國 立 交 通 大 學

光 電 工 程 研 究 所

碩士論文

分散式控制混合型被動光纖網路之光線路終端設計

Optical line terminal design for novel distributed control hybrid passive optical network

研　究　生：方 建 和

指 導 教 授：陳 智 弘 老師

中 華 民 國 九 十 六 年 七 月

分散式控制混合型被動光纖網路之光線路終端設計

Optical line terminal design for novel distributed control hybrid passive optical network

研 究 生 :方建和　　Student : Chien-Ho Fang

指導教授 :陳智弘 老師　　Advisor : Associate Prof. Jyehong Chen

國 立 交 通 大 學

光 電 工 程 研 究 所

碩 士 論 文

A Thesis

Submitted to Institute of Electro-Optical Engineering

College of Electrical Engineering and Computer Science

National Chiao Tung University

In Partial Fulfillment of the Requirements

For the Degree of Master In

Institute of Electro-Optical Engineering

July 2007

Hsinchu, Taiwan, Republic of China

中 華 民 國 九 十 六 年 七 月

致謝

Acknowledgements

一轉眼兩年的碩士生涯終於要畫上句點了，很感謝在許多人的幫助下才能完成這篇碩士論文。

首先要感謝我的指導教授陳智弘老師帶領我進入光通訊這個領域，使我對光通訊有更進一步的了解，同時在陳老師亦師亦友的教導下，學習到很多待人接物和求學應有的態度。另外特別要感謝工研院簡清雲學長，在他細心和不厭其煩的教導下，以及實驗期間不斷容忍我所犯的過錯給我鼓勵。如果沒有簡清雲學長的耐心指導相信今天這篇論文無法順利誕生。還有魏嘉建學長、蔡昇佑學長、林澤雨學姊，以及五樓的其他學長們，無論是在學業及論文的研究，軟硬體的使用上也給了我許多幫助，常常很多遇到的難題在請教他們之後都能迎刃而解。

還有陪伴我度過這兩年碩士生涯的同學，曹正、彭晟峰、江俊臻，不論是在課業上，還是日常生活上，都幫我度過了許多的難關。還有一年級的眾學弟，因爲有了你們，光電聯誼盃的歡樂，聚餐時的八卦等，都將是難忘的回憶。

最後，我要感謝我的家人，分享著我碩士生涯的點滴，並在背後默默地支持。要感謝的人實在太多，總之，感謝大家！

分散式控制混合型被動光纖網路之光線路終端設計

學生：方建和　　指導教授 ： 陳智弘 老師

國立交通大學光電工程研究所碩士班

摘要

人們對頻寬的需求愈來愈大，我們最常用的非對稱數位用戶迴路已無法提供足夠的頻寬，所以愈來愈多的人開始使用光纖到家這項新的技術。光纖到家相對於非對稱數位用戶迴路來說，頻寬大了 20 至 30 倍。而在光纖到家的技術中，最常被聽見的非被動光網路莫屬了。以太被動光網路可提供的頻寬為 1.25 Gbps, ，這個數字遠超過數位用戶迴路所提供的頻寬。使用分時多工與分頻多工，我們會發現他們的時槽與波長的利用是沒有效率的，因為他們之中的閒置時槽與波長沒有辦法被共同使用，所以就有人提出動態頻寬配置。而以前動態頻寬配置的方式皆是由光網路單元送信號給光線路終端，然而光網路單元與光線路終端的距離都是在 20 公里以上，所以我們發現動態頻寬配置沒有辦法及時被更新，所以我們提出分散式的動態頻寬配置。分散式動態頻寬配置是將信號從光網路單元送至分離器即可，這大大地減少動態頻寬配置被更新的時間。而我們採用分時多工與分波多工混用是因為如果每個光網路單元都使用一個波長的話，那所耗費的成本會是相當地昂貴。而在光線路終端、光網路單元和動態頻寬配置，我們都是用現場可程式化閘陣列來完成。

Optical line terminal design for novel distributed control hybrid passive optical network

Student：Chien-Ho Fang Advisor : Dr. Jyehong Chen

Institute of Electro-Optical Engineering

National Chiao-Tung University

Abstract

More and more people require more and more bandwidth. It can not supply enough bandwidth by xDSL, so passive optical network (PON) is widespread little by little. TDM (Time Division Multiplexing) PON and WDM (Wavelength Division Multiplexing) PON are both bandwidth inefficient because the free slots and wavelength can not be shared. So someone use “Dynamic Bandwidth Control (DBA)" to solve this problem. The prevailing method of centralized control DBA is Interleaved Polling with Adaptive Cycle Time. But the distance from the ONU to the OLT is over 20 km, the round trip time is too long. The DBA can not be update immediately. So we let the calculation of DBA is near the splitter. The splitter is closer to the ONU and far from the OLT. So it can update DBA immediately. And we use mixed structure including TDM and WDM. Then we call it “Hybrid". So the name of our architecture is “Distributed Control DBA Hybrid PON".

Contents

List of Figures

Chapter 1

Introduction

In 1990s, rapid progress of different types of the Digital Subscriber Line(DSL) technology and its widespread adoption was the factor impeding the development of the fiber to the home(FTTH). The cost of DSL of replacing existing cooper infrastructure is lower than FTTH. The data rate and the distance of the DSL is 24 Mbps/5 km and 50 Mbps/1.5 km. But more and more bandwidth are needed. It can not satisfy people with the DSL. Then the bandwidth drivers is shown as follows. Something we know need a lot of bandwidth like: high definition TV (HDTV), peer web usage (Youtube), business conferencing and telecommuting, distance learning, video on demand. And something we can see in several years like: telemedicine, high definition video on demand, complex multi-party games. Above all require a lot of bandwidth.

In recent years, people all over the world start to interest in access method based on the optical fiber. The first reason is that the Internet has become phenomenally popular and the number of customers requiring broadband access and willing to pay for it is increasing steadily. The second reason is that the new services have been developed that require more bandwidth that cannot be provided by DSL. There are many reasons I did not mention. In short, more and more people require more and more bandwidth. DSL is overwhelmed by FTTH in the supply of bandwidth.

We talk about the FTTH technology now. FTTH was implemented by many types of technology. Among these types of technology, the passive optical network(PON) is one of the most famous. The PON structure is shown as Fig. 1.1. The OLT is the abbreviation of

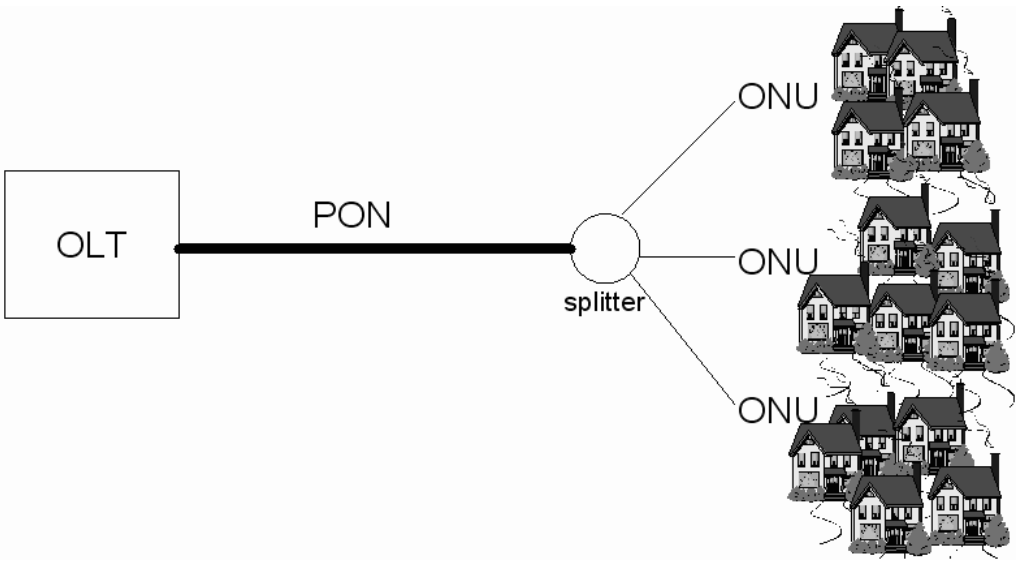

Figure 1.1: The architecture of PON.

the Optical Line Terminal and the ONU is the abbreviation of the Optical Network Unit. The advantage of the PON is all passive devices are used inside it. The maintainance of the passive devices is lower than the electrical equipment. And the infrastructure remain usable for tens years. The other advantage is that PON is point to multiple points technology. It means one OLT can support several ONUs. It can save much cost and many materials. In next chapter, I will introduce the popular technology of transmission in recent years.

Chapter 2

The conventional and novel architecture of PON

In this chapter, we will introduce many types of the technology about PON.

2.1 Point-to-Point Ethernet [1]

The widespread adoption of the Ethernet protocol in local networks and the growing demands for fast access networks are the key factors that drove the vendors grouped into Ethernet in the First Mile Alliance (EFMA) to work on new architectures enabling FTTH. The work of the EFMA task group a number of specifications has been released in recent years and different interfaces to the physical layer were defined.

- 100 BASE-LX
- 100 BASE-BX
- 1000 BASE-LX
- 1000 BASE-BX

In the proposed extensions the two end stations communicate with each other over a point-to-point connection in full duplex mode over a single (LX) or dual (BX) strand of single or multi mode optical fiber. The supported maximum transmission speed is

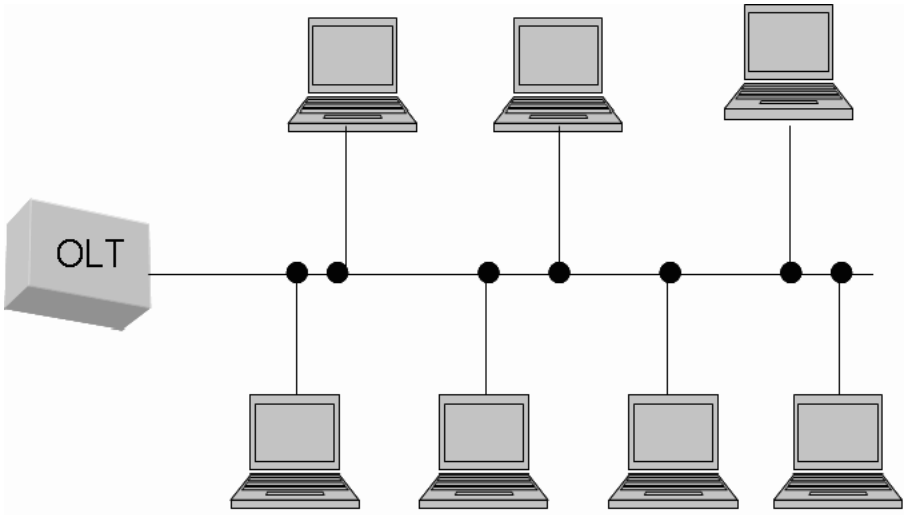

Figure 2.1: Configuration of Point-to-Point FTTH Network.

100Mbit/s for slower links (100BASE-X) or 1000Mbit/s for faster links (1000BASE-X). In comparison with the different DSL architectures the proposed specifications enable connecting stations which are up 10km apart. The typical architecture of network based on point-to-point connections is shown in Fig. 2.1.

But point-to-point Ethernet has a fatal disadvantage. The data collisions will happen when two nodes send the data simultaneously. When the data collision happen, the data will be given up. Although the data rare of point-to-point Ethernet can achieve 1000 Mbps, the data collision will decrease the data rate of the point-to-point Ethernet greatly.

2.2 Ethernet passive optical network(EPON) [2]

2.2.1 Overview

A Passive Optical Network (PON) is a single, shared optical fiber that uses inexpensive optical splitters to divide the single fiber into separate strands feeding individual subscribers. PON are called “passive” because other than at the CO and subscriber endpoints, there are no active electronics within the access network. Using these techniques to create a passive optical infrastructure, Ethernet in the First Mile PON (EFMP) builds a point-to-multi-point fiber topology that supports a speed of 1 Gbps for up to 20 km. The

structure is shown as Fig. 1.1. Eliminating the need for electrical equipment in the first mile network is a key facet of the EFMP topology. Another advantage is that much less fiber is required than in point-to-point topologies.

2.2.2 EPON network

An EPON network includes an optical line terminal (OLT) and an optical network unit (ONU). The OLT resides in the Central Office (CO). This would typically be an Ethernet switch or Media Converter platform. The ONU resides at or near the customer premise. It can be located at the subscriber residence, in a building, or on the curb outside. The structure is shown as Fig. 1.1.

2.2.3 EPON system

EPON is configured in full duplex mode in a single fiber point-to-multi-point (P2MP) topology. Subscribers, or ONUs, see traffic only from the headend. Each subscriber cannot see traffic transmitted by other subscribers, and peer-to-peer communication is done through the headend, or OLT. The headend allows only one subscriber at a time to transmit using a Time Division Multiplex Access (TDMA) protocol. EPON systems use an optical splitter architecture, multiplexing signals with different wavelengths for downstream and upstream as such:

- 1490 nm for downstream
- 1310 nm for upstream

2.2.4 The downstream and the upstream of EPON

The downstream of EPON is shown as Fig. 2.2. The users only get the information they want because the data go through the filters before the user get the data. And the upstream of EPON is shown as Fig. 2.3. The upstream of EPON use TDMA protocol. If the area is high load, the bandwidth allocation is good. If the area is low load, the bandwidth allocation is terribly inefficient because many slots are not be used.

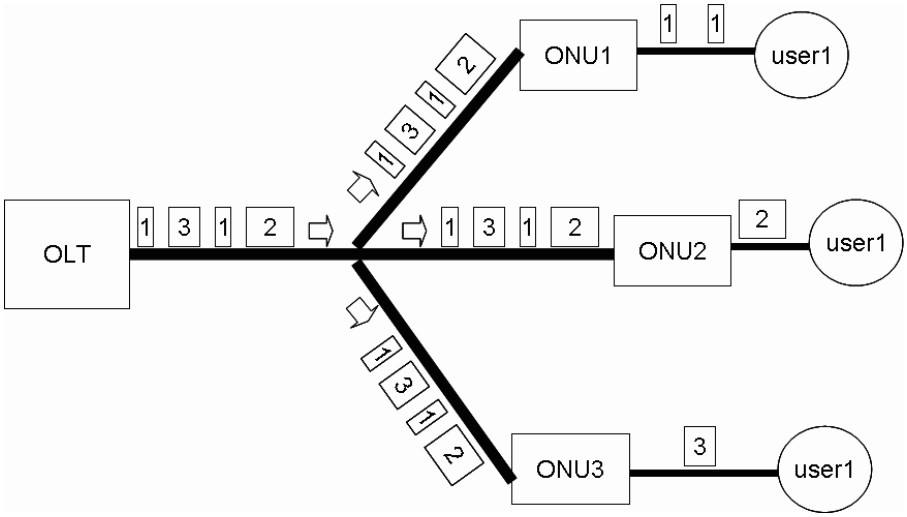

Figure 2.2: Downstream EPON Operation .

2.3 EPON using the centralized control dynamic bandwidth allocation (DBA)

EPON using the centralized control dynamic bandwidth allocation(DBA) means before the ONUs send the data to the OLT, the ONUs must transmit the test signal to the OLT. Then the OLT send the ack to the ONUs. It is more efficient than the EPON using TDMA. The test signal and the ack from the ONUs and the OLT will waste some bandwidth, but it save more bandwidth than EPON using TDMA protocol in the low load area. Because EPON using TDMA protocol in low load area lead to many slots not to be used. It waste much bandwidth.

2.3.1 Round trip time(RTT) of EPON

Round trip time(RTT) of EPON means the time required for a signal to travel from the ONU to the OLT. The calculation of the RTT is shown as the Fig. 2.4. The signal is sent from the ONU at T1, and the signal arrive the OLT at T2. The RTT of EPON is 2(T2-T1). In the EPON structure, the distance from the OLT to the ONU is very far(usually exceed 20 Km), so the RTT is the key point of the EPON structure. Because the RTT is not small, the DBA will not be update immediately. So the centralized control DBA of EPON is not

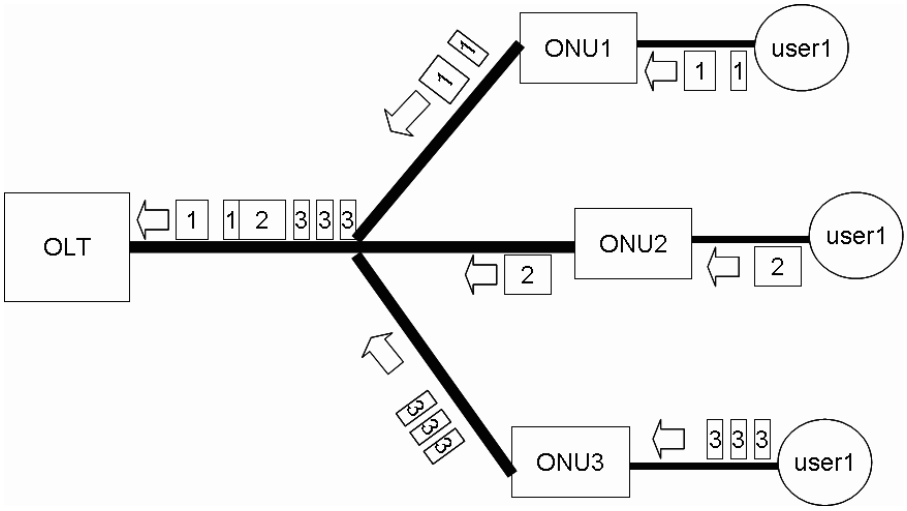

Figure 2.3: EPON upstream control.

good too.

2.3.2 Interleaved polling with adaptive cycle time (IPACT)

The prevailing method of centralized control DBA is Interleaved Polling with Adaptive Cycle Time (IPACT). Now we introduce IPACT shortly. The ONUs tell the OLT the demands of the time slots. Then the OLT will assign the time slots that is needed to each ONU. It is shown as Fig. 2.5. And now we will show you an example of IPACT. It is shown as Fig. 2.6. The RTT is 5 slots delay. At first, the queue size of each ONU equals zero. The OLT ask each ONU how many packets stored in the buffer. Then the ONUs send an ack to the OLT. The ack is the queue size of each buffer. Then the OLT assign the slots to each ONU. If the time slots of each ONU finish, it will send the new queue size to the OLT. If all ONUs are low load, IPACT is terribly inefficient. The shadow means the slots are not used.

2.4 EPON using the distributed control DBA

As the last section said, the centralized control DBA of EPON can not update the queue size of each buffer immediately, so we use the distributed control DBA to solve this problem. The distributed control DBA means the test signal will not be sent from the ONUs to

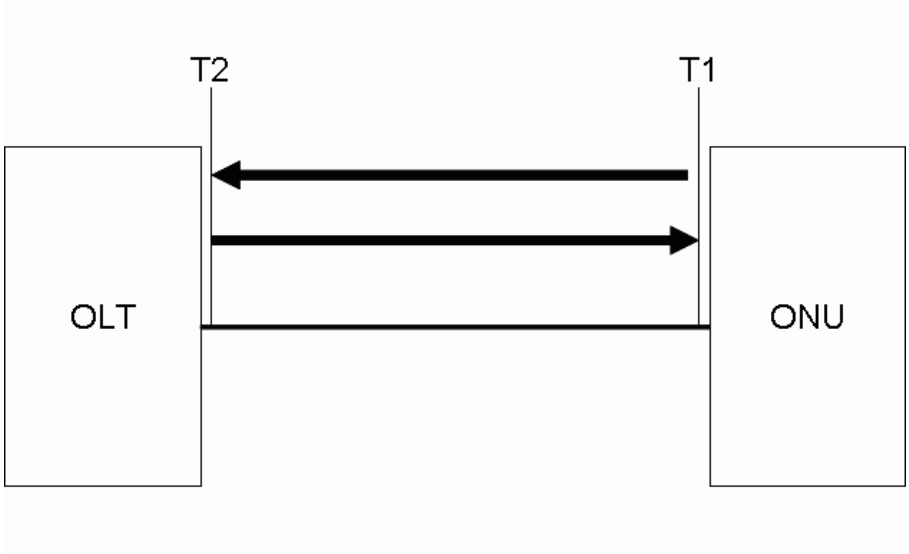

Figure 2.4: RTT of EPON.

the OLT. The test signal only is sent from the ONUs to the splitter, as shown in Fig. 2.7. It decrease the RTT greatly because the distance from the ONUs to the splitter is much smaller than the ONUs to the OLT. The broadcasting signal use the wavelength of 1550 nm. It is different from the upstream(1310 nm) and the downstream(1490 nm).

2.4.1 The concept of the distributed control DBA

The concept of the distributed control DBA is shown as Fig. 2.8. The left of the Fig. 2.8 is about the operation of the control message and the right of the Fig. 2.8 is about the data uploading. The control message is separated from the data uploading because they use different wavelength. If the ONUs send the control message to the splitter and then it will return the message about the DBA. Because the distributed control DBA has short RTT, it can update the queue size of each buffer immediately.

2.4.2 The example of the distributed control DBA

The example of the distributed control DBA is shown as Fig. 2.9. The control message include the queue size of the ONUs and the message of DBA. All the ONUs send the queue size to the splitter and after calculation the message of DBA will be sent back to

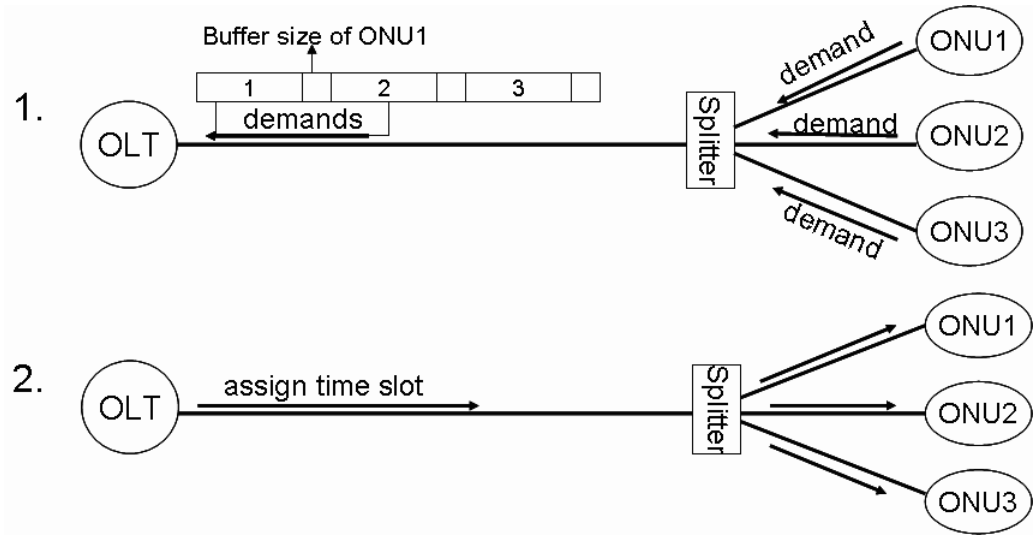

Figure 2.5: The concept of IPACT.

the ONUs. At first, two packets in the ONU1 are transmitted and then one packet in the ONU2 are sent. Originally one packet in the ONU2 are transmitted, but the message of DBA tell ONUs only the packet in the ONU1 can be sent. After that, two packets in the ONU3 are transmitted. Then the packet in the ONU1 will follow the ONU3. But the message of DBA tell all ONUs that only the packets in the ONU2 can be sent.

2.5 WDM PON

The structure of WDM PON is shown as Fig. 2.10. All ONUs use different wavelength. because each ONU use specific wavelength, the data collision will not happen. But the wavelength can not be shared by other ONUs. So WDM PON has a fatal disadvantage in the bandwidth allocation. And there is another disadvantage in WDM PON. Each ONU use specific wavelength, so the cost of WDM PON is very expensive. Maybe two or more ONUs use a wavelength is better.

2.6 The new architecture of PON

We introduce many structure of PON so far. All above have its fatal disadvantage. So we take the mixed structure of PON discussed above. The name of the new structure is

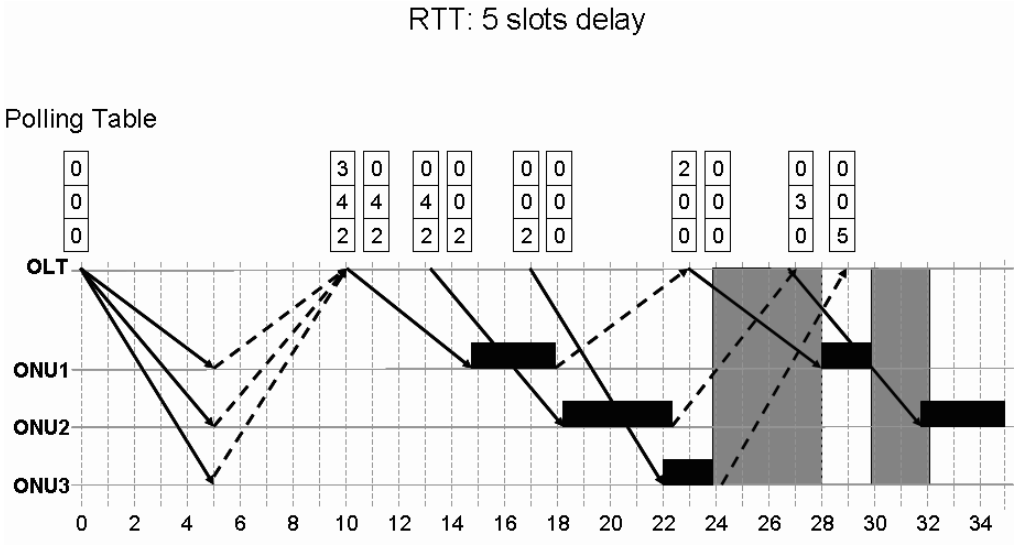

Figure 2.6: An example of IPACT.

"distributed control DBA Hybrid PON". The "hybrid" means the mixed structure of the TDM and WDM PON. We will introduce this new architecture as follows.

2.6.1 The architecture of distributed control hybrid PON

The architecture of distributed control hybrid PON is shown as Fig. 2.11. The thick arrow means data packets uploading(1310 nm). The thin arrow means the control signal broadcasting(1550 nm). Between the ONUs and the AWG uses TDM PON, and the other part use WDM PON. The AWG is the abbreviation of "Arrayed Waveguide Grating". The AWG make the different wavelength separated. The distributed control DBA increase the bandwidth efficiency greatly and WDM PON between the AWG and the OLT make the cost much lower. So this new structure makes the bandwidth efficient and the cost lower. We will introduce the details of the architecture as follows.

2.6.2 The OLT

We use "Field Programmable Gate Array(FPGA)" to implement the OLT. The OLT data flow diagram is shown as Fig. 2.12. The OLT data flow include two parts: the upstream and the downstream. There are several modules including the PON Processor, the ONU

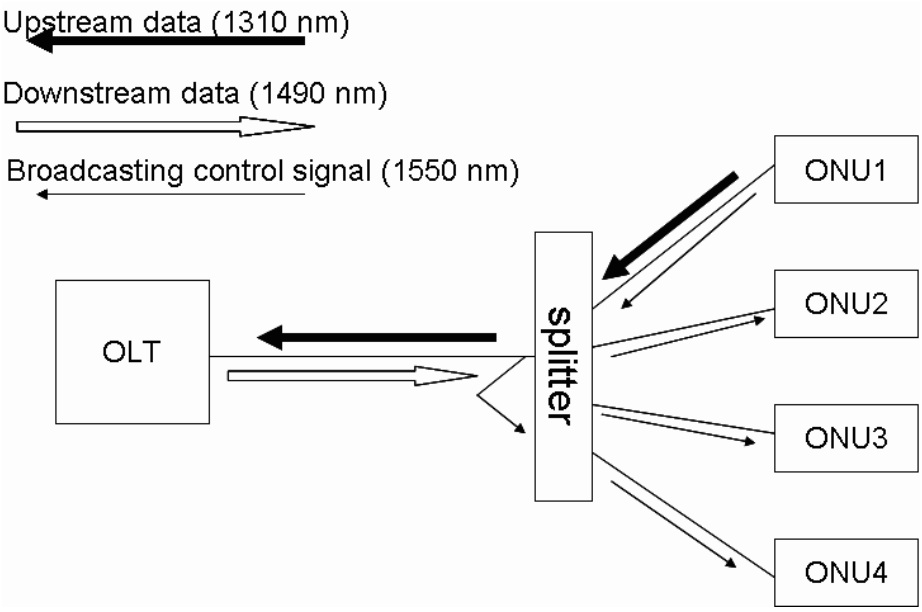

Figure 2.7: Distributed control DBA.

Buffers, the Multiplexers, Gigabit Ethernet Media Access Control(GbeMAC), and Gigabit Media Independent Interface(GMII) in the upstream. And there are several modules including Gigabit Media Independent Interface(GMII), GbeMAC, the Downstream Buffer and the Framer in the downstream. The functions of these modules will introduce in the chapter3.

2.6.3 The ONU

The same as the OLT, we implement the ONU by using FPGA. The ONU data flow diagram is shown as Fig. 2.13. There are three parts in the ONU data flow diagram: the upstream, the downstream and the DBA control. There are several modules in the upstream including Media Independent Interface (MII), Ethernet Media Access Control (Ethernet MAC), the Buffer, the Framer and the Data Buffer. The MII module do something of transfer 4 bits data into 16 bits data. The Ethernet MAC module count the length of the data transmitted. The Buffer module works like a memory. It store the data transmitted and the length of the data transmitted. The Framer module add the header before each packet transmitted. The header include the preamble, delimiter , ONU-ID and the length of each packet. Finally, the Data Buffer module store the data sent from the Framer mod-

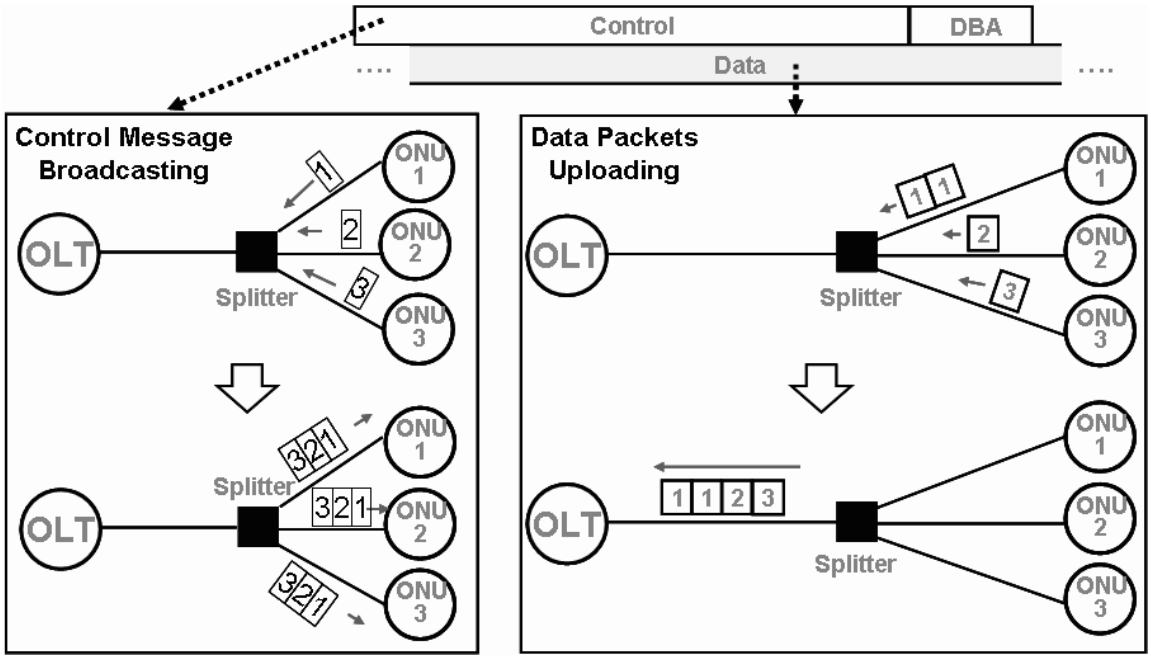

Figure 2.8: The concept of the distributed control DBA.

ule and not send the data to the next stage until the Framer module get the signal from the DBA Processor module. There are two modules in the downstream including the PON MAC module and Media Independent Interface(MII). The PON MAC module read the MAC address to know where the data are sent. The MII module do something of transfer 16 bits data to 4 bits data. It is opposite to MII in the upstream. The third part is the DBA control. All the modules in the DBA control part do something of calculating the queue size in each ONU . If ONU1 has the largest queue size, and the DBA processor send the signal to the Data Buffer. Then the data stored in the Date buffer can be sent to the PHY.

2.6.4 The data format of the upstream

The data format of the upstream is shown as Fig. 2.14. Each packet has the fixed length (280 bytes). The length of Ethernet packet is 46 to 1500 bytes, so each Ethernet packet is divided into 1 to 6 parts. If the packet equal to 280 bytes, the Idle will not be added. But if the packet is less than 280 bytes, the Idle will be added and expand to 280 bytes.

The header include the preamble, the delimiter, the ONU-ID and the payload-length. The preamble include 8 bytes and does something of synchronization. The delimiter

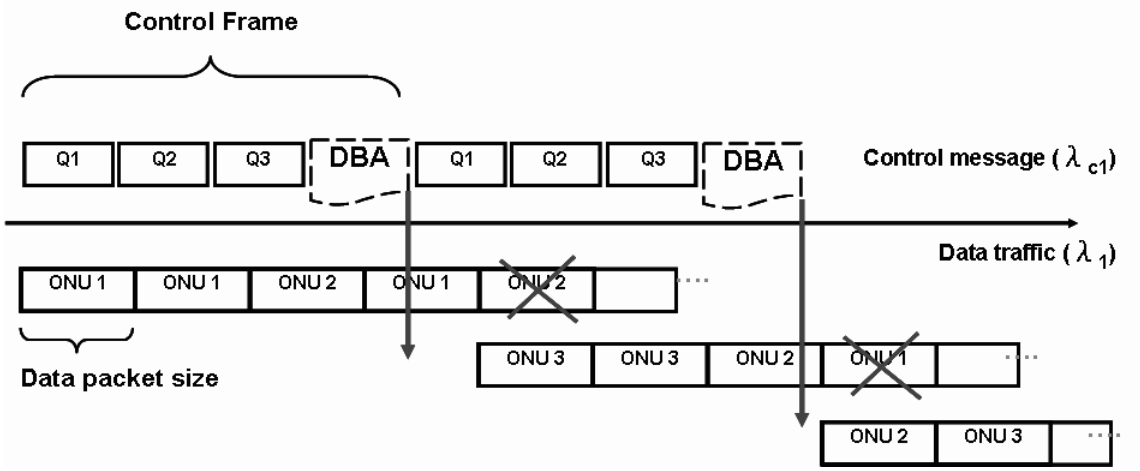

Figure 2.9: The example of the distributed control DBA.

include 1 bytes. It tells us the data is coming. The ONU-ID include 1 bytes. It tells us the data come from which ONU. The payload-length include 2 bytes. It tells us how many bytes the packet include. The most significant bit (MSB) of the payload-length is a special bit. Ethernet packet may be divided into several frames. If the MSB of the payload-length equal 1, it means this frame is the last frame of the Ethernet packet.

2.6.5 The data format of the downstream

The data format of the downstream is shown as Fig. 2.15. It is different from the upstream. It does not has the fixed length. The header will be added before the Ethernet packet. The Idle is added when no Ethernet packet are sent. The header include PSYNC, the delimiter and the payload-length. PSYNC is like the preamble. It is for synchronization. And the work of the delimiter and the payload-length is the same as that in the upstream.

2.7 Summary

TDM (Time Division Multiplexing) and WDM (Wavelength Division Multiplexing) are both bandwidth inefficient because the free time slots and wavelength can not be shared. We use dynamic bandwidth allocation (DBA) to make up for it. Then the prevailing method of centralized control DBA is Interleaved Polling with Adaptive Cycle Time

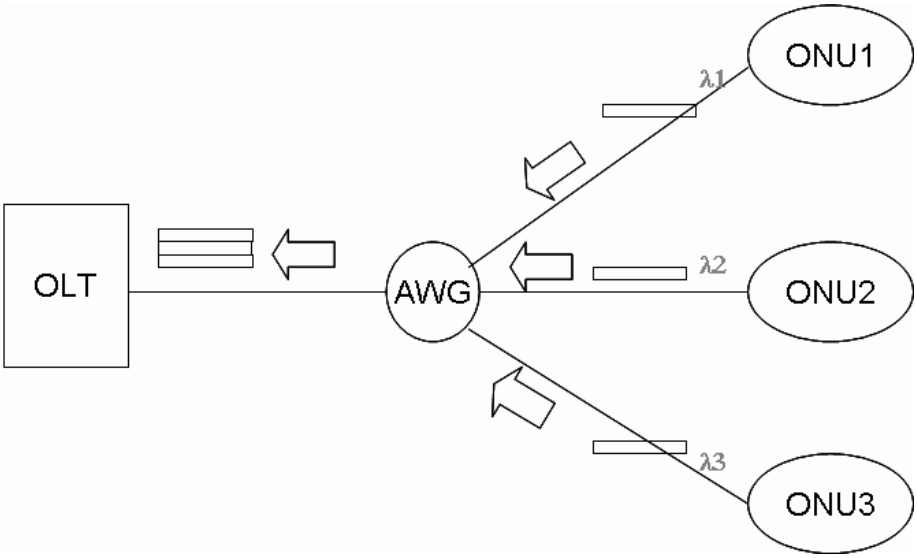

Figure 2.10: The architecture of WDM PON.

(IPACT). But IPACT has a fatal disadvantage. The round trip time is very long. So we use the distributed control DBA. Because the broadcasting signal only is sent from the ONUs to the splitter, the round trip time is short. The queue size will be update immediately.

Finally, we will show the comparison between IPACT and distributed control DBA. Fig. 2.16 show the "Packet Mean Delay" of IPACT and DHPON. Obviously, the packet mean delay of IPACT is much greater than DHPON. It is due to the long round trip time of IPACT. The farther the distance from the ONU to the OLT is, the longer of the packet mean delay it has. When the traffic load is over 0.8, the packet mean delay of IPACT increase rapidly. The packet mean delay almost does not change even the traffic load is over 0.8. Fig. 2.17 show the "Packet Drop Rate" of IPACT and DHPON. We introduce the concept "Packet Drop rate" first. Each Ethernet packet is assigned a packet life time before it is sent. So if the packet delay is greater than the packet life time, the packet will drop. From Fig. 2.17 we know that when the traffic load is over 0.92, the packet drop rate of IPACT almost equal 1. It means all the packets are dropped. Whatever the traffic load is, the packet drop almost equal zero.

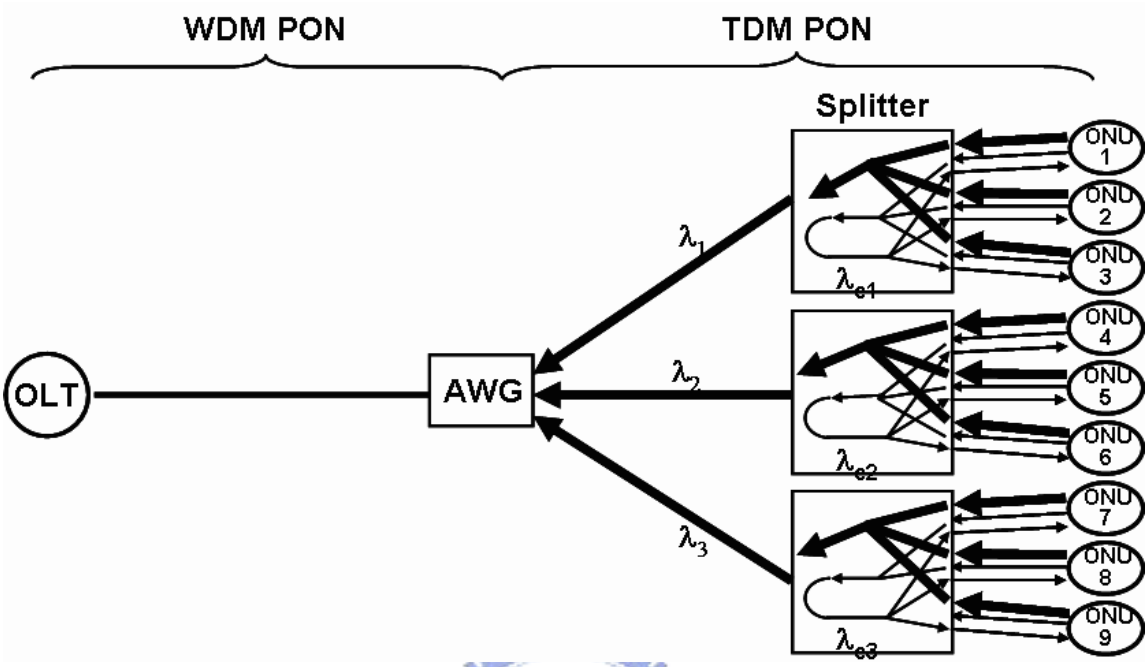

Figure 2.11: The architecture of distributed control DBA hybrid PON.

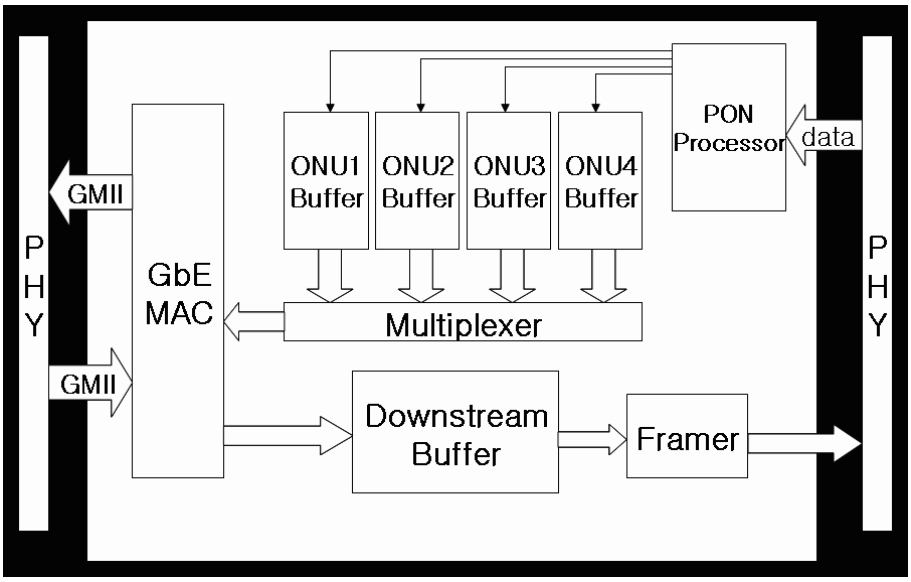

Figure 2.12: The OLT data flow diagram.

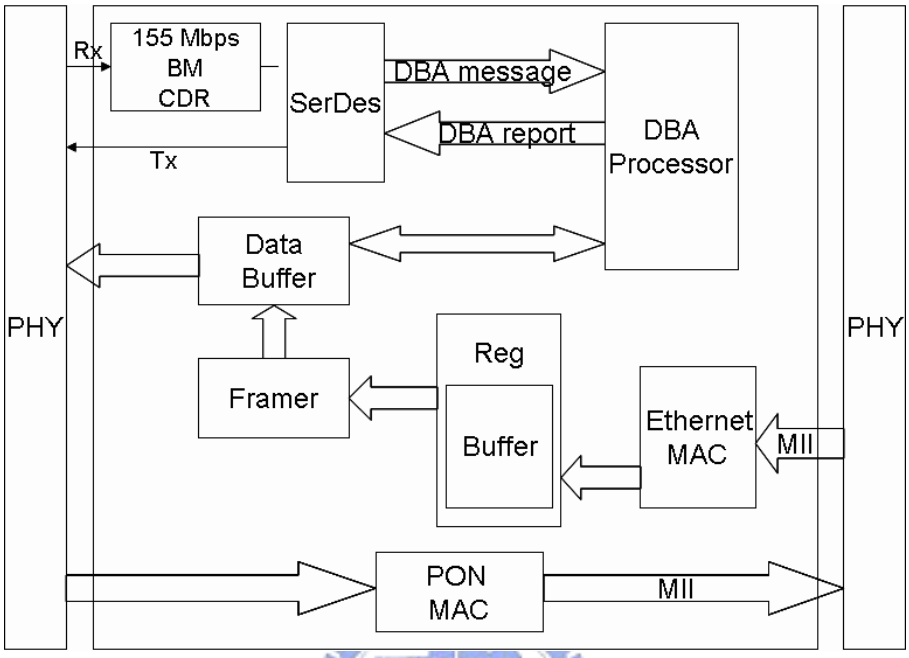

Figure 2.13: The ONU data flow diagram.

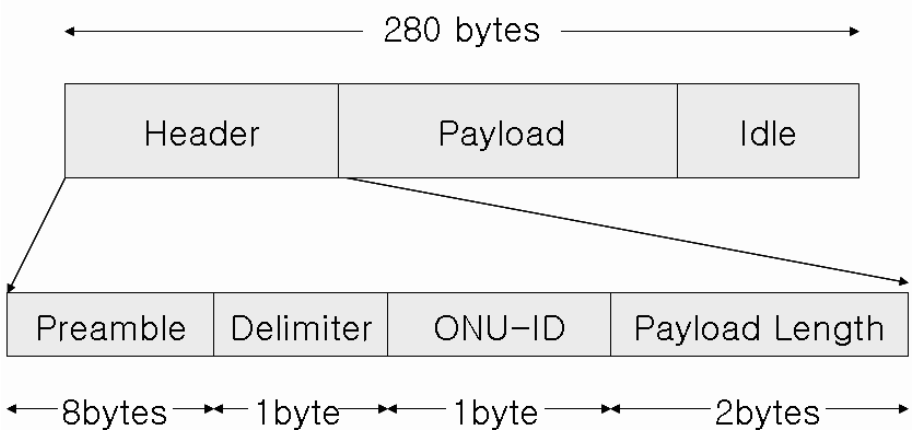

Figure 2.14: The data format of the upstream.

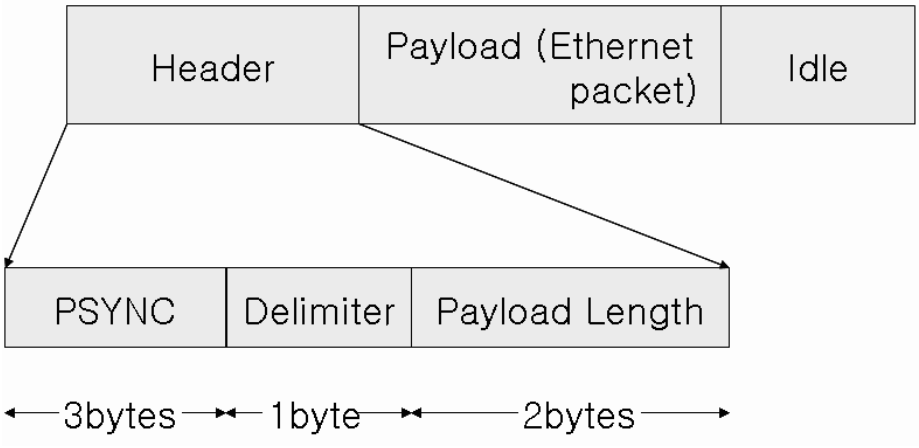

Figure 2.15: The data format of the downstream.

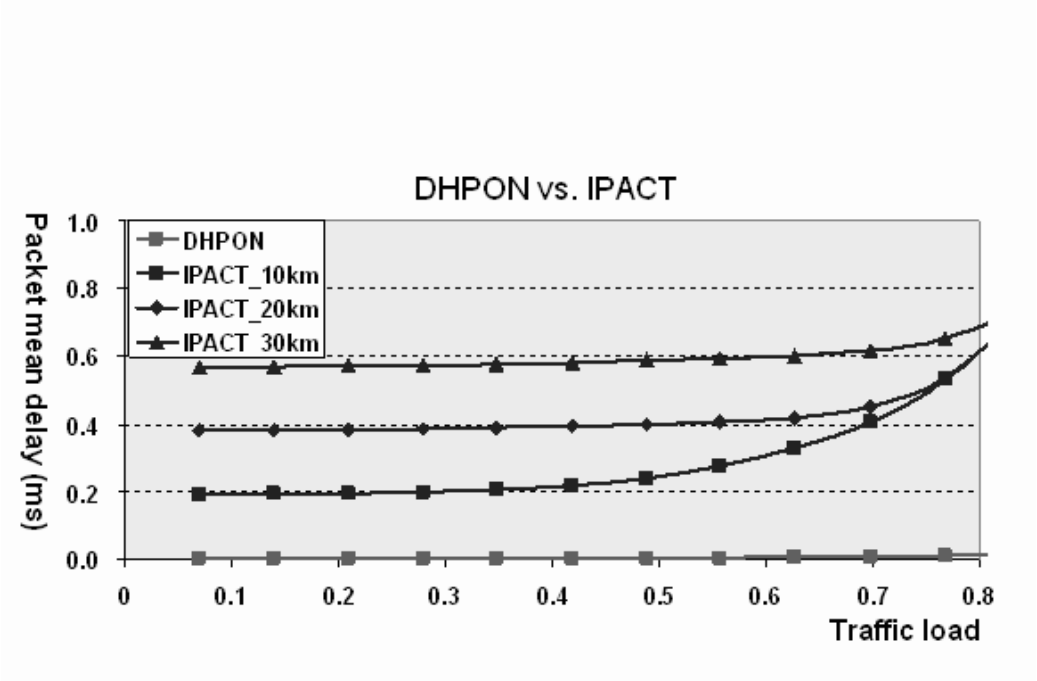

Figure 2.16: Packet mean delay of IPACT and DHPON.

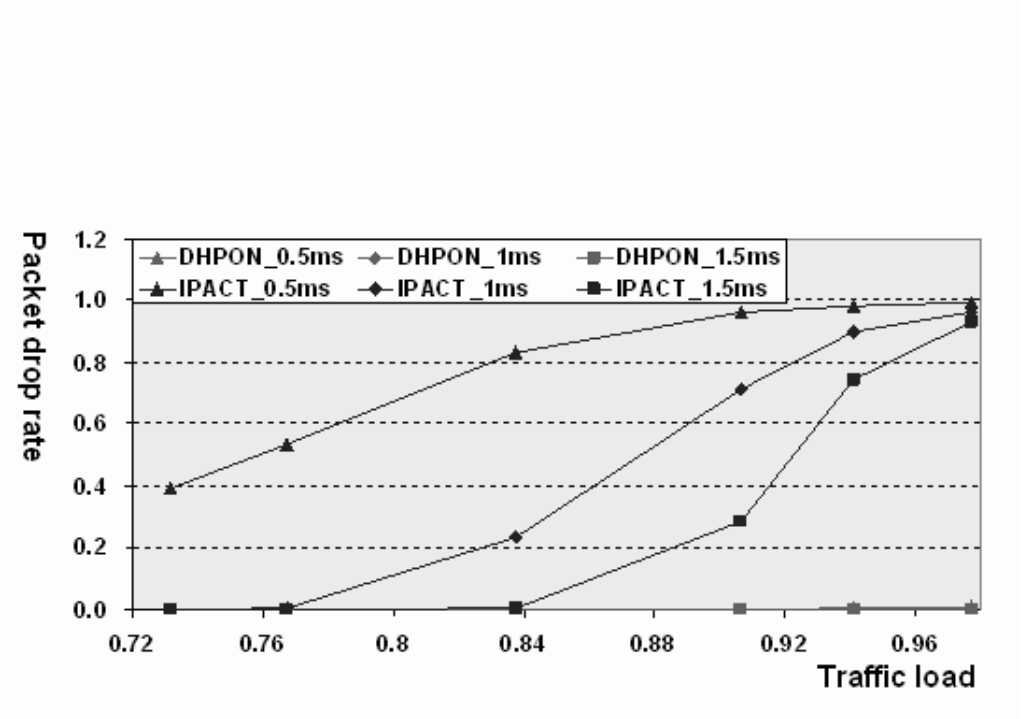

Figure 2.17: Packet drop rate of IPACT and DHPON.

Chapter 3

Simulation in software

In this chapter, we discuss the simulation of the OLT system. The OLT system is divided into two parts: the upstream and the downstream. There are some modules in the upstream including the PON Processor, the ONU Buffer, the Multiplexer and Gigabit Media Independent Interface (GMII). In the downstream, there are some modules including GMII, the Downstream Buffer and the Framer. The complete OLT data flow diagram is shown as Fig. 3.1. The most important is the data transmission rate between GMII and PHY is 125 Mbps and the other is 77.76 Mbps. Then I introduce the functions of these modules in the rest of this chapter.

3.1 The upstream

Before I introduce the functions of these modules, I introduce the format of the data transmitted first, as shown in Fig. 3.2. The preamble includes 8 bytes. It is just useless data but do something of synchronization. And behind the preamble is the delimiter. The delimiter includes only one bytes. It do something of telling us data is coming. And we put the data transmitted in the different ONU Buffers just depending on the ONU-ID. The payload length means the number of bytes of the data transmitted. The payload means the data transmitted. All the packets have fixed length(280 bytes), so if the number of bytes of the data transmitted is fewer than 280 bytes, it will expand by adding the idle. All above is the description of the data format. As the last section said, there are some

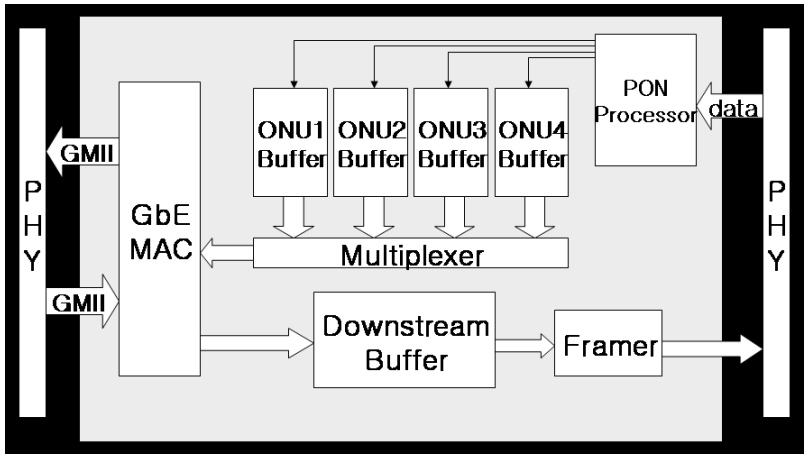

Figure 3.1: OLT data flow diagram.

modules included in the upstream and the downstream. I introduce the functions of these modules one by one.

3.1.1 PON Processor

The input of the data is shown as the Fig. 3.3. It include CLK and data-input. CLK means the clock signal, and data-input means the data transmitted. In the upstream, the PON Processor is the first module. The format of the data transmitted is shown as Fig. 3.2. When the data is transmitted into FPGA, the packets will lose some preamble. So we must do something for calibration. We shift the data to the right place by adding two registers, and then we will get the right packets. The data after calibration is shown as Fig. 3.4. After calibration, the PON Processor will read and store the ONU-ID, the payload_length

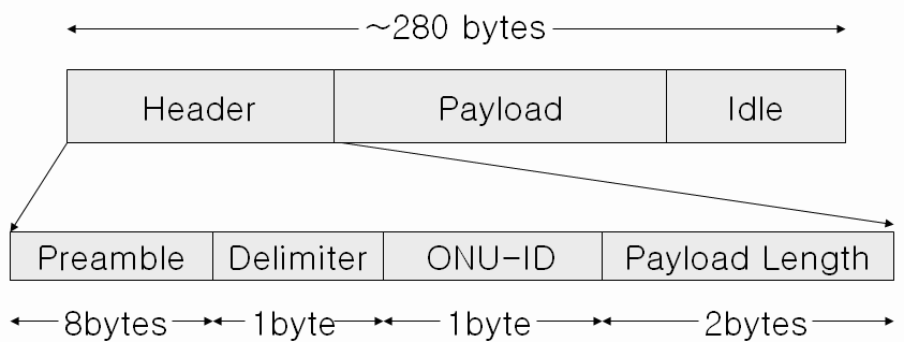

Figure 3.2: The format of the data transmitted.

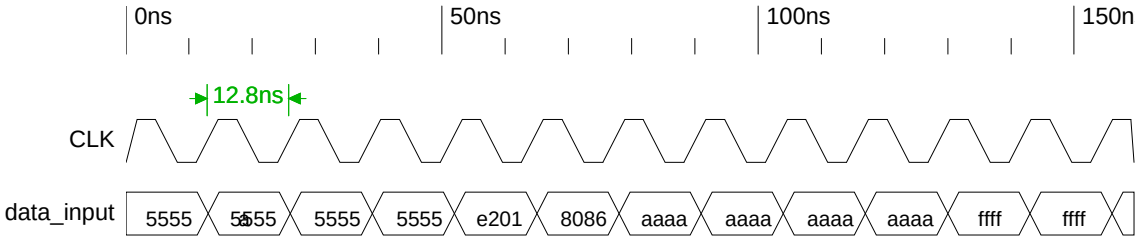

Figure 3.3: The input of the data.

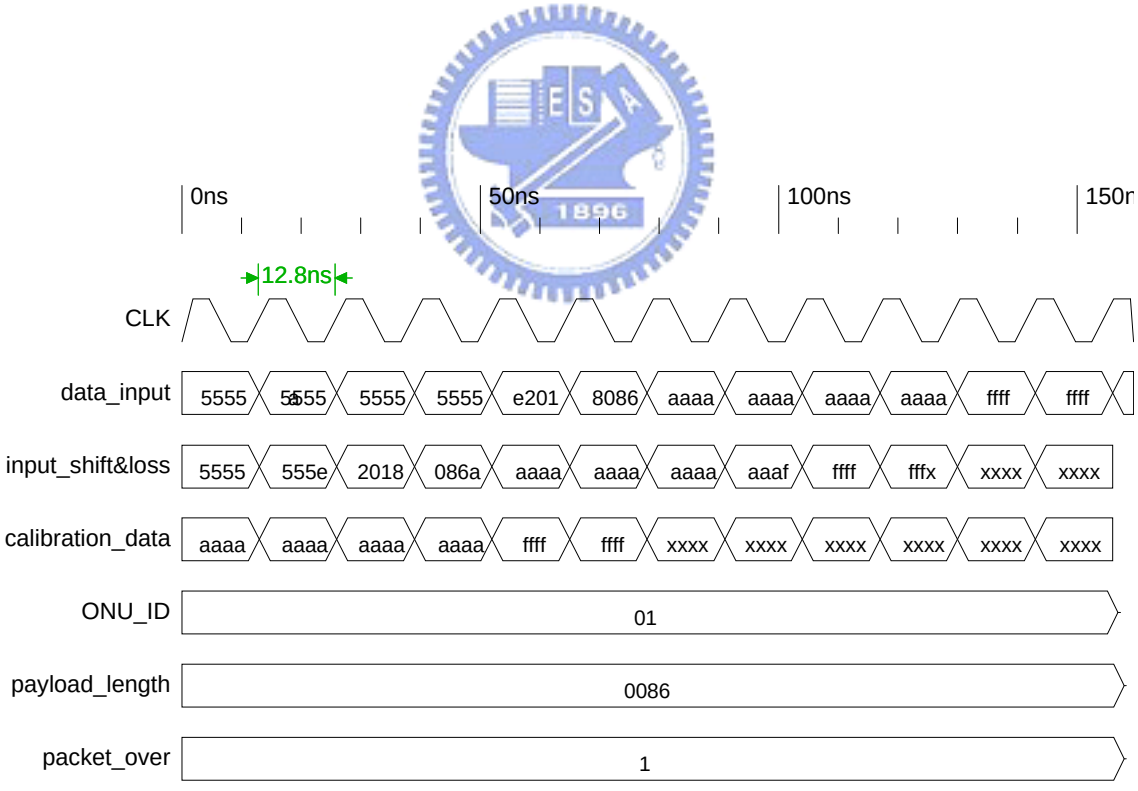

Figure 3.4: After calibration.

and the packet_over. Then the PON Processor send the ONU-ID, the payload_length and the packet_over to the next module.

3.1.2 ONU Buffer and Multiplexer

The input of the ONU Buffer module connects with the output of the PON Processor module, as shown in Fig. 3.5. There are four inputs because there are four different ONU-ID. So we put the data into the ONU Buffers depending on the ONU-ID.

The length of Ethernet packet is 46 to 1500 bytes. The length of the data transmitted is 280 bytes, so every Ethernet packet may be divided into 1–6 parts, and we set the most significant bit(MSB) of the payload-length of the last frame of each Ethernet packet be "1". So if we receive "1" at the MSB of the payload-length, it means the whole Ethernet packet we get. Then we must know which ONU Buffer has the most whole Ethernet packets, so we have four registers to count the number of the packets stored in the ONU Buffers. The data in the ONU Buffer has the most packets will be sent to next module first. If the number of the packets in the ONU Buffers equal to each other, the order of the data transmission is ONU Buffer1, ONU Buffer2, ONU Buffer3 and ONU Buffer4. The order of the data transmission is shown as Fig. 3.6.

And the other issue is when two ONUs send the frame simultaneously, what is the order of the frame? If the packet1 is divided into frameA and frameB and the packet2 is divided into frameC and frameD, which frame will be sent first? Which frame will be sent last? There is no particular answer. I just certainly know the frameB and frameD are sent later than frameA and frameC, as shown in Fig. 3.7. So when the MSB of the payload-length equals 1, we know this frame is the last frame of the Ethernet packet.

3.1.3 Gigabit Media Independent Interface(GMII)

The input of the GMII module connects with the output of the ONU Buffer and Multiplexer module, as shown in Fig. 3.8. In the GMII module, two functions will be complete. One is to transfer 16 bits data into 8 bits data. The other is when TXEN is on, the data is valid. When TXEN is off, the data is invalid. Another important thing is the output

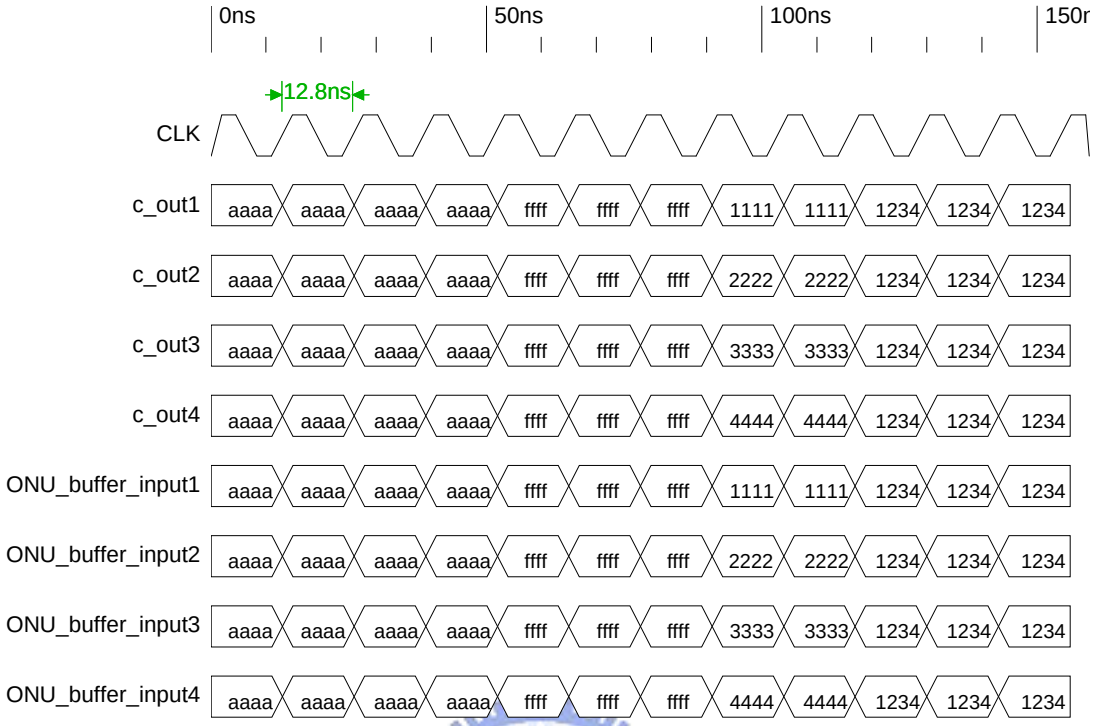

Figure 3.5: PON Processor output.

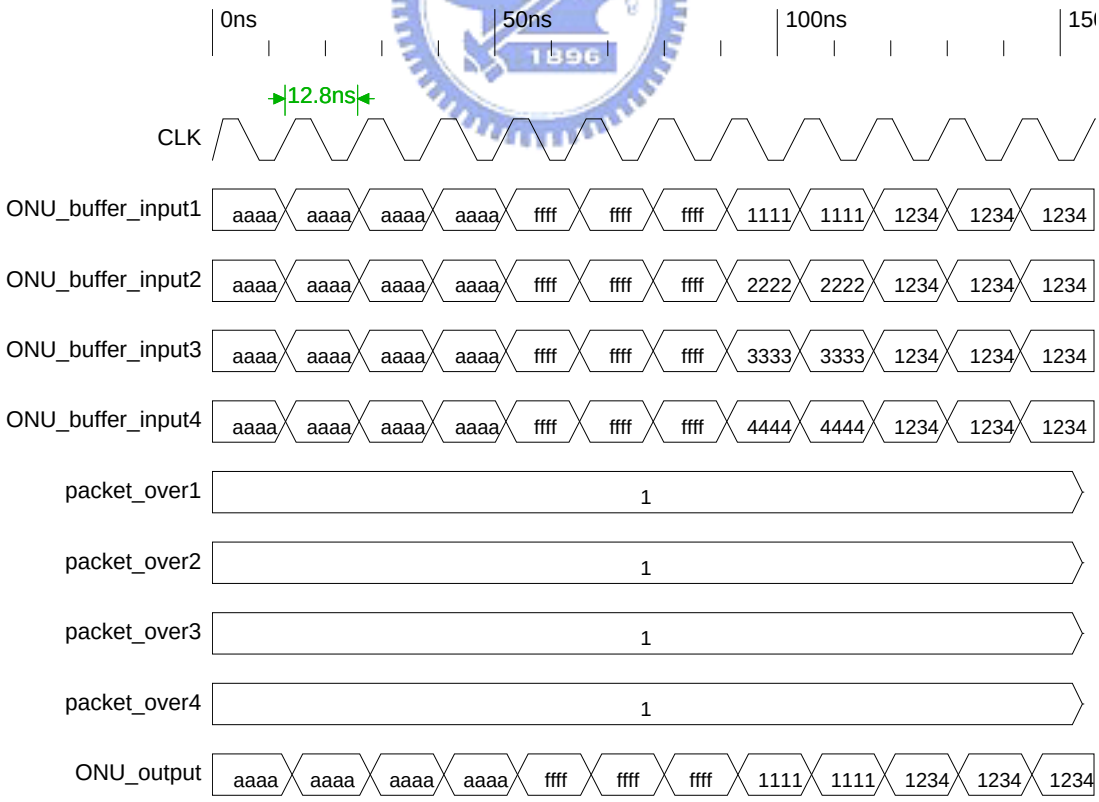

Figure 3.6: ONU_Buffer.

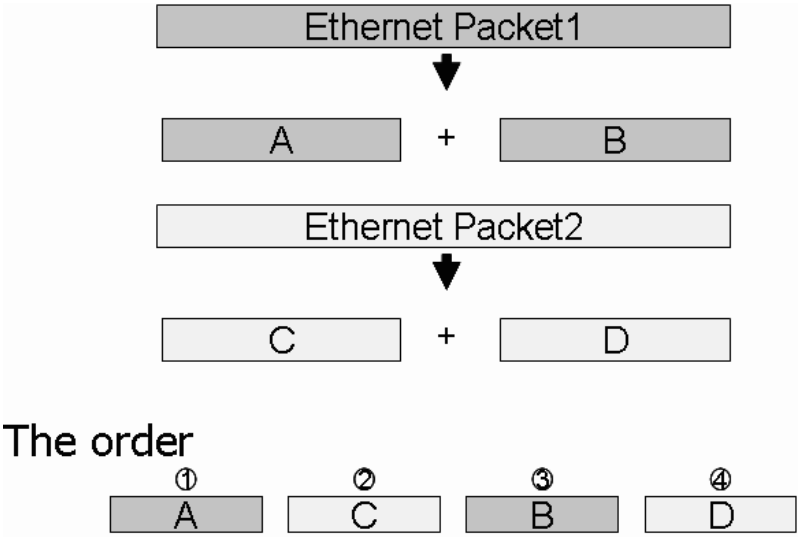

Figure 3.7: The order of the frame.

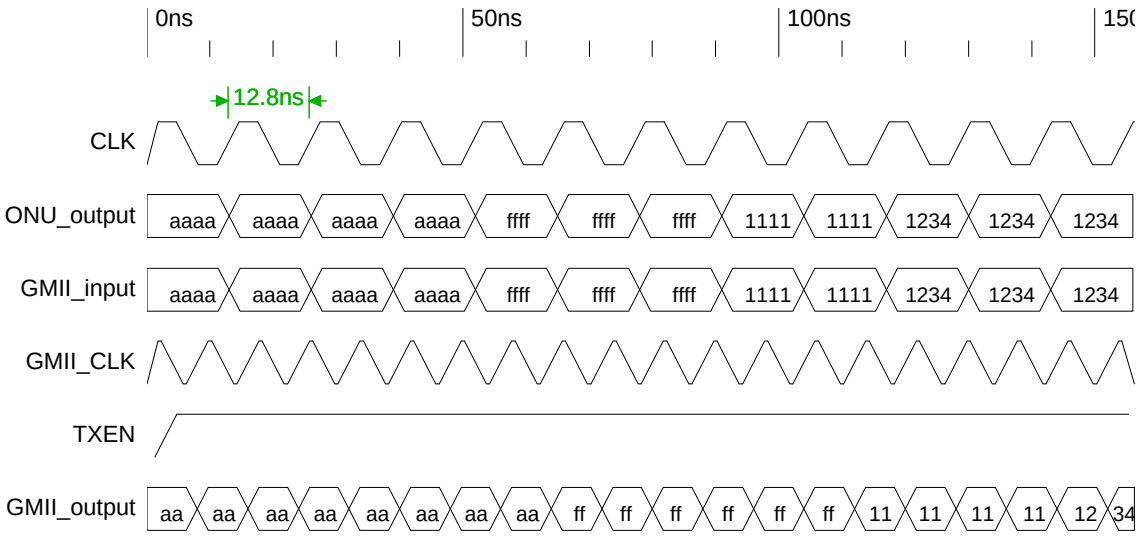

Figure 3.8: GMII module.

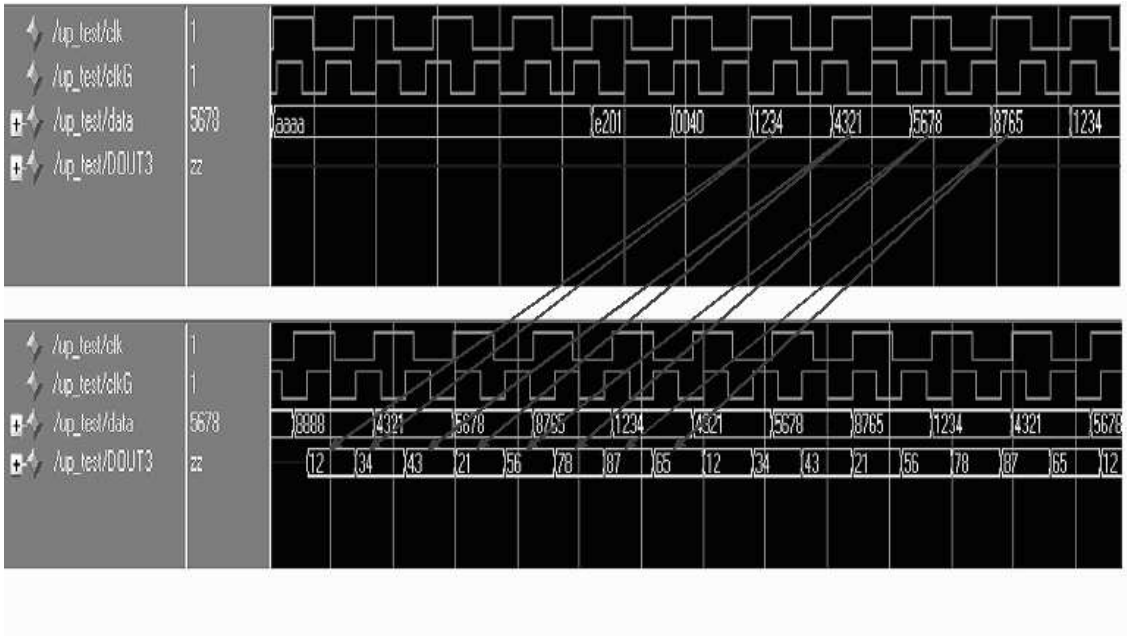

Figure 3.9: The final result of the upstream.

of the GMII module is triggered by the 125 Mbps. It is different from the module before GMII, just because it is the last module in the FPGA. After the data leave FPGA, it will be transmitted in 1 Gbps. Out of the GMII module, the data are transmitted 8 bits parallel, so we only need 125 Mbps transmission rate can achieve 1 Gbps. It is shown in Fig. 3.8. And the final result of the simulation in the computer is shown as Fig. 3.9.

3.2 The downstream

There are some modules in the downstream. It includes the GMII module, the Length module, the Data_buffer module and the Framer module. The same as the upstream, the data transmission rate between PHY and GMII is 125 Mbps, and the other is 77.76 Mbps. I will introduce the functions of these modules one by one.

3.2.1 GMII

The input data of the downstream is shown as Fig. 3.10 . The function of the GMII module in the downstream is almost the same as the GMII module in the upstream. The

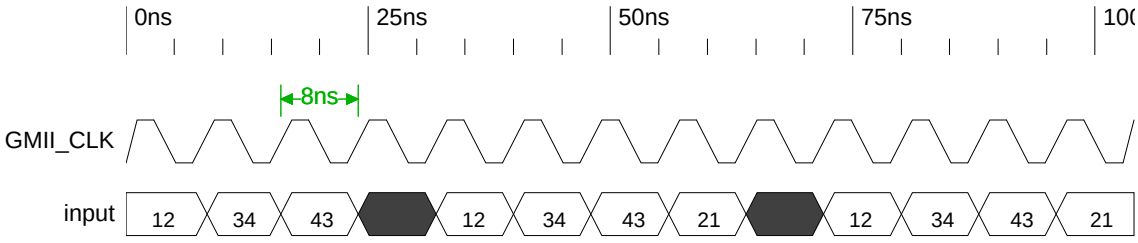

Figure 3.10: The input of the data.

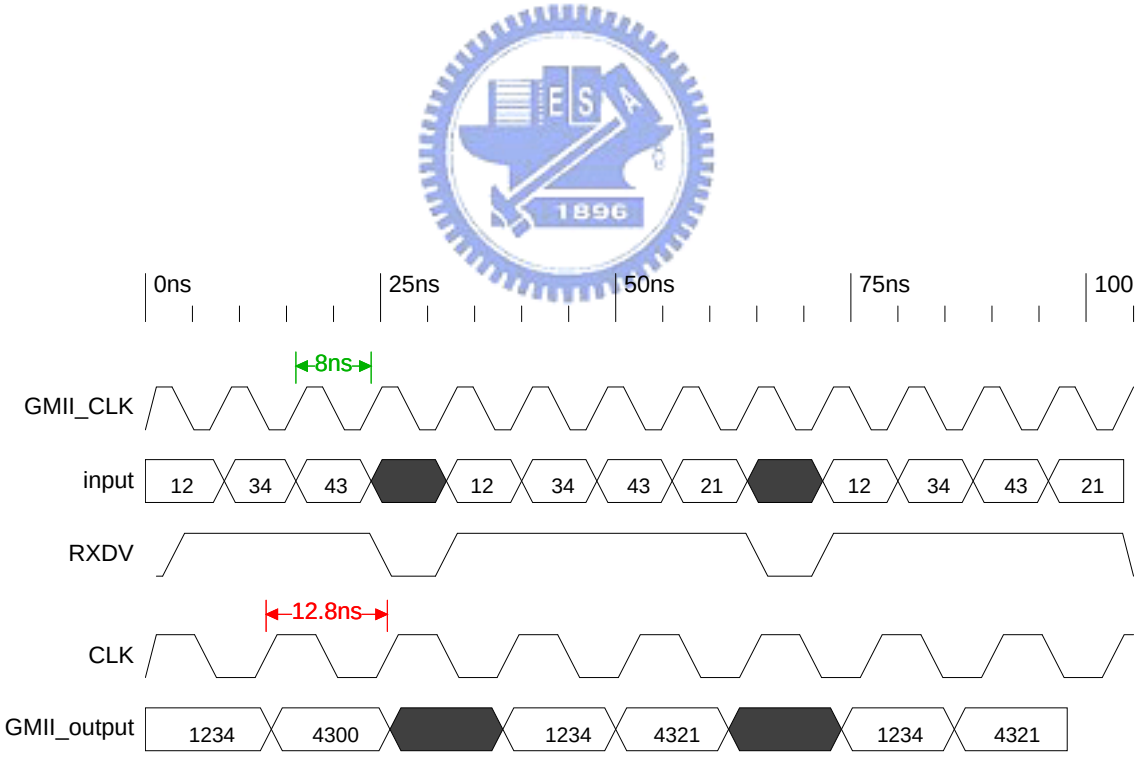

Figure 3.11: Download_GMII.

only difference is the upstream GMII module transfer 16 bits data into 8 bits data, but the downstream GMII module transfer the 8 bits data to 16 bits data. When the RXEN is on, the data is valid. When RXEN is off, the data is invalid.

3.2.2 Length

The data transmitted include the information of the payload length of the data in the upstream, but it does not in the downstream. When the data transmitted in the GMII module, it counts the length of the packet and store the result in the Length Module. So the Length module works like the memory. The modules behind the Length get the information of the length of the packets from the Length Module. The function of the Length module is shown as Fig. 3.12

3.2.3 Data_buffer

The input of the Data_buffer module connects with the output of the GMII module. It works like memory. The Data_buffer module store the data from the GMII module and then give it to the Framer module. The function is shown as Fig. 3.13

3.2.4 Framer

The last module in the downstream is the Framer module. The Framer module do something of adding the header before the data transmitted. The header include the preamble(6

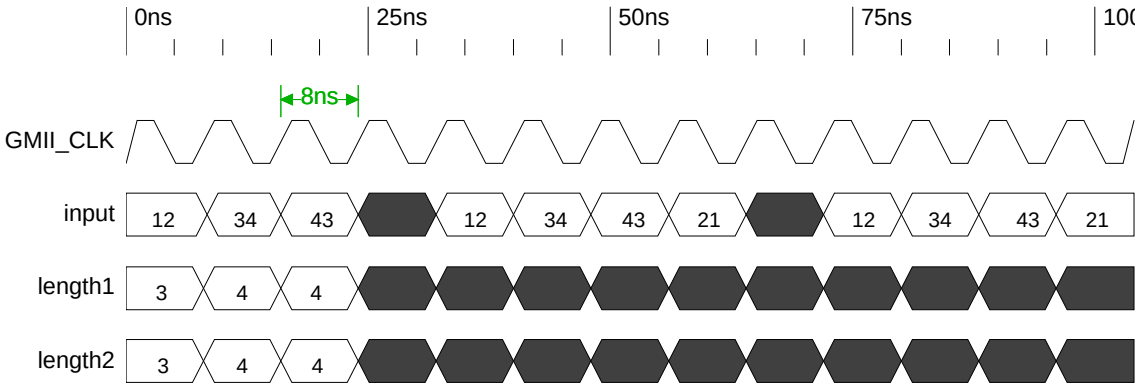

Figure 3.12: Length module.

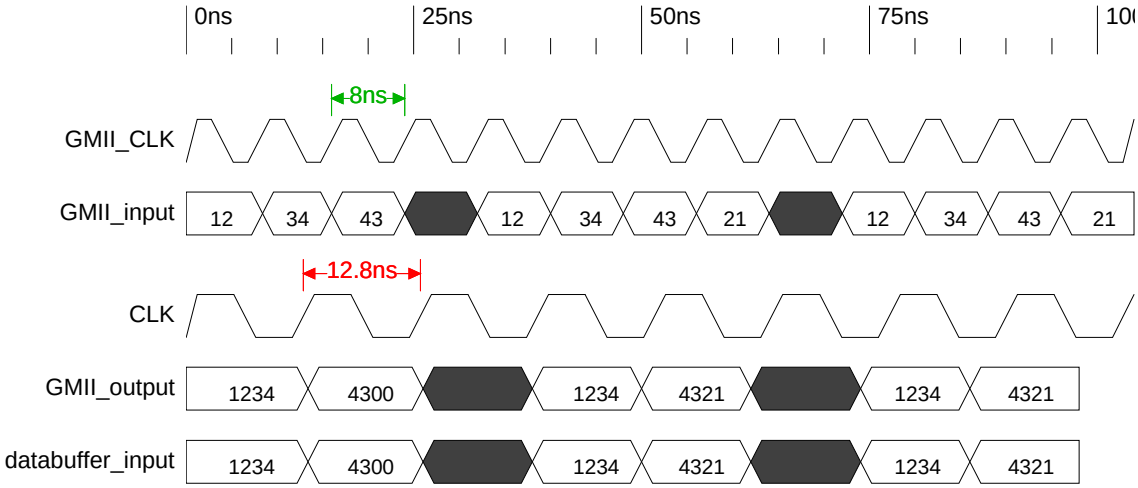

Figure 3.13: Data_buffer module.

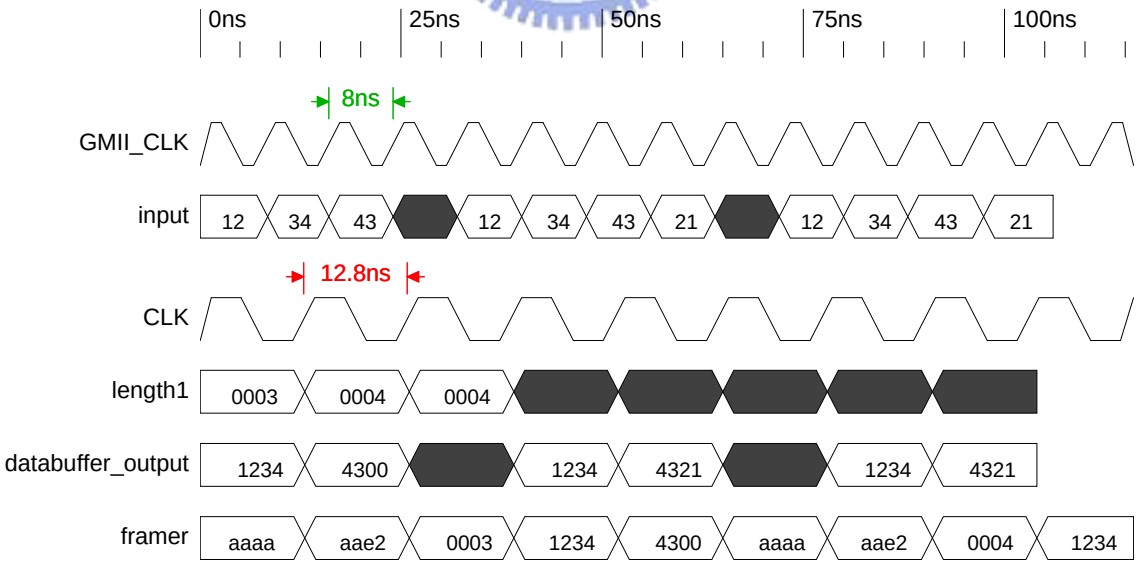

Figure 3.14: Framer module.

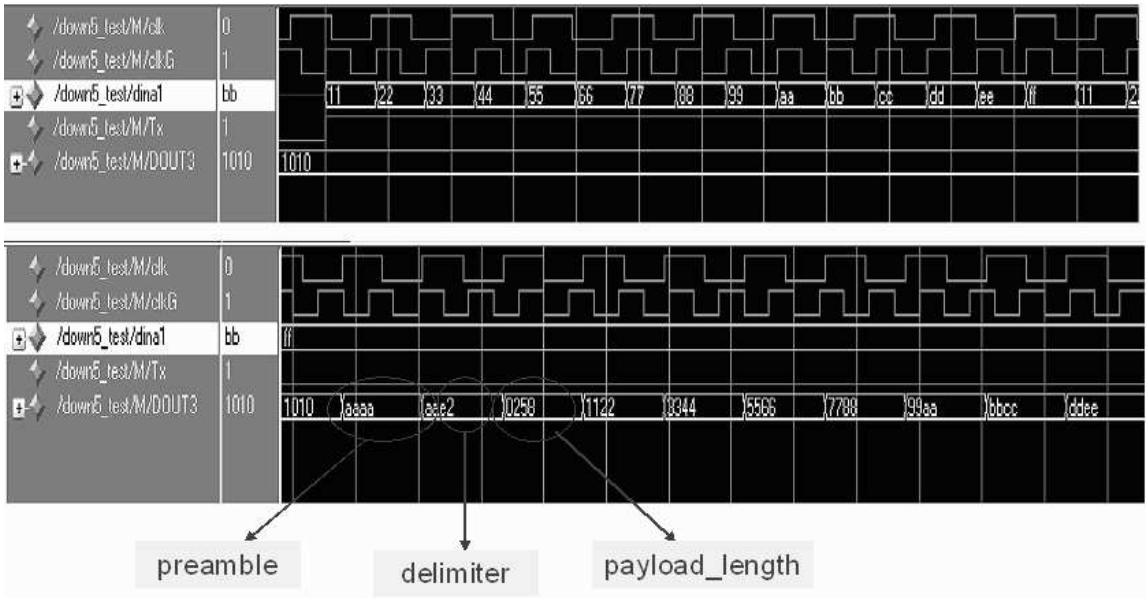

Figure 3.15: The final result of the downstream.

bytes), the delimiter(1 bytes) and the payload_length(2 bytes). The preamble include six consecutive “10101010”. The delimiter is “11100010”. The payload_length is from the Length module and the data transmitted is from the Data_buffer module. All the result is shown as Fig. 3.14. And the final result of the simulation in the computer is shown as Fig. 3.15.

Chapter 4

Devices on board

This chapter we will introduce the important devices on the board. These devices include field programmable gate array(FPGA), serializer and deserializer(SerDes) and Gigabit Media Independent Interface(GMII).

4.1 Field programmable gate array (FPGA)

A field programmable gate array is a semiconductor device containing programmable logic components called "logic blocks", and programmable interconnects [3]. Logic blocks can be programmed to perform the function of basic logic gates such as AND, OR, and XOR, or more complex combinational functions such as decoders or simple mathematical functions. In most FPGAs, the logic blocks also include memory elements, which may be simple flip-flops or more complete blocks of memories. A FPGA designed by Xilinx is shown as Fig. 4.1

A hierarchy of programmable interconnects allows logic blocks to be interconnected as needed by the system designer, somewhat like a one-chip programmable breadboard. Logic blocks and interconnects can be programmed by the customer/designer, after the FPGA is manufactured, to implement any logical function, hence the name "field-programmable".

FPGAs are usually slower than their application-specific integrated circuit (ASIC) counterparts, as they cannot handle as complex a design, and draw more power. But their advantages include a shorter time to market, ability to re-program in the field to fix

Figure 4.1: FPGA.

bugs, and lower non-recurring engineering costs. Vendors can sell cheaper, less flexible versions of their FPGAs which cannot be modified after the design is committed. The designs are developed on regular FPGAs and then migrated into a fixed version that more resembles an ASIC.

4.1.1 Architecture

The typical basic architecture consists of an array of configurable logic blocks and routing channels [3]. Multiple I/O pads may fit into the height of one row or the width of one column in the array. Generally, all the routing channels have the same width (number of wires).

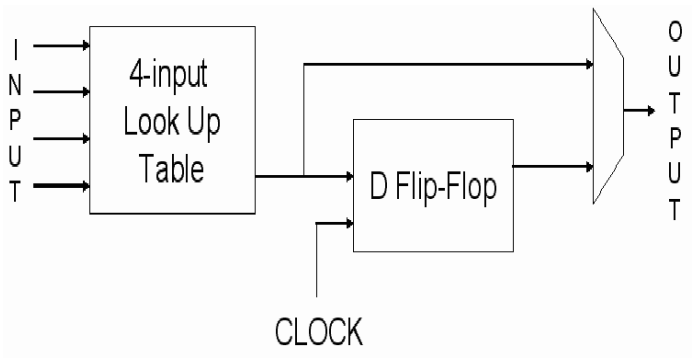

Figure 4.2: Logic block.

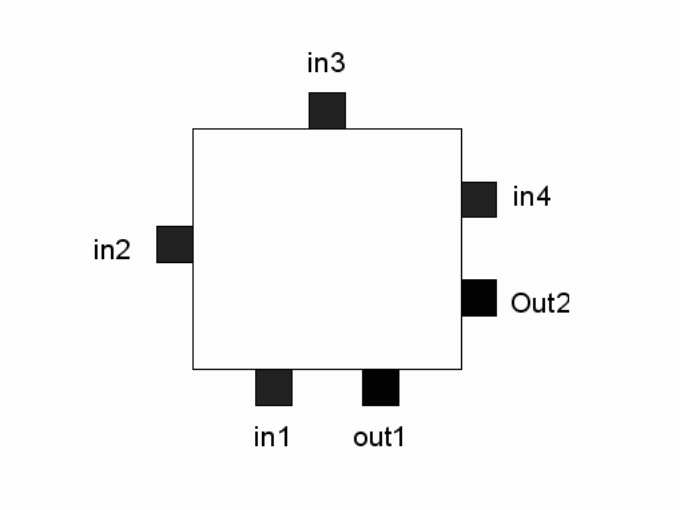

Figure 4.3: Logic block Pin location.

A classic FPGA logic block consists of a 4-input lookup table (LUT), and a flip-flop, as shown in Fig. 4.2. In recent years, manufacturers have started moving to 6-input LUTs in their high performance parts, claiming increased performance.

There is only one output, which can be either the registered or the unregistered LUT output. The logic block has four inputs for the LUT and a clock input. Since clock signals (and often other high-fanout signals) are normally routed via special-purpose dedicated routing networks in commercial FPGAs, they and other signals are separately managed. For this example architecture, the locations of the FPGA logic block pins are shown in Fig. 4.3.

4.1.2 Design process

The design process is shown as Fig. 4.4 [4].

- System design: Designer has to decide what portion of his functionality has to be implemented on FPGA and how to integrate that functionality with rest of the system.
- I/O assignment and analysis: I/O streams of FPGA are integrated with rest of the Printed Circuit Board.

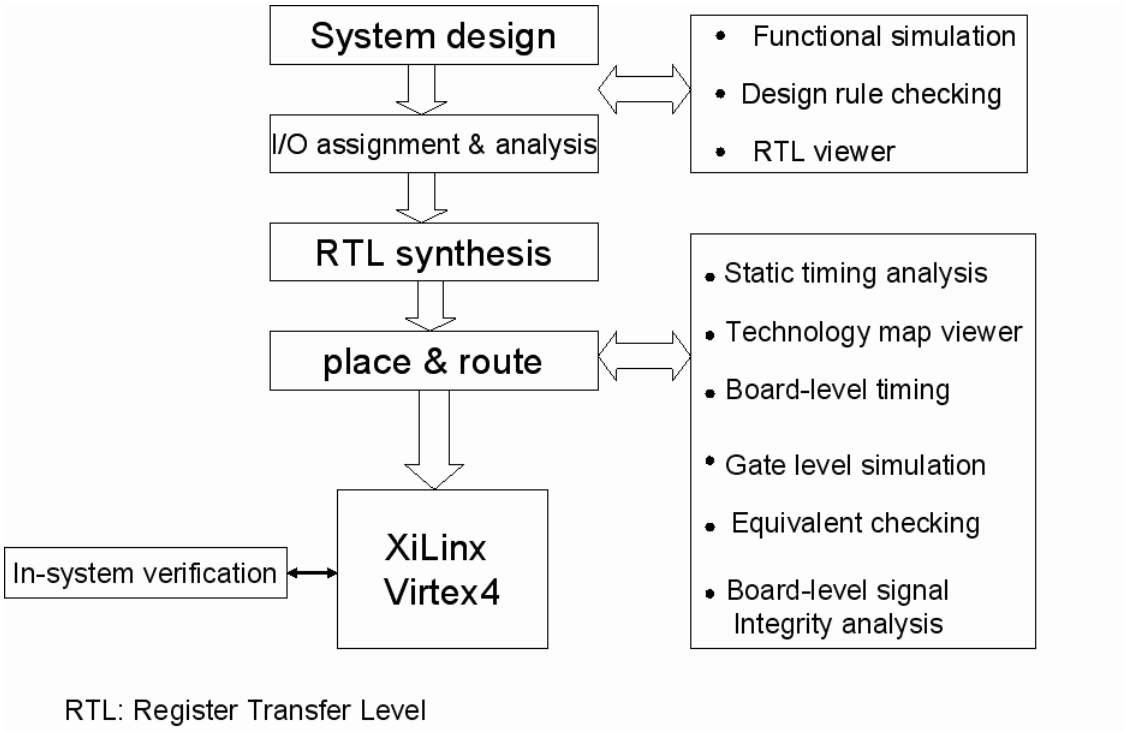

Figure 4.4: Design process.

- RTL synthesis: Once design has been defined CAD tools are used to implement the design on a given FPGA. Synthesis includes generic optimization, slack optimizations and power optimizations.

- Place and route: The place and route tool will then export a device configuration that can be used to target the FPGA device in actual hardware.

- In-system verification: Timing tools are used to determine maximum clock frequency of the design. The design is loading onto the target FPGA device and testing will be done in real environment.

4.2 Serializer and deserializer(SerDes)

A SerDes or serializer/deserializer is an integrated circuit (IC or chip) transceiver that converts parallel data to serial data and vice-versa [5]. The transmitter section is a serial-to-parallel converter, and the receiver section is a parallel-to-serial converter. Multiple

SerDes interfaces are often housed in a single package.

SerDes chips facilitate the transmission of parallel data between two points over serial streams, reducing the number of data paths and thus the number of connecting PINs or wires required. Most SerDes devices are capable of full-duplex operation, meaning that data conversion can take place in both directions simultaneously. SerDes chips are used in Gigabit Ethernet systems, wireless network routers, fiber optic communications systems, and storage applications. Specifications and speeds vary depending on the needs of the user and on the application. Some SerDes devices are capable of operating at speeds in excess of 10 Gbps.

4.3 Gigabit Media Independent Interface(GMII)

GMII is the abbreviation of Gigabit Media Independent Interface [6]. GMII interfaces are backward compatible with the Media Independent Interface specification.

The GMII specification is used to define the interface between the Media Access Control (MAC) device and the physical layer (PHY). The interface defines speeds up to 1000 Mbps, implemented using an eight bit data interface clocked at 125 MHz. It can also operate on fall-back speeds of 10/100 Mbps as the MII specification.

4.3.1 Transmitter

- GTXCLK: clock signal for gigabit TXD (125 MHz).
- TXCLK: clock signal for 10/100 Mbps signals.
- TXD[7:0]: data to be transmitted.
- TXEN: transmitter enable
- TXER: transmitter error (used to corrupt a packet)

There are two clocks, depending on whether the PHY is operating at gigabit or 10/100 Mbps. For gigabit speeds, the GTXCLK is supplied to the PHY and the TXD, TXEN, TXER signals are synchronized to this. Otherwise for 10/100 Mbps the TXCLK (supplied

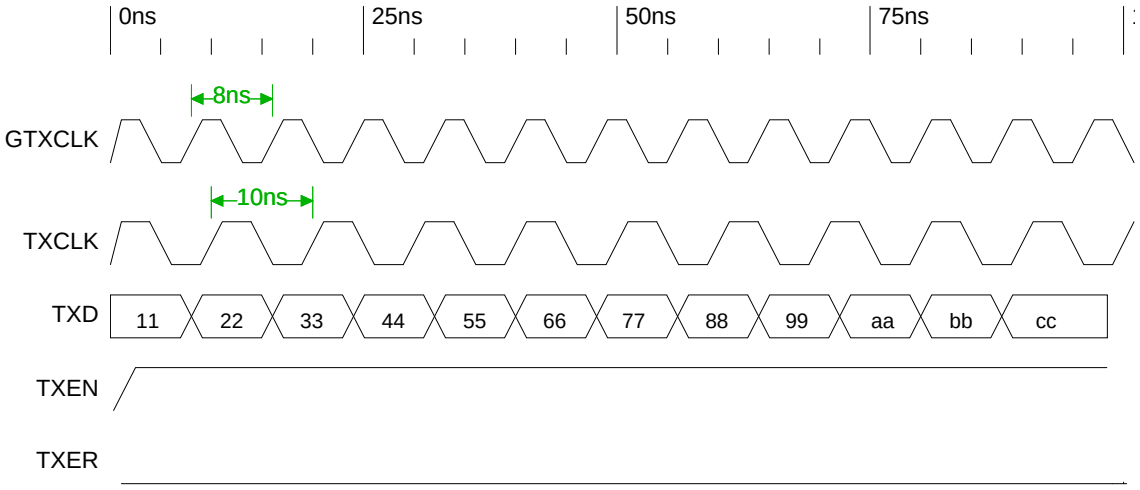

Figure 4.5: Signal timing of TX.

by PHY) is used for synchronizing those signals. This operates at either 25 Mbps for 100 Mbps or 2.5 Mbps for 10 Mbps connections. The receiver clock is much simpler, with only one clock, which is recovered from the incoming data. Hence the GTXCLK and RXCLK are not coherent. The timing of all signals of the transmitter is shown as Fig. 4.5

4.3.2 Receiver

- RXCLK: received clock signal (recovered from incoming received data).
- RXD[7:0]: received data.
- RXDV: signifies data received is valid.

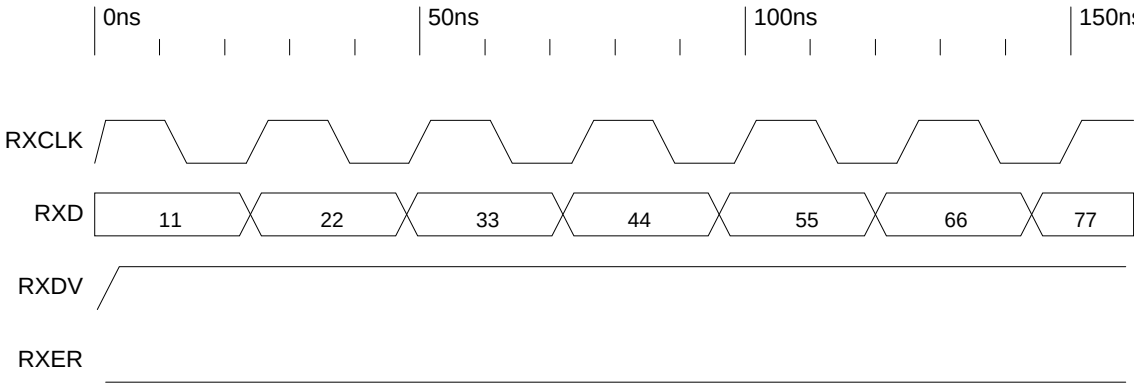

Figure 4.6: Signal timing of RX.

- RXER: signifies data received has errors.

The timing of all signals of the receiver is shown as Fig. 4.6

Chapter 5

Verification in hardware

5.1 Upstream

In this chapter, we will show the result of the on-board testing. And the testing board is shown as Fig. 5.1. Then we will show the result of the verification on board. The upstream result is shown as Fig. 5.2,Fig. 5.3 and Fig 5.4. The input includes four Ethernet packets. The first and the second Ethernet packets are both less than 280 bytes and the third and the fourth Ethernet packet are over 280 bytes. So the third and the fourth Ethernet packet will be divided into two frames. These four Ethernet packet are sent from different ONU. So there are six frames will be transmitted. These six frames are transmitted every 2.048 us. But the result does not show these four Ethernet packet are separated. The first white part is the data sent from ONU1 and the second white part is the data sent from ONU2. The third white part and the fourth white part is too near to distinguish by our eyes. This is because the third and the fourth Ethernet packet both are divided into two frame. We assume that the third Ethernet packet is divided into frameA and frameB and the fourth Ethernet packet is divided into frameC and frameD. The transmission order is frameA, frameC, frameB and the frameD, as shown in Fig. 5.6 . So the frameC arrive the ONU buffer earlier than the framerB. The frameA are not sent until the frameB is coming. So when the frameA and the frameB get together and is sent to the next stage, the frameD is coming. The frameB and the frameD get together immediately and is sent to the next stage. So the third Ethernet packet is so close to the fourth Ethernet packet.

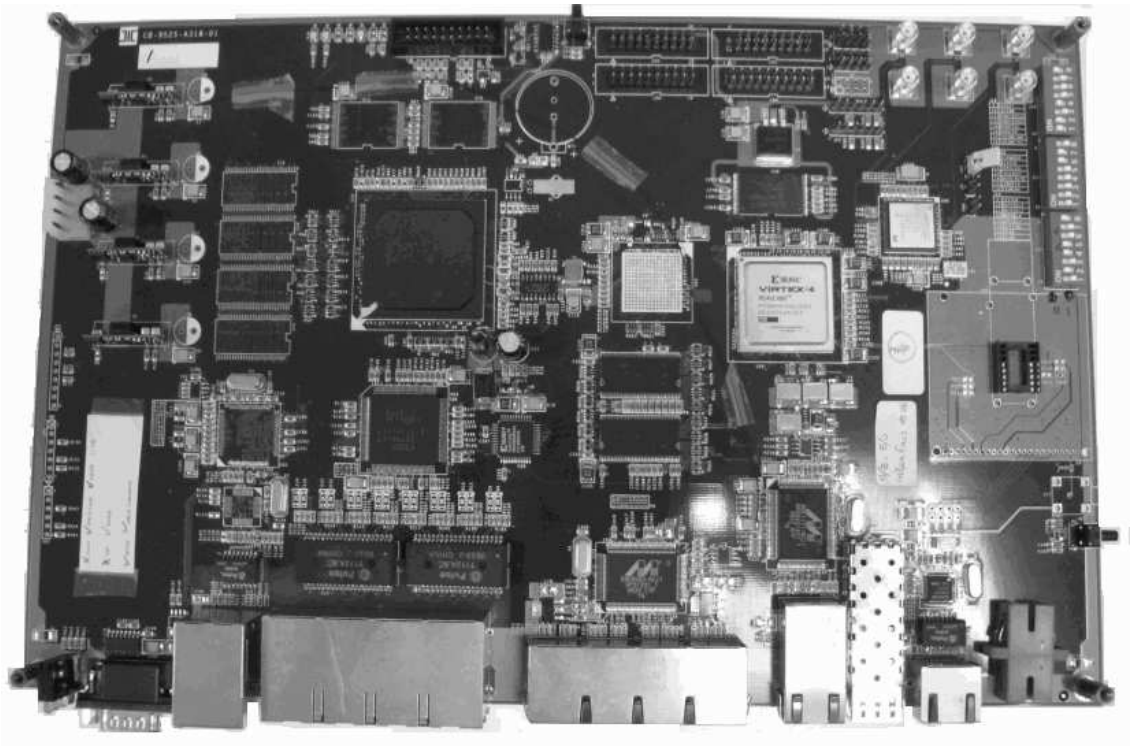

Figure 5.1: OLT testing board.

5.2 Downstream

Then we will show the downstream result of the on-board testing. The result is shown as Fig. 5.5. We set the time scale of the first line of Fig. 5.5 much larger than the other. So we just see the white and the shadow. The details of the data is shown as from the second line to the last line.“AAAA AAE2 00BB” of second line means the header. The header include 3 bytes PSYNC, 1 byte delimiter and 2 bytes payload-length. After the header, all the data is Ethernet packet. This Ethernet packet totally include 187 bytes.

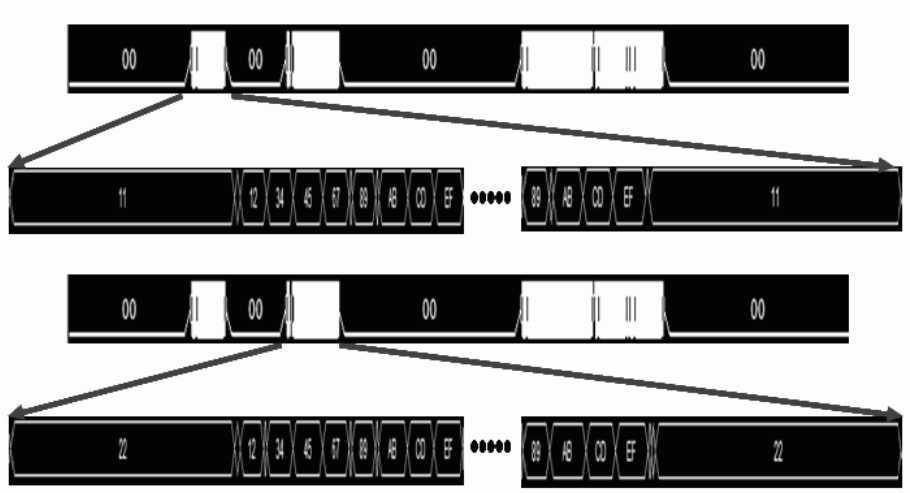

Figure 5.2: Upstream result of on-board testing (a).

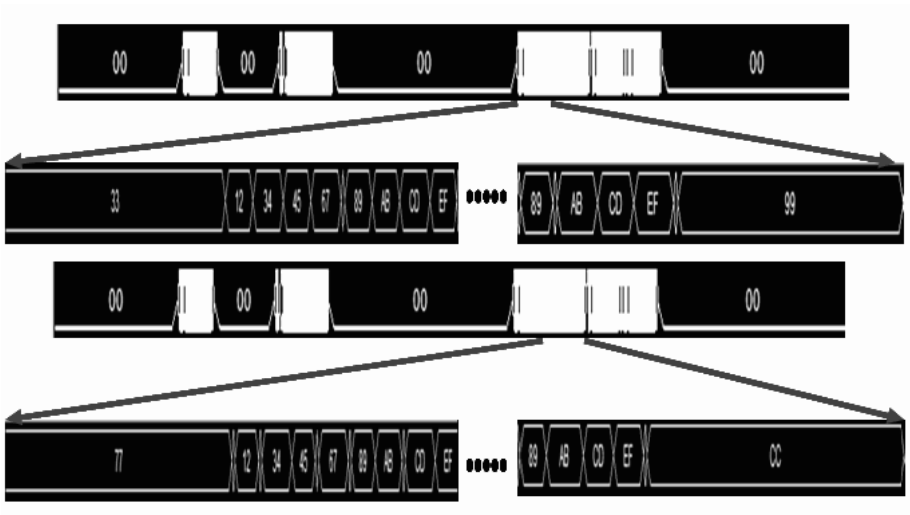

Figure 5.3: Upstream result of on-board testing (b).

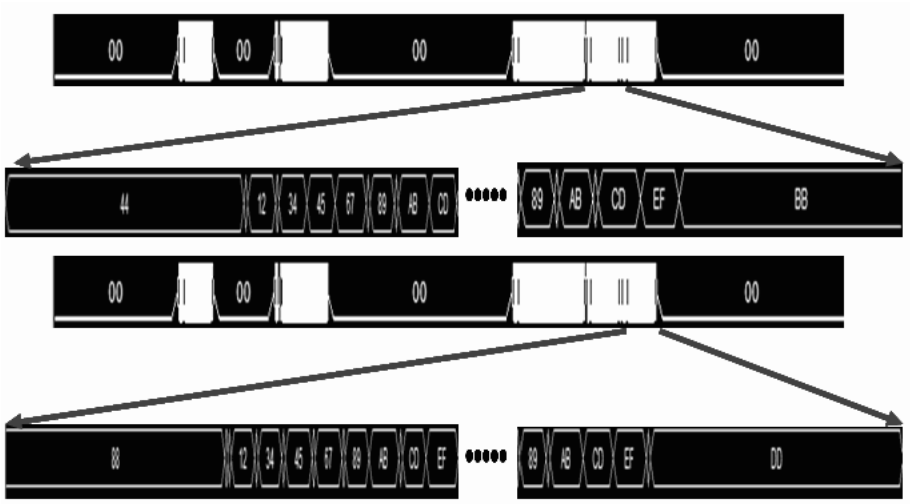

Figure 5.4: Upstream result of on-board testing (c).

Figure 5.5: Downstream result of on-board testing.

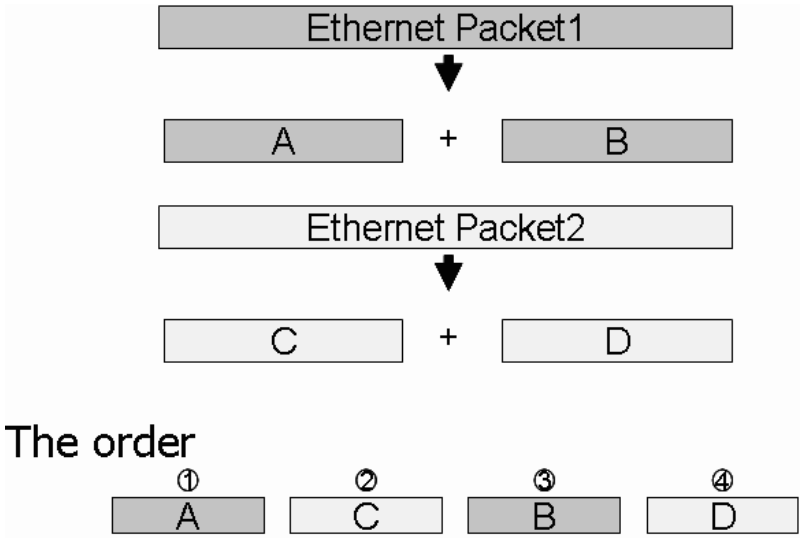

Figure 5.6: The transmission order.

Chapter 6

Conclusions

First, we use “Xilinx ISE” to complete all of the functions of the OLT. Then we use “Synplicity” to synthesize the circuit. After that we use “Modelsim” to complete the simulation. All above is the simulation in software. When we complete the simulation in the computer, we must load the program to the testing board to be verified. There are two parts to be verified including the upstream and the downstream. In the upstream, we just verify the loopback circuit, and the final result is exact. In the downstream, the input data is produced from “Ethview”. Then the data will be through GMII, FPGA and the SerDes. The final result is exact too. We will do something about connecting the OLT with the ONU. It is a loop from the data are sent from the different ONUs to the OLT and then the ack are sent back from the OLT to each ONU.

Bibliography

[1] Dawid Nowak and John Murphy, "FTTH: The Overview of Existing Technologies," Opto-Ireland, 2005 .

[2] "Ethernet Passive Optical Network EPON A Tutorial." Reproduced with permission of the Metro Ethernet Forum, The Metro Ethernet Forum, 2005. http://metroethernetforum.org/PDFs/EFMA/EFM%20PON%20Tutorial%20v2.pdf

[3] http://en.wikipedia.org/wiki/FPGA

[4] http://www.altera.com.cn/literature/manual/intro_to_quartus2.pdf

[5] http://searchstorage.techtarget.com/sDefinition/0,290660,sid5_gci1006456,00.html

[6] http://en.wikipedia.org/wiki/GMII

[7] Michael P. McGarry, Martin Maier and Martin Reisslein, "ETHERNET PONs: A SURVEY OF DYNAMIC BANDWIDTH ALLOCATION(DBA) ALGORITHM," IEEE Optical Communication, 2004.

[8] A. Hadjiantonis, S. Sherif, A. Khalil, T. Rahman, G. Ellinas, M. F. Arend, and M. A. Ali, "A NOVEL DECENTRALIZED ETHERNET-BASED PASSIVE OPTICAL NETWORK ARCHITECTURE," IEEE Communications Society, 2004.

[9] Bill Zakowski and Amedia Networks, "Passive vs. Active Optical Networking," BROADBAND PROPERTIES, 2004.

[10] Jing Xie, Shengming Jiang and Yuming Jiang, "A DYNAMIC BANDWIDTH ALLOCATION SCHEME FOR DIFFERENTIATED SERVICES IN EPONs," IEEE Oplical Communications, 2004.