

Technická univerzita v Liberci

Fakulta mechatroniky a mezioborových inženýrských studií

Diagnostický systém jádrově založených SoC obvodů s nízkými nároky na paměť

Autoreferát disertační práce

Ing. Zbyněk MADER

Studijní program: P 2612 Elektronika a informatika
Studijní obor: 2612V045 Technická kybernetika

Pracoviště: Katedra elektroniky a zpracování signálů
Fakulta mechatroniky a mezioborových inženýrských studií
Technická univerzita v Liberci
Hálkova 6, 461 17 Liberec I

Školitel: prof. Ing. Ondřej Novák, CSc.

Rozsah disertační práce:

Počet stran: 107
Počet příloh: 3
Počet obrázků: 64
Počet tabulek: 13

Abstrakt

Tato disertační práce je zaměřena na testování logických jader v SoC obvodu. Práce předkládá návrh diagnostického systému SoC obvodu, který testuje jednotlivá jádra, přičemž systém vyhodnocuje každé jádro buď jako dobré nebo defektní.

Na základě analýzy RESPIN architektury byl vybudován testovací mechanismus tohoto systému, který splňuje požadavky normy IEEE 1500. Testovací mechanismus používá standardní testovací rozhraní jader. Jádra jsou upravena tak, aby obsahovala paralelní skenovací řetězce a odpovídala tak požadavkům RESPIN architektury. K vytvoření mechanismu testovacího přístupu je využito metodologie návrhu založené na dynamické rekonfiguraci části FPGA. Celý testovací mechanismus je řízen procesorem a řadičem, které mohou pracovat současně. Testovací data jsou uložena v komprimovaném tvaru v paměti SoC obvodu. K vytvoření komprimovaných testovacích dat je využit algoritmus zvaný COMPAS. Tento algoritmus využívá maximálního překrytí testovacích vzorků, ze kterého je schopen komprimovaná data vygenerovat. Dekompresi těchto dat probíhá s pomocí skenovacích řetězců testovacích jader, která jsou součástí SoC obvodu. Navržený diagnostický systém byl implementován společně s jádry jako systém na čipu a realizován v FPSLIC AT94K obvodu firmy Atmel.

V práci je představena nová metodika testování jader v SoC obvodu, která vede k redukci testovacích dat, testovacích pinů, doby testu a plochy na čipu.

Obsah

Abstrakt	2
1 Úvod	4
2 Metody DFT	4
3 Architektura RESPIN	5
4 Cíle disertační práce	6
5 Diagnostický systém DyRESPIN	7
5.1 Testovací rozhraní jádra	8
5.2 Mechanismus testovacího přístupu	10
6 Řízení systému DyRESPIN	11
7 Implementace	13
8 Porovnání	14
9 Závěr	16
Literatura	19
Přehled publikací	20

1 Úvod

Díky současnému vysokému stupni integrace jsme schopni umístit celý elektronický systém na jeden čip [1, 2, 3]. Takový systém je nazýván jako systém na čipu a je sestaven z jednotlivých bloků zvaných vnořená jádra. Tato jádra mohou být již vytvořena danou existující technologií; mohou se skládat z podbloků, které plní určitou funkci. Jádra je tak možné uvést na trh ve velmi krátkém čase a mohou být opětovně použita i v jiných návrzích. S nástupem SoC obvodů vzniká nový pohled na tržní hospodářství, který zavádí tzv. dodavatele jader jako subjekt, který dodává již hotová funkční jádra, a uživatele jader, který se vzhledem k jádrům chová jako spotřebitel, integruje potřebná jádra do jediného čipu a vytváří tak funkční SoC obvod. Dodavatel jader, který je často návrhářem i prodejcem svých vytvořených jader, disponuje znalostmi nejen o funkci svých jader, ale také o jejich struktuře, zatímco uživatelé jader jsou často omezeni pouze na jejich funkci. Z tohoto důvodu, kromě všech výhod, které SoC obvody přináší, vzniká několik problémů z hlediska návrhů a testování SoC obvodů [1, 2, 4, 5], které vyžadují nové strategie testování jak samotných jader, tak celého systému [6, 7, 8, 9, 10, 11].

Složitost čipu se neustále zvyšuje a zvětšuje se i hustota tranzistorů na čipu. Cena testovatelnosti celého SoC obvodu se taktéž zvyšuje, ba dokonce začíná převyšovat cenu samotné výroby SoC obvodů [12]. To znamená, že pokud cena testu obvodu není nízká, testovatelnost může mít negativní dopad i na cenu návrhu [8, 12, 13], což vede ke zvýšení celkové ceny na výrobu čipu, a tím i ceny SoC obvodů. Proto se uživatelé jader vyhýbají jakékoliv změně v návrhu dodaných jader a pracují s nimi tak, jak byly dodány. Je-li třeba upravovat návrhy jader, aby vyhovovaly různým testovacím technikám, které by rád využil uživatel jader, je celý návrh SoC obvodů založených na jádrech nevykonný, pomalý a nákladný, tím by v konečném důsledku znemožnil rychlé uvedení obvodu na trh. Cíle této disertační práce spočívají ve vytvoření diagnostického systému SoC obvodu založeném na jádrech. Tento návrh systému přispívá ke snížení celkové ceny za test obvodu díky vytvoření levného testovacího mechanismu s využitím vhodné testovací metodiky návrhu. Takový levný způsob testování SoC obvodu vyžaduje pouze základní znalosti již nabízených jader s jejich základními testovacími vlastnostmi a k testování využívá jader (paměť, procesor), která jsou již běžně součástí SoC obvodů. Nazval jsem tedy tuto disertační práci „Diagnostický systém jádrově založených SoC obvodů s nízkými nároky na paměť“.

2 Metody DFT

Za účelem rychlého uvedení jader na trh dodavatelé jader nabízí svá jádra i s vlastnostmi snadného testování. Návrhový proces, kterým se do jádra zavádí schopnost testování, se nazývá DFT (Design-for-Test). Jádro s DFT vlastnostmi nám umožní řídit a sledovat stavy vnitřních uzlů obvodů, čímž se zvýší možnosti testovatelnosti jádra.

Skenově založený DFT transformuje paměťové prvky (klopné obvody) v jádře do řetězce, kde každý klopný obvod je přepracován do skenovací buňky. Řetězec z takto vytvořených skenovacích buněk se nazývá skenovací řetězec. Se zvyšujícím počtem skenovacích buněk ve skenovacím řetězci se také zvyšuje doba nasunutí testovacího vektoru do řetězce a taktéž doba vysunutí odezvy ze skenovacího řetězce. Tento jeden skenovací řetězec je možné rozdělit do několika kratších skenovacích řetězců. V jádře pak vzniká několik paralelně uspořádaných řetězců, doba nasunutí a vysunutí testovacích dat je kratší, avšak zvyšuje se počet testovacích vstupů a výstupů jádra. Skenově založený DFT tedy

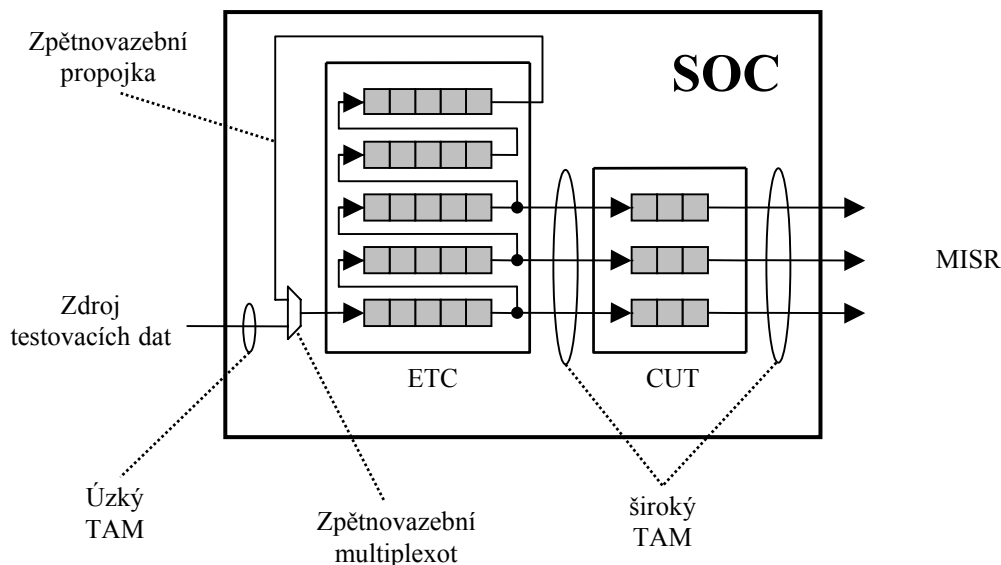
zvyšuje možnosti zpřístupnit vnitřní uzly jádra, nastavit vnitřní stavy jádra a sledovat jejich odezvu.

Systémový návrhář SoC obvodu má na starosti začlenění takto navržených jader do systému a musí navrhnout patřičnou propojovací logiku mezi jádry včetně možnosti testovatelnosti této logiky. Dále musí vybavit jádra vhodným testovacím rozhraním (wrapperem) tak, aby vyhovovaly samotnému DFT celého systému. Návrh DFT celistvého systému se kromě testovacích rozhraní jader skládá z mechanismu testovacího přístupu (TAM) a také ze zdroje testovacích dat a odezvy na test. Návrh systémového DFT je tak přenechán zcela do rukou návrháře SoC obvodu. Cílem je samozřejmě vytvořit SoC obvod co nejlevněji, nejrychleji a nejsnadněji. Je proto dobré využívat již nabízených jader se skenově založeným DFT pokud možno i s wrappery, které jsou standardizovány normou IEEE 1500. V případě potřeby jakkoliv modifikovat jádra (provést tzv. „upgrade“ jádra), by mělo být toto jádro vyměnitelné bez jakýchkoliv nároků na zásahy do jeho wrapperu a okolí. Na mechanismus testovacího přístupu jsou kladeny nároky především na jeho flexibilitu šířky pásma vzhledem k testovacímu přístupu různých jader. Posledním charakteristickým rysem DFT na úrovni systému je zdroj testovacích dat a odezva na test. Testovací data by měla využívat ke svému uložení co nejmenší kapacitu paměti, v případě použití generátoru testovacích vzorků by měl zaujímat co nejmenší plochu na čipu. Odezva na test by měla být zpracována do komprimované formy.

Samozřejmostí je použití vhodné architektury pro testování obvodu vyhovující zavedením již dříve zmíněné normy IEEE 1500. Každý výrobce si tedy může vybrat z pestré nabídky různých architektur nebo navrhnout svou vlastní architekturu vhodnou k testování jader. Na konferenci IEEE ETW 2001 byla prezentována pány Dorchem a Wunderlichem architektura zvaná RESPIN [14]. Tato architektura umožňuje rekonfigurovat zapojení jednotlivých jader obvodu tak, že každé jádro je testováno za pomoci jader okolních, ve kterých probíhá dekomprimace testovacích dat. Použitá testovací data jsou tedy v komprimovaném tvaru, není tedy kladen vysoký nárok na velikost paměti pro uložení testovacího stimulu. Dekomprese testovacích dat probíhá ve skenovacích řetězcích okolních jader, není tedy nutné vyžadovat extra hardware pro dekompresi testovacích vzorků. RESPIN architektura nevyžaduje vysokou přenosovou kapacitu pro přenos komprimovaných dat, je proto možné využít jakýkoliv procesor, který je již součástí nynějších SoC obvodů. RESPIN architektura je kompatibilní s normou IEEE 1500. Tuto architekturu jsem s přihlédnutím na všechny její výhody zvolil pro náš diagnostický systém

3 Architektura RESPIN

RESPIN architektura je vhodná pro testování vnořených jader s pravidelnou strukturou, kde kombinační obvody jsou důsledně odděleny od paměťových členů. Taková jádra již byla zmíněna v předcházející kapitole v souvislosti s jejich skenově založeným DFT. RESPIN architektura v podstatě využívá pro dekomprimaci testovacích vzorků tyto paměťové členy v testovacím jádře, které jsou uspořádány do skenovacího řetězce. Rovněž testovací přístup testovaného jádra vyžaduje vnitřní uspořádání paměťových členů do skenovacích řetězců. Je tedy zřejmé, že jádra vyhovující RESPIN architektuře musí obsahovat skenově založený DFT. Příklad RESPIN architektury je vyobrazen na obrázku 1.



Obrázek 1: RESPIN architektura

Hlavní myšlenka RESPIN architektury spočívá v tom, že testované jádro zvané CUT (Core Under Test) se testuje za pomoci jiného testovacího jádra zvaného ECT (Embeded Tester Core), podle potřeby může být na místě jednoho ETC použito více ETC. Testovací přístup k jednotlivým jádrům zajišťuje TAM. Do ETC je připojen zvnějšku obvodu úzký TAM, který přivádí do jádra silně komprimované testovací vzorky. Přenosová kapacita úzkého TAMu je nízká. Spojení mezi ETC a CUT je zajištěno širokým TAMem. Ten přenáší již dekomprimované testovací vzorky paralelním přenosem a jeho přenosová kapacita je vysoká. Odezvy na testovací vzorky mohou být akumulovány ve vícevstupovém příznakovém analyzátoru (MISR), který zde zaujímá funkci zpracování odezvy testovaného jádra na jednotlivé testovací vzorky. Do MISRu jsou tyto odezvy rovněž přenášeny po širokém TAMu paralelním přenosem. Tímto je dosaženo vynikajících vlastností testování jádrově založených SoC obvodů při použití vysoce komprimovaných testovacích dat.

4 Cíle disertační práce

Hlavním cílem této disertační práce je navrhnout diagnostický systém pro SoC obvody, který je schopen testovat vnořená logická jádra. Je třeba vytvořit vhodný DFT návrh na úrovni systému, přičemž má být brán zřetel na malé hardwarové navýšení v SoC obvodu tímto diagnostickým systémem. Je tedy třeba zvolit vhodnou testovací metodu a vytvořit úsporné testovací okolí jader. Celková doba testu obvodu má být co nejkratší. Dále je kladen důraz na využití co nejmenší kapacity paměti, ve které budou uloženy testovací vzorky. Požadavek je také kladen na rozhraní SoC obvodu, které má být standardizováno. Tímto bude mimo jiné zajištěna možnost použití stávajících automatických testovacích zařízení (ATE). Dalším znakem diagnostického systému má být univerzálnost jeho použití v praxi, má se tedy uplatnit nejen v procesu testování bezprostředně po výrobě ASIC obvodů, ale i v diagnostice SoC obvodu vytvořeném v FPGA, kdy může být kdykoliv ověřena bezporuchovost jader. Tímto bude přispěno nejen k rychlému uvedení SoC obvodů na trh, ale i k jeho nízké ceně se zárukou vysoké spolehlivosti. Abychom dosáhli dobrých výsledků v návrhu diagnostického systému, musí se splnit tyto cíle:

- transformace jader do testovatelné podoby
- optimalizace testovací architektury
- návrh mechanismu testování podle normy IEEE 1500
- návrh řízení mechanismu testování z hlediska malé plochy na čipu a krátké doby testu
- implementace mechanismu a řízení na SoC obvodu
- praktická realizace SoC obvodu s diagnostickým systémem
- provést rozbor výsledků a srovnání s jinými srovnatelnými metodami

Předpokladem detekce a lokalizace defektního jádra našim diagnostickým systémem v SoC obvodu je použití plně skenových jader.

Je zde třeba důrazně říci, že tato disertační práce se nezabývá tvorbou testovacích dat ani způsobem jejich generování. Tímto problémem se zabývá jiná skupina našeho týmu. V této disertační práci pracuji s již předpřipravenými deterministickými testovacími daty v komprimovaném tvaru.

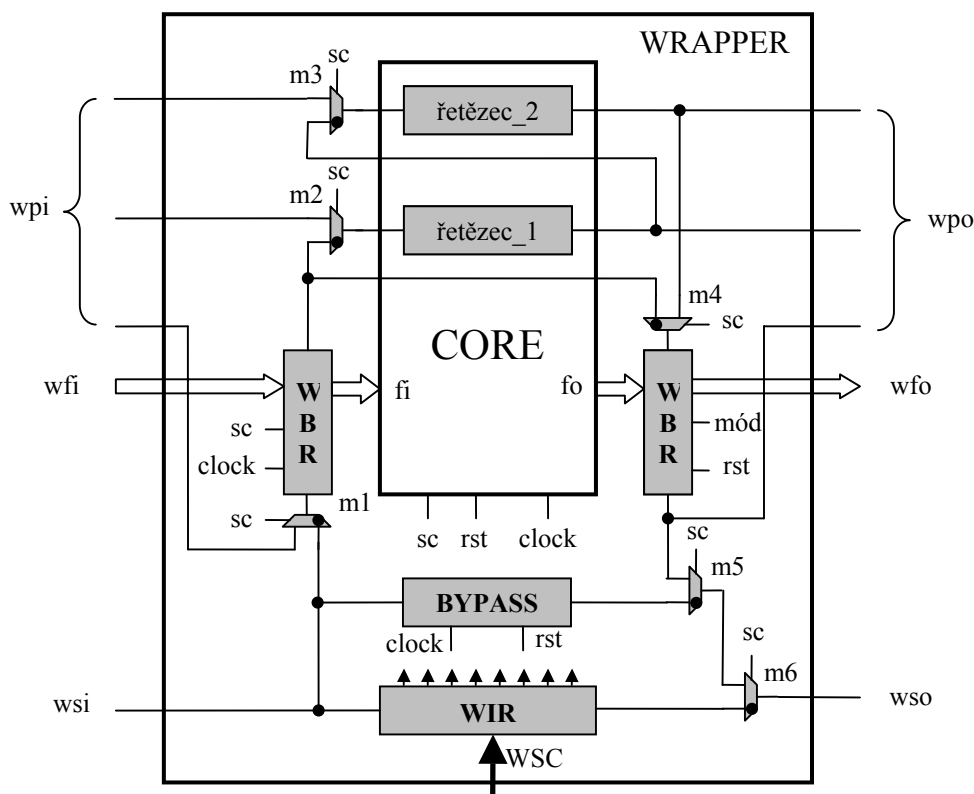
5 Diagnostický systém DyRESPIN

Jak je již zmíněno v předcházející kapitole 2, náš diagnostický systém je založen na RESPIN architektuře. Lze tedy tvrdit s přihlédnutím na tuto RESPIN architekturu, že testovací data budou tvořena testovacím stimulem v komprimovaném tvaru. K vytvoření komprimovaných testovacích vzorků a jejich přerovnání tak, aby vyhovovaly RESPIN architektuře, se dá výhodně použít již vytvořený kompresní algoritmus COMPAS [15]. Přes všechny výhody RESPIN architektury je to také jeden z důvodů, proč jsem diagnostický systém založil na této architektuře. Algoritmus COMPAS je založen na základě překrývání testovacích vektorů [16]. Při kompresi dat algoritmem COMPAS nedochází ke ztrátě informace, jedná se tedy o bezeztrátovou kompresi. Způsob generování komprimovaných testovacích posloupností ale není předmětem této disertační práce, zabývá se jím jiná skupina našeho týmu. V praxi bude celý postup přípravy návrhu SoC obvodu vypadat následovně. Dodavatelé jader dodají uživateli požadovaná jádra se skenově založeným DFT včetně testovacích vektorů. Dodané testovací vektory jsou v nekomprimovaném tvaru. K jejich převodu do komprimovaného tvaru se použije program COMPAS. Ten však pracuje na základě netlistu daného obvodu, který nemusí být dodavatelem jader zpřístupněn. V případě, že uživatel jader nemá netlist jádra k dispozici, komprimaci testovacích vektorů provádí výrobce jader na základě poskytnutí programu COMPAS. V opačném případě se komprimace testovacích vektorů provádí přímo u uživatele jader.

V této kapitole 5 bude postupně popsán celý návrh diagnostického systému pro testování jader, který jsem nazval DyRESPIN systém. Pojmenování systému „DyRESPIN“ vplyne dále z kontextu této kapitoly. V kapitole 2 byly uvedeny tři obecné znaky charakterizující diagnostický systém SoC obvodu: 1) Zdroj testovacích dat a odezva na test, 2) wrapper a 3) TAM. Pokud neuvažuji v bodě 1 zdroj testovacích dat, bude se popis celého systému týkat především testovacího rozhraní jádra (wrapperu), mechanismu testovacího přístupu (TAMu) a komprese odezev na jednotlivé testovací kroky jádra CUT.

5.1 Testovací rozhraní jádra

Testovací rozhraní jádra neboli wrapper musí vyhovovat RESPIN architektuře a zároveň pracovat v módech, které splňují kritéria testování v této architektuře. Tvůrci RESPIN architektury navrhují wrapper, jehož součástí je zpětnovazební multiplexor včetně zpětnovazební propojky (viz obrázek 1). Toto řešení však značně omezuje možnosti testování v RESPIN architektuře zvláště při použití více testovacích jader a také výrazně komplikuje funkčnost wrapperu včetně jeho módů vzhledem ke standardu IEEE 1500. My přicházíme s takovým návrhem wrapperu do našeho DyRESPIN systému, že z wrapperů vyčleníme zpětnovazební multiplexor a propojku zpětné vazby a tyto umístíme do TAMu. Struktura takového wrapperu je vyobrazena na obrázku 2. Tento wrapper vyhovuje standardu IEEE 1500, je zajištěn paralelní testovací přístup pro přenos dekomprimovaných testovacích vzorků a odezev na ně, přísun komprimovaných vzorků se provádí přímo ze zdroje rovněž paralelním testovacím přístupem. Povinné módy wrapperu Bypass a Extest vyhovují standardu, jako třetí povinný vnitřní testovací mód byl zvolen paralelní testovací Intest mód. Pro DyRESPIN systém je potřebný zpětnovazební multiplexor se zpětnovazební propojkou začleněn do TAMu. TAM je dle normy IEEE 1500 uživatelsky definován, je tedy zcela na uživateli, jakým způsobem je vytvořen a co bude obsahovat. TAMu se podrobně věnuje celá kapitola 5.2. Zjednodušeně lze však říct, že TAM je tvořen dynamicky rekonfigurovatelným blokem v FPGA, který zajišťuje všechna propojení mezi paralelními porty wrapperů včetně zpětnovazební propojky a zpětnovazební multiplexor je součástí jedné buňky v tomto dynamicky rekonfigurovatelném bloku. Tímto je vysvětlen název testovacího systému DyRESPIN, jenž využívá k testování jádra RESPIN architekturu za pomoci dynamicky rekonfigurovatelného TAMu.



Obrázek 2: Testovací rozhraní jádra

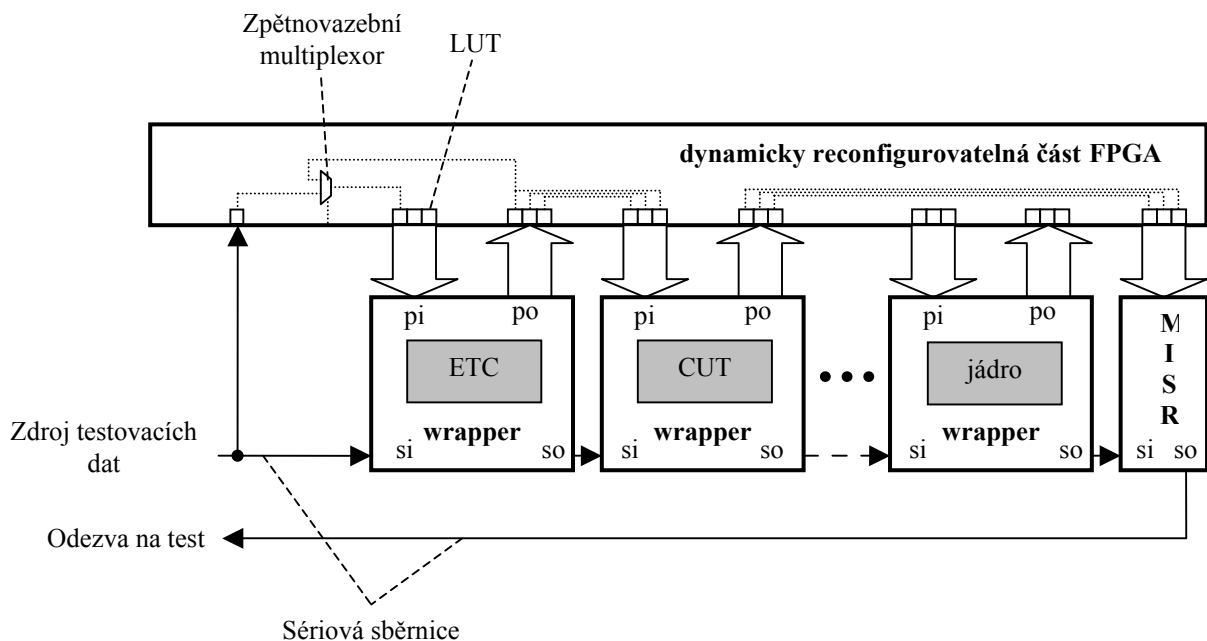
Instrukční registr značený jako WIR plní ve wrapperu úlohu řídicí jednotky. Do instrukčního registru je možné přivést data přes sériový vstup *si*. Data charakterizují instrukci, která je do instrukčního registru zapsána. Pomocí této zapsané instrukce a řídicích signálů WSC dokáže instrukční registr vytvořit řídicí signály pro další registry a multiplexory obsažené ve wrapperu. Wrapper je zavedením instrukce uveden do určitého módu a pracuje v určitých režimech. Wrapper v Bypass módu (charakterizován instrukcí WS_BYPASS) propojuje jádro ke svému okolí a ponechává jej plně funkční. Wrapper v sériovém Extest módu (charakterizován instrukcí WS_EXTEST) umožní otestovat propojky mezi dvěma jádry. Wrapper v paralelním Intest módu (charakterizován instrukcí Wx_INTEST) zajistí testovatelnost CUT jádra. Instrukční soubor wrapperu se v první řadě skládá ze tří povinných instrukcí zmíněných v předcházejícím odstavci. Wx_INTEST je jakákoliv instrukce, která zajistí testování jádra a je zcela na uživateli, jak si ji nadefinuje. S výhledem na využití RESPIN architektury k testování jádra je jako třetí povinná instrukce zvolena WP_INTEST, která říká, že jde o testování jádra za pomoci paralelního testovacího přístupu se zapojením vnitřních skenovacích řetězců jádra. Kromě těchto povinných instrukcí bylo třeba uživatelsky nadefinovat instrukci WS_CIRCULAR vhodnou pro wrapper systému DyRESPIN. Pomocí této instrukce se uvádí wrapper do cirkulárního módu. Konfigurace wrapperu v cirkulárním módu pouze propojuje WBR registr s vnitřními skenovacími řetězci jádra sériově v pořadí: vstupní WBR → řetězec_1 → řetězec_2 → výstupní WBR. Do takto vzniklého sériového řetězce je zajištěn sériový přístup přes paralelní testovací port do vstupu vstupního WBR registru a dále je zajištěn paralelní přístup rovněž přes paralelní testovací port z výstupů všech registrů obsažených v jejich sériovém propojení vyjma vstupního WBR. Tím, že wrapper neobsahuje zpětnou vazbu, je možné na výstup výsledného sériového řetězce navázat další řetězce wrapperů s jádry ETC. Cirkulární mód wrapperu daný instrukcí WS_CIRCULAR je však odkázán na sériové propojení řetězců s pevně daným vstupem. Z důvodů větší flexibility v možnostech zvolení vhodné délky řetězce v ETC jádře, kterému odpovídá týkající se řetězec v CUT jádře, je v systému DyRESPIN uživatelsky definována další instrukce WP_CIRCULAR. Tato instrukce vychází z již dříve popsání instrukce WS_CIRCULAR, avšak propojení řetězců ve wrapperu není sériové, ale každý řetězec ve wrapperu je samostatně připojen ke svému paralelnímu vstupu a výstupu. Operativnost wrapperů ve všech zmíněných módech ukazuje tabulka 1.

instrukce	m1	m2	m3	m4	m5	m6	mód WBR
WS_BYPASS	X	X	X	X	1	0	funkční
WS_EXTEST	1	X	X	1	0	0	vnější testovací
WP_INTEST	0	0	0	1	X	X	vnitřní testovací
WS_CIRCULAR	0	1	1	0	X	X	vnitřní testovací
WP_CIRCULAR	0	0	0	1	X	X	vnitřní testovací

Tabulka 1: Nastavení multiplexorů a WBR ve wrapperu

5.2 Mechanismus testovacího přístupu

Hlavní úlohou TAMu je postarat se o přenos testovacích vzorků na čipu. Je tedy využít jednak pro přenos stimulu ze zdroje testovacích vzorků do CUT a jednak pro přenos odezvy na test z CUT do modulu (zde MISRu) pro jeho následné zpracování. Podle definice z normy IEEE 1500 je TAM implementován přímo na čipu. Existuje několik způsobů a mnoho architektur, jak vytvořit TAM. K jádru je často připojen stejný typ TAMu jak pro přivedení stimulu, tak pro odvedení odezvy, ale nemusí tomu tak být. Především v RESPIN architektuře jsou vyžadovány různé typy TAMů (viz. obrázek 1). Pro vytvoření výkonného TAMu pro RESPIN architekturu bylo třeba se zpětně podívat na všechny výhody a nevýhody struktur TAMů a testovacích sběrnic doposud zveřejněných a prezentovaných. Také bylo třeba nahlédnout do oblasti sítí na čipu. V souvislosti se sítěmi na čipu byla uvažována i oblast dynamické rekonfigurace FPGA, kde bylo poukazováno na možná řešení dynamického propojení rekonfigurovatelných modulů na dynamicky rekonfigurovatelných FPGA.



Obrázek 3: TAM pro DyRESPIN systém

Z takto získaných poznatků byl navržen mechanismus testovacího přístupu pro DyRESPIN systém, který je vykreslen na obrázku 3. Pro lepší orientaci v propojkách TAMu jsou do obrázku doplněny wrappery a MISR. Celý mechanismus testovacího přístupu je složen ze dvou technologicky odlišných částí, přičemž každá tato část je určena pro přenos určitého typu dat. Jedna část TAMu je tvořena klasickou sériovou jednobitovou sběrnicí propojující sériové vstupy *si* a sériové výstupy *so* všech wrapperů a MISRu. Po této sběrnicí jsou do instrukčních registrů přenášeny instrukce určené pro všechny wrappery. Pomocí této sběrnice se také odvádí výsledný příznak z MISRu po ukončení testu jádra. Technologicky je sběrnice tvořena pevným a neměnným sériovým propojením mezi porty *si* a *so* tak, jak je patrné v obrázku.

Druhou část TAMu tvoří paralelní sběrnice, které patřičně propojují pouze paralelní testovací porty wrapperů s ETC a CUT jádry a paralelní port MISRu. Přenos komprimovaných testovacích dat ze zdroje testovacích dat probíhá po jednobitové propojce. Tato komprimovaná data vstupují do ETC vstupním paralelním portem přes dynamicky

rekonfigurovatelné FPGA, přičemž je z tohoto vstupního paralelního portu využít pouze jednobitový vstup. Fyzicky jsou tyto sběrnicce, dá-li se těmto propojkám říkat sběrnicce, vytvořeny v FPGA programovatelném poli. Každý paralelní port wrapperu v závislosti na jeho bitové délce je připojen k určitému počtu LUTů (jeden LUT odpovídá jedné propojce vedoucí z paralelního portu). Další skupina LUTů je vyhrazena pro MISR a jeden samostatný LUT je vyhrazen pro propojku ze sériové sběrnicce. V závislosti na tom, jaké jádro je nakonfigurováno jako ETC a CUT, rekonfiguruje se toto programovatelné pole tak, že se vytvoří mezi LUTy propojení, jak to vyžaduje RESPIN architektura. Současně se musí vytvořit v tomto FPGA poli zpětnovazební multiplexor, který je umístěn mezi příslušné propojky. Jádra, která nebyla nastavena jako ETC nebo CUT, jsou nastavena do Bypass módu a výstupy jejich přiřazených LUTů zůstávají v rozpojeném stavu. Tyto jádra mohou i nadále pracovat ve funkčním režimu. Proces, který vytváří paralelní sběrnicci, se nazývá dynamická rekonfigurace FPGA a musí být vykonán vždy před každým započatým testováním určitého jádra CUT. V případě, že jádra se svými wrappery a další komponenty jsou součástí programovatelného pole FPGA, je třeba dynamicky rekonfigurovat pouze část obvodu FPGA vyhrazenou pro paralelní sběrnicce.

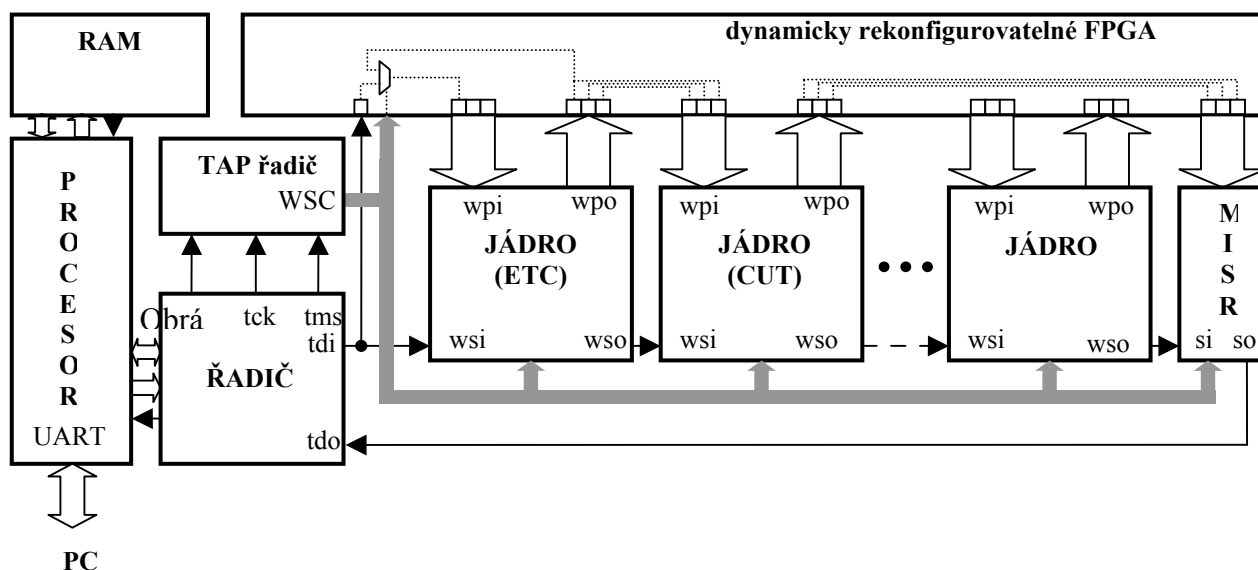
Zejména v současnosti, kdy se cena velkých FPGA obvodů dorovnává ceně ASIC obvodů, začínají se mnohé komponenty a jádra dodávat v podobě určené pro naprogramování do FPGA. Dynamické rekonfigurace se tedy už nevyužívá pouze za účelem záměny již nepotřebného nebo defektního funkčního bloku za nový funkční blok v dané aplikaci nebo při úpravě v rozmístění funkčního bloku v důsledku vzniklé poruchy v jedné buňce FPGA, ale také ke včlenění testovacích propojení před započítím testu, což je právě náš případ. Další výhodou využití dynamické rekonfigurace v TAMu v oblasti testování je výrazné zjednodušení řízení TAMu. V našem případě jsou zapotřebí tzv. bitstreamy. Každá konfigurace propojek je charakterizována jedním bitstreamem. Tyto bitstreamy jsou v podstatě jednoduché soubory bitů a jsou uloženy v paměti. Řízení dynamické rekonfigurace v přeměnu těchto bitstreamů na danou konfiguraci části TAMu zajišťuje svými nástroji, architekturou obvodu a svými postupy výrobce FPGA obvodů s vlastností dynamické rekonfigurace.

6 Řízení systému DyRESPIN

Hlavní úlohou řízení systému je, jak vyplývá z funkce RESPIN architektury, plynule zajistit dodávku jednotlivých bitů komprimovaných testovacích vzorků do ETC jádra tak, aby celý proces testu SoC obvodu, nebo jeho částečný test, běžel plynule a proběhl v co nejkratším čase. Další důležitou vlastností řídicího mechanismu je zajistit kompatibilitu s okolím. To znamená, že vstupy a výstupy řídicího systému by se měly držet již zavedených norem. Samozřejmě také HW náročnost řídicího systému je požadována co nejmenší a je vhodné v systému použít již vyzkoušené a praxí osvědčené prvky. Navrhnul jsem tedy řídicí systém, který těmto nárokům vyhovuje, jeho blokové schéma je vyobrazeno na obrázku 4.

Řízení DyRESPIN systému se skládá ze čtyř modulů, 1) TAP (Test Access Port) řadiče, 2) vlastního řadiče systému, 3) procesoru a 4) paměti RAM. Vstupní rozhraní TAP řadiče přijímá hodinový signál *tck* a řídicí signál *tms*, oba tyto signály jsou normovány standardem IEEE 1149.1. Výstupní rozhraní TAP řadiče vytváří skupinu WSC signálů normovaných podle standardu IEEE 1500. Pomocí WSC signálů jsou řízeny všechny wrappery, MISR a zpětnovazební multiplexor v TAMu. Vlastní řadič systému je ovládán povely z procesoru přes řídicí sběrnicci a potvrzovacím signálem řadič posílá zpět procesoru informaci o přijatém povelu. Přenos dat mezi procesorem a řadičem probíhá přes

jednabitovou sběrnici. Na základě přijatého povelu řadič vytváří souslednost řídicích signálů pro TAP řadič tak, aby testování jader proběhlo co nejrychleji. Současně se řadič stará o přenos dat z *tdi* a *tdo* portů na datovou sběrnici procesoru. Řadič je vytvořen HW návrhem, protože je na něj kladen vysoký nárok na rychlost. Je třeba si uvědomit, že řídicí signály WSC wrapperů v průběhu testování jádra se mění s každým hodinovým cyklem, proto i změny *tck* a *tms* signálů musí být taktovány na stejné frekvenci. Procesor je z hlediska řízení nadřazen řadiči. S řadičem procesor komunikuje již zmíněnou datovou a řídicí sběrnici, potvrzení o přijatém povelu řadičem je v procesoru vyvoláno přerušení. Procesor oproti řadiči vykonává složitější procesy spojené s testováním, které však nejsou tak časově náročné. Procesor je schopen komunikace s vnějším SoC obvodu, zde s PC, a také je mu umožněn přístup do paměti RAM. Poslední modul tvoří samotná paměť RAM, která slouží k uchování především komprimovaných dat a bitstreamů pro dynamicky rekonfigurovatelnou sběrnici.



Obrázek 4: Řízení DyRESPIN systému

Řídicí systém je schopen provádět diagnostiku jader, přitom se drží zavedených standardů. Řadič DyRESPIN systému je automat vybaven čítačem, který vytváří z přijatých řídicích slov z vyšší úrovně sled po sobě jdoucích bitů pro řídicí signál *tms*. Procesor ovládá průběh testování jádra, řídí tedy sled jednotlivých fází testování počínaje dynamickou rekonfigurací FPGA přes fázi nastavení wrapperů a samotného testování jádra a konče fází vyčtení výsledného příznaku z MISRu. Procesor a DyRESPIN řadič mohou pracovat současně. Počítač komunikuje s procesorem přes UART rozhraní a s pomocí vhodného uživatelského rozhraní je možné nejen spouštět test určitého jádra, ale také ovládat jednotlivé kroky testování jádra, zadávat testovací data do systému a vyčíst výsledek testu.

7 Implementace

DyRESPIN systém jsem implementoval na obvodu firmy Atmel FPSLIC AT94K [17]. Tento obvod se skládá z FPGA, které má vlastnost dynamické rekonfigurace, z 8-bitového RISC procesoru AVR a z paměti SRAM, která má kapacitu 36k bajtů. K AVR procesoru je mimo jiné připojena fixní periférie UART rozhraní potřebná pro komunikaci diagnostického systému s PC. Paměť SRAM může sloužit k uchování komprimovaných testovacích vzorků s bitstreamy pro dynamickou rekonfiguraci. Do této paměti je možné přistupovat jak z procesoru, tak z FPGA. FPGA má kapacitu 40 tisíc hradel a 2864 registrů, což není mnoho, ale na vytvoření prototypu DyRESPIN diagnostického systému včetně jednoduchých jader je tato kapacita dostačující.

Obecně se tedy můžeme na tento obvod dívat jako na systém na čipu s diagnostickým vybavením. V FPGA jsou naprogramovány všechny wrappery s příslušnými jádry, MISR a oba řadiče řídicího systému, je zde také vytýčená oblast pro dynamicky rekonfigurovatelnou část TAMu. AVR procesor je využit, jak již bylo uvedeno v předcházející kapitole, ke spuštění testu, čtení dat z paměti, zápis dat do paměti a dále pro řízení částečné rekonfigurace FPGA. AVR procesor je v FPSLIC obvodu vytvořen v podobě hardwarového jádra. K jeho naprogramování jsem využil volně dostupný překladač GNU a programovací jazyk C určený pro AVR procesory. Komponenty určené pro FPGA jsem popsal ve VHDL jazyce.

Realizace proběhla na stavebnici (angl. Kit) ATSTK94, která užívá právě tento dříve zmíněný obvod FPSLIC AT94K a je dodávána s vhodnými softwarovými nástroji sloužícími k simulaci a syntéze, k vytvoření všech konfiguračních bitstreamů pro FPGA a také z nástroje, pomocí kterého je možné vše zavést do obvodu.

Byly provedeny dva experimenty. V prvním experimentu jsem implementoval celý diagnostický systém včetně tří jader do FPGA a využil jsem hardwarové jádro AVR procesoru. Jako jádra mi posloužily tři ISCAS benchmarkové obvody (S298, S382 a S444). Jako testované CUT jádro jsem v experimentu zvolil obvod S444, testovací ETC jádra jsou zapotřebí dvě, zvolil jsem tedy obvod S382 jako ETC₁ a obvod S298 jako ETC₂. K takto přiděleným jádrům jsem navrhnul komponentu nazvanou TAM_BAC, ve které jsou popsány propojky mezi těmito jádry včetně zapojení zpětnovazebního multiplexoru. Pro tuto konfiguraci jsem vygeneroval statický bitstream. Podobným způsobem jsem navrhnul druhou komponentu nazvanou TAM_ABC, avšak zde jako ETC₁ jsem zvolil obvod S298 a jako ETC₂ obvod S382. Z této druhé komponenty jsem opět vygeneroval druhý statický bitstream. Tím jsem vytvořil podmínky k principiálnímu provedení jednoduché diagnostiky SoC obvodu, která vyhodnocuje CUT jádro (tedy S444 obvod) jako dobré nebo poruchové za pomoci dvou různých testů. Velikost hardwaru takového diagnostického systému včetně jader zaujímala 34% celkové kapacity FPGA. Dynamická rekonfigurace TAMu trvala několik tisíc taktovacích cyklů procesoru, při 4MHz hodinové frekvenci procesoru doba rekonfigurace TAMu trvala méně než 1 ms. Velikost jednoho statického bitstreamu pro dynamickou rekonfiguraci TAMu činí přibližně 2k bajtů paměti SRAM.

Druhý experiment jsem provedl za použití tří větších jader, konkrétně jsem použil jak pro CUT, tak pro ETC jádra vždy ten samý ISCAS obvod S1423. Vytvoření statických bitstreamů bylo podobné, jako v předcházejícím experimentu. Kapacita FPGA v tomto případě byla využita na 73%. Je vidět, že velikost obvodu FPSLIC AT94K pro větší jádra by již byla nepostačující, nemělo by tedy smysl další experimenty provádět. Nicméně funkčnost diagnostického systému DyRESPIN jsem ověřil na těchto dvou výše popsaných experimentech. Výsledný příznak jednotlivých testů jader v tomto systému na čipu

realizovaném v FPSLIC obvodu získaný počítačem se shodoval s provedenou funkční simulací tohoto obvodu za použití tzv. „testbench“ souboru se stejnými vstupními daty.

8 Porovnání

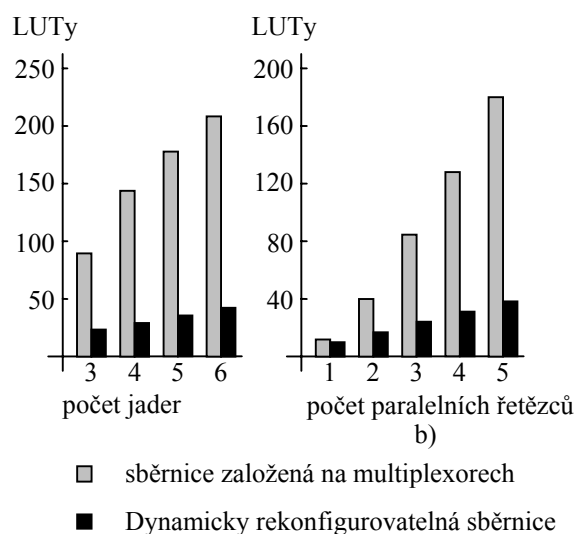
Skenově založený samočinně testovací vestavěný systém je v současné době nejvhodnějším řešením diagnostického systému integrovaného v SoC obvodu. Požadavky kladené na takový diagnostický systém jsou:

- Mechanismus testovacího přístupu má být jednoduchý a nenáročný na řízení, má být optimální k testovací architektuře návrhu.
- Wrapper má vyhovovat vzniklé normě IEEE 1500 určené pro testování jader v SoC obvodech.
- Velikost paměti pro uchování testovacích vzorků má být co nejmenší.
- Speciální hardwarové bloky sloužící k dekomprimaci testovacích vzorků, k vyhodnocení odezev na testy nebo jako generátor testovacích vzorků mají zaujímat co nejmenší plochu na čipu.
- Celková doba potřebná pro diagnostiku SoC obvodu má být co nejkratší.

Pokud tedy chceme srovnávat náš diagnostický systém DyRESPIN s jinými metodami a technikami, musíme se na něj dívat jako na celek, tzn. jako na diagnostický systém, jehož nedílnou součástí je kompresní algoritmus COMPAS sloužící ke generaci komprimovaných testovacích vektorů. Takto pojatý systém lze srovnávat s ostatními nejenom v oblasti hardwaru potřebného pro diagnostiku, ale i časové oblasti, kdy je posuzována doba potřebná k otestování jádra. Také je důležité množství bitů potřebných k otestování jádra.

Mechanismus testovacího přístupu

Pro srovnání TAMů je nejvhodnější pracovat s architekturami systému podobných vlastností. Vezměme tedy RESPIN architekturu, která pro své propojení jader mezi sebou a MISRem využívá odlišné návrhy sběrnic. V práci jsem zrekapituloval, že jediná vhodná univerzální sběrnice pro RESPIN architekturu je sběrnice založená na multiplexorech. Vytvořil jsem tedy návrhy různých sběrnic v FPGA obvodu založených na multiplexorech v závislosti na počtu paralelních řetězců v jádrech a počtu jader v SoC obvodu. Výsledky zobrazují grafy znázorněné na obrázku 5, kde je vidět množství LUTů potřebných k vytvoření



sběrnice založené na multiplexorech a dynamicky rekonfigurovatelné sběrnice. V grafu a) uvažujeme různý počet jader v SoC obvodu v závislosti na množství LUTů, všechna jádra mají konstantní počet paralelních řetězců. Se vzrůstajícím počtem jader v obvodu rapidně roste počet LUTů v multiplexorově založené sběrnici. Oproti tomu v dynamicky rekonfigurovatelné sběrnici s každým dalším jádrem přidaným do obvodu vzrůstá v této sběrnici počet LUTů minimálně. V grafu b) je situace obdobná, obvod má však stálý počet jader roven třem avšak mění se v nich počty paralelních řetězců.

Mechanismus řízení

Zavedením normy IEEE 1500 týkající se testování vnořených jader by měla být snaha všech výrobců SoC obvodů používat normovaný wrapper. Každý takový wrapper má povinný WSP port, přes který sériově putují datové signály, přes tento port je také wrapper řízen řídicími signály WSC. Pokud diagnostický systém obsahuje hardwarové bloky, jako jsou např. generátor náhodných testů, dekompresor nebo příznakový analyzátor, musí být rovněž tyto bloky vybaveny stejným portem jako wrappery, aby byla zajištěna správná synchronizace jednotlivých testovacích kroků. Pro různá řešení diagnostických systémů musí být vytvořen vždy podobný mechanismus řízení. Z tohoto důvodu se nadále nebude uvažovat nutné hardwarové navýšení o tento řídicí mechanismus, protože pro různá řešení zaujímá přibližně stejnou plochu na čipu.

Metoda testování

K našemu diagnostickému systému DyRESPIN je možné připojit stávající ATE zařízení disponující rozhraním dle standardu IEEE 1149.1. Od ATE se vyžaduje pouze jeden vstupní a jeden výstupní datový kanál. Při použití takového ATE zařízení lze porovnat náš systém se známou metodou testování zvanou „Boundary Scan“ (zkráceně BS). Tato metoda se používá pro testování propojek mezi integrovanými obvody umístěnými na desce s plošnými spoji, ale může být aplikována i v SoC obvodech k testování jader. Testovací systém založený na BS testování potřebuje k vykonání testu mnohem delší dobu oproti našemu testovacímu systému, protože testování jádra v našem systému může probíhat na vyšší testovací frekvenci, než ATE. Tohoto výsledku je docíleno díky vlastnosti RESPIN architektury pracující s komprimovanými testovacími vzorky, které mohou přicházet do systému pomaleji oproti testovací metodě BS pracující s nekomprimovanými vzorky. Další zkrácení doby testu je v důsledku použití paralelního přenosu dekomprimovaných dat mezi ETC a CUT našeho systému. BS metoda testování žádného paralelního přenosu dat nevyužívá.

Požadavky na paměť a doba testu CUT

Nejjednodušším řešením, jak generovat testovací vzorky, je použít generátor náhodných testovacích vzorků např. LFSR registr, celulární automat, Johnsonův čítač atd. Tyto generátory nekladou žádné nároky na paměť, protože žádné paměti potřeba není. V případě umístění generátoru na čip SoC obvodu vzniká pouze nárok na hardware samotného generátoru. Problém ale nastává v dlouhé době testování takovým generátorem, proto vznikají tzv. generátory smíšených testovacích vzorků pracující jak s náhodnými, tak s deterministickými testovacími vzorky. Jiný typ zařízení, které vytváří testovací vzorky je dekompresor. Dekompresor pracuje většinou s deterministickými komprimovanými testovacími daty, přijímá tedy tyto data v komprimovaném tvaru a vytváří z nich dekomprimované testovací vektory. V podstatě lze na dekompresor pohlížet jako na generátor testů, který využívá deterministických testovacích dat v komprimovaném tvaru. Ať se jedná o jakoukoliv techniku, která k tvorbě testovacích vzorků používá deterministické testovací vzorky, musí být tyto vzorky uloženy v paměti.

Na vybraných ISCAS obvodech dobře známých kompresních metod a kompresního algoritmu COMPAS bylo provedeno srovnání a sestavena tabulka 2 s počty bitů potřebných k provedení úplného testu obvodu. Z výsledků pak můžeme vidět, že pro dané ISCAS obvody DyRESPIN systém ve spojitosti s COMPAS algoritmem využívá nejmenší kapacitu paměti pro uložení komprimovaných testovacích dat.

Název obvodu	MinTest [18]	Stat. Coding [19]	LFSR Reseeding [20]	Illinois Scan [21]	FDR Codes [22]	EDT [23]	RESPIN ++ [24]	COMPAS [15]
	Počet bitů	Počet bitů	Počet bitů	Počet bitů	Počet bitů	Počet bitů	Počet bitů	Počet bitů
s13207	163,100	52,741	11,285	109,772	30,880	10,585	26,004	4,024
s15850	58,656	49,163	12,438	32,758	26,000	9,805	32,226	7,737
s38417	113,152	172,216	34,767	96,269	93,466	31,458	89,132	21,798
s38584	161,040	128,046	29,397	96,056	77,812	18,568	63,232	6,675

Tabulka 11: Požadavky na paměť v závislosti na různých kompresních metodách

Také doba testování je kratší, než u jiných metod, přičemž počet bitů nezbytných k provedení všech testů obvodu je přibližně srovnatelný, jak ukazuje další tabulka 3.

Název obvodu	folding counters [25]			RESPIN [14]		Compas [15]	
	Počet vstupů	Počet bitů	Počet hodinových cyklů	Počet bitů	Počet hodinových cyklů	Počet bitů	Počet hodinových cyklů
c7552	207	9E3	4E6	-	-	5E3	1E6
c2670	233	5E3	4E6	-	-	4E3	8E5
s13207	700	25E3	2E7	4E3	2E8	4E3	3E6
s15850	611	18E3	2E7	6E3	2E8	7E3	4E6
s38417	1,664	136E3	2E8	3E4	5E9	2E4	3E7
s38584	1,464	33E3	6E7	6E3	2E8	6E3	9E6

Tabulka 3: Porovnání doby testování při přibližně stejném počtu testovacích bitů

Hardwarové nároky

DyRESPIN systém se vyznačuje tím, že používá výhradně deterministická data, nepotřebuje tedy žádný generátor náhodných testovacích vzorků. Taktéž k dekomprimaci testovacích vzorků nepotřebuje žádný speciální hardware, protože dekomprimace probíhá v paralelních řetězcích jiného testovacího jádra. Hardwarová úspora je značná i na výše zmíněném mechanismu testovacího přístupu, který mimo jiné nemá ani řídicí mechanismus.

9 Závěr

Předložená disertační práce je zaměřena na testování logických vnořených jader do SoC obvodu. Práce předkládá návrh diagnostického systému, který jednotlivá jádra testuje a určí, zda je jádro poruchové, či nikoliv.

Splnění cílů disertační práce

Jako logická jádra jsou v systému použity tzv. ISCAS benchmark obvody, které jsem upravil tak, aby obsahovaly požadovaný počet paralelních skenovacích řetězců patřičné délky. Takto upravená jádra odpovídají požadavkům testovací architektury RESPIN. Tato

architektura je vhodná k dekomprimaci testovacích vzorků vygenerovaných kompresním algoritmem COMPAS.

Na základě podrobné analýzy RESPIN architektury byl vybudován testovací mechanismus splňující požadavky normy IEEE 1500. Výhodou oproti originálnímu návrhu RESPIN architektury je použití standardního testovacího rozhraní jader. Unikátním a jednoduchým způsobem řeším mechanismus testovacího přístupu v FPGA. Tímto byla vnesena do oblasti návrhu testovacích sběrnic nová metodologie založená na dynamické rekonfiguraci FPGA.

Bylo navrženo řízení testovacího mechanismu tvořeného procesorem a řadičem. Procesor a řadič pracují současně. Procesor řídí celý průběh testování, řadič vykonává jen časově náročné operace. Touto koordinací bylo dosaženo krátké doby testu jádra. Pokud bereme v úvahu, že procesor je v současnosti již standardní součástí systému na čipu, plocha potřebná pro řadič je malá. Testovací data jsou uložena v paměti v komprimovaném tvaru, požadavek na využití malé kapacity paměti pro tyto data je také splněn. Celý diagnostický systém byl společně s jádrem implementován jako systém na čipu a realizován v FPSLIC AT94K obvodu.

Porovnání výsledků a srovnání s jinými metodami bylo provedeno v rámci 8 kapitoly. Zde je patrné, že naše metodika testování vykazuje dobré výsledky jak v celkové době testování jádra, tak v počtu testovacích bitů potřebných k vykonání testu jádra, což má příznivý dopad na velikost paměti určené pro tyto testovací bity. Taktéž absence speciálního hardwaru v podobě dekompresoru dat systém nevykazuje v této oblasti žádné hardwarové navýšení.

Přínos disertační práce pro praxi

Aplikací standardu IEEE 1500 byl vytvořen mechanismus pro testování vnořených jader v systémech na čipu. Na základě této normy byl vybudován diagnostický systém, který testuje logická jádra v SoC obvodech. V disertační práci je zkoumána tzv. RESPIN architektura, která byla posléze modifikována pro její použití v praktickém návrhu. V systémech na čipu mohou být použita jádra s běžně užívaným skenovacím DFT návrhem. Nejsou tedy zapotřebí žádné zásahy do takto nabízených jader od dodavatelů, jádra jsou tak připravena k okamžitému použití, což přispívá k rychlému uvedení SoC obvodu s těmito jádery na trh. Diagnostický systém užívá standardních rozhraní dle norem IEEE 1500, IEEE 1149.1 a sériového UART rozhraní. Test SoC obvodu lze spouštět jak z PC přes UART rozhraní, tak ze stávajících ATE zařízení podporujících IEEE 1149.1 rozhraní.

Systémový DFT návrh využívá k testování komprimované testovací vektory uložené v paměti ATE nebo v paměti samotného SoC obvodu. Dekomprimace testovacích vzorků probíhá ve volných testovacích jádrech, která jsou součástí SoC obvodu. Řízení užívá k testování obecného procesoru. Nejen procesor, ale i paměť je standardní součástí většiny současných SoC obvodů, oboje mohou být proto využity i k testovacím účelům. Z praktického pohledu nese takový systémový DFT návrh v SoC obvodu minimální hardwarové navýšení, kapacita paměti určená pro data k testování je vzhledem k jiným testovacím metodám taktéž minimální.

Velká výhoda našeho systému tkví v jeho unifikaci. Hardwarový návrh systému je pro všechny SoC obvody univerzální, tzn. že při použití různých jader v SoC obvodu se mění pouze software (testovací data v paměti), hardware zůstává původní.

V souvislosti se spotřebou obvodu je všeobecně známo, že SoC obvody jsou navrhovány na jejich funkční režim, nikoliv testovací režim. Diagnostický systém využívá při procesu testování kromě vlastního testovaného jádra navíc pouze jádra testovací, ostatní jádra v SoC obvodu nejsou do testovacího procesu zapojeny. V konečném důsledku tedy nedochází

k tak vysokému proudovému odběru, jako v případě zapojení více jader a tím i k nadměrnému přehřátí obvodu.

SoC obvody s diagnostickým systémem DyRESPIN najdou uplatnění v procesu testování bezprostředně po vyrobení ASIC obvodů v případě, že takový obvod obsahuje částečně i FPGA s dříve zmíněnými vlastnostmi. Diagnostický systém dále nachází uplatnění v SoC obvodech vytvořených v FPGA, kde je možné prověřování bezporuchovost jader, kdykoliv je vznesen požadavek na test jádra.

Návrh prototypu SoC obvodu v podobě funkčního vzorku obsahující diagnostický systém DyRESPIN je důkazem toho, že takový obvod je realizovatelný a tudíž může být DyRESPIN systém použit při praktickém systémovém DFT návrhu SoC obvodů.

Přínos disertační práce pro vědecký obor

Přínos práce pro vědecký obor diagnostika číslicových obvodů spočívá ve vytvoření nové metodiky testování logických jader v SoC obvodech. Ke generaci testovacích vzorků je použita metoda překrývání testovacích vektorů. Vygenerovaná komprimovaná data jsou uložena v paměti. K dekomprimaci testovacích dat je využito vlastností RESPIN architektury. Navržený testovací mechanismus umožňuje implementovat tento diagnostický systém do SoC obvodu založeného na vnořených jádrech. Takový SoC obvod lze prakticky realizovat v FPGA. Disertační práce zde tvoří poslední články této metodiky testování.

Dalším přínosem pro vědecký obor je využití výhod dynamické rekonfigurace FPGA v oblasti testování. Pomocí dynamické rekonfigurace části FPGA se vytváří mechanismus testovacího přístupu k jádrům. Návrh TAMu touto metodologií je výrazně jednodušší oproti jiným návrhům. Taktéž není doposud známo, že by se k vytvoření testovací sběrnice využívalo právě dynamické rekonfigurace FPGA. Zavedením této metodologie do DFT návrhů se rozšiřují možnosti dalšího využití dynamické rekonfigurace FPGA.

Budoucí práce

Doposud se realizace SoC obvodu s diagnostickým systémem prováděla na obvodu AT94K od firmy Atmel a používali jsme relativně malá jádra z důvodů omezené kapacity FPGA tohoto obvodu. Nyní hodláme přejít na novou platformu s obvodem Virtex 4 od firmy Xilinx. Tento obvod nabízí podstatně větší FPGA a umožňuje jeho dynamickou rekonfiguraci. Součástí obvodu je i HW jádro procesoru PowerPC, které bude možno využít k testovacím účelům. S tímto obvodem bude možné provádět experimenty i s velkými jádry.

Náš diagnostický systém je v současnosti schopen testovat jádro komprimovanými vzorky a získat výslednou odezvu na test taktéž v komprimované podobě ve formě příznaku. Systém je třeba rozšířit o možnost získávat příznaky jednotlivých testovacích kroků, eventuálně zaznamenávat přímo odezvy jádra na testovací vzorky. Tato vlastnost systému bude mít velký význam pro skupinu našeho týmu zabývající se generováním testovacích vektorů. Budeme tak schopni vyhodnotit okamžité pokrytí poruch jádra v jakémkoliv okamžiku průběhu testu.

Diagnostický systém nyní pracuje tak, že komprimovaná testovací data získává řadič z procesoru, který je vyčítá z paměti. Systém by se měl upravit tak, aby řadič načítal testovací data přímo z paměti a procesor by byl využit pouze k přípravě a spuštění testu, v průběhu testování jádra by byl volný.

S takovými výše uvedenými úpravami diagnostického systému je samozřejmě spjata vybudování odpovídajícího uživatelského rozhraní v PC. Z PC se budou nejen spouštět a vyhodnocovat testy jader, ale bude zajištěno i monitorování průběhu testu s možností ručního vkládání testovacích bitů a krokování testu.

Literatura

- [1] R. K. Gupta and Y. Zorian, "Introducing Core-Based System Design", *IEEE Design & Test of Computers*, vol. 14, pp. 15–25, Dec. 1997.
- [2] Y. Zorian, E. J. Marinissen, and S. Dey, "Testing Embedded-Core-Based System Chips", *IEEE Computer*, vol. 32, pp. 52–60, June 1999.
- [3] Y. Zorian, S. Dey, and M. Rodgers, "Test of Future System-on-Chips", in *Proceedings International Conference on Computer-Aided Design (ICCAD)*, pp. 392–398, Nov. 2000.
- [4] B. T. Murray and J. P. Hayes, "Testing ICs: Getting to the Core of the Problem", *IEEE Computer*, vol. 29, pp. 32–38, Nov. 1996.
- [5] A. M. Rincon, C. Cherichetti, J. A. Monzel, D. R. Stauffer, and M. T. Trick, "Core Design and System-on-a-Chip Integration", *IEEE Design & Test of Computers*, vol. 14, pp. 26–35, Dec. 1997.
- [6] L. Whetsel, "Core Test Connectivity, Communication, & Control", in *Proceedings IEEE International Test Conference (ITC)*, pp. 303–312, IEEE Computer Society Press, Oct. 1998.
- [7] L. Whetsel, "Optaddressable Test Ports: An Approach to Testing Embedded Cores", in *Proceedings IEEE International Test Conference (ITC)*, pp. 1055–1064, IEEE Computer Society Press, Sept. 1999.
- [8] J. Rajski, "DFT for High-Quality Low Cost Manufacturing Test", in *Proceedings of the Asian Test Symposium (ATS)*, pp. 3–8, IEEE Computer Society Press, Nov. 2001.
- [9] H. Vranken, T. Waayers, H. Fleury, and D. Lelouvier, "Enhanced Reduced Pin-Count Test for Full-Scan Design", in *Proceedings IEEE International Test Conference (ITC)*, pp. 738–747, IEEE Computer Society Press, Oct. 2001.
- [10] R. Kapur, R. Chandramouli, and T. Williams, "Strategies for Low-Cost Test", *IEEE Design & Test of Computers*, vol. 18, pp. 47–54, Dec. 2001.
- [11] J. Bedsole, R. Raina, A. Crouch, and M. S. Abadir, "Very Low Cost Testers: Opportunities and Challenges", *IEEE Design & Test of Computers*, vol. 18, pp. 60–69, Sept. 2001.
- [12] ITRS, "The International Technology Roadmap for Semiconductors, 2001 Edition", <http://public.itrs.net/>.
- [13] B. Bottoms, "The Third Millennium's Test Dilemma", *IEEE Design & Test of Computers*, vol. 15, pp. 7–11, Oct. 1998.
- [14] R. Dorsch and H-J Wunderlich, "Reusing Scan Chains for Test Pattern Decompression", in *Proceedings IEEE ETW*, pp. 24–32, 2001.
- [15] O. Novák, J. Zahrádka, Z. Pliva, "COMPAS – Compressed Test Pattern Sequencer for Scan Based Circuits", *Springer: Lecture Notes in Computer Science 3463*, ISSN 0302-9743, pp. 403–414, 2005.
- [16] O. Novák, J. Nosek, "Test Pattern Decompression Using a Scan Chain", in *Proceedings IEEE International Symposium on Defects and Fault Tolerance in VLSI Systems*, pp. 110 – 115, 2001.
- [17] Datasheet AT94K [online]. http://www.atmel.com/dyn/resources/prod_documents/doc1138.pdf
- [18] C. Barnhart, V. Brunkhorst, F. Distler, O. Farnsworth, B. Keller, B. Koenemann, „OPMISR: The foundation for compressed ATPG vectors“, in *Proceeding ITC*, pp. 748-757, 2001.
- [19] A. Jas, J. Ghos-Dastir and N. A. Touba, „Scan Vector Compression/Decompression Using Statistical Coding“, in *Proceeding VTS*, 1999

- [20] C. V. Krishna, N. A. Touba, „Reducing Test Data Volume Using LFSR Reseeding with Seed Compression”, in *Proceeding of ITC*, pp.321-330, 2002.
- [21] A. R. Pandey, H. J. Patel, „Reconfiguration Technique for Reducing Test Time and Test Data Volume in Illinois Scan Architecture Based Designs”, in *Proceeding IEEE VLSI Test Symp*, pp. 9-15, 2002.
- [22] A. C Chandra and K. Chakrabarty, „Frequency/Directed Run Length (FDR) Codes with Application to System/on/Chip Test Data Compression”, in *Proceeding VTS*, pp. 42-47, 2001.
- [23] J. Rajski, et al, „Embedded Deterministic Test”, in *Proceeding IEEE Trans. on CAD*, vol. 23, No. 5, pp. 776-792, May 2004.
- [24] L. Schäfer, R. Dorsch, H. J. Wunderlich, “RESPIN++ – Deterministic Embedded Test“, in *Proceedings of the Seventh IEEE European Test Workshop (ETW'02)*, IEEE Computer Society Press, 2002.
- [25] S. Hellebrand, H. G. Liang, H. J. Wunderlich, „A mixed mode BIST scheme based on reseeding of folding counters”, in *Proceeding of ITC*, 2000.

Přehled publikací

- [26] Z. Mader, „Diagnostika SoC obvodů s využitím RESPIN architektury“, *seminář PAD*, Moravany nad Váhom, Slovakia, str. 66-71, Sept. 2004, ISBN 80-969202-0-0.
- [27] Z. Mader, “System-on-a-Chip Diagnostics Using RESPIN Architecture”, in *Proceedings EDCC*, Budapest, Hungary, May 2005.
- [28] O. Novák, Z. Mader, “SOC Diagnostic System Using the RESPIN Architecture”, *IFIP WG 10.5 International Conference on Very Large Scale Integration System-on-Chip*, Perth, Western Australia, October 2005, ISBN 07298-0610-3.
- [29] Z. Mader, M. Jarkovský, “SoC diagnostic Design Using RESPIN Architecture”, *IEEE Workshop DDECS*, Prague, Czech Rep., pp. 241-243, April 2006, ISBN 1-4244-0184-4.
- [30] Z. Mader, “Samočinné testování jader v SoC obvodech s malými HW požadavky”, *seminář PAD*, Papradno, Slovakia, str. 7-12, September 2006, ISBN 80-969202-2-7.
- [31] O. Novák, Z. Plíva, J. Jeníček, Z. Mader, M. Jarkovský, “Self Testing SoC with Reduced Memory Requirements and Minimized Hardware Overhead”, in *Proceedings DFT*, Arlington, USA, pp. 300-308, October 2006, ISBN 0-7695-2706-X.
- [32] O. Novák, Z. Plíva, J. Jeníček, Z. Mader, “Self Testing SoC with Reduced Memory Requirements and Minimized Hardware Overhead”, *Journal of Electronic Testing: Theory and Applications*, Springer, připraveno k publikaci.