[1,*]Guangxuan Chen, Qiang Liu

[2]Yuanyuan Huo

A Proposed FPGA Design and Implementation of Quintuple Service Identification Module for 10G EPON ONU

Journal of Electrical Systems

Abstract: - As one of the mainstream technical solutions for optical access network, 10G EPON is increasingly favored by telecommunications suppliers. However, only as a simple two-layer bearer system, 10G EPON has defects such as transparent data transmission and lack of concern for protocols and services. As a result, the 10G EPON system is not efficient in carrying new services such as web video services, P2P services, VoIP services, instant messaging services, streaming media services, and online game services, and the bandwidth is seriously wasted. For this reason, this paper designs a set of 10G EPON ONU implementation scheme that supports service identification, and then implements FPGA implementation on the core modules of the DPI part of the scheme, such as quintuple identification, port identification, uplink packet feature extraction cache, data FIFO and other sub-modules. Finally, a simulation test is carried out on the designed scheme. The results show that the accuracy rate and recall rate of the two main indicators of the services identification function have reached more than 90% and 80% respectively, and the expected design goal has been completed.

Keywords: 10G EPON; Service Identification; DPI; Quintuple Identification; Service Classification.

I. INTRODUCTION

The access network is providing users with more and more services. In addition to traditional services as telephone communication, web browsing, e-mail, there are also a series of high-speed, high-quality services such as P2P services, web video services, instant messaging services, VoIP services, streaming media services, online game business, online shopping, e-commerce, an so on. [1]. The new business traffic represented by P2P downloads has accounted for more than 70% of the entire Internet traffic. P2P services and various video services consume a large amount of bandwidth, which makes operators have to expand the backbone network [2]. However, the operator's income has not increased with the expansion of the network, and it has become necessary for the operator to change the way of network management and to introduce intelligent technology to tap the potential value of the network. It is the general trend to identify various services in the network and perform refined management of the services [3].

Based on this consideration, this paper proposes a 10G EPON ONU implementation scheme that supports service identification, and designs and implements it through FPGA. The feature of this solution is that it can realize 10Gbps high-throughput service identification on the hardware. In the field of business identification, identification methods are usually divided into software identification and hardware identification [4]. It is more convenient to implement in software, and there are many related algorithms. There are also many existing service identification devices implemented by software, such as "iSIE, Internet Service Identification Engine". However, the software identification device has a disadvantage, that is, with the gradual increase of the user-side bandwidth in the network, it is difficult for the software service identification device to achieve a processing speed of more than 1 Gbps. Therefore, the application occasion of the software service identification device is greatly restricted. Although some scholars have proposed to implement business identification on hardware, few can achieve higher throughput on hardware. Therefore, this paper designs a high-throughput service identification scheme implemented by hardware, and the throughput can reach 10Gbps.

*Corresponding author: Guangxuan Chen

[1] Department of Computer and information Security, Zhejiang Police College, Hangzhou 310054, China

[2] College of Geosciences and Engineering, North China University of Water Resources and Electric Power, Zhengzhou 450046, China

II. RELATED WORK

Wherever Times is specified, Times Roman or Times New Roman may be used. If neither is available on your word processor, please use the font closest in appearance to Times. Avoid using bit-mapped fonts if possible. True-Type 1 or Open Type fonts are preferred. Please embed symbol fonts, as well, for math, etc.

There are two existing service identification methods, one is packet-by-packet identification, and the other is flow-by-flow identification [5]. For packet-by-packet identification, since each packet has characteristics, the packets can be identified one by one by matching the characteristics. Flow-by-flow identification refers to identifying data packets according to the type of flow [6]. At present, the following methods are widely used in the field of business identification.

(1) Port number based identification method: This is a method of packet-by-packet detection. Although many services use dynamic random port numbers for communication, some common services can still be identified through the identification of fixed port numbers. For example, the port number for FTP file download is 21, the port number for DNS service is 53, and the SMTP port number for mail service is 25[6].

(2) quintuple based identification method: <source IP address, destination IP address, protocol type, source port number, destination port number> is called a quintuple, and a flow with the same quintuple is called a quintuple group flow, hereinafter referred to as flow. Since the packets of the same flow have the same service type, other data packets of this flow can be identified by identifying the first few data packets of the flow and the quintuple extracted from the data packets.

(3) Deep Packet Inspection (DPI): For services with characteristic words in the packet content, the packet can be identified by keyword matching. For services (such as QQ application) that have keywords in all data packets of a flow, they can be identified by packet-by-packet. For flows with keywords only in some data packets (such as BT), the quintuple of the packets can be extracted after matching the packets with keywords, and then identify the other packets according to the quintuple[7].

(4) Deep Flow Inspection (DFI): For encrypted P2P services and unknown P2P services, we can use machine learning to identify them by analyzing their traffic statistics and traffic behavior characteristics. Traffic statistics characteristics include packet length and packet interval in the service flow [7]; traffic behavior characteristics refer to the number of connections initiated at the same time, and whether TCP and UDP are used at the same time. If the traffic characteristics or behavior characteristics used continue to be collected until the end of the flow, then the identification of the flow is not real-time. If the characteristics of the flow can be identified only by using the first several data packets of the flow, and the subsequent packets in the flow can be identified and controlled after the flow is identified, then the identification of the flow is considered to be real-time.

III. DESIGN SOLUTIONS OF 10G EPON ONU THAT SUPPORT SERVICE IDENTIFICATION

A. Schematic Diagram of 10G EPON System

Figure 1 shows a schematic diagram of a 10G EPON system. Adding the service identification function to the 10G EPON system requires adding service identification modules on both the OLT and ONU sides to identify and control the upstream and downstream data at the same time. Each ONU only identifies and controls its upstream data, and the OLT identifies and controls its downstream data at each PON port. The OLT assigns LLID to all downstream frames, and uses 1 bit to identify whether the frame is recognized by the OLT; the ONU assigns LLID to all upstream frames, and uses 1 bit to identify whether the frame is identified by the ONU. When the rule base for service identification needs to be upgraded, the new rule base is first configured to the OLT through the CPU, and then the OLT broadcasts the OAM frame to each ONU. After the ONU receives the OAM frame, it configures the service identification module of the ONU through the CPU.

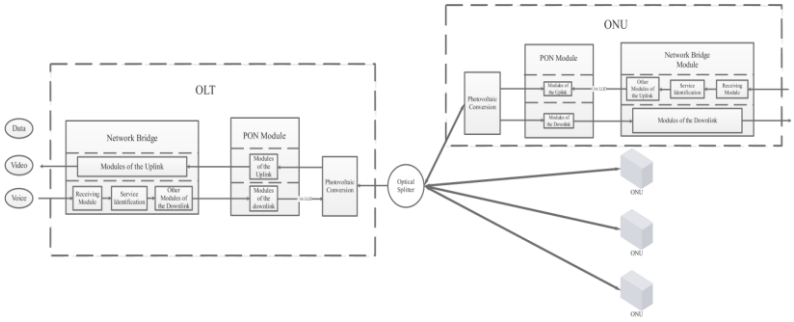

Figure 1. Schematic Diagram of a 10G EPON System

The service identification modules in ONU and OLT are of the same design. In order to support the identification of mainstream Internet services, the service identification modules use a combination of Deep Packet Inspection (DPI) and Deep Flow Inspection (DFI). DFI technology analyzes the characteristics of the flow to realize the identification of the flow, so it is necessary to store the flow characteristics, and identify the characteristics after the collection is finished. For a throughput of 10Gbps, there are a lot of streams to be processed. It is too costly to let all the streams identified through DFI. Therefore, it is more economical to use DFI to identify services that cannot be identified by DPI, that is, DPI and DFI are identified in a serial manner.

Here, the DFI adopts a time-effective method, that is, the service type of the flow is judged by the packet length and direction at the message exchange phase of a flow, so that the classification result is given before the data part of the flow is processed. Bidirectional packet features are used in the identification process, that is, whether it is ONU or OLT, when performing DFI identification, (source address A+source port B+destination address C+destination port D+protocol E) and (source address C+source port D+destination Address A + destination port B + protocol E) are regarded as the same flow. In order to avoid entering too much redundant information, only the packet features not recognized by DPI are allowed to enter the DFI module.

The ONU only identifies the upstream data, and the OLT only identifies the downstream data. Therefore, if the ONU wants to use the unrecognized packet features in the downstream, it needs the OLT to inform whether the packet has been identified; similarly, if the OLT wants to use the unidentified packet features in the upstream, it needs the ONU to inform whether the packet has been identified. Therefore, in the service identification system, the communication between the ONU and the OLT uses only a 1-bit LLID, so that the DFI modules of the ONU and the OLT can perform the identification efficiently without dealing with excessive redundancy (the DPI-identified packet characteristics)

B. Design Scheme of 10G EPON ONU that Supports Service Identification

Figure 2 is the block diagram of the 10G EPON ONU that supports service identification. The MPCP processing module, filtering and packet feature extraction modules are all related to the service identification function.

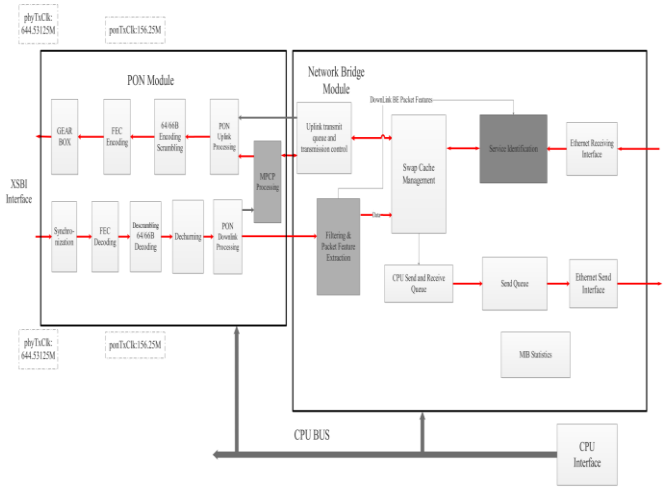

Figure 2. Design diagram of 10G EPON ONU that supports service identification

For the upstream direction, in the bridge module of the ONU, the service identification module mainly completes the analysis of the frame content and completes the service identification. The MPCP module is responsible for the processing of the MPCP protocol, including the processing of mpcp frames such as registration, report, authorization, etc., as well as the upstream transmission control. It is also responsible for forwarding Ethernet data frames for the upstream and marking the LLID for the upstream frames.

For the downstream direction, in the bridge module of the ONU, the filtering and packet feature extraction modules complete the identification of the LLID; the filtered frame configured by the CPU is judged whether the frame has been identified by the OLT of the opposite end; the packet length feature of the frame that unidentified by the opposite OLT will be extracted and transmitted to the service identification module. In this way, all frame data pass normally.

C. FPGA Design Solution of 10G EPON ONU

Figure 3 shows the FPGA block diagram of 10G EPON ONU service identification, which is mainly composed of two parts: DPI part and DFI part. The DPI part includes a port identification module, a keyword identification module, a quintuple identification module, etc.; the DFI part includes a DFI preprocessing module, a DFI classifier, and a quintuple identification module.

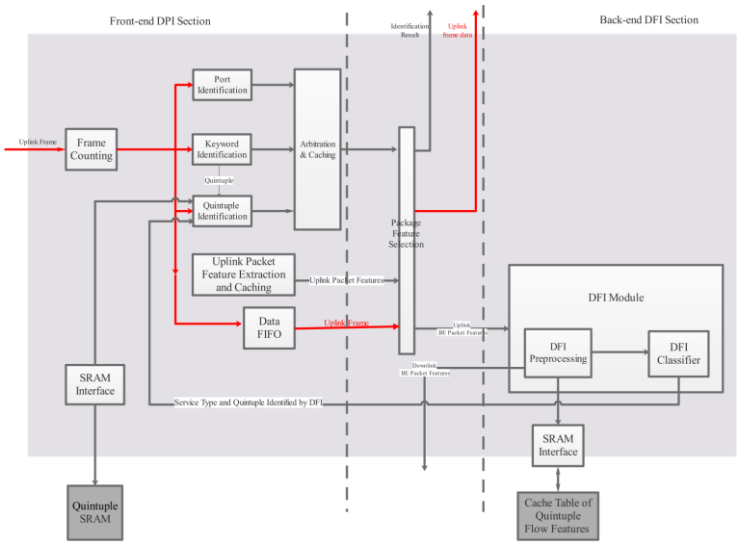

Figure 3. FPGA Block Diagram of 10G EPON ONU Service Identification

The DPI part uses keyword identifiers to identify non-encrypted services, and the DFI part uses packet feature vectors to classify and identify encrypted P2P services through a classifier. The identified quintuple included in the quintuple identification module includes services identified by the keyword identification module and services identified through DFI. This solution innovatively combines DPI and DFI technologies to exert the advantages of both and expand the scope of identifiable services. Here, the DPI and the DFI adopt a concatenated structure, and the DFI only recognizes the service data that is not recognized by the DPI, which can reduce the data throughput processed by the DFI.

At the front end, the system resources occupied by the serial and parallel combination of the port identification module, the keyword identification module and the quintuple identification module are similar, and the total delay is also on an order of magnitude. If the parallel method is adopted, according to the comprehensive judgment of the subsequent identification results, it can be determined which service belongs to with a high accuracy rate and the delay is the lowest. So, it was finally decided to combine these three modules with similar behavior in parallel manner (fixed or non-fixed-position matching of frame data).

Since there are IP fragmented packets in the network environment, and these fragmented packets have no port information, the port identification module and the quintuple identification module will fail and output wrong results. Here, only the keyword recognition module works properly. Based on the recognition that the fragmented packets are all P2P service data packets, the identification of the fragmented data packets is carried out in the port

identification module. When the port identification module identifies the fragment, the identification result of the P2P data packet is directly marked instead of port comparison.

The arbitration cache module receives the identification results of the port identification module, the keyword identification module and the quintuple identification module, performs arbitration on the three identification results of the same frame to obtain a comprehensive result, and temporarily stores the comprehensive result in the cache. The identification result is output only when the read request is cached. The identification arbitration module can configure the priority of the results of keyword identification, the quintuple identification and the port identification according to the CPU.

The upstream packet feature extraction and cache module is responsible for extracting the packet length feature of the frame data, which is performed simultaneously with the three matching modules. After the packet feature of the frame is extracted, it is stored in the buffer, and the corresponding packet feature is output when the frame header of the frame data is output from the data FIFO.

The frame data passes through the data FIFO with a delay, and each frame is delayed for a maximum processing time of 195 clock cycles to ensure that the frame identification result has been given when the frame header of each frame comes out. The frame identification results and data output, the packet characteristics of the BE packet are given to the subsequent DFI module.

The packet feature selection module is responsible for selecting the packet features of all upstream data packets. All identification results are output from the service identification system in a one-to-one correspondence with frames. If the identification result of a certain packet is BE, then send this packet feature to the subsequent DFI module for continue identification, and other non-BE packet features are directly discarded.

In the DFI part, only knowing the packet length and direction of the first three messages of a flow can determine which service the flow belongs to. The real-time performance of stream feature identification has been greatly improved, and it has a high identification accuracy. The DFI module only identifies streams that DPI does not identify.

Both the quintuple stream feature cache table and the quintuple table are stored in the SRAM, and their size does not exceed 4M. According to the characteristics of commonly used storage devices, SRAM can achieve higher actual throughput, so it is selected for storage in our high-speed system.

The DFI preprocessing module completes the scheduling control of packet characteristics of the upstream and downstream and the integration of messages. Since the data throughput is still relatively large, a higher processing speed is required. Interaction with SRAM is pipelined to meet data throughput requirements.

The classifier needs to process the previous stream feature before the next stream feature arrives, so the real-time performance of the classifier also needs to be considered. In order to achieve a balance between resources and processing time, the operation is changed to partial parallelism and resource multiplexing. The classification rules in the classifier can be customized. When the service category changes, the OLT receives the new classification rules and transmits them to each ONU through OAM frames, and each ONU assigns the new classification rules to the classifier through the CPU.

Through the cooperation of DPI and DFI, the mainstream services on the Internet can be accurately identified. The precision of final business identification is over 90%, and the recall rate is over 80%.

IV. FPGA DESIGN AND IMPLEMENTATION OF QUINTUPLE IDENTIFICATION MODULE

The quintuple identification method is also an early-developed service identification method. It is an idea of association identification, and it is assumed that the flow satisfying a quintuple carries the same service. Using the quintuple identification technology alone to identify services has inherent defects, because the same quintuple stream may carry different services and cannot be subdivided. Therefore, the quintuple identification method is often used as an auxiliary identification method in recent years, and it can be assumed that the same quintuple stream carries the same service in a short period of time.

The quintuple identification module is indispensable in the 10G EPON ONU implementation scheme that supports service identification. In this scheme, the quintuple identification module works together with the DPI

and DFI modules. The quintuple identification module stores the quintuple of the already identified services by the previous keyword identification module and the classifier module and the corresponding service type in the hash table. For the business data flow that comes later, extract the quintuple information in the frame, including source IP address, destination IP address, source port, destination port and protocol type, and then conduct classification through querying and matching the entries in the quintuple table.

When the quintuple of the newly arrived service data stream is the same as that in the quintuple table, it indicates that this service is a known service, and output the corresponding service type; When the quintuple of the newly arrived service data stream is different from the quintuple in the quintuple table, then output the type of the service as unknown service.

A. Requirements Analysis

1) Functional Requirements

- Extract the quintuple information of each frame, find the quintuple table for matching, and output the recognition result;
- Store the recognition results of the previous keyword recognition module and the classifier module in the quintuple table;
- Periodically age the entries in the quintuple table;
- Correctly handle conflicts between matching, updating, and aging requests.

2) Performance requirements

- Store more than 2,100 quintuples as a hash table;
- The correct rate of quintuple matching is more than 90% (that is, the conflict rate is below 10%);
- Linear speed processing is achieved, and the throughput is maintained at 10Gbps;
- The aging time of each quintuple entry is 1.5s.

B. Difficult Problem in the Design

1) Scale analysis of quintuple stream hash table

A typical application scenario is that the residential users are connected to the ONUs, the access rate of each ONU is 160Mbps, the bandwidth allocated by the ONU to each household is 10Mbps, then the number of residential users connected to the ONU is 16. Considering that each household has 2 PCs and 2 smart terminals (such as mobile phones and pads) . It can be assumed that a terminal running P2P generates 60 data streams in 1 second, and a terminal that does not run P2P generates 20 data streams in 1 second. Home users run P2P applications through ordinary PCs, and other terminals do not run P2P applications. The maximum usage rate of P2P is assumed to be 60%, then the number of data streams reaching the ONU within seconds is $16\times2\times60\%\times60+16\times2\times40\%\times20+16\times2\times20=2048\approx2100$

2) Choice of hash algorithm

The hash solution adopts the Bob + Bit Extraction algorithm [8] [9], which can meet the requirement that the correct rate of quintuple matching is over 90%, that is, the collision rate is less than 10%. An ONU will concurrently have 2100 different streams per unit time (1s). The designed hash address has 12 bits, which can ensure that the quintuple matching rate is higher than 90%. The reasoning process is as follows: Assuming that the length of hash bucket is 2[10], to ensure that the correct rate is above 90%, the calculated hash address is at least 12 bits, and the hash table address is 2^{12}=4096. Assuming that the hashing algorithm has good hashability [11] and the hash value distribution is relatively uniform, then the probability of storing at most 2 entries in the same address is 98.465% according to the computational formula.

Thus, it can be seen from the above calculation that the probability of no conflict in the quintuple table can reach 98.465%. In addition, according to the experimental data (750,000 data streams of debugging record), the

experimental results show that the conflict rate of Bob algorithm is 2.5×10^{-6}, based on which the final conflict rate can be estimated to be 0.8% [8], which meets the requirement that the conflict rate is below 10%, and can be used in the quintuple identification module.

3) Linear Speed Processing with Throughput Remains at 10Gbps

This design scheme can meet the line-speed processing for medium and long frames, and can process 8 bytes per clock cycle, that is, the throughput is maintained at 64bit×156.25MHz=10Gbps. For the most severe conditions in the network environment, that is, the shortest frame and the shortest frame interval, how to deal with it becomes the difficulty in the design of this module. The processing of one frame by this module includes the following three steps:

- Hash calculation, which takes about 3 clock cycles in total [12];
- FPGA access off-chip SRAM, which takes about 2-3 clock cycles;
- Data string comparison, which takes about 1 clock cycle.

As we all know that the quintuple can be read at least at the 46th byte of a frame. For TCP stream, the shortest frame is 72 bytes, leaving 26 bytes of data, that is, 3 clock cycles, that is to say, the identification result cannot be given before the end of the frame. The idea of pipeline operation is designed: in the time of reading the data of the current frame, the hash value of the next frame is calculated at the same time, and the data of the previous frame is compared and output the service identification result. It can be seen from the above analysis that the pipeline operation can meet the linear speed processing.

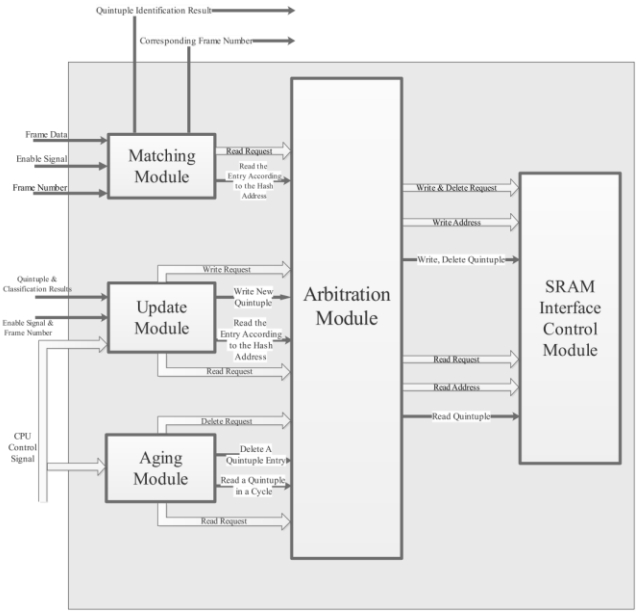

Figure 4. FPGA design of quintuple service identification module

4) Requirements for Off-chip SRAM Throughput

In the extreme circumstance, all the frames and frame intervals in the system are the shortest; the keyword identification module writes a quintuple for each packet identified, and the write throughput is 16×8bit/60.8ns=2.11Gbps; the DFI module writes the characteristics of every three packet into the quintuple when they have been identified, so the write throughput is 16×8bit/182.4ns=0.70Gbps, and the total write throughput is 2.81Gbps.

Here, every packet will be read in the process, and the maximum throughput is 32×8bit/60.8ns=4.21Gbps. Therefore, the throughput required by the quintuple table is 2.81Gbps in the write direction and 4.21Gbps in the read direction, and with a total of 7.02Gbps if the off-chip SRAM storage satisfies the add operation. Here, we choose QDR-SRAM with a data width of 36bit and a clock frequency of 400MHz[12], which can provide the

highest throughput of 36×4bit/10ns=14.4Gbps. Therefore, the throughput of the selected SRAM can meet the requirements.

C. FPGA Design of Quintuple Service Identification Module

Figure 4 is the FPGA diagram of the quintuple identification module, where the linear arrows represent the data flow, and the box-shaped arrows represent the control flow. The input of this module comes from 4 modules, the first one is the frame data, enable signal and frame number output from the frame delimiter counting module, the second one is the quintuple, classification result, enable signal and frame number output from the keyword identification module, the third one is the quintuple, classification result, enable signal and frame number from the DFI classifier, and the fourth one is the control signal from the CPU; the output has only one destination, which is to output the classification result, enable signal and frame number to the identification result judgment cache module[13][14].

According to functional requirements and performance requirements, five sub-modules are designed, which are matching module, update module, aging module, arbitration module, and SRAM interface control module.

V. SIMULATION AND ANALYSIS

In order to better verify the functions of the designed module, we use the excitation source on the simulation platform to simulate and test the module. Among them, the excitation source A inputs a data frame in XGMII format, the data transmission interface is 8 bytes, and the control information transmission interface is 1 byte. The excitation source B inputs the quintuple and the classification result, and simulates the function of the keyword identification module and the DFI classifier. The excitation source C simulates the data of the quintuple table in the SRAM.

First, we carry out the simulation for the matching module and the update module.

A. Identify the First Packet of a Flow by Keyword

1) Excitation Source Design

Identify the first packet of the flow by the keyword, configure the quintuple to the quintuple identification module, and identify the subsequent data packets through the quintuple identification module. It can be seen from Figure 5 that the keyword identification module configures the quintuple of frames whose identification results are x"04" and x"08" to the quintuple identification module. The excitation source settings are as follows: a total of 4 streams are designed, namely HTTP, BT, PPStream, and PPlive, with a total of 8000 frames.

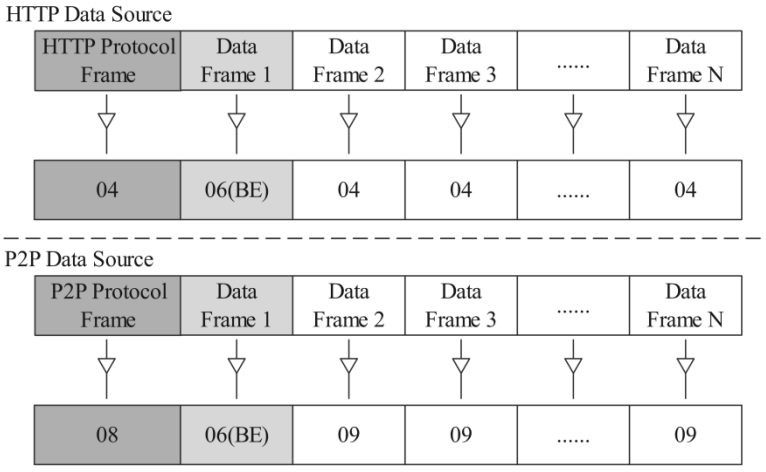

Figure 5. HTTP and P2P data source

2) Result Analysis

The test results show that the workflow of the quintuple recognition module is consistent with the design, that is, the quintuple input by the keyword recognition module is correctly updated to the quintuple table; extract the IP address, protocol, port and other information according to the position of the frame header and the length of the IP

header; then compared it with the rules in the quintuple table, and the identification result is output after one clock cycle.

3) Result Verification

According to the experimental results, the identification results of the four frames are "04", "08", "08", "08", and the identification results of the data frames are "04", "09", "09", " 09", the identification result of the first data frame of the 4 frames is "06", which is consistent with the expected result.

B. Identify the First Few Packets of the Flow Through DFI

Identify the first few data packets of the flow through DFI, configure the quintuple to the quintuple identification module, and identify the subsequent data packets through the quintuple identification module. Here, there are five DFI simulation rules, the identification results are all P2P (x"09"), all other message feature identification results are non-P2P (x"05"), and the excitation source is designed as 10 flows, of which 6 are TCP flows, 4 UDP flows, the number of uplink frames is 2033, and the number of downlink frames is 34.

The simulation results that the workflow of the quintuple identification module is consistent with the design, and the quintuple input by the DFI module is correctly updated to the quintuple table, extract the IP address, protocol, port and other information according to the position of the frame header and the length of the IP header, and then compare with the rules in the quintuple table, and output the identification result after one clock cycle.

After verification, the identification result of the frames in the first three message collection stages of each flow is "06", a total of 78 frames. In addition, the subsequent frames can be correctly identified.

C. Conflict Handling

The excitation source is set to 2433 frames, and there are both DPI-identified flows and DFI-identified flows. The keyword identification module and the DFI module can simultaneously input quintuple updates to the quintuple identification module.

From the simulation results, it can be seen that when the keyword and DFI configure quintuple to the quintuple identification module at the same time, the quintuple identification module correctly handles the conflict and does not miss a quintuple that needs to be updated. Th uplink frame data at the output end of the service identification module is completely consistent with the uplink frame data at the input end, and the actual identification result at the output end is completely consistent with the expected identification result at the input end.

By simulating the aging sub-module, arbitration sub-module, and service identification capability in extreme cases in the quintuple identification module, the results show that the design scheme proposed in this paper is consistent with the expected function.

In addition, the simulation results of the port identification module, the uplink packet feature extraction cache module, and the data FIFO module also show the expected effect.

VI. CONCLUSION AND FUTURE WORK

This paper proposes a FPGA design and implementation method for modules in 10G EPON ONU. These modules are mainly the core modules of the DPI part, including the port identification module, the quintuple identification module, the uplink packet feature extraction cache module, and the data FIFO module. In the process of design and implementation, the functional requirements, performance requirements, block diagrams, sub-module design, interface signals, and simulation tests of each module are analyzed.

Ethernet is the mainstream business of the 10G PON system, so in the FPGA development process of the 10G PON system, the input excitation source of the XGMII interface Ethernet frame is adopted in the function simulation of the module, and the output signal of the tested module needs to be verified at the same time in order to verify whether the logic of the tested module is correct, such as Ethernet frame transceiver module, service identification module, service queue scheduling module, etc. Based on the above reasons, this paper designs a flexible and configurable 10GE Ethernet frame excitation generation and result verification simulation platform to provide functional simulation verification methods for FPGA development.

The experimental verification follows the general FPGA development process. First, the most basic function and performance simulation under Modelsim is carried out, followed by the joint debugging of the modules, then the down-board debugging function and linear speed processing performance, and finally the test of accuracy and recall in the actual network environment. Each simulation and test includes the design of the excitation source, the simulation results, the result description, the result verification, and the verification description.

The 10G EPON ONU design scheme that supports service identification provided in this paper, is more generally applicable to point-to-point systems in the network, and this feature can be fully utilized to apply to more network equipment; In addition, the two main performance indicators, i.e., precision and recall are set above 90% and above 80% respectively, which is a conservative goal considering the laboratory environment and implementation complexity, and if conditions permit, there is room for improvement.

Future work will focus on making more improvements at the level of algorithms and implementation methods involved in the two technologies of DPI and DFI.

ACKNOWLEDGMENT

This work was supported by the 2021 Cooperative High-Tech Project between Zhejiang Police College and Shanghai Jiao Tong University under Grant No. 31511080401.

REFERENCES

[1] Rayapati, Bhargav Ram , and N. Rangaswamy, "Bridging electrical power and entropy of ONU in EPON," Optoelectronics Letters, vol. 17, no. 2, pp. 1-5, 2021.

[2] Rahalkar, Chaitanya, and D. Gujar, "Content Addressed P2P File System for the Web with Blockchain-Based Meta-Data Integrity," IEEE 2019 International Conference on Advances in Computing, Communication and Control (ICAC3), Mumbai, India, 2020.

[3] Yamauchi H, Nakao A, Oguchi M, “Service Identification Based on SNI Analysis,” IEEE 17th Annual Consumer Communications & Networking Conference (CCNC), Las Vegas, NV, USA, pp. 1-6, 2020.

[4] Zhang Q, Zeng Q, and Li Y, “Research and design of low noise broadband frequency source based on FPGA and DDS technology,” IEEE MTT-S International Wireless Symposium (IWS), Shanghai, China, 2020.

[5] Charyyev B, Gunes M H, “IoT Traffic Flow Identification using Locality Sensitive Hashes,” ICC 2020 - 2020 IEEE International Conference on Communications, Dublin, Ireland, 2020.

[6] Qiu Y, "Time based resource-consumption-aware spectrum assignment for multicast traffic in elastic optical networks," Optical Fiber Technology, vol. 59, no. 11, pp. 102325.1-102325.7, 2020.

[7] Kyriakopoulos C, Nicopolitidis P, Papadimitriou G and E Varvarigos, “Adapting Spectrum Resources using Predicted IP Traffic in Optical Networks,” 17th International Joint Conference on e-Business and Telecommunications, vol 1, pp. 176-182, 2020.

[8] Cheng G, Jian G, “A method to device flow hash algorithm based traffic character,” 11th IEEE International Conference on Communication Technology, pp. 41-45, 2008.

[9] Ramakrishna M. V. , E. Fu , and E. Bahcekapili, "A Performance Study of Hashing Functions for Hardware Applications," Proccedings of the International Conference on Computing and Information, vol. 36, no. 2, pp. 23-26, 1994.

[10] Peng L, Yang B, Chen Y, “Effective packet number for early stage internet traffic identification,” Neurocomputing, vol. 156, pp. 252-267, 2015.

[11] Choi, Yongmin, “On the Accuracy of Signature-based Traffic Identification Technique in IP Networks,” IEEE/IFIP International Workshop on Broadband Convergence Networks, Munich, Germany, pp. 1-12, 2007.

[12] Kuon I, Tessier R, and Rose J, “FPGA Architecture: Survey and Challenges,” Foundations and Trends in Econometrics, vol. 2, no. 22, pp. 135-253, 2007.

[13] Lv Y, Bi M, Zhai Y, and H Chi, “Study on the Solutions to Heterogeneous ONU Propagation Delays for Energy-Efficient and Low-Latency EPONs,” IEEE Access, vol. 8, pp. 193665-193680, 2020.

[14] Li Y, Qian C, and Zhang Q, “Fair and efficient DWBA algorithm based on SLA differentiated polling interleaved scheduling for NG-EPON,” Optical Fiber Technology, vol. 61, pp. 102451-102461, 2021.