

VHDL OPIS KONFIGURABILNOG MIKROKONTROLERA ZA IMPLEMENTACIJU NA FPGA

Stanković, D. B.¹, Mitić, M. D., Đorđević, G. Lj.,
Elektronski fakultet u Nišu

I UVOD

Zajedničke osobine svih *embedded* sistema, generalno gledano, su njihova kompaktnost i orijentisanost ka nekoj specifičnoj primeni. *Embedded* sistemi niske i srednje složenosti, po pravilu, zasnovani su na 8-bitnim mikrokontrolerima i mikroracunarima na čipu [1]. Savremene familije mikrokontrolera odlikuju se velikom raznovrsnošću ugrađenih perifernih jedinica, što, u mnogim slučajevima, omogućava relizaciju sistema sa minimalnim brojem dodatnih komponenata. Međutim, postoje primene, pre svega u oblasti telekomunikacija i digitalne obrade signala, koje zahtevaju veliku moć obrade podataka uz relativno skromne zahteve u pogledu upravljanja. Optimalno rešenje za ove namene je ono koje kombinuje jednostavan mikrokontroler sa složenim hardverskim akceleratorima u jedinstveni sistem na čipu [2]. Zahvaljujući stalnom povećanju logičkog kapaciteta uz istovremeni pad cene, danas se sve veći broj sistema na čipu realizuje u FPGA tehnologiji [3]. Otuda se javlja potreba za razvojem jezgra mikrokontrolera prilagođenog implementaciji na FPGA. Zbog ograničenih hardverskih resursa koji su raspoloživi u FPGA, od jezgra se zahteva niska hardverska složenost uz mogućnost konfiguracije koja bi omogućila dodatnu optimizaciju u pogledu zauzeća hardverskih resursa. Takođe, poželjno je da jezgro bude kompatibilno, do nivoa skupa instrukcija, sa nekom standardnom familijom mikrokontrolera, čime bi se omogućilo korišćenje raspoloživih alata za razvoj softvera. Kao primer jednog takvog rešenja, ovde će biti predstavljeno konfigurabilno jezgro mikrokontrolera zasnovanog na PIC *mid-range* familiji mikrokontrolera (cmPIC). Jezgro je opisano pomoću VHDL jezika i prilagođeno implementaciji na Xilinx FPGA [4]. cmPIC jezgro podržava veliki broj konfiguracionih opcija koje se odnose na veličinu programske i registarske memorije, uključivanje/isključivanje različitih perifernih jedinica, kao i statičku konfiguraciju režima rada pojedinačnih perifernih jedinica.

Osnovne prednosti pristupa kod koga se mikrokontroler, dostupan u obliku konfigurabilnog jezgra, implementira na FPGA zajedno sa dodatnim specijalizovanim hardverskim blokovima, u odnosu na rešenja koja koriste standardni mikrokontroler na čipu u sprezi sa FPGA, ogledaju se u sledećem: a) Implementacija dizajna koji je stvarno orijentisan specifičnoj primeni. Za razliku od mikrokontrolera na čipu, gde se aplikacija prilagođava mikrokontroleru, mikrokontrolersko jezgro se može konfigurirati prema zahtevima konkretne aplikacije. b) Niža cena sistema, s obzirom na manji broj potrebnih komponenti. c) Performanse

koje postiže mikrokontrolersko jezgro implementirano na FPGA su bolje od performansi odgovarajućeg mikrokontrolera na čipu. Razlog za to je mogućnost rada na višim taktnim frekvencijama.

U sekciji II date se osnovne karakteristike arhitekture PIC familije mikrokontrolera, koja je poslužila kao osnova za razvoj konfigurabilne cmPIC arhitekture. U sekciji III opisana je cmPIC arhitektura sa naglaskom na rešenja koja se tiču konfigurabilnosti. Sekcija IV opisuje tok projektovanja zasnovan na cmPIC jezgru. Ostvareni rezultati koji se odnose na hardversku složenost i performanse cmPIC jezgra implementiranog na konkretno FPGA kolo izloženi su u sekciji V.

II ARHITEKTURA PIC MIKROKONTROLERA

Kao osnov za realizaciju cmPIC-a uzeta je arhitektura *Microchip*-ovih PIC mikrokontrolera familije *mid-range* [5]. PIC mikrokontroleri se odlikuju visokim performansama koje su posledica njihove arhitekture koja ima mnoga svojstva RISC arhitekture procesora a u koja spadaju: *harvard* arhitektura (odvojena memorija za instrukcije i podatke), smanjen (redukovan) skup instrukcija, protočnost kod izvršenja instrukcija, registarska fajl arhitektura, *long word* instrukcije, *single word* i *single cycle* instrukcije, ortogonalan skup instrukcija (omogućavaju izvršenje bilo koje operacije na bilo kom registru uz pomoć bilo kog načina adresiranja). Sama RISC arhitektura je pogodna za realizaciju na FPGA jer ne zahteva upotrebu velike količine hardvera pa je i to uzeto u obzir prilikom izbora arhitekture za realizaciju konfigurabilnog mikrokontrolera. Pored visokih performansi i niske cene (kada su u pitanju hardverski resursi), prednost korišćenja ove arhitekture je i to što postoji veliki broj alata koji su razvijeni kao njena podrška, počevši od kompajlera, debugera i simulatora pa do kompletnih radnih okruženja za brzi razvoj aplikacija za ove kontrolere. Takođe, javno je dostupna velika količina softvera razvijenog za različite primene ovih mikrokontrolera.

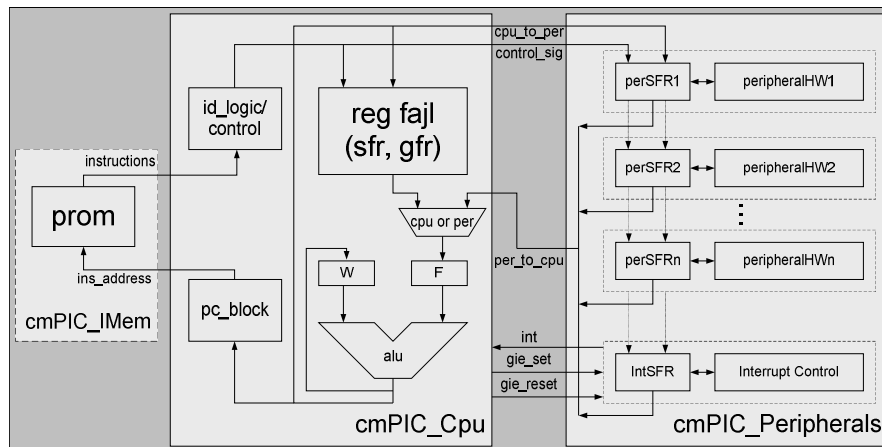
III ARHITEKTURA KONFIGURABILNOG MIKROKONTROLERA cmPIC

Arhitektura cmPIC-a (slika 1) se sastoji iz tri konfigurabilna dela: programska memorija (*cmPIC_Imem*), CPU (*cmPIC_Cpu*) i periferni podsistem (*cmPIC_Peripherals*).

A. Programska memorija

Blok instrukcione memorije (IM) sadrži program koji cmPIC izvršava. Uobičajeno je da se kod *embedded* sistema za smeštanje programa koristi diskretna ROM/PROM ili *Flash* memorija ugrađena u samom čipu mikrokontrolera. S obzirom da FPGA ne poseduje ugrađeni *Flash*, a da je naš

¹ Stipendista Ministarstva za nauku i zaštitu životne sredine



Slika 1. Arhitektura konfigurabilnog mikrokontrolera cmPIC

cilj bio realizacija celokupnog sistema na jednom čipu, usvojeno je da se IM opisuje kao RAM. Sadržaj ovog RAM-a je unapred definisan i nalazi se u izvornom fajlu *prom.vhd* koji se pridodaje projektu cmPIC-a prilikom sinteze. Organizacija instrukcione memorije je n reči po 14 bitova pri čemu svaka reč predstavlja jednu instrukciju. Broj reči n je konfiguracioni parametar koji je određen veličinom programa koji će se izvršavati na cmPIC-u. Maksimalna veličina IM je određena PIC arhitekturom na 8K reči, odnosno raspoloživim resursima FPGA čipa. Dodatni konfiguracioni parametar IM je i način njene implementacije u FPGA: a) blok RAM (namenska RAM memorija koja je raspoloživa u pojedinim FPGA čipovima), b) distribuirani RAM (koriste se LUT blokovi).

B. CPU

CPU jedinica predstavlja jezgro cmPIC arhitekture koje objedinjuje: (a) upravljačku jedinicu i instrukcioni dekoder (blok *id_logic/control*), (b) programski brojač sa pridruženom stek memorijom (*pc_block*), (c) ALU jedinicu, (d) registarsku memoriju (*reg_file*) i (e) interfejs prema periferijskom podsistemu.

Upravljačka jedinica omogućava dvo-stepeno, protočno izvršenje instrukcija (dok traje izvršenje jedne instrukcije vrši se pribavljanje sledeće). Sve instrukcije (osim instrukcija grananja za koje je potrebno duplo veće vreme) se izvršavaju u okviru jednog instrukcionog ciklusa. Instrukcioni ciklus je podeljen na četiri pod-ciklusa, ili stanja, Q0, Q1, Q2 i Q3. Nakon reset-a CPU se nalazi u stanju Q0, a programski brojač se postavlja na 0. U stanju Q0 obavljaju se sledeće aktivnosti: (1) vrši se upis pribavljene instrukcije iz programske memorije (prom) u instrukcioni registar. (2) Na osnovu koda instrukcije određuje se način adresiranja (direktno/indirektno), kako bi se pripremili uslovi za pribavljanje operandu u sledećem stanju. (3) Ukoliko je prethodna pribavljena instrukcija bila instrukcija grananja, u instrukcioni registar se upisuje NOP (no operation) i vrši se grananje na adresu koja je u prethodnom ciklusu od 4 takta bila upisana u programski brojač. (4) Detektuje se postojanje zahteva za opsluživanjem prekida. U stanju Q1 vrši se dekodiranje instrukcije u bloku *id_logic* i pribavljanje direktno ili indirektno adresiranog operandu iz registarske memorije. Selektovani operand se upisuje registar *F* koji predstavlja jedan od dva ulaza u ALU. Drugi ulaz je uvek *W* (*working*) registar. U stanju Q2 ALU izvršava operaciju (utvrđenu na osnovu dekodiranja instrukcije u prethodnom stanju) nad operandima u registrima *W* i *F* (nad oba ili nad

samo jednim od njih). Sve aritmetičko logičke instrukcije sadrže po jedan bit koji definiše da li se rezultat rada ALU smešta u *W* ili u registarsku memoriju. U stanju Q3 vrši se upis rezultata u određeni registar. Takođe na kraju ovog ciklusa se postavlja signal koji označava da li do grananja dolazi da bi se u sledećem ciklusu od 4 takta izvršilo grananje na adresu grananja.

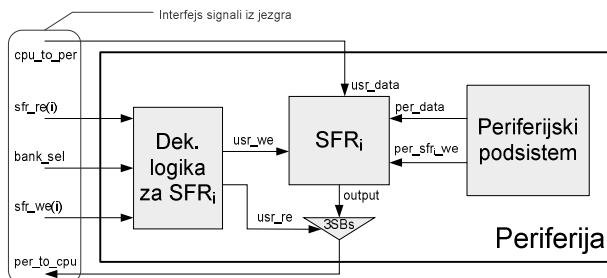
Instrukcije poziva potprograma (CALL), kao i prihvatanje zahteva za prekid zahtevaju pamćenje adrese povratka iz potprograma ili prekidne rutine u glavni program. Kod PIC arhitekture, za ovu namenu, koristi se namenska stek memorija (deo bloka *pc_block*) dubine 8 reči, što određuje maksimalni broj ugnježenih poziva potprograma ili prekidnih rutina. Kod cmPIC arhitekture, dubina steka predstavlja konfigurabilni parametar, koji se može podesiti prema zahtevima konkretnog programa. Na taj način, prevazilazi se ograničenje o dozvoljenoj dubini poziva potprograma koje imaju svi standardni PIC mikrokontroleri.

Registarska memorija kod PIC mikrokontrolera je podeljena na banke maksimalne veličine 2^7 bajtova (jer je dužina adrese operandu 7 bitova) a svaka banka je podeljena na dva regiona: region sa registrima specijalne namene (SFR) (prvih 16 ili 32 bajta u zavisnosti od tipa mikrokontrolera) i region sa registrima opšte namene (GPR). Prilikom pristupa registarskoj memoriji, CPU isporučuje adresu registra u okviru banke, dok se izbor banke vrši nezavisno posredstvom bitova za selekciju banke koji su deo namenskog registra. Dodatno, adresni prostor od 16 reči na kraju svake banke je deljiv između svih banaka. Kod cmPIC arhitekture, broj banaka kao i broj GPR u okviru svake banke se mogu konfigurisati po potrebi. U maksimalnoj konfiguraciji, registarska memorija sadrži 384 GPR-ova. Posredstvom SFR-ova obavlja se pristup periferijskim jedinicama. Iako deo jedinstvenog adresnog prostora, ovi registri su, fizički, deo periferijskih jedinica, a njihov broj zavisi od broja i tipova ugrađenih periferija.

C. Periferijski podsistem

Arhitektura PIC familije mikrokontrolera karakteristična je po velikom broju raznorodnih periferijskih jedinica. Dodatno, periferijske jedinice tipično podržavaju nekoliko različitih režima rada. Za pristup periferijskim jedinicama kod PIC arhitekture koristi se princip memorijski preslikanog ulaza-izlaza. Svakoj jedinici pridružen je skup SFR-ova putem kojih je moguće programski regulisati režim rada i razmenjivati podatke sa jedinicom. Raspoloživost fleksibilnih hardverskih jedinica omogućava adaptaciju mikrokontrolera

za različite primene bez potrebe ugradnje dodatnih eksternih komponenti. Međutim, kod većine uobičajenih primena, mnoge perifernijske jedinice ostaju neiskorišćene, ili se koriste samo u jednom režimu rada. cmPIC arhitektura prevazilazi ovaj problem tako što pruža mogućnost projektantu da u procesu konfiguracije arhitekture uključi samo one jedinice koje su potrebne za konkretnu aplikaciju.



Slika 2. Interfejs jezgra cmPIC prema periferijama

Polazeći od zahteva lake konfigurabilnosti i proširljivosti, kod cmPIC arhitekture periferijske jedinice su izdvojene u zaseban podsistem, koji je putem jednostavnog interfejsa spregnut sa CPU jedinicom (Slika 2). Interfejs čine: (a) dve jednosmerne 8-bitne magistrale podataka, *cpu_to_per* (za upis podataka u SFR-ove) i *per_to_cpu* (za čitanje podataka iz SFR-ova); (b) skup adresnih i upravljačkih signala za adresiranje SFR-ova i izbor operacije (upis/čitanje). Uključivanje/islučivanje periferija se, samim tim, svodi na uključivanje/islučivanje njihovih SFR-ova, a to se svodi na jednostavnu modifikaciju logike za dekodiranje adresa memorije.

Sam SFR predstavlja specifičan konfigurabilni blok. Kod pojedinih SFR-ova ne koriste se sve bitske pozicije. Neke bitove može da menja samo periferija preko svojih signala, neke samo CPU preko instrukcija programa a neke i periferija i CPU. U projektu cmPIC arhitekture SFR je opisan konfigurabilnim VHDL modulom kome se prilikom instanciranja, putem dve 8-bitne maske, prenose informacije o tipu i načinu rada svake bitske pozicije.

Prekidi

cmPIC može da ima različite izvore prekida. Prekidi mogu da potiču iz CPU jezgra, a mogu da ih generišu i periferne jedinice. Sistem za obradu prekida sadrži SFR-ove koji pamte flegove za pojedine prekide i bitove dozvole za te prekide, kao i globalni bit dozvole za prekide. Sistem za obradu prekida je povezan sa perifernim modulima preko odgovarajućeg signala zahteva za prekid. Taj signal postavlja odgovarajući fleg koji se briše programski, nakon obrade prekida. Sa jezgrom je povezan preko tri signala (Slika 1): *sig_int*, *sig_gie_set* i *sig_gie_reset*. Sistem za obradu prekida proverava stanja fleg bitova i odgovarajućih bitova dozvole. Ukoliko postoji zahtev za prekidom i on je dozvoljen, sistem generiše signal *sig_int* kojim obaveštava jezgro o zahtevu za prekid. Jezgro prihvata zahtev i prelazi na obradu prekida. Da bi se sada onemogućilo prihvatanje novog zahteva za prekid, jezgro šalje signal *sig_gie_reset*. Po završetku obrade prekida jezgro se vraća na normalno izvršavanje programa i dozvoljava nove zahteve za prekid, šaljući *sig_gie_set*.

Implementirane periferije

U arhitekturi cmPIC-a, implementirane su sledeće periferije: **TIMERO** – modul koji sadrži 8-bitni brojač, ima mogućnost izbora izvora takta između internog i eksternog takta i pri izboru eksternog izvora takta za izbor ivice takta. Takođe

sadrži i 8-bitni programabilni preskaler. Generiše zahtev za prekid pri prelasku brojača sa FFh na 00h.

TIMER1 - modul koji sadrži 16-bitni brojač, ima mogućnost rada u tri moda: kao sinhroni tajmer (izvor takta je interni), kao asinhroni brojač (izvor takta je eksterni) i kao sinhroni brojač (eksterni izvor takta je sinhronizovan). TIMER1 može biti isključen. Takođe ima i unutrašnji reset ulaz, koji generiše CCP modul. Generiše zahtev za prekid pri prelasku brojača sa FFFFh na 0000h.

TIMER2 - modul koji sadrži 8-bitni brojač sa programabilnim preskalerom, postskalerom (njegov izlaz generiše zahtev za prekid) i period registrom.

CCP1 i CCP2 – mogu da rade u jednom od tri režima: kao 16-bitni *capture* registar (pamti stanje TIMER1), kao 16-bitni *compare* registar (upoređuje svoju trenutnu vrednost sa TIMER1 registrom, u slučaju jednakosti generiše CCPIF prekid) i kao PMW modul (trajanje zadato u TIMER2).

UART – (*Universal Asynchronous Receiver Transmitter*) modul je serijski U/I modul. Moguće je konfigurisati ga da radi serijski prijem podataka ili serijsku predaju podataka. U okviru svakog režima moguće je još i podešavati pojedine parametre.

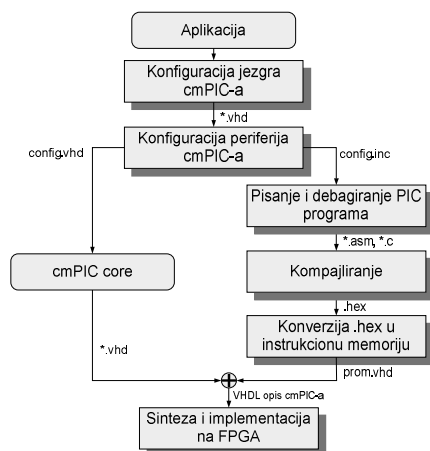
Konfigurabilnost periferijskog sistema

Konfigurisanje periferijskog podsistema cmPIC-a je moguće izvršiti na dva nivoa, kao: (1) dinamičko konfigurisanje i (2) statičko konfigurisanje. **Dinamičko konfigurisanje** se odnosi na programsko upravljanje režimom rada periferijskih jedinica i ostvaruje se upisom specifičnih bitskih sekvenci u upravljačke registre jedinice. **Statičko konfigurisanje** se vrši u fazi izbora arhitekture sistema. Projektant bira koje periferijske jedinice, i u okviru jedinice koje funkcionalnosti su mu potrebne. Razlikuju se dva podnivoa statičkog konfigurisanja: (1) **na nivou sistema** - svodi na isključivanje pojedinih periferija, tako da se one (zajedno sa pridruženim SFR-ovima) uopšte ne sintetišu; (2) **na nivou periferije** - svodi se na fiksiranje pojedinih vrednosti u upravljačkim registrima periferije. U procesu sinteze, izbacuju se svi oni putevi, i hardver koji se našao na njima, koji se nikada ne koriste. Na taj način, dobija se jedinica, manje složenosti, koja može da radi samo u izabranom režimu rada. Obe statičke metode konfiguracije imaju za cilj smanjenje zauzeća FPGA, jer smanjuju količinu hardvera koju je potrebno sintetisati.

IV TOK PROJEKTOVANJA

Tok projektovanja *embedded* aplikacije zasnovan na korišćenju cmPIC arhitekture predstavljen je na Slici 3. Polazeći od unapred postavljenih zahteva aplikacije, projektant, u prvoj fazi projektovanja, utvrđuje potrebe za hardverskim resursima na osnovu kojih konfigurira jezgro i periferijski podsistem cmPIC arhitekture. Kao rezultat ove aktivnosti generišu se: (a) VHDL paket koji sadrži izabrane vrednosti konfiguracioni parametara (*config.vhd*) i pridružuje se VHDL projektu cmPIC mikrokontrolera i (b) datoteka koja sadrži sve relevantne podatke o arhitekturi bitne za razvoj softvera (definiše simbolička imena i adrese SFR-ova, veličinu registerske memorije i sl.) i pridružuje se softverskom projektu.

U sledećoj fazi, projektant piše i debuguje programa na asemblerskom jeziku za PIC mikrokontrolere ili na C-u. Nakon toga se napisani program kompajlira.



Slika 3. Tok projektovanja

Za programe pisane na assembleru moguće je koristiti integrisano razvojno okruženje *MPLab* kompanije *Microchip* koje je besplatno. Za kompajliranje C programa postoji veliki broj komercijalnih i takođe i javno dostupnih kompajlera kao što su npr. [6] i [7]. Kao izlaz iz kompajlera dobije se kompajlirani program u *.hex* formatu kojim bi se napunio mikrokontroler da je u pitanju regularan dijagram toka. Kod upotrebe konfigurabilnog mikontrolera potrebno je načiniti još nekoliko koraka. Potrebno je prevesti *.hex* fajl u VHDL opis programske memorije (*prom.vhd*) za šta se koristi poseban program za konverziju. Ovako dobijena VHDL datoteka se uključuje u projekat i predstavlja memoriju u kojoj su smeštene instrukcije programa koje cmPIC izvršava. Konačno, vrši se sinteza i implementacija na FPGA kolu.

V REZULTATI

Implementacijom realizovane arhitekture na FPGA kolu Xilinx FPGA Spartan3 došlo se do podataka o zauzeću, koji se nalaze u Tabeli. Puna konfiguracija zauzima 832 slice-ova (1375 LUT-ova) i može da radi na maksimalnoj frekvenciji od 64.8 MHz. Minimalna konfiguracija, koja sadrži samo jednu memorijsku banku i od periferija samo TIMER0 zauzima samo 451 slice-ova (669 LUT-ova) i može da radi na frekvenciji od blizu 100 MHz, što čini uštedu u površini od blizu 50%. Najviše mesta zauzima CPU sa GPR, gde GPR-ovi zauzimaju mesto srazmerno broju banaka, tako da pri konfigurisanju cmPIC-a treba voditi računa o racionalnom korišćenju registara. Od svih periferija, najviše mesta zauzima UART. U zavisnosti od konfiguracije i uključenih periferija, moguće je postići smanjenje zauzeća i do više od 50%.

VI ZAKLJUČAK

U ovom radu je predstavljen konfigurabilni mikrokontroler cmPIC pogodan za korišćenje u sistemima na čipu za primene kod *embedded* sistema za telekomunikacije i digitalnu obradu signala. Predstavljeno rešenje odlikuje se fleksibilnošću koju poseduje procesor opšte namene uz istovremenu optimalnu iskorišćenost hardverskih resursa. Dalji rad vezan za cmPIC obuhvata dodavanje novih periferija, generisanje softverskog okruženja koje bi korisniku olakšalo konfigurisanje mikontrolera i koje bi omogućilo grubu procenu zauzeća hardverskih resursa, a takođe može se modifikovati i unutrašnja arhitektura kako bi se dodatno ubrzalo izvršenje instrukcija.

Tabela 1. Rezultati implementacije

komponenta		zauzeće	
		# slice	# LUT
CPU	bez GPR	190	310
GPR	1 banka, 96 registara	68	136
	2 banke, 192 registara	138	275
	4 banke, 384 registara	280	547
TIMER0	puna verzija	27	48
	takt eksterni/interni	26	46
	isključen preskaler	18	33
	fiksiran preskaler	19-23	34-41
	minimum	14	26
TIMER1	puna verzija	24	43
	fiksiran preskaler	20-22	36-37
	takt eksterni/interni	23/22	41/40
	minimum	17	28
TIMER2	puna verzija	34	45
	fiksiran postskaler	29	44
	fiksiran preskaler	29-32	40-44
	minimum	24	37
CCP	puna verzija	27	41
	PMW mode	20	17
	compare mode	24	22
	capture mode	9	10
UART	puna verzija	87	103
	receiver	67	73
	transmitter	50	53

LITERATURA

- [1] T. Noergard, *Embedded Systems Architecture*, Elsevier, Amsterdam, 2004.
- [2] A. Jantsch, *Modelling Embedded Systems and SoC's*, Morgan Kaufmann, San Francisco, 2003.
- [3] R. Munden, *ASIC and FPGA Verification: A Guide to Component Modelling*, Morgan Kaufmann, 2004.
- [4] P.J.Ashenden, *The Designer's Guide to VHDL*, 2nd edition, Morgan Kaufmann, San Francisco, 2001.
- [5] Microchip, *PicMicroMID-range MCU Family Reference Manual*, Microchip Technology Inc. <http://www1.microchip.com/downloads/en/DeviceDoc/33023a.pdf>, 1997.
- [6] HI-TECH, HT-PICC Compiler, <http://www.htsoft.com/>
- [7] Mikroelektronika, MicroC for PIC, <http://www.mikroelektronika.co.yu>

Abstract

VHDL description of a configurable microcontroller based on a PIC mid-range family of microcontrollers (cmPIC) is presented. The microcontroller is suitable for FPGA implementation and could be used as a CPU core of various systems on chip. Different parameters of this microcontroller such as, memory size, turning on/off peripherals and similar, of this microcontroller are configurable. We have implemented cmPIC on Xilinx Spartan3 FPGA and the results related to performance and FPGA chip usage for different configuration options are presented. Configurable microcontroller presents a valuable solution in applications such as audio and video signal processing and thus they can be used in the design of a variety of telecommunications systems.

VHDL CORE OF A CONFIGURABLE MICROCONTROLLER INTENDED FOR FPGA IMPLEMENTATION

Stanković, D. B., Mitić, M. D., Đorđević, G. Lj.