

Implementation of Instrument for Testing Performance of Network Based on RFC2544 Test

Niu Ling¹, Feng Gaofeng² and Duan Mei-Xia³

¹*Zhou Kou Normal University, Zhoukou 466001, China;*

²*JiYuan Vocational And Technical College, JiYuan Henan 454650, China*

³*North China University of Water Resources and Electric Power, Zhenzhou 450011, China*

Niuling@zknu.edu.cn¹, fengjyjava@126.com², Duanmeixia@ncwu.du.cn³

Abstract

RFC2544 is an international standard to test network interconnect devices. The study provided the design idea of the instrument based on the LINUX system platform for testing network performance and the detail test schedule. This paper introduces the RFC2544 relevant standard and testing condition. And also discusses the key technique which concerns the design of the system's hardware and software. The test results indicate that the design idea and test schedule are viable which being effective for testing performance of network.

Keywords: *RFC2544, throughput, Qt, loopback test frame, back-to-back*

1. Introduction

With the rapid development of the Internet, the expansion of network scale, the network interconnection device has been widely used. The switch, router and other network interconnection device have become an important part of the network. The performance of network interconnection device will directly influence the network size, network stability and reliability, so it is necessary to accurately evaluate it. A good network performance test system is helpful to ensure the computer network can be normal, safe, efficient, rational use and operation. The network performance test instrument has become essential to evaluating performance of various types of network and network acceptance.

Current, RFC (Request For Comments) 2544 is the most authoritative standard to test and evaluate network interconnection equipment, which is proposed by the Internet engineering task force. RFC2544 is an international standard for the network equipment testing, it provides the concrete test method for performance index defined in RFC1242, at the same time, and it details the test report format. A network performance tester is introduced in this paper, taking Linux as a system software platform, S3C2440 as the core processor, and adopts the large scale FPGA circuit to realize the network performance test. At the same time, creatively put forward the network performance test across the network, has realized the leap subnet network performance test.

2. RFC2544 Testing Standards

2.1. RFC2544 Test Index and Conditions

RFC2544 developed by the IETF is an International standard for Performance evaluation for network interconnection devices. It provides the concrete test method for performance

index defined in RFC1242, and the test report format detailed provisions. According to the regulations, the network interconnect device test includes the following main index:

(1) Throughput: The index is to test the router packet forwarding capability. Usually refers to the routers' the maximum forwarding packets per second in the no packet loss conditions. General can be used to find the limit points of dichotomy

(2) Latency: The index is to test router's interval from receiving packets to forward the packet in the throughput, range, the results are the mean value of 20 times repeated the test.

(3) Packet Loss Rate: The index is to test router's proportion between dropping packet, and received packet under different loads. Different load is usually from throughput to line speed (the highest rate of packet transmission on the line), step is generally linear velocity 10%.

(4) Back-to-Back Frame: The index is to test router can handle the maximum number of packets taking transmission in the minimum packet interval under no packet loss conditions. The test is actually the ability to test the router cache. If the router has the wire speed ability (throughput = interface media line speed), the test has no meaning.

2.2. RFC2544 Test Structures

According to RFC2544, there are two kinds of test structure for network interconnection equipment. They are the master-slave test mode with two instruments and a single test mode with transceiver function instrument.

Single machine structure is a high performance test device with the simultaneous sending and receiving port to execute the testing process. In the process of testing, test equipment sent test data to receiving port of the device under test (DUT), the data are processed by DUT, and then the send port of DUT sends the processed data to the receiving port of the test equipment. According to the sent and received data, he test device quantitatively evaluates the performance of the DUT.

The dual machine structure of the test device (tester) is the single machine structure divided into 2 parts: the sender and the receiver. Usually this test structure needs another device (usually a computer) to control the process of the synchronization or time synchronization, which is complex to realize.

Single machine structure is recommended by RFC2544, because it is on the same test instrument, so it does not have the problem of clock synchronization. However the design of the dual-port restricted the volume of the testing Instrument, which makes it inconvenient to use in the test site. In the design, it is considered that the instrument will be used in different applications to test different objects. So we use two sets of test equipment. One test takes a variety of tests as a master; the other one is used to implement data loopback of the test frames of the network performance. For this kind of test case, the test parameters are all bidirectional test results.

3. Design and Implementation of a Network Tester

Based on RFC2544 test design and development, the network test instrument includes the data acquisition module hardware system and data processing and analysis module software system.

By using the high performance FPGA circuit, the hardware module can collect the data. On the basis of FPGA circuit, we run embedded Linux system to complete the entire data transceiver control function, accept the upper software commands and data interaction function. FPGA chip is mainly to complete the line speed data generating and receiving, timing processing and counting statistics and other functions.

The software module does the work such as data processing and data analysis, it also provides the simple and flexible man-machine dialogue interface to the user, and support the TCP/IP protocol.

3.1. Hardware Design

During the network performance testing of the different subnets, two test instruments are usually accessed to different subnet segment, one tester is used as the host sending data of the test frame, the other one is set to loop back . In the design of the test, the connection diagram is shown in Figure 1. In the testing process, we need to set the tester routing on the router to make sure that the UDP test frame can reach the test instrument in the subnet.

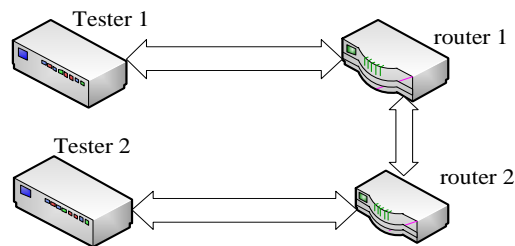


Figure 1. Test Router Connection

According to the RFC 2544, the test equipment can configure parameters flexibly. Such as frame size, frame rate, frame number, frame interval, and then send test frame in a certain period to the device under test (DUT). This need to control the speed of the occurrence of testing frame. The equipment throughput is usually tested in the first, and then the next step is the test of latency, packet loss rate, and back to back.

In the design, during the throughput test, we adjust the send rate with bisearch algorithm till the speed of the send frame of the DUT equal to the received frame of the DUT; the wire-speed stepping rate is 1%. So the instrument needs to control byte time parameters strictly. While using software methods have such problem as the long time delay and low precision, and cannot meet the needs of the actual test. Therefore, it is necessary to use the method of the hardware timer. So the time precision can be achieved one byte, namely 12.5 ns. The sender adjusts the wire speed by adjusting the test frame interval and the data flow interval. The time intervals of the network data flow are presented in the diagram in Figure 2.

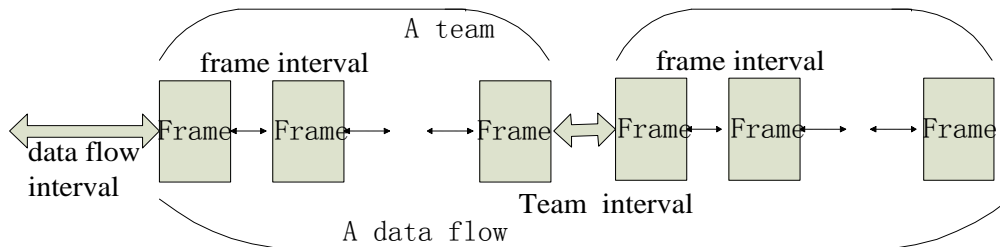


Figure 2. Network Data Flow Time Interval

The test frame circuit is mainly in the FPGA, including sending RAM, address generator, CRC checksum, framing and other circuits. The encapsulating timing circuit manages to read out a variety of software data, which is edited by the software and stored in the RAM.

According to the real time, the encapsulating timing circuit also inserts the precise time identifier corresponding number of the test frame, which makes it easy to test the network delay or other parameters.

Deframing circuit gets UDP testing frame data and the relevant test frame time parameters of the MAC frame. In order to calculate the network time parameter and the network packet loss rate, the Deframing circuit receives and analyzes the test frame to realize the timing and synchronization of data bits, analyzes the application layer data to determine whether the received data frame is the remote loop back end protocol test frame, and analyzes the related parameters of test frame, such as the test frame delay received and the test frame received out of order.

During the network performance testing of the different subnets, when the test frame goes out of the router, the source IP and the source MAC address of the test frame will be replaced by IP address and MAC address of the router. In this situation, the other loopback test instruments cannot find the IP address of the sender, and cannot send loopback test frame to the source. In order to achieve inter-network performance testing, the test frame of the sender is designed as UDP test frame. And the UDP port number of the UDP test frame can be set arbitrarily, put the source IP address of the sender into the UDP datagram. The loopback test instruments resolve the IP address of the source of the UDP packet from the UDP datagram, and this IP address will be used as the destination IP address. At the same time, the parameter identification should be added to the test frame, which indicates the behavior of the current test.

The function of the loopback end circuit is loopback sending the test frame data. The loopback end circuit will judge whether the test frame is to the loopback port. Only the test frame is sent to the loopback port, it can be sent back. The key of design of the loopback end is how to use hardware to do such work, the exchange between the source MAC address and destination MAC address, the exchange between the source IP address and destination IP address. At the same time, the hardware should realize judgment of this machine MAC address and this machine IP address, and then realize judgment of 2544 test frame, as well as, the hardware should realize CRC at the loopback end ensuring the loopback frame test packets to be returned to the sending end for statistics.

The frame format and the realization of the schematic diagram shown in Figure 3.

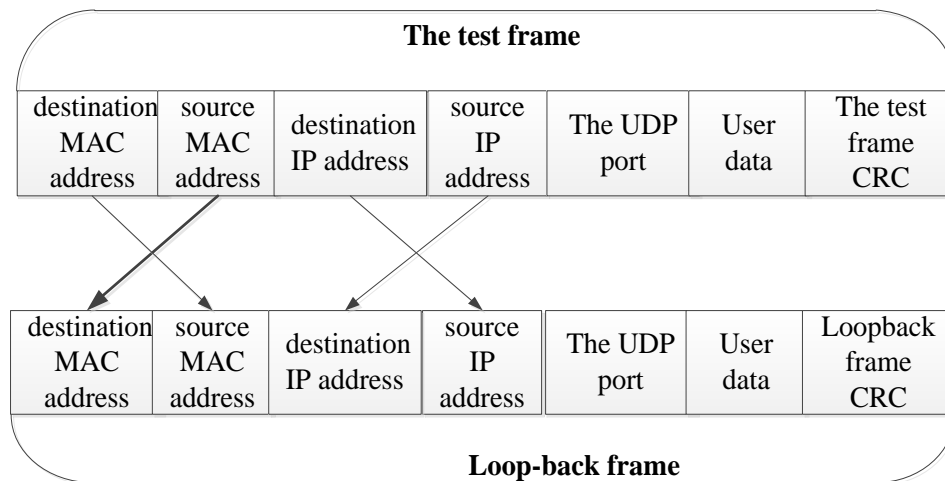


Figure 3. Frame Format and the Loopback Implementation Schematic Diagram

3.2. The Software Design

Operating systems built on the Linux kernel are used for embedded systems such as consumer electronics machine control, industrial automation, navigation equipment, spacecraft flight software, and medical instruments in general.

Embedded Linux refers to the standard Linux after miniaturized cutting out processing, which can be solidified in the memory chip capacity of only a few K or a few M byte or single chip, dedicated Linux operating system suitable for specific embedded applications.

Embedded Linux is one of the developing direction of the embedded operating system. Module design of Linux technology offers a very competitive advantage condition for embedded Linux: a very important point for the embedded operating system is that requires less resources, and be able to carry on the management and control of dynamic for loading resources, The modular design of embedded Linux is just the perfect characteristic.

The advantages of embedded Linux over proprietary embedded operating systems include multiple suppliers for software, development and support; no royalties or licensing fees; a stable kernel; the ability to view, modify and redistribute the source code. Technical disadvantages include a comparatively large memory footprint (kernel and root filesystem); complexities of user mode and kernel mode memory access, and a complex device driver framework.

The embedded Linux operating system has the smallest kernel. It runs stability, and has no copyright trial fee. so it is preferred to be used in this system. The software design adopts the hierarchical design, the design hierarchy diagram as shown in Figure 4.

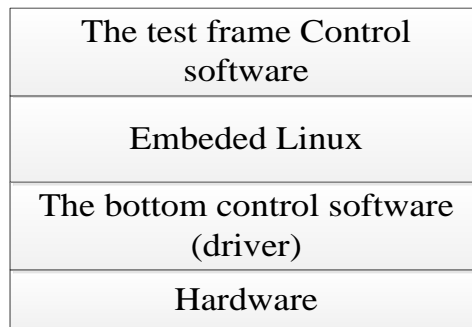


Figure 4. The Whole Software Hierarchy Diagram

3.3. The Driver Module Realization

The device driver is an important part of the Linux kernel. It is the interface between the operating system kernel and the machine hardware. The main work of the driver is to read data, such as the statistics register in the design, the test frames send counter, and the test frame reception counter. The device driver program also needs to operate data in the receiving RAM area, or interrupt mode to loop back data and related statistical register operation. So it is convenient to design all of the device as a character device. The driver interfaces between the application program and the hardware character device as Figure 5.

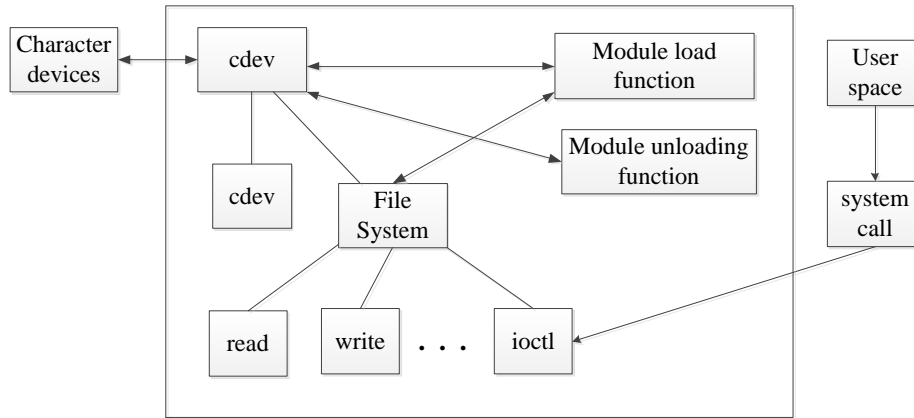


Figure 5. Schematic Diagram of Interface

3.4. The Application Layer Software Design

Qt is an iconic product of Trolltech Company. Trolltech Company was founded in Norway in 1994, but the core development team of the company has already begun in 1992 R & D Qt products, and in 1995 launched the first commercial version of Qt. Until now, Qt has been used by the software developers cross platform all over the world, and Qt functionality has been continuous improvement and improved.

Qt is the first to provide uniform beautiful graphics programming interface for cross platform software developers, but now it provides programming interfaces to the unified network and database operation. Just as Microsoft operating system provides a friendly and delicate user interface, Today, due to Trolltech's cross platform Qt development framework, but also makes the Unix, Linux operating system has a more convenient and beautiful man-machine interface.

Qt for Embedded Linux is a C++ framework for GUI and application development for embedded devices. It runs on a variety of processors, usually with Embedded Linux. Qt for Embedded Linux provides the standard Qt API for embedded devices with a lightweight window system. Qt for Embedded Linux applications write directly to the framebuffer, eliminating the need for the X Window System and saving memory. The Linux framebuffer is enabled by default on all modern Linux distributions. For development and debugging purposes, Qt for Embedded Linux provides a virtual framebuffer.

The system uses Qt- Embedded-4.5 for the development of human-machine interface. The main interface and the sub interface are designed completely in Qt Designer. Qt is used to realize the interface switch as well as the man-machine interface and the underlying SCM control system interaction.

The data of the loopback frame is spited by the application layer software. At first, the software unpacks the UDP data, and then to spin off UDP data from the Loopback data frame, so it judge whether data a network performance test frame, then to read network time parameters calculating delay parameters.

The design of network performance testing software is as follows:

(1) To judge whether the address is in the network segment: if it is, then directly send the loopback test frame, start the remote loopback device into the loop state; if it is not, first find the gateway, then send the loopback test frame, tart another remote loopback device into the loop state.

(2) To send transmit path verification test frame, for network access authentication.

(3) To send a test frame, combined with the hardware design, take a network performance test

(4) To send stop test frames, let the loop device to exit the loop state

Network performance software testing process as Figure 6

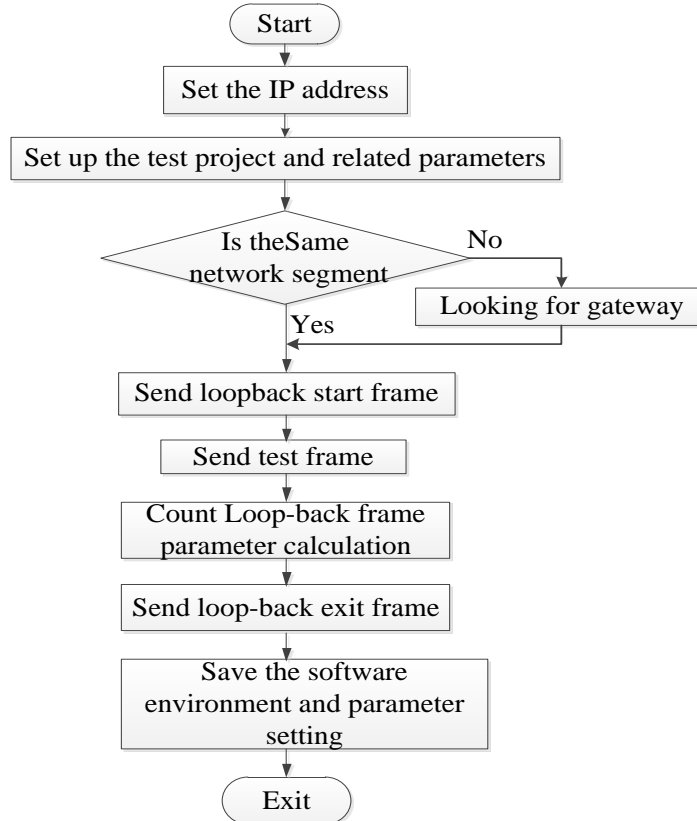


Figure 6. Network Performance Software Testing Process

4. The Test Results

According to the network composed of 2 household routers, We conducted a series of tests such as throughput, latency, frame loss rate and the back-to-back frame test. The results show that the test results and the general product test results are basically consistent. The analysis of the results shows that, short frame network data through a router cost more which leading to the smaller throughput. It is also found in the router testing, that the test instruments need to be able to support IGMP simulation, because the router will send out a query command. If there is no response to the query frame, the router will stop the test frame forwarding to the tester of the subnet.

5. Conclusion

Using data acquisition system of the front hardware and data processing and analysis system of the background software, in Latency test, by using high precision counter in the FPGA module for clock synchronization and counting, its timing accuracy is up to 40 ns, which greatly improved the accuracy of time delay test.

In this paper, a new network tester based on RFC2544 test is proposed, with Linux is used as a software platform, and the large-scale FPGA circuit is used to realize the network performance cross network test. The realization is simple, which can ensure the stable and reliable performance, and network interconnection equipment can be completely applied to the actual test by this method. The instrument can be set through the parameters of the default make user conveniently carries on the network performance parameter testing, and through the way of chart make users can intuitively understand the specific network performance. This instrument has been successfully applied to various kinds of communication equipment test site, test scheme and design method for the instrument on network performance test of higher speed, such as 10G network testing also provides reference ideas.

Acknowledgment

This research is supported natural science of China foundation (61103143).

References

- [1] IETF Network Working Group- RFC2544, Benchmarking Methodology for Network Interconnection Devices [S], (1999) March.
- [2] M. Sloman and J. Moffet, "Domain Management for Distributed Systems", Integrated Network Management I, B. Meandzija and J. Westcott, eds., pp. 505-516
- [3] R. Yavatkar, D. Pendarakis, and R. Guerin, RFC 2753, A Framework for Policy Based Admission Control, (2000) January.
- [4] V. Paxson and Guy Almes, RFC2330, Framework for IP Performance Metrics, (1998) May.
- [5] V. Paxson, A. Adams and M. Mathis, "Experiences with NIMI", In proceedings of Passive and Active Measurement, (2000).
- [6] J. Case, M. Fedor nad M. Schoffstall, RFC1157, "A Simple Network Management Protocol SNMP", (1990) May.
- [7] IP Performance Metrics Working Group IPPM, available at: <http://www.ietf.org/html.charters/ippm-charter.html>, 2002
- [8] G. Almes, S. Kalidindi and M. Zekauskas, RFC2681, A Round-trip Delay Metric for IPPM, (1999) September.
- [9] R. Yavatkar and R. Guerin, RFC2753, A Framework for Policy-based Admission Control, (2000) January.
- [10] Internet Engineering Task Force Policy Working Group, <http://www.ietf.org/html.charters/policy-charter.html>
- [11] G. Goh, "Policy Management Requirements, System Management Department", HP Laboratories Bristol, (1998) April.
- [12] Hewlett-Packard Company, A Primer on Policy-based Network Management ,OpenView Network Management Division, Hewlett-Packard Company, September 14,1999
- [13] D. D. Clark and D. R. Wilson, "A Comparison of Commercial and Military Computer Security Policies", In Proceedings of IEEE Symposium on Security and Privacy, (1987).
- [14] J. D. Moffett and M. S. Sloman, "Policy Hierarchies for Distributed Systems Management", IEEE JSAC Special Issue on Network Management, vol. 11, no. 9, (1993) December, pp. 1404-1414.
- [15] R. Weis, "Policy Definition and Classification: Aspects, Criteria and Examples", In Proceedings of the IFIP/IEEE International Workshop on Distributed Systems: Operations & Management, Toulouse, France, (1994) October 10-12.

Authors



Niu Ling, she received the B. Eng degree in Computer science from Henan normal university and M. Eng degree in Computer science from Chengdu University of Technology. She is currently researching on computer application technology.



Feng Gaofeng, he received the computer science degree from Henan Normal University, China, and the master degree in computer science from Beijing University of Posts and Telecommunications. He is a member of China Computer Federation and Association of Fundamental Computing Education in Chinese Universities in Beijing, China.



Duan Mei Xia, she received the B. Eng. degree electronic technology from Henan normal university and M. Eng. degree in Computer science from Chengdu University of Technology. She is currently researching on embed system and measure.

