APPLICATION-SPECIFIC PROCESSOR FOR STATEFUL NETWORK TRAFFIC PROCESSING

Jan Kučera
Bachelor Degree Programme (3), FIT BUT
E-mail: xkucer73@stud.fit.vutbr.cz

Supervised by: Lukáš Kekely
E-mail: ikekely@fit.vutbr.cz

Abstract: This paper presents the design of an application-specific processor for high-speed networks. The unique combination of hardware acceleration and software flexibility based on a new concept of flexible flow-based monitoring called Software Defined Monitoring (SDM) will allow creation of complex network security and monitoring applications. The system enables offloading of the computation of various aggregations and statistics from the main system CPU to the FPGA accelerator. The firmware of the proposed system (application-specific processor) is implemented in the FPGA chip and it allows packet capture and monitoring of the network traffic at full speed of a 100 Gbps network interface.

Keywords: FPGA, 100 Gbps, Processor, Network Measurement, Software Defined Monitoring

1 ÚVOD

Současný trend ve vývoji počítačových sítí vede k přechodu na vyšší rychlosti komunikace a nasazování nových technologií s přenosovou rychlostí 40 Gb/s a 100 Gb/s na páteřních linkách a v datových centrech. Vývoj komunikačních sítí se však musí promítnout i do oblasti bezpečnosti a monitorování těchto sítí. Aktuálně používané postupy totiž nedosahují požadované výkonnosti pro tuto novou generaci vysokorychlostních sítí. Řešení uvedeného problému se neobejde bez podpory hardwarové akcelerace časově kritických operací. Při návrhu celého systému monitorování je rovněž důležité volit vhodné rozložení činností mezi hardwarovou a softwarovou vrstvu. Hardwarová akcelerace nabízí vysoký výkon a rychlost zpracování, avšak výsledný systém je špatně rozšiřitelný. Východiskem je úzké provázání se softwarovým řízením, které poskytne dostatečnou flexibilitu. Touto problematikou se zabývá koncept softwarově definovaného monitorování (SDM, Software Defined Monitoring) [1] využitelný pro flexibilní monitorování sítí, které je založené na sledování síťových toků.

Základní myšlenkou konceptu SDM je umožnění softwarem řízeného hardwarového předzpracování síťových dat a kontrolovaná ztráta příchozích informací. Několik prvních paketů nového síťového toku je odesláno k softwarovému zpracování. Softwarový řadič dále rozhodne, jaký způsob hardwarového předzpracování bude zvolen, a zavede do firmwaru pravidlo pro následující pakety tohoto síťového toku. Následně jsou k danému síťovému toku do softwaru odesílána pouze předzpracovaná data ve formě extrahovaných informací ze záhlaví paketů nebo agregovaných např. NetFlow statistik k celému síťovému toku. V případě nutnosti detailnější analýzy příchozích síťových dat lze zvolit i nadále odesílání celých paketů k plně softwarovému zpracování.

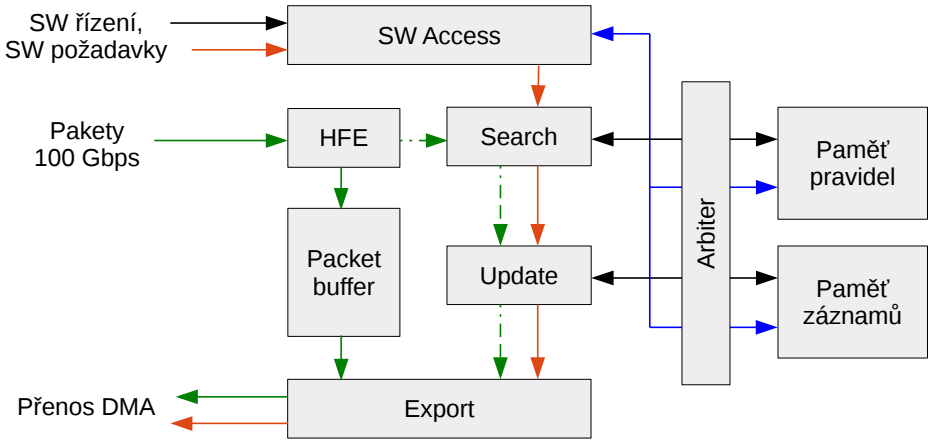
Cílem práce je návrh architektury a implementace firmwaru uvedeného systému, jehož stručné představení je úkolem následující kapitoly.

2 FIRMWARE SYSTÉMU

Celý systém je navržen pro hardwarovou akcelerační kartu s čipem FPGA Virtex-7 poskytující rychlé rozhraní PCI-Express gen3, externí statické paměti QDR a podporující technologii 100G Ethernet.

Firmware FPGA je vytvořen s využitím platformy NetCOPE [2] pro realizaci příjmu a vysílání dat na síťových rozhraních Ethernet a zajištění komunikace po sběrnici PCI-Express. Součástí platformy je také řadič přímého přístupu do paměti (Direct Memory Access, DMA) umožňující rychlé přenosy dat do operační paměti počítače.

Jádro firmwaru FPGA je tvořeno vlastní implementací aplikačně specifického procesoru. Zpracování všech vstupních požadavků probíhá zřetězeně za účelem dosažení vysoké propustnosti. Činnost procesoru je řízena instrukcemi uloženými v rámci pravidel v externí paměti. Instrukce určuje, jaká operace má být provedena s každým příchozím paketem ze vstupního síťového rozhraní. Každé pravidlo tak udává způsob hardwarového předzpracování dat. Po provedení instrukce jsou data předávána softwarové vrstvě buď ve formě původního paketu, jeho zkrácené verze, extrahovaných informací ze záhlaví paketu nebo agregovaných statistik o daném síťovém toku. Schéma architektury navrženého procesoru je zobrazeno na obrázku 1.



Obrázek 1: Architektura navrženého procesoru.

Zpracování všech příchozích paketů (zeleně znázorněná datová cesta) začíná analýzou a extrakcí jejich záhlaví v jednotce *HFE (Header Field Extractor)*. Původní přijatý paket je dočasně odložen do *FIFO paměti (Packet buffer)*. Extrahovaná data jsou v jednotném formátu předána k dalšímu zpracování (přerušovaná, zeleně znázorněná datová cesta). Jednotka *Search* dále vyhledá pravidlo v externí paměti příslušející příchozímu paketu. Je-li pravidlo nalezeno, je předáno spolu s extrahovanými daty jednotce *Update*, kde je provedena operace dle obsažené instrukce – aktualizace záznamu o příslušném síťovém toku v externí paměti (adresa záznamu je součástí pravidla). Blok *Export* dále zajistí vyzvednutí přijatého paketu z vyrovnávací paměti, jeho další zpracování (zkrácení, zahození) a případně odeslání paketu nebo extrahovaných dat do softwaru. V případě, že není nalezeno odpovídající pravidlo, se provede výchozí způsob zpracování (všechny neznámé příchozí pakety jsou ve výchozím stavu odesílány do softwaru k další analýze). Odesílání do softwaru se realizuje přímým přístupem do operační paměti (DMA).

Paměť pravidel je spravována softwarovým řadičem skrze jednotku *SW Access* (modře označená datová cesta). Softwarový přístup k paměti záznamů slouží především k počáteční inicializaci záznamu a pro ladicí účely. V případě požadavku na exportování záznamu o síťovém toku do softwaru nebo exportování záznamu se současným vynulováním záznamu či odstraněním pravidla (červeně znázorněná cesta) je tento požadavek předán jednotce *Search*, která na základě obsažených informací o síťovém toku vyhledá a případně odstraní pravidlo. Součástí pravidla je také adresa požadovaného záznamu, ta je předána jednotce *Update*, která zajistí načtení a případné vynulování záznamu. Záznam je následně odeslán jednotkou *Export* do kanálu DMA. Tento přístup umožňuje dodržení atomicity a korektního zpracování při práci s pravidly a záznamy s ohledem na souběžně probíhající zpracování příchozích paketů. Mimoto poskytuje jednotka *SW Access* přístup ke stavu a konfiguraci procesoru.

Pro vyhledávání a jako způsob přístupu k *paměti pravidel* bylo využito principu kukaččího hašování (Cuckoo Hashing) [3]. V souvislosti s tímto přístupem bylo třeba řešit problematiku přístupu do paměti QDR. Jakoukoliv modifikaci dat je nutné provádět ve dvou fázích (čtení a následný zápis), přičemž každá operace přístupu do externí paměti má určitou latenci a požadavky jsou z důvodu vysoké propustnosti zpracovávány zřetězeně. Pro zajištění atomicity těchto operací vznikla jednotka *Arbiter*, která řeší správnou synchronizaci a řízení přístupu k externí paměti.

Další problematickou úlohou je analýza a extrakce záhlaví příchozích paketů. Jednotka *HFE* je založena na analyzátoru záhlaví paketů publikovaném na odborné konferenci ANCS [4]. Díky tomu dosahuje implementace vysoké propustnosti a nízké latence při minimálním množství spotřebovaných zdrojů FPGA.

Návrh jednotky *Update* spravující záznamy o síťových tocích byl proveden s ohledem na budoucí rozšiřitelnost. Modulární architektura jednotky umožňuje kromě základních NetFlow statistik přidávání podpory dalších typů záznamů pro další (dosud neznámé) monitorovací metody. Pro vytváření nových modulů se počítá s využitím techniky HLS (High Level Synthesis), která urychlí jejich vývoj díky možnosti popisu na vyšší úrovni abstrakce ve formě jazyka C/C++.

3 ZÁVĚR

Zde představený firmware systému jsem v závěru loňského roku plně implementoval. V rámci další fáze vývoje jsem provedl také jeho funkční verifikaci. S využitím techniky HLS byly experimentálně vytvořeny další tři moduly pro sběr statistik o síťových tocích. Systém je navržen tak, aby dosahoval dostatečné výkonnosti pro potřeby monitorování 100 Gbps sítí a je také vhodný pro akceleraci analýzy aplikačních protokolů na těchto sítích. V současnosti je sestavení systému na cílovou akcelerační kartu podporující 100G Ethernet závislé na dostupnosti platformy NetCOPE pro tuto kartu. Základní funkčnost systému však byla ověřena na vývojové akcelerační FPGA kartě Fiberblaze FB8XG@V7690 [5]. Dále probíhá analýza reálné propustnosti systému a integrace se softwarovým řízením systému s cílem nasazení v praxi na reálné síti.

PODĚKOVÁNÍ

Tato práce vznikla v rámci projektu Velká infrastruktura CESNET pod označením LM2010005 podporovaného Ministerstvem školství, mládeže a tělovýchovy České republiky a za podpory projektu Architektury paralelních a vestavěných počítačových systémů VUT v Brně FIT-S-14-2297.

REFERENCE

[1] KEKELY, L. *Hardwarová akcelerace aplikací pro monitorování a bezpečnost vysokorychlostních sítí*. Diplomová práce. Brno: FIT VUT v Brně, 2013.

[2] MARTÍNEK, T., KOŠEK, M. NetCOPE: Platform for Rapid Development of Network Applications. *Proceedings of the 11th IEEE Workshop on Design and Diagnostics of Electronic Circuits and Systems*, DDECS 2008. Bratislava, SK, 2008, s. 1–6. ISBN 978-1-4244-2277-7.

[3] PAGH, R., RODLER, F. F. Cuckoo Hashing. *Proceedings of the 9th Annual European Symposium on Algorithms*, ESA 2001. Aarhus, DK: Springer, 2001, s. 121–133. ISBN 978-3-540-42493-2.

[4] PUŠ, V., KEKELY, L., KOŘENEK, J. Low-latency Modular Packet Header Parser for FPGA. *Proceedings of the 8th ACM/IEEE Symposium on Architectures for Networking and Communications Systems*, ANCS 2012. New York, USA: ACM, 2012, s. 77–78. ISBN 978-1-4503-1685-9.

[5] FIBERBLAZE. *Products: Unconfigured FPGA Cards* [online]. [cit. 2014-03-01]. URL: `http://www.fiberblaze.com/products/unconfigured-fpga-cards.html`