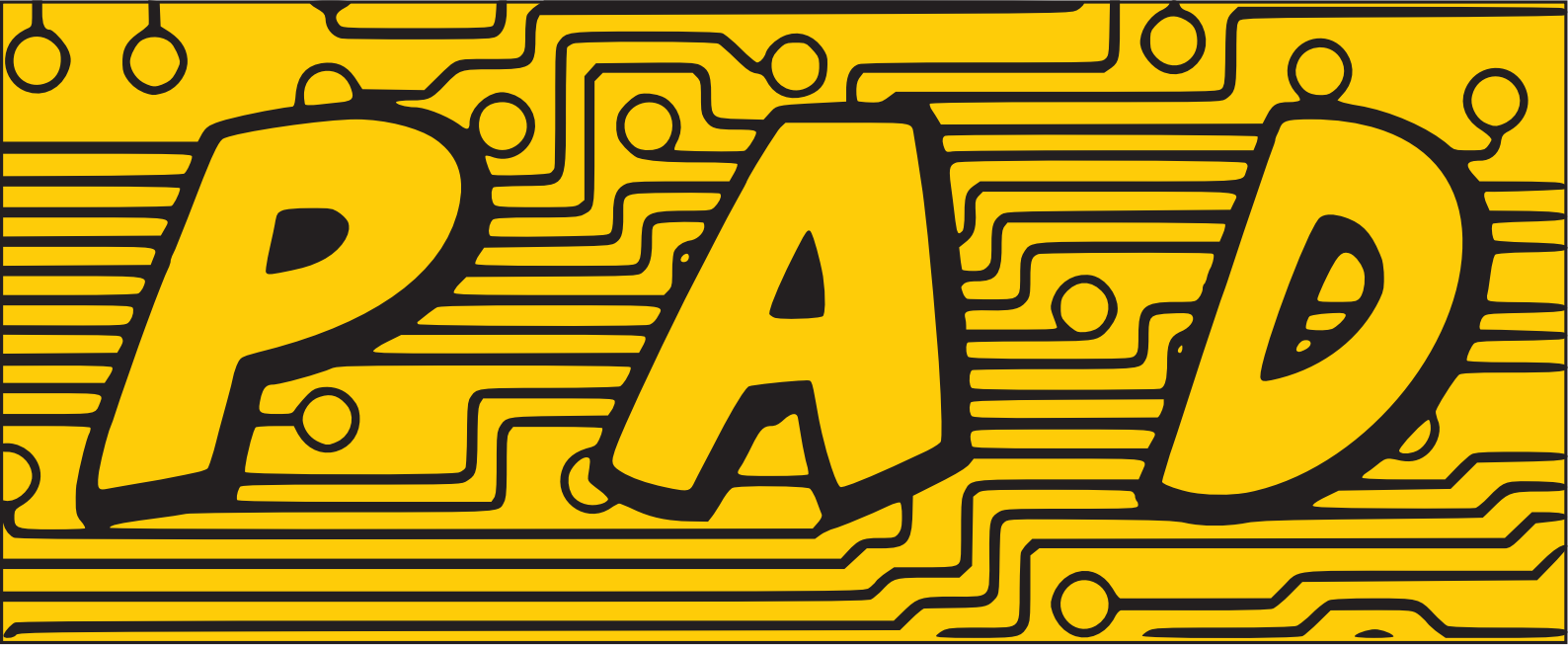


Sborník příspěvků

Počítačové architektury & diagnostika 2015

Česko-slovenský seminář pro studenty doktorského studia

Zlín - Fakulta aplikované informatiky, UTB ve Zlíně

2. - 4.9.2015

Počítačové architektury a diagnostika PAD 2015

česko-slovenský seminář pro studenty doktorského studia

Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky
2. – 4. 9. 2015

Sborník příspěvků

Fakulta aplikované informatiky

ISBN: 978-80-7454-522-1

Úvodní slovo

Prof. Jan Hlavička označoval návrh architektury a diagnostiku za dvojčata, protože jedno bez druhého postrádá smysl [1]. Obě disciplíny čerpají z poznatků více než sedmdesáti let úsilí o zdokonalování počítačů, cíl se nemění – cílem zůstává spolehlivost – mění se ale nástroje a prostředky, mění se technologie výroby, vlastnosti a způsob využití jak technických, tak programových prostředků počítačů. Seminář Počítačové architektury a diagnostika PAD 2015 se odehrává v prostředí moderní architektury města Zlín. Volba místa konání semináře není náhodná. Tak jako architektura města i architektura čipů dnes prodělává svoji obnovu. Nové trendy vývoje kladou důraz na nároky stávající architektury v nových podmínkách, na její spolehlivou funkčnost, vyšší bezpečnost provozu a menší energetické nároky.

V architektuře počítačů, podobně jako při výstavbě města, jsou patrné snahy využívat osvědčené stavební bloky. Nacházet novou funkčnost a nová propojení těchto bloků je rovněž vedeno snahou o zvyšování bezpečnosti provozu. Při řešení logistiky stavebních bloků města se skutečně setkáváme s problematikou, která má obdobu v systému na čipu. Na architektuře města se to projevuje např. tím, že se v ulicích mění provoz z dvojsměrného na jednosměrný. Propojení bloků do systému na čipu (SoC) se dnes již často neobejde bez techniky propojení pomocí sítě na čipu (NoC) – systém je sice o něco pomalejší, ale může být spolehlivější při využití kódového zabezpečení, které je pro bezpečný provoz sítí velmi propracované.

Podobných analogií je možné nalézt více, jednou z nich je například řešení nesouladu mezi rychlostí výpočetních jednotek a rychlostí pamětí: vede to k důmyslným konstrukcím vícestupňových pamětí. Paměti jsou účelově připojovány tak, aby se dosahovalo optimální doby čekání při ukládání a dostupnosti mezivýsledků při výpočtech. Ještě výraznější roli má tento požadavek při řešení systémů s více výpočetními jádry. Zde opět vystupuje do popředí diagnostika jako nezbytný předpoklad bezporuchového provozu systému na čipu.

Při hodnocení prací účastníků semináře PAD 2015 je aspekt spolehlivosti při provozu vyhodnocován jako prioritní. Tento přístup se osvědčuje i v projektech mobilních sítí 5G. Ty jsou dnes ve zkušebním provozu, ale během krátké doby budou každodenní realitou. To již bude většina dnešních přispěvatelů tohoto sborníku působit jako vedoucí projektů sítí 5G.

Podle zvyklostí z minulých let bylo vyhodnocení příspěvků na PAD 2014 provedeno pro jednotlivé ročníky studia. Byli oceněni tito studenti doktorského studijního programu:

1. ročník:
Ing. Lukáš Kekely – cena prof. Ing. Jana Hlavičky, DrSc. za vynikající výsledky v doktorském studiu,
Ing. Josef Kokeš – ocenění za výborné výsledky v doktorském studiu,
Ing. Adam Crha – ocenění za výborné výsledky v doktorském studiu,
Ing. Martin Kováč – ocenění za výborné výsledky v doktorském studiu.

2. ročník:
Ing. Miroslav Siebert – ocenění za výborné výsledky v doktorském studiu,

3. ročník:
Ing. Štefan Krištofík – cena prof. Ing. Jana Hlavičky, DrSc. za vynikající výsledky v doktorském studiu,
Ing. Marcela Šimková – ocenění za výborné výsledky v doktorském studiu.

Oceněným studentům blahopřejeme.

Ve Zlíně 2. září 2015

Za celý organizační výbor PAD 2015 Karel Vlček

[1] Hlavička, J.: Design and Diagnostics as Twin Disciplines, Proc. DDECS1995, pp. 9-13, ISBN 80-901751-9-8

Programový výbor

Dobai Roland	FIT VUT v Brně
Dohnal Jan	ON SEMI
Drábek Vladimír	FIT VUT v Brně
Dudáček Karel	FAV ZČU v Plzni
Fišer Petr	FIT ČVUT v Praze
Gramatová Elena	FIIT STU v Bratislavě
Jaroš Jiří	FIT VUT v Brně
Jelemenská Katarína	FIIT STU v Bratislavě
Jeníček Jiří	TU v Libereci
Kořenek Jan	FIT VUT v Brně
Koutný Tomáš	FAV ZČU v Plzni
Kotásek Zdeňek	FIT VUT v Brně
Kubátová Hana	FIT ČVUT v Praze
Lórencz Robert	FIT ČVUT v Praze
Novák Ondřej	FMIMS TU v Liberci
Pleštil Antonín	ASIC CENTRUM
Plíva Zdeněk	FMIMS TU v Liberci
Racek Stanislav	FAV ZČU v Plzni
Rozkovec Martin	FMIMS TU v Liberci
Růžička Richard	FIT VUT v Brně
Schmidt Jan	FIT ČVUT v Praze
Skrbek Miroslav	FIT ČVUT v Praze
Stopjaková Viera	FEI STU v Bratislavě
Strnadel Josef	FIT VUT v Brně
Vladimír Smotlacha	FIT ČVUT v Praze
Vavřička Vlastimil	FAV ZČU v Plzni
Vlček Karel	FAI UTB ve Zlíně
Zahradnický Tomáš	FIT ČVUT v Praze

Organizační výbor

Korbel Jiří
Knot Tomáš
Král Erik
Kunčar Aleš
Lebedová Jana
Matýsek Miroslav
Pokorný Pavel
Pospíšilík Martin
Prokopová Zdenka
Slovák Dalibor
Sysel Martin
Šilhavý Petr
Šilhavý Radek
Vlček Karel

Obsah

Na vybranou položku lze přejít kliknutím myší.

Evoluční návrh nízkopříkonových obvodů

Vojtěch Mrázek
Výpočetní technika a informatika, 1. ročník, prezenční forma
Školitel: Lukáš Sekanina, Specialista: Zdeněk Vašíček

Fakulta informačních technologií, Vysoké učení technické v Brně
Božetěchova 2, 612 66 Brno

imrazek@fit.vutbr.cz

Abstrakt. Práce se zabývá tématem snižování příkonu digitálních obvodů. Rozebírá výsledky aktuální práce v oblasti využití nekonvenčních metod pro snížení spotřeby integrovaných obvodů. Prvně je ukázán evoluční návrh obvodů na úrovni tranzistorů, kde se podařilo snížit náročnost návrhu. Díky tomu byly vytvořeny obvody s desítkami tranzistorů, což doposud nebylo pomocí evolučního návrhu možné. Dále byl tento přístup akcelerován v FPGA Zynq se zrychlením $4.7\times$. Byl navržen nový přístup k evoluční optimalizaci těchto obvodů s ohledem na příkon. Tato metoda využívá nový způsob odhadu spotřeby založený na pravděpodobnostním modelu. Mimo to jsou diskutovány možnosti návrhu s ohledem obvodů na úrovni hradel na spotřebu. Navíc je představen i způsob snižování spotřeby omezením funkčnosti, tzv. aproximační počítání. Tento příklad je demonstrován na ukázce výpočtu mediánu.

Klíčová slova. Příkon, VLSI, tranzistorová úroveň, hradla, aproximační počítání.

1 Úvod

Vzhledem k velkému rozmachu mobilních zařízení a další elektroniky, která pracuje nepřetržitě, stoupají požadavky na snižování spotřeby těchto zařízení. Jedny z prvků, které ovlivňují spotřebu, jsou digitální integrované obvody ASIC. Právě těmito obvody se budu dále více zabývat. Optimalizace spotřeby probíhá, stejně jako návrh, na více úrovních [10]. Nejnižší úrovní, která se zabývá příkonem, je mikroelektronika. Nastupují nové technologie výroby a se snižujícími se rozměry spotřeba klesá. Mění se však i další parametry obvodů navržených na moderních technologiích, a proto je musí návrhové systémy respektovat. Spotřebu můžeme ovlivnit mimo jiné napájecím napětím, taktováním, ale také i rozmístěním jednotlivých tranzistorů (tzv. layoutem). Velmi významný vliv má propojení jednotlivých tranzistorů a volba správných zapojení. Dále můžeme příkon optimalizovat na úrovni propojení mezi hradly a dalšími stavebními bloky obvodů. Spotřebu těchto obvodů můžeme upravovat i na nejvyšších úrovních, jako je optimalizace softwaru nebo úprava požadované funkčnosti s využitím aproximačního počítání.

Nižší úrovně, jako je úroveň technologická a úroveň masek, nebudou v práci rozebírány, protože k těmto možnostem optimalizace často nemá vývojář přístup. Kapitola 2 se zabývá optimalizací spotřeby na úrovni tranzistorů a jejich propojení. V kapitole 3 bude prezentován způsob snižování příkonu omezením funkčnosti. Možnosti pokračování práce a popsání celkového tématu disertace je možné nalézt v kapitole 4.

2 Návrh na úrovni tranzistorů

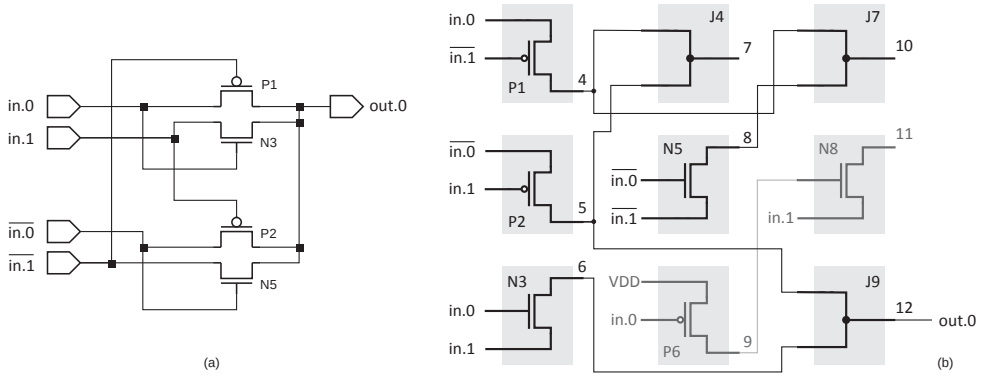
Tranzistorová úroveň popisu umožňuje významně optimalizovat příkon celého obvodu. Například pro implementaci čtyřvstupového obvodu AND-OR-INVERT může být ušetřeno 60 % prostředků při přechodu z popisu na úrovni hradel na úroveň tranzistorů. Při tvorbě jednotlivých bloků se na rozdíl od CMOS technologie nemusíme omezovat na ustálený způsob zapojení *komplemementární logiky CMOS*, ale můžeme využít řadu dalších možností. Tím může být například *Pass–Transistor logic* (normální nebo komplementární verze), *Single–Rail Pass–Transistor logic*, dynamické přístupy jako je *pseudo n-MOS* nebo diferenční přístup *Differential Cascade Voltage Switch logic* [7].

V posledních letech několik autorů ukázalo výhody techniky evolučního návrhu obvodů popsaných na úrovni tranzistorů. Tato metoda pracuje na principu generování a testu řady kandidátních řešení. Proto také výkonnost použitého simulátoru má významný vliv na škálovatelnost celého evolučního přístupu. Pro urychlení evoluce Žaloudek et al. navrhl přístup založený na jednoduchém simulátoru [13]. Díky nepřesnostem v simulaci nebyla řada nalezených řešení funkční v reálném prostředí. Jiný přístup navrhl Trefzer [8], který místo použití simulátoru použil rekonfigurovatelný analogový obvod. Nicméně bylo ukázáno, že přibližně 50 % nalezených řešení nebylo funkčních v přesném simulátoru SPICE. Později Walker et al. použil techniku hrubé síly pro evoluci obvodů s odolností vůči variabilitě výrobního procesu [11]. Pro evaluaci využil cluster přesných SPICE simulátorů.

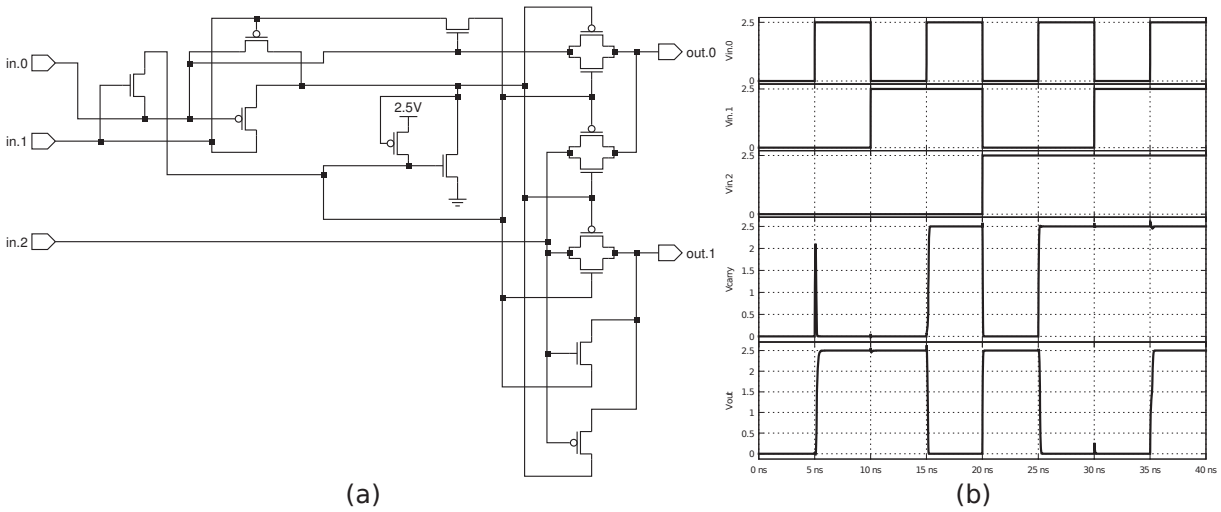
Jak je vidět, předcházející přístupy měly dva základní problémy. Prvním byl problém časové náročnosti simulace ve SPICE, kde model každého tranzistoru má stovky parametrů. Pokud se autoři snažili použít vlastní způsob simulace, projevovala se odchylka od reality. Při větších obvodech se tyto nedostatky začaly projevovat mnohem více a kvůli tomu se podařilo nalézt pouze velmi malé obvody s jednotkami tranzistorů.

2.1 Evoluční návrh obvodů

Vzhledem k náročnosti výpočtu bylo pro návrh obvodů nutné použít vlastní diskrétní simulátor. Tento simulátor vychází z vícehodnotové simulace a snaží se co nejvíce kopírovat chování tranzistorů včetně degradace signálů [12]. Pro návrh byla navržena vlastní reprezentace obvodů odvozená z kartézského genetického programování [1]. Jedná se o pevnou mřížku uzlů, kde každý uzel může plnit funkci *nmos* nebo *pmos tranzistoru*, nebo *propojky*. Bylo ukázáno, že tato reprezentace je dostačující a jsme schopni v ní definovat všechny obvody. Ukázka propojení je znázorněna na obrázku 1.



Obrázek 1: Příklad kandidátního obvodu implementující funkci XNOR s použitím osmi tranzistorů (čtyři jsou využity pro implementaci invertované hodnoty proměnných $\overline{in.0}$ a $\overline{in.1}$) v (a) schematické a (b) interní reprezentaci. Parametry dle [1] jsou následující: n_i=4 (0,V_{dd},in.0,in.1), n_o=1 (out.0), n_c=3, n_r=3, l=2. Chromozom: *(2,-3,pmos)(-2,3,pmos)(3,2,nmos)(4,5,junction)(-3,-2,nmos)(1,2,pmos)(4,8,junction)(9,3,nmos)(5,6,junction)(12).*



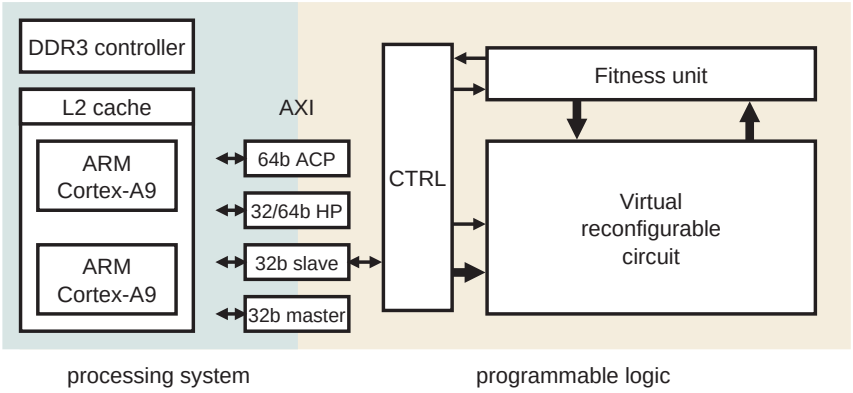
Obrázek 2: (a) Nejmenší a nejrychlejší nalezený obvod obsahující 14 tranzistorů, který plní funkci jednobitové sčítačky. (b) Výstupní signál získaný pomocí SPICE simulátoru.

Jako algoritmus evolučního návrhu byla použita evoluční strategie $(1 + \lambda)$ [1]. Pomocí této techniky se podařilo zlepšit úspěšnost návrhu obvodů na úrovni tranzistorů. Oproti předcházejícím pracím, kde se jednalo o obvody s jednotkami tranzistorů, se při použití nové techniky podařilo objevit obvody obsahující např. 25 tranzistorů. Také se podařilo vytvořit 14 tranzistorovou sčítačku (obrázek 2), která je téměř identická s řešením vytvořeným člověkem – expertem [6].

Další výsledky jsou dostupné v článku shrnujícím problematiku evolučního návrhu obvodů na úrovni tranzistorů [3].

2.2 Akcelerace evolučního návrhu

Pro evoluční návrh je výhodné, abychom byli schopni rychle ohodnotit kandidátní řešení. Výše popsaná metoda nabízí přímo možnost paralelizace na úrovni hardware. Protože se jedná o pevnou mřížku uzlů, je možné realizovat takzvaný virtuální rekonfigurovatelný obvod (VRC). Jedná se o obvod v FPGA, který obsahuje entity a dynamické přepínání funkce těchto entit a jejich propojení. Oproti existujícím VRC, které byly určeny pro kombinační logiku na úrovni hradel, musí náš VRC, určený pro úroveň tranzistorů, umožňovat modelovat tok proudu všemi směry. To komplikuje řešení dynamického propojení entity, takže je nutné implementovat spojování jednotlivých signálů do jednoho pomocí stromové redukce.



Obrázek 3: Architektura HW akcelerátoru v systému Zynq.

Další náročnou částí HW akcelerace je to, že evoluční algoritmus je poměrně složitý a obtížně implementovatelný v FPGA, zejména díky složitosti operace mutace. Aby nedošlo ke zpomalení evolučního návrhu, byl pro řízení použit výkonný procesor. S výhodou byla využita platforma systému na čipu Xilinx Zynq, který kombinuje dvoujádrový procesor ARM s taktem 1 GHz s FPGA částí. Rozložení komponent akcelerátoru je vidět na obrázku 3.

Při použití akcelerátoru dochází ke zrychlení evolučního návrhu pro obvody s 5 vstupy a 80 prvky přibližně $4.7\times$. Samotný akcelerátor bez použití evolučního algoritmu je dokonce $25\times$ rychlejší než diskrétní simulace. Zrychlení oproti SPICE simulátoru je dokonce více než tisícinásobné. Podrobné výsledky byly publikovány v článku [2].

2.3 Optimalizace obvodů s ohledem na příkon

Předcházející práce se většinou zabývaly optimalizací na počet tranzistorů. Je zřejmé, že na spotřebu mají vliv i další parametry, zejména přepínací aktivita jednotlivých tranzistorů. Proto bylo nutné navrhnout metodu, která bude správně odhadovat příkon obvodu popsaného na úrovni tranzistorů. Využívá se výsledků z diskrétní simulace, kde ke každému tranzistoru je možné určit četnost výskytu kombinace nastavení vstupu *source* a vstupu *gate*. Pro aktivní stavy se pak vypočítá pravděpodobnost přepnutí ze stavu a do stavu b (kde stav je dvojice hodnoty *source* a *gate*), která je dána

$$P_{a\to b} = 2 \cdot \frac{C_a}{2^i} \cdot \frac{C_b}{2^i}, \tag{1}$$

s tím, že i určuje počet vstupů a C_x určuje četnost stavu x. V simulátoru SPICE byla určena spotřeba pro každou možnou dvojici přepnutí a celková spotřeba tranzistoru je dána jako suma součinů pravděpodobností přechodů a změřených spotřeb pro danou dvojici přechodů. Pro tuto metodu byla na vzorku 200 obvodů zjištěna věrnost odhadu, která určuje to, že pokud je reálná hodnota spotřeby prvního obvodu větší, respektive menší, než spotřeba druhého, tak i odhadnutá spotřeba prvního obvodu musí být větší, respektive menší, než odhadnutá spotřeba druhého. U testovaných obvodů se věrnost odhadu pohybovala mezi 75 – 100 %.

Vzhledem k mírné chybě odhadu byly výsledky jednou za $1000--3000$ generací validovány pomocí simulátoru. Jako případová studie byla představena optimalizace čtyřbitových násobiček, které byly optimalizovány na úrovni hradel. Jedná se o osmivstupové obvody, které obsahují přibližně 300 tranzistorů. Ukázalo se, že touto metodou jsme schopni snížit příkon o 4 – 12 % s tím, že zpoždění obvodu zůstane přibližně zachováno, nebo se zlepší. Také se ukázalo, že je lepší používat metodu kombinující diskrétní i numerickou simulaci ve SPICE spíše než metodu používající pouze numerickou simulaci, protože tvoří variabilnější řešení, která nejsou tolik závislá na použité technologii. Výsledky jsou zpracovány v článku [4], který bude prezentován na konferenci EUC.

3 Systémové snižování příkonu

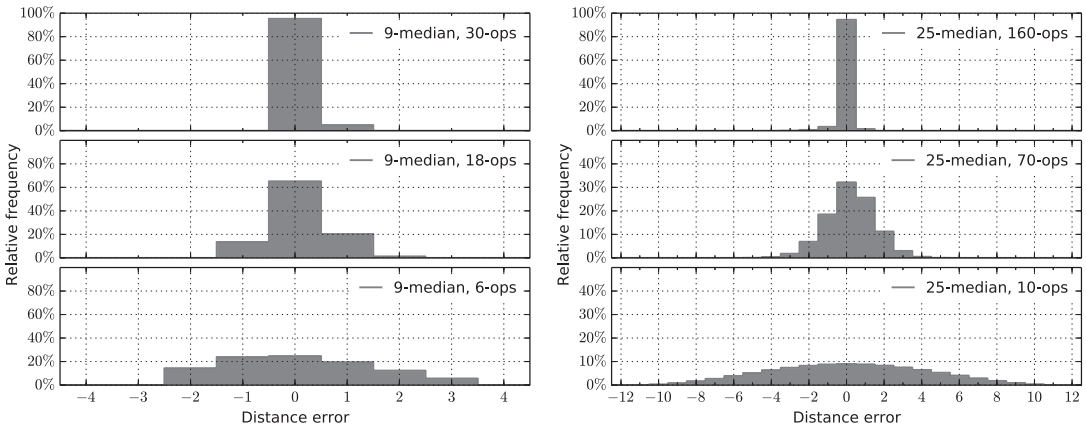
Dalším faktorem, kterým můžeme ovlivnit spotřebu, je úprava celkového chování systému. Ukazuje se, že v některých případech jsme ochotni snížit požadavky na funkčnost za cenu menší spotřeby a rychlejšího zpracování. Typickým příkladem jsou multimediální aplikace nebo jiné zpracování signálů. Celému přístupu se často říká *aproximační počítání*.

3.1 Návrh mediánového filtru

Ukázkou tohoto přístupu bude výpočet mediánu. Tato funkce byla aproximována pomocí kartézského genetického programování z optimálního řešení [9]. Při návrhu se používaly pouze bloky s funkcí *minimum* a *maximum*. Hlavním problémem návrhu je zjištění chyby aproximace. Ta se nejčastěji určuje

jako suma odchylek od plně funkčního řešení. Například pro algoritmu počítající medián z devíti osmibitových vstupů je potřeba k plnému ohodonocení $2^{8^9} = 2^{72}$ testů, což je neřešitelné. Proto se testy řeší náhodným výběrem vstupních vektorů. Potom je však chyba závislá na rozložení náhodného výběru.

Úkolem tedy bylo navrhnout metriku, která by správně vyhodnotila chybu. Vyjdeme z faktu, že díky použitému typu funkce je výsledek kandidátního řešení vždy jedním ze vstupních prvků. Další vlastností mediánu je to, že z množiny o velikosti $2n+1$ vybere právě $(i+1)$-tý nejmenší prvek. Proto při libovolné permutaci množiny $\{-n, -n+1, \ldots, 0, \ldots, n-1, n\}$ je validní medián 0. Hodnota, kterou aproximační medián vrátí při libovolné permutaci uvedené množiny, určuje odchylku pozice nejmenšího prvku. Tato metrika nám tedy určuje odchylku polohy mediánu a je nezávislá na vstupních datech. Výsledky pro různé aproximační mediány jsou vidět na obrázcích 4.



(a) Medián pro 9 vstupů

(b) Medián pro 25 vstupů

Obrázek 4: Histogram chyb v pozici mediánu pro jednotlivé aproximace. Hodnota OPS určuje počet operací *min / max* v aproximovaném řešení, kdy pro plně funkční 9 vstupový medián je potřeba 38 operací a pro 25 vstupů 220 operací.

3.2 Měření příkonu na mikroprocesorech

Dále bylo nutné prakticky ověřit, že použitím aproximace dochází ke zlepšení celkového příkonu. Proto byly aproximované mediány implementovány v mikroprocesorech. Jako ukázky jsem vybral osmibitový procesor akumulátorové architektury řady PIC16, dále šestnáctibitový procesor registrové architektury řady PIC24. Zástupce moderních procesorů řady ARM byl procesor STM32F4. Na těchto procesorech byl implementovaný medián a byl zkoumán vliv aproximace na spotřebu oproti plně funkčnímu řešení. Ukázalo se, že pro 9 vstupů a při povolení chyby o 1 pozici dochází k redukci příkonu o 21 %. U odchylky 2 pozic, kde správný medián nebyl určen v 35 % případů, je redukce příkonu 52 %. Podobně u 25-vstupého byl při odchylce o 5 pozic snížen příkon o 27 %. Kompletní výsledky jsou v článku na GECCO [5].

4 Cíl disertace

Předcházející kapitoly shrnují optimalizaci příkonu na tranzistorové úrovni a na úrovni aproximace funkce. Dalším místem optimalizace je úroveň hradel. Výsledky ve výzkumné skupině EHW@FIT ukazují, že evoluční algoritmy mají velký potenciál v optimalizaci obvodů na této úrovni. Zatím však jediným cílem optimalizace byla výsledná plocha na čipu. Ukazuje se však, že pro určení spotřeby je nutné obvod analyzovat detailněji. Na toto téma bylo publikováno mnoho literatury a možností odhadu

spotřeby je více. Celá disertační práce by měla ukázat, že evoluční metody návrhu digitálních obvodů jsou schopny efektivně optimalizovat spotřebu na různých úrovních — od tranzistorů, přes hradla, až po elementární softwarové metody. Kromě toho by měla nabídnout ucelený přehled metod, které je možné využít pro rychlý odhad příkonu.

5 Závěr

Práce se zabývá optimalizací spotřeby obvodů s využitím evolučních algoritmů. Byl představen nový přístup k návrhu obvodů na úrovni tranzistorů, který zlepšuje schopnost navrhnout obvody z jednotek na desítky tranzistorů. Tato metoda byla potom hardwarově akcelerována v čipu Xilinx Zynq. Byla také představena metoda odhadu příkonu těchto obvodů, se kterou jsme schopni optimalizovat spotřebu obvodů obsahující stovky tranzistorů. Práce také ukazuje možnosti snižování spotřeby na systémové úrovni, které jsou demonstrovány při aproximaci výpočtu mediánu. Ukázalo se, že při použití aproximací dochází k reálnému snížení spotřeby mikroprocesorů. Úspěšnost aproximace mediánu je vyjádřena novou metrikou založenou na pozici odchylky.

Poděkování Tato práce vznikla za podpory projektu FIT-S-14-2297 Architektury paralelních a vestavěných počítačových systémů.

Reference

[1] Miller, J. F.: *Cartesian Genetic Programming*. Springer Verlag, 2011.

[2] Mrazek, V.; Vasicek, Z.: Acceleration of transistor-level evolution using Xilinx Zynq Platform. In *Evolvable Systems (ICES), 2014 IEEE International Conference on*, Dec 2014, s. 9–16.

[3] Mrazek, V.; Vasicek, Z.: Evolutionary Design of Transistor Level Digital Circuits Using Discrete Simulation. In *Genetic Programming*, *LNCS*, ročník 9025, Springer, 2015, ISBN 978-3-319-16500-4, s. 66–77.

[4] Mrazek, V.; Vasicek, Z.: Automatic design of low-power arithmetic approximate VLSI circuits. In *13th IEEE International Conference on Embedded and Ubiquitous Computing, EUC*, 2015. V tisku., str. 8.

[5] Mrazek, V.; Vasicek, Z.; Sekanina, L.: Evolutionary Approximation of Software for Embedded Systems: Median Function. In *GECCO'15 Conference*, ACM, 2015, ISBN 978-1-4503-3488-4, s. 795–801.

[6] Shams, A.; Bayoumi, M.: A novel high-performance CMOS 1-bit full-adder cell. *IEEE Tr. on Circuits and Systems II: Analog and Digital Signal Processing*, ročník 47, č. 5, May 2000: s. 478–481, ISSN 1057-7130.

[7] Soudris, D.; Piguet, C.; Goutis, C.: *Designing CMOS Circuits for Low Power*. European low-power initiative for electronic system design, Springer, 2002, ISBN 9781402072345.

[8] Trefzer, M.: *Evolution of Transistor Circuits*. Dizertační práce, Ruprecht-Karls-Universität Heidelberg, 2006.

[9] Vasicek, Z.; Sekanina, L.: Evolutionary Approach to Approximate Digital Circuits Design. *IEEE Transactions on Evolutionary Computation*, ročník 19, č. 3, 2015: s. 432–444, ISSN 1089-778X.

[10] Venkatachalam, V.; Franz, M.: Power Reduction Techniques for Microprocessor Systems. *ACM Comput. Surv.*, ročník 37, č. 3, New York, NY, USA: ACM, Září 2005: s. 195–237, ISSN 0360-0300.

[11] Walker, J.; Hilder, J.; Tyrrell, A.: Towards evolving industry-feasible intrinsic variability tolerant CMOS designs. In *IEEE Congress on Evolutionary Computation*, 2009, s. 1591–1598.

[12] Weste, N. H.; Harris, D.: *CMOS VLSI design: a circuits and systemes perspective*. Boston, USA: Addison-Wesley, třetí vydání, 2005, ISBN 0-321-14901-7, 968 s.

[13] Zaloudek, L.; Sekanina, L.: Transistor-Level Evolution of Digital Circuits Using a Special Circuit Simulator. In *Evolvable Systems: From Biology to Hardware*, *LNCS*, ročník 5216, Springer Verlag, 2008, s. 320–331.

Využití funkční verifikace pro ověřování metodik pro zajištění odolnosti proti poruchám

Jakub Podivínský
Informatika a výpočetní technika, druhý ročník, prezenční studium
Školitel: Zdeněk Kotásek

Fakulta informačních technologií, Vysoké učení technické v Brně
Božetěchova 2, 612 66 Brno
ipodivinsky@fit.vutbr.cz

Abstrakt. Náplní tohoto článku je představení práce zabývající se aplikací funkční verifikace pro testování metodik pro zajištění odolnosti proti poruchám v systémech založených na FPGA. Je zde představena problematika obvodů FPGA, vlivu poruch a možnosti jejich eliminace. Na základě těchto poznatků jsou formulovány cíle práce a návrh jejich dosažení. Uveden je také přehled dosažených výsledků při řešení této práce. Mezi ně patří experimentální elektromechanický systém, první verze platformy pro testování metodik pro zajištění odolnosti proti poruchám a zejména verifikační prostředí pro procesor běžící na FPGA, který bude tvořit jádro dalšího experimentálního systému. Dosavadní výzkumná práce je shrnuta v časopise Microprocessors and Microsystems [1].

1 Úvod a motivace

Číslicové systémy hrají stále větší roli v našem každodenním životě, setkáváme se s nimi v nejrůznějších aplikacích. Jsou hojně využívány v průmyslové výrobě, používají se jako řídicí systémy v dopravních prostředcích, medicíně, telekomunikacích a podobně. Současný trend je přesouvání stále větší zodpovědnosti právě na číslicové řídicí systémy, což obvykle vede ke snížení hmotnosti mechanické části a tím i snížení provozních nákladů, například v letectví nebo automobilismu [2]. Důsledkem je, že používané číslicové systémy jsou stále komplexnější, což vede k neustálému růstu míry jejich integrace. Tento jev má za následek větší náchylnost takových systémů k poruchám, a to jak k poruchám vzniklým při návrhu a implementaci systému, tak k poruchám vzniklým za provozu systému. Výskyt takové poruchy může mít nedozírné následky, a to nejen ve formě finančních ztrát, ale může dojít i k ohrožení lidských životů (např. porucha v řídicím systému letadla). Pro bezpečnostně kritické aplikace je tedy velmi důležité hledat cesty, jak zajistit, aby vznik poruch nijak neohrozil provoz těchto systémů. Tato práce se zabývá tvorbou platformy pro ověřování metodik pro zajištění odolnosti proti poruchám (*FT - Fault Tolerance*) v systémech založených na FPGA. Jako základní vyhodnocovací mechanismus poslouží funkční verifikace.

2 Současný stav poznání

Před formulací cílů práce je třeba krátce uvést do problematiky FPGA, odolnosti proti poruchám a funkční verifikace.

2.1 Obvody FPGA a výskyt poruch

Programovatelná hradlová pole (*FPGA*) jsou obvody, které je možné programovat jak před použitím, tak za běhu aplikace bez přerušení činnosti ostatních částí obvodu pomocí *částečné dynamické rekonfiguraci* (PDR – Partial Dynamic Reconfiguration) [3]. FPGA jsou stále populárnější a nacházejí uplatnění v řadě aplikací, především

díky zmíněné programovatelnosti, snadnému návrhu, flexibilitě, snižující se spotřebě ale také klesajícím cenám. FPGA se skládá z matice konfigurovatelných logických bloků (*CLB*), které jsou propojeny pomocí programovatelné propojovací sítě. Mimo CLB obsahují i řadu dalších prvků. Konfigurace jednotlivých bloků a propojovací sítě je uložena v paměti SRAM ve formě bitové posloupnosti (tzv. *bitstream*). V současné době jsou nejpoužívanější FPGA s konfigurací uloženou v paměti SRAM.

Poruchy v FPGA typicky vznikají vlivem vysoce energetických částic [4]. Dopad takové částice na FPGA může způsobit nežádoucí zákmit na přenášeném signálu, což je efekt označovaný jako *Single Event Transient* (SET). Při zásahu paměťové buňky může dojít ke snížení napětí, což vede ke změně uložené hodnoty. Tento efekt se nazývá *Single Even Upset* (SEU) a je to nejčastější porucha postihující FPGA obvody.

2.2 Implementace a ověřování odolnosti proti poruchám

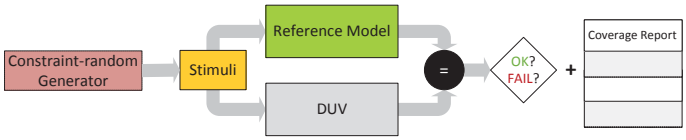
Při zajišťování odolnosti proti poruchám můžeme použít vhodnou modifikaci základních typů *redundance* (*hardwarová, časová, informační a programová redundance*). Základní technikou pro detekci a maskování poruchy je technika TMR (*Triple Modular Redundancy*), která je schopna díky ztrojení funkčních bloků maskovat jednu poruchu. Následná částečná dynamická rekonfigurace umožňuje opravit poruchou napadenou část konfigurační paměti FPGA a obnovit bezporuchový stav.

Čekat na přirozený výskyt poruch (SEU) je velmi neefektivní. Omezující jsou zde parametry MTTF (*Mean Time To Failure*) a MTBF (*Mean Time Between Failures*), které se mohou pohybovat i v řádech několika let, proto je nutné tyto poruchy vhodným způsobem simulovat. Simulační metoda pro emulaci vlivu SEU poruch v konfigurační paměti FPGA je představena v [5]. Autoři kombinují simulaci a topologickou analýzu systému, který je implementován v FPGA. Nástroj pro injekci poruch do FPGA je náplní článku [6]. Podporuje různé modely poruch použitelné v FPGA, které jsou implementovány ve VHDL, ale je třeba modifikovat původní návrh a doplnit další hradla a spoje pro injekci poruch. Techniky založené na injekci poruch do reálného FPGA bez nutnosti změny původního systému jsou představeny v [7]. Jsou založeny na PDR, která umožňuje přečíst část aktuální konfigurační paměti, invertovat zvolený bit a následně zapsat tuto část zpět do paměti. Výzkumem v oblasti injekce poruch se také zabývají někteří členové týmu doc. Kotáska. Ing. Jan Kaštil a další prezentují v [8] externí injektor poruch do FPGA. Tento injektor je založen na generování poruch mimo FPGA (generování probíhá v PC), tedy není omezen na konkrétní desku s FPGA čipem.

2.3 Funkční verifikace číslicových systémů

Funkční verifikace ověřuje, zda systém odpovídá specifikaci monitorováním jeho vstupů a výstupů v simulačním prostředí (např. *ModelSim*). Jedná se o rozšířenou sofistikovanější verzi běžně používaných *testbench*. Pro usnadnění tvorby verifikačních prostředí existuje standardizovaný jazyk *SystemVerilog* [9], metodika UVM (*Universal Verification Methodology*) [10] a knihovny metodiky UVM.

Verifikovaný systém je na Obrázku 1 označen jako DUV (*Device Under Verification*), výstupy tohoto systému jsou porovnávány s referenčním modelem (*Reference Model*). Pokud je zjištěna neshoda mezi výstupy DUV a referenčního modelu, znamená to, že jejich funkce nejsou ekvivalentní. Výstupem funkční verifikace je také zpráva o pokrytí klíčových funkcí verifikovaného systému (*Coverage Report*), která nám říká, jak důkladně byly ověřeny jednotlivé specifikované funkce. Pro získání vysokého pokrytí je třeba zajistit přísun dostatečného počtu vhodných vstupů (*Stimuli*). To je úkolem generátoru, který tyto vstupy generuje (*Constraint-random Generator*).



Obrázek 1: Základní princip funkční verifikace.

3 Cíle disertační práce a návrh zvoleného řešení

Ze studia aktuálního stavu poznání v oblasti FT systémů založených na FPGA vyplynula potřeba nalézt odpověď na několik otázek: *Jaké budou výsledky FT metodik na reálných systémech? Lze spoléhat na to, že ne všechny poruchy v elektronické části systému se projeví na chování řízené mechanické aplikace? Lze využít funkční verifikaci pro ověřování vlivu poruch na FT systémy?* Na základě těchto otázek byly formulovány cíle disertační práce:

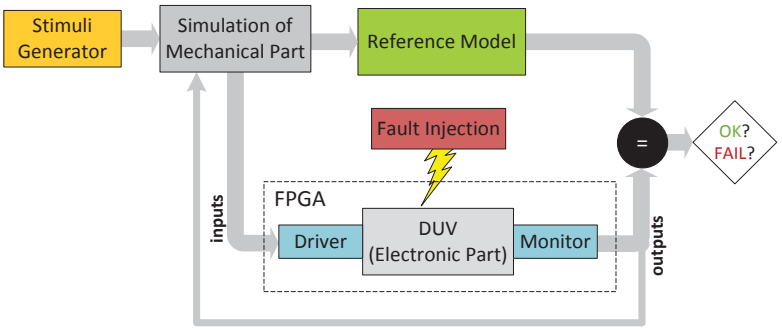
1. Navrhnout a vytvořit platformu, která bude založená na technologii FPGA a bude sloužit k testování FT metodik a k sledování vlivu poruch nejen na výstup elektronické části, ale také na řízenou mechanickou aplikaci. Bude využita technika funkční verifikace a injektor poruch vyvinutý týmem doc. Kotáska. Jádrem platformy bude experimentální elektromechanická aplikace.
2. V návaznosti na vytvořenou testovací platformu navrhnout proces ověřování FT metodik s přihlédnutím ke specifikům elektromechanických systémů. Tyto postupy budou využívat navrženou a vytvořenou testovací platformu. Proces bude navržen na základě experimentování s vytvořenou platformou. Součástí bude popis činností před zahájením procesu ověřování. Proces bude ověřen a demonstrován na dalším experimentálním systému. Tím dojde k zobecnění získaných výsledků.

3.1 Platforma pro ověřování FT metodik a proces ověřování FT metodik

Platforma bude koncipována jako rozšířené prostředí pro funkční verifikaci (Obrázek 2) a umožní ověřovat FT na experimentálních elektromechanických systémech. V uvedeném schématu lze nalézt jak elektronickou část (*Electronic Part*), tak mechanickou část (*Mechanical Part)* experimentálního systému. Elektronická část zde reprezentuje verifikovaný obvod (DUV), rozdíl oproti klasické verifikaci je umístění DUV na FPGA, což s sebou nese potřebu komunikace mezi FPGA a PC. Součástí každého verifikačního prostředí je referenční model (*Reference Model*), jehož výstup je porovnáván s výstupem DUV, které bude, díky použití injektoru poruch (*Fault Injector*), vystaveno působení poruch. Jelikož se zaměřujeme na ověřování metodik cílených na FPGA, poruchy budou injektovány pouze do FPGA (typicky SEU poruchy) a nikoliv do ostatních elektronických prvků (čidla atd.). Samozřejmě bude třeba vygenerovat takové stimuly, které povedou k dostatečnému funkčnímu pokrytí, což zajistí generátor stimulů (*Stimuli Generator*). V případě elektromechanické aplikace se jedná o různé konfigurace experimentální elektromechanické aplikace (různé verifikační scénáře). K vyhodnocení míry funkčního pokrytí lze s výhodou použít nástroje pro funkční verifikaci.

Postup ověřování FT metodik je rozdělen na tři fáze. V *první fázi* je potřeba vytvořit verifikační prostředí, včetně referenčního modelu, pro elektronickou řídicí jednotku. Je třeba vzít v úvahu mechanickou část tak, aby byla součástí verifikačního prostředí. V první fázi se nijak nevyužívá FPGA. Odhalují se chyby v implementaci tak, aby nebyly později zaměněny s projevy injektovaných poruch. Výsledkem této fáze bude sdělení, zda elektronická řídicí jednotka správně zpracuje vygenerované verifikační scénáře a také sada použitých scénářů, pro které je zaručené správné chování. *Druhá fáze* již využívá verifikaci na FPGA, což umožňuje použít injektor poruch. Vstupem

jsou ověřené verifikační scénáře, každý je několikrát opakován, při každém opakování je injektována jiná porucha či sada poruch a jsou sledovány jejich projevy. Výstupem je seznam poruch, které v kombinaci s daným verifikačním scénářem způsobily nesprávný výstup elektronického řadiče. Tyto poruchy budou detailněji analyzovány v *třetí fázi*. Každá injektovaná porucha tak bude zařazena do jedné z těchto kategorií: (1) Výstup z DUV a referenčního modelu se shoduje, porucha se neprojevila. (2) Výstupy se neshodují, navzdory tomu mechanická část splnila svůj úkol. (3) Výstupy se neshodují a zároveň došlo k nesplnění cíle mechanické část. Poruchy s vlivem na mechanickou část budou detailněji analyzovány. Důležitou fází při návrhu procesu ověřování bude navržení vhodných kombinací poruch se zaměřením na ověřovanou FT metodiku. Například při ověřování TMR bude zajímavé chování při výskytu stejné poruchy ve dvou kopiích stejného funkčního bloku.



Obrázek 2: Využití funkční verifikace pro testování odolnosti proti poruchám.

4 První verze testovací platformy

Robot pro hledání cesty v bludišti byl zvolen jako *první experimentální elektromechanický systém*. Elektronická část je zde reprezentována řídicí jednotkou robota implementovanou v FPGA, zatímco mechanickou část reprezentuje robot v bludišti. Není použit reálný robot, ale robot a jeho prostředí jsou pouze simulovány pomocí volně dostupného simulačního prostředí *Player/Stage* [11], což umožňuje velmi snadno měnit verifikační scénáře, tedy počáteční a cílovou pozici a bludiště, které robot prochází. Ověřované FT metodiky budou aplikovány na elektroniku v FPGA, tedy na řídicí jednotku robota. Tato řídicí jednotka je navržena tak, aby reprezentovala komplexní systém obsahující nejrůznější aspekty návrhu číslicových systémů (sekvenční a kombinační obvody, konečné automaty, paměti a sběrnice), což umožní testovat různě zaměřené metodiky.

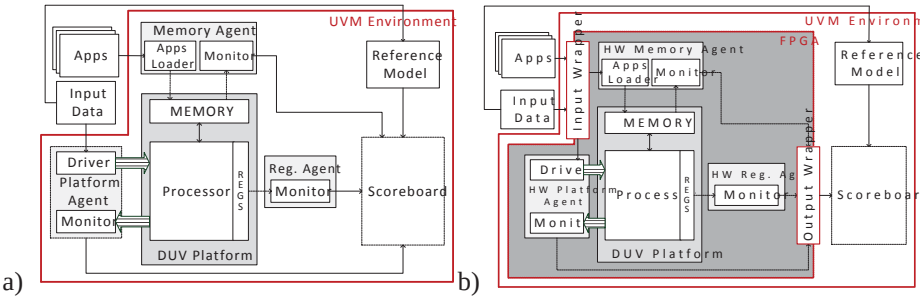
Byla implementována *první verze platformy*, prozatím bez aplikace principu funkční verifikace. Experimentálním systémem v první verzi platformy se stal zmíněný robot pro hledání cesty v bludišti. Platformu tvoří dva základní bloky - počítač a FPGA deska. Tyto bloky jsou spolu propojeny pomocí dvou komunikačních kanálů, první je rozhraní JTAG využívané injektorem poruch, druhý komunikační kanál slouží pro přenos dat mezi simulačním prostředím robota a jeho řídicí jednotkou. První verze platformy je rozdělena na 3 základní funkční části: (1) vývojová deska ML506 s FPGA Virtex 5, kde je implementována řídicí jednotka, (2) simulační prostředí *Player/Stage* [11] pro simulaci robota a kontrolu reakcí mechanické části na pokyny řídicí jednotky běžící na PC a (3)injektor poruch, který také běží na PC, umožňuje vkládat poruchy do řídicí jednotky robota [8].

5 Verifikační prostředí pro procesor běžící na FPGA

V další práci bych se chtěl zaměřit také na vytvoření jiného experimentálního elektromechanického systému. V současné době jsou k řízení nejrůznějších systémů používány procesory, můžeme se s nimi setkat v automobilech, letadlech a podobně. Chtěl bych se tedy zaměřit i na elektromechanický systém, kde bude elektronická část tvořena procesorem implementovaným na FPGA. To umožní ověřovat FT metodiky zaměřené nejen na číslicové systémy, ale také tzv. *Software Fault Tolerant* techniky. Rozhodl jsem se použít Codix RISC procesor [12], který je možné vygenerovat pomocí nástroje Codasip Studio (FIT vlastní akademickou licenci). Jedná se o 32-bitový procesor typu RISC (*Reduced Instruction Set Computer*) se 7 stupni zřetězeného zpracování, 32 volně použitelnými registry, 512 kB paměti a instrukční sada obsahuje 59 instrukcí.

S odkazem na *první fázi* procesu ověřování FT metodik je nutné nejprve vytvořit *prostředí pro funkční verifikaci pro procesor* a spustit verifikaci bez injekce poruch. Verifikační prostředí podle standardu UVM ukazuje Obrázek 3a, toto verifikační prostředí je implementováno v jazyce SystemVerilog a lze jej automaticky generovat pomocí nástrojů Codasip Studia.

Druhá fáze v procesu ověřování kvality metodik pro zajištění odolnosti proti poruchám je funkční verifikace systému implementovaného v FPGA, v tomto případě se jedná o procesor běžící na FPGA. V této fázi se také dostává ke slovu injekce poruch do FPGA. Pro tyto účely je třeba upravit původní verifikační prostředí tak, aby procesor běžel na FPGA, takové verifikační prostředí ukazuje Obrázek 3b. Je třeba poznamenat, že téměř všechny UVM komponenty byly přesunuty do FPGA, mimo referenční model (*Reference Model*) a porovnání výstupů (*Scoreboard*). UVM agenti a jejich vnitřní komponenty jsou nahrazeny HW agenty v FPGA. Komunikace mezi softwarovou a hardwarovou částí verifikačního prostředí je zajištěna použitím proprietárního rozhraní. Výstupy referenčního modelu jsou porovnávány s výstupy DUV, které jsou získány z HW prostřednictvím komponenty *Output Wrapper*. Porovnává se obsah paměti a registrového pole po provedení každého programu a také data na výstupním portu procesoru. Komponenta *Input Wrapper* slouží k odesílání programů, které vykonává procesor, a vstupních dat.



Obrázek 3: Verifikační prostředí pro procesor a) čistě SW verze, b) rozšíření o FPGA.

Při experimentování s akcelerovaným verifikačním prostředím bylo zjištěno, že verifikační prostředí akcelerované pomocí FPGA potřebuje pro splnění stejného úkolu kratší čas. Bylo dosaženo zrychlení přibližně 1,7x, což je horší výsledek, než bylo očekáváno. Je to způsobeno zejména komplexním referenčním modelem, který běží pomaleji, než DUV na FPGA. Nicméně pro účely testování metodik pro zajištění odolnosti proti poruchám není míra zrychlení důležitá. Významné je, že procesor běží na FPGA a je zajištěna komunikace se SW stranou, což umožňuje využití v navrhované platformě.

6 Závěr

V tomto článku byla představena základní myšlenka disertační práce zabývající se využitím funkční verifikace pro testování FT metodik v systémech založených na FPGA. Byl zde představen úvod do řešené problematiky, který položil základy pro formulaci cílů disertační práce. Na základě představených cílů byl nastíněn způsob řešení. Druhá část článku se zabývala prezentací již realizovaných výsledků, mezi které patří implementace experimentálního elektromechanického systému, první verze platformy pro testování, prozatím bez aplikace funkční verifikace. Představen byl také koncept akcelerace funkční verifikace procesoru pomocí FPGA, nebylo dosaženo příliš dobrých výsledků ve zrychlení verifikace, ale představený procesor běžící na FPGA bude použit jako základ dalšího experimentálního systému. Jak již bylo řečeno, dosavadní výsledky jsou shrnuty v časopise Microprocessors and Microsystems [1].

Další práce bude směřována především k rozšíření první verze platformy o techniky funkční verifikace, bude tedy vytvořeno verifikační prostředí pro řídicí jednotku robota běžící na FPGA. Toto verifikační prostředí bude dále rozšířeno o injekci poruch do FPGA, což umožní provádění dalších rozsáhlejších experimentů.

Reference

[1] Podivinsky, J.; Cekan, O.; Simkova, M.; Kotasek, Z.: The evaluation platform for testing fault-tolerance methodologies in electro-mechanical applications, Microprocessors and Microsystems, 2015, ISSN 0141-9331, Přijato k publikaci, http://dx.doi.org/10.1016/j.micpro.2015.05.011.

[2] Leen, G.; Heernan, D.: Expanding automotive electronic systems. Computer, ročník 35, č. 1, 2002. ISSN 0018-9162, s. 88-93.

[3] XILINX: Partial Reconfiguration User Guide.

[4] Ceschia, M.; Violante, M.; Reorda, M.; aj.: Identification and classification of single-event upsets in the configuration memory of SRAM-based FPGAs. *IEEE Transactions on Nuclear Science*, ročník 50, č. 6, 2003. ISSN 0018-9499, s. 2088-2094.

[5] Bernardeschi, C.; Cassano, L.; Domenici, A.; aj.: Accurate simulation of SEUs in the configuration memory of SRAM-based FPGAs. In *IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems* (DFT), 2012. s. 115-120.

[6] Rudrakshi, S.; Midasala, V.; Bhavanam, S.: Implementation of FPGA based fault injection Tool (FITO) for testing fault tolerant designs. *IACSIT International Journal of Engineering and Technology*, ročník 4, č. 5, 2012. s. 522-526.

[7] Alderighi, M.; Casini, F.; d'Angelo, S.; aj.: Evaluation of single event upset mitigation schemes for SRAM based FPGAs using the FLIPPER fault injection platform. In *IEEE International Symposium on Defect and Fault-Tolerance in VLSI Systems*, 2007. s. 105-113.

[8] Straka, M.; Kastil, J.; Kotasek, Z.: SEU Simulation Framework for Xilinx FPGA: First Step Towards Testing Fault Tolerant Systems. In *Conference on Digital System Design*, IEEE Computer Society, 2011. ISBN 978-0-7695-4494-6, s. 223-230.

[9] IEEE Std. 1800-2005, *IEEE Standard for SystemVerilog - Unified Hardware Design, Specification, and Verification Language*. 2005.

[10] Rosenberg, S.; Meade, K.: *A practical guide to adopting the universal verification methodology (UVM).* Cadence Design Systems, 2013.

[11] Gerkey, B.; Vaughan, R. T.; Howard, A.: The player/stage project: Tools for multi-robot and distributed sensor systems. In *Proceedings of the 11th international conference on advanced robotics*, ročník 1, 2003, s. 317-323.

[12] Codasip: Codix RISC. Květ. 2015. URL <https://www.codasip.com/products/codix-risc/>

Principy generování verifikačních stimulů

Ondřej Čekan

Informatika a výpočetní technika, druhý ročník, prezenční studium
Školitel: Zdeněk Kotásek

Fakulta informačních technologií, Vysoké učení technické v Brně
Božetěchova 2, 612 66 Brno
icekan@fit.vutbr.cz

Abstrakt. Tento článek pojednává o tématu disertační práce a shrnuje aktuální stav poznání v oblasti generování stimulů založeného na omezujících podmínkách. Je zde představen návrh a základní parametry generátoru stimulů, který je vhodný pro použití především ve funkční verifikaci číslicových systémů. V rámci článku jsou rovněž definovány cíle disertační práce a dosavadní práce představující návrh a realizaci univerzálního generátoru stimulů, který je použit v praxi pro generování assemblerovských programů pro procesory. Výzkumná práce je shrnuta v časopise Microprocessors and Microsystems [1].

1 Úvod

Elektronické obvody se dostávají stále více do popředí našeho každodenního života a dalo by se říci, že od 21. století si bez nich svět již nedokážeme představit. Mikrokontroléry, které mají na starost řízení určitých zařízení, se nacházejí dnes i tam, kde bychom to ani nečekali. Jedná se o hračky, kuchyňské spotřebiče, chytré přívěsky nebo dokonce šperky. Takovéto použití představuje spíše high-end současné doby a obvody, které takovéto zařízení řídí, nejsou abnormálně spolehlivé. Na druhou stranu ale existují aplikace, u nichž jakýkoli nedostatek v návrhu nebo ve funkci systému může znamenat vážné riziko a v ohrožení se nacházejí především lidské životy. Takovéto aplikace se označují jako kritické z hlediska bezpečnosti (safety-critical) a představují je oblasti automobilového průmyslu, letectví, kosmonautiky či medicíny.

V případě, že cílíme na systémy, které neobsahují návrhové ani implementační chyby, které by mohly způsobit nekorektní chování, musí být tyto systémy důkladně otestovány. V úvahu se berou obvyklé, ale i neobvyklé kombinace vstupních hodnot, které mohou v daném systému nastat. Jelikož neustále roste složitost systému, tak roste i složitost spojená s důkladným ověřením jeho správnosti [2]. Jednoduché systémy není složité otestovat manuálně. U komplexnějších systémů je manuální testování velmi časově náročné. Rovněž doposud vyvinuté formální techniky pro formální ověření rozsáhlých systémů selhávají. Z tohoto důvodu byla vytvořena technika zvaná funkční verifikace, která na základě vstupních a výstupních hodnot ze systémů ověřuje jejich korektnost. Hodnoty, které vstupují do systému, se typicky získávají pomocí generátoru stimulů (testů), který je předmětem disertační práce a tedy i tohoto článku.

2 Současný stav poznání

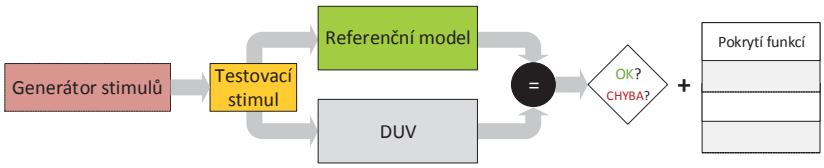
V této kapitole je shrnut současný stav poznání, se kterým bylo nutné se seznámit a na který je navazováno v rámci tématu disertační práce. Jedná se o popis funkční verifikace a problému s omezujícími podmínkami.

2.1 Funkční verifikace

Funkční verifikace [3] je proces, při kterém dochází k ověření správnosti funkce systému vzhledem k jeho specifikaci tím, že jsou nastavovány jeho vstupy a sledovány

jeho výstupy. Funkční verifikace probíhá v simulaci, a tudíž neposkytuje důkaz o korektnosti systému. Oproti tomu využívá přídavné techniky, čímž simulaci zefektivňuje.

Funkční verifikace je založena na dvou systémech, které paralelně testujeme se stejnými vstupními daty. První systém je hardwarové zařízení popsané v HDL (Hardware Description Language) jazyce, označované jako DUT (Device Under Test) či DUV (Device Under Verification), které ověřujeme na správnost vzhledem ke specifikaci. Druhý systém je model ověřovaného systému, který odpovídá stejné specifikaci a typicky bývá implementován v jiném programovacím jazyce. Model rovněž bývá typicky implementován jiným vývojářem, aby se zamezilo stejným implementačním chybám a stejným chybám ve špatně pojaté specifikaci. Model bývá označován jako *referenční* či *golden* model. Na vstup těchto dvou systémů je tedy přiveden stejný testovací stimul, který je typicky získán pomocí generátoru testovacích stimulů. Výstupy těchto systémů jsou porovnány na rovnost. Výstupem funkční verifikace je výsledek porovnání výstupů obou systémů a také informace o pokrytí (coverage) [4] klíčových funkcí systému. V rámci simulačního prostředí je možno definovat metriky a podmínky (klíčové funkce), které mají být v daném systému sledovány. V případě naplnění všech definovaných podmínek je dosaženo 100% pokrytí funkcí systému, což je z hlediska funkční verifikace klíčové. Tyto klíčové funkce se typicky volí tak, aby byly pokryty veškeré větve HDL kódu systému, všechny jeho stavy, hraniční hodnoty vstupních dat a podobně. Výše popsaný princip je znázorněn na Obr. 1.



Obr. 1: Princip funkční verifikace.

2.2 Problém s omezujícími podmínkami

Problém s omezujícími podmínkami **Chyba! Nenalezen zdroj odkazů.** neboli v angličtině *Constraint Satisfaction Problem* (CSP) je obecný matematický problém, který je definován množinou proměnných, které mohou nabývat hodnot z konečné, neprázdné a diskrétní domény, a množinou omezení (tzv. constraints). Každé omezení je definováno nad určitou podmnožinou proměnných, pro něž z dané domény specifikuje platné hodnoty, které mohou nabývat. Výsledkem řešení problému s omezujícími podmínkami je jedno nebo všechna přiřazení hodnot proměnným tak, že jsou splněna všechna omezení.

Definice 1. *Nechť X je množina proměnných, D je doména hodnot a C je množina omezení. Pak CSP je definován jako trojice (X,D,C), kde pro každé c ϵ C existuje dvojice (t,R), kde t je n-tice proměnných a R je n-ární relace nad D.*

Mezi typické příklady CSP patří problém N dam nebo problém obarvení grafu.

2.3 Generování testovacích stimulů založené na omezujících podmínkách

Jak bylo ukázáno v podkapitole o funkční verifikaci, generátor testovacích stimulů zde hraje významnou roli. Nalezení vhodných testovacích stimulů může ulehčit celému procesu funkční verifikace a tím urychlit vývoj systému. Generování testovacích stimulů se dá rozdělit na dvě hlavní kategorie, manuální generování a automatické generování [6].

Manuální generování představuje inženýrskou činnost v podobě ručního vytváření testů. Verifikační inženýři detailně rozumí návrhu a struktuře verifikovaného systému,

a tudíž se dokáží zaměřit na úskalí systému, na jeho mezní hodnoty, přechody mezi stavy a podobně. Manuální tvorba testů ale představuje značně časově náročnou činnost, jejíž výsledky nemusí být vždy dostačující.

Druhou možností je využití podpůrných programů, které dokáží vytvářet testy automaticky. Takovéto programy jsou založeny především na řešení problému s omezujícími podmínkami, jejichž řešením je platný testovací stimul pro daný systém. Velkou výhodou automaticky generovaných testů je jejich náhodnost a rychlost, díky kterým se ověří i atypické a nepředvídatelné kombinace vstupních hodnot, které v daném systému mohou nastat a které při manuálním vytváření zpravidla nevznikají.

V praxi se používá kombinace obou přístupů generování, kde je nejprve systém otestován na malé množině manuálně vytvořených testů a až poté je verifikován na automaticky generovaných testech tak dlouho, dokud je to přínosné.

Požadavky na generátor testovacích stimulů

Požadavky na generátor testovacích stimulů se mohou v řadě případů lišit, což je dané především oblastí použití generátoru. V tématu disertační práce se zaměřuji na použití generátoru testovacích stimulů v oblasti funkční verifikace, proto všechny ostatní oblasti použití nebudeme brát v úvahu. V oblasti funkční verifikace existují následující požadavky:

Parametrizovatelnost
: Parametrizovatelnost je důležitou vlastností pro úpravu chování generátoru, který na základě zadaných parametrů generuje požadovaný výstup, díky němuž se přizpůsobí aktuálním potřebám testovacích stimulů pro dosažení vysokého pokrytí.

Rychlost
: Generátor musí být rychlý, aby nebrzdil již tak zaneprázdněnou a časově vytíženou simulaci verifikovaného systému.

Náhodnost
: Skutečně náhodné testy jsou předpokladem pro rovnoměrné generování všech možných kombinací vstupních hodnot. Tímto dojde k otestování nepředvídatelných a mezních případů v systému.

Univerzálnost
: Jelikož je každý systém jedinečný, je vhodné navrhnout generátor takovým způsobem, aby nebylo nutné vytvářet vždy nový generátor pro specifický systém, ale aby bylo možné využít již existující generátor testů pro různé systémy.

2.4 Aktuální výzkum v oblasti generování testovacích stimulů

Aktuální výzkum v oblasti generování testovacích stimulů se zabývá automatickým generováním testů pro jeden konkrétní systém, především programů pro procesor typu RISC. Požadované programy jsou v této oblasti získávány z několika vstupních struktur, které popisují daný procesor. Tyto vstupní bloky nejsou speciálně navrženy pro široké použití, jelikož využívají popisy používané především pro procesory. Jako vstup je použit popis instrukční sady (ISA) procesoru [7], který je zkombinovaný s dalším popisem. Práce [8] využívá jako druhý popis určité prvky mikroarchitektury procesoru. Disertační práce [9] z loňského roku ze Slovenské technické univerzity v Bratislavě využívá jako druhý popis VHDL (VHSIC Hardware Description Language) popis procesoru. Následně využívá genetický algoritmus (GA) k úpravě programu tak, aby hodnota fitness (pokrytí) ve funkční verifikaci byla co nejvyšší.

Další práce [10], která automaticky generuje programy pro procesory, využívá reprezentaci programů pomocí acyklických grafů. Jako vstup generátoru je použita vlastnoručně navržená instrukční knihovna, která popisuje assemblerovskou syntaxi jednotlivých instrukcí a jejich platné kombinace operandů. Zde se používá již obec-

nější popis testů pro různé procesory, ovšem formát a kombinace všech možných operandů pro jednotlivé instrukce mohou být poměrně složité a rozsáhlé.

Pro generování programů pro procesory lze využít rovněž technik genetického programování. Práce [11] ukazuje generování assemblerovských programů na základě definice instrukčních maker a jejich spojování do složitějšího celku (programu). GA zde hraje klíčovou roli ve výběru těchto maker a přiřazení jejich operandů.

Ze současného výzkumu v oblasti generování testovacích stimulů vyplývá potřeba navrhnout a vytvořit univerzální generátor testovacích stimulů vhodný pro použití ve funkční verifikaci. Generátor by měl být parametrizovatelný a měl by umožňovat rychlé vytváření testovacích scénářů. Velkou výhodou může být rovněž jeho použití jak v generování stimulů pro hardware, tak i pro software.

3 Cíle disertační práce

Na základě studia současného stavu poznání v oblasti funkční verifikace číslicových systémů a v oblasti generování testovacích stimulů pomocí omezujících podmínek byly definovány hlavní cíle a podcíle disertační práce, které musí být splněny, aby bylo naplněno téma disertační práce. Hlavní cíle a jejich podcíle jsou následující:

Cíl 1: Navrhnout a vytvořit univerzální generátor testovacích stimulů založený na řešení problému s omezujícími podmínkami, který bude vhodný především pro použití ve funkční verifikaci.

- Generátor testovacích stimulů musí být schopen parametrizace, aby jej bylo možné použít ve funkční verifikaci. Generátor tímto dokáže za běhu verifikace zpracovávat omezující podmínky a přizpůsobovat tak generované testovací stimuly k dosažení vyššího pokrytí funkcí systému.
- Vstupy pro generátor budou získávány ze speciálně navržených struktur, pomocí kterých bude definován formát generovaných testovacích stimulů a omezující podmínky, které se uplatní v procesu generování těchto stimulů.

Cíl 2: Navrhnout obecný a jednotný popis (jazyk) stimulů různých systémů, pomocí něhož lze popsat veškeré podmínky a zákonitosti nutné k vygenerování platného testovacího stimulu. Výsledkem této činnosti bude vytvoření postupů generování testů pro různé systémy.

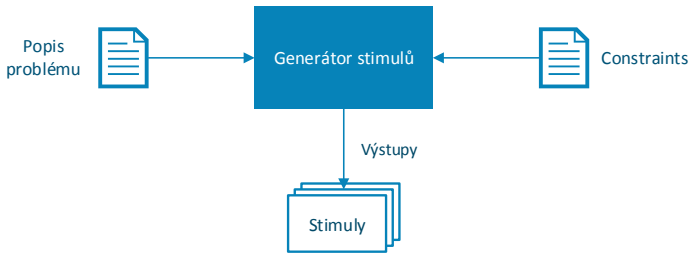
- Hlavním podcílem, který povede ke splnění druhého cíle, je vytvoření sad vstupních popisů pro různé číslicové obvody (procesory, funkční jednotky, jednotky odolné vůči poruchám, atd.), které navržený generátor použije ke generování testovacích stimulů.
- Navržené vstupní popisy budou zobecněny a na základě nich bude definován jazyk pro jednotný popis stimulů různých číslicových systémů. Z tohoto popisu pak budou vyextrahovány klíčové principy generování testů pro dané typy obvodů.

4 Dosavadní práce

Dosavadní práci na tématu disertační práce shrnuje tato kapitola. Je zde představen navržený princip univerzálního generování stimulů, který je použit v praxi ve společnosti Codasip [12] pro generování programů v jazyce symbolických instrukcí (assembler) pro vybrané procesory. Tyto programy představují první případ použití tohoto generátoru.

4.1 Univerzální generátor stimulů

Princip univerzálního generování, který je ukázán na Obr. 2, má za cíl zjednodušit a urychlit vytváření testovacích stimulů pro různé systémy.



Obr. 2: Princip univerzálního generování stimulů.

Základní idea je založena na dvou specifických popisech, které definují formát generovaných dat a podmínky určující, jak má být s těmito daty nakládáno při jejich generování. Popis definující formát generovaných dat je označen jako *Popis problému* a popis definující podmínky a omezení je nazýván *Constraints*. Tyto dva popisy představují vstup do samotného generátoru, který na základě jejich obsahu generuje validní stimuly na jeho výstup. Tyto stimuly pak již mohou být předány konkrétnímu systému jako vstupní data.

Uživatel při vytváření vlastních testovacích stimulů neprogramuje žádný kód, ale pouze specifikuje požadovaný formát stimulů a omezení, které generátor na základě řešení problému s omezujícími podmínkami vyřeší. Důležitým předpokladem pro tento popsaný princip generování je široká množina vstupní abecedy, která musí obecně popisovat problémy a omezení testovacích stimulů.

Tento princip generování stimulů je parametrizovatelný. Dokáže za běhu zpracovávat a měnit omezující podmínky změnou vstupního popisu, a tudíž je vhodný pro použití v procesu funkční verifikace.

Popis problému představuje první vstup generátoru stimulů. Pro definování problému, tedy toho, co chceme generovat, jsou k dispozici tři základní části: *Nahrazení*, *Proměnné* a *Syntaxe*.

Část *Nahrazení* definuje identifikátor a všechny možné substituce, za které je možno daný identifikátor nahradit. Jedná se o obdobu výčtového datového typu. Nahrazuje se za konkrétní řetězec z definované množiny hodnot. V každém novém cyklu generování dochází k náhodnému zvolení určité substituce pro daný identifikátor. V případě programů pro procesory je tato část použita především pro definici dostupných registrů daného procesoru.

Část *Proměnné* definuje proměnné v obecném slova smyslu. Pro každou proměnnou je přiřazena náhodná hodnota v závislosti na jejím datovém typu a to v každém cyklu generování. V případě programů pro procesory je tato část použita pro definici a přiřazení přímých operandů anebo řetězců jako návěští skokových instrukcí.

Část *Syntaxe* syntakticky popisuje řetězce, jeden po druhém, které mají být náhodně generovány na výstup generátoru jako součást stimulu. V jednotlivých definovaných řetězcích se mohou nacházet identifikátory z částí *Nahrazení* a *Proměnné*. Tyto identifikátory jsou nahrazeny za konkrétní nebo náhodnou hodnotu v závislosti na typu identifikátoru. Syntaktická část představuje statické hodnoty v generovaném řetězci, zatímco zbylé dvě části představují dynamické (měnící se) hodnoty v generovaném řetězci. V případě, že chceme tedy generovat assemblerovské programy, bude tato část obsahovat definici instrukční sady daného procesoru.

Constraints reprezentují podmínky a omezení pro generované stimuly. Omezením možných řešení je zajištěno generování platných scénářů pro zvolený systém. Constraints představují především omezení pro datové hodnoty (proměnná může nabývat pouze hodnot z určitého rozsahu) nebo závislostní omezení (některá kombinace hodnot nemůže nastat po aktuálně generované kombinaci). Constraints jsou unikátní pro každý systém, a tedy různá omezení jsou aplikována na různé systémy.

Generátor stimulů kombinuje *Syntaxe*, *Nahrazení* a *Proměnné* tak, že všechna omezení jsou splněna, a tedy žádné není porušeno. Výstupem generátoru je posloupnost řádků, která odpovídá definovanému problému a která tvoří výsledný stimul.

5 Závěr

V této práci byl představen aktuální stav poznání a z toho plynoucí návaznost na téma disertační práce, jejímž základem je generování stimulů pro použití především ve funkční verifikaci. Rovněž byly stanoveny cíle disertační práce a navrženy způsoby, jak k těmto cílům dojít. V rámci dosavadní práce byl navržen a implementován generátor testovacích stimulů, který generuje stimuly na základě dvou vstupních popisů. Tyto vstupní popisy představují vlastnoručně navržené a definované struktury, které byly navrženy tak, aby byly co nejuniverzálnější a aby bylo možno pomocí nich popsat stimuly různých systémů. Vytvořený generátor byl použit v praxi. Na základě specifikace byly vytvořeny vstupní popisy pro generování assemblerovských programů pro procesory od společnosti Codasip. V další práci budou vytvořeny popisy pro generování stimulů pro jiné typy systémů a to pouze na základě jejich definování pomocí již zmiňovaných navržených vstupních struktur. Na základě různých popisů budou sepsány principy generování stimulů pro dané typy systémů, v čemž je možno vidět zobecnění získaných poznatků. Dosavadní práce je shrnuta v časopise Microprocessors and Microsystems [1].

Reference

[1] Podivinsky, J.; Cekan, O.; Simkova, M.; Kotasek, Z.: *The evaluation platform for testing fault-tolerance methodologies in electro-mechanical applications*, Microprocessors and Microsystems, 2015, ISSN 0141-9331, Přijato k publikaci, http://dx.doi.org/10.1016/j.micpro.2015.05.011.

[2] Roy, S.; Ramesh, S.: *Functional verification of system on chips - practices, issues and challenges*. In Proceedings of ASP-DAC 2002, 2002, s. 11-13, doi:10.1109/ASPDAC.2002.994873.

[3] Meyer, A.: *Principles of Functional Verification*. Amsterdam: Elsevier Science, 2003, ISBN 978-0-7506-7617-5, 216 s.

[4] Tasiran, S.; Keutzer, K.: *Coverage metrics for functional validation of hardware designs.* Design Test of Computers, IEEE, ročník 18, č. 4, červen 2001, ISSN 0740-7475, s. 36-45.

[5] Kottho, L.: *Constraint Solvers: An Empirical Evaluation of Design Decisions*. ArXiv e-prints, leden 2010: s. 1-23, <1002.0134>.

[6] Yuan, J.; Pixley, C.; Aziz, A.: *Constraint-based verification*. New York: Springer, 2006, ISBN 978-0-387-25947-5, I-XII, 1-253 s.

[7] Patterson, D. A.: *Reduced Instruction Set Computers*. Commun. ACM, ročník 28, č. 1, leden 1985: s. 8-21, ISSN 0001-0782, doi:10.1145/2465.214917.

[8] Belkin, V.; Sharshunov, S.: *ISA Based Functional Test Generation with Application to Self-Test of RISC Processors*. In Design and Diagnostics of Electronic Circuits and systems, 2006 IEEE, duben 2006, s. 73-74, doi:10.1109/DDECS.2006.1649575.

[9] Hudec, J.: *An efficient technique for processor automatic functional test generation based on evolutionary strategies*. In Proceedings of the ITI 2011, 33rd International Conference on Information Technology Interfaces, červen 2011, ISSN 1330-1012, s. 527-532.

[10] Corno, F.; Sanchez, E.; Reorda, M.; aj.: *Automatic test program generation: a case study*. IEEE Design and Test of Computers, ročník 21, č. 2, březen 2004: s. 102-109, ISSN 0740-7475, doi:10.1109/MDT.2004.1277902.

[11] Corno, F.; Reorda, M.; Squillero, G.; aj.: *A genetic algorithm-based system for generating test programs for microprocessor IP cores*. In Proceedings of the 12th IEEE International Conference on Tools with Artificial Intelligence (ICTAI 2000), IEEE Computer Society, listopad 2000, ISBN 0-7695-0909-6, s. 195-198.

[12] Codasip: Codasip - Delivering the Power of ASIP [online]. 2014 [cit. 2015-01-06]. URL <http://www.codasip.com>

EFFICIENT REPROGRAMMING FOR RESOURCE CONSTRAINED EMBEDDED DEVICES

Ondrej Kachman

Applied Informatics, 1st class, full-time study
Supervisor: Doc. Ing Ladislav Hluchý, PhD.

Institute of Informatics, Slovak Academy of Sciences
Dúbravská cesta 9, 845 07 Bratislava, Slovakia

ondrej.kachman@savba.sk

Abstract. The efficient reprogramming methods are very important for devices with constrained resources. The resource constraints of these devices are memory, computational power and energy sources. To reprogram devices efficiently, memory management of these devices is important, update data transferred through network must be minimal and update agent on the devices must be small. This paper analyzes these areas and sums up some possible improvements for the given problem.

Keywords: reprogramming, embedded system, differencing algorithm

1 Introduction

With the recent advances in the Internet of Things (IoT) area, number of small computers and embedded devices is rising every day. There have been 5 billion IoT devices in 2013 and predictions say, that by 2020 there will be more than 20 billion devices connected to the internet. These devices can perform various tasks like providing home security, preventing natural disasters, monitoring structural health of buildings etc., and sometimes they can be deployed in physically inaccessible areas where they are powered only by batteries [1]. The most important area to mention here is the area of wireless sensor networks (WSNs).

Embedded devices based on 8-bit or 16-bit microcontrollers usually have limited computational power and memory. If they are powered by batteries, it is important to handle these resources efficiently. Network communication, especially wireless in case of WSNs, is very costly. 1 bit transferred by wireless connection can consume as much energy as 1000 instructions executed by microcontroller. This opens the problem of efficient reprogramming, as sometimes devices with firmware developed in test conditions may not function properly once deployed to remote location and only firmware updates can fix them. Transferring whole firmware images can be very costly, especially if the updates come in more versions after one another [2]. There are methods to

save energy on the data that need to be transferred and this paper analyzes these methods and discusses some possible ways to improve them.

2 Embedded Systems Initialization and Updates

Firmware is usually a rather small program run by embedded devices. Its main objective is to perform tasks the devices were developed for. Apart from firmware, devices can have a small program called bootloader stored in their memory. Bootloader can perform some system initializations before firmware is loaded and can also be responsible for firmware updates. Bootloader can load firmware sections to different parts of device's ROM and RAM memories, depending on a boot scenario developer chooses. This can affect update capabilities of the device. Firmware can be divided into three main sections (also referred to as segments) [3]:

- .bss section allocated for uninitialized data
- .data section containing initialized data
- .text section containing instructions of the firmware

2.1 Boot Scenarios

Depending on available amount of RAM memory, firmware code can run from ROM or RAM. Programs run from RAM tend to execute faster, but not all embedded devices have enough RAM to hold the entire firmware code and use of paging can be inefficient for resource constrained devices. If firmware image can be copied to RAM, it can be compressed in ROM saving space and bootloader decompresses it into RAM. There are three basic boot scenarios [3]:

1. Execution from ROM. Bootloader copies .data section and allocates .bss section in RAM, then jumps to the address in ROM where first instruction of the firmware is stored. Device continues execution of firmware from ROM and works with data in RAM. This scenario does not include compression.
2. Execution from RAM. Every firmware section is allocated/copied into RAM and bootloader exits with jump into RAM to the first instruction of the firmware. It is possible to compress firmware into ROM, but bootloader must include decompressing algorithm.
3. Network boot, execution from RAM. Bootloader initializes network connection and downloads the firmware image that can optionally be compressed into RAM. Then it allocates .bss section and copies .data section to their respective addresses and runs new firmware.

2.2 Firmware Updates

In case firmware update is needed, bootloader can be used for network boot to download and rewrite new image. But for some devices, this may not be feasible and they may need to perform updates on-the-fly. This requires firmware to be able to download

update data and include a small sub-program called update agent, that performs updates [4]. For resource constrained devices, update data transferred need to be minimal, usually in exchange for execution time. The term delta file is introduced. Delta file is a file that contains all the information needed to update firmware to newer version, but does not necessarily contain full source code [5]. Improving similarity of two firmware versions can help delta file generators to achieve better size of the delta file.

First step that can help to improve the similarity is update conscious compilation process [2]. This requires to alter the compiler in such way, that it preserves register allocations for variables in both firmware versions. Once register must store another variable, move instructions are injected into the code. This helps to maintain similarity as variables are allocated in the same registers for both versions, but execution time of the new version may increase due to extra move operations. If program versions are too different, this approach can downgrade program quality and efficiency.

Great area of research in the problem of efficient reprogramming is the area of differencing algorithms. These are used to generate delta files. The input of these algorithms are usually the new and the old version of the firmware. If the old firmware is denoted as F_{base} and the new firmware as $F_{version}$, we generate a delta file Δ [6]:

$$F_{base} + F_{version} \rightarrow \Delta_{(base,version)}$$

On the target device, update agent receives the delta file and must reconstruct the new firmware image from the old firmware.

$$F_{base} + \Delta_{(base,version)} \rightarrow F_{version}$$

The update agent should be also able to check the integrity of the delta file in case it was received corrupted. Update applied from corrupted file could cause device malfunction and result in a system failure. Developer must consider some redundant data to secure the delta file integrity. Usually, CRC codes or hash functions are used.

3 Differencing Algorithms

The differencing algorithms responsible for delta file generation must solve two main optimization problems:

1. How to determine identical parts between the two firmware images
2. How to encode update data into delta file efficiently

The problem of searching for identical part of the two files is solved by algorithms for detection of longest common subsequences between strings. Although this problem is NP-hard for an arbitrary number of strings, for 2 strings it can be calculated in polynomial time. Encoding of the delta file depends on target device's memory resources, choice of a right boot scenario can help developers to minimize delta files.

3.1 Longest Common Subsequence

Formal definition of a longest common subsequence (LCS) problem for 2 strings is as follows. There are two strings:

$$C = c_1, c_2, c_3, \dots, c_x$$

$$A = a_1, a_2, a_3, \dots, a_m$$

String *C* is a subsequence of *A* if there exists mapping $f: \{1,2,3, \dots, x\} \rightarrow \{1,2,3, \dots, m\}$, such that $f(i) = k$ if symbol $c_i = a_k$ and f is monotone strictly increasing function. For the last 50 years, many algorithms that solve LCS problem were designed. Some of them were used to propose methods for efficient remote updates of embedded devices. One of these algorithms is the Hirschberg's algorithm. Designed in 1975, it was recently used in the article describing efficient code update method for WSNs [7]. In the area of WSNs, two more algorithms called RMTD (reprogramming with minimal transferred data) and R-sync are often mentioned [5]. Every algorithm differs in time and space complexity, and the best one with space complexity $O(n)$ and time complexity $O(n(\log n)$ is called DASA and it is based on suffix arrays [8].

3.2 Delta Files

Once every LCS has been calculated, sequences that changed their position within the file need to be included in the delta file. However, there is no need to include the whole code of the sequence, only copy operation, as target device can copy common sequence to its new position in the memory. New code has to be included in the delta file whole with the address it needs to be added to. The basic encoding of the delta file then consists of the two main operations [6]:

$$< ADD >< ADDRESS_{new} >< n >< BYTE_1, BYTE_2, \dots, BYTE_n >$$

$$< COPY >< ADDRESS_{old} >< ADDRESS_{new} >< n >$$

COPY and ADD keywords encode operation. $ADDRESS_{old}$ of the copy operation specifies an address from which data should be copied and $ADDRESS_{new}$ specifies an address to which data should be copied or added. Number n specifies number of bytes to be copied or added and bytes BYTE represent data to be added to firmware. There are ways to optimize size of encoded operations. If a new firmware is reconstructed byte by byte at a target device, $ADDRESS_{new}$ can be left out from both ADD and COPY operations [5].

Update propagation is usually carried out by networking protocols. There are some protocols designed especially for firmware version synchronization in wireless sensor networks. There are two protocols mentioned in most of the analyzed literature, Multihop Over-the-Air Programming (MOAP) and Deluge. Deluge is considered state-of-the-art protocol even after 11 years, but it propagates updates in firmware pages, it does not use delta files. Redesigning Deluge to use delta files could lead to performance improvements of this protocol [7].

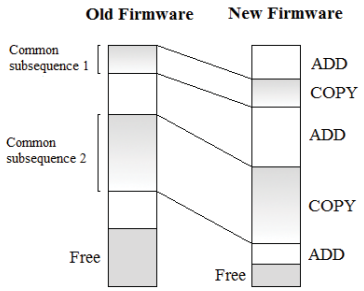


Fig. 1. Reconstructing new firmware image using ADD and COPY operations

Once the delta file is propagated to the target device, update agent uses it to perform firmware update. It is a requirement for the update agent code to be small in size, but include all the functions needed to perform integrity check of the delta file and reconstruction of the new firmware. The process of firmware image reconstruction is illustrated in the Fig. 1.

3.3 Improving Differencing Algorithms

There may still be some space for improvement in differencing algorithms, this subchapter presents some ideas that might be promising. One way may be through better management of relocatable code. Relocatable code is source code that will run properly from any position in the program memory, as long as reference instructions (jumps, branches etc.) have good absolute addresses. This can be used to fragment program memory and reduce copy operations that need to be encoded into the delta file. Code relocations must be supported by linker and can be performed with splitting .text section of the firmware into subsections. The number of possible subsections is usually limited, but these limits are to thousands of subsections (8192 subsections for Atmel's AVR compiler and linker for its 8-bit microcontrollers). Firmware can be split to stand-alone functions or modules, every function and module can have its own section assigned, and leaving space before and after this section can lead to smaller delta files. Microcontrollers with Harvard architecture should be able to take full advantage of relocatable code, microcontrollers with Von Neumann architecture might require safe write operations so non-volatile data are not overwritten.

Delta file encoding techniques might be improved too. It is unnecessary to encode operations using byte when only 2 operations exist, one bit per operation should be enough. This requires delta file to be split into two sections, one short section encoding number of operations and operations, the other encoding addresses and data. With relocatable code introduced, amount of data needed to perform upgrades might reduce.

4 Conclusion

This article is focused on methods used for efficient reprogramming of resource constrained embedded devices. These devices are used mostly in WSNs, and energy saving

on wireless communication is crucial for them. When firmware update is needed, data needed for full update should be minimal. This problem is closely related to memory management of the devices and differencing algorithms that generate delta files. Perfecting these algorithms may lead to very efficient and secure reprogramming of embedded devices. The outcome of research are some open problems that need to be investigated and may lead to some improvements in the existing methods.

Acknowledgement

This work has been supported by Slovak national project VEGA 2/0192/15.

References

[1] J. Höller, V. Tsiatsis, C. Mulligan, S. Karnouskos, S. Avesand and D. Boyle, From Machine-to-Machine to the Internet of Things, Oxford: Academic Press, 2014. ISBN 978-0-12-407684-6

[2] Y. Huang, Z. Mengying and C. J. Xue, "WUCC: Joint WCET and Update Conscious Compialtion for Cyber-physical Systems," in *Design Automation Conference (ASP-DAC), 2013 18th Asia and South Pacific*, Yokohama, IEEE, 2013, pp. 65 - 70.

[3] Q. Li, Real-Time Concepts for Embedded Systems, San Francisco: CMP Books, 2003. ISBN 1-57820-124-1

[4] G. Jurković and V. Sruk, "Remote Firmware Update for Constrained Embedded Systems," in *Information and Communication Technology, Electronics and Microelectronics (MIPRO), 2014 37th International Convention*, Opatija, IEEE, 2014, pp. 1019 - 1023.

[5] W. Dong, Y. Liu, C. Chen, J. Bu, C. Huang and Z. Zhao, "R2: Incremental Reprogramming Using Relocatable Code in Networked Embedded Systems," in *IEEE Transactions on Computers*, vol. 62, Shanghai, IEEE, 2012, pp. 1837 - 1849.

[6] R. C. Burns and D. D. E. Long, "A linear time, constant space differencing algorithm," in *Performance, Computing, and Communications Conference, 1997*, Phoenix, IEEE, 1997.

[7] B. Mazumder and J. O. Hallstrom, "An Efficient Code Update Solution for Wireless Sensor Network Reprogramming," in *Embedded Software (EMSOFT), 2013 Proceedings of the International Conference*, IEEE, 2013, pp. 1 - 10 .

[8] B. Mo, W. Dong, C. Chen, J. Bu and Q. Wang, "An efficient differencing algorithm based on suffix array for reprogramming wireless sensor networks," in *Communications (ICC), 2012 IEEE International Conference*, Ottawa, IEEE, 2012, pp. 773 - 777.

PRAKTICKÉ ASPEKTY LINEÁRNÍ KRYPTOANALÝZY BLOKOVÝCH ŠIFER

Josef Kokeš
Informatika, 2. ročník, prezenční
Školitel: Róbert Lórencz

Fakulta informačních technologií
České vysoké učení technické v Praze
Thákurova 9, 16000 Praha 6, Česká republika

josef.kokes@fit.cvut.cz

Abstrakt. Při analýze šifry Baby Rijndael jsme narazili na několik zvláštností v chování techniky lineární kryptoanalýzy. Zaměřili jsme se na důkladný průzkum těchto vlastností a odhalili dosud nepopsané závislosti mezi výběrem lineárních aproximací a úspěšností odhalení šifrovacího klíče. Ukazujeme, že mezi jednotlivými lineárními aproximacemi panují značné kvalitativní rozdíly, přestože pravděpodobnostní odchylka jednotlivých aproximací je stejná. Podobné rozdíly nalezneme také při aplikaci těchto aproximací na odhalení různých bitů klíče.

Klíčová slova. Kryptoanalýza, lineární kryptoanalýza, Baby Rijndael, lineární aproximace, hledání klíče.

1 Úvod

Ve světě, ve kterém objem i hodnota informací neustále roste [10], musíme klást velký důraz i na ochranu těchto informací – před zničením, ale také před zneužitím. Lidstvo se s těmito problémy potýká už staletí, klasické šifrovací systémy ovšem trpěly řadou slabin plynoucích z nedostatečného porozumění problematice a zaměření na utajení šifrovacích postupů. Moderní kryptologie už plně staví na tzv. Kerckhoffsově principu, totiž že bezpečnost šifry má být postavena výhradně na bezpečnosti klíče [12]. Důležitým nástrojem pro zajištění tohoto principu je *kryptoanalýza*, která se snaží nalézt způsoby, jak prolomit bezpečnost minulých, současných i budoucích šifer a zjistit utajenou informaci snáze než vyzkoušením všech možných klíčů. Důsledně prováděná kryptoanalýza nás může ochránit před chybami, které jsou důsledkem tzv. "Schneierova zákona" – *Každý, od netušícího amatéra až po nejlepšího kryptografa, dokáže navrhnout [šifrovací] algoritmus, který on sám nedokáže prolomit. [...] Těžké je vytvořit algoritmus, který nedokáže prolomit ani nikdo jiný.*[16]

Velkým podnětem pro výzkum kryptoanalytických technik byla soutěž Advanced Encryption Standard z let 1997-2001, jejíž cílem bylo vybrat kvalitní symetrickou blokovou šifru jako nový šifrovací standard. Přímo v rámci soutěže podstoupili kandidáti na AES vyčerpávající kryptoanalýzu ze strany autorů konkurečních návrhů, další kryptoanalýza pak následovala i v dalších letech až do současnosti. Základními technikami jsou lineární [15], diferenciální [2] a algebraická kryptoanalýza a jejich varianty a rozšíření:

- Analýza tzv. impossible differentials [17][4][3] (nemožných diferenciálů) vychází z diferenciální

kryptoanalýzy, ovšem na rozdíl od běžného hledání diferenciálů s maximální pravděpodobností se naopak snaží nalézt diferenciály, které v určitém místě uvnitř šifry nastat *nemohou*.

- Bumerangový útok [18] také vychází z diferenciální kryptoanalýzy a prodlužuje její dosah tím, že rozděluje šifru na dvě poloviny a pracuje s takovými dvojicemi šifrových a otevřených textů, aby jedna dvojice pomohla druhé dvojici dosáhnout až do konce šifry.

- Analýza tzv. related keys [5][6] je podobná diferenciální kryptoanalýze, hledá slabiny v šifrovacím algoritmu za předpokladu, že pro zašifrování téhož otevřeného textu použijeme dva různé klíče, které mezi sebou mají předem definovaný vztah; kryptoanalytik nezná hodnotu klíčů, ale má zaručenou existenci tohoto vztahu.

- Biclique přístupy [7] uplatňují teorii úplných bipartitních grafů pro provedení tzv. Meet-in-the-Middle útoků, ve kterých se kryptoanalytik snaží vyjádřit šifru prostřednictvím dvou funkcí, jedné začínající na straně otevřeného textu a druhé začínající na straně šifrového textu, které se uvnitř šifry definovaným způsobem setkají.

- ... a další útoky, jako Demirci-Selçuk Meet-in-the-Middle útok [9] nebo reflection útoky [11]

Vývoj je v této oblasti velmi výrazný, vidíme v něm však několik slabých míst:

1. Zdá se, že převážná většina výzkumu je zaměřena na diferenciální kryptoanalýzu a její modifikace. Ostatní základní kryptoanalytické techniky se zdají být poměrně opomíjené, přinejmenším v nich jsou nové výsledky publikovány podstatně vzácněji.

2. Zřejmě vůbec není zkoumána možnost, jak propojit jednotlivé kryptoanalytické techniky tak, aby i dílčí výsledky v jedné z nich (např. odhalený jeden bit klíče) pomohly druhé zvýšit svoji účinnost.

3. Velmi malá pozornost je věnována praktickému ověření výsledků. Do značné míry je to pochopitelné, moderní šifry jsou natolik silné a klíče natolik dlouhé, že i výrazný teoretický úspěch má stále takovou výpočetní složitost, že praktické ověření není zvládnutelné v rozumném čase.

V našem výzkumu kryptoanalýzy se snažíme tyto nedostatky odstranit.

2 Náš přístup

V našem výzkumu kryptoanalytických technik se snažíme eliminovat nedostatky popsané v předchozí kapitole. Zaměřujeme se převážně na lineární kryptoanalýzu, která je soudobým vývojem poněkud potlačována, věnujeme však značnou pozornost potenciálním styčným bodům s ostatními kryptoanalytickými technikami. Nadějně vypadá zejména možnost propojení lineární a algebraické kryptoanalýzy: lineární kryptoanalýza může lépe odhalovat hodnoty některých bitů klíče, což následně může zjednodušit problém algebraické kryptoanalýzy v řešení rozsáhlé soustavy rovnic. Tuto možnost nyní zkoumají další členové našeho týmu.

Druhým podstatným aspektem našeho přístupu je volba použité šifry. Z praktického hlediska by bylo vhodné soustředit se na některou z běžně používaných šifer, zejména Rijndael (AES). Bohužel je ale složitost těchto šifer tak velká, že většinu navrhovaných útoků nelze ani vyzkoušet, natož komplexně prověřit na větším množství případů. Rozhodli jsme se proto vyjít z šifry Baby Rijndael [1], která má pro analýzu dvě výhodné vlastnosti.

1. Baby Rijndael vychází z *reálně používané šifry* Rijndael. Její autor Cliff Bergman ji navrhl tak, aby základní principy šifry zůstaly zachovány, ale zmenšila se velikost bloku a klíče. Struktura

šifry je příslušně upravena, lze ale ukázat [13][14], že tento návrh dodržuje všechny relevantní designové principy, rozhodnutí a také požadavky, které si stanovili autoři Rijndaelu [8] při návrhu své šifry. V tomto ohledu je Baby Rijndael přijatelným modelem Rijndaelu; útoky fungující proti Baby Rijndaelu nemusí nutně být prakticky realizovatelné i proti Rijndaelu, lze ale očekávat, že aspoň základní mechanismy pracovat budou.

2. Detailní analýza Baby Rijndael je *výpočetně zvládnutelná*. Šifra používá blok i klíč o délce 16 bitů, tzn. existuje jen 65536 různých otevřených textů a 65536 různých klíčů. Prolomit tak slabou šifru hrubou silou by samozřejmě bylo triviální, důležitější ale je, že můžeme relativně snadno vyzkoušet chování šifry ve zvolené situaci pro všechny možné otevřené texty a všechny možné klíče a lépe tak vyhodnotit *skutečné* chování příslušné kryptoanalytické techniky. To by mělo pomoci při stanovení *metodologie*, jak uplatnit techniku i na rozsáhlejší šifry, které úplné prověření kvůli své rozsáhlosti znemožňují.

Náš přístup je na těchto vlastnostech založen. V rámci jednotlivých zkoumaných podproblémů se snažíme zformulovat způsob využití lineární kryptoanalýzy pro útok na šifru, následně sestavit a implementovat algoritmus útoku a tento pak prakticky prověřit. Poslední krok přitom probíhá pokud možno vyčerpávajícím způsobem, tzn. snažíme se ověřit chování útoku vzhledem ke všem možným klíčům.

V dalším textu budeme používat následující termíny:

Optimální aproximace Lineární aproximace šifry, jejíž pravděpodobnostní odchylka dosahuje maximální dosažitelné hodnoty.

Skutečný klíč Klíč resp. část klíče, který byl skutečně použit k zašifrování otevřeného textu. Jinými slovy, jde o klíč, který se snažíme odhalit.

Kandidátní klíč Klíč resp. část klíče, který byl nějakým algoritmem označen jako hledaný klíč. V optimálním případě jsou kandidátní a skutečný klíč totožné a útok byl úspěšný.

Seznam kandidátních klíčů Taková permutace všech možných klíčů, která vznikne postupným vybíráním kandidátních klíčů a jejich řazením za sebe: kandidátní klíč se nachází na první pozici v této permutaci; kdybychom kandidátní klíč z kryptoanalýzy zcela vyřadili, určí nám kryptoanalýza jiný kandidátní klíč, který nalezneme na druhé pozici v permutaci; atd.

Pozice skutečného klíče Pozice skutečného klíče v seznamu kandidátních klíčů. Nejlepší hodnota je 1 značí správně nalezený klíč, hodnota n značí klíč nalezený na n-tý pokus.

Aktivní S-box Substituční box, případně jiný nelineární prvek šifry, kterým prochází použitá lineární aproximace.

3 Dosažené výsledky

V rámci výzkumu jsme se zaměřili na následující oblasti:

3.1 Vliv zvolené lineární aproximace na úspěšnost odhalení skutečného klíče

Podle stávající teorie lineární kryptoanalýzy by úspěšnost odhalení klíče měla záviset pouze na pravděpodobnostní odchylce zvolené lineární aproximace a na počtu vzorků otevřeného a šifrového textu, a to v tom smyslu, že je-li ϵ pravděpodobnostní odchylka, pak potřebujeme $\frac{1}{\epsilon^2}$ vzorků [15]. V rámci mé diplomové práce [13] jsme však narazili na případ, kdy aproximace se stejnou pravděpodobnostní odchylkou a se stejným (dostatečným) počtem vzorků vykazovaly odlišnou průměrnou úspěšnost odhalení klíče.

	Aktivní S-boxy v poslední rundě					
(aktivní box označen 1)	0011	0101	0110	1001	1010	1100
Pravděpodobnostní odchylka	$\pm\frac{1}{256}$	$\pm\frac{1}{256}$	$\pm\frac{1}{256}$	$\pm\frac{1}{256}$	$\pm\frac{1}{256}$	$\pm\frac{1}{256}$
Počet optimálních aproximací	3840	48	48	48	48	3840
Počet možných klíčů	256	256	256	256	256	256
Průměrná pozice skutečného klíče	114,75	49,58	111,91	111,90	49,58	114,72
Medián pozice skutečného klíče	114,91	49,85	111,77	111,65	49,88	115,08
Směr. odch. pozice skutečného klíče	2,89	6,03	2,23	2,32	6,03	2,77

Tabulka 1: Vliv zvolené lineární aproximace na úspěšnost odhalení skutečného klíče. Ideálem je pozice 1, značící, že pro všechny použité klíče byl správný klíč nalezen na první pokus. Pozice $PocetMoznychKlicu/2$ odpovídá v průměru zcela náhodnému výběru klíče, tzn. nejhoršímu možnému případu.

Průměrný počet správně nalezených bitů	4,63
Maximální počet správně nalezených bitů	4,90
Minimální počet správně nalezených bitů	4,40

Tabulka 2: Úspěšnost lineárních aproximací s aktivními S-boxy typu 0101 při odhalování části klíče. Ideální výsledek by byl 8 bitů (klíč odhalen zcela správně), 4 bity značí nejhorší možný výsledek (výsledek odpovídá náhodnému výběru).

Během prvního a zejména druhého roku doktorského studia jsme tedy provedli vyčerpávající šetření všech optimálních lineárních aproximací se dvěma aktivními S-boxy v poslední rundě šifry, a to pro všechny kombinace klíče a otevřeného textu. Pro každou aproximaci a klíč jsme měřili pozici skutečného klíče a proměnlivost této pozice. Výsledky jsou zachyceny v tabulce 1.

Vidíme, že úspěšnost jednotlivých aproximací se výrazně liší, přestože by podle teorie měly dávat zhruba stejné výsledky. Jako nejúspěšnější se jeví aproximace, které v poslední rundě střídají aktivní a neaktivní S-boxy; ostatní aproximace měly v průměru cca 2,25-krát horší pozici správného klíče a také menší variabilitu tohoto ukazatele.

Tento experiment bohužel ukázal, že schopnost nalézt správný klíč je i pro aproximace typu 0101, které byly v průměru nejúspěšnější, spíš špatná: Pozice 49,58 značí, že potřebujeme provést v průměru 50 testů na vybrání správného klíče, zatímco kdybychom postupovali zcela náhodně nebo systematicky např. v lexikografickém pořadí, potřebovali bychom 128 testů. Úspora získaná lineární kryptoanalýzou je v tomto případě jen cca 1,25 bitu, je ovšem více než vykompenzována potřebou kompletní sady vzorků otevřeného a šifrového textu (zatímco pro ostatní metody by nám stačil vzorek jeden) a výpočetní náročností samotné lineární kryptoanalýzy.

3.2 Počet správně odhalených bitů

Pro všech 48 aproximací s aktivními S-boxy typu 0101, které se v předchozím experimentu projevily jako průměrně nejúspěšnější, jsme dále provedli analýzu, kolik bitů kandidátního klíče se shoduje s odpovídajícími bity správného klíče, tzn. podařilo se je odhalit správně, a které bity to byly. Odpověď na první otázku ukazuje tabulka 2.

Tento výsledek není povzbudivý, zkoumáme-li ale nalezené bity nikoliv jako celek ("kolik bitů se podařilo odhalit správně"), ale každý zvlášť ("jak často se podařilo správně odhalit bit 1"), zjistíme, že úspěšnost se výrazně liší: Nejúspěšnější bit (bit č. 3) má v nejlepší aproximaci pravděpodobnost nalezení 70,34 %, mnohem víc než očekávaných průměrných 57,88 %. 26 různých kombinací aproximace a bitu má pravděpodobnost správného nalezení větší než 65 %, z toho 8 aproximací definuje bit 3, 8 bit 11, 5 bit 0 a 5 bit 8. Naproti tomu nejlepší aproximace např. pro bit 1 má pravděpodobnost splnění jen 60,94 %.

To dále potvrzuje hypotézu, že úspěšnost lineární kryptoanalýzy závisí na mnohem více faktorech než jen na velikosti pravděpodobnostní odchylky a na počtu vzorků otevřeného a šifrového textu.

	průměrná úspěšnost
11	82,19 %
3	82,01 %
8	79,85 %
0	79,47 %

Tabulka 3: Úspěšnost odhalení vybraných bitů klíče, využijeme-li k jeho určení váženého průměru pěti nejkvalitnějších lineárních aproximací s aktivními S-boxy v poslední rundě typu 0101. Očekávaná úspěšnost jedné průměrné lineární aproximace by byla 57,88 %.

Tento výsledek je poměrně povzbudivý, protože nám dává návod, jak s dobrou pravděpodobností vytipovat hodnotu konkrétního bitu klíče. Tuto hodnotu pak můžeme implementovat do jiných kryptoanalytických technik a tím tyto techniky vzájemně propojit.

3.3 Další výsledky

Zjistili jsme i další zajímavé a nečekané výsledky, které však nejsou dosud precizně zpracovány, aby zde mohly být publikovány. Měřili jsme se úspěšnost odhalení všech čtyř "nejlepších bitů" současně. Sledovali jsme závislost úspěšnosti odhalení těchto bitů na počtu vzorků, která vykazuje překvapivě lineární charakter. Předběžné výsledky také ukazují na to, že jednotlivé klíče nemají shodné chování, ale že některé klíče jsou snáze prolomitelné lineární kryptoanalýzou než jiné klíče. To by ovšem bylo zcela proti očekávání, že v šifrách typu Rijndael jsou všechny klíče stejně dobré.

Zpracování těchto výsledků bude předmětem dalšího zkoumání.

4 Cíle dizertační práce

Původně byla moje dizertační práce zaměřena na zkoumání vlastností blokových šifer, zejména šifry Rijndael. V průběhu výzkumu se však objevily aspekty, které tento výzkum výrazně komplikují. Kromě rozsáhlosti tématu je to i otázka, nakolik je zvolená technika využití zmenšeného modelu šifry pro dosažení tohoto cíle vhodná.

V důsledku toho se cíl poněkud proměnil. Soustředíme se nyní na teoretické i praktické aspekty lineární kryptoanalýzy, jak se projevují na šifře Baby Rijndael, a s ohledem na možné vztahy k dalším kryptoanalytickým technikám, zejména algebraické kryptoanalýze. Podstatnou součástí by mělo být stanovení metodologie, která by umožnila určit klíčové parametry pro úspěšnost kryptoanalýzy a na jejich základě vybrat optimální postup pro provedení útoku na šifry – a to i jiné šifry než Baby Rijndael.

5 Závěr

V našem výzkumu jsme navázali na už dříve odhalené zvláštnosti ve fungování lineární kryptoanalýzy při jejím použití na šifru Baby Rijndael. Díky zaměření na komplexní analýzu této techniky pro všechna možná vstupní data jsme zjistili, že lineární kryptoanalýza vykazuje u šifra Baby Rijndael značné odlišnosti od teorií popsaného chování: Nejen že optimálních lineárních aproximací šifry je relativně mnoho, ale tyto aproximace vykazují navzájem výrazně odlišné chování a úspěšnost v odhalování klíče. Také odlišnosti v úspěšnosti odhalování jednotlivých bitů klíče jsou velmi významné. Nyní potřebujeme zjistit, *proč* k těmto odlišnostem dochází a *jak* je využít pro útok nejen na Baby Rijndael, ale i na další šifry – nejlépe i na samotný Rijndael. To je předmětem aktuálního výzkumu.

Reference

[1] Bergman, C.: A Description of Baby Rijndael. Iowa State University, 2005.

[2] Biham, E., Shamir, E.: Differential Cryptanalysis of DES-like Cryptosystems. Lecture Notes in Computer Science Volume 537, 1991, pp 2-21.

[3] Biham, E.: Impossible cryptanalysis of Skipjack. CRYPTO '98, 1998. Available online: http://video.google.com/videoplay?docid=-3129573377272841l2&hl=en [cit. 2015-06-04].

[4] Biryukov, A.: Miss-in-the-middle attacks on IDEA. CRYPTO '98, 1998. Available online: http://video.google.com/videoplay?docid=-5370812042739937841&hl=en [cit. 2015-06-04].

[5] Biryukov, A., Khovratovich, D.: Related-key Cryptanalysis of the Full AES-192 and AES-256. Lecture Notes in Computer Science Volume 5912, 2009, pp 1-18.

[6] Biryukov, A., Dunkelman, O., Keller, N., Khovratovich, D., Shamir, A.: Key Recovery Attacks of Practical Complexity on AES Variants With Up To 10 Rounds. Cryptology ePrint Archive, Report 2009/374, 2009.

[7] Bogdanov, A., Khovratovich, D., Rechberger, C.: Biclique Cryptanalysis of the Full AES. Advances in Cryptology – ASIACRYPT 2011. AES Round 1 Technical Evaluation CD-1: Documentation. NIST, August 1998.

[8] Daemen, J., Rijmen, V.: The design of Rijndael: AES – the Advanced Encryption Standard. Springer-Verlag, 2002, ISBN 3-540-42580-2.

[9] Demirci, H., Selçuk, A. A.: A Meet-in-the-Middle Attack on 8-Round AES. In: Nyberg, K. (ed.) FSE 2008. Lecture Notes in Computer Science Volume 5086, 2008, pp. 116–126.

[10] Gantz, J., Reinsel, D.: Extracting Value From Chaos. IDC, 2011.

[11] Kara, O.: Reflection Attacks on Product Ciphers. Cryptology ePring Archive, Report 2007/043, 2007.

[12] Kerckhoffs, A.: La cryptographie militaire. Journal des sciences militaires, Volume IX, 1883.

[13] Kokeš, J.: Cryptanalysis of Baby Rijndael. Diploma thesis, Faculty of Information Technology, Czech Technical University in Prague, 2013.

[14] Kokeš, J., Lórencz, R.: Baby Rijndael as a Reduced-size Model of AES/Rijndael. 2014. Not yet published (pending review).

[15] Matsui, M.: Linear Cryptanalysis Method for DES Cipher. Lecture Notes in Computer Science 765, 1994, ISBN 978-3-540-57600-6, pp 386-397.

[16] Schneier, B.: Memo to the Amateur Cipher Designer. Crypto-Gram, October 15, 1998. Available online: https://www.schneier.com/crypto-gram/archives/1998/1015.html [cit. 2015-06-04].

[17] Shamir, A.: Impossible differential attacks. CRYPTO '98, 1998. Available online: http://video.google.com/videoplay?docid=1283860161748060032&hl=en [cit. 2015-06-04].

[18] Wagner, D.: The Boomerang Attack. 6th International Workshop on Fast Software Encryption (FSE '99). Rome: Springer-Verlag. pp. 156–170.

Rychlé bezztrátové kompresní algoritmy vhodné pro hardware

Ing. Matěj Bartík
ČVUT FIT & CESNET
matej.bartik@fit.cvut.cz
1. ročník, prezenční forma

Dr. Ing. Sven Ubik
CESNET
ubik@cesnet.cz
Školitel

Ing. Pavel Kubalík, Ph. D.
ČVUT FIT
pavel.kubalik@fit.cvut.cz
Školitel specialista

***Abstrakt*—Výzkum se zabývá bezztrátovým kompresním algoritmem LZ4 (založeném na LZ77) a jeho vhodností pro kompresi multimediálních dat a univerzální paketovou kompresi pro síťové technologie [1]. Pro tyto účely je nutné zajistit co největší propustnost/latenci. Kompresní poměr sám o sobě není klíčový. Cílovou aplikací má být možnost spolupráce v reálném čase v oblastech citlivých na zpoždění.**

***Index Terms*—Bezztrátová, Komprese, Algoritmus, LZ4, LZ77, FPGA, Latence, Propusnost**

I. Úvod

V současné době se dostávají do popředí kompresní algoritmy, které se zaměřují na rychlost komprese a dekomprese namísto kompresního poměru. Tyto algoritmy (LZ4, LZO, FastLZ a jiné) jsou založeny na algoritmu LZ77 [2] (Lempel–Ziv 77). Tento algoritmus se řadí mezi jednoprůchodové slovníkové kompresní metody [3]. LZ77 je kvůli své rychlosti dekomprese využíván pro kompresi bitstreamu pro FPGA [4].

A. Současný stav

V současnosti probíhá vývoj nové verze platformy CESNET MVTP–4K (Modular Video Transfer Platform), která slouží k přenosu obrazových dat v reálném čase až do rozlišení 4K. Toto zařízení využívá k přenosu dat rozhraní optický 10G Ethernet. Mezi nevýhody současného zařízení patří obrovské nároky na přenosové pásmo. Některé z plánovaných funkcí nové verze MVTP:

1) Tvorba odlehčené verze MVTP pro přenos multimediálních dat (1080p25) pro koncerty uskutečňované na velké vzdálenosti. Současná verze MVTP pro jeden kanál využije přibližně 1,1 Gbps.
2) Příprava na podporu vyšších rozlišení (8K), větší snímkovou frekvenci nebo podporu 3D stereoskopického zobrazení. Pro všechny tyto aplikace je doporučené použití rozhraní SDI (Serial Digital Interface) ve variantě SDI-12G. Z názvu je patrné nutné přenosové pásmo. Je však možné některé informace nepřenášet, například zatemňovací oblast. Podpora 8K bude probíhat prostřednictvím rozdělení obrazu na kvadranty, každý o rozlišení 4K. Jako komunikační rozhraní může posloužit 40G Ethernet.

B. Motivace

Z požadavků na přenosové pásmo plánovaných inovovaných verzí MVTP je patrné, že potřebné přenosové pásmo je pouze o pár procent vyšší, než běžně používané standardy pro síťovou komunikaci. Je tedy nutné nalézt kompresní algoritmus vyhovujícím následujícím požadavkům (které vycházejí z vlastností rozhraní SDI):

- Zvládne všechny typy dat — univerzální bezztrátový algoritmus,
- vysoká propustnost (náročné video přenosy),
- nízká latence (pro spolupráci v reálném čase),
- potřebujeme šetřit přenosové pásmo — nízký overhead pro nekomprimovatelná data,
- malé nároky na zdroje FPGA,
- kompresní poměr je až poslední kritérium — potřebujeme ušetřit 20% – 15%.

Pokud se z nějakého důvodu nepovede zajistit dostatečný kompresní poměr, naší prioritou zabránit výskytu artefaktů v obraze. V námi používaném systému MVTP–4K existují metody (duplikace řádků, duplikace celých snímků), které zabraňují vzniku artefaktů při chybách (ztráty packetů). Ztráta snímku nám vadí méně, než výskyt artefaktů v obraze, které se vyskytuje při ztrátové kompresi (ať už na při kompresi jednotlivých snímků pomocí JPEG nebo mezisnímkové komprese typu H.264/H.265).

C. Řešený problém

Současný výzkum bezztrátových kompresních algoritmů v hardwaru se ubírá směrem maximální propustnosti. Existují články [5], [6] zabývající se implementací kompresních algoritmů pro co nejvyšší propustnost. Výsledná propustnost tohoto systému je 9,17 Gbps, respektive 8,5 Gbps [7]. V článku je použita kompresní metoda LZ77 s nerealisticky malým slovníkem a velikostí výhledu v jednotkách bitů. Autoři naznačují, že použitím pipeliningu bude možné navýšit propustnost nad 10 Gbps.

Většina autorů článků [8] se zabývá pouze popisem implementace kompresního algoritmu (typicky deriváty LZ77) bez detailnějšího popisu vlastností (propustnost [9], latence, použitá architektura FPGA), popřípadně bez jakéhokoliv hlubšího srovnání. V současné době směr(y) výzkumu neřeší tyto problémy (až na vzácné výjimky):

- Ověření vhodnosti (rychlé) bezztrátové komprese pro multimediální data [10],
- neexistuje žádná implementace rychlé kompresní metody v HW/FPGA,
- stávající algoritmy jsou interně (maximálně) 8–bitové, což je dnes hlavním důvodem pro nízkou propustnost [11],
- chybí metriku pro srovnání kompresních algoritmů (kompresní poměr/plocha, plocha/propustnost),
- chybí „corpus“ obsahující především obrazová data vhodná jak pro ztrátovou i bezztrátovou kompresi [12].

Současný směr výzkumu ztrátových a beztrátových kompresních algoritmů v softwaru se ubírá dvěma směry: Maximální kompresní poměr (především u ztrátových metod) a rychlost zpracování v reálném čase.

II. Společné znaky rychlých kompresním metod

Kompresní algoritmy optimalizované pro rychlost (de)komprese staví na stejných základech. V tomto případě jde o algoritmus LZ77 [2], [3]. Tento algoritmus lze charakterizovat několika vlastnostmi:

- Slovníková metoda (hledají se pouze shody bez potřeby statistických nebo kontextových výpočtů), není potřeba mnoho složitých výpočtů (růst propustnosti),
- je univerzální pro všechny typy dat,
- jednoprůchodová (vhodné pro zkrácení latence),
- asymetrická (doba/složitost dekomprese je vždy menší, než komprese),
- nedá se efektivně paralelizovat jinak než kompresí nezávislých bloků dat,
- prohledávání slovníku se děje lineárním průchodem (snižuje propustnost, zlepšuje kompresní poměr).

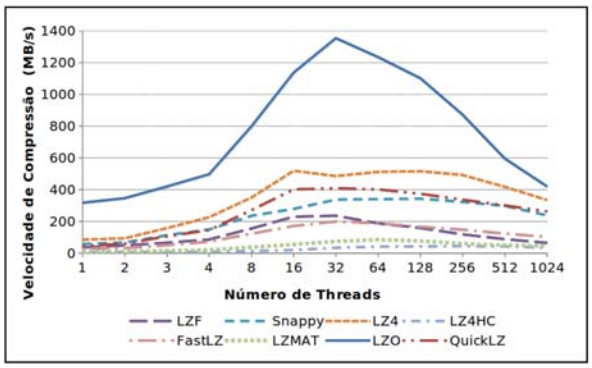
Rychlé kompresní metody (např.: LZ4 [13], LZO [14]) z LZ77 zachovávají všechny důležité vlastnosti. LZ77 sám o sobě je pouze popsán ve formě „pseudokódu“, který není optimalizován pro žádnou z běžně dostupných platforem a některé části nejsou optimální zhlediska výpočetní složitosti (lineární prohledávání). V případě rychlých algoritmů je lineární prohledávání nahrazeno hledáním přez rozptylovací (hash) tabulku. Tím se však připravujeme o možnost nalezení některých shod, tedy klesá kompresní poměr.

LZ77 i rychlé kompresní algoritmy jsou bajtově orientované. Rychlé kompresní algoritmy však využívají výhod moderních procesorových architektur (nezarovnané čtení a zápis do paměti), práce s většími bloky dat naráz (SIMD), softwarový pipelining, data/kód algoritmu pouze v cache, a množství dalších optimalizací, které jsou šité na míru cílové platformě [11]. Maximální rychlost (de)komprese algoritmů s procesorově závislými optimalizacemi dosahuje propusnosti paměťového subsystému cílového systému (v řádu GBps).

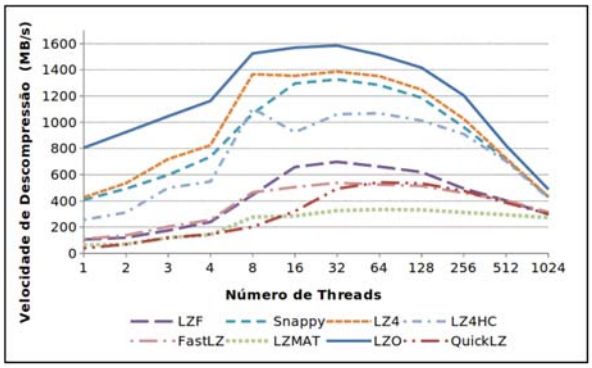
A. Rešerše rychlých kompresních algoritmů a jejich vhodnost pro multimediální data

Jak bylo řečeno výše [5], [6], algoritmy postavené na LZ77 mají potenciál dosáhnout propustnosti vyšší než 10 Gbps při použití v HW/FPGA. Současné rychlé algoritmy na standardní PC architektuře (x86) jsou pomalejší než LZ77 v HW/FPGA, ale vůči původní implementaci LZ77 mají stále vyšší propustnost. Je zde předpoklad, že při implementaci rychlých kompresních metod v HW/FPGA lze dosáhnout ještě vyšší propusnosti. Podmínkou pro úspěšnou portaci je přenesení stávajících optimalizací (pro standardní PC architekturu) a nalezení nových optimalizací, které by těžily z výhod HW/FPGA zpracování.

Dalším bodem rešerše je vhodnost těchto algoritmů pro kompresi multimediálních dat v reálném čase. V publikaci [10] se autoři zabývají vlastnostmi (rychlost komprese (Obr. 1.), dekomprese (Obr. 2.), kompresní poměr (Obr. 3.), škálovatelnost, real–time zpracování) velkého množství rychlých kompresních algoritmů. Jako testovací data autoři zvolili multimediální stream o rozlišení 4K60p kombinovaný s 3D stereoskopickým zobrazením. Jedním z výsledků je konstatování, že v současné době existují pouze dva rychlé bezztrátové kompresní algoritmy, které svými vlastnostmi umožňují real–time přenos multimediálních dat s kompresním poměrem vhodným pro naše použití a to LZ4 [13] a LZO [14].

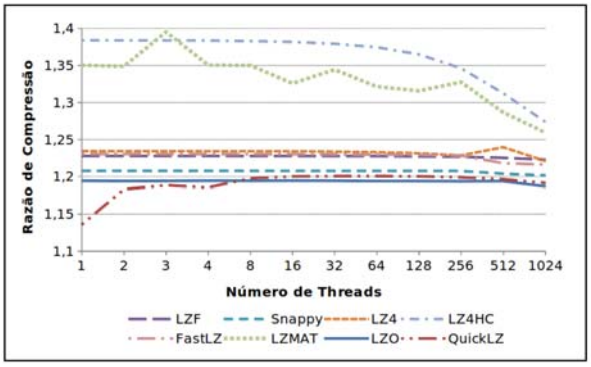


Obrázek 1. Propustnost komprese (MB/s) v závislosti na počtu vláken [10]



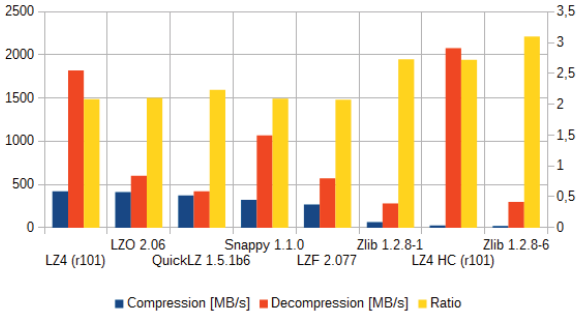
Obrázek 2. Propustnost dekomprese (MB/s) v závislosti na počtu vláken [10]

Metoda LZO dosahuje vyšší propustnosti při kompresi než LZ4. Rychlost dekomprese je možné považovat za srovnatelnou. Průměrný kompresní poměr LZ4 je o 5% vyšší než u LZO. Na základě dalších výkonostních srovnání (Obr. 4.) [13] lze konstatovat, že oba algoritmy jsou pro naše účely srovnatelné. Pro další výzkum jsem zvolil orientaci na kompresní algoritmus LZ4. Jedna z výhod LZ4 oproti LZO spočívá v garantovaném overheadu pro nekomprimovatelná data (0,4%), což lze považovat za výhodu pro síťové aplikace (packetová



Obrázek 3. Kompresní poměr v závislosti na počtu vláken [10]

komprese). Další výhodou je „svobodnější“ licenci (BSD vs. GPL), která umožňuje případné výsledné IP jádro integrovat do komerečních produktů.



Obrázek 4. Další srovnání některých rychlých kompresních algoritmů [13]

III. Vlastní výzkum

Vlastní výzkum se dělí na část teoretickou (detailní rozbor kompresního algoritmu LZ4) a praktickou (experimentální měření a implementace LZ4 ve VHDL).

A. Rozbor kompresního algoritmu LZ4

Provedl jsem analýzu referenčního zdrojového kódu (v jazyce C) ve verzi r127 a po vypuštění procesorových optimalizací jsem sestavil pseudokód LZ4. Pseudokód předpokládá následující parametry:

- I : Vstupní buffer
- O : Výstupní buffer
- Isize : Velikost vstupního bufferu

```
pointer ip = 0; // address to I
pointer op = 0; // address to O
hash_table HT; // Zeroed

while (ip < Isize-5) {
  h_adr = read U32 *ip, calculate hash;
  read possible match address HT(h_adr);
  store current address HT(h_adr)=ip;
  if !(match found) ||
     !(distance < offset_limit) ip++;
  else {
    if (ip > Isize-12) break;

    // writing to O buffer
    encode Token;
    encode Literals length;
    copy literals;
    encode Offset;
    encode Match length;

    increase input and output pointers;
  }
}

encode last literals;
return output pointer (data size);
```

Z pseudokódu je patrné, že algoritmus lineárně prohledává vstupní buffer. Algoritmus pracuje s rozptylovací (hash) tabulkou pro ukládání potenciálních shod. Výpočet hashe se děje pomocí multiplikativního hashování [15] s prvočíslem 2654435761. Toto prvočíslo je nejblíže hodnotě $\frac{2^{32}}{\phi} = 2654435769$, kde ϕ je hodnota Zlatého řezu. V případě FPGA může být implementace provedena pomocí DSP bloků. Tímto se odstraní největší nedostatek LZ77, a to lineární prohledávání paměti na shody. Drobným vylepšením LZ4 je možnost překryvu shod.

Na vypočítanou adresu (do hashovací tabulky) se ukládá původní adresa dat a zároveň se zkoumá, zda na dříve uložené adrese nejsou shodná data. V případě shody a dodržení maximální vzdálenosti shody se do výstupního bufferu uloží LZ4 sekvence definovaná v Tabulce I.

Jméno	Účel	Velikost
Token	Match & Literals length (nibbles)	1 Byte
Literals length	Literal length extension (optional)	n Bytů
Literals	Copied literals	n Bytů
Offset	Match distance (1–65535)	2 Bytů
Match length	Match length extension (optional)	n Bytů

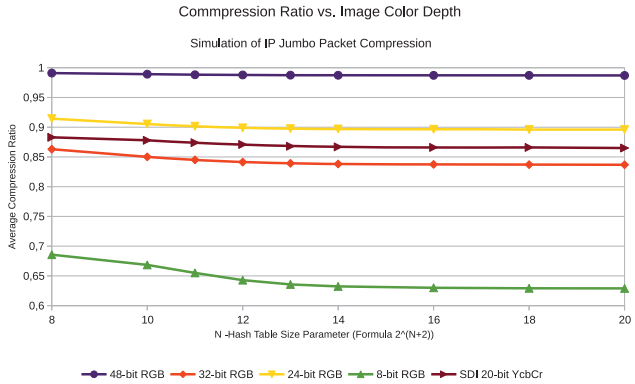
Tabulka I
LZ4 Sequence format [13]

Zbylá data jsou na konci zakódována jako samostatná sekvence. Referenční implementace metody LZ4 pracuje s operandy různých délek (8, 16, 32 a 64 bitů) a čtení/zápis do paměti probíhá z nezarovnaných adres. Proto je s výhodou použit pokročilý paměťový řadič v moderních procesorových architekturách. Například kopírování literálů do výstupu probíhá s maximální šířkou slova použitého procesoru, není tedy použito kopírování po jednotlivých bytech. Stejně tak se algoritmus pokouší alespoň „softwarově“ provádět zarovnaný zápis/čtení. To se pozitivně projevuje na celkové propustnosti [16]. Jediným parametrem (N) ovlivňující kompresní

poměr, je velikost hashovací tabulky o velikosti 2^N. Změnou tohoto parametru se zvyšuje/snižuje pravděpodobnost nálezu shody v rozptylovací tabulce.

B. Ověření kompresního poměru LZ4 nad vlastními daty

Důležitou částí rešerše LZ4 bylo ověřit vhodnost kompresního algoritmu pro multimediální data. Pro vlastní testování jsem použil několik snímků, z archivu CineGrid [12], které neprošly žádnou z metod ztrátové komprese. Vybrané snímky by měly reprezentovat všechny typy scén (barevnost, entropie), které se vyskytují při přenosu multimediálních dat. Originální snímky mají nadstandardní 48–bitovou hloubku barev. Na posledních bitech každé barevné složky se tedy vyskytuje „náhodný šum“. Snímky s touto bitovou hloubkou jsou téměř nekomprimovatelné, jak je patrné z tabulky II (její vizualizace je na obrázku 5). Proto jsem se rozhodl snížit bitovou hloubku (snížit entropii) pomocí grafického editoru na běžně používané bitové hloubky barev. Ukázalo se, že s klesající bitovou hloubkou pro barevné složky dochází k nárustu kompresního poměru. Je pravděpodobné, že se jednotlivé pixely začínají „více opakovat“. Druhý test se zabýval kompresním poměrem našich vlastních dat, zachycených ve formě paketů ze systému MVTP (SDI 20–bit YCbCr).



Obrázek 5. Vztah mezi kompresním poměrem a bitovou hloubkou snímku.

Z naměřených výsledků je možné konstatovat, že kompresní poměr se u běžně používaných bitových hloubek (20, 24 a 32–bitů) pohybuje mezi 10%—15%. To není mnoho, ale pro námi zamýšlené použití se jeví kompresní poměr jako dostačující. Platforma MVTP nepřenáší „zatemňovací“ oblasti, které se počítají do celkové propustnosti média. Protože komprimované datové packety jsou již „předkomprimovány“ odebráním „zatemňovacích“ oblastí, bude celková úspora přenosového pásma ještě vyšší.

C. Portace LZ4 na syntetizovatelný kód

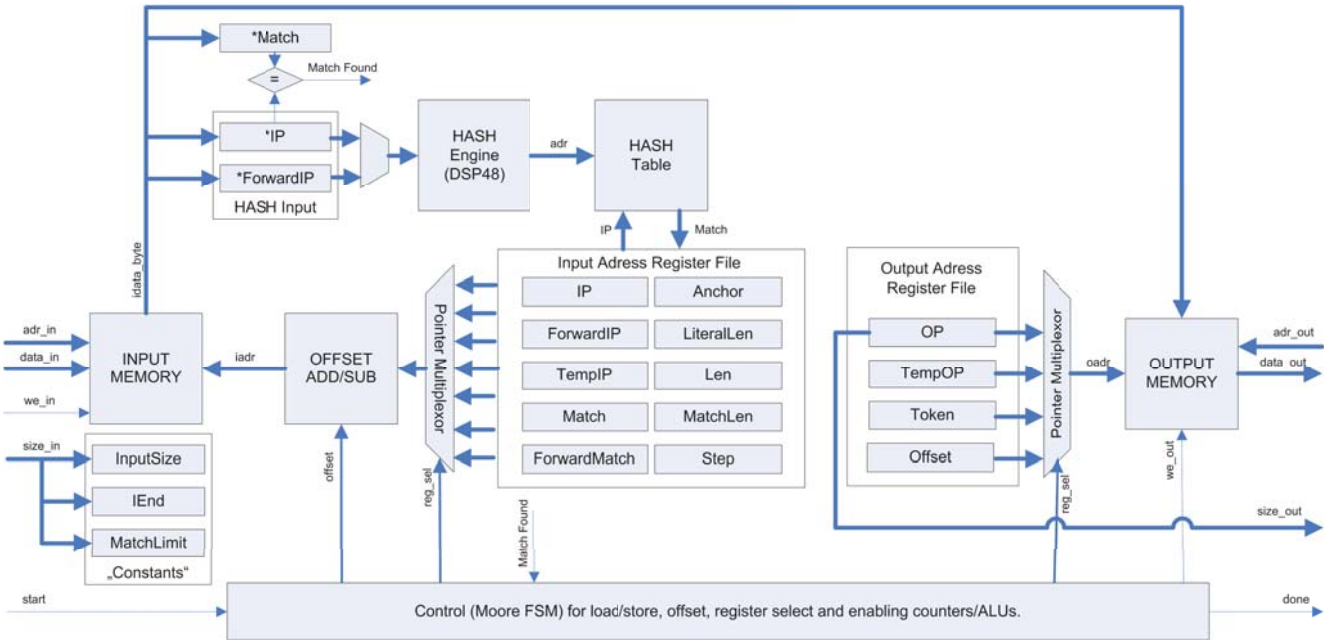
Protože LZ4 je stejně jako LZ77 asymetrická kompresní metoda, je rychlost dekomprese (nebo její složitost) vyšší (nižší) než u komprese. Proto jsem se v první řadě soustředil na vývoj, implementaci a testování kompresní části LZ4. Je zde předpoklad, že pokud se podaří dosáhnout potřebných parametrů komprese (rychlost, propustnost) na straně kompresní části, dosáhne se potřebných parametrů i na straně dekomprese.

Tabulka II
ZÁVISLOST KOMPRESNÍHO POMĚRU, VELIKOST HASHOVACÍ TABULKY, BITOVÁ HLOUBKA BAREVNÝCH SLOŽEK A ZPŮSOBU KÓDOVÁNÍ BAREVNÝCH SLOŽEK

8–bit (RGB)							
File	**N**	**8**	**10**	**11**	**12**	**13**	**14**
00000050.tif		0,113	0,109	0,108	0,108	0,108	0,107
00000200.tif		0,628	0,614	0,601	0,591	0,585	0,583
00001000.tif		0,572	0,556	0,545	0,536	0,531	0,529
00005000.tif		0,743	0,720	0,698	0,679	0,670	0,666
00010000.tif		0,819	0,802	0,787	0,772	0,763	0,758
00015000.tif		0,623	0,607	0,598	0,590	0,585	0,582
00020000.tif		0,729	0,712	0,701	0,689	0,680	0,676
Average		0,686	0,669	0,655	0,643	0,636	0,632
24–bit (RGB)							
File	**N**	**8**	**10**	**11**	**12**	**13**	**14**
00000050.tif		0,181	0,176	0,174	0,173	0,172	0,172
00000200.tif		0,854	0,844	0,840	0,837	0,835	0,834
00001000.tif		0,850	0,836	0,831	0,827	0,826	0,825
00005000.tif		0,972	0,961	0,955	0,951	0,948	0,947
00010000.tif		0,978	0,973	0,970	0,968	0,966	0,966
00015000.tif		0,887	0,875	0,871	0,869	0,869	0,869
00020000.tif		0,945	0,943	0,943	0,941	0,941	0,941
Average		0,914	0,905	0,902	0,899	0,898	0,897
32–bit (RGB)							
File	**N**	**8**	**10**	**11**	**12**	**13**	**14**
00000050.tif		0,168	0,165	0,165	0,165	0,165	0,165
00000200.tif		0,792	0,778	0,772	0,768	0,766	0,764
00001000.tif		0,768	0,750	0,747	0,743	0,741	0,740
00005000.tif		0,930	0,915	0,907	0,901	0,898	0,897
00010000.tif		0,951	0,941	0,934	0,929	0,926	0,924
00015000.tif		0,823	0,808	0,804	0,802	0,801	0,800
00020000.tif		0,914	0,909	0,907	0,905	0,904	0,904
Average		0,863	0,850	0,845	0,841	0,839	0,838
48–bit (RGB)							
File	**N**	**8**	**10**	**11**	**12**	**13**	**14**
00000050.tif		0,249	0,244	0,243	0,241	0,241	0,241
00000200.tif		0,958	0,955	0,954	0,954	0,953	0,953
00001000.tif		0,992	0,989	0,987	0,986	0,986	0,985
00005000.tif		1,003	1,003	1,002	1,002	1,002	1,002
00010000.tif		1,001	1,000	0,999	0,998	0,998	0,998
00015000.tif		0,990	0,989	0,988	0,988	0,987	0,987
00020000.tif		1,002	0,999	0,999	0,999	0,999	0,999
Average		0,991	0,989	0,988	0,988	0,988	0,987
SDI 20-bit (YCbCr)							
File	**N**	**8**	**10**	**11**	**12**	**13**	**14**
Average		0,883	0,878	0,874	0,871	0,868	0,867

Prvním krokem bylo přepsání sekvenčního kódu v jazyce C do popisu vhodného pro hardware (VHDL). Vnitřní architektura (Obrázek 6.) je bloku je principiálně jednoduchý stavový automat, který vykonává jednotlivé instrukce algoritmu v jazyce C. Pokud to bylo nutné, byly komplexní instrukce rozděleny na „mikroinstrukce“. Tento blok pro maximální jednoduchoust pracuje pouze v 8–bitovém režimu. Výsledky syntézy pro současnou a budoucí verzi MVTP jsou v tabulkách III a IV.

I když popisovaný blok má zanedbatelnou propustnost a obrovskou latenci, obsahuje všechny části (struktury), které se



Obrázek 6. Vnitřní architektura LZ4 bloku.

Tabulka III
XC6VLX240T–2FF1156 Resource Utilization

Slice Logic Utilization	Used	Available	Ratio
Number of occupied Slices	207	37680	1%
Number of Slice Registers	445	301440	1%
Number of Slice LUTs	733	150720	1%
Number used as logic	676	733	92%
Number used exclusively as route-thrus	57	563	8%
Number using RAMB36E1	8	416	1%
Number using RAMB18E1	1	832	1%
Number of DSP48E1s	3	768	2%
Number of bonded IOBs	78	600	13%
Average Fanout of Non-Clock Nets	4.13		
Maximum frequency (minimum latency)	220.556 MHz (4.534 ns)		

Tabulka IV
XC7A100T-3FFG676 Resource Utilization

Slice Logic Utilization	Used	Available	Ratio
Number of occupied Slices	251	15850	1%
Number of Slice Registers	375	126800	1%
Number of Slice LUTs	762	63400	1%
Number used as logic	715	762	93%
Number used exclusively as route-thrus	47	563	7%
Number using RAMB36E1	8	135	1%
Number using RAMB18E1	1	270	1%
Number of DSP48E1s	3	240	2%
Number of bonded IOBs	78	300	26%
Average Fanout of Non-Clock Nets	4.14		
Maximum frequency (minimum latency)	173.040 MHz (5.779 ns)		

budou vyskytovat i v optimalizované verzi. Jako „úzká hrdla“ limitující propustnost bloku se jeví především 8–bitové zpracování (některé části LZ4 mohou být až 64–bitové). Relativně nízká pracovní frekvence obvodu je způsobena paměťovým řadičem Block RAM, který podporuje operace s offsety. Dále je patrné, že maximální počet bloků, které se vejdou do FPGA je omezen především požadovanou velikostí vstupních a výstupních bufferů (implementované v Block RAM). U starších FPGA architektur může být omezující i počet DSP bloků.

IV. Další výzkum a cíl dizertační práce

Na základě rešerše a současných výsledků je možné pokračovat ve výzkumu v následujících směrech:

A. Metody pro přenos procesorových optimalizací na FPGA

Cílem je zdokumentovat procesorové a jiné optimalizace, kterými se odlišují rychlé kompresní algoritmy od původní LZ77, najít v nich průnik nejčastěji používaných a navrhnout způsoby, jak tyto optimalizace přenést do HW/FPGA tak, aby měli pozitivní přínos.

B. Tvorba nových optimalizicí těžících z FPGA architektury

Dalším cílem je návrh nových optimalizací, který by využili rozdílných vlastností FPGA a dnešních procesorových architektur. Takovým příkladem může být náhrada softwarového pipelingu (který je závislý na překladači) za skutečný hardwarový pipeling, využití víceportových Block RAM pro lepší paralelní zpracování.

V současné době je například navržená optimalizace, která dokáže „smazat“ obsah rozptylovací tabulky v nižším počtu taktů, než to dokáže procesor lineárním průchodem paměťových buňek. Obsah rozptylovací tabulky musí být „vynulován“ před každým během algoritmu.

C. Vytvoření nativně 64–bitového kompresního algoritmu

Protože hlavním limitem propusnosti je vnitřní 8–bitová architektura, nabízí se možnost vytvoření nativně 64–bitového algoritmu. Nutná hodinová frekvence k zajištění 10 Gbps propusnosti je 156,25 MHz. Tento kmitočet je nižší, než syntézní výsledky běžných obvodů pro FPGA posledních generací.

V. Závěr

Tato práce se zabývá rychlými bezztrátovými kompresními algoritmy a možností jejich využití pro kompresi multimediálních dat v reálném čase. Na základě rešerše jsem zvolil pro experimenty kompresní algoritmus LZ4. Citované publikace a vlastní experimentalní výsledky potvrzují vhodnost pro navržené použití (přenos multimediálních dat). Byl proveden rozbor vlastností a principů činnosti algoritmu LZ4 tak, aby mohl být efektivně implementován v HW/FPGA. Pro tyto účely bylo nutné pochopit procesorově závislé optimalizace LZ4 a navrhnout nové optimalizace, které by těžily z vlastností FPGA architektury. Práce se dále zabývá analýzu kompresního poměru multimediálních dat v závislosti na bitové hloubce barevných složek.

Poděkování

Tento výzkum byl podpořen projektem SGS15/020/OHK3/1T/18.

Reference

[1] Munteanu D. An FPGA-based Network Processor for IP Packet Compression. [Online]. Available: http://pages.cpsc.ucalgary.ca/˜carey/papers/2005/FPGA.pdf

[2] J. Ziv and A. Lempel, “A Universal Algorithm for Sequential Data Compression,” IEEE Transactions on Information Theory, vol. 23, no. 3, pp. 337–343, 1977.

[3] Solomon, D.: Data Compression: The Complete Reference (Fourth ed.). 20007, Springer. ISBN 9781846286032.

[4] Khu, A.: Xilinx FPGA Configuration Data Compression and Decompression. [Online]. Available: http://tinyurl.com/nhhgqbj

[5] Mehboob, R.; Khan, S.A.; Ahmed, Z.; Jamal, H.; Shahbaz, M., ”Multigig lossless data compression device,”Consumer Electronics, IEEE Transactions on , vol.56, no.3, pp.1927,1932, Aug. 2010, doi: 10.1109/TCE.2010.5606348

[6] El Ghany, M.A.A.; Salama, A.E.; Khalil, A.H., ”Design and Implementation of FPGA-based Systolic Array for LZ Data Compression,”Circuits and Systems, 2007. ISCAS 2007. IEEE International Symposium on , vol., no., pp.3691,3695, 27-30 May 2007, doi: 10.1109/ISCAS.2007.378644

[7] Papadopoulos, K.; Papaefstathiou, I., ”Titan-R: A Reconfigurable Hardware Implementation of a High-Speed Compressor,”Field-Programmable Custom Computing Machines, 2008. FCCM ’08. 16th International Symposium on , vol., no., pp.216,225, 14-15 April 2008 doi: 10.1109/FCCM.2008.14 [Online]. Available: http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=4724904

[8] Rigler, S.; Bishop, W.; Kennings, A., ”FPGA-Based Lossless Data Compression using Huffman and LZ77 Algorithms,”Electrical and Computer Engineering, 2007. CCECE 2007. Canadian Conference on , vol., no., pp.1235,1238, 22-26 April 2007, doi: 10.1109/CCECE.2007.315 [Online]. Available: http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=4232974

[9] NAQVI S. Optimized RTL design and implementation of LZW algorithm for high bandwidth applications. [Online]. Available: pe.org.pl/articles/2011/4/68.pdf

[10] Ruan Delgado Gomes, Yuri Gonzaga Gonçalves da Costa, Lucenildo Lins Aquino Júnior, Manoel Gomes da Silva Neto, Alexandre Nóbrega Duarte, and Guido Lemos de Souza Filho. 2013. A solution for transmitting and displaying UHD 3D raw videos using lossless compression. In Proceedings of the 19th Brazilian symposium on Multimedia and the web (WebMedia ’13). ACM, New York, NY, USA, 173-176.

[11] Kane, J.; Qing Yang, ”Compression Speed Enhancements to LZO for Multi-core Systems,”Computer Architecture and High Performance Computing (SBAC-PAD), 2012 IEEE 24th International Symposium on , vol., no., pp.108,115, 24-26 Oct. 2012

[12] CineGrid Exchange. [Online]. Available: http://cinegrid.org/

[13] Collet, Y.: RealTime Data Compression: Development blog on compression algorithms. [Online]. Available: tinyurl.com/qc9yve4

[14] Oberhumer, M.:LZO real-time data compression library [Online]. Available: http://www.oberhumer.com/opensource/lzo/

[15] Knuth, D. E.: The Art of Computer Programming, Volume 3: (2Nd Ed.) Sorting and Searching. Redwood City, CA, USA: Addison Wesley Longman Publishing Co., Inc., 1998, ISBN 0-201-89685-0.

[16] Fiedler, O.: LZ-Family Data Compression Methods, Bachelor Thesis, 2014.

High Performance Computing on Low Power Devices

Vojtěch Nikl

Computer Science and Engineering, 1st year, full-time study
Supervisor: Jiří Jaroš

Faculty of Information Technology, Brno University of Technology
Božetěchova 2, 612 66 Brno

inikl@fit.vutbr.cz

Abstract. Nowadays, the power efficiency of modern processors is becoming more and more important next to the overall performance itself. Many programming tasks and problems do not scale very well with higher number of cores due to being memory or communication bound, therefore it is often not beneficial to use faster chips to achieve better runtimes. In this case, employing slower low power processors or accelerators may be much more efficient, mainly because it is possible to get the same results using much less energy and possibly after the same amount of time, given the algorithm can scale to higher number of cores after applying some adjustments fitting the low power architecture. This paper describes the benefits of using low power chips for building an HPC cluster, the group of algorithms where this approach can be useful, results achieved so far and future plans.

Keywords: HPC, parallelism, low power, processor architecture, supercomputers, k-Wave toolbox, MPI, OpenMP, performance evaluation, numerical methods

1 Introduction

Even though computer processors have come a long way in terms of performance, there are still many tasks and problems which require large amounts of computing power to be successfully solved. For some time, hardware engineers haven't been purely focusing on raw performance, but the energy consumption has also become a very important factor. Using low powered processors can be much more efficient for certain kinds of algorithms, mainly for memory and communication bound problems. Unfortunately, this is where algorithms' scalability come into play.

Some algorithms scale very well. For example, *Finite difference* or *Partial differential equation* techniques, used for modeling computational electrodynamics, have constant communication complexity per core for any given size of the domain, because each unit communicates only with its closest neighbors. *Computational fluid dynamics methods* and especially its *Lattice Boltzmann* class also scale very well, although the memory demands are often rather high.

However, there is also the opposite class of algorithms, which scale very poorly. Usually, these algorithms require some sort of global or semi-global communication. Typical example are *Spectral methods* of *Partial differential equations*. These methods work with domains, represented by matrices, where each point of the matrix represents a value of some given attribute. The use of the *fast Fourier transform* is very often involved, which requires one or more so called global all-to-all communications. This results in a very intensive communication overhead especially for bigger domains and the communication dominates over the computation. Employing faster chips brings almost no benefit. This class of algorithms can be more suitable for low power processors, because the computation/communication ratio positively increases and there is more room for overlapping the communication and computation steps, while getting the results using much less energy.

2 Motivation

Today's supercomputers are usually based on the x86 architecture, specifically the Intel Xeon one. One of many examples is the Anselm cluster[1], located in Ostrava, Czech Republic. It consists of 209 2x8 core Intel Xeon E5-

[1] https://docs.it4i.cz/anselm-cluster-documentation/hardware-overview

2665 2.6GHz nodes, each with atleast 64GB of RAM. Each node requires approximately 230W of energy under full load, while providing about 400 GFlop/s of theoretical performance. Most of that energy is dissipated into heat and therefore it requires very intensive and expensive cooling system. Some systems chose a little bit different approach towards higher power effectiveness. One of them is the Fermi cluster[2], located in the Cineca organization, Bologna, Italy. It consists of 10,240 nodes, each integrating 16 core IBM PowerA2 1.6GHz processor. While the overall performance per node is about half of the Anselm one's, the peak power consumption is only 55W. This results in almost twice as good performance per Watt ratio. *The Green 500 list*[3] provides a ranking of the most energy-efficient supercomputers in the world and Fermi is very close to the top at the 49th place, having over 2 GFlop/s per Watt. The most efficient supercomputer has about 5.3 GFlop/s per Watt.

Searching for even better efficiency, the low power processors used in tablets, smartphones and embedded systems seem to be the best bet. They have come a long way and today's smartphones have up to 8-core processors, which are capable of doing some very intensive tasks, such as recording and playing videos in 4K resolution. These chips are built with an emphasis on low power consumption to extend the battery life as much as possible. For example a "soon to be available" ARM based chip nVidia Tegra X1[4] and its GPU can provide 0.512 TFlop/s in single precision while consuming only about 10W of energy. The fastest nVidia GPU, Titan X[5], provides 6.14 TFlops with 250W of power consumption. The Tegra is more than twice as efficient while providing more than 50 GFlops per Watt!

Unfortunately, this approach has a few downsides. Since the low power processors are less powerful, it is often necessary to employ much more of them to reach the same level of overall performance. As mentioned earlier, the scalability of different algorithms may become a problem. For example if it is required to have 8x more cores in order to reach the same theoretical raw performance, the number of messages sent during one all-to-all communication phase increases by a factor of 64!

Another problem may be the amount of system memory. Fermi has 16GB of RAM per node, which can quickly become a limiting factor when calculating extensive simulations with many domains. The Tegra X1 chip supports only up to 4GB of LPDDR4 RAM, so the problem escalates even more. For example, a realistic ultrasound simulation performed by the k-Wave toolbox, introduced in the next section, typically uses a grid size of 1024^3. The amount of memory required to run this simulation is about 128GB, which means at least 2 nodes of Anselm (32 cores), but at least 8 nodes of Fermi (128 cores) or 32 Tegra chips (256 cores), excluding the memory requirements for an operating system, MPI buffers etc. A simulation with a grid size of 4096^3, which is one of the future goals of k-Wave, requires about 8192GB of RAM, so at least 2048 Tegra chips (16,384 cores) need to be employed. Next to the need of high overall performance, memory demands are very often another reason why it is important to use a higher number of cores in many HPC areas, assuming a given algorithm can even provide scalability this high.

The main goal is to explore the possibility of using such a low power architecture cluster for calculating a subset of specific tasks, which are less suitable for current clusters mentioned above. The main focus will be aimed towards spectral methods and algorithms and the exploration of new extreme scaling techniques. And here a very important question arises. Is it actually worth it, considering all pros and cons, to use these low power processors as the main computational units in a cluster and get a very efficient machine in terms of both initial and running costs, but maybe not so efficient in terms of application scalability and overall performance?

3 The k-Wave project

The k-Wave toolbox [1] is designed to simulate ultrasound wave propagation in soft-tissues and bone, modelled as fluid and elastic media, respectively. The simulation of ultrasound wave propagation through biological tissue has a wide range of practical applications including planning therapeutic ultrasound treatments of various brain disorders such as brain tumours, essential tremor, and Parkinson's disease. The major challenge is to ensure the ultrasound focus is accurately placed at the desired target within the brain because the skull can significantly distort it. Performing accurate ultrasound simulations, however, requires the simulation code to be able to exploit several thousands of processor cores and work with datasets on the order of tens of TB.

In the k-Wave toolbox, the k-space pseudospectral method is used to solve the system of governing equations described in detail by Treeby in [2]. These equations are derived from the mass conservation law, momentum

[2] http://www.hpc.cineca.it/content/ibm-fermi-user-guide
[3] http://www.green500.org/
[4] http://www.nvidia.com/object/tegra-x1-processor.html
[5] http://maxwell.nvidia.com/titan-x

conservation law, and an empirically derived acoustic pressure-density relation that accounts for acoustic nonlinearity, absorption, and heterogeneity in the material properties [2].

The k-space and pseudospectral methods gain their advantage over finite difference methods due to the global nature of the spatial gradient calculations [3]. This permits the use of a much coarser grid for the same level of accuracy. However, the global nature of the gradient calculation, in this case using the 3D fast Fourier transform (FFT), introduces additional challenges for the development of an efficient parallel code. Specifically, the FFT requires a globally synchronizing all-to-all data exchange. This global communication can become a significant bottleneck in the execution of spectral models. Fortunately, considerable effort has already been devoted to the development of distributed memory FFT libraries, such as FFTW [4] or PFFT [5], that show reasonable scalability of up to tens of thousands of processing cores.

A recently introduced hybrid MPI/OpenMP decomposition approach of calculating the FFTs [6] is able to scale almost linearly up to 16,384 cores of the Fermi cluster (see Fig. 1). This is a big asset to k-Wave, which was previously limited by its pure MPI 1D decomposition, which allowed to employ no more than about 2048 cores. This hybrid approach requires one global MPI all-to-all communication, which can be very time demanding, especially for bigger domains. Fig. 2 shows that on typical Intel Xeon Sandy Bridge based clusters, Zapat and Anselm, the communication step takes about 80% of the whole computation.

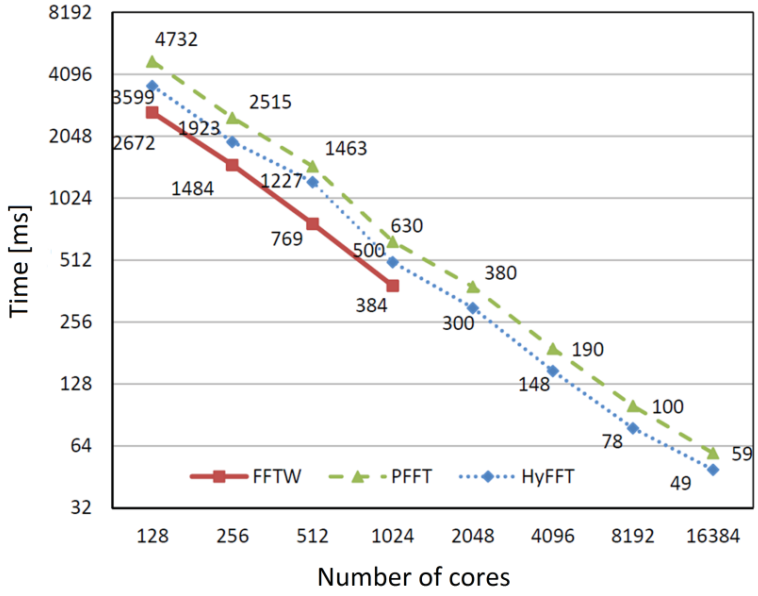


Figure 1: FFT on Fermi, 1024^3 grid points

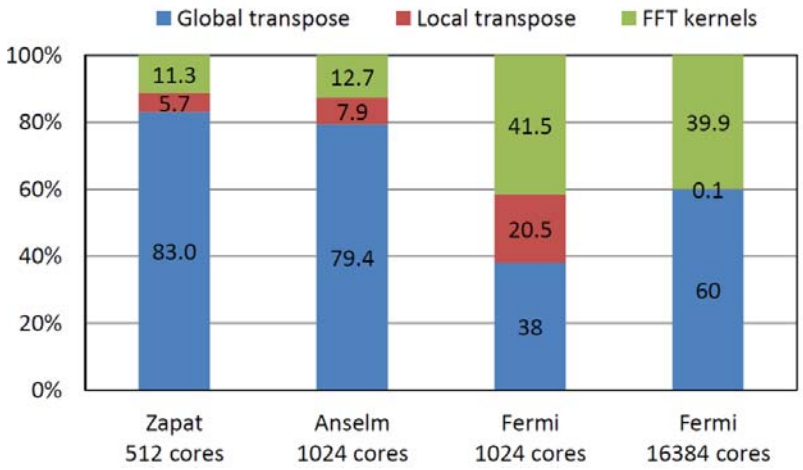


Figure 2: Hybrid FFT Time Distribution, 1024^3 grid points

This is very inefficient in terms of power consumption, because the CPUs are running during the whole time. On Fermi, however, the communication part goes down to about 60% due to slower CPUs, which may allow for better utilization of resources, such as overlapping the communication and computation of 2 or more FFTs running in parallel. This shows that for communication bound algorithms, it is not the performance performance per core that matters the most, but rather a fast interconnecting network and hiding the application latencies and communication overheads.

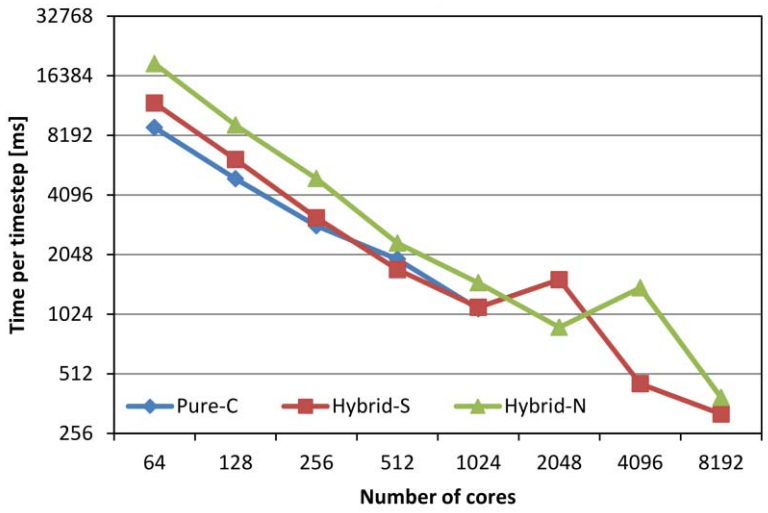


Figure 3: k-Wave on SuperMUC, 1024^3 grid points

A whole k-Wave simulation using the hybrid decomposition was benchmarked on the SuperMUC cluster[6] using up to 8192 cores [7]. Fig. 3 shows that both the Hybrid-S (1 MPI process per socket, 1 thread per core) and Hybrid-N (1 MPI process per node, 1 thread per core) setups offer almost 4 times higher performance over Pure-C (1 MPI process per core), which yields efficacy of almost 50%, which is not so bad considering the code is proven to be communication and memory bound. The peaks in execution time directly correspond to the communication share. In a typical run, the communication share is about 50%, while in those exceptional cases the communication share springs up to 75%. The profile confirmed that the distributed transposition is not done optimally and a custom routine may need to be implemented to ensure the correct behavior.

4 Future plans

During late summer of 2015, a small cluster consisting of 18 Tegra X1 kits (distributed as16 compute nodes, 1 I/O node and 1 login node) will be bought and installed at our faculty. This architecture is able to run on Linux based operating system with most of its standard compilers and libraries used in HPC, such as GNU package, MPI libraries, FFTW, Matlab, CUDA etc. k-Wave has already been successfully compiled and run on the ARM processor of Tegra K1[7], the predecessor of X1, and the next step is to port it to X1 while making efficient use of the heterogeneous CPU+GPU architecture. Previous section showed that the k-space pseudospectral method of k-Wave is a good candidate for low power architectures, mainly because it is very memory and communication bound. The scalability is tested to be good enough to utilize a high number of cores, while the power consumption and heat dissipation of running the whole simulation is expected to be much lower. This means that a single surgery is expected to be much cheaper for the patient, while keeping its planning and simulation time under clinically important 24 hours.

This cluster will serve as a "proof of concept" tool for verifying the (dis)advantages of this architecture over the common architectures on suitable problems. The current task is to benchmark numerical methods for solving partial differential equations, mainly finite differential, finite element, spectral and boundary element methods,

[6] https://www.lrz.de/services/compute/supermuc/systemdescription/
[7] http://www.nvidia.com/object/tegra-k1-processor.html

and compare their performance, accuracy and memory demands on various problems. After this is done on a common cluster, later on the main goal will be to port these algorithms on the tegra cluster and compare the performance, energy demands, efficiency etc.

The goal of the Ph.D. thesis is to define a set of algorithms suitable for low power architectures, port the code, exploit the scalability as much as possible, adjust the properties of the algorithms to fit the architecture and benchmark the performance, power consumption and resource utilization. The conclusion will be: *Algorithm X can run on hardware Y more effectively (with precise specification what that means) under conditions Z.*

5 Conclusion

This paper described the motivation behind using low power architectures for solving specific tasks instead of the common architectures used today. The nVidia Tegra X1 processor was presented as one of the most power efficient architectures, while providing performance comparable to today's desktop PCs. The k-Wave project described above is one of many applications suitable for this architecture due to being memory and communication bound and being able to scale to a very high number of cores at the same time. The presented future work is focusing on the tegra cluster, which is going to verify the benefits and advantages of low power architectures for certain kinds of algorithms. The final goal of the Ph.D. thesis is to identify these sets of algorithms, adjust and benchmark them on both the common and low power architecture and define the conclusion about advantages of low power architectures.

References

1. B. E. Treeby and B. T. Cox. k-Wave: MATLAB toolbox for the simulation and reconstruction of photoacoustic wave fields. *Journal of Biomedical Optics*, 15(2):021314, 2010.
2. B. E. Treeby, J. Jaros, A. P. Rendell, and B. T. Cox. Modeling nonlinear ultrasound propagation in heterogeneous media with power law absorption using a k-space pseudospectral method. *The Journal of the Acoustical Society of America*, 2012(131):4324–4336, 2012.
3. T. D. Mast, L. P. Souriau, D.-L. D. Liu, M. Tabei, A. I. Nachman, and R. C. Waag. A k-space method for large-scale models of wave propagation in tissue. *IEEE Trans. Ultrason. Ferroelectr. Freq. Control*, 48(2):341–354, 2001.
4. M. Frigo and S. G. Johnson. The Design and Implementation of FFTW3. *Proceedings of the IEEE*, 93(2):216–231, 2005.
5. P. Michael. PFFT-An extension of FFTW to massively parallel architectures. *Society for Industrial and Applied Mathematics*, 35(3):213–236, 2013.
6. V. Nikl and J. Jaros. Parallelisation of the 3D Fast Fourier Transform Using the Hybrid OpenMP/MPI Decomposition. In *Mathematical and Engineering Methods in Computer Science, LNCS 8934*, pages 100–112. Springer International Publishing, 2014.
7. J. Jaros, V. Nikl and B. E. Treeby. Large-scale Ultrasound Simulations Using the Hybrid OpenMP/MPI Decomposition, EASC 2015, Edinburgh (accepted paper).

Obecná polymorfní logika a její složitost

Radek Tesař
Informatika a výpočetní technika, ročník druhý, kombinované studium
Školitel: Richard Růžička

FIT VUT Brno
Božetěchova 2, Brno

itesar@fit.vutbr.cz

Abstrakt. Hypotéza: Existuje třída problémů, kterou lze efektivně řešit polymorfní elektronikou. Kritériem hodnocení je počet tranzistorů jednotlivých hradel nebo počet hradel ve složitějších obvodech. Klasické tranzistory (bipolární, unipolární) považujeme za ekvivalentní polymorfním. Dále existuje třída problémů, které nelze efektivně řešit polymorfní elektronikou (ale „neefektivně“ řešitelná je).

Klíčová slova. Ambipolární tranzistor, číslicová logika, organická elektronika, polymorfní elektronika, logické hradla, číslicové obvody.

1 Úvod

V současné době se v oblasti elektrotechniky stále více diskutuje o nových technologiích, jmenovitě nanotechnologiích, organických polovodičích, ambipolárních technologiích a s tím spojené polymorfní elektronice [1]. Ta slibuje řešit požadavek na stále větší hustotu funkcionality integrovaných obvodů v závislosti na ploše čipu, spotřebě, případně dalších kritériích. Polymorfní elektronika se tedy intenzivně zkoumá (např. [2]), nicméně v pozadí zůstává teoretický vývoj této oblasti. Pro polymorfní obvody totiž nelze použít běžné návrhové metody a logiku. Většina vědeckých skupin, zabývající se takovou elektronikou, proto používá nějaké formy generických algoritmů, různé druhy rozhodovacích stromů a podobně [3]. Chybí však teoretický základ polymorfní elektroniky, logické vazby a v návaznosti na to pak rozhodnutí, pro jakou třídu aplikací je taková elekronika vhodná.

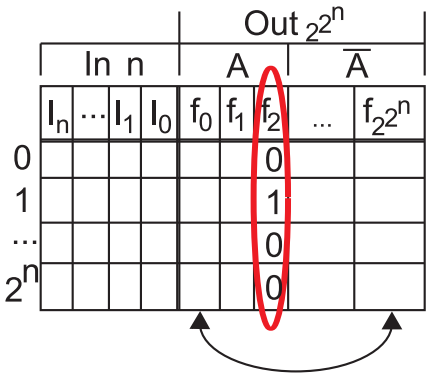
Začali jsme tedy zkoumat vlastnosti polymorfních obvodů a pokusili jsme se nastínit základní vlastnosti polymorfismu v logických obvodech.

2 Logická tabulka

Nejprve shrnu některé známé ale i ne příliš běžné skutečnosti, důležité pro pochopení dalšího textu.

Na obrázku 1 je strukura obecné tabulky logické funkce platná pro dvojkovou soustavu, kterou publikoval v roce 1904 například [5]. Tato tabulka nám ukazuje závislost počtu vstupů logického obvodu n na počtu kombinací uvedených vstupů 2^n a počtu výstupních funkcí 2^{2^n} tímto obvodem realizovatelných, kde $n \in \mathbb{Z}$. Každá logická funkce spadá buď do oblasti A, kde $A \in < 0, \frac{2^{2^n}}{2} - 1 >$ nebo $\overline{A}$, kde $\overline{A} \in < \frac{2^{2^n}}{2}, 2^{2^n} - 1 >$ a má negaci funkce v opačné oblasti.

Množina 2^{2^n} funkcí nám tedy určuje stavový prostor všech možných problémů, pro n vstupních proměnných. Na vektor výstupní funkce lze nahlížet jako číslo



Obrázek 1: Strukura logické tabulky

$$p(x) = \sum_{i=0}^{N} a_i x^i, N \in \mathbb{Z} \tag{1}$$

Členu $a_i x^i$ pak odpovídá řádek x^i, a proměnná a_i má hodnotu odpovídající výstupní proměnné v tomto řádku. Příklad: $0.2^3 + 0.2^2 + 1.2^1 + 0.2^0 = 0010_b = 2_d$. Označení fce tedy odpovídá její evaluaci a negace funkcí jsou pak symetricky uspořádány.

3 Speciální případy

Pro je $n = 0$ platí, že $2^0 = 1$, to znamená, že tabulka má pouze jeden řádek, ale žádnou vstupní proměnnou a existují právě $2^{2^0} = 2$ funkce, které jsou vzájemně inverzní. První funkce spadá do oblasti A (kontradikce), zatím co její negovaná funkce (tautologie) spadá do oblasti $\overline{A}$. Zmiňované kontradikce a tautologie jsou tedy nejobecnější funkce a nejsou závislé na žádné vstupní proměnné.

Pro jednu vstupní proměnnou $n = 1$ platí, že $2^1 = 2$, to znamená, že tabulka má dva řádky a existují právě $2^{2^1} = 4$ funkce, kde 2 spadají do oblasti A a dvě do oblasti $\overline{A}$. První funkce f_0 v oblasti A je výše zmíněná kontradikce, negace této funkce je tautologie (funkce f_3), spadající do oblasti $\overline{A}$. Další dvě vzájemně inverzní funkce jsou identita ID (f_1) a negace NOT (f_2).

Pro dvě vstupní proměnné $n = 2$ platí, že $2^2 = 4$, a existují právě $2^{2^2} = 16$ funkcí. Všech 16 funkcí je zobrazeno v tabulce 2. Některé tyto funkce poprvé použil v reléových logických obvodech Claude Elwood Shannon [4], na základě prací Augusta De Morgana [6] a George Boolea [7].

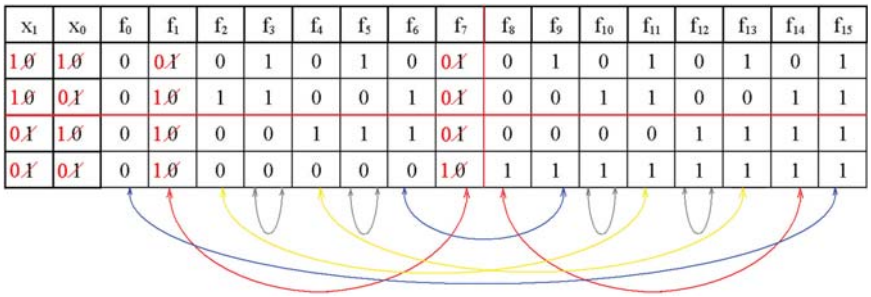
Platí tedy, že každá funkce z $A \in < 0, \frac{2^{2^n}}{2} - 1 >$ existuje negace této funkce v $\overline{A}$, které jsou definovány následovně:

$$\forall f_a, a \in < 0, \frac{2^{2^n}}{2} - 1 >, \exists f_b, b \in < \frac{2^{2^n}}{2}, 2^{2^n} - 1 > : f_a = \overline{f_b} \Rightarrow b = 2^{2^n} - a. \tag{2}$$

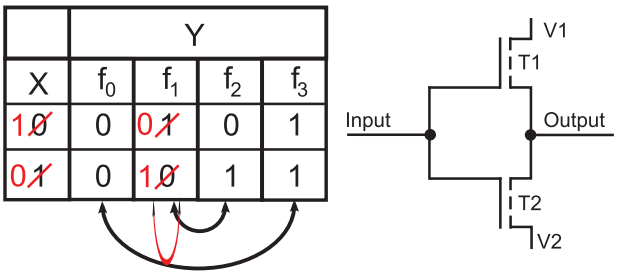
Poznámka: Inverzní funkci získáme velmi snadno zařazením invertoru na výstup logické funkce.

4 Polymorfní obvody

Polymorfní obvody jsou takové obvody, které jsou schopny záměrné a definované změny funkce za různých podmínek [3]. Příkladem může být změna logického hradla z funkce NAND na NOR, při změně teploty, nebo úrovně napájecího napětí (viz například [8]). Praktické využití polymorfismu v elektronice



Obrázek 2: Stavový prostor všech fcí pro n=2 a jejich doplňkové funkce.



Obrázek 3: Realizace doplňkové funkce (negace), vpravo hradlo, které ji realizuje

se objevuje především díky ambipolárním tranzistorům, které jej umožňují přímočaře použít. Neexistuje však zatím teorie polymorfní logiky, která by umožnila zkoumání vlastností polymorfních funkcí, případně tříd problémů, pro které je tato logika vhodná. Proto jsme se zaměřili především na tuto oblast.

Zkoumáním jsme zjistili, že existují přirozené polymorfní funkce (takzvané doplňkové funkce k původní funkci), které jsou tvořeny podobně jako například negace. Doplňkové hradla realizující tyto funkce jsou konstrukčně nejjednodušší (vycházejí z původní funkce) a nejmenší, co se týká počtu tranzistorů. Důvodem je, že vychází z konvenčních (minimalizovaných) hradel a všechny tranzistory jsou využity v obou funkcích. Dále je však možné vytvářet polymorfní hradla, které realizují i jiné kombinace funkcí.

Doplňkovou funkci dostaneme při zachování zapojení obvodu a přepnutí polymorfního obvodu do druhého stavu. Máme–li například polymorfní hradlo, které mění funkci změnou polarity napájení, pak toto hradlo změnou napájení realizuje doplňkovou funkci k funkci f_n, kterou budeme značit $c(f_n)$. Získání takové doplňkové funkce je přímočaré – viz tabulka na obrázku 3 a dostaneme ji negací všech vstupních i výstupních proměnných.

Doplňková funkce se může zobrazit sama na sebe – rezistentní funkce, viz obrázek 3. Druhou možností je zobrazení na negaci funkce. Totéž dostaneme použitím invertoru, což ale zvýší složitost obvodu. Příkladem je tautologie na obrázku 3, která se zobrazí na kontradikci. Poslední možnost je zobrazení na jinou funkci. Z hlediska polymorfních obvodů nás zajímá především poslední případ.

5 Vlastnosti doplňkových funkcí

Doplňkové funkce jsou obousměrně inverzní a zachovávají své vlastnosti. Na obrázku 2 tvoří žluté funkce (f_2/f_{11} a f_4/f_{13}), nebo červené funkce (NAND/NOR a AND/OR) symetrický systém. Modré jsou funkce doplňkové, které jsou současně negací funkce (tautologie/kontradikce, XOR/XNOR). Šedé funkce jsou funkce jedné proměnné a zobrazí se samy na sebe (jsou rezistentní). To je způsobeno tím, že funkce jedné proměnné může realizovat pouze identitu/negaci.

- Tvoří – li funkce úplný logický systém, tvoří doplňková fce také úplný systém. Příkladem je polymorfní hradlo NAND/NOR.
- Pokud $f_x = c(f_x)$, pak je funkce rezistentní. Funkce jsou sudé a asymetrické podle osy $\frac{2^n}{2}$. Jsou to funkce jedné proměnné (šedé fce na obrázku 2).
- Pokud $f_x = f_y, f_x \wedge \neq c(f_x) \wedge c(f_x) = c(f_y)$, pak je doplňková fce k f_x současně její negací. Funkce jsou sudé a symetrické podle osy $\frac{2^n}{2}$. Na obrázku 2 jsou modré. Proto je např XOR obtížně realizovatelné polymorfními obvody.
- Pokud $f_x \neq f_y \wedge c(f_x) \neq c(f_y)$, pak funkce f_x a f_y tvoří spolu se svými doplňkovými funkcemi symetrický systém. Symetrický systém značí, že funkce f_x a f_y jsou symetricky uspořádány podle osy $\frac{2^{2^n}}{2}$ a proto jsou vzájemně negované. Funkce $c(f_x)$ a $c(f_y)$ jsou také symetricky uspořádány podle osy $\frac{2^{2^n}}{2}$ a proto jsou opět vzájemně negované. Funkce jsou liché, proto jsou vždy asymetrické. Můžeme počítat

$$(2^{2^n} - 1) - f_x = c(f_y)(2^{2^n} - 1) - c(f_x) = f_y, n \in \mathbb{Z} \tag{3}$$

- Platí – li pro f_x že $|H| = |L|$ pak platí totéž i pro doplňkovou funkci (jedná se o „sudou funkci“). Platí – li pro f_x, že $|H| < |L|$, pak platí pro doplňkovou funkci $|H| > |L|, |H_{f_x}| = |L_{c(f_x)}|$. Funkce je asymetrická a „prohazuje“ počet H a L.
- Je – li fce závislá na určitých vstupních proměnných, je doplňková fce závislá na stejných vstupních proměnných.
- Z hlediska polymorfních obvodů vytvářejí zajímavé struktury především asymetrické funkce.

6 Obecný polymorfismus

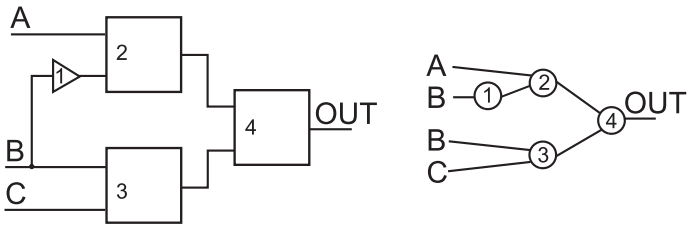
Doposud jsme se zabývali pouze polymofními obvody a hradly, které realizovaly polymorfně pouze dvě funkce (např. již zmiňované NAND/NOR, atd). Jedná se o speciální případ polymorfismu, který je nutno zobecnit, což jsme učinili následovně: Obecný polymorfní obvod můžeme zapsat jako

$$f(x) = \sum_{i=0}^{N} \sum_{j=0}^{M} p_i a_j x^j, N \in \mathbb{Z}, M \in \mathbb{Z} \tag{4}$$

Kde p_i značí polymorfní stupeň, který nabývá diskrétních hodnot $< 0, 1 >$. Jedná se tedy o obdélníkovou matici $n * m$ prvků, kde n je počet polymorfních funkcí o m polynomech (m řádků log. tabulky o $\sqrt{m}$ vstupních proměnných). Takto lze tedy vytvořit i vícefunkční polymorfní hradla.

Výhodou tohoto zápisu je absorbce všech informací – hodnoty vstupních proměnných v daném řádku logické tabulky, odpovídající hodnoty výstupních fcí a hodnoty „polymorfismu“. Všechny polymorfní členy v tomto zápisu musí být vždy nepravdivé, kromě jediného. Pravdivost polymorfního členu vyplývá z pravdivosti vstupní podmínky (např. polarita napájecího napětí) a určuje která funkce je za daných podmínek aktivní.

Předpokládejme, že takové hradlo bude reagovat na změnu velikosti napájecího napětí U_{min} až U_{max} změnou výstupní funkce tak, že všechny funkce v rozsahu $f = 2^{2^n}$ budou rozmístěny lineárně. Pro $n = 1$ bude mít takové polymorfní hradlo 4 funkce, pro $n = 2$ bude počet funkcí 16, atd. Vycházíme z předpokladu, že analogová změna napětí v nějakém rozsahu je spojitá veličina, proto je možno daný interval rozdělit na libovolný počet diskrétních částí a tak realizovat polymorfní hradlo s libovolným počtem výstupních funkcí.



Obrázek 4: Obecné schéma zapojení a jeho syntaktický strom (vpravo).

Pro takové hradlo je tedy stavový prostor N všech realizovatelných funkcí v rozsahu $f = 2^{2^n}$ dán změnou velikosti napájecího napětí v rozsahu U_{min} až U_{max}. Pokud tedy připojíme takové hradlo na regulovatelný zdroj napětí a budeme plynule měnit napájecí napětí takového hradla potenciometrem od U_{min} do U_{max} v čase od t_0 do t_{max}, budeme mít na výstupu hradla v jednotlivých časových úsecích t všechny funkce realizovatelné takovým hradlem pro daný počet vstupních proměnných. Mějme tedy nějaký problém, který lze řešit v tomto stavovém prostoru N. Pokud budeme postupně měnit napájecí napětí takového polymorfního hradla, dostaneme řešení problému o složitosti 2^{2^n} v lineárním čase a to nejhůře t_{max}. Další zmenšení časové složitosti je možné použitím invertoru zařazeného na výstup hradla a zpřístupněním obou stavů současně – tím bychom snížili složitost na $t\frac{max}{2}$.

7 Syntaxe a sémantika

Syntaxí rozumíme schéma zapojení, tedy jak jsou jednotlivé hradla nebo bloky propojeny. Syntaxi lze snadno zobrazit různými stromy (viz obrázek 4). Určuje tedy propojení bloků (motiv DPS), který nelze měnit. Na rozdíl od toho sémantika určuje funkci bloků nebo hradel – pro stejný syntaktický strom existuje více sémantických modelů. Cílem je v polymofní elektronice nalézt vhodné sémantické modely realizující potřebné funkce pro konkrétní syntaxi.

Problém tedy je, že máme dvě nebo více funkcí $(f_0, f_1, ..f_n)$, pro které potřebujeme vytvořit vhodné zapojení (syntaxi) tak, aby byly realizovatelné dostupnými polymorfními hradly (sémantika).

Na obrázku 4 je schéma zapojení obecného obvodu (vlevo) a jeho syntaktický strom (vpravo). Ve schématu jsou logické hradla kresleny záměrně jako obecné, protože změnou logické funkce každého hradla dosáhneme změnu výstupní funkce. Podívejme se na schéma z pohledu konvenční elektroniky. Můžeme psát

$$\Pi_{n=0}^{M} 2^{2^{n^{x_n}}}, M \in \mathbb{Z} \tag{5}$$

Kde x_n značí počet n – vstupých logických prvků (hradel). Pro schéma z obrázku 4) pak vychází pro obsažený celého stavového prostoru $2^{2^{0^0}}.2^{2^{1^1}}.2^{2^{2^3}} = 16384$ variací s opakováním. Pokud však uvažujeme polymorfní elektroniku, roste nám složitost následovně:

$$\Pi_{n=0}^{M}\Pi_{p=0}^{Q} 2^{2^{n^{x_n^p}}}, M \in \mathbb{Z}, Q \in \mathbb{Z} \tag{6}$$

Kde x_n značí počet n – vstupých logických prvků (hradel), které mají p-tý stupeň polymorfismu. Důležité však je, že obvody s $p = 1$ jsou konvenční logické obvody (bez polymorfní funkce). Pro schéma z obrázku 4) pak vychází pro obsažení celého stavového prostoru dokonce $2^{2^{0^0}}.2^{2^{1^{1^4}}}.2^{2^{2^{3^{16}}}} = 1.6069e^{60}$ variací s opakováním!

8 Závěr

Cílem práce bylo prokázat, že existuje ucelený set logických pravidel pro polymorfní elektroniku. V tom budeme pokračovat a zkoumat další pravidla, které bude možno využít při návrhu polymorfní elektroniky, stejně jako v současné číslicové elektronice například pomocí Booleovy algebry. Velmi důležitým cílem je také výzkum v oblasti syntézy a sémantiky. Dalším cílem je definovat, pro jakou třídu aplikací lze přirozeně využít polymorfní elektroniku, případně kde se již využití takové elektroniky nevyplatí. Ideálním výsledkem by pak byly návrhové pravidla, pomocí kterých bude možno jednoduše navrhovat a ověřovat polymorfní logické obvody.

Reference

[1] *Polymeric Polymorphic Electronics: Towards Multifunctional Logic Elements Based on Organic Semiconductor Materials*, Růžička, R., Šimek, V., Proceedings of CSE 2012 International Scientific Conference on Computer Science and Engineerig, Košice, SK, FEI TU v Košiciach, 2012, pages 154 – 161, ISBN 978-80-8143-049-7

[2] *Taking evolutionary circuit design from experimentation to implementation: some useful techniques and a silicon demonstration*, Stoica A., et al., Computers and Digital Techniques, IEE Proceedings – (Volume:151, Issue: 4), 2004, Pages: 295 – 300, DOI: 10.1049/ip-cdt:20040503.

[3] *Evolutionary Design of Gate-Level Polymorphic Digital Circuits* Lukáš Sekanina, Applications of Evolutionary Computing, Springer – Verlag Berlin Heidelberg, 2005, Pages: 185 – 194, DOI: 10.1007/978-3-540-32003-6-19.

[4] *A symbolic analysis of relay and switching circuits*, Shannon, Claude Elwood. Massachusetts Institute of Technology 1940, Thesis (M.S.) – Massachusetts Institute of Technology, Dept. of Electrical Engineering. Pages: 72.

[5] *Sets of Independent Postulates for the Algebra of Logic*, Huntington, E. V., Transactions of the American Mathematical Society, 5:3 (1904), Pages: 288 – 309.

[6] *Formal logic, or, The calculus of inference, necessary and probable*, De Morgan, Augustus. London: Taylor and Walton, 1847, pages: 384. Call number AJW-4210.

[7] *The mathematical analysis of logic: being an essay towards a calculus of deductive reasoning*, Boole, George, 1847, Cambridge : Macmillan, Barclay, & Macmillan, pages: 82, Call number ADN-0289.

[8] *REPOMO32 - New reconfigurable polymorphic integrated circuit for adaptive hardware*, Sekanina, L.; Ruzicka, R.; Vasicek, Z.; Prokop, R.; Fujcik, L., Evolvable and Adaptive Hardware, 2009. WEAH '09. IEEE Workshop on, vol., no., pages 39 – 46, April 30 2009 – March 2 2009, doi: 10.1109/WEAH.2009.4925666

[9] *Applications of Boolean Algebra: Claude Shannon and Circuit Design*, Janet Heine Barnett. Colorado State University – Pueblo, Loci, July 2013, DOI:10.4169/loci004000

Novel Error Detection and Correction Method Combining Time and Area Redundancy

Jan Bělohoubek

Informatics, 1th class, full-time study
Supervisors: Petr Fišer, Jan Schmidt
Faculty of Information Technology, Czech Technical University in Prague
Thákurova 9, 160 00 Praha 6, Czech Republic
jan.belohoubek@fit.cvut.cz

Abstract. In this paper, a novel fault-tolerant circuits design method is briefly described. It combines time and area redundancy to achieve error-correction abilities similar to triple-modular redundancy (TMR) and the area-overhead close to a duplex system. New logic gates design allowing complete stuck-at fault testability is presented.

Keywords: generalized C-element, area-overhead, time-overhead, duplex system, error-correction, stuck-at-fault, design for test, offline-test

1 Introduction

Methods for construction of dependable systems [1] are based on redundancy – *area redundancy* (hardware duplication), *time redundancy* (recomputation, software redundancy), or *information redundancy* (coding).

Short-duration transient faults can be well detected (and their effects corrected) using time redundancy, while *long-duration transient* faults and *permanent* faults using area redundancy. For *intermittent* faults it depends on their behaviour [1].

In this paper *stuck-at fault model* for *permanent faults* is used. Stuck-at faults associated with all gate inputs and outputs are considered. The term *complete fault-coverage* represents all gate-level stuck-at fault coverage.

In *duplex systems* [1], area redundancy is used for error detection (see Figure 1). Error-correcting representatives of the area-redundancy domain are *N-modular redundancy (NMR) systems* [1]. The simplest one is the *triple-modular redundancy (TMR) system* with the area overhead more than triple. Naturally, the correctness check is being performed *online*.

In the time-redundancy domain, a fault can be identified by running *offline tests*, for which the system operation must be temporarily interrupted.

An offline test of at least one module in a duplex system should allow to select the correct system output. However, to have a complete fault-coverage, offline tests are usually time consuming, moreover, a complete fault-coverage needs not be always achieved.

To avoid the mentioned offline testing problems, a new circuit construction method is proposed.

2 Time-Extended Duplex Scheme

Using the circuit duplication combined with universal short-duration offline tests, a system called *time-extended duplex scheme* can be constructed (see Figure 1). The offline test is performed while the computation is temporarily paused and then resumed without data violation. In the proposed method, the test takes no more than tens of computational cycles – it is denoted as *short-duration test*.

Permanent faults consequences in time-extended duplex scheme are detected using a universal short-duration offline test of module `M1*`. Transient faults consequences are eliminated by recomputation.

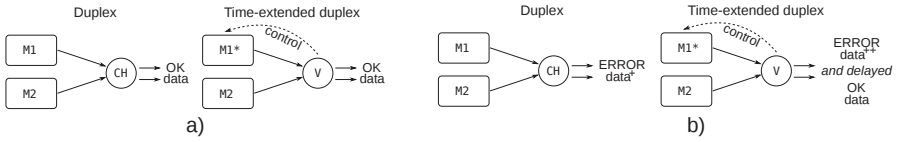


Fig. 1. Duplex and *time-extended duplex* comparison. a) The fault-free behaviour of both schemes is the same. b) In case of fault, duplex produces `ERROR` signal and incorrect data (`data`$^{+}$). On the other hand – time-extended duplex produces `ERROR` signal and *fault symptoms* (`data`$^{++}$). After a defined delay it produces `OK` signal and correct data.

3 Proposed Circuit Design Principle

For the *universal short-duration offline testing*, a simple test consisting of `all-zero` and `all-one` vectors is proposed. Also similarly simple (`all-zero` and `all-one`) output vectors will be observed at the outputs in a fault-free circuit. Detection of all stuck-at faults is required at the same time.

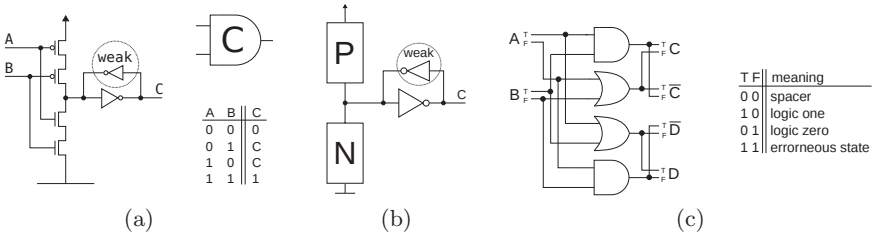


Fig. 2. a) semi-static C-element implementation, symbol and truth table, b) generalized C-element (the memory element is driven by N-MOS and P-MOS parts) and c) classical dual-rail logic `AND` and `OR` implementation. `C` $=$ `A` $\cdot$ `B`, `D` $=$ `A` $+$ `B`. `NOT` is implemented as wire-swap only: `T` $\leftrightarrow$ `F`.

As it was in detail presented in [3], if such simple test vectors are to be used for testing, it is not possible to use common logic gates such as `AND` and `OR`.

To propagate all types of faults the same way, the used gates should be symmetric. The second condition is *monotonicity* – the circuit should contain no inverters.

To achieve the monotonicity, principles from dual-rail logic [4] were adopted. In dual-rail logic [2], one logic value is encoded using two signals, see Figure 2 c) (`T` and `F`). If `T` is `0` and `F` is `1`, it represents the logic zero value, `10` represents logic one and `00` and `11` have a special meaning. Note that dual-rail logic with complementary signals allows to implement the `NOT` function as a wire-swap only.

The proposed gates are based on symmetric and state-holding *C-elements*. Probably the most commonly used implementation of a C-element is the *semi-static C-element* [4] (see Figure 2 a)).

If C-element is extended by *preset signals*, it can realise any monotonic gate function such as `AND`, `NAND`, `OR` and `NOR`. The adapted C-element matches the *generalized C-element* structure in Figure 2 b).

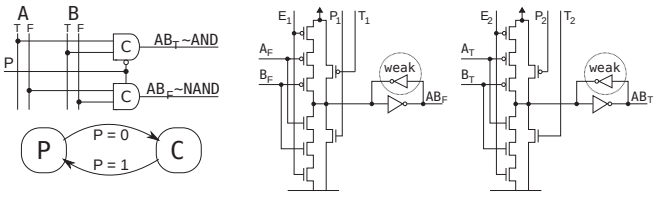


Fig. 3. Dual-rail `AND/NAND` gate implementation with preset. $\mathtt{AB_T}$ is the output of the `AND` gate and $\mathtt{AB_F}$ is the output of the `NAND` gate. Signals $\mathtt{E_1}$ and $\mathtt{E_2}$ are input enable signals and signals $\mathtt{T_1}$, $\mathtt{T_2}$ are only for test purposes. The illustrated implementation is a single-rail `AND` gate replacement. The single-rail `NAND` gate replacement differs only in a wire-swap: $\mathtt{AB_T} \leftrightarrow \mathtt{AB_F}$. The state transition diagram shows the gate-operating phases (P $\sim$ preset; C $\sim$ computation).

In Figure 3, the resulting dual-rail `AND/NAND` gate implementation is shown. The C-element based gate operates in two phases – *preset* and *computation*. In the preset phase the output value is preset using signals $\mathtt{P_1}$ and $\mathtt{P_2}$. In computational phase the value is preserved or changed depending on the states of the gate input signals.

Both phases are regularly switching during computation. Please, refer to [3] for the detailed description.

Using the proposed approaches, any combinational circuit can be constructed. The resulting circuit can be tested using short-duration offline test described also in [3]. The signals ($\mathtt{T_1}$, $\mathtt{T_2}$, $\mathtt{E_1}$ and $\mathtt{E_2}$) are used during the test.

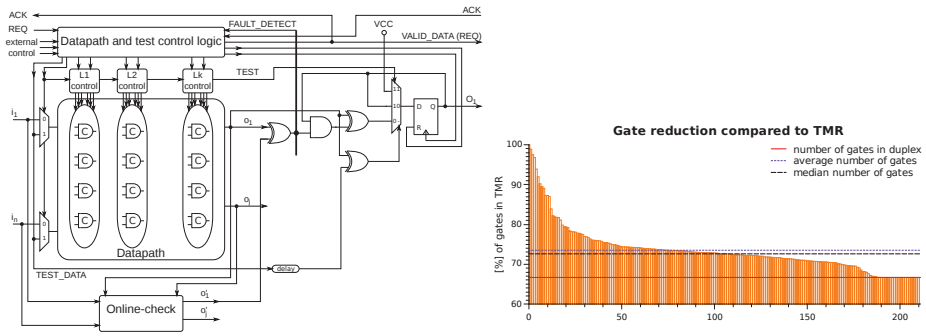
Since no fault masking is possible, the method is natively able to test multiple stuck-at faults and the reconvergences [5]

4 Time-extended duplex implementation

The detailed *time-extended duplex* system scheme is in Figure 4 a).

As the presented *time-extended duplex* exploits time-redundancy, computation time for every cycle is not predictable. This implies that a system based on the proposed method is *globally asynchronous*.

The *Online check* module in Figure 4 a) corresponds to the module `M2` in Figure 1, *Datapath* and *Li control* modules correspond to the module `M1*`. *Datapath and test control logic module*, together with the remaining logic implements the *Voter* in Figure 1. Modules `M2` and `M1*` can perform the same logic function.



(a) (b)

Fig. 4. a) *Time-extended duplex* implementation containing the *datapath* module constructed using the proposed gates. b) Number of C-element-based gates in time-extended duplex compared to the number of gates in the common implementation of TMR. Each vertical line represents one circuit. The circuits are sorted in descending order by the area-overhead.

4.1 Error-Correction Principle

The presented circuit works as follows: if no fault is detected by the online-checker, the result computed in the *Datapath* module is stored in the output register (the flip-flop in Figure 4 a)) and the `VALID_DATA` signal is asserted. Conversely, if a fault is detected (signal `FAULT_DETECT`), the universal short-duration offline test described in [3] is launched. This test generates *fault-symptoms*. These are accumulated in the output register. After the offline test is finished, the circuit recomputes the response using the same inputs as in the first phase. If a fault is detected again, it is corrected using fault-symptoms stored in the output register (only outputs where both tests – online and offline – report the fault presence are corrected). The corrected result is stored in the output register in place of the symptoms. If the fault is not detected again, new – uncorrected – results will be stored. This approach eliminates both transient-fault and permanent-fault effects. Finally, the `VALID_DATA` signal is set.

4.2 The Circuit Construction

The *Datapath module* is based on dual-rail logic, as indicated in Section 3. The dual-rail implementation can be derived from any single-rail implementation in a straightforward way – each single-rail gate is replaced by two complementary gates forming dual-rail one. Therefore, the number of single-rail gates is approximately doubled.

In the next step, denoted as a *reduction*, the area overhead is minimised. Each signal in the single-rail circuit is represented by two complementary signals when transformed into the dual-rail logic. A signal (and also its driving gate) needs to be duplicated only if it is present both in its direct and inverted form in the original single-rail implementation. Otherwise, the respective unused complementary gate can be removed. As shown in Figure 4 b), this reduction significantly decreases the area overhead. Note that the resultant datapath module is monotonic – no inverters are present in the module.

The inputs $\mathtt{i}_1 \ldots \mathtt{i}_n$ in Figure 4 a) correspond to inputs of the original single-rail circuit and their negations, while some inputs polarities can be removed by the reduction.

As described above, the proposed gates work in two phases. The gate can leave the preset phase after all gate inputs were stabilized. Additionally the test described in [3] requires level-based gate control. This implies that the control signals can be connected together only for the gates in the same depth. This is why separated control logic is required for every circuit level

The remaining logic implementation is not discussed since it can be synthesised using the standard approaches based on Figure 4 a).

4.3 Preliminary Results

Simulations to discover properties of the datapaths implemented using the proposed method were done using more than 200 combinational circuits from the *LGSynth'91 Benchmark* [6]. As the basis for experiments, *AIG*s [7] produced by the *ABC* [8] tool were used for all the circuits. From AIGs, two-input `NAND` gate based circuits were derived.

The resultant number of gates in the combinational part of the time-extended duplex system compared to the traditional implementation of TMR is presented in Figure 4 b). The time-extended duplex contains modules `M1*` and `M2`. The TMR is composed of three identical modules corresponding to `M2`.

Even though (for the most of benchmark circuits) the number of gates after the reduction is less than twice the number of gates in the original circuit, the real overhead is greater, because the proposed gates are larger. To achieve smaller gates, the semi-static C-element was replaced by the *dynamic C-element* [4].

The switching activity is increased thanks the preset phase of the proposed method. Based on the experiments, the dynamic power consumption strongly depends on the values of the circuit inputs. It can be at best comparable to the common logic implementation, if the share of ones and zeroes in the data input vectors are close to 1.

5 Conclusions and Future Work

The area overhead (the number of transistors) in time-extended duplex schema is comparable to TMR, however the number of gates is reduced and the 100% gate-level stuck-at fault coverage is achieved.

This can be also employed to find the exact fault-rates after the fabrication in the defined parts of the circuit. The testability of the remaining logic using the standard test approaches is in general not affected by presented method.

The main disadvantage of the proposed method is that the switching activity rises exponentially when the number of ones and the number of zeroes is not balanced in input vectors. This issue will be explored in the future too.

Limiting the design to two-input gates is not sufficient. The traditional two-input `NAND` gate is composed of 4 transistors, but the proposed two-input gate based on dynamic C-element is composed of 10 transistors. The size ratio for two-input gate is thus 2.5. The size ratio for the 3-input gate is 2 and for the 4-input gate 1.75. The greater fan-in should influence not only the resulting area overhead but also the circuit depth – the test length and the control logic complexity is in general related to the circuit depth.

The transistor-level stuck-at fault coverage and testability will be explored into detail because the proposed tests covers only the gate-level stuck-at faults.

The main control can be realised using the 16-state Moore machine but the detailed control logic elaboration is not done yet.

Acknowledgement

The author would like to thank to Ivo Háleček and Michael Hájek. This research has been in part supported by CTU grant SGS15/119/OHK3/1T/18.

References

1. I. Koren and C. M. Krishna, *Fault-Tolerant Systems*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2007.
2. A. Davis and S. M. Nowick, "An introduction to asynchronous circuit design," Tech. Rep., 1997.
3. J. Bělohoubek, "Novel gate design methodology for short-duration test," in *19th International Student Conference on Electrical Engineering – POSTER 2015*, May 2015.
4. J. Sparsø and S. Furber, *Principles of Asynchronous Circuit Design: A Systems Perspective*, 1st ed. Kluwer Academic Publishers, Boston, 2001.
5. L. Biwei, C. Shuming, and H. Xiao, "Analysis of glitch reconvergence in combinational logic ser estimation," in *Second Asia International Conference on Modeling Simulation, 2008. AICMS.*, May 2008, pp. 1015–1020.
6. S. Yang, "Logic synthesis and optimization benchmarks user guide: Version 3.0," Tech. Rep., 1991.
7. A. Biere, "AIGER," http://fmv.jku.at/aiger/, 2007.
8. A. Mishchenko *et al.*, "ABC: A system for sequential synthesis and verification," http://www.eecs.berkeley.edu/~alanmi/abc, 2012.

ENERGETICKÁ AUTONÓMNOSŤ IMPLANTOVATEĽNÝCH SENZORICKÝCH UZLOV

Martin Kováč, Viera Stopjaková
Mikroelektronika, 2. ročník, denná prezenčná forma štúdia
Školiteľ: Viera Stopjaková

Fakulta elektrotechniky a informatiky, Slovenská technická univerzita v Bratislave
Ilkovičova 3, 812 19 Bratislava

martin_kovac@stuba.sk

Abstrakt. Tento príspevok pojednáva o priestorovej a energetickej náročnosti komunikačného modulu (subsystému) v tzv. aktívnych biosenzorických/senzorických implantátoch (ABSI). Zároveň obsahuje konkrétne vyšpecifikované ciele a navrhované riešenia danej problematiky ako objekty výskumu dizertačnej práce. Príspevok obsahuje výsledky numerických simulácií uskutočnených v prostredí ANSYS HFSS, pomocou ktorých bola verifikovaná aplikovateľnosť úplne nového nami navrhnutého konceptu pre ABSI systémy. Koncept rieši otázku ich priestorovej a energetickej náročnosti na základe integrácie antény na čip spolu so zvyškom systému, využitia nízkopríkonových bezdrôtových techník a kombinácie puzdrenia čipu s prispôsobovacou dielektrickou vrstvou (PDV), čím sa zvýši zisk antény. Prvotné dosiahnuté zlepšenie radiačnej účinnosti v niektorých prípadoch bolo viac ako 20 dB (t.j. 100-krát).

Kľúčové slová. anténa na čipe, aktívne biosenzorické/senzorické implantáty, WBAN, UWB technológia

1 Úvod

V poslednej dobe sa tzv. systémy asistovaného života (angl. ambient-assisted living - AAL) stávajú už neoddeliteľnou súčasťou života mnohých ľudí a významným spôsobom skvalitňujú ich život. Jednou z najdôležitejších cieľov AAL programu je úspešné zavedenie konceptu elektronickej domácej starostlivosti, ktorá je založená na bezdrôtovom prenose dát medzi telovou sieťou (angl. WBAN) a pacientovým personál-nym serverom. Personálny server následne zabezpečuje prenos dôverných osobných údajom do personálneho serveru lekára.

Jeden z najkritickejších bodov v tejto sieti sa javí vytvorenie komfortnej a bezpečnej monitorovacej bezdrôtovej siete pozostávajúcej zo senzorických/biosenzorických uzlov prípadne HUBov umiestnených na/v ľudskom tele. Špeciálna starostlivosť musí byť venovaná tzv. aktívnym implantovateľným medicínskym zariadeniam (AIMZ), ktorých chápanie je veľmi všeobecné. Preto zavedieme termín ABSI a budeme ho chápať ako druh AIMZ s permanentným rezervoárom elektrickej energie (napr. batéria), ktorý zabezpečuje jeho monitorovaciu/stimulačnú autonómnosť a možnosť inicializovať sofistikovanejší typ bezdrôtového prenosu. Širšie využitie ABSI systémov je značne brzdené vysokou spotrebou komunikačného modulu, čo platí hlavne v prípade komunikácie využívajúcej rádiové vlny (RF). Mnohí vedci sa možno i preto domnievajú, že budúcnosť patrí ultrazvukovej komunikácií a komunikácií postavenej na vodivých vlastnostiach tkaniva. Prioritu im pripisujú aj vďaka nižším stratám v živom tkanive

(vzhľadom na pracovné frekvenčné pásmo), ktoré majú v porovnaní s RF komunikáciou. Do návrhu však prinášajú nové komplikácie ako konštrukcia elektród, začlenenie piezoelektrického meniča do návrhu a pod. Veľkým benefitom RF a ultrazvukovej komunikácie je, že dokážu preniesť informáciu i mimo živého organizmu. Tento fakt sa stal hlavnou motiváciou nášho výskumu, pretože externý kontroler/čítač by v takomto prípade mohol byť súčasťou už aj tzv. smart zariadení ako sú napríklad smart telefóny alebo hodinky priamo s podporou GSM, WIFI, Bluetooth atď.

2 Východiskový stav

V počiatočnej fáze výskumu sme sa zamerali na redukciu spotreby energie ABSI systémov, konkrétne na bezdrôtový komunikačný modul, ktorý patrí v súvislosti so spotrebou medzi tie najproblematickejšie (spotreba zvyčajne vyššia ako $1\ mW$). Nami navrhnuté riešenie pozostáva z tzv. hybridného vysielača/prijímača, pričom vysielač je založený na impulznej širokopásmovej komunikácií (IR-UWB) a prijímač na úzkopásmovej komunikácii v konfigurácii Wake-up (Tab. 1). Keďže FCC (angl. Federal Communications Commission) vyhradila pre UWB komunikáciu frekvenčný rozsah $3,1 - 10,6\ GHz$, pokladáme za rozumné pre úzkopásmovú komunikáciu využiť najbližšie nižšie ISM pásmo t.j. $2,45\ GHz$. Priamo z pracovných frekvencií je zrejmé, že v najlepšom prípade sa bude maximálna hĺbka umiestnenia implantátu pohybovať na úrovni jednotiek cm (stredná hĺbka implantátu, diskrétne riešenia antény) z dôvodu vysokého útlmu v mäkkom tkanive. Avšak použitie vyšších frekvencií vedie k možnosti integrácie antény priamo na čip (prispeje k miniaturizácii ABSI), čo predstavuje jeden z cieľov práce. Antény realizované na čipe (špeciálne ak sa jedná o štandardný CMOS proces) sú ale charakteristické veľmi malou radiačnou účinnosťou a teda aj ziskom, čo by ich diskvalifikovalo pre použitie v ABSI. Z toho dôvodu sme museli vymyslieť spôsob ako čo najmarkantnejšie zvýšiť radiačnú účinnosť antén realizovaných na čipe a zabezpečiť tak ich aplikovanosť i pre takýto typ systémov.

Všeobecne stanovené ciele:	**Riešenia:**
1. redukcia spotreby plochy	použitie hybridnej konfigurácie vysielač/prijímač
2. miniaturizácia ABSI	integracia antény na čip (štandardný CMOS proces)
3. zachovanie aplikovateľnosti RF komunikácie pre ABSI 4. splnenie limitov týkajúcich sa vplyvu EM poľa na ľudské zdravie	*zvýšenie radiačnej účinnosti integrovanej antény prostredníctvom PDV*

Tab. 1: Všeobecne stanovené ciele a riešenia dizertačnej práce.

2.1 Jadro (základ) navrhnutého konceptu ABSI

Analýza šírenia elektromagnetickej vlny vo vodivom prostredí tvoreného ľudským tkanivom a návrh diskrétnych antén pre ABSI systémy je súčasťou množstva výskumných prác [2–4]. Iba práca [2] sa však zaoberá hlbšou analýzou vplyvu izolátora (v našom ponímaní sa jedná o PDV) na vlastnosti šíriacej sa elektromagnetickej vlny a vlastnosti poľa. Analýzu uskutočnil na základe analytického odvodenia pre polypropylén, PEEK, polyamid, oxid hlinitý a oxid zirkoničitý.

Okrem tejto práce si zaslúži špeciálnu pozornosť aj práca Dissanayake-ho, ktorý analyzoval vplyv izolátora (PDV) na vlastnosti UWB antény [5, 6]. K zaujímavým zisteniam prišiel vyšetrením koeficientu odrazu Γ na hranici medzi izolátorom a ľudským tkanivom v prípade elektromagnetického poľa budeného Hertzian-ovým elektrickým dipólom. Pri návrhu diskrétnej UWB antény požil ako izolátor glycerol, ktorý má relatívnu permitivitu blízku relatívnej permitivite mäkkého tkaniva pri frekvenciách vyšších ako $3,1\ GHz$.

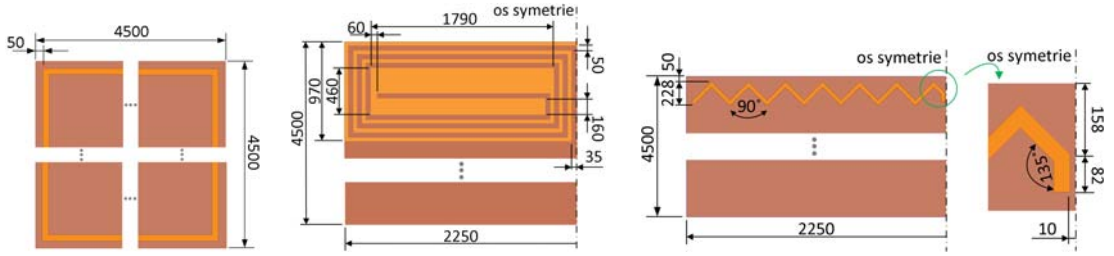
S uvedených poznatkov vyplýva viacero užitočných zistení, na základe ktorých je vhodné v počiatočnej fáze koncept postaviť a následne modifikovať v závislosti od špecifických podmienok vzťahujúcich sa na realizáciu antény na čipe:

- Zahrnutie izolátora do návrhu vedie k zníženiu vložných strát,

- Relatívna permitivita oboch rozhraní by mala byť rovnaká alebo aspoň približne rovnaká,
- Hrúbka izolátora by mala byt čo najväčšia,
- Ako zdroj budenia elektromagnetického žiarenia je lepšie použiť magnetický než elektrický zdroj,
- Variabilita koeficientu odrazu Γ v závislosti od zmeny vodivosti tkaniva (uvažovaná vodivosť izolátora je rovná nule) je zanedbateľná, dominantná je variabilita v závislosti od neidentity jednotlivých permitivít tvoriacich rozhranie izolátor (PDV)–tkanivo.

3 Antény integrované na čipe

Návrh antény na čipe realizovaný v štandardnej CMOS technológii sa v mnohých aspektoch líši od "klasického" návrhu antény v dôsledku existencie mnohých parazitných javov (nízky merný odpor samotného substrátu, malá vzdialenosť medzi najvyššou metalizačnou úrovňou a substrátom, malá efektívna dĺžka antén realizovaných na čipe v porovnaní s ich pracovnou frekvenciou, atď.). Tieto javy negatívnym spôsobom ovplyvňujú výslednú radiačnú účinnosť antén. Efektívnejší návrh antény v štandardnom CMOS procese je až pre vyššie frekvencie t. j. desiatky GHz. Radiačná účinnosť i pri takto vysokých frekvenciách však závisí od konfigurácie systému a môže byť zvýšená pomocou optimalizačných techník akými sú začlenenie tieniacej vrstvy medzi anténu a vonkajší povrch substrátu, či použitie zložitejších štruktúr antén, použitie superstrátovej konfigurácie alebo tvarovacej šošovky a pod [1]. Existuje teda reálny predpoklad, že získané výsledky by bolo možné ďalej vylepšiť práve použitím uvedených techník, nakoľko verifikácia konceptu bola zatiaľ uskutočnená len na základných neoptimalizovaných anténach (Obr. 1). Naopak prítomnosť PDV s vysokou permitivitou, ktorá je základom navrhnutého konceptu, môže úplne zmeniť charakter vyššie spomínaných javov.



(a) Slučková anténa. (b) Slot anténa. (c) Dipólová anténa.

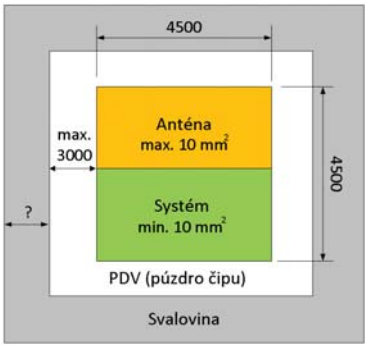
Obr. 1: Štruktúra a rozmery navrhnutých antén (rozmery sú v μm).

Na overenie vhodnosti ako aj vlastností navrhnutého konceptu sme použili tri testovacie vzorky–tri typy antén, konkrétne slučkovú anténu (zastupuje budenie s využitím magnetického zdroja elektromagnetického žiarenia, Obr. 1a), ďalej tzv. slot anténu (zastupuje kvázi Huygensov zdroj, Obr. 1b) a elektrickú dipólovú anténu (zastupuje budenie s využitím elektrického zdroja elektromagnetického žiarenia, Obr. 1c). Rozmery jednotlivých antén neboli žiadnym spôsobom optimalizované, pretože cieľom tohto príspevku je len pojednať o možných výhodách, ktoré by mohol navrhnutý koncept priniesť z pohľadu ABSI. Doplňme ešte, že simulácie antény typu slot v prirodzenom prostredí vyplnenom vzduchom (simulácia 1) a za prítomnosti ľudského tkaniva (simulácia 2) nebola vykonaná (pozri Tab. 2). V prvom prípade išlo o časovú náročnosť a v druhom o nedostatočnú výpočtovú mohutnosť použitých hardvérových prostriedkov.

Zámerom tohto príspevku je verifikácia možnosti integrácie antény na čip spolu so zvyškom systému (vytvorenie tzv. systém na čipe) pre ABSI na základe navrhnutého konceptu realizovaného v štandardnej TSMC $90\ nm$ CMOS technológii. Principiálna schéma (prislúchajúca simulácii 2 – pozri Tab. 2) nového konceptu je zobrazená na Obr. 2[1].

[1] Platí pre dipólovú a slot anténu. Slučková anténa je v tomto prípade realizovaná po obvode čipu.

Centrálna plocha je zabraná samotným čipom, ktorý je fyzicky rozdelený na dve hlavné časti z dôvodu potlačenia vzájomnej elektromagnetickej interferencie medzi systémom (aktívna časť) a anténou (pasívna časť). Holý čip je následne obalený PDV vrstvou, ktorá zároveň tvorí puzdro čipu. Poznamenajme však, že fyzické rozdelenie čipu na aktívnu a pasívnu časť nám umožňuje vytvoriť v puzdre dutinu v aktívnej oblasti a tým zjednodušiť kontaktovanie čipu. Vzhľadom na to, že existuje predpoklad priameho kontaktu PDV s hostiteľským prostredím je dôležité aby PDV bezpodmienečne spĺňala požiadavky na biokompatibilitu a biostabilitu. Takto zapuzdrený čip sa následne stáva súčasťou živého tkaniva (svaloviny), hĺbka implantácie je cieľom budúceho výskumu.



Obr. 2: Principiálna schéma konceptu ABSI

4 Dosiahnuté výsledky

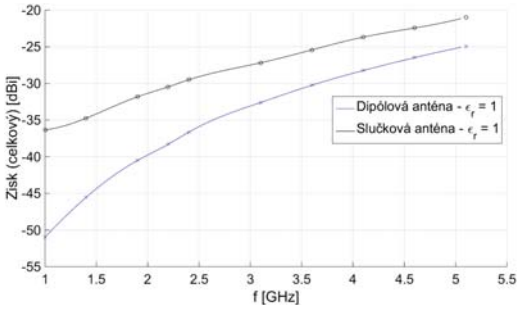
Na verifikáciu aplikovateľnosti nami navrhnutého konceptu sme využili komerčne dostupný simulátor ANSYS HFSS^{TM}, ktorý je založený na metóde konečných prvkov, pričom na riešenie Maxwell–ových rovníc využíva frekvenčnú oblasť. Uvedené vlastnosti ho robia vhodným kandidátom (z pohľadu presnosti výpočtu) pre riešenie takéhoto typu problému, akou je analýza antény realizovanej na čipe v oblasti desiatok GHz (elektricky malé štruktúry). Vzhľadom na zvolenú technológiu výroby (štandardný TSMC $90\ nm$ CMOS proces), v ktorej bude systém na čipe navrhnutý, jediným a vopred daným parametrom je radenie a vlastnosti jednotlivých vrstiev danej technológie s výnimkou hrúbky substrátu. Okrem variability hrúbky substrátu, pohybujúcej sa v rozmedzí $250 - 500\ \mu m$ (použité $250\ \mu m$), vlastnosti antény môže byť modifikované ďalšími parametrami ako šírka metalizácie (použitá $35\ \mu m$), počet metalizačných úrovní (použitých 10 úrovní), štruktúra a typ antény (Obr. 1) atď., ktoré sú z veľkej časti definované návrhovými pravidlami výrobnou technológiou čipu. Z tohto dôvodu je nutné návrh dotiahnuť až na topografickú úroveň (layout) zabezpečujúcu vyrobiteľnosť a zachovanie vlastnosti navrhnutej antény. Poznamenajme, že detailnejší popis radenia a vlastnosti jednotlivých vrstiev je možné nájsť v [7].

Parametre ako napríklad prítomnosť PVD a fantómu tkaniva, vlastnosti a typ použitej antény, závisí od typu realizovaného scenára (simulácie). Konkrétne parametre použité pri jednotlivých scenároch a ich ciele sú prehľadne zhrnuté v Tab. 2 (ε_r - relatívna permitivita, tg δ - stratový činiteľ, σ - vodivosť, h - hrúbka). Prvý je zameraný na vyšetrenie zisku antén v bežnom prostredí vyplneného vzduchom. Druhý scenár modeluje situáciu antén v ich neprirodzenom in–vivo prostredí ako súčasť ABSI systémov. Posledný scenár bol navrhnutý na porovnanie radiačnej účinnosti antén klasického konceptu (prostredie vyplnené vzduchom, simulácia 1) a navrhnutého konceptu (prostredie vyplnené PDV, simulácia 3).

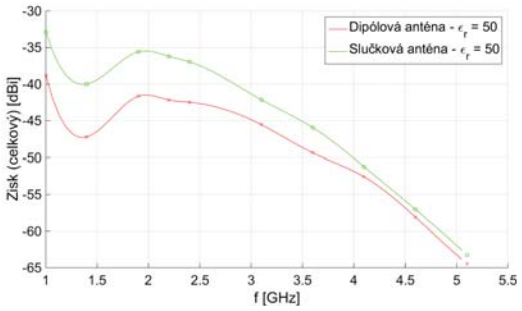
Simulácia	**Typ antény**	**Prispôsobovacia vrstva (PDV)**	**Vlastnosti PDV**	**Fantóm**
1	dipólová, slučková	nie	-	nie
2	dipólová, slučková	áno	$tg\ \delta = 0,002\ \varepsilon_r = 50\ h = 2\ mm$	áno
3	dipólová, slučková, slot	nie	-	nie
Simulácia	**Vlastnosti fantómu**	**Simulačné prostredie**	**Vlastnosti simulačného prostredia**	**Cieľ simulácie**
1	-	vzduch	$tg\ \delta = 0\ \varepsilon_r = 1,001\ h = 4\ cm$	Zistenie radiačnej účinnosti v prostredí vzduch
2	$Re\{\varepsilon_r\} = 54,8$ $\sigma = 0,98\ S/m\ (1\ GHz)$ $Re\{\varepsilon_r\} = 49,5$ $\sigma = 4,04\ S/m$ $(5,1\ GHz)\ h = 5\ cm$	vzduch	$tg\ \delta = 0\ \varepsilon_r = 1,001\ h = 4\ cm$	Zistenie zisku antény umiestnenej v tkanive
3	-	PDV	$tg\ \delta = 0\ \varepsilon_r = 50\ h = 1,1\ cm$	Overenie zlepšenia radiačnej účinnosti

Tab. 2: Parametre a vlastnosti prvkov použitých v jednotlivých simuláciách.

Získané výstupy zisku antén a radiačných účinností sú uvedené na Obr. 3 a Obr. 4. Obr. 3a zobrazuje typický priebeh zisku elektricky malých antén realizovaných na čipe s nízko–rezistívnym substrátom v naturálnom prostredí (simulácia 1), kedy radiačná účinnosť (v tomto prípade definovaná prostredníctvom zisku antény) rastie so zmenšujúcou sa vlnovou dĺžkou. Keď antény vložíme do vodivého prostredia – svaloviny (simulácia 2), nárast útlmu elektromagnetickej vlny pre vysoké frekvencie je väčší než nárast zisku elektricky malých antén (Obr. 3b). Významným zistením z pohľadu konceptu je fakt, že zisk antén, ktoré sú implantované do svaloviny $5\ cm$ hlboko, je takmer rovnako veľký zisk ako antény v naturálnom prostredí. Za zmienku stojí tiež potvrdenie lepšej radiačnej účinnosti slučkovej antény (budenie s využitím magnetického zdroja elektromagnetického žiarenia) ako to bolo predpokladané v predchádzajúcom texte.



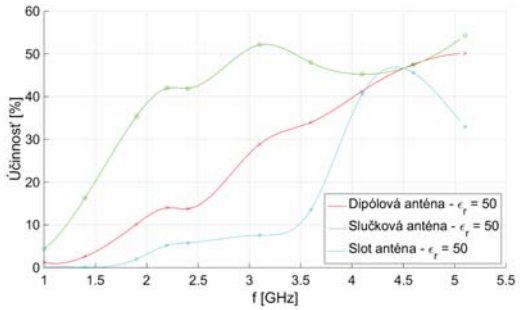
(a) Vzduch (simulácia 1).



(b) Ľudské tkanivo - svalovina (simulácia 2).

Obr. 3: Celkový zisk (zosilnenie) dipólovej a slučkovej antény umiestnenej v jednotlivých prostrediach.

Najväčšiu výpovednú hodnotu v súvislosti s možnými očakávanými výhodami analyzovaného konceptu prezentuje graf uvedený na Obr. 4, ktorý zobrazuje radiačnú účinnosť prislúchajúcu simulácii 3. Tá bola navrhnutá tak, aby sme dokázali pozorovať len vplyv začlenenia PDV na zmenu vlastností integrovanej antény bez vplyvu vodivého tkaniva. Poznamenajme, že stratový činiteľ homogénneho prostredia bol v tomto prípade rovný nule, čo inak povedané predstavuje ideálny prípad. Obr. 4 preukazuje markantné zlepšenie radiačnej účinnosti už pri pomerne nízkych hodnotách frekvencie. To je výhodné práve z pohľadu ABSI, kde sa útlm vo vodivom tkanive zvyšuje s rastúcou hodnotou frekvencie. Pre porovnanie možno uviesť, že maximálna účinnosť v prípade simulácie 1 bola dosiahnutá pri najvyššej frekvencii t.j. $5\ GHz$ a mala hodnotu len $0,17\ \%$ (dipólová anténa), resp. $0,51\ \%$ (slučková anténa).



Obr. 4: Radiačná účinnosť antén umiestnených v prostredí tvorenom PDV (simulácia 3).

Okrem vyššie analyzovaných parametrov, istú výpovednú hodnotu má pre nás aj vyšetrenie vstupnej impedancie jednotlivých antén, čo vedie k dvom významným zisteniam. Prvé zistenie z pohľadu konceptu, ktoré sa dalo logicky predpokladať je, že zmena prostredia s nízkou relatívnou permitivitou na prostredie s vysokou relatívnou permitivitou sa odzrkadlí aj na zmene vstupnej impedancie. Druhým,

pre nás dôležitejším poznatkom je skutočnosť, že reálne a imaginárne zložky vstupných impedancií v prípade simulácie 2 a simulácie 3 sú takmer identické, čo okrem vzájomného prispôsobenia rozhrania PDV–tkanivo implikuje tiež "uväznenie" veľkej časti blízkeho poľa (jeho reaktívnej časti) v PDV vrstve a nie vo vodivom tkanive. Logickým dôsledkom tohto javu je teda vyššia radiačná účinnosť antén. Sekundárnym dôsledkom je teoretická možnosť jednoduchšieho modelovania komunikačného kanála, pretože elektromagnatická vlna je vo vodivom prostredí charakterizovaná prevažne už len radiačnou zložkou blízkeho poľa a vlastnosťami ďalekého poľa. Toto zistenie sa môže v neskoršom výskume uplatniť pri analýze a návrhu fyzickej vrstvy RF komunikácie.

5 Ciele dizertačnej práce

Na základe vykonanej analýzy, zistených poznatkov a potrieb, ako aj doteraz dosiahnutých výsledkov prezentovaných v tomto príspevku, boli hlavné ciele dizertačnej práce sformulované nasledovne:

- Navrhnúť nový koncept komunikačného modulu za účelom zníženia spotreby energie a redukcie rozmerov ABSI systémov.
- Preskúmať možnosti integrácie UWB antény na čip a definovať jej optimálnu štruktúru z pohľadu požiadaviek ABSI systémov.
- Vyšetriť a vyhodnotiť vplyv vlastnosti prispôsobovacej (dielektrickej) vrstvy na parametre navrhnutej UWB antény.
- Zrealizovať fyzický návrh antény na čipe (navrhnúť topografiu antény) vo vybranej CMOS technológii a vyhodnotiť dosiahnuté parametre.
- Navrhnúť a verifikovať model komunikačného bezdrôtového kanála a vyhodnotiť jeho prínos pre ďalší výskum a realizáciu komunikačného modulu v rámci ABSI.

6 Záver

Aplikovateľnosť navrhnutého konceptu bola verifikovaná prostredníctvom numerických simulácií, pričom sme zaznamenali v niektorých prípadoch zlepšenie radiačnej účinnosti antény pri daných podmienkach až o 20 dB. Výsledky môžu byť ešte vylepšené optimalizáciou konceptu (štruktúra antény, vlastnosti PDV, vyšetrenie vplyvu metalizácie ostatných subsystémov na parametre antény atď), ktorá bude cieľom budúceho výskumu, kde sa primárne zameriame na ďalšie zvýšenie radiačnej účinnosti antény a jej širokopásmovosť. Najväčšiu výzvu vidíme v realizácií puzdrenia, ktoré okrem biokompatibility a biostability musí disponovať vhodnými elektromagnetickými vlastnosťami charakteristickými pre PDV.

Poďakovanie

Článok je jedným z výstupov výskumnej práce projektu s názvom "Centrum výskumu závažných ochorení a ich komplikácií", ITMS projektu: 26240120038. "Projekt je spolufinancovaný zo zdrojov EÚ. Podporujeme výskumné aktivity na Slovensku".

Literatúra

[1] H.M. Cheema and A. Shamim. The last barrier: on-chip antennas. Microwave Magazine, IEEE, 14(1):79-91, Jan 2013.
[2] F. Merli. Implantable Antennas for Biomedical Applications. ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE, 2011.
[3] A. A. Y. Ibraheem. Implanted antennas and intra-body propagation channel for wireless body area network, 2014.
[4] A. Johansson. Wireless communication with medical implants: antennas and propagation. Lund University, 2004.
[5] T. Dissanayake, K.P. Esselle, and M.R. Yuce. Dielectric loaded impedance matching for wideband implanted antennas. Microwave Theory and Techniques, IEEE Transactions on, 57(10):2480-2487, Oct 2009.
[6] T. Dissanayake, K.P. Esselle, and M. Yuce. UWB antenna impedance matching in biomedical implants. In Antennas and Propagation, 2009. EuCAP 2009. 3rd European Con-ference on, pages 3523-3526, March 2009.
[7] Martin Kováč, Viera Stopjaková, and Daniel Arbet. UWB Communication for Implantable Biosensors within WBAN Systems. YBERC 2014 , pages 6-11, 2014.

Operační systém na dynamicky rekonfigurovaných procesorech

Ing. Petr Cvek
Technická kybernetika, 4. ročník, prezenční forma studia
Školitel: Prof. Ing. Ondřej Novák, CSc.

Ústav ITE, FM
Technická univerzita v Liberci, Studentská 1402/2, 461 17 Liberec 1

petr.cvek@tul.cz

Abstrakt. Příspěvek přibližuje prostředky a návrh systému s rekonfigurovatelným hardware. Hlavním cílem je návrh plánovače, který je schopen modifikovat hardware s periodou blížící se periodě přepínání úloh ve víceúlohovém systému. A vyvinutí toho plánovače jakou součást klasického plánovače úloh. Účelem tohoto typu rekonfigurace je dosažení lepšího využití prostředků FPGA se zároveň co nejnižšími náklady na vývoj aplikací. Aplikace si ale může dodat vlastní výpočetní jednotku, kterou podle dodaných parametrů plánovač zařadí do použitelných prostředků.

Klíčová slova. FPGA, rekonfigurace, plánovač úloh, Microblaze, SMP, Linux

1 Úvod

Práce je zaměřena na problematiku rekonfigurace FPGA (Field Programmable Gate Array) pro univerzální víceúlohový operační systém . Obvody FPGA jsou již od 80. let používány pro snadnou implementaci hardware pro testování návrhu. Díky rostoucí složitosti integrace je možné v posledních zhruba deseti letech implementovat do FPGA celý SoC (System On a Chip) s dostatečně vybaveným procesorem, který je schopen provozovat víceúlohový operační systém s ochranou paměti. Zároveň je u moderních FPGA možné z testovacích, zabezpečovacích [1] nebo ladících důvodů přistupovat k již naprogramované konfigurační paměti. V případě, že se jedná o zápis, je tento přístup nazýván rekonfigurace a v případě čtení potom readback. Obě operace lze ještě dále dělit na částečnou, kde se operace provádí jen na podmnožině konfigurace a/nebo na dynamickou, kde dochází k operaci bez zásahu do obvyklé činnosti zbylého návrhu. FPGA architektury se také liší podle typu prostředků, na které lze readback a/nebo rekonfiguraci aplikovat [2].

Práce bude popisovat rekonfiguraci na jedné z nejvyspělejších architektur, které v době psaní existovaly a to Xilinx FPGA řady 7 (Kintex XC7K325T-2FFG900C). Tato architektura umí rekonfiguraci po sloupcích CLB (Configurable Logic Block), což je základní jednotka sdružující logiku pro tvorbu kombinačních i sekvenčních obvodů. Z dalších vlastností je readback, při kterém je uložen aktuální stav proměnlivých prvků CLB (Flip-Flop, LUTRAM, ...) a který může být dál zpracován v systému (například nahrán na jinou lokaci na čipu).

V případě implementovaného SoC je možné použít rekonfiguraci pro odstranění nevýhod klasického návrhu. Pomocí rekonfigurace je možné měnit účel a prostředky SoC podle aktuálních požadavků operačního systému. V současné době existují různé, mnohdy jednoúčelové návrhy, které se snaží tyto požadavky generovat a řešit. Jednotlivé relevantní projekty byly popsány ve zdrojích [3–5].

1.1 Motivace

Motivací pro výzkum rekonfigurovatelných systémů je rostoucí potřeba stále více inteligentnějších systémů, možnost změnit funkci nepoužívaných oblastí čipu na používané prostředky a v neposlední řadě taky možnost opravy poruch, které jsou se snižujícím se procesem výroby stále více dominantní. Výzkum rekonfigurace v klasickém operačním systému je navíc výhodný ve snížení nároků na vývoj aplikace.

2 Prostředky rekonfigurovatelného hardware

Víceprocesorový SoC obvykle vyžaduje úpravu existujících prvků systému a přidání nových funkcí. Následující sekce popíšou základní z nich.

2.1 Meziprocesorová komunikace

Základním požadavkem při implementaci SoC, který má podporovat více procesorů, je schopnost jednotlivých procesorů mezi sebou navzájem komunikovat. To je nutné například v případě SMP, kdy plánovač úloh potřebuje přesunout úlohu z jednoho procesoru na druhý.

Komunikace mezi procesory je obvykle implementována pomocí speciálního přerušení IPI (Interprocessor interrupt), které může kontaktovat jiný procesor (například LAPIC [6] na architektuře x86).

2.2 Routování přerušení od periferií

Pokud má být konkrétní procesor v SoC schopen provádět kód ovladačů v jádru operačního systému, je nutné, aby byl schopen přijmout požadavek na přerušení. Obvykle stačí vhodně upravit jednoprocesorový řadič přerušení z podpory M:1 na M:N. S výjimkou systémů s redundancí výpočetních jednotek je obvykle potřebné, aby mohl být signál přerušení dopraven k libovolnému procesoru. Signál přerušení k ostatním procesorům je obvykle maskován.

2.3 Časovače

Pro víceúlohový preemptivní operační systém je vhodné použít pro každý procesor samostatný zdroj přerušení od programovatelného časovače. Jako zdroj přerušení spadá SMP časovač spíše do třídy meziprocesorové komunikace, neboť přísluší právě jednomu procesoru. Z hlediska operačního systému je možné použít pouze jeden globální časovač, to ale vnáší značná omezení na plánování úloh, protože pak dostanou všechny zároveň běžící procesy stejně velké kvantum času. Stejná situace lze v případě potřeby vytvořit vhodným naprogramováním všech SMP časovačů, takže je řešení časovač pro každý procesor vhodnější.

2.4 Ochrana před datovými hazardy nad sdílenou pamětí

V případě, že dojde k přerušení běhu programu, který zrovna pracuje se sdílenou pamětí, je nutné, aby byla zachována konzistence dat. Tento problém existuje i na jednoprocesorových víceúlohových systémech, ale na víceprocesorových je nutno tento hazard řešit i mezi jednotlivými procesory. Pro tyto účely je implementován hardware (arbitr exkluzivního přístupu), který kontroluje veškeré přístupy do operační paměti a rozhoduje zda bude požadavek operace proveden nebo ignorován. Z hlediska instrukční sady je pro exkluzivní přístup do paměti nutná podpora, stejně tak na paměťovém rozhraní. Jeden z mechanismů - LLSC (linked load, store conditional) definuje speciální ochranný interval instrukcí, kde první instrukce aktivuje arbitr a zažádá o čtení dat a kde poslední instrukce požádá arbitr o uložení a vrácení oznámení o úspěšnosti. Případné přerušení nebo modifikace s vyšší prioritou zabrání uložení hodnoty z toho ochranného intervalu.

Nad mechanismy ochrany před takovými hazardy je poté možné vybudovat prvky řízení kódu, jako semafory, mutexy apod.

2.5 Koherentní cache

Víceprocesorový systém s implementovanými datovými cache, který je navržen pro sdílení operační paměti, vyžaduje zachování konzistence dat v případě změny jedné z kopií zpracovávané informace. V případě použití rekonfigurace procesoru, který používá datovou cache, je nutné, aby byl řadič cache schopen správně vyřešit stav rekonfigurace (procesor bude odstraněn, modifikován nebo nahrazen).

2.6 Podpora programovacího portu FPGA

Rekonfigurace z vnitřních zdrojů FPGA čipu využívá speciální komponentu. Například obvody Xilinx mají port ICAP (Internal Configuration Access Port), což je rozhraní zapojitelné do hardwarového projektu. Přes ICAP rozhraní se může projekt napojit na konfigurační registry FPGA a poslat do konfigurační paměti nový popis hardware (bitstream). Stejné konfigurační rozhraní je sdíleno i JTAG (Joint Test Action Group) portem, nicméně ICAP je paralelní rozhraní a tak je konfigurace přes něj rychlejší. Obvyklé rychlosti jsou jednotky až stovky milionů slov za sekundu (bitová šířka slova je 16 bitů u řady Spartan-6 a 32 bitů u ostatních řad).

Pro podporu dynamické částečné rekonfigurace je nutné zajistit co nejrychlejší zápis (a čtení) dat. Pro tyto účely je vhodné použít mechanismy operací s pamětí (jako přímý přístup do paměti apod.) a jednoúčelové výpočetní jednotky (některé části bitstreamu je nutné přepočítat nebo upravit - zvlášť v případě readbacku). Během částečné rekonfigurace je upravovaná oblast nepoužitelná pro výpočet. V operačním systému, který tento hardware používá, je samozřejmě nutné implementovat ovladače a mechanismy pro optimální využití. Popis softwarových prostředků bude následovat v kapitole 3.

2.7 Podpora inicializace při aktivní rekonfiguraci

Během rekonfigurace hardware je nutné dodržet specifikace výrobce FPGA. Například hlavní požadavek architektury Kintex (nejen) je neměnnost hodinových signálů během modifikace konfigurace. Programování nového hardware není okamžité a daná oblast se tak může dostat do neznámého stavu [7]. Na Xilinx FPGA se tak doporučuje přivést hodinové signály přes ovladatelný buffer (BUFGCE, BUFHCE).

Stejně tak je nutná úprava datových rozhraní, které mohou překračovat hranice rekonfigurované oblasti, na nich nebude připojen žádný zdroj signálu a mohou tak přijímat šum z okolních propojů na čipu.

3 Software

Jakákoliv implementace některého hardwarového prostředku vyžaduje příslušnou podporu v software. Meziprocesorová komunikace vyžaduje podporu jádra operačního systému pro posílání zpráv z jednoho procesoru na druhý (v Linuxovém jádru se jedná o skupinu funkcí *SMP_call_function_**). Ovladač řadiče přerušení v SMP systému musí být schopen směrovat přerušení na vybraný (například nečinný) procesor. Plánovač úloh musí být schopen spočítat časová kvanta procesu pro každý SMP časovač. Operační systém musí implementovat systém zámků, mutexů, aj. pro jádro a uživatelský prostor pomocí hardwarové implementace (výjimka při selhání exkluzivního zápisu, obalení sdílených informací jádra, ...). Implementace s podporou rekonfigurace vyžaduje ovladač programovacího portu a možnost jeho použití aplikací, která provede rekonfiguraci. Stejně tak je nutná podpora jádra operačního systému pro samotný rekonfigurovatelný hardware, stávající možnosti jsou popsány v následující kapitole.

3.1 Hotplug procesoru

Rekonfigurace „hlavního“ procesoru, tedy procesoru, na kterém může být spuštěno jádro nebo aplikace z jiného hlavního procesoru, vyžaduje několik kroků:

- Ukončení procesů na procesoru *A*: Procesor *X*, který zpracovává požadavek na odstranění procesoru *A*, odešle příkaz pro přemigrování procesů. Obvuklé řešení je pomocí meziprocesorové komunikace.
- Uvedení procesoru *A* do konzistentního stavu: Zakáže se libovolné přerušení pro procesor *A*, v případě koherentní cache se provede zapsání hodnot do operační paměti, procesor *A* se zastaví. Procesor *X* zastaví hodinové signály pro procesor *A*.
- Odstranění hardware procesoru *A*, naprogramování hardware procesoru *B*: Procesor *X* použije programovací port FPGA pro vložení hardware procesoru *B*.
- Reset hardware procesoru *B* (nepovinné) a aktivace hodinových signálů procesoru *B*. Reset není povinný v případě vhodné architektury procesoru a použití readback funkcionality FPGA. V takovém případě by se do systému vrátil procesor ve stejném stavu jako byl odstraněn. Pro obecné operační systémy se tato funkce příliš nehodí. Například v operačním systému GNU/Linux není možné zachovat konzistenci systému v případě, že by byl zastaven a odebrán procesor s libovolným běžícím procesem.
- Inicializace procesoru *B* a oznámení procesoru *X* o úspěšně provedené rekonfiguraci.

3.2 Hotplug koprocesoru

Na rekonfigurovatelné prostředky je možné nahlížet také ve smyslu výpočetního koprocesoru. Koprocesor obecně nezpracovává události generované systémem (přerušení, zpracování výpadků stránek virtuální paměti, plánování úloh, ...), ale používá se jako pomocná výpočetní jednotka. Jádro operačního systému tedy koprocesor ke své činnosti nepotřebuje a případná nedostupnost koprocesoru během rekonfigurace nezpůsobí selhání.

Z programovacího hlediska lze využití koprocesoru rozdělit na:

- Úroveň instrukčního kódu, kde vykonání instrukce koprocesoru při nedostupném koprocesoru způsobí vyvolání hardwarové výjimky. Tento typ koprocesoru může být například klasický matematický koprocesor (FPU/MMX/SSE instrukce na x86 architektuře). Použití instrukčního kódu z hlediska rekonfigurace může vyžadovat zásah do syntézy architektury procesoru a s tím spojené problémy s časováním při vyhrazování prostoru pro rekonfiguraci.
- Úroveň zařízení v adresním prostoru systému. Zde se ke koprocesoru přistupuje přes registrová okna a mapované paměti. Tímto koprocesorem může být například zachytávací videokarta. Nedostupnost během rekonfigurace musí řešit ovladač v operačním systému (např PCI hotplug).
- Podmnožinou koprocesoru v adresním prostoru systému je samostatný akcelerační procesor s vlastním přístupem do (vlastní) paměti. Příkladem může být experimentální hybridní AMP (Asymmetric multiprocessing) platforma Parallella [8], která obsahuje klasické procesory ARM, ke kterým je připojen výpočetní čip Epiphany (16 nebo 64 RISC procesorů). Tyto procesory mohou být dostatečně univerzální, aby na nich bylo možné spustit stejný typ operačního systému, ale často bývají použity pro specifické DSP (Digital signal processing) algoritmy. Z hlediska operačního systému na ně nelze přemigrovat libovolnou úlohu nebo zpracování systémové události (přerušení). V operačním systému GNU/Linux konfiguruje tyto koprocesory vrstva *remoteproc*.

- Volně propojené systémy (cluster). Zde mohou mít jednotlivé uzly libovolnou architekturu procesoru, kde je na každém z nich instance operačního systému. Migrace aplikací je minimální a omezená variabilitou architektury. Jednotlivé uzly si spíše předávají samotná data. Rekonfiguraci (a výpadek) jednoho z uzlů může okamžitě nahradit uzel jiný.

3.3 Emulace

Koprocesor se během rekonfigurace stává nedostupným pro aplikaci. V některých případech ale může být požadováno, aby aplikace pokračovala v činnosti dál. Zde ale například zavolání již neexistující instrukce způsobí hardwarovou výjimku a obvykle ukončení (pád) aplikace. Obejití této nevýhody je možné pomocí emulace instrukcí (například emulace matematického koprocesoru nejen na x86). Většina procesorových architektur disponuje mechanismem oznámení adresy instrukce, která způsobila výjimku, lze tedy ze zachycené výjimky odvodit selhanou instrukci a softwarově emulovat její funkci.

Emulaci lze v GNU/Linux řešit pomocí zachycení signálů SIGFPE (výjimka matematického koprocesoru) a SIGILL (výjimka neznámé instrukce), které se generují v jádře a posílají do uživatelského prostoru. Emulace v uživatelském prostoru může potom existovat ve formě speciální knihovny.

3.4 Plánovač úloh na rekonfigurovatelném systému

Ve víceúlohovém operačním systému je nutné, aby se procesorový čas rozdělil mezi aktivní procesy. Plánovač úloh (scheduler), který má za úkol tento problém řešit, se rozhoduje na základě následujících kritérií:

- Pořadí vykonávaných procesů: Toto kritérium implementuje zejména FIFO (First in, first out) plánovač, který se hodí do systémů, kde záleží spíše na determinističnosti než na maximálním využití času procesoru.
- Latence procesů: Neboli jak dlouho trvá běh procesu.
- Odezva procesu: Zde se jedná o dobu, kdy jeden proces čeká než přijde na řadu (nový proces, nebo proces, který vyčerpal svoji část procesorového času).
- Počet procesů, kterým bude přidělen procesor z jednotku času: Důležité pro systémy s mnoha aplikacemi.
- Míra přiděleného procesorového času jednotlivým procesům: Plánovač úloh musí řešit i procesy, které vytváří podprocesy, rozdělování procesorového času uživatelům v systému a případně i různý výkon procesorů v SMP systému (například ARM big.LITTLE, kde jsou aktivní obě množiny procesorů). Jednotlivým procesům lze také přiřadit různé priority, kdy mají přijít na řadu a jak dlouhý časový úsek mají zkonzumovat.

Řešení optimálního pořadí procesů z hlediska předdefinovaných preferencí je obecně NP-hard úloha, kterou u interaktivních systému komplikuje fakt, že procesy mohou náhodně vznikat a zanikat.

V operačním systému GNU/Linux je v současné době plánovač úloh CFS (Completely Fair Scheduler) [9] založený na R-B (Red-Black) stromu, který umožňuje třídění procesů podle předvypočítané priority (čím méně procesorového času proces spotřeboval, tím má vyšší prioritu). Výhoda R-B stromu je ta, že proces s nejvyšší prioritou je k dispozici ihned (složitost *O(k)*, jádro si navíc udržuje informace o procesu s nejvyšší prioritou). Po spotřebování přiděleného času (kvantum) lze proces zařadit zpět do R-B stromu se složitostí *O(log N)*.

Dalším plánovačem v aktuálním Linuxovém jádru je FIFO a RR (Round robin). FIFO i RR se specializují na tzv. realtime úlohy, což jsou úlohy, které vyžadují minimální latence a vysoké odezvy. Obvyklá využití realtime aplikací je řízení fyzických mechanismů, například letecký/automobilový autopilot, zpracování digitálních signálů v komunikačním řetězci, regulace reaktoru apod. V těchto aplikacích

může být i pouhé zpoždění výpočtu (vlivem převzetí priority jiným procesem a větším než maximální hodnota v návrhu) bráno jako selhání. FIFO plánovač je podobný jako RR. U obou existuje fronta procesů, které se postupně vykonávají. FIFO plánovač má na rozdíl od plánovače RR neomezené množství času, po který může být proces vykonáván, dokud ho nepřeruší proces s vyšší prioritou, není zastaven při čekání na I/O nebo se sám nevzdá procesoru. Plánovač RR k tomuto přidává ještě maximální čas, po který může být proces vykonáván. Po jeho překročení je proces přepnut na další v řadě.

Linuxové jádro umožňuje definici vlastní politiky plánovače úloh. Což může být využito při implementaci plánovače, co musí vzít v potaz prostředky rekonfigurace.

Explicitní přiřazení množiny procesů na množinu procesorů je také možno ovlivňovat pomocí systému *cgroups* [10].

4 Cíle disertační práce

Moderní víceprocesorové systémy jsou založené na pevně vyrobeném hardware, který musí být natolik obecný, aby uspokojivě zpracoval libovolnou úlohu. To představuje značnou nevýhodu pro specializované úlohy, které musí být rozděleny na mnoho jednodušších částí. V případě FPGA lze tuto nevýhodu teoreticky odstranit pomocí prostředků rekonfigurace a přizpůsobit tak výpočetní systém aktuálně zpracovávané aplikaci.

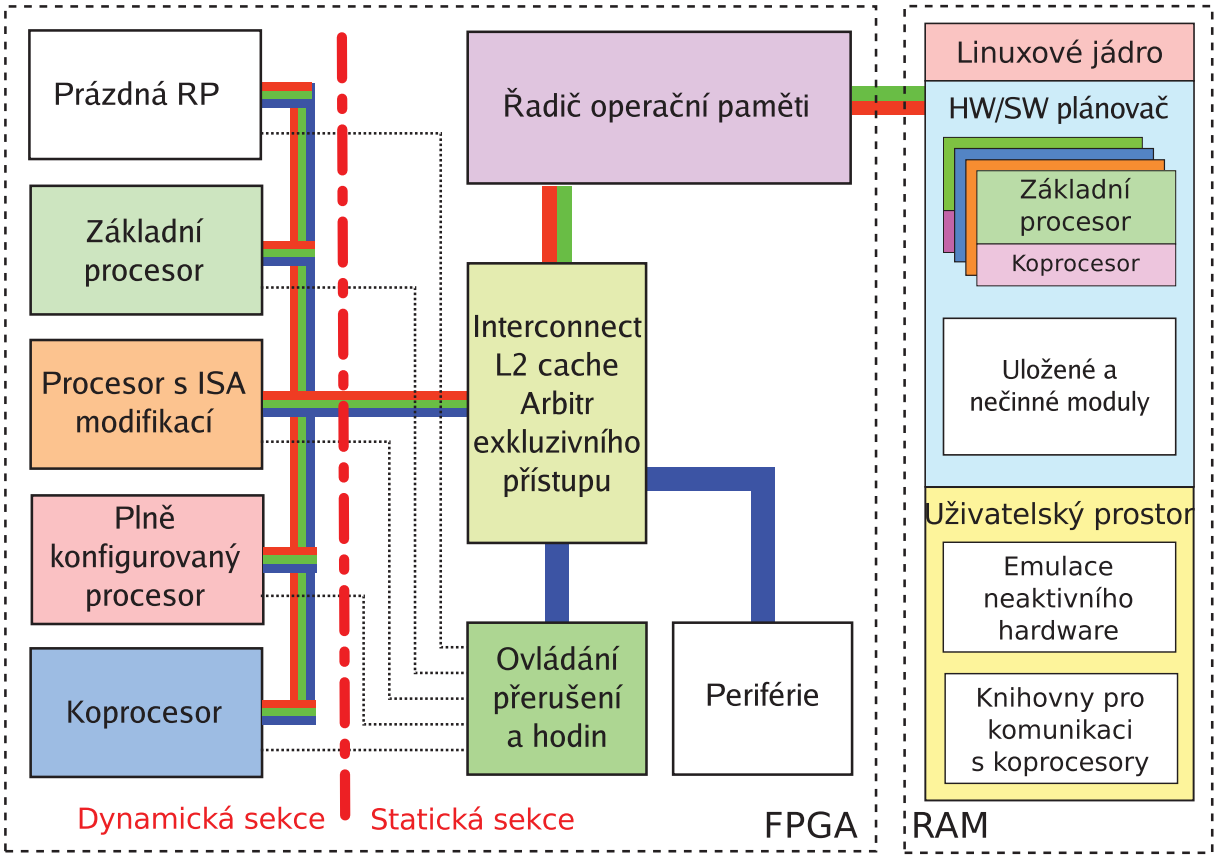
Obsahem výzkumu je vytvoření postupů návrhu pro tvorbu a použití víceprocesorových heterogenních systémů, které jsou schopné multiplexovat výpočetní úlohy v čase a umístění na FPGA. Cílem je potom popis plánovače úloh pro toto použití. S navrženým plánovačem úloh potom bude možné úlohu dynamicky přesouvat mezi obecnou a specializovanou výpočetní jednotkou za účelem efektivního využití zdrojů v FPGA obvodu.

Na rozdíl od popisovaných existujících řešení [11–17] (porovnání v [3–5]) se výzkum zaměřuje na transparentní rekonfiguraci. Transparentní rekonfigurací je myšlena taková rekonfigurace, která nevyžaduje spolupráci prováděné aplikace. Ovládání rekonfigurace (například plánovačem úloh nebo emulací instrukcí) je pak součástí operačního systému. Aplikace si potom nemusí být vědoma, že je rekonfigurace použita.

Implementace systémů podle navrhované metodologie by poté nemusela používat rekonfigurovatelné prostředky jako speciální součást systému.

Pro testování hypotéz výzkum počítá s vytvořením demonstrační platformy na vývojové desce KC705 s FPGA Xilinx Kintex a úpravou operačního systému GNU/Linux.

Na obrázku 1 je zobrazen navrhovaný systém. V levé části jsou jednotlivé oblasti dynamického hardware, tedy procesory a koprocesory. Ve střední části je statický design (tj. podpůrný hardware, systémové rozhraní, ...). V pravé části je zobrazena operační paměť. Zde je ilustrováno rozložení operačního systému. Plánovač úloh sestavuje frontu hardware, která bude postupně nahrávána do dynamické sekce FPGA. Stejně tak zde může být cache neaktivních (ko)procesorů apod. V uživatelské části operačního systému budou umístěny knihovny a emulační software pro (ko)procesory, které jsou sice vyžadovány aplikací, ale nemají dostatečnou prioritu pro umístění na FPGA čip.



Obrázek 1: Navrhovaný systém

4.1 Dosažené výsledky

Pro ujištění se, zda je hypotéza rekonfigurovatelných procesorů na základě požadavků aktivní úlohy ve víceúlohovém operačním systému možná, bylo třeba nejprve implementovat testovací systém. Některé schopnosti rekonfigurace jsou totiž omezeny nedostupností různých prostředků (například nedostupná dokumentace vnitřního zapojení FPGA pro prostředky ukládání aktuálního stavu a relokace, uzavřenost zdrojového kódu pro kritický hardware v systému, nekompletní podpora víceprocesorového systému apod.). Implementace systému proto postupuje paralelně s výzkumem plánovače a zkoumá možná řešení.

Na testovacím systému bylo do této doby provedeno několik měření základních vlastností víceprocesorového systému. Popis těchto měření byl prezentován ve zdroji [3, 5].

V době psaní toho článku je možné provádět dynamickou částečnou rekonfiguraci (manuálně na vyžádání, ale z operačního systému). Zároveň je možné vyměnit rekonfigurací procesor za procesor s omezenými prostředky a chybějící softwarově emulovat (experimentálně ověřeno emulátorem všech variant instrukcí celočíselného násobení a dělení instrukční sady Microblaze).

5 Publikace

Popisy jednotlivých částí práce byly publikovány na konferenci ECMSM 2013 [5]. V době psaní tohoto článku, byl přijatý článek na dosud neproběhlou konferenci ECMSM 2015. Porovnání systémů a některé naměřené výsledky byly prezentovány na minulých konferencích PAD 2012, 2013 a 2014.

6 Závěr

Výzkum víceprocesorového hybridního systému umožňuje zvýšit výpočetní výkon úlohy pro obvod FPGA ve smyslu multiplexace výpočetních jednotek v čase (a lokaci na čipu). Stejně tak sleduje trend zvyšování výpočetního výkonu pomocí paralelizace. Od výzkumu se očekává přínos v metodě vývoje paralelních výpočetních aplikací realizovaných pomocí částečné dynamické rekonfigurace FPGA.

Obsah výzkumu byl zpracován jako teze dizertační práce. V současné době probíhá jedna z posledních fází: návrh Linuxového plánovače úloh.

Poděkování

Rád bych poděkoval mému školiteli prof. Ing. Ondřejovi Novákovi, CSc. a kolegovi Ing. Martinu Rozkovcovi, PhD za cenné rady a zkušenosti během mého výzkumu. Stejně také za podporu výzkumu v rámci projektu SGS 2015 na Technické Univerzitě v Liberci.

Reference

[1] Xilinx. *Soft Error Mitigation Controller, PG036*, 2014.

[2] Xilinx. *Partial Reconfiguration, UG909*, 2014.

[3] P. Cvek and O. Novak. Generic gnu/linux reconfiguration platform proposal. In *Electronics, Control, Measurement, Signals and their application to Mechatronics (ECMSM), 2015 IEEE 13th International Workshop of*, June 2015.

[4] P. Cvek. Gnu/linux and reconfigurable multiprocessor fpga platform. 2014.

[5] P. Cvek, T. Drahonovsky, and M. Rozkovec. Gnu/linux and reconfigurable multiprocessor fpga platform. In *Electronics, Control, Measurement, Signals and their application to Mechatronics (ECMSM), 2013 IEEE 11th International Workshop of*, pages 1–5, June 2013. doi: 10.1109/ECMSM.2013.6648932.

[6] Intel. *MultiProcessor Specification*, 1997.

[7] C. Beckhoff, D. Koch, and J. Torresen. Short-circuits on fpgas caused by partial runtime reconfiguration. In *Field Programmable Logic and Applications (FPL), 2010 International Conference on*, pages 596–601, Aug 2010. doi: 10.1109/FPL.2010.117.

[8] Adapteva. *Parallella 1.x Reference Manual*, 2014.

[9] IBM. Inside the linux 2.6 completely fair scheduler, 2009. URL `<http://www.ibm.com/developerworks/linux/library/l-completely-fair-scheduler/index.html>`.

[10] Paul Menage. Linux kernel documentation: Cgroups, 2015. URL `<https://www.kernel.org/doc/Documentation/cgroups/cgroups.txt>`.

[11] A. Krasnov, A. Schultz, J. Wawrzynek, G. Gibeling, and P.-Y. Droz. Ramp blue: A message-passing manycore system in fpgas. In *Field Programmable Logic and Applications, 2007. FPL 2007. International Conference on*, pages 54 –61, 2007. doi: 10.1109/FPL.2007.4380625.

[12] D. Gohringer and J. Becker. High performance reconfigurable multi-processor-based computing on fpgas. In *Parallel Distributed Processing, Workshops and Phd Forum (IPDPSW), 2010 IEEE International Symposium on*, pages 1 –4, 2010. doi: 10.1109/IPDPSW.2010.5470800.

[13] V. Rana, M. Santambrogio, D. Sciuto, B. Kettelhoit, M. Koester, M. Porrmann, and U. Ruckert. Partial dynamic reconfiguration in a multi-fpga clustered architecture based on linux. In *Parallel and Distributed Processing Symposium, 2007. IPDPS 2007. IEEE International*, pages 1 –8, 2007. doi: 10.1109/IPDPS.2007.370363.

[14] M.A. Kinsy, M. Pellauer, and S. Devadas. Heracles: Fully synthesizable parameterized mips-based multicore system. In *Field Programmable Logic and Applications (FPL), 2011 International Conference on*, pages 356 –362, 2011. doi: 10.1109/FPL.2011.70.

[15] Hoyden Kwok-Hay So and R.W. Brodersen. Improving usability of fpga-based reconfigurable computers through operating system support. In *Field Programmable Logic and Applications, 2006. FPL '06. International Conference on*, pages 1 –6, 2006. doi: 10.1109/FPL.2006.311236.

[16] E. Matthews, L. Shannon, and A. Fedorova. Polyblaze: From one to many bringing the microblaze into the multicore era with linux smp support. In *Field Programmable Logic and Applications (FPL), 2012 22nd International Conference on*, pages 224–230, 2012. doi: 10.1109/FPL.2012.6339185.

[17] G. Kuzmanov, G. Gaydadjiev, and S. Vassiliadis. The molen processor prototype. In *Field-Programmable Custom Computing Machines, 2004. FCCM 2004. 12th Annual IEEE Symposium on*, pages 296 – 299, 2004. doi: 10.1109/FCCM.2004.55.

Kapilárne siete internetu vecí

Ondrej Perešíni, školiteľ: Tibor Krajčovič
Ústav počítačových systémov a sietí, 2014/2015, PhD prezenčné štúdium, 1.ročník

Fakulta informatiky a informačných technológií, Slovenská technická univerzita v Bratislave
Ilkovičova 2, 842 16 Bratislava 4
ondrej.peresini@stuba.sk

Abstract. Idea internetu vecí nie je nová, avšak k násobnému rozširovaniu rôznych typov zariadení komunikujúcich cez sieťové technológie dochádza až vplyvom miniaturizácie polovodičových čipov a ich energetickej spotreby. Najväčším problémom znemožňujúcim rýchle rozširovanie takýchto zariadení je absencia štandardizovaného sieťového protokolu, ktorý by abstrahoval sieťovú komunikáciu od použitej komunikačnej technológie. Cieľom dizertačnej práce je návrh takéhoto protokolu, ktorý je založený na zadefinovanom matematickom modeli, ktorý zohľadňuje rôzne komunikačné limity plynúceho z takýchto sietí.

Keywords: internet vecí, homogénne siete, inteligentná domácnosť, decentralizovaná sieť, vnorený hardvér

1 Úvod a súčasný stav problematiky

Vývoj systémov inteligentného riadenia domácností prináša množstvo otázok a výziev, ktoré smerujú najmä na oblasť ich zabezpečenia a vzájomnej kompatibility. Súčasné implementácie inteligentných domácností pozostávajú z viacerých senzorických modulov a akčných členov. Nezriedka býva v takomto systéme nevyhnutná aj ďalšia centrálna jednotka, ktorá buď zabezpečuje logiku a automatizáciu riadenia alebo sprístupňuje používateľské rozhranie. Jednotlivé moduly komunikujú len s centrálnou jednotkou a nedokážu inicializovať vzájomnú komunikáciu medzi sebou, čo by umožnilo efektívnejšiu výmenu informácií. Práve do tejto oblasti vstupuje myšlienka internetu vecí, ktorej cieľom je pospájať všetky senzory do jedného systému, ktorý bude zdieľať pripojenie do internetu. Všetky takéto zariadenia pritom poskytujú určitú funkcionalitu, ktorá prináša dodatočnú hodnotu pre používateľa, keďže inteligentné objekty vzájomne kooperujú s fyzickými alebo virtuálnymi zdrojmi. Takéto zdroje sú charakteristické vysokým stupňom heterogénnosti, čím poskytujú rôznu funkcionalitu aj zariadeniam, ktoré boli pôvodne len jednoúčelové. Na to aby priraďovanie takýchto zdrojov fungovalo korektne, tak je treba navrhnúť dostatočne abstraktný model opisujúci danú komunikáciu a interoperabilitu medzi nimi v rámci kapilárnych sietí, ktoré pozostávajú z rôznych bezdrôtových, mobilných a pevných sietí. Každý modul má individuálne a špecifické požiadavky na prenosové parametre a popri tom je žiaduce minimalizovať energetické nároky tak, aby niektoré senzory mohli fungovať na jeden zdroj energie až niekoľko rokov.

2 Definícia problémových oblastí

Internet vecí

Internet vecí je reprezentovaný množstvom zariadení, ktoré spolu vzájomne komunikujú a prinášajú dodatočnú funkcionalitu, ktorú by jednotlivé zariadenia samostatne nedokázali priniesť. Dobrým príkladom je domáca televízia a inteligentný termostat. Termostat obsahuje teplotný senzor, ktorý umožňuje regulovanie teploty, avšak nedokáže zobrazovať podrobnejšie informácie o nameraných teplotách. Idea internetu vecí umožní termostatu využiť funkcie iných zariadení, kde napr. televízia poskytne funkciu zobrazenia. Používateľ si tak môže pozrieť grafické priebehy teploty a výkonu vykurovania alebo chladenia, čo by samostatný termostat nedokázal.

Myšlienka internetu vecí reprezentuje súbor vzájomne prepojených zariadení s pripojením do internetu [1], ktoré pozostávajú zo senzorov, akčných členov, projekčných a audiovizuálnych zariadení a mnohých ďalších zariadení, pričom tieto zariadenia nemusia logicky a ani funkčne súvisieť. Zariadenia vzájomne komunikujú [2] a poskytujú si funkcionalitu obohacujúcu celý systém. Komunikácia pritom môže byť vedená heterogénnymi kapilárnymi sieťami, najmä z dôvodu rozdielnych komunikačných požiadaviek, kde senzory na rozdiel od kamerového systému nepotrebujú vysokú dátovú priepustnosť. Každý modul musí byť jednoznačne identifikovaný a musí poskytovať zoznam svojich funkcionalít a požiadaviek, tak aby mohli spolu komunikovať aj viac-menej nekompatibilné moduly. Vzniknutý systém pozostáva z dynamickej štruktúry zariadení s dodatočnou funkcionalitou. Pre zaručenie vzájomnej komunikácie v rámci heterogénneho systému je nutné vyšpecifikovať štrukturálny model [3] definujúci vzájomnú výmenu dát. Takýto model [4] zabezpečuje správu senzorov, činností jednotlivých modulov, užívateľského prístupu, rolí a oprávnení.

Bezpečnostný model.

Dôležitým kritériom nasadenia internetu vecí je dodržanie bezpečnostných pravidiel na zariadeniach a systéme [5], ktoré sú opísané v nasledujúcich aspektoch systému:

- Dôvernosť – služby môžu obsahovať citlivé údaje a preto by všetky zúčastnené zariadenia malí spĺňať stanovené bezpečnostné kritériá ako napr. šifrovanie.
- Integrita – medzi službami v rámci jedného systému môžu byť preposielané kritické údaje, ktoré nesmú byť žiadnym spôsobom zmenené.
- Dostupnosť – v prípade výpadku modulu príde celý systém o danú funkcionalitu, takže zariadenia musia spĺňať požiadavku na neustálu dostupnosť.
- Autentifikácia [6] – overenie identity používateľa musí byť spoľahlivé a presné. Neoprávnený prienik do systému znamená kompromitáciu údajov a nastavení.
- Autorizácia a správa prístupov – priradenie prístupových práv na základe systémového profilu. Prístupové práva sú viacúrovňové podľa úrovne privilegovanosti.
- Dôveryhodnosť – cieľový systém musí byť jednoznačne identifikovateľný a preukázateľný, že sa jedná o správny systém a nie o útočníkom podvrhnutý model.
- Auditovanie – prístupov a požiadaviek užívateľov. Systém musí viesť záznam o vykonaných službách a k nim pričleneným požiadavkám.

Kapilárne siete

Pri presadzovaní myšlienky internetu vecí je dôležité zabezpečiť schopnosť vzájomnej komunikácie medzi zariadeniami. Bez možnosti zdieľania informácií sa zariadenia stávajú jednoúčelovými senzormi alebo akčnými členmi bez ďalšej pridanej hodnoty.

Väčšina súčasne používaných bezdrôtových sietí je založená na infraštruktúrnom modeli, ktorý obsahuje zariadenia dvoch kategórií: prístupové body a koncové zariadenia. Základnou vlastnosťou takýchto sietí je, že všetky koncové zariadenia sa pripájajú len na zvolený prístupový bod, cez ktorý realizujú všetky svoje dátové prenosy. Komunikáciu moderuje a riadi centrálna autorita, ktorá je reprezentovaná jedným alebo viacerými prístupovými bodmi. Druhý komunikačný model typu Ad-hoc neobsahuje centrálnu autoritu a všetky bezdrôtové zariadenia spadajú pod jeden komunikačný segment, ktorý je riadený všetkými komunikačný uzlami. Každý uzol dokáže priamo komunikovať so susedným uzlom pokiaľ je v dosahu pokrytia a aj bez prístupového bodu.

Myšlienka internetu vecí spočíva v automatickom prepájaní komunikujúcich zariadení bez potreby rozsiahlej konfigurácie a budovania infraštruktúrneho pokrytia, čo nekorešponduje s infraštruktúrnym modelom vzhľadom na cenu, energetickú efektívnosť a citlivosť na útoky. Čiastočným riešením popisovaných problémov je použitie Ad-hoc modelu, ktorý prináša autonómnosť od infraštruktúry, väčšie pokrytie a zväčša aj menšiu energetickú náročnosť spoločne s nízkou cenou vyplývajúcou z absencie prístupových bodov. Robustnosť riešenia navyše umožňuje samo-uzdravovanie takejto siete, kde sú za určitých podmienok nahradzované nefunkčné alebo napadnuté moduly.

Klasifikácia parametrov M2M (Machine-to-Machine) sietí.

Pri definícii systémového modelu je nutné špecifikovať niekoľko parametrov:

- Oneskorenie – závislosť od času potrebného na prešírenie informácie v rámci siete.
- Priepustnosť – určiť mieru závislosti jednotlivých modulov od priepustnosti.
- Spoľahlivosť – prenosu musí byť zaručená nezávisle od sieťovej technológie.
- Priorita prístupu – pre kritické aplikácie, QoS a prioritizácia komunikácie.
- Mobilita – určujúca mieru migrácie zariadení a vplyv na komunikačné vlastnosti.
- Zabezpečenie – komunikácie a smerovania najmä v Ad-hoc sieťach.
- Energetická efektívnosť – a úsporné cykly spánku pre zariadenia napájané z batérie.
- Správa toku komunikácie – zamedzujúca stratu údajov a smerovacích tabuliek v prípade cyklických úsporných režimov zariadení.
- Rozšíriteľnosť – pre nárast počtu komunikujúcich uzlov a funkcionalít.

Najpoužívanejšie komunikačné protokoly.

Keďže M2M komunikácia využíva heterogénne komunikačné technológie, tak je nesmierne dôležité zabezpečiť vhodný komunikačný model, ktorý umožní bezproblémovú, transparentnú a nezávislú výmenu dát. Napriek vysokým energetickým nárokom je pre M2M komunikáciu často používaná technológia Wi-Fi (*IEEE 802.11*), ktorej výhodou je všade-prítomnosť a jednoduchý prístup do internetu. Naopak medzi nízkoenergetické protokoly patria napríklad Bluetooth Smart, či IEEE 802.15.4. Tieto tech-

nológie sú cielené na aplikácie s nízkymi nárokmi na priepustnosť ako napríklad senzorické siete a rôzne kontrolné systémy [7]. Zatiaľ čo IEEE 802.15.4 reprezentuje len nižšie (MAC) prístupové vrstvy, tak na zabezpečenie pripojenia do internetu [8] je ešte potrebné použiť protokol vyššej vrstvy ako napríklad 6LoWPAN (*IPv6 over Low power Wireless Personal Area Networks*) [9 10]. Vďaka použitiu týchto komunikačných protokolov je možné sprístupniť pripojenie do internetu aj pre nízkoenergetické zariadenia. Jednou nevýhodou samotného IEEE 802.15.4 je na rozdiel od IEEE 802.11 nevyhnutnosť použitia prístupovej brány pre prístup do internetu [11], keďže 802.11 a MAC/IP protokol používa inú štruktúru komunikačného rámca.

3 Ciele a tézy dizertačnej práce

Cieľom dizertačnej práce je návrh a realizácia otvorenej architektúry internetu vecí zabezpečujúcej komunikáciu s rôznymi požiadavkami na poskytovanú kvalitu služby cez kapilárne siete pozostávajúce z bezdrôtových, mobilných a pevných sietí. Z tohto dôvodu je nutné zhodnotiť vlastnosti jednotlivých komunikačných metód a predstaviť riešenie optimalizácie komunikácie a smerovania v rámci vzniknutej heterogénnej siete. Navrhnutý model kapilárnych sietí musí rešpektovať dynamiku migrácie zariadení a konektivitu do internetu. Medzi najdôležitejšie oblasti rozvoja projektu patrí:

- Návrh modelu integrovaného zabezpečenia na rôznych úrovniach systému.
- Návrh pokročilých metód Ad-hoc smerovania a pripojenia do internetu.
- Globálna správa siete a kolektívne rozhodovanie podľa aktuálnych parametrov systému (senzorické siete majú iné požiadavky na smerovanie, kvalitu služieb a komunikačný protokol ako v prípade multimediálnych zariadení).
- Formovanie lokálnych Ad-hoc sietí spájajúcich rôzne komunikačné technológie.
- Model pre vzájomné zdieľanie dostupných prostriedkov v rámci segmentu.
- Návrh systému propagácie poskytovanej funkcionality a spracovania požiadaviek ostatných modulov podľa vopred zadefinovaných pravidiel.
- Návrh modelu komunikačného protokolu, správy prístupov a riadenia zahltenia.

Navrhovaný model umožňuje:

- Rozšírenie jednoúčelových zariadení o doplnkovú funkcionalitu vyplývajúca z členstva v segmente s ostatnými kooperujúcimi zariadeniami.
- Segmentáciu podľa príslušnosti modulov k jednému systému.
- Používateľský zásah do takéhoto systému.
- Vzdialenú správu jednotlivých zariadení.

Komunita vzájomne zoskupených zariadení podlieha pod správu jedného Ad-hoc systému. Je nutné definovať komunikačný protokol popisujúci:

- Objavovanie susediacich zariadení, ich poskytovaných služieb a funkcionalít.
- Propagáciu typov zariadení, ich funkcií a požiadaviek.
- M2M Ad-hoc komunikáciu v rámci heterogénneho segmentu.
- Automatické zabezpečenie pripojenia do internetu cez viacero brán.

Dôležitým cieľom je návrh modelu, ktorý bude dostatočne robustne špecifikovať poskytované služby a bude ich vedieť klasifikovať vo forme stromu (*„capabilitiy tree"*). Tieto služby a funkcionality musia byť dynamicky distribuované v rámci celého systému. Pre úspešné dosiahnutie stanovených cieľov projektu je nutné vyriešiť problémy, ktoré vyplývajú z téz dizertačnej práce. Projekt smeruje k návrhu otvoreného systému združujúceho množstvo rôznorodých zariadení, ktoré komunikujú cez heterogénne siete. Toto zadanie si vyžaduje vyšpecifikovať matematický model, ktorý bude:

- Definovať metriku kvantifikujúcu dynamické správanie Ad-Hoc sietí.
- Definovať model smerovacieho protokolu v heterogénnych sieťach (DV, LS a iné).
- Matematicky vyjadriť dopady pridávania ďalších zariadení do existujúceho systému.
- Navrhnúť model prioritizácie komunikácie (podľa typu služby, náročnosti na priepustnosť, oneskorenia, zabezpečenia a typu poskytovanej služby).
- Špecifikovať kategorizáciu zariadení, služieb a komunikačného protokolu.
- Navrhnúť model realizácie kolektívneho riadenia systému, prístupu a zahltenia.

Matematický model musí vyjadrovať limitné hranice pre:

- Kapacity a limity jednotlivých komunikačných kanálov.
- Dynamiku zmien komunikačných parametrov pri zmene kvality spojenia a pri použití úsporných režimov na rôznych zariadeniach.
- Maximálnu zmenu počtu a výskytu zariadení pre zabezpečenie nepretržitej služby.

Sieťový model musí obsahovať mechanizmy pre:

- Vzájomnú detekciu rôznych zariadení.
- Propagáciu funkcionalít a požiadaviek v rámci siete.
- Zabezpečenie priamej komunikácie zariadení bez použitia mediátora.
- Automatickú konfiguráciu pripojenia do internetu.

4 Záver a smerovanie výskumu

Analýza odhalila viaceré problémy tejto oblasti, ktoré spočívajú v uzavretosti používaných štandardov a vzájomnej nekompatibility medzi rôznymi riešeniami. Je nutné zabezpečiť bezproblémovú komunikáciu viacerých a hlavne rôznych zariadení, ktoré si budú vzájomne vymieňať informácie a tak tvoriť systém s doplnkovou funkcionalitou, akú by samostatné zariadenia nevedeli poskytnúť. Súčasné smerovacie a komunikačné protokoly sa sústreďujú len na jednu komunikačnú technológiu, čo ich robí prinajlepšom neefektívne pri použití v spojení s inými technológiami. Jednotlivé oblasti Ad-hoc sietí a M2M komunikácie sú už značne preštudované, avšak heterogénne kapilárne siete prinášajú značný potenciál pre ďalší rozvoj. Výstupom projektu bude návrh komunikačného protokolu, ktorý bude zjednocovať spôsob zdieľania funkcionalít a požiadaviek. Zo zabezpečenia aplikačnej komunikácie vyplýva aj návrh vhodného sieťového protokolu, ktorý premostí siete s rôznymi prístupovými metódami. Tento protokol musí byť dostatočne robustný a samo-adaptujúci, tak aby sa prispôsobil požiadavkám celého systému a zároveň umožňoval využívať všetky prostriedky kapilárnych sietí.

5 Literatúra

1. F. Wortmann, K. Fluchter.: Internet of Things: Technology and Value Added. Bus Inf Syst Eng 57(3). 2015. s. 221–224
2. W. Zhu, H. Lu, X. Cui.: Distributed Relation Discovery in Internet of Things. 2014 International Conference on Cloud Computing and Big Data. s. 39-46
3. E. Cavalcante, M. P. Alves, T. Batista, F. C. Delicato, P. F. Pires.: An Analysis of Reference Architectures for the Internet of Things. CobRA'15, May 6, 2015, s. 13-16
4. S. Alam, M. M. R. Chowdhury, J. Noll.: Interoperability of Security-Enabled Internet of Things. Wireless Pers Commun 2011. s. 567–586
5. A. F. Skarmeta, J. L. Hernández-Ramos, M. Victoria Moreno.: A decentralized approach for Security and Privacy challenges in the Internet of Things. 2014 IEEE. WF-IoT. s. 67-72
6. I. Alqassem.: Privacy and security requirements framework for the internet of things (IoT). ICSE Companion 2014. ACM, New York, NY, USA, s. 739-741
7. N. Accettura, M. R. Palattella, M. Dohler, L. A. Grieco, G. Boggia.: Standardized Power-Efficient & Internet-Enabled Communication Stack for Capillary M2M Networks. WCNC 2012 Workshop on IoT, Embracing M2M Communications and Beyond. s. 226-231
8. S. M. Sajjad, M. Yousaf.: Security Analysis of IEEE 802.15.4 MAC in the context of Internet of Things. 2014 Conference on Information Assurance and Cyber Security. s. 9-14
9. M. R. Palattella, N. Accettura, X. Vilajosana, T. Watteyne, L. A. Grieco, G. Boggia, M. Dohler.: Standardized Protocol Stack for the Internet of (Important) Things. IEEE Communication surveys & Tutorials, Vol. 15, No. 3, Third Quarter 2013. s. 1389-1406
10. R. Giuliano, F. M. A. Neri, A. M. Vegni.: Security Access Protocols in IoT Networks with Heterogenous Non-IP Terminals. 2014 IEEE International Conference on Distributed Computing in Sensor Systems. s. 257-262
11. T. Zachariah, N. Klugman, B. Campbell, J. Adkins, N. Jackson, P. Dutta.: The Internet of Things Has a Gateway Problem. HotMobile'15, Feb. 12–13, 2015, Santa Fe, USA. s. 27-32

Univerzálny BIST pre testovanie vnorených pamätí v systémoch na čipe

Juraj Šubín

Aplikovaná informatika, 1. ročník, denná forma
Školiteľka: doc. RNDr. Elena Gramatová, PhD.

Fakulta informatiky a informačných technológií
Slovenská technická univerzita
Ilkovičova 2, 84216 Bratislava 4

juraj.subin@stuba.sk

Abstrakt. Vstavané samočinné testovanie pamätí (BIST) je v súčasnosti najrozšírenejšou metódou pre testovanie pamätí vnorených v systémoch na čipe. Je to dané predovšetkým vysokým počtom pamätí (stovky až tisícky), a tiež ich nedostupnosťou pre privedenie testovacích signálov z externého testovacieho zariadenia. Cieľom je navrhnúť novú stratégiu pre generovanie blokov BIST pre testovanie viacerých pamätí v jednom čase, či už paralelnou, alebo sériovou metódou. Vhodnou kombináciou týchto metód je snaha dosiahnúť čo najlepšie výsledky z týchto hľadísk: pridaná plocha na čipe z dôvodu testovania, spotreba energie a čas testovania. Stratégia testovania vnorených pamätí bude implementovaná v systéme pre automatické generovanie architektúry BIST, ktorého vstupom budú parametre pamätí, výber testov march, cieľový parameter architektúry, ako aj obmedzenia zadané používateľom a výstupom bude opis optimálnej architekrúry BIST v jazyku VHDL.

Keywords: testovanie, vnorená pamäť, systém na čipe, VHDL.

1 Úvod

Pamäte zaberajú v súčasnosti najväčší podiel z plochy systémov na čipe (SoC – *system on chip*), približne 86 % a viac [1]. Často sú integrované hlboko v jednotlivých funkčných blokoch a ich porty sú neprístupné z externého prostredia. Principiálne existujú dve možnosti na otestovanie vnorených pamätí, a to buď pomocou automatického testovacieho zariadenia (ATE – *automatic test equipment*), alebo pomocou testovacej logiky, ktorá je umiestnená priamo na čipe (BIST – *built-in self-test*). Kvôli spomínaným problémom s dostupnosťou portov pamätí je testovanie použitím ATE takmer nerealizovateľné. Ďalším faktorom komplikujúcim použitie ATE je počet vnorených pamätí, ktorý sa v súčasnosti pohybuje v stovkách, až tisíckach. Vzhľadom na uvedené skutočnosti vychádza testovanie pamätí použitím testovacej logiky umiestnenej na čipe ako jediné prijateľné riešenie.

adfa, p. 1, 2011.

Otestovať stovky až tisícky pamätí pomocou testovacej logiky umiestnenej na čipe však nie je triviálna úloha. Pri návrhu architektúry BIST je potrebné brať do úvahy plochu na čipe, ktorú si architektúra vyžiada, čas testovania alebo spotrebu energie počas testovania. Návrh takejto komplexnej architektúry BIST manuálne a intuitívne je príliš zložité, a preto sa žiada vyvinúť vhodný systém na automatické generovanie architektúry BIST podľa zadaných parametrov.

Článok je rozdelený do štyroch častí vrátane úvodu a záveru. Druhá časť opisuje výsledky analýzy vlastností, parametrov architektúr BIST pre testovanie vnorených pamätí v SoC, tretia časť je zameraná na ciele a potenciálne tézy dizertačnej práce a posledná, štvrtá časť, je záverom.

2 Analýza vlastností systému a využívaných metód

Pre správne fungovanie systému je potrebné zvoliť vhodné metódy testovania a riadenia, ktoré budú využívané na riešenie čiastkových problémov, ako je zdieľanie testovacej logiky viacerými pamäťami alebo aké testy bude navrhnutá architektúra BIST generovať.

Doteraz bolo publikovaných niekoľko systémov, napr. [2], [3]. Všetky existujúce systémy majú svoje špecifické obmedzenia, prípadne vyhradené zameranie. V niektorých prípadoch boli systémy vyvíjané podľa požiadaviek konkrétnej firmy. Z uvedených skutočností vyplýva motivácia na návrh univerzálnejšieho alebo komplexnejšieho systému na automatické generovanie architektúry BIST.

2.1 Sledované parametre generovanej architektúry BIST

Navrhovaný systém generovania architektúr BIST by mal používateľovi poskytovať možnosť voľby parametra, ktorý bude predstavovať hlavný cieľ – prvoradú vlastnosť generovanej architektúry BIST. Parametre, z ktorých sa predpokladá výber, sú: Plocha na čipe, spotreba energie a čas testovania.

Napriek používateľom zvolenému parametru, systém nebude môcť generovať architektúru BIST len za účelom dosiahnutia optimálnych výsledkov zameraných na zvolený parameter, ale bude musieť brať do úvahy aj ostatné parametre.

Pre dosiahnutie čo najmenšej plochy na čipe sa bude systém snažiť maximalizovať možnosti zdieľania blokov BIST medzi pamäťami. Obmedzenia budú maximálny čas testovania a maximálna spotreba energie systému na čipe.

Najnižšia spotreba energie sa teoreticky dosahuje pri sériovom testovaní všetkých pamätí. Hlavným obmedzením pre sériovú aplikáciu testovania bude maximálny čas testovania.

Najkratší čas testovania sa teoreticky dosahuje pri využití paralelného testovania v maximálnej možnej miere. V tomto prípade bude najväčším obmedzením maximálna spotreba energie SoC.

2.2 Zdieľanie bloku BIST

Bez možnosti zdieľania jednotlivých blokov BIST by nebolo možné splniť požiadavku na čo najmenšiu plochu na čipe. Zdieľaný blok BIST môže vykonávať testovanie viacerých pamätí buď paralelne, alebo sériovo. Samozrejme, pre zdieľanie bloku BIST viacerými pamäťami musia byť zadefinované pravidlá a sú to:

- Zdieľanie bloku BIST medzi pamäťami rôzneho typu sa v systéme nepredpokladá. Napríklad pamäte typu ROM a typu RAM sa testujú rozdielnymi metódami a takýto zdieľaný blok by pravdepodobne zabral toľko plochy na čipe, ako samostatný blok pre každú z pamätí.
- Dôležitým obmedzením pre zdieľanie bloku BIST je vzdialenosť pamätí, resp. ich umiestnenie v SoC. Systém bude musieť pri generovaní architektúry brať do úvahy aj maximálnu dĺžku prepojenia medzi blokom BIST a testovanou pamäťou, aby pri šírení testovacích signálov nedochádzalo k oneskoreniam a prípadným hazardom.
- V súčasných systémoch na čipe sa spravidla nachádza viacero časových domén, a teda pamäte pracujú aj na rôznych operačných frekvenciách. Momentálne sa zdieľanie bloku BIST medzi pamäťami pracujúcimi na rôznych operačných frekvenciách nepredpokladá. Avšak pri nájdení vhodnej metódy na realizáciu bloku BIST s podporou takejto funkcionality sa zdieľanie nevylučuje.
- V niektorých existujúcich prístupoch je možné zdieľať blok BIST pre paralelné testovanie pamätí len vtedy, ak pamäte majú rovnakú dĺžku slova [2], alebo ak pamäte majú rovnaký počet slov [3].
- Pre zdieľanie bloku BIST pre sériové testovanie pamätí sa takisto vyskyje v existujúcich prístupoch niekoľko obmedzení. V [2] sa môžu sériovo testovať len pamäte s rovnakou spotrebou energie. V [3] musia mať pamäte rovnakú dĺžku slova a v [4] musia byť pamäte rovnakého typu aj veľkosti.

Jedným z cieľov práce je navrhnúť metódu, ako odstrániť vyššie spomenuté obmedzenia. Metóda by mala umožňovať paralelné aj sériové testovanie pamätí s rôznou veľkosťou aj s rôznou dĺžkou slova. Napríklad v článku [5] autori dosiahli schopnosť paralelného testovania pamätí rôznych veľkostí tolerovaním redundantných vykonaní testovacích operácií nad menšími pamäťami. Nevýhoda tohto prístupu je v spotrebe energie, kedy pamäte menších veľkostí môžu byť od testovania odstavené, kým sa dokončí testovanie pamätí väčších veľkostí.

2.3 Testovacie algoritmy

Dôležitým aspektom testovania pamätí je výber vhodného testovacieho algoritmu, ktorý bude mať čo najnižšiu zložitosť a čo najvyššie pokrytie porúch. V minulosti sa používali algoritmy typu Galloping, Walking, Butterfly alebo Sliding Diagonal [7]. Ich výhodou je dobré pokrytie porúch, nevýhodou vysoká zložitosť, ktorá rastie exponenciálne v závislosti od veľkosti testovanej pamäte. Z toho dôvodu sú v súčasnosti najrozšírenejšie algoritmy typu march, ktoré dosahujú rovnaké pokrytie porúch ako vyššie spomenuté algoritmy, avšak ich veľkou výhodou je lineárna zložitosť [6].

Testy typu march pozostávajú z postupnosti operácií zápisu a čítania, vykonávaných na bunkách pamäte. Rôznymi kombináciami týchto operácií sa dá dosiahnuť rôzne pokrytie modelov porúch. Operácie sú zoskupené do elementov. Element testu march je skupina operácií, ktoré sa vykonajú na jednej pamäťovej bunke za sebou a až potom sa prechádza na testovanie ďalšej pamäťovej bunky. V navrhovanom systéme si bude používateľ môcť vyberať z testov typu march.

Oproti existujúcim systémom je potrebné používateľovi ponúknuť aj možnosť voľby adresnej schémy, ktorá sa použije v teste. Vhodnou adresnou schémou sa dá test zacieliť na poruchy v adresnom dekóderi. Klasická lineárna adresná schéma, v ktorej sa vykonáva inkrementácia alebo dekrementácia adries deteguje len zlomok porúch, ktoré sa môžu v adresnom dekóderi vyskytnúť. Ak sa vyžaduje pokrytie napríklad aj dynamických porúch v adresnom dekóderi, je potrebné použiť takú adresnú schému, v ktorej majú po sebe idúce adresy Hammingovu vzdialenosť rovnú 1 [8]. Príklady niekoľkých adresných schém sú uvedené v tabuľke 1.

Tabuľka 1. Príklady adresných schém [9]

Krok	Li	Ac	Gc	$2^i = 4$	Pr	Wc
0	0000	0000	0000	0000	0000	-
1	0001	1111	0001	0100	0001	0001
2	0010	0001	0011	1000	0011	0000
3	0011	1110	0010	1100	0111	0001
4	0100	0010	0110	0001	1111	-
5	0101	1101	0111	0101	1110	0010
6	0110	0011	0101	1001	1101	0000
7	0111	1100	0100	1101	1010	0010
8	1000	0100	1100	0010	0101	-
9	1001	1011	1101	0110	1011	0100
10	1010	0101	1111	1010	0110	0000
11	1011	1010	1110	1110	1100	0100
12	1100	0110	1010	0011	1001	-
13	1101	1001	1011	0111	0010	1000
14	1110	0111	1001	1011	0100	0000
15	1111	1000	1000	1111	1000	1000

Možnosť použitia niektorej z vyššie uvedených adresných schém v teste march dokumentujú pravidlá, ktoré vymedzujú voľnosť úprav testov march bez toho, aby sa zmenilo ich pokrytie porúch [8]:

- Adresná sekvencia (poradie adries jednotlivých buniek pamäte) môže byť ľubovoľne zvolená, ak sa každá adresa pamäte vyskytuje v danej sekvencii presne raz a sekvenciu je možné zopakovať v opačnom poradí.
- Adresná sekvencia pre inicializáciu pamäte môže byť ľubovoľne zvolená, ak sa každá adresa pamäte vyskytuje v danej sekvencii aspoň raz.
- Ak je test napísaný symetricky, testovacie údaje môžu byť invertované.

- Údaje, s ktorými pracuje operácia zápisu/čítania nemusia byť nevyhnutne rovnaké pre všetky adresy, ak pravdepodobnosť detekcie základných porúch ostane nezmenená.
- Vstupné údaje sa nedefinujú pre operáciu čítania.
- Výstupné údaje sa nedefinujú pre operáciu zápisu.

2.4 Obmedzenia

V systéme sa predpokladá niekoľko pevne definovaných obmedzení za účeľom zaručenia správneho fungovania generovanej architektúry: Každá pamäť prislúcha pod jeden blok BIST, bude zadaný maximálny počet prepojení medzi pamäťou a blokom BIST, bude zadaná maximálna dĺžka prepojenia medzi pamäťou a blokom BIST.

2.5 Používateľsky definované obmedzenia

Systém by mal používateľovi poskytovať možnosť voľby nasledovných parametrov (za účelom vyjadrenia preferencií alebo špecifikácie technologických obmedzení): Používateľsky určené zoskupenia pamätí, maximálny počet pamätí prislúchajúcich pod jeden blok BIST, parametre napájania, maximálny čas testovania.

3 Ciele dizertačnej práce

Cieľ dizertačnej práce je automaticky zostaviť architektúru BIST jednak podľa požiadaviek používateľa (zadanie hlavných parametrov), ako aj podľa existujúcich metód testovania pre pokrytie žiadaných porúch v pamätiach. Výsledná architektúra by mala byť opísaná jazykom VHDL alebo Verilog. K dosiahnutiu týchto cieľov je potrebné riešiť nižšie formulované tézy dizertačnej práce:

- Identifikácia jednotlivých blokov v súčasných BIST architektúrach pre 2-D pamäte a ich modifikácie pre ich univerzálnejšie a flexibilnejšie využitie.
- Výber typov pamätí, ktoré budú objektom testovania v SoC, a pre ktoré má byť automaticky navrhnutý optimálny BIST.
- Špecifikovanie požiadaviek, prípadne obmedzení, pre aplikáciu architektúr samočinnej testovateľnosti vnorených pamätí rôznej veľkosti a definovanie parametrov (napr. čas testovania, spotreba energie a pod.) pre meranie efektivity zabezpečenia testovateľnosti všetkých pamätí na čipe testovaných sekvenčne alebo paralelne.
- Návrh novej metódy a univerzálnej architektúry samočinnej testovateľnosti s aplikáciou na vybrané typy pamätí so zabezpečením ich dlhšej životnosti a spoľahlivosti v SoC a flexibilnej pre rôzne architektúry a štruktúry SoC.
- Návrh systému na automatické generovanie špecifickej architektúry BIST pre ľubovoľnú zostavu pamäťových blokov v SoC s požiadavkami na ich testovanie, čo vlastne budú parametre tohto systému. Výstup sa predpokladá vo forme VHDL.
- Implementácia a overenie funkčnosti nového systému pri stanovených požiadavkách a merateľných parametroch pre rôzne zoskupenia pamätí.

4 Záver

Stúpajúca miera integrácie čoraz väčšieho počtu jadier a funkčných blokov na čipe si vyžaduje nové prístupy k riešeniu ich testovania, a to so zameraním sa na testovanie pamätí, ktoré zaberajú najväčšiu časť plochy čipu. Návrh vstavanej architektúry vykonávajúcej samočinné testovanie na vysoké počty pamätí nie triviálna úloha. V tejto oblasti sa po analýze identifikovali otvorené miesta, kde by bol priestor pre vývoj systému automatického generovania testovacej architektúry ponúkajúceho navyše oproti existujúcim prístupom napríklad možnosť voľby testovacieho algoritmu, adresnej schémy, alebo poskytujúceho výstup vo formáte VHDL. Ďalšia práca bude sústredená na špecifikáciu nového systému a vhodných blokov BIST.

Poďakovanie

Táto práca bola podporovaná Agentúrou na podporu výskumu a vývoja na základe zmluvy č. SK-CZ-2013-0173 a národného projektu VEGA 1/1008/12.

Literatúra

[1] International Technology Roadmap for Semiconductors (ITRS). 2011. Test and Test Equipment. http://www.itrs.net/Links/2011ITRS/2011Chapters/2011Test.pdf (2.5.2013)
[2] Kahng, A.B., Kang, I.: Co-Optimization of Memory BIST Grouping, Test Scheduling, and Logic Placement, in IEEE Design, Automation and Test in Europe Conference and Exhibition, 2014, p. 1-6.
[3] Miyazaki, M., Yoneda, T., Fujiwara, H.: A Memory Grouping Method for Sharing Memory BIST Logic, in Asia and South Pacific Conference on Design Automation, 2006, pp. 671-676.
[4] Yu-Jen, H., Che-Wei, Ch., Jin-Fu, L.: A Low-Cost Built-In Self-Test Scheme for an Array of Memories, in IEEE European Test Symposium, 2010, pp. 75-80.
[5] Huang, D.C., Jone, W.B.: A Parallel Transparent BIST Method for Embedded Memory Arrays by Tolerating Redundant Operations, in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2002, pp. 617-628.
[6] Hamdioui, S., Gaydadjiev, G., Van De Goor, A. J.: The State-of-art and Future Trends in Testing Embedded Memories, in Records of the International Workshop on Memory Technology, Design and Testing, 2004, pp. 54-59.
[7] Du, X., Mukherjee, N., Cheng, W.-T., Reddy, S.M.: Full-Speed Field-Programmable Memory BIST Architecture, in Proceedings of IEEE International Test Conference, 2005, pp. 1165-1173.
[8] Bosio, A., Dilillo, L., Girard, P., Pravossoudovitch, S., Virazel, A.: Advanced Test Methods for SRAMs. Springer, 2010, p. 171, ISBN 978-1-4419-0937-4.
[9] Van De Goor, A.J., Kukner, H., Hamdioui, S.: Optimizing Memory BIST Address Generator Implementations, in IEEE International Conference on Design & Technology of Integrated Systems in Nanoscale Era, 2011, p. 6.

FPNN – neuronové sítě v FPGA

Martin Krčma

Informatika a výpočetní technika, první ročník, prezenční studium
Školitel: Zdeněk Kotásek

Fakulta informačních technologií, Vysoké učení technické v Brně
Božetěchova 2, 612 66 Brno
ikrcma@fit.vutbr.cz

Abstract. Tento příspěvek se pojednává o problémech implementace neuronových sítí v hradlových polích, prezentuje koncept FPNA jakožto jedno z možných řešení. Představuje techniku mapování neuronových sítí na FPNN a využití tohoto procesu ke zvýšení odolnosti FPNN proti poruchám. Dále seznamuje s výzkumnou a publikační činností autora a jeho budoucími plány.

Keywords: Neuronové sítě, FPNA, FPNN, FPGA

1 Úvod

Neuronové sítě jsou jedním z významných modelů na poli softcomputingu. Mají schopnost učit se, generalizovat, pamatovat si, což jim dává řadu možných využití, především v klasifikačních, aproximačních, predikčních a kontrolních úlohách. V dnešní době zažívají neuronové sítě vzestup oblíbenosti díky tzv. hlubokým (*deep*) sítím, tedy sítím s větším množstvím skrytých vrstev. Tyto sítě vykazují dobré výsledky v rozpoznávání obrázků a řeči.

Neuronové sítě se potýkají se škálou různých problémů. Jedním z problémů, který mnohé výzkumníky od neuronových sítí odrazuje, je ten, že je obtížné odhalit jak neuronová síť dané konfigurace vlastně pracuje vzhledem k tomu, že si tato síť ukládá naučené stimuly jakožto množinu reálných vah, která je člověkem prakticky nečitelná.

Zaměříme-li se ale na problémy implementační, trpí neuronové sítě především dvěma problémy vycházejícími z jejich samotné podstaty. Těmito problémy jsou časová a prostorová složitost. Tyto problémy vycházejí především z potřeby neuronovou síť učit a také z jejich masivně paralelní struktury.

Těmto problémům čelíme, i pokud chceme implementovat neuronové sítě v FPGA (například kvůli spotřebě energie nebo rozměrům). Problém časové složitosti je možné do určité míry řešit neuronovým sítím i FPGA vlastním paralelismem a řetězením.

Problém prostorové složitosti je možné snížit použitím méně implementačně náročné struktury jako je například FPNA/FPNN, které rozebírá následující část příspěvku.

adfa, p. 1, 2011.

2 Koncept FPNA

Koncept *Field Programmable Neural Arrays* (FPNA) [1] byl navržen s důrazem na zjednodušení implementace neuronových sítí v hradlových polích (FPGA) za pomoci zlepšení jejich vlastností tak, aby více reflektovaly vlastnosti FPGA. Toto zjednodušení vychází z jejich hlavní vlastnosti – vysoce přízpůsobitelné a flexibilní struktury, která umožňuje dosáhnout sdílení zdrojů mezi synaptickými propoji originální neuronové sítě.

2.1 FPNA

FPNA je definováno [1] jako orientovaný graf. Uzly a hrany tohoto grafu representují dva odlišné typy výpočetních jednotek. Uzly jsou nazývány *aktivátory* a representují originální neurony, přičemž provádějí ty samé akce – sběr potenciálu pomocí bázové funkce (u aktivátoru realizováno iteračně) a výpočet aktivační funkce. Hrany jsou nazývány *spoje*, propojují aktivátory a za pomoci afinních operátorů slouží jako aproximace původního synaptického propojení neuronů. Oba typy jednotek jsou dohromady nazývány *neurální zdroje*.

FPNA kvůli absenci dalších detailů nepopisuje konkrétní objekt, ale celou třídu možných objektů. Pro dosažení plně specifikovaného konkrétního objektu tedy potřebujeme něco dalšího. Zbývající specifikaci poskytuje *Field Programmable Neural Network* - FPNN.

2.2 FPNN

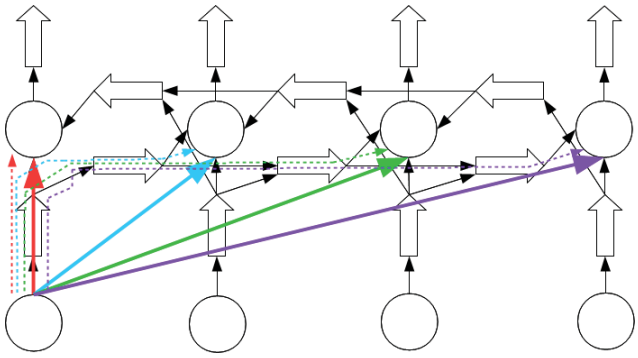
FPNN [1] je jedna z možných instancí FPNA. Definuje konkrétní parametry spojů a aktivátorů. Specifikuje také datové propojení neurálních zdrojů. K tomu využívá čtyř typů binárních příznaků (S, r, s, R) definujících (ne)existenci konkrétního propojení mezi konkrétními dvěma neurálními zdroji (vstup-spoj, spoj-aktivátor, aktivátor-spoj a spoj-spoj).

Nejzajímavější je příznak *R*, který specifikuje propojení mezi dvěma spoji. Toto je klíčová vlastnost FPNN, možnost propojování spojů mezi sebou a konstruování posloupností propojených spojů. To pak umožňuje konstruovat FPNN rozličných struktur a tím pádem i struktur výhodných pro implementaci v FPGA.

Možnost konstruovat různé struktury umožňuje vytvářet různé typy FPNN. Jedním z takových typů je **mřížové** FPNN. Toto FPNN má strukturu ve tvaru mříže, která je utvořena tak, že výstup každého aktivátoru je napojen pouze na jeden spoj a tento spoje je napojen na jeden aktivátor v následující vrstvě. Kromě toho je napojen také na *propojovací sled*, dva proti sobě jdoucí řetězce spojů uvnitř vrstvy, které zajišťují šíření dat mezi všemi aktivátory ve vrstvě (viz. Obr. 1).

V takovém FPNN dochází k výraznému sdílení zdrojů mezi původními synaptickými propoji. Skrze jeden spoj totiž pomyslně prochází jedna a více synapsí, jak ukazuje Obr. 1, kde tlusté šipky znázorňují původní propoje a tenké tečkované šipky pak cestu, kudy toto propojení vede uvnitř FPNN. Toto sdílení šetří zdroje FPGA, navíc výrazně snižuje složitost propojení a tím přináší další úsporu. Další výhodou FPNN

této struktury je, že jeho tvar je podobný tvaru propojovací sběrnice FPGA, což jej činí v FPGA snadněji implementovatelné.



Obr. 1. Mřížové FPNN a ukázka aproximace synaptického propojení.

2.3 Mapování neuronových sítí na FPNN

Jedním z výzev, kterým při práci s FPNN čelíme, je nalezení způsobu, jak na mřížová FPNN namapovat již naučenou neuronovou síť pouze s využitím vah získaných z této sítě, tedy aniž bychom museli učit samotné FPNN. Takový postup je užitečný v případě, že nemáme dostupná trénovací data, nebo nechceme investovat prostředky do učení.

Abychom mohli vyřešit tento problém, je potřeba napřed definovat cílové prostředky FPNN, na které budeme mapovat, tedy afinní operátory spojů:

- Každý spoj *(p,n)* disponuje afinním operátorem pro každý jeden aktivátor *p*, který je propojen s aktivátorem *n* (*x* jsou vstupní data spoje):

$$\alpha_{(p,n)} = W_n(p) \times x + T_n(p); W_n(p), T_n(p), x \in \mathbb{R} \quad (1)$$

Nyní, nechť každý spoj *e* má přiřazenu množinu TC_e, obsahující všechny řetězce spojů končící tímto spojem a začínající napojením na aktivátor předchozí vrstvy. Pro prostřední spoj ve spodní části propojovacího sledu na Obr. 1 by množina TC_e obsahovala dva řetězce – jeden řetězec tvořený třemi spoji (včetně jeho samého) pro aproximaci zelené synapse vycházející z nejlevnějšího aktivátoru, a druhý řetězec pro aproximaci synapse jdou ze sousedního aktivátoru (tvořený dvěma spoji).

Každý řetězec v TC_e množině je zakončen spojem *e*, který je poslední a jehož parametry afinních operátorů závisejí na hodnotách předchozích spojů v řetězci, protože celý řetězec realizuje posloupnost násobení. Je tedy zřejmé, že hodnoty parametrů spojů předcházejících v řetězci spoji *e* je nutno vypočítat dříve, než bude možné počítat spoj *e*. A protože tyto spoje jsou samy konci jiných řetězců, platí výrok i pro ně. Jednotlivé spoje tedy jednak leží v několika různých řetězcích a několik řetězců samy zakončují. Některé řetězce se skládají pouze z jednoho spoje (ty, které vedou od aktivátoru předchozí vrstvy přímo k aktivátoru následující vrstvy) a proto je možné je

vypočítat rovnou a s takto získanými hodnotami začít výpočet o jeden spoj delších řetězců a v každém dalším kroku prodlužovat.

Každý z řetězců aproximuje nějakou synapsi původní sítě. Můžeme napsat, že každá synapse S, končící ve spoji e (tj. aproximovaná řetězcem z TC_e), je aproximovaná hodnotou A_S, která je dána součinem spojů v řetězci:

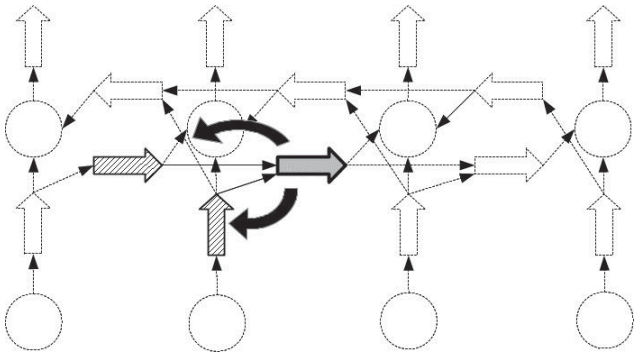
$$A_S = \prod_{x \in TC_e} W_x = \left(\prod_{x \in TC_e - \{e\}} W_x\right) \times W_e \quad (2)$$

Při známé hodnotě výrazu $\prod_{x \in TC_e - \{e\}} W_x$ (což je zajištěno pořadím výpočtu od nejkratších řetězců k dlouhým), je tedy úlohou nalezení hodnoty W_e^S, tedy hodnoty, kterou by mělo nabývat W_e pro přesnou aproximaci synapse S. Při předem dané hodnotě váhy W_S synapse S je způsob výpočtu zřejmý:

$$W_e^S = \frac{W_S}{\prod_{x \in TC_e - \{e\}} W_x} \quad (3)$$

Tímto způsobem jsme tedy schopni pro každou synapsi S vypočítat hodnotu W_e^S, která v součinu s předchozí částí řetězce činí hodnotu A_S, jež by se měla co nejvíce blížit hodnotě W_S. To, jak je tento požadavek splněn závisí na počtu dostupných afinních operátorů ve spoji e. Pokud je jich stejný počet jako synapsí, které spoj aproximuje, tak můžeme každému afinnímu operátoru přiřadit patřičnou hodnotu W_e^S a aproximace bude přesná. Pokud ne, je nutné najít kompromis. Způsobů nalezení kompromisu můžeme použít celou řadu, například aritmetický průměr, medián nebo vážený průměr s rozličnými metodami určení vah. Tyto metody a jejich experimentální výsledky při použití speciálního typu FPNN, v němž každý spoj disponuje pouze jedním afinním operátorem, jsem publikoval na IEEE DDECS [3].

Představený způsob mapování nám, kromě samotného přenosu neuronové sítě na FPNN, nabízí ještě další zajímavou možnost. A to je možnost potenciálně zabezpečit FPNN proti poruchám, aniž bychom museli přidávat mnoho redundance (způsob detekce chyb je v tomto kontextu jiným problémem). S poruchou ale musíme počítat dopředu a zvolit spoje, které chceme takto zabezpečit (metriky určující vhodnost konkrétních spojů k zabezpečení jsou dalším problémem, řešitelným heuristikami založenými například na hodnotách afinních operátorů, poloze v řetězci apod.). Tato metoda předpokládá rozšíření původního modelu takovým způsobem, který by umožnil volitelně učinit libovolný spoj transparentním pro procházející data (tedy přeměnit jej na registr). V případě poruchy by pak bylo možné chybný spoj takto vypojit z FPNN a zabránit tak předem neznámému dopadu této poruchy. Chybějící afinní operátory se pak můžeme pokusit kompenzovat pomocí spojů, které vyloučený spoj v řetězcích předcházejí. Pokud rozdělíme synapse aproximované chybným spojem mezi odpovídající předchůdce v řetězcích, dojde při procesu mapování k započítání těchto synapsí do výsledných afinních operátorů těchto spojů. Zabezpečovaný spoj a zbytek spojů v návazných řetězcích se pak přepočítají, aby reflektovaly změny ve svých předchůdcích. Metodu ilustruje Obr. 2.



Obr. 2. Zabezpečení šedého spoje jeho dvěma předchůdci.

Po tomto zabezpečení pak předchůdci zabezpečeného spoje v případě jeho odstavení částečně kompenzují jeho absenci. S ohledem na počet dostupných afinních operátorů a konkrétní podobu aproximovaných váhových vektorů tato metoda může mít ale i negativní dopad na výslednou přesnost aproximace. Efekt metody je tedy odvislý od konkrétního případu a použitého typu FPNN. V provedených experimentech se prokázala v některých případech schopnost metody posílit odolnost proti poruchám, i negativní dopad v jiných případech.

3 Výzkumná činnost

FPNN jsem zabýval již ve své činnosti předcházející doktorskému studiu [2]. Během ní jsem vytvořil VHDL implementaci FPNN. Tato implementace vykonávala výpočty v pevné řádové čárce a byla parametrizovatelná z ohledu bitových šířek před a za řádovou čárkou. Podporovala několik aktivačních funkcí, přičemž využívala rychlé a nenákladné aproximace. Dalším výsledkem mojí práce je balík aplikací sloužících pro práci s FPNN s názvem **PyFPNN**. Aplikace byly napsány v Pythonu verze 3. Balík se skládá z těchto aplikací:

1. **Generátor FPNN** – aplikace, která dokáže generovat mřížové FPNN pro zadanou dopřednou vrstvenou síť. Výstupem aplikace je textový popis ve formátu používaném všemi aplikacemi balíku.
2. **Generátor VHDL** – aplikace, která pro FPNN zadané textovým popisem dokáže vygenerovat VHDL kód s popisem architektury implementující zadané FPNN. Implementace je založena na vytvořené implementaci neurálních zdrojů. Aplikace dokáže rovněž pro takovou architekturu vygenerovat testbench.
3. **Mapovací aplikace** – aplikace, která dokáže mapovat naučenou neuronovou síť na zadané FPNN odpovídající struktury. Vstupem je textový popis FPNN a soubor s váhami původní neuronové sítě. Výstupem je textový popis namapovaného FPNN. Aplikace podporuje různé metody mapování, které umožňuje kombinovat. Výsledek mapování může být optimalizován pomocí optimalizačního algoritmu.

4. **Simulátor** – aplikace, která umožňuje softwarově simulovat běh FPNN. Vstupem je textový popis FPNN a vstupní data načítaná ze souboru nebo ze standartního vstupu. Výstupem jsou datové vektory snímané z výstupu FPNN. Aplikace má také zvláštní mód, který umožňuje FPNN testovat – tedy srovnávat jeho výstup s referenčním výstupem. Simulaci je možné provádět v číslech s pevnou nebo plovoucí řádovou čárkou. V režimu s pevnou řádovou čárkou je možné výstup simulace použít pro verifikaci VHDL implementace.

Během své výzkumné činnosti v doktorském studiu jsem se věnoval hlavně další práci s FPNN. Moje práce se dosud zaměřovala na experimenty s FPNN, prováděné především pomocí PyFPNN, zaměřené na zkoumání vlivu kompromisů v architektuře, zmiňovaných výše, na sílu FPNN a na hledání metod jejího zvýšení. Dále na vytváření formálních modelů odvozených typů FPNN a konstrukce algoritmů s těmito modely pracujících. Jednalo se především o algoritmy konstruující FPNN daného modelu a pak dále algoritmy mapování neuronových sítí na FPNN.

Vytvořené modely jsem dále rozšířil o podporu prostředků odolnosti proti poruchám. Toto rozšíření umožňuje využívat neurální zdroje zabezpečené redundancí. Zavedl jsem ale také metodu posilující odolnost proti poruchám bez využití redundance.

Na IEEE DDECS jsem rovněž publikoval příspěvek [3], který popisoval principy fungování metod mapování naučených sítí na FPNN (využívajíce některých metod implementovaných v PyFPNN). Příspěvek rovněž prezentoval možnosti kombinování metod a jejich vylepšování pomocí dvou rozšíření. Rovněž byly zahrnuty experimentální výsledky mapování.

V nejbližší době plánuji provést další epxerimenty s vytvořenými modely přizpůsobenými k zajišťování odolnosti proti poruchám a jejich další výzkum a rozvoj. Ve svém dalším výzkumu se hodlám věnovat dalšímu rozšiřování modelů a PyFPNN zejména s ohledem na přizpůsobení různým typům hlubokých neuronových sítí. Následně chci provést experimenty s odolností hlubokých sítí proti poruchám a navrhnout metody jejího zvyšování. Poté chci vytvořit novou architekturu pro implementaci v FPGA a provést experimenty. Podporu této architektury vložím do balíku PyFPNN.

4 Bibliografie

1. Girau, B.: FPNA: Applications and Implementations. In FPGA Implementations of Neural Networks, editation A. R. Omondi; J. C. Rajapakse, Springer US, 2006, ISBN 978-0-387-28487-3, p.81-136, 10.1007/0-387-28487-7-4.
   URL http://dx.doi.org/10.1007/0-387-28487-7-4
2. Martin Krcma: The neural networks acceleration in FPGA, master thesis, Brno, FIT BUT in Brno, 2012
3. KRČMA Martin, KAŠTIL Jan a KOTÁSEK Zdeněk. Mapping trained neural networks to FPNNs. In: IEEE 18th International Symposium on Design and Diagnostics of Electronic Circuits and Systems. Belgrade: IEEE Computer Society, 2015, s. 157-160. ISBN 978-1-4799-6779-7.

Lokalizační systémy pro složky integrovaného záchranného systému

Aleš Kunčar
Inženýrská informatika, 1. ročník, prezenční studium
Školitel: Martin Sysel

Fakulta aplikované informatiky, Univerzita Tomáše Bati ve Zlíně
Nad Stráněmi 4511, 760 05 Zlín

kuncar@fai.utb.cz

Abstrakt. Tato práce pojednává o inerciálních navigačních systémech, jenž jsou využívány pro navigování a snímání polohy ve vnitřních prostorách budov. Systém je vytvořen z cenově dostupných inerciálních senzorů (akcelerometr, gyroskop, magnetometr) od výrobce Seeed a GHI Electronics. Naši snahou je navrhnout vhodné filtry ke zpřesnění určení polohy v místech bez signálu GPS (budovách, tunelech, městské zástavbě, atd.).

Klíčová slova: akcelerometr, gyroskop, inerciální navigační systémy, rozšířený Kalmanův filtr

1 Úvod

Přesné určování polohy a v následném navigování ve vnitřních prostorách budov patří v současné době k důležitým aspektům v průmyslu i v běžném životě. Proto tato oblast přitahuje v posledních několika letech pozornost řadu výzkumných pracovníků a společností zabývajících se navigačními systémy. Nejčastěji používané navigační systémy GPS (Global Positioning System) nelze ve vnitřních prostorách budov využívat vlivem nedostupnosti satelitního signálu, jenž je potřeba k určení aktuální polohy.
V dnešní době již existují navigační systémy, které na základě naměřených hodnot zrychlení, náklonu a rotace ze senzorů (akcelerometr, gyroskop) dokážou vypočítat odhad aktuální polohy s dostatečně velkou přesností. Poněvadž tyto systémy s velkou přesností dosahují vysokých cen, je zde snaha zpřesňovat cenově dostupné senzory za použití dalších senzorů (magnetometr [1], kompas), externích signálů (UWB [2], WLAN [3] nebo RFID), různých filtrů a dalších technik.

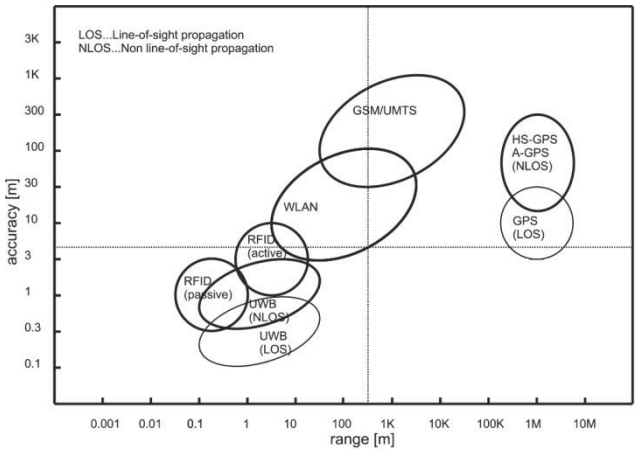


Fig. 1. Přesnost a dosah pozičních systémů

Tento článek pojednává o základních principech inerciálních navigačních systémů a rozšířeném Kalmanově filtru, který je nejčastěji používán k minimalizaci chyb. Dále je popsán aktuální stav, což je analýza senzorů. V poslední části článku je stručně uveden cíl disertační práce.

2 Inerciální navigační systémy

Inerciální navigační systémy (INS) [4] jsou samostatné nebo nezávislé jednotky, neboli využívají techniky, které je činí nezávislé na externích technologických systémech. Za inerciální navigační systémy tedy nelze považovat navigace, jež využívají satelitní systémy GPS, ačkoliv se tyto skupiny mohou mezi sebou kombinovat (spolupracovat).

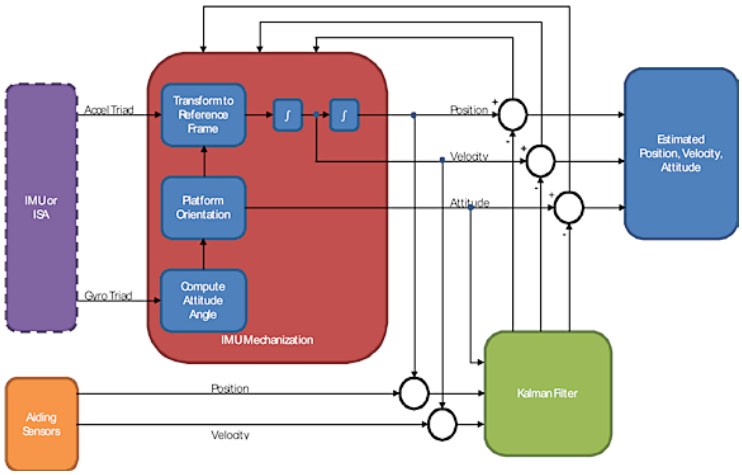


Fig. 2. Blokové schéma INS

Tyto systémy se obvykle skládají z inerciální měřící jednotky (IMU – Inertial Measurement Unit) a řídící jednotky. Řídící jednotka zpracovává a vyhodnocuje naměřené data z pohybových senzorů (akcelerometrů) a sensorů rotace (gyroskop, kompas), které se nachází na měřící jednotce.
Akcelerometry měří okamžité zrychlení a gyroskopy náklon a rotaci pohybujícího se objektu. Z takto naměřených hodnot lze následně vypočítat délka a směr pohybu od výchozí polohy a tím pádem i aktuální polohu objektu.

2.1 Akcelerometry

Akcelerometry [5] patří do skupiny inerciálních senzorů využívajících setrvačnosti hmoty pro měření rozdílu mezi kinetickým a gravitačním zrychlením. V současné době se nejčastěji používají mikro-elektromechanické akcelerometry (MEMS – Micro-Electro-Mechanical System), jenž pracují na principu mechanického kmitavého systému.
MEMS akcelerometry jsou tvořeny destičkou z polykrystalického křemíku tvarovaná do dvou pružných tětiv, ukotvených do monokrystalického křemíkového substrátu. Tuhost *k* mechanického oscilátoru představují tětivy, které jsou spojeny s hřebínkem reprezentující seismickou hmotnost *m*. Každý zub tohoto hřebínku tvoří střední pohyblivou elektrodu X soustavy diferenciálních kapacitních senzorů s proměnným vzduchovým dielektrikem. Jako pevné elektrody slouží systém nosníků Y a Z. Při působení horizontálního zrychlení se zvýší kapacita mezi elektrodami X, Z a poklesne mezi elektrodami X, Y. Při opačném působení zrychlení se kapacita mění obráceně.

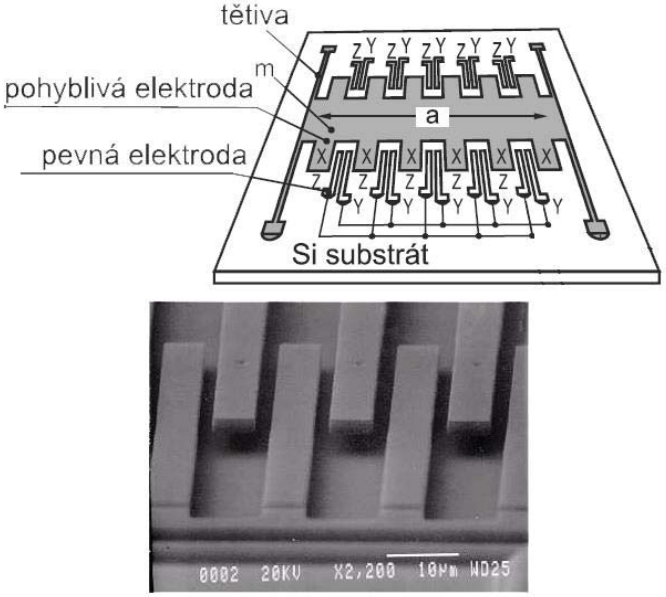


Fig. 3. Struktura MEMS akcelerometru

Existuje několik typů akcelerometrů, které využívají pro měření zrychlení pomocí následujících principů:

- **piezoelektrické akcelerometry** - využívají piezoelektrický krystal, který generuje náboj úměrný síle působící na každý objekt při zrychlení
- **piezoresistivní akcelerometry** - využívají křemíkovou mechanickou strukturu, kde zrychlení odpovídá změně odporu
- **akcelerometry s proměnnou kapacitou** - využívají křemíkovou mechanickou strukturu, kde zrychlení odpovídá změně kapacity

2.2 Gyroskopy

Gyroskopy [5] patřící do stejné skupiny jako akcelerometry měří natočení a otáčení. Nejčastěji používané MEMS gyroskopy pracují na principu Coriolisovy síly. Tyto senzory umějí měřit rotaci pouze vzhledem k jedné ze tří os X (roll axis), Y (pitch axis) a Z (yaw axis), proto se využívají tři ortogonálně umístěné.

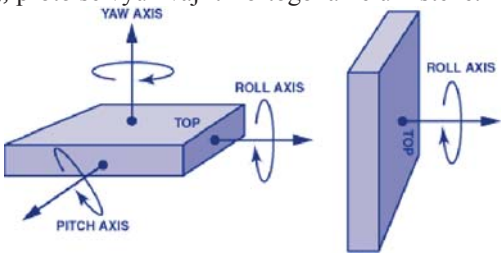


Fig. 4. Osy měření natočení a rotace

Samotný čip neboli senzor je tvořen mechanickými mikrosoučástkami a elektronickými obvody. Rezonující struktura přesně dané hmotnosti, která je upevněna ve vnitřním rámu, se vlivem vlastní mechanické rezonance pohybuje kolmo ke směru otáčení. Tak vznikají Coriolisovy síly stlačující vnější pružiny rámu a způsobí posun měřících plošek (elektrod) ovlivňující velikost kapacity, která je přímo úměrná úhlové rychlosti otáčení.

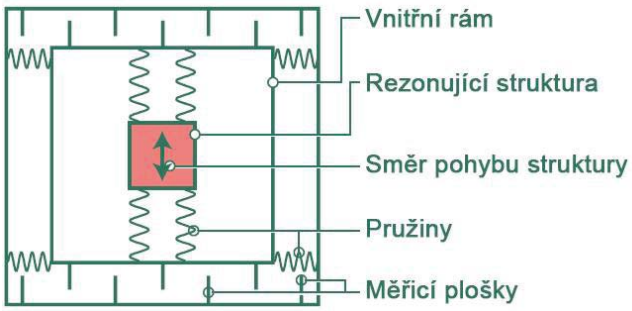


Fig. 5. Struktura MEMS gyroskopu

3 Rozšířený Kalmanův filtr

Pro použití Kalmanova filtru se předpokládá linearita dynamického systému. Existuje však i mnoho procesů modelovaných pomocí nelineárního dynamického systému. Základním principem rozšířeného Kalmanova filtru (EKF - Extended Kalman Filter) [6], [7] je to, že dynamický systém se nejprve linearizuje a následně na takto modifikovaný systém je aplikován obyčejný Kalmanův filtr.
Jednotlivé kroky algoritmu rozšířeného Kalmanova filtru jsou uvedeny v následující tabulce.

Table 1. Kroky rozšířeného Kalmanova filtru algoritmu

Počáteční podmínky	$\mu_{0\|0}$ $\Sigma_{0\|0}$
Jacobiho matice	$A_{t-1} = \frac{\partial g(x_{t-1}, u_{t-1}, w_{t-1})}{\partial x_{t-1}} \| x_{t-1} = \mu_{t-1\|t-1}, u_{t-1} = u_{t-1}, w_{t-1} = 0$ $C_t = \frac{\partial f(x_t, u_t, v_t)}{\partial x_t} \| x_t = \mu_{t\|t-1}, u_t = u_t, v_t = 0$ $F_t = \frac{\partial f(x_t, u_t, v_t)}{\partial v_t} \| x_t = \mu_{t\|t-1}, u_t = u_t, v_t = 0$ $G_{t-1} = \frac{\partial g(x_{t-1}, u_{t-1}, w_{t-1})}{\partial w_{t-1}} \| x_{t-1} = \mu_{t-1\|t-1}, u_{t-1} = u_{t-1}, w_{t-1} = 0$
Predikční krok	$\mu_{t\|t-1} = g(\mu_{t-1\|t-1}, u_{t-1}, 0)$ $\Sigma_{t\|t-1} = A_{t-1}\Sigma_{t-1\|t-1}A'_{t-1} + G_{t-1}W_{t-1}G'_{t-1}$
Datový krok	
inovace	$e_t = y_t - \hat{y}_{t\|t-1} = y_t - f(\mu_{t\|t-1}, u_t, 0)$ $R_{t\|t-1} = C_t\Sigma_{t\|t-1}C'_t + F_tV_tF'_t$
Kalmanův zisk	$K_t = \Sigma_{t\|t-1}C'_tR^{-1}_{t\|t-1}$
Filtrace	$\mu_{t\|t} = \mu_{t\|t-1} + K_te_t$ $\Sigma_{t\|t} = [I - K_tC_t]\Sigma_{t\|t-1}$
Zpětný krok (smoothing)	$\mu_{t-1\|N} = \mu_{t-1\|t-1} + J_{t-1}(\mu_{t\|N} - \mu_{t\|t-1})$ $\Sigma_{t-1\|N} = \Sigma_{t-1\|t-1} + J_{t-1}(\Sigma_{t\|N} - \Sigma_{t\|t-1})J'_{t-1}$ $J_{t-1} = \Sigma_{t-1\|t-1}A'_{t-1}(\Sigma_{t\|t-1})^{-1}$

4 Analýza akcelerometru

Inerciální měřící jednotka je tvořena 8-bitovým MEMS akcelerometrem od společnosti Seeed/GHI Electronics, který je připojen k základní desce FEZ Spider Mainboard a modulu USB Client EDP. Tyto komponenty jsou vhodné pro rychlou tvorbu prototypů a v následném testování.

Fig. 6. MEMS akcelerometr od Seeed/GHI Electronics

Naměřená data jsou následně přenesena pomocí USB kabelu do počítače, kde je možné tyto data dále analyzovat.
Takto sestavený obvod byl využit v experimentu, jenž byl rozdělen do dvou částí. V první části byl senzor položen na rovné podložce vždy kolmo k jednotlivým osám, aby mohlo být změřeno gravitační zrychlení. Z takto naměřeného gravitačního zrychlení byla zjištěna hodnota přesná hodnota 1g vůči jednotlivým osám. Tímto způsobem bylo docíleno přesnějšího výsledku měření.
Druhá část se zabývala měřením předem dané dráhy o délce 34 cm. Naměřená data byly následně analyzovány v programu Microsoft Excel.

Table 2. Naměřená a vypočtená data

Měření č.	Naměřená dráha	Chyba
1.	43,52 cm	21,87 %
2.	39,11 cm	13,07 %
3.	38,23 cm	11,06 %
4.	41,97 cm	18,99 %
5.	37,35 cm	8,97 %
6.	40,2 cm	15,42 %
7.	41,73 cm	18,52 %
8.	38,37 cm	11,39 %
9.	38,55 cm	11,8 %
10.	40,19 cm	15,4 %

Z naměřených a vypočtených výsledků je patrné, že tento 8-bitový MEMS akcelerometr není dostatečně přesný i při tak krátkých časových intervalech měření, které nepřekračovaly 5 vteřin. I přes tyto nepřesnosti v měření, které byly způsobeny nerovnostmi na vytyčené dráze, jenž způsobovaly na naměřených datech rychlé a signifikantní nárůsty akcelerace, budou senzory nahrazeny 16-bitovými. Největší slabinou použitých senzorů byl parametr Output Data Rate (ODR – frekvence vzorkování dat), který dosahoval hodnoty 50 Hz, což není vhodně pro budoucí aplikace.

5 Závěr

V tomto článku byl představen směr disertační práce, který tkví v návrhu lokalizačního systému pracujícího ve vnitřních prostorách budov, tunelech a městské zástavbě sestávajícího se pouze z cenově dostupných inerciálních senzorů (akcelerometr, gyroskop, magnetometr, …). V důsledku cenové dostupnosti jsou senzory méně přesné a bude je nutné doplnit povětšinou matematickým aparátem (filtrem) pro minimalizaci chyb. Další metodou, která bude předmětem budoucích experimentů, bude snaha minimalizovat chybu přidáním většího počtu inerciálních senzorů, případně měřením charakteristik lidské chůze a následném využití metody Zero Velocity Update (ZUPT). Výsledkem tohoto výzkumu bude lokalizační systém vhodný pro složky integrovaného záchranného systému.

Poděkování

Tento článek vznikl za podpory projektu Interní grantové agentury, IGA/FAI/2015/012.

Reference

1. YUN, Xiaoping, Eric R. BACHMANN, Hyatt MOORE a James CALUSDIAN. Self-contained Position Tracking of Human Movement Using Small Inertial/Magnetic Sensor Modules. In: *Proceedings 2007 IEEE International Conference on Robotics and Automation* [online]. 2007 [cit. 2015-06-17]. DOI: 10.1109/robot.2007.363845.
2. HARMER, D., YAROVOY, A., SCHMIDT, N., WITRISAL, K., RUSSELL, M., FRAZER, E., BAUGE, T., INGRAM, S., NEZIROVIC, A., LO, A., XIA, L., KULL, B., DIZDAREVIC, V., "An ultra-wide band indoor personnel tracking system for emergency situations (Europcom)," Radar Conference, 2008. EuRAD 2008. pp.404-407, 30-31 Oct. 2008
3. LEPPAKOSKI, H., J. COLLIN a J. TAKALA. Pedestrian navigation based on inertial sensors, indoor map, and WLAN signals. In: *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* [online]. 2012 [cit. 2015-06-17]. DOI: 10.1109/icassp.2012.6288192.
4. WOODMAN, Oliver. *An introduction to inertial navigation* [online]. 2007 [cit. 2015-06-18]. ISSN 1476-2986. Dostupné z: https://www.cl.cam.ac.uk/techreports/UCAM-CL-TR-696.pdf
5. *Měření otáček, rychlosti a zrychlení* [online]. [cit. 2015-06-18]. Dostupné z: http://measure.feld.cvut.cz/system/files/files/cs/vyuka/predmety/A3B38SME/08_rychlost%20zrychleni%20vibrace%20TEXT.pdf
6. MRÁZ, Jaroslav. *Diplomová práce* [online]. Plzeň, 2012 [cit. 2015-06-19]. Dostupné z: https://otik.uk.zcu.cz/bitstream/handle/11025/2649/DP-Mraz.pdf?sequence=1. Diplomová práce. Západočeská univerzita v Plzni.
7. STEHNOVÁ, Martina. *Metody filtrace a jejich využití v ekonomii* [online]. Brno, 2012 [cit. 2015-06-19]. Dostupné z: http://is.muni.cz/th/269500/prif_m/DP.pdf. Diplomová práce. Masarykova univerzita.

SAFETY OF COMMUNICATION AND AUTHENTICATION IN DATA WAREHOUSE FOR REMOTE LABORATORIES AND LABORATORY MANAGEMENT SYSTEM

L. Pálka[1], *F. Schauer*[1,2]

[1]Tomas Bata University in Zlín, Faculty of Applied Informatics, Nad Stráněmi 4511, CZ-760 05 Zlín, Czech Republic. E-mail: lukas.palka@email.cz
[2]Faculty of Education, Trnava University in Trnava, SK-918 43 Trnava, Slovak Republic

ABSTRACT

In spite of the fact that remote laboratories have been existing for at least three decades, virtually no attention has been devoted to the security of this new subject. The paper deals with the security of the data storage of the Data Center - RemLabNet (DTC), with remote laboratories working under the Laboratory Management System (LMS). Especially, the communication risks for the data storage and corresponding data processing to ensure the operation of the data warehouse are described in detail.

Index Terms— security data, data communication, data storage, rig, RemLabNet, remote experiments, data warehouse, trust authentication, C2 auditing

1. INTRODUCTION - REMOTE LABORATORIES AND LABORATORY MANAGEMENT SYSTEMS - STATE OF THE ART

At the present stage of the development of Information Communication Technologies (ICT) there are plenty simulations and remote experiments for science and education purposes [1][3][9]. Remote experiments and informatics resources are tools that are closely related and definitely need to process and store substantial amounts of data. Data, used with remote laboratories (RL), may have the form of simple queries, data analysis, comparative analysis and data mining for associative analysis, extrapolation or predictive trend analysis. Surprisingly, in spite of the fact the RL have been existing for at least three decades [1], virtually no attention has been devoted to the security of this new ICT subject [8].

The present paper deals with the communication and safeguarding of the data processed and especially stored in the DTC with remote laboratories, especially that with Laboratory Management System (LMS).

The layout of the paper is following. In Chapter 1, the typical scheme of the communication of a typical remote experiment (RE) , built as the finite-state machine (FSM) [2], using the Internet School Experimental System (ISES) physical hardware, is described[10] (for the ease of reading we will next denote the set of the individual remote experiment by the word rig). Also, the control program compiling and the type of the data generated and transferred is shortly described. More details may be found in corresponding literature [2][8]
The Chapter 2 is devoted to describing the architecture Remote Experiments and the corresponding integrating management system, called for our purposes Remote Laboratory Management System (RLMS) [10]. The Chapter 3 is devoted to the actual communication design and database scheme of remote laboratories. The Chapter 4 is then focused on the communication risks access of remote laboratories attack of a typical DW of a university datacenter (DTC) with LMS for remote laboratories. The final chapter 5 is oriented on communication risks access of remote laboratories based defense with Trust Calculation, followed by conclusions.

2. ISES REMOTE EXPERIMENT (RE) AND REMOTE LABORATORY MANAGEMENT SYSTEM (RLMS) – TOOLS USED

Only recently has emerged a serious problem stemming from analysis of research data. ISES is a powerful tool for process and experiments control, acquisition, collecting and data processing in real time. Let us mention the basic features of the ISES system, more detailed description may be found elsewhere [2][10]. The basis of the system is ISES board, which is available in several versions, differing depending on the number of inputs/outputs and also on type of communication with the control PC (by PCI card, USB connector, Wi-Fi). To this board are, by a unique connector, plugged in sensors like: ammeter, voltmeter, thermometer, position sensor, ohmmeter, load cell, anemometer,

microphones, sonar, light gate, pH meter, conductivity meter, heart rate monitor, etc. [8]. The layout arrangement of the RE is in Figure 1.

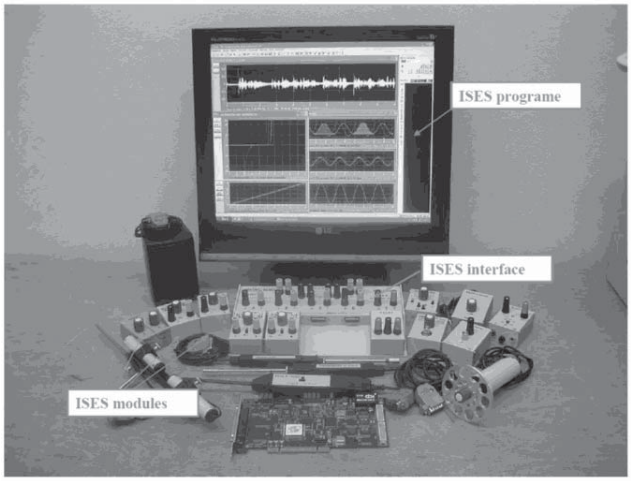


Fig. 1. ISES – Internet School Experimental System

The most important component is the Measureserver module, functioning as finite-state machine (FSM) controlled by the controlling program of the PSC script file. The main feature of the Measureserver, is to communicate with the physical hardware and to check the setup of the ISES panel and its sensors/meters and to take care about their data collection and processing. Other parts of the system are ImageServer for life view of the remote experiment, Web server for the communication between RE and the client. Also, aprt of the RE is the communication web page as the interface communicating with the RE over the Internet by the client.

The inevitable part of the RE system is the data warehouse for the storage of data for all above systems. It is a centralized repository service to Measureserver, web server, image server and other components of the solution.

In this article we will discuss this last part of the system with respect of data security, but not only from the perspective a single RE, but of the whole RLMS. The layout arrangement of the RE is in Figure 2

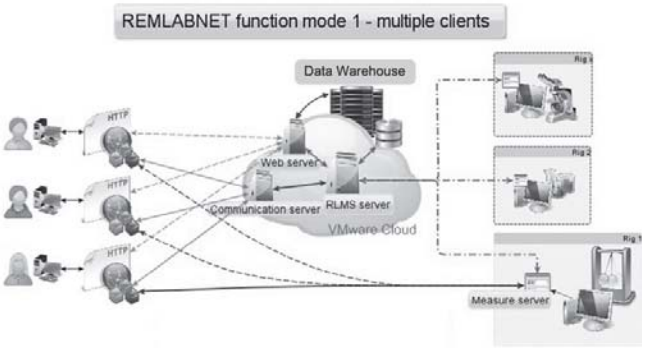


Fig. 2. REMLABNET function mode 1 – multiple clients

A serious problem stemming from security aspects of e-laboratories has emerged only recently[19]. Let us describe first the data generated and that are processed in every ISES rig working on the communication principle server-client and the functioning of the superordinate Remote Laboratory management system (RLMS). The controlling of every rig by client is enabled via Web interface, by means of which the user can perform the appropriate settings, options, and starting or stopping of the remote experiment (RE).

The measured data from the experiment delivered from the MeasureServer are stored in the data storage . RLMS is a system for a database-driven Web application. This is seen from the figure 2, where LMS is divided terms of safety in three parts. Database-driven Web applications are very common in today's Web-enabled society. LMS consist of a back-end database with Web pages that contain server-side script written in a programming language that is capable of extracting specific information from a database depending on various dynamic interactions with the user.

Remote experiments problematic is the topic of scientific activities of the group since 2005, when the first remote experiment started to be built. We relied on the enormous know-how of Assoc. Prof. F. Lustig from Department of Physics Education of Faculty of Mathematics and Physics, Charles University in Prague, where the universal and very useful modular computer oriented set Internet School Experimental System (ISES) was designed at the beginning of 90th [Lustig, F.: "Počítačem podporované školní experimenty s měřícím systémem ISES pod Windows", In: sborník MEDACTA 97 - vzdělávanie v meniacom sa svete, 404-408, Ústav didaktickej technológie, PF UKF v Nitre, Nitra, ISBN 80-967339-9-0, 1997 and Lustig, F. and Schauer, F.: "Creative laboratory experiments for basic physics using computer data collection and evaluation exemplified on the ISES", Proceedings first european conference on Physics

Teaching in Engineering Education, 125-131, Copenhagen, Denmark, ed. Oehlenschlaeger, 1997].

3. COMMUNICATION DESIGN AND DATABASE SCHEME OF REMOTE LABORATORIES

The goal of the setup of the standardized data model is to ensure that data will not be duplicated in DB and also guarantee the simplest form of data consistency. It is therefore necessary to accept a flexible data model design in order to associate the data into various contexts without increase of redundancy [14].

The first three types of standardized forms, which are generally followed in the data model, called the first normal form (1NF), second normal form (2NF) and third normal form (3NF).

1NF - The first normal form eliminates repeating of groups and requires that each line has its own and unique identifier or key [14]

2NF - The second normal form must meet the requirements of 1NF and exclude the partial key dependences due to the location of fields within the table itself independently of fields that depend on the entire key [14],

3NF - The third normal form must satisfy the requirements of 1NF and 2NF and in addition must eliminate the dependence of non-core fields due to their location in a separate table. At this stage, all the single non-key fields depend on the key [14].

Already in the design of the data model the safety of data must be considered. The data model should eliminate duplication model and be clear and well describable. The attributes of each item of the table must clearly state, which data will be saved in the given field. The designer of the model should stick to established methods and proceed systematically, where each table should have its own meaning, its connection to another table and it should posses clear indexation. As a whole the data model should operate lucidly, hierarchically and logically. Design of a data model for remote laboratories will be described in subsequent work on the data warehouse for remote laboratories design.

4. COMMUNICATION RISKS ACCESS OF REMOTE LABORATORIES ATTACK

Remote laboratories and their remarkable development all over the world and forcing our system is already integrated with friendly remote laboratories. In particular, the Sahara project led by the University of Technology Sydney (UTS). Labshare's mission is to create a nationally shared network of remote laboratories in the context of Australian education institutions, in order to address the issues of underutilization laboratory, accessibility, flexibility and foster the availability of high-quality experiments. It has developed both Supporting software systems as well as an organizational model That will Encourage and support cross-institutional sharing.
It is a joint initiative of the Australian Technology Network (ATN) partners: UTS; Curtin University of Australia; University of South Australia (UniSA); Royal Melbourne Institute of Technology (RMIT); and Queensland University of Technology (QUT).[3]

There is space for close cooperation with laboratories constructed in the Czech Republic and especially in the remote laboratory project Easy remote ISES. These laboratories should communicate, migrate, integrate, share data and be able to offer their services to students around the world.

These difficulties have you see many problems, especially with virtualization of their services to the cloud, authentication method sand credibility. In the next chapter I will describe what to expect due to the integration of services, especially data storage and the associated authentication.

Today the future of data warehousing by definition will probably conceived together with technologies such as clustering, geo-clustering, virtualization or cloud. The future is now much relies on the cloud and in conjunction with a data warehouse brings many difficulties to data security. In our case it is not otherwise a data warehouse delivers services to the cloud.

In the cloud, we must calculate with the fact that our data will travel. Travel from one server to another. Data security is therefore as secure as the weakest link in the infrastructure. This Article may be not updated server and its services, lack of monitoring people's access to the servers and the like.

The most common risks of operating a data warehouse in the cloud can be considered as data theft, data manipulation, authentication threats, malicious software and attacks from his own people.

Data Theft in the cloud in the data warehouse great threat and should never be neglected safety data and should take it into account already in implementation. In remote laboratories working with very sensitive data, and particularly data manipulation can cause a number of problems, as well as protection reports and knowledge base. The topic of data theft is therefore systematically deal with the authentication methods, set permissions, auditing to

encryption. These methods should be combined, and today is not always fully used.

Data manipulation is based on the issue of data theft. UC-purpose handling, but do not underestimate. In the future, with us this problem more and more touch and. A service that is unavailable and no offense to destroy the data physically by fire etc. already nowadays easily solves backup technology. Before working with the data, however, is not clear counter tool. Data that are not provable and ou can not go back to the point where the data is wrong, it can lead in many cases to the total degradation of data warehouse and data declarations for worthless. Data manipulation in the future is a big threat and broken new ground. And no tools are not the only tool in the data warehouse is em auditing and transactions processing the data. However, this technology is not the long-term analysis and the possibility of returning the consistency of the data.

Authentication threats in the cloud threatens abused another account and again manipulation, feeding false data warehouse and other threats. Due to the vision of information technology is a threat of abuse login using the newly introduced terminology in this work and cloning of authenticity. This term, I would like to express method, abuse authentication using other systems that also requires authentication. Virtually in the world today, each user has multiple accounts and now it is a problem in any system remember another password and even better a different username. Today and in the future, everything points to use a single authentication and the approaches call them userID, is repeated in other systems. Defense is only one and it HW means for authentication. Cards, chips, tokens and discipline users.

Malicious software in cloud. If you use cloud storage to share data with other users, the virtual server can become a promoter of computer threats. Location convenient to exchange data and documents can be dragged into your computer infected and lose all your data. Some computer threats can also steal passwords and other sensitive data by which the attackers can then hack into your cloud storage.

Attack from the inside - cloud services, which lost management settings can be threatened from within a person with administrator privileges. For systems that are dependent on cloud providers, it is a very serious risk. Even if the content is encrypted cloud, the system is vulnerable to an attack led by a man with administrator access and the ability to manipulate data.

5. COMMUNICATION RISKS ACCESS OF REMOTE LABORATORIES BASED DEFENSE

Securing access to cloud-based services like remote laboratories created by UTB Zlin presents challenges that aren't easily addressed using conventional security controls. In cloud environments, systems and their data store in database are virtualized and may migrate dynamically to different network locations. This makes it difficult to effectively restrict access using traditional security controls such as firewalls, which rely on fixed locations of systems and a more static nature of the data. We need much more granular and dynamic controls that are linked to the resources themselves rather than just their network location.

To meet these rapidly changing in cloud environments, systems we need a highly flexible and dynamic architecture. The architecture should enable us to more quickly adopt new devices, use models, and capabilities; provide security across an increasingly complex environment; and adapt to a changing threat landscape.

The new architecture replaces this with a dynamic, multitiered trust model that exercises more fine-grained control over identity and access control, including access to specific resources. This means that for an individual user, the level of access provided may vary dynamically over time, depending on a variety of factors—such as whether the user is accessing the network from a highly secure managed device or an untrusted unmanaged device.

The architecture is based on four cornerstones: [18]
Trust Calculation: dynamically determining whether a user should be granted access to specific resources like run rigs etc.

Security Zones: the infrastructure is divided into multiple security zones that provide different levels of protection.

Balanced Controls: to increase flexibility and the ability to recover from a successful attack

User and Data Perimeters: this means an increased focus on user awareness as well as data protection built into the information assets.

Trust Calculation
The trust calculation plays an essential role in providing the flexibility required to support a rapidly expanding number of devices and usage models. The calculation enables us to dynamically adjust users' levels of access, depending on factors such destination score, taking into account the controls available to mitigate risk. As shown in Figure 3, the

result of this calculation determines whether the user is allowed access and the type of access provided.

Source Score Trust in the source, or requestor, is calculated based on the following factors:

Who: The identity of the user or service requesting access
What: The device type
Where: The user's or service's location. For example, a user who is inside the school MAN network for remote laboratories is more trusted than the same user connecting through a public network.

Destination Score This is calculated based on the same three factors, but these are considered from the perspective of the destination the information the source is trying to access.

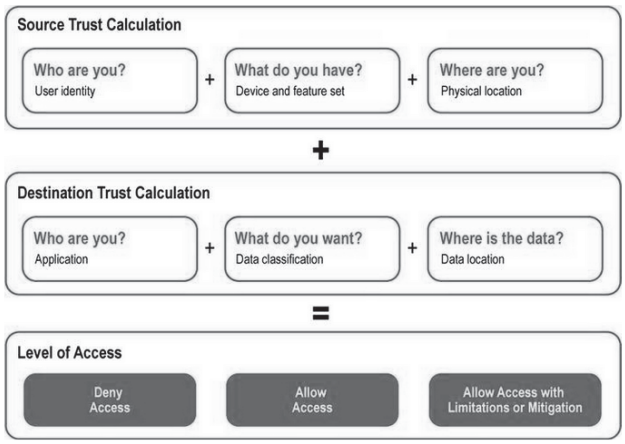


Fig. 3. Trust calculation. Source: Intel Corporation, 2012

Calculating Trust
The trust calculation adds the source score and the destination score to arrive at an initial trust level. The available controls are then considered to make a final decision about whether access is allowed and, if so, how. This calculation is performed by a logical entity called a policy decision point (PDP), which is part of the authentication infrastructure and makes access control decisions based on a set of policies.

If we are unable to verify the location of a high-security device such as a managed PC, we would allow less access.

6. REPORTS – VIEW OF DATABASE TOOL BASED COMMUNICATION DEFENSE

We may call the inspection of data by viewing as a function – view. Remote experiments will be used by students and each connection to the rig can be formed login file, describing his/her activity during connection in a form of a report, susceptible to inspection, without resorting to any comment or evaluation that can be published. View is then the first step to the data protection and thus security.

Authentication and Auditing of database tool based communication defense
In order to make the data warehouse accessible to the end user he/she has to obtain access rights to its own system and successively to the data warehouse.

Database systems contain system tables that store user information, monitoring login / logout, store information about the user´s activities i.e. with tables the user processed, which operations carried out and also information about SQL queries used. The system administrator can then control the amount of information stored in the system tables by setting the system audit. In the case of full system audit we have to keep in mind the number of users can result in huge system tables, so it is immediately necessary to set rules for their archiving or the data deleting from tables in a pre defined cycle. In our case the remote laboratories ISES approaches to the data warehouse will be set up using SQL Server Security services. Auditing of the data warehouse will be performed using C2 auditing capability, which is a built-in functionality of the SQL Server, described bellow.

C2 auditing servers for
- Detection of abuse of accounts and corresponding preventing mechanisms
- Inspection and policies to prevent abuse (watch to apply strong passwords)
- Mechanisms to detect and correct a potential weak spots of the system

Auditing needs should be considered from the very beginning, cataloguing data and manipulate data with properly.
The remote laboratory data should be considered from two points of view: the crucial data of the experiment´s results and the data that are stored, but of inferior importance for the result of measurements. Data of experimental results, evaluation of experiments, records and analyzes are very important for the measurement process. On the other hand temporary data from ongoing trials are unimportant and do not require logging, when accessing them. Data security is important and has to be taken in account system analysis and when compiling the DB environment, ie. the structure and design of tables, diagrams and view of data.

Authentication and Authorization Settings of database tool based communication defense
Authentication, or if a user authentication is an authentication mode and one of the key issues in security SQL Server and WEB server.

To enable the end users to work with data warehouse, we give them access rights. These can be defined at three levels, namely: the operating system level, database level and the level of the database tables. Most users will read data only by an end tool. In our case using a web server and a web site, communicating with a web server. Web server, from the security point of view is understood as an application server, whom it is necessary to secure for the security package by including in to the infrastructure according to the site DMZ rules communications and set firewall settings.

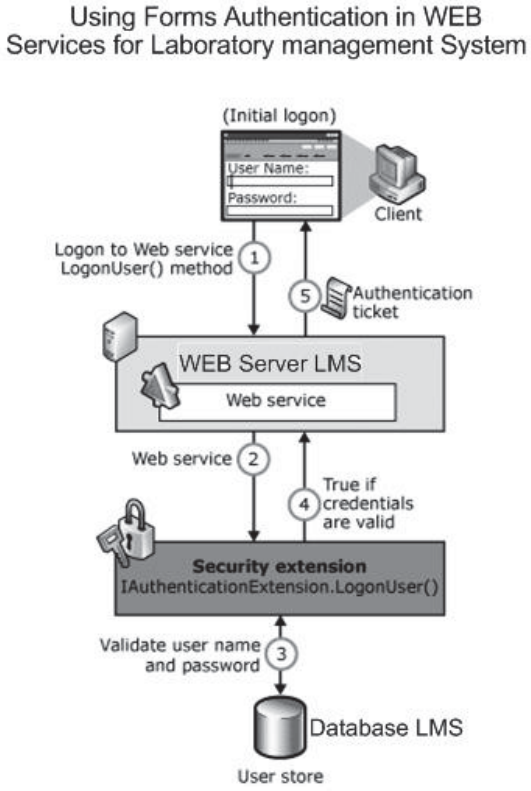


Fig. 4. Schematic representation authentication client

Data Encryption of database tool based communication defense

A much more sophisticated approach to data security in data storage is data encryption. Data encryption can be done at the level of the whole table, one record or one column in the database table. When choosing the proper encrypting we have to keep in mind circumstances as the existence of primary and foreign keys that may breach the integrity of the database during encrypting process, leading to the deleting the links between tables. This is seen from the figure 4.

Data encryption this has a ill-defined effect on data manipulation, leading to the impossibility of data manipulation with standard SQL. The golden rule her is to use the encryption for remote laboratory results data only by teacher and authentication into the system. On the other hand we have to realize the ill effect of encryption on CPU, as coding and decoding of data is causing the increase of the system response in order of milliseconds to seconds.
The feature will protect us that there will be theft or physical storage of the discs, disc images, and the like. Before breaking passwords that control is ineffective. Encryption can also be paired with certificates that can be stored on removable media (tokens or cards).

7. CONCLUSIONS

This work describes a series of recommendations and procedures to secure data storage in the scheme of the data warehouse for the needs of remote laboratories. The work includes a vision for the future regarding security and direction of data warehouses, aiming to direct readers to the problems of data warehouses from all perspectives and to learn, what are the risks of today and how to comprehend security. In the course of the work on the data warehouse we learned how to realize a safety problem as well as design security. We believe, that the article sheds some light on a number of acute problems but simultaneously opened to us many other questions to consider in connection with data warehouse security. The article also describes the introduction of new terms on issues of security and shares experience in terms of theory and our practical experience.

Acknowledgments

The paper was published thanks to the Grant of the Internal Agency of UTB No IGA/FAI/2015/. One of us acknowledges the partial support of the Slovak Research and Development Agency, project no. APVV-0096-11, the Scientific Grant Agency VEGA, project no. 2/0157/12, and the KEGA Agency projects No 011TTU-4/2012 and 020TTU-4/2013.

8. RELATION TO PRIOR WORK

The work presented here has focused on the remote experiments and systems that ensure the operation. The Remote Laboratory Management System (RLMS), REMLABNET, for the integrating and management of remote experiments for starting university level and secondary schools is presented. Its building was initiated both from the extensive use and expertise in Internet School Experimental System (ISES) and remote experiments built and the lack of a similar system for secondary schools in Europe. RLMS is built using new components, designed for the purpose, as Measureserver, web space management, data warehouse, communication board of RLMS, etc. The communication server will provide beside connection and diagnostics services also services for the teacher´s comfort as white board, IP telephony, simulation inclusion, test management

and reservation management. For the sake of safety, optimal access to all experiments and economical exploitation the virtualized cloud will be used.

9. REFERENCES

[1] The whole system is detail described in the project proposal Submitted Project Grant Agency of the Czech Republic: INFORMATICS MEANS FOR GRID OF e-LABORATORIES – PROJECT REMLABNET, 2013.

[2] KRBEČEK, Michal. Possible utilization of the artificial intelligence elements in the creation of remote experiments. [online]. 2012, č. 1 [cit. 2013-06-26]

[3] Grid Remote Laboratory Management System. Sahara Reaches Europe. 2013, č. 1

[4] Database-Level Roles [online]. 2012 [cit. 2013-06-26], http://msdn.microsoft.com/en-us/library/ms189121.aspx

[5] ALEXANDER, David, Amanda FINCH, David SUTTON a Andy TAYLOR. Information Security Management Principles. 2. vyd. bcs, 2013. ISBN 9781780171753

[6] SCHULZ. Cloud and Virtual Data Storage Networking. teChapterChapterbooks, 2011. ISBN 978-1439851739

[7] Data Warehouse [online]. 2013 [cit. 2013-06-26], http://en.wikipedia.org/wiki/Data_warehouse

[8] SCHAUER, František, František LUSTIG a Miroslava OŽVOLDOVÁ. Innovations 2011: World Innovations in Engineering Education and Research: Internet Natural Science Remote e-Laboratory (INTRE-L) for Remote Experiments. USA: iNEER, 2011, s. 51-68. 1. ISBN 978-0-9818868-2-4.

[9] SCHAUER, František a Miroslava OŽVOLDOVÁ. Plug and play system for hands on and remote laboratories. In: Proceedings of 8th International Conference on Hands-on Science. Ljubljana: University of Ljubljana, 2011, s. 17-21. ISBN 978-989-95095-7-3

[10] KRBEČEK Michal, František SCHAUER, Roman JAŠEK. Security aspects of remote e-laboratories. Zlín: UTB ve Zlíně, Fakulta aplikované informatiky, 2012

[11] PÁLKA Lukáš,Data Warehouse services [online]. 2013 [cit. 2013-06-26], http://datawarehouse.cz/Data_warehouse

[12] Data Warehouse [online]. 2013 [cit. 2013-06-25], http://www.1keydata.com/datawarehousing/datawarehouse.html

[13] PÁLKA Lukáš, Methods and Tools Related to Data Security and the Protection of Microsoft SQL Servers. Zlín UTB, 2012

[14] LABERGE, Robert. The Data Warehouse Mentor: Practical Data Warehouse and Business Intelligence Insights. -: 2011. ISBN-10: 0071745327

[15] CLARKE, Justin. SQL Injection Attacks and Defense. USA: Elsevier, 2012. ISBN 978-1-59749-963-7.

[16] GERŽA Michal, František SCHAUER, Roman JAŠEK. Security of ISES MeasureServer© module for remote experiments against malign attacks, Zlín: UTB ve Zlíně, Fakulta aplikované informatiky, 2013.

[17] BRUCHEZ, Rudi. Microsoft SQL Server 2012 Security Cookbook. UK: Packt Publishing, 2012. ISBN ISBN 978-1-84968-588-7

[18] HARKINS, Malcolm. Managing Risk and Information Security: Protect to Enable. LLC: Apress Media, 2013. ISBN 978-1430251132

[19] Paper REV 2014 - 31102013 – F. Schauer 2014

[20] T. Dulík, M. Bližňák, "Security measures in virtual laboratory of microprocessor technology," DAAAM International Vienna, Proceedings of the 21st International DAAAM Symposium "Intelligent Manufacturing & Automation: Focus on Interdisciplinary Solutions", Vienna, 2010, pp. 1203-1204, ISBNISSN 978-3-901509-73-5

[20] D. Lowe, P, Newcombe and B. Stumpers, "Evaluation of the Use of Remote Laboratories for Secondary," Res. Sci. Educ/Science Education, Springer Science and Business Media B.V. 2012, DOI 10.1007/s11165-012-9304-3.

Nonmetallic-carbon nanotube ”buckypaper” networks applied on plastic substrates as a passive antenna construction and gas sensor

JIRI MATYAS[1,2a], ROBERT OLEJNIK[2,b], KAREL VLCEK[1,c], PETR SLOBODIAN[2,d]

[1]Faculty of Applied Informatics, Department of Computer and Communication Systems, Tomas Bata University in Zlin, Nad Stranemi 4511, 760 05 Zlin, Czech Republic
[2]Centre of Polymer Systems, University Institute, Tomas Bata University in Zlin, trida Tomase Bati 5678, 760 01 Zlin, Czech Republic

[a] matyas@fai.utb.cz, [b] olejnik@cps.utb.cz, [c] vlcek@fai.utb.cz, [d] slobodian@cps.utb.cz,
http://www.utb.cz

Abstract: Carbon nanotubes were used in the form of entangled networks. This material was employed for construction of a passive antenna. The networks were made by a filtration method through a polyurethane membrane. The non-metallic carbon material was adjusted to the required shape in order to ensure the best parameters. The prepared prototype of the experimental antenna was tested by measurement of its parameters in the anechoic chamber. The sample was also simulated and compare with a model before preparing. The prepared antenna seems to be well adapted for 1.284 GHz. Carbon nanotubes (CNT) can be used in particular applications that employ microstrip antennas. In addition, there is the possibility of their integration to electrically nonconductive surfaces where traditional materials, such as copper, for various reasons, cannot be used. Carbon nanotube networks was also tested for gas sensitivity. The network shows good sensitivity and, after oxidation treatment, the sensitivity is increased due to formation of carboxylic and carbonilic group on the carbon nanotube walls.

Key-Words: - carbon nanotubes, microstrip antenna,buckypaper, polymethylmetacrylate (PMMA),

1 Introduction

In many fields of science, carbon nanotubes are one of the most promising materials. Gradually, there are increasing possibilities for the use of carbon nanotubes in electronics. The main reasons for employing these materials are their characteristics, which are, for instance, good conductivity (after complex adjustments), or great potential for miniaturization. As the composite materials can perform several functions simultaneously, they can be applied to various products and surfaces not only in the ICT field [5]. This contribution is focused on the use of carbon nanotubes in the passive antenna, which consists of a layer of carbon nanotubes applied to polymethyl methacrylate (PMMA). The antenna described in this article operates in the 1.284 GHz bandwidth.

When preparing “buckypaper “ it is impossible to determine in advance the resistivity value of the carbon layer. Nevertheless, the aim of the research is to produce the layer with the lowest possible resistivity. The low resistivity ensures a better suitability for the material for the construction of the antenna. This characteristic can be influenced by the use of specific chemicals and by the type of carbon nanotubes. The produced specimen of the antenna employs multi-walled carbon nanotubes (MWCNT). The main reason for choosing MWCNT is the fact that the composite does not have to be perfectly pure; for the pure composite it would be more appropriate to use the single-walled carbon nanotubes (SWCNT). In addition, the final price of the subsequent applications to the devices plays an important role. Therefore, the price is an important criterion. From the perspective of the layer resistivity, better results of some measurements would be obtained if single-walled carbon nanotubes (SWCNT) were used [6]. However, this would lead to the price of the final product being much higher than in the case of multi-walled carbon nanotubes (MWCNT).

2 Experimental

At present, the most frequently used and well-known electrically conductive materials for the construction of passive antennas are as follows: copper, gold, silver, aluminum, and stainless steel.

These materials are well described and therefore, based on simulations of their behavior within the various constructions, their performance as passive antennas can be estimated. The carbon nanotubes are a new and quite attractive material. It can be used for constructing antennas in general and also for those operating at higher GHz frequencies. However, the main problem related to carbon nanotubes in antennas is too high of a resistivity value of the carbon layer, which is coated onto the polymethyl methacrylate substrate (PMMA).

The antenna was subjected to measurements in a professional anechoic chamber. Due to the use of an anechoic chamber very accurate measurements of antenna parameters could be confirmed throughout the course of the exercise without external factors distorting the final values. The graph in Fig.1 shows measured values in comparison with the simulation. When the curve of the actual measurements is compared to a simulation curve, one can notice that the antenna achieves similar values as compared with the simulation, which is desirable. For measurements in the anechoic chamber the source horn antenna and signal generator (type SMR 20) by Rohde & Schwarz were used.

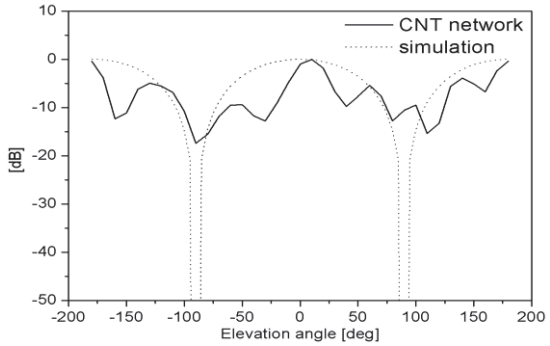


Fig.1 Radiation pattern values of the antenna in comparison with the simulation.

The measured antenna is equipped with a semi-rigid coaxial cable and a gold-plated SMA connector. The ground plate consists of the copper layer on the "Cuprextit" substrate. The coaxial cable is cold mounted on the carbon layer. In order to ensure a high quality contact, a silver electrically conductive paste with the fast curing characteristics was used; on application it hardens the joint. Such a joint is perfectly protected against damage that could occur during manipulation with the antenna. The carbon strip is very fragile, therefore, it is always necessary to use some kind of substrates on which the strip is placed. In this case, the polymethyl methacrylate (PMMA) substrate was used. From the perspective of the actual microstrip antenna the polymethyl methacrylate (PMMA) is a nonconductive substrate. The substrates based on polymethyl methacrylate (PMMA) are used only in certain types of antennas for portable devices. A number of wireless portable devices consist mainly of plastic materials. The advantage of polymeric substrates makes it possible to apply it directly to the antenna and to plastic covers of portable communication devices. The purified MWCNTs produced by the chemical vapor deposition of acetylene were supplied by Sun Nanotech Co. Ltd, China. According to the supplier, the nanotubes have diameters of 10–30 nm, length 1–10 μm, purity >90 % and electrical resistivity 0.12 Ω.cm.

0.6 g of the as-received MWCNTs were dispersed in 100 ml of the nitric acid (65 wt%) in a 250 ml round bottom flask equipped with a condenser and the dispersion was refluxed for 5 h. After that, the resulting dispersion was diluted in water and filtered. The resulting solid was washed up to neutral pH, and the sample was dried overnight. We prepared two types of carbon nanotubes pure one called *china-pure* and oxidized form called *china-HNO_3*.

MWCNT were analyzed via high-resolution transmission electron microscopy (TEM) using microscope JEOL JEM 2010 at the accelerating voltage of 160 kV. Before that, samples had been deposited on a copper grid (SPI, USA) and dried.

The figures show the TEM analysis of the multi-walled carbon nanotubes (Fig.2) and individual carbon nanotubes with around 15 rolled graphene layers (Fig.3). Both forms were used for the preparation of an aqueous dispersion.

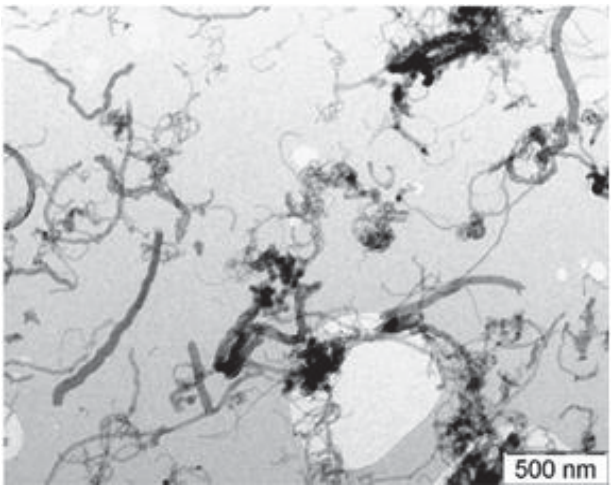


Fig.2 TEM image of multi-walled carbon nanotubes deposited on the carbon film.

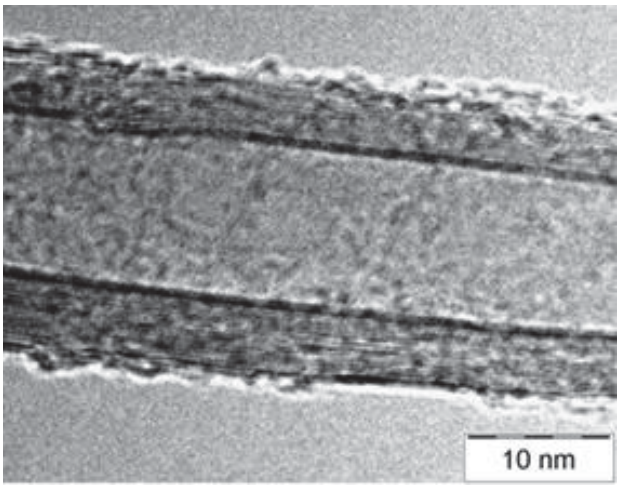


Fig.3 HRTEM detailed view of the structure of the nanotube.

The procedure is following: 1.6 g of MWCNT and approximately 50 ml of deionized water were mixed using a mortar and pestle. The paste was then diluted in deionized water with sodium dodecyl sulfate (SDS) and 1-pentanol. Consequently, NaOH aqueous solution was added to adjust the pH to the value of 10 [1]. The final nanotube concentration in suspension was 0.3 wt%, and the concentrations of SDS and 1-pentanol were 0.1 and 0.14 M, respectively [2]. The suspension was sonicated in UZ Sonoplus HD 2070 kit for 2 h at a temperature of about 50 ◦C. For making the entangled MWCNT network on a polyurethane porous membrane [3], a vacuum filtration method was used [10].

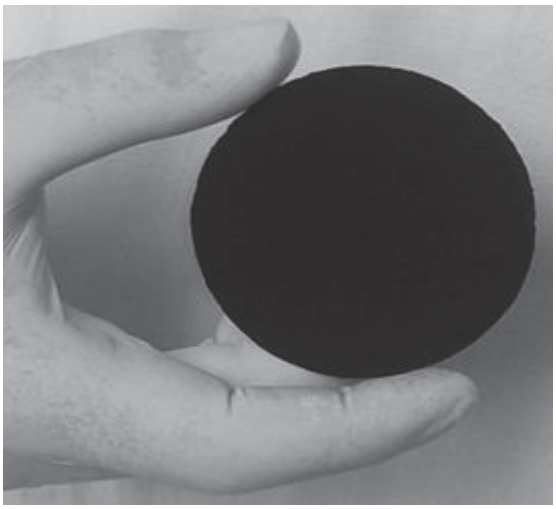


Fig.4 Free-standing randomly entangled MWCNT network (disk diameter 75 mm and thickness 0.15 mm).

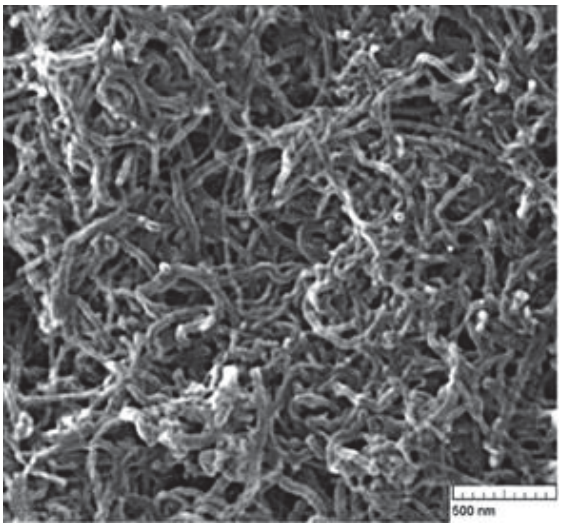


Fig.5 SEM image of the surface of entangled MCWNT network of buckypaper.

The formed disk-shaped network (Fig.4) was washed several times by deionized water and methanol in situ, and then peeled off and dried between filter papers. The thickness of the obtained disks was typically 0.15–0.46 mm (Fig.4). The structure of the MWCNT network was analyzed by a scanning electron microscope (SEM) technique.
SEM analysis was carried out using Vega II LMU (Tescan, Czech Republic) with a beam acceleration voltage set at 10 kV.The microscope image clearly shows the carbon nanotubes network, which was created by crossing the tubes during a vacuum filtration process (Fig.5). The resulting network has a visible porous structure. The carbon nanotubes network, sometimes called buckypaper, was used for the preparation of the specimen. The buckypaper was cut to a rectangle shape with dimensions 5x 15 mm [4, 11].

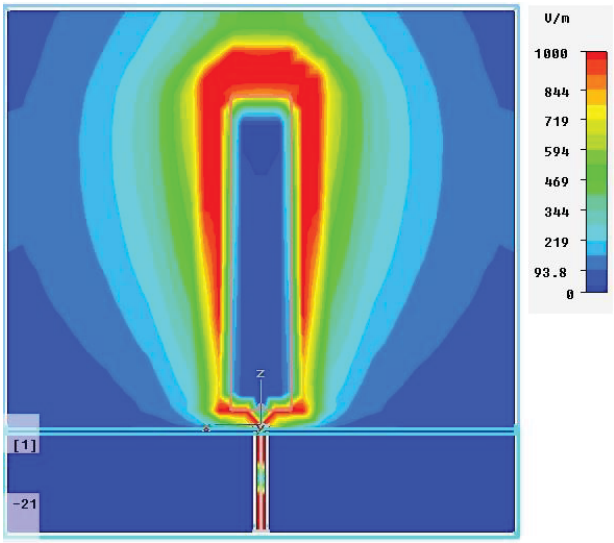


Fig.6 Simulation of electromagnetic field of proposed antenna

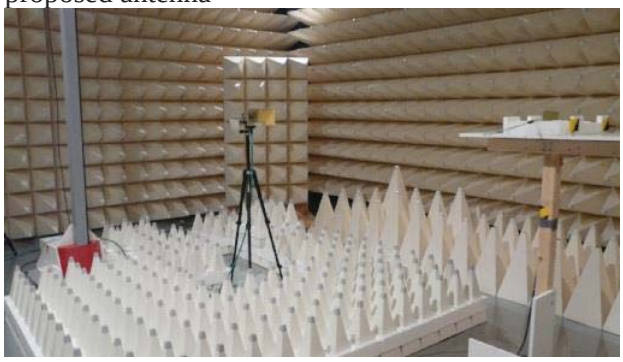


Fig.7 Measuring of proposed antenna in anechoic chamber

The Fig.7 shows us anechoic chamber used for accurate measurement of antenna characteristics. We use an anechoic chamber from Frankonia. Measuring devices inside our (faculty) anechoic chamber are calibrated and supplied by Rohde & Schwarz.

3 Results

The measurements of the antenna were performed using the R&S FSH3 spectrum analyzer (the model with a tracking generator and without a preamplifier) and FSH-Z2 bridge capable of measuring within a bandwidth of 100 kHz to 3 GHz. For experimental measurements this range is sufficient. The following graphs depict the operation of the measured antenna.

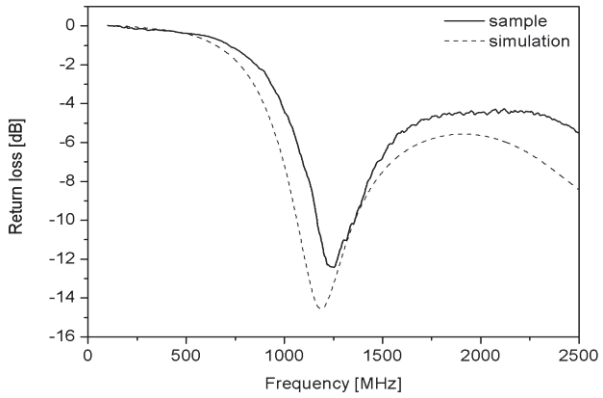


Fig.8 The course of reflection coefficient "r" depends on frequency in dB.

The course of the reflection (Fig.8) coefficient "r" (expressed in dB) based on the frequency. In accordance with the marker, at a frequency of 1.284 GHz the antenna shows a minimum reflection -11.48 dB and thus, its best impedance matching is just at this frequency. The measured curve is in good agreement with simulation which is also shown in Fig.8. The quantification for the frequency is as follows: -11.48 dB corresponds to r = 0.2667, and this corresponds to a standing wave ratio VSWR = 1.73 (1 = perfect matching, 100 % is emitted; the infinity = does not match at all, nothing is emitted). At this frequency, the power transmission is the best as it reaches 93 %, which means that at the frequency of 1.284 GHz 93 % of the power is to be emitted while 7 % reflected (it returns back to the exciter). The farther away from each side of the frequency the worse it is. The reflection coefficient grows bigger and so does the standing wave ratio. Therefore, all negative aspects connected to impedance mismatching of the antenna to the signal source (50 ohm) appear, namely worse efficiency, overloading of the excitation source, and harmful interference owing to the emission of the standing wave [12, 13].

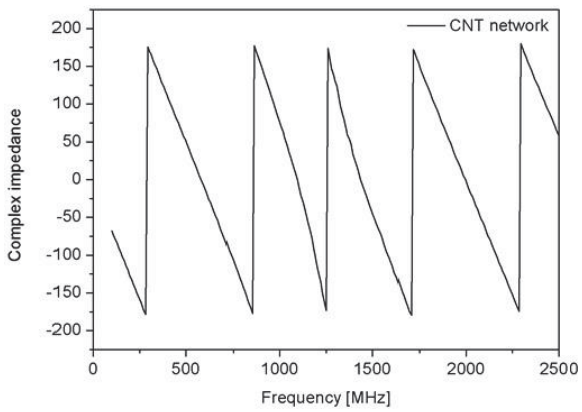


Fig.9 The course of phase of impedance depends on frequency.

The graph in Fig.9 shows the course of the phase of complex impedance at the given frequency. The antenna shows a real part of impedance (where the graph intersects x axis, this means for y = 0) within the frequency range of 100 MHz to 2.5 GHz at 8 frequencies. At the frequency of 1.284 GHz it is 50 ohms for which the characteristics were measured (Fig.8). Outside 50 ohms, Fig.8 also shows the ideal impedance matching within the measuring range for the other 7 values [7]. The simulation of the proposed antenna (Fig.6) was simulated in CST microwave studio.

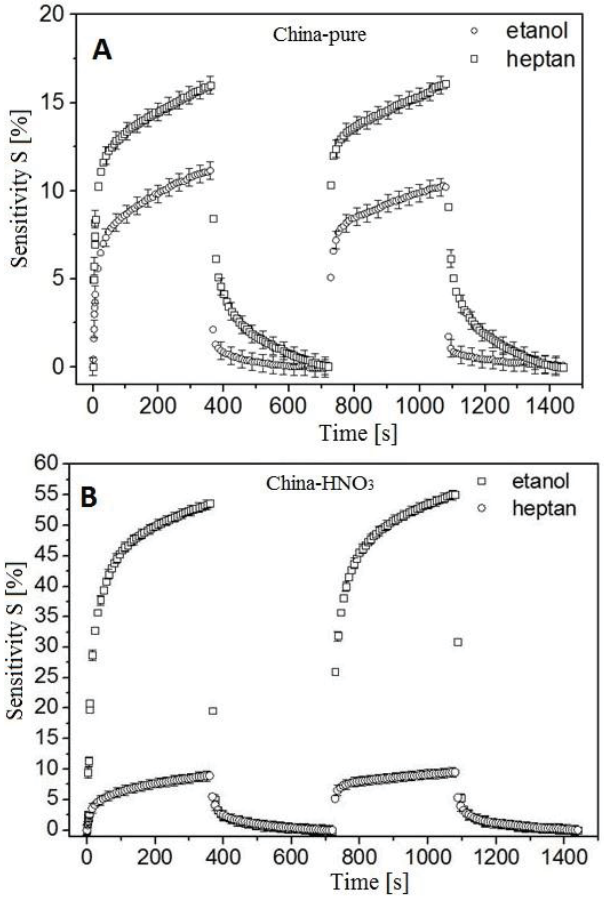


Fig. 10 Two adsorption/desorption cycles for a) pure carbon nanotubes network *china-pure* and b) network oxidized by concentric nitric acid *china-HNO_3*

The strips made of CNT networks, sometimes called buckypaper, Fig. 4 were exposed to the vapors of two different solvents: ethanol and heptane. The conditions were 25 °C and each adsorption/desorption cycle was 6 minute long.

The data in Fig. 10 represent two adsorption to/desorption from cycles for specimens made of pure CNT networks exposed to ethanol and heptane. Then Fig. 10 part b) demonstrates analogical data for network made of HNO_3 oxidized tubes. Over all, the adsorption of organic molecules increases resistance with time, which is presented in the figure as sensitivity or gas response S. S is defined as a percentage change of resistivity explain by equation 1.

$$S = \Delta R/R_0 \,.\, 100 \;\; [\%] \qquad (1)$$

Where S represents sensitivity of the sensor in % and R_0 is initial resistivity.

Solvent	vapor pressure at 25 °C [kPa]	volume in air [%]	sensitivity S [%] *china-pure*	sensitivity S [%] *china-HNO3*
Ethanol	5.95	5.87	10	53
Heptane	5.33	5.26	17	7

Tab. 1 Summary of solvent properties mainly vapor pressure at 25 °C volume fraction of saturated vapor above solvent and sensitivity response for *china-pure* and oxidized *china-HNO_3*.

4 Conclusion

Multiwall carbon nanotubes were used to prepare CNT network (buckypaper) by vacuum filtration method. Presented data shows that this multifunctional could be used as an antenna and a gas sensing element. The antenna works at 1.284 GHz and could be used in telecommunication technologies. Main advantages are light weight and the possibility to be incorporated to a mobile phone cover. The carbon nanotubes network as an antenna could replace commonly used materials in the future.

All measured values proved that carbon nanotubes of the multifunctional polymer can be used for the antenna operating in the bandwidth of 1.284 GHz. Therefore, the antenna operating within this bandwidth will be suitable for the field of communication technologies. The main advantages of the antenna are low weight and the possibility for applying to plastic covers of portable telecommunication devices and wearable electronics. Portable telecommunication devices, such as mobile phones, use for communication antennas made of copper foils and strips, which

could be replaced by the developed antenna. The majority of widespread technologies, e.g. GSM, UMTS, or LTE operate at frequencies for which it is possible to produce a customized antenna made of carbon nanotubes [8, 9].

Multiwall carbon nanotubes were also used for preparation of gas and vapors sensing elements. Their response to adsorption/desorption cycles were determined as a change of macroscopic resistance. The response was measured by a two point method. CNT network has good sensitivity and assumed selectivity defined by polarity of used solvent. Chemical oxidation method was use as a way how to introduced carboxylic and carbonilic group on the carbon nanotubes walls. The oxidation by concentric nitric acid was reached higher sensitivity for polar solvent in this case represents by ethanol and decreasing response for nonpolar solvent heptane. Finally, it was found that measured response has good reversibility.

Acknowledgement

The work behind the article was supported by the Internal Grant Agency of Tomas Bata University in Zlin with No. IGA/FAI/2015/007. We also acknowledge the support of the Ministry of Education, Youth and Sports of the Czech Republic - Program NPU I (LO1504).

References:

[1] C.S. Chern and L.J. Wu, Microemulsion polymerization of styrene stabilized by sodium dodecyl sulfate and short-chain alcohols, Polym. Sci. Part A: Polym. Chem. 39 (2001), pp. 3199–3210.

[2] D. Kimmer, P. Slobodian, D. Petras, M. Zatloukal, R. Olejnik, and P. Saha, Polyurethane/MWCNT nanowebs prepared by electrospinning process, J. Appl. Polym. Sci. 111 (2009),pp. 2711–2714.

[3] X.L. Xie, Y.W. Mai, and X.P. Zhou, Dispersion and alignment of carbon nanotubes in polymer matrix: A review, Mater. Sci. Eng. R 49 (2005), pp. 89–112.

[4] J. Li, "Carbon nanotubes applications: chemical and physical sensors," in Carbon Nanotubes: Science and Applications, M. Meyyappan, Ed., CRC Press, Boca Raton, Fla, USA.

[5] G.W. Hanson, Fundamental transmitting properties of carbon nanotube antennas,IEEE *Transactions on Antennas and Propagation* 53 (2005) 3426–3435.

[6] Mehdipour, A.; Rosca, I.D.; Sebak, A.; Trueman, C.W.; Hoa, S.V., "Carbon Nanotube Composites for Wideband Millimeter-Wave Antenna Applications," Antennas and Propagation, IEEE Transactions on , vol.59, no.10, pp.3572,3578, Oct. 201,doi: 10.1109/TAP.2011.2163755.

[7] BALANIS, Constantine A. Antenna Theory: Analysis and Design. 2nd edition. New York: John Wiley & Sons, Inc, 1997, 941 s. ISBN 0-471-59268-4.

[8] Huey Shin Wong, Mohammad Tariqul Islam, and Salehin Kibria, "Design and Optimization of LTE 1800 MIMO Antenna," *The Scientific World Journal*, vol. 2014, Article ID 725806, 10 pages, 2014. doi:10.1155/2014/725806

[9] Weifeng Sun, Guoqiang Zhang, Jingjing Zhou, and Vijay Bhuse, "Next-Generation Internet and Communication," *The Scientific World Journal*, vol. 2014, Article ID 342471, 2 pages, 2014. doi:10.1155/2014/342471.

[10] M. T. Islam and M. Samsuzzaman, "Miniaturized Dual Band Multislotted Patch Antenna on Polytetrafluoroethylene Glass Microfiber Reinforced for C/X Band Applications," *The Scientific World Journal*, vol. 2014, Article ID 673846, 14 pages, 2014. doi:10.1155/2014/673846.

[11] MATYAS, Jiri, Robert OLEJNIK, Karel VLCEK a Petr SLOBODIAN. The use of carbon nanotubes applied to plastic substrates for the construction of a passive antenna. In: Latest Trends in Circuits, Systems, Signal Processing and Automatic Control: *Proceedings of the 5th International Conference on Circuits, Systems, Control, Signals (CSCS ´14).* Salerno: WSEAS Press, 2014, s. 297-300. ISBN 978-960-474-374-2ISSN 1790-5117.

[12] Ahmed Ali , Herve Aubert, Fractal superlattice cover with variable lacunarity for patch antenna directivity enhancement analysis and design, WSEAS TRANSACTIONS on COMMUNICATIONS, v.7 n.10, p.1035-1044, October 2008.

[13] S. Raghavan , N. Jayanthi, Design of planar inverted - F antenna for wireless applications, WSEAS TRANSACTIONS on COMMUNICATIONS, v.8 n.8, p.863-872, August 2009.

Sponzoři konference

Společnost CROSS Zlín byla založena v roce 1994 a zabývá se vývojem, dodávkou, instalací a údržbou špičkových technologických zařízení pro silniční dopravu. Je v tomto oboru jednou z největších tuzemských společností s ryze českou vlastnickou strukturou. Všechny produkty a výrobkové řady (hardware a software) jsou realizovány vlastními zdroji společnosti.

CROSS Zlín je vedoucím dodavatelem světelných signalizačních zařízení a systémů v České republice. Společnost je také velmi aktivní v exportu těchto zařízení do dalších zemí EU a do Ruska. Má dominantní postavení v oblasti silniční meteorologie na území České republiky a je také největším tuzemským výrobcem produktů pro detekci dopravy a vážení za jízdy. Produktové portfolio doplňují proexportně orientované parkovací systémy a platební terminály.

Firma ASICentrum s.r.o. (název je odvozen od Application Specific Integrated Circuits - zákaznické integrované obvody) vznikla v roce 1992. Počátkem roku 2001 se stala součástí mezinárodní skupiny SWATCH GROUP prostřednictvím 51% podílu, který získala švýcarská firma EM Microelectronic. Ta byla založena v roce 1975 a v současnosti je 100% dceřinná firma společnosti SWATCH GROUP.

EM Microelectronic Marin je významným výrobcem integrovaných obvodů. Své zkušenosti z návrhu a výroby obvodů s extrémně nízkým příkonem, původně pro hodinky, zhodnotila v řadě dalších aplikací jako RFID, počítačové periferie, systémy pro automobily, průmysl a zdravotnictví. Pro řadu významných zákazníků z mnoha zemí světa vyrábí ročně stovky miliónů čipů.

Společnost HELLA je mezinárodně orientovaný nezávislý rodinný podnik s více než 32.000 zaměstnanci a více než 100 zastoupeními ve více než 35 zemích. Koncern HELLA vyvíjí a vyrábí v obchodním odvětví Automotive komponenty a systémy osvětlení a elektroniky. Dále zahrnuje společnost HELLA v segmentu Aftermarket také jednu z největších obchodních sítí pro díly a příslušenství do automobilů a diagnostických a servisních služeb v Evropě. Kromě toho vyvíjí společnost HELLA v segmentu Special Applications výrobky pro speciální vozidla a zcela nezávislé aplikace, jako pouliční osvětlení nebo průmyslové osvětlení. Ve společných podnicích s partnery vznikají navíc i kompletní moduly, klimatizační jednotky a palubní sítě do automobilů. S více než 6 000 zaměstnanci ve výzkumu a vývoji patří společnost HELLA k hlavním průkopníkům inovací na trhu. S obratem cca 5,8 mld. eur za účetní rok 2014/2015 se koncern HELLA zařadil mezi 50 největších dodavatelů dílů pro automobilový průmysl a patří do stovky největších průmyslových podniků v Německu.

Časopis DPS Elektronika od A do Z svým zaměřením pokrývá celé spektrum oboru elektroniky.

Na svých stránkách se pravidelně věnuje aktuálním trendům ve vývoji elektroniky, novinkám v technologických postupech, výrobních zařízeních a materiálech. Přináší také informace o nových elektronických součástkách a zajímavých možnostech jejich použití, pravidelně se věnuje měřicí a testovací technice, stejně jako softwarovým nástrojům pro vývoj a výrobu elektronických zařízení.

V neposlední řadě přináší i ekonomické informace či pravidelně aktualizovaný kalendář odborných akcí pořádaných v České republice i v zahraničí.

Časopis je dvouměsíčník a vychází jak v tištěné, tak elektronickém podobě. Čtenářům je distribuován formou předplatného a cíleně také odborné veřejnosti na významných domácích i zahraničních veletrzích, výstavách, konferencích a seminářích.

Evektor je mezinárodní společnost a patří mezi přední vývojové a výrobní společnosti působící v leteckém průmyslu v České republice. Kromě letectví má Evektor rozsáhlé vývojové aktivity v automobilovém a strojírenském průmyslu. Společnost byla založena v roce 1991 a již od roku 1992 působí v oblasti vývoje a konstrukce letadel. Historicky Evektor nazvazuje na tradice společnosti Aerotechnik CZ, která se stala jeho součástí v roce 1996, a která byla založena v roce 1970 a během své existence vyrobila, mimo jiné, téměř 200 letounů řady L-13 Vivat.

Společnost Evektor je aktivně zapojena do mezinárodní spolupráce a účastní se výzkumných a vývojových projektů podporovaných Evropskou unií a Ministerstvem průmyslu a obchodu ČR.

Evektor je jeden z leaderů celoevropského výzkumného programu Clean Sky 2. Tento program je zaměřen na výzkum nových technologií v letectví, které mají snížit spotřebu paliva, emise skleníkových plynů CO2 a NOx a na snížení hlukových emisí, které vznikají při provozu letounů. Cílem je snížit tyto emise o 20 - 30% ve srovnání se současným stavem.

Počítačové architektury a diagnostika PAD 2015
česko-slovenský seminář pro studenty doktorského studia

Sborník příspěvků

Název: Počítačové architektury a diagnostika PAD 2015
Autoři: Kolektiv autorů příspěvků
Vydavatel: UTB ve Zlíně
Fakulta aplikované informatiky
Pořadí vydáni: První
Rok vydání: 2015

ISBN: 978-80-7454-522-1