



**ELEKTROTEHNI KI FAKULTET
BEOGRAD**

RISC procesor 5 stepeni pipeline

student: Nikola Nenin
2010/0428

profesor: dr Veljko Milutinovi
asistent: Živojin Šuštran

Beograd
15.09.2016.

RISC procesor sa 5-stepenim pipeline-om

Nikola Nenin

email: nn100428d@student.etf.rs

1. DEFINISANJE PROJEKTA

1.1 Uvod

Projekat iz predmeta *Ra unarski VLSI sistemi* nam prvenstveno služi za savladavanje osnovnih koncepata jezika VHDL, kao i sticanje iskustva u projektovanju i testiranju procesora kroz samostalan rad. Arhitektura procesora opisanog u projektnom zadatku bliska je sa arhitekturom modernih procesora koji su danas u upotrebi.

1.2 Ciljevi projekta

Cilj projekta jeste sticanje praktičnog iskustva u projektovanju hardvera korišćenjem alata Quartus II i ModelSim firme Altera i jezika VHDL.

2. OPIS DIZAJNA

2.1 Zabeleške uz dizajn

Projektovan je 32-bitni procesor opšte namene koji je povezan sa dve keš memorije, instrukcijskom i memorijom za podatke.

Procesor sadrži proto nu obradu. Broj stepeni je 5. Sve hazarde je potrebno hardverski razrešiti uz poštavanje slede ih zahteva: 1) za instrukcije skoka se koristi prediktor skoka i 2) zaustavljati proto nu obradu samo kad prosle ivanje nije mogu e i kada postoji zavistnost po podacima. Pored traženih zahteva težiti ka što boljim performansama. Prediktor skoka treba da bude realizovan po principu keša sa dvobitnom šemom.

Jezgro ima 32 registara opšte namene širine 32-bita. Registri su obeleženi sa R0 – R31. Registri specijalne namene su: 1) PC je pokaziva na slede u instrukciju i 2) SP pokaziva na vrh steka. SP pokazuje na prvu slobodnu lokaciju i stek raste ka nižim adresama. Instrukcijski set dat je u slede em poglavlju. Prilikom generisanja izuzetka prekinuti rad procesora.

2.2 Faze dizajna

Procesor je dizajniran po fazama, svaka od 5 faza pipeline-a je dizajnirana posebno i testirana samostalno, tek nakon završetka svih faza su one povezane i napravljen je jedan entitet koji predstavlja procesor.

2.2.1 Interfejs izme u procesora i memorije

Za komunikaciju izmedju procesora i keš memorije za podatke koriste se adresne linije (addr), linije za upis podataka (data_rd) i linije za itanje podataka (data_wr). Radi organizacije itanja i upisa u memoriju uvedeni su dodatni kontrolni signali i to: rd i wr. Interfejs izme u procesora i keš memorije za instrukcije ine: adresne linije (pc) i linije za itanje podataka (word)

OP[31:29]	000	001	010	011	100	101
OP[28:26]						
000	LOAD	ADD	AND	SHL	JMP	BEQ
001	STORE	SUB	OR	SHR	JSR	BNQ
010			XOR	SAR	RTS	BGT
011			NOT	ROL		BLT
100	MOV	ADDI		ROR	PUSH	RGE
101	MOVI	SUBI			POP	RLE
110						
111						

Tipovi operacija (Tabela 1)

2.2.2 Faze izvršavanja instrukcije

Izvršavanje instrukcija se vrši kroz 5 faza i to su slede e faze: faza dohvatanja instrukcije (IF), faza dekodovanja instrukcije (ID), faza izvršavanja instrukcije (EX), faza za pristup memoriji za podatke (MEM), faza za upis rezultata na odgovaraju u lokaciju ili u odgovaraju i registar (WB).

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
opcode		Rd					Rs1					immediate																				
opcode		Rd					Rs1					Rs2					Reserved															
opcode		Rd					Rs1					Reserved																				
opcode		Rd					Reserved					immediate																				
opcode		Reserved					Rs1					immediate																				
opcode		Rd					Reserved					immediate					Reserved															
opcode		immediate					Rs1					Rs2					immediate															
opcode		Rd					Reserved																									
opcode		Reserved					Rs1					Reserved																				
opcode																																
opcode		Reserved					Rs1																									
opcode																																

Tipovi instrukcija (Tabela 2)

3. IMPLEMENTACIJA

Implementacija je izvedena koriste i alate Quartus II i ModelSim. Status nakon sinteze i analize pomo u Quartus II alata dat je u nastavku (Slika 1)

Flow Status	Successful - Thu Sep 15 20:53:22 2016
Quartus II 64-Bit Version	13.0.1 Build 232 06/12/2013 SP 1 SJ Web Edition
Revision Name	CPU
Top-level Entity Name	CPU
Family	Cyclone II
Device	EP2C70F896I8
Timing Models	Final
Total logic elements	7,691
Total combinational functions	5,539
Dedicated logic registers	2,628
Total registers	2628
Total pins	165
Total virtual pins	0
Total memory bits	0
Embedded Multiplier 9-bit elements	0
Total PLLs	0

Izveštaj sintetizovanog modela (Slika 1)

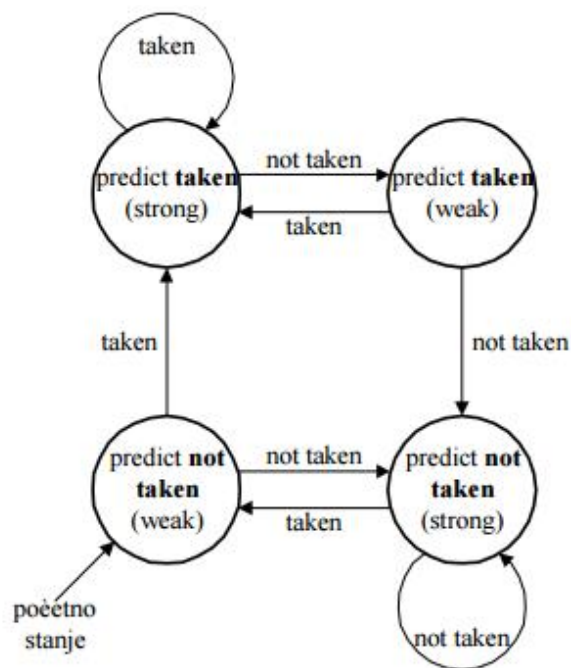
3.1 Hazardi

U toku izvršavanja instrukcija može do i do više razli itih hazarda i oni se mogu podeliti na 3 tipa: strukturalni hazardi, hazardi podataka i upravlja ki hazardi. Strukturalni hazardi nastaju zbog potrebe da se istovremeno pristupa istom resursu od strane razli itih instrukcija koje se nalaze u razli itim stepenima pipeline-a. Hazardi podataka nastaju kada je pristup nekom podatku od strane neke instrukcije u nekom stepenu pipeline-a uslovljeno prethodnim pristupanjem tom istom podatku od strane neke druge instrukcije iz nekog drugog stepena pipeline-a. Upravlja ki hazardi nastaju zbog skokova i drugih instrukcija koje menjaju vrednost PC-a.

3.2 Prediktor skoka

Prediktor skoka (Slika 2) je poseban entitet u okviru realizacije dizajna procesora.

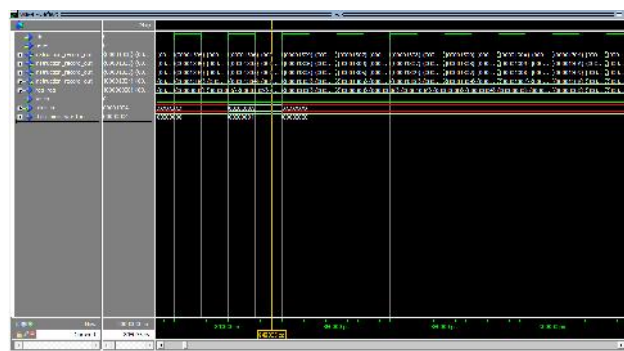
Dizajniran je u vidu tabele i može se na i u etri stanja: WEAK_TAKEN, WEAK_NOT_TAKEN, STRONG_TAKEN, STRONG_NOT_TAKEN.



Dvobitna šema (Slika 2)

4. TESTIRANJE I VERIFIKACIJA

Simulacija je zamišljena u vidu procesa. Jedan koji predstavlja beskona nu petlju u kojoj se na svakih 5ns generiše signal kloka procesora. Proces je startovanje sistema tako što se takt zadrži signal reset aktivan i nakon toga ugasi i zapo ne izvršavanje. Okruženje za testiranje je ModelSim gde se prati dijagram vremenski zavisnih signala u okviru implementacije sistema (Slika 3).



Vremenski dijagram (Slika 3)

4.1 Javni test

Test koji je dostavljen od strane predmetnog asistenta.

```
PC = 00001000
SUB R6, R6, R6
SUB R4, R4, R4
LOAD R6, R6
LOAD R5, R6
SUB R5, R5, #1
BGT R5, R4, -2
STORE R5, R6
HALT
```

4.2 Test svih instrukcija

Test koji proverava sve mogu e instrukcije u okviru zadate projektne specifikacije.

A9	MOVI	R0, 100
AA	LD	R1, R0, 0
AB	MOV	R2, R1
AC	ST	R2, R0, 1
AD	ADD	R3, R1, R2
AE	SUB	R4, R3, R2
AF	ADDI	R5, R2, 3
B0	SUBI	R6, R2, 2
B1	ST	R3, R0, 2
B2	ST	R4, R0, 3
B3	ST	R5, R0, 4
B4	ST	R6, R0, 5
B5	AND	R10, R3, R4
B6	OR	R11, R3, R4
B7	NOT	R12, R3, R4
B8	XOR	R13, R1, R2
B9	ST	R10, R0, 6
BA	ST	R11, R0, 7
BB	ST	R12, R0, 8
BC	ST	R13, R0, 9
BD	JMP	R13, BF
BE	ST	R13, R0, 0
BF	SHL	R3, 1
C0	SHR	R4, 1
C1	SAR	R5, 2
C2	ROL	R6, 1
C3	ROR	R6, 1
C4	ST	R3, R0, A
C5	ST	R4, R0, B
C6	ST	R5, R0, C
C7	ST	R6, R0, D
C8	MOVI	R30, 1000
C9	MOVI	R19, 0
CA	PUSH	R13
CB	POP	R20
CC	JSR	R30, 0
CD	BEQ	R1, R2, 1
CE	ST	R13, R0, 0
CF	BEQ	R1, R20, 1
D0	JSR	R30, 0
D1	BNQ	R1, R2, 1
D2	JSR	R30, 0
D3	BNQ	R1, R20, 1
D4	ST	R13, R0, 0
D5	BGT	R3, R4, 1
D6	ST	R13, R0, 0
D7	BLT	R3, R4, 1
D8	JSR	R30, 0
D9	BGE	R3, R4, 1
DA	ST	R13, R0, 0
DB	BLE	R3, R4, 1
DC	JSR	R30, 0
DD	BGE	R1, R2, 1

DE	ST	R13, R0, 0
DF	BLE	R1, R2, 1
E0	ST	R13, R0, 0
E1	HALT	
1000	ST	R20, R19, 0
1001	ADDI	R19, R19, 1
1002	ADDI	R20, R20, 1
1003	RTS	

4.3 Provera hazarda

Test koji služi za analizu i uspešno razrešavanje svih mogu ih hazarda koji se mogu na i u okviru implementacije.

0	MOVI	R0, 1000
1	LD	R1, R0, 0
2	ADDI	R1, R1, 1
3	MOV	R2, R1
4	MOV	R3, R2
5	ADD	R4, R1, R2
6	ST	R1, R0, 1
7	ST	R2, R0, 2
8	ST	R3, R0, 3
9	ST	R4, R0, 4
A	HALT	

4.4 Sortiranje niza

Test koji za ulazni skup re i dati skup sortira neopadaju e.

0	MOVI	R0, 0
1	MOVI	R31, 1000
2	MOVI	R30, 10
3	JSR	R0, 1200
1200	LD	R29, R31, 0
1201	MOV	R19, R31
1202	SUBI	R30, R30, 1
1203	BEQ	R30, R0, 8
1204	ADDI	R31, R31, 1
1205	MOV	R2, R31
1206	MOV	R1, R30
1207	JSR	R0, 1300
1208	BGE	R3, R29, 2
1209	ST	R3, R19, 0
120A	ST	R29, R13, 0
120B	JMP	R0, 1200
120C	RTS	
1300	LD	R3, R2, 0
1301	MOV	R13, R2
1302	SUBI	R1, R1, 1
1303	BEQ	R1, R0, 7
1304	ADDI	R2, R2, 1
1305	LD	R4, R2, 0
1306	BGE	R4, R3, 2
1307	MOV	R3, R4
1308	MOV	R13, R2
1309	SUBI	R1, R1, 1
130A	BNQ	R1, R0, -7
130B	RTS	

5. ZAKLJUČAK

Ovaj projekat je omogućio jedan osnovni uvid u to kako se dizajnira procesor i koji su izazovi sa kojima se inženjeri susreću u tom procesu. Tako je objedinio znanja iz domena arhitekture računara koja su sada dobila konkretnu primenu.

Ono što je nedostatak jeste to što ne oslikava u potpunosti fizičke karakteristike kola u vidu prirodnog kašnjenja koje nastaje zbog šuma.

6. LITERATURA

Spisak literature korišten u izradi domaćeg zadatka u IEEE formatu, npr:

- [1] V.A. Pedroni, Circuit Design with VHDL, Cambridge, MA, USA, The MIT Press, August 2004.
- [2] C. Talarico, VHDL guidelines for synthesis, SIEMENS Semiconductor group, Sophia-Antipolis, FRANCE
- [3] J. Korević, Arhitektura i organizacija računara - Pipeline
- [4] <http://esd.cs.ucr.edu/labs/tutorial/>
- [5] <http://esd.cs.ucr.edu/labs/tutorial/>

7. PRILOZI

Skica šeme celokupnog sistema:

1. CPU
 - IF
 - ID
 - EX
 - MEM
 - WB
 - Hazard controller
 - Predictor
 - Register file
2. Instruction cache
3. Data cache

Svi detalji entiteta i signala su detaljno nacrtani uz sve potrebne signale na šemama koje se nalaze u okviru priloga ovog rešenja.