

8. rujna 2016.

Zavod za elektroniku, mikrelektroniku,  
računalne i inteligentne sustave

## Arhitektura računala 2

pismeni ispit

1. Oblikujte i implementirajte mikroprogram za instrukciju DBJNZ <sup>DBJNZ increment</sup> (~~decrement~~ b and jump relative if not zero). Neka je operacijski kod instrukcije DBJNZ \$23, te neka se mikroprogram za fazu pribavi nalazi na adresi \$ef. ~~Navedite kako bismo trebali izmijeniti put podataka ako bismo htjeli izvesti analognu instrukciju DAJNZ koja bi umjesto registra B koristila A.~~
2. U memoriji računala s pojednostavnjenim modelom procesora, počevši od adrese \$0020, nalazi se sljedeći niz heksadekadskih bajtova: \$22, \$20, \$24, \$25, \$26, \$20. Poznati su sljedeći heksadekadski operacijski kodovi instrukcija koje adresiraju nultu stranicu: INC - \$20, DEC - \$21, LDA - \$22, ADD - \$23, STA - \$24, SUB - \$25 i JMP - \$26. Odredite ponašanje programa i skicirajte stanje na sabirnici kroz 12 taktova signala vremenskog vođenja.
3. Predložite prijevod priloženog potprograma za arhitekturu MIPS pod pretpostavkom da se argumenti p, p2 i q prenose preko registara \$4, \$5 i \$6. Prikažite dvije verzije potprograma: jednu neoptimiranu te jednu optimiranu za procesor s jednostrukim izdavanjem.

`void main(int *p, int *p2, int *q) {`

...operacijski kodovi instrukcija koje adresiraju nultu stranicu: INC - \$20, DEC - \$21, LDA - \$22, ADD - \$23, STA - \$24, SUB - \$25 i JMP - \$26. Odredite ponašanje programa i skicirajte stanje na sabirnici kroz 12 taktova signala vremenskog vođenja.

3. Predložite prijevod priloženog potprograma za arhitekturu MIPS pod pretpostavkom da se argumenti p, p2 i q prenose preko registara \$4, \$5 i \$6. Prikažite dvije verzije potprograma: jednu neoptimiranu te jednu optimiranu za procesor s jednostrukim izdavanjem.

```
void memcpy(int *p, int *p2, int *q){  
    while (p!=p2){  
        *q=*p;  
        ++p;  
        ++q;  
    }  
}
```

4. Prosječno vrijeme pristupa memoriji na nekom računalu s jednorazinskom priručnom memorijom je 2.4 ciklusa takta. Pri tome, u slučaju pogotka, podatku se može pristupiti u 1 periodu takta, a u slučaju promašaja, potrebno je 80 perioda za pristup glavnoj memoriji.

Koliki mora biti postotak pogodaka u drugoj razini priručne memorije s vremenom pristupa od 6 perioda takta ako prosječno vrijeme pristupa želimo poboljšati za 65%?

*Per formansu*

5. Prikažite izvođenje dretvi prikazanih slikom na tri superskalarna procesora koji podržavaju:

- (a) simultanu višedretvenost;
- (b) finozrnatu višedretvenost;
- (c) grubozrnatu višedretvenost s prekidima na vertikalno neiskorištenim priključcima.

Pretpostavite da protočne strukture triju procesora nisu specijalizirane te da svaki od procesora može izdati do četiri instrukcije u svakom periodu signala takta.



31	29	26	24	22	19	17	15	13	7	0
CA	CB	COP	CSH	CMB	CAB	CBB	CST	CNA	CEM	

CA	CB	COP	CSH
00 ... nema prijenosa 01 ... $L \leftarrow [0, F(CEM)]$ 10 ... $L \leftarrow [F(CEM), 0]$ 11 ... $L \leftarrow A$	000 ... nema prijenosa 001 ... $R \leftarrow B$ 010 ... $R \leftarrow B^*$ 011 ... $R \leftarrow PC$ 100 ... $R \leftarrow SR$ 101 ... $R \leftarrow 0$	00 ... suma uz $C=0$ 01 ... suma uz $C=1$ 10 ... ne koristi se 11 ... ne koristi se	00 ... $MB \leftarrow S$ 01 ... $MB \leftarrow shr S$ 10 ... $MB \leftarrow shl S$ 11 ... $MB \leftarrow IN$
CMB	CAB	CBB	CST
000 ... nema prijenosa 001 ... $A \leftarrow MB$ 010 ... $B \leftarrow MB$ 011 ... $PC \leftarrow MB$ 100 ... $SR \leftarrow MB$ 101 ... $OUT \leftarrow MB$	00 ... $H(1) \leftarrow 0$ 01 ... $H(1) \leftarrow 1$ 10 ... $H(1) \leftarrow SR(1)$ 11 ... $H(1) \leftarrow SR(0)$	00 ... $H(0) \leftarrow 0$ 01 ... $H(0) \leftarrow 1$ 10 ... $H(0) \leftarrow SR(1)$ 11 ... $H(0) \leftarrow MB(15)$	00 ... SR se ne mijenja 01 ... $SR(0) \leftarrow ZT$ 10 ... $SR(1) \leftarrow MB(15)$ 11 ... $SR(0) \leftarrow ZT$ $SR(1) \leftarrow MB(15)$

110  $PR \leftarrow MB$

Slika uz zadatak 1: mikroinstrukcijska riječ.



