

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ  
FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

# **POLYMORFNÍ ELEKTRONIKA**

**HABILITAČNÍ PRÁCE**

pro obor

VÝPOČETNÍ TECHNIKA A INFORMATIKA

Brno 2011

**Ing. Richard Růžička, Ph.D.**

# Abstrakt

Polymorfní elektronika je oblast elektroniky, zkoumající číslicové elektronické obvody, které dokážou plnit více než jednu funkci, přičemž jejich zapojení zůstává stále stejné a aktuálně prováděná funkce závisí na stavu okolního prostředí. Všechny funkce jsou do obvodu zabudovány úmyslně již při jeho návrhu, počítá se s nimi. Polymorfní obvody jsou ve srovnání s konvenčními vícefunkčními obvody rozměrově velmi úsporné. Změna funkce nastává přirozeně a okamžitě a detekce stavu prostředí je přímo vestavěna do obvodu. Typickým prostředím, ovlivňujícím funkci polymorfních obvodů, bývá napájecí napětí, teplota, intenzita osvětlení a podobně, tedy veličiny, které ovlivňují chování polovodičových tranzistorů. Koncept polymorfní elektroniky je v této práci zobrazen, není vázán pouze na komponenty sestavené z polovodičových unipolárních tranzistorů, i když současné aplikace na nich stojí. Mezi základní problémy, komplikující rozvoj polymorfní elektroniky, patří: návrh polymorfních obvodů, hledání vhodných základních stavebních bloků (hradel) a identifikace oblastí aplikace, kde polymorfní elektronika přinese výhodu a pokrok oproti známým řešením. V této práci lze nalézt řešení všech těchto základních problémů. Obsahuje popis téměř všech dnes existujících polymorfních hradel a jejich hodnocení. Dále jsou představeny a diskutovány známé metody syntézy polymorfních obvodů. Řada obvodů byla navržena evolučními technikami, zejména metodou Kartézského genetického programování. Byly ale navrženy i neevoluční (konvenční) metody syntézy. V práci je popsána i řada aplikací. Většina aplikací byla realizována a ověřena buď simulačně, nebo přímo fyzickou realizací. Obecně lze polymorfní obvody nasadit v aplikacích, kde se předpokládá adaptace obvodu na měnící se podmínky nebo rychlá a elegantní rekonfigurace. Toho lze využít v obvodech, které se samy přizpůsobí nepříznivým podmínkám – omezí svoji spotřebu, tepelné vyzařování, při zachování alespoň nezbytné základní činnosti. Prosadit se však mohou i tam, kde se předpokládá obvod sice monofunkční, ale s nějakou přidanou vlastností. Tato přidaná vlastnost přitom může být využívána jen zřídka, ale díky typické vlastnosti polymorfních obvodů – kompaktní a prostorově velmi nenáročná konstrukci – může být získána za velmi příznivou cenu. Může to být v oblasti vestavěné diagnostiky, bezpečnosti a podobně. Principů polymorfní elektroniky lze využít nejen u kombinačních, ale i u sekvenčních číslicových obvodů. V práci je ukázáno, jak lze využít vlastností polymorfních obvodů pro návrh bezpečných adaptivních obvodových řadičů. Pro reálné experimenty s polymorfními obvody byl navržen a vyroben první rekonfigurovatelný polymorfní obvod na světě. Text práce a zejména autorem provedené definice a tvrzení odráží kromě očekávaného zobecňujícího a vědeckého přístupu také praktickou zkušenost, kterou autor za léta práce s polymorfními obvody získal.

# Abstract

Polymorphic electronics is a new domain of electronics comprising digital circuits, which are able to perform more than one function. Polymorphic circuits typically have one stable structure for all functions and actually performed function depends on an environment. All functions of the circuit are designed intentionally. Polymorphic circuits are very area efficient in comparison to conventional multi-function circuits. Change of the circuit function comes immediately (with no delay) and sensitivity to the environment is embedded to the circuit. Typically, power supply voltage, temperature or a special signal serves as the environment which determines the function of the circuit, i.e. physical quantities that affect behaviour and parameters of semiconductors. But the concept of polymorphic electronics is generalised in this thesis, although today's applications are based on unipolar semiconductor transistors. There are three basic problems of polymorphic electronics: design methods for polymorphic circuits, lack of suitable polymorphic components (gates) and identification of appropriate application areas. In this thesis, several possible solutions of all three main problems of polymorphic electronics are described. The thesis contains description and evaluation of almost all existing polymorphic gates. Also known methods of polymorphic circuits' synthesis are introduced and discussed. Lot of polymorphic circuits were designed using evolutionary design methods, especially using Cartesian Genetic Programming. But also non-evolutionary (conventional) design methods were proposed. A range of applications is described in the thesis. Majority of them were implemented and verified by physical realisation. Generally, polymorphic circuits can be used in applications that assume adaptation of the circuit to variable environment or smart and fast reconfiguration. Such behaviour is useful for circuits that must adapt itself to unfriendly environment e.g. by restriction of power consumption or heat dissipation with preservation of necessary basic functionality. Polymorphic electronics is profitable also in applications that are basically mono-functional, but need some additional feature. The additional feature may be used rarely, but due to typical area effectiveness of polymorphic circuits, the additional feature is keenly priced. This is attractive e.g. for embedded diagnostics, security applications etc. Principles of polymorphic electronics are useful for both combinational and sequential circuits. It is shown in the thesis how to utilise polymorphic electronics for design of safe adaptive circuit controllers. For real experiments, the first reconfigurable polymorphic chip in the world was designed and manufactured. The text of the thesis, and namely definitions and assertions stated in the thesis done by the author, reflects also practical experiences acquired during years of work with polymorphic electronics.

# Poděkování

Tato habilitační práce se zabývá výzkumnou oblastí nazvanou polymorfní elektronika. S tímto pojmem přišel před deseti lety Adrian Stoica, vědec, podnikatel a snílek (jak sám sebe charakterizuje), vedoucí výzkumné skupiny ve slavných laboratořích tryskových pohonů (JPL) NASA. Spolu se svými kolegy Didierem Keymeulenem a Ricardem Zebulem publikovali základní teze polymorfní elektroniky a provedli první praktické experimenty, na které jsem mohl navázat. Za to jim patří můj dík.

Výzkum polymorfní elektroniky přinesl na FIT VUT v Brně prof. Ing. Lukáš Sekanina, Ph.D., který pracoval v roce 2004 několik měsíců v týmu A. Stoicy jako výzkumník. Byl to právě prof. Sekanina, který inspiroval mne i další moje kolegy k tomu, abychom se polymorfní elektronikou zabývali. Díky němu, jeho myšlenkám i tvrdé práci při získávání prostředků na další výzkum dnes máme první konfigurovatelný polymorfní obvod na světě a na něm řadu funkčních aplikací, prokazujících více než co jiného životaschopnost Stoicových ideí. Bez něj by tato práce sotva vznikla.

Do týmu, který se podílel a podílí na FIT VUT v Brně na výzkumu v oblasti polymorfní elektroniky, patří také kolegové Ing. Zbyšek Gajda, Ph.D., jehož disertační práce přinesla nové metody syntézy polymorfních obvodů, doc. Ing. Zdeněk Kotásek, CSc. a Ing. Lukáš Stareček, kteří využívají polymorfní elektroniku v diagnostice číslicových obvodů, Ing. Michal Bidlo, Ph.D., který přispěl k výzkumu developmentu polymorfních obvodů, Ing. Zdeněk Vašíček, který se podílel na celé řadě aplikací, například na experimentech s fyzickou neklonovatelnou funkcí v polymorfních obvodech, Ing. Václav Šimek, který má velký podíl na tvorbě kitu s konfigurovatelným polymorfním čipem REPOMO a Ing. Luděk Žaloudek, který evolucí na úrovni tranzistorů vytvořil některá nová polymorfní hradla. Díky jejich výsledkům skýtá tato práce na polymorfní elektroniku daleko ucelenější pohled.

Děkuji své manželce Michaele za podporu v období, kdy tato práce vznikala, za její lásku a péči, bez níž bych v tuto chvíli ještě zdaleka nebyl s prací hotov.

Výzkum, popsáný v této práci, byl prováděn v rámci projektů „Metody návrhu polymorfních číslicových obvodů“, GAČR, GA102/06/0599, 2006-2009, „Natural computing na nekonvenčních platformách“, GAČR, GP103/10/1517, 2010-2013, „Zvyšování spolehlivosti a provozuschopnosti v obvodech SoC“, GAČR, GA102/09/1668, 2009-2011 a „Výzkum informačních technologií z hlediska bezpečnosti“, CEZ MŠMT, MSM0021630528, 2007-2013.

Richard Růžička  
září 2011

# Obsah

1 Úvod	1
1.1 Obvody měnící funkci	1
1.2 Cíle práce	2
1.3 Struktura práce	2
2 Polymorfní elektronika	5
2.1 Vymezení pojmu	5
2.2 Komponenty pro polymorfní obvody	6
2.3 Historie	8
2.4 Prostředí	9
2.5 Známé způsoby řízení funkce polymorfního obvodu	10
2.5.1 Řízení napájecím napětím	10
2.5.2 Řízení teplotou	10
2.5.3 Řízení speciálním signálem	11
3 Polymorfní hradla	12
3.1 Hradlo jako základní stavební prvek kombinačních číslicových obvodů	12
3.2 Unipolární tranzistor typu MOS	12
3.3 Hradlo v technologii CMOS	14
3.4 Polymorfní hradlo – definice a požadavky	16
3.5 Známá polymorfní hradla	18
3.5.1 Polymorfní hradla řízená napájecím napětím	18
3.5.2 Hradla řízená teplotou	40
3.5.3 Hradla řízená speciálním signálem	43
3.5.4 Polymorfní grafenové hradlo	52
4 Syntéza polymorfních obvodů	54
4.1 Cíle a zásady návrhu polymorfních obvodů	54
4.2 Ad hoc ruční návrh polymorfních obvodů	55
4.3 Kartézské genetické programování	55
4.4 Evoluční návrh polymorfních obvodů	57
4.5 Polymorfní multiplexování	59
4.6 Návrh polymorfních obvodů s využitím BDD	60
4.7 Optimalizace polymorfních obvodů	62
5 Experimentální platforma pro polymorfní obvody	63
5.1 Konfigurovatelné obvody	63
5.2 REPOMO	66
5.3 REPOMO32/kit	70
6 Aplikace kombinačních polymorfních obvodů	76
6.1 Možnosti využití vícefunkčních obvodů s reakcí na prostředí	76
6.2 Aplikace polymorfních obvodů v diagnostice a testování	77
6.2.1 Zkrácení testu číslicového obvodu s využitím polymorfních hradel	77
6.2.2 Samotestující se sčítačky	78
6.3 Zvýšení odolnosti proti poruchám aplikací polymorfní elektroniky	81
6.4 Obvody se skrytou funkcí	87
6.5 Využití polymorfních hradel v návrzích jednofunkčních obvodů	93

7 Sekvenční polymorfní obvody	94
7.1 Sekvenční obvody	94
7.2 Polymorfní klopné obvody	95
7.3 Polymorfní čítače a řadiče	97
7.4 Aplikace polymorfních čítačů a řadičů	99
8 Závěr	107
8.1 Shrnutí	107
8.2 Zhodnocení práce	110
8.3 Budoucnost	111
Literatura	113

# 1 Úvod

## 1.1 Obvody měnící funkci

Obvody, které mění funkci, nachází dnes uplatnění v celé řadě aplikací. Mezi nejvýznamnější patří adaptace na zhoršené podmínky, ať už se jedná o poruchu vlastního obvodu, změnu vstupních dat nebo je třeba zpracovávat vstupní data jiným způsobem. Konvenční způsob řešení změny funkce obvodu během jeho činnosti předpokládá, že se pro možné funkce připraví příslušné obvody a mezi nimi se podle potřeby přepíná. Tento způsob však klade vysoké nároky na prostor a tím pádem i cenu řešení. Další možností je využití rekonfigurovatelného obvodu [Sek00], [Hig99]. Rekonfigurovatelný obvod obsahuje sadu univerzálních funkčních elementů a programovatelnou propojovací síť (viz též část 5.1 této práce). V případě využití rekonfigurovatelného obvodu jsou prostorové nároky obecně o něco nižší, než kdyby byly fyzicky implementovány obě či více funkcí s přepínáním, i když je i zde třeba počítat s jistou režii – prvky rekonfigurovatelného obvodu nejsou vždy pro požadovanou funkci optimální, je třeba počítat i s pamětí pro uložení konfigurací a mechanismem, který ji bude podle požadavků měnit. Nevýhodou je ale časová prodleva při rekonfiguraci – změna konfigurace není okamžitá a v průběhu rekonfigurace je obvod mimo provoz.

Pokrok v možnostech návrhu číslicových obvodů vlivem podpory výkonnou výpočetní technikou, ale zejména nové možnosti, jak nalézat možné hardwarové implementace různých funkcí – zvláště evoluční techniky [Bac96], [Kva00], [Eib07] – otevírají také nové možnosti realizace obvodů s více funkcemi. J. Miller, autor Kartézského genetického programování [Mil00a], říká: „Předností konvenčních návrhových metod je, že dokážou pracovat s velkým počtem vstupních proměnných, nejsou však schopny se adaptovat na nové stavební logické bloky a vyžadují velké množství analytické práce k tomu, aby vyprodukovaly malé vylepšení. Evoluční algoritmus naopak postupně přivádí řešení k požadované pravdivostní tabulce. Pracuje tak v mnohem větším prostoru logických funkcí, v jehož rámci mnohé z nich nereprezentují požadovanou funkci. Jedná se o jediný způsob, jak objevit radikálně nová řešení“ [Mil00b], [Sek09a]. Evoluční návrh přitom také na rozdíl od konvenčních metod návrhu není omezený v použití komponent, z nichž se obvod může skládat.

Byly to právě evoluční techniky návrhu a experimenty s nimi, které přivedly A. Stoicu, R. Zebuluma a D. Keymeulena z Jet Propulsion Laboratories NASA na myšlenku jediného obvodu, který by mohl realizovat více funkcí, aniž by se změnil. Změna funkce by nastala pouze na úrovni pracovních bodů tranzistorů. Pracovní bod tranzistorů lze ovlivnit různými způsoby, mezi nejběžnější a nejpřirozenější patří změna (napájecího) napětí, teploty čipu, osvětlení čipu atd. Takové obvody nazvali autoři polymorfními obvody a oblast, která se jimi zabývá, polymorfní elektronikou [Sto01].

Lze říci, že evoluční techniky umožnily opětné sblížení číslicové a analogové elektroniky, zatím běžné na úrovni meziregistrových přenosů, případně na systémové úrovni (mixed-signal circuits), tentokrát ovšem na úrovni tranzistorů. Polymorfní obvody lze v určitých ohledech považovat za analogové, v jiných za číslicové. Jejich zamýšlené funkce a použití však patří jistě do oblasti číslicové elektroniky. Polymorfní obvod je typický tím, že je rozměrově úsporný při možnosti realizovat více funkcí a dále tím, že funkce může být měněna netradičním způsobem. Obvod může reagovat na změnu prostředí a změnou funkce například zmírnit jeho nepříznivé vlivy (adaptovat se na prostředí).

Ukazuje se, že možnosti takto postavených obvodů jsou poměrně široké. Původní záměr návrhu kompaktních adaptivních obvodů byl v průběhu let značně rozšířen, byly nalezeny další aplikace a také jiné možnosti návrhu polymorfních obvodů než evoluční.

## 1.2 Cíle práce

Polymorfní elektronika se ukazuje jako slibný koncept v oblasti návrhu kompaktních (rozměrově a cenově výhodných) obvodů, které lze použít v oblastech s potřebou změny funkce obvodu v průběhu jeho činnosti. Velkou výhodou je přirozeně vestavěná citlivost na prostředí. Koncept polymorfní elektroniky s sebou přináší také řadu problémů, na jejichž uspokojivém řešení závisí, nakolik budou polymorfní obvody zajímavé z pohledu praktických aplikací v doménách, kde již existují úspěšná konvenční řešení a zda se prosadí v oblastech, kde dosud nevýhody jiných koncepcí brání nalezení úspěšného řešení.

Mezi základní problémy polymorfní elektroniky patří:

- návrh polymorfních obvodů – metody syntézy polymorfních obvodů, které by byly spolehlivé, nenáročné, škálovatelné,
- existence vhodných základních stavebních bloků (hradel) s parametry, které by byly srovnatelné nebo lepší než parametry konvenčních hradel používaných dnes v číslicové elektronice,
- identifikace oblastí aplikace, kde polymorfní elektronika přinese výhodu a pokrok oproti známým řešením.

Autor této práce experimentuje již pět let s polymorfní elektronikou, navrhl možnosti řešení některých problémů a řadu polymorfních obvodů také prakticky vyzkoušel v obvodové realizaci. Při této práci získal řadu zkušeností a odhalil praktické aspekty oblasti polymorfní elektroniky. V této práci jsou jeho poznatky a zkušenosti shrnuty. Autor se však pokusil nejen shromáždit a formulovat své vlastní myšlenky, ale snažil se vytvořit komplexní pohled na tuto relativně novou výzkumnou oblast. Proto práce obsahuje i formální definici polymorfní elektroniky a problémů s ní souvisejících, přehled známých řešení výše uvedených problémů syntézy polymorfních obvodů, přehled polymorfních hradel a jejich zhodnocení i přehled aplikací, kde polymorfní elektronika nachází uplatnění. V práci lze tedy nalézt řešení všech tří naznačených základních problémů polymorfní elektroniky.

## 1.3 Struktura práce

V kapitole 2 je představen koncept polymorfní elektroniky, jak jej navrhl A. Stoica s kolektivem z laboratoří JPL NASA a také vývoj konceptu v průběhu let tak, jak tým A. Stoicy publikoval své práce z této oblasti. Protože A. Stoica představuje spíše ideu polymorfní elektroniky a dále se věnuje evolučnímu návrhu jednoduchých polymorfních komponent (nejčastěji hradel), provedl autor v této kapitole na základě publikovaných prací a vlastní zkušenosti formální definování polymorfního obvodu. Dále je v této kapitole diskutováno, proč je vhodné navrhovat polymorfní obvody na úrovni hradel (jako struktury s polymorfními komponenty). Zbytek kapitoly je věnován charakteristickému rysu polymorfních obvodů, totiž reakci na prostředí. Je popsáno, co může být prostředím z pohledu polymorfní elektroniky, jak prostředí ovlivňuje a determinuje funkci číslicového obvodu a jaké veličiny charakterizující prostředí se typicky v polymorfní elektronice využívají. Je diskutováno, jak vliv prostředí obohacuje číslicovou elektroniku, v čem naopak situaci komplikuje, a jsou identifikovány dva základní způsoby využití polymorfismu obvodů: obvod buď přirozeně reaguje na změny prostředí (adaptace) nebo



je prostředí uměle ovlivňováno s cílem ovlivnit chování obvodu (netradiční způsob šíření signálu v obvodě).

Kapitola 3 je věnována komponentám polymorfních obvodů – polymorfním hradlům. Protože jde o obvody na úrovni tranzistorů a jejich parametry jsou silně závislé na parametrech tranzistorů a jejich zapojení, je nejprve stručně představen unipolární tranzistor a důležité parametry, použití unipolárního tranzistoru ke konstrukci číslicových logických prvků. Dále je uvedena definice polymorfního hradla a jsou stanoveny požadavky kladené na polymorfní hradlo s ohledem na jeho konkurenceschopnost vůči konvenčním hradlům. Následuje přehled známých polymorfních hradel, u každého jsou vždy shrnuty jeho vlastnosti a parametry a jsou ohodnoceny z hlediska ideálních parametrů. Velký prostor je věnován popisu a experimentům s hradly navrženými na pracovišti autora (na jejichž návrhu se autor též podílel nebo je sám navrhl), zejména polymorfnímu hradlu NAND/NOR řízenému napájecím napětím. Toto hradlo bylo již několikrát fyzicky realizováno na čipu, jsou proto popsána základní měření hradla a naměřené parametry, jeho chování v mezních situacích, detailně se autor věnuje experimentům se změnou funkce hradla a jevům, které změnu provází. Dále jsou popsány experimenty se změnou teploty hradla a reakcí na ně. Výsledky experimentů jsou z pohledu polymorfní elektroniky velmi významné a ukazují, že hradlo, ač původně navrženo jako řízené úrovní napájecího napětí, může být též úspěšně využito pro polymorfní obvody řízené teplotou. Téměř všechna dnes známá polymorfní hradla jsou navržena jako struktury z komplementárních unipolárních tranzistorů (CMOS). Koncept polymorfní elektroniky je však na vlastním řešení komponent nezávislý. V budoucnu bude možné jej využít i pro realizaci obvodů pokročilejšími technologiemi. Proto je na konci kapitoly 3 zařazeno stručné představení multifunkčního grafenového hradla, vyrobeného nanotechnologickými postupy, jehož využití v polymorfních obvodech je také možné.

Problém syntézy polymorfního (vícefunkčního) obvodu z komponent (hradel) a dosud známá řešení tohoto problému představuje kapitola 4. Formálně se jedná o hledání jediného grafu, který by popisoval obvod schopný realizovat více funkcí, přičemž změny funkce se dosahuje pouze změnou funkce (některých) použitých komponent. V současné době jsou jednoduché polymorfní obvody navrhovány buď přímo ad hoc, nebo (nejčastěji) jsou výsledkem evolučního návrhu. Pro návrh číslicových obvodů (nejen polymorfních) se velmi osvědčuje metoda Kartézského genetického programování, která je také stručně představena. Z. Gajda ve své zcela nedávno obhájené disertační práci navrhuje dvě inovativní metody syntézy polymorfních (vícefunkčních) obvodů, které lze označit za konvenční, protože nevyužívají evolučního algoritmu ani jiných heuristik. Jde o metodu tzv. polymorfního multiplexování a metodu PolyBDD (založenou na binárních rozhodovacích stromech). Gajdovy metody produkují výsledky plně funkční, avšak ne vždy zcela optimální z pohledu velikosti obvodů (ceny). Proto i Gajda doporučuje následnou optimalizaci výsledku např. evolučním algoritmem.

V kapitole 5 je popsán první vyrobený konfigurovatelný obvod s polymorfními hradly, který může být využit pro návrh polymorfních obvodů evolučními technikami a ověření (jakkoli navržených) polymorfních obvodů fyzickou realizací. Na návrhu obvodu se podílel autor práce a s obvodem provedl řadu experimentů pro ověření vlastností samotného konfigurovatelného obvodu i řadu realizací různých aplikací polymorfních obvodů popsaných v různých částech této práce. Konfigurovatelný polymorfní obvod REPOMO byl navržen na základě zkušeností s konfigurovatelnými obvody komerčních výrobců i zkušeností s návrhem konfigurovatelných číslicových obvodů pro speciální použití. Ty jsou na začátku kapitoly 5 stručně shrnuty. Pro obvod REPOMO dále autor navrhl experimentální platformu, která podporuje snadnou změnu funkce polymorfních hradel uvnitř obvodů různými způsoby (ručně i automatizovaně), snadnou konfiguraci obvodu (ručně i automatizovaně), generování stimulů pro obvod a snímání odezev na ně, předzpracování výsledků generovaných obvodem a dále změnu teploty čipu v širokém rozsahu. Tato platforma je také v kapitole 5 popsána.

Kapitola 6 je zaměřena na aplikace polymorfní elektroniky. Jsou popsány aplikace, které pro polymorfní elektroniku navrhuje tým A. Stoicy, hlavní pozornost je však věnována aplikacím, navrženým a ověřeným na pracovišti autora a aplikacím, na nichž se autor podílel. Je představena aplikace polymorfní elektroniky v diagnostice číslicových obvodů, kde použití polymorfních hradel pomáhá snížit náklady na testování. Dále je představen koncept samotestujících se sčítaček, u nichž použití polymorfních hradel vede na úsporu plochy čipu. Informace o chybné činnosti některé z jednobitových sčítaček je navíc elegantně propagována přes signály přenosu. Polymorfní obvody mohou být také užity ke zvýšení odolnosti obvodu vůči poruchám. To dokládá aplikace polymorfní elektroniky v oblasti číslicových filtrů. Filtr s polymorfními hradly může v případě nepříznivých podmínek samočinně omezit svoji spotřebu při zachování rozumné kvality výstupu. Dále je představena aplikace polymorfních obvodů v bezpečnosti a kryptografii. Je popsán návrh realizace fyzické neklonovatelné funkce s využitím polymorfních obvodů. Navržená fyzická neklonovatelná funkce je výhodná z hlediska ceny implementace a experimenty na reálných čípech ukazují slibnou účinnost.

Kapitola 7 představuje možnosti uplatnění polymorfní elektroniky v sekvenčních číslicových obvodech. Je možné stavět sekvenční číslicové obvody z polymorfních klopných obvodů (klopných obvodů schopných měnit svoji funkci vlivem prostředí), jak ukazuje popsáný návrh A. Stoicy a aplikace jím navrženého polymorfního klopného obvodu v prostředí řízeném reverzibilním čítači s nulováním. Tento přístup však naráží na neexistenci dostatečné škály polymorfních klopných obvodů (které je ovšem patrně možné navrhovat evolučně) a především na neexistenci postupů syntézy polymorfních kombinačních obvodů z polymorfních klopných obvodů. Autor práce proto navrhl metodu návrhu sekvenčních polymorfních obvodů s využitím konvenčních klopných obvodů a přidané polymorfní kombinační logiky. V tomto případě je možné úspěšně použít známé metody návrhu, které prezentuje kapitola 4 této práce. Autor dále popsal svoji metodu návrhu sekvenčních řadičů pro adaptivní a proti poruchám odolné obvody a demonstroval funkčnost na příkladech realizovaných řadičů a s nimi provedených experimentech.

Kapitola 8 shrnuje poznatky a informace popsané v této práci a zamýšlí se nad možnostmi a perspektivou polymorfní elektroniky v budoucnosti.

## 2 Polymorfní elektronika

### 2.1 Vymezení pojmu

Pod pojmem „polymorfní elektronika“ se v této práci rozumí oblast elektroniky, zkoumající *číslicové elektronické obvody*, které dokážou plnit *více než jednu funkci*, přičemž jejich *zapojení zůstává stále stejné* a aktuálně prováděná *funkce závisí na stavu okolního prostředí*. Všechny funkce jsou do obvodu zabudovány úmyslně již při jeho návrhu, počítá se s nimi. Jde tedy o smysluplné funkce, nikoliv například o poruchový stav vyvolaný tím, že jsou překročeny provozní parametry obvodu. Stav okolního prostředí lze popsat exaktně nejčastěji hodnotou nějaké fyzikální veličiny, pro konkrétní stav prostředí lze jednoznačně určit, jakou funkci bude obvod plnit.

Charakteristické pro polymorfní obvody je, že:

- 1 polymorfní obvod je svými nároky na prostor (počet tranzistorů, plocha čipu) výrazně skromnější, než kdyby bylo konvenčními postupy navrženo několik monofunkčních obvodů a ty se podle stavu prostředí přepínaly,
- 2 přepínání mezi funkcemi se děje přirozeně a okamžitě, není třeba čekat například na dokončení rekonfigurace nebo nějakého podobného sekvenčního děje, kterým by změna funkce obvodu byla podmíněna,
- 3 detekce stavu prostředí je přirozenou součástí obvodu, je strukturálně (na úrovni tranzistorů) provázána s jeho dalšími funkcemi,
- 4 detekce stavu prostředí je distribuována, většinou nelze jednoznačně lokalizovat část obvodu, která by byla určena pouze k detekci stavu prostředí a od níž by byla informace o stavu prostředí rozvedena k ostatním funkčním blokům obvodu.

Uvedené charakteristické vlastnosti zároveň ukazují výhody polymorfní elektroniky a nastiňují důvody, pro které se polymorfní elektronika jeví jako perspektivní výzkumná oblast.

Jak bylo uvedeno výše, polymorfní elektronika zkoumá číslicové elektronické obvody, kde funkce obvodu je jednoznačně definována, chování obvodů (stavy výstupu pro konkrétní data na vstupech) lze popsat s využitím booleovy algebry. V takových případech lze jednoznačně říci, zda obvod plní funkci  $F_1$  či jinou funkci (například  $F_2$ ).

Číslicový obvod lze popsat jako acyklický graf  $G = (V, E, \varphi)$ , kde  $V$  je množina vrcholů (bran komponent obvodu),  $E = \{(a,b) \mid a,b \in V\}$  je množina hran (spojů obvodu) a  $\varphi$  je zobrazení, které každému vrcholu  $z \in V$  přiřazuje komponentu  $z$  množiny  $K$ ,  $\varphi: V \rightarrow K$  (lépe řečeno, definuje ekvivalenční třídy vrcholů-bran náležících jedné obvodové komponentě – podrobněji formálně například v [Ruz07a]).

#### **Definice 2.1** (polymorfního obvodu)

Polymorfní obvod je číslicový elektronický obvod, který lze popsat grafem  $G$ , určujícím jednoznačné propojení komponent  $z$  množiny  $K$ , přičemž tento obvod je schopen realizovat v každém okamžiku jednu ze smysluplných funkcí  $z$  množiny  $\Phi = \{F_1, \dots, F_n\}$  a  $|\Phi| > 1$ .

Nechť  $X$  je fyzikální veličina, nabývající hodnot  $z$  oboru reálných čísel  $\mathbf{R}$ , popisující prostředí, v němž obvod pracuje. Existuje zobrazení  $\pi: Y \rightarrow \Phi$ , kde  $Y = \{I_i \mid I_i \subset \mathbf{R}\}$  je nějaká množina

intervalů hodnot veličiny  $X$ . Jestliže v čase  $t_1$  tato veličina nabývá hodnoty  $X(t_1) \in I_k$ , kde  $I_k \subset \mathbf{R}$  je interval na množině reálných čísel, pak obvod popsaný grafem  $G$  v čase  $t_1$  realizuje funkci  $F_k \in \Phi$ , stručně  $\pi(I_k) = F_k$ . Jestliže v čase  $t_2$  veličina  $X$  nabývá hodnoty  $X(t_2) \in I_m$ , kde  $I_m \subset \mathbf{R} \wedge I_m \cap I_k = \emptyset$ , pak obvod popsaný grafem  $G$  v čase  $t_2$  realizuje funkci  $F_m \in \Phi$ , stručně  $\pi(I_m) = F_m$ .

□

### Poznámka 2.1 k definici 2.1

Prostředím, zmíněným v definici 2.1 a vybírajícím jednu z  $|\Phi|$  funkcí, která je právě obvodem realizována, může být například úroveň napájecího napětí, teplota čipu, úroveň osvětlení nebo podobně. Někdy za „prostředí“ může být považováno i napětí na speciálním vodiči – signál, který je do obvodu přiveden. Obecně lze na prostředí pohlížet jako na další vstup (nebo vstupy) logického obvodu, který ovlivňuje hodnotu výstupu, přitom však nejde o klasické logické signály.

□

### Poznámka 2.2 k definici 2.1

Mohou existovat takové intervaly hodnot veličiny  $X$ , pro které funkce obvodu není definována.

□

### Důsledek definice 2.1

Protože z definice 2.1 vyplývá, že struktura obvodu – graf  $G$ , tedy způsob propojení jednotlivých komponent se nemění, přitom se ale může se změnou prostředí změnit funkce obvodu, musí svoje chování se změnou prostředí měnit komponenty obvodu.

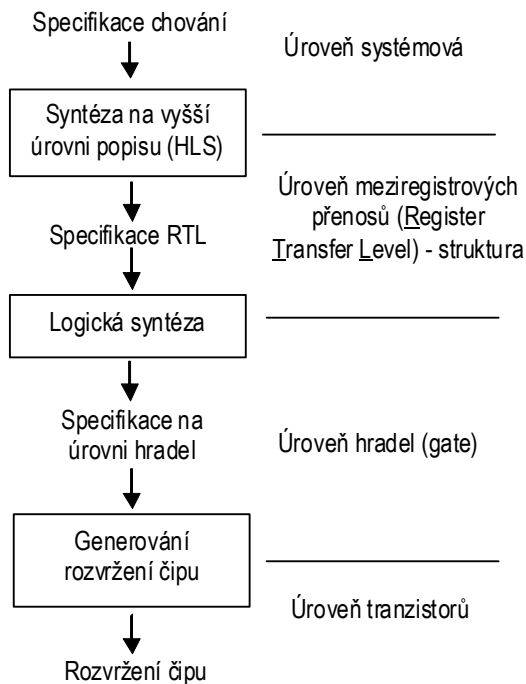
□

Pro polymorfni elektroniku je tedy klíčové:

- nalézt vhodné komponenty, které by dokázaly předvídatelně a deterministicky reagovat na změnu prostředí a
- nalézt postup, kterak z těchto komponent sestrojít jediný graf  $G$  tak, aby v žádaných režimech činnosti těchto komponent (pro relevantní stav prostředí) obvod reprezentovaný grafem  $G$  realizoval vždy konkrétní požadovanou funkci.

## 2.2 Komponenty pro polymorfni obvody

V současnosti už téměř není možné při návrhu složitých obvodů postupovat ručně a navrhnout celou strukturu obvodu nakreslením schématu obvodu postaveného z jednotlivých komponent, které lze vytvořit na čipu - tranzistorů. Proto vzniká řada automatizovaných návrhových systémů a nástrojů podporujících návrh číslicových obvodů. Charakteristická je hierarchie úrovní popisu struktury obvodu. Typicky se z pohledu struktury rozlišují úroveň systémová, úroveň meziregistrových přenosů, úroveň hradel a úroveň tranzistorů (viz obrázek 2.1). [Mae92], [Lee97], [Wak00].



Obrázek 2.1: Úrovně popisu obvodu

Má-li se naplnit požadavek na co nejmenší nároky na prostor obvodu, který dokáže plnit více než jednu funkci, aniž by se změnilo jeho zapojení (je tedy sestaven z vícefunkčních komponent – viz důsledek definice 2.1), je vhodné volit komponenty na co nejnižší úrovni granularity (obrázek 2.1).

Z toho plyne, že nejlepších výsledků (z pohledu nároků na velikost obvodu) lze dosáhnout při návrhu na úrovni tranzistorů. Bylo by možno říci, že na této úrovni se postupuje při návrhu polymorfních hradel (viz kapitola 3). Při návrhu na úrovni tranzistorů se však nejedná o změnu funkce logického obvodu, protože tranzistor není číslicovým prvkem v pravém smyslu slova, chová se analogově (podrobněji pojednáno v kapitole 3).

Nejnižší vhodnou úrovní pro návrh<sup>1</sup> (synézou) vícefunkčních (tedy i polymorfních ve smyslu definice 2.1) číslicových obvodů je proto úroveň hradel. Hledají se taková hradla, která dokážou realizovat více než jednu smysluplnou logickou funkci v závislosti na prostředí.

Typickými nevýhodami (nejčastěji plynoucími z charakteru polymorfní elektroniky) jsou:

- Základní stavební prvky polymorfních obvodů, polymorfní hradla, nejsou k dispozici. Je třeba je navrhout a vyrobit, aby bylo možno s polymorfními obvody dále experimentovat. Tuto problematiku řeší kapitola 3 této práce.
- Návrhové metody, které by umožnily navrhovat cíleně polymorfní obvody, nejsou k dispozici. Je třeba takové metody vyvinout. Tuto problematiku řeší kapitola 4 této práce.
- Kompaktní rozměry obvodu jsou vyváženy zpravidla zhoršením jiných parametrů (rychlost, spotřeba). Jde o kompromis.

<sup>1</sup> Návrhem (syntézou) číslicového (logického) obvodu se rozumí hledání takového grafu  $G$ , který plní zadanou logickou funkci  $F$ . Funkce  $F$  přitom pracuje s vektory složenými z logických hodnot – vektorem hodnot na vstupu, případně také vektorem popisujícím stav (v případě sekvenčních logických obvodů).

## 2.3 Historie

Pojem „Polymorfni elektronika“ ve smyslu oboru zabývajícího se elektronickými obvody, které mění svoji funkci jako přirozenou reakci na změnu podmínek či prostředí, v němž pracují, bez přepínání či rekonfigurace, použili poprvé A. Stoica, R. Zebulum a D. Keymeulen. V roce 2001 vyšel jejich článek nazvaný „Polymorphic Electronics“ na konferenci „Evolvable Systems: From Biology to Hardware“ [Sto01]. Tento článek lze považovat za první veřejnou zmínku o polymorfni elektronice, jak ji chápe tato práce. V článku [Sto01] se však autoři odkazují na výzkumnou zprávu NASA týmu vedeného A. Stoicou nazvanou „Polymorphic electronics – A novel type of circuits with multiple functionality“ z června 2000. Myšlenka polymorfni elektroniky pochází právě z této doby.

Autoři v článku [Sto01] popisují polymorfni elektroniku jako nové paradigma obvodů s přidáním dalšími funkcemi, přičemž změna funkce nepřichází s přepínáním výstupů obvodů navržených pro jednotlivé požadované funkce či rekonfigurací obvodu, ale je vestavěnou součástí obvodu již od jeho návrhu a je způsobena spíše změnou charakteristik použitých prvků a jejich pracovních bodů. Autoři také navrhují zkrátit pojem „Polymorphic Electronics“ na „Polytronics“.

Ve své výzkumné zprávě z dubna 2004 nazvané „Polymorphic Electronic Circuits“ [Sto04a] titíž autoři naopak uvádí, že polymorfni elektronika je součástí širší technologické disciplíny – polytroniky (angl. polytronics), která je důsledkem evolvable hardware. Polytronika podle této výzkumné zprávy je disciplína zahrnující nejen elektronické, ale i optické a potenciálně další multifunkční systémy na zpracování informací.

Podle [Sto04a] předcházely definování polytroniky experimenty s evolučním návrhem obvodů, které by svojí různou funkcí při různých teplotách kompenzovaly vliv teploty na současné polovodičové prvky (výzkumná zpráva nazvaná „EHW Approach to Temperature Compensation of Electronics“. Obrovský potenciál návrhu elektronických obvodů technikou evolvable hardware (EHW) [Sek06] dovolil hledat obvodová řešení, která by konvenčního návrháře konvenčními technikami návrhu ani hledat nenapadlo.

Je třeba podotknout, že pojem „Polytronics“ se ve významu, který mu přikládají výzkumníci z týmu A. Stoicy, příliš nepoužívá. V současnosti se pojem „Polytronics“ vyskytuje spíše v souvislosti s elektronickými obvody postavenými na bázi polymerů (též polymerní elektronika, organická elektronika) [Mar05].

Tým A. Stoicy, který se primárně (v souladu s posláním jejich domovského pracoviště – NASA Jet Propulsion Laboratory) zabývá výzkumem elektronických technologií pro dlouhé nepilotované kosmické mise, kde je žádoucí odolnost proti poruchám, jejich kompenzace, automatické přizpůsobení se tvrdým a extrémním podmínkám, navrhl novou oblast – polymorfni elektroniku – při experimentech s evolučním návrhem spolehlivých a odolných obvodů pro extrémní prostředí. Výzkumníci z týmu A. Stoicy spatřili velký potenciál aplikací obvodů z této nové oblasti přesahující výzkumné zájmy dané posláním jejich domovského pracoviště a naznačili jej ve svých publikacích.

Ve výzkumu polymorfni elektroniky a experimentech s praktickými realizacemi polymorfni obvodů pokračuje ve světě dále několik dalších výzkumných týmů. Lze říci, že nejúspěšnějším co do počtu a kvality výsledků výzkumu je zatím výzkumný tým z pracoviště autora této práce.

Současný pohled na oblast polymorfni elektroniky je popsána zejména v části 2.1 této práce včetně autorem vytvořené formální definice polymorfni elektroniky (definice 2.1).

## 2.4 Prostředí

Jak již bylo uvedeno v části 2.1 této práce, jedním z význačných charakteristických rysů polymorfních obvodů je změna funkce podle stavu okolního prostředí obvodu. Formální definice polymorfního obvodu (definice 2.1) počítá se stavem prostředí vyjádřeným hodnotou nějaké fyzikální veličiny. Fyzikální veličiny jako teplota nebo napětí jsou vyjádřeny hodnotou z oboru reálných čísel. Protože se jedná o polymorfní číslicové obvody, není možné, aby funkce obvodu byla funkcí reálné hodnoty (tj. měnila se spojitě). Jsou tedy definovány intervaly hodnot veličiny popisující prostředí obvodu, pro které platí, že je-li aktuální hodnota veličiny popisující prostředí uvnitř tohoto intervalu, obvod realizuje určitou konkrétní funkci.

Pro každou možnou funkci číslicového polymorfního obvodu je definován interval hodnot veličiny popisující prostředí obvodu. Formálně je tento vztah popsán zobrazením  $\pi$  (viz definice 2.1).

Zobrazení  $\pi: Y \rightarrow \Phi$ , kde  $Y = \{I_i \mid I_i \subset \mathbf{R}\}$  je nějaká množina intervalů hodnot veličiny popisující prostředí obvodu a  $\Phi = \{F_i \mid i = (1, n) \wedge n > 1\}$  je množina funkcí, které je obvod schopen realizovat.

Musí platit, že:  $\forall I_j, I_k \in Y: I_j \cap I_k = \emptyset \leftrightarrow j \neq k$ , tedy že všechny intervaly z  $Y$  jsou navzájem disjunktní, protože číslicový obvod v jednom okamžiku může vykonávat pouze jednu funkci. Veličina popisující prostředí přitom má konkrétní hodnotu, která musí spadat do nejvýše jednoho intervalu.

Naopak, jak praví poznámka 2.2, ne pro každou možnou hodnotu veličiny popisující prostředí musí být funkce obvodu definována, tzn. intervaly z množiny  $Y$  na sebe nemusí bezprostředně navazovat. Odpovídá to praktické situaci, kdy při změně funkce růstem nebo poklesem veličiny popisující prostředí může existovat nějaký přechodný stav, kdy obvod nerealizuje žádnou smysluplnou funkci. Takové chování sice není ideální, v praxi však u některých polymorfních obvodů může nastat zejména s ohledem na požadavek kompaktních rozměrů polymorfních obvodů (blíže viz charakteristika polymorfních obvodů č. 1 v části 2.1).

V konvenční elektronice by k podobnému chování – změně funkce na základě změny hodnoty nějaké fyzikální veličiny – bylo třeba přidat do obvodu senzor příslušné veličiny, dále vyhodnocovací obvod a také nějaký přepínač funkcí, případně distribuovat informace o vyhodnoceném stavu k těm komponentám, u kterých bude funkce přepínána. U polymorfních obvodů tomu tak není. Naopak, využívá se citlivosti samotných obvodových prvků na stav prostředí (příslušnou fyzikální veličinu). Jak praví důsledek definice 2.1, změna funkce se odehrává v samotných komponentách obvodu (konkrétně v polymorfních hradlech). Senzor příslušné fyzikální veličiny je tak svým způsobem distribuován. To přináší výhody (odpadají globální rozvody informace o stavu prostředí, které bývají v systémech na čipu nákladné či obtížně realizovatelné, obvod se stává kompaktnějším, prostorově méně náročným, levnějším), ale i nevýhody (přepnutí funkce nemusí nastat u všech komponent současně, což na jedné straně může mít za následek vznik nedefinovaných či hazardních stavů, na straně druhé však vhodným návrhem může být tento jev potlačen či dokonce využit – viz aplikace v kapitole 6).

Na využití nejvýznačnějšího rysu polymorfních obvodů – změnu funkce podle stavu prostředí, lze v zásadě použít dva protichůdné pohledy (způsoby využití polymorfismu obvodů):

- Z1 Polymorfní obvod je navržen tak, aby změnu funkce vyvolanou změnou prostředí využil ve prospěch zařízení, v němž pracuje – kompenzoval vliv teploty, kolísajícího napětí a podobně.
- Z2 Prostředí je měněno záměrně s cílem vyvolat změnu funkce obvodu. Využívá se toho, že informace o požadované funkci není třeba distribuovat po čipu, případně další funkce jsou při pouhém pohledu na obvod a jeho zapojení skryty, nejsou v prvním plánu a běžně se neaktivují (skrytá funkce, fyzická neklonovatelná funkce, diagnostická funkce – viz aplikace popsané v kapitole 6).

Jak již bylo naznačeno v části 2.3, k objevení polymorfní elektroniky došlo hledáním obvodů (s využitím evolučních technik), které by se chovaly podle pohledu Z1. Objevením polymorfní elektroniky se však následně otevřel i pohled Z2.

## 2.5 Známé způsoby řízení funkce polymorfního obvodu

### 2.5.1 Řízení napájecím napětím

Řízení funkce polymorfního obvodu napájecím napětím je jedním z nejtypičtějších způsobů ovlivňování funkce polymorfních obvodů. Napájecí napětí je analogová veličina (nabývá hodnot z oboru reálných čísel), která je přirozeně přítomná v každém číslicovém obvodu. Její vliv je ovšem v konvenčních číslicových obvodech záměrně potlačen. Naopak u polymorfních obvodů je její využití velice přirozené. Jedná se o elektrickou veličinu, lze tedy ji samotnou snadno v obvodě ovlivňovat v případě, kdy se polymorfních obvodů využívá jako elegantně rekonfigurovatelných obvodů (způsob využití Z2). Informace o napájecím napětí je totiž zcela přirozeně přítomna téměř v každém místě obvodu, není třeba žádného dalšího rozvodu signálu nesoucího informaci o požadované funkci. Druhý pohled (způsob využití Z1) je v případě napájecího napětí také velmi aktuální. Vhodně navržený obvod může změnou funkce reagovat na pokles či naopak vzrůst napájecího napětí, jenž často vypovídá o stavu zdroje, a tak zamezit havárii obvodu z důvodu nedostatku energie zdroje nebo alespoň následky havárie zmírnit. Obvod může například při poklesu napájecího napětí vhodně redukovat svoji činnost na jen nejnnutnější kroky či omezit svoji přesnost a tak zajistit alespoň nejnnutnější funkce při nižší spotřebě energie – více o tomto způsobu využití polymorfní elektroniky včetně příkladů a praktických experimentů je v kapitole 6 a 7 této práce.

Číslicové polymorfní obvody, jejichž funkce je řízena napájecím napětím, jsou ve skutečnosti alespoň částečně obvody analogovými. Týká se to zejména komponent obvodu – hradel, vlastní struktura obvodu je navrhována čistě dle zásad číslicového návrhu (ovšem postupy návrhu se „vícefunkčností“ obvodu komplikují – podrobněji viz kapitola 4). Polymorfní hradla, jejichž funkce je řízena napájecím napětím, pokud jsou realizována konvenčními polovodičovými technologiemi a sestávají z běžných tranzistorů, jsou založena na tom, že alespoň některé z tranzistorů pracují v lineárním režimu, kde je pracovní bod velmi závislý na napájecím napětí. To je jejich hlavní odlišnost od konvenčních hradel, kde tranzistory pracují přísně v saturačním režimu. Daní za „vícefunkčnost“ polymorfních hradel řízených napájecím napětím je pak pomalejší činnost a větší spotřeba v porovnání s konvenčními hradly. Podrobně o této problematice pojednává kapitola 3 této práce.

### 2.5.2 Řízení teplotou

Citlivosti současných polovodičových technologií založených na křemíku na teplotu křemíkového čipu lze s výhodou využít. Zatímco u konvenčních číslicových obvodů se vliv



teploty záměrně potlačuje (cílem je dosáhnout stále stejné – „té pravé“ – funkce obvodu v co nejširším teplotním rozsahu, protože obvody se zejména vlastní činností ohřívají), u polymorfních obvodů se vliv teploty nechá na obvod vhodně zapůsobit – obvod je třeba navrhnout tak, aby se pracovní bod některých vybraných tranzistorů s rostoucí teplotou vhodně posunul. Princip je v zásadě totožný s návrhem polymorfních obvodů řízených napájecím napětím. Proto (a experimenty to potvrzují) jsou polymorfní hradla řízená napájecím napětím citlivá na teplotu a naopak. Změnu funkce (naštěstí na stejnou smysluplnou funkci v obou případech) lze proto u hradel vyvolat oběma způsoby.

Protože teplota je neelektrická veličina, přichází u polymorfních obvodů řízených teplotou v úvahu hlavně způsob využití Z1. Jen těžko si lze představit, že by kvůli změně funkce obvodu (rekonfiguraci) ve smyslu způsobu využití Z2 byl obvod záměrně zahříván. Zařízení na ohřívání čipu a regulaci jeho teploty (aby obvod udržel požadovanou funkci) by patrně bylo natolik komplikované, že by převážilo výhody polymorfní elektroniky (takové zařízení bylo pro účely experimentování s polymorfními obvody přesto vyvinuto – viz kapitola 5).

Je vhodné připomenout, že kompenzace vlivu teploty změnou funkce obvodu za účelem zvýšení spolehlivosti obvodu stála u zrodu polymorfní elektroniky. V obdobných aplikacích mají polymorfní obvody řízené teplotou jistý potenciál využití [Sto01].

### 2.5.3 Řízení speciálním signálem

Autoři myšlenky polymorfní elektroniky navrhli také možnost řízení funkce polymorfních obvodů nějakým speciálním elektrickým signálem. Z hlediska definice je řídicí veličinou napětí na speciálním signálu. V případě, že  $|\Phi| = 2$ , tedy že polymorfní obvod je schopen provádět právě dvě různé funkce, může být polymorfní obvod chápán také jako konvenční číslicový obvod s jedním vstupem navíc, protože řídicí signál je platný v jednom ze dvou intervalů napětí podobně jako běžný logický signál. V případě tří a více funkcí vybíraných (jedním) speciálním signálem však už obvod vykazuje všechny rysy polymorfních obvodů. Některé tranzistory musí pracovat v lineárním režimu, aby mohly reagovat na různé napěťové úrovně.

Polymorfní obvody řízené speciálním signálem mají také svoje výhody a svoje potenciální aplikace. Především je možné signál měnící funkci generovat opět v nějakém elektronickém obvodu a využít tak jednu z hlavních vlastností polymorfní elektroniky – přirozenou a okamžitou změnu funkce kompaktního obvodu bez rekonfigurace, či redundance a přepínání. Nebo je možné připojit jakýkoliv sensor, který převádí neelektrickou veličinu na elektrickou, a tak vytvořit obvod, který reaguje i na veličinu, na kterou nejsou přímo k dispozici polymorfní komponenty.

## 3 Polymorfní hradla

### 3.1 Hradlo jako základní stavební prvek kombinačních číslicových obvodů

Je-li množina  $B = \{0, 1\}$  množinou hodnot, které může nabývat logická proměnná, pak funkce  $f: B^n \rightarrow B$ , je tzv. boolovská nebo též logická funkce. Číslo  $n \leq 1$  je celé číslo. Některé funkce jsou význačné a užívají se často, mají proto své pojmenování (negace – NOT, logický součet – OR, logický součin – AND, a podobně).

Hradlo nebo též logický člen je základním stavebním prvkem kombinačních číslicových obvodů. Realizuje logickou (boolovskou) funkci  $f$ . Je to jednoduchý kombinační obvod, pro nějž je charakteristické, že má jediný výstup (který může nabývat hodnot log. 0 nebo log. 1), vstup může být jeden nebo několik.

Logické hodnoty jsou nejčastěji reprezentovány napětím. Drtivá většina systémů je navržena tak, že hodnota log. 0 je reprezentována potenciálem blízkým potenciálu záporného pólu zdroje. Jestliže je (a opět - drtivá většina systémů to tak má) záporný pól zdroje vzat jako referenční (tzv. zem), pak hodnota log. 0 je reprezentována nulovým napětím. Hodnota log. 1 je pak reprezentována potenciálem blízkým potenciálu kladného pólu zdroje, což (v systému s uzemněným – referenčním – záporným pólem zdroje) znamená, že hodnota log. 1 je reprezentována napětím blízkým napájecímu napětí.

Typicky je dnes hradlo realizováno jako elektronický obvod. Úplný soubor logických funkcí lze realizovat hradly sestavenými z tranzistorů. Je známa celá řada technologií realizace hradel z tranzistorů, mezi ty nejznámější patří technologie TTL, rozšířená zejména díky úspěchu řady integrovaných obvodů 74XX firmy Texas Instruments. První obvody této řady byly uvedeny na trh již v první polovině 60. let a prakticky definovaly průmyslový standard, užívaný dodnes, ať už se jedná o napěťové úrovně reprezentující logické hodnoty, napájení číslicových systémů nebo nabízené obvody a jejich uspořádání v pouzdře. Hradla i složitější obvody v technologii TTL sestávají z bipolárních tranzistorů. Tím je dána i většina omezení logických obvodů realizovaných technologií TTL. Patří mezi ně poměrně úzké pásmo napájecích napětí (typicky kolem 5V), vlastní spotřeba, definování pásem napěťových úrovní pro reprezentaci logických hodnot nebo tzv. zisk hradel (vyjádřený počtem vstupů následných obvodů, kterými lze zatížit jeden výstup).

Další velmi rozšířenou technologií výroby logických obvodů je technologie CMOS. Pro logické obvody v technologii CMOS je typické, že se využívá komplementární dvojice unipolárních tranzistorů – jeden je typu p-mos a druhý je typu n-mos. Z toho pochází i název technologie, který je zkratkou anglického názvu „Complementary Metal-Oxide Semiconductor.“

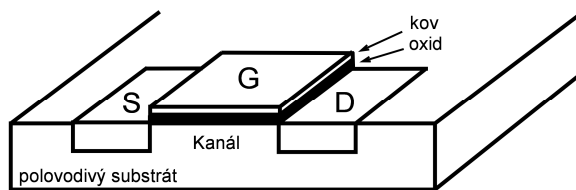
### 3.2 Unipolární tranzistor typu MOS

Dominantním materiálem pro výrobu aktivních součástí je dnes křemík. Je to prvek se čtyřmi valenčními elektrony, v krystalové mřížce čistého křemíku tyto čtyři elektrony tvoří právě vazbu se čtyřmi sousedními atomy. Nezbyvají žádné volné nosiče náboje. Proto je čistý křemík špatným vodičem. Vodivost lze zvýšit přidáním malého množství příměsí. Pokud se přidá prvek

s pěti valenčními elektrony, například arsen, budou jeho čtyři valenční elektrony tvořit vazbu s okolní čtveřicí atomů křemíku a pátý lze malou energií odpoutat od "svého" atomu, aby se z něj stal nosič náboje volně se pohybující krystalovou mřížkou. Získá se polovodič s elektronovou vodivostí, tzv. polovodič typu N. Pokud se přidá prvek se třemi valenčními elektrony, například bór, bude jeden elektron v jedné ze čtveřic vazeb atomů v mřížce chybět. Tento chybějící elektron si atom bóru může „vypůjčit“ od sousedního atomu v mřížce, ten zase od sousedního atomu atd. Tímto způsobem krystalovou mřížkou putuje „díra“, chybějící elektron, což lze považovat za nosič kladného náboje. Získá se polovodič s děrovou vodivostí, polovodič typu P. Zde jsou majoritními nosiči elektrického náboje díry, elektrony jsou nosiči minoritními. U polovodiče typu N je tomu právě naopak.

Nejjednodušší polovodičovou součástí je dioda. Dioda vznikne tam, kde je vytvořeno rozhraní, ostrý přechod mezi polovodičem typu N a P. Dioda se chová tak, že pokud napětí na straně P převyšuje napětí na straně N (o alespoň tzv. difúzní napětí, které se pro polovodiče založené na křemíku pohybuje kolem 0,7V), přes přechod teče elektrický proud (přechod je polarizován v propustném směru). Napětí opačné polaroty proud nezpůsobí, přechod je v tomto směru neprůchozí (přechod je polarizován v závěrném směru).

Unipolární (dnes nejčastěji MOS) tranzistor je „sendvičová“ struktura sestávající z vrstev polovodičů typu N a P a izolantu. Tato struktura je vytvořena sérií chemických procesů v krystalickém křemíku. Základem je monokrystal křemíku. Ten se dále lokálně upravuje difúzí atomů příměsí (aby vznikla místa s polovodivostí typu N či P) či oxidací (oxid křemíku je nevodivý, tedy oxidací vznikají místa, která izolují) a nanáší se na něj další vrstvy např. kovu pro vytvoření přívodních vodičů.

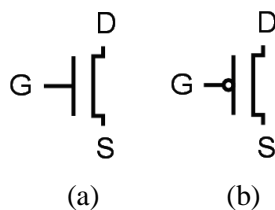


Obrázek 3.1: Struktura tranzistoru typu MOS.

Na obrázku 3.1 je vidět řez strukturou dvou možných typů tranzistorů typu MOS (Metal Oxide Semiconductor, tedy kov-oxid-polovodič). Podle toho, jaký typ vodivosti mají jeho hlavní elektrody, rozlišujeme n-MOS a p-MOS tranzistor. Jeho dvě hlavní elektrody se nazývají „Drain“ a „Source“. Jsou vždy silně dotované na opačnou vodivost, nežli je prostředí okolo nich, tzv. substrát. Mezi elektrodami D a S je tedy vždy alespoň jeden závěrně polarizovaný přechod PN a proto mezi nimi nemůže téct proud. Pokud ovšem v polovodiči mezi elektrodami D a S vytvoříme elektrické pole přivedením napětí mezi elektrodu G (z anglického „Gate“) a substrát, můžeme tímto elektrickým polem „přilákat“ do prostoru mezi elektrodami D a S dostatek minoritních nosičů náboje. Je třeba poznamenat, že proud mezi substrátem a elektrodou G díky izolující vrstvě oxidu křemíku téci nemůže. Pokud je elektrické pole dostatečně silné, překoná lokálně počet původně minoritních nosičů počet nosičů majoritních a polovodič v tomto místě změní svůj typ vodivosti na opačný. Pak bude vodivost prostoru mezi elektrodami D a S stejná jako vodivost vlastních elektrod, na cestě mezi elektrodami nebude žádný polovodičový přechod. Bude zde vodivá cesta, dostatek volných nosičů náboje stejné polaroty. Prostor mezi elektrodami D a S, kde dojde vlivem elektrického pole ke změně typu vodivosti polovodiče, se nazývá kanál. Tranzistor typu MOS je tedy součástí, kde změnou napětí na elektrodě G lze měnit vodivost mezi elektrodami D a S. Tranzistor typu MOS lze považovat za napětím řízený spínač.

Soustava elektroda G - substrát se bude chovat podobně jako kondenzátor - dvě elektrody (G a substrát) jsou navzájem izolované, v prostoru mezi nimi je elektrické pole. Proud poteče vždy jen při změnách napětí na těchto elektrodách, v klidu ale proud neteče (ve skutečnosti teče téměř zanedbatelný tzv. svodový proud, který je dán nedokonalostí izolační vrstvy).

Jestliže bude unipolární tranzistor typu n-MOS, bude to kladné napětí na elektrodě G (kladné vůči substrátu), které otevře vodivou cestu mezi elektrodami D a S. U tranzistoru typu p-MOS to je záporné napětí (vůči substrátu). Toto napětí musí mít určitou minimální velikost, aby vodivý kanál vznikl. Napětí, při kterém právě vzniká vodivá cesta mezi elektrodami D a S se nazývá „prahové napětí“. Kanál zpravidla i u plně otevřeného tranzistoru má nějaký (nenulový) elektrický odpor. Velikost odporu kanálu závisí na řadě faktorů, mezi nejvýznamnější patří fyzické rozměry kanálu (délka a šířka).

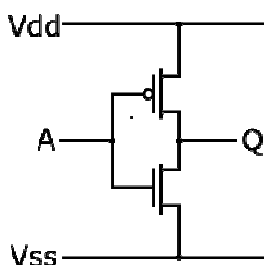


Obrázek 3.2: Schematické značky unipolárních tranzistorů typu MOS.

Obrázek 3.2 ukazuje schematické značky tranzistoru n-MOS (a) a p-MOS (b), obvyklé v případě, že tranzistory jsou užity v nějakém číslicovém obvodu a počítá se s tím, že budou pracovat buď plně otevřené nebo plně uzavřené. Schematické značky se liší tím, že tranzistor p-MOS má u elektrody G kroužek. Ten značí, že se tranzistor otevírá záporným napětím. Zejména pro analogové aplikace tranzistorů typu MOS se používají značky jiné. Polarita PN přechodů je tam znázorněna šipkou a kreslí se také čtvrtá elektroda, značící připojení substrátu. Při použití značek jako na obrázku 3.2 se předpokládá implicitní připojení substrátu k napájecímu zdroji obvodu, ale do schématu se toto připojení nekreslí.

### 3.3 Hradlo v technologii CMOS

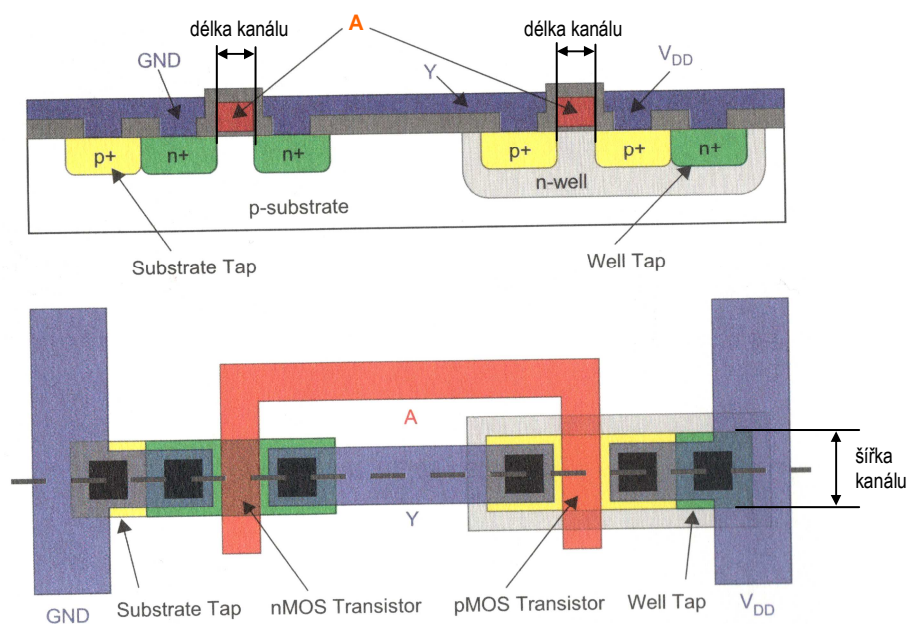
Technologie CMOS byla navržena již v roce 1963 pracovníkem firmy Fairchild Semiconductors (a v roce 1967 patentována jako US patent No. 3,356,858 [Wan67]). Typické hradlo, vyrobené technologií CMOS, obsahuje komplementární dvojici unipolárních tranzistorů v zapojení, jaké je vidět například na obrázku 3.3.



Obrázek 3.3: Invertor (hradlo NOT) v technologii CMOS.

Na obrázku 3.3, který ukazuje realizaci nejjednoduššího logického členu, invertoru, lze dobře demonstrovat roli, jakou komplementární dvojice unipolárních tranzistorů hraje. Horní tranzistor (typu p-MOS) se otevře právě tehdy, pokud je na vstupu A hradla úroveň log. 0 - potenciál blízký zápornému pólu zdroje  $-V_{ss}$ , který je zemí. V takovém případě se na výstupu Q hradla objeví potenciál blízký kladnému pólu zdroje  $-V_{dd}$ , což odpovídá hodnotě log. 1. Naopak, pokud na vstupu A hradla bude log. 1, otevře se spodní (n-MOS) tranzistor a na výstup Q tak propustí potenciál  $V_{ss}$ , což je interpretováno jako úroveň log. 0. Obvod z obrázku 3.3 tedy obrací logickou úroveň, realizuje logickou funkci jedné proměnné zvanou *inverze*, proto může být nazýván hradlem s funkcí NOT, invertorem.

Za povšimnutí stojí fakt, že každá ze dvou možných logických úrovní na vstupu hradla otevírá právě jeden tranzistor, druhý zůstává uzavřen. Každý z tranzistorů se v klidu nachází vždy v jednom ze dvou stavů: je buď plně otevřen, nebo plně uzavřen.



Obrázek 3.4: Skutečné provedení CMOS invertoru na křemíkovém čipu.

Na obrázku 3.4 (převzato z [Wes05]) je vidět fyzická realizace CMOS invertoru, jehož schéma je na obrázku 3.3. Červeně označený vodič je vstup invertoru A, který tvoří zároveň i elektrody G obou tranzistorů. Substrát (základní křemíková monokrystalická destička) je typu P a je připojen na zem ( $V_{ss}$ ). Tvoří zároveň i kanál tranzistoru n-MOS. Proto je možné tranzistor typu n-MOS otevřít úrovní log. 1. Pro tranzistor p-MOS je v substrátu vytvořena oblast s vodivostí typu N (angl. n-well) a ta je připojena na  $V_{dd}$ , aby bylo možno tranzistor typu p-MOS otevírat log. 0 na elektrodě G.

V obrázku 3.4 jsou vyznačeny dva důležité fyzické rozměry, které rozhodujícím způsobem určují jejich parametry. Jsou to šířka kanálu a délka kanálu tranzistoru. Vodivost kanálu v otevřeném stavu je přímo úměrná šířce kanálu a nepřímo úměrná délce kanálu. V režimu saturace pro proud kanálem  $I_{dsat}$  platí, že [Wes05]:

$$I_{dsat} = k \cdot (W/L) \cdot (V_{dd} - V_t)^2,$$

kde  $k$  je konstanta závislá na technologii,  $W$  je šířka kanálu,  $L$  je délka kanálu a  $V_t$  je prahové napětí, napětí, při kterém se kanál otevírá a začíná být vodivým.

Ve skutečnosti je vodivost tranzistoru n-MOS a p-MOS se stejnými rozměry kanálu odlišná – p-MOS má horší vodivost, proto se zpravidla jeho kanál musí dělat širší než by tomu bylo u n-MOS tranzistoru. Na obrázku 3.4 jsou však pro jednoduchost oba tranzistory stejně velké.

Hradlo, realizované technologií CMOS, má z elektrotechnického pohledu velmi zajímavé parametry:

- Díky nízkému odporu otevřeného unipolárního tranzistoru (kanálu mezi elektrodami D a S) a velmi vysokému odporu uzavřeného unipolárního tranzistoru (ve srovnání s bipolárními tranzistory) jsou výstupní napěťové úrovně velmi kvalitní (velmi se blíží potenciálu kladného, resp. záporného pólu zdroje). To mimo jiné způsobuje také vysokou šumovou imunitu hradla.
- Protože odpor elektrody G unipolárního tranzistoru je velmi vysoký (elektroda G je od kanálu izolována oxidem kovu, tranzistor se otevírá pouze elektrickým polem a nikoliv proudem, jako je tomu u bipolárního tranzistoru), zatěžuje vstup hradla výstupy jemu předcházejících členů naprosto minimálně.
- Vysoký vstupní odpor a fakt, že vždy je otevřen pouze jeden z komplementární dvojice tranzistorů způsobují, že obvody realizované technologií CMOS mají v klidu velmi nízkou spotřebu. Spotřeba vzroste pouze v okamžiku, kdy dochází ke změně úrovně – jeden z tranzistorů se právě otevírá a druhý právě uzavírá.
- Hradla jsou velice jednoduchá na výrobu a prostorově nenáročná. Samotný MOS tranzistor je velmi jednoduchá struktura a také hradla z takových tranzistorů realizovaná jich neobsahují mnoho. To umožňuje vysoký stupeň integrace se zachováním nízkých nákladů.

Výše uvedené parametry způsobily, že technologie CMOS se v průběhu 80. let stala dominantní technologií výroby (nejen) číslicových obvodů.

### 3.4 Polymorfni hradlo – definice a požadavky

#### Definice 3.1:

Polymorfni hradlo je elektronický obvod s  $k$  vstupy a jedním výstupem, který je schopen realizovat množinu logických funkcí  $\Phi = \{f_i: B^k \rightarrow B\}$ , kde  $|\Phi| > 1$ . Která z funkcí  $f_i$  je v daném okamžiku hradlem realizována, určuje jednoznačně stav prostředí. □

#### Poznámka 3.1 k definici 3.1:

Předpokládá se, že v jednom okamžiku hradlo realizuje pouze jednu konkrétní funkci. Výstup hradla je jen jeden a tudíž v jednom okamžiku může nabývat pouze jedné konkrétní hodnoty  $y$  z množiny  $B$ . Pokud je na vstupu vektor  $A \in B^k$ , pak právě  $y = f_i(A)$ . Někdy se také říká, že hradlo je v režimu  $f_i$ . □

#### Poznámka 3.2 k definici 3.1:

Z definice 3.1 vyplývá, že dvě různé funkce  $f_a, f_b \in \Phi$ , realizované jedním polymorfni hradlem, mají vstupní vektor o stejném počtu bitů. Mohou však existovat polymorfni hradla, u nichž v některých režimech stav jejich výstupu ovlivňují jen některé z  $k$  jejich vstupů, ty zbylé na stav výstupu vliv nemají. Takovou situaci lze popsat následujícím výrazem:

$\forall f: B^j \rightarrow B \exists f': B^{j+1} \rightarrow B$  taková, že  $\forall x_1x_2x_3\dots x_j \in B^j: f(x_1x_2x_3\dots x_j) = y$ , pak právě  $f'(x_1x_2x_3\dots x_j 0) = y \wedge f'(x_1x_2x_3\dots x_j 1) = y$

Praktický dopad platnosti výše uvedeného výrazu je zejména v případech, kdy hradlo fakticky realizuje funkci  $f'$  (má  $j+1$  vstupů), avšak právě funkce  $f$  má nějaké význačné pojmenování. Například hradlo se dvěma vstupy realizuje funkci  $f'$ , kdy  $f'(00) = 1, f'(01) = 1, f'(10) = 0, f'(11) = 0$ . Je zřejmé, že na hodnotu funkce má vliv pouze první ze vstupů, druhý výstup nijak neovlivní, ať je v jakémkoli stavu. Funkce  $f$  je potom funkce negace. Hradlo realizuje funkci negace prvního ze vstupů, NOT  $x_1$ .

□

### Příklad 3.1

Dvouvstupové hradlo z obrázku 3.8 při napájení  $V_{dd} = 3,3V$  realizuje funkci  $f_1$ , kde  $f_1(00) = 1, f_1(01) = 0, f_1(10) = 0, f_1(11) = 0$ , pokud je teplota čipu nižší než asi  $120^\circ C$ . Funkce  $f_1$  je funkcí negovaného logického součtu, označovaného zkratkou NOR. Když se čip ohřeje nad uvedenou teplotu (ale ne více než  $130^\circ C$ , hradlo realizuje funkci  $f_2$ , kde  $f_2(00) = 1, f_2(01) = 0, f_2(10) = 1, f_2(11) = 0$ . I když funkce  $f_2$  je funkcí dvou proměnných, bylo by možné najít funkci  $f_2'$ , která je funkcí jedné proměnné, protože výsledek funkce  $f_2$  nezávisí na prvním ze vstupů.  $f_2'(0) = 1$  a  $f_2'(1) = 0$ . Jde o funkci negace. Protože funkci  $f_2'$  lze podle poznámky 3.2 rozšířit na funkci dvou proměnných a takto rozšířenou funkcí bude právě funkce  $f_2$ , lze říci, že hradlo v tomto režimu realizuje funkci negace druhého vstupu. První vstup hradla v tomto režimu nemá význam, hodnota na něm neovlivňuje výstup.

□

### Poznámka 3.3 k definici 3.1

Prostředím, zmíněným v definici 3.1 a vybírajícím jednu z  $|\Phi|$  funkcí, která je právě hradlem realizována, může být obecně cokoli, co se v daném obvodu nepovažuje přímo za logický signál, nesoucí nějakou obvodem zpracovávanou informaci. Může to být například úroveň napájecího napětí, teplota, úroveň osvětlení a podobně. Někdy za „prostředí“ může být považován i speciální elektrický signál (ne nutně jen s logickými úrovněmi), rozvedený ke všem hradlům. Více o prostředí lze nalézt v části 2.4 této práce.

□

Předpokládá se, že polymorfni hradla budou v obvodech kombinována s běžnými hradly a dalšími obvodovými prvky, jak je to běžné u konvenčních číslicových obvodů. Návrh číslicových obvodů složených čistě z polymorfni hradel zatím nedává příliš smysl a také samotný proces návrhu by se tím velmi komplikoval. Polymorfni hradlo by tedy na fyzické úrovni mělo být konstruováno tak, aby bylo snadno kombinovatelné s běžnými hradly. V ideálním případě by mělo být kompatibilní s obvody CMOS.

Pro aplikace polymorfni elektroniky, kde se předpokládá nahrazení některých konvenčních hradel polymorfni hradly, aby byla nějakým podstatným způsobem rozšířena kvalita obvodu (přidání diagnostické funkce, přidání skryté funkce, přidání nouzové funkce), je žádoucí, aby polymorfni hradla měla rozměry (plocha čipu, počet zapojených tranzistorů) alespoň řádově srovnatelné s rozměry konvenčních hradel. V opačném případě by totiž kvalita řešení získaného použitím polymorfni hradel mohla být snížena nebo znehodnocena příliš velkým nárůstem plochy čipu a tím i příliš velkým nárůstem ceny obvodu. Ideální je, pokud jedno polymorfni hradlo realizující  $|\Phi|$  funkcí nezabírá více plochy čipu než je součet plochy zabrané  $|\Phi|$  konvenčními hradly realizujícími stejné funkce. Podobná tvrzení přiměřeně platí i pro spotřebu hradla.

Na základě zkušeností a experimentů s publikovanými polymorfni hradly, jakož i návrhu některých vlastních polymorfni hradel a dále na základě zkušeností s aplikacemi polymorfni číslicové elektroniky, autor stanovil vlastnosti a kritéria, které by polymorfni hradlo mělo mít a splňovat. Jsou přehledně uvedeny v následující definici.

**Definice 3.2:**

Požadavky na vlastnosti ideálního polymorfního hradla lze definovat takto:

- P1. **vysoký vstupní odpor** odpovídající odporu izolovaného hradla (elektrody) unipolárního tranzistoru typu MOS, aby hradlo nezatěžovalo svými vstupy jemu předcházející logickou síť,
- P2. **nízký výstupní odpor** odpovídající odporu plně otevřeného kanálu unipolárního tranzistoru typu MOS, aby logické úrovně na výstupu hradla byly jasně definované a pohybovaly se pokud možno blízko potenciálů zdroje bez ohledu na zatížení výstupu další logickou sítí, čímž bude zajištěna přiměřená šumová imunita obvodů,
- P3. **rozsah napájecího napětí** v rozsahu napájecího napětí CMOS obvodů,
- P4. **akceptace logických úrovní** platných pro hradla CMOS (log. 0 do  $0,3 V_{dd}$  a log. 1 nad  $0,7 V_{dd}$  [Weste and Harris: CMOS VLSI Design]),
- P5. **výstupní úrovně** v rozsahu platných výstupních úrovní hradel CMOS,
- P6. **krátká doba propagace signálu** ze vstupu na výstup ( $t_{pd}$ ),
- P7. **malé rozměry**,
- P8. **nízká spotřeba**.

□

## 3.5 Známá polymorfní hradla

### 3.5.1 Polymorfní hradla řízená napájecím napětím

Na polymorfní hradla, jejichž funkce je řízena (vybírána) úrovní napájecího napětí, je možno pohlížet jako na zvláštní druh analogových obvodů. Z předchozího popisu a zvláště podmínek formulovaných v definici 3.2 je zřejmé, že od polymorfního hradla se navenek očekává chování jako od číslicového obvodu (zejména co se týká vstupních a výstupních logických úrovní – tedy hradlo pracuje výhradně s číslicovými úrovněmi), avšak pokud jde o jeho vnitřní strukturu, ne všechny tranzistory budou v ustáleném stavu pracovat v režimu saturace. Má-li být hradlo citlivé na úroveň napájecího napětí a tudíž má-li se jeho chování s úrovní napájecího napětí měnit, musí mít vestavěn nějaký druh detektoru úrovně napětí. Díky požadavku na jednoduchost struktury hradla, který vyplývá z podmínky P7 definice 3.2, bude takový detektor tvořen pravděpodobně jedním či několika málo tranzistory pracujícími v lineárním režimu. Zpravidla to nebývá tranzistor, který přímo spíná výstupní úroveň. Požadavky P2 a P5 (které jsou vzhledem ke kompatibilitě polymorfních hradel s běžnými hradly klíčové) vedou na řešení, kdy výstupní úroveň spínaly tranzistory pracující v saturaci (ideálně navíc log. 1 tranzistor typu p-mos a log. 0 tranzistor typu n-mos).

Protože pro polymorfní hradla je typická situace, kdy pro stejnou kombinaci vstupů v některých případech má mít výstup hodnotu např. log. 0 a v jiných případech (třeba pro jiné napájecí napětí) zase hodnotu log. 1, přičemž požadavek P7 nedovoluje užít klasického multiplexoru či přenosového hradla, řeší se konflikt úrovní často tak, že v některých případech výstup budí dva nebo i více otevřených tranzistorů, přivádějících na výstup různé úrovně. Logicky však musí „zvítězit“ jen jedna. Který tranzistor určí výstupní úroveň je v takovém případě nastaveno zejména vodivostí jejich kanálu (šířka vs. délka kanálu).

Rozdíl fyzických rozměrů mezi tranzistory, které jsou využity v jednom polymorfním hradle, je dalším typickým rysem polymorfních hradel řízených napájecím napětím. Zatímco u konvenčních CMOS hradel se v jednom hradle zpravidla vyskytují tranzistory jednotných rozměrů a jejich role při generování výstupní logické úrovně je určena prakticky pouze jejich

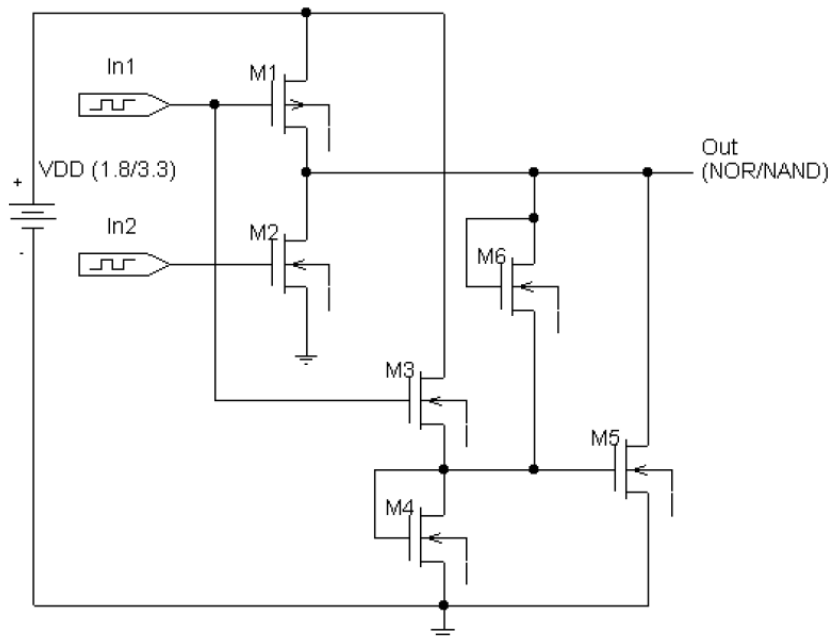


zapojením, u polymorfních hradel řízených napájecím napětím je i rozměr tranzistoru (zejména jeho kanálu) výrazným rysem určujícím fungování hradla a velikost kanálu (a tím určenou jeho vodivost) je třeba vůči ostatním tranzistorům případně se podílejícím na buzení výstupu vyladit. Je zřejmé, že takovým laděním se určuje i kvalita hradla ve smyslu stupně naplnění požadavků P2, P5, P7 a nakonec i P8.

S největší pravděpodobností zapojení s využitím dvou výše popsanych řešení (tranzistor pracující v lineárním režimu a „soupeření“ otevřených tranzistorů) musí způsobit trvalý průchod proudu mezi  $V_{dd}$  a zemí, což působí proti dokonalému naplnění požadavku P8 definice 3.2. I když lze vhodným nastavením parametrů tranzistorů tento proud omezit, bude pravděpodobně klidová spotřeba polymorfních hradel řízených napájecím napětím v některých situacích (pro některé vstupní kombinace) řádově větší, než je tomu u klasických CMOS číslicových hradel.

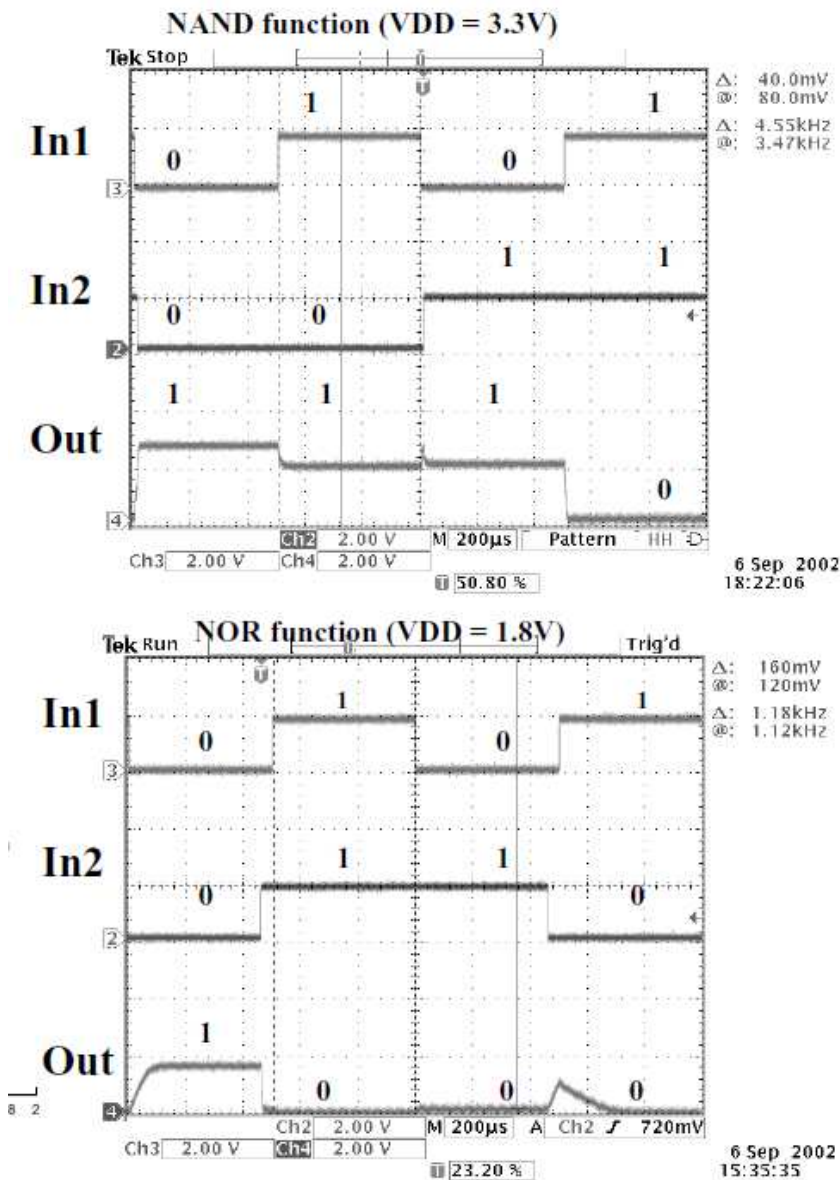
### Hradlo NAND/NOR řízené $V_{dd}$ , navržené v JPL

Toto hradlo vzniklo v laboratořích NASA při experimentech s evolučním návrhem elektronických obvodů. Hradlo je tedy výsledkem evolučního návrhu. Autoři se na příkladu multifunkčního (polymorfního) hradla pokoušeli ukázat, že pomocí evolučního návrhu je možné vytvářet takové návrhy obvodů, které jsou nezávislé na platformě, na které vznikly. Za normálních okolností evoluční návrh přirozeně využívá všech vlastností obvodu, na kterém probíhá, včetně jeho nedokonalostí a nehomogenit, proto návrh nemusí být přenositelný, na jiné platformě nemusí fungovat [Tho96], [Tho99], [Sto04b]. Že hradlo funguje i na jiných platformách, než na jaké bylo vyevolvováno (patrně na FPTA obvodě používaném v JPL [Zeb00], [Sto00]), bylo ověřeno implementací na čipu s využitím technologie HP 0,5  $\mu\text{m}$ . Zajímavostí je, že se jednalo patrně o první fyzickou realizaci obvodu navrženého evolucí v obvodu typu ASIC. Do té doby byly obvody navržené evolucí jen simulovány nebo realizovány v FPGA. Hradlo bylo publikováno v [Sto04b].



Obrázek 3.5: Polymorfní hradlo NAND/NOR navržené v JPL [Sto04b].

Zapojení hradla je vidět na obrázku 3.5. Hradlo realizuje dvě funkce –  $f_1 = \text{NAND}$  a  $f_2 = \text{NOR}$ . Autoři uvádí, že funkci  $f_1$  hradlo realizuje bezpečně pro napájecí napětí  $V_{dd} = 3,3 \text{ V}$  a funkci  $f_2$  bezpečně pro  $V_{dd} = 1,8 \text{ V}$ . Tato napětí zřejmě vychází z použité technologie. Jak se hradlo chová mezi těmito napětími, nebylo publikováno. Nebyly bohužel publikovány ani základní parametry použitých tranzistorů. Přitom jak bylo popsáno výše, u polymorfního hradla řízeného  $V_{dd}$  jsou parametry tranzistorů (rozměry kanálu a z toho plynoucí odpor kanálu) zásadní. Je proto velmi obtížné bez znalostí parametrů tranzistorů toto hradlo simulovat nebo znovu implementovat.



Obrázek 3.6: Chování polymorfního hradla NAND/NOR navrženého v JPL.

V [Sto04b] byly také publikovány průběhy zaznamenané na výstupu hradla v reakci na různé kombinace vstupních logických úrovní. Jsou zobrazeny na obrázku 3.6 (obrázek převzat z [Sto04b]). Je vidět, že hradlo v zásadě plní předpokládané funkce. Je však také vidět, že hradlo se nechová vždy tak dokonale, jako je tomu u konvenčních hradel CMOS. Přestože není rychlost změn úrovní na vstupech nijak rychlá (asi 1kHz), výstup reaguje s jistým zpožděním, hrany signálu na výstupu nejsou v některých případech příliš ostré. Takové chování je dáno

konstrukcí hradla s minimem tranzistorů, kdy nutně musí v některých stavech docházet k současnému otevření několika tranzistorů budících výstup. Tehdy „zvíťezí“ silnější tranzistor za cenu zvýšené spotřeby a zhoršené úrovně výstupu. Z obrázku 3.6 je zřejmé, že taková situace nastává zejména v okamžiku rozdílných vstupních úrovní (kombinace 01 a 10). V takové situaci jsou buď oba tranzistory M1 a M2 otevřeny (a zkratují  $V_{dd}$  k zemi) nebo naopak oba uzavřeny a pak je buzení výstupu na ostatních tranzistorech, které však již nejsou zapojeny podle zásad CMOS logiky. Patrně nejhůře se hradlo chová v režimu NOR při vstupní kombinaci 10, kdy výstup má mít úroveň log. 0, avšak jak je vidět z obrázku 3.6, výstup dosáhne v jednom okamžiku až úrovně 1 V, což je při  $V_{dd} = 1,8$  V již za hranicí tolerance pro úroveň log. 0.

Parametry a splnění požadavků

Požadavek			
P1	vstupní odpor	hradla tranzistoru MOS	✓✓
P2	výstupní odpor	závisí na stavu (sepnutém tranzistoru)	✓
P3	napájecí napětí	1,8 – 3,3 V	✓✓
P4	vstupní úrovně	nebylo publikováno, nelze ověřit	?
P5	výstupní úrovně	Až 1 V v log. 0; 1,8/2,5 V log. 1 pro $V_{dd} = 1,8/3,3$ V	✗
P6	$t_{pd}$	nebylo publikováno, odhadem až jednotky $\mu$ s	?
P7	rozměry	6 tranzistorů	✓✓
P8	spotřeba	nebylo publikováno, ale pravděpodobně v některých režimech velká	?

Tabulka 3.1: Sledované parametry hradla NAND/NOR JPL a splnění požadavků kladených na polymorfní hradla.

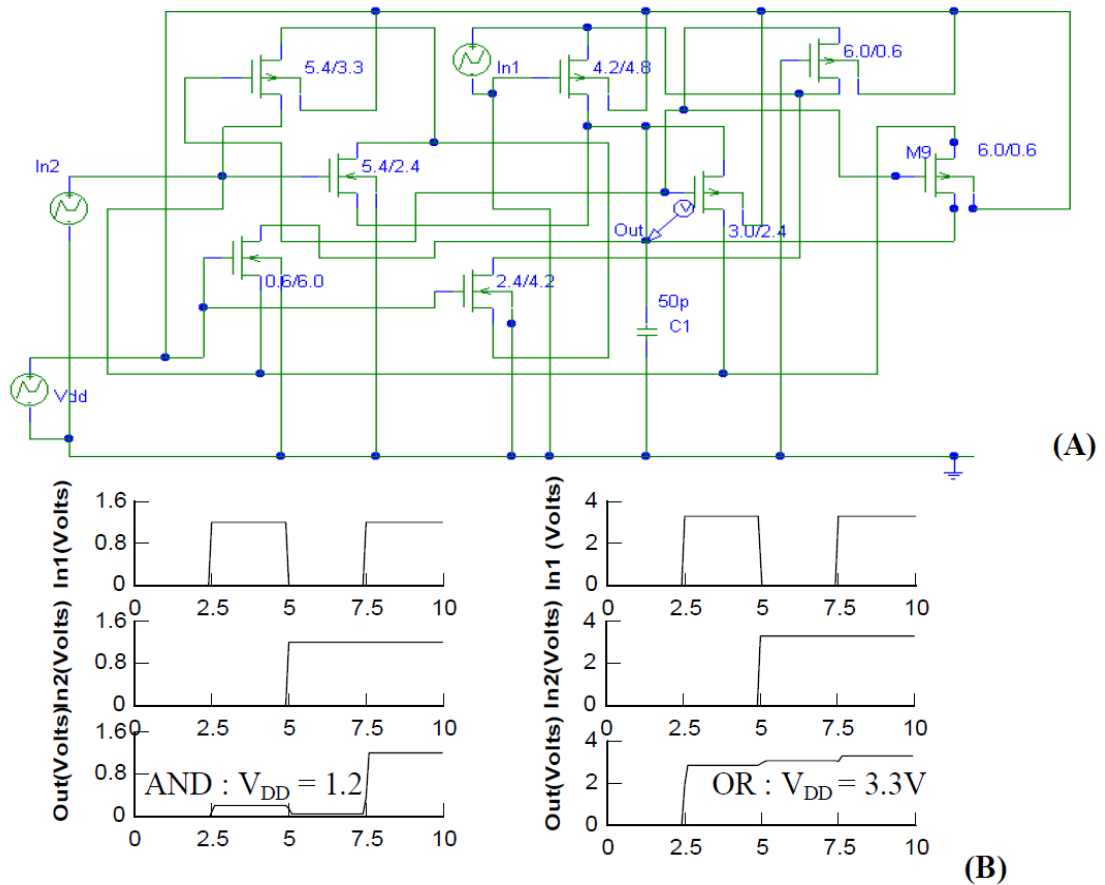
### Hradlo AND/OR řízené $V_{dd}$ , navržené v JPL

Dalším polymorfním hradlem, řízeným napájecím napětím, které bylo navrženo s pomocí evolučních technik kolektivem Adriana Stoicy v laboratořích JPL NASA, je hradlo realizující dvě funkce  $f_1 = \text{AND}$  pro  $V_{dd} = 1,2$  V a  $f_2 = \text{OR}$  pro  $V_{dd} = 3,3$  V [Sto02]. Toto hradlo však na rozdíl od hradla NAND/NOR nebylo realizováno. Jeho funkce byla pouze ověřena simulací. Hradlo sestává z osmi tranzistorů a jeho zapojení je vidět na obrázku 3.7 (obrázek převzat ze [Sto02]). Na rozdíl od hradla NAND/NOR z JPL u tohoto hradla byly publikovány parametry tranzistorů (délka a šířka kanálu), a v [Sto02] se přímo uvádí, že tyto parametry byly také předmětem evolučního návrhu. To je zcela pochopitelné – jak již bylo uvedeno, právě ve vhodném nastavení parametrů užitých tranzistorů a jejich vzájemném vyvážení je klíč k polymorfismu hradla řízeného  $V_{dd}$ . Parametry jednotlivých tranzistorů jsou vidět v obrázku 3.7 (u každého tranzistoru je uvedena šířka/délka kanálu) a je zřejmé, že rozměry tranzistorů se mezi sebou liší až o řád. Tranzistory se vzhledem k předpokládanému napájecímu napětí jeví poměrně velké.

Struktura hradla je velmi nekonvenční, je na první pohled znát, že šlo o výsledek automatizovaného evolučního návrhu. V zapojení hradla jsou porušovány základní principy návrhu hradel v technologii CMOS. Například vstupy hradla nejsou zapojeny pouze na hradla (elektrody G) tranzistorů (nelze zaručit vysoký vstupní odpor), žádný z tranzistorů nespíná na výstup  $V_{dd}$  (úroveň log. 1 se na výstup spíná ze vstupů).

Výsledky simulací navrženého hradla na obrázku 3.7 ukazují, že hradlo se chová celkem korektně v obou režimech, alespoň pokud jde o výstupní úrovně. Na kvalitu dynamických parametrů hradla nelze usuzovat, protože simulovány byly jen pomalé děje – měřítko

vodorovných os je v milisekundách, frekvence změn vstupů (a výstupu) hradla je tedy jen v řádu stovek Hz.



Obr. 3.7: Polymorfní hradlo AND/OR navržené v JPL a jeho chování.

Požadavek			
P1	vstupní odpor	přes kanál tranzistoru mohou přímo budit výstup	✘
P2	výstupní odpor	závisí na stavu (sepnutém tranzistoru)	✓
P3	napájecí napětí	1,2 – 3,3 V	✓✓
P4	vstupní úrovně	nebylo publikováno, nelze ověřit	?
P5	výstupní úrovně	Do 0,3 V v log. 0; 1,2/3 V log. 1 pro $V_{dd} = 1,2/3,3$ V	✓
P6	$t_{pd}$	nebylo publikováno, odhadem až jednotky $\mu$ s	?
P7	rozměry	8 tranzistorů	✓
P8	spotřeba	nebylo publikováno, ale pravděp. v některých režimech velká	?

Tabulka 3.2: Sledované parametry hradla AND/OR JPL a splnění požadavků kladených na polymorfní hradla.

### Hradlo NAND/NOR řízené V<sub>dd</sub>, navržené FIT/FEKT

Pro účely testování navržených polymorfních obvodů bylo ve spolupráci s Ing. Romanem Prokopem, Ph.D. z Ústavu mikroelektroniky Fakulty elektrotechniky a komunikačních

technologíí VUT v Brně navrženo polymorfní hradlo se dvěma vstupy řízené napájecím napětím [Ruz08a]. Cílem bylo navrhnout hradlo kompatibilní s obvody CMOS (předpoklad použití s konvenčními hradly CMOS na jednom čipu). Protože byla k dispozici technologie CMOS AMIS 0,7 $\mu$ m, která předpokládá  $V_{dd}$  do asi 5,5 V, bylo zvoleno, že hradlo bude mít dvě funkce  $f_1$  a  $f_2$ , přičemž funkci  $f_1$  bude hradlo bezpečně realizovat při napětí 5V, funkci  $f_2$  pak při napětí nižším, ale také obvyklým pro logické obvody – 3,3V. Jako logické funkce byly zvoleny  $f_1 = \text{NAND}$  a  $f_2 = \text{NOR}$ , protože obě tyto funkce jsou logicky úplné, tj. je možné pomocí takové funkce realizovat všechny ostatní funkce. Logicky úplné funkce byly zvoleny v naději, že takto bude usnadněn (zejména evoluční) návrh složitějších kombinačních obvodů využívajících tato hradla. Logickým požadavkem byla „konkurenceschopnost“ navrženého hradla v porovnání s konvenčními hradly CMOS, pokud jde o jeho rozměry.

Návrh polymorfního hradla řízeného napájecím napětím vyžaduje poněkud specifický přístup. Na jednu stranu je třeba pamatovat na to, že půjde o obvod, který má vykazovat chování charakteristické pro číslicové obvody – zejména tranzistory spínající výstup by měly mít charakteristické zapojení, jaké je vidět například na obrázku 3.3 (komplementární dvojice MOS tranzistorů), na druhou stranu jde však o návrh analogového obvodu, který musí reagovat na měnící se napětí  $V_{dd}$  a (pouze) pro některé kombinace vstupních úrovní „přinutit“ výstupní komplementární dvojici chovat se odlišně pro různé úrovně  $V_{dd}$ . V případě, kdy  $f_1 = \text{NAND}$  a  $f_2 = \text{NOR}$ , je rozdíl ve výstupní úrovni pro takové (dvě) kombinace vstupních úrovní, kdy na každém vstupu je jiná hodnota. Přehledně to ukazuje tabulka 3.3.

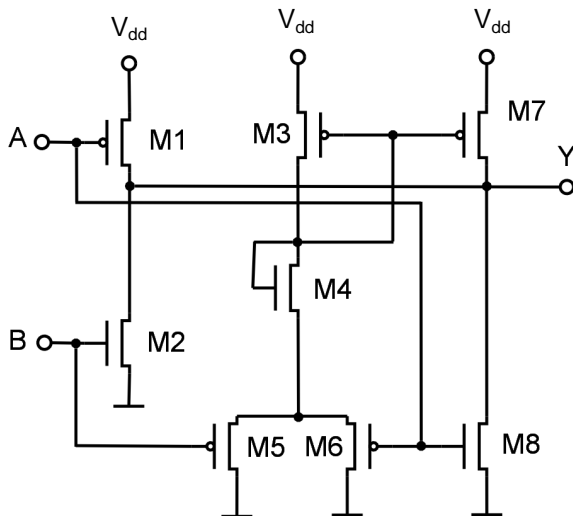
	a	b	$V_{dd} = 5 \text{ V}$ $f_1(ab)$	$V_{dd} = 3,3 \text{ V}$ $f_2(ab)$
1	0	0	1	1
2	0	1	1	0
3	1	0	1	0
4	1	1	0	0

Tabulka 3.3: Funkce realizované hradlem NAND/NOR řízeným  $V_{dd}$ .

Obrázek 3.8 ukazuje schéma navrženého hradla. Je vidět, že hradlo sestává z osmi MOS tranzistorů (3 n-MOS a 5 p-MOS). Pro srovnání: běžné hradlo NAND nebo NOR v technologii CMOS sestává ze čtyř tranzistorů. Velikost navrženého polymorfního hradla je z tohoto pohledu příznivá – má stejnou „cenu“ jako dvojice běžných CMOS hradel (lze říci, že toto hradlo v určitých aplikacích dvě běžná hradla nahrazuje). Pokud by ale dvojice běžných hradel (jedno hradlo NAND a jedno NOR) měla funkčně nahradit navržené polymorfní hradlo, musela by být ještě doplněna o detektor napěťové úrovně napájecího napětí a nějaký přepínač, který by dle napěťové úrovně volil pro výstup jedno z dvojice běžných hradel. Pro určitou třídu aplikací může být proto navržené polymorfní hradlo výhodnější než řešení využívající konvenčních obvodových struktur. Při srovnání se stejným hradlem, navrženým v NASA JPL [Sto04b], popsaným výše, vychází navržené hradlo rozměrově poněkud hůře (8 tranzistorů versus 6 tranzistorů hradla z JPL), avšak jeho parametry, implikující praktickou použitelnost v reálných obvodech, tento handicap více než vyvažují.

Je však třeba poznamenat, že samotné porovnání počtu tranzistorů nevyjadřuje zcela přesně poměr ceny dvou hradel. O něco přesnějším ukazatelem by byl poměr ploch na čipu, které hradla zaberou. Jak bylo již uvedeno, u polymorfních hradel řízených  $V_{dd}$  je třeba uplatňovat tranzistory různé velikosti. Právě velikosti tranzistorů (ovlivňující zejména odpor kanálu otevřeného tranzistoru, ale i prahové napětí) a jejich vzájemné poměry jsou mimo jiné podstatné pro to, kdy (při jakém napětí) se změní chování obvodu jako celku a tudíž i logická funkce,

realizovaná hradlem. Zatímco u běžných CMOS hradel se rozměry jednotlivých tranzistorů podstatně neliší, u polymorfních hradel může být mezi tranzistory téhož hradla rozdíl až řádový.



Obrázek 3.8: Schéma hradla NAND/NOR řízeného  $V_{dd}$ , které bylo navrženo na FIT a FEKT.

tranzistor	šířka kanálu	délka kanálu
M1	1 $\mu\text{m}$	12 $\mu\text{m}$
M2	1,7 $\mu\text{m}$	10 $\mu\text{m}$
M3	7 $\mu\text{m}$	1 $\mu\text{m}$
M4	8 $\mu\text{m}$	2 $\mu\text{m}$
M5	20 $\mu\text{m}$	0,7 $\mu\text{m}$
M6	20 $\mu\text{m}$	0,7 $\mu\text{m}$
M7	70 $\mu\text{m}$	1 $\mu\text{m}$
M8	2,2 $\mu\text{m}$	20 $\mu\text{m}$

Tabulka 3.4: Parametry tranzistorů použitých v hradle NAND/NOR řízeném  $V_{dd}$ .

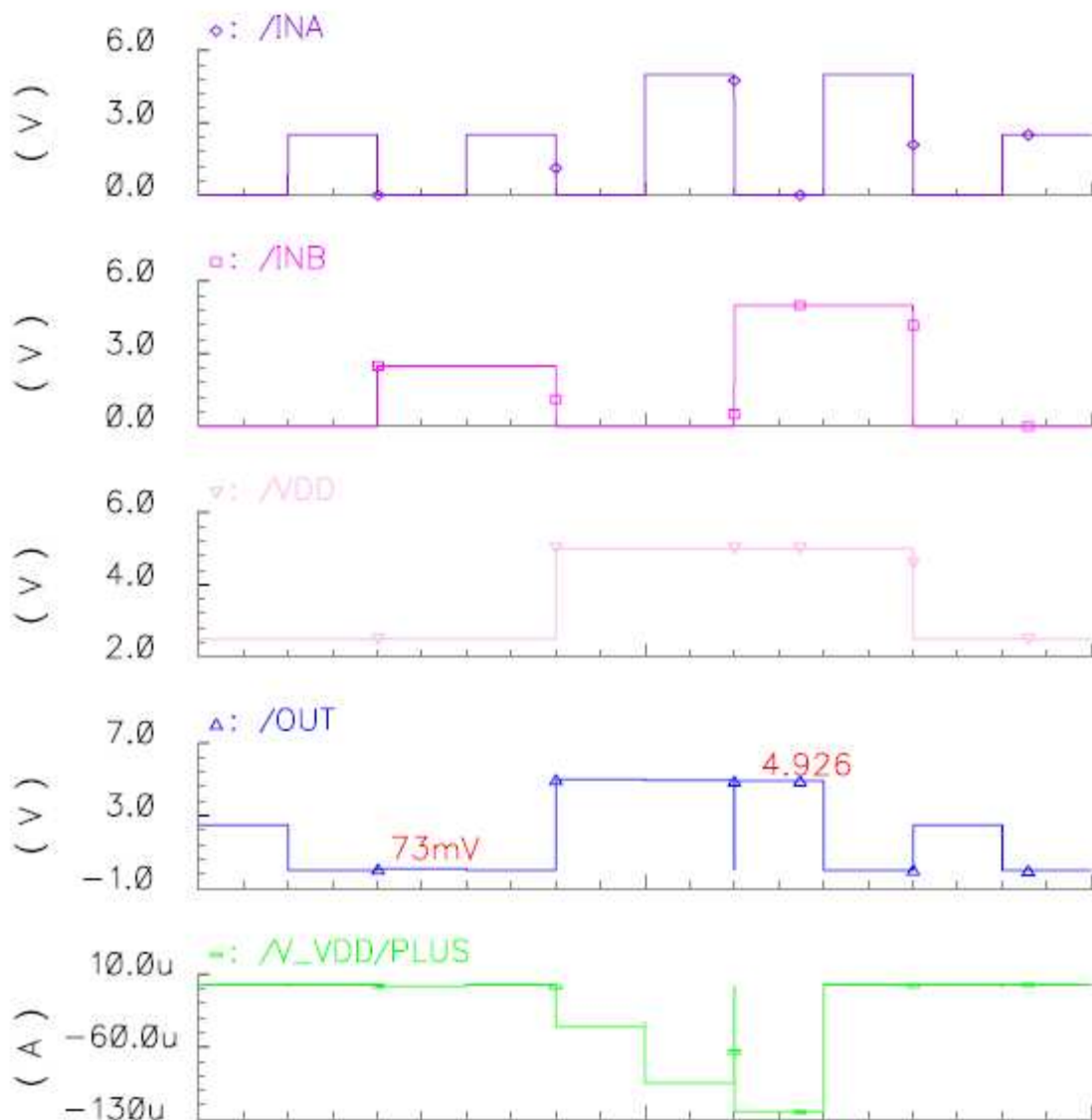
V tabulce 3.4 jsou uvedeny základní parametry (rozměry kanálu) tranzistorů hradla z obrázku 3.8. Pro realizaci hradla byla využita technologie AMIS 0,7  $\mu\text{m}$ .

Jak tedy hradlo z obrázku 3.8 funguje? Tranzistory M1 a M2 jsou zapojeny téměř tradičně – jako komplementární dvojice v invertoru (obrázek 3.3). Rozdíl je však v tom, že elektrody G těchto dvou tranzistorů nejsou spolu propojeny. Jako invertor budou však fungovat v situacích, kdy hodnoty na obou vstupech A, B hradla jsou stejné (řádky 1 a 4 tabulky 3.3). V tabulce 3.4 je vidět, že komplementární dvojice tranzistorů M1, M2 má rozměry přibližně stejné. Oproti tranzistorům, užívaným v obyčejných CMOS hradlech, však mají poměrně dlouhé (až desetkrát delší) a úzké (opět až desetkrát) kanály. To znamená, že odpor jejich kanálu v otevřeném stavu bude výrazně větší, než je tomu u tranzistorů v běžných CMOS hradlech. Pro jednoznačné definování úrovně výstupu to příliš nevádí, větší odpor je však velmi prospěšný v situaci, kdy se otevřou oba tranzistory (a taková situace možná je – viz řádek č. 2 tabulky 3.3) – tehdy teče trvalý proud přes kanály obou tranzistorů a je omezen právě jen jejich odporem. Odpor kanálů tranzistorů tak omezuje klidovou spotřebu hradla.

V situaci, kdy vstup A = 1 a B = 0 (řádek 3 tabulky 3.3) jsou oba vstupní tranzistory zavřeny a výstupní úroveň musí zajistit komplementární dvojice M7, M8. Zde se otevírá buď M8, má-li být hradlo v režimu NOR, nebo je překonán otevřením tranzistoru M7, který na výstup vnutí

hodnotu log. 1. Tranzistor M7 je totiž záměrně konstruován tak, aby byl silnější – je větší, má daleko vodivější kanál (krátký a široký). Z tabulky 3.4 je zřejmé, že rozdíl v rozměrech kanálu je markantní a vodivost kanálu tranzistoru M7 vůči M8 bude lepší o nejméně o dva řády. Tranzistor M7 se otevírá až tehdy, pokud je napájecí napětí větší, než je určitá hranice. Tato hranice je nastavena parametry tranzistorů M3 a M4. Je zřejmé, že pokud se tranzistory M3 a M4 otevřou, způsobí to otevření i tranzistoru M7.

Situace, kdy vstup A = 0 a B = 1 (řádek 2 tabulky 3.3) způsobí sice otevření obou tranzistorů M1 a M2, avšak díky jejich mírně rozdílným parametrům převáží M2 a na výstupu se objeví log. 0 (viz tabulka 3.4). Pouze však pro tak malá  $V_{dd}$ , že ještě nedojde k otevření M3. Hradlo je pak v režimu NOR. Jakmile se M3 otevře, otevře se i M7 a ten, protože je nejsilnější (vodivost jeho kanálu bude nejméně o dva řády lepší než M2), přebíjí otevřený M2 a výstup dostává na potenciál blízký  $V_{dd}$ , tedy log. 1. Hradlo pak běží v režimu NAND.



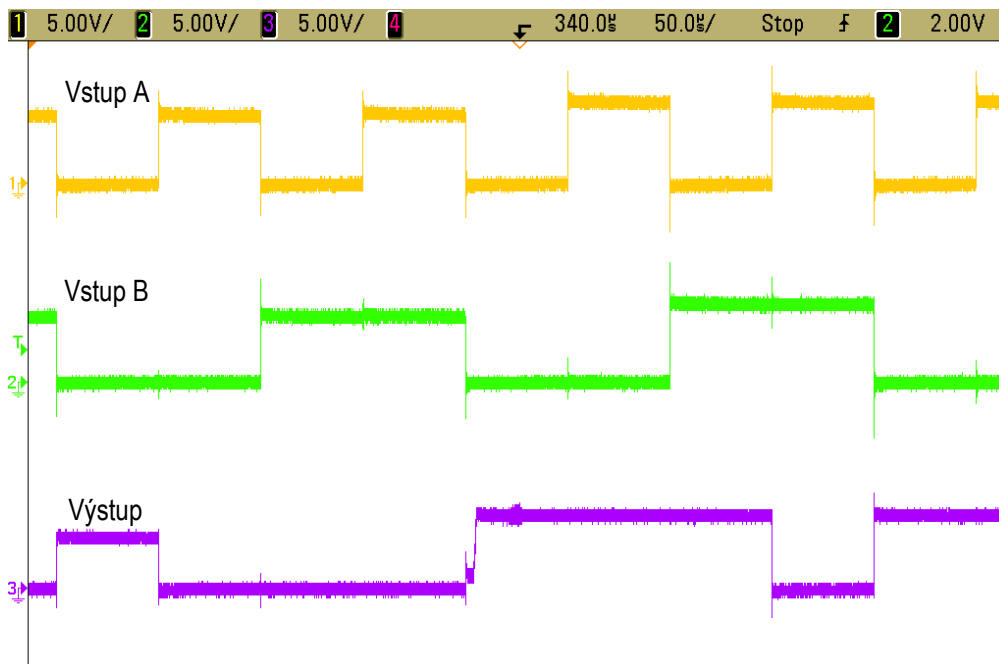
Obrázek 3.9: Simulace chování polymorfního hradla NAND/NOR řízeného  $V_{dd}$ , které bylo navrženo na FIT a FEKT.

Tranzistory M5 a M6 zabraňují, aby se otevřel tranzistor M7 v situaci, kdy oba vstupy jsou v úrovni log. 1. V této situaci musí zůstat výstup v log. 0, i když napájecí napětí bude 5 V. Úroveň log. 0 je na výstup přivedena tranzistory M2 a M8. Tranzistory M5 a M6 jsou však zavřené a nedovolí, aby se otevřely M3 a M4.

Z výše uvedeného popisu rolí a parametrů jednotlivých tranzistorů pro různé vstupní úrovně a různé režimy činnosti hradla vyplývá, že popsané hradlo funguje na rozdíl od běžného CMOS hradla (např. NAND nebo NOR) poněkud nesymetricky. Proto i parametry hradla (jako například spotřeba nebo zpoždění signálu, délky hran) budou různé pro různé kombinace vstupních úrovní a režimů hradla. Toho však lze využít a vhodným návrhem minimalizovat dobu, po kterou hradlo bude pracovat s vyšší statickou spotřebou.

Na obrázku 3.9 je vidět simulace chování hradla z obrázku 3.8. Průběhy IN A a IN B jsou signály generované pro vstupy hradla. Jsou generovány tak, aby se na vstupech postupně vystřídaly kombinace 00, 10, 01 a 11 s úrovní log. 1 asi 3V a potom stejné kombinace s úrovní log. 1 asi 5 V. Spolu se změnou úrovní se mění také napájecí napětí hradla ( $V_{DD}$ ). Lze říci, že napájecí napětí hradla už neslouží jen k dodávce energie, ale dodává také další vstupní informaci. Takto se prověří funkce hradla pro všechny vstupní kombinace v obou režimech.

Výstup hradla (OUT) se při simulaci chová podle očekávání. Pro napájecí napětí 3 V hradlo realizuje funkci NOR. Úroveň log. 0 nepřekračuje asi 0,073V, což je bohatě v toleranci pro CMOS výstup v log. 0. Je vidět, že pro kombinaci vstupů 01 je úroveň log. 0 mírně horší. Je to způsobeno tím, že v této situaci je otevřen i tranzistor M1, který výstup připojuje k  $V_{dd}$ . Je však slabší než M2, který je otevřen také a výstup převáží do log. 0. Protože úroveň  $V_{dd}$  nestačí k otevření M3 a M4, neotevře se ani M7. Poslední průběh na obrázku 3.9 ukazuje spotřebu hradla – proud tekoucí přes  $V_{dd}$ . Pokud je hradlo v režimu NOR, je spotřeba hradla malá. Dokonce i v situaci, kdy jsou otevřeny oba tranzistory M1 a M2 (vstupní kombinace 01) a přes jejich kanály tak teče proud z  $V_{dd}$  do země, není spotřeba hradla velká. Tranzistory M1 a M2 jsou slabé a vodivost jejich kanálů v otevřeném stavu není velká.



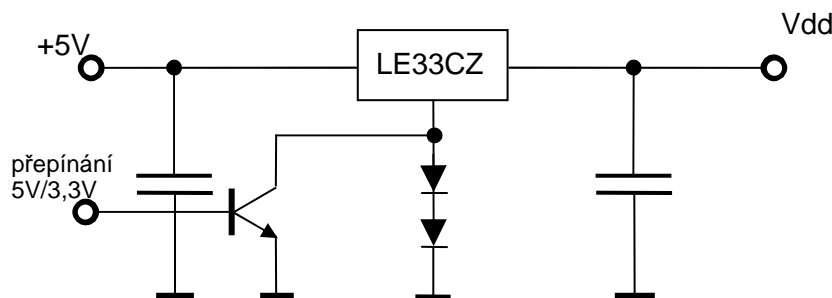
Obrázek 3.10: Skutečné průběhy na polymorfním hradle NAND/NOR řízeném  $V_{dd}$ , navrženém na FIT a FEKT.



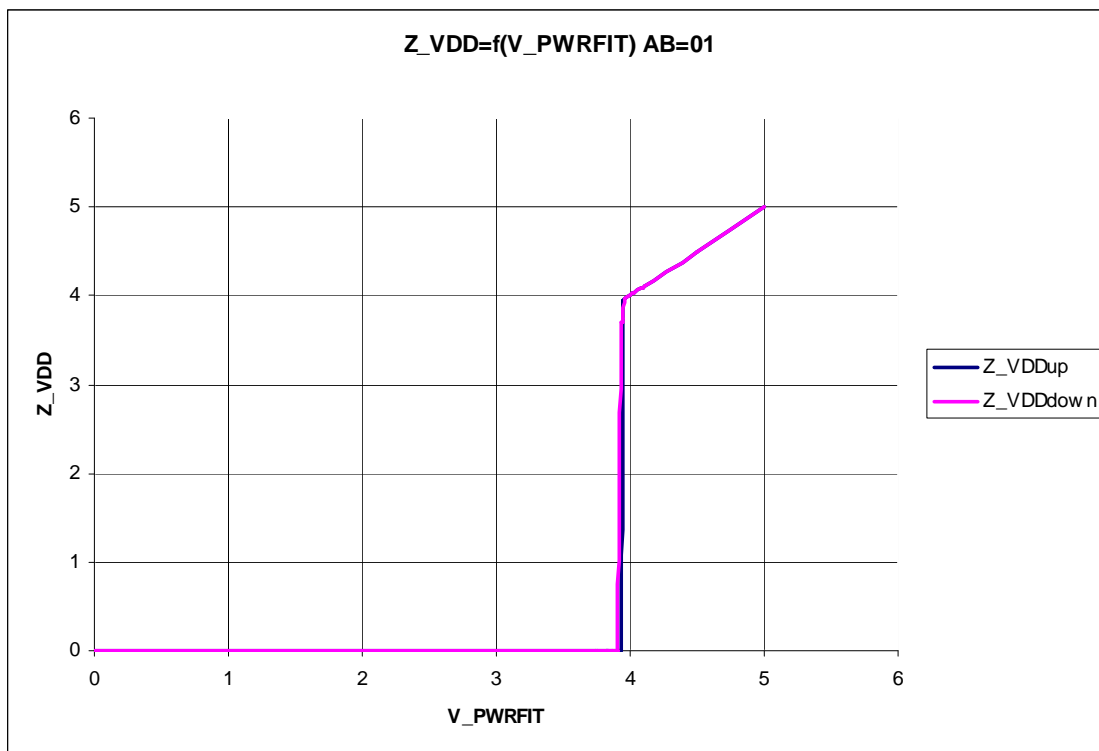
Pro napájecí napětí 5 V hradlo realizuje funkci NAND. Úroveň log. 1 na výstup zajišťuje tranzistor M1 a M7. Protože pro kombinace 00, 10 a 01 se otevírají i tranzistory M5 a M6 a díky vyššímu napájecímu napětí také M3 a M4, dochází k propojení  $V_{dd}$  a země také přes kanály tranzistorů M3, M4 a M5/M6. Protože tyto tranzistory mají kanál podstatně vodivější než M1 a M2, proud přes ně tekoucí je také podstatně vyšší. Je to vidět na průběhu proudu. Pro kombinaci 00 na vstupu je to už kolem  $40 \mu\text{A}$  (proud teče jen přes M3, M4 a M5/M6), pro kombinaci 10 je to už kolem  $80 \mu\text{A}$  (přidává se proud přes M7 a M8) a pro kombinaci 01 dokonce přes  $100 \mu\text{A}$  (proud teče přes M7 a M2 namísto M8, přičemž M2 je mírně vodivější než M8, proto je proud větší než v předchozím případě). Vstupní kombinace 11 je v režimu NAND nejúspornější, protože jsou uzavřeny tranzistory M5 a M6. Tím je také uzavřen M7,  $V_{dd}$  a zem nejsou nikde přes tranzistory propojeny. Úroveň log. 1 na výstupu je velmi dobrá (4,926 V) zejména díky silnému tranzistoru M7.

Z průběhu spotřeby hradla pro různé vstupní kombinace a režimy vyplývá, že hradlo je úspornější v režimu NOR než v režimu NAND. Vyšší spotřeba je způsobena detektorem úrovně  $V_{dd}$  (tranzistory M3 a M4 a „hradlovací“ tranzistory M5 a M6) a „přetahováním“ o úroveň výstupu mezi tranzistorem M7 a M2 nebo M8 (hraje roli jen pro vyšší  $V_{dd}$ , tedy režim NAND). Vyšší spotřebou se platí za jednoduchost zapojení – celé hradlo včetně detektoru úrovně  $V_{dd}$  obsahuje jen 8 tranzistorů.

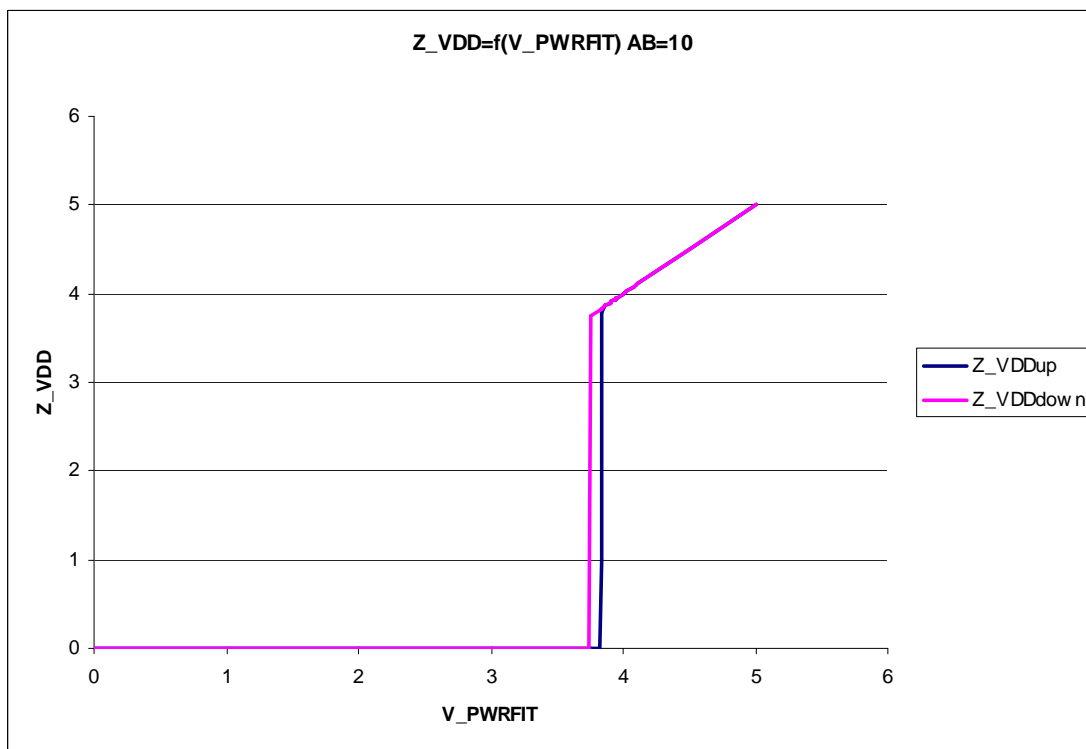
Obrázek 3.10 ukazuje vlastně tytéž časové průběhy, jako jsou vidět na obrázku 3.9. V tomto případě však jde o průběhy naměřené osciloskopem na skutečně vyrobeném hradle. V časovém okamžiku, který se nachází přibližně uprostřed obrázku, došlo ke skokové změně úrovně napájecího napětí z asi 3,3 V na 5 V (spolu s tím byly také zvýšeny úrovně vstupních signálů pro hradlo). Zdroj, který byl pro měření polymorfních hradel řízených  $V_{dd}$  navržen a realizován, je vidět na obrázku 3.11. Na časovém průběhu úrovně na výstupu hradla v obrázku 3.10 je vidět, že hradlo zareagovalo a změnilo realizovanou funkci z NOR na NAND. Je třeba poznamenat, že změnit napájecí napětí zcela skokově není snadné. Nárůst napájecího napětí probíhá vždy v nějakém konečném časovém intervalu. Z předchozího popisu je zřejmé, že ke změně režimu hradla dochází otevřením či zavřením tranzistorů M3 a M4. Tato změna je (jak je patrné v obrázku 3.9) doprovázena změnou napájecího proudu (odběru ze zdroje). Taková změna může (a často také vyvolá) změnu úrovně  $V_{dd}$  díky nenulovému vnitřnímu odporu zdroje (a napájecích vodičů). Naneštěstí, jak je vidět z obrázku 3.9, při nárůstu  $V_{dd}$  (změně funkce z NOR na NAND) dochází k nárůstu napájecího proudu, což způsobí malý lokální pokles  $V_{dd}$  a naopak. Navíc je ještě napájecí proud krátkodobě zvýšen díky překlápění tranzistorů (některé se ještě nezavřely a jiné se již otevírají). Je tak vlastně vytvořena záporná zpětná vazba, která způsobí, že než se napájecí napětí znovu ustálí, může dojít ke krátkodobému návratu k původní funkci. Hradlo se vlastně několikrát rychle překlápí z jedné funkce do druhé. Tento jev může způsobit na výstupu hradla krátký přechodný děj, pokud se v tomto okamžiku se změnou funkce zároveň má změnit výstupní logická úroveň. Na obrázku 3.10 je tento přechodný děj také vidět.



Obrázek 3.11: Zdroj navržený pro měření polymorfního hradla řízeného  $V_{dd}$ .



Obrázek 3.12: Závislost výstupu na napájecím napětí pro kombinaci 01 na vstupech hradla.

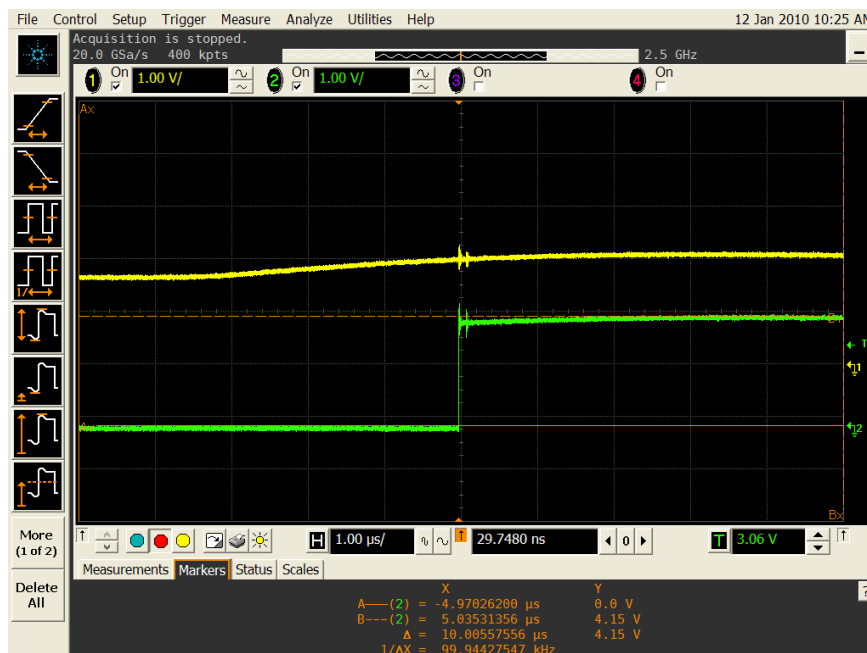


Obrázek 3.13: Závislost výstupu na napájecím napětí pro kombinaci 10 na vstupech hradla.

Obrázky 3.12 a 3.13 ukazují, kdy dochází ke změně funkce realizované hradlem. Je vidět, že ke změně funkce hradla dochází v poměrně úzkém pásmu mezi 3,7 a 3,9 V. Jak již bylo řečeno, změna funkce se na výstupu projevuje jen pro vstupní kombinace, kdy je na každém vstupu

rozdílná úroveň (viz též tabulka 3.3). Rozdíl mezi situací 01 (obrázek 3.12) a 10 (obrázek 3.13) je v tom, jak je na výstup přiváděna log. 0 v režimu NOR. Zatímco pro  $AB = 01$  je log. 0 přivedena na výstup přes M2 (přebijí log. 1 od M1), pro  $AB = 10$  je zdrojem log. 0 na výstupu otevřený M8. Změnu funkce hradla na NAND – změnu výstupní úrovně – realizuje vždy M7. Důsledkem je mírně rozdílné chování hradla pro oba případy. V případě  $AB = 01$  probíhá změna funkce vždy kolem 3,93 V, ať se M7 otevírá nebo zavírá. V případě  $AB = 10$  vykazuje hradlo jistou hysterezi. Při nárůstu  $V_{dd}$  se funkce mění na NAND při 3,84 V, při poklesu se hradlo navrácí k funkci NOR až při 3,75 V. Taková hystereze zřejmě může vést k tomu, že přechodný děj, zaznamenaný například na obrázku 3.10 bude o něco kratší nebo méně výrazný.

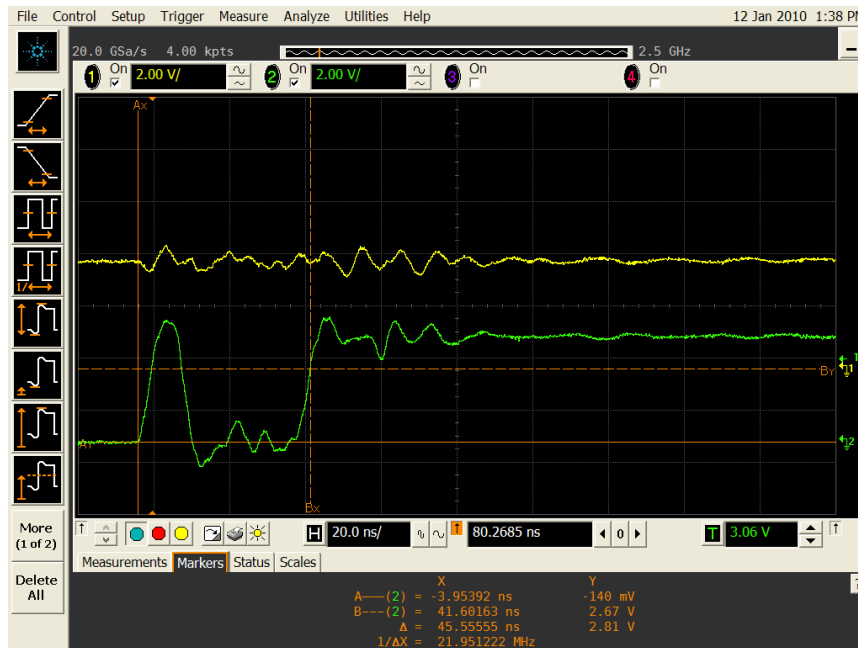
Na obrázku 3.14 je skutečný časový průběh změny funkce polymorfního hradla. Žlutá stopa patří napájecímu napětí  $V_{dd}$ , zelená stopa výstupu hradla. Napájecí napětí bylo zvýšeno tak rychle, jak to bylo vzhledem ke konstrukci zdroje a rozvodům napájení možné. Jeden horizontální dílek je 1  $\mu\text{s}$ ,  $V_{dd}$  se tedy změní z asi 3,3 V na 5V za několik jednotek mikrosekund. Přechodový děj proběhl poměrně příznivě, došlo jen k několika málo zákmitům  $V_{dd}$  díky kolísajícímu odběru v okamžiku přepínání tranzistorů hradla. Tyto zákmity se projeví i na výstupu hradla, zdá se, že výstup v okamžiku, kdy dojde k otevření tranzistoru M7 (a tudíž k překlopení do log. 1) prakticky sleduje napájecí napětí. V tomto případě nedošlo k oscilaci výstupu. Vysvětlení příznivého přechodného děje lze hledat dílem v hysterezi (viz přenosová funkce na obrázku 3.13), dílem v setrvačnosti celého hradla (zákmity mají periodu v desítkách nanosekund, ale tak rychle reagovat tranzistor velikosti M7 reagovat nedokáže), dílem v konstrukci zdroje a rozvodů napájení (včetně vhodně zvoleného blokování).



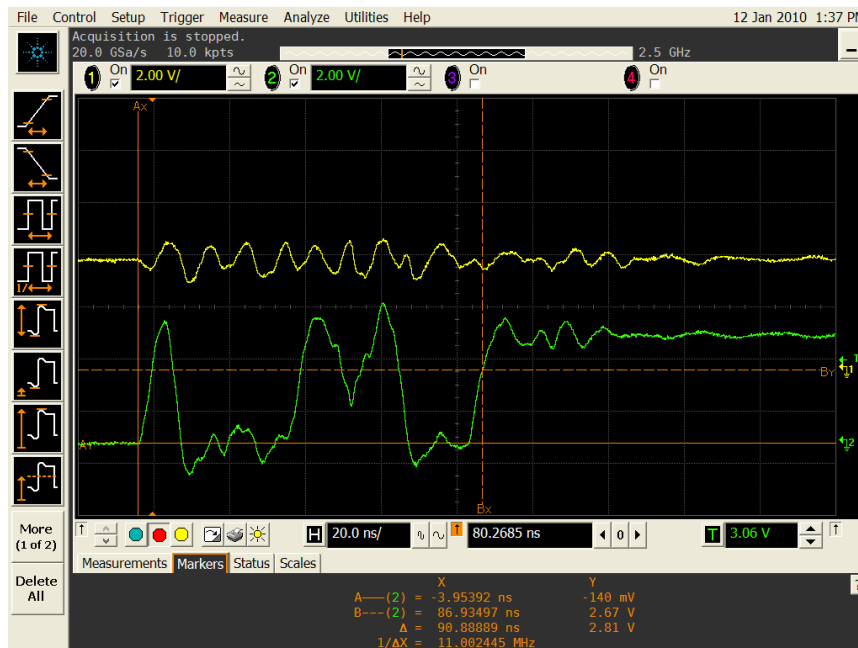
Obrázek 3.14: Změna funkce hradla z NOR na NAND při nárůstu napájecího napětí.

Obrázky 3.15 a 3.16 ukazují podobnou situaci, ale podrobněji. Opět se jedná o zachycení okamžiku změny funkce hradla z NOR na NAND při nárůstu  $V_{dd}$ . Změna funkce je vyjádřena změnou hodnoty výstupu pro navzájem různé vstupní logické úrovně. Zatímco obrázek 3.15 ukazuje chování hradla pro kombinaci vstupů 10, obrázek 3.16 ukazuje totéž pro kombinaci 01. Je zřejmé, že hypotéza o vlivu hystereze (srov. obr. 3.12 a 3.13) se potvrdila. Přechodný děj je skutečně v případě vstupní kombinace 10 kratší než v případě vstupní kombinace 01. Rozdíl

v přechodném ději pro různé vstupní vektory je také způsoben rozdílným proudem, který hradlo odebírá z  $V_{dd}$ . Z obrázku 3.9 je patrné, že pro vstupní kombinaci 10 má hradlo menší spotřebu než pro vstupní kombinaci 01, proto bude i pokles  $V_{dd}$  způsobený úbytkem na vnitřním odporu zdroje a rozvodech napájení daleko menší, zotavení napájení tak bude trvat kratší dobu. Výsledkem působení zmíněných faktorů je, že zatímco pro vstupní kombinaci 10 trvá přechodný děj asi 45 ns, pro kombinaci 01 je to již 90 ns.

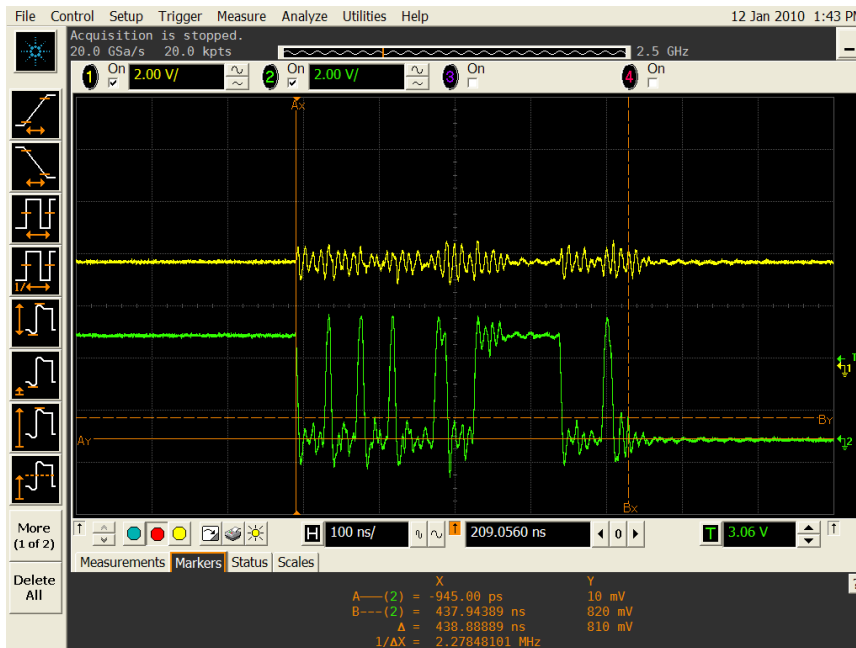


Obrázek 3.15: Změna funkce hradla z NOR na NAND při nárůstu napájecího napětí – detail pro vstupní kombinaci 10.

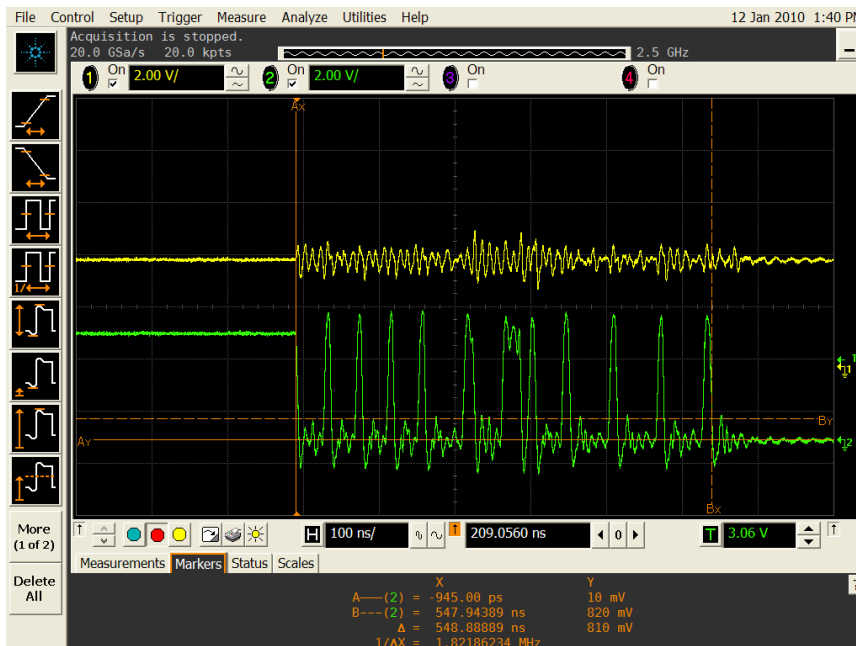


Obrázek 3.16: Změna funkce hradla z NOR na NAND při nárůstu napájecího napětí – detail pro vstupní kombinaci 01.

Daleko horší situace nastala při měření přechodného děje při změně funkce z NAND na NOR, tedy při poklesu  $V_{dd}$ . Faktorem, který patrně nejvíce zhoršuje realizaci takové změny, je nemožnost rychlého poklesu napětí v rozvedech napájení. Tomu brání blokovací kondenzátory a svůj díl má zřejmě také regulace napájecího napětí. I v případě poklesu napájecího napětí je (stejně jako při nárůstu) o něco příznivější přechodný děj při kombinaci vstupů 10. Zde trvá přechodný děj asi 440 ns, zatímco pro vstupní kombinaci 01 trvá 550 ns.

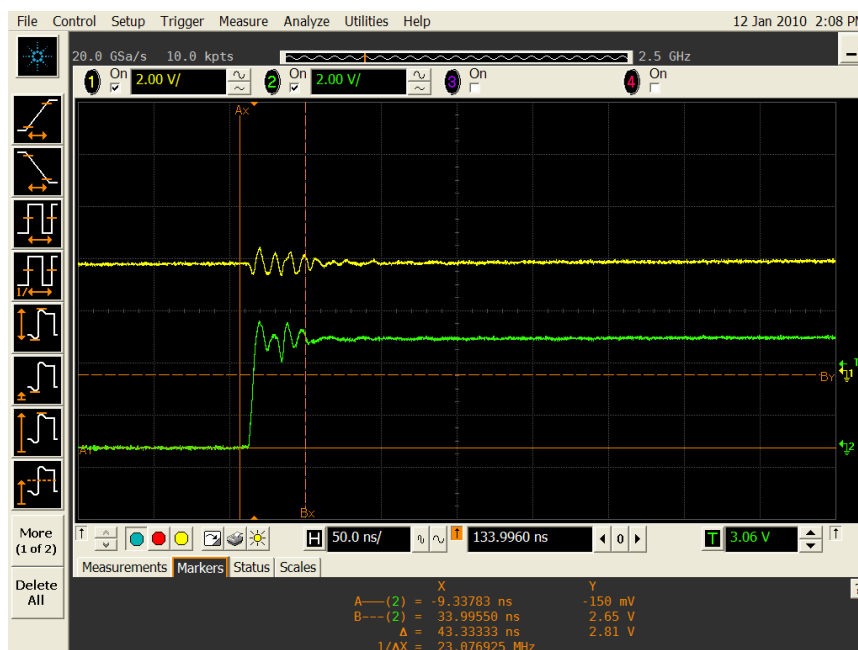


Obrázek 3.17: Změna funkce hradla z NAND na NOR při poklesu napájecího napětí – detail pro vstupní kombinaci 10.



Obrázek 3.18: Změna funkce hradla z NAND na NOR při poklesu napájecího napětí – detail pro vstupní kombinaci 01.

Je třeba říci, že přechodné děje zachycené na obrázcích 3.14 – 3.18 byly zkoumány při jediném jednoduchém zapojení hradla a jeho napájení. Napájecí zdroj ani rozvody napájení a blokovací kapacity nebyly nijak optimalizovány. I když je patrně třeba u zde popisovaného hradla vždy s nějakým přechodným dějem počítat, lze jeho následky zmírnit právě vhodným návrhem zdroje a rozvodů napájení. Na průběh přechodného děje však může mít vliv i samotné zapojení polymorfního číslicového obvodu, v němž se hradlo (hradla) nachází. Proudové rázy v napájecím rozvodu, způsobující záškrt výstupu hradla, mohou být například kompenzovány zřetěžením více polymorfních hradel. Obrázek 3.19 opět ukazuje situaci, kdy polymorfní hradlo NAND/NOR vlivem nárůstu  $V_{dd}$  mění funkci z NAND na NOR a protože jeho vstupy jsou buzeny rozdílnými hodnotami, projeví se to na výstupu změnou logické úrovně. Za tímto hradlem (na jeho výstup) jsou však navíc zapojeny ještě další dvě stejná hradla, ovšem se spojenými vstupy (chovají se jako invertory). Obrázek 3.19 ukazuje výstup druhého z nich (zelená stopa). Je vidět, že přechodný děj (zachycený například na obrázku 3.15) byl v následujícím řetězu hradel pohlcen [Šim10].



Obrázek 3.19: Změna režimu polymorfních hradel NAND/NOR řízených  $V_{dd}$ , zapojených za sebou.

	$V_{dd} = 3,3 \text{ V}$	$V_{dd} = 5 \text{ V}$
$V_{iL}$	max. 0,79 V	max. 1,09 V
$V_{iH}$	min. 1,37 V	min. 1,78 V
$V_{oL}$	0,0 V	0,0 V
$V_{oH}$	3,29 V	4,97 V

Tabulka 3.5: Statické parametry hradla

V tabulce 3.5 je vidět, jaké jsou meze akceptovaných vstupních úrovní a jaké jsou generované výstupní úrovně. Je vidět, že prahové napětí hradla, které rozlišuje mezi úrovněmi log. 0 a log. 1 na vstupu, je oproti standardu CMOS posunuto poněkud níže. To znamená, že jako log. 1 hradlo interpretuje poměrně nízké napětí. To by mohl být potenciální problém při přenosu signálu log. 0, ke kterému by se při přenosu přidal šum s amplitudou větší než 1 V. Takový šum by mělo

standardní hradlo CMOS interpretovat jako trvalou log. 0, s měřeným polymorfním hradlem by však mohl nastat problém, hradlo by se už mohlo rozkmitat. V praxi lze však takový šum či rušení očekávat spíše na spojích na úrovni desky či systému, velmi zřídka však na čipu. Přitom aplikace tohoto hradla se předpokládá právě na čipu mezi konvenčními hradly, kde spoje jsou velice krátké.

Naměřené výstupní úrovně hradla (viz tabulka 3.5) jsou velice kvalitní a snesou srovnání s konvenčními hradly CMOS. Pokles výstupního napětí v log. 1 oproti  $V_{dd}$  je minimální a nemění se výrazně, ani když je hradlo zatíženo několika dalšími hradly. Protože se předpokládá aplikace hradla na čipu v kombinaci s konvenčními CMOS hradly, nemá smysl zatížení výstupu větší zátěží než několik CMOS hradel zkoumat.

Dynamické parametry hradla z obrázku 3.8 jsou shrnuty v tabulce 3.6. Je tam uvedena doba, za kterou dojde ke změně úrovně na výstupu Y hradla (z log. 0 na log 1. –  $t_{pLH}$  a z log. 1 na log 0. –  $t_{pHL}$ ) v reakci na změnu vstupní úrovně pro oba vstupy (A i B). Je vidět, že doba propagace signálu hradlem je poměrně silně závislá jak na tom, ze kterého vstupu se signál hradlem šíří a také na tom, zda jde o vzestupnou či sestupnou hranu signálu. Jak bylo již uvedeno, hradlo má v sobě (na rozdíl od konvenčních hradel CMOS) zabudovánu jistou nesymetrii. Proto jsou u údajů v tabulce 3.6 uvedeny v závorce i tranzistory, které se na buzení výstupní úrovně v daném případě podílí. Výrazné rozdíly mezi hodnotami doby propagace signálu bylo možno (vzhledem ke konstrukci hradla) očekávat – jsou dány právě rozdílnou velikostí jednotlivých tranzistorů a také tím, že v některých případech dochází k přebíjení úrovní přiváděných některými tranzistory napětím přivedeným jiným tranzistorem.

	$V_{dd} = 3,3 \text{ V}$	$V_{dd} = 5 \text{ V}$
$t_{pLH} \text{ A} \rightarrow \text{Y}$	65 ns (M1)	3 ns (M1 + M7)
$t_{pHL} \text{ A} \rightarrow \text{Y}$	56 ns (M8)	25 ns (M2 + M8)
$t_{pLH} \text{ B} \rightarrow \text{Y}$	81 ns (M1)	3 ns (M1 + M7)
$t_{pHL} \text{ B} \rightarrow \text{Y}$	71 ns (M2)	26 ns (M2 + M8)

Tabulka 3.6: Doba propagace signálu hradlem v obou režimech pro oba vstupy.

Z hodnot uvedených v tabulce 3.6 lze odvodit mezní frekvenci hradla, tedy maximální frekvenci, se kterou se mohou měnit logické hodnoty, zpracovávané hradlem. V konvenčním pojetí návrhu by se mělo vycházet z nejvyšší hodnoty naměřeného zpoždění signálu, což je 81 ns. V takovém případě by to bylo asi 12,3 MHz. To není ve srovnání s mezní frekvencí konvenčních hradel mnoho. Avšak je třeba vzít v úvahu, že poměrně nízkou mezní frekvencí je vykoupena jednoduchost hradla (které vykonává dvě různé funkce a obsahuje ještě vestavěný detektor úrovně  $V_{dd}$ ).

Vzhledem k tomu, jakým způsobem je u hradla řešena detekce úrovně  $V_{dd}$  (tranzistory M3 a M4 pracují v lineárním režimu), lze očekávat závislost změny funkce hradla také na teplotě. Za normálních okolností není změna chování číslicového obvodu vlivem okolí žádoucí, avšak u polymorfního obvodu jistá závislost funkce na vlivech prostředí žádoucí je. Vedle  $V_{dd}$  to může být právě teplota, která může ovlivňovat funkci realizovanou hradlem. Jedno polymorfní hradlo závislé na teplotě již (ve dvou variantách) bylo navrženo [Sto01], avšak žádné jeho konkrétní aplikace popsány nebyly.

Polymorfní hradlo z obrázku 3.8 bylo proto podrobena dalším experimentům, při kterých byl čip s hradly zahříván na teplotu až 140°C. Bylo využito obvodu REPOMO, který hradla obsahuje, a kitu pro experimenty s tímto obvodem, který byl pro tyto účely vybaven topnou komorou s měřením a regulací teploty obvodu (podrobnosti viz kapitola 5).

Při prvním experimentu byl čip napájen  $V_{dd} = 3,3$  V (hradlo tedy bylo v režimu NOR) a postupně byla zvyšována teplota. Ukázalo se, že hradlo při překročení určité teploty (asi  $125^{\circ}\text{C}$ ) nejprve přejde do zcela nového (třetího) režimu, kdy realizuje funkci negace vstupu B (NOT B – viz poznámka 3.2 a příklad 3.1 na začátku této kapitoly). Další zvyšování teploty (přes  $135^{\circ}\text{C}$ ) vede k tomu, že hradlo podruhé změní režim na NAND.

Je třeba poznamenat, že pokud je hradlo ovládáno napájecím napětím, režim NOT B u něj prakticky nebyl pozorován, hradlo i při velmi pozvolném nárůstu či poklesu napájecího napětí přechází relativně krátkým (řádově desítky ns) přechodným dějem (kdy výstup může několikrát zakmitat) z režimu NOR přímo do režimu NAND, případně zpět. Kmitání výstupu se však projevuje pouze právě tehdy, pokud má výstup při změně režimu změnit úroveň – tj. pokud na vstupech jsou rozdílné úrovně (viz řádky 2 a 3 tabulky 3.3). Nelze tedy mluvit o dalším režimu ani při přechodném ději, jde pouze o několikrát rychlé přepnutí mezi novým a starým režimem, než se hradlo definitivně ustálí v novém režimu. Naproti tomu při řízení hradla teplotou experimenty ukázaly, že hradlo vykazuje další (třetí) režim, režim funkce NOT B, i staticky.



Obrázek 3.20: Hradlo mění funkci z NOR na NOT B při teplotě asi  $124^{\circ}\text{C}$  ( $V_{dd} = 3,3$  V).

Co je třeba, aby hradlo z obrázku 3.8 fungovalo v režimu NOT B? Pohled do tabulky 3.3 napovídá, že režim NOT B se od režimů NOR liší jen v jednom řádku – v řádku 3. Aby se změnil režim hradla z NOR na NOT B, musí pro vstupní kombinaci 10 (řádek 3 tabulky 3.3) být na výstupu log. 1. Tu přivádí na výstup tranzistor M7, pokud jsou otevřeny M3 a M4. Ty pracují v lineárním režimu, lze proto předpokládat, že budou ovlivňovány teplotou nejvíce. Tento tranzistor přivádí log. 1 na výstup i v případě vstupní kombinace 00 a v případě vstupní kombinace 01. Pro funkci NOT B ale vstupní kombinace 01 znamená, že na výstupu musí zůstat log. 0 (jako v režimu NOR), tranzistor M7 se proto nesmí „se svojí“ log. 1 na výstupu prosadit. Čím se liší chování tranzistorů hradla pro vstupní kombinace 01 a 10? Pro vstupní kombinaci 01 přivádí na výstup log. 0 (pokud nepřeváží M7 se svou log. 1) tranzistor M2, pro vstupní kombinaci 10 pak tranzistor M8. Přitom tranzistor M8 je navržen jako podstatně slabší než M2, protože je překonáván pouze M7, kdežto M2 je překonáván dvojicí M1 a M7. Patrně díky tomu je při zahřívání hradla dříve (při nižší teplotě) překonán tranzistorem M7 tranzistor M8 než M2 a tudíž pro vstupní kombinaci 10 se objeví na výstupu log. 1 při nižší teplotě než



pro vstupní kombinaci 01. Obrázek 3.20 ukazuje situaci, kdy při teplotě asi 124°C začíná hradlo měnit svoji funkci z NOR na NOT B – přibývá nový „jedničkový“ vrchol funkce pro vstupní kombinaci 10. Je zachycen přechodný děj – výstup nemá ještě trvání log. 1 po celou dobu, kdy na vstupech je kombinace 10, jsou vidět zákmity. Při dosažení teploty 125°C však již hradlo dokonale realizuje funkci NOT B (viz obrázek 3.21).



Obrázek 3.21: Hradlo realizuje spolehlivě funkci NOT B při teplotě 125°C ( $V_{dd} = 3,3$  V).



Obrázek 3.22: Hradlo mění funkci na NAND – přechodový děj při teplotě asi 135°C ( $V_{dd} = 3,3$  V).

Pokud se čip s polymorfním hradlem z obrázku 3.8 ohřívá dále nad teplotu 125°C, dojde při teplotě asi 135°C k další změně režimu. Hradlo začne realizovat funkci NAND tak, jako by se napájecí napětí zvedlo nad 4 V, i když  $V_{dd}$  zůstává na úrovni 3,3 V. Opět lze, tak jako při přechodu od funkce NOR k NOT B (při teplotě o asi 10°C nižší), pozorovat jistý přechodný děj, kdy výstup hradla pro vstupní kombinaci 01 (při níž se liší hodnota výstupu pro funkci NOT B a NAND) kmitá (zachycuje to obrázek 3.22). Při teplotě 137°C je již výstup pro všechny vstupní kombinace stabilní a odpovídá funkci NAND – viz obrázek 3.23.



Obrázek 3.23: Hradlo realizující spolehlivě funkci NAND při teplotě 137°C ( $V_{dd} = 3,3$  V).

Je třeba poznamenat, že uvedené teploty platí pro konkrétní realizaci čipu (pro experimenty zobrazené na obrázcích 3.20 – 3.23 byl použit čip REPOMO výr. č. 15), pro jiné čipy se díky rozptýlu parametrů použité technologie teplotní meze, při nichž dochází ke změně režimu hradla, liší, avšak ne více než asi o  $\pm 3\%$ .

Jestliže polymorfní hradlo z obrázku 3.8, jehož funkci lze řídit změnou úrovně  $V_{dd}$ , dokáže změnit realizovanou funkci také působením teploty, i když je úroveň  $V_{dd}$  stále stejná, jak se bude v závislosti na teplotě chovat, pokud se i  $V_{dd}$  bude měnit? Jestliže změna chování hradla je způsobena změnou pracovního bodu tranzistorů M3 a M4 a tuto změnu je možno způsobit jak změnou  $V_{dd}$ , tak změnou teploty, bude zřejmě možné úrovní  $V_{dd}$  ovlivnit, při jaké teplotě hradlo změní funkci, a naopak. Tuto hypotézu se podařilo experimentálně potvrdit, výsledky experimentu ukazuje tabulka 3.7.

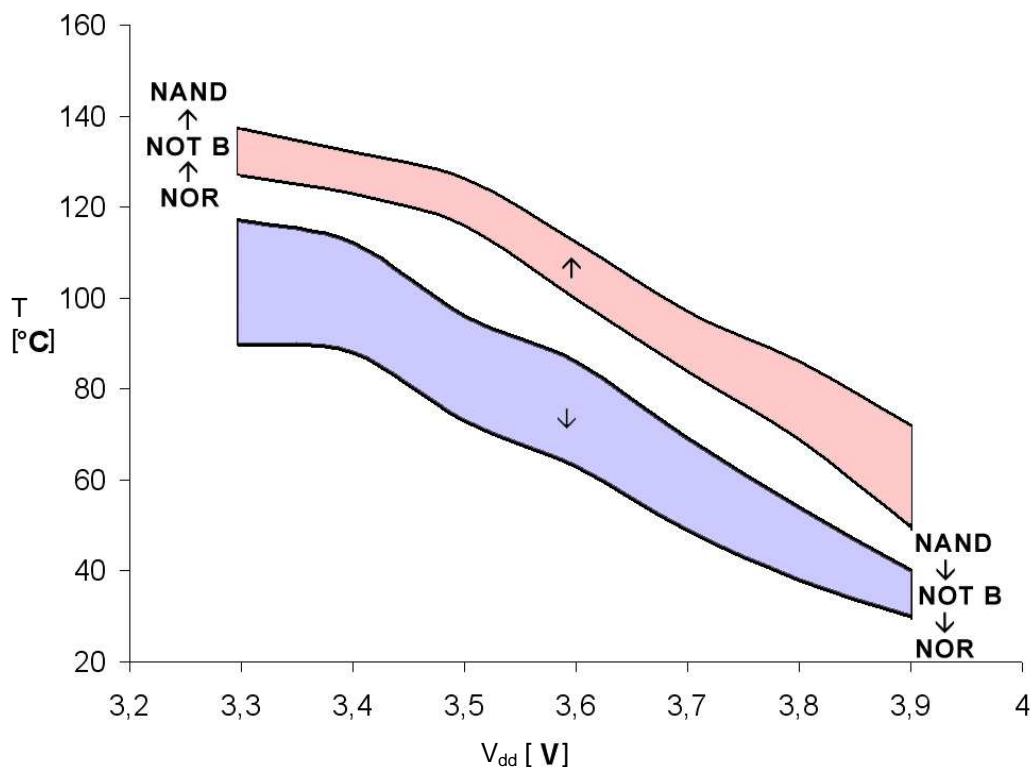
V tabulce 3.7 je vidět, že čím vyšší je napájecí napětí, tím nižší teplota čipu stačí, aby hradlo změnilo režim činnosti. Bylo zkoumáno chování hradla při  $V_{dd}$  v rozsahu 3.3 V (což je nominální napájecí napětí pro základní režim činnosti hradla – NOR) až 3.9 V. Pro vyšší  $V_{dd}$  než 3.9 V již dochází ke změně režimu hradla na NAND i při běžné teplotě a zvyšování teploty již chování hradla dále nemůže ovlivnit, hradlo setrvává v režimu NAND.

V <sub>dd</sub>	NOR → NOT B	NOT B → NAND	NAND → NOT B	NOT B → NOR
3,3	127°C	137°C	117°C	90°C
3,4	123°C	132°C	112°C	88°C
3,5	116°C	126°C	96°C	73°C
3,6	100°C	112°C	86°C	63°C
3,7	84°C	97°C	69°C	49°C
3,8	69°C	86°C	54°C	38°C
3,9	50°C	72°C	40°C	30°C

Tabulka 3.7: Vztah teploty, při níž se mění funkce hradla, a napájecího napětí.

Levá polovina tabulky, podbarvená červeně, přitom ukazuje teploty, při nichž se mění funkce hradla, pokud se čip ohřívá (teplota stoupá). Naproti tomu pravá polovina tabulky, modře podbarvená, ukazuje teploty, při nichž se mění funkce hradla, pokud se čip ochlazuje. Je tedy vidět, že i při řízení funkce hradla teplotou se uplatňuje jistá hystereze (srov. obr. 3.12 a 3.13, kde je vidět hystereze při řízení hradla pomocí V<sub>dd</sub>). Pro úplnost je třeba dodat, že tento experiment byl proveden s čipem REPOMO v. č. 19.

Graf na obrázku 3.24 přehledněji znázorňuje, jak lze napájecím napětím ovlivnit teplotní meze, při nichž hradlo mění svůj režim. Je vidět, že teplotu, při které hradlo změní realizovanou funkci, lze bez problémů nastavit právě napájecím napětím. Tato skutečnost dává hradlu velkou flexibilitu, pokud jde o jeho využití v obvodech, které mají reagovat na změnu teploty.

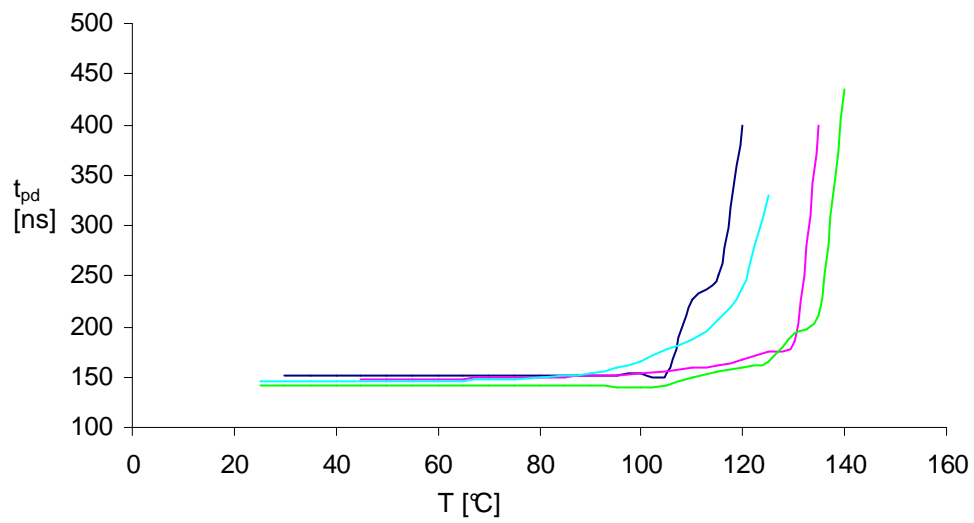


Obrázek 3.24: Závislost režimu hradla na teplotě a napájecím napětí.

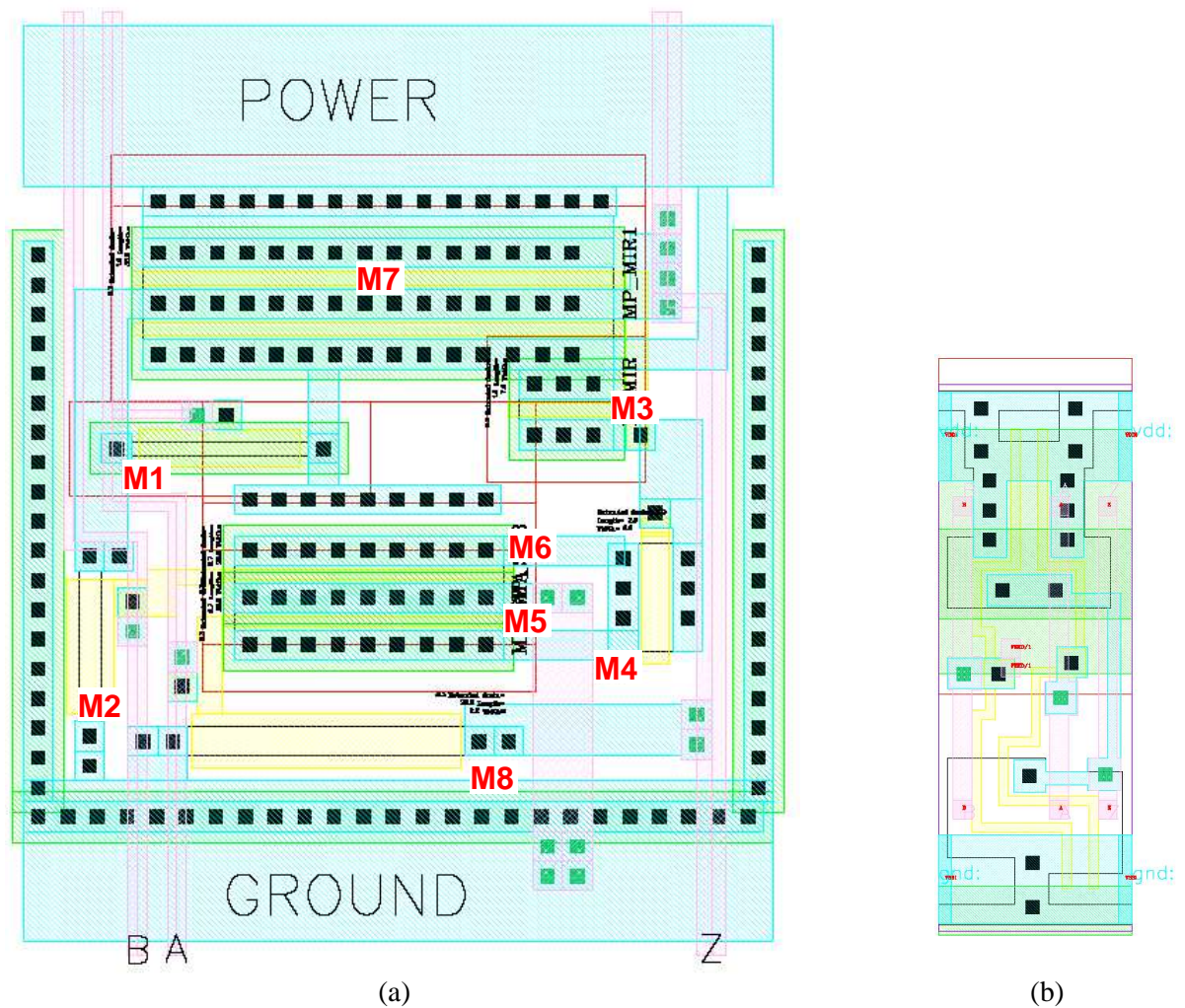
Součástí experimentů se změnou teploty čipu s hradlem z obrázku 3.8 bylo také měření dynamických parametrů hradla. Tabulka 3.8 ukazuje, jak se mění zpoždění (doba propagace signálu hradlem) při rostoucí teplotě čipu s hradlem. Jak bylo uvedeno výše, experimenty se změnou teploty čipu byly prováděny na kitu s obvodem REPOMO (protože kit byl navržen s podporou pro tento druh experimentů – viz kapitola 5). Proto jsou hodnoty  $t_{pd}$  i při normální teplotě mnohem vyšší než je uvedeno pro samostatné hradlo v tabulce 3.6. Čip REPOMO totiž neumožňuje měřit pouze jediné hradlo, proto hodnoty naměřené a uvedené v tabulce 3.8 jsou součtem zpoždění polymorfního hradla a tří konvenčních. Protože experimenty ukázaly, že zpoždění konvenčních hradel užitých na čipu REPOMO není příliš teplotně závislé a pohybuje se kolem 20 ns, odráží hodnoty v tabulce 3.8 závislost polymorfního hradla na teplotě celkem dobře. S teplotou se zpočátku ani zpoždění polymorfního hradla příliš nemění, rapidní zhoršení parametrů nastává až nad 120 °C. Obrázek 3.25 ukazuje závislost zpoždění na teplotě pro hradla na čtyřech různých náhodně vybraných vyrobených čipech (čipy REPOMO vyr. č. 15, 19, 25 a 26).

v.č.	25	26	15	19
T[°C]	$t_{pd}$ [ns]	$t_{pd}$ [ns]	$t_{pd}$ [ns]	$t_{pd}$ [ns]
25			141	146
30	151,9		141,5	146
35	151,7		141,5	146
40	151,6		141,4	146
45	151,5	148	141,3	146
50	151,5	148,1	141,2	146
55	151,4	148,2	141,2	146
60	151,3	148,3	141,1	146
65	151,2	148,4	141	146
70	151,2	148,6	140,9	147
75	151,2	148,9	140,8	148
80	151,2	149,3	140,7	149
85	151,3	149,9	140,6	152
90	151,5	150,7	140,6	154
95	151,9	151,9	140,5	159
100	152,6	153,3	140,5	165
105	153,6	155,6	140,6	177
110	227	158,6	150,2	188
115	250	161,3	156	206
120	400	167,4	159,8	238
125		175	165,7	330
130		184,6	193,6	
135		400	210,8	
140			435,6	

Tabulka 3.8: Zpoždění hradla v závislosti na teplotě.



Obrázek 3.25: Závislost zpoždění hradla  $t_{pd}$  na teplotě.



Obrázek 3.26: Fyzická realizace hradla na křemíkovém čipu technologií AMIS 0,7  $\mu\text{m}$  (a) ve srovnání s obyčejným hradlem NAND (b) vyrobeném stejnou technologií.

Na obrázku 3.26(a) je fyzická struktura hradla tak, jak je realizováno na křemíkovém čipu. Jednotlivé tranzistory jsou v obrázku popsány a jsou zcela zřejmé proporce jejich velikostí (viz též tabulka 3.4). V pravé části obrázku (3.26 (b)) je konvenční hradlo NAND, realizované stejnou technologií. Kanály těchto tranzistorů jsou dlouhé jen 0,7  $\mu\text{m}$ , zato široké 7 či 14  $\mu\text{m}$  pro n-MOS resp. p-MOS tranzistor. Polymorfnímu hradlu zcela dominuje tranzistor M7, který je navržen tak, aby v případě změny funkce hradla jím na výstup přivedená úroveň přebila ostatní tranzistory. Musí to tak být, protože toto polymorfní hradlo (na rozdíl od konvenčního) může mít za normálních okolností otevřen současně jak p-MOS, tak i n-MOS tranzistor a je třeba jednoznačně definovat, který převáží a určí tak spolehlivě výstupní úroveň. Je vidět, že tranzistory M1, M2 a M8, které přivádí úroveň na výstup hradla, mají oproti tranzistorům se stejnou rolí v konvenčním hradle kanál užší a delší, mají tedy větší odpor v otevřeném stavu. Tranzistory M5 a M6 mají velikost téměř srovnatelnou s tranzistory konvenčního hradla, mají také podobnou roli – realizují logickou funkci, která povoluje změnu funkce hradla pro určité vstupní kombinace a pro jinou zakazuje. Tranzistory M3 a M4 slouží jako detektor napětí, jejich rozměry nemá smysl srovnávat s rozměry tranzistorů zajišťujících logickou funkci v konvenčním hradle. Na obrázku 3.26(b) je uvedeno konvenční hradlo NAND (4 tranzistory), vyrobené stejnou technologií, pro porovnání rozměrů. Zatímco polymorfní hradlo zabírá na čipu plochu 55 x 68  $\mu\text{m}$ , konvenční hradlo NAND jen asi 15 x 40  $\mu\text{m}$ . Na druhou stranu, polymorfní hradlo je schopno realizovat dvě (nebo tři) různé funkce, mezi kterými se přepíná, a obsahuje také detektor  $V_{\text{dd}}$ , případně teploty.

Parametry a splnění požadavků:

Požadavek			
P1	vstupní odpor	hradla tranzistoru MOS	✓✓
P2	výstupní odpor	závisí na stavu (sepnutém tranzistoru)	✓
P3	napájecí napětí	3 – 5,5V	✓✓
P4	vstupní úroveň	Pod 0,8/1,1V log. 0, nad 1,4/1,8 V log. 1 pro $V_{\text{dd}} = 3,3/5$ V	✓✓
P5	výstupní úroveň	0/3,29 V pro $V_{\text{dd}} = 3,3$ V; 0/4,97 V pro $V_{\text{dd}} = 5$ V	✓✓
P6	$t_{\text{pd}}$	desítky ns	✓
P7	rozměry	8 tranzistorů	✓
P8	spotřeba	V režimu NAND pro kombinaci vstupů 01 až 100 $\mu\text{A}$ . V režimu NOR pod 1 $\mu\text{A}$	✗

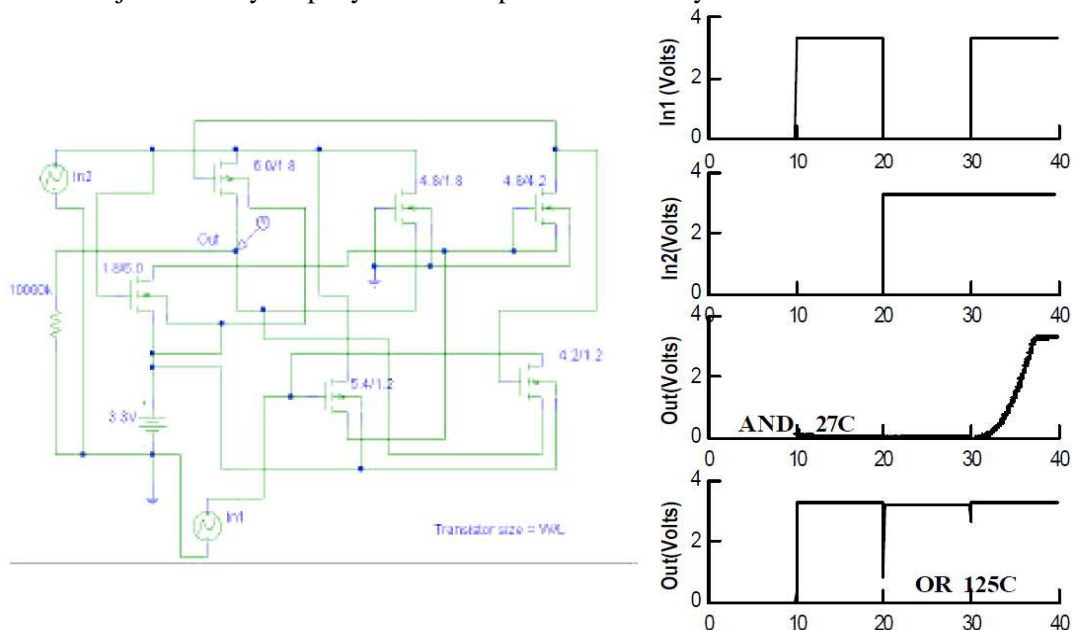
Tabulka 3.9: Sledované parametry hradla a splnění požadavků kladených na polymorfní hradla.

### 3.5.2 Hradla řízená teplotou

Zatím neexistuje příliš mnoho polymorfních hradel řízených teplotou. Patrně je to způsobeno tím, že oblast aplikací polymorfní elektroniky není zatím příliš bohatá a řízení funkce obvodu teplotou je chápáno spíše jako zajímavá hříčka. Na druhou stranu, číslicové obvody realizované na bázi anorganických polovodičů vykazují poměrně silnou teplotní závislost (i když je přirozenou snahou ji u konvenčních obvodů co nejvíce eliminovat a u dnes masivně používaného křemíku je mezi polovodiči ještě poměrně malá) a tak lze očekávat, zejména u obvodů (hradel), kde některé tranzistory pracují v lineárním režimu, že teplota jejich funkci ovlivňovat bude. Pozitivně to bylo prokázáno na hradle NAND/NOR, které bylo navrženo týmem z FIT a FEKT VUT jako řízené napájecím napětím (viz hradlo z obrázku 3.8). Toto hradlo, jak bylo popsáno výše, také mění funkci s teplotou [Ruz11a].

## AND/OR Stoica #1

Hradlo, zobrazené na obrázku 3.27 (převzato z [Sto01]) bylo navrženo evolucí v týmu A. Stoicy v NASA JPL. Návrh byl proveden s využitím SPICE modelu, jako fitness funkce byla zvolena kombinace kvality řešení (blížkost ideálním řešením) při dvou teplotách pro dva režimy (AND a OR) hradla. Výsledkem je hradlo s šesti tranzistory, které realizuje funkci AND při teplotě 27°C, při teplotě 125°C realizuje funkci OR. Jak je vidět na obrázku 3.27 vpravo, odezvy na vstupní signál nejsou příliš kvalitní, zejména v režimu AND, kdy log. 1 na výstupu nabíhá velmi pozvolně – hodnoty na časových osách jsou v milisekundách! Takto se chovající hradlo zřejmě nemá příliš velké uplatnění v reálných obvodech. Bohužel nebylo publikováno (a patrně ani ověřováno), jak se hradlo chová pro teploty ležící mezi dvěma sledovanými body, jak vypadá přechodný děj při změně funkce, když teplota pozvolna roste, a ani jak se hradlo chová při jiném napájecím napětí než při  $V_{dd} = 3,3V$ , při kterém byla prováděna simulace. Je třeba poznamenat, že polymorfního chování je zde také dosaženo velmi rozdílnou velikostí tranzistorů (rozhodně nejde o pouhou aplikaci tranzistorů stejné velikosti, běžné při návrhu konvenčních CMOS hradel). Tranzistory zřejmě pracují v lineárním režimu a spotřebu hradla lze odhadovat jako poměrně vysokou (zejména v porovnání s konvenčními CMOS hradly). Nectnosti jsou však vykoupeny velmi kompaktními rozměry.



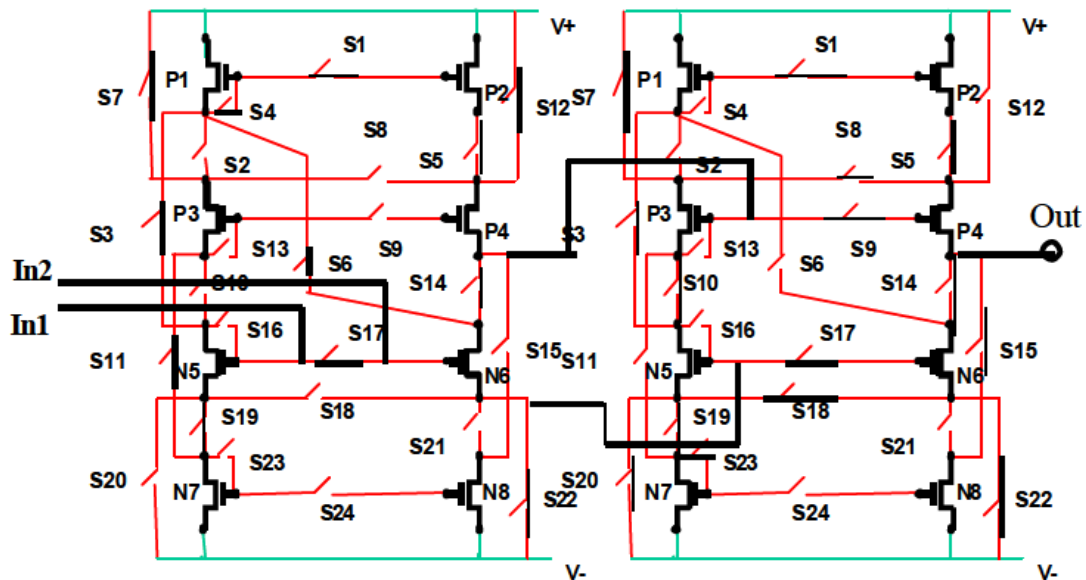
Obrázek 3.27: Teplotou řízené polymorfní hradlo AND/OR a jeho chování.

Požadavek			
P1	vstupní odpor	vstupy jsou využívány jako zdroje pro výstup	✗
P2	výstupní odpor	výstup je generován ze vstupů	✗
P3	napájecí napětí	$V_{dd} = 3,3V$	✓
P4	vstupní úrovně		?
P5	výstupní úrovně	blíží se $V_{dd}$ , je ale otázka, jaké budou reálně vzhledem k výstupnímu odporu hradla	? ✓
P6	$t_{pd}$	zejména v režimu AND je až 5 ms!	✗
P7	rozměry	6 tranzistorů s relativně malým rozměrem kanálu	✓✓
P8	spotřeba	patrně velká v některých stavech	? ✗

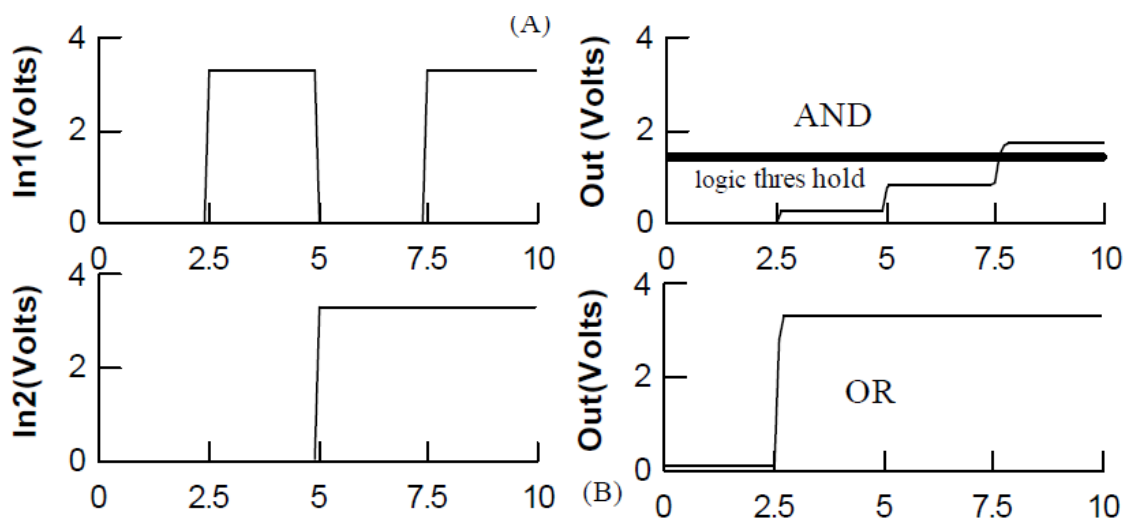
Tabulka 3.10: Sledované parametry hradla a splnění požadavků kladených na polymorfní hradla.

## AND/OR Stoica #2

Dalším experimentálním polymorfním hradlem navrženým evolucionářským týmem A. Stoicy [Sto01] je opět teplotou řízené hradlo AND/OR. Jeho schéma je vidět na obrázku 3.28 (obrázek převzat ze [Sto01]). Tentokrát jde však o evoluční návrh přímo v hardware s využitím dvou buněk programovatelného obvodu FPTA (Field-Programmable Transistor Array) [Sto00], [Zeb00]. Při návrhu byl obvod umístěn v komoře s regulovanou teplotou. Celé hradlo sestává z osmi tranzistorů, přičemž vstupy jsou zapojeny do první z užitých čtyřtranzistorových buněk, výstup je vyveden z druhé buňky. Každá buňka je konfigurovatelná pomocí 24 spínačů (na obrázku jsou označeny S1 – S24). Obvod se chová jako hradlo AND pro teplotu 5°C, pro teplotu 90°C se chová jako OR. Jak je vidět z průběhů na obrázku 3.29, chování obvodu je oproti předchozímu popsanému hradlu AND/OR o něco kvalitnější, zejména pokud jde o rychlost odezvy (časové osy opět v ms). Výstupní úrovně v režimu AND jsou však nepříliš kvalitní. Pozoruhodné je, že obě hradla AND/OR mají horší chování při nižších teplotách než při vyšších. Přitom konvenční hradla jsou vždy navrhována tak, aby pracovala spolehlivě při teplotách blízkých pokojové teplotě, při teplotách nad 70°C se jejich parametry naopak zhoršují.



Obrázek 3.28: Teplotou řízené hradlo AND/OR.



Obrázek 3.29: Chování teplotou řízeného hradla AND/OR z obrázku 3.25



Požadavek			
P1	vstupní odpor	hradla tranzistoru MOS	✓✓
P2	výstupní odpor	sepnutý tranzistor	✓✓
P3	napájecí napětí	$V_{dd} = 3,3V$	✓
P4	vstupní úrovně		?
P5	výstupní úrovně	v režimu AND je log. 1 pod 2V	✗
P6	$t_{pd}$		✓
P7	rozměry	8 tranzistorů	✓✓
P8	spotřeba	patrně velká v některých stavech	? ✗

Tabulka 3.11: Sledované parametry hradla a splnění požadavků kladených na polymorfni hradla.

### 3.5.3 Hradla řízená speciálním signálem

Polymorfni hradla řízená speciálním signálem tvoří specifickou třídu hradel. Splňují definici 3.1, tedy provádí více než jednu funkci podle stavu prostředí, avšak prostředí v tomto případě degraduje na další elektrický signál, tedy informace o prostředí je stejného charakteru, jako informace obvodem zpracováváné. Dokonce pokud jde o hradlo realizující právě dvě funkce, je prostředím signál nesoucí binární informaci, tedy vlastně logický signál. Takové polymorfni hradlo s  $n$  vstupy a dvěma funkcemi je pak vlastně „jen“ konvenčním hradlem s  $n+1$  vstupem. I takové hradlo ale autoři myšlenky polymorfni elektroniky považují za polymorfni (viz [Sto01]). Smysl těchto hradel lze spatřovat hlavně v tom, že jde o úspornou implementaci složitější logické funkce. Jsou dobře použitelná pro experimentování, případně evoluční návrh složitějších polymorfni obvodů. Přepínání mezi funkcemi je totiž velice snadné, říditelné běžnými logickými obvody, což je při evolučním návrhu v HW velmi užitečné.

#### NAND/NOR McDermott

V roce 1997 (tedy před tím, než tým Adriana Stoicy formuloval principy polymorfni číslicové elektroniky) si Mark McDermott a John Turner z Cyrix Corporation dali patentovat hradlo, které realizuje jednu ze dvou logických funkcí – buď funkci NAND nebo funkci NOR, podle toho, jaké hodnoty nabývá speciální řídicí signál  $C$  [McD97]. Předpokládali využití v poli náhradních prvků na čipu, na němž je realizován procesor. Po vyrobení je čip testován a v případě zjištěných chyb mohou být některá vadná hradla nahrazena z pole náhradních hradel pouhým přidáním další vrstvy vodivých propojek (tedy bez zásahu do křemíku). Jsou-li náhradní hradla vícefunkční (v kontextu současného poznání lze říci i polymorfni), podstatně se podle autorů zvyšuje flexibilita oprav. Autoři také předpokládají možné nasazení v programovatelných logických polích.

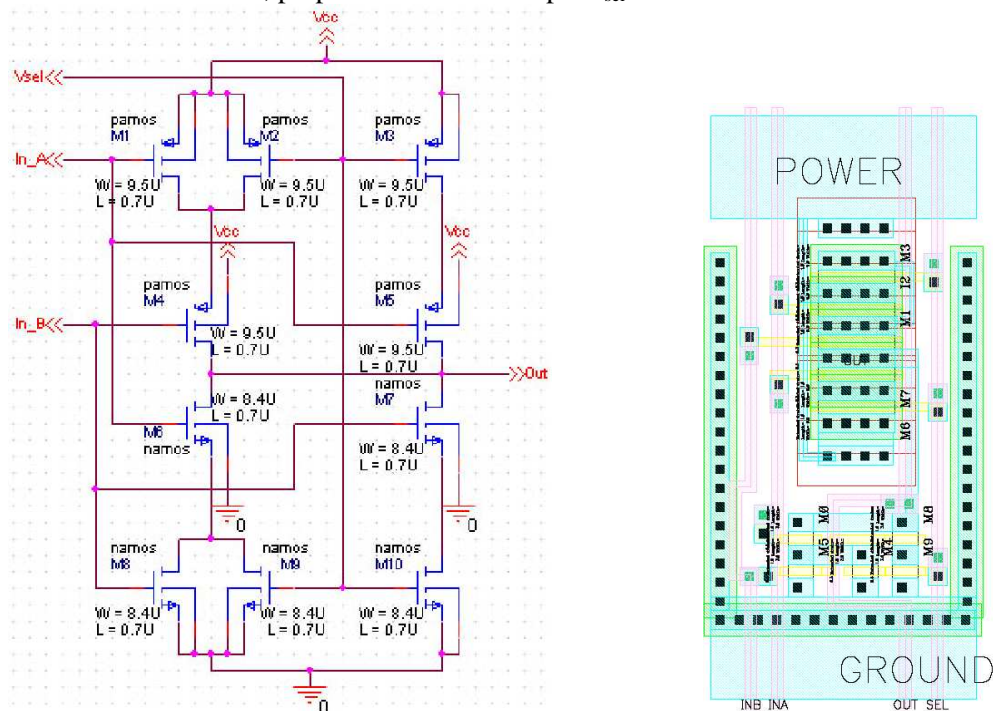
Hradlo vzniklo v podstatě sloučením konvenčního CMOS NAND a NOR hradla. Zatímco hradlo NAND má p-MOS tranzistory, které spínají  $V_{dd}$  na výstup, zapojeny paralelně (postačuje log. 0 na libovolném vstupu, aby výstup měl hodnotu log. 1), zatímco n-MOS tranzistory, které spínají na výstup zem, v sérii (je třeba log. 1 na obou vstupech, aby na výstupu byla log. 0), u NOR hradla je tomu přesně naopak. Proto autoři vložili do designu konfigurační tranzistory C1 a C2 (viz obrázek 3.30, převzatý z [McD97]), které pro vstup  $C=0$  ruší (zkratem) logický součin vstupů na p-MOS tranzistorech a pro  $C=1$  ruší logický součin vstupů na n-MOS tranzistorech. Pro vstup  $C$  na úrovni blízké zemi se tak hradlo chová jako NAND, pro  $C$  na úrovni blízké  $V_{dd}$  se chová jako NOR. Protože jsou p-MOS tranzistory výhradně využity pro vedení  $V_{dd}$  a n-MOS k vedení potenciálu země na výstup (to je v souladu s pravidly dobrého návrhu hradel





### NAND/NOR Stareček

L. Stareček navrhl v rámci práce na své disertaci v oblasti diagnostiky s využitím polymorfní elektroniky polymorfní hradlo realizující dvojici funkcí NAND a NOR s dodržováním základních principů návrhu hradel CMOS [Ruz07b], [Ruz08b]. Jak již bylo několikrát řečeno, funkce NAND a NOR jako logicky úplné jsou výhodné zejména v případě užití evolučního návrhu (snazší nalezení validního řešení) a také proto je autor hradla zvolil. Hradlo sestává z deseti tranzistorů, prakticky jsou využity tranzistory běžných rozměrů, jaké se užívají pro realizaci konvenčních hradel CMOS. Tranzistory typu p-MOS jsou důsledně využívány ke spínání log. 1, tranzistory typu n-MOS zase pro spínání log. 0. Hradlo je v podstatě ekvivalentní patentovanému řešení McDermotta a Turnera jak po stránce rozměrů, tak i po kvalitativní stránce. Hradlo bylo fyzicky realizováno technologií AMIS 0,7  $\mu\text{m}$ . Obrázek 3.32 ukazuje schéma zapojení hradla a také jeho fyzickou realizaci. Rozměry jsou 31 x 68  $\mu\text{m}$ . Funkci, realizovanou hradlem, přepíná úroveň na vstupu  $V_{\text{sel}}$ .



Obrázek 3.32: Hradlo NAND/NOR, řízené speciálním signálem, navržené L. Starečkem a jeho fyzická realizace.

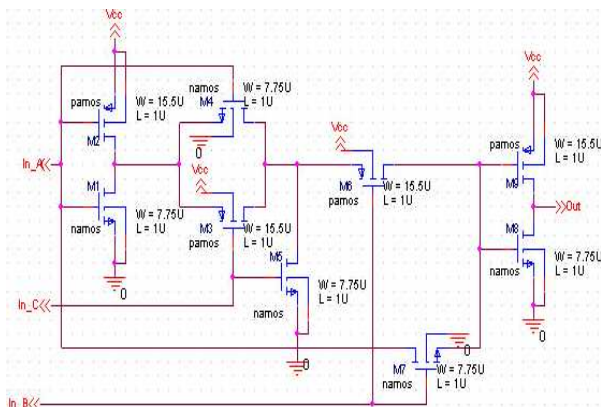
Požadavek			
P1	vstupní odpor	hradla tranzistoru MOS	✓✓
P2	výstupní odpor	sepnutý tranzistor obvyklé velikosti	✓✓
P3	napájecí napětí	v obvyklém rozsahu (do 5 V)	✓✓
P4	vstupní úroveň	v obvyklém rozsahu	✓✓
P5	výstupní úroveň	v obvyklém rozsahu	✓
P6	$t_{\text{pd}}$	patrně v obvyklém rozsahu, nebylo zkoumáno	?
P7	rozměry	10 tranzistorů	✓
P8	spotřeba	v obvyklém rozsahu	✓✓

Tabulka 3.14: Sledované parametry hradla a splnění požadavků kladených na polymorfní hradla.

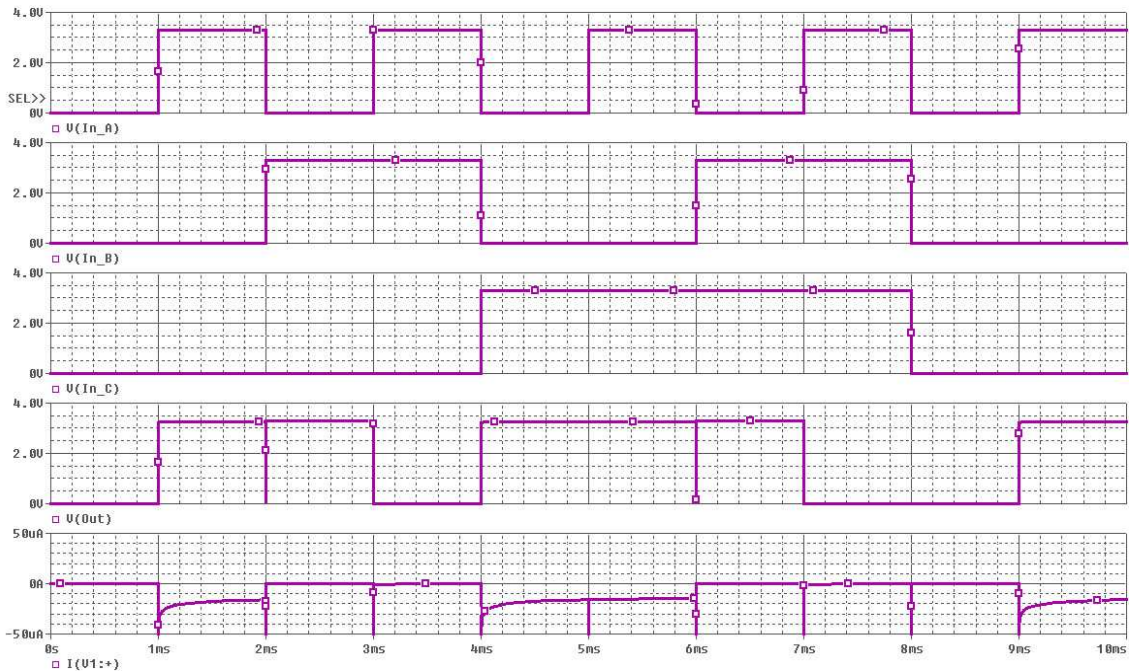
### NAND/XOR Růžička

Obrázek 3.33 ukazuje schéma dalšího polymorfního hradla řízeného speciálním signálem, které navrhl autor této práce [Ruz08b] pro účely experimentování s konvenčním návrhem polymorfních sekvenčních obvodů (viz kapitola 7). Výhodou hradla NAND/XOR je, že obě funkce se liší jen v jednom vrcholu (řádku tabulky), což může návrháři usnadnit návrh.

Hradlo sestává z devíti tranzistorů, což činí hradlo konkurenceschopným. V porovnání s konvenčním řešením, kdy hradlo NAND lze sestavit ze 4 tranzistorů a hradlo XOR z dalších 4 – 14ti tranzistorů (podle typu implementace a požadované kvality) [Bui00], [Ves99] a další tranzistory bude třeba na multiplexor výstupu, je navržené hradlo velmi prostorově úsporné. Použité tranzistory jsou stejné, jako se v dané technologii používají pro implementaci konvenčních číslicových hradel, proto je možné přímé srovnání velikosti pomocí počtu tranzistorů. Výstupní úrovně hradla jsou kvalitní díky použití tradičního zapojení s komplementární dvojicí tranzistorů budící výstup přímo z rozvodů napájení. Hradlo nebylo implementováno, ale bylo simulováno v profesionálním návrhovém systému pro integrované obvody vyráběné technologií AMIS 0,7  $\mu\text{m}$ . Vstupy A a B hradla jsou datové, vstupem C se řídí funkce hradla. Pokud je na tomto vstupu napětí blízké potenciálu země, hradlo realizuje funkci XOR. V případě, že napětí na tomto vstupu je blízké  $V_{\text{dd}}$ , hradlo realizuje funkci NAND. Simulace ukázala, že hradlo se chová podle očekávání, generuje kvalitní výstupní úrovně, pouze v některých situacích má oproti konvenčním hradlům mírně vyšší statickou spotřebu. Výsledky simulace ukazuje obrázek 3.34. Nejspodnější průběh zobrazuje právě spotřebu hradla. V kritických situacích dosahuje statická spotřeba asi 20  $\mu\text{A}$ . V porovnání se spotřebou polymorfních hradel řízených napájecím napětím nebo teplotou to však není tak mnoho.



Obrázek 3.33: Hradlo NAND/XOR, řízené speciálním signálem, navržené autorem práce.



Obrázek 3.34: Chování a spotřeba hradla NAND/XOR.

Požadavek			
P1	vstupní odpor	hradla tranzistoru MOS	✓✓
P2	výstupní odpor	sepnutý tranzistor obvyklé velikosti	✓✓
P3	napájecí napětí	v obvyklém rozsahu (simulováno pro 3,3 V)	✓✓
P4	vstupní úroveň	v obvyklém rozsahu	✓✓
P5	výstupní úroveň	v obvyklém rozsahu	✓
P6	$t_{pd}$	patrně v obvyklém rozsahu, nebylo zkoumáno	?
P7	rozměry	9 tranzistorů	✓
P8	spotřeba	v některých stavech mírně vyšší	✓

Tabulka 3.15: Sledované parametry hradla a splnění požadavků kladených na polymorfní hradla.

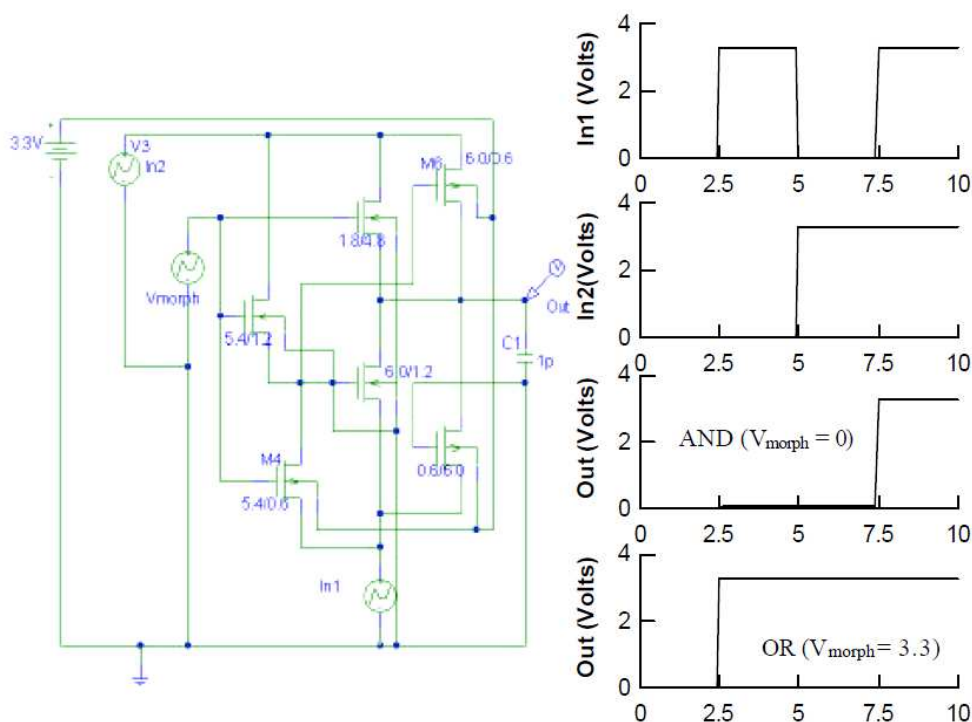
### AND/OR Stoica

A Stoica a kolektiv publikovali ve své práci [Sto01] řadu evolucí navržených hradel. Je mezi nimi i hradlo, jehož schéma je na obrázku 3.35 (převzato z [Sto01]). Hradlo realizuje logickou funkci AND pro napětí na speciálním řídicím signálu  $V_{morph}$  blízkém potenciálu země a logickou funkci OR pro  $V_{morph}$  blízké  $V_{dd}$ . Jde vlastně o třívstupové hradlo realizující funkci zvanou majorita [Val84]. Hradlo sestává z pouhých šesti velmi nekonvenčně zapojených tranzistorů. Za povšimnutí stojí fakt, že tranzistory mají dosti rozdílné rozměry: od širokého a krátkého kanálu (poměr 10:1) až k úzkému a dlouhému kanálu (poměr 1:10). Tak velké rozdíly nebývají u dvojfunkčních polymorfních hradel řízených speciálním signálem obvyklé.

Funkčnost hradla byla ověřena pouze simulací. Spolu se schématem byly publikovány také průběhy napětí na vstupech a výstupech (jsou vidět taktéž na obrázku 3.35). Z nich je vidět, že hradlo se chová poměrně korektně, pokud jde o kvalitu logických úrovní na výstupu, snad jen

log. 0 v režimu AND pro nenulový některý vstup je mírně horší, nežli by byla u konvenčního CMOS hradla. Časová osa je v milisekundách, proto z publikovaných výsledků není možné usuzovat téměř nic o dynamických parametrech hradla. Zajímavé je, že žádný z tranzistorů nespíná  $V_{dd}$ , to je pouze připojeno na substrát. Znamená to však, že log. 1 pro výstup bude vždy čerpána ze vstupů, což bude mít patrně neblahý vliv na kvalitu úrovní při větším zatížení výstupu hradla.

I když jde o dvouvstupové hradlo se dvěma funkcemi, přepínanými dvoustavovým signálem, které by tedy bylo možno chápat jako konvenční třívstupové hradlo (hradlo realizující booleovskou funkci tří proměnných, zde konkrétně funkci majorita), lze ze schématu hradla usuzovat, že parametry hradla patrně nebudou příliš dobré (zejména ve srovnání s parametry ostatních zde popsanych hradel stejné kategorie). Srovnání s konvenčními CMOS hradly toto hradlo nesnese. Je to cena, která se platí za kompaktní rozměry implementace (pouhých šest tranzistorů).



Obrázek 3.35: Hradlo AND/OR, řízené speciálním signálem, navržené týmem A. Stoicy.

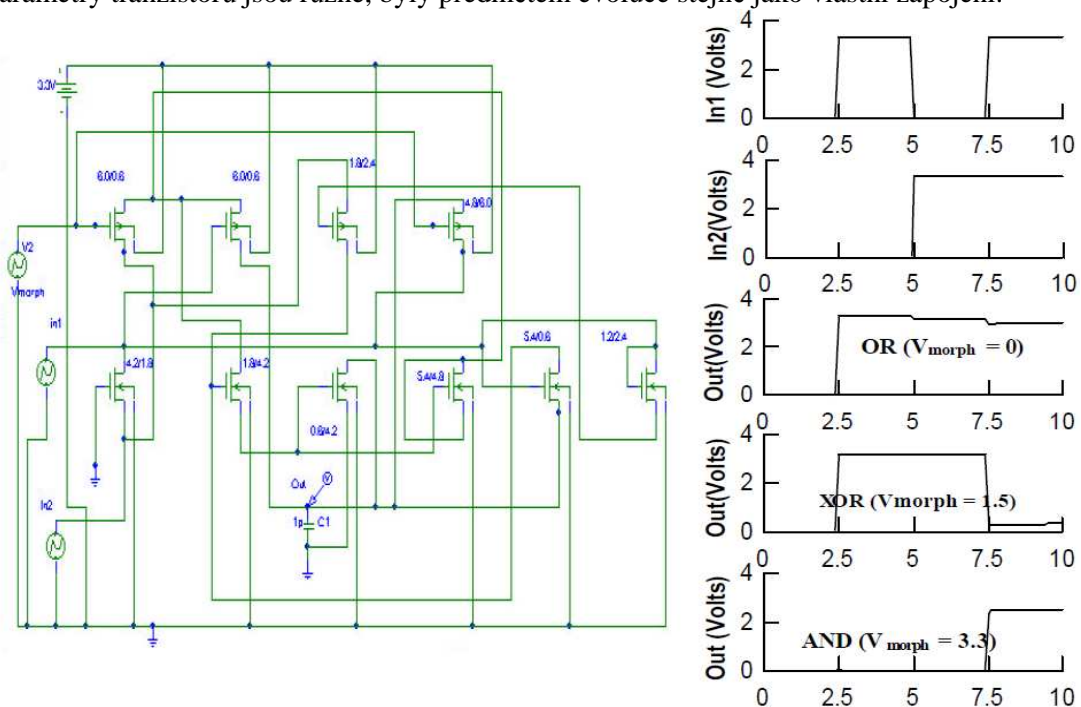
Požadavek			
P1	vstupní odpor	vstupy přímo spínány na výstup	✘
P2	výstupní odpor	závisí na předcházející síti	✘
P3	napájecí napětí	v obvyklém rozsahu (simulováno pro 3,3 V)	✓✓
P4	vstupní úrovně	v obvyklém rozsahu	✓✓
P5	výstupní úrovně	v režimu AND pro 01 a 10 mírně horší, jinak OK	✓
P6	$t_{pd}$	nebylo zkoumáno	?
P7	rozměry	6 tranzistorů	✓✓
P8	spotřeba	nebylo zkoumáno, patrně vyšší	?

Tabulka 3.16: Sledované parametry hradla a splnění požadavků kladených na polymorfní hradla.

### AND/OR/XOR Stoica

V článku [Sto01] bylo publikováno také hradlo, které realizuje tři různé logické funkce podle úrovně řídicího signálu. Protože se stále jedná o jeden jediný řídicí signál, je zřejmé, že zde už nemůže být řeč o třívstupovém konvenčním hradle (jako to bylo možné u hradla realizujícího dvě funkce, přepínané speciálním signálem). Hradlo, jehož schéma je na obrázku 3.36 (obrázek převzat z [Sto01]), realizuje funkci OR pro  $V_{\text{morph}}$  na úrovni blízké zemi, funkci XOR pro  $V_{\text{morph}}$  okolo 1,5 V a funkci AND pro  $V_{\text{morph}}$  blízké  $V_{\text{dd}}$ . Hradlo bylo simulováno, výsledky simulace jsou ve formě průběhů na vstupech a výstupu zobrazeny také na obrázku 3.36. Je vidět, že výstupní úrovně jsou v některých případech horší.

Hradlo je sestaveno z devíti tranzistorů. Ani zde (podobně jako u hradla AND/OR) není na výstup v žádném stavu spínáno přímo  $V_{\text{dd}}$  pro získání log. 1,  $V_{\text{dd}}$  je připojeno pouze na substrát. Parametry tranzistorů jsou různé, byly předmětem evoluce stejně jako vlastní zapojení.



Obrázek 3.36: Hradlo AND/OR/XOR, řízené speciálním signálem, navržené týmem A. Stoicy.

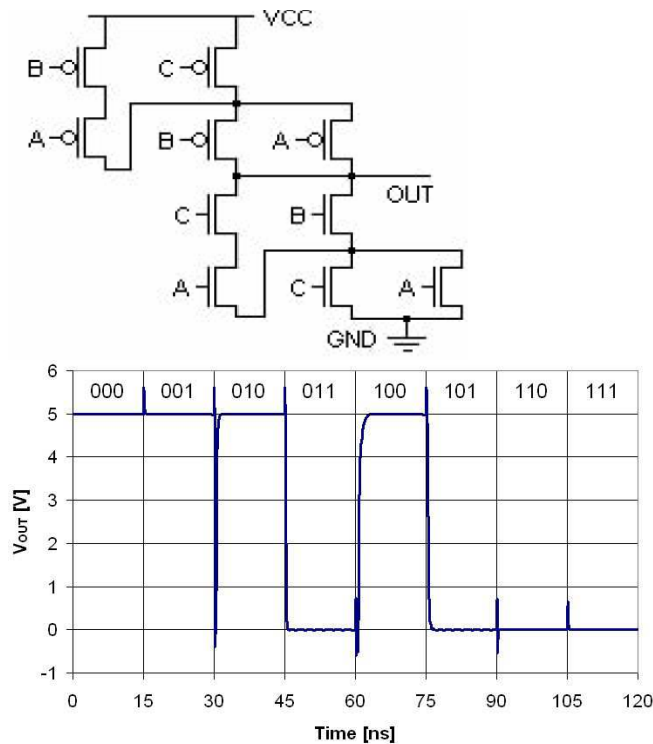
Požadavek			
P1	vstupní odpor	vstupy přímo spínány na výstup	✘
P2	výstupní odpor	závisí na předcházející síti	✘
P3	napájecí napětí	v obvyklém rozsahu (simulováno pro 3,3 V)	✓✓
P4	vstupní úrovně	nebylo možno zjistit	?
P5	výstupní úrovně	v některých případech horší	✘
P6	$t_{\text{pd}}$	nebylo zkoumáno	?
P7	rozměry	9 tranzistorů	✓✓
P8	spotřeba	nebylo zkoumáno, patrně vyšší	?

Tabulka 3.17: Sledované parametry hradla a splnění požadavků kladených na polymorfní hradla.



### NAND/NOR Źaloudek

Při experimentech s evolučním návrhem navrhl polymorfni hradlo řižené speciálním signálem také L. Źaloudek, student doktorského studia z FIT VUT v Brně [Źal08]. Hradlo se velmi podobá návrhu McDermotta a Turnera, má i stejný počet tranzistorů (10). Tranzistory jsou použity obvyklých rozměrů. Lze předpokládat, že bude mít i podobné kvality. Hradlo bylo simulováno, výsledky simulace jsou vidět spolu se schématem na obrázku 3.37 (převzato z [Źal08]).



Obrázek 3.37: Hradlo NAND/NOR, řižené speciálním signálem, navržené L. Źaloudkem.

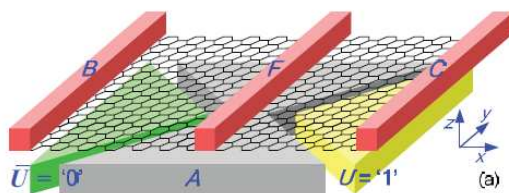
Požadavek			
P1	vstupní odpor	hradla tranzistoru MOS	✓✓
P2	výstupní odpor	sepnutý tranzistor obvyklé velikosti	✓✓
P3	napájecí napětí	v obvyklém rozsahu (5 V)	✓✓
P4	vstupní úrovně	v obvyklém rozsahu	✓✓
P5	výstupní úrovně	v obvyklém rozsahu	✓✓
P6	$t_{pd}$	jednotky ns	✓✓
P7	rozměry	10 tranzistorů	✓
P8	spotřeba	patrně v obvyklém rozsahu	? ✓

Tabulka 3.18: Sledované parametry hradla a splnění požadavků kladených na polymorfni hradla.

### 3.5.4 Polymorfní grafenové hradlo

V roce 2010 publikoval tým S. Tanachutiwata z University of Albany v New Yorku experimenty s prvkem vyrobeným na bázi grafenu (vrstva uhlíku o síle 1 atom), který realizuje několik různých logických funkcí podle toho, jaká napětí jsou přiložena na jednotlivé jeho vstupy [Tan10]. Pro různá napětí se tvoří v grafenu různě polarizované polovodiivé přechody. Rozměry je toto hradlo srovnatelné s rozměry konvenčního CMOS NAND hradla vyrobeného technologií 22  $\mu\text{m}$ , jde tedy o velmi malý prvek. Velkou výhodou grafenových prvků je však rychlost.

Základní funkce prvku je dvouvstupový multiplexor, kdy elektroda A určuje, který z datových vstupů (B nebo C) bude přiveden na výstup F. Elektrody U určují polarizaci přechodů – viz obrázek 3.38 (obrázek převzat z [Tan10]).



Obrázek 3.38: Grafenový tranzistor jako polymorfní hradlo řízené speciálním signálem.

		$U = 1$		$U = 0$	
$V_B$	$V_C$	Function	Output	Function	Output
B	C		$AC + \bar{A}B$		$\bar{A}C + AB$
1	0		$\bar{A}$		A
0	C		AC		$\overline{AC}$
B	1		$A + B$		$\overline{A + B}$

Tabulka 3.19: Přehled funkcí grafenového hradla.

V tabulce 3.19 (převzaté z [Tan10]) jsou uvedeny funkce, které hradlo dokáže realizovat podle toho, jak jsou polarizovány polovodičové přechody či jak jsou využívány jednotlivé vstupy. Na první pohled se zdá, že s poměrně malým a jednoduchým prvkem lze realizovat mnoho logických funkcí a uvedené hradlo proto značně převyšuje možnosti popsaných polymorfních hradel založených na konvenčních křemíkových tranzistorech. Ve skutečnosti však jde o mírně odlišný koncept, kdy dvě základní varianty multiplexoru uvedené v prvním řádku tabulky 3.19 jsou v dalších řádcích tabulky jen rozpracovány do speciálních případů pro konkrétní hodnoty některého ze vstupů. Je třeba poznamenat, že ve smyslu definice polymorfního hradla (definice 3.1) i s přihlédnutím k poznámce 3.2 k této definici popsané hradlo nebude polymorfní s osmi funkcemi (jak by napovídala letmý pohled do tabulky 3.19), ale polymorfní se dvěma funkcemi. Lze ale říci, že dvojice funkcí  $f_1$  a  $f_2$ , vybíraná speciálním signálem U, může být u stejného prvku různá podle toho, jak se prvek zapojí do obvodu.

Pokud by se jako prostředí, ovlivňující funkce hradla (ve smyslu poznámky 3.3 k definici 3.1), uvažoval vektor, složený ze dvou signálů (např. U a B nebo U a C), bylo by možno na uvedené

hradlo nahlízet jako na hradlo realizující čtyři funkce. Takový pohled však již opouští pole dané koncepcí polymorfni elektroniky a míří spíše zpět do oblasti konvenční kombinační logiky.

## 4 Syntéza polymorfních obvodů

### 4.1 Cíle a zásady návrhu polymorfních obvodů

Podstatou návrhu polymorfních číslicových obvodů je hledání grafu  $G$  (viz část 2.1 a v ní uvedená definice 2.1), reprezentujícího zapojení obvodu (propojení jednotlivých komponent jejich vstupů a výstupů), aby obvod dokázal realizovat více funkcí (všechny zamýšlené funkce obvodu tvoří množinu  $\Phi = \{F_1, \dots, F_n\}$ ). Při změně funkce obvodu se tedy musí změnit pouze funkce komponent, zapojení (graf  $G$ ) zůstává stále stejné.

Pokud jde o úroveň popisu, na kterých návrh probíhá (viz obrázek 2.1), u návrhu číslicové polymorfní elektroniky v současnosti přichází v úvahu prakticky pouze úroveň hradel (bližší zdůvodnění viz část 2.2). Samotná polymorfní hradla se pak navrhují na úrovni tranzistorů (viz kapitola 3).

Praktické výsledky experimentů s návrhem polymorfních obvodů ukazují, že nemá smysl hledat obvod složený čistě z polymorfních komponent (hradel). Funkční obvody bývají složeny z konvenčních hradel propojených s polymorfními hradly, dokonce u valné většiny navržených obvodů konvenční hradla počtem převažují. Často také postačuje uplatnit v návrhu polymorfní hradla jednoho typu, zvláště pak, pokud realizují logicky úplné funkce (například hradlo typu NAND/NOR). Použití více typů hradel by možná mohlo vést na optimálnější řešení (např. z hlediska plochy obvodu nebo jeho rychlosti), ovšem za cenu nárůstu složitosti problému návrhu (zvětšení stavového prostoru). Některé metody (např. metody popsané v částech 4.5 a 4.6 této práce) přímo počítají s tím, že bude použito pouze jednoho druhu polymorfního hradla v celém obvodě.

Problém, který zde před návrhářem stojí, je daleko složitější než návrh konvenčních číslicových obvodů. Metody návrhu konvenčních obvodů řeší nalezení grafu obvodu  $G$  pro jedinou funkci  $F$ . Pro tuto funkci je často nalezen vhodný kanonický tvar jejího zápisu, z něhož pak lze jednoduchými pravidly odvodit graf  $G$ . Pak může následovat optimalizace grafu. Tyto metody předpokládají jistý tvar hledaného řešení, který vlastně implikuje graf  $G$ . Pro polymorfní obvody toto nelze dobře použít, protože se hledá jeden graf  $G$  pro  $n$  funkcí  $F_1$  až  $F_n$  a problém nalezení stejného tvaru všech funkcí  $F_1$  až  $F_n$  (jen s různými elementárními funkcemi na stejných místech), implikujícího jediný graf  $G$  s vícefunkčními (polymorfními) hradly, není tak triviální. Je třeba poznamenat, že problém syntézy polymorfního obvodu neřeší jak – jakým prostředím, fyzikální veličinou – a pro jaké její hodnoty bude obvod měnit svoji funkci. Tento problém je dán čistě výběrem polymorfních hradel a jejich vlastností, které se v obvodě uplatní.

Jednodušší obvody je možné navrhnout metodou ad hoc ručně, avšak nelze říci, nakolik se takto navržené obvody blíží optimálnímu řešení. Při návrhu polymorfních obvodů není možné použít ani konvenční optimalizační metody pro návrh číslicových obvodů.

Jak prokázal například Sekanina [Sek06] a před ním naznačili jiní (Koza [Koz92], Miller [Mil00a], Thompson [Tho96]), evoluční techniky dokážou nalézat nekonvenční, zajímavá a užitečná řešení jak v oblasti návrhu, tak i v oblasti optimalizace zapojení číslicových obvodů. Ostatně i sama myšlenka polymorfní elektroniky vznikla jako vedlejší efekt při experimentech s evolučním návrhem obvodů [Sto01]. Prakticky všechny složitější polymorfní obvody byly navrženy technikou Kartézského genetického programování [Mil00a], která je pro návrh kombinačních logických obvodů velmi výhodná.

Problematikou návrhu polymorfních obvodů se detailně zatím zabýval pouze Gajda [Gaj11], ten také jako první ve spolupráci se Sekaninou navrhl „konvenční“ návrhové metody (nezaložené čistě na evolučním či jinak heuristickém přístupu) pro polymorfní obvody.

## 4.2 Ad hoc ruční návrh polymorfních obvodů

Menší a jednodušší polymorfní obvody je někdy možné navrhnout (syntetizovat) bez použití nějakých technik, pouze s využitím zkušeností návrháře. Podařilo se to zejména u některých jednodušších polymorfních hradel (zde ovšem šlo o návrh na úrovni tranzistorů) – některá tato hradla jsou popsána v kapitole 3 – nebo na úrovni hradel jednoduchou kombinační logiku například pro generování dalšího stavu jednoduchých polymorfních čítačů či řadičů – kapitola 7. Zpravidla je to možné u obvodů obsahujících nejvýše jedno či dvě polymorfní hradla. Pak je často možné vytipovat místo, kde bude polymorfní hradlo v obvodě zapojeno, a okolní části obvodu navrhovat pak konvenčními metodami syntézy číslicových obvodů. U obvodů se složitější funkcí a s více možnými polymorfními hradly už není v silách běžného návrháře udržet při návrhu konzistentní představu o fungování obvodu ve více režimech hradel.

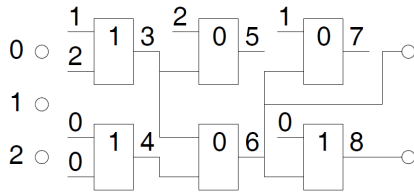
## 4.3 Kartézské genetické programování

Metoda Kartézského genetického programování (Cartesian Genetic Programming, CGP) byla představena J. Millerem a P. Thomsonem v roce 1999 [Mil00a]. Jde o variantu genetického programování, kde jsou kandidátní řešení reprezentována pomocí orientovaných grafů. Je to metoda výhodná pro navrhování kombinačních číslicových obvodů [Sek09a].

Obvod je v tomto případě modelován jako pole elementů (uzlů grafu), což mohou být například hradla, uspořádaných pro přehlednost a snadnou kódovatelnost do matice  $m \times n$ . Počet primárních vstupů a výstupů celého obvodu je pevně určen. Každý z uzlů může realizovat jednu z  $k$  funkcí. Uzly v určitém sloupci je možno propojovat s uzly předchozích sloupců (je pevně dáno, do jaké hloubky  $L$ ), není možné propojení mezi uzly v jednom sloupci a s uzly v následujících sloupcích (žádná zpětná vazba). Všechny zdroje signálů v obvodě (primární vstupy i výstupy hradel) mají svoje označení ve formě celočíselné hodnoty. Právě tímto číslem je v chromozómu konkrétního obvodu kódováno, který signál je připojen na který vstup. I označení funkce uzlu (hradla) je kódováno celočíselnou hodnotou. Chromozóm obvodu má pak tvar posloupnosti celočíselných hodnot. Je tomu tak proto, aby se maximálně zjednodušila práce s chromozómy při generování dalších generací obvodů. Každý uzel potřebuje k jednoznačnému určení své role  $y + 1$  celočíselných hodnot. Přitom  $y$  je počet vstupů uzlu a jedna hodnota navíc je pro identifikaci funkce uzlu. Celý obvod je pak identifikován chromozómem o délce  $(m \cdot n) \cdot (y + 1) + q$  celočíselných hodnot, kde  $q$  je počet primárních výstupů obvodu – posledních  $q$  hodnot chromozómu určuje, které signály generují výstup obvodu.

Na obrázku 4.1 je jednoduché schéma obvodu, který byl navržen metodou Kartézského genetického programování [Ruz06]. V obrázku je jasně vidět maticová struktura  $2 \times 3$  hradla ( $n = 2$ ,  $m = 3$ ), primární vstupy obvodu jsou označeny čísly 0, 1 a 2. Dále každý výstupní signál hradla má svoje číselné označení, které plynule navazuje na očíslování primárních vstupů. To v případě znázorněném na obrázku 4.1 znamená, že výstup prvního hradla má označení 3, výstup druhého hradla 4 atd. Z obrázku 4.1 je dále zřejmé, že pro jednoznačnou identifikaci role hradla v obvodě je třeba celkem tří celočíselných hodnot. To proto, že dvě celočíselné hodnoty určují, jaký signál je přiveden na dvojici vstupů hradla, třetí hodnota pak určuje funkci

realizovanou hradlem. Chromozom obvodu na obrázku 4.1 by pak vypadal takto: 1, 2, 1; 0, 0, 1; 2, 3, 0; 3, 4, 0; 1, 6, 0; 0, 6, 1; 6, 8.



Obrázek 4.1: Obvod navržený pomocí CGP.

Za povšimnutí stojí, že některá hradla se v příkladu obvodu na obrázku 4.1 prakticky nepodílí na jeho činnosti – zde konkrétně hradla s výstupy číslo 5 a 7. Přesto se ale v obvodě vyskytují, jsou do něj zapojena. To je jedna z význačných vlastností Kartézského genetického programování a plyne přímo z jeho definice. Chromozom každého obvodu vždy popisuje úplnou matici uzlů (hradel), i když výstupy některých z nich nejsou dále využity. Délka chromozómu je tedy vždy stejná, ale jednotlivé obvody chromozómem popisované mohou být různě velké. Další optimalizací by bylo u finálního řešení možno nevyužitá hradla vypustit. U členů jednotlivých generací to ale nejen nemá smysl, ale může to být dokonce kontraproduktivní.

Při generování další generace řešení se v Kartézském genetickém programování používá výhradně operátoru mutace. Náhodně vybraný gen (celočíslná hodnota na určitém místě v chromozómu) je změněn na náhodně vygenerovanou novou celočíselnou hodnotu. Parametrem operace je počet takto modifikovaných genů. Je třeba ohlídat, aby nově vygenerovaná hodnota byla platná (je-li to hodnota určující funkci uzlu, aby byla v příslušném rozsahu platných funkcí, je-li to číslo signálu, pak aby bylo v rozsahu signálů existujících v obvodě a navíc aby byla dodržena zásada „žádná zpětná vazba“ a maximální hloubka  $L$ ).

S ohledem na způsob kódování chromozómu je zřejmé, že samotná mutace postačí k tomu, aby byl možný posun ve stavovém prostoru. Změnou jediné hodnoty v chromozómu je totiž možné i to, aby se z původně nevyužitého hradla stalo hradlo, které se účastní operací v obvodě a ovlivňuje tak některý primární výstup obvodu (a naopak). Mutace tím do jisté míry nahrazuje křížení, které v tradičních evolučních technikách právě hraje roli explorativního operátoru [Sek09a]. Dokonce existují studie, které ukazují, že křížení není v některých případech použití CGP vůbec vhodné.

Každá populace sestává z  $(1 + \lambda)$  jedinců, přičemž následující generace se tvoří tak, že ze současné se vybere ten obvod, který vykazuje nejvyšší fitness a tento obvod je základem nové populace spolu se svými  $\lambda$  mutanty. Obvykle se volí  $\lambda = 4$ . Pokud existuje více obvodů se stejnou hodnotou fitness, použije se do nové populace ten, který v předchozí generaci nebyl „rodičem“. Počáteční populace se buď vygeneruje náhodně, nebo s použitím již existujícího řešení. Evoluce končí buď nalezením dostatečně kvalitního řešení (dle požadované hodnoty fitness) nebo vyčerpáním předem stanoveného počtu generací. Fitness se u CGP vypočítá jako hodnota  $B$ , kde  $B$  je počet správných hodnot výstupů pro všechny vstupní kombinace. Pokud jsou všechny hodnoty správně, vypočítá se fitness jako  $B + (mn - z)$ , kde  $z$  je počet v obvodě použitých hradel a  $mn$  je celkový počet hradel, která jsou k dispozici. Člen  $(mn - z)$  tedy ohodnocuje kvalitu nalezeného řešení z pohledu velikosti. Čím větší hodnota fitness funkce, tím blíže je řešení optimu hledaného řešení.

## 4.4 Evoluční návrh polymorfních obvodů

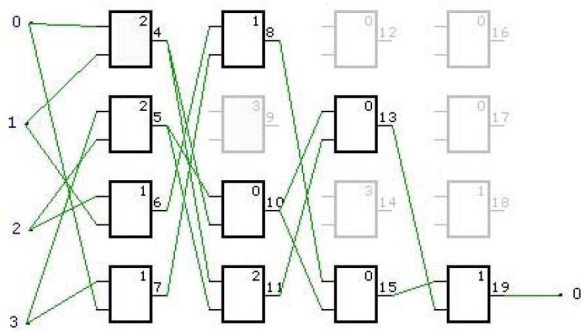
Syntéza polymorfních obvodů založená na evolučních technikách je velmi úspěšná a podařilo se najít řadu zajímavých a dobře uplatnitelných řešení. Jak říká sám autor CGP J. Miller, evoluční algoritmus pracuje na mnohem větším prostoru logických funkcí než konvenční metody syntézy, i když ne všechny reprezentují požadovanou funkci. Postupně však přivádí řešení k požadované pravdivostní tabulce hledané logické funkce. Konvenční metody syntézy sice dokážou pracovat s velkým množstvím vstupních proměnných, ale je obtížné zařídit, aby pracovaly s novými logickými stavebními bloky. Je třeba velké množství analytické práce k nalezení jen o málo lepšího řešení, než je známé. Zdá se proto, že evoluční algoritmy jsou jediným způsobem, jak najít radikálně nová řešení. I pro syntézu polymorfního obvodu to platí více než dokonale. Nejvýznamnější nevýhodou evoluční syntézy je, že nalezená řešení nebývají škálovatelná. Nalezení složitějších obvodů vyžaduje prohledávání velkého stavového prostoru a práci s velkými chromozómy, což z nalezení vhodného řešení činí úlohu velmi časově náročnou a snižuje pravděpodobnost nalezení kvalitního řešení.

Evolučním návrhem polymorfních obvodů se zabýval tým A. Stoicy, který s myšlenkou polymorfní elektroniky přišel [Sto04b], tým W. Lua z univerzity v Hefei, Čína [Luo07] a tým L. Sekaniny z FIT VUT v Brně.

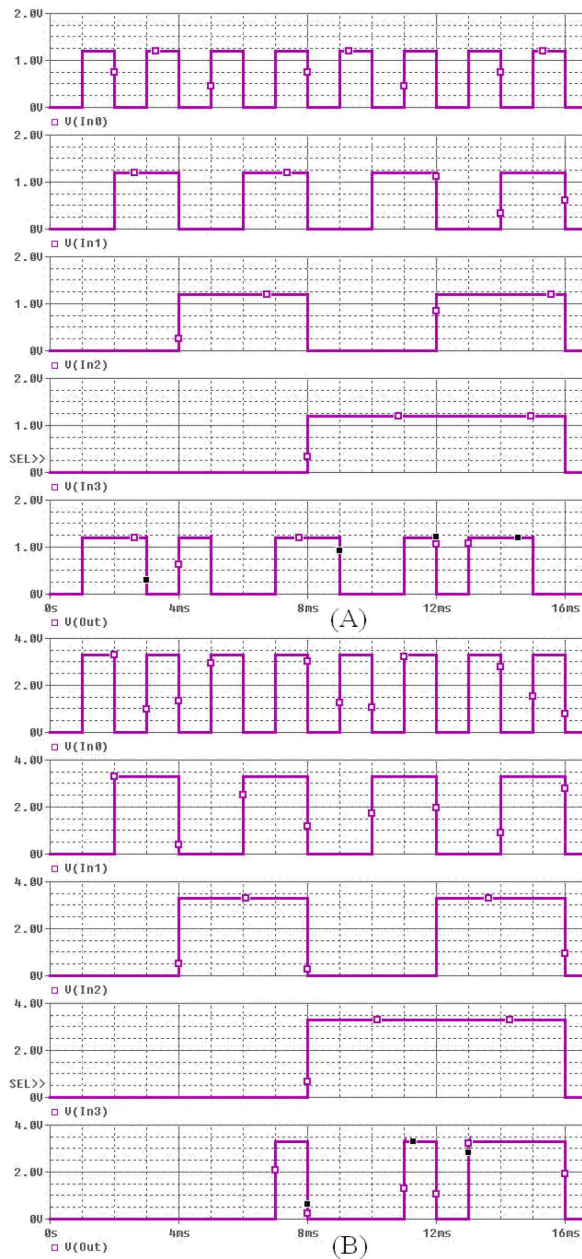
Tým A. Stoicy se zabýval hlavně evolučním návrhem na úrovni tranzistorů. Výsledkem jejich práce jsou některá polymorfní hradla. Jsou popsána v kapitole 3 této práce. Autoři využívají klasického genetického algoritmu s velikostí populace 50, počtem generací 100 – 200 a podílem křížení 30%, mutací 8% [Sto01]. Některé experimenty prováděli přímo v hardware s využitím tranzistorového pole FPTA k evaluaci vyvinutých jedinců (viz např. obrázek 3.28).

Členové týmu z FIT VUT v Brně použili k syntéze polymorfních obvodů téměř výhradně Kartézského genetického programování (CGP). Metoda CGP byla obecně popsána v části 4.3 této práce. Při syntéze polymorfních obvodů je hlavní rozdíl v použité fitness funkci, která musí hodnotit navržená řešení pro všechny režimy činnosti použitých polymorfních hradel. Proto například pro polymorfní hradla realizující dvě různé funkce  $f_1$  a  $f_2$  může fitness funkce vypadat následovně:  $\text{fit} = 1 + (B_1 + B_2)(mn + 1) + z$ , kde  $B_1$  je počet špatných hodnot výstupů obvodu pro všechny vstupní kombinace při polymorfních hradlech v režimu  $f_1$ ,  $B_2$  je počet špatných hodnot výstupů obvodu pro všechny vstupní kombinace při polymorfních hradlech v režimu  $f_2$ ,  $z$  je počet použitých hradel a  $mn$  je počet všech hradel, která jsou k dispozici. Poslední člen se uplatňuje, jen pokud je nalezené řešení plně funkční, tj.  $B_1$  i  $B_2 = 0$ . Je třeba poznamenat, že v tomto případě je fitness funkce konstruována tak, že čím je její hodnota nižší, tím blíže je řešení optimu hledaného řešení. Příklad polymorfního obvodu navrženého Kartézským genetickým programováním je vidět na obrázku 4.2. Jedná se o obvod, který obsahuje tři polymorfní hradla, realizující funkce NAND/NOR (hradla označená „0“) a dále konvenční hradla OR (označená 1) a XOR (označená 2). Tento obvod realizuje funkci sudé parity, pokud jsou polymorfní hradla v režimu NAND a funkci majority, pokud jsou polymorfní hradla v režimu NOR. Obrázek 4.3 ukazuje chování obvodu v simulaci.

Z obrázku 4.2 je zřejmé, že matice elementů pro CGP sestává ze čtyř řádků a čtyř sloupců, obvod má 4 primární vstupy. Každý uzel může plnit jednu ze čtyř funkcí, kromě OR, XOR a NAND/NOR ještě AND, která ovšem není v obvodu využita (i když některým nevyužitým hradlům v matici je přiřazena). Hloubka propojení byla omezena na  $L = 2$  (i když ve výsledném řešení není maximální hloubky nikde využito, všechna hradla jsou zapojena na výstupy hradel předchozího sloupce, hradla prvního sloupce pak přímo na primární vstupy). Velikost populace byla  $(1 + \lambda) = 15$ , mutace modifikovala vždy jeden gen chromozómu a evoluce se zastavila po 100 000 generacích. Ze 100 pokusů bylo nalezeno funkční řešení v 62 případech.



Obrázek 4.2: Polymorfní obvod parita/majorita navržený CGP.

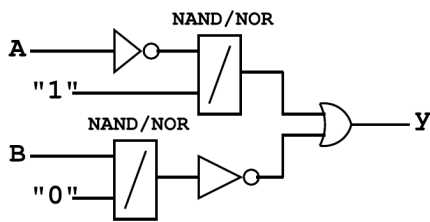


Obrázek 4.3: Výsledky simulace obvodu parita (A)/majorita (B), jehož schéma je na obrázku 4.2.



## 4.5 Polymorfní multiplexování

Technika polymorfního multiplexování, navržená Gajdou a Sekaninou [GaS11] je jakýmsi hybridem mezi konvenčním způsobem návrhu vícefunkčních obvodů a polymorfní elektronikou. Principem je zachování vlastností polymorfní elektroniky jako je vestavěná citlivost na prostředí, ale přitom ponechává možnost využít konvenční prostředky syntézy číslicových obvodů. Polymorfní hradla jsou v obvodě navrženém technikou polymorfního multiplexování využita pouze k přepínání výstupu (výstupů) obvodu podle stavu prostředí. Vlastní funkce pro různé režimy jsou však plněny částmi obvodu sestavenými konvenčními metodami syntézy z konvenčních hradel.



Obrázek 4.4: Možná implementace polymorfního multiplexoru.

Jedna z možných implementací polymorfního multiplexoru s využitím existujících polymorfních hradel NAND/NOR je vidět na obrázku 4.4 [GaS11]. Takový multiplexor je použitelný pro dvojfunkční polymorfní obvod.

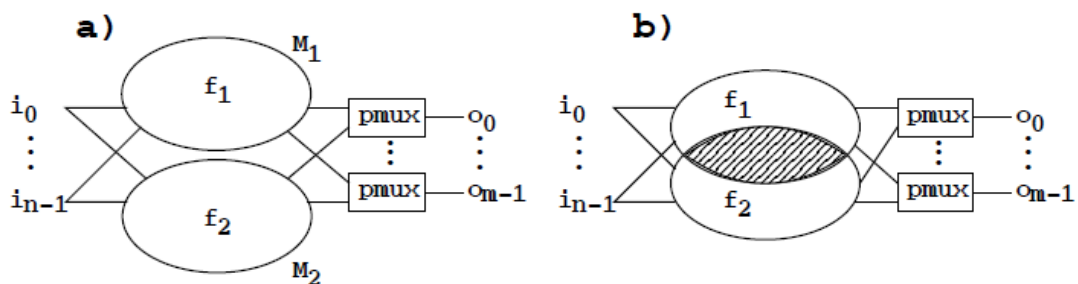
Pokud jsou hradla v režimu NAND, na výstup se přenáší hodnota ze vstupu A:

$$((\text{NOT } A) \text{ NAND } 1) \text{ OR } (\text{NOT } (B \text{ NAND } 0)) = (A \text{ OR } 0) \text{ OR } (B \text{ AND } 0) = A \text{ OR } 0 = A$$

Pokud jsou hradla v režimu NOR, na výstup se přenáší hodnota ze vstupu B:

$$((\text{NOT } A) \text{ NOR } 1) \text{ OR } (\text{NOT } (B \text{ NOR } 0)) = (A \text{ AND } 0) \text{ OR } (B \text{ OR } 0) = 0 \text{ OR } B = B$$

Přitom přepínání mezi vstupy A a B se děje právě tím, čím je měněna funkce polymorfních hradel – prostředím (napájecím napětím, teplotou atd.). Pokud by hradla NAND/NOR byla řízena speciálním signálem (viz část 3.5.3), pak by vlastně šlo o běžný multiplexor. Je třeba poznamenat, že implementace multiplexoru na úrovni hradel není příliš efektivní. Například s využitím NAND/NOR polymorfních hradel řízených  $V_{dd}$ , které vznikly na FIT/FEKT VUT, by polymorfní multiplexor z obrázku 4.4 potřeboval 24 tranzistorů (z toho některé relativně velkých rozměrů). Jistě však je možné na úrovni tranzistorů navrhnout realizaci daleko úspornější.



Obrázek 4.5: Polymorfní obvod, plnící dvě různé funkce  $f_1$  a  $f_2$ , navržený s využitím polymorfních multiplexorů.

Má-li se navrhnout obvod, který bude plnit dvě různé funkce  $f_1$  a  $f_2$  (podle stavu prostředí) technikou polymorfního multiplexování, postupuje se tak, že se nejprve konvenčními metodami syntézy číslicových obvodů navrhnu dva obvody  $M_1$  a  $M_2$ , z nichž jeden plní funkci  $f_1$  a druhý funkci  $f_2$ . Výstupy těchto obvodů se pak na primární výstupy výsledného polymorfního obvodu přepínají polymorfními multiplexory. Zapojení je vidět na obrázku 4.5 a). Je zřejmé, že takto navržený obvod nebude příliš optimální z hlediska velikosti. Konvenční metody syntézy obvodů  $M_1$  a  $M_2$  samozřejmě nabízí možnosti hledat co nejušpornější implementaci funkcí  $f_1$  a  $f_2$ . Velikost obvodu jako celku lze ale minimalizovat zejména tím, že se pokud možno co největší část logiky realizující funkce  $f_1$  a  $f_2$  mezi moduly  $M_1$  a  $M_2$  sdílí, jak naznačuje obrázek 4.5 b). Také je vhodné minimalizovat počet výstupů, pro které je nutné užít polymorfní multiplexování (hammingova vzdálenost mezi  $f_1$  a  $f_2$ ).

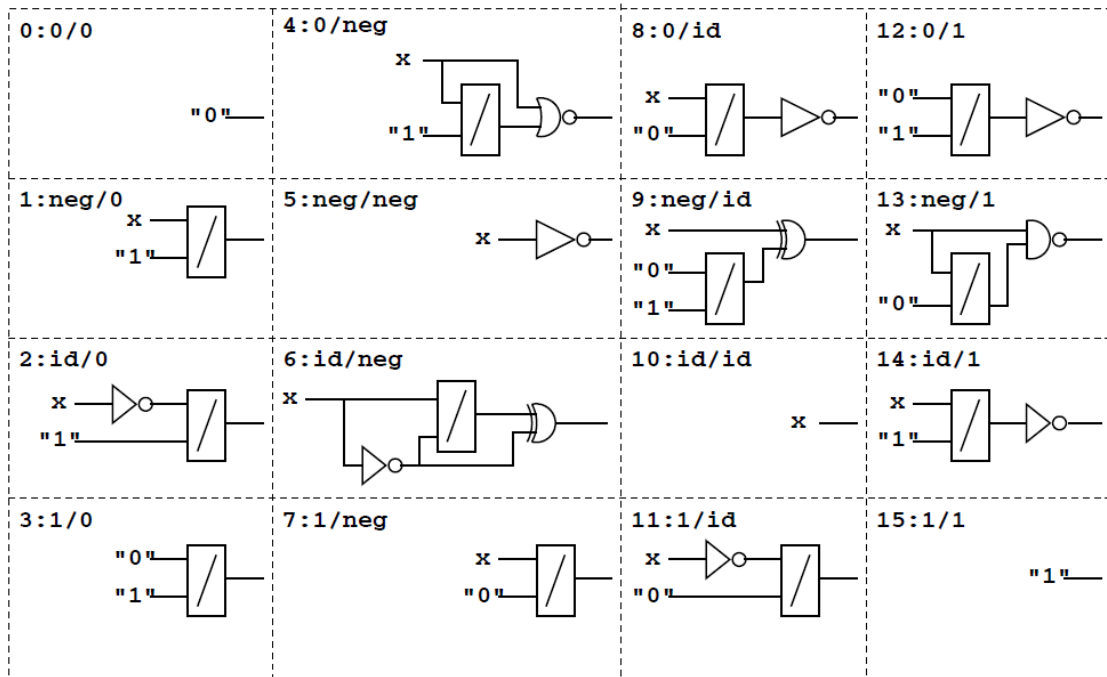
## 4.6 Návrh polymorfních obvodů s využitím BDD

Gajda [Gaj11] navrhl metodu syntézy polymorfních obvodů využívající binárních rozhodovacích diagramů (BDD). Tuto metodu nazval PolyBDD. Využívá tzv. Multi-terminal BDD, což je rozšíření binárních rozhodovacích diagramů o možnost, aby terminální uzly diagramu obsahovaly celočíselné hodnoty. V metodě PolyBDD tyto hodnoty reprezentují možné relace mezi hodnotou vstupní binární proměnné a výstupní hodnotou. Pro jednu proměnnou to může být funkce identity, negace, trvalé 0 nebo trvalé 1. V případě polymorfního obvodu, který má realizovat dvě různé funkce pro dva možné režimy činnosti užitých polymorfních hradel, existuje 16 možných kombinací. Hodnoty v terminálních uzlech MTBDD budou proto celá čísla z intervalu 0 – 15.

Metoda PolyBDD je popsána v [Gaj11] nebo [GaS11] jako posloupnost následujících kroků:

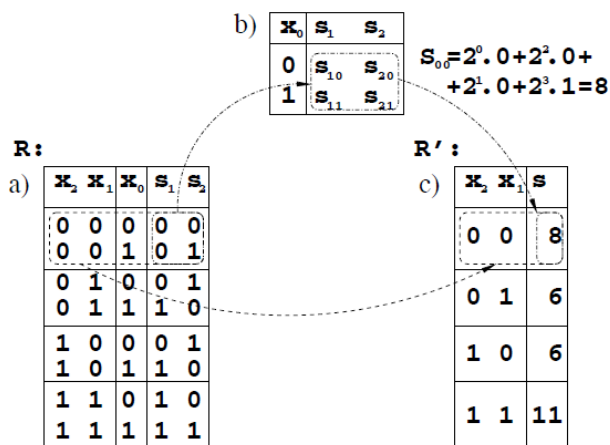
1. Necht' má být navržen polymorfní obvod, realizující funkce  $F_1$  a  $F_2$ , obě jsou funkcemi  $n$  proměnných. Tyto funkce necht' jsou popsány úplnými pravdivostními tabulkami  $R_1$  a  $R_2$ . Každá pravdivostní tabulka má tedy  $2^n$  řádků. Pravdivostní tabulky  $R_1$  a  $R_2$  se sloučí do jedné pravdivostní tabulky  $R$  s  $2^n$  řádky a  $n + 2$  sloupci tak, že prvních  $n$  sloupců tvoří vstupní proměnné  $x_0 - x_{n-1}$ , zbylé dva sloupce pak hodnoty výstupů funkcí  $F_1$  a  $F_2$  pro danou kombinaci vstupů.
2. Vybere se tzv. rozhodovací proměnná  $x_c$  (kde  $c$  je z intervalu 0 –  $n-1$ ) a tabulka se rozdělí na  $2^{n/2}$  segmentů (řádků) takovým způsobem, že v jednom segmentu je vždy ta dvojice řádků, které se liší pouze hodnotou rozhodovací proměnné  $x_c$ . Z každého segmentu se vypočte tzv. signatura podle polynomu  $S = 2^3 \cdot s_{21} + 2^2 \cdot s_{20} + 2^1 \cdot s_{11} + 2^0 \cdot s_{10}$ , kde  $s_{21}$ ,  $s_{20}$ ,  $s_{11}$  a  $s_{10}$  jsou bity ze sloupců příslušejících hodnotám výstupů funkcí  $F_1$  a  $F_2$  ( $s_{10}$  je hodnota výstupu pro funkci  $F_1$  a  $x_c = 0$ ,  $s_{11}$  je hodnota výstupu pro funkci  $F_1$  a  $x_c = 1$ ,  $s_{20}$  je hodnota výstupu pro funkci  $F_2$  a  $x_c = 0$  a konečně  $s_{21}$  je hodnota výstupu pro funkci  $F_2$  a  $x_c = 1$ ). Vznikne tabulka  $R'$  s  $2^{n/2}$  řádky a  $n$  sloupci, kde prvních  $n-1$  sloupců tvoří kombinace hodnot vstupních proměnných kromě  $x_c$  a poslední sloupec tvoří vypočtené signatury  $S$ .
3. Tabulka  $R'$  se konvertuje vhodným algoritmem pro zpracování MTBDD na optimalizovaný rozhodovací diagram. Optimalizace spočívá zejména v redukci identických (redundantních) větví, redukci identických uzlů se stejnými větvemi a přeuspořádání vstupních proměnných.
4. Vzniklý PolyBDD se transformuje na schéma obvodu tak, že neterminální (rozhodovací uzly) se přímo mapují na dvouvstupové multiplexory přepínané příslušnou proměnnou a terminální uzly se implementují jako obvodové struktury realizující příslušnou kombinaci funkcí proměnné  $x_c$  s využitím polymorfního hradla. Pro každou kombinaci funkcí je k dispozici řešení s polymorfním hradlem podle signatury. Například pro využití

polymorfních hradel NAND/NOR je přehled řešení dle kombinací funkcí a z toho odvozené signatury vidět na obrázku 4.6.



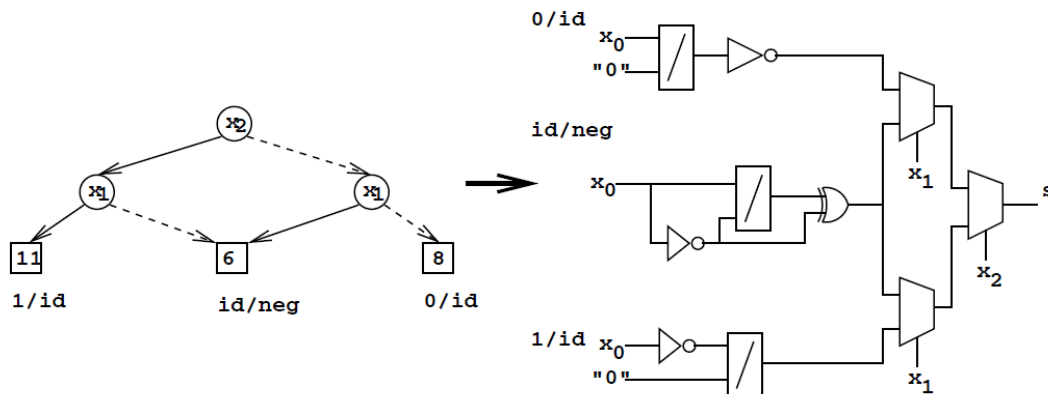
Obrázek 4.6: Přehled řešení všech dvojic kombinací funkcí rozhodovací proměnné s hradlem NAND/NOR a příslušných signatur.

Použití metody PolyBDD bude demonstrováno na příkladu obvodu, který má realizovat funkce  $F_1$  „majorita tří proměnných“ a  $F_2$  „sudá parita pro tři proměnné“. Na obrázku 4.7 a) je pravdivostní tabulka R sdružující hodnoty výstupů pro obě tyto funkce. Jako rozhodovací proměnná byla vybrána proměnná  $x_0$ . V tabulce R na obrázku 4.7 a) je vyznačeno i rozdělení na čtyři segmenty podle proměnné  $x_0$  (v každém segmentu jsou hodnoty proměnných  $x_1$  a  $x_2$  stejné). Z hodnot výstupu obvodu ( $s_1$  pro  $F_1$  a  $s_2$  pro  $F_2$ ) v každém segmentu se vypočte hodnota signatury S segmentu dle matice zobrazené na obrázku 4.7 b) realizující konverzní polynom. Hodnoty signatur pro všechny segmenty pak obsahuje tabulka R', která je pro daný příklad vidět na obrázku 4.7 c).



Obrázek 4.7: Sdružená tabulka pro funkce majorita/parita tří proměnných a) a konverze této tabulky na tabulku signatur pro konstrukci MTBDD c) pomocí konverzní matice b).

Když je k dispozici tabulka  $R'$ , je možné zkonstruovat rozhodovací diagram. Redukovaný diagram pro funkce  $F_1$  a  $F_2$  z příkladu uvedeného na obrázku 4.7 je vidět na obrázku 4.8 vlevo. Je vidět, že došlo k redukci (sloučení) terminálních uzlů se signaturou 6, které se v tabulce  $R'$  dvakrát opakovaly. Z takového diagramu je pak sestrojeno schéma obvodu (obrázek 4.8 vpravo). Jak napovídá algoritmus, neterminální (rozhodovací) uzly se implementují jako dvouvstupové multiplexory přepínání příslušnou proměnnou. Rozhodovací proměnná se pak objevuje na vstupu podsítí realizujících kombinace jejích funkcí nalezené výpočtem polynomu signatury. Je zvoleno řešení využívající polymorfní hradla NAND/NOR, řešení terminálních uzlů pro příslušné signatury ukazuje obrázek 4.6.



Obrázek 4.8: Rozhodovací diagram pro dvojici funkcí majorita/parita pro tři proměnné vytvořený metodou PolyBDD a výsledný obvod obsahující polymorfní hradla NAND/NOR.

## 4.7 Optimalizace polymorfních obvodů

Obvody syntetizované pomocí metod popsaných v částech 4.5 a 4.6 jsou sice plně funkční a splňují většinu požadavků kladených na polymorfní obvody (viz kapitola 2). Jsou však zpravidla zbytečně velké. Takto syntetizované obvody se vyznačují tím, že polymorfní hradla jsou v nich zastoupena velmi řídko, slouží jen jako senzorové přepínače vstupů (v případě obvodů syntetizovaných metodou PolyBDD) nebo výstupů (v případě obvodů syntetizovaných metodou polymorfního multiplexování). Nevyužívá se tak plně potenciálu polymorfních hradel, jejichž použití vede ke kompaktním a úsporným řešením. Gajda a Sekanina [GaS11] vyslovují hypotézu, že zvětšení podílu polymorfních hradel v obvodě a jejich zapojení hlouběji ve struktuře obvodu vede k úspornější implementaci. Tuto hypotézu dokazují optimalizací obvodů syntetizovaných popsány metodami pomocí evolučních technik.

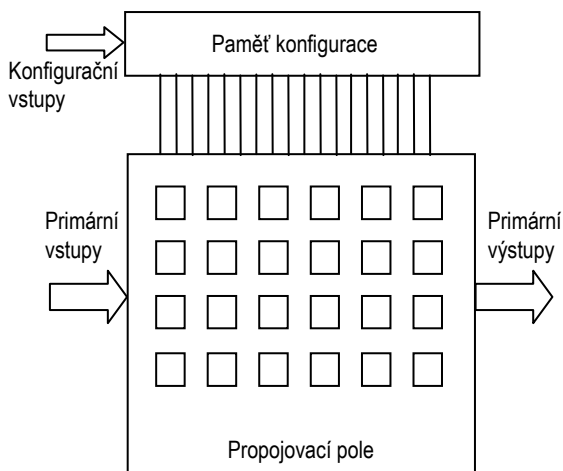
Pro účely optimalizace polymorfních obvodů syntetizovaných popsány neevolučními metodami používají Gajda a Sekanina opět Kartézského genetického programování. Protože Kartézské genetické programování pracuje vždy s dvouvstupovými elementy, je třeba vícevstupové elementy (například multiplexory používané v přepínací síti obvodů syntetizovaných metodou PolyBDD) implementovat pomocí více dvouvstupových hradel.

Jako výchozí obvod pro CGP je použito neoptimalizované funkční řešení syntetizované některou z metod. Každé kandidátní řešení je pak ohodnoceno pomocí fitness funkce  $1 + (B_1 + B_2)$ , kde  $B_1$  je počet špatných hodnot výstupů obvodu pro všechny vstupní kombinace při polymorfních hradlech v režimu  $f_1$ ,  $B_2$  je počet špatných hodnot výstupů obvodu pro všechny vstupní kombinace při polymorfních hradlech v režimu  $f_2$ . Za povšimnutí stojí, že do hodnoty fitness funkce se nijak nepromítá počet hradel, tedy velikost obvodu. Přitom právě ta je předmětem optimalizace. Přesto tak nastavená evoluce vede k optimalizaci řešení [GaS11].

## 5 Experimentální platforma pro polymorfní obvody

### 5.1 Konfigurovatelné obvody

Již mnoho let se v elektronice používají integrované obvody, které jsou elektricky konfigurovatelné. Zejména pro rychlé prototypování či kusovou výrobu elektronických zařízení jsou takové obvody neocenitelné. Staly se téměř plnohodnotnou alternativou k zakázkově vyrobenému integrovanému obvodu (Application Specific Integrated Circuit, ASIC). Konfigurovatelné obvody mají nejčastěji podobu pole (matice) elementů, z nichž má být sestaven obvod. Dále obsahují konfigurovatelnou propojovací síť a v neposlední řadě paměť konfigurace. Obrázek 5.1 ukazuje zjednodušeně typickou strukturu konfigurovatelného obvodu.



Obrázek 5.1: Typická struktura konfigurovatelného obvodu.

Vznik konfigurovatelných obvodů byl zprvu zřejmě motivován myšlenkou cenově dostupné alternativy k ASIC obvodům pro aplikace vyráběné v malém množství, kde by se vývoj a výroba obvodu ASIC nevyplatila. Později, zejména s rozvojem technologií polovodičových pamětí (SRAM, Flash), kdy odpadla nutnost výroby propojovací masky či programování obvodu nákladným speciálním zařízením, se rozšířilo použití konfigurovatelných obvodů i do oblasti prototypování a v současnosti, kdy jsou k dispozici vyspělé technologie pro implementaci konfiguračních pamětí a technologie umožňující na čipu tvořit velmi drobné struktury pro výrobu rozsáhlých a přitom efektivních a rychlých propojovacích sítí a velkého množství elementů, využívá se konfigurovatelných obvodů i k rekonfigurovatelnému počítání [Har01], [Bob07] či k evolučnímu návrhu obvodů přímo v hardware [Tho96], [Tho99], [Sek09a].

Elementy, z nichž lze pomocí konfigurace sestavovat uvnitř konfigurovatelného obvodu různá zapojení, mohou být buď samotné tranzistory (např. FPTA [Lan05], [Zeb00], [Sto00]), hradla, či složitější struktury (více vstupové kombinační bloky, bloky na úrovni RT).

Průkopníkem konfigurovatelných polí elementů na křemíkovém čipu byla firma Fairchild se svým polem komponentů DTL logiky Fairchild Micromatrix 4500. Pole tvořilo 32 hradel [Mar67]. O rok později následovalo pole třiceti čtyřhradlových buněk v technologii TTL od firmy Sylvania. Další z prvních konfigurovatelných obvodů uvedla na trh firma Motorola pod označením XC157 v roce 1969. Bylo to pole 12ti hradel, který mohla být propojena pomocí propojovací vrstvy navzájem a k třicítce vývodů pouzdra, které tvořily primární vstupy a

výstupy. V tomto obvodu se ještě nenachází oddělená konfigurační paměť (jako je vidět třeba na obrázku 5.1), lze ale říci, že tato paměť je reprezentována propojovací maskou, jde vlastně o paměť typu ROM [Mot69]. Brzy následovaly obvody s 25ti a 80ti hradly na jednom čipu.

O obvodech, které byly programovatelné maskou (konfigurační paměť typu ROM) lze hovořit jako o tzv. částečně zakázkových obvodech (semi-custom devices), protože pro získání obvodu s vlastní konfigurací bylo třeba spolupracovat s výrobcem. V 70. a počátkem 80. let držela 40% světového trhu s částečně zakázkovými obvody anglická firma Ferranti se svými ULA (Uncommitted Logic Array) z roku 1972. Šlo o pole konfigurovatelných elementů (tranzistory, které se v základním propojení chovaly jako dvou vstupové NOR hradlo), ty bylo možno vzájemně propojovat s využitím jedné zákazníkem specifikované metalické vrstvy. Firma Ferranti měla velmi propracovaný a efektivní systém propojování, proto i pro poměrně složité obvody bylo možno vystačit s jednou propojovací vrstvou. V 70. letech nabízela obvody s 200 – 1000 hradly pro rychlost až 20MHz, počátkem 80. let už byly k dispozici rodina 50ti obvodů s až 10 000 hradly a rychlostí 80 MHz. Zákazníkům firma dodávala i celé návrhové systémy sestávající ze software typu CAD na minipočítačích PDP. Návrhový systém zahrnoval knihovny prvků, podporu návrhu obvodů včetně verifikace, simulace, generování testů. Návrhový systém komunikoval s centrem firmy po lince s protokolem DECNET. Zákazník tak byl velmi blízko fyzické realizaci, s plnou verzí návrhového systému mohl provést celý fyzický návrh sám a do firmy jen odeslat podklady k finální verifikaci a výrobě, i když samozřejmě firma nabízela pro příležitostné zákazníky i služby komplexního návrhu [Smi10].

Velkou oblibu si získala programovatelná hradlová pole PAL firmy MMI, uvedená na trh v roce 1978. Tyto obvody již bylo možno programovat elektricky, firma dodávala i software pro syntézu konfiguračních dat z popisu funkce obvodu prostřednictvím booleovských rovnic [Líš93]. Konkurenční firma Lattice Semiconductors v roce 1985 představila obvody řady GAL (Generic Array Logic), které již obsahovaly konfigurační paměť typu EEPROM a bylo je možno mazat, měnit konfiguraci a programovat přímo v aplikaci. To otevřelo cestu ke konfigurovatelným obvodům typu CPLD (Complex Programmable Logic Devices), které obsahují více obvodů podobných řadě GAL na jednom čipu [Líš93].

V současnosti patří mezi nejužívanější konfigurovatelné obvody FPGA (Field-Programmable Gate Array). Konfigurovatelné elementy v matici jsou ekvivalentní logické síti obsahující mnoho hradel, mohou realizovat složitější kombinační funkci a často obsahují i sekvenční prvky (klopné obvody, registry). Moderní FPGA umožňují dále propojovat tyto bloky s komponenty typu paměť RAM atd. Speciální buňky jsou užity k připojení k primárním vstupům a výstupům. I jejich funkce může být složitější než jen pouhé propojení dvou uzlů. Konfigurační paměť je u obvodů FPGA tvořena pamětí typu SRAM [Sah10].

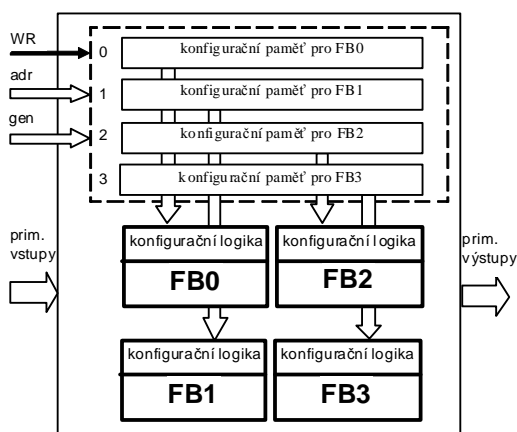
Dnes tvoří konfigurovatelné obvody typu FPGA velkou část trhu s integrovanými obvody. Existuje několik velkých firem, které je vyrábí (Xilinx, Altera, Atmel, Lattice, Actel). Pro jejich výrobu se využívají nejmodernější technologie výroby integrovaných obvodů a tak je možné vytvářet i v laboratorních podmínkách ty nejrychlejší a nejrozsáhlejší aplikace.

Téměř všichni výrobci konfigurovatelných obvodů až do současnosti navrhovali svoje produkty jako alternativu k čipům ASIC pro menší série a prototypové aplikace v elektronických zařízeních. Předpokládá se tedy, že zapojení obvodu bude navrženo návrhářem, odladěno a pak nasazeno v aplikaci. Dnes ale existují i takové způsoby využití konfigurovatelných obvodů, které předpokládají změnu zapojení obvodu v průběhu jeho činnosti. Příkladem takové aplikační oblasti může být rekonfigurovatelné počítání, evoluční návrh obvodů přímo v hardware nebo sebemodifikující se obvody (např. za účelem zvýšení spolehlivosti nebo odolnosti proti poruchám). Pro tuto aplikační oblast jsou výhodné obvody se způsobem konfigurace, jako mají například FPGA.

Protože aplikace, u nichž je třeba rekonfigurace za běhu programu, jsou většinou zatím pouze ve vývoji a ve stádiu pokusného nasazení a navíc jde o obvody s velmi různorodou strukturou, není znám konfigurovatelný obvod, který by pro tuto třídu aplikací byl komerčně vyráběn. V minulosti měla tuto úlohu plnit řada FPGA obvodů Xilinx 6200, avšak pro tyto účely se neprosadila [Sek09a].

Sekanina spolu s autorem této práce v roce 2000 navrhli konfigurovatelný obvod pro účely evolučního návrhu číslicových obvodů a biologií inspirovaného adaptivního hardware. Ve své práci [Sek00] formulovali požadavky na takový obvod následovně:

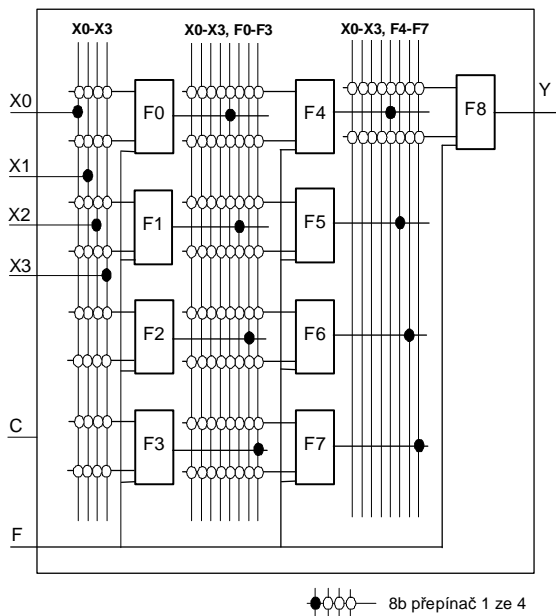
- paměť konfigurace typu RAM, aby bylo možno neomezeně měnit konfiguraci při běhu zařízení, například v průběhu evoluce bylo možno nahrát každé kandidátní řešení do obvodu a evaluovat jej,
- konfigurovatelné elementy mají vhodnou formu a granularitu, jsou „ušity na míru“ aplikaci, která má být v obvodu implementována, čímž je zajištěno, že konfigurační data popisující element jsou krátká a výstižná (např. pokud se evolucí vyvíjí zapojení na úrovni RT, není vhodné, aby každý element byl popsán dlouhou sekvencí bitů popisující jeho vnitřní strukturu, s takovými daty evoluční algoritmus pracuje zbytečně obtížně a zdoluhavě),
- formát konfiguračních dat je známý a jednoduchý, protože konfigurační data je při evoluci třeba tvořit on-line s co nejjednodušším zařízením (v tomto bodě selhává většina komerčních FPGA, protože formát jejich konfiguračních dat nebývá dobře dokumentován a je tedy možné používat maximálně několik konfigurací dopředu připravených syntézou v příslušném vývojovém systému),
- náhodná kombinace v konfiguračních datech nesmí způsobit havárii aplikace nebo dokonce poškodit čip, protože evoluce může generovat prakticky libovolné kombinace konfiguračních dat,
- konfigurační proces je dostatečně rychlý, ideálně paralelní sběrnici, protože je třeba během evoluce vyzkoušet velké množství kandidátních řešení (většina současných FPGA má však sériový způsob konfigurace dlouhým konfiguračním řetězcem, protože se počítá s tím, že konfigurace se nahrává pouze jednou, při startu aplikace – tzv. bootování FPGA),
- je podporována částečná rekonfigurace, v opačném případě by bylo nutno zdoluhavě konfigurovat celý čip, přitom často je třeba měnit jen malou část (jen část čipu může být vyhrazena pro evaluaci kandidátních řešení, v případě adaptivního hardware také často stačí adaptovat jen určitý úsek obvodu).



Obrázek 5.2: Rekonfigurovatelný čip pro evoluční návrh číslicových obvodů.

Rekonfigurovatelný obvod, publikovaný v [Sek00], byl implementován v běžném komerčním FPGA. Jeho blokové schéma je vidět na obrázku 5.2. Navržený obvod má paralelní konfiguraci

s náhodným přístupem do konfigurační paměti, zápis konfigurační informace tedy bude dostatečně rychlý, navíc je možné změnit libovolnou část konfigurační informace, aniž by byl ovlivněn zbytek obsahu konfigurační paměti. Ve schématu na obrázku 5.2 je vidět, že jsou navrženy čtyři funkční bloky (FB0 – FB3, čtyři identické matice elementů s konfigurovatelným propojením) a každý z nich má svůj samostatný úsek konfigurační paměti. Toto rozdělení bylo zvoleno s ohledem na využití k evolučnímu návrhu číslicových obvodů, případně k evoluci řízené adaptaci. Takto rozdělený obvod podporuje evaluaci čtyř kandidátních řešení z jedné populace současně.



Obrázek 5.3: Struktura funkčního bloku rekonfigurovatelného obvodu z obrázku 5.2.

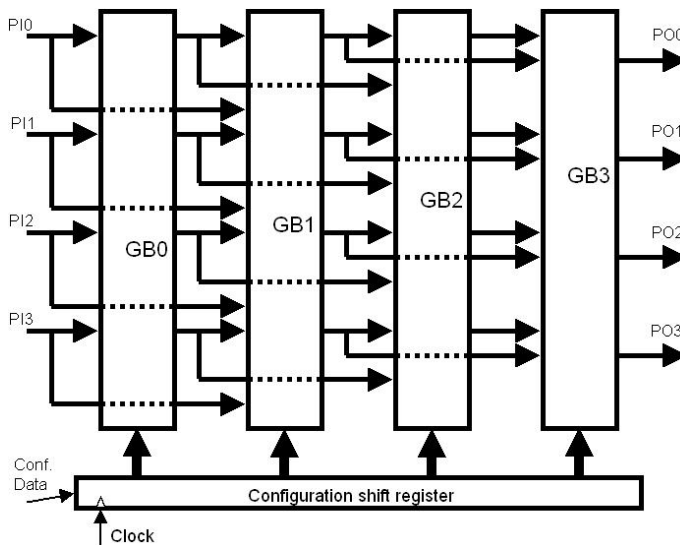
Vnitřní strukturu funkčního bloku FB<sub>x</sub> z obrázku 5.2 ukazuje obrázek 5.3. Čtveřice primárních vstupů X0 – X3 je rozvedena ke všem elementům F0 – F8. Elementy prvního sloupce F0 – F3 mohou na svoje vstupy mít připojen kterýkoliv z primárních vstupů, elementy dalších sloupců pak buď kterýkoliv z primárních vstupů nebo kterýkoliv z výstupů elementů předchozího sloupce. Propojení přepínačů je řízeno obsahem konfigurační paměti. Dále je konfigurovatelná funkce každého elementu (přes signál F).

## 5.2 REPOMO

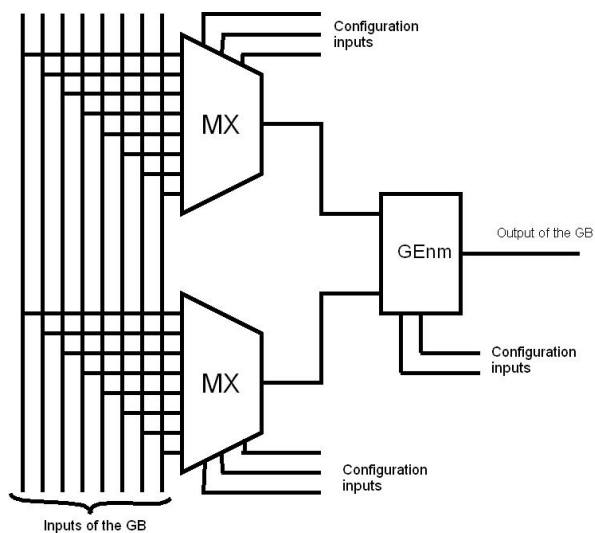
Pro účely experimentování s polymorfními obvody a také pro podporu evolučního návrhu polymorfních obvodů byl L. Sekaninou a autorem této práce navržen s využitím zkušeností z předchozích návrhů speciální konfigurovatelný obvod. Avšak na rozdíl od konfigurovatelného obvodu popsaného v [Sek00], tento polymorfní rekonfigurovatelný obvod nemohl být realizován v běžně dostupném FPGA, protože jeho elementy musí být schopny pracovat jako polymorfní hradla, což není s běžnou logikou realizovatelné (šlo by maximálně o přibližnou simulaci). Jak je zřejmé z kapitoly 3, polymorfní hradla jsou speciální struktury navržené na úrovni tranzistorů, proto pro realizaci přichází v úvahu jedině obvod ASIC. Obvod byl navržen a realizován ve spolupráci s kolegy z Ústavu mikroelektroniky Fakulty elektrotechniky a komunikačních technologií a vyroben v několika desítkách exemplářů technologií AMIS 0,7 μm za podpory projektu GA102/06/0599 „Metody návrhu polymorfních číslicových obvodů“. Obvod je zapouzdřen v keramickém pouzdru DIL s 28 vývody.



První návrhy obvodu byly publikovány v práci [Ruz06], obvod byl nazván „Rekonfigurovatelný polymorfní modul“ – REPOMO (REconfigurable POLymorphic MOdule) s ohledem na skutečnost, že návrh předpokládá možnost propojení více obvodů (modulů) pro realizaci rozsáhlejších návrhů polymorfních obvodů. Při návrhu struktury obvodu autoři dále vycházeli z předpokladu, že modul bude sloužit pro evaluaci kandidátních řešení obvodů navržených Kartézským genetickým programováním (viz část 4.3 této práce), proto je struktura konfigurovatelných elementů přísně maticová a umožňuje hloubku propojení  $L = 2$ . Hloubka propojení byla omezena s ohledem na velikost konfiguračních dat (konfigurační data pro jeden element zabírají nejvýše 8 bitů – viz požadavky formulované v části 5.1) a s ohledem na zkušenosti s návrhem obvodů pomocí CGP, které ukazují, že taková hloubka propojení je postačující pro nalezení kvalitních řešení.



Obrázek 5.4: Základní struktura REPOMO.

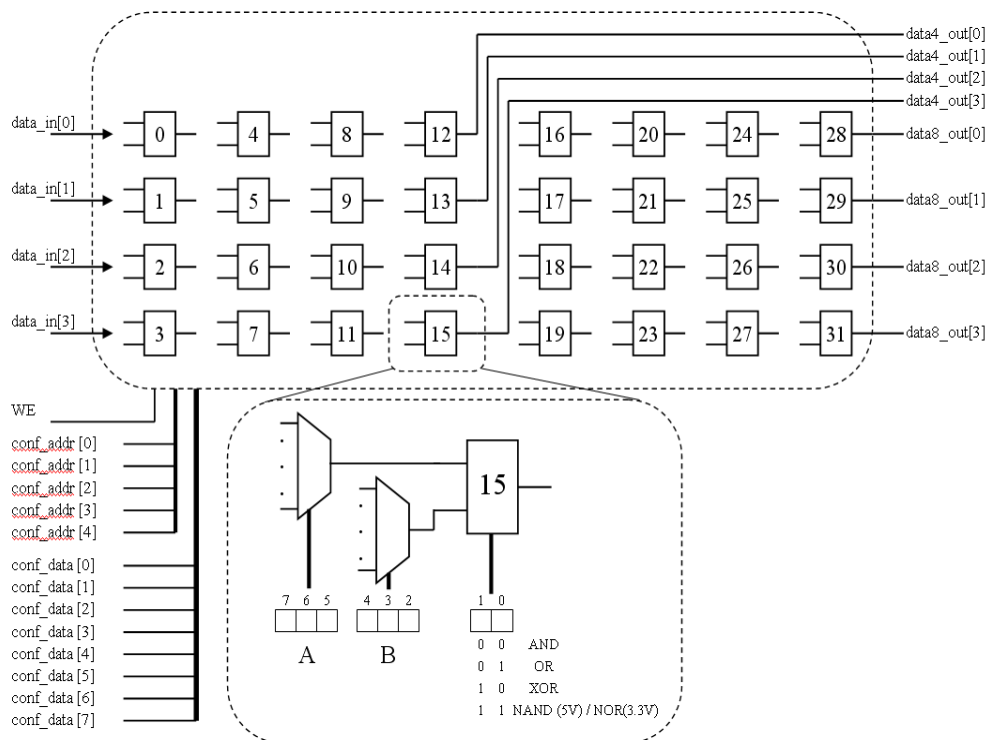


Obrázek 5.5: Struktura  $\frac{1}{4}$  sloupce hradel GB obvodu REPOMO.

Základní strukturu obvodu REPOMO ukazuje schéma na obrázku 5.4. Je tam vyznačeno i propojení jednotlivých sloupců s hloubkou propojení  $L = 2$ . Obvod má čtyři primární vstupy a

také čtyři primární výstupy. Na obrázku je znázorněna i konfigurační paměť ve formě posuvného registru se sériovým plněním. Fyzická realizace REPOMO na čipu má ale z důvodů uvedených v předchozí části této kapitoly paralelní ukládání konfigurace do konfigurační paměti typu RAM.

V každém sloupci, na obrázku 5.4 označeném GBx, mohou pracovat až 4 dvouvstupová hradla. Jejich zapojení do obvodu je vidět na obrázku 5.5. Na obrázku je pouze jedno hradlo, každý GB obsahuje čtyři takové elementy. Každé hradlo v GBx může být zapojeno na jeden ze čtyř výstupů hradel GBx-1 nebo na jeden ze čtyř výstupů GBx-2. Pro každý vstup každého hradla se tedy vybírá z osmi zdrojů signálu pomocí osmivstupového multiplexoru (viz obrázek 5.5) třemi bity konfiguračních dat.



Obrázek 5.6 Celková struktura REPOMO s formátem konfiguračních dat.

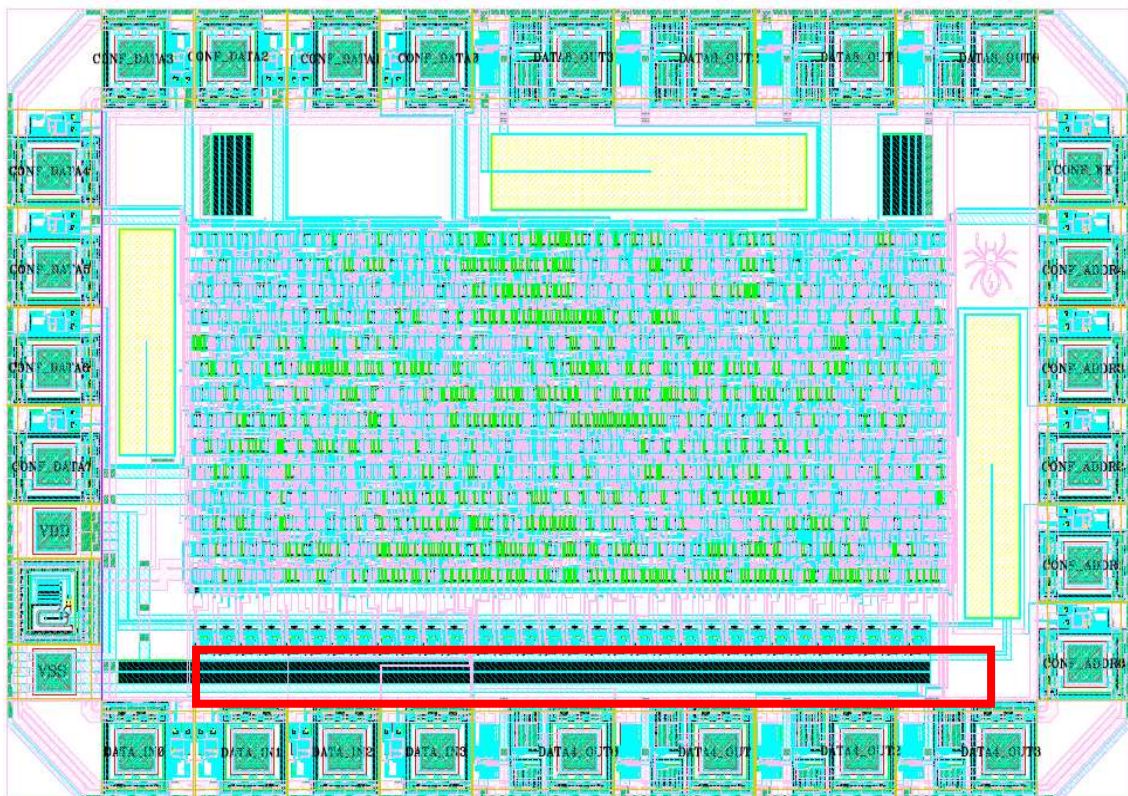
Na čipu REPOMO je celkem 32 elementů s hradly uspořádaných do matice 8 sloupců x 4 řádky. Na jeden čip se tedy vešly dvě struktury z obrázku 5.4. Na obrázku 5.6 je vidět přehledně uspořádání modulu REPOMO. V souladu s původním návrhem jsou primární výstupy vyvedené z výstupů hradel sloupce GB3, další pak z výstupů sloupce GB7. Obrázek také přehledně ukazuje způsob konfigurace (ve výřezu). Každý element je konfigurovatelný jedním bajtem. Bity 6, 5, 3 a 2 udávají řádek, z něž výstup některého předchozího hradla je připojen na vstup hradla, bity 7 a 4 pak určují, zda jde o výstup hradla z bloku GBx-1 (bit obsahuje hodnotu 0) nebo výstup z hradla bloku GBx-2. Pro elementy prvního sloupce GB0 (na obrázku 5.6 jsou to elementy číslo 0, 1, 2 a 3) nemají konfigurační bity 7 a 4 význam, bity 6, 5, 3 a pak určují číslo primárního vstupu, který je na vstup daného hradla připojen. Pro elementy druhého sloupce GB1 (na obrázku 5.6 jsou to elementy číslo 4, 5, 6 a 7) pak hodnota 1 konfiguračních bitů 7 a 4 znamená, že se připojují primární vstupy. Konečně bity 1 a 0 konfiguračního bajtu udávají, jakou funkci daný element realizuje. Přehledně to ukazuje tabulka 5.1. Každý element může realizovat funkci konvenčního hradla AND nebo OR, případně XOR nebo polymorfního hradla NAND/NOR. Při vhodné konfiguraci (například oba vstupy připojené na stejný signál – bity 7, 6, 5 konfiguračního bajtu elementu stejné jako bity 4, 3, 2 a nastavení funkce například na AND

nebo OR) může element plnit i funkci prostého spoje (i když bude vykazovat určité zpoždění) a je tak možné za cenu obětování jednoho elementu zvýšit hloubku propojení o jeden sloupec.

Funkce	bit 1	bit 0
AND	0	0
OR	0	1
XOR	1	0
polymorfní NAND/NOR	1	1

Tabulka 5.1 Konfigurace funkce elementu v REPOMO

Každému elementu odpovídá právě jedno osmibitové konfigurační slovo. Konfigurační slova lze do konfigurační paměti obvodu (typu SRAM) vkládat prostřednictvím osmi vstupů `conf_data[7..0]`, konfigurace je paralelní. Adresa konfigurační paměti se vybírá vstupy `conf_addr[4..0]`, konfigurační paměť má tedy náhodný přístup. Je možno rekonfigurovat libovolný element, aniž by se ovlivnila konfigurace ostatních. Adresy odpovídají číslům elementů uvedeným na obrázku 5.6 – element číslo 0 má konfigurační slovo uloženo v konfigurační paměti na adrese 0, element číslo 1 na adrese číslo 1 atd.



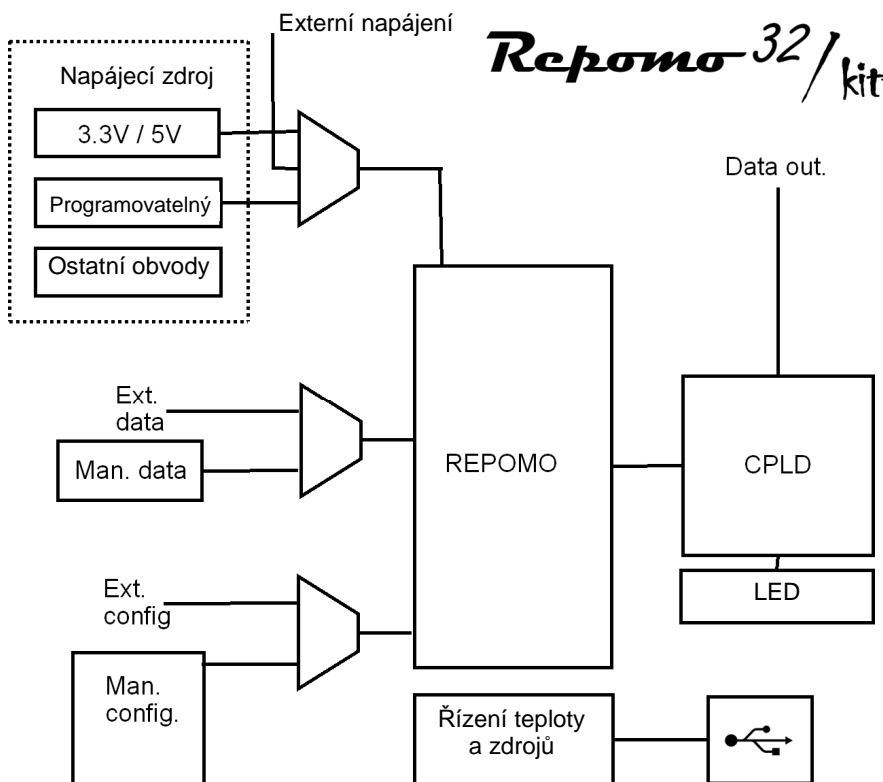
Obrázek 5.7: Rozvržení čipu REPOMO (v červeném rámečku sada 32 polymorfních hradel).

Na obrázku 5.7 je fyzické rozvržení čipu REPOMO. Čip byl vyroben technologií AMIS s rozměrem elementu  $0,7 \mu\text{m}$ , celkové rozměry čipu jsou  $2,9 \times 1,97 \text{ mm}$ . Byla využita polymorfní hradla NAND/NOR, navržena ve spolupráci FIT a FEKT VUT v Brně, která byla již dříve experimentálně fyzicky realizována stejnou technologií. Vlastní použité polymorfní hradlo, jeho parametry a chování, je popsáno detailně v části 3.5.1 této práce. Na obrázku 5.7 jsou polymorfní hradla (32 hradel pro 32 konfigurovatelných elementů) pro porovnání plochy hradel vůči zbytku čipu vyznačena červeným rámečkem. Po obvodu čipu se nachází

kontaktovací plošky jednotlivých vývodů čipu pro připojení vodičů k vývodům pouzdra. U každé plošky jsou také soustředěny pomocné budičové obvody a ESD ochrana. Vlastní střed čipu tvoří propojovací síť (osmivstupové multiplexory s příslušnými metalickými spoji, viz obrázek 5.5) [Sek09c]. Napájení čipu REPOMO je odvozeno od napájení použitých polymorfních hradel. Protože všechny ostatní obvody na čipu jsou řešeny technologií CMOS a mají tudíž široký rozsah napájecího napětí, celý čip REPOMO pracuje s takovým napájením a úrovními logických signálů, jaké mají právě polymorfní hradla.

### 5.3 REPOMO32/kit

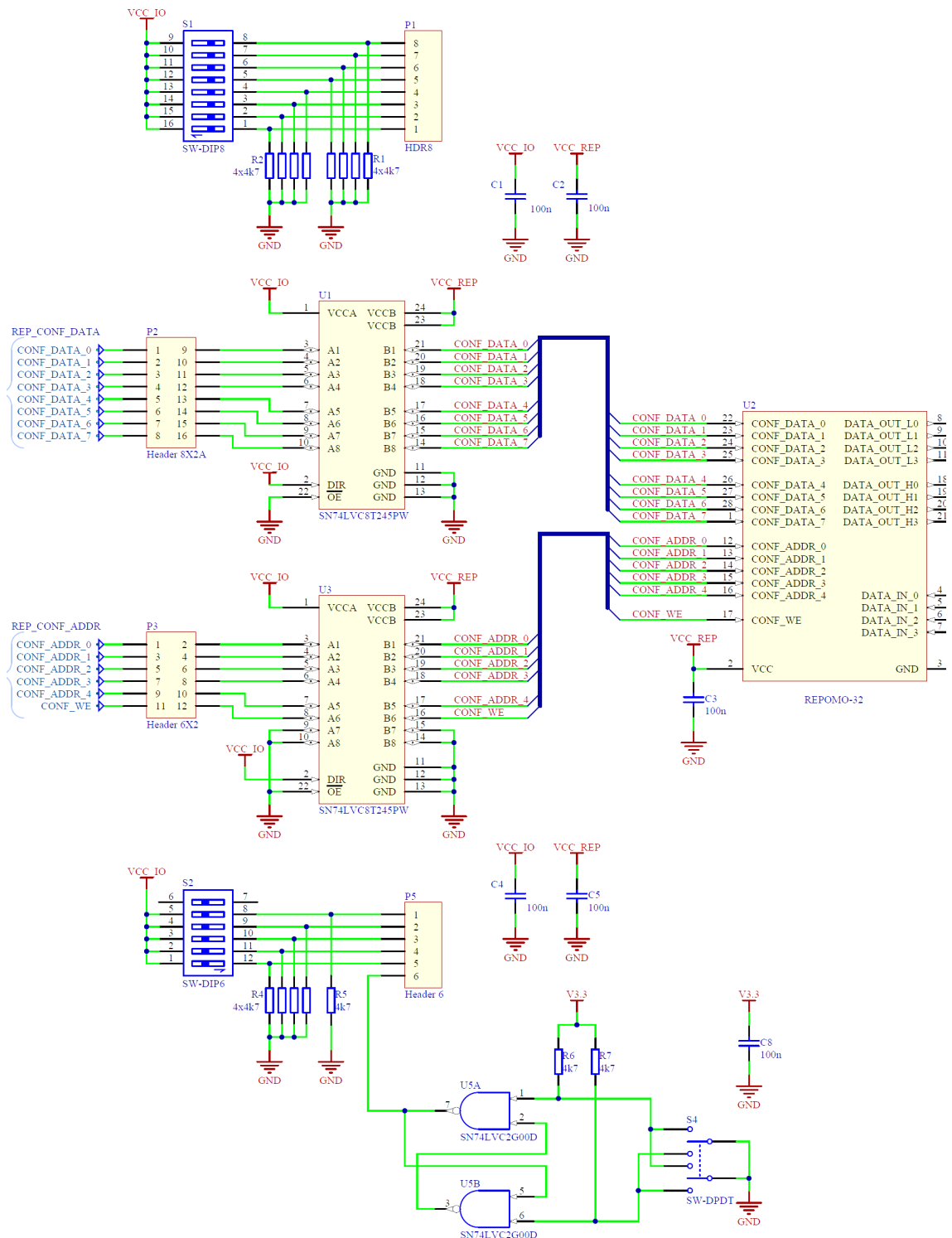
Pro podporu experimentování s čipem REPOMO byl autorem práce ve spolupráci s V. Šimkem vyvinut kit, který dokáže poskytovat čipu různá napájecí napětí, podporuje konfiguraci čipu v manuálním či automatickém režimu, podporuje manuální či automatické generování logických signálů pro vstupy obvodu a sledování i další zpracování výstupních signálů čipu. Navíc také dokáže měnit a udržovat pracovní teplotu čipu pro experimenty s polymorfními obvody řízenými teplotou.



Obrázek 5.8: Blokové schéma kitu pro obvod REPOMO.

Obrázek 5.8 ukazuje blokové schéma kitu. Uprostřed je vlastní obvod REPOMO, kit mu poskytuje napájení, stimuly na jeho vstupy a zpracovává výstupy a také umožňuje jeho konfiguraci. Všechny funkce mohou být nastavovány manuálně, kit tedy může pracovat jako samostatná a nezávislá jednotka, na níž dokáže experimenty provádějící pracovník nastavit přímo veškeré parametry. Je však také možné připojit kit k počítači či jinému generátoru vstupních stimulů a simulovat tak složitější chování. Rovněž konfigurace může probíhat automatizovaně. Výstupy pak mohou být snímány osciloskopem, logickým analyzátozem, počítačem či zpracovány jiným druhem zařízení. V tomto režimu může kit také sloužit jako

podpora pro evoluční návrh polymorfních obvodů, kdy v čipu REPOMO jsou ověřována kandidátní řešení. Automatizovaně i manuálně je možno rovněž měnit napájecí napětí a teplotu čipu.

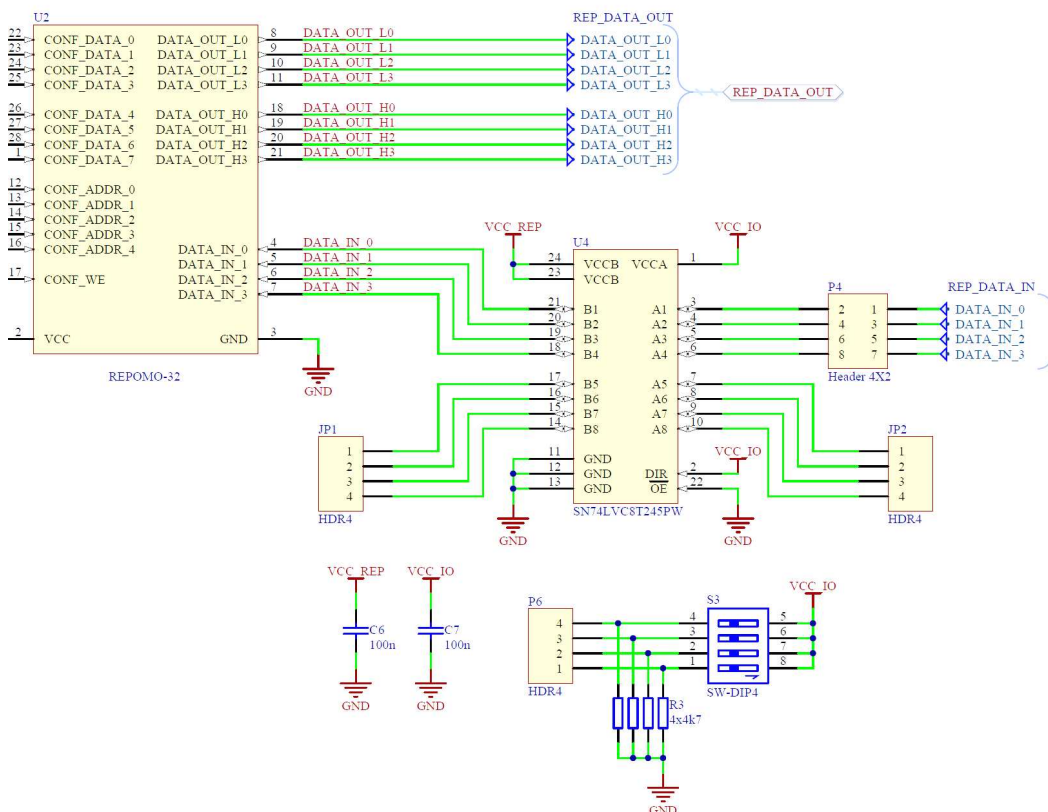


Obrázek 5.9: Schéma konfigurační části kitu.

Na obrázku 5.9 je vidět řešení konfigurační části kitu. Jak konfigurační data, tak i adresa pro konfigurační paměť jsou od čipu odděleny budiči 74LVC8T245 (budiče U1 a U3) [TI05], aby bylo možno generovat externě konfigurační signály například z počítače s běžnými logickými

úrovněmi, i když vlastní čip pracuje s odlišným napájecím napětím, nebo se jeho napájecí napětí dokonce mění. Proto mají budiče vnější stranu (stranu A) napájenou napájecím napětím z pomocného zdroje (3,3 V, případně je možné přivést pro 100% kompatibilitu napájení přímo z přístroje, který konfigurační informaci generuje), zatímco vnitřní strana budičů (strana B) již pracuje se stejným napájecím napětím jako REPOMO. Manuální konfigurace je možná pomocí přepínačů DIP, kdy na přepínačích S2 se v binárním kódu nastaví příslušná adresa konfigurační paměti (číslo elementu, který se bude konfigurovat) a na přepínačích S1 v binárním kódu konfigurační slovo (formát viz část 5.2). Vlastní zápis do konfigurační paměti se provede stiskem tlačítka S4 v okamžiku, kdy je již adresa i data nastavena. Tlačítko S4 generuje zápisový signál conf\_we (je ošetřeno proti mechanickým zákmitům klopným obvodem z U5A/U5B). Tak řešená konfigurace dovoluje ručně změnit kteroukoli buňku konfigurační paměti bez ovlivnění zbytku obsahu. Výběr manuální či automatizované (z vnějšího zdroje dat) konfigurace se provádí zkratovacími propojkami zapojenými na špičky P1/P2 a P3/P5.

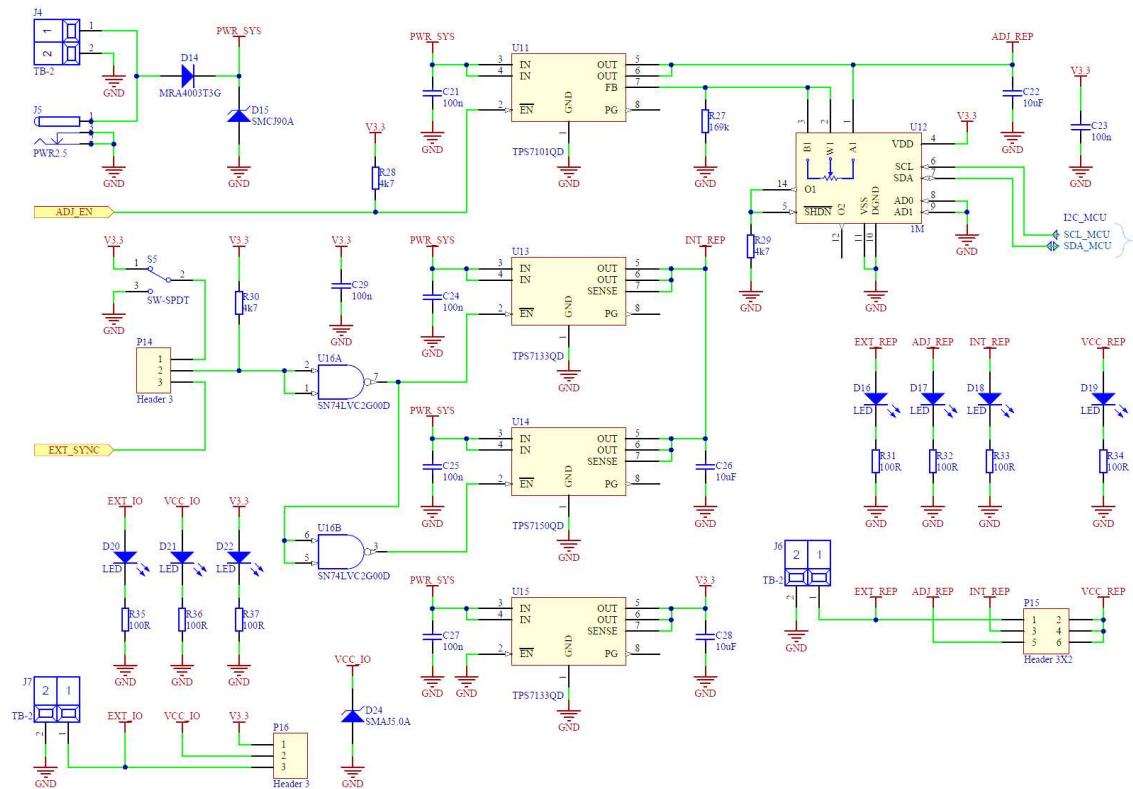
Velice podobným způsobem jako konfigurace čipu je řešen i vstup dat pro primární vstupy matice elementů uvnitř čipu REPOMO. Vstupní data je možno zadávat buď manuálně pomocí DIP přepínačů (přepínače S3 na obrázku 5.10) nebo přivádět z externího zdroje. Signály jsou pro přizpůsobení úrovní opět odděleny budičem (budič U4). Volba manuální nastavení/externí zdroj dat je prováděna opět zkratovacími propojkami na poli kolíků P4/P6. Protože obvod REPOMO má pouze čtyři primární vstupy a budič 74LVC8T245 je schopen zpracovat 8 signálů, jsou zbývající 4 vstupy a výstupy budiče vyvedeny samostatně k volnému použití pro případné další signály, které by v aplikaci bylo třeba přizpůsobit.



Obrázek 5.10: Schéma části kitu s datovými vstupy.

Primární výstupy obvodu REPOMO jsou přivedeny do obvodu CPLD typu Xilinx XC9572XL [Xil07]. Tento obvod je na kitu přítomen pro možnost okamžitého automatizovaného (před) zpracování výstupů z čipu. V nejjednodušší variantě jeho konfigurace pouze zobrazuje stav výstupních signálů REPOMO na svítivých diodách, aby bylo možno okamžitě odečítat stav výstupů při samostatném užívání kitu. Je však možno do obvodu implementovat například fitness funkci pro automatické vyhodnocování kandidátních řešení, je-li kit využíván pro evoluční návrh polymorfních obvodů metodou Kartézského genetického programování. Obvod CPLD na kitu posloužil také pro zachycování vzájemného časového posuvu hran výstupních signálů při vyhodnocování experimentů s fyzickými neklonovatelnými funkcemi (blíže viz kapitola 6).

Poměrně komplikovaně je řešena zdrojová část kitu. Je tomu tak proto, že pro čip REPOMO s polymorfními hradly závislými na napájecím napětí musí být za účelem experimentů možno generovat napájecí napětí v širokém rozsahu, ale přitom dostatečně kvalitní a stabilní, aby experimenty byly nedokonalostí zdroje ovlivněny co nejméně. Na druhou stranu však musí zdrojová část poskytovat napájení i ostatním podpůrným obvodům na kitu. Schéma zdrojové části ukazuje obrázek 5.11.



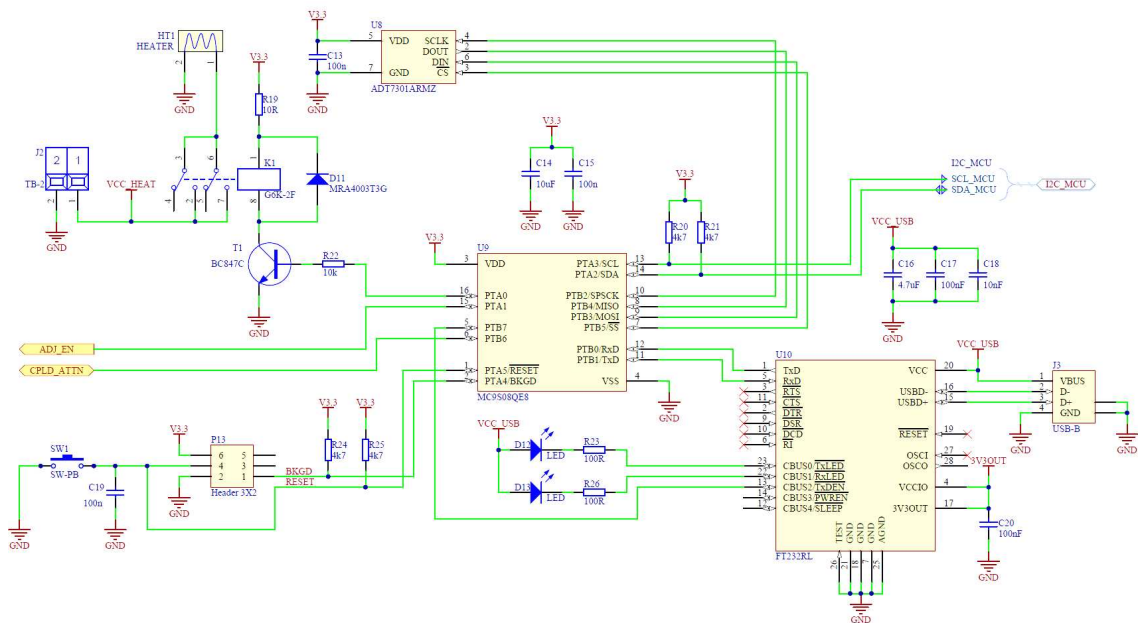
Obrázek 5.11: Schéma zdrojové části kitu.

Zdroj pro čip REPOMO může pracovat v následujících režimech:

- ruční ovládání napájení přepínačem 3,3/5 V,
- automatické přepínání 3,3/5 V vnějším signálem (např. z řídicího počítače),
- plynulá regulace napájecího napětí řízená mikrokontrolérem (možné řízení též počítačem přes USB rozhraní),
- externí napájení ze zdroje mimo kit.

Jak již bylo zmíněno, pro experimenty s polymorfními hradly, jejichž funkce je řízena napájecím napětím, jsou nároky na zdroj vysoké. S ohledem na stabilitu a minimální zvlnění napájecího napětí byly zvoleny lineární regulátory napájecího napětí s nízkým úbytkem řady TPS71XX [TI94]. Pro napájení pomocných obvodů kitu včetně CPLD je osazen regulátor TPS7133 (pozice U15 na obrázku 5.11) s pevným výstupním napětím 3,3 V.

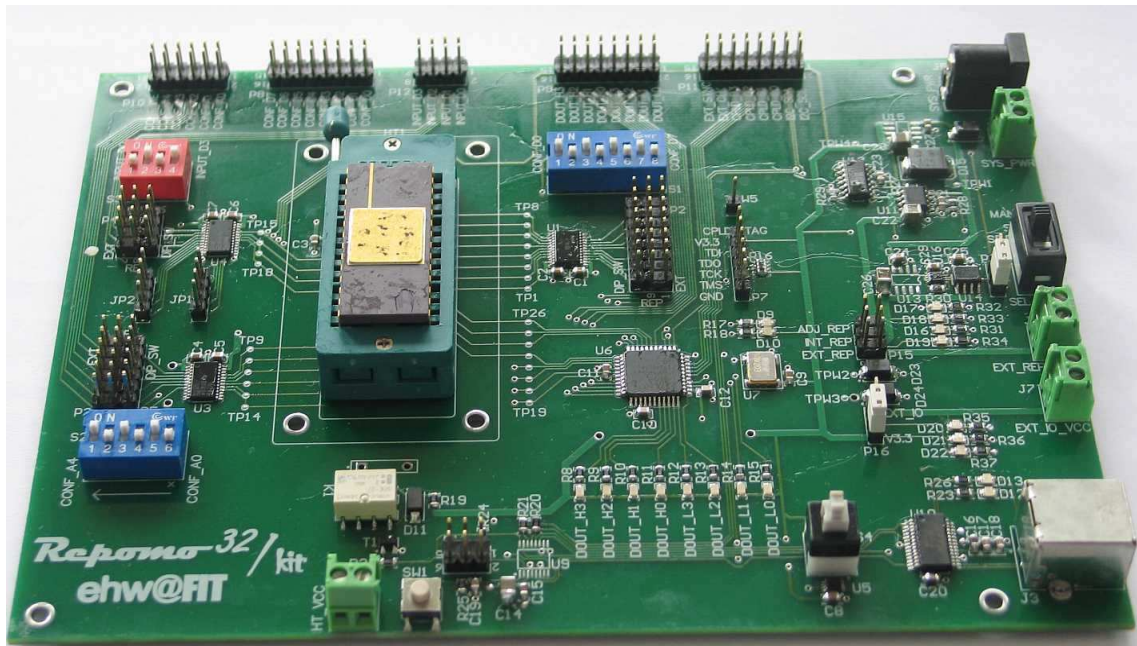
Napájení pro čip REPOMO zajišťuje buď dvojice zdrojů s pevným výstupním napětím TPS7133 (3,3 V) a TPS7150 (5,0 V), mezi kterými se přepíná buď ručně přepínačem S5 nebo logickým signálem z vnějšího zdroje (log. 0 znamená, že REPOMO bude napájeno 5V, log. 1 znamená, že napájení bude 3,3 V, volba mezi ručním přepínáním a přepínáním vnějším signálem se provádí zkratovací propojkou na pinech P14) nebo zdroj TPS7101 (pozice U11) s nastavitelným výstupním napětím. Nastavení výstupního napětí zdroje se děje elektronickým potenciometrem U12, ovládaným po sběrnici I2C z mikrokontroléru (viz též obrázek 5.12). Výstupní napětí tohoto zdroje je možno teoreticky nastavovat v rozmezí 1,2 až 8,1 V (dáno rozsahem elektronického potenciometru 1 M $\Omega$ ) s krokem asi 27 mV (protože použitý digitální potenciometr AD5241 [AD09] na pozici U12 má 256 kroků).



Obrázek 5.12: Schéma ovládací části kitu s mikrokontrolérem.

Na obrázku 5.12 je vidět část kitu s řídicím mikrokontrolérem (U9) pro řízení prostředí, ve kterém polymorfní obvod pracuje. Mikrokontrolér může autonomně řídit zdroj v režimu plynulého nastavení napájecího napětí (přes sběrnici I2C nastavovat potenciometr U12) nebo řídit teplotu čipu prostřednictvím elektrického vyhřívání komory s čipem REPOMO. Pro řízení teploty slouží jako akční člen topná spirála HT1, ovládaná přes relé K1. Aby bylo možno přesně regulovat teplotu, je zajištěna zpětná vazba přes teplotní čidlo ADT7301 (pozice U8 na obrázku 5.12). Čidlo je k mikrokontroléru připojeno prostřednictvím sběrnice SPI, dosahuje přesnosti  $\pm 1^\circ\text{C}$  a informace o teplotě je kódována na 13ti bitech [AD11]. Mikrokontrolér je také možno propojit s řídicím počítačem přes sériovou linku emulovanou na rozhraní USB (obvodem FT232RL, pozice U10 [FT10]). Řídicí počítač tak může sledovat i ovlivňovat jak napájecí napětí, tak teplotu polymorfního obvodu.





Obrázek 5.13: Skutečný vzhled kitu.

Na obrázku 5.13 je fotografie hotového REPOMO32/kit-u. V patici s nulovou základací silou je vložen integrovaný obvod REPOMO. Patice s nulovou základací silou byla zvolena proto, aby bylo možno snadno měnit čipy při ověřování shody či rozdílů v parametrech jednotlivých čipů (velmi užitečné například pro experimenty s fyzickou neklonovatelnou funkcí, podrobněji viz kapitola 6 této práce). Na vyfotografované desce není osazena ohřívací komora, vyrobená z plechu, kterou se překrývá obvod REPOMO a která obsahuje topné těleso.

## 6 Aplikace kombinačních polymorfních obvodů

### 6.1 Možnosti využití vícefunkčních obvodů s reakcí na prostředí

Již A. Stoica a jeho tým v práci, ve které představili koncepci polymorfní elektroniky [Sto01], uvádí řadu možných oblastí, kde by polymorfní elektronika mohla být s výhodou uplatněna. Jako hlavní přednost vidí možnost navrhovat obvody, které mají k „hlavní“ funkci přidanou jednu nebo více dalších funkcí, které mohou být v případě potřeby vyvolány. Tyto funkce „navíc“ mohou být přirozeně aktivovány za příslušných vnějších podmínek nebo prostě vyvolány v případě potřeby. Možné využití funkcí „navíc“ vidí tým A. Stoicy zejména v bezpečnostní oblasti, například:

- autentizační mechanismus či „vodoznak“ pro identifikaci,
- ochrana proti reverznímu inženýrství (skutečná funkce se projeví jen za určitých podmínek),
- ochrana před neautorizovaným použitím zakódováním biometrických dat do obvodu,
- poskytnutí dalšího komunikačního kanálu,
- netradiční technika pro infiltraci senzorů do chráněných oblastí a získávání dat z nich,
- netradiční značkovací technika, atd.

Stoica se svým týmem například navrhuje polymorfní obvod, který za normálních podmínek funguje jako generátor hodinového signálu. Když pak je aplikován speciální klíč (určitá teplota nebo průběh teploty, elektromagnetické impulsy, změna napájecího napětí atd.), obvod může vygenerovat dávku impulsů, které odemknou či probudí speciální kódovací schéma.

Jiná aplikace s využitím biometrických dat může například využít polymorfní elektroniky tak, že nasnímaný otisk prstu či jiný biometrický vzor se převede například na matici popisující různá napětí, která se pak aplikují na různé části obvodu. To způsobí nastavení té správné „druhé“ funkce ve všech částech obvodu tak, že celek funguje zamýšleným způsobem. Tak může vzniknout personalizovaný čip s biometrickými údaji přímo „zadrátovanými“ v křemíku.

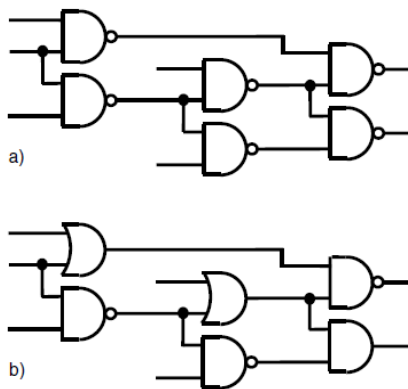
Dále je v práci [Sto01] zmíněn potenciál polymorfní elektroniky sloužit v aplikacích, kde je třeba zvýšená odolnost proti poruchám a možnost adaptace na změny prostředí. Přidané funkce mohou převzít kontrolu nad zařízením, když hrozí nějaké detekovatelné nebezpečí. Polymorfní elektronika také umožňuje vyvíjet systémy, které velmi rychle mění funkce podle toho, jak se vyvíjí okolní prostředí. Je to možné díky tomu, že změna funkce je, na rozdíl od tradiční rekonfigurace, u polymorfních obvodů téměř okamžitá. Oproti přepínání několika modulů zase polymorfní elektronika nabízí daleko méně prostorově náročné implementace.

Autoři navrhuje například aplikaci, kdy se nějaký obvod musí adaptovat na zhoršené podmínky jako je klesající napájecí napětí vlivem vybíjející se baterie či podobně. Jako příklad uvádí 16ti bitový převodník, který může v případě poklesu napětí zdroje přirozeně omezit vzorkování třeba na 8 bitů, případně i omezit vzorkovací frekvenci. Fakticky ale tým A. Stoicy žádnou aplikaci konkrétně neřešil, jde pouze o návrhy a náměty. Ani další tým, který se zabývá polymorfní elektronikou, tým W. Lua z Číny, nepublikoval žádné konkrétní aplikace, pouze některé svoje návrhy dvojfunkčních obvodů s blíže nespecifikovaným účelem [Luo07]. Řadu aplikací však navrhl a implementoval tým z FIT VUT v Brně. Některé nejzajímavější a reprezentativní návrhy budou popsány dále v této kapitole a také v kapitole následující.

## 6.2 Aplikace polymorfních obvodů v diagnostice a testování

### 6.2.1 Zkrácení testu číslicového obvodu s využitím polymorfních hradel

L. Stareček [Sek08] navrhl metodu pro zlepšení testovatelnosti číslicových obvodů, využívající polymorfní elektroniky. Experimenty prokázal, že záměnou některých hradel v obvodě, aniž by se jinak struktura obvodu změnila, lze generovat podstatně kratší testovací sekvenci k otestování poruch, než kdyby šlo o obvod s původními hradly. Když se vhodně zvolená hradla v takovém obvodě nahradí polymorfními hradly, je možné, aby obvod v jednom režimu pracoval (v tomto režimu musí polymorfní hradla realizovat takovou funkci, kterou mají mít hradla, nahrazená polymorfními hradly, dle původního návrhu obvodu) a ve druhém režimu probíhal test. Test bude podstatně kratší a tudíž i levnější. Pokud se použijí například polymorfní hradla řízená napájecím napětím, není ani třeba přidávat žádné další spoje. Obvod bude pracovat při takovém napájecím napětí, které zajistí polymorfním hradlům požadovanou funkci. Pro test se jednoduše napájecí napětí změní, aby se změnila i funkce polymorfních hradel. Předpokládá se generování testu, který testuje strukturu obvodu a nikoliv vnitřní funkci hradel.



Obrázek 6.1: Benchmarkový obvod C17 a) a jeho modifikace pro kratší test b).

Na obrázku 6.1 a) je vidět benchmarkový obvod C17 ze sady benchmarkových obvodů ISCAS-85 [Brg85]. Sestává ze šesti hradel typu NAND. Pokud je pro něj generován test pro 100% pokrytí poruch nástrojem Mentor Graphics FlexTest, je vygenerován test obsahující devět testovacích vektorů. Pokud se dvě z hradel obvodu změní na NOR (viz obrázek 6.1 b), test generovaný pro obvod za stejných podmínek sestává z pouze pěti testovacích vektorů.

Problém spočívá v tom, jak identifikovat ta hradla, jejichž záměna povede k nejlepší redukci objemu testu. Vzhledem k tomu, že stavový prostor může být rozsáhlý a jeho prohledání hrubou silou může být nepraktické, navrhuje L. Stareček použít heuristické metody odvozené od horolezeckého algoritmu.

Výsledky experimentů byly publikovány v [Sek08] a [Sta08] a jsou přehledně shrnuty v tabulce 6.1. Zdá se, že je možno dosáhnout redukce délky testu průměrně o 30%. Je však také vidět, že redukce délky testu je vykoupena zvětšením zabrané plochy (zde vyjádřeno počtem tranzistorů implementace). Prvních šest obvodů v tabulce jsou rozsahem malé a jednoduché kombinační obvody – tři různé implementace plně jednobitové sčítačky (fulladd1, fulladd2 a fulladd3), tříbitový komparátor (comp3bit), kodér kódu 1z8 na binární kód (enc8to3) a dekodér binárního kódu na kód 1z8 (dec3to8). Tyto obvody svým rozsahem ještě dovolily prohledat celý stavový prostor. Avšak jen malé procento možných modifikací obvodu vedlo k úspoře. U sčítaček se jednalo o asi 17% všech možností, u komparátoru jen necelých 1,5% modifikací vedlo k úspoře,

u dekodéru asi 2% a u kodéru méně než 1%. Přitom u posledně jmenovaných tří obvodů už počet prohledávaných kombinací dosáhl řádu statisíců.

Zbývá čtveřice obvodů v tabulce jsou „benchmarkové“ obvody ze sady ISCAS-85 [Brg85]. Pro tyto obvody již bylo použito heuristického prohledávání, protože svým rozsahem překračují možnosti prozkoumávání všech kombinací. I tak trvalo hledání vhodného řešení asi 4 dny. Nejzajímavějšího výsledku bylo dosaženo s obvodem c6288 (jedná se o 16ti bitovou násobičku). Při nárůstu počtu tranzistorů o pouhé 1% bylo dosaženo redukce počtu testovacích vektorů o 21%. Naproti tomu pro obvod c1355 (32bitový SEC korekční obvod) nebylo vůbec nalezeno řešení, které by počet testovacích vektorů snížilo.

Obvod	počet hradel	počet modifikovaných hradel	redukce délky testu o	zvýšení počtu tranzistorů o	původní pokrytí poruch	nové pokrytí poruch
fulladd1	4	1	17%	19%	100%	100%
fulladd2	6	3	37%	60%	100%	100%
fulladd3	5	2	37%	29%	100%	100%
comp3bit	11	2	18%	13%	100%	100%
enc8to3	13	1	9%	11%	100%	100%
dec3to8	11	4	50%	32%	100%	100%
c6288	2416	11	21,7%	1%	99,56%	99,56%
c432	160	27	44,1%	37%	99,24%	99,82%
c2670	1269	43	34,4%	7%	95,74%	96,56%
c1355	546		0%			

Tabulka 6.1: Některé výsledky aplikace metody L. Starečka.

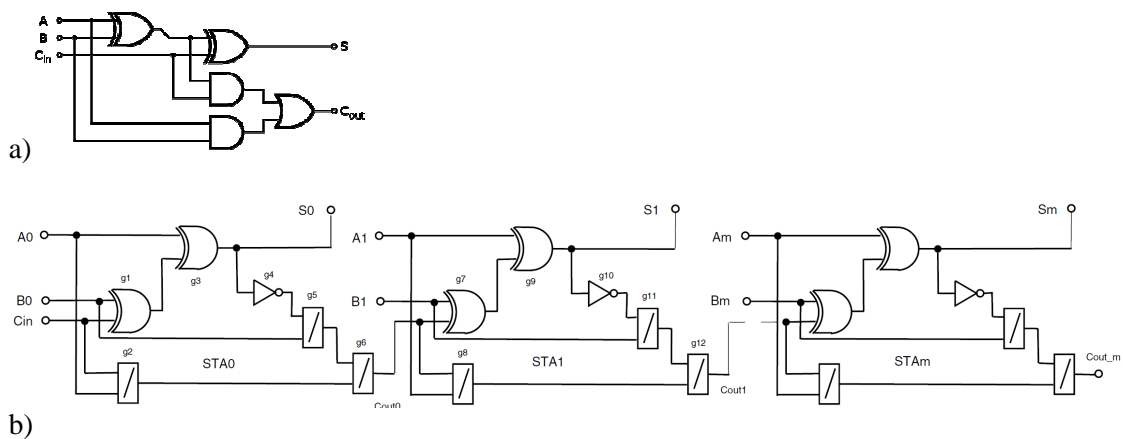
## 6.2.2 Samotestující se sčítačky

L. Sekanina navrhl a ve spolupráci s autorem této práce experimentálně ověřil použití polymorfni elektroniky v návrhu samotestujících se obvodů [Ruz08a]. Tradiční přístup k samotestujícím se obvodům znamená vně obvodu umístěný kontrolní obvod (checker), který posuzuje správnost výsledku produkovaného obvodem buď porovnáním výstupu z více zdrojů nebo detekcí bezpečnostního kódu (například vlastní obvod produkuje výsledek zakódovaný v nějakém bezpečnostním kódu a kontrolní obvod pak posuzuje, jestli výstup je platným kódovým slovem v užitém kódu) [Pra96], [Dia79]. Nový přístup představil M. Garvie [Gar05], který pomocí evolučního návrhu vytvořil malé kombinační obvody, které mají vlastnost samodetekce chyby a přitom jsou rozsahem menší než konvenční řešení. Takové obvody vlastně slučují funkce vlastního obvodu a kontrolního obvodu do jednoho kompaktního celku. Obvod má pak speciální signály, které indikují, zda je funkční (výstup je platný) či nikoliv. Při návrhu složitějších systémů sestávajících z takových jednoduchých samotestujících se obvodů se pak speciální signály nesoucí informaci o bezchybném stavu či případné poruše jednotlivých komponent musí agregovat, aby byla získána globální informace o funkčnosti či poruše celého systému nebo dílčího podsystemu.

L. Sekanina navrhl ve své práci [Sek07] několik jednobitových sčítaček s funkcí samotestování, které výstup indikující funkční nebo chybový stav nepotřebují. Tuto informaci lze získat na jejich datových výstupech. Koncept využívá polymorfni hradel se dvěma režimy činnosti (hradla realizují dvě různé logické funkce). Za normálních okolností (bez poruchy) obvod generuje stejný výstup, ať jsou polymorfni hradla v jakémkoliv režimu. Pokud se však vyskytne v obvodě porucha, změna režimu hradel způsobí změnu výstupu signálu přenosu. Takový způsob indikace poruchy navíc dovolí propagaci informace o poruše strukturou obvodu po existujících spojích. Otestování obvodu lze kdykoliv v průběhu činnosti obvodu provést tak, že

se změny režim činnosti hradel. Pokud se použijí například polymorfní hradla řízená úrovní napájecího napětí, není třeba ani dalšího rozvodu pro signál k otestování, opět se využijí stávající rozvody v obvodě.

Myšlenka využití polymorfních hradel použitých v návrhu samotestující se sčítačky byla předtím publikována ve výzkumné zprávě týmu A. Stoicy z NASA [Zeb06a]. Tam publikovaná sčítačka byla poměrně velká, sestávala ze 14ti polymorfních hradel (tedy 84 tranzistorů). Indikuje poruchu na obou výstupech (součet S i přenos Cout), což je nepraktické zejména z pohledu agregace informací o poruše jednotlivých jednobitových sčítaček (zatímco signál přenosu se propaguje přes všechny bity, součet nikoliv). Sčítačka nebyla nikdy implementována a ověřena.



Obrázek 6.2: Běžná implementace jednobitové sčítačky a) a jednobitové sčítačky se samotestováním zapojené do vícebitové sčítačky s postupným šířením přenosu b).

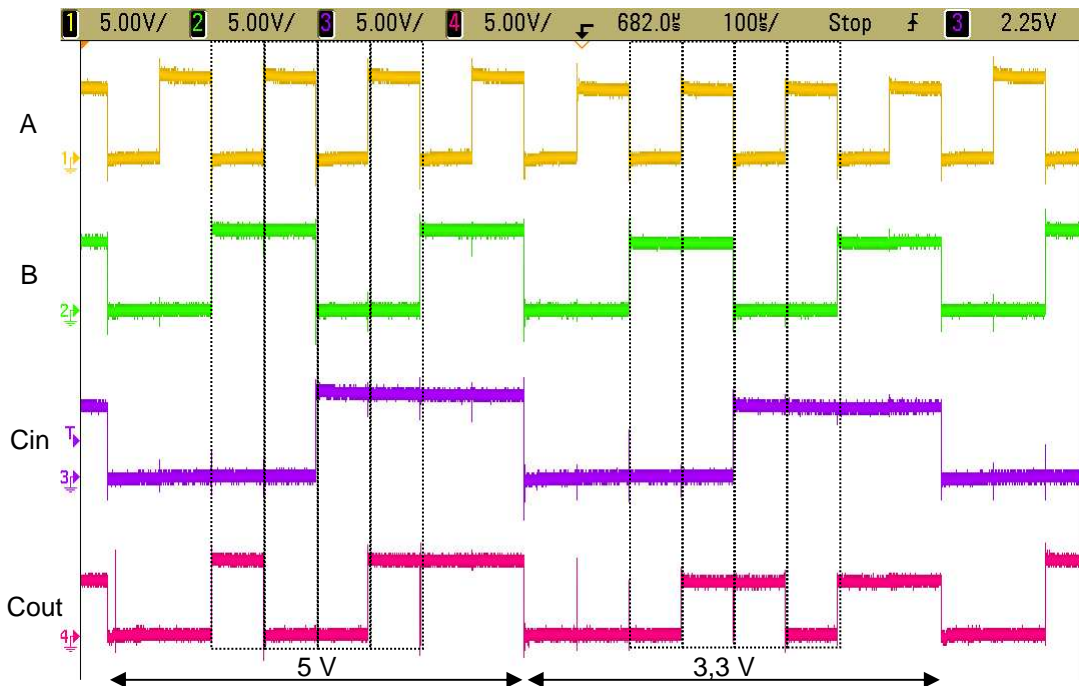
Na obrázku 6.2 b) je vidět zapojení nejlepší navržené samotestující se jednobitové sčítačky do vícebitové sčítačky s postupným šířením přenosu (obrázek ukazuje tři instance jednobitové sčítačky) [Ruz08a]. Síť, která realizuje součet S dvou vstupních bitů A a B a přičtení přenosu z předchozího bitu C<sub>in</sub> je řešena konvenčně pomocí hradel typu XOR (lze srovnat s konvenčním řešením jednobitové sčítačky na obrázku 6.2 a). Polymorfní hradla jsou použita v síti generující výstup signálu přenosu Cout. Konvenční sčítačka využívá hradel typu AND a OR (obrázek 6.2 a), zatímco navržená samotestující se sčítačka zde používá tři polymorfní hradla NAND/NOR. Cena implementace úplné samotestující se jednobitové sčítačky je 42 tranzistorů (2x 8 tranzistorů na XOR hradla, 3x 8 tranzistorů hradla NAND/NOR řízená V<sub>dd</sub>, 1x 2 tranzistory inverter). Známé implementace konvenční úplné jednobitové sčítačky sestávají z minimálně 24 tranzistorů [Mar02]. Protože za úspěšné implementace samotestujících se obvodů se považují takové, které jsou menší než dvojnásobek konvenční implementace (bez vlastnosti samotestování), lze navrženou samotestující se sčítačku považovat za přínosnou. Existují sice menší implementace samotestujících se sčítaček – například v [Mar02] je zmíněna samotestující se sčítačka obsahující 36 tranzistorů, ale představená sčítačka s polymorfními hradly navíc, na rozdíl od konvenčního přístupu k samotestujícím se obvodům, nepotřebuje speciální signály ani pro spuštění testu, ani pro výstup informace o poruše.

Klíčem k pochopení, jak lze detekovat poruchu v samotestující se sčítačce, je tabulka 6.2. Obvod dokáže detekovat poruchy typu trvalá 0 nebo trvalá 1 na kterémkoliv signálu uvnitř hradla. Signály jsou označeny podle toho, výstup kterého hradla signál budí (M<sub>g1</sub> – M<sub>g6</sub>). Z tabulky 6.2 je vidět, že ne každá kombinace na vstupu způsobí signalizaci každé poruchy (změnou stavu výstupu Cout při změně funkce hradel g2, g5 a g6). V případě, že je třeba kompletní otestování, lze najít úplné pokrytí s minimálním počtem testovacích vektorů. Takové

pokrytí může být třeba {001, 010, 011, 101}, {001, 010, 101, 110} nebo {010, 100, 101, 110}. Jako praktické se jeví on-line testování při běžném provozu, kdy pokud se vyskytnou na vstupu vhodná data, provede se změna režimu polymorfních hradel (například pokud jde o hradla řízená  $V_{dd}$ , provede se krátkodobá změna napájecího napětí) a klopným obvodem se zachytí případný výskyt hrany na Cout. Výskyt hrany indikuje některou z poruch, kterou testovací vektor, právě se nacházející na vstupu, pokrývá (hvězdička v tabulce 6.2).

testovací vektor Cin, B, A	0	0	0	0	1	1	1	1	porucha
$M_{g1}$			*	*	*	*			trvalá 0
$M_{g2}$		*			*				trvalá 0
$M_{g3}$		*	*		*				trvalá 0
$M_{g4}$				*		*	*		trvalá 0
$M_{g5}$	*		*						trvalá 0
$M_{g6}$									trvalá 0
$M_{g1}$		*				*	*		trvalá 1
$M_{g2}$				*		*	*		trvalá 1
$M_{g3}$				*		*	*		trvalá 1
$M_{g4}$		*	*		*				trvalá 1
$M_{g5}$					*		*		trvalá 1
$M_{g6}$							*	*	trvalá 1

Tabulka 6.2: Pokrytí poruch testovacími vektory pro samotestující se sčítačku z obrázku 6.2 b).

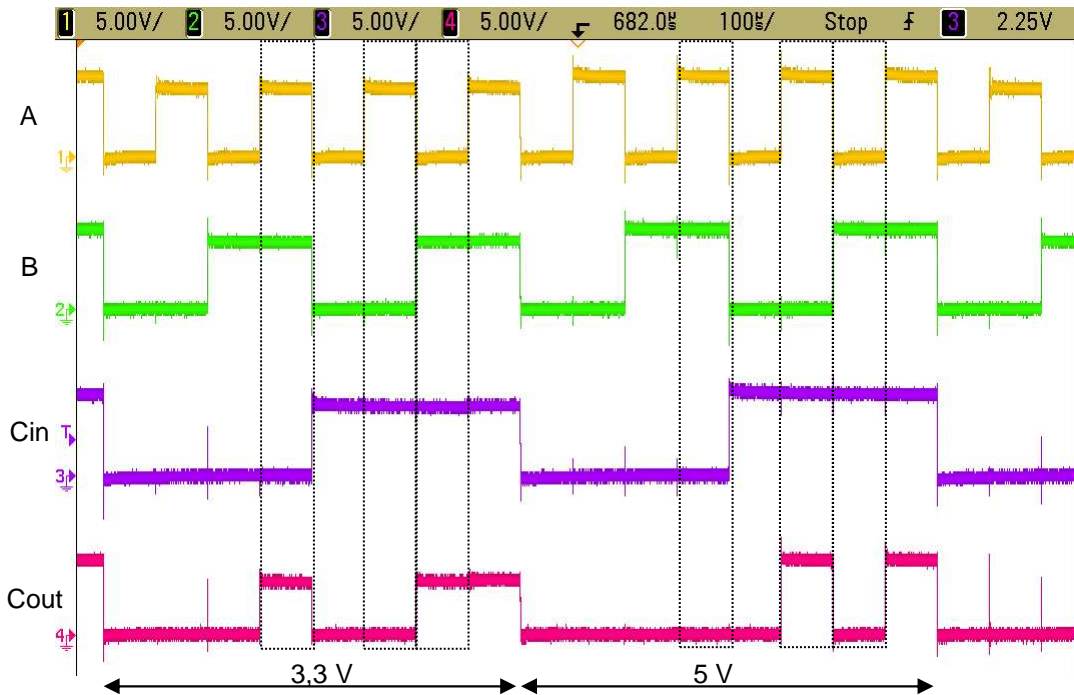


Obrázek 6.3: Průběh signálů na vstupech A, B a Cin a na výstupu Cout sčítačky při poruše typu trvalá 0 na výstupu hradla  $g1$ .

Samotestující se sčítačka, jejíž schéma je na obrázku 6.2 b), byla realizována z polymorfních hradel NAND/NOR (které byly navrženy ve spolupráci FIT a FEKT VUT v Brně) a konvenčních CMOS hradel XOR z obvodu 4030 a předpokládané chování bylo prakticky ověřeno podle tabulky 6.2. Na vstupy sčítačky byly přikládány všechny kombinace hodnot s

opakovací frekvencí 9,6kHz, napětí zdroje bylo přepínáno mezi 3,3 V a 5V s frekvencí 1,2 kHz. (tedy vždy po úplné sadě testovacích vektorů). Sčítačka může samozřejmě pracovat na daleko větší frekvenci (viz limit frekvence polymorfních hradel uvedený v kapitole 3), uvedená frekvence byla zvolena s ohledem na v experimentech použitý zdroj a snadnost vyhodnocení.

Na obrázku 6.3 je záznam experimentu, kdy na signálu buzeném výstupem hradla g1 byla uměle vytvořena porucha typu trvalá log. 0. Je vidět, že pro vstupní vektory 010, 011, 100, 101 (vyznačeno čárkovaně) je výstup Cout odlišný pro režim polymorfních hradel NAND ( $V_{dd} = 5$  V) a NOR ( $V_{dd} = 3,3$  V), zatímco pro ostatní vstupní vektory je úroveň pro obě hladiny  $V_{dd}$  stejná. Je to v souladu s prvním řádkem tabulky 6.2.



Obrázek 6.4: Průběh signálů na vstupech A, B a Cin a na výstupu Cout sčítačky při poruše typu trvalá 1 na výstupu hradla g3.

Na obrázku 6.4 je záznam experimentu, kdy na signálu buzeném výstupem hradla g3 byla uměle vytvořena porucha typu trvalá log. 1. Je vidět, že pro vstupní vektory 011, 101, 110 (vyznačeno čárkovaně) je výstup Cout odlišný pro režim polymorfních hradel NAND ( $V_{dd} = 5$  V) a NOR ( $V_{dd} = 3,3$  V), zatímco pro ostatní vstupní vektory je úroveň pro obě hladiny  $V_{dd}$  stejná. Je to v souladu s devátým řádkem tabulky 6.2.

### 6.3 Zvýšení odolnosti proti poruchám aplikací polymorfní elektroniky

V kapitole 2 bylo ve vymezení pojmu „polymorfní elektronika“ uvedeno, že aktuálně provádění funkce polymorfního obvodu závisí na stavu okolního prostředí a dále že polymorfní obvody se vyznačují kompaktní stavbou – svými rozměry jsou výrazně skromnější, než kdyby šlo o konvenční implementaci vícefunkčního obvodu. Tyto vlastnosti přímo předurčují polymorfní obvody pro nasazení v aplikacích, kde je třeba předejít havárii systému způsobené nepříznivými okolními podmínkami reakcí s předstihem. Již A. Stoica s kolektivem ve své úvodní práci o polymorfní elektronice [Sto01] navrhuje aplikaci, kdy při rostoucí teplotě či poklesu napájecího napětí polymorfní obvod realizující například analogově-číslicový převodník samočinně omezí

šířku slova a četnost vzorkování, aby tak snížil energetickou ztrátu obvodu po dobu trvání nepříznivých podmínek a zabránil tak totálnímu kolapsu systému zachováním alespoň minimální funkčnosti. Obdobnou aplikaci (ovšem na úrovni sekvenčních obvodů) popisuje také kapitola 7.

L. Sekanina s autorem této práce a Z. Gajdou publikovali jinou aplikaci polymorfní elektroniky pro zvýšení odolnosti obvodu proti poruchám snížením spotřeby v kritických okamžicích [Sek09b]. Podobně jako u aplikace navržené týmem A. Stoicy, i v tomto případě jde o aplikaci na zpracování signálů. Autoři navrhují filtry s konečnou impulsní odezvou (FIR), které mohou omezit svoji spotřebu snížením počtu členů (koeficientů), přičemž zbývající členy se rekonfigurují tak, aby odezva nového (jednoduššího, s menším počtem členů, koeficientů) filtru byla co nejpodobnější původnímu. Snížení počtu koeficientů samozřejmě znamená, že tvar odezvy nelze přesně dodržet, filtr s menším počtem koeficientů bude mít zřejmě odezvu o něco horší, parametry filtrovaného signálu se tím patrně zhorší. Ušetří se však na spotřebě obvodu a překoná se období nedostatku energie či zvýšené teploty systému (a okolí) za cenu zhoršených parametrů, ale bez úplného výpadku funkce.

Číslicová filtrace je důležitou oblastí moderního zpracování signálů. Nachází uplatnění v řadě aplikací, mezi nejčastější patří potlačování šumu a jiných nežádoucích složek signálů. Číslicová filtrace upravuje hodnoty vzorků signálu podle určitého algoritmu. Filtry se nejčastěji popisují lineárními diferenčními rovnicemi s konstantními koeficienty. Rovnice má následující tvar:

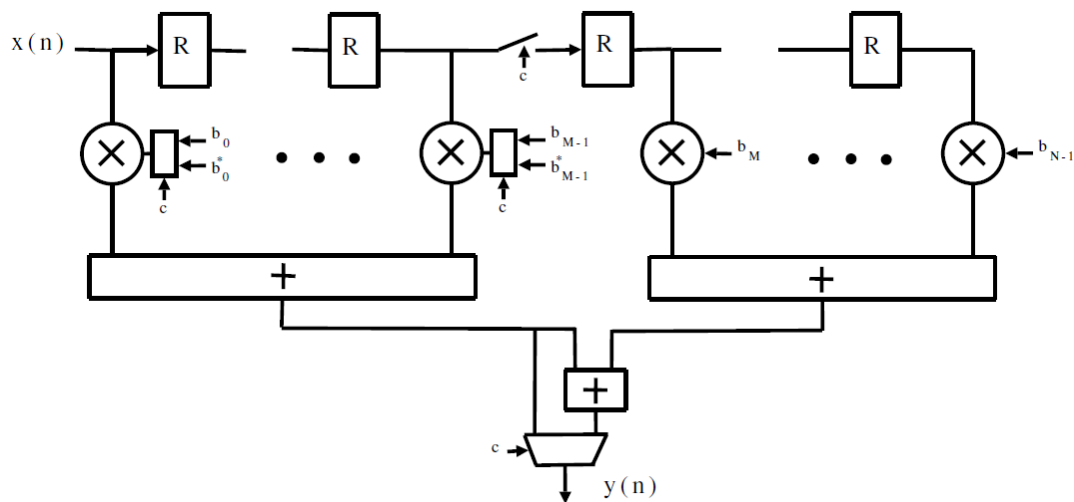
$$y(n) = \sum_{k=0}^N b_k x(n-k) - \sum_{k=1}^J a_k y(n-k)$$

kde  $y$  je výstupní signál,  $x$  je vstupní signál a  $a_k, b_k$  jsou konstantní koeficienty. Je vidět, že filtr kombinuje  $N$  předchozích vzorků vstupního signálu a  $J$  předchozích kroků výstupního signálu vážených koeficienty  $a$  a  $b$ . Filtry, které popisuje výše uvedená rovnice, se též nazývají filtry s nekonečnou impulsní odezvou (Infinite Impulse Response, IIR). Zvláštním případem filtrů jsou tzv. filtry s konečnou impulsní odezvou (Finite Impulse Response, FIR), které jsou popsány diferenční rovnicí v tomto tvaru:

$$y(n) = \sum_{k=0}^N b_k x(n-k)$$

Je vidět, že u FIR filtru se pro nalezení  $n$ -tého výstupního vzorku používají pouze vstupní vzorky, konkrétně  $N$  posledních vstupních vzorků. Každý vzorek je násoben konstantou  $b$ , součet těchto součinů pak tvoří výstupní vzorek. Pro implementaci filtru jako číslicového obvodu je třeba  $N$  registrů pro uchování posledních  $N$  vzorků vstupního signálu a dále strukturu obvodů, které provádí funkci násobení konstantou a přičtení předchozího mezisoučtu. Přitom operace násobení a její implementace číslicovým obvodem je náročná jak z pohledu složitosti (plochy, počtu tranzistorů) a z pohledu spotřeby energie, tak i z pohledu zpoždění signálu (rychlosti obvodu). V literatuře jsou známy i implementace FIR filtrů bez násobiček [Wad94], [MaP02], [Mas07], kde se vlastní násobička nahrazuje bitovými posuvy a součty, případně rozdíly. To je možné zejména v případě násobení konstantou. Někteří autoři také používají evolučních technik k optimalizaci či návrhu filtru [Har95]. J. Miller dokonce navrhl evolučními technikami zcela nekonvenční filtry na úrovni hradel, kde tradiční struktura násobení a sečítání není vůbec zřetelná [Mil99].



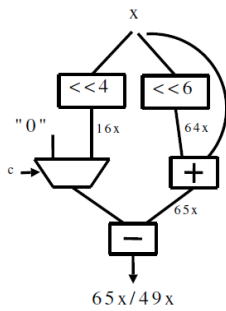


Obrázek 6.5: Struktura polymorfního FIR filtru s nouzovým režimem.

Na obrázku 6.5 je bloková struktura polymorfního FIR filtru, jak ji v roce 2009 publikovali L. Sekanina s autorem této práce a Z. Gajdou [Sek09b]. Registry  $R$  uchovávají posledních  $N - 1$  vzorků vstupního signálu  $x$ , aktuální vzorek  $x(n)$  je k dispozici přímo na vstupu. Obvod dále obsahuje  $N$  struktur pro násobení konstantou  $b$  a strukturu pro sečtení násobků, která produkuje výstupní vzorek  $y(n)$ . Potud je struktura navrženého filtru shodná s klasickou implementací vycházející z výše uvedené diferenční rovnice popisující činnost číslicového filtru s konečnou impulsní odezvou. Z obrázku 6.5 je však vidět, že celý filtr je rozdělen na dvě části. Levá část sečítá násobky prvních  $M$  vzorků ( $M < N$ ), pravá část pak zbylých  $N - M$  vzorků. Za normálních okolností filtr pracuje se všemi  $N$  předchozími vzorky. Pokud však nastane situace, že je třeba omezit spotřebu či energetické vyzařování filtru (například z důvodu klesající zásoby energie v baterii či zvyšující se teploty čipu), je možné omezit počet koeficientů filtru na  $M$  nezbytných. Pravá strana filtru se odpojí, filtr přejde do nouzového režimu. Ve schématu na obrázku 6.5 odpojení (přepínání mezi normálním a nouzovým režimem činnosti) zajišťuje signál  $c$ .

Samotné pomnutí posledních  $N - M$  koeficientů však nutně musí vést ke změně charakteristiky (odezvy) filtru, což není žádoucí. I v nouzovém režimu by filtr měl vykonávat takovou úlohu, jaká je potřeba, aby funkce celého systému byla ovlivněna co nejméně. Pro číslicové filtry však platí, že čím více koeficientů se uplatňuje, tím více se odezva číslicového filtru blíží ideální a naopak. K filtru s  $N$  koeficienty lze tedy zřejmě nalézt filtr s  $M$  koeficienty (kde  $M < N$ ) takový, že odezva tohoto filtru je tvarem podobná. Hodnotami koeficientů se však tyto dva filtry liší. Aby tedy byla zachována (alespoň přibližně) odezva filtru implementovaného dle struktury naznačené na obrázku 6.5 jak v normálním, tak i v nouzovém režimu, musí dojít při přepnutí režimu i ke změně koeficientů  $b_0 - b_{M-1}$  na koeficienty  $b_0^* - b_{M-1}^*$  (a zpět).

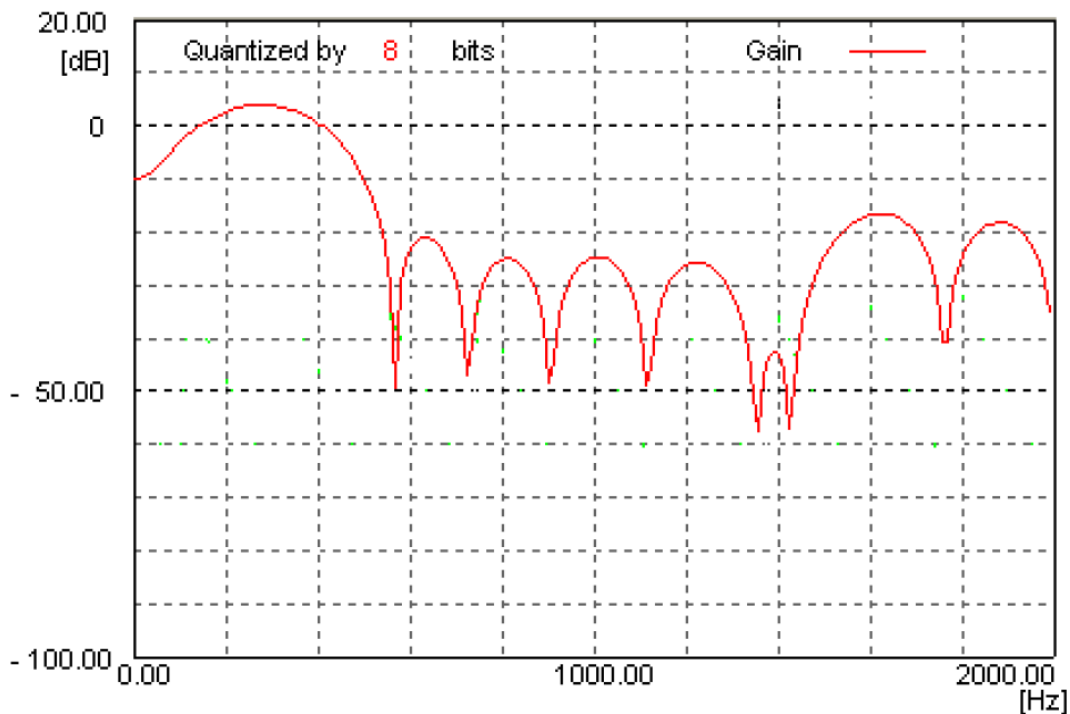
Za předpokladu, že uzly násobení konstantou jsou realizovány bez použití násobičky, tj. s využitím bitových posuvů a sčítání/odečítání, je třeba zajistit rekonfiguraci takových uzlů, protože obě konstanty jsou vlastně součástí struktury obvodu. Na obrázku 6.6 je příklad uzlu, který v jednom případě má provádět násobení konstantou 65 a ve druhém případě konstantou 49. Pravá větev obvodu zajišťuje násobení vstupní hodnoty  $x$  konstantou 65, které je realizováno jako bitový posuv o šest míst vlevo (odpovídá násobení 64) a přičtením ještě jednou  $x$ . Změna funkce na násobení konstantou 49 je realizována odečtením 16ti násobku  $x$  ( $65 - 16 = 49$ ). Má-li obvod násobit 65ti, odečítá se však nula. Změnu zajišťuje multiplexor ovládaný přepínacím signálem  $c$ .



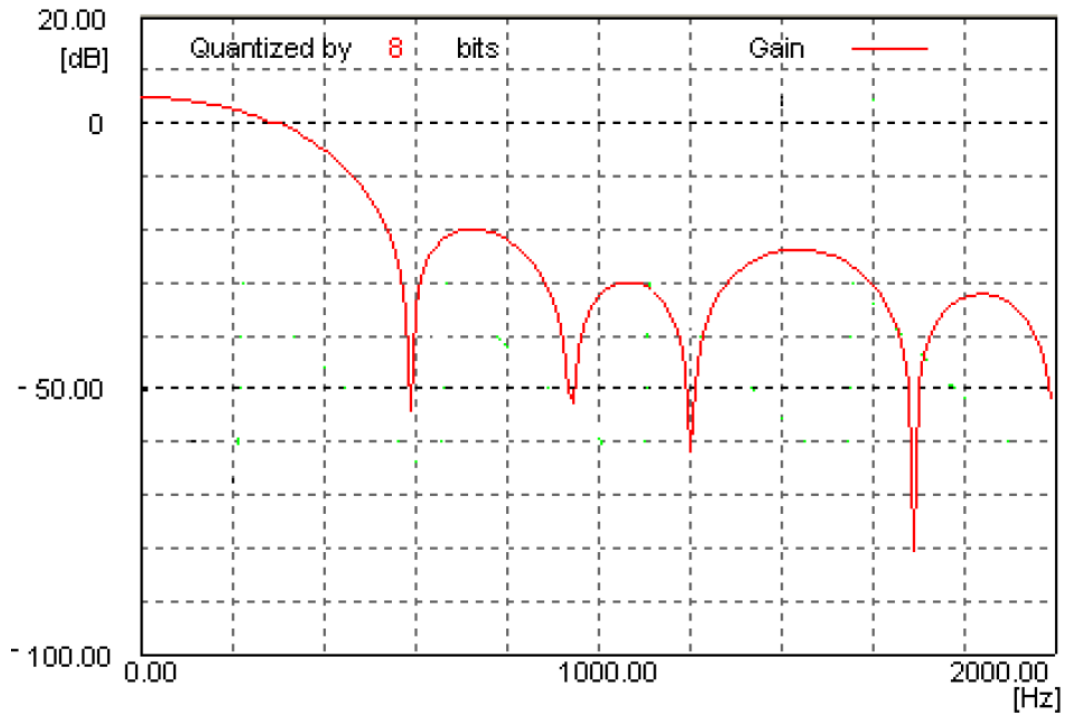
Obrázek 6.6: Obvod pro násobení konstantou 65 nebo 49.

Pro změnu režimu obvodu násobení konstantou lze s výhodou použít polymorfní elektroniky. Multiplexor na obrázku 6.6 může být realizován jako polymorfní (viz část 4.5 této práce), nebo může být celý obvod navržen jako polymorfní například některou metodou evolučního návrhu (třeba CGP, viz část 4.4 této práce). V případě využití polymorfních hradel řízených úrovní  $V_{dd}$  (nebo teplotou) signál ke změně funkce  $c$  nemusí být vůbec ve struktuře obvodu fyzicky implementován. Změna funkce obvodu může být realizována právě změnou funkce zapojených polymorfních hradel.

Jako příklad pro demonstraci výše popsaného přístupu k návrhu polymorfních FIR filtrů s nouzovým režimem činnosti se sníženou spotřebou byl zvolen filtr typu dolní propust, který by měl propouštět složky signálu s kmitočtem nižším než 500 Hz a pro složky s kmitočtem nad 500 Hz by měl mít podstatný útlum. Vzorkovací kmitočet signálu by byl 4 kHz a velikost vzorku 8 bitů. V normálním režimu by pracoval s 20ti osmibitovými koeficienty, v nouzovém režimu musí stačit 10 osmibitových koeficientů. Obrázek 6.7 ukazuje příklad odezvy navrženého filtru, který splňuje výše uvedenou specifikaci. Je vidět, že pro signály pod 500 Hz se útlum pohybuje pod 10 dB, naopak nad 500 Hz neklesá příliš pod 20 dB pro téměř celý možný frekvenční rozsah vstupního signálu (vzorkovací kmitočet 4000 Hz).



Obrázek 6.7: Odezva FIR filtru typu dolní propust pod 500 Hz s 20ti koeficienty.



Obrázek 6.8: Odezva FIR filtru typu dolní propust pod 500 Hz s 10ti koeficienty.

koef. $i$	konvenční FIR N = 20		konvenční FIR N = 10		polymorfní FIR, normální režim		polymorfní FIR, nouzový režim	
	$b_i$	hradel	$b_i$	hradel	$b_i/b_i^*$	hradel	$b_i^*$	hradel (poly)
0	00h	0	08h	0	00/08h	10	08h	10(2)
1	F7h	67	10h	0	F7/10h	67	10h	67(16)
2	F6h	64	18h	32	F6/18h	93	18h	93(28)
3	F0h	26	20h	0	F0/20h	27	20h	27(4)
4	F4h	26	22h	32	F4/22h	71	22h	71(21)
5	F8h	26	22h	32	F8/22h	72	22h	72(13)
6	02h	0	20h	0	02/20h	23	20h	23(8)
7	10h	0	18h	32	10/18h	41	18h	41(11)
8	18h	32	10h	0	18/10h	42	10h	42(11)
9	20h	0	08h	0	20/08h	24	08h	24(14)
10	20h	0			20h	0		
11	18h	32			18h	32		
12	10h	0			10h	0		
13	02h	0			02h	0		
14	F8h	26			F8h	26		
15	F4h	26			F4h	26		
16	F0h	26			F0h	26		
17	F6h	64			F6h	64		
18	F7h	67			F7h	67		
19	00h	0			00h	0		
násobení		482		128		711		470
10-vstup. sčítačka		666		333		666		333
výst. sčítačka		77				77		
výst. mux						32		32
registry		912		432		912		432
celkem		2137		893		2398		1267
%		100		42		112		59

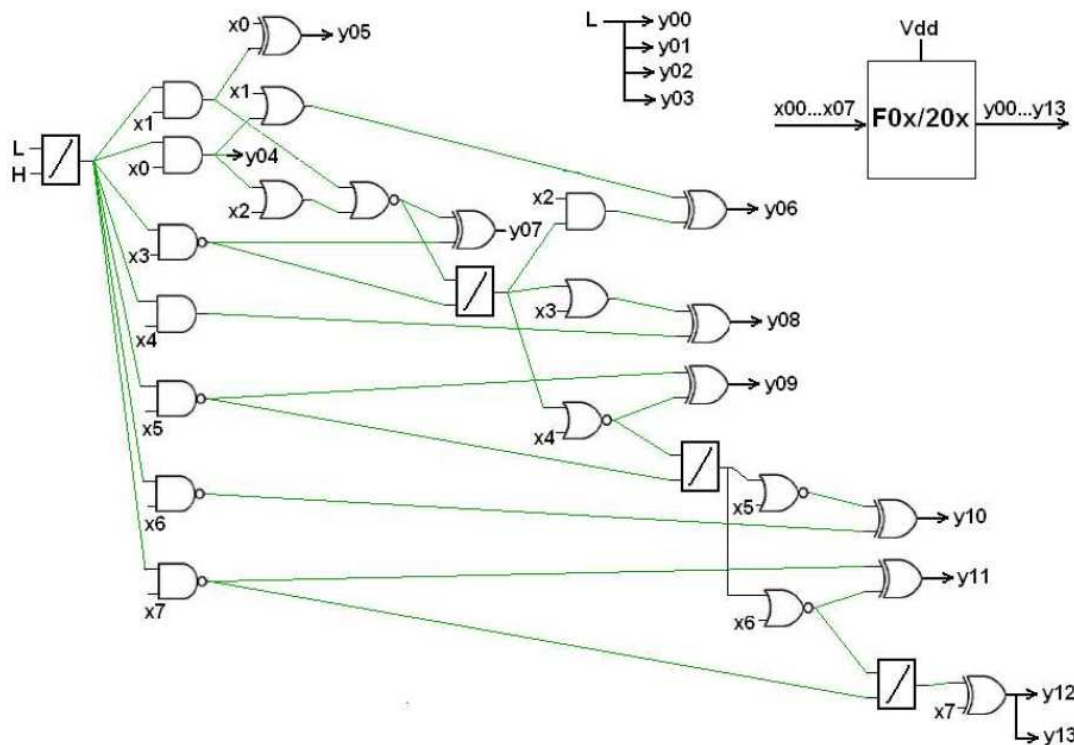
Tabulka 6.3: Koeficienty a počty hradel pro konvenční FIR filtry s 10ti a 20ti koeficienty a pro polymorfní FIR filtr v normálním a nouzovém režimu.

Na obrázku 6.8 je vidět odezva filtru se stejnými parametry, pouze počet koeficientů byl omezen na 10. Filtr má opět pro signály do 500 Hz malý útlum (ne více než 10 dB), pro kmitočty nad 550 Hz je útlum vždy větší než 20 dB.

Koeficienty pro filtry, jejichž odezvy ukazují obrázky 6.7 a 6.8, jsou vidět v tabulce 6.3. Ke každému koeficientu je zároveň uveden i počet hradel, která jsou potřeba pro implementaci obvodu realizujícího násobení tímto koeficientem. Je třeba poznamenat, že koeficienty, které jsou pouhou mocninou 2, nepotřebují žádné hradlo. Realizace násobení koeficientem, který je  $2^k$ , znamená pouhé propojení vodičů tak, aby se bity vstupní hodnoty  $x$  posunuly o  $k$  bitů vlevo. Je proto žádoucí hledat takové řešení filtru, kde by bylo co nejvíce koeficientů právě mocninou 2.

Řádek „násobení“ v tabulce je pak součtem hradel, která jsou potřeba pro realizaci násobení všemi koeficienty filtru. Tyto násobky se pak sečítají desetivstupovou sčítačkou (viz obrázek 6.5), pro jejíž realizaci je třeba 333 hradel. Filtr s 20ti koeficienty potřebuje takové sčítačky dvě (tedy 666 hradel) a navíc sčítačku výsledků těchto dvou desetivstupových sčítaček (77 hradel). Registry pro uchování 19ti předchozích vzorků potřebují k realizaci 912 hradel, celkem tedy filtr s 20ti koeficienty potřebuje 2137 hradel. Tento obvod je brán jako referenční pro porovnání velikosti polymorfního řešení, aby bylo možno zhruba odhadnout spotřebu. Poslední čtyři sloupce tabulky totiž obsahují vyčíslení počtu hradel, která jsou aktivní v normálním a nouzovém režimu, je-li filtr realizován jako polymorfní. K optimalizaci jednotlivých polymorfních obvodů realizujících násobení koeficienty bylo užito CGP.

Na obrázku 6.9 je uveden příklad realizace obvodu násobení konstantami F0h a 20h. Pro filtr popsaný v tabulce 6.3 je to obvod realizující násobení koeficienty  $b_3/b_3^*$ . Polymorfní hradla NAND/NOR jsou znázorněna obdélníkem. Symboly L a H označují trvalou úroveň log. 0 a log. 1.



Obrázek 6.9: Příklad realizace obvodu násobení konstantou F0h nebo 20h (koeficienty  $b_3/b_3^*$ ) s polymorfními hradly.

Poslední řádek tabulky ukazuje, že v normálním režimu pracující polymorfni filtr (počítá se plných 20 koeficientů) má aktivních 2398 hradel, což je o 12% více, než kdyby bylo použito konvenční řešení. Za možnost přejít do nouzového úsporného řešení se zachováním základní funkčnosti filtru se tak platí mírně zvýšenou spotřebou. Pokud však filtr přejde do nouzového režimu (kdy stále pracuje a filtruje signál!), je třeba napájet pouze 1267 hradel, což je 59% konvenčního řešení. Oproti normálnímu řešení pracuje dokonce jen necelých 53% hradel. Tato čísla dávají hrubou představu o úspoře energie (případně redukci tepelného vyzařování) filtru. Ve skutečnosti snížení spotřeby filtru závisí na použitých hradlech a poměru jejich spotřeby ke spotřebě konvenčních hradel. V kapitole 3 popsaná polymorfni hradla, řízená  $V_{dd}$  nebo teplotou, mají zpravidla spotřebu vyšší, takže úspora nebude tak výrazná. O hradlech řízených speciálním signálem to však obecně neplatí, jejich spotřeba může být srovnatelná se spotřebou konvenčních hradel. Na druhou stranu, budou-li použita hradla řízená  $V_{dd}$ , případně se snížení  $V_{dd}$  pojme jako další opatření k redukci spotřeby, povede snížení  $V_{dd}$  v nouzovém režimu ke snížení spotřeby. Podle rovnice

$$\frac{P_{3,3V}}{P_{5V}} = \frac{k \cdot q \cdot (3,3)^2}{k \cdot (5)^2} = 23\% ,$$

kde konstanta  $k$  závisí na dynamických parametrech obvodu a provozním kmitočtu (ideálně v obou režimech činnosti filtru stejné) a  $q$  je poměr hradel pro nouzový a normální režim činnosti (0,53, viz výše), bude spotřeba v ideálním případě (polymorfni hradla mají srovnatelnou spotřebu s konvenčními) v nouzovém režimu jen 23% spotřeby filtru v normálním režimu.

## 6.4 Obvody se skrytou funkcí

Jak již bylo mnohokrát řečeno, polymorfni elektronika má jako jednu z klíčových vlastností možnost realizace více funkcí jedním obvodem, přičemž ve struktuře polymorfniho obvodu typicky není možné identifikovat či oddělit struktury, které by realizovaly jednotlivé funkce. Této vlastnosti lze využít i v oblasti bezpečnosti, kdy zařízení, vykonávající svoji primární funkci, může díky užití principů polymorfni elektroniky při návrhu mít ještě další, skrytou funkci. Tato funkce může sloužit pro autentizaci uživatele (jak navrhuje například Stoica a další v [Sto01]) nebo může například sloužit ke spolehlivé identifikaci konkrétní instance obvodu.

Obvody, které obsahují v hardware funkci pro jednoznačnou identifikaci konkrétního čipu (instance obvodu), se nazývají obvody s fyzickou neklonovatelnou funkcí (Physical Unclonable Function, PUF). Tato technika, obdobná jednosměrné funkci v software, se v kryptografii používá již dlouho. Pojem „fyzická neklonovatelná funkce“ se objevuje v literatuře v roce 2002 [Pap02]. Fyzická neklonovatelná funkce musí být snadno vyrobitelná, ale prakticky neduplikovatelná. Na vstupní stimul reaguje nepredikovatelným způsobem díky tomu, že stimul složitým způsobem interaguje se strukturou čipu. Implementace kryptografických mechanismů založených na fyzických neklonovatelných funkcích předpokládá tyto tři základní vlastnosti fyzických neklonovatelných funkcí:

1. funkce dává různé výstupy na různých čípech,
2. při generování více výstupů na jednom čipu funkce generuje různé výstupy,
3. pro různé vstupy dává funkce různé výstupy.

Fyzická neklonovatelná funkce může být využita jako základní stavební blok nejrůznějších bezpečnostních mechanismů. Jedním z nich je hardwarový generátor náhodných čísel. Existuje řada známých řešení hardwarových generátorů náhodných čísel, mezi nejrozšířenější patří

využití kruhového oscilátoru [Mai09]. Je-li dostatečně splněna podmínka č. 2, pak je konstrukce generátoru náhodných čísel velmi jednoduchá. Výstup fyzické neklonovatelné funkce se vloží do hashovací funkce a výsledek je použitelný jako náhodné číslo.

Jiná možnost využití fyzické neklonovatelné funkce je při autentizaci odpovědí na výzvu (challenge-response authentication). Fyzická neklonovatelná funkce se používá jako aproximace ideální funkce typu náhodná věštba (Random Oracle Function), kdy na každou výzvu funkce odpovídá náhodnou odpovědí vybranou z její výstupní domény. Přitom je důležité, aby na stejnou výzvu funkce vždy odpověděla stejným způsobem. Autentizace je prováděna ve dvou krocích. V prvním, inicializačním kroku, je generováno několik párů výzva-odpověď a tyto páry jsou uloženy do databáze pro budoucí autentizace. Ve druhém, autentizačním kroku, je z databáze vybrána již dříve uložená výzva (ovšem taková, která ještě nikdy nebyla použita) a aplikována na čip. Jestliže odpověď je správná (nebo se jí dostatečně blíží s ohledem na to, že reálná implementace fyzické neklonovatelné funkce je pouze aproximací ideální funkce), čip je považován za autentický, protože z principu fyzické neklonovatelné funkce, jen autentický čip je schopen generovat autentický výstup. Z důvodu ochrany proti odposlechu se výzva, jednou použitá k autentizaci, již znovu nepoužije [Suh07].

Pro řadu aplikací se zvýšenými nároky na bezpečnost může být výhodné, aby každá instance (čip, případně uzel v síti) měla svůj unikátní klíč. Takový klíč může být užitečný třeba pro zabezpečení komunikace. Výstup fyzické neklonovatelné funkce nemůže být sice použit přímo pro tyto účely, protože díky vlivu teploty a dalším vlivům se může výstup fyzické neklonovatelné funkce pro stejný vstup drobně lišit, je ale možné použít například samoopravný kód [Suh07]. Postup sestává opět ze dvou kroků. V prvním (inicializačním) kroku je výstup fyzické neklonovatelné funkce kódován samoopravným kódem a je extrahován syndrom, který je trvale uložen (není třeba, aby byl utajen). Potom, když ve druhém kroku je fyzickou neklonovatelnou funkcí generován klíč, je výstup funkce korigován pomocí syndromu tak, aby výstup generoval správný (bezchybný) klíč.

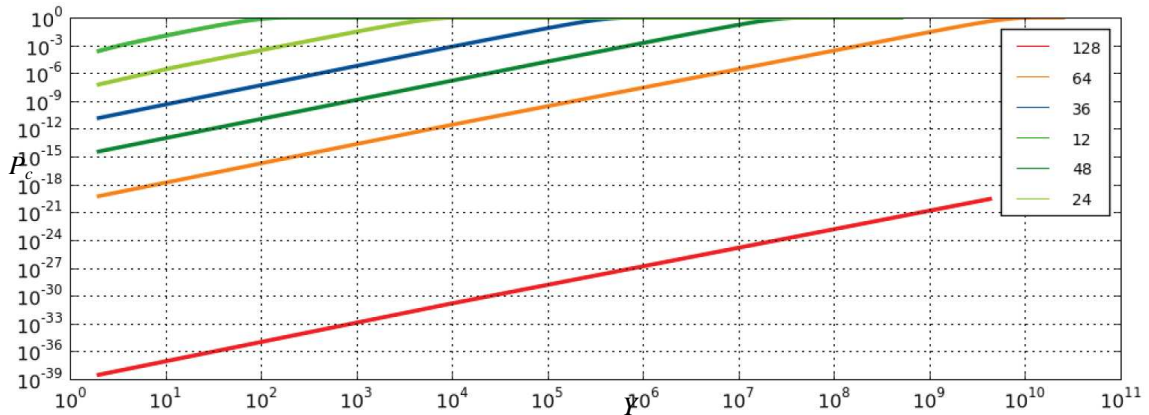
Fyzická neklonovatelná funkce může být také využita pro jednoznačnou identifikaci konkrétního čipu, konkrétního kusu, konkrétní instance obvodu. Získání unikátního identifikátoru konkrétního čipu je úloha velice podobná generování klíče pro kryptografické účely. I zde je třeba získat číslo, které je unikátní pro konkrétní čip a přitom nepodléhá okolním vlivům. Konvenční přístupy využívají unikátní kódy uložené do nevolatilní paměti, modifikaci části struktury čipu laserovým paprskem. Pozdější práce počítají s identifikací čipu podle drobných odchylek při výrobě [Lof00], případně sledováním odchylek parametrů tranzistorů na čipu [SuY08] či sledováním počátečních stavů v buňkách RAM na čipu a odchylkám ve zpoždění různých cest v obvodě [Gua07]. Tyto přístupy ovšem (na rozdíl od identifikátoru v nevolatilní paměti nebo modifikace laserem) generují náhodný vzorek identifikující čip. Na druhou stranu, vzhledem k tomu, že odchylky výrobního procesu jsou mimo možnosti ovlivnění při výrobě, je obtížné napodobit čip s přesně stejnými odchylkami.

Náhodný identifikátor čipu daný neovlivnitelnými variacemi parametrů namísto přidělení při výrobě s sebou přináší také možnost kolize identifikátorů u některých vyrobených čipů. Není vyloučeno, že se podaří vyrobit dva čipy s velmi blízkými odchylkami. Pravděpodobnost shody identifikátoru čipu délky  $X$  bitů mezi  $Y$  vyrobenými čipy lze popsat vztahem:

$$P_c = 1 - \prod_{n=1}^Y \left( 1 - \frac{n-1}{2^X} \right)$$

Na obrázku 6.10 je grafické znázornění výše uvedeného vztahu. Dává lepší představu o tom, jaké jsou ve skutečnosti hodnoty pravděpodobnosti pro konkrétní množství vyrobených čipů a

šířku slova identifikátoru čipu. Je vidět, že mezi 10 000 vyrobenými čipy se pravděpodobnost shody identifikátoru pohybuje pod  $10^{-10}$  pro již poměrně malý počet bitů identifikátoru – 64 bitů. Je-li identifikátor dlouhý 128 bitů, pravděpodobnost shody je již menší než  $10^{-30}$ .



Obrázek 6.10: Pravděpodobnost shody identifikátoru čipu vyrobeného v množství  $Y$  pro různé šířky slova.

Jak bylo zmíněno, fyzická neklonovatelná funkce může dávat na stejném čipu pro stejný vstup mírně odlišný výsledek díky teplotním a dalším vlivům na parametry tranzistorů a jiných struktur na čipu. Hammingova vzdálenost jednotlivých výstupních slov pro stejný vstup funkce na stejném čipu pak udává počet nestabilních bitů slova identifikátoru. V takovém případě lze použít například samoopravného kódu [Suh07]. Naproti tomu hammingova vzdálenost mezi dvěma identifikátory délky  $X$  získanými ze dvou různých čipů by měla být ideálně  $X/2$ . Pokud je o hodně menší, znamená to, že existuje příliš mnoho kolizí mezi identifikátory a že odchylky výrobního procesu, na nichž je postavena fyzická neklonovatelná funkce generující indikátor čipu, jsou nedostatečné pro realizaci skutečně efektivně fungující fyzické neklonovatelné funkce.

Myšlenka využít polymorfní elektroniky jako nekonvenční platformy pro realizaci fyzické neklonovatelné funkce vznikla na FIT VUT v Brně. Toto využití polymorfní elektroniky navrhl P. Hanáček a dále rozpracoval a experimentálně ověřil tým L. Sekaniny, jehož členem je i autor této práce. V současné době je návrh a výsledky experimentů v publikaci.

Při implementaci fyzické neklonovatelné funkce polymorfními obvody se využívá faktu, že polymorfní obvody a z nich složená polymorfní hradla mají v sobě vestavěnou citlivost na okolní prostředí, že okolní prostředí dokáže změnit jejich funkci (viz definice 2.1). Změna prostředí může být právě podnětem, vstupem fyzické neklonovatelné funkce. Na základě tohoto podnětu pak funkce generuje výstup, jednoznačný identifikátor obvodu. Není třeba dalších signálů ani nárůstu plochy čipu kvůli fyzické neklonovatelné funkci, tato je „funkcí v pozadí“, za normálních okolností i ty části obvodu, které jsou součástí implementace fyzické neklonovatelné funkce, jsou činné v běžné funkci obvodu. Dokonce návrh skupiny L. Sekaniny a P. Hanáčka umožňuje využívat obě (či více) funkcí polymorfního obvodu k běžné činnosti, fyzická neklonovatelná funkce je ještě něco „navíc“, pouze využívá některých vlastností polymorfní elektroniky, u konvenčních obvodů nepřítomných.

Principem je využití faktu, že odchylky výrobního procesu způsobí, že při změně prostředí nedojde k překlopení všech hradel v jednom okamžiku. Kritická hodnota fyzikální veličiny popisující prostředí (z definice 2.1), při které dojde ke změně funkce polymorfního hradla, se díky odchylkám ve výrobním procesu pro jednotlivá hradla v obvodě patrně mírně liší. Protože

prostředí jako fyzikální veličina se mění spojitě (byť jakkoli rychle), musí zákonitě dojít k tomu, že polymorfní hradla zapojená v jednom obvodě nezmění svoji funkci zcela současně. Pořadí přepnutí jednotlivých polymorfních hradel v obvodě při změně prostředí je zřejmě pro každý vyrobený čip unikátní a z něj proto může být odvozen identifikátor konkrétního čipu – výstup fyzické neklonovatelné funkce.

Myšlenka byla vyzkoušena s polymorfními rekonfigurovatelnými čipy REPOMO, jejichž struktura a vlastnosti jsou popsány v kapitole 5. Tyto čipy byly k experimentům zvoleny zejména proto, že jsou zatím jedinou dostupnou fyzickou realizací polymorfních obvodů. Protože však nebyly navrhovány s ohledem na realizaci fyzické neklonovatelné funkce (primárně byla struktura čipu optimalizována na použití s kartézským genetickým programováním a obvody jím generovanými), mají některá omezení, s nimiž je třeba v experimentech počítat:

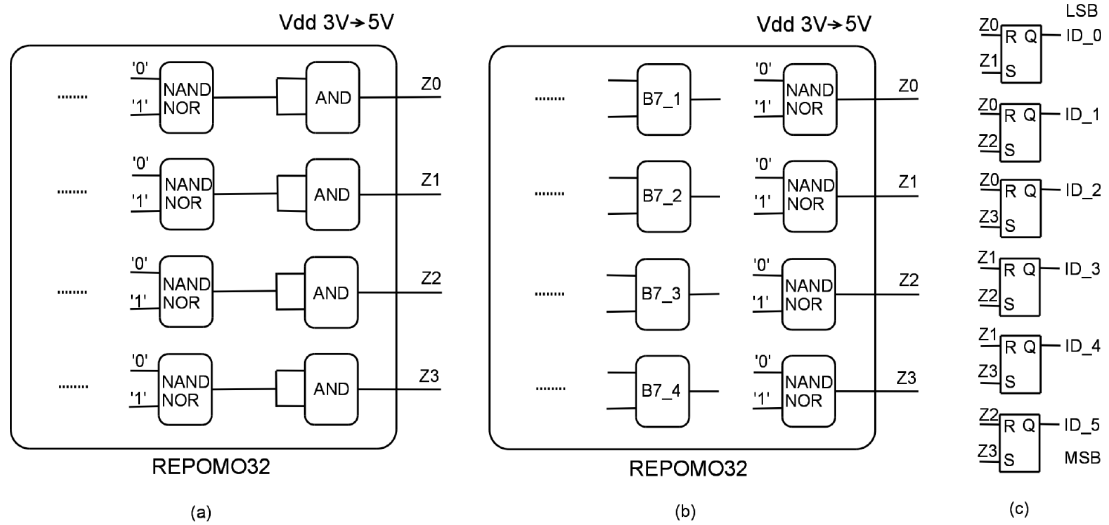
- omezený počet výstupů vede k vícekrokovému získávání identifikátoru čipu – je třeba realizovat fyzickou neklonovatelnou funkci s několika různými podněty (konfiguracemi obvodu a vstupy) pro získání vícebitové signatury,
- omezená hloubka propojení znamená, že některé elementy musí být konfigurovány jako propoje (funkce identity), aby bylo možno dostat vstupní stimuly k elementům uvnitř struktury obvodu a od nich odezvy na primární výstupy,
- REPOMO je čistě kombinační obvod, uvnitř nejsou datové registry ani záchytné klopné obvody – tyto je třeba realizovat vně obvodu.

Naštěstí je čip REPOMO navržen jako konfigurovatelný obvod, proto dopad všech uvedených nevýhod lze překonat postupnou rekonfigurací obvodu a získáváním identifikátoru čipu v několika krocích.

Obrázek 6.11 ukazuje způsob získávání identifikátoru čipu REPOMO. Výstupy polymorfních hradel jsou zapojeny na klopné obvody typu RS, které zachytí a svým stavem zobrazí ten ze vstupů, který změnil svoji úroveň dříve. Protože v jednom bloku hradel jsou 4 hradla (elementy – viz též obrázek 5.6) a obvod má 4 primární výstupy, porovnávají se v jednom kroku právě výstupy čtveřice hradel. Klopné obvody RS mají dva vstupy, porovnávají se tedy vždy dvě hradla mezi sebou. Podle toho, které z nich změnil funkci dříve (vstupy polymorfních hradel jsou nastaveny tak, aby změna funkce hradla způsobila změnu úrovně na jeho výstupu), zachytí příslušný klopný obvod buď log. 0 nebo log. 1. Jak je vidět na obrázku 6.11 c), k porovnání čtveřice hradel je třeba šest klopných obvodů. V jednom kroku lze získat šest bitů identifikátoru čipu. Obecně je počet bitů, které lze získat v jednom kroku, roven  $n(n-1)/2$ , kde  $n$  je počet hradel, jejichž výstupy lze současně srovnávat.

Obrázky 6.11 a) a b) znázorňují, že pro získání další šestice bitů identifikátoru je třeba obvod REPOMO rekonfigurovat, aby klopné obvody na primárních výstupech zachytily rozdíly v okamžiku překlápění dalšího sloupce hradel. Klopné obvody byly implementovány vně obvodu REPOMO, bylo použito obvodu typu CPLD, který je na kitu REPOMO32/kit (viz část 5.3 této práce a obrázek 5.8). Pro získání jedné šestice bitů identifikátoru musí být vždy jeden sloupec hradel konfigurován jako polymorfní hradla, elementy v ostatních sloupcích by měly být konfigurovány pro funkci identity. Toho se může docílit například tím, že elementy se nakonfiguruje na funkci AND a jejich vstupy se propojí, jako je to vidět na obrázku 6.11 a). Postupným testováním jednotlivých sloupců hradel v obvodě REPOMO je možné pro každý čip získat 8x6, tedy 48 bitů identifikátoru.





Obrázek 6.11: Získávání identifikátoru čipu REPOMO porovnáním okamžiků změny funkce polymorfních hradel při změně  $V_{dd}$ .

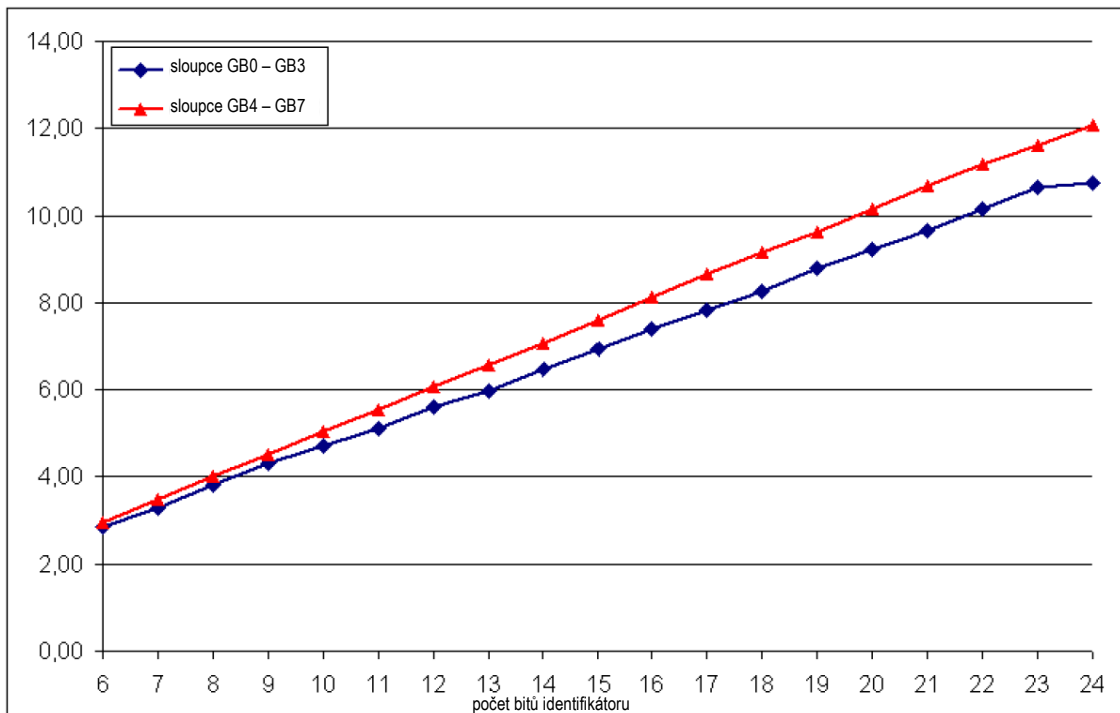
čip / sloupec	GB0	GB1	GB2	GB3	GB4	GB5	GB6	GB7
A	35	0F	25	01	07	3F	27	1E
B	3F	27	25	35	30	1A	21	3F
C	27	27	38	37	38	3F	3C	37
D	0B	27	21	25	38	27	08	1F
E	27	34	35	27	0A	1E	25	3E
F	0A	3E	07	07	18	03	3E	20
G	21	20	21	07	3C	27	1A	18
H	1F	38	35	3F	34	0F	25	0A
I	34	21	1F	0B	27	18	30	0B
J	0B	00	25	03	34	30	0B	1F
K	1E	37	27	00	35	3C	20	1A
L	3E	3F	25	1F	03	01	1F	01
M	27	34	3E	27	37	08	35	0B
N	27	03	08	0F	00	18	3F	37
O	00	37	18	21	3C	35	21	1E
P	35	3F	0F	03	0F	3F	1A	3F
Q	20	3F	3F	21	21	37	0B	01
R	3F	3E	34	21	21	0B	07	07
S	0F	1F	21	3F	34	0A	20	21
T	3F	07	20	21	35	30	3C	07
U	0B	1F	25	1F	1F	25	03	01
% nestabilních bitů	12,68	17,94	11,64	15,33	7,72	9,41	8,40	5,56

Tabulka 6.4: Šestibitová identifikace pro 8 sloupců (GB0 – 7) 21 testovaných čipů (A – U).

Při experimentech bylo prověřeno a srovnáno celkem 21 kusů čipů REPOMO. Identifikátory získané pro jednotlivé čipy a jejich jednotlivé sloupce hradel jsou přehledně vidět v tabulce 6.4. Měření bylo provedeno 1000x pro každý čip a byly vyhodnoceny odchylky v šestibitových částech identifikátoru pro každý sloupec. Procento nestabilních bitů je uvedeno v posledním řádku tabulky 6.4. Rozdíly procenta nestabilních bitů pro jednotlivé sloupce jsou patrně způsobeny zejména tím, že se pro různou vzdálenost hradel od primárních výstupů různě projevuje zpoždění cesty mezi hradlem a výstupem. Pro celý 48 bitový identifikátor čipu je průměrně 11% bitů nestabilních. Jak bylo uvedeno výše, vliv nestabilních bitů identifikátoru lze potlačit například použitím samoopravného kódu.

Průměrná hammingova vzdálenost mezi identifikátory libovolných dvou různých čipů z testované sady je 22,83 s rozptylem  $\pm 4,11$ , což se blíží ideální teoretické hodnotě 24. Poměr  $\log_2 1$  a  $\log_2 0$  v identifikátorech je 1,25.

Pozoruhodné je, že části identifikátoru vzešlé ze sloupců hradel GB4 – GB7 jsou o něco spolehlivější než ty ze sloupců GB0 – GB3 (viz poslední řádek tabulky 6.4). Obrázek 6.12 ukazuje rozdíl v hammingově vzdálenosti pro první a druhou čtveřici sloupců pro různé velikosti identifikátoru. Jak již bylo naznačeno, je to patrně způsobeno rozdílnou délkou cesty k primárním výstupům. Vlivy zpoždění a jeho proměnlivost vlivem různých faktorů je u vzdálenějších hradel větší. U návrhů obvodů, kde by se s implementací fyzické neklonovatelné funkce jako prostředku identifikace čipu počítalo od počátku, by bylo proto vhodné umístit RS klopné obvody pro zachycení bitů identifikátoru co nejbližší k výstupům polymorfních hradel. Tím by se také podstatně zkrátila doba identifikace, protože by nebylo třeba obvod několikrát rekonfigurovat. Provedené experimenty ukazují, že procento nestabilních bitů identifikátoru by mohlo být okolo 5%. Právě takové hodnoty dosahuje procento nestabilních bitů u posledního sloupce hradel, jejichž výstupy jsou bezprostředně připojeny na primární výstupy obvodu. Současné implementace identifikace čipu fyzickou neklonovatelnou funkcí v ASIC nebo FPGA vykazují procento nestabilních bitů okolo 4% [SuY08], což je srovnatelné.



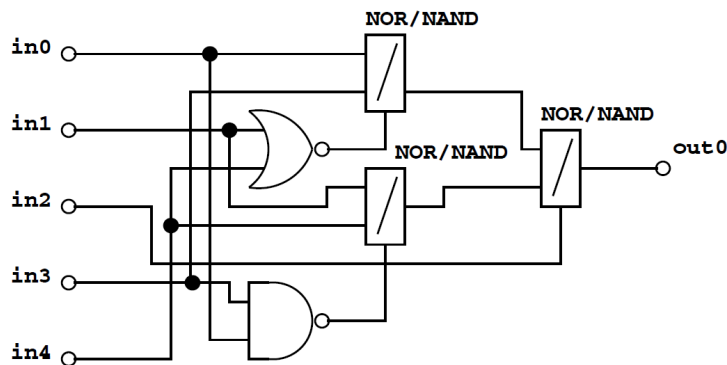
Obrázek 6.12: Průměrná hammingova vzdálenost identifikátorů různé délky pro sloupce GB0 – GB3 a GB4 – GB7.

Obvod REPOMO nebyl navrhován pro identifikaci pomocí fyzické neklonovatelné funkce, přesto však i experimenty s ním ukazují, že myšlenka jednoznačné identifikace čipu založené na fyzické neklonovatelné funkci odvozené od pořadí překlápění funkce jednotlivých polymorfních hradel je perspektivní.

Z naměřených hodnot hammingovy vzdálenosti identifikátorů různých čipů (obrázek 6.12) plyne, že pro jednoznačnou a spolehlivou identifikaci čipu REPOMO z měřené sady 21 kusů by postačoval desetibitový identifikátor.

## 6.5 Využití polymorfních hradel v návrzích jednofunkčních obvodů

Polymorfní hradla, zejména polymorfní hradla řízená speciálním signálem, která realizují dvě různé funkce (a lze je tedy vlastně chápat jako realizaci složitější funkce s  $n+1$  proměnnou, kde  $n$  je počet datových vstupů hradla), je možné použít také jako konvenční hradla v návrhu konvenčních kombinačních obvodů. Výhodou tohoto přístupu je úspornější implementace, nevýhodou je, že konvenční návrhové metody s atypickými stavebními bloky (funkcemi) pracovat nedokážou. Návrháři jsou proto zatím odkázáni na evoluční techniky návrhu. Na druhou stranu, pro evoluční návrh složitějších obvodů je výhodné mít k dispozici komplexnější bloky. Správný výběr základních stavebních bloků pro evoluci (druhu i velikosti/složitosti) je pro úspěch evolučního návrhu zásadní.



Obrázek 6.13: Obvod majority pěti proměnných realizovaný s využitím polymorfních hradel.

Příkladem obvodu, u něhož využití polymorfních hradel při jeho návrhu vede na efektivní implementaci, i když se jedná o obvod s jedinou funkcí, je obvod realizující funkci majorita pro pět vstupních proměnných. Schéma obvodu je na obrázku 6.13 [Gaj07]. Obvod byl navržen technikou kartézského genetického programování (viz část 4.3 této práce). Byla povolena libovolná hloubka propojení, populace měla (1+15) jedinců. Poměrně složitá fitness funkce v tomto případě kombinovala (v uvedeném pořadí významnosti) ohodnocení funkčnosti, počtu tranzistorů, počtu hradel, zpoždění. Evoluční návrh byl opakován stokrát a produkován vždy milión generací. Nejdříve byla aplikována mutace na tři geny v chromozómu, po dosažení 100% funkce obvodu byl počet mutací zvýšen na sedm. V práci [Gaj07] provedli autoři dokonce sadu experimentů s různými množinami bloků (hradel), aby prokázali, že když má evoluce k dispozici i složitější bloky (jako například polymorfní hradla NAND/NOR nebo NOR/NAND), může produkovat v určitých případech daleko lepší řešení, než kdyby byly k dispozici jen základní hradla.

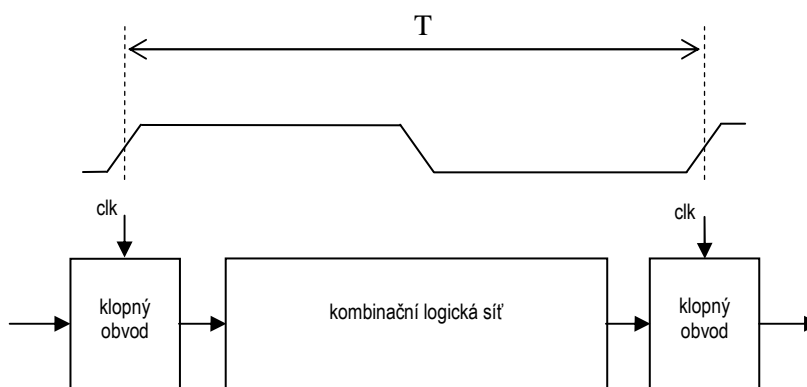
Z obrázku 6.13 je patrné, že nalezené řešení potřebuje k implementaci pouze 32 tranzistorů (1x 4 tranzistory pro hradlo NAND, 1x 4 tranzistory pro hradlo NOR a 3x 8 tranzistorů pro hradlo NOR/NAND). Nejlepší známé řešení přitom potřebuje 38 tranzistorů [Qui01]. Zajímavé je, že evoluční návrh se stejnými parametry, pouze s množinou hradel ochuzenou o polymorfní hradla (evoluce mohla uplatnit dvouvstupová hradla AND, OR, XOR, NAND, NOR, třívstupová AND, OR, XOR, NAND, NOR, invertor a multiplexor) našel řešení s cenou 40 tranzistorů.

## 7 Sekvenční polymorfní obvody

### 7.1 Sekvenční obvody

Dosud se tato práce zabývala polymorfními obvody, které lze zařadit do kategorie tzv. kombinačních číslicových obvodů. To jsou obvody, u kterých hodnota výstupu obvodu závisí pouze na kombinaci hodnot, které jsou aktuálně na vstupu obvodu. Sekvenčními obvody se pak nazývají takové číslicové obvody, u nichž hodnota výstupu nezávisí pouze na aktuálním stavu vstupů, ale také na tom, s jakými hodnotami obvod pracoval dříve. Lze také hovořit o tzv. stavu obvodu, obvod má jistý druh paměti.

Při návrhu sekvenčních číslicových obvodů hrají roli paměťové elementy, které se také nazývají klopné obvody. Klopné obvody nemusí přitom v sekvenčních obvodech sloužit nutně jako paměť. Mohou sloužit jako prvky, které rozdělují obvod na části, pracující v čase postupně, jako prvky, které rozlišují jednoznačně v určitých bodech obvodu minulou hodnotu od současné a budoucí [Wes05].

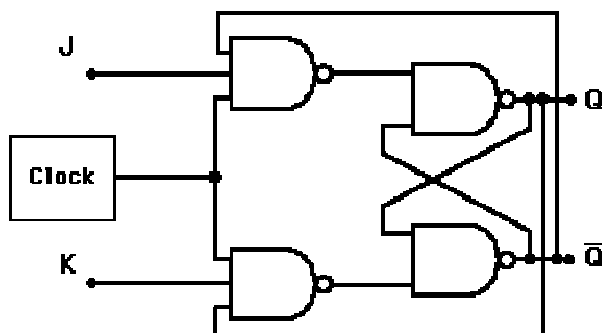


Obrázek 7.1: Sekvenční číslicový obvod.

Na obrázku 7.1 je vidět, jak je v sekvenčním číslicovém obvodu kombinační část proložena klopnými obvody, jak je obvod rozdělen na jednotlivé úseky. Hodnota vstupu kombinační logické sítě je zachycena z předchozího kroku v klopném obvodu, hodnota výstupu kombinační logické sítě se zachycuje do klopného obvodu pro aplikaci do následné logické sítě v dalším kroku. Celý děj je řízen hodinovým signálem, za jednu periodu hodinového signálu musí dojít k propagaci datového signálu ze vstupu na výstup kombinační sítě, aby každý klopný obvod při příslušné hraně hodinového signálu, která provádí zápis informace do klopného obvodu z jeho vstupu, měl na svém vstupu platný výsledek z předcházející kombinační sítě.

Klopné obvody je možné realizovat podobně jako hradla na úrovni tranzistorů, nebo též modelovat na úrovni hradel. Podle uspořádání vstupů a jejich reakce na ně existuje řada různých typů klopných obvodů. U některých se synchronizace provádí speciálním vstupem hodinového signálu, jiné jsou asynchronní. Na obrázku 7.2 je příklad řešení klopného obvodu typu JK sestaveného z hradel typu NAND. Dvojice hradel vpravo tvoří vlastní klopný obvod. Je vidět, že stav klopného obvodu (hodnota, na kterou je nastaven) je v něm (na jeho výstupu) udržována pomocí zpětné vazby. Jedná se o klasický asynchronní klopný obvod typu RS (reset-set), který lze překlápat nastavením log. 0 na jeden ze vstupů. Hradla vlevo slouží pro synchronizaci zápisu hodnoty do klopného obvodu pomocí hodinového signálu. Zajistí, že překlápání klopného obvodu není možné pouze změnami na vstupech J a K, ale že je třeba ještě hodinový

signál, konkrétně jeho vzestupnou hranu. Právě při vzestupné hraně hodinového signálu (signál „Clock“ na obrázku 7.2) se překlopí klopný obvod podle toho, jaké hodnoty jsou zrovna na vstupech J a K.



Obrázek 7.2: Klopný obvod typu JK z hradel NAND.

Chování klopného obvodu, stejně jako třeba chování hradla, je možné popsat například též pravdivostní tabulkou. Pravdivostní tabulka však neobsahuje pouze hodnoty log. 0 a 1, jako je tomu u kombinačních obvodů, ale musí odrážet vliv stavu obvodu od minulé periody hodinového signálu. Pravdivostní tabulku pro klopný obvod typu JK ukazuje tabulka 7.1.

J	K	clk	$Q_{n+1}$
0	0	↑	$Q_n$
0	1	↑	0
1	0	↑	1
1	1	↑	$\overline{Q_n}$

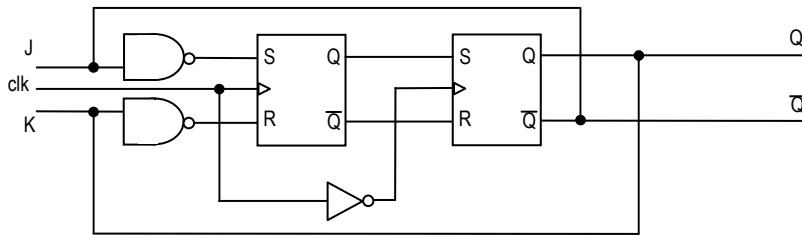
Tabulka 7.1: Pravdivostní tabulka klopného obvodu JK.

Pravdivostní tabulka 7.1 popisuje přesně chování klopného obvodu typu JK. Symbol ↑ značí okamžik vzestupné hrany signálu. Jindy, nežli v tomto okamžiku se stav (hodnota výstupu) klopného obvodu nezmění. Z tabulky je zřejmé, že stav klopného obvodu se nezmění, ani když nastane vzestupná hrana a na vstupech J a K jsou log. 0. To je vyjádřeno symbolem  $Q_n$ , který reprezentuje stav před příchodem hrany hodinového pulsu. Symbolem  $Q_{n+1}$  je vyjádřen stav klopného obvodu po příchodu hodinového pulsu. Pro  $J = K = 0$  tedy platí, že  $Q_{n+1} = Q_n$ , klopný obvod se nepřeklopí.  $J = K = 1$  naopak způsobí překlopení klopného obvodu, úroveň výstupu se změní, ať byla předtím 0 nebo 1.

## 7.2 Polymorfní klopné obvody

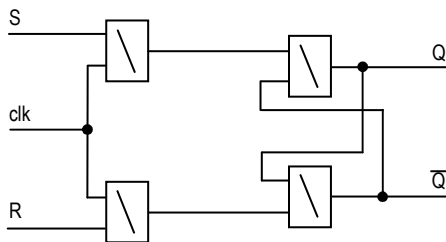
Tým A. Stoicy navrhl a v roce 2006 publikoval v technické zprávě NASA [Zeb06b] polymorfní klopné obvody typu JK, které byly sestaveny z kombinace konvenčních a polymorfních hradel. Čítač sestavený z takových klopných obvodů může měnit funkci podle stavu prostředí.

Stoica navrhuje klopný obvod JK sestavený ze dvou klopných obvodů RS v zapojení master-slave (inverter v cestě hodinového signálu mezi prvním a druhým klopným obvodem RS způsobí, že druhý klopný obvod reaguje na opačnou hranu než první) – viz obrázek 7.3.



Obrázek 7.3: Polymorfní klopný obvod typu JK.

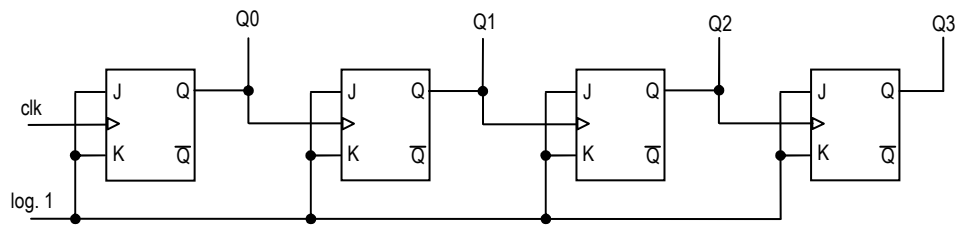
Klopné obvody typu RS, použité v klopném obvodu typu JK z obrázku 7.3, jsou sestaveny výhradně z polymorfních hradel. Jejich struktura je vidět na obrázku 7.4. Použitá hradla jsou podle [Zeb06b] typu NAND/NOR/AND. Struktura navrženého klopného obvodu je velmi podobná klasickému zapojení z obrázku 7.2. Pokud jsou použita hradla s funkcí NAND, klopný obvod je překlápěn do stavu log. 1 (na výstupu Q) úrovní log. 1 na vstupu S a do stavu log. 0 úrovní log. 1 na vstupu R. Je-li stejná struktura sestavena s využitím hradel NOR, funkce vstupů se obrací a obrací se také vstupní logické úrovně. Protože tyto klopné obvody jsou použity v klopném obvodu z obrázku 7.3, je otázkou, zda je označení „polymorfní JK klopný obvod“, které používá Stoica, korektní.



Obrázek 7.4: Implementace polymorfního klopného obvodu RS z polymorfních hradel NAND/NOR/AND.

Jako možná aplikace takto navrženého polymorfního klopného obvodu se v [Zeb06b] uvádí asynchronní čítač, který podle režimu polymorfních hradel použitých v klopných obvodech může buď čítat vzestupně (hradla v režimu NAND) nebo čítat sestupně (hradla v režimu NOR), případně se vynulovat (hradla v režimu AND). Čítač, v němž lze polymorfní klopné obvody použít, je vidět na obrázku 7.5. Naznačené řešení čítače sestává ze čtyř klopných obvodů, jde o čtyřbitový čítač. Jak je však z obrázku 7.5 zřejmé, jde o naprosto standardní zapojení asynchronního binárního čítače a je triviální rozšířit čítač přidáním dalších klopných obvodů na libovolný počet bitů.

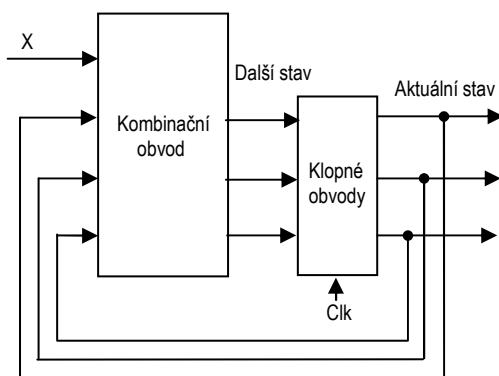
Ovládání čítače závisí na použitých polymorfních hradlech. Autoři zamýšleli svůj návrh jako optimalizované řešení standardního obousměrného binárního čítače s nulováním, který potřebuje jeden další logický signál pro určení směru čítání a druhý pro nulování. Pokud by použitá hradla byla řízena speciálním signálem (viz část 3.5.3 této práce), pak by k ovládání čítače místo dvou logických signálů postačil jeden analogový. To by ušetřilo jeden spoj (ovšem rozváděný ke všem hradlům). Hlavní úspora proti konvenčnímu řešení se však skrývá v přídatné logice, kterou by se musely ovládací logické signály u každého klopného obvodu zpracovávat. Autoři hovoří o úspoře tří hradel u každého klopného obvodu.



Obrázek 7.5: Asynchronní čítač s polymorfními klopnými obvody typu JK.

### 7.3 Polymorfní čítače a řadiče

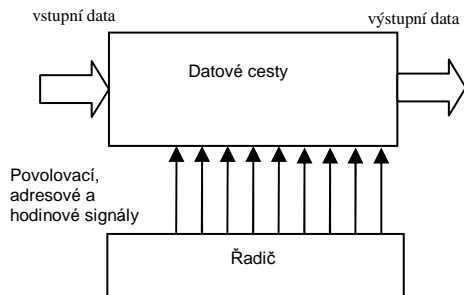
Polymorfní sekvenční číslicové obvody, které jsou sestaveny z polymorfních klopných obvodů, jak je navrhuje tým A. Stoicy a jak jsou popsány v předchozí části této kapitoly, mají kromě výše zmíněných výhod také svoje nevýhody. Polymorfní klopné obvody bude náročné navrhovat. Patrně je bude možné navrhnout technikami evolučního návrhu, protože jde o obvody poměrně malé. Jejich využití ve složitějších návrzích ale i tak nebude jednoduché. Jak bylo popsáno v části 4.4, evoluční techniky návrhu mají problematickou škálovatelnost. Není známo, jak tým A. Stoicy dospěl k řešení klopného obvodu, který je na obrázcích 7.3 a 7.4 a jak dospěl k návrhu čítače na obrázku 7.5, který navržené klopné obvody velmi elegantně využívá. Konvenční techniky návrhu pro sekvenční polymorfní obvody dosud nejsou známy. Autor této práce proto navrhl pro sekvenční polymorfní obvody opačný postup, než zvolil tým A. Stoicy. Návrh spočívá ve využití konvenčních klopných obvodů, které jsou doplněny polymorfní kombinační logikou. Techniky syntézy polymorfních kombinačních obvodů jsou již známy (viz kapitola 4 této práce), pro návrh sekvenčního obvodu s polymorfní logikou lze pak použít konvenční techniky návrhu sekvenčních obvodů. Použitelnost tohoto postupu je dále demonstrována na příkladech a je též popsána možná aplikace sekvenčních polymorfních obvodů v oblasti systémů odolných proti poruchám [Ruz09], [Ruz10a], [Ruz11b], [Ruz07c].



Obrázek 7.6: Struktura sekvenčního obvodu (čítače nebo řadiče).

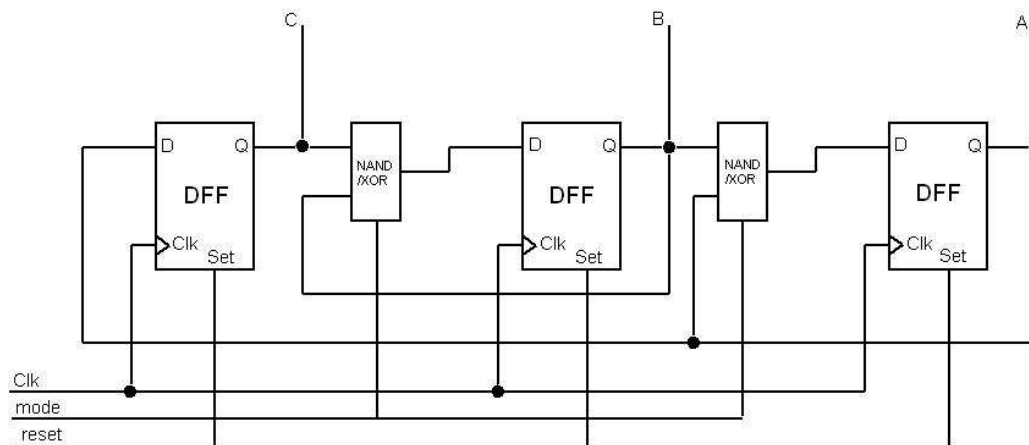
Na obrázku 7.6 je vidět struktura typického sekvenčního obvodu, který může být použit jako čítač nebo obvodový řadič. Klopné obvody jsou zde zpravidla jednoduché obvody typu D, slouží pouze jako paměť aktuálního stavu. Kombinační síť, znázorněná na obrázku 7.6 vlevo, pak pro každý aktuální stav (na vstupu) dokáže odvodit stav, který má následovat. Chování kombinačního obvodu pro výpočet dalšího stavu lze popsat tabulkou. Tímto způsobem lze implementovat například libovolný čítač v libovolném kódu s libovolným počtem stavů (počet stavů nemusí být nutně roven  $2^n$ , kde  $n$  je počet bitů stavu, může být i nižší).

Má-li obvod být implementací obecného stavového automatu (například Mealyho), tedy stroje, který má i datový vstup, podle něhož se (také) rozhoduje o dalším stavu, musí mít kombinační síť pro výpočet následujícího stavu ještě další vstup. Ten je označen na obrázku 7.6 jako X. Obvod lze pak aplikovat například jako řadič sekvenčně řídicí nějaký složitější číslicový obvod. Pro jeho datovou část například může generovat řídicí a povolovací signály (signály zápisu do registrů – datová část bude mít strukturu v takovém stylu, jako je vidět na obrázku 7.1) a tím řídit tok dat obvodem podle daného algoritmu.



Obrázek 7.7: Kombinační číslicový obvod s řadičem.

Je-li kombinační logická síť realizována jako polymorfní obvod, pak prostředí, ovlivňující polymorfní hradla v ní uplatněná, lze chápat jako svého druhu další vstup typu vstupu X z obrázku 7.6 (viz též poznámka 2.1 k definici 2.1).

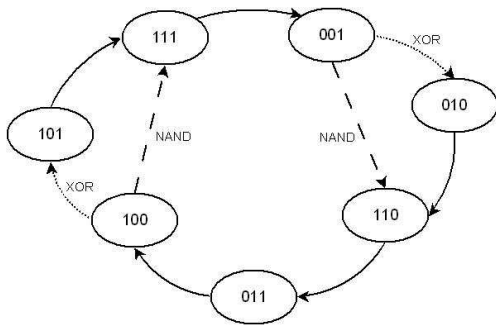


Obrázek 7.8: Příklad synchronního čítače s polymorfními hradly.

Na obrázku 7.8 je příklad tříbitového synchronního čítače, kde jsou v kombinační části, generující hodnoty následujícího stavu, použita polymorfní hradla NAND/XOR, řízená speciálním signálem (konkrétně hradlo NAND/XOR navržené autorem práce – podrobnosti viz část 3.5.3 této práce). Je třeba poznamenat, že pokud by bylo použito například hradlo řízené úrovní  $V_{dd}$ , odpadl by signál výběru režimu čítače „mode“ a jeho rozvod.

Čítač na obrázku 7.8 se chová tak, že pokud jsou polymorfní hradla v režimu XOR ( $mode = 0$ ), čítač čítá v celkem sedmi stavech. Pokud jsou hradla v režimu NAND ( $mode = 1$ ), čítač čítá jen v pěti stavech z původních sedmi. Přehledně to ukazuje obrázek 7.9.





Obrázek 7.9: Diagram přechodů obvodu z obrázku 7.8.

Obrázek 7.9 ukazuje diagram přechodů polymorfního čítače s hradly NAND/XOR. Je vidět, že v režimu hradel XOR (mode = 0) se postupně s hodinovým signálem obvod dostává do stavů 001, 010, 110, 011, 100, 101, 111, zatímco v režimu hradel NAND (mode = 1) se stavy 010 a 101 vynechávají. Rozhodování se děje ve stavu 001: v režimu 0 se pokračuje na 010 (tečkovaná hrana), zatímco v režimu 1 se čítač překlápí rovnou do stavu 110 (čárkovaná hrana) a také ve stavu 100: v režimu 0 se pokračuje na 101 (tečkovaná hrana), zatímco v režimu 1 se přeskočí rovnou do stavu 111 (čárkovaná hrana). V ostatních stavech čítače režim polymorfních hradel nemá význam, následující stav je vždy stejný, protože výstup polymorfních hradel je pro dané vstupní kombinace shodný v obou režimech hradel.

Polymorfní hradla NAND/XOR jsou, jak již bylo naznačeno v části 3.5.3 této práce, pro tento typ úlohy vhodná, protože pravdivostní tabulka funkcí NAND a XOR se liší pouze v jednom řádku, rozdíly v chování čítače v obou režimech jsou proto jen mírné.

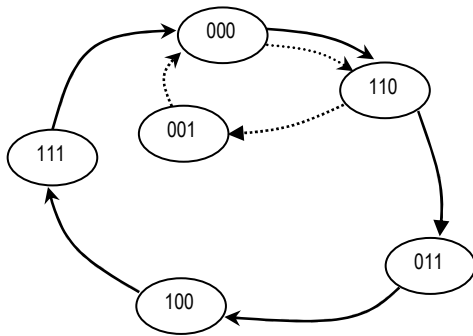
## 7.4 Aplikace polymorfních čítačů a řadičů

Autor této práce navrhl a experimentálně ověřil aplikaci polymorfních čítačů/řadičů, jejichž koncept byl naznačen v předchozí části. Aplikace využívá polymorfismu obvodů k elegantní adaptaci systému na zhoršené podmínky, jako je například pokles napájecího napětí vlivem úbytku energie v akumulátoru nebo zvýšení teploty čipu v důsledku růstu okolní teploty a tím zhoršené možnosti chlazení. V takových případech dojde k samočinné „rekonfiguraci“ (u polymorfních obvodů nelze mluvit o rekonfiguraci v pravém slova smyslu, ve skutečnosti jde o vhodně zvolenou změnu funkcí komponent systému) tak, aby systém snížil svoji spotřebu (tepelné vyzařování) omezením náročných funkcí a zachováním funkcí jen opravdu nezbytných k přežití systému do doby, než se vnější podmínky upraví a systém bude moci opět (díky polymorfním obvodům zcela samočinně) obnovit svoji funkci. Přitom cena takové aplikace bude ve srovnání s konvenčním řešením díky vlastnostem polymorfní elektroniky (viz též kapitola 2 této práce) s největší pravděpodobností výrazně nižší. Pro konvenční řešení by patrně bylo třeba navrhnout dva moduly pro běžnou a nouzovou funkci a k nim přepínač řízený senzorem nebo obvod řešit jako rekonfigurovatelný, kde režie rekonfigurace (paměť konfigurací, řídicí obvody pro změnu konfigurace, konfigurovatelný obvod s propojovací či přepínací sítí a konfigurovatelnými elementy) představuje také nezanedbatelný nárůst plochy čipu a s tím související nárůst ceny a snížení spolehlivosti. Systémy, které se dokážou adaptovat na vnější podmínky a zachovat svoji činnost i v případě nepřízně okolního prostředí a přežít do doby, kdy se podmínky opět zlepší, najdou uplatnění například v kosmických aplikacích nebo všude tam, kde se předpokládá autonomní činnost zařízení bez obsluhy člověkem.

Jádrem aplikace je polymorfní řadič (viz část 7.3) navržený tak, aby v jednom režimu procházel všemi stavy (inicioval všechny činnosti, které je třeba, aby systém vykonával), zatímco

v druhém (nouzovém) režimu procházel jen některými stavy (inicioval jen činnosti naprosto nezbytné pro udržení chodu systému a funkce, které není možné oželeť), případně vykonával jiné stavy, v nichž se provádějí činnosti náhradní za ty, které byly vypuštěny. Ovšem takové, které jsou omezené na minimum, jsou méně energeticky náročné než standardní činnosti.

Na obrázku 7.10 je stavový diagram jednoduchého řadiče obvodu s pěti základními stavy (000, 110, 011, 100 a 111). Těchto pět základních stavů je třeba, aby obvod řízený řadičem vykonával svoji běžnou činnost. V případě nouze – například při poklesu napětí v baterii, což je příznakem, že zásoba v energii dochází – lze omezit činnost obvodu tak, že budou prováděny jen činnosti realizované ve dvou z těchto pěti stavů (v obrázku 7.10 jsou to stavy 000 a 110) a činnosti realizované ve zbývajících třech stavech se nahradí jedním novým náhradním stavem 001 (s omezeným rozsahem tak, aby systém úplně nepřestal pracovat, ale omezil se na jen to nejnútnejší, například se omezí vzorkování nebo četnost vzorků, ale tak, aby systém nezůstal úplně bez vstupních dat). Řadič v nouzovém režimu vykonává stavy propojené tečkovanými hranami, zatímco v normálním režimu stavy propojené plnými hranami. V literatuře se lze pro takové chování setkat s označením elegantní degradace. Jakmile nepříznivé podmínky pominou, řadič se zase automaticky vrátí k základnímu režimu, obnoví plnou činnost systému.



Obrázek 7.10: Diagram přechodů řadiče s normálním režimem činnosti a nouzovým režimem činnosti (tečkované hrany).

Autor práce pro návrh polymorfniho obvodového řadiče s vlastností elegantní degradace vytvořil, ověřil a publikoval [Ruz10a] postup návrhu, který popisuje Algoritmu 7.1.

V algoritmu 7.1 se vyskytují tři množiny stavů. Předně je to množina  $S$  všech stavů, které má řadič mít v normálním režimu své činnosti. Pro příklad z obrázku 7.10 by bylo  $S = \{000, 110, 011, 100, 111\}$ . Dále množina stavů  $M$ , v níž jsou stavy, které musí být vykonávány v nouzovém režimu. Tam patří jistě nezbytné stavy z  $S$ , ale také „náhradní“ stavy, které nejsou v množině  $S$  – stavy, v nichž se realizují omezené činnosti jako náhrada za „plné“ činnosti realizované ve stavech z množiny  $S$ , které byly pro nouzový režim vypuštěny (tedy z rozdílů množin  $S \setminus M$ ).  $M = \{000, 110, 001\}$ . Třetí množina stavů je množina  $D$ . V ní jsou možné stavy řadiče, které nejsou ani v množině  $S$ , ani v množině  $M$ . Takové stavy vzniknou v případě, že počet smysluplných stavů dohromady (sjednocení množin  $S \cup M$ ) je menší než  $2^b$ . Přitom číslo  $b$  je počet bitů, pomocí nichž je vyjádřen stav – počet klopných obvodů v sekvenční části řadiče.

Při činnosti řadiče může dojít k situaci, kdy se řadič ocitne ve stavu, nenáležícím ani do množiny  $S$ , ani do množiny  $M$ . Tato situace může nastat buď díky nějakému přechodnému jevu (polymorfni hradla použitá v obvodu nezmění funkci současně v jeden okamžik) nebo existuje možnost přechodu do některého stavu z množiny  $D$  jako důsledek minimalizace sítě realizující funkce  $F1$  a  $F2$  (minimalizované funkce jsou označeny  $F1'$  a  $F2'$ ). Pak je důležité ověřit, zda ze stavů z množiny  $D$  vedou přechody do některého z platných stavů (z  $S$  nebo  $M$ ), zda stavy z množiny  $D$  nejsou „izolované“. Pokud nevedou, je vhodné zajistit, aby takové přechody

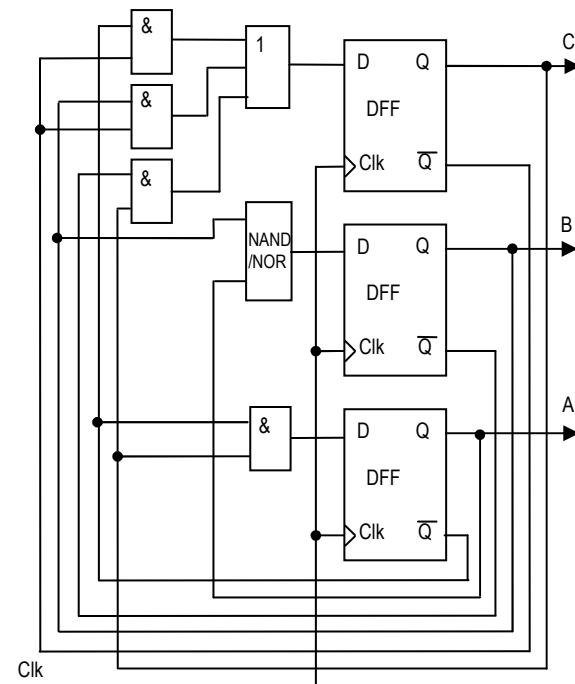
existovaly. Tak bude čítač bezpečný, sám se dokáže po jednom nebo několika hodinových impulsích dostat zase do stavu, který je smysluplný a dále pokračuje normálně už jen stavy z množiny  $\mathbf{S}$  nebo  $\mathbf{M}$  podle režimu. Aby případný přechodný děj neovlivnil činnost obvodu řadičem řízeného, musí se dále zajistit, aby stavy z množiny  $\mathbf{D}$  nebyly spojeny s žádnými činnostmi řízeného obvodu.

#### Algoritmus 7.1:

1. Nechť  $\mathbf{S}$  je množina stavů, které má obvodový řadič v normálním režimu mít. Každý stav je kódován jako binární číslo sestávající z  $\mathbf{b}$  bitů, přičemž  $\mathbf{b} = \lceil \log_2 |\mathbf{S}| \rceil$ .
2. Určí se množina nezbytných stavů  $\mathbf{M}$  (stavů, v nichž konané činnosti musí být vykonány vždy, bez ohledu na ohrožení obvodu), typicky  $\mathbf{M} \cap \mathbf{S} \neq \emptyset$ .
3. Vhodně se zvolí množina hradel  $\Gamma$ , z nichž bude realizována kombinační síť pro výpočet následujícího stavu. Množina by měla obsahovat konvenční hradla, ale také alespoň jedno polymorfní hradlo (hradlo realizující funkce  $f_1/f_2$ , přičemž řízení funkce hradla by mělo být takovou veličinou, jaká charakterizuje vliv ohrožující funkce obvodu a intervaly této veličiny, které vymezují funkce  $f_1$  a  $f_2$  hradla, by měly korespondovat s normálním a ohrožujícím stavem prostředí).
4. Specifikuje se vhodnou formou (pravdivostní tabulka, booleovská rovnice) logická funkce generující následující stav  $\mathbf{F1} = \{(\mathbf{s}_n, \mathbf{s}_{n+1}) \mid \mathbf{s}_n \in \mathbf{S} \wedge \mathbf{s}_{n+1} \in \mathbf{S}\}$  pro normální režim obvodu a  $\mathbf{F2} = \{(\mathbf{m}_n, \mathbf{m}_{n+1}) \mid \mathbf{m}_n \in \mathbf{M} \wedge \mathbf{m}_{n+1} \in \mathbf{M}\}$  pro nouzový režim obvodu.
5. Vytvoří se graf  $\mathbf{G} = (\mathbf{V}, \mathbf{E}, \varphi: \mathbf{V} \rightarrow \Gamma)$  – struktura kombinačního číslicového obvodu implementující funkce  $\mathbf{F1}'$  a  $\mathbf{F2}'$  s využitím hradel z množiny  $\Gamma$ . Pro syntézu obvodu se využije vhodná technika návrhu polymorfních obvodů (viz techniky popsané v kapitole 4). Funkce  $\mathbf{F1}'$  a  $\mathbf{F2}'$  jsou optimalizované (minimalizované) realizace funkcí  $\mathbf{F1}$  a  $\mathbf{F2}$ , tedy platí, že  $\mathbf{F1} \subset \mathbf{F1}'$  a  $\mathbf{F2} \subset \mathbf{F2}'$ .
6. Graf  $\mathbf{G}$  je polymorfní kombinační obvod generující následující stav pro elegantně degradující obvodový řadič.
7. Má-li řadič být bezpečný, je třeba zkontrolovat, zda stavy z množiny  $\mathbf{D} = (2^b \setminus (\mathbf{S} \cup \mathbf{M}))$  nejsou v implementaci řadiče grafem  $\mathbf{G}$  izolované, tj.  $\exists \mathbf{x}_{n+1} \in (\mathbf{S} \cup \mathbf{M}) : (\forall \mathbf{d}_n \in \mathbf{D} : (\mathbf{d}_n, \mathbf{x}_{n+1}) \in \mathbf{F1}' \vee (\mathbf{d}_n, \mathbf{x}_{n+1}) \in \mathbf{F2}')$ .
8. Ve stavech z množiny  $\mathbf{D}$  nesmí být prováděna žádná činnost.

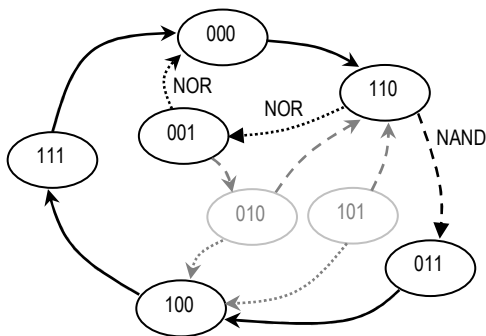
Pro diagram přechodů řadiče z obrázku 7.10 byl algoritmem 7.1 navržen řadič, jehož zapojení je vidět na obrázku 7.11. Jako polymorfní hradla byla do množiny  $\Gamma$  použita hradla NAND/NOR řízená  $V_{dd}$ .

Polymorfní řadič, jehož schéma je vidět na obrázku 7.11, pracuje podle přechodového diagramu, který je na obrázku 7.10. Je-li polymorfní hradlo v režimu NAND, realizuje stavy z množiny  $\mathbf{S} = \{000, 110, 011, 100, 111\}$ . Pokud napětí zdroje klesne, polymorfní hradlo začne pracovat v režimu NOR a řadič realizuje stavy z množiny  $\mathbf{M} = \{000, 110, 001\}$ . Na třech bitech existují však ještě kombinace (stavy) 010 a 101, které nejsou ani v množině  $\mathbf{S}$ , ani v množině  $\mathbf{M}$ . To jsou stavy, které tvoří množinu  $\mathbf{D}$ . Obrázek 7.12 ukazuje stavový diagram popisující úplné chování řadiče, jehož schéma je na obrázku 7.11, včetně stavů z množiny  $\mathbf{D}$  a přechodů, které z nich a do nich vedou.



Obrázek 7.11: Realizace polymorfního řadiče dle přechodového diagramu z obrázku 7.10.

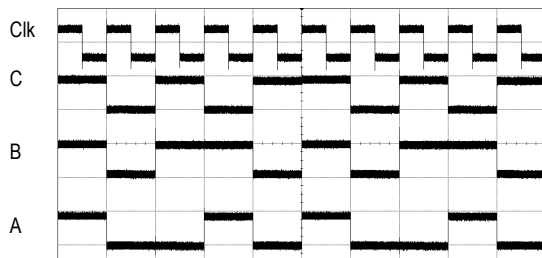
Na obrázku jsou plnými čarami vyznačeny ty přechody, které nezávisí na režimu řadiče. Provedou se vždy, ať je řadič v normálním či nouzovém režimu. Čárkovanými čarami jsou vyznačeny přechody, které se provedou pouze v normálním režimu a tečkovanými čarami ty, které se naopak provedou pouze v nouzovém režimu. Nevyužité stavy z množiny **D** jsou šedivé. Je vidět, že do stavu 010 se může řadič dostat v okamžiku, kdy se změní režim řadiče a řadič právě vykonává stav 001. Ze stavu 010 se však jedním hodinovým impulsem řadič dostane do normálního stavu 110 a dále pokračuje v souladu s předpokládaným chováním. Do stavu 101 se řadič může dostat pouze vlivem přechodového jevu při změně funkce, protože žádný přechod do tohoto stavu nevede. Zotavení z tohoto stavu je vidět na obrázku. V případě normálního režimu stačí jeden hodinový impuls, aby se řadič dostal do smysluplného stavu 110, dále pak pokračuje v souladu s předpokládaným chováním. V případě nouzového režimu bude ještě třeba vykonat „normální“ stavy 110 a 111, než se řadič dostane do nouzové smyčky 000 – 110 – 001. Znamená to ale pouze oddálení náběhu úsporného režimu o dva takty hodin, což ve většině aplikací nebude kritické.



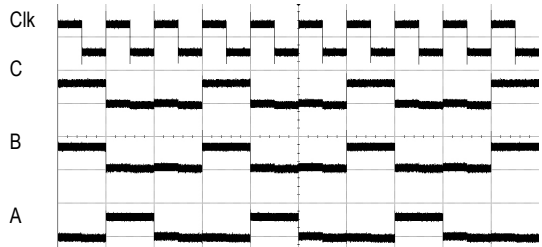
Obrázek 7.12: Úplný stavový diagram řadiče z obrázku 7.11.

Navržený polymorfní řadič byl implementován s využitím konfigurovatelného obvodu REPOMO (viz kapitola 5 této práce) a jeho činnost byla experimentálně ověřena včetně změny režimu řadiče změnami úrovně napájecího napětí.

K implementaci byly využity pouze první čtyři sloupce hradel z obvodu REPOMO (viz obrázek 5.6). Tři ze čtyř primárních vstupů ( $data\_in[0-2]$ ) byly využity jako vstupy aktuálního stavu. Tři primární výstupy ( $data4\_out[0-2]$ ) byly využity jako výstup následujícího stavu. Na výstupy obvodu REPOMO byly zapojeny tři standardní klopné obvody CMOS typu D, použité jako paměť stavu řadiče. Jejich výstupy byly propojeny se vstupy obvodu REPOMO (podobně, jako je vidět na obrázku 7.6). Obvod REPOMO byl konfigurován následující konfigurací sekvencí: 0000011, 00100111, 01001011, 0000000, 10101000, 00111000, 00011000, 10010111, 10011000, 00000101, 01001000, 01101100, 00000101, 01001000, 01101100, 00000000. Obrázek 7.13 ukazuje časové průběhy na výstupech aktuálního stavu řadiče C, B a A v závislosti na hodinovém signálu Clk v normálním režimu řadiče. Je zřetelně vidět, že řadič neustále vykonává stavy 111, 000, 110, 011, 100.



Obrázek 7.13: Časový průběh výstupů řadiče z obrázku 7.11 v normálním režimu.

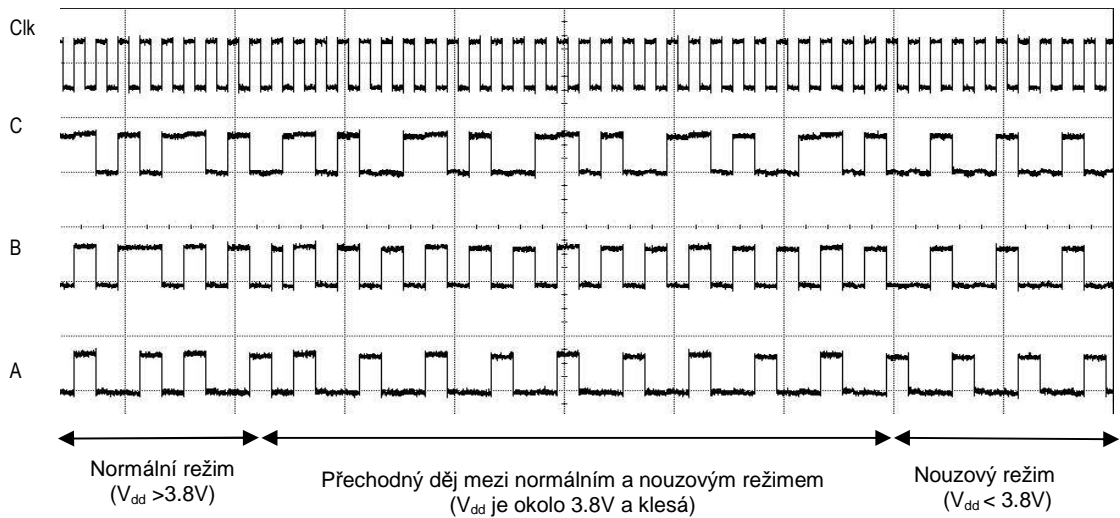


Obrázek 7.14: Časový průběh výstupů řadiče z obrázku 7.11 v normálním režimu.

Na obrázku 7.14 jsou časové průběhy na výstupech aktuálního stavu řadiče C, B a A v závislosti na hodinovém signálu Clk v nouzovém režimu řadiče. Je zřetelně vidět, že řadič neustále vykonává stavy 110, 001, 000.

Časové průběhy výstupních signálů řadiče na obrázku 7.15 zachycují okamžik, kdy dochází ke změně funkce v důsledku poklesu napájecího napětí. Pokles napájecího napětí je pozvolný, což odpovídá reálné situaci pomalého vybíjení baterie. V levé části obrázku řadič pracuje se stavy z množiny **S**, až po sekvenci stavů 111, 000, 110 nenásleduje 011, jak by odpovídalo normálnímu režimu řadiče, ale stav 001, což znamená přechod do nouzového režimu (překlopení použitého polymorfního hradla z režimu NAND do režimu NOR). Protože však polymorfní hradlo v situaci změny funkce také má určitý přechodný děj, kdy funkce se může několikrát změnit, než se ustálí na nové (viz obrázky 3.14 a 3.15), i řadič, v němž je hradlo použito, bude určitý přechodný děj vykazovat. Proto po prvním výskytu stavu 001 nenásleduje stav 000, jak by odpovídalo chodu řadiče v nouzovém režimu, ale opět 111, po něm 000, 110, 001 a pak dokonce 010, tedy jeden ze stavů množiny **D**. Díky schopnosti řadiče zotavit se z těchto stavů se řadič opět vrátí k vykonávání stavů, které mají význam. Během přechodného

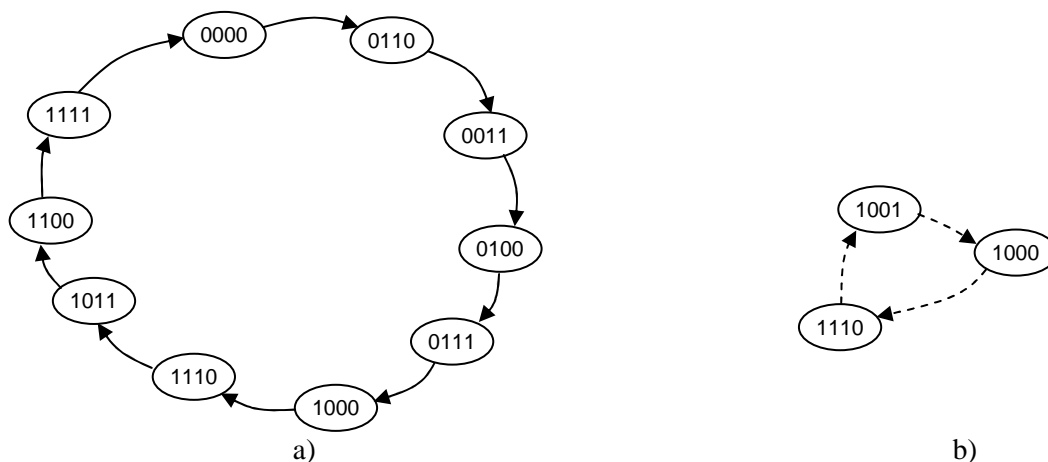
děje se stav 010 vyskytne ještě asi čtyřikrát (viz obrázek 7.15), pak už ale řadič bezchybně vykonává stavy z množiny  $\mathbf{M}$  (na obrázku 7.15 vpravo).



Obrázek 7.15: Přechodný děj mezi normálním a nouzovým režimem.

Autor této práce experimentoval také s citlivostí polymorfního hradla NAND/NOR na teplotu. Výsledky experimentů se samotným hradlem jsou popsány v kapitole 3.5, pro ověření chování na složitějším obvodu byl s využitím algoritmu 7.1 navržen další polymorfní řadič. Polymorfní řadič řízený teplotou může aktivně působit proti přehřívání čipu ve zhoršených provozních podmínkách, například v situaci, kdy chlazení systému má z důvodů vysoké okolní teploty sníženou účinnost (kosmická sonda vystavená momentálně intenzivnímu slunečnímu záření). V literatuře se systémům, které dokážou reagovat na růst teploty čipu snížením energetické ztráty obvodu, říká též „temperature aware“ systémy [Bao08], [Xie06].

Na obrázku 7.16 a) je diagram přechodů navrhovaného řadiče pro normální režim činnosti (množina  $\mathbf{S}$  dle algoritmu 7.1) a na obrázku 7.16 b) pro nouzový režim činnosti (množina  $\mathbf{M}$  dle algoritmu 7.1). Mezi množinami jsou sdílené stavy 1110 a 1000. To jsou stavy, které je nutno vykonat v každém případě, i ve stavu nouze. Činnosti prováděné ve stavu 1001 nouzově nahrazují všechny ostatní činnosti obvodu. Protože stavů v normálním režimu je 10 a k tomu ještě další nouzový stav 1001, kódují se stavy na čtyřech bitech. Řadič bude potřebovat k zapamatování stavu čtyři klopné obvody.

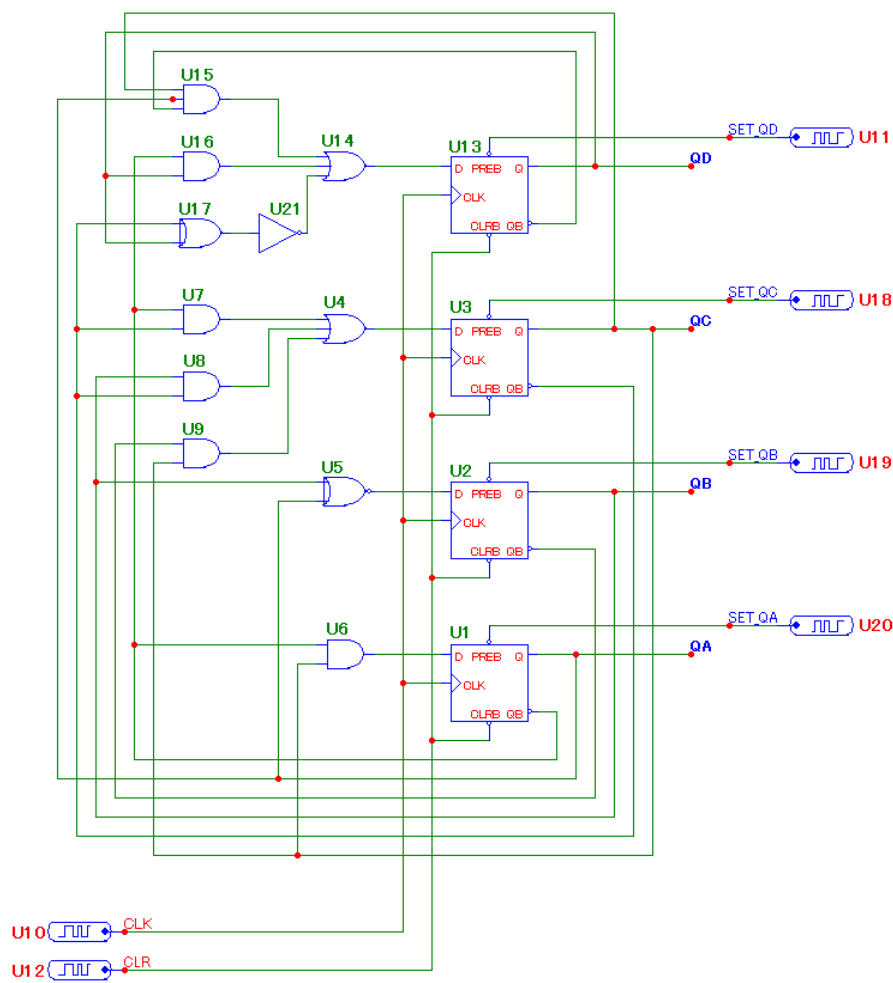


Obrázek 7.16: Diagram přechodů řadiče v normálním a) a v nouzovém režimu činnosti b).

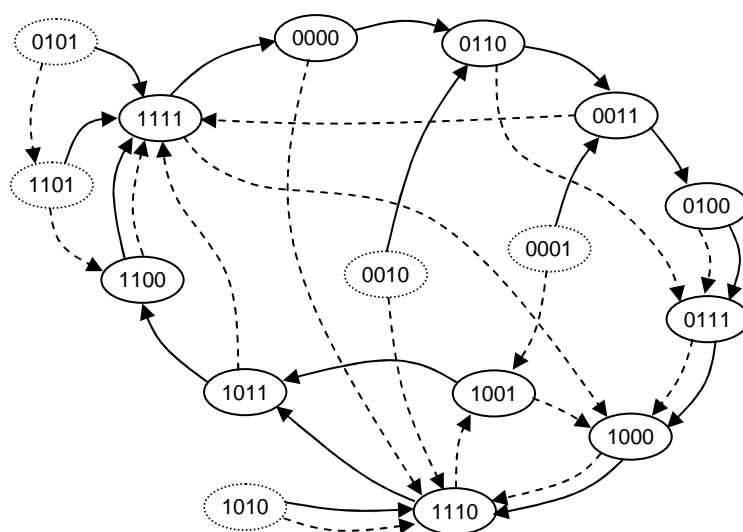
Algoritmem 7.1 byl navržen obvod, jehož schéma je na obrázku 7.17. Obvod obsahuje dvě polymorfní hradla, na obrázku mají označení U5 a U17.

Obrázek 7.18 ukazuje stavový diagram popisující úplné chování řadiče, jehož schéma je na obrázku 7.17, včetně stavů z množiny  $\mathbf{D}$  a přechodů, které z nich a do nich vedou.

Na obrázku jsou plnými čarami vyznačeny ty přechody, které budou prováděny v normálním režimu řadiče. Čárkovanými čarami jsou vyznačeny přechody, které se provedou pouze v nouzovém režimu. Nevyužité stavy, tvořící množinu  $\mathbf{D}$ , jsou označeny tečkovanou čarou. Jde o stavy 0101, 1101, 0001, 0010 a 1010. Z obrázku je vidět, že se jedná o stavy, do nichž nevede žádná hrana, řadič se tedy do nich může dostat jedině díky přechodnému ději. Je také zřejmé, že z těchto stavů se řadič může velmi rychle zotavit, vždy z nich vede krátká cesta do některého platného stavu.



Obrázek 7.17: Realizace polymorfního řadiče dle přechodového diagramu z obrázku 7.16.



Obrázek 7.18: Úplný stavový diagram řadiče z obrázku 7.17.



## 8 Závěr

### 8.1 Shrnutí

Existuje celá řada aplikací, kde je výhodné, aby číslicový obvod dokázal změnit svoji funkci třeba i v průběhu své činnosti – například se adaptoval na měnící se podmínky, ať už se změna odehrává vně obvodu nebo uvnitř (např. porucha některých částí obvodu). Ne vždy lze takový systém implementovat na úrovni software běžícím na univerzálním výpočetním stroji, kde otázka změny funkce je již z podstaty dobře řešitelná. Jsou případy, kdy je nutné jisté funkce řešit v hardware (z důvodů rychlosti, prostorových nároků, spotřeby, ...) a hledat techniky, jak zajistit změnu funkce samotného obvodu při zachování parametrů, které jsou na něj kladeny.

Je-li třeba navrhnout obvod, který plní více než jednu funkci, postupuje se zpravidla tak, že se navrhnou obvody pro všechny požadované funkce samostatně. Takto navržené obvody se pak implementují samostatně a přepínají podle požadované funkce (například signálem ze senzoru detekujícího prostředí) nebo se uloží jako konfigurace pro nějaký konfigurovatelný obvod, do něžž se pak nahraje ta z připravených konfigurací, která reprezentuje funkci právě po obvodu požadovanou. Výhodou první uvedené možnosti je okamžité přepnutí z jedné funkce na jinou, za to se však platí velkými prostorovými nároky (i když je teoreticky možné určité části obvodu sdílet pro více funkcí). Druhá uvedená možnost je výhodnější prostorově (i když i zde je poměrně velká režie vzhledem k tomu, že konfigurovatelný obvod musí být do jisté míry univerzální a dále je potřeba paměťový prostor na uložení konfigurací a mechanismus jejich záměny), avšak změna konfigurace není okamžitá – po určitou dobu, nutnou ke změně konfigurace, obvod není funkční.

V poslední době se řada výzkumníků věnuje možnosti adaptace hardwaru pomocí evolučních algoritmů. Předpokládá se využití konfigurovatelného obvodu, kdy konfiguraci vytváří s přihlédnutím ke stavu prostředí (též vstupních dat a jejich jakosti, stavu obvodu a jakosti jeho výstupu) evoluční algoritmus. V tomto případě není potřeba mít předem připravené a uložené konfigurace obvodu pro různé stavy prostředí, ale na konfigurovatelný obvod se v tomto případě kladou daleko vyšší nároky. Mimo jiné je třeba, aby veškeré (i náhodné) konfigurace obvodu byly bezpečné z pohledu konfigurovatelného obvodu i vlastní aplikace, protože evoluční algoritmus může během své práce generovat prakticky cokoliv.

Návrh pomocí evolučních technik (off-line) však stál u zrodu polymorfní elektroniky. Právě tento způsob návrhu dokáže generovat i velmi nekonvenční řešení, běžnými technikami návrhu prakticky nedostupná. Je možné navrhovat i obvody, které budou prostorově relativně nenáročné, avšak mohou s drobnými změnami (nikoliv ve struktuře, ale v pracovních bodech použitých tranzistorů) realizovat dvě či více rozdílných (smysluplných) funkcí. Navíc, pracovní body tranzistorů (a tím pádem funkci obvodu) lze ovlivňovat různými způsoby. Obvod se proto stává přirozeně citlivým na prostředí.

Pod pojmem „polymorfní elektronika“ se v této práci rozumí oblast elektroniky, zkoumající číslicové elektronické obvody, které dokážou plnit více než jednu funkci, přičemž jejich zapojení zůstává stále stejné a aktuálně prováděná funkce závisí na stavu okolního prostředí. Všechny funkce jsou do obvodu zabudovány úmyslně již při jeho návrhu, počítá se s nimi. Jde tedy o smysluplné funkce, nikoliv například o poruchový stav vyvolaný tím, že jsou překročeny provozní parametry obvodu. Stav okolního prostředí lze popsat exaktně nejčastěji hodnotou nějaké fyzikální veličiny, pro konkrétní stav prostředí lze jednoznačně určit, jakou funkci bude obvod plnit.

Polymorfní obvody jsou typicky rozměrově velmi úsporné (ve srovnání s „konvenčními“ vícefunkčními obvody), změna funkce nastává přirozeně a okamžitě (není třeba čas na rekonfiguraci, po který je obvod mimo provoz) a detekce stavu prostředí je přímo vestavěna do obvodu, je často distribuována, protože každý tranzistor, který má vhodně nastaven pracovní bod, reaguje na prostředí sám za sebe.

Typickým „prostředím ovlivňujícím funkci“ polymorfních obvodů bývá napájecí napětí, teplota, intenzita osvětlení a podobně, tedy veličiny, které ovlivňují chování polovodičových tranzistorů. Koncept polymorfní elektroniky je však v této práci zobrazen. Je v zásadě možné použít stavební bloky postavené na jiném principu než běžné polovodičové unipolární tranzistory, které budou citlivé na jiné fyzikální veličiny, a tím získat polymorfní obvod reagující na zcela jiné veličiny charakterizující prostředí. V současné době však až na malé výjimky jsou všechny experimenty prováděny s obvody na bázi polovodičových unipolárních tranzistorů. Proto i část této práce se věnuje současným známým polymorfním hradlům postaveným na bázi polovodičových tranzistorů.

Polymorfní hradla jsou chápána jako základní stavební blok polymorfních číslicových obvodů. Polymorfní číslicové obvody se tedy navrhnou podobně jako konvenční (monofunkční) číslicové obvody na úrovni hradel. Polymorfní hradlo je obvod, realizující v různých režimech různé elementární logické funkce, přičemž režim hradla je ovlivněn prostředím. Práce přináší přehled v současnosti známých polymorfních hradel a jejich srovnání jak mezi sebou, tak i s konvenčními hradly, kterým při návrhu vícefunkčních obvodů v jistém smyslu konkurují. V práci je ukázáno, že již dnes existují polymorfní hradla, která jsou prakticky použitelná, a lze s nimi dosáhnout funkčních implementací vícefunkčních obvodů, které jsou lepší než používaná konvenční řešení. Řada z nich byla fyzicky realizována a experimentálně ověřena, což je v práci rovněž popsáno.

Protože jednou ze stěžejních předností polymorfních obvodů jsou jejich kompaktní rozměry, bylo by ideální provádět návrh na úrovni tranzistorů. Pro malé obvody (typicky realizace elementárních logických funkcí, tedy hradla) to skutečně možné je, je možno použít evolučního návrhu i ad-hoc expertního návrhu zkušeného návrháře (ideálně se zkušenostmi z návrhu analogové i číslicové techniky). Návrh na úrovni tranzistorů však naráží na problém škálovatelnosti. Proto byly navrženy metody návrhu na úrovni hradel (elementárních logických funkcí) podobně, jako je tomu u konvenčních číslicových obvodů. Na této úrovni již není třeba uvažovat o mechanismu citlivosti na prostředí, návrh se soustředí jen na vytvoření zapojení hradel-komponent (vytvoření grafu, kde vrcholy tvoří vstupy a výstupy hradel a hrany spoje mezi nimi), které bude realizovat příslušné funkce pro různé režimy hradel.

Pro návrh polymorfních obvodů na úrovni hradel je jistě možné použít evolučních technik. V práci jsou popsány možnosti a dosažené výsledky takového návrhu. Technikou Kartézského genetického programování, vyvinutou a používanou primárně pro návrh konvenčních číslicových obvodů na úrovni hradel, byla navržena řada polymorfních obvodů, které v mnoha parametrech předčí známá konvenční řešení. Problém škálovatelnosti návrhu se evolučními technikami ale neřeší zcela, pouze se odsouvá do vyšších řádů. Proto vznikly a v práci jsou popsány také dvě ne-evoluční (lze říci konvenční) techniky návrhu polymorfních obvodů. Je to technika polymorfního multiplexování, kdy se polymorfním multiplexorem (reagujícím na prostředí) přepínají konvenčně navržené obvody realizující požadované funkce, a technika využívající binárních rozhodovacích diagramů (a jejich implementace číslicovým obvodem) k realizaci dvojfunkčních číslicových obvodů. Pro požadovanou dvojici funkcí se vytvoří společná tabulka logických hodnot vstupů a výstupů, z ní se pak definovaným postupem vytvoří multiterminální binární rozhodovací diagram (vrcholy mohou obsahovat i celočíselné hodnoty) a ten je dle daného postupu transformován na zapojení obvodu, přičemž hodnoty v terminálních uzlech diagramu implikují konkrétní podsíť s polymorfními prvky.

Obvody vytvořené popsány technikami však nebývají příliš optimální, pokud jde o rozměry, proto je vhodné provést ještě následnou optimalizaci. Ta může být provedena opět evolučními technikami, které jsou pro optimalizace ověřeným nástrojem.

Pro experimenty s fyzicky realizovanými polymorfními obvody a také pro podporu návrhu polymorfních obvodů zejména technikou Kartézského genetického programování byl navržen polymorfní konfigurovatelný obvod. Obsahuje polymorfní hradla NAND/NOR, řízená napájecím napětím, která byla již dříve navržena a prakticky ověřena, a dále konvenční hradla. Obvod byl nazván REPOMO (Rekonfigurovatelný polymorfní modul) a obsahuje 32 programovatelných elementů s konfigurovatelnou funkcí a propojením, uspořádaných v matici 8 sloupců x 4 řádky. Obvod má 4 primární vstupy a 4 primární výstupy, přičemž je možné použít více modulů současně pro vytvoření složitějšího obvodu. Pro tyto účely jsou vyvedeny na piny pouzdra ještě výstupy ze 4. řady programovatelných elementů. V práci je obvod detailně popsán a jsou popsány také experimenty, které s ním byly provedeny, ať už pro ověření jeho konstrukce nebo pro ověření různých aplikací polymorfních obvodů. Je třeba poznamenat, že se jedná patrně o první fyzicky vyrobený rekonfigurovatelný polymorfní obvod na světě. Pro snadné experimentování s obvodem byl dále navržen kit, prostřednictvím kterého je možno obvod konfigurovat, ovlivňovat jeho prostředí (napájecí napětí, teplotu), generovat stimuly pro jeho vstupy a vyhodnocovat jeho výstupy. Kit podporuje jak samostatnou činnost obvodu, tak i jeho řízení prostřednictvím řídicího počítače.

V práci je popsána také řada aplikací polymorfních obvodů. Většina aplikací byla realizována a ověřena buď simulačně nebo přímo fyzickou realizací. Obecně lze polymorfní obvody nasadit v aplikacích, kde se předpokládá adaptace obvodu na měnící se podmínky nebo rychlá a elegantní rekonfigurace. Možnosti naznačuje příklad využití v obvodech, které se samy přizpůsobí nepříznivým podmínkám (omezí svoji spotřebu, tepelné vyzařování, při zachování alespoň nezbytné základní činnosti) – příklad s redukcí počtu koeficientů číslicového filtru v části 6.3. Prosadit se však mohou i tam, kde se předpokládá obvod sice monofunkční, ale s nějakou přidanou vlastností. Tato přidaná vlastnost přitom může být využívána jen zřídka, ale díky typické vlastnosti polymorfních obvodů – kompaktní a prostorově velmi nenáročná konstrukci – může být získána za velmi příznivou cenu. Příkladem takové aplikace může být využití v diagnostice (viz příklad zkrácení testu v části 6.2.1) nebo pro přidání vlastnosti samotestování (samotestující se jednobitové sčítačky s propagací informace o poruše popsané v části 6.2.2). Dále se levně přidaná funkce hodí v oblasti bezpečnosti, jak ukazuje příklad fyzické neklonovatelné funkce pro autentizaci nebo identifikaci čipu, popsány v části 6.4.

Zvláštní postavení mezi polymorfními obvody mají ty obvody, které mění funkci podle hodnoty speciálního signálu řízení funkce. Pokud jde o polymorfní obvody se dvěma funkcemi, lze tento signál chápat vlastně jako další logický vstup. Polymorfní hradlo s  $n$  vstupy a dvěma funkcemi, řízené speciálním signálem, může být chápáno jako konvenční hradlo s  $n+1$  vstupem a může být použito v návrzích konvenčních číslicových obvodů. Jeho využití předpokládá nasazení zejména evolučních technik návrhu, které jsou schopny pracovat i s netypickými komponenty. Výsledkem pak může být obvod, který je v některých parametrech (typicky rozměry, zabraná plocha čipu) lepší než dosud známé konvenční řešení.

Principů polymorfní elektroniky lze využít nejen u kombinačních, ale i u sekvenčních číslicových obvodů. Při jejich návrhu může být využito speciálně navržených polymorfních klopných obvodů, které (podobně jako polymorfní hradla) mění realizovanou funkci podle stavu veličiny popisující prostředí. Pro návrh takových obvodů však dosud nebyly navrženy vhodné postupy. Druhá alternativa, kdy se využívají konvenční klopné obvody doplněné polymorfní kombinační logikou, umožňuje použít konvenční techniky návrhu kombinačních obvodů v kombinaci s některou popsanou technikou návrhu kombinačních polymorfních obvodů.

Sekvenční polymorfni obvody nachází uplatnění v podobných oblastech jako kombinační. V práci je ukázáno, jak lze využít vlastností polymorfnií obvodů pro adaptivní obvodové řadiče. Je navržena a popsána metoda syntézy polymorfnií obvodových řadičů, které samočinně reagují na zhoršené podmínky omezením činnosti řízeného obvodu při zachování základních funkcí nezbytných pro „přežití“ zařízení do doby, než se vnější podmínky zlepší. Funkčnost metody je demonstrována na několika příkladech včetně ověření fyzickou realizací.

## 8.2 Zhodnocení práce

Práce je v první řadě komplexním přehledem současného stavu poznání výzkumné oblasti, nazvané autory této myšlenky „polymorfni elektronika“. Patrně v současné době neexistuje práce, která by se polymorfni elektronice věnovala takto komplexně. Autor si je vědom, že by si téma proto zasloužilo zpracování v anglickém jazyce, aby bylo dostupné pro celosvětovou vědeckou komunitu. Primárním účelem však bylo vytvořit kvalifikační práci, proto se autor soustředil především na obsah a použil k jeho naplnění svůj mateřský jazyk. Autor však zvažuje budoucí vytvoření anglické verze a možnost vydání případně rozšířené a doplněné verze u vhodného vydavatele.

V práci lze nalézt jak (v rámci možností úplnou) definici oblasti polymorfni elektroniky, tak řadu možných odpovědí na tři základní problémy polymorfni elektroniky, formulované v úvodu této práce, totiž:

- problém syntézy polymorfnií obvodů,
- problém existence vhodných základních stavebních bloků (hradel) a
- problém nalezení dostatku aplikací polymorfni elektroniky, v nichž by polymorfni obvody znamenaly nesporný přínos oproti konvenčním řešením.

Kromě hodnoty spočívající v komplexním uchopení výzkumné oblasti zvané „polymorfni elektronika“ obsahuje práce také řadu poznatků a výsledků, které jsou vlastní prací autora nebo na jejichž vzniku a realizaci se autor podílel. Řada těchto výsledků byla postupně publikována na mezinárodním fóru a jsou přístupné čtenářům v uznávaných mezinárodních databázích vědeckých článků. Mezi zcela konkrétní výsledky, které jsou vlastní prací autora a jimiž přispěl k výzkumu polymorfni elektroniky, lze zařadit:

- Formální definici polymorfniího obvodu (definice 2.1 a s ní související poznámky).
- Formální definice prostředí ovlivňující funkci polymorfniího obvodu (rozpracovaná v části 2.4 této práce).
- Pojmenování a rozdělení způsobů využití polymorfni elektroniky na dvě základní oblasti (způsoby Z1 a Z2 uvedené v části 2.4).
- Formální definice polymorfniího hradla (definice 3.1 a s ní související poznámky).
- Definování kvalitativních požadavků kladených na polymorfni hradla (definice 3.2).
- Hodnocení všech popsaných polymorfnií hradel podle požadavků definice 3.2 – u všech hradel popsaných v části 3.5 této práce.
- Veškeré experimenty s polymorfniím hradlem NAND/NOR, řízeným  $V_{dd}$ , které bylo navrženo a realizováno ve spolupráci FIT a FEKT VUT v Brně od počátku jeho návrhu a výroby.
- Autorství návrhu hradla NAND/NOR řízeného speciálním signálem – hradlo je popsáno v části 3.5.3.
- Autorství návrhu hradla NAND/XOR, řízeného speciálním signálem – hradlo je popsáno v části 3.5.3.

- Spoluautorství návrhu polymorfního konfigurovatelného čipu REPOMO32 a experimenty ověřující parametry vyrobeného čipu – viz část 5.2.
- Spoluautorství návrhu kitu REPOMO32/kit pro experimenty s polymorfními obvody a návrh polymorfních číslicových obvodů – viz část 5.3.
- Veškeré experimenty s řízením polymorfních hradel NAND/NOR teplotou s využitím čipu REPOMO a kitu REPOMO32/kit – viz část 3.5.1.
- Podíl na navržených aplikacích polymorfní elektroniky a experimentech s nimi: samotestující se sčítačky (viz část 6.2.2), polymorfní filtry (část 6.3), fyzická neklonovatelná funkce (viz část 6.4).
- Autorství postupu návrhu adaptivních polymorfních řadičů (algoritmus 7.1) a veškeré realizace a experimenty s nimi – viz část 7.3.

Protože autor osobně provedl mnoho experimentů s fyzickou realizací polymorfních obvodů, při nichž na jednu stranu získal odpovědi na některé otázky, ale na druhou stranu objevil další aspekty, které nebyly na první pohled zřejmé, odráží text práce a zejména autorem provedené definice a tvrzení kromě očekávaného zobecňujícího a vědeckého přístupu také praktickou zkušenost, kterou autor za léta práce s polymorfními obvody získal. Ta může být užitečná dalším při pokračování výzkumu na poli polymorfní elektroniky.

### 8.3 Budoucnost

Koncept polymorfní elektroniky se autorovi jeví jako slibný směr při úvahách o kompaktních vícefunkčních obvodech, od nichž se očekává adaptace na různé podmínky. Ukázalo se, že je možné nalézt použitelné a fungující postupy, jak syntetizovat kompaktní vícefunkční obvody z vícefunkčních komponent. Další výzkum syntézy polymorfních obvodů by bylo třeba zaměřit zejména na metody generující optimálnější výsledky (z pohledu rozměrů obvodu – počtu hradel, tranzistorů, plochy čipu). Současný stav, kdy přiměřeně optimálních výsledků se dosahuje zejména s použitím evolučních technik, je pro budoucnost problematický s ohledem na omezenou škálovatelnost těchto postupů. Zde je ještě dostatek prostoru pro další výzkum.

Snahu o hledání aplikací pro polymorfní elektroniku, motivovanou prokázáním životaschopnosti této myšlenky, lze v současné době považovat za úspěšnou. Je řada aplikačních oblastí, kde polymorfní elektronika přináší pokrok a kvalitativně lepší řešení nežli jsou řešení dosud známá. Tato práce přehledně shrnuje řadu z nich včetně demonstrace výsledků a jejich kvalitativního zhodnocení. Pokud se myšlenka polymorfní elektroniky ve světové vědecké komunitě a mezi pracovníky aplikovaného výzkumu rozšíří, budou zřejmě další aplikace vznikat přirozeně.

Třetí oblast problémů, oblast základních stavebních bloků polymorfních obvodů, polymorfních hradel, se jeví zatím jako nejméně se rozvíjející. Drtivá většina hradel je postavena na principu konvenčních unipolárních tranzistorů. Na jednu stranu to je nutnost, vyplývající z potřeby integrace polymorfních hradel s běžnými monofunkčními konvenčními hradly, na stranu druhou je to ale faktor, který je pro hradla nejvíce limitující. Polymorfním hradlům, jejich vlastnostem a praktickým aspektům jejich použití je v této práci věnována značná pozornost. Zkušenosti s nimi ukazují, že požadavek na vícefunkčnost způsobenou změnami pracovního bodu tranzistorů v kombinaci s požadavky na malé rozměry vynucuje, aby tranzistory pracovaly v lineárním režimu. To má negativní vliv na spotřebu hradel v klidu, nesrovnatelnou s klidovou spotřebou klasických hradel CMOS. Navíc tento způsob návrhu vede na rozdílné rozměry kanálu tranzistorů a negativně tak ovlivňuje kvalitu logických úrovní v obvodech.

Koncept polymorfní elektroniky, jak je předestřen v této práci, však není závislý na konkrétní fyzické realizaci základních stavebních bloků, realizujících logické funkce. Proto je perspektivní i pro budoucí logické systémy, které se oprostí od klasického schématu realizace hradel technologií CMOS. První vlašťovkou může být například grafenové multifunkční hradlo (popsané v části 3.5.4 této práce), kde je přirozeným způsobem vestavěn potenciál k realizaci více různých základních logických funkcí.

---

## Literatura

- [AD09] AD5241/AD5242, I2C-Compatible, 256-Position Digital Potentiometers. Katalogový list, revize C, Analog Devices, 2009, 20 stran.
- [AD11] ADT7301,  $\pm 1^\circ\text{C}$  Accurate 13-Bit Digital Temperature Sensor. Katalogový list, revize B, Analog Devices, 2011, 16 stran.
- [Bäc96] Bäck, T.: *Evolutionary Algorithms in Theory and Practice*. Oxford University Press, New York, Oxford, 1996.
- [Bao08] Bao, M., Andrei, A., Eles, P., Peng, Z.: Temperature-aware voltage selection for energy optimization. Proc. of DATE, 2008.
- [Bob07] C. Bobda: *Introduction to Reconfigurable Computing: Architectures*. Springer, 2007, 362 stran.
- [Brg85] Brglez, F., Fujiwara, H.: A neutral netlist of 10 combinatorial benchmark circuits and a target translator in FORTRAN. Int. Symposium on Circuits and Systems, Special Session on ATPG and Fault Simulation, 1985, str. 663 – 698.
- [Bui00] Bui, H. T., Al-Sheraidah, A. K., Wang, Y.: New 4-transistor XOR and XNOR designs. Proc. of 2<sup>nd</sup> IEEE Asia Pacific ASIC 2000, str. 25 – 28.
- [Dia79] Diaz, M., Azéma, P., Ayache, J.-M.: Unified design of selfchecking and fail-safe combinational circuits and sequential machines. IEEE Trans. Computers, 28(3), 1979, str. 276 – 281.
- [Eib07] Eiben, A. E., Smith, J. E.: *Introduction to Evolutionary Computing*. 2. vydání, Springer 2007.
- [FT10] FT232R USB UART IC. Katalogový list, verze 2.09, Future Technology Devices International Ltd., 2010, 43 stran.
- [Gaj07] Gajda, Z., Sekanina, L.: Reducing the Number of Transistors in Digital Circuits Using Gate-Level Evolutionary Design. 2007 Genetic and Evolutionary Computation Conference, New York, ACM, 2007, str. 245 – 252.
- [Gaj11] Gajda, Z.: *Evoluční přístup k syntéze a optimalizaci běžných a polymorfních obvodů*. Disertační práce. UPSY FIT VUT v Brně, 2011, 92 stran.
- [Gar05] Garvie, M.: *Reliable Electronics through Artificial Evolution*. Disertační práce. University of Sussex, 2005.
- [GaS11] Gajda, Z., Sekanina, L.: On Evolutionary Synthesis of Compact Polymorphic Combinational Circuits. Journal of Multiple-Valued Logic and Soft Computing, roč. 17, č. 6, 2011, Philadelphia, str. 607 – 631, ISSN 1542-3980.
- [Gua07] Guajardo, J., Kumar, S. S., Schrijen, G. J., Tuyls, P.: FPGA intrinsic PUFs and their use for IP protection. Cryptographic Hardware and Embedded Systems Workshop, LNCS Series, vol. 4727, 2007, str. 63 – 80.

- 
- [Har01] Hartenstein, R.: A decade of reconfigurable computing: a visionary retrospective. Proceedings of the Conference on Design, Automation and Test in Europe (DATE 2001), Munich, IEEE Press, 2001, str. 642 – 649.
- [Har95] Harris, S. P., Ifeachor, E. C.: Automating IIR filter design by genetic algorithm. Proc. of the First IEE/IEEE International Conference on Genetic Algorithms in Engineering Systems: Innovations and Applications (GALESIA'95), no. 414. IEE, 1995, str. 271 – 275.
- [Hig99] Higuchi, T., Kajihara, N.: Evolvable Hardware Chips for Industrial Applications. Communication of the ACM. Vol. 42, No.4. 1999, str. 60 – 66.
- [Koz92] Koza, J. R.: Genetic Programming: On the Programming of Computers by Means of Natural Selection. MIT Press, 1992, 840 stran, ISBN 0-262-11170-5.
- [Kva00] Kvasnička, V., Pospíchal, J., Tiňo, P.: Evolučné algoritmy. STU Bratislava, 2000.
- [Lan05] Langeheine, J.: Intrinsic Hardware Evolution on The Transistor Level. Disertační práce. Rupertus Carola University of Heidelberg, 2005.
- [Lee97] Lee, M. T. C.: High-Level Test Synthesis of Digital VLSI Circuits. Artech House Boston, London 1997.
- [Líš93] Líška, M., Šulo, V., Strelec, J.: Programovatelná logická pole. Grada, 1993, 451 stran.
- [Lof00] Lofstrom, K., Daasch, W. R., Taylor D.: IC Identification Circuit Using Device Mismatch. Proc. of International Solid State Circuits Conference, 2000, str. 372 – 373.
- [Luo07] Luo, W., Zhang, Z., Wang, X.: Designing polymorphic circuits with polymorphic gates: a general design approach. IET Circuits Devices Syst., 2007, 1, (6), str. 470 – 476.
- [Mae92] Maerz, S.: High-Level Synthesis. The Synthesis Approach to Digital System Design (Michel, P., Lauther, U., Duzy, P., eds.), Kluwer Academic Publishers, 1992, str. 115 – 220.
- [Mai09] Maiti, A., Raghunandan, N., Reddy, A., Schaumont, P.: Physical unclonable function and true random number generator: a compact and scalable implementation. Proceedings of the 17th Great Lakes Symposium on VLSI, 2009, ACM.
- [MaP02] Martinez-Peiro, M., Boemo, E. I., Wanhammar, L.: Design of high-speed multiplierless filters using a nonrecursive signed common subexpression algorithm. IEEE Trans. Circuits Syst. II, 2002, vol. 49, no. 3, str. 196 – 203.
- [Mar02] Marienfeld, D., Ocheretnij, V., Gössel, M., Sogomonyan, E. S.: Partially duplicated code-disjoint carry-skip adder. Proc. of the 17th IEEE Int. Symposium on Defect and Fault Tolerance in VLSI Systems, IEEE, 2002, str. 78 – 86.
- [Mar05] Mari, D.: Organic Electronics and Polytronics. OnBoard Technology, duben 2005, s. 34 – 35.



- 
- [Mar67] Marvin, C. E., Walker, R.M.: Customizing by interconnections. Electronics, únor 1967.
- [Mas07] Maskell, D. L.: Multiplierless multiple constant multiplication. IET Circuits Devices Syst., 2007, vol. 1, no. 2, str. 175 – 180.
- [McD97] McDermott, M. W., Turner, J. E.: Configurable NAND/NOR Element. United States patent 5592107, leden 1997.
- [Mil00a] Miller, J., Thomson, P.: Cartesian Genetic Programming. Proc. of the 3<sup>rd</sup> European Conference on Genetic Programming EuroGP 2000, LNCS 1802, Springer 2000, str. 121 – 132.
- [Mil00b] Miller, J., Job., D., Vassilev, V.: Principles in the Evolutionary Design of Digital Circuits – Part I. Genetic Programming and Evolvable Machines, ročník 1, č. 1, 2000, str. 8 – 35.
- [Mil99] Miller, J.F.: On the filtering properties of evolved gate arrays. Proc. of the 1st NASA/DoD Workshop on Evolvable Hardware, IEEE Computer Society, 1999, str. 2 – 11.
- [Mot69] Motorola Semiconductor Data Book, Fourth Edition. Motorola Inc., 1969. str. IC-73.
- [Pap02] Pappu, R. S.: Physical One-Way Functions. Disertační práce. Massachusetts Institute of Technology, 2002.
- [Pra96] Pradhan, D. K.: Fault-Tolerant Computer System Design. Prentice Hall, 1996.
- [Qui01] Quintana, J. M., Avedillo, M. J., Jiménez, R.: Practical Low-Cost CPL Implementations of Threshold Logic Functions. Proc. of the 11th ACM Great Lakes Symposium on VLSI, West Lafayette, Indiana, ACM, 2001, str. 139 – 144.
- [Ruz06] Růžička, R., Sekanina, L.: Evolutionary Circuit Design in REPOMO - Reconfigurable Polymorphic Module. Proceedings of the Second IASTED International Conference on Computational Intelligence, Anaheim, ACTA Press, 2006, str. 237 – 241.
- [Ruz07a] Růžička, R.: Podpora návrhu pro snadnou testovatelnost číslicových obvodů popsaných na úrovni meziregistrových přenosů, Brno, FIT VUT, 2007, 130 stran, ISBN 978-80-214-3551-3.
- [Ruz07b] Růžička, R., Stareček, L.: Development of Building Blocks for Polymorphic Digital Circuits. Proceedings of the Work in Progress Session of 10th Euromicro DSD 2007, Linz, JKUL, 2007, str. 33 – 34.
- [Ruz07c] Růžička, R.: New Polymorphic NAND/XOR Gate. Proceedings of 7th WSEAS International Conference on Applied Computer Science, Venice, WSEAS, 2007, str. 192 – 196.
- [Ruz08a] Růžička, R., Sekanina, L., Prokop, R.: Physical demonstration of Polymorphic Self-checking Circuits. Proc. of the 14th IEEE On-Line Testing Symposium, IEEE CS, 2008, str. 31 – 36.

- 
- [Ruz08b] Růžička, R.: On Bifunctional Polymorphic Gates Controlled by a Special Signal. WSEAS Transactions on Circuits, roč. 7, č. 3, 2008, Athens, str. 96 – 101, ISSN 1109-2734.
- [Ruz08c] Růžička, R., Prokop, R.: Bifunctional NAND/NOR Gates as Building Blocks for Polytronics. Proceedings of CSE 2008, Stará Lesná, TU v Košiciach, 2008, str. 200 –207.
- [Ruz09] Růžička, R.: Dependable Controller Design using Polymorphic Counters. Proc. of 12th Euromicro Conference on Digital System Design, Los Alamitos, IEEE CS, 2009, str. 355 – 362.
- [Ruz10a] Růžička, R.: Gracefully Degrading Circuit Controllers Based on Polytronics. Proc. of 13th Euromicro Conference on Digital System Design, Los Alamitos, IEEE CS, 2010, str. 809 – 812.
- [Ruz11a] Růžička, R., Šimek, V., Sekanina, L.: Behavior of CMOS Polymorphic Circuits in High Temperature Environment. Proceedings of the 2011 IEEE Symposium on Design and Diagnostics of Electronic Circuits and Systems, Cottbus, IEEE CS, 2011, str. 447 –452.
- [Ruz11b] Růžička, R., Šimek, V.: Chip Temperature Selfregulation for Digital Circuits Using Polymorphic Electronics Principles. Proceedings of 14th Euromicro Conference on Digital System Design, Los Alamitos, ICSP, 2011, str. 205 – 212.
- [Sah10] Saha, A., Manna, N.: Digital Principles and Logic Design. Jones and Bartlett Publishers, LLC, Sudbury, 2010, str. 193 – 212.
- [Sek00] Sekanina, L., Růžička, R.: Design of the Special Fast Reconfigurable Chip Using Common FPGA. Proc. of Design and Diagnostics of Electronic Circuits and Systems - IEEE DDECS'2000, Smolenice, 2000, str. 161 – 168, ISBN 80-968320-3-4.
- [Sek06] Sekanina, L.: Evolutionary Approach to the Implementation Problem. Habilitační práce. Brno, FIT VUT, 2006, 127 stran.
- [Sek07] Sekanina, L.: Evolution of Polymorphic Self-Checking Circuits. Proc. of Evolvable Systems: From Biology to Hardware, Berlin, Springer, 2007, str. 186 – 197.
- [Sek08] Sekanina, L., Stareček, L., Kotásek, Z., Gajda, Z.: Polymorphic Gates in Design and Test of Digital Circuits. International Journal of Unconventional Computing, roč. 4, č. 2, 2008, Philadelphia, str. 125 – 142, ISSN 1548-7199.
- [Sek09a] Sekanina, L., Vašíček, Z., Růžička, R., Bidlo, M., Jaroš, J., Švenda, P.: Evoluční hardware: Od automatického generování patentovatelných invencí k sebemodifikujícím se strojům, Praha, Academia, 2009, 328 stran, ISBN 978-80-200-1729-1.
- [Sek09b] Sekanina, L., Růžička, R., Gajda, Z.: Polymorphic FIR Filters with Backup Mode Enabling Power Savings. Proc. of the 2009 NASA/ESA Conference on Adaptive Hardware and Systems, Los Alamitos, IEEE CS, 2009, str. 43 – 50.

- 
- [Sek09c] Sekanina, L., Růžička, R., Vašíček, Z., Prokop, R., Fucik, L.: REPOMO32 - New Reconfigurable Polymorphic Integrated Circuit for Adaptive Hardware. Proc. of the 2009 IEEE Symposium Series on Computational Intelligence - Workshop on Evolvable and Adaptive Hardware, Nashville, IEEE CIS, 2009, str. 39 – 46.
- [Šim10] Šimek, V., Růžička, R., Sekanina, L.: On Analysis of Fabricated Polymorphic Circuits. Proc. of the 13th Int. IEEE Symposium on Design and Diagnostics of Electronic Circuits and Systems, Vienna, IEEE CS, 2010, str. 281 – 284.
- [Smi10] Smith, C.: The ZX Spectrum ULA. ZX Design and Media, Glamorgan, 2010, str. 29 – 70.
- [Sta08] Stareček, L., Sekanina, L., Kotásek, Z.: Reduction of Test Vectors Volume by Means of Gate-Level Reconfiguration. Proc. of 2008 IEEE Design and Diagnostics of Electronic Circuits and Systems Workshop, Bratislava, IEEE CS Press, 2008, str. 255 – 258.
- [Sto00] Stoica, A., Keymeulen, D., Zebulum, R., Thakoor, A., Daud, T., Klimeck, Y., Tawel, R., Duong, V.: Evolution of analog circuits on field programmable transistor arrays. Proc. of the 2<sup>nd</sup> NASA/DoD Workshop on Evolvable Hardware, IEEE CS, červenec 2000, str. 99 – 108.
- [Sto01] Stoica, A., Zebulum, R. S., Keymeulen, D.: Polymorphic electronics. Proc. of Evolvable Systems: From Biology to Hardware Conference, volume 2210 of LNCS, Springer 2001, str. 291–302.
- [Sto02] Stoica, A., Zebulum, R., Keymeulen, D., Lohn, J.: On polymorphic circuits and their design using evolutionary algorithms. Proc. of IASTED International Conference on Applied Informatics (AI2002). Innsbruck, 2002.
- [Sto04a] Stoica, A.: Polymorphic Electronic Circuits. NASA Tech Briefs (NPO- 21213), Vol. 28, No. 4 (duben 2004), str. 38
- [Sto04b] Stoica, A., Zebulum, R. S., Guo, X., Keymeulen, D., Ferguson, I., Duong, V.: Taking evolutionary circuit design from experimentation to implementation: Some useful techniques and a silicon demonstration. IEE Proc.-Comp. Digit. Tech., 151(4), 2004, str. 295 – 300.
- [Suh07] Suh, G. E., Devadas, S.: Physical unclonable functions for device authentication and secret key generation. Proc. of Design Automation Conference, 2007, str. 9 – 14.
- [SuY08] Su, Y., Holleman, J., Otis, B. P.: A digital 1.6 pj/bit chip identification circuit using process variations. IEEE Journal of Solid-State Circuits 43, 1, 2008, str 69 – 77.
- [Tan10] Tanachutiwat, S., Lee, J. U., Wang, W., Sung, C. Y.: Reconfigurable multi-function logic based on graphene P-N junctions. Proc. of DAC 2010, str. 883 – 888.
- [Tho96] Thompson, A.: Silicon Evolution. Proc. of Genetic Programming, GP'96, MIT Press, 1996, str. 444 – 452.
- [Tho99] Thompson, A., Layzell, P., Zebulum, S.: Explorations in Design Space: Unconventional Electronics Design Through Artificial Evolution. IEEE Transactions on Evolutionary Computation, ročník 3, č. 3, 1999, str. 167 – 196.

- 
- [TI05] SN74LVC8T245, 8-bit Dual Supply Bus Transceiver with Configurable Voltage Translation and 3-state Outputs. Katalogový list, Texas Instruments, 2005, 24 stran.
- [TI94] TPS71XX, Low Dropout Voltage Regulators. Katalogový list, Texas Instruments, 1994, 38 stran.
- [Val84] Valiant, L. G.: Short monotone formulae for the majority function. *J. Algorithms*, 5(3), str. 363 – 366, 1984.
- [Ves99] Vesterbacka, M.: A new six-transistor CMOS XOR circuit with complementary output. *Proc. of 42<sup>nd</sup> Midwest Symposium on Circuits and Systems, IEEE CS*, 1999, str. 796 – 799.
- [Wad94] Wade, G., Roberts, A., Williams, G.: Multiplier-less fir filter design using a genetic algorithm. *IEE Proceedings in Vision, Image and Signal Processing*, 1994, vol. 141, no. 3, str. 175 – 180.
- [Wak00] Wakerly, J. F.: *Digital Design, Principles and Practices*. Prentice Hall, New Jersey, 2000, 3. vydání, 920 stran.
- [Wan67] Wanlass, F. M.: Low Stand-by Power Complementary Field Effect Circuitry. United States patent 3356858, prosinec 1967.
- [Wes05] Weste, N. H. E., Harris, D.: *CMOS VLSI Design: A Circuits and Systems Perspective*. Pearson/Addison-Wesley, 3. vydání, Boston 2005, 967 stran. ISBN 0-321-14901-7.
- [Xie06] Xie, Y., Hung, W. L.: Temperature-aware task allocation and scheduling for embedded multiprocessor systems-on-chip (MPSoC) design. *Journal of VLSI Singnal Processing*, vol. 45, no. 3, 2006, str. 177 – 189.
- [Xil07] XC9572XL High Performance CPLD. Katalogový list, Xilinx, 2007, 10 stran.
- [Žal08] Žaloudek, L., Sekanina, L.: Transistor-level Evolution of Digital Circuits Using a Special Circuit Simulator. *Proc. of Evolvable Systems: From Biology to Hardware*, Berlin, Springer, 2008, str. 320 – 331.
- [Zeb00] Zebulum, R., Stoica, A., Keymeulen, D.: A Flexible Model of a CMOS Field Programmable Transistor Array Targeted for Hardware Evolution. *Third Int. Conference on Evolvable Systems: From Biology to Hardware (ICES2000)*, Edinburgh, duben 2000, str. 274 – 283.
- [Zeb06a] Zebulum, R. S., Stoica, A.: Multifunctional Logic Gates for Built-In Self-Testing. *NASA Tech Briefs*, 30(3):10, 2006.
- [Zeb06b] Zebulum, R. S., Stoica, A.: Ripple Counters Controlled by Analog Voltage. *NASA Tech Briefs*, 30(3):2, 2006.