100 Gbit/s network monitoring with on-the-fly reconfigurable rules for multi-encapsulated packets

Tamás Tóthfalusi, László Kovács
AITIA International Inc.
Telecommunication Division
Budapest, Hungary
Email: {tothfalusi, lkovacs}@aitia.ai

Péter Orosz, Pál Varga
Department of Telecommunications and Media Informatics,
Budapest University of Technology and Economics
Budapest, Hungary
Email: {orosz, pvarga}@tmit.bme.hu

***Abstract*—Before the advent of FPGAs (Field Programmable Gate Arrays), hardware acceleration of networking equipment has been implemented through static architectural elements. The new generations of these highly flexible FPGA architectures can be reconfigured on-the-fly, allow parallell processing of data arriving at high-speed, and even contain hundreds of DSPs (Digital Signal Processors), to further parallelize certain, computationally intensive tasks. This paper demonstrates some of the capabilities of a new, FPGA-based networking platform, C-GEP. This multi-purpose, programmable platform can support various tasks, from being a high-speed switch/router, through pre-processing packets for Deep Packet Inspection, towards being the Forwarding Plane element in the SDN infrastructure. The current demonstration contains two further implementations of 100 Gbit/s-capable applications: a traffic generator – firing off multi-encapsulated packets – and a lossless traffic monitor – that is able to classify and steer the monitored traffic to further post-processors.**

I. Introduction

In this demonstration, we introduce a reconfigurable, high-throughput capable packet processing platform, called C-GEP. The name stands for Combinatorial Gigabit Ethernet Evaluation Platform; this paper describes a demonstration for one of its versions that support 100Gbit/s Ethernet [1] communication.

The main idea of the demonstration is to show that C-GEP is able to process the traffic arriving at 100 Gbit/s: it is able to parse, chop and filter packets based on various rules, and forward them for further processing. Besides demonstrating these capabilities, we show that the platform is also capable of acting as a full-speed traffic generator. Furthermore, since C-GEP is based on a Virtex-6 FPGA (Field Programmable Gate Array), it has dynamic hardware reconfiguration capabilities; and it can even host hardware-accelerated protocol-implementations, such as PTP (Precision Time Protocol) – which we also include in the demonstration.

C-GEP is the successor of the C-Board platform [2][3], that had similar capabilities – but it reached these with four FPGAs and up to 2x10Gbit/s. The general description of the 100Gbit/s capable version of C-GEP appears in [4].

II. Demonstration setup

The demo composition is based on two C-GEP prototype boards, which are directly connected to each other through a 100Gbit/s optical interface. One of the boards is configured as a traffic generator, which transmits Ethernet frames in loop mode. The pattern (e.g. packet sequence) and the content is reconfigurable through the management interface. The other board operates as a lossless monitoring probe, applying hardware-based packet processing phases. Next to the real-time packet parsing and classification, the firmware adds a very precise (3.2 ns resolution) timestamp to the incoming packets. The monitor device synchronizes its internal timer module to a PTP (Precision Time Protocol) server module. This synchronization requires a 1Gbit/s Ethernet management interface.

Figure 1 illustrates the demo setup including three laptops as controllers of management and presentation functions: (i) content modification and generator control, (ii) filter reconfiguration, (iii) classification results presented in Wireshark protocol analyzer software.

The monitor device inserts the timestamp in the Ethernet frame, as an extension header. To show the hardware-based delta-time between packets, Wireshark algorithms was extended through a self-made script.

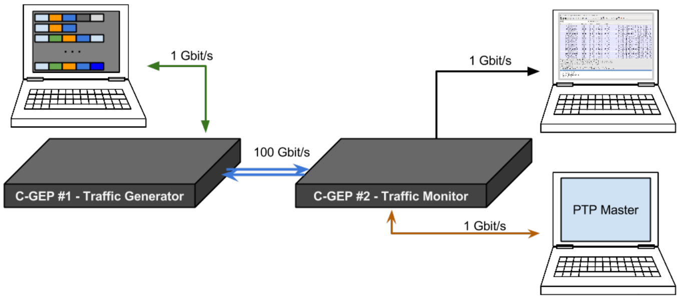


Fig. 1. Demonstration setup

A. *Traffic Generator*

The traffic generator is one of the main features to represent our platform's capabilities. The traffic generator firmware assembles the Ethernet frame content based on a predefined pattern, and transmits it on the 100Gbit/s interface at line rate. Since the central processing element is an FPGA chip, the transmit parameters can be reconfigured during operation.

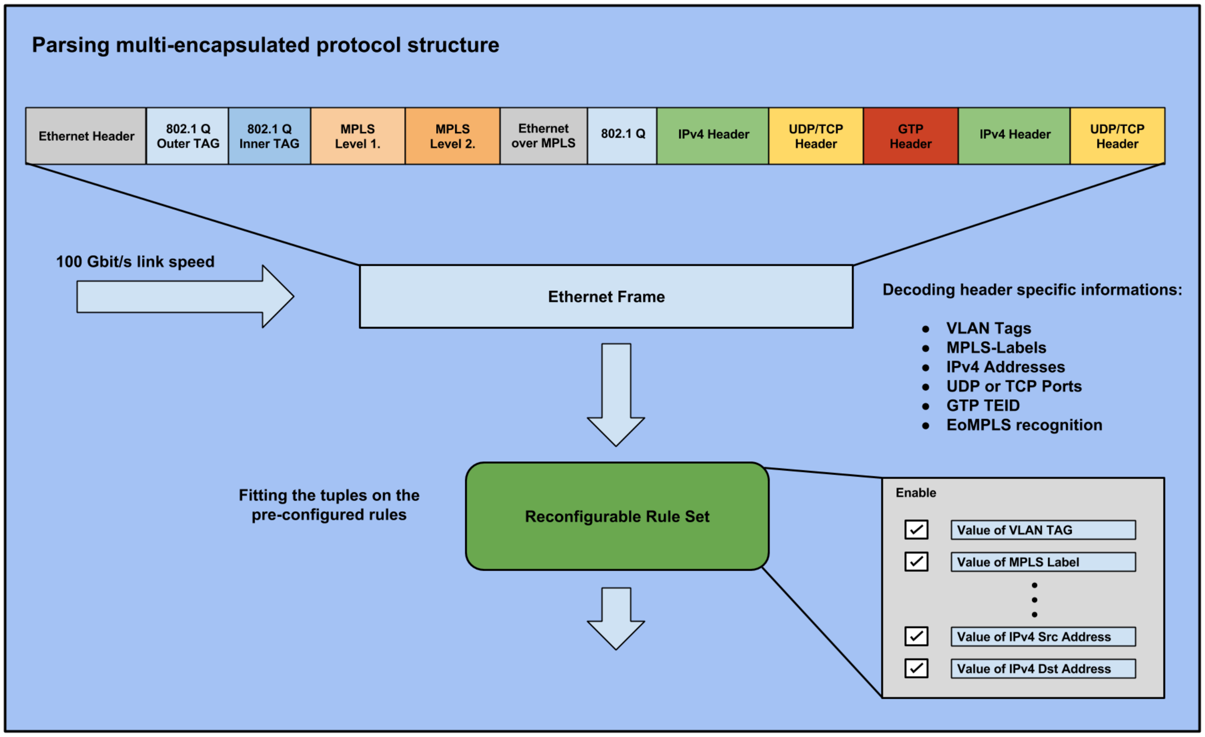


Fig. 2. Operation flow of the monitor firmware

Next to the inter-frame gap, the content and the length of the Ethernet frames can be also modified.

We have created a management web-page in order to offer better configuration and analysis capabilities. This enables some important control over the board, where the outgoing traffic speed is also measured and represented by a flow chart.

The demonstration firmware is ready to store 128 (or nx128) different Ethernet frame content, in which the maximum size of multi-encapsulated headers is 160 (or nx160) bytes. In order to control the traffic pattern, the sequence of the transmitted packets can also be defined. The size of the frame sequence storage is 512 (nx512). Aftercommiting the configuration, the firmware repeats the packet sequence over and over again.

B. Traffic Monitor

Since the demonstration setup focuses on network monitoring, the other C-GEP of the setup is programmed as a traffic monitoring board. As an architectural feature, the FPGA is directly connected to the 100G/10G/1G interfaces, providing appropriate support for high-speed packet processing within the chip. The demonstration firmware of the monitor core realizes the first DPI (Deep Packet Inspection) phases (e.g. packet parsing, packet classification), represented by Figure 2.

To achieve 100Gbit/s processing speed in a tipically *few hundred MHz operating frequency*-based device, the implemented DPI phases follow a pipeline architecture model, and applies FPGA logic features (e.g. DSP).

State of the art FPGAs are designed for high speed processing, and contain several hundreds of DSP (Digital Signal Processing) elements, aimed for general accumulation, substraction, or multiplication tasks. However, these logic blocks can be also used for pattern matching. During packet filtering, the protocol information is fitted on a preconfigured rule set, where DSPs offer high-speed rule-set matching.

Since Xilinx-DSP primitive can operate even on 48-bit input data width, a 32-bit wide *IPv4 address* concatenated with a 16-bit wide *port* can be compared for exact match in one step, saving other resources.

Another filtering approach is the *IP range match* calculation. The range check operation is realized by substracting the IP address of the rule (B) from the IP address of the packet (A). If the result overflows ([32+1]-th bit is '1'), then B >A – which means that the packet is in the lower address range. The procedure is similar for the upper end of the range, naturally with the A and B IP addresses being swapped. Considering source and destination address ranges, the engine needs 2x2 DSPs for this task.

Because of the pipeline architecture requirements, filter modules must propagate internal information. DSP elements can be also configured to propagate the incoming data without modification, next to the pattern matching function. Figure 3 depicts the internal scheme of a DSP block (within the FPGA), configured as a protocol filtering element.

Setting up DSPs as it's original function (e.g. counting, and accumulation), the demo firmware of the monitor core can calculate interface bandwith (or packet speed) by accumulating the byte (packet) count per second.

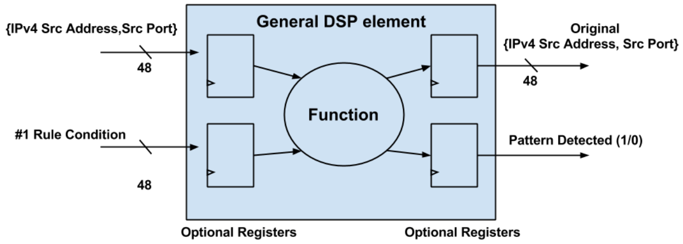


Fig. 3. General DSP element as a filter engine

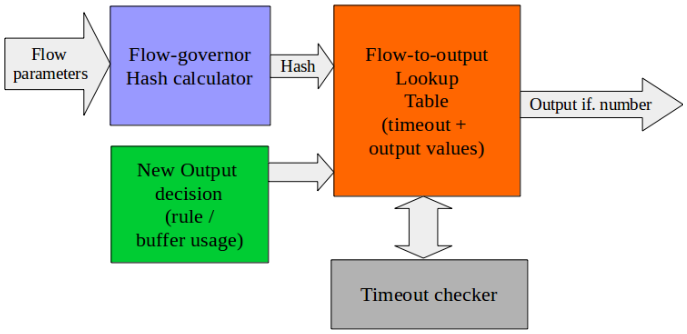

Flow parameters
Flow-governor Hash calculator
Hash
Flow-to-output Lookup Table (timeout + output values)
Output if. number
New Output decision (rule / buffer usage)
Timeout checker

Fig. 4. Traffic-based dynamic packet steering

III. Flow-based packet steering

IP packets with the same governing parameters belong to a single flow. In order to support further, sofware-based traffic analysis, we can guarantee that every packet that belongs to a given flow is forwarded to the same output interface during monitoring. This is called packet steering. The first flow parameters are detected by decoding the IP packet.

A simple IP flow is identified by the following 5-tuple:

- Source/Destination IP addresses,
- Layer 4 protocol type (fe. UDP/TCP),
- Layer 4 source/destination ports.

A fragmented IP flow (for IP reassembly) is identified by the following 3-tuple:

- Source/Destination IP addresses,
- IP identification field.

In order to support monitoring inside te mobile packet data core network, G-GEP allows filtering for GTP (GPRS Tunneling Protocol) parameter field. For GTP-U steering (U stands for User plane), the following 6-tuple parameters are deducted from the packets:

- GTP-U TEID (Tunnel Endpoint Identifier) field
- GTP encapsulated source/destination IP addresses
- Layer 4 protocol type (fe. UDP/TCP)
- Layer 4 source/destination ports

The governing parameters are used to generate a hash value. The hash value is then used to address a lookup table, and read out the output interface value from it. The table also holds a timestamp value. On every output decision the stored timestamp value is updated with the current timestamp + a timeout value. If the timestamp that has been read back from the table is less than the current time (timeout occured), then a new output interface number is chosen for the given record. The output interface number is selected by rule preferences, and/or by output buffer usage, in order to avoid overloading interfaces. This general workflow is depicted by Figure 4.

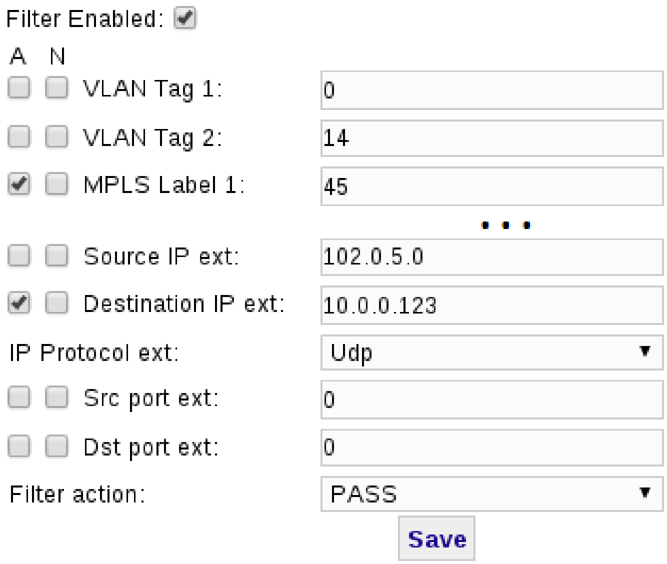
Because of the high traffic throughput, we have to perform the lookup in every clock cycle (worst case). In order to overcome this issue, an overlapped lookup mechanism is used.

IV. On-the-fly reconfigurable rule set

The FPGA is a central element of the C-GEP platform, mainly responsible for lossless packet processing. It is directly connected to an on-board management PC through a PCI-express interface. This feature allows the operator to observe the incoming and outgoing traffic measures, and link events. Packet processing operates based on a reconfigurable rule-set.

Beside other features, this demonstration shows how does the web-based user interface operates in order to to simply check the incoming traffic on the 100Gbit/s interface, and how the classification rules can be reconfigured.

The web-interface allows on-the-fly reconfiguration of the rules, without the need for interrupting the packet capture, or redirecting the incoming traffic to a default path. Figure 5 depicts some of the adjustable conditions.



Filter Enabled: ☑

A N

☐ ☐ VLAN Tag 1: 0

☐ ☐ VLAN Tag 2: 14

☑ ☐ MPLS Label 1: 45

. . .

☐ ☐ Source IP ext: 102.0.5.0

☑ ☐ Destination IP ext: 10.0.0.123

IP Protocol ext: Udp ▼

☐ ☐ Src port ext: 0

☐ ☐ Dst port ext: 0

Filter action: PASS ▼

Fig. 5. Web-based user interface

The rule set is always synchronized to the decoded header information, and to the internal packet path during the classification phase. This approach allows uploading the new rules while – in parallell – continously calculating the matching rules for the incoming packets.

Figure 6 depicts a simple simulation result of a reconfiguration process. When a reprogramming event (UpdateEN signal is high) and a new packet arrival event (SOFIN signal is high) occurs in the same clock cycle, the incoming packet will get fitted to the previously active conditions. Synchronization signals grant, that the packet is tested on that information, which was active in the arriving clock cycle. In this simulation, the RuleMatched signal equals to the rules most significant bit. The simulation figure shows, that the new Rule contains 0, but the RuleMatched signal is equal to the previous rule content.

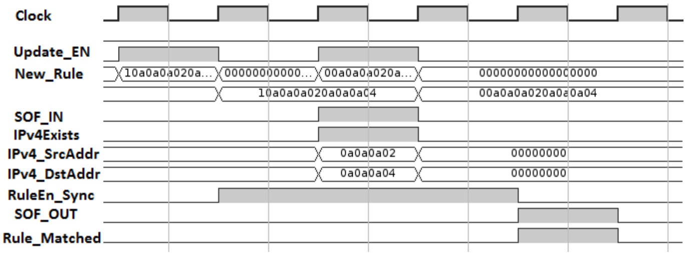


Fig. 6. Rule reconfiguration process

V. Conclusion

This demonstration aims at providing an overview of the highly configurable C-GEP platform, especially its 100Gbit/s-capable version. The C-GEP platform is programmable, provides highly accurate time-stamping for network monitoring, and it is designed for lossless packet capture. It can be widely used by research engineers, application developers as well as operators in order to meet the challenges of new network and service paradigms such as Software Defined Networking and Network Function Virtualization – at high line rates.

VI. Acknowledgement

This research was partially funded by the national project C-GEP, GOP-1.1.1-11-2012-0031, in Hungary. The AITIA part of this research was partially funded by the national project DIANA, KMR_12-1-2012-0207, in Hungary.

References

[1] *802.3ba-2010 – CSMA/CD Access Method and Physical Layer Specifications Amendment 4: Media Access Control Parameters, Physical Layers, and Management Parameters for 40 Gb/s and 100 Gb/s Operation*, IEEE Standard 802.3ba-2010.

[2] P. Varga, I. Moldovan, D. Horvath, and S. Plosz, "A Low Power, Programmable Networking Platform and Development Environment," in *Network-Embedded Management and Applications (NEMA)*, Niagara Falls, Canada, 2010, pp. 19–36.

[3] I. Moldovan and P. Varga, "A Flexible Switch-Router with Reconfigurable Forwarding and Linux-based Control Element," in *IEEE 10th International Symposium on Electronics and Telecommunications (ISETC)*, Timisoara, Romania, 2012, pp. 217–220.

[4] P. Varga, L. Kovacs, T. Tothfalusi, and P. Orosz, "C-GEP: 100 Gbit/s Capable, FPGA-based, Reconfigurable Networking Equipment," in *IEEE 16th International Conference on High Performance Switching and Routing*, Budapest, Hungary, 2015.