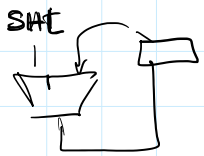
martedì 7 novembre 2017 16:19

ALU { aritmetico logiche corte (ADD SUB AND OR SHR SHL ..)
" " lunghe (MUL DIV)

Ch.

mul∅. // μ-programma che calcola IR.Ra IR.Rb IR.Rc
REG[Ra] * REG[Rb] → REG[Rc]

32

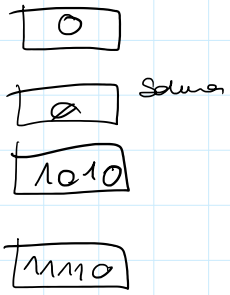


12 × 34
48
36-
408

2 3
10 × 11
10
100
110
4+2

32 come (SHL)

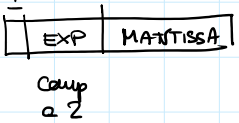
32 × 32
64

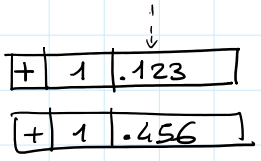


101 × 110
shl 1010
10100
1 1110
3 iterazioni

50 τ
mul∅.
mul 1.

2 2
1.23 × 4.56





.XXXX

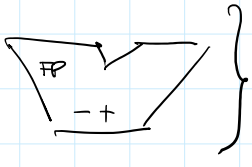
1.23 45.6

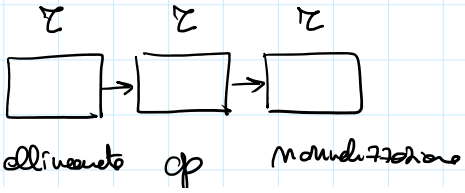
1.23
45.60
46.83

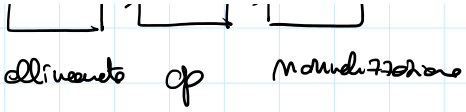
+ 2 .4683

1,23
4.56
5,79

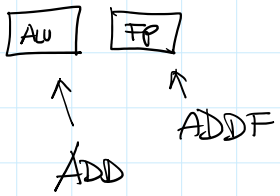
+ 1 .579

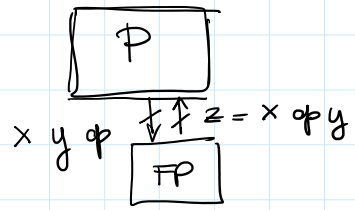






8086
x86





~~8~~286 287

martedì 7 novembre 2017 16:44

$$T = \sum_{i=0}^{n-1} (k_i \tau \cdot p_i)$$

k_i: # cicli spesi nell'interpete per calcolare uno ISTR D-RISC; p_i: percentuali delle istr nel codice "tipo"

$2\tau + t_a + \tau = 3\tau + t_a$

ARITM - LOG CORTE

$T_{ch_0} + T_{ch_1} + T_{exec}$ (singolo τ)

$T_{ch_0} = \tau$

$T_{ch_1} = t_a + \tau$

chiamata e la decodifica costano $2\tau + t_a$

ARITM LOGICHE LUNGHE $52\tau + t_a$

$T_{ch_0} + T_{ch_1} + T_{exec}$

$2\tau + t_a + 50\tau$

$4\tau + 2t_a$

MEMORIA (LD, ST)

$T_{ch_0} + T_{ch_1} + T_{exec}$

$2\tau + t_a$ $2\tau + t_a$

$3\tau + t_a$

SALTO

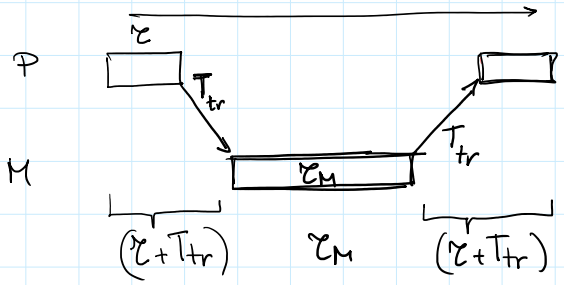
INCONDIZ. $\frac{CH}{2\tau + t_a} + \frac{exec}{\tau}$

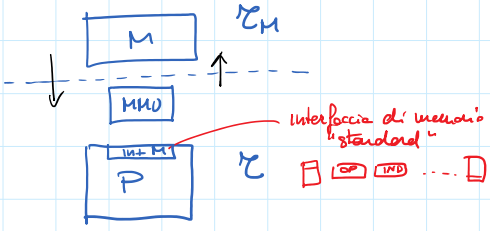
CONDIZIONATI $\frac{CH}{2\tau + t_a} + \frac{exec}{2\tau}$

$4\tau + t_a$

$t_a = 2(\tau + T_{tr}) + \tau_M$ (10^τ N, 50^τ)

$= 72\tau$





T

$B = \frac{1}{T}$

Alcate	lughe	LD/ST	GOTO	IF
40%	10%	30%	10%	10%

$$T = 8.3\tau + 1.3 t_a$$

MIX

$T_{M1} \overset{?}{<=>} T_{M2}$

k_i τ p_i

↑

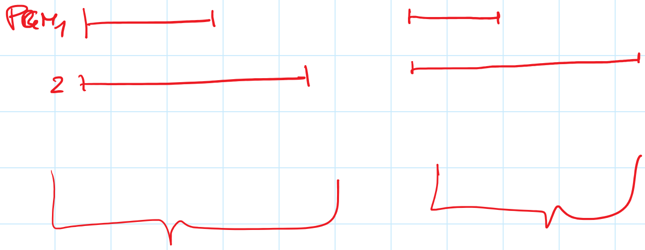
CPI

$B = \frac{1}{T}$ } **MIPS** / MFLOPS

BENCHMARK

{ programmi } PGM_1 ⋮ PGM_m

M_1 M_2



tempo di completamento

$T_c = \#\text{istruzioni} \cdot T$

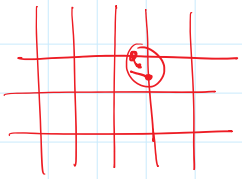
MEMORIE

STATICHE

porte AND/OR
6

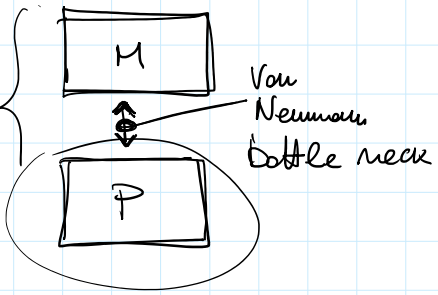
molto costose
molto veloci 5-6 T
non hanno bisogno del refresh

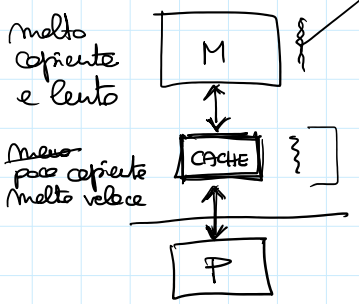
DINAMICHE



poco costose
lente
refresh

80 T





PC

#stati o(100) #var di cond o(10)

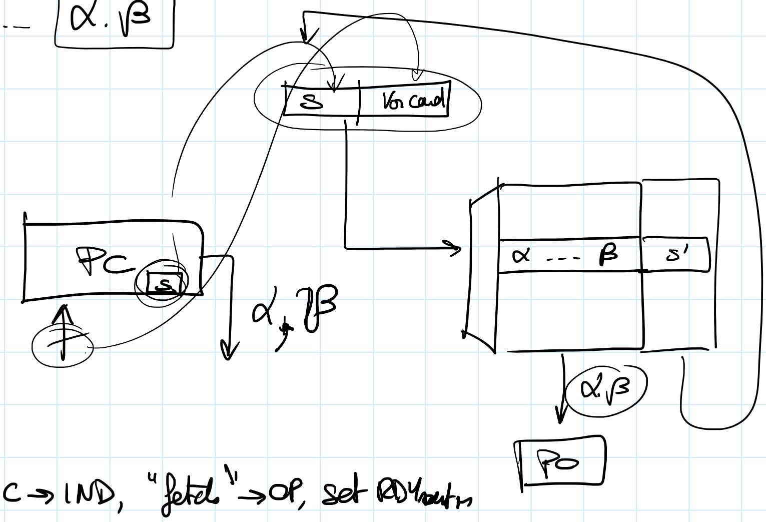
ch0. $\alpha_1 = \alpha_2 = \alpha_3 = \beta_1 \ldots$ α.β

ch1.

add0.
sub0
shl0
shr0.

ld0
ld1
st0
st1
st2

Goto0
if<0
if<1
if>0
if>1



ch0. IC → IND, "fetch" → OP, set RDYout

INT OR(ESITO) RDYinM F