FPGA Based "Intelligent Tap" Device for Real-Time Ethernet Network Monitoring

Rafał Cupek, Piotr Piękoś, Marcin Poczobutt, and Adam Ziębiński

Silesian University of Technology,
Institute of Informatics
rcupek@polsl.pl, piotr.piekos@pidi.pl,
marcin.poczobutt@gmail.com, adam.ziebinski@polsl.pl

Abstract. This paper describes an "Intelligent Tap" – hardware device dedicated to support real-time Ethernet networks monitoring. Presented solution was created as a student project realized in Institute of Informatics, Silesian University of Technology with support from Softing A.G company. Authors provide description of realized FPGA[1] based "Intelligent Tap" architecture dedicated for Real-Time Ethernet network monitoring systems. The practical device realization and feasibility study conclusions are presented also.

Keywords: real-time Ethernet networks, network diagnosis, packet-analysis methods, FPGA, RISC soft-core architecture.

1 Introduction

The real-time industrial network area is now changing from solutions based on standard serial communication into area of dedicated real-time (RT) Ethernet based solutions. Another fact is the increasing size of newly created industrial networks. In fact the network structures used for the distributed industrial control systems have become more and more complicated. New functional requirements for horizontal and vertical communication are given. Despite these facts there are very few solutions which can help engineers to startup and maintain industrial networks based on real-time Ethernet. There are some general purpose network analyzers, but very few solutions are dedicated to low level Ethernet based industrial networks [1].

The traffic structure in the real-time Ethernet network consists of large number of deterministically exchanged small Ethernet packets. To support RT Ethernet network monitoring the preliminary filtration of data is necessary due to large amounts of data incoming in one instance of time [2]. Current paper contains "Intelligent Tap" hardware device presentation which is dedicated to support industrial network monitoring.

[1] FPGA – Field Programmable Gate Array – an integrated circuit which internal structure can be modified (programmed) after manufacturing.

A. Kwiecień, P. Gaj, and P. Stera (Eds.): CN 2010, CCIS 79, pp. 58–66, 2010.

Due to large number of transmitted packets real-time Ethernet network diagnosis using packet-analysis methods becomes a task, which requires high computational power and memory consumption [3]. One of solutions that allow to reduce amount of data, received by dedicated network-analysis software, is a filtering device implemented as a dedicated hardware. This paper clarifies the "Intelligent Tap" hardware filtering device concept, together with its principle of the operation and presents the realized solution for real-time Ethernet traffic filtration embedded into FPGA structure as a configurable IP core[2] named Ethernet Frame Filter. This idea is presented in Fig 1.

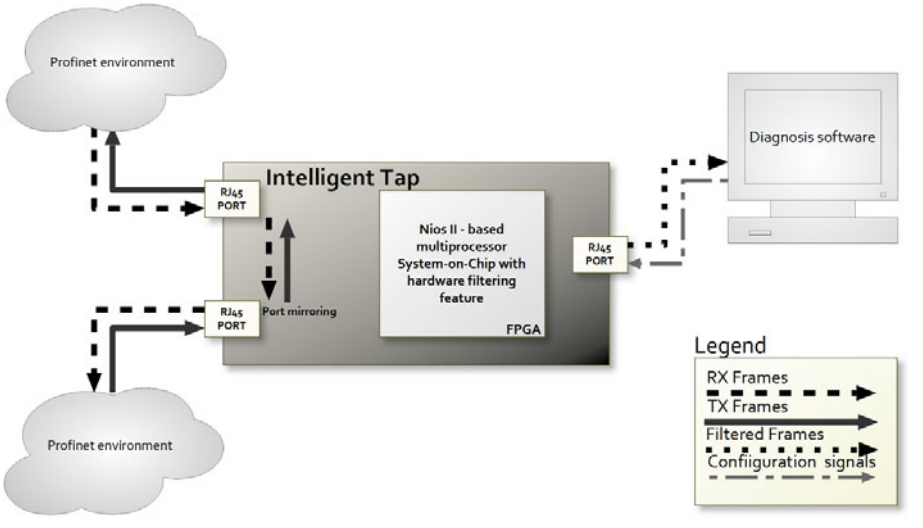

Fig. 1. "Intelligent Tap" hardware device idea

Three main goals for considered the "Intelligent Tap" architecture were formulated:

- Design of a sufficient method of Ethernet packets filtration in hardware, being able to process up to 100 Mb/s full duplex data flow.
- Development of a specification of hardware filtering module that could be embedded into Altera's family FPGA chips as an IP core.
- Delivery of a working prototype of a filtering device on particular platform: Altera's Cyclone III Development Board.

2 "Intelligent Tap" System on Chip Structure

The network monitoring process based on "Intelligent tap" filtering device consists of two main parts. First one is the software level: filtering schema preparation. The

[2] IP core – reusable unit of logic that is the intellectual property (IP) of one party. IP cores are commonly used as building blocks of FPGA designs.

user of the monitoring system is responsible for the filtering pattern preparation. For this task a dedicated filter description language and the related software components were created. This software allows for user defined mask preparation and is responsible for its translation into proper low-level register configuration of the IP core. Second part of realized solution is FPGA based "Intelligent Tap" hardware architecture presented in Fig. 2. Each rectangle in Fig. 2 embodies single System on Programmable Chip Module. Presented system comprises of NIOS II embedded processor, an IP core responsible for filtration, two Ethernet MAC's[3] and a number of standard system components. Those include the following: SDRAM controller that allows access to external SDRAM chip used as CPU instruction container as well as data storage. CFI Flash controller connected through a tri-state bridge, and EPCS controller[4] – both capable of initial FPGA configuration and program startup. Sysid component ensures safe compilation of software projects, pll (phase locked loop) divides external crystal oscillator frequency into several clock domains.Timer provides an interval-timer for Avalon-based processor systems and jtag uart module allows USB connectivity during design process.

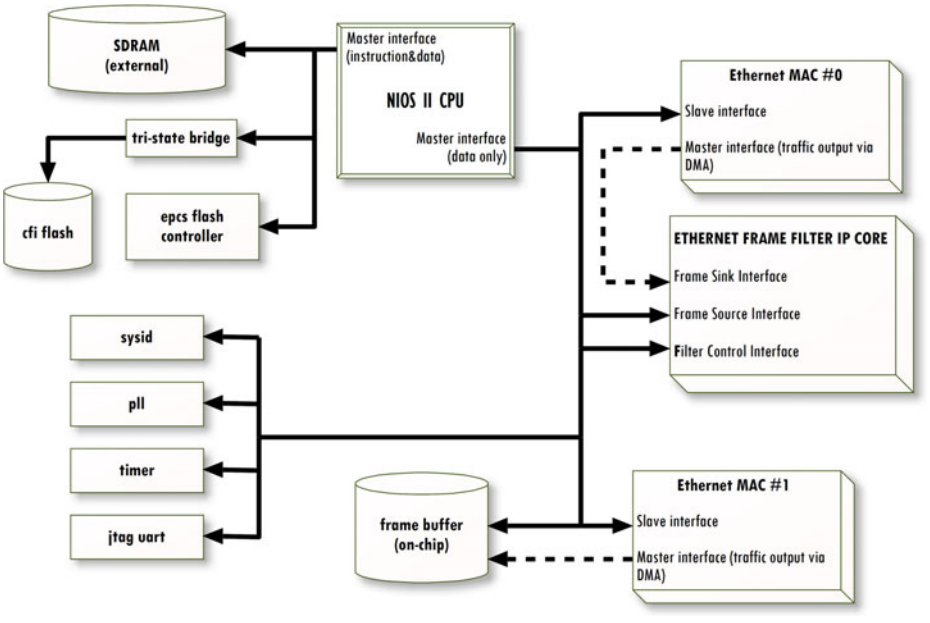

Fig. 2. The FPGA embedded "Intelligent tap" hardware structure

[3] MAC: Medium Access Controller. A piece of hardware that implements data communication protocol sub-layer specified in OSI model. In FPGA design commonly occurs as an IP core.

[4] EPCS chip controller – an IP core controller for AlteraâĂŹs Serial Configuration Device a.k.a. EPCS device.

Three different data channels can be distinguished in the Fig. 2:

1. *Instruction and data Avalon buses* represented by solid arrows indicates the communication between instruction master port of the Nios II CPU and peripherals that are able to store processor's instructions (program). Data master port is also connected to those peripherals to allow accessing data located on those mediums or to write to them. This includes following modules: EPCS chip controller, CFI Flash (through tri-state Bridge) and SDRAM. Last memory device (SDRAM) was used during testing. It stored both program and data information. Initial FPGA configuration loaded from EPCS chip should be considered overwritten.
2. *Data Avalon buses* (solid arrows) show typical Avalon Memory Mapped Interface's interconnections between NIOS II and those of its peripherals that do not consist program memory. Media Access Controllers, Ethernet Frame Filter, Frame Buffer memory and standard SOPC components (PLL, Sysid, timer, jtag uart) are accessed this way by NIOS II CPU.
3. *Direct Memory Access buses*[5] represented by dashed arrows define those interconnections that are omitting CPU and define communication between system's modules. Ethernet packets are sent from MAC using DMA interface. Two DMA sinks are present in the system: Ethernet Frame Filter module and Frame Buffer.

Filtration schema – once translated from the script query into register compatible sum of minterms form – is downloaded into FPGA structure (Ethernet Frame Filter (EFF) IP Core). This Ethernet Frame Filter preparation phase have to be done before actual network monitoring. Second phase starts after the IP core configuration is uploaded – data capturing begins. Media Access Controllers Ethernet MAC 0, ETHERNET MAC 1 insert the packets directly inside internal frame buffer of the Tap. This is done via DMA channel. If the packet arrived successfully to the buffer, MAC asserts an interrupt, informing CPU about its arrival. This causes an update (software controlled) of Tap status registers and starts the filtering process. Depending on the filtering result, an interrupt is asserted, informing that the frame matched filter's current configuration and can be sent to external buffer (IP stack, SDRAM etc.). If the filtering result was positive, packet is copied. This concludes data capturing phase, which repeats itself in the loop until filtering process is terminated.

Depending on network's payload and reference clock driving IP core, another frame might be copied into EFF buffer, while the previous one is being sent. In order to provide fast and "transparent" filtering, buffer space is paged – internal controller alternating inserts frames into upper or lower part of the buffer. Since dual-port memory is implemented, it allows parallel reading from lower part, while the upper is being overwritten by next incoming frame and vice versa. Presented Tap device is implemented on Cyclone III FPGA (40k logical elements) using only peripherals located on DBC3C40 [4] development board.

[5] Direct Memory Access (DMA): feature that allows access to certain hardware subsystems independently of the central processing unit.

Two Ethernet ports are functional, however only one of them is "equipped" with hardware filtering capability. Second Ethernet port serves as monitor port. Filtering output is redirected onto PHY[6] responsible for this port. Although not saved on any mass storage medium, filtering results (number of frames on filtering output) can also be viewed on seven segment display located on board. On-board reference clock generator provides 50 MHz signal. It is later divided by Phase-locked loop module to generate 75 MHz reference clock for all peripherals included in System on Programmable Chip design.

3 Performance Tests

Researches of the "Intelligent Tap" performance under heavy load conditions was performed in simplified Ethernet topology. Tap Ethernet port (the one capable of hardware filtering) was connected directly to packet generator soft-ware running on PC. Ethernet Frame Filter module was configured so that every incoming frame was passed to the external monitor port. Such EFF setup simulated maximum CPU load condition, since it resulted in execution of Interrupt Service Routine every time the packet arrived to EFF.

This assured execution of maximal number of instructions per time unit. As a packet generator AnetTest software was chosen. It allows creation of packets with different lengths varying from smallest possible valid Ethernet packet to largest ones defined by IEEE 802.3 standard. Generated packets can also be sent with arbitrary defined time intervals between them. In every tested scenario no time intervals were added, therefore maximum effective payload was determined by overall system performance – a PC computer equipped with Intel's Core2 Duo CPU and 100 Mb/s compatible Ethernet network card, governed by Microsoft Windows operating system. Such configuration allowed generation of following bandwidth utilization efficiency: 7.8 Mb/s for smallest possible Ethernet packet, up to 83.5 Mb/s for maximal Ethernet packet size. This radical change in payloads occurred due to system's limited ability to generate high packet per second rate (it did not exceeded 14 000 packets/s for AnetTest generator). The results of high payload testing are presented in Fig. 3.

Since maximal bandwidth utilization efficiency for 64 bytes packet size equals 84.2%, and AnetTest generated approximately ten times lower payload, different generator was used to provide more comprehensive results. This flooding software executed on UNIX based server station with 100 Mb/s compatible Ethernet network card. Generated traffic achieved levels close to maximal (approx. 77 Mb/s), using 60-bytes ARP[7] packets. Test results present that for larger packets (exceeding 600 bytes) number of packets were lost (they did not shown up on the monitor port of Technology Demonstrator). It appears that the maximal supported payload dangles around 41 Mb/s for frames larger than 600 bytes. This problem was not explicitly identified within internal system structure; however

[6] PHYceiver is a device that operates at physical layer of OSI network model.
[7] ARP – Address Resolution Protocol.

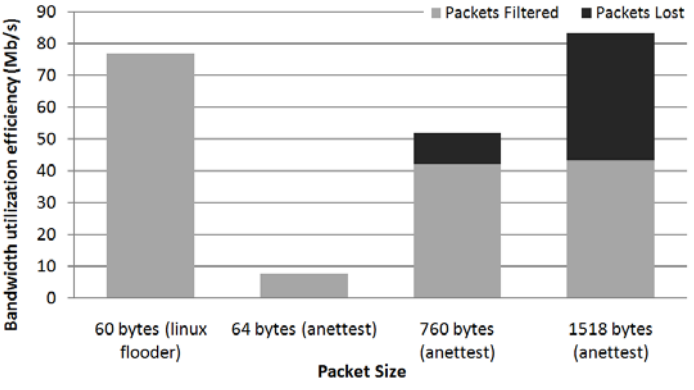

Fig. 3. Results of high payload testing

few factors that causing it were identified. Since packet size is the variable causing the problem, such threshold is most likely defined by ineffective copying operations (the bigger the frame is, the more clock cycles are needed to transfer the packet from one buffer to another within system on chip). Copying operation from the Ethernet Frame Filter IP core's internal storage to frame buffer on-chip memory takes place during execution of interrupt service routine with priority higher than MAC IRQ[8] priority. Accordingly, it cannot be halted by an incoming packet event (announced as MAC interrupt request), which results in packet loss. Additionally, software loaded to FPGA did not support multi-threading, so despite of DMA channel established between MAC and filter, and filter's support for multi-threading is impossible to receive new packet from MAC while the previous one is being sent to the monitor port of the Tap device. Running an optimized version of the software and increasing the reference clock frequency resolved the problem. This leads to the conclusion that system in presented form oscillates on the border of theoretical magnitude of network traffic. It does not completely fulfill 100 Mb/s constraint for software compiled in debug mode and reference clock of 75 MHz, while it obtains 100 Mb/s functionality after the increase of reference clock to 100 MHz and execution of optimized software in release mode.

4 "Intelligent Tap" Behavior in Profinet I/O Environment

Although presented "Intelligent Tap" may be used for any 10/100 Mb/s Real-time Ethernet frame filtering the practical tests were made in reference Profinet I/O environment that consisted of three Profinet compatible IO devices and Siemens PLC (Programmable Logic Controller) (S300 Series). All of mentioned devices were connected to Profinet compatible 8-port Siemens switch. During normal

[8] IRQ – Interrupt ReQuest.

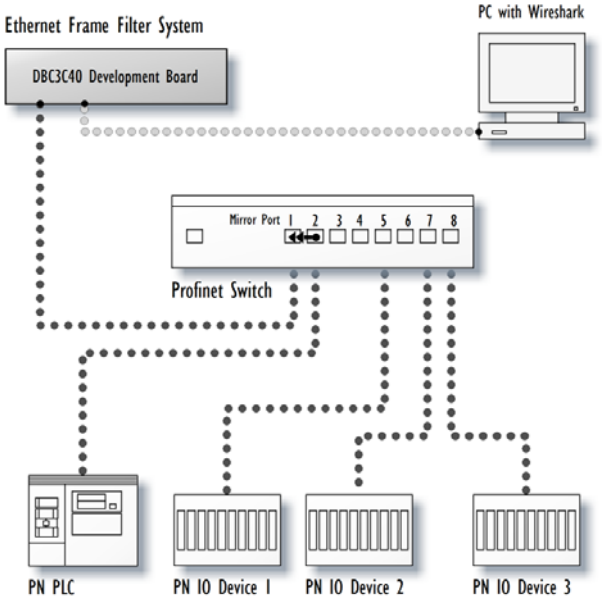

Fig. 4. The Profinet I/O "Intelligent Tap" test environment

operation, reference environment generated some (around 15–20%) VLAN[9] tagged frames [5,6]. The Profinet I/O test environment is presented in Fig. 4.

The Ethernet Frame Filter System gathered outgoing and incoming data from the PLC device through the mirror port of the switch. Testing procedures were common for following presented below PROFINET I/O communication scenarios:

- Startup of Profinet devices. Communication establishment between IO Supervisor and IO Device was investigated. Traffic consisted mainly of broadcast/multicast, LLDP[10] and DCP[11] frames. Network load did not exceed 0.01 Mb/s.
- Normal network operation where vast majority of the traffic was generated by RT frames. Bandwidth utilization in this case was evaluated to be approximately 2 Mb/s.
- Malfunctions imitation of different elements from reference topology by disconnecting or powering off some of the devices. Different elements were forced to produce alarm type frames and reestablish the communication after plugging them back into the network.

[9] VLAN – Virtual Local Area Network, is a group of hosts with a common set of requirements that communicate as if they were attached to the same broadcast domain, regardless of their physical location.
[10] LLDP – Link Layer Discovery Protocol.
[11] DCP – Device Control Protocol.

Various filtering schema were uploaded to the "Intelligent Tap" device in order to investigate its behavior under three scenarios presented above:

- all packets transmitted to PC with Wireshark software,
- no packets transmitted to PC with Wireshark software,
- only alarm frames were allowed to pass to the PC,
- bidirectional communication between only one of the IO devices and PLC,
- unidirectional communication between only one of the IO devices and PLC,
- multicast and broadcast frames only,
- real time traffic only.

More sophisticated scenarios utilizing all features of "Intelligent Tap" device were tested also for example: combination of different filtration schema uploaded to "Intelligent Tap" device. Scenarios investigated had exploited most of the cases that can occur in practical applications. Positive results were obtained in every investigated set of circumstances; however one must remember that reference topology assembled using only few Profinet devices.

5 Conclusions

This article describes "Intelligent Tap" student's project results and practical tests of network monitoring hardware device realized in Institute of Informatics Silesian University of Technology with cooperation with Softing A.G. The Tap hardware support for software real-time Ethernet networks monitoring was presented. Described solution was implemented and tested as a preliminary study for a commercial device that would compete with several other available network monitors, such as ProfiTAP device from Grid Connect Inc.

Although realized in Profinet I/O environment tests didn't find any data losing the heavy load performance tests had shown that in some extreme conditions transmitted frames may be lost. Lost packets were caused by very have network load and long frame transmitted. Because as the principle industrial networks use short packets and lower traffic load planned in the network design process [7] such condition weren't observed. Because of the more and more complicated network structure we can't guaranty that such conditions will not appear in real installation.Additional restriction is the fact that second Ethernet port serves as monitor port only. Such configuration allows for one directional network traffic monitoring or requires additional switch device with mirroring port as it was done in realized test environment presented in Fig 4.

Presented "Intelligent Tap" device has shown that embedded FPGA architectures based on one processor may by insufficient for real-time Ethernet network monitoring. Presented architecture may be improved by multi-processor parallel architecture which will offer parallel processing of receiving and sending RT data and filtering process procedures. Although presented solution can't be used for critical data registration many network monitoring applications may use it for statistical real-time network analysis and for network maintenance and startup tasks as well.

References

1. Cupek, R., Bregulla, M., Huczala, L.: PROFINET I/O network analyzer. In: Kwiecień, A., Gaj, P., Stera, P. (eds.) 16th Conference on Computer Networks, CN 2009, Wisła, Poland. CCIS, vol. 39, pp. 242–251. Springer, Heidelberg (2009)
2. Cupek, R., Huczala, L.: Passive PROFINET I/O OPC DA Server. In: 14th IEEE International Conference on Emerging Technologies and Factory Automation (2009)
3. 802.1AB Station and Media Access Control Connectivity Discovery. IEEE Computer Society, New York (2005)
4. Nios II Hardware Development Tutorial. Altera Corporation, San Jose (2007)
5. Popp, M., Weber, K.: The Rapid Way to PROFINET. PROFIBUS Nutzerorganisation e.V., Karlsruhe (2004)
6. PROFINET Technology and Application In: Siemens information materials, Karlsruhe (2005)
7. Kleines, H., Detert, S., Drochner, M., Suxdorf, F.: Performance Aspects of PROFINET IO. IEEE Transactions on nuclear science 55(1) (February 2008)