Design of a Switching System for 10GBASE-T Ethernet

by

Haojun Luo

A Thesis Presented in Partial Fulfillment
of the Requirements for the Degree
Master of Science

Approved November 2010 by the
Graduate Supervisory Committee:

Joseph Hui, Chair
Junshan Zhang
Martin Reisslein

ARIZONA STATE UNIVERSITY

December 2010

ABSTRACT

Ethernet switching is provided to interconnect multiple Ethernets for the exchange of Ethernet data frames. Most Ethernet switches require data buffering and Ethernet signal regeneration at the switch which incur the problems of substantial signal processing, power consumption, and transmission delay. To solve these problems, a cross bar architecture switching system for 10GBASE-T Ethernet is proposed in this thesis. The switching system is considered as the first step of implementing a multi-stage interconnection network to achieve Terabit or Petabit switching.

By routing customized headers in capsulated Ethernet frames in an out-of-band control method, the proposed switching system would transmit the original Ethernet frames with little processing, thereby makes the system appear as a simple physical medium for different hosts. The switching system is designed and performed by using CMOS technology.

ACKNOWLEDGMENTS

I would like to express my sincere gratitude and respect to my advisor and mentor Dr. Hui, for his continued support and invaluable guidance, during the course of the degree, without which this work would not have been possible. His trust in my capabilities and patience in helping me understand new concepts have been the motivating force for this work. I am grateful to Dr. Junshan Zhang and Dr. Martin Reisslein, for agreeing to be on my Masters Committee and for their time and efforts in reviewing this work. I am indebted to my family for their unconditional love and support. Finally, I would like to thank all my friends who were also important in the successful realization of this thesis.

TABLE OF CONTENTS

Page

CHAPTER Page

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Ethernet and Switching Network Architectures

Ethernet, as the dominant networking technology nowadays, has moved from 10 Megabit per second (Mbps) to 10 Gigabit per second (Gbps) just in 3 decades, with 40 Gbps and 100 Gbps being developed. The rapid growth of bandwidth-intensive applications such as high-performance computing, enterprise computing, virtualization, and video on demand has caused an explosion of bandwidth use. Internet Small Computer System Interface (iSCSI), Fibre Channel over Ethernet (FCoE) and Network Access Server (NAS) storage are beginning to adopt Ethernet speed increasing to 10 Gbps. Therefore, Ethernet switch, as the interconnect network for Ethernets, requires higher link capacity and better scalability.

An ideal Ethernet switch would be one that routes frames from any network segment to any other segment with small time delay, irrespective of the number of segments in the path. This kind of switch should be scalable cost-effectively from a very small number of ports to thousands of ports without losing any of the performance.

The scalability of an Ethernet switch depends on its underlying architecture. Of the three architectures currently employed by the industry

for increasing capacity – shared buses, shared memory, and crossbars – crossbar switches provide the best scalability.

Typical shared-bus switches as shown in Figure 1.1 have maximum capacity ranging from 640 Mbps to 2.5 Gbps [1]. When bus contention overhead is taken into account, their aggregate capacity decreases.

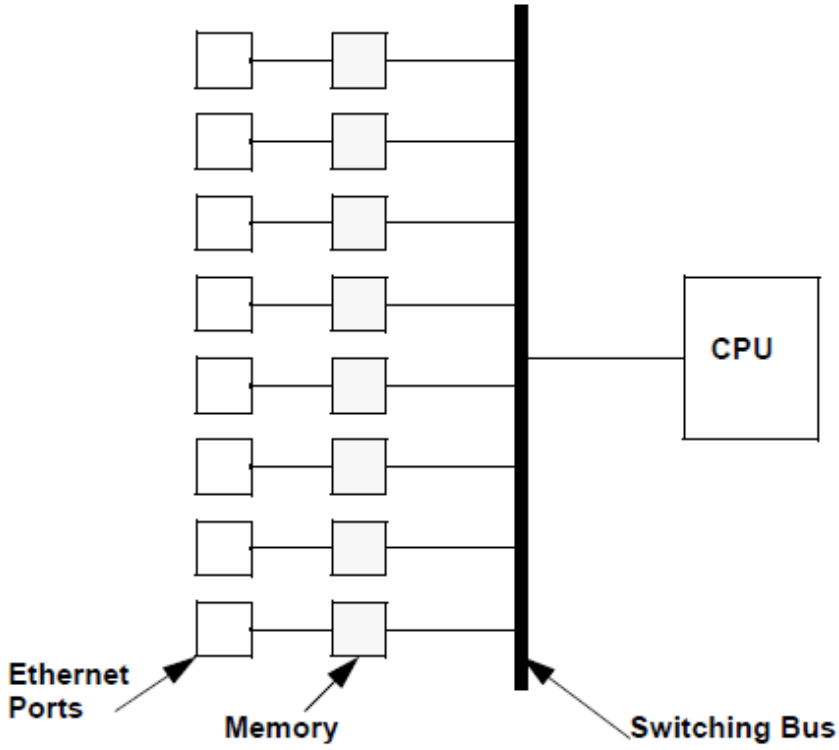

Figure 1.1 Typical shared bus switch

Typical shared memory switching system, as shown in Figure 1.2, has aggregate capacity ranging from 4 Gbps to around 10Gbps [1]. The

major disadvantage of the shared memory architecture is the complexity and cost that are added as attempting to increase bandwidth.

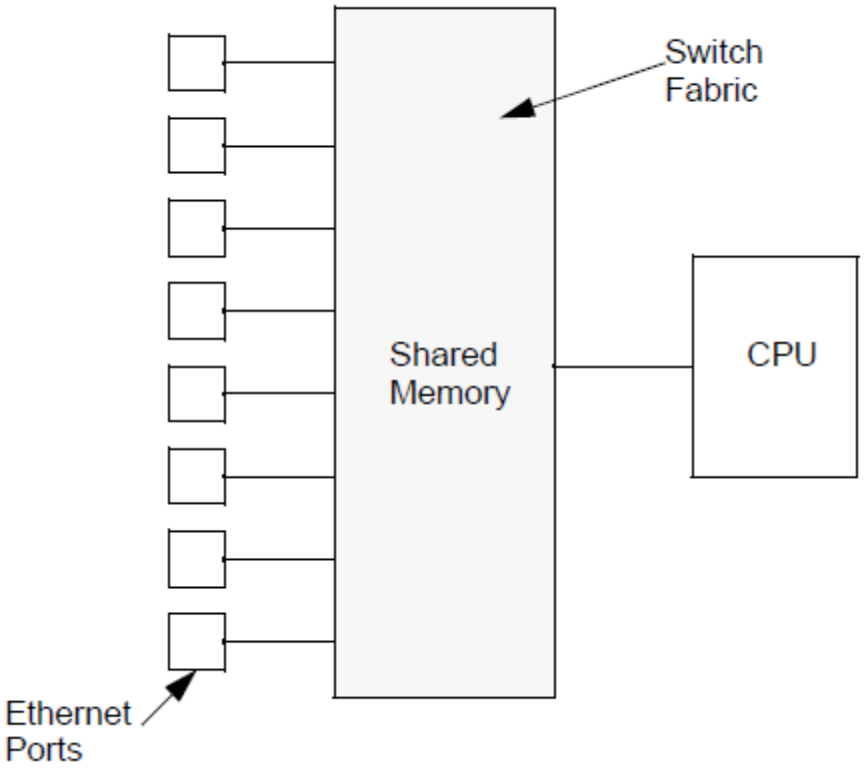

Figure 1.2 Typical shared memory switch

Crossbar switches can be designed to easily scale to higher capacity because data is passed through the switch in parallel through dedicated switching elements, as shown in Figure 1.3, rather than a shared resource such as a shared bus or shared memory. Each connection from a crossbar fabric input segment to a crossbar fabric output segment represents a dedicated path through the switch. Therefore,

adding more ports to the crossbar provides a corresponding linear increase in the crossbar switch's bandwidth.

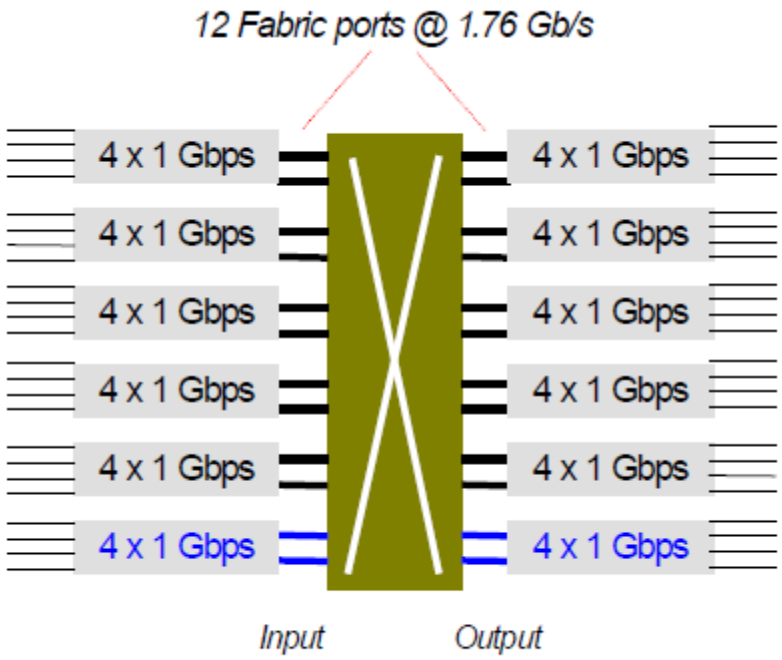

Figure 1.3 Typical crossbar switch

However, crossbar switches suffer a condition called contention occurs when multiple inputs beginning connection to same output. Due to this potential for contention at the output connections, worst case output link utilization can fall as low as 60 percent of raw capacity [2][12]. Unless enhancements are made to the basic crossbar design, crossbar contention can make the shared memory design relatively more efficient

than the crossbar switch. For this reason, an enhanced multi-stage crossbar switching network is introduced in the next section.

1.2 CSMA/TS Multi-stage Switching Network

Carrier Sense Multiple Access (CSMA) is a protocol [3] in which a communication node senses the presence or absence of a carrier on a transmission medium before attempts to transmit. There are two widely known techniques that extend the CSMA protocol [4].

First, we have Carrier Sense Multiple Access with Collision Detection (CSMA/CD). In this modification a node follows the basic CSMA protocol - verifying the medium is not busy before transmitting, but with the enhancement that after a node begins transmitting, it must monitor the medium to detect a collision. A transmitting node may detect a collision using any techniques, typically involves comparing transmitted data to received data. If they are different, a collision is assumed and a jamming signal is transmitted immediately. The jamming signal causes any conflicting nodes to stop their transmission and back off by random amounts of time before reattempting transmission.

Second, we have Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA). This modification requires a handshake before transmission. The transmitting node issues a Request-to-Send (RTS) and must receive back a Clear-to-Send (CTS) from the intended receiving

node before beginning a transmission. The handshake serves as a notification to all other nodes to refrain from transmitting, thus avoiding collisions. This extension to the CSMA protocol is primarily intended for use with wireless transmission media where CSMA/CD would not work because it is a node cannot listen while transmitting, enhance failing to detect a collision.

Both CSMA/CD and CSMA/CA utilize timing (temporal switching) of the transmission to avoid collision. A third technique, Carrier Sensing Multiple Access in Time-Space (CSMA/TS), was introduced [3] that utilizes spatial switching to avoid collision. Through CSMA/TS, carrier sensing is performed Step-By-Step (SBS) for multiple links in a path and possibly over alternative paths. Hence carrier sensing is performed not only in time, but also in the space of multiple links and multiple paths.

In CSMA/TS, the end-to-end connection established between two nodes involves a number of Cross Bar Switches (CBS) that are interconnected in stages to form a Multi-stage Interconnection Network (MIN). One example of multi-stage switching network is shown below [3].

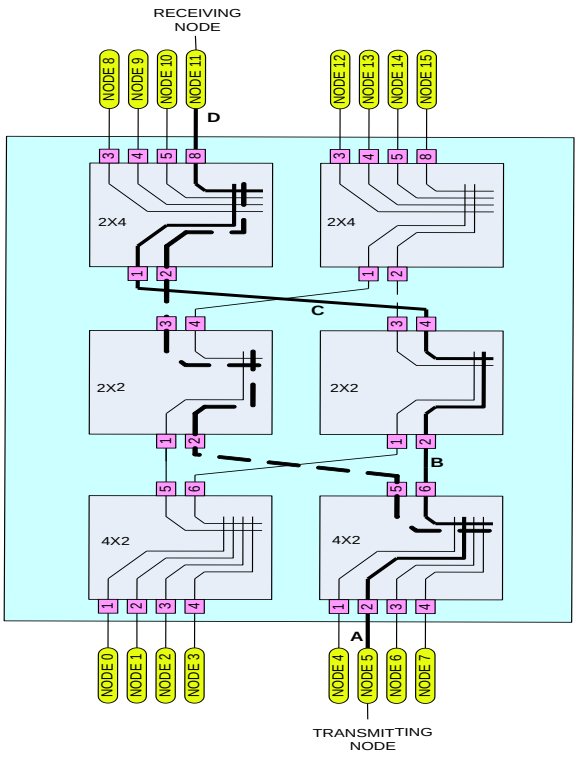

Figure 1.4 An example of multi-stage switching network

In Figure 1.4, a three-stage CBS is presented. Ethernet packages are routed through the MIN in two possible paths (either the thick black route or the dashed route). For example, input node 5 may choose the black route labeled “ABCD”. It then establishes the path step-by-step by carrier sensing the internal nodes. If a carrier is detected, it immediately tries the alternative dashed path. In that case, CSMA randomizes transmission over space. If a carrier is detected in the final node, no alternative path would work either, and CSMA then may have transmitted over time by carrier sensing the medium after a random delay.

In a multi-stage switching network for CSMA/TS, there are two modes for controlling carrier sensing. First a persistent CSMA/TS-P for which links sensed idle are seized and not released even if downstream links are found busy. For example, in a two-stage switching network in Figure 1.6, node 5 may seize the idle output 6. When it proceeds to sense output 8 in the next stage switch, it may sense a carrier as another input node may be sending data at the same time to node 11. The seized output 6 in the first stage is not released, despite node 11 being sensed busy. Node 5 completes the connection when node 11 is sensed idle later.

Alternatively, a fast release CSMA/TS-FR may release links in an earlier stage should later stage links be found busy. In our example, output 6 in the first stage is released after node 11 is sensed to be busy. Another attempt to connect to output 11 is attempted after a random delay.

Previous work [5] has been done to analyze the throughput comparison between multi-stage switching networks as shown in Table 1.1.

Table 1.1 Throughput comparisons of multi-stage switching networks

THROUGHPUT RESULTS

% Throughput				
# of stages	packet length distribution	method	TS-P	TS-FR
1	Fixed	analysis	58.8%	-
	Exponential	analysis	50%	-
2	Exponential	analysis	33.3%	-
		simulation	35%	40%
3	Exponential	simulation	40%	45%

From the results, we found that the throughput of a three-stage network is greater than that of a two-stage network by 5% because of availability of multiple parallel paths through the switch fabric. Also, using Fast Release (TS-FR) mechanism increases throughput by 5% compared to the Persistent (TS-P) carrier sensing mechanism. This is due to releasing of intermediate links if the connection could not be completed, thus allowing other connections to proceed.

In this thesis, a one-stage cross bar switch is proposed and designed as the first step implementation of CSMA/TS Multi-stage switching network.

Routing control of multi-stage switching network is difficult to perform in parallel and fast. Switches built using MIN are typically controlled by the centralized Store Program Control (SPC) method. Route

establishment is often performed sequentially and therefore does not scale well with increasing traffic and switch size. In this thesis, instead of using SPC for routing, a separate control plane with parallelism is proposed and designed. Combined with the one-stage cross bar switch, a switching system for Ethernet is designed.

1.3 Thesis Outline

The remainder of the thesis is organized as follows. Chapter 2 gives a brief review of proposed one-stage switching system. Chapter 3 presents the design of physical plane of the switch. Chapter 4 describes the components design of Tri-state switch and control plane in the system. Simulation results are given in Chapter 5. Finally, Chapter 6 concludes the entire thesis.

CHAPTER 2

AN OVERVIEW OF PROPOSED SWITCHING SYSTEM

2.1 An Overview of Proposed Switching System

In the proposed switching system, electrical Ethernet signals are transmitted as is through a one-stage cross bar switch network called the Physical Plane. Without regeneration and buffering of Ethernet signal and data, much of the processing and delay is removed. This requires a shift of accessing, routing and other functions to the Ethernet interface, which employs carrier sensing in both space and time to route data through the Physical Plane. The Ethernet interface translates the destination Ethernet address into routing address. The routing address is sent prior to sending Ethernet data.

At the Ethernet switch, this route information, together with other control signals for a Request-to-Send (RTS) signal, are diverted to a Control Plane that performs three functions. First, route information is decoded through an address de-multiplexer. Second, contention for a destination Ethernet is resolved through a contention resolution multiplexer. Third, the Ethernet with a successful RTS is acknowledged with a Clear-to-Send (CTS) signal, while the Control Plane completes the connection of the two Ethernets in the Physical Plane. After CTS signal is received, the Ethernet interface begins transmitting Ethernet data frames.

Larger switches, such as a two-stage SS switching network and a three-stage SSS switching network could be built based on the architecture of the one-stage Switch Plane (SP). The step-by-step and out-of-band control architecture is extended for the MIN. Choosing alternative paths to avoid blocking is controlled by the Control Plane to improve throughput of the switching network.

2.2 The Components of the Switching System

The architecture of proposed Ethernet switch is shown in Figure 2.1. The Host Bus Adapter (HBA) that generates the 10GBASE-T Ethernet signal is connected to the Switch Plane (SP) via Category 6a (CAT6a) cables [6]. Each CAT6a cable has 4 pairs of twisted wires for which each pair carries 16 Pulse Amplitude Modulation (PAM16) symbols for 10GBASE-T Ethernet [13]. As an illustration in Figure 2.1, the 4 pairs carrying differentially encoded signals called DA+/-, DB+/-, DC+/-, and DD+/- are switched through 4 separate Physical Plane (PP). HBA, Control Plane and Physical Plane are connected through an interface named Tri-stage Switch which provides different connections among these components. For simplicity of illustration, a Switch Plane with 4 HBAs is shown in this thesis. The SP in practice can be of size 32x32 or larger, interconnecting 32 or more HBAs. The HBAs are numbered sequentially in binary representation, i.e. the inputs are numbered in the figure to down

as 00, 01, 10 and 11, respectively. This number is referred to as an Input Segment Address ISA.

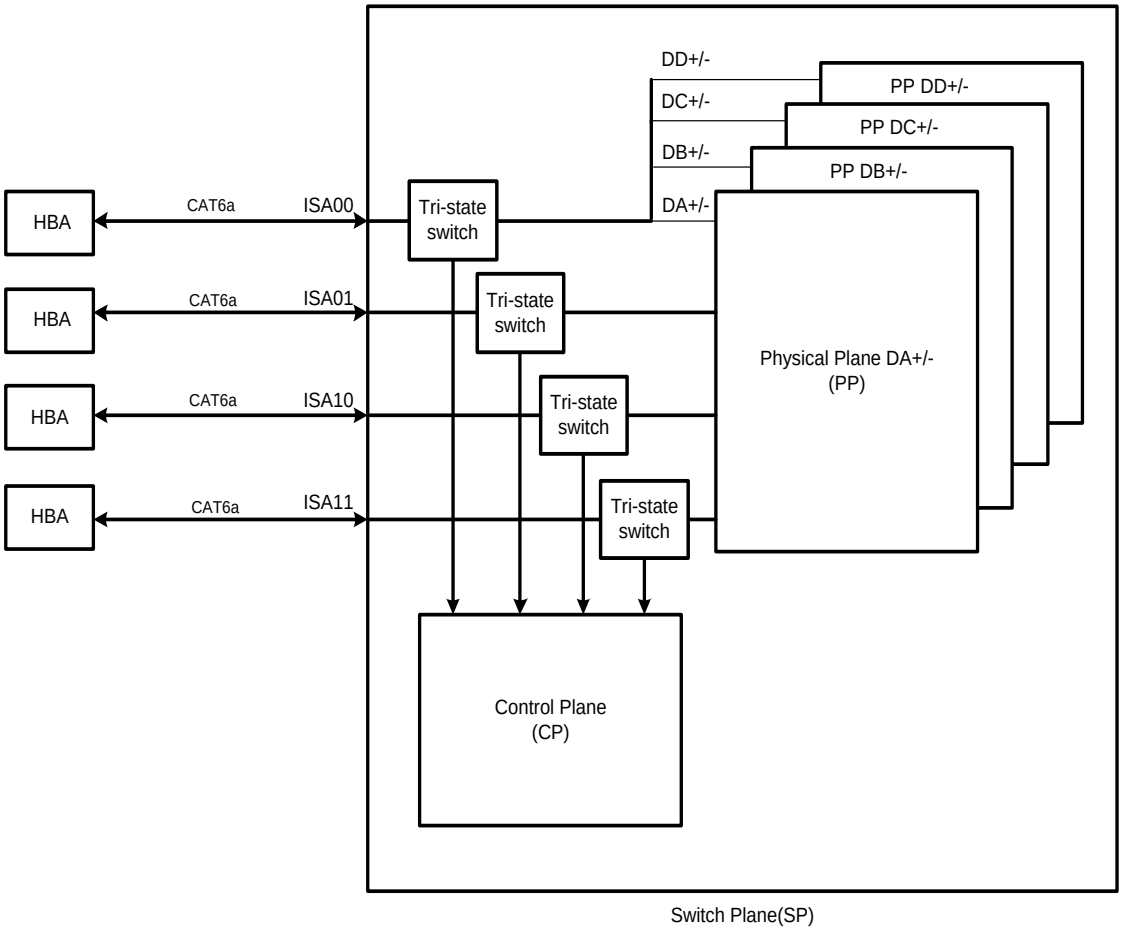

Figure 2.1 The architecture of the Ethernet switch

2.3 Host Bus Adapter Architecture

The HBA provides routing and carrier sensing information for switch control as shown in Figure 2.2. A cache table provides the translation of frequently used 64-byte Ethernet Media Access Control addresses into the

Destination Segment Address (DSA), which represents the physical location of the destination Ethernet segment to be connected to. The HBA is also responding for sending primitives to CP and PP in the switch, receiving primitives from CP before transmitting Ethernet packages as well as performing other network functions such as address resolution. An example function is address resolution such as finding the DSA of the destination Ethernet if that translation is not found in the cache. Another example function is the handshake involved when an HBA is initially connected to a switch, exchanging information such as Ethernet addresses of the HBA, segment address on which the HBA is established and the parameters setting for 10GBASE-T Physical Layer (PHY) transceivers to achieve reliable data transmission under 10Gbps.

An HBA takes data input from the host through the PCIe port. The data is then processed by 10GBASE-T Media Access Controlled before being transmitted by the 10GbE PHY through a connector named RJ45. From the connector the signal is transmitted on Ethernet transmission medium such as a CAT6a cable.

Control primitives are also transmitted and received using the same PCIe port and CAT6a cable. They are processed by the Control Plane MAC and the CP PHY. Hybrid circuit, MUX, and DEMUX are used to

implement Time-Division Multiplexing (TDM) between Ethernet packages and control signals.

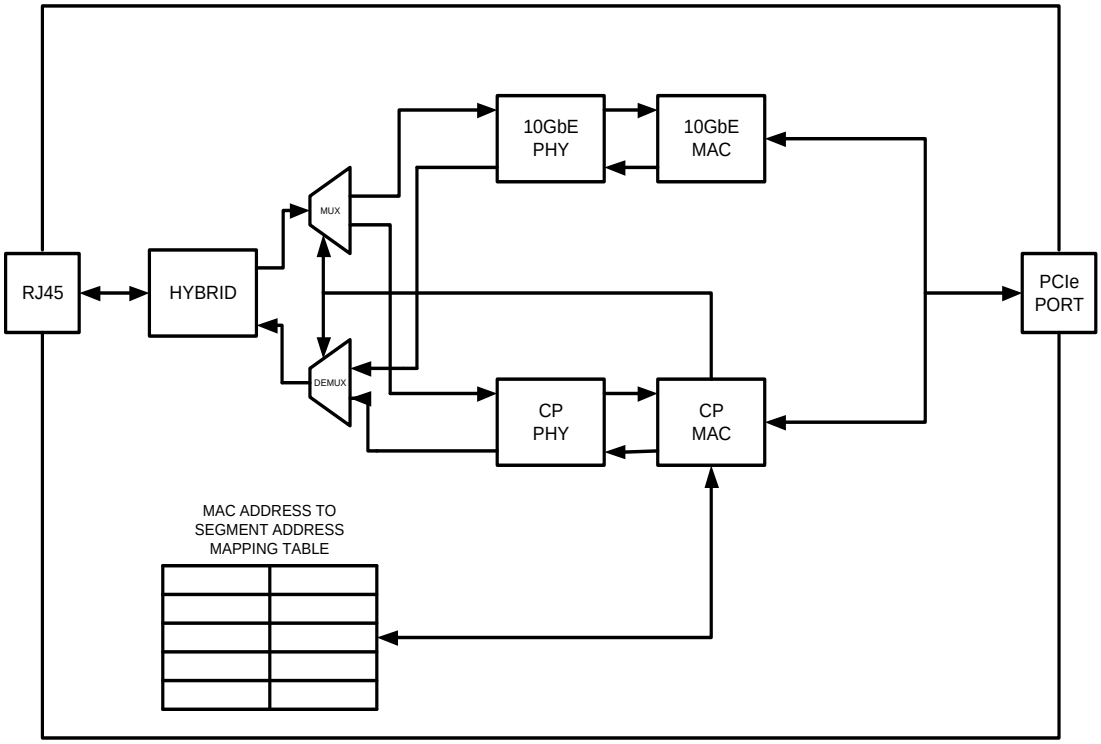

Figure 2.2 The architecture of Host Bus Adaptor

2.4 Control Architecture of the Switching System

The control algorithm of the switching system is based on a handshaking protocol for which one HBA needs to send Request-to-Send (RTS) signal and upon receiving Clear-to-Send (CTS) acknowledgement signal from the switch, it starts to transmit Ethernet frames. Otherwise, the HBA will send the next RTS in queue, or resend this RTS after a random

delay. This section will describe the control architecture of the switching system.

The architecture of a Tri-state switch is shown in Figure 2.3. It severs the function as providing possible connectivity between three components: HBA, CP and PP. The HBA to CP connection is a bidirectional buffer for which primitives are driven to send and receive between them, the HBA to PP connection is an analog switch for transmitting Ethernet signals which is identical to the cross bar switch in PP, and the CP to PP connection is an one direction buffer to drive the primitives sending from CP to PP. These switches in the Tri-state switch are controlled by the control lines shown in the figure.

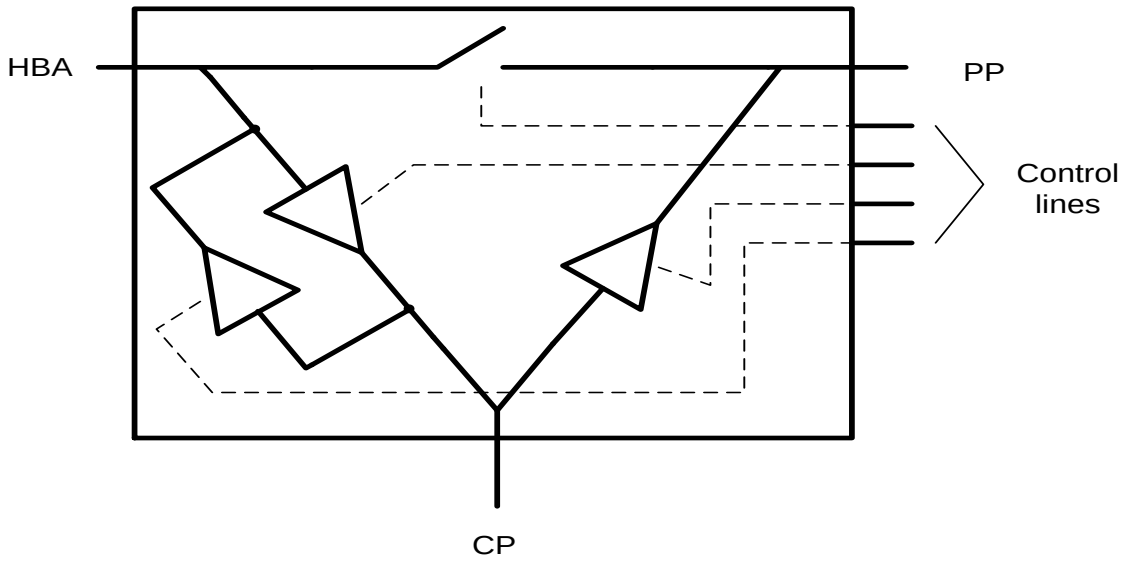

Figure 2.3 The architecture of a Tri-state switch

The architecture of Control Plane is shown in Figure 2.4. The first purpose of the control plane is to detect and decode the RTS primitive from a source HBA. The second purpose of the control plane is routing this RTS signal to the proper place for contention resolution. The decoded RTS contains the address of the destination Ethernet segment DSA. It is self-routed using its DSA through a de-multiplexer. The de-multiplexer is structured as a tree for which consecutive bits of DSA are used to set a route position at each level of the tree. The RTS subsequently opens a pathway in the de-multiplexer for contention resolution for the requested destination Ethernet segment.

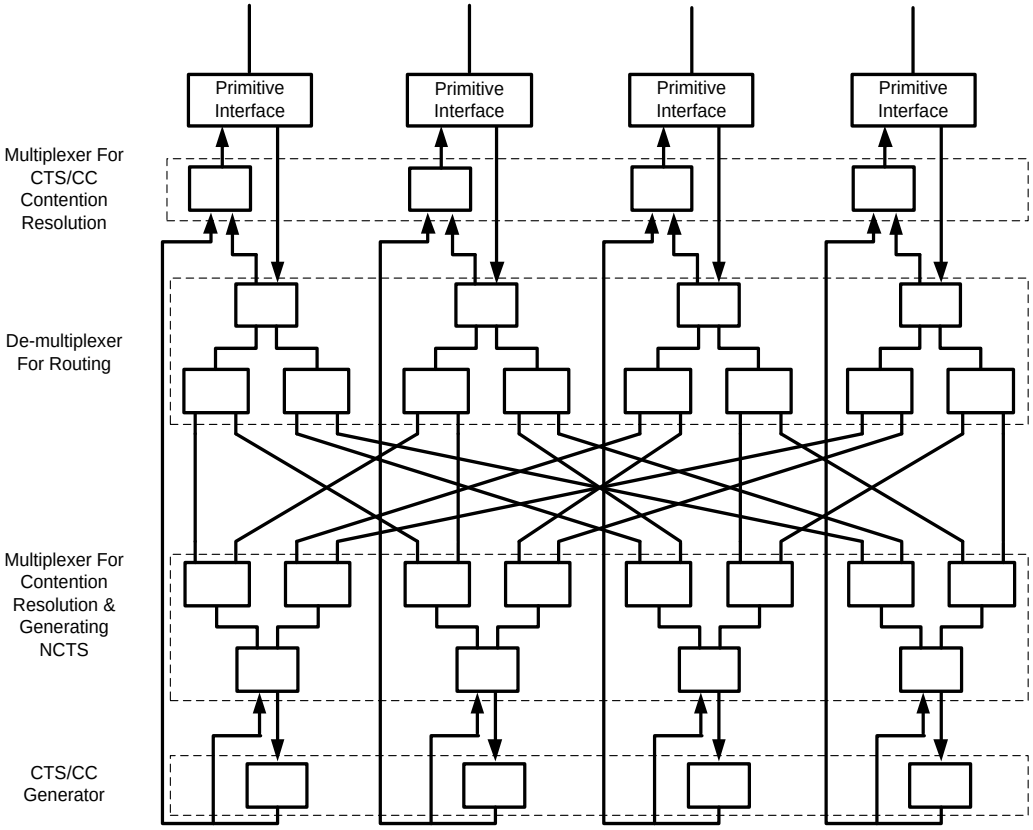

Figure 2.4 The architecture of Control Plane

The third purpose of the control plane is to gather the RTS for the same destination Ethernet segment for contention resolution. After the RTS passed through the de-multiplexed by using up its DSA for routing, RTSs from different inputs for the same DSA are grouped together as input to a multiplexer for that DSA.

The tree is made up of Contention Resolvers (CR) that resolves contention of RTS on a first come first serve (FCFS) basis. Each CR has N inputs (N=2 in the figure) for which an arrival RTS could connect to its

single output if that output is not already connected to any input; otherwise the arriving RTS is pre-empted by an earlier arriving RTS, thereby loses the contention on a FCFS basis. The state of a CR is its state of connection, i.e. if the CR is not connected or if connected, which input the CR is connected to.

The surviving RTS of a CR then proceeds to contend at the next level of the multiplexer. The RTS wins the final contention resolution process of the DSA at the bottom of the multiplexer which connects to the DSA requested.

One example illustrates address decoding and contention resolution processes and trees with the example of 2 RTSs from input ISA 01 and 11, both contending for the DSA 10 in Figure 2.5. The RTS from input 11 is made earlier than the RTS from input 01, winning the contention resolution as the RTS from input 01 loses the contention at the bottom of the multiplexer.

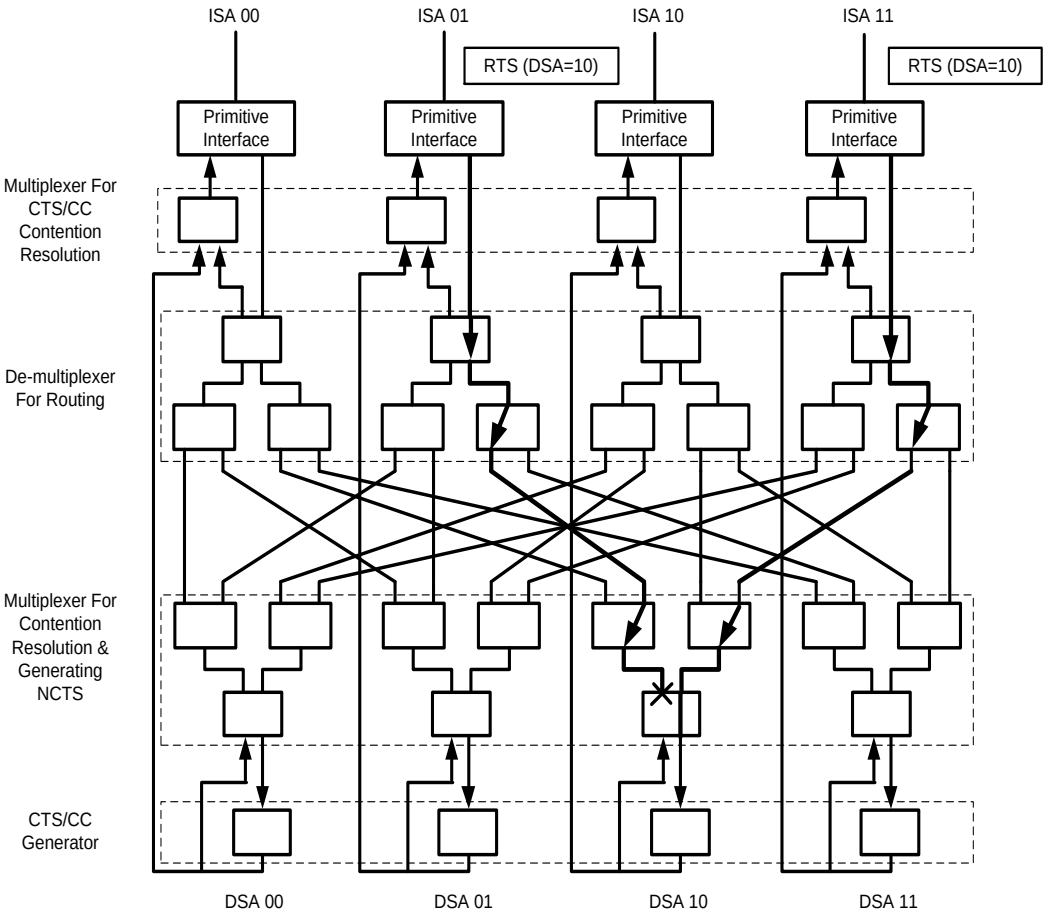

Figure 2.5 The routing example in Control Plane

The RTS that survives overall wins the contention resolution process. That RTS from the requesting ISA for access to the DSA is now cleared to send, and a CTS signal will be generated and sent from the DSA back to the ISA. The CTS signal propagates backward using the path opened from the winning ISA that has now connected to the DSA. Another kind of contention may occur when CTS or CC primitive arrives at the

same segment address. The multiplexers are introduced to solve that issue on a FCFS basis.

An RTS that fails to reach the bottom of the multiplexer loses the contention resolution. The sending ISA fails to receive the CTS signal. A Not-Clear-To-Send (NCTS) signal will propagate backward to clear the opened path and reset the multiplexer and de-multiplexer to their initiating state. The initiating HBA may buffer the data for later attempts to send, or send the next RTS in queue instead. The NCTS Generator is contained in the multiplexer, and not shown in the Figure 2.6.

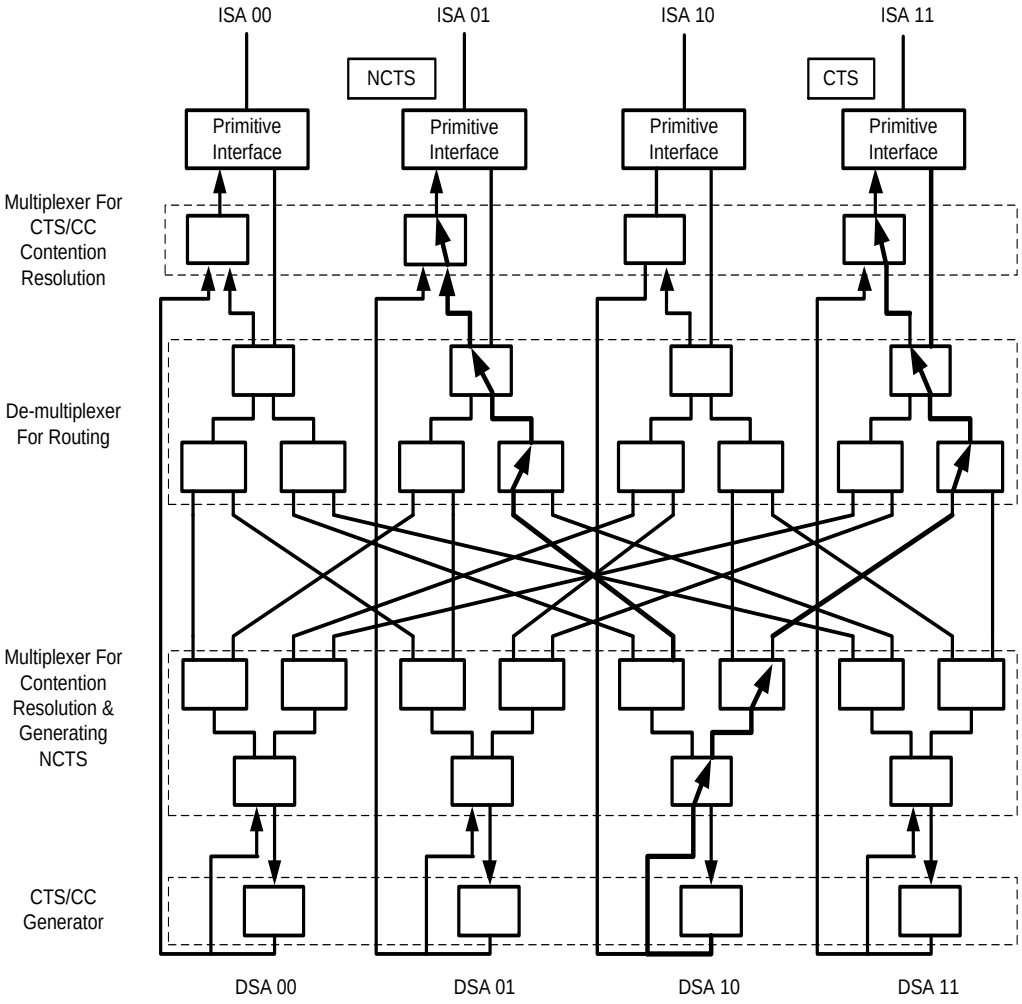

Figure 2.6 Clear-To-Send signaling in Control Plane

The forth purpose of the control plane is then to connect the input Ethernet with a successful RTS to the requested destination Ethernet. The cross point is referenced by the address pair (ISA, DSA). The implementation shows in Figure 2.7 allowing for the CP to send two CC primitives arriving at the cross point (ISA, DSA) simultaneously. The cross

bar switch at (ISA, DSA) receives simultaneously the CC primitive for the crosspoint to close.

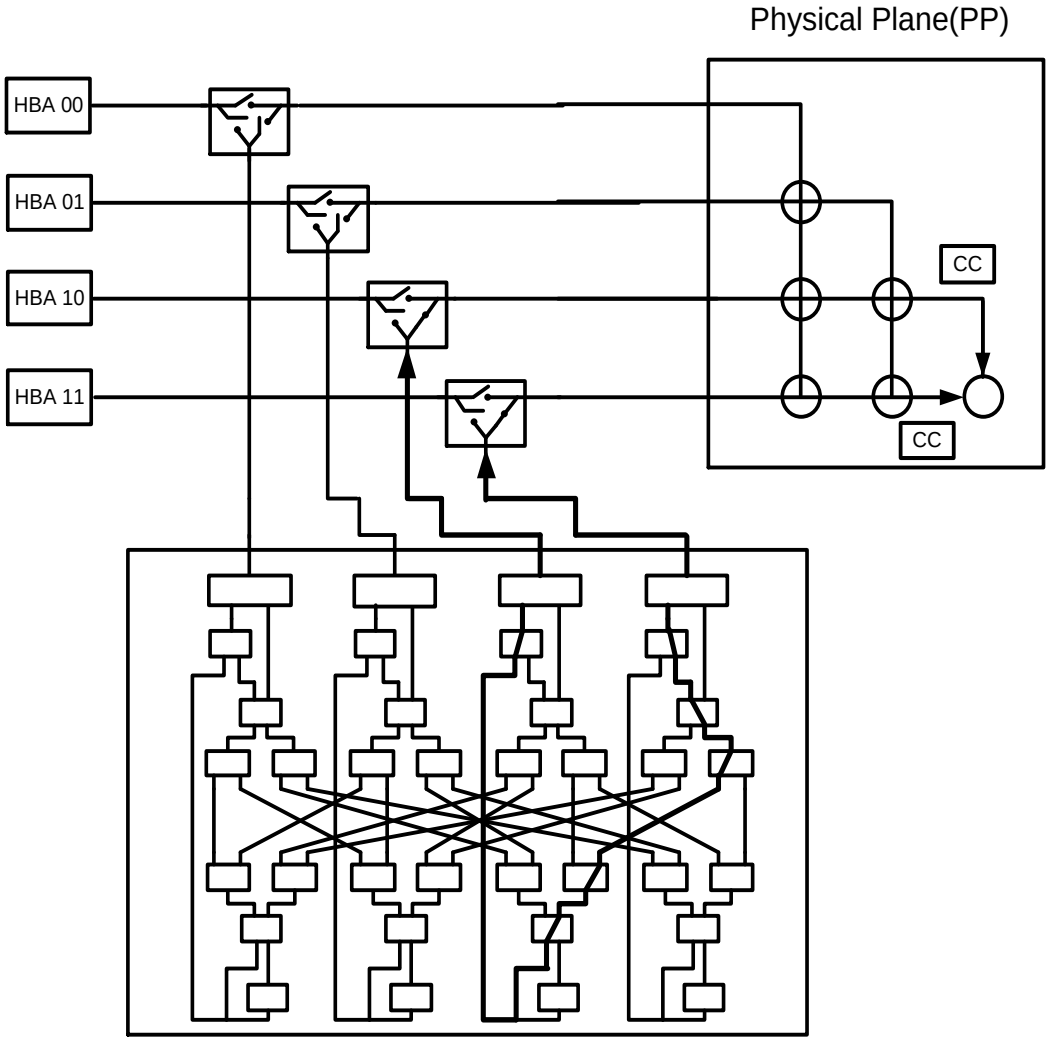

Figure 2.7 The process of completion of connection request by closing cross point in Cross Bar Switch

A possible problem arises when we attempt to close two crosspoints, say (ISA, DSA) and (ISA', DSA') simultaneously. The CC commands travels on four Ethernet segments ISA, ISA', DSA, and DSA'.

Subsequently the crosspoints (ISA, DSA') and (ISA', DSA) may also be inadvertently closed. While this is unlikely because the two CCs may not be simultaneous, we can avoid this by sending distinguishable connection commands CC and CC'.

Simple logic has to be implemented at each cross bar switch to detect simultaneously arriving CCs that are identical. Upon detection, the crosspoint is closed. Likewise, simple logic has to be implemented at each crosspoint to Disconnect Commands (DC) that are identical. Upon detection, the crosspoint is opened and thereby disconnects. The design of the cross bar switch is describe in Chapter 4. Once the CTS is received by the IS, data transmission through the PP begins. Upon completion of data transmission, a DC is sent from the IS through the PP to open the crosspoint.

2.5 Timing diagram of the switching system

The timing exchange diagram of the switching system is shown in Figure 2.8.

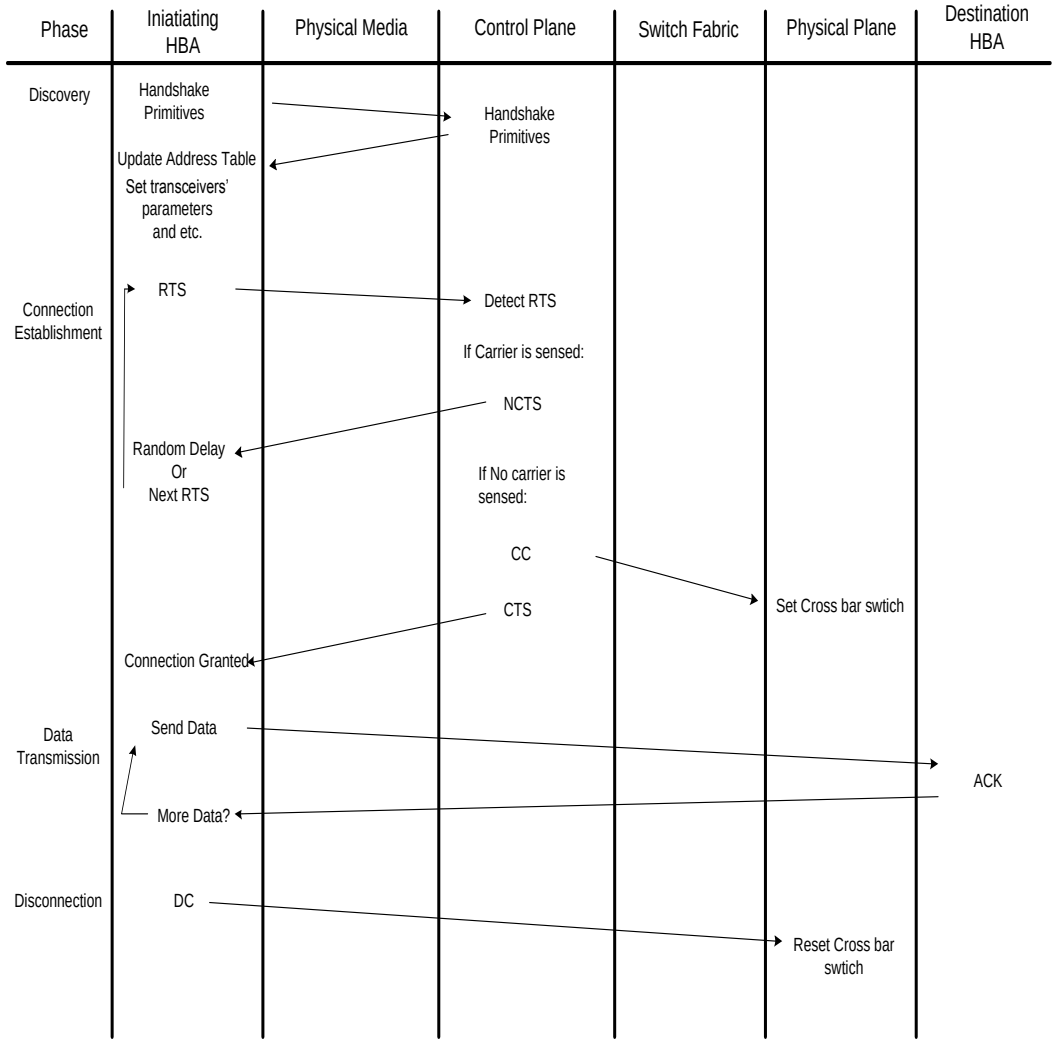

Figure 2.8 The timing exchange diagram of the switching system

During the Discovery Phase, all necessary information for the handshake primitives between HBAs and the switch are exchanged, such as the MAC address to segment address translation table, the transceiver setting parameters in 10G PHY chip located in each HBA, etc.

During the Connection Establishment Phase, the HBA will generate the RTS primitive. At the switch side upon receiving the primitive from the HBA, CP will detect the RTS header, and process the operations described above. If the CP find the destination node is idle for connection, CTS and CC primitives will be sent to the HBA and PP respectively. If the destination node is found busy, NCTS primitive will be sent to the HBA. The HBA will transmit the Ethernet Frame or attempt to send the RTS again, depending on receiving CTS or NCTS code words. Tri-state switches therefore play the role of providing connectivity among HBA, CP and PP.

During Data Transmission Phase, the HBA operates normal transmission of 10GBASE-T frames.

During Disconnection Phase, the HBA will generate the Disconnect Command primitive sent to PP, and the PP will decode the DC primitive and release the connection.

CHAPTER 3

DESIGN OF PHYSICAL PLANE

3.1 An Overview of Physical Plane

Figure 3.1 shows a block diagram of a 4x4 cross bar switch. A cross point allows 10GBASE-T Ethernet signals to be transmitted across 2 segments in full duplex mode. A cross point is closed when a transmission gate is in a conductive state, otherwise, in a non-conductive state, the cross point is considered open. Simple control circuit for detecting Connect Command and Disconnect Command determines if the cross point should be open or close.

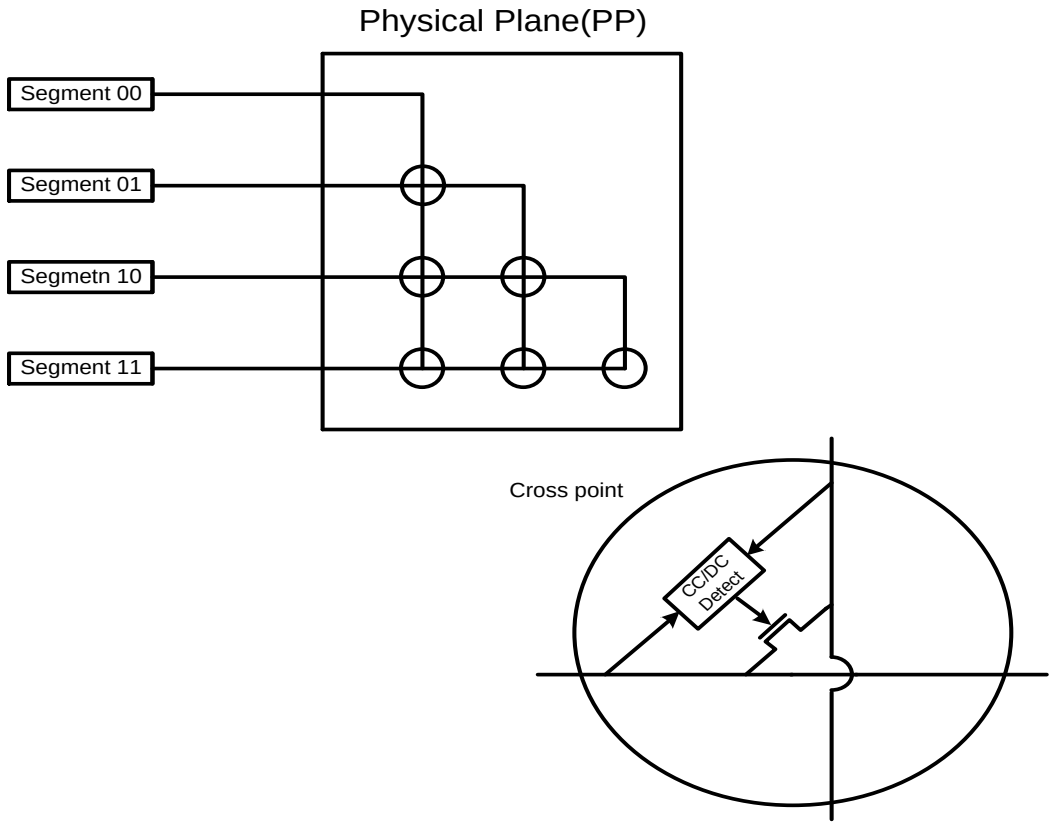

Figure 3.1 A 4x4 cross bar switch

3.2 Design of the Cross Bar Switch

The proposed differential cross bar switch architecture is shown in Figure 3.2. Two analog switches are constructed in a π-switch configuration, which results in minimum reflection loss in the conductive condition while maintaining good frequency response in the non-conductive condition. The sensor block inside the control logic circuit is designed to detect CC or DC primitives, and subsequently sends control information to the ON/OFF Logic block. The ON/OFF Logic block then

processes the coming control signals, and controls the state of these analog switches.

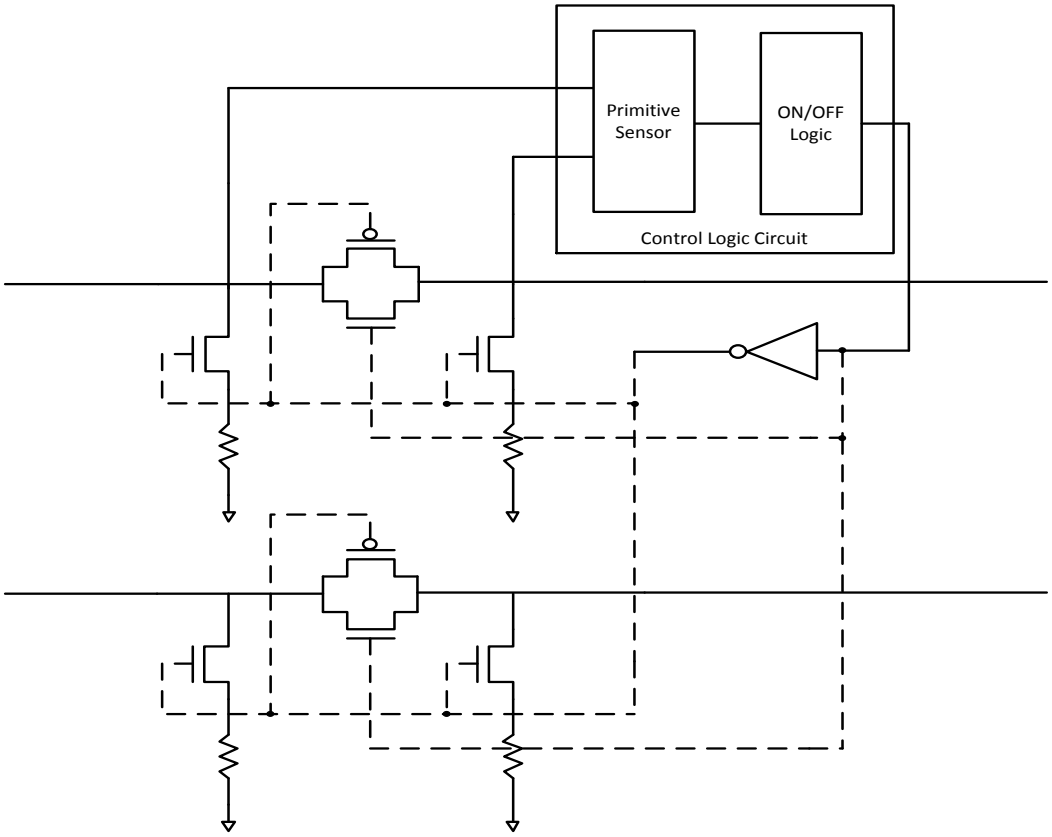

Figure 3.2 The architecture of a cross bar switch

A. Design of the analog switch

The most basic form of an analog switch is a voltage-controlled device. In the "on" state, resistance can be less than 1 ohm, while in the "off" state, resistance increases to several hundreds of mega ohms with pico amperes leakage currents. When it goes to the metal-oxide-semiconductor field-effect transistor (MOSFET) technology, a voltage is

applied to the oxide-insulated gate electrode which can provide a conducting channel between the two other contacts called source and drain. The source and drain of the MOSFET are interchangeable. The channel can be of n-type or p-type and referred accordingly as a NMOS or a PMOS [9].

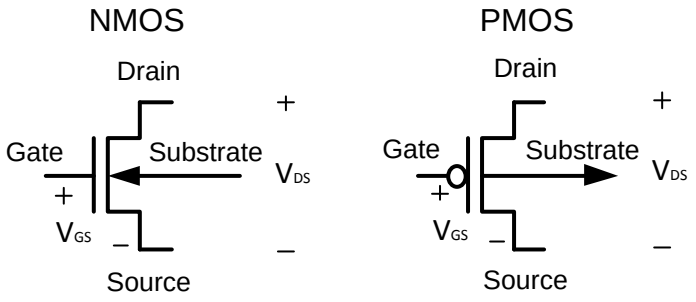

Figure 3.3 Voltage designations of MOSFETs

The operation of a MOSFET can be separated in three modes, depending on the voltages at the terminals. The three operation modes are cutoff, linear, and saturation. In cutoff mode, V_{GS} is less than V_{th}, where V_{th} is the threshold voltage of the device and the transistor is turned off with little leakage current between the drain and the source. When V_{GS} is greater than V_{th} and V_{DS} is less than V_{GS}- V_{th}, the transistor is turned into linear mode and a channel is created which allows current to flow between the drain and the source. The device operates as a resistor, controlled by the source and drain voltages. When V_{GS} is larger than V_{th} and V_{DS} is

larger than V_{GS}-V_{th}, the transistor goes into saturation mode. Since the drain voltage is much higher than the gate voltage, the electrons spread out and conduction is not through a narrow channel but through a broader, two- or three-dimensional current distribution extending away from the interface and deeper in the substrate. As the lack of channel region near the drain, the drain current is now weakly dependent upon drain voltage and controlled primarily by the gate-source voltage. Hence, from an analog switch design aspect, MOSFETs need to operate either in cutoff mode or linear mode.

There are three types of MOSFET analog switch are shown in Figure 3.3. In the following sections, we will call them as NMOS, PMOS and CMOS (Complementary Metal Oxide Semiconductor) switch accordingly. With different voltages attached to the substrate and gate of the device, different voltage swings of input signals would apply to the switches. For example, with a V_{DD} voltage attached to the gate and a V_{ss} voltage attached to the substrate of the NMOS switch, the input and output voltage swing would be from V_{ss} to V_{DD}- V_{th}. In that regard, signals with voltage level around V_{DD} would be distorted when it goes through the switch. Similarly, PMOS switch is good for transmitting signals with voltage level around V_{DD} which gives a V_{th} rise when the V_{SS} level signals goes through the switch. A CMOS switch, combining the advantages of

both NMOS and PMOS by connecting the PMOS and NMOS in parallel allows signals with full swing from V_{SS} to V_{DD} to go through the switch without any offset.

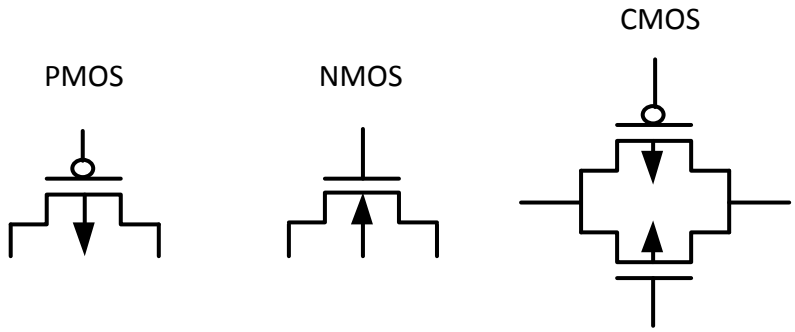

Figure 3.4 NMOS, PMOS and CMOS switches

Furthermore, the on-resistance of NMOS and PMOS changes nonlinearly with signal voltage. This nonlinear resistance can cause DC accuracy as well as AC distortion. Using a CMOS switch solves this problem. On-resistance is minimized and linearity is also improved. Figure 3.4 shows the on-resistance characteristic of the PMOS, NMOS and an improved CMOS switch with flattened resistance [8]. For these reasons mentioned above, CMOS switch is chosen as the analog switch for our application.

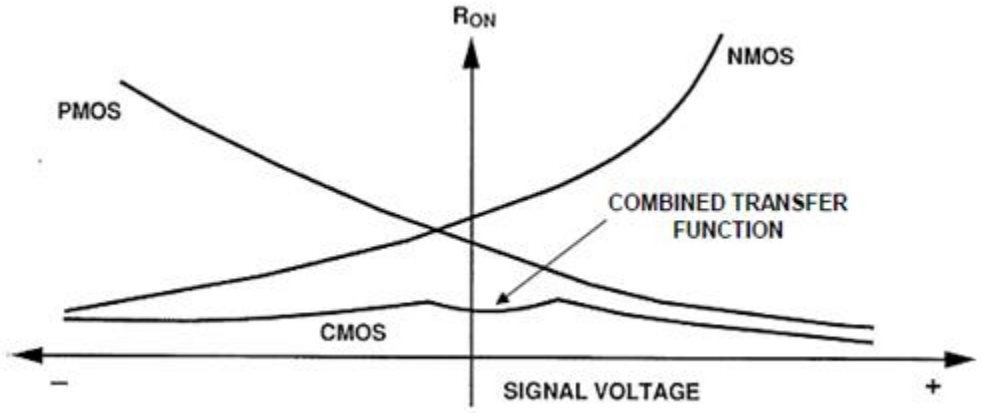

Figure 3.5 On-resistances of NMOS, PMOS and CMOS switches

In our proposed switching system, two more NMOS switches are added to form a π-switch configuration to reduce the reflection loss introduced to the system.

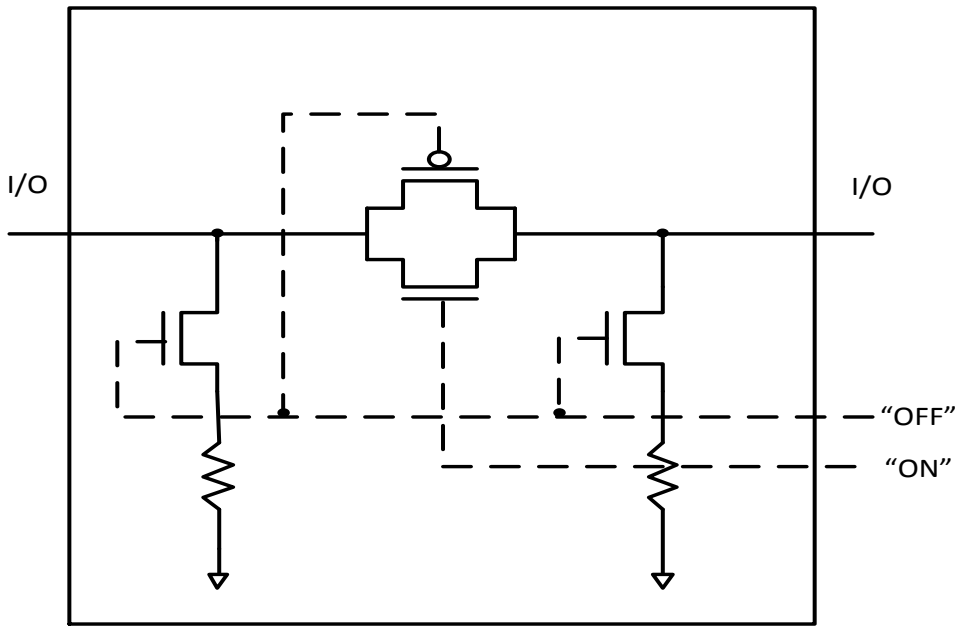

Figure 3.6 A π-switch configuration

When the “ON” pin is set to high, the CMOS switch is in conductive state, while the two NMOS switches are in non-conductive state. When the “OFF” pin is set to high, the two NMOS switches are in conductive state, and CMOS switch is in non-conductive state. This makes the input impedance set to a pre-set value close to the HBA impedance to minimize the reflection loss at both ends. In our case, the pre-set resistance value for both branches is 30 ohms approximately.

B. Design of the control logic circuit

In the proposed cross bar switch, the control logic circuit is directly attached to the analog switch. By sensing the CC or DC primitives, the analog switch is set to close or open. The main design consideration of the control logic circuit is to minimize the impairment caused by the sensing circuit on the analog switch. We also maximize the accuracy of the sensing and control logic circuits.

As shown in Figure 3.6, there are three amplifiers sensing the voltage changes at two nodes when CC or DD primitives are sent and by amplifying the voltage, the ON/OFF logic can decode the primitives and set the analog switch respectively.

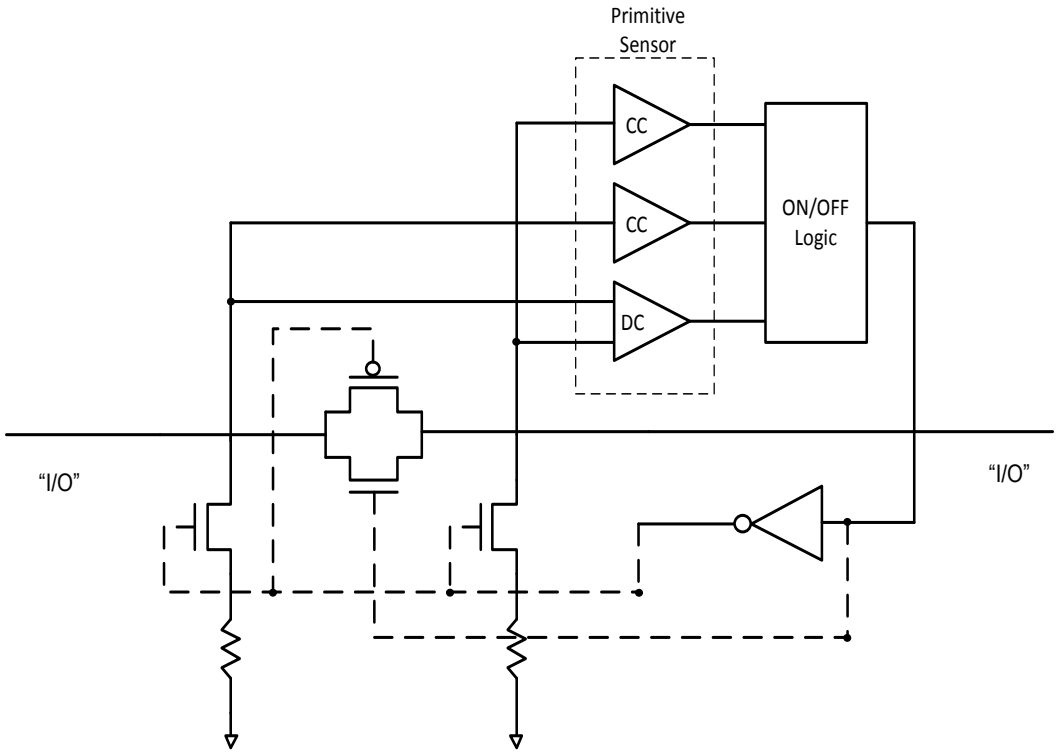

Figure 3.7 A cross bar switch with Control Logic Circuit

When the analog switch is in "OFF" state, two NMOS transistors let the two I/O nodes connect to ground through resistances. As the CC primitives sent from the CP at both nodes simultaneously, the voltage changes can be detected by CC sensor amplifiers and converted into digital format for ON/OFF Logic processing.

When the analog switch is in "ON" state, it actually behaves as a resistance. As the DC primitive sent from the HBA, the voltage changes

can be detected by DC sensor amplifier, and converted into digital format for ON/OFF Logic processing.

CHAPTER 4

DESIGN OF TRI-STATE SWITCH AND CONTROL PLANE

4.1 Design of Tri-state Switch and Control Plane

In this chapter, more detailed design of Tri-state switch and Control Plane is presented by using TSMC 0.25um technology. Figure 4.1 demonstrates the structure of the Tri-state switch and Control Plane of the proposed switching system. The left hand side of this structure is connected to Ethernet cables and the right hand side is attached to the Physical Plane.

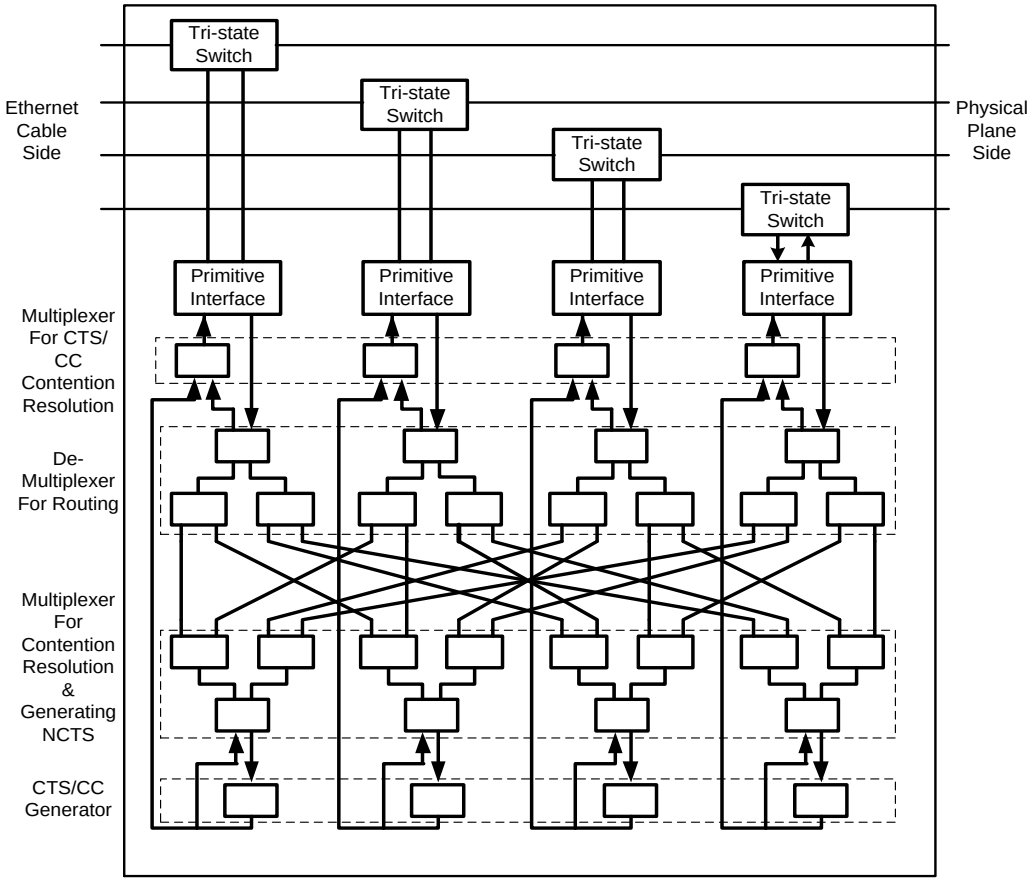

Figure 4.1 The structure of Tri-state switches and Control Plane

A. Primitives of the system

As described in the control architecture, there are several primitives operating in the switching system. Table 4.1 shows these primitives and their abbreviations.

Table 4.1 Primitive Abbreviation Table

Abbreviation	Primitive
RTS	Request To Send

RTS_SOF	RTS Start Of Frame
RTS_DSA	RTS Destiny Segment Address
RTS_CR	RTS Contention Resolution
RTS_CC	RTS Connect Command
RTS_EOF	RTS End Of Frame
NCTS	Not Clear To Send
CTS	Clear To Send
CC	Connect Command
DC	Disconnect Command
HBA_CP_EN	HBA to CP connection enabled
CP_HBA_EN	CP to HBA connection enabled
CP_PP_EN	CP to PP connection enabled
RTS_DSA0	RTS Destiny Segment Address “0”
RTS_DSA1	RTS Destiny Segment Address “1”

Five primitives, RTS_SOF, RTS_DSA, RTS_CR, RTS_CC and RTS_EOF, form the primitive RTS. Each primitive serves its function as described in Chapter 2. For example, after CP detects the RTS_SOF, it is enabled to receive RTS_DSA for routing (RTS_DSA0 or RTS_DSA1 respectively). When the routing process ends, RTS_CR is received to resolve contention. If the RTS primitive successfully reaches the bottom of

the “tree” structure, for contention resolution the RTS_CC is received to trigger the CP sending connection commands to PP as well as CTS’s to HBAs. Otherwise, RTS_CC would let the CP send NCTS primitives to HBAs. The last code word, RTS_EOF would disable the CP to receive other primitives.

We propose to encode these primitives with simple analog pulses as shown in Figure 4.2. As a result, control primitives and Ethernet frame signals can be distinguished in terms of different voltage amplitudes. For 10G Ethernet signals defined in IEEE802.3 standard, the transceivers carry transmitting power up to 5.2 dBm with 100 ohm impedance Ethernet cable load [9]. In other words, the voltage swing of Ethernet signals in each pair of Category 6a cable is around 0.72 volt unbalanced. For the control primitives, 2.5 volt voltage swing is proposed.

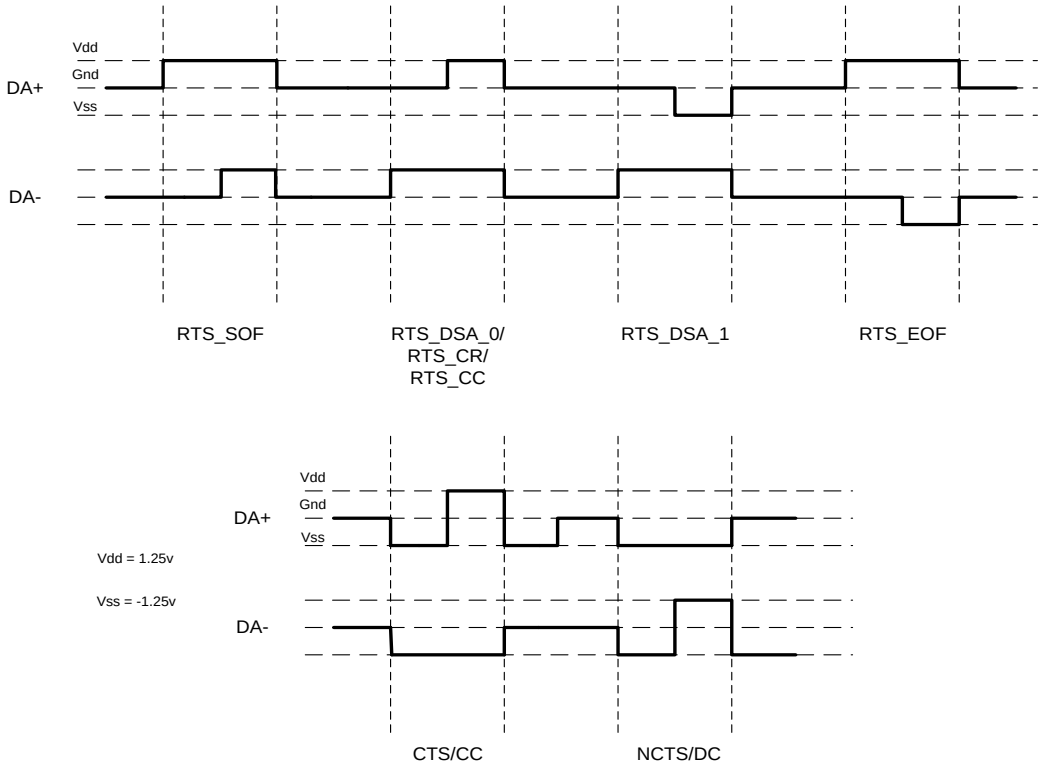

Figure 4.2 Waveforms of control primitives

B. Design of Tri-state Switch

The structure of a Tri-state switch is shown in Figure 4.3. As described in Chapter 2, it provides possible connectivity between three components: HBA, CP and PP. The connection between the HBA and PP is the CMOS switch for transmitting Ethernet signals which is mentioned in Chapter 3. The HBA to CP connection is a bidirectional buffer for which primitives are driven to send and receive between them. It is controlled by the “HBA_CP_EN” and “CP_HBA_EN” enable signals sent from Primitive Interface in the CP. The CP to PP connection is a one direction buffer to

drive the primitives sending from CP to PP. It is controlled by the "CP_PP_EN" enable signal also from the CP.

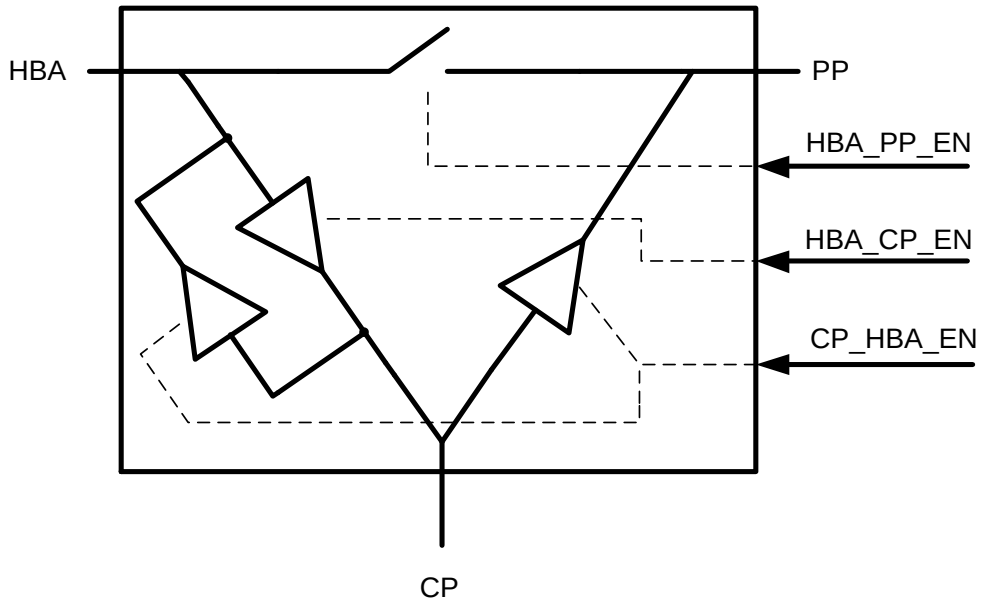

Figure 4.3 The structure of a Tri-state switch

C. Design of Primitive Interface

Figure 4.4 shows the block diagram of one port labeled Primitive Interface. Upon receiving the RTS primitive from an HBA through the Tri-state switch, the PI will decode the RTS into three signals RTS_DSA0, RTS_DSA1, and RTS_EN respectively as the inputs of a de-multiplexer. The decoded signals are all impulse signals with proper pulse width. The PI controls its respective Tri-state switch through HBA_CP_EN,

CP_HBA_EN, and CP_PP_EN once the PI decoded the RTS primitive. CTS or NCTS primitives are driven by the PI to signal a HBA or the PP.

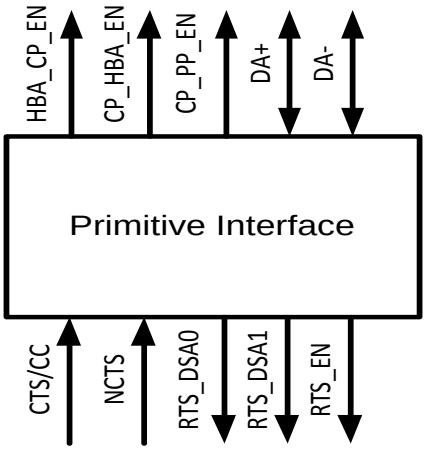

Figure 4.4 The block diagram of a Primitive Interface

The PI is designed by using different voltage level to support the decode circuit as well as the drive circuit.

D. Design of De-multiplexer and Multiplexer

The constructions of a De-multiplexer (De-MUX) and a Multiplexer (MUX) are shown in Figure 4.5 and Figure 4.6 respectively. The De-MUX routes requests while the MUX resolves conflicts. In Figure 4.5, Initially RTS_DSA0, RTS_DSA1, and RTS_EN will go through the buffer switch in the middle of the three buffer switches. If the pulse signal of the segment address “0” (RTS_DSA0) arrives first at the Arbiter, the outputs of the Arbiter will set the top buffer switch close and the middle and bottom

buffer switches open to let the next segment address primitive go to the next level de-multiplexer. If the impulse signal of RTS_DSA1 comes first, then the next primitive will be routed to the bottom switch. At the same time, the paths for transmitting CTS/CC and NCTS/DS primitives are controlled to be open or close respectively by the Arbiter. The Arbiter would reset the states of these switches receiving CTS/CC or NCTS/DC.

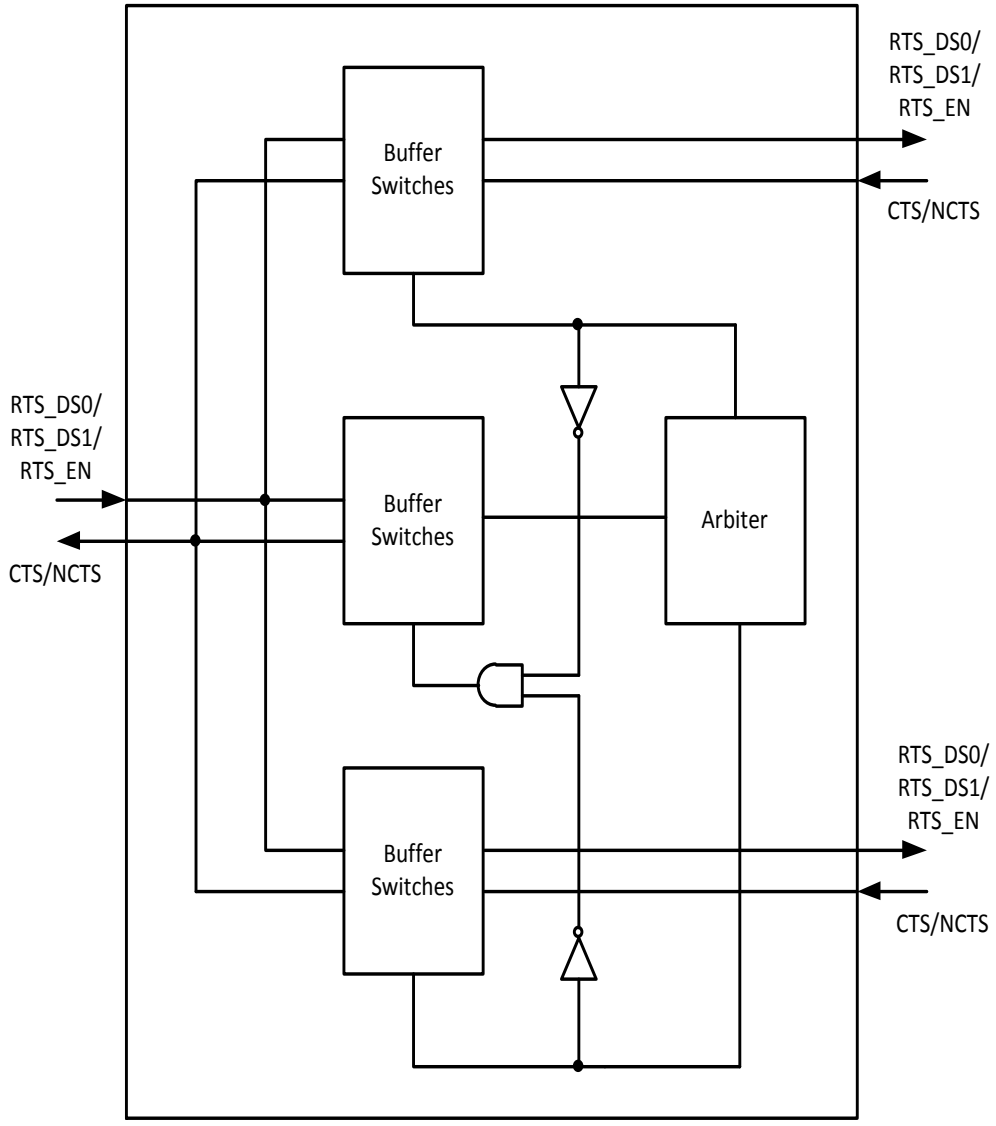

Figure 4.5 The structure of a De-Multiplexer

In Figure 4.6, the multiplexer is constructed similar to that of the de-multiplexer. The Arbiter performs the First Come First Server (FCFS) rule when a contention occurs and routes the winner node to the next level multiplexer. The loser node will be routed to the NCTS generator to send

the NCTS primitive back to the source HBA. The multiplexers and de-multiplexers are reset as the NCTS routes through.

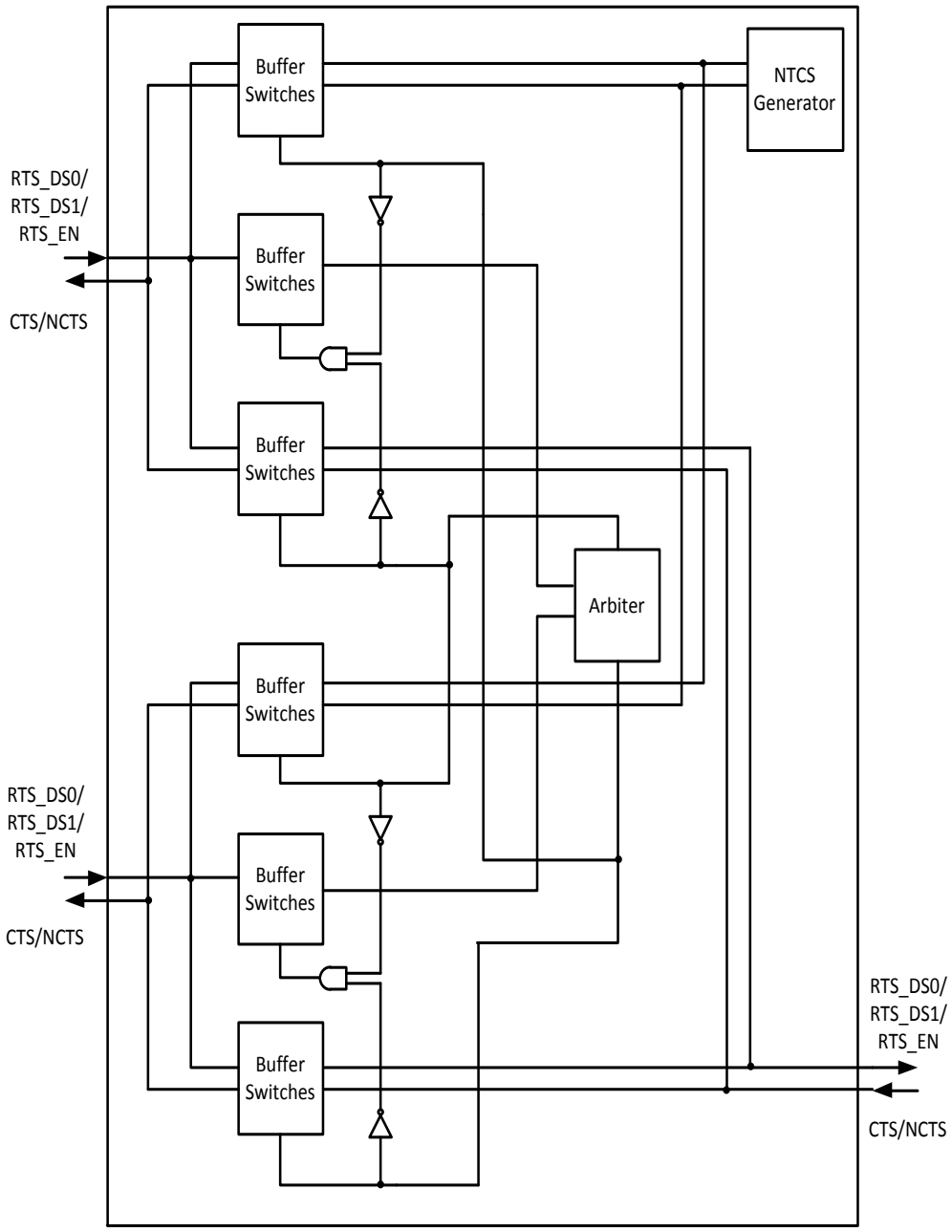

Figure 4.6 The structure of a Multiplexer

E. Design of CTS/NCTS Primitive Generator

Once the RTS primitive has reached the CTS generator or routed to the NCTS generator in the multiplexer, an impulse will be generated and sent back to the responding PI through CTS/NCTS paths opened by the RTS primitive. Figure 4.7 shows a typical impulse generator which is used in the CP with proper delay setup. By modifying the delay time, different width impulse could be generated. The PI would encode the coming CTS or NCTS impulse and to resend the source HBA.

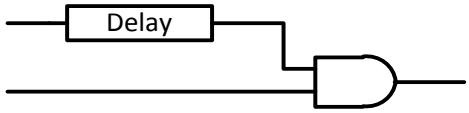

Figure 4.7 CTS/NCTS Primitive Generator

CHAPTER 5

SIMULATION

In order to prove the design concept and evaluate the performance, the design has been performed for the TSMC 0.25um CMOS process. We present in this chapter preliminary results of simulation by using Cadence and Advanced Design System software suites.

5.1 Simulation of Tri-state Switch and Control Plane

The simulation result of the Tri-state switch and Control Plane in terms of switching correctness and switching delay is illustrated in this section. Only one example is demonstrated as mentioned in section 2.4. Both HBA 11 and HBA 01 sent RTS primitives to HBA 10. However HBA 11 initiates the request earlier and successfully receives the CTS primitive. At the same time, the responding cross bar switch is set for data connection. On the other hand, HBA 01 loses the contention and receives the NCTS from the Control Plane.

A. Simulation setup

According to the control algorithm, a specific format of RTS primitive would be sent from HBA 01 and HBA 11, go through the Tri-state switch to arrive at the CP. Figure 5.1 shows the input signals of the switching system. HBA 01 has been delayed for 2 Nano seconds to send the RTS to the switch.

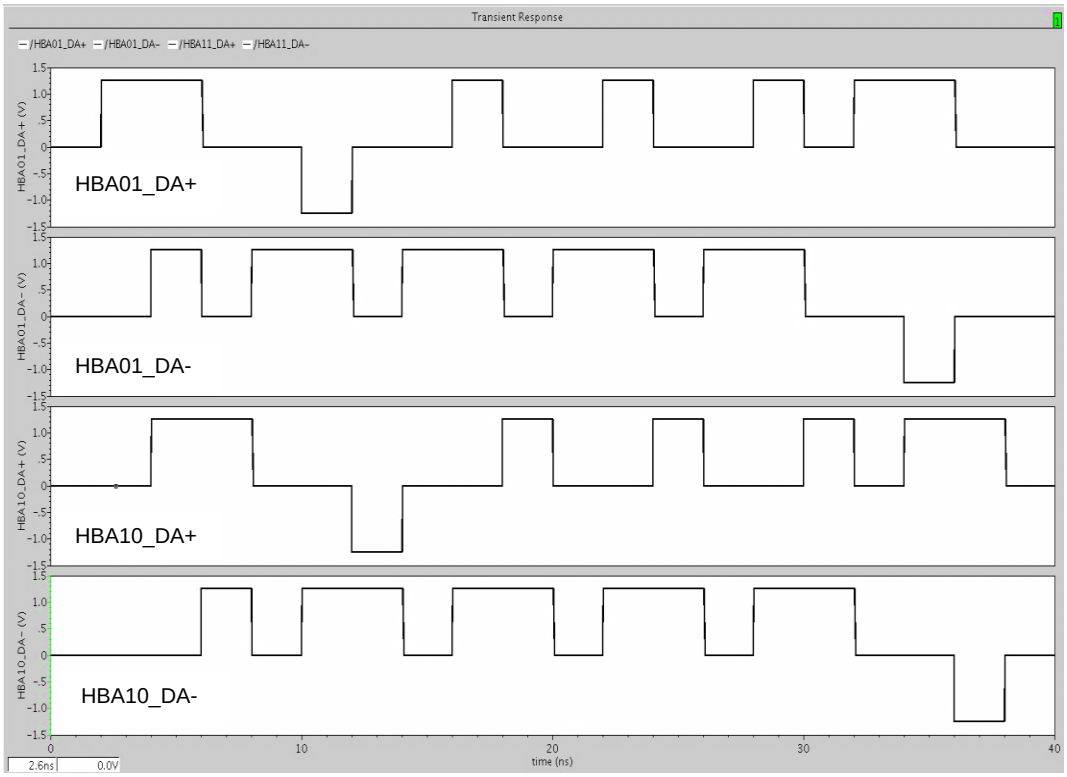

Figure 5.1 HBA01 and HBA11input signals

B. Results and analysis

The Primitive Interface component described in Chapter 4 plays the role of interconnecting the Tri-state switch and the rest part of CP. Figure 5.2, 5.3, 5.4 and 5.5 show the signals of Primitive Interface HBA 01 and HBA 11.

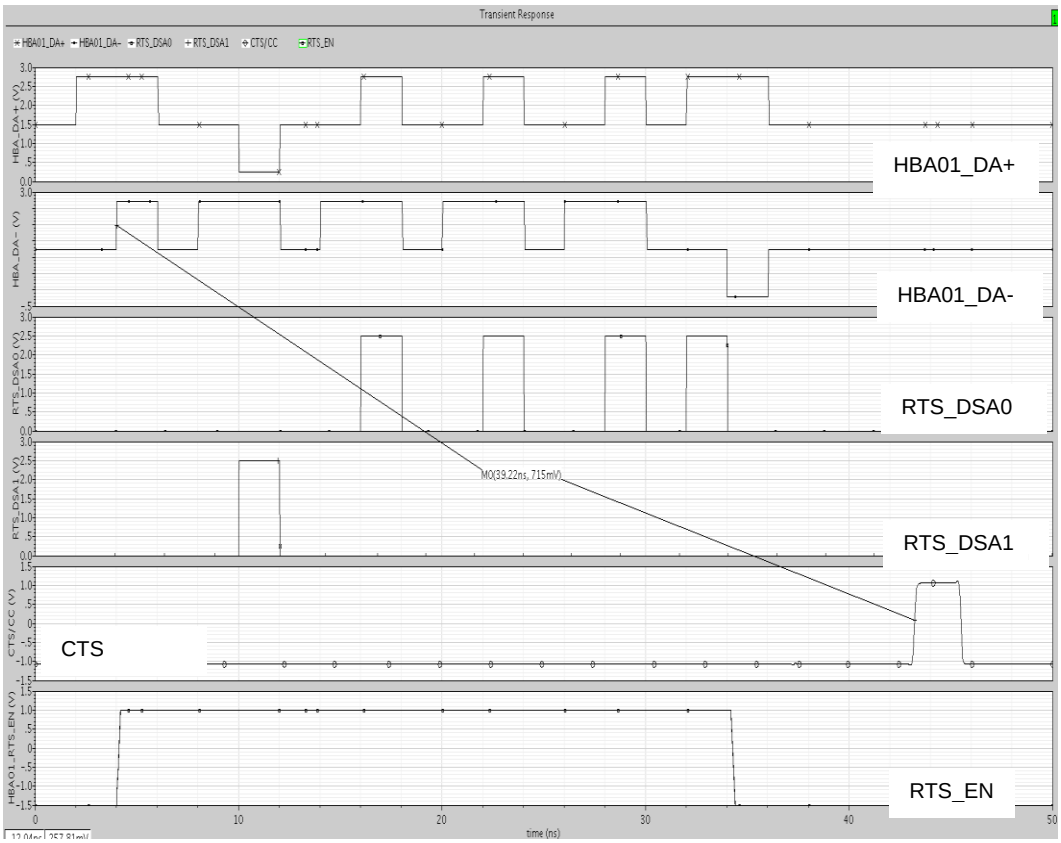

Figure 5.2 PI signals of HBA01 (a)

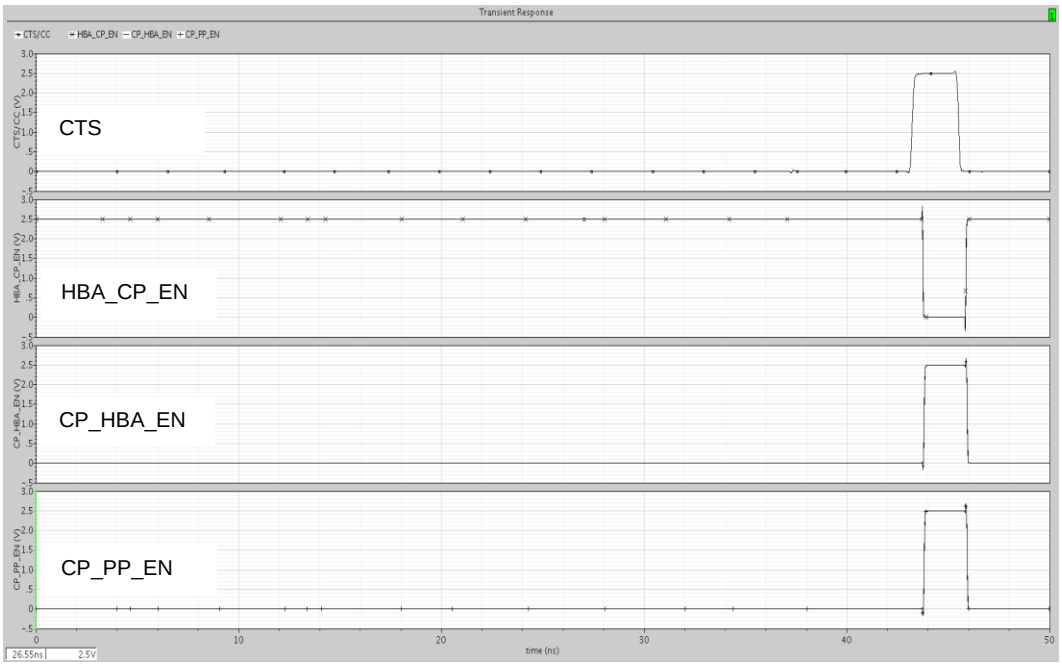

Figure 5.3 PI signals of HBA01 (b)

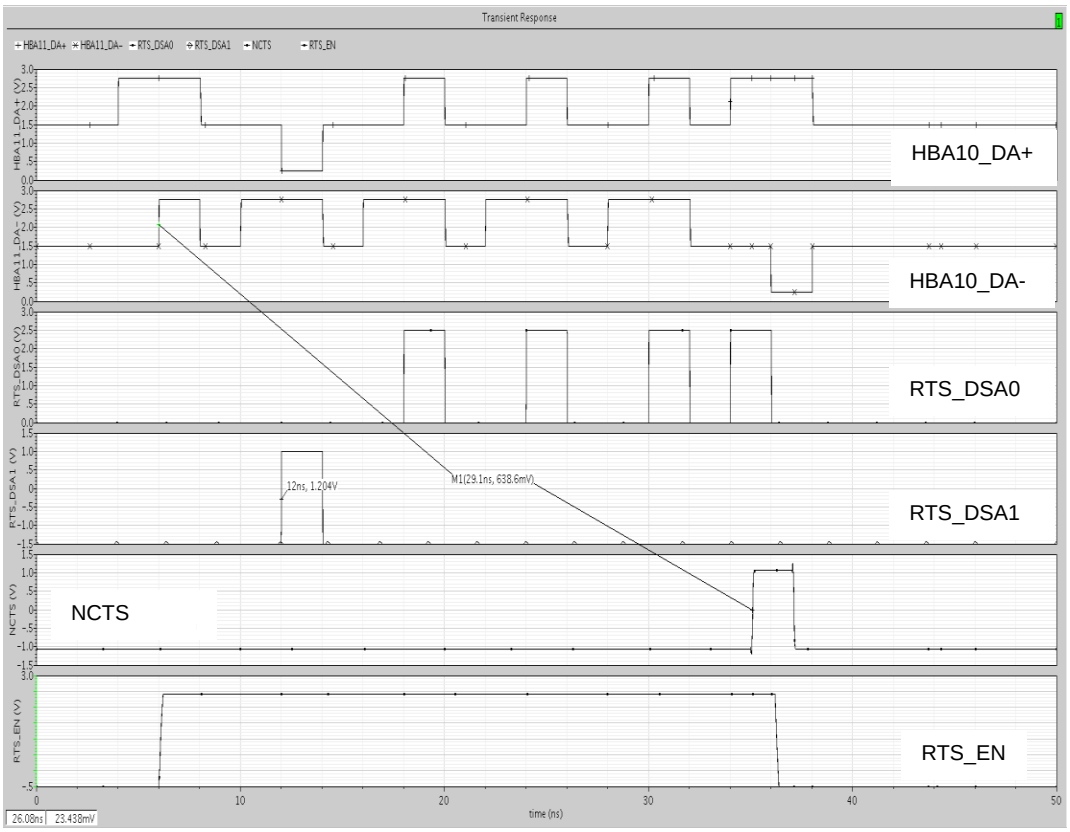

Figure 5.4 PI signals of HBA10 (a)

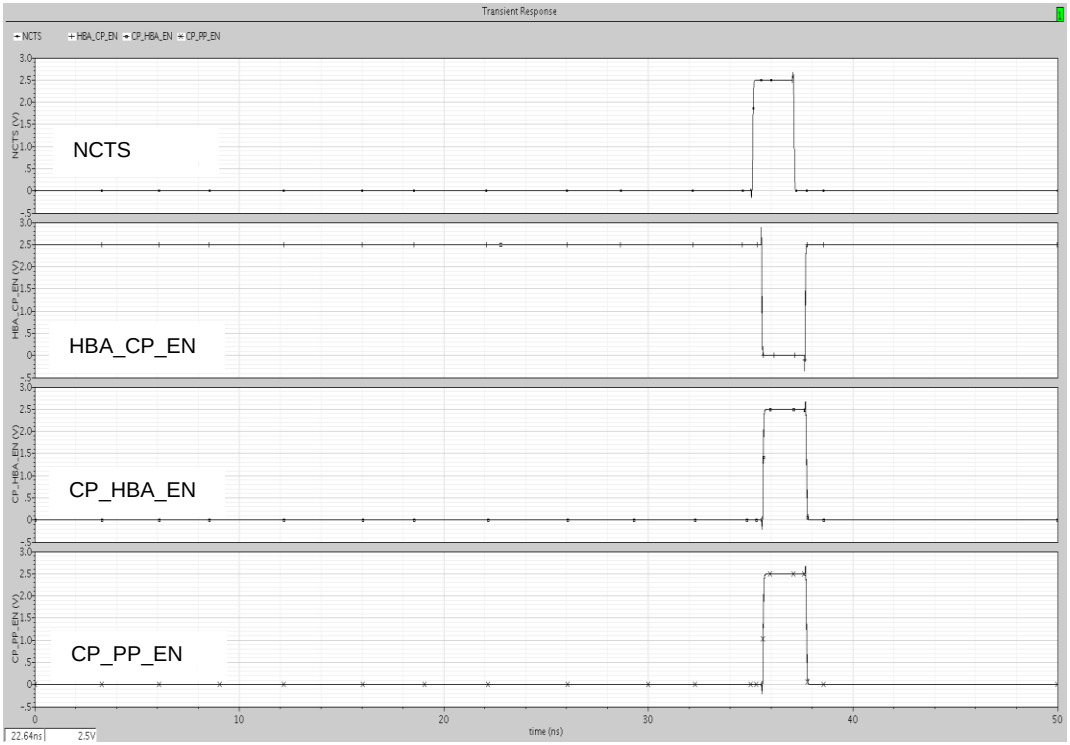

Figure 5.5 PI signals of HBA10 (b)

From the results shown in Figure 5.2, 5.3, 5.4 and 5.5, we found that the Tri-state switch and CP are working properly. The delay from sending a RTS from one HBA until it receives CTS or NCTS primitives from CP is 39.21ns or 29.10ns respectively.

5.2 Simulation of the Cross Bar Switch

The simulation result of the cross bar switch is presented in two parts: signal integrity test simulation and functionality test simulation.

A. Simulation setup

In order to test the signal integrity of the switching system, a system level simulation configuration is proposed as shown in Figure 5.3.

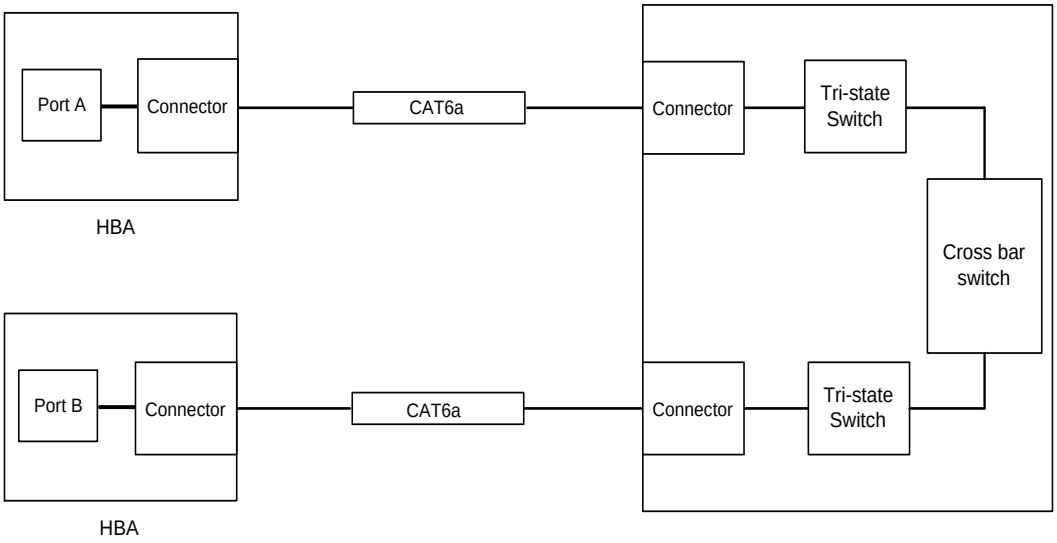

Figure 5.6 Signal integrity simulation configuration

For the top HBA of Figure 5.3, a power source (Port A) with 100 ohm differential impedance load is used to simulate 10GBASE-T PHY source. The receive end is the bottom HBA with 100 ohm impedance load (Port B). Between these two nodes are four connectors for two of CAT6a Ethernet cables, two Tri-switches which are on “HBA_PP” state and the cross bar switch. The length of the CAT6a cable is 9 meters. The cable model is published by the IEEE802.3an task force group [10]. The Connector model is obtained from the published data of a commercial RJ45 connector.

There are two operations of the cross bar switch. For turning “on”, two CC primitives have to be sent to the responding cross bar switch

simultaneously, and for turning “off”, a DC primitive has to be sent to the Tri-state switch and the cross bar switch.

B. Results and analysis

By importing the S-parameter models of connector and cable, the insertion loss and reflection loss of the entire data path from one HBA (Port A) to another HBA (Port B) as shown in Figure 5.3 are simulated. The insertion loss (IL) is the dB expression of the transmission coefficient |S21|. It is given by [11]

$$IL = -20\log_{10}|S_{21}|\ dB$$

The reflection loss is the dB expression of the reflection coefficient |S11|. It is given by [11]

$$RL = -20\log_{10}|S_{11}|\ dB$$

Only the first one pair (DA+/-) of four pairs of data paths (DA+/-, DB+/-, DC+/- and DD+/-) is simulated.

For cross bar switch with “ON” state, Figure 5.4 shows the IL versus frequency curves between Port A and Port B. For comparison, the IL requirement defined by the IEEE802.3an standard is shown in the figure. The RL versus frequency curves are shown in Figure 5.5.

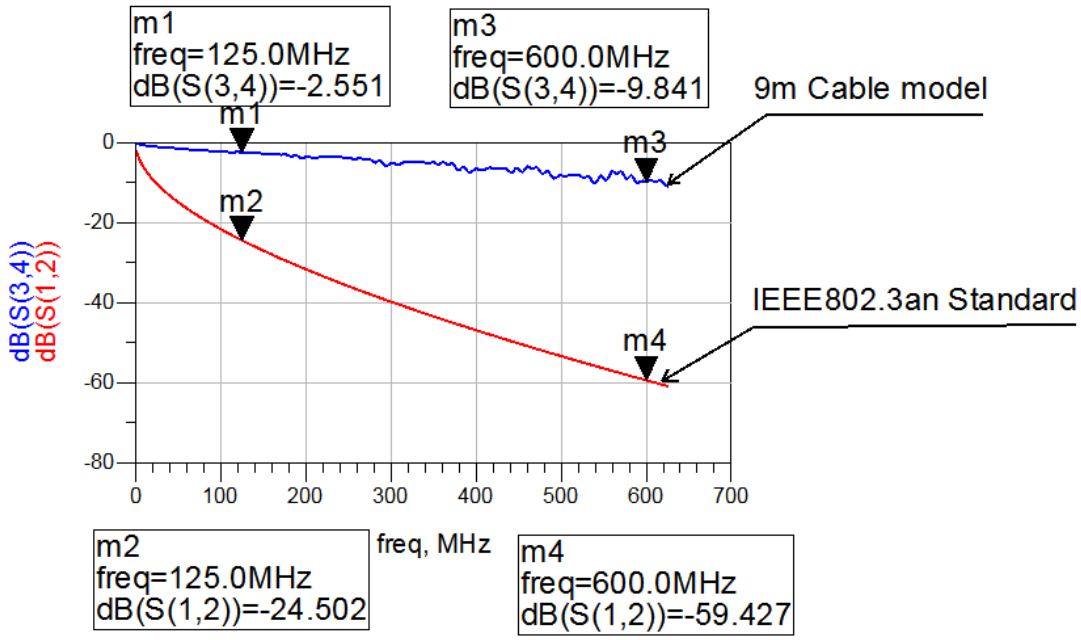

Figure 5.7 IL of the switching system and IEEE802.3an standard (a)

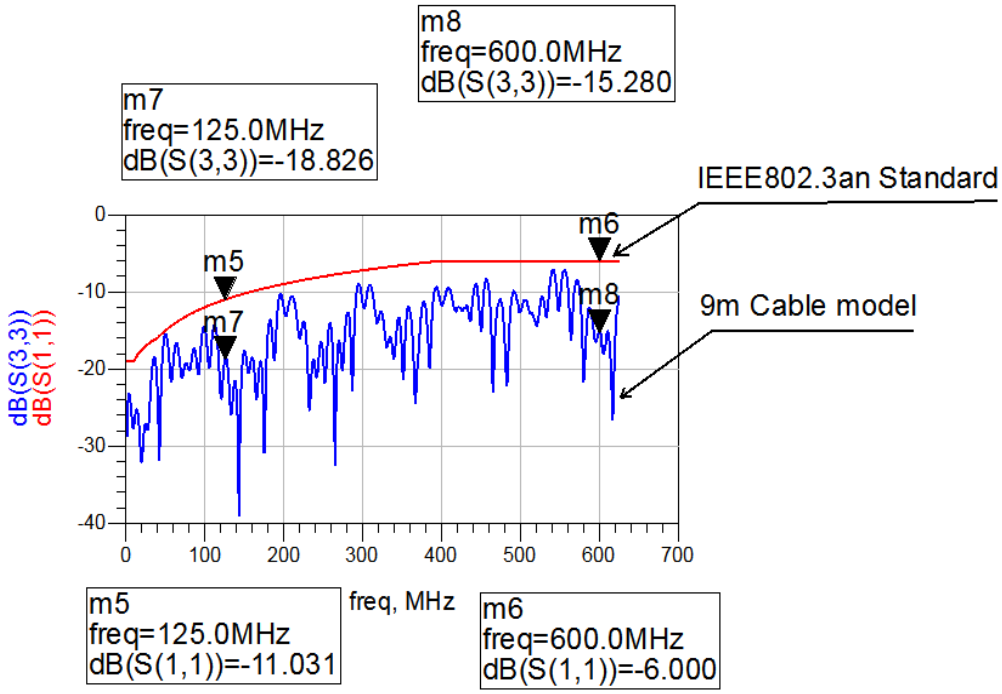

Figure 5.8 RL of the switching system and IEEE802.3an standard (b)

For cross bar switch with “OFF” state, Figure 5.6 and Figure 5.7 show the IL and RL curves respectively.

Besides using 9 meters cable model, different lengths of cable model are simulated as shown in Figure 5.8. From the simulation results, it is observed that the IEEE802.3an standard requirement cannot be met if the cable length extends to 100 meters. However, at the expense of a shorter length cable, the proposed switching system could still meet the IEEE802.3an standard requirement.

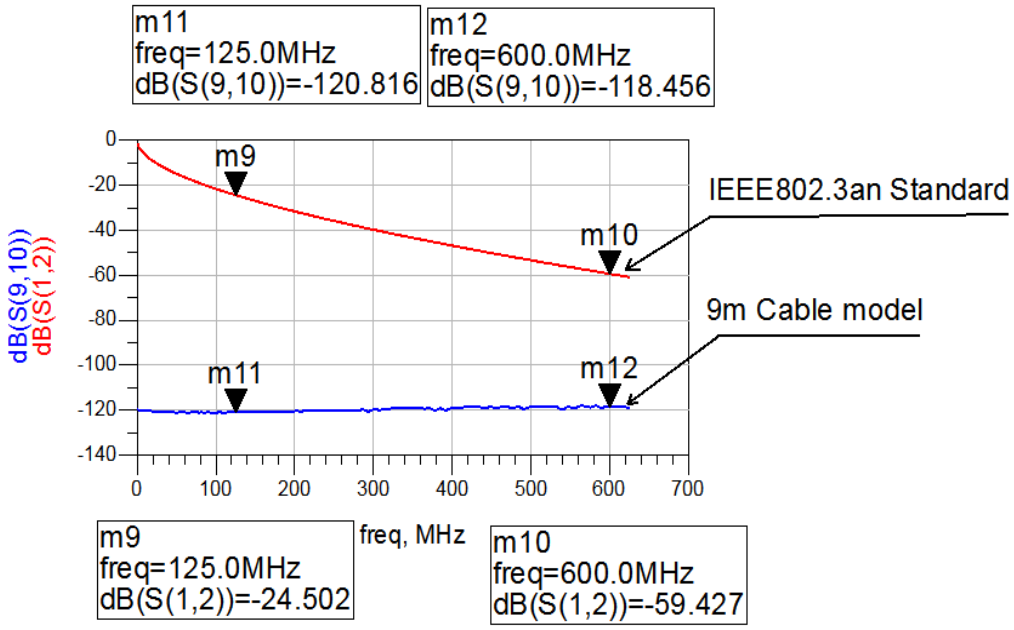

Figure 5.9 IL of the switching system and IEEE802.3an standard (a)

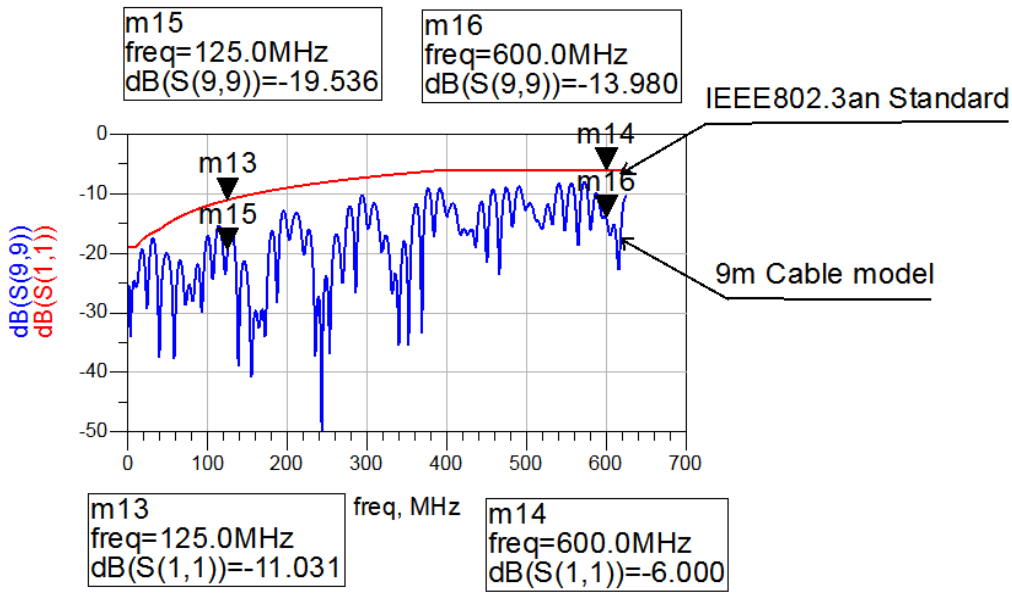

Figure 5.10 RL of the switching system and IEEE802.3an standard (b)

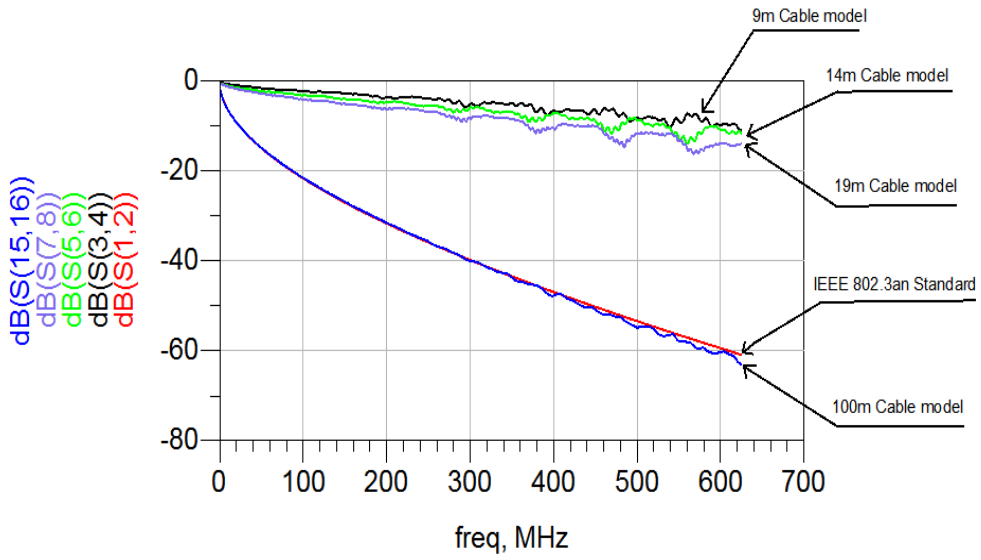

Figure 5.11 IL of different lengths of cable model

For functionality test, the simulation result is shown in Figure 5.9. When the cross bar switch receives the CC primitive, it then turns the EN signal high. When the cross bar switch receives the DC primitive, it then turns the EN signal low. The delay from decoding CC or DC primitive until setting the cross bar switch are 627.4ps and 1.322ns respectively.

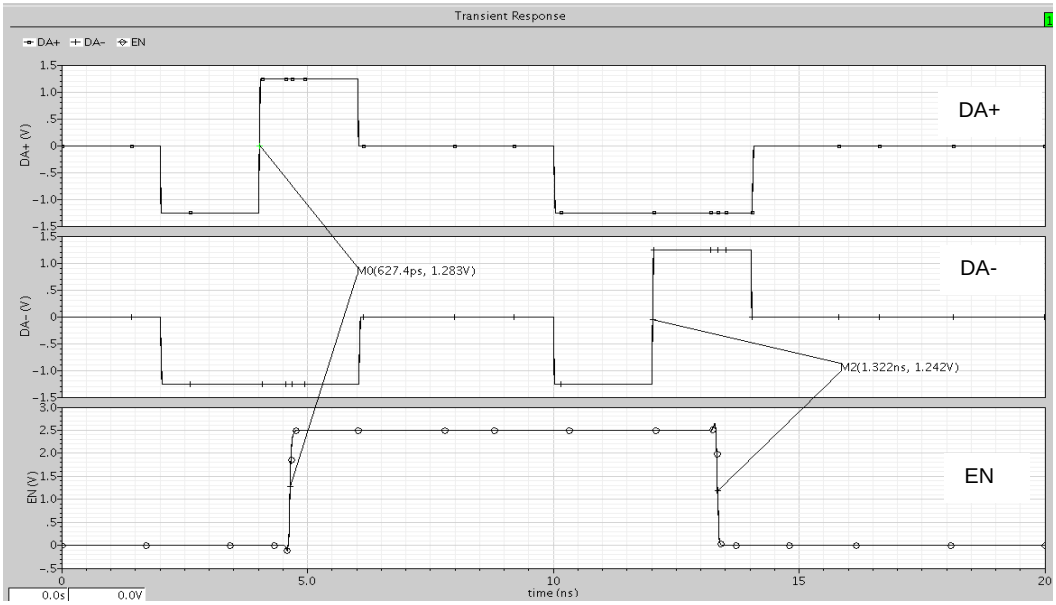

Figure 5.12 Simulation results of functionality test

CHAPTER 6

CONCLUSION AND FUTURE WORK

6.1 Conclusion

In this thesis, a novel switching system has been described for 10GBASE-T Ethernet. We describe a first stage implementation for CSMA/TS. We consider also multi-stage cross bar switch network. The proposed system reduces latency and power consumption compared with traditional switching systems. By routing customized headers in capsulated Ethernet frames, the proposed switching system would transmit the original Ethernet frames with little processing, thereby makes the system appear as a simple physical medium for different hosts. Two major parts of the switching system, namely the Control Plane and Physical Plane, are designed to be implemented using CMOS technology.

The Control Plane in the system is designed allow to parallel process primitives to be sent from hosts. Output contention is resolved on first-come-first-serve basis. The Physical Plane is designed to meet the requirement for 10GBASE-T Ethernet cabling segment specifications. Simulation works have demonstrated the operation of this switching system, establishing and releasing connection between two hosts.

In conclusion, an analog-based 10G Ethernet switching system is introduced and designed as a new method to reduce latency and energy

compared with the traditional share-memory or crossbar switches. Based on this design, multi-stage switching network could be further developed for CSMA/TS technique.

6.2 Future Work

All the design works are done under the assumption that the Host Bus Adapter could send any signals to the switch to realize the proposed switching system, we have to develop further for the 10G Ethernet HBA as described in Chapter 2. Other network functions mentioned in Chapter 2 also have to be developed in this Host Bus Adapter to fully realize the switching system.

A better analysis and/or simulation may be necessary to evaluate the signal integrity test for the switching system. We have preliminary results that the PP could transmit 10G Ethernet signals to achieve a reliable connection.

REFERENCES

[1] "Performance Optimized Ethernet Switching," Cajun White Paper #1, LucentTechnologies.

[2] "Cisco 12000 Gigabit Switch Router," White Paper, Cisco Systems, 1997.

[3] Joseph Y. Hui and David A. Daniel, "Terabit Ethernet: a Time-Space Carrier Sense Multiple Access Method", IEEE Globecom 2008.

[4] IEEE Computer Society, IEEE Std 802.3™-2005, Part 3 Carrier Sense Multiple Access with Carrier Detection (CSMA/CD) Access Method and Physical Layer Specifications, June 9, 2005.

[5] Apurva N Sahasrabudhe, Terabit Ethernet, M.S. Thesis, Arizona State University, May 2009.

[6] IEEE Computer Society, IEEE Std 802.3an™-2006, Part 3 Carrier Sense Multiple Access with Carrier Detection (CSMA/CD) Access Method and Physical Layer Specifications, Amendment 1: Physical Layer and Management Parameters for 10 Gb/s Operation, Type 10GBASE-T, September, 2006.

[7] R. Jacob Baker, CMOS Circuit Design, Layout, and Simulation, Second Edition, Wiley-Interscience, 2008.

[8] "Analog Switches and Multiplexer Basics," *White Paper*, Analog Devices.

[9] IEEE P802.3 10GBASE-T Study Group (*http://grouper.ieee.org/groups/802/3/10GBT/index.html*).

[10] 10GBASE-T Task Force, *http://grouper.ieee.org/groups/802/3/an/index.html*.

[11] Eric Bogatin, Signal and Power Integrity Simplified, Second Edition, Prentice Hall, 2004.

[12] Norbert Eicker, Thomas, Lippert, "A Scalable Ethernet Clos-Switch," NIC Symposium 2006, NIC Series, Vol. 32, pp.307-314, 2006.

[13] G. Ungerboeck, "10GBASE-T modulation & coding, set of fixed precoders and start-up," IEEE P802.3an Task Force Meeting, Nov. 2004 (*http://www.ieee802.org/3/an/public/nov.04/ungerboeck.1.1104.pdf*).

This document was generated using the Graduate College Format Advising tool. Please turn a copy of this page in when you submit your document to Graduate College format advising. You may discard this page once you have printed your final document. DO NOT TURN THIS PAGE IN WITH YOUR FINAL DOCUMENT!

65

65