

ZAPISI

PV 1. bit = predznak
ostali so veličnost

ODMIK 2^{n-1}
• $[2] \rightarrow [10]$ - odmik
• $[10] + \text{odmik} \rightarrow [2]$

DEC JEVIJE
• oče. e.
• deli mantise (v des.)
 $E \text{ max vrednost } (255/2047)$
 $\rightarrow m = 0 \rightarrow \infty$
 $\rightarrow m \neq 0 \rightarrow N \text{ a } N$

DE NORMIRANA
 $E = 0 \text{ imp. bit.} = 0$
 $e = -126 \text{ namesto } -127$
• L z vsakim bitom D
• na koncu VSOTA

REG.	NAMEN
r0	nula
r1-23	splošni namen
r24	prvi param za podprogram
r25	drugi param za podprogram
r26	bazni za dolge skoke
r27	bazni za dolge klice
r28	vred. ki jo vrača podprogram
r29	kazalec na okvir
r30	skladovni kazalec
r31	povratni naslov

SKOK IN RAZVELJAVLJANJE

bne ... IF ID EX MEM WB
IF ID ...
IF ...
razveljavljeno
novi ukazi (od tam, kamor smo skočili)
št. raz. u. = $(n-1) \cdot 2$
v zanki \rightarrow iteracije

POMIKI (SHIFT)
• LEVI $\leftarrow \sim \cdot 2^n$
• DESNI $\leftarrow \sim : 2^n$
• LOGIČNI: v izpraženem mestu grejo 0
• ARITMETIČNI: levi bit se ne spreminja (predznak se ohranja)

1'K (negacija)
 \oplus : enako kot PV
 \ominus : invert ali pa oče. $2^n - 1$
PRELIV
potrebno: enak predznak
zadostno: vsota drug p.z.

MNOŽ.
• e sešt. \rightarrow začošaj
• mantisi množimo (z imp. bitom!)
• in normiramo
• predznak XOR

BIN. MNOŽENJE

$x_2 x_1 x_0 \cdot y_2 y_1 y_0$
 $x_2 y_2 \quad x_1 y_1 \quad x_0 y_0$
 $x_2 y_1 \quad x_1 y_0 \quad x_0 y_0$
konj
↓
 $x_2 y_0 \quad x_1 y_0 \quad x_0 y_0$
• prva 2 param (L proti D) shrani v r24 in r25
• ostale param porine na sklad (D proti L)
• izvede klic (ukaz CALL)

PUSH reg
sv 0(r30), reg
subui r30, r30, #4

POP reg
addui r30, r30, #4
lw reg, 0(r30)

IF	ID	EX	MEM	WB
prevzem ukaza	dekodiranje ukaza	izvrševanje	dostop do pomnilnika	shranjevanje rezultata

trajanje prog.:
• št. ukazov: pred zanko + št. iter x ukazi v zanki + po zanki
• PN: pred zanko + št. iter x v zanki + po zanki
• KN: št. skokov x 2
• latenca $5 - 1 = 4$

ENDIAN - HIP
BIG: prvi (najtežji) del na najnižjem nasl.
LITTLE: najlažji del na najnižjem nasl.
m 3 0
m+1 2 1
m+2 1 2
m+3 0 3
 $b_3 b_2 b_1 b_0$

2'K $\cdot 1'K + 1$
 $\oplus [10] \rightarrow [2]$
 $\ominus [10] \rightarrow [2]$
 \rightarrow invert
 $\rightarrow +1$
 \rightarrow predznak
 \rightarrow predznak
 \rightarrow veličnost
 $\rightarrow -2^{n-1}$

FP
• zaokrožimo k najbližjemu še predstavljenemu št. (če je enako potem k sedemu)

SEŠT.
• 1. št. je isto z večim eks.
• pomik mantise 2. št.
• seštej mantisi
• prenos? \rightarrow zmanjšaj m in e

NAPAKA PRI REZANJU DEC.

• napaka: $N' - N$
• ne more preseči $r-k$
• zadostiti moramo $r-k \leq E_{max}$
• poiščemo k, da velja:

$k \geq \log_r(1/E_{max})$
 $k = \lceil \log_r(1/E_{max}) \rceil$

• s sklada pobriše param
(sklad. kaz. + 4 x št. prenešenih param)

KRATKI KLICI: call r31, podprog(r0)
DOLGI KLICI: lhi r27, #PODPROG
addui r27, r27, #PP
call r31, 0(r27)

premoščanje: operand dobimo že v EX/MEM
ukazi: ALE \rightarrow L/S

(v isti fazi/periodi je lahko naslednji (odvisni) ukaz v ID)

NASLAVLJANJE
v HIP (samo)
• TAKOŠNJE 2 načina
- 16b fazej. operand
- razširitev na 32 (predznak)
• BAZNO
- odmik D_i je 16b r2'K
- dejanski naslov: $A = Rb + D_i$
- bazni reg.: katerikoli

FP IEEE 754
des. natanč.: 7 mest
SP (32b) enojna
S E m
31 30 -23 22 - 0
DP (64b) dvojna 16 mest
S E m
31 30 -23 22 - 0

$E = e + 127$ $E = e + 1023$
 $(-1)^s \cdot 1, m \cdot 2^e$ prava 2⁹ odmikom
mantiso NORMIRANO
01101 \rightarrow 1,01101
prvi bit je 1, implicitni

N odrežemo na k decimalk, dobimo N'
• abs. napaka: $|N' - N|$
 $k \geq \log_r(1/E_{max})$
 $k = \lceil \log_r(1/E_{max}) \rceil$

PO VRATITVI IZ PODPROG.

premoščanje: operand dobimo že v EX/MEM
ukazi: ALE \rightarrow L/S
(v isti fazi/periodi je lahko naslednji (odvisni) ukaz v ID)

NASLAVLJANJE
v HIP (samo)
• TAKOŠNJE 2 načina
- 16b fazej. operand
- razširitev na 32 (predznak)
• BAZNO
- odmik D_i je 16b r2'K
- dejanski naslov: $A = Rb + D_i$
- bazni reg.: katerikoli

TAKOŠNJE 2 načina
- 16b fazej. operand
- razširitev na 32 (predznak)
• BAZNO
- odmik D_i je 16b r2'K
- dejanski naslov: $A = Rb + D_i$
- bazni reg.: katerikoli

$$CPI = \sum_{i=1}^n CPI_i \cdot p_i = \frac{\text{št. ciklov}}{\text{št. ukazov}} = \frac{\text{čas izvaj.} \cdot \text{frek}}{\text{št. ukazov}}$$

št. u.p. za ukaz i $\frac{f_{u_i}}{f_u}$
rel. pogostost vrste ukaza

čas izvajanja = CPE čas = $\frac{\text{št. ukazov}}{MIPS \cdot 10^6} = \text{št. ukaz.} \cdot CPI \cdot t_{CPE}$

$f_{CPE} = \frac{1}{t_{CPE}}$

$$MIPS = \frac{f_{CPE}}{CPI \cdot 10^6} = \frac{1}{CPI \cdot t_{CPE} \cdot 10^6}$$

u.p. (čas za izvedbo 1 cikla)

$CPI_{idealni} = \frac{\text{št. ukaz.} \cdot CPI}{f_{CPE}}$

$N = \text{št. pom. dostopov (vsah/R/W)}$
 $M_i = \text{povp. št. operandnih pom. dostopov na ukaz}$
 $CPI_{idealni}$ predpostaviti, da ni zgrešitev

POVEČANJE ŠT. UKAZOV V DANEM ČASU: (latenca)

- manjši CPI (čeprav je trajanje ukaza enako ($N \cdot t_{CPE}$))
- krajši t_{CPE} : če uspemo narediti enostavne podop.

čas shran. rez. v reg. $t_{CPE} = t_{SHRANJEVANJE} + t_{PODOPERACIJA}$

z več stopnjami lahko zmanjšamo $t_{PODOPERACIJA}$, $t_{SHRANJEVANJE}$ na NE

$I = \text{št. ukazov}$
 $H = \text{povp. verj. zad.}$
 $K_z = \text{povp. zgr. kazen}$
 $B = 2^b = \text{velikost bloka}$
naslov besede 0 je vedno večkratnik dolžine bloka

$S = \text{št. setov} = 2^s$ stopnja asociativnosti
 $E = 2^e = \text{št. blokov v setu}$ (št. AP v setu = št. primerjal.)
 $M_b = S \cdot E = 2^{s+e} = \text{št. blokov v PP}$
 $M = M_b \cdot B = S \cdot E \cdot B = 2^{s+e+b}$ pom. besed

CPE čas = (CPE periode izvir + CPE periode čak) $\cdot t_{CPE}$

CPE per izvir = $I \cdot CPI_{ideal}$ $N = I \cdot (1 + M_i)$

CPE per čak = $N \cdot (1 - H) \cdot K_z$

CPE čas = $I \cdot (CPI_{ideal} + M_i \cdot (1 - H) \cdot K_z) \cdot t_{CPE}$

$H_R \neq H_W$: CPE per čak = $N_R \cdot (1 - H_R) \cdot K_{zR} + N_W \cdot (1 - H_W) \cdot K_{zW}$

$H_{Upp} \neq H_{Opp}$: št. zgr.: $N \cdot (1 - H) \rightarrow I \cdot (1 - H_{Upp}) + I \cdot M_i \cdot (1 - H_{Opp})$

verj. zgr.: $1 - H = 1 - H_{Upp} + M_i \cdot (1 - H_{Opp})$

CPE per čak = $(I \cdot (1 - H_{Upp}) + I \cdot M_i \cdot (1 - H_{Opp})) \cdot K_z$

PP povečujemo s povečevanjem S/E/B (najlažje B)

čas dostopa = $t_a = t_{ap} + (1 - H) t_{ag}$ $t_a = t_{ap} + (1 - H) t_b$ čas za prenos bloka

$H = \frac{N_p}{N} = \frac{N_p}{N_p + N_g}$ UVEDBA L2 (zmanjša zgr. kazen)

CPE per čak = $N \cdot (1 - H_{L1}) \cdot K_{zL1}$ lokalna

$K_{zL1} = t_{BzL1} + (1 - H_{L2}) \cdot K_{zL2}$ verjetnost zgrešitve

bloka iz L2 v L1 (zad. v L2)

globalna verj. zgr. v L2 = $(1 - H_{L1}) \cdot (1 - H_{L2})$

$t_a = t_{apL1} + (1 - H_{L1}) t_{apL2} + (1 - H_{L1}) (1 - H_{L2}) \cdot t_{ag}$

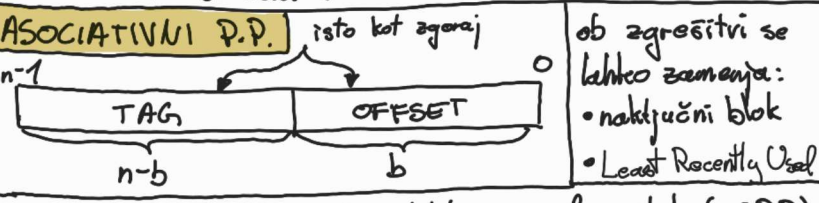
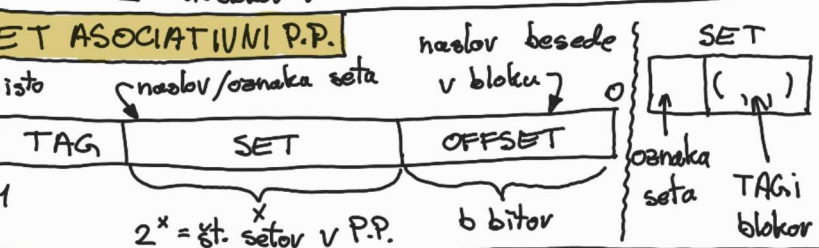
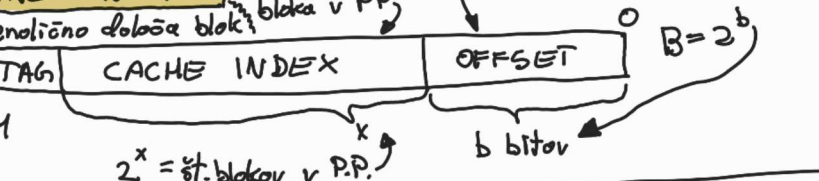
AMDAHLLOV ZAKON (delež)

$f \dots$ zaporedni del prog.
 $(1 - f) \dots$ vzporedni del prog.

dolovanje je N-krat hitrejšo (npr. paral. N CPU)

$S(N) = \frac{1}{f + (1 - f)/N} = \frac{N}{1 + (N - 1)f}$

pohitritev (odvisna od N)
ta del se da paralelizirati
pogostejšo na N rač. ... zmanjšano čas



po TAG-u ugotovljamo ali je blok v predpomnilniku (vsi P.P.)

ZGREŠITVE

- OBVEZNE: prvi dostop (bloka ni v PP)
- KONFLIKTNE: zamenjava blokov, ki se preslikajo v isti set
 - pri set asoc. in direkt., pri asoc. jih ni
 - ko je treba zamenjati blok, ki se bo kmalu spet uporabil - ker se oba bloka preslikata v isti set

Št.	Op. koda	func	Format	Ukaz	Št.	Op. koda	func	Format	Ukaz
0	000000	x	1	ADDI	36	100100	x	1	LB
1	000001	x	1	SUBI	37	100101	x	1	LH
2	000010	x	1	ADDUI	38	100110	x	1	LW
3	000011	x	1	SUBUI	39	100111	x	1	BEQ
4	000100	x	1	ANDI	40	101000	x	1	SB
5	000101	x	1	ORI	41	101001	x	1	SH
6	000110	x	1	XORI	42	101010	x	1	SW
7	000111	x	1	LHI	43	101011	x	1	
8	001000	x	1	SEHI	44	101100	x	1	J
9	001001	x	1	SNEI	45	101101	x	1	CALL
10	001010	x	1	SLTI	46	101110	x	1	TRAP
11	001011	x	1	SGTI	47	101111	x	1	RFE
12	001100	x	1	SCTUI	48	110000	0	2	ADD
13	001101	x	1	SCTUI	49	110000	1	2	SLL
14	001110	x	1		49	110001	0	2	SUB
15	001111	x	1		49	110001	1	2	SRL
16	010000	x	1	SLLI	50	110010	0	2	ADDU
17	010001	x	1	SRLI	51	110010	1	2	SRA
18	010010	x	1	SRAI	51	110011	0	2	SUBU
19	010011	x	1		51	110011	1	2	EI
20	010100	x	1		52	110100	0	2	AND
21	010101	x	1		53	110100	1	2	DI
22	010110	x	1		53	110101	0	2	OR
23	010111	x	1		53	110101	1	2	MOVER
24	011000	x	1		54	110110	0	2	XOR
25	011001	x	1		54	110110	1	2	MOVRE
26	011010	x	1		55	110111	x	2	
27	011011	x	1		56	111000	0	2	SEQ
28	011100	x	1		57	111001	0	2	SNE
29	011101	x	1		58	111010	0	2	SLE
30	011110	x	1		59	111011	0	2	SGT
31	011111	x	1		60	111100	0	2	SLTU
32	100000	x	1	LBUI	61	111101	0	2	SGTUI
33	100001	x	1	LHI	62	111110	0	2	NOT
34	100010	x	1		63	111111	x	2	
35	100011	x	1	BNE					