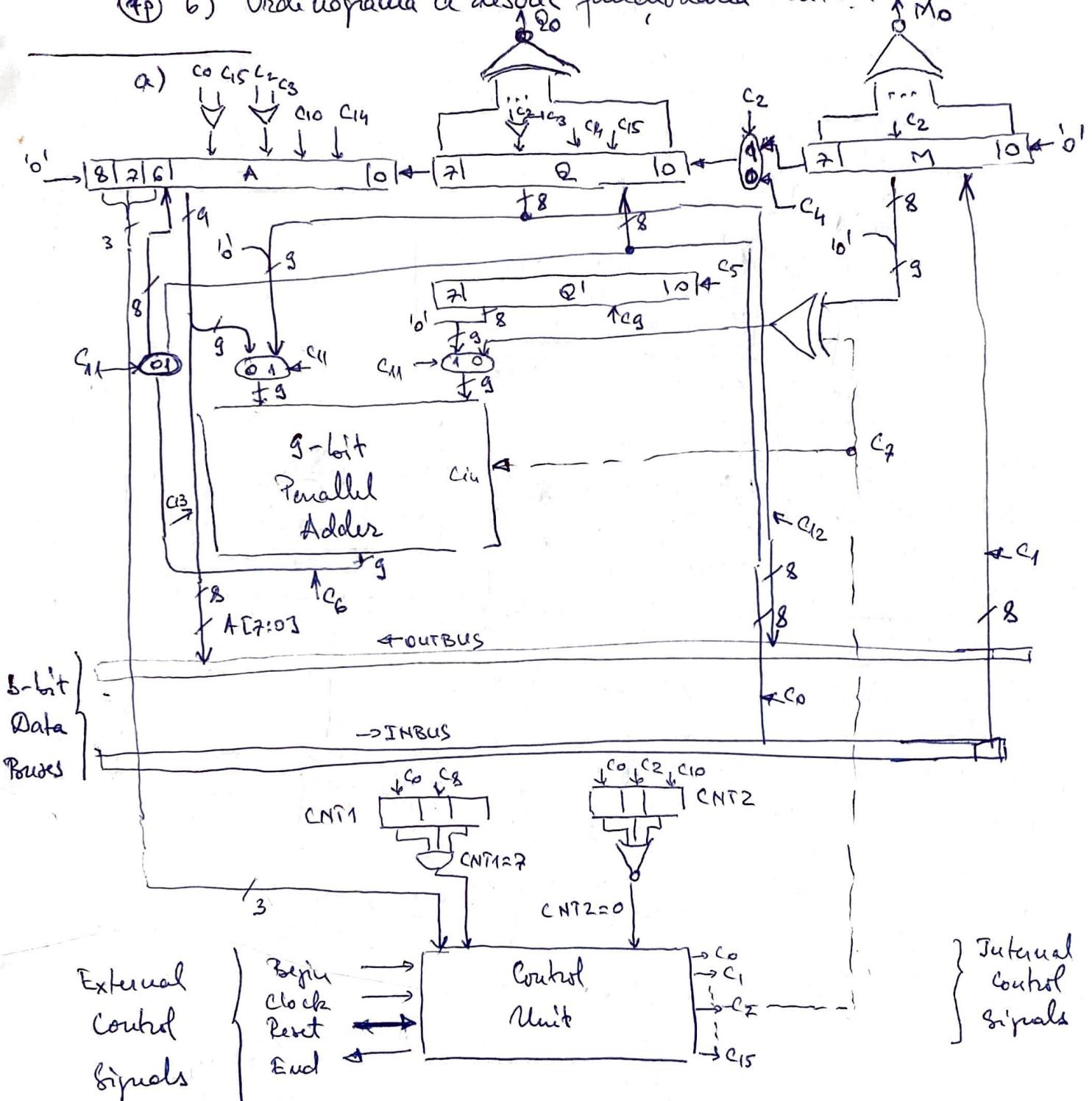


2p) ① Să se completeze descrierea unui împărțitor SRT radix 2, astfel încât să se trateze cazurile speciale de împărțire la zero (când atât cîntul cît și restul sunt "1...1") și împărțirea a lui zero. Se cere:

A2
B1

①p) a) Schema bloc hardware + semnale de control.

②p) b) Ordinograma ce descrie funcționarea unității de control.



b)

Start/
stop

Zähler

Bereit

C0

$A := 0, CNT1 := 0, CNT2 := 0,$
 $Q := INBUS, Q' := 0;$

C1

 $M := INBUS;$

C14

$M0 \rightarrow 1 | A := "1...14";$
 $Q := "1...14";$
 $Q0 \rightarrow 1 | A := 0,$
 $Q := 0;$

C15

1

0

$M[7] \rightarrow 0 | AC8[0:7], Q, M[7:1] :=$

$A[7:0], Q, MC2[0];$

$MC0[0] := '0', CNT2 := CNT2 + 1;$

C2

001
010
011

C3, C4

100

101

110

C3, C5

$A, Q[7:1] := A[7:0], Q[7:0],$
 $Q'[7:1] := Q'[6:0],$
 $Q'[0] := '0, Q[0] := '1;$

000
111

Q3

$A, Q[7:1] := A[7:0], Q[7:0],$
 $Q[7:1] := Q[6:0],$
 $Q[0] := '0, Q[0] := '1;$

 $A := A - M$ $A := A + M;$ $CNT1 := CNT1 + 1;$

C6, C7

1

0

$A[7] \rightarrow 1 | A := A + M,$
 $Q' := Q + 1;$

C8

$CNT2 := CNT2 + 1;$

C10

$Q := Q - Q';$

$OUTBUS := Q;$

C12

$OUTBUS := A;$

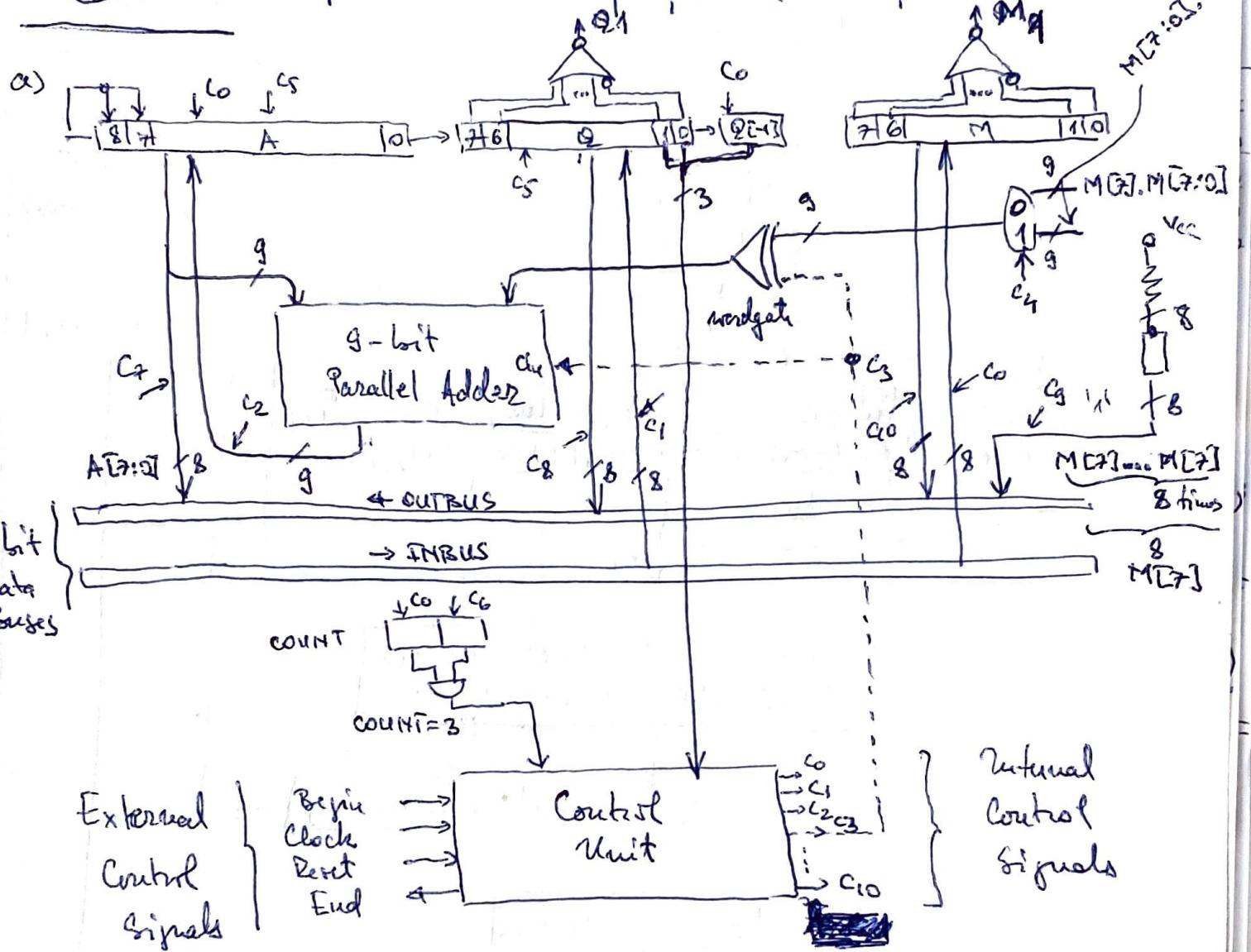
Start
stop End

RP ① Să se completeze descrierea unui înmulțitor Booth radix 4, astfel încât să trateze cazul special al înmulțirii cu 1 (unul din cele două operații este egal cu 1). Se cere:

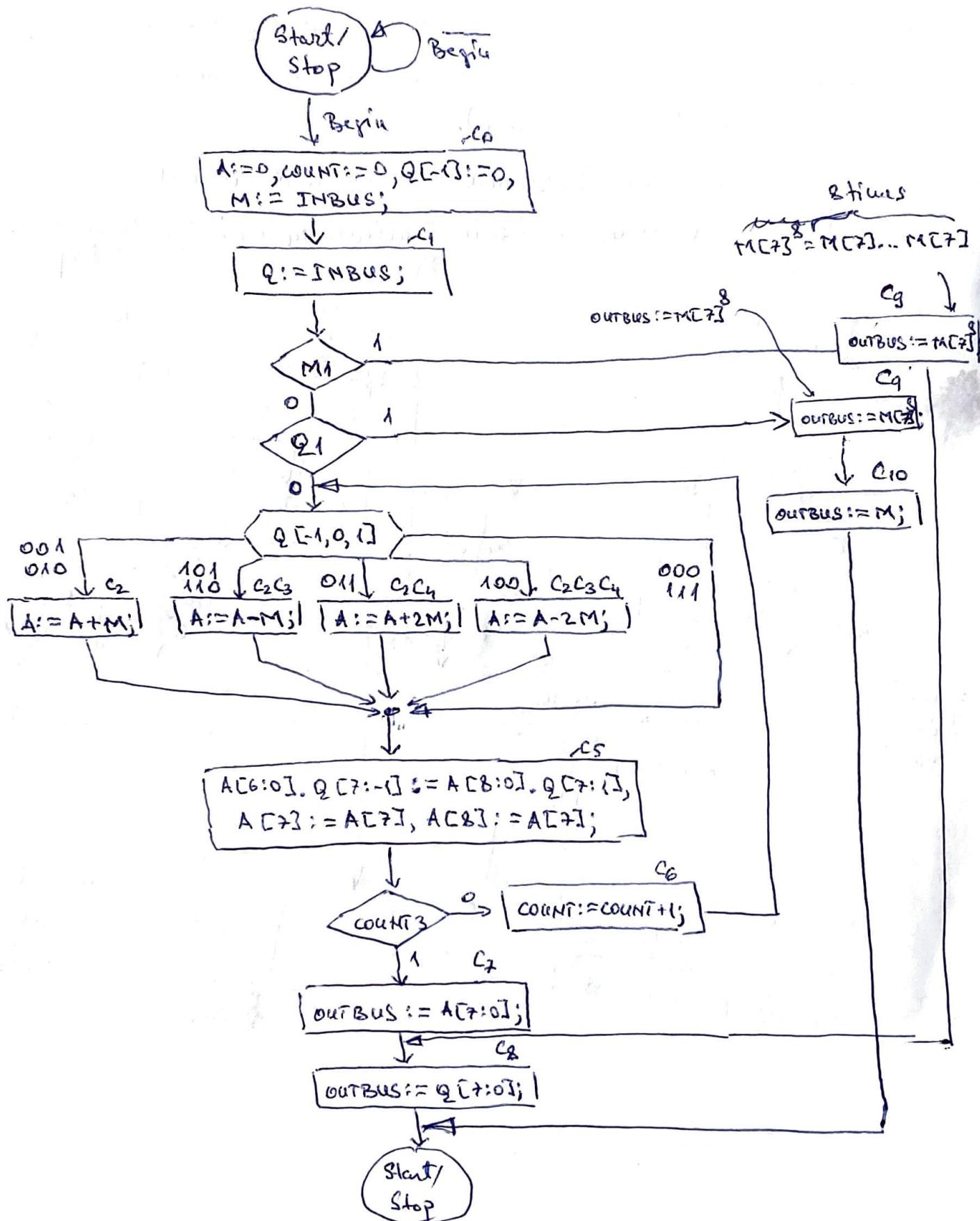
A1
B2

① a) Schema bloc hardware + semnalele de control

① b) Ordinea parcurgândă ce descrie funcționarea unității de control.



b)



2) Se dă o mașină de calcul A, pe care executăm A1, A2, B1, B2 un program ce constă în
 1000 (A1)
 2000 (A2)
 3000 (B1)
 4000 (B2)

Acste instrucțiuni sunt de 4 tipuri, conform tabelului de mai jos, unde sunt precizate frecvența, respectiv media de cicluri de clock pentru fiecare tip de instrucțiune.

	A1		A2		B1		B2	
Type	CPI	Freq	CPI	Freq.	CPI	Freq	CPI	Freq
Arith	4	40%	5	30%	4	50%	4	40%
CMP	2	30%	3	30%	3	20%	3	40%
Branch	2	20%	2	20%	3	20%	2	10%
Other	3	10%	2	20%	2	10%	2	10%

La mașină A, frecvența de clock este 2 GHz.

Un nou procesor cu o viteză ce implementează un set diferit de instrucțiuni, fapt ce determină o degradare a frecvenței de clock cu 15% la noua mașină de calcul (B).

Pe de altă parte, programul pe care îl rulează, determină la mașină B o reducere cu 10% a instrucțiunilor aritmetice și brațe, dar și a instrucțiunilor CMP cu 5%.

Mașina B reduce CPI pentru tip de instrucțiune cu 1 c.c. la "Other" și îl crește cu 1 c.c. la instrucțiunile CMP. La celelalte tipuri de instrucțiuni (Arith și Branch), CPI rămâne neschimbat. Să se compute cantitatea mașinile A și B după

- a) CPU time
- b) MIPS

$$\textcircled{2} \quad \boxed{\text{A1}} \quad \text{Clock cycle time} = \frac{1}{2 \cdot 10^9 \text{ Hz}} = 0.5 \text{ ns}$$

A1, A2, B1, B2

$$\text{a) CPU time}_A = 1000 \times (4 \times 0.4 + 2 \times 0.3 + 2 \times 0.2 + 3 \times 0.1) \times 0.5 \text{ ns}$$

$$= 1450 \text{ ns} = \boxed{1.45 \mu\text{s}}$$

A better

\textcircled{2} CPI times_B
IC instructions_B

$$\text{CPI Arith}_B = 4$$

$$\text{CPI CMAP}_B = 3$$

$$\text{CPI Branch}_B = 2$$

$$\text{CPI others}_B = 2$$

$$\text{CPU time}_B = \underbrace{(360 \times 4 + 285 \times 3 + 180 \times 2 + 100 \times 2)}_{1440 \quad 855 \quad 360 \quad 200} \text{ Clock cycle time}_B$$

$$\text{IC Arith}_A = 1000 \times 0.4 = 400$$

$$\text{IC CMAP}_A = 300$$

$$\text{IC Branch}_A = 200$$

$$\text{IC Others}_A = 100$$

$$\text{IC Arith}_A = \frac{400 - 40}{40} = 360$$

$$\text{IC CMAP}_B = \frac{300 - 15}{15} = 285$$

$$\text{IC Branch}_B = \frac{200 - 20}{20} = 180$$

$$\text{IC Others}_B = 100$$

$$\text{Clock cycle time} \quad \text{Clock rate}_B = \cancel{2.16 \text{ GHz} \times 1.15} \quad (2 + 0.15 \times 2) \text{ GHz}$$

$$= 1.7 \text{ GHz} =$$

$$\Rightarrow \text{Clock cycle time}_B = \frac{1}{1.7 \text{ GHz}} = \frac{1}{1.7 \times 10^9 \text{ Hz}} = \frac{1}{1.7} \text{ ns} =$$

$$= 0.588 \text{ ns}$$

$$\text{CPU time}_B = 2855 \times 0.588 \text{ ns} = 1678.74 \text{ ns} = \boxed{1.67874 \mu\text{s}}$$

$$\text{b) MIPS}_A = \frac{\text{IC} \times \text{Clock frequency}}{\text{CPI}_A \cdot 10^6} = \frac{\text{Clock frequency}}{\text{CPI}_A \cdot 10^6} =$$

$$= \frac{2 \cdot 10^9 \text{ Hz}}{2.9 \cdot 10^6} = 0.6865 \cdot 10^3 = \boxed{689.65}$$

$$\text{IC}_B = 360 + 285 + 180 + 100 = 925$$

$$\text{CPI}_B = \frac{2855}{925} = 3.0865$$

$$\text{MIPS}_B = \frac{3.0865}{3.0865 \cdot 10^6} = \boxed{550.785}$$

A2 (2)

3.8

a) CPU time_A = $2000 \times (5 \times 0.3 + 3 \times 0.3 + 2 \times 0.4) \times 0.5 \mu s =$
= 3200 μs = $3.2 \mu s$

A is better

CPU time_B → IC_{ArithA} ≈ 600 $\xrightarrow{-5\%}$ 540 = IC_{ArithB}
IC_{CMPA} ≈ 600 $\xrightarrow{-10\%}$ 570 = IC_{CMPB}
IC_{BranchA} ≈ 400 360 = IC_{BranchB}
IC_{Other} ≈ 400 → 400 = IC_{OtherB}

CPI_{ArithB} = 5 CPU time_B = $\frac{2700}{(540 \times 5 + 570 \times 3 + 360 \times 2 + 400)} \times 0.5 \mu s =$
CPI_{CMPB} = 4
CPI_{BranchB} = 2.
Other_B = 1
 $= 6100 \times 0.588 \mu s = 3586.8 \mu s = 3.5868 \mu s$

3.5868 μs

b) CPI_B = $\frac{6100 \text{ clock cycles}}{540 + 570 + 360 + 400 \text{ instructions}} = \frac{6100}{1870} = 3.262$

MIPS_A = $\frac{2 \cdot 10^9 \text{ Hz}}{3.2 \cdot 10^6} = 625$

> A is better

MIPS_B = $\frac{1.7 \cdot 10^9 \text{ Hz}}{3.262 \cdot 10^6} = 521.153$

(2)

$$2 + 1 \cdot 2 + 0 \cdot 2 = 3,4 = CPI_A$$

[B1]

a) $CPU_{time_A} = 3000 \times (\underbrace{4 \times 0,5 + 3 \times 0,2 + 3 \times 0,2 + 2 \times 0,1}_{2 + 1 \cdot 2 + 0 \cdot 2} \times 0,5 \mu s) = 5100 \mu s = \boxed{5,1 \mu s}$

A is better

$$IC_{Arith_A} = 1500$$

 $\xrightarrow{-10\%}$

$$1350 = IC_{Arith_B}$$

$$CPI_{Arith_B} = 4$$

$$IC_{CMP_A} = 600$$

 $\xrightarrow{-5\%}$

$$570 = IC_{CMP_B}$$

$$CPI_{CMP_B} = 4$$

$$IC_{Branch_A} = 600$$

 $\xrightarrow{-10\%}$

$$540 = IC_{Branch_B}$$

$$CPI_{Branch_B} = 3$$

$$IC_{Others_A} = 300$$

 $\xrightarrow{9600}$

$$300 = IC_{Others_B}$$

$$CPI_{Others_B} = 1$$

$$CPU_{time_B} = (1350 \times 4 + 570 \times 4 + 540 \times 3 + 300) \times 0.588 \mu s = 9600 \times 0.588 = 5644.8 = \boxed{5.6448 \mu s}$$

b) $MIPS_A = \frac{2 \cdot 10^9 \text{ Hz}}{3,4 \cdot 10^6} = \boxed{588,235}$

A is better

$$IC_B = 1350 + 570 + 540 + 300 = 2760$$

$$CPI_B = \frac{9600 \text{ c.c.}}{2760 \text{ inst.}} = 3,478$$

$$MIPS_B = \frac{1,7 \times 10^9}{3,478 \times 10^6} = \boxed{488,787}$$

②

B2

$$3 \cdot 2 = CPI_A$$

a) CPU time $= 4000 \times (\underbrace{4 \times 0.4 + 3 \times 0.4 + 2 \times 0.1 + 2 \times 0.1}_{A} \times 0.5 \mu s) = 8400 \mu s = \boxed{6.4 \mu s}$

A is better

$IC_{Arith_A} = 1600$	$\xrightarrow{-10\%} 1440 = IC_{Arith_B}$	$CPI_{Arith_B} = 4$
$IC_{CMPE_A} = 1600$	$\xrightarrow{-5\%} 1520 = IC_{CMPE_B}$	$CPI_{CMPE_B} = 4$
$IC_{Branch_A} = 400$	$\xrightarrow{-10\%} 360 = IC_{Branch_B}$	$CPI_{Branch_B} = 2$
$IC_{Other_A} = 400$	$400 = Other_B$	$CPI_{Other_B} = 1$

$12960 \text{ clock cycles}$

$IC_B = \frac{5760}{1440 \times 4} + \frac{6080}{1520 \times 4} + \frac{720}{360 \times 2} + 400 = 3720$

$CPU \text{ time}_B = (12960 \times 0.588 \mu s) = \boxed{7.62 \mu s}$

$$IC_B = 1440 + 1520 + 360 + 400 = 3720 \text{ clock cycles}$$

b) $CPI_B = \frac{12960 \text{ clock cycles}}{IC_B} = \frac{12960}{3720} = 3.4839$

$MIPS_A = \frac{3.2 \cdot 10^9 \text{ Hz}}{3.2 \cdot 10^6} = \boxed{625}$

A is better

$MIPS_B = \frac{1.7 \cdot 10^9 \text{ Hz}}{3.4839 \cdot 10^6} = \boxed{487.959}$

A1, A2, B1, B2

③ Un sistem de calcul are un procesor ce trimitе referinte către memoria cache la o frecvență de 10^8 word/second, unde 10% sunt scieri și 90% citiri.

2P Cache-ul are un miss rate de 10% și un block de 16 curenți.

BUS-ul are un bandwidth de 10^9 bytes/secundă și o latime (bus bandwidth) de 1 curant.

La un moment, 20% din blocuri sunt dirty.
Se calculează procentajul din bus bandwidth folosit din cauză imperfecțiunii cache-ului în următoarele căzuri.

1P a) Cache-ul este WB + Write Allocate

A1 1P b) Cache-ul este WT + Write No Allocate

B2 1P The rest as in A1 + un contează procentajul de blocuri dirty

a) Cache-ul este WT + Write Allocate

b) Cache-ul este WT + Write No Allocate

1P

A2 Miss rate = 8% și un block de 16 curenți

20% Scieri 80% Citiri ; Bus bandwidth = 4 words

10% Blocuri dirty

a) Cache-ul este WB + Write Allocate

b) Cache-ul este WT + Write No Allocate

The rest as in A2 + un contează procentajul de blocuri dirty

B1 1P a) Cache-ul este WT + Write Allocate

b) Cache-ul este WT + Write No Allocate

1P

③ A1

a) Accesses via the Bus = $10^8 \text{ words/sec} \times 0.1 \times (0.1 \times \text{Write Miss Penalty} + 0.9 \times \text{Read Miss Penalty})$

$$\text{Write Miss Penalty} = \underbrace{0.2 \times 8 \text{ BUS Writes}}_{\cancel{\text{Allocate Update}}} + \underbrace{8 \text{ BUS Reads}}_{\cancel{\text{Allocate}}} = 1.2 \times 8 \text{ BUS Accesses} = 9.6 \text{ BUS Accesses}$$

$$\text{Read Miss Penalty} = \underbrace{0.2 \times 8 \text{ BUS Writes}}_{\text{Update}} + \underbrace{8 \text{ BUS Reads}}_{\text{Allocate}} = 1.2 \times 8 \text{ BUS Accesses} = 9.6 \text{ BUS Accesses}$$

$$\Rightarrow \text{Accesses via the Bus}_{WB+WA} = 10^7 \times 9.6 \text{ acc/sec.}$$

$$\% \text{ Used Bus}_{WB+WA} = \frac{10^7 \times 9.6}{10^{10}} = \frac{9.6}{10^3} = \boxed{0.96\%} [9.6\%]$$

b) Accesses via the Bus $WT+WNA =$

$$= 10^8 \text{ words/sec} \times 0.1 \times (0.1 \times \text{Write Miss Penalty} + 0.9 \times \text{Read Miss Penalty})$$

$$+ 10^8 \text{ words/sec} \times 0.9 \times 0.1 \times \text{Write Hit Penalty}$$

No ^{miss} allocation, no update!

$$\text{Write Miss Penalty} = 1 \text{ BUS Write} = 1 \text{ BUS Acc}$$

$$\text{Read Miss Penalty} = 8 \text{ BUS Reads} = 8 \text{ BUS Acc.}$$

$$\text{Write Hit Penalty} = 1 \text{ BUS Write} = 1 \text{ BUS Acc.}$$

$$\text{Accesses via the Bus}_{WT+WNA} = 10^7 \times (0.1 \times 1 + 0.9 \times 8) + 10^7 \times 0.9 \times 1 = 10^7 (7.3 + 0.9) = 8.2 \times 10^7$$

$$\% \text{ Used Bus}_{WT+WNA} = \frac{8.2 \times 10^7}{10^9} = \frac{8.2}{100} = \boxed{8.2\%}$$

③ B2

$$\text{a) BUS Accesses}_{WT+WA} = 10^8 \times 0.1 \times (0.1 \times \text{Write Miss Penalty} + 0.9 \times \frac{\text{Read Miss Penalty}}{\text{Penalty}}) + 10^8 \times 0.1 \times 0.9 \times \text{Write Hit Penalty}$$

$$\text{Write Miss Penalty} = \underbrace{8 \text{ BUS Reads}}_{\text{Allocate}} + 1 \text{ BUS Write} = 9 \text{ BUS accesses.}$$

$$\text{Read Miss Penalty} = \underbrace{8 \text{ BUS Reads}}_{\text{Allocate}} = 8 \text{ BUS accesses}$$

$$\text{Write Hit Penalty} = 1 \text{ BUS Write} = 1 \text{ BUS Access}$$

$$\Rightarrow \text{BUS Accesses}_{WT+WT} = 10^8 \times (0.1 \times 9 + 0.9 \times 8) + \cancel{10^8 \times 0.9 \times 1} = 10^8 \times 8.1 + 10^8 \times 0.9 = 9 \times 10^7$$

$$\% \text{ BUS Access}_{WT+WA} = \frac{9 \times 10^7}{10^9} = \frac{9}{100} = 9\%$$

b)

$$\text{BUS Accesses}_{WT+WNA} = 10^8 \times 0.1 \times (0.1 \times 1 + 0.9 \times 8) + 10^8 \times 0.9 \times 1 = 8.2 \times 10^7$$

$$\% \text{ BUS Access}_{WT+WNA} = \frac{8.2 \times 10^7}{10^9} = 8.2\%$$

③ A2

a) Access via the BUS_{WB+WA} = $10^8 \text{ words/sec} \times 0.08 \times$
 $\times (0.2 \times \text{Write miss Penalty} + 0.8 \times \text{Read miss Penalty})$

Write miss Penalty = $0.9 \times \frac{16 \text{ Bytes/Block}}{\frac{16}{4} \text{ words bus width}} \text{ Bus writes} + \frac{16}{4} \text{ Bus reads} =$
 $\underbrace{\hspace{10em}}_{\text{Update}} \quad \underbrace{\hspace{10em}}_{\text{Allocate}}$
 $= (0.1 \times 4 + 4) \text{ Bus accesses} = 1.1 \times 4 = 4.4 \text{ Bus accesses}$

Read miss Penalty = Write miss Penalty = 4.4 Bus accesses

BUS Accesses_{WB+WA} = $10^7 \times 0.8 \times 4.4 = 3.52 \times 10^7 \text{ BUS accesses/sec}$

$\Rightarrow \% \text{ BUS Used}_{WB+WA} = \frac{3.52 \times 10^7}{10^9} = \frac{3.52}{100} = \boxed{3.52\%}$

b) Accesses via the BUS_{WT+WNA} =

= $10^8 \text{ words/sec} \times 0.08 \times (0.2 \times \text{Write miss Penalty} + 0.8 \times \text{Read miss Penalty})$,

+ $10^8 \text{ words/sec} \times 0.92 \times 0.2 \times \text{Write hit Penalty}$

No write, no allocation, no update!

Write miss Penalty = 1 BUS Write = 1 BUS Acc

Read miss Penalty = 4 BUS Reads = 4 BUS Acc

Write hit Penalty = 1 BUS Write = 1 BUS Acc

Accesses via the BUS_{WT+WNA} = $10^7 \times 0.8 \times (0.2 \times 1 + 0.8 \times 4) +$
 $+ 10^7 \times 0.92 \times 2 = 10^7 \times 2.72 + 10^7 \times 1.84 = 4.56 \times 10^7$

$\% \text{ BUS Used}_{WT+WNA} = \frac{4.56 \times 10^7}{10^9} = \boxed{4.56\%}$

B1 3

a) $\text{BUS Accesses}_{WT+WA} = 10^8 \times 0.08 \times (0.2 \times \text{Write miss Penalty} + 0.5 \times \text{Read miss Penalty}) + 10^8 \times 0.92 \times 0.2 \times \text{Write Hit Penalty}$

Write miss Penalty = $\underbrace{4 \text{ BUS Reads}}_{\text{Allocated}} + 1 \text{ BUS Write} = 5 \text{ BUS accesses}$

Read miss Penalty = $\underbrace{4 \text{ BUS Reads}}_{\text{Allocated}} = \underbrace{4 \text{ BUS accesses}}_{\text{Allocated}}$

Write Hit Penalty = 1 BUS Write = 1 BUS access

$$\begin{aligned}\text{BUS Accesses}_{WT+WA} &= 10^8 \times (0.2 \times 5 + 0.8 \times 4) + 10^8 \times 0.92 \times 1 = \\ &= 10^7 \times 0.8(0.2 \times 5 + 0.8 \times 4) + 10^7 \times 0.8 \times 0.92 = \\ &= 10^7 \times 3.36 + 10^7 \times 0.736 = 4.096 \times 10^7\end{aligned}$$

$$\% \text{ BUS Used}_{WT+WA} = \frac{4.096 \times 10^7}{10^9} = \boxed{4.096\%}$$

b) $\text{BUS Accesses}_{WT+WA} = 10^7 \times 0.8 \times (0.2 \times 1 + 0.8 \times 4) + 10^7 \times 0.92 \times 2 \times 1 = 4.56 \times 10^7$

$$\% \text{ BUS Used}_{WT+WA} = \frac{4.56 \times 10^7}{10^9} = \boxed{4.56\%}$$

(4) Considerăm un computer ce are un spațiu de memorie virtual de 4 TiB și un spațiu de memorie fizică de 256 GiB, unitatea adusabilită fiind byte-ul. 1 cadrant (word) are 64 de biti, un block are 16 cadrante și o pagină are dimensiunea de 16 KiB. TLB-ul acestui computer are 64 intrări și folosește maparea directă. Dimensiunea de date a memoriei cache este 256 KiB, cache-ul implementând o politică de tip Write Back pentru scriere și Write Allocate în cazul unei write miss. Cache-ul este set-associativ pe 4 căi, și utilizează politica de retrasare LRU cu registru de vârstă și cer următoarele:

- (4) a) Formulele care relatează adresa virtuală, respectiv fizică
 (4) b) Dimensiunile totale ale TLB-ului și cache-ului
 (4) c) Schema bloc comună a TLB-ului și cache-ului

$$4 \text{ TiB} = 2^2 \cdot 2^{40} \text{ B} = 2^{42} \text{ B}$$

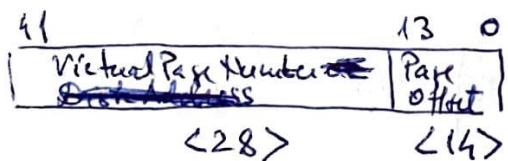
$$\begin{array}{r} 38- \\ 16 \\ \hline 22 \end{array}$$

$$256 \text{ GiB} = 2^8 \cdot 2^{30} = 2^{38} \text{ B}$$

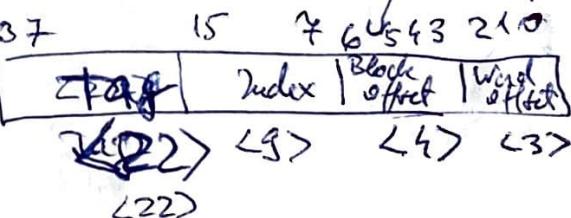
$$16 \text{ KiB} = 2^4 \cdot 2^{10} \text{ B} = 2^{14} \text{ B}$$

$$\begin{array}{r} 42- \\ 14 \\ \hline 28 \end{array} \quad \begin{array}{r} 38- \\ 14 \\ \hline 24 \end{array}$$

Virtual Address



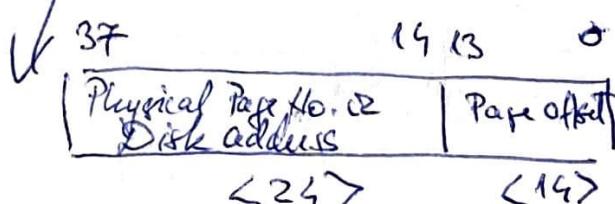
Physical Address



$$1 \text{ cache block} = 2 \text{ words} = 2^4 \cdot 2^3 \text{ B} = 2^7 \text{ B}$$

$$256 \text{ KiB Data cache} = 2^{18} \text{ B} \Rightarrow \frac{2^{18} \text{ B}}{2^2 \text{ s.a.}} = 2^{16} \text{ B / bank}$$

$$\frac{2^{16} \text{ B / bank}}{2^7 \text{ B / block}} = 2^9 \text{ blocks / bank}$$



b) Cache total size = $2^2 \times 2^9$ (1 bit + 1 bit + 22 bits + 2 bits + 2⁷ B)
 s.a. block \downarrow dirty tag age data
 LRU

$$= \underbrace{2^{11} \times 2^7 B}_{\text{data}} + 2^{11} \times 26 \text{ bits} \approx 256 \text{ KiB} + 2^{11} \times 32 \text{ bits}$$

$$\approx 256 \text{ KiB} + 2^2 \cdot 4 \text{ B} \cdot 1 \text{ K}$$

TLB entries = 64 = $2^6 \Rightarrow$ index size = 6
 tag = $28 - 6 = 22$ bits

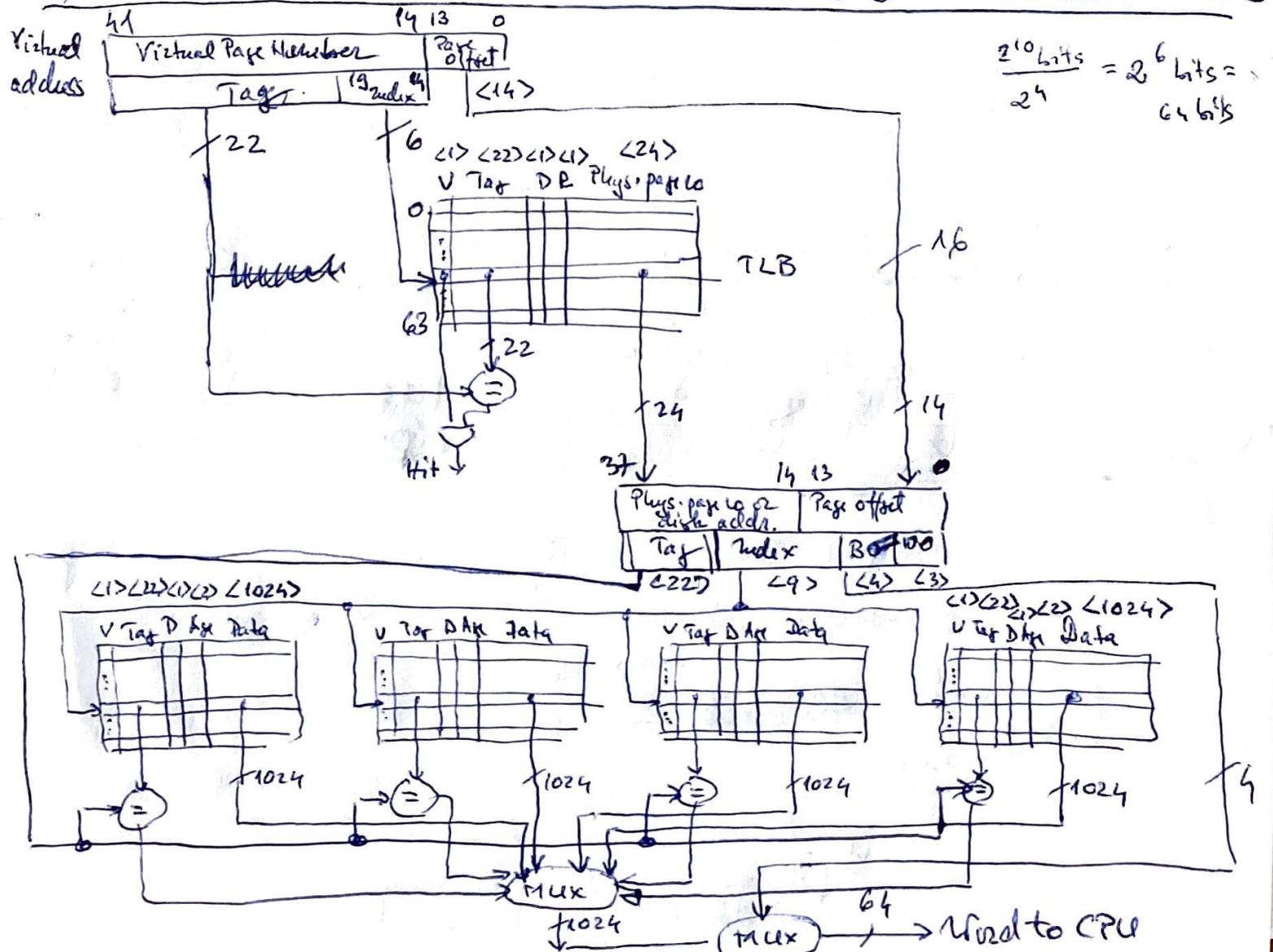
$$= 256 \text{ KiB} + 16 \text{ KiB}$$

$$\approx 272 \text{ KiB}$$

TLB total size = ~~28-6-22~~ $2^6 \times (1 \text{ bit valid} + 1 \text{ bit dirty} + 1 \text{ bit acc} + 22 \text{ bits tag} + 24 \text{ bits physical page no.})$

$$= 2^6 \times (49 \text{ bits}) \approx 2^6 \times 64 \text{ bits} = 2^6 \times 2^3 B = 2^9 B \approx 0.5 \text{ KiB}$$

c)



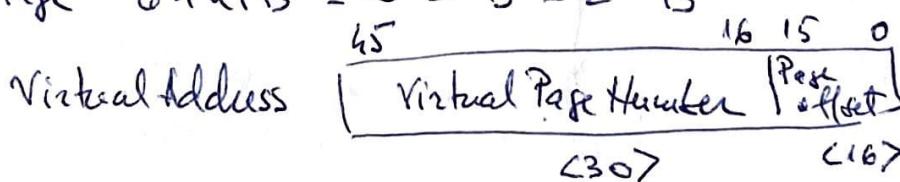
Q1 Considerăm un computer ce are un spațiu de memorie virtual de 64 TiB și un spațiu de memorie fizică de 128 GiB, unitatea adresabilă fiind byte-ul. Un word (word) are 64 de biti, un block are 8 cuvinte și o pagină are dimensiunea de 64 KiB. TLR-ul acestui computer are 32 de încărăci și folosește o magazin full asociativă.

Dimensiunea de date a memoriei cache este 512 KiB, cache-ul implementând o politică de tip Write Through pentru scris și Write Allocate în cazul unei write miss. Cache-ul este set-associativ pe 2 căi și utilizează politică randomă de replasare.

Se cer următoarele:

- 1P a) Formulele curierelor de adrese virtuale, respectiv fizice.
- 1P b) Dimensiunile totale ale TLR-ului și cache-ului.
- 1P c) Schema bloc comună a TLR-ului și cache-ului.

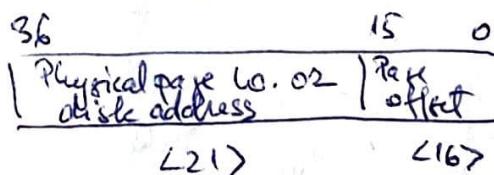
$$\begin{aligned} \text{a) VA } 64\text{TiB} &= 2^6 \cdot 2^{40}\text{B} = 2^{46}\text{B} \\ \text{PA } 128\text{GiB} &= 2^7 \cdot 2^{30}\text{B} = 2^{37}\text{B} \\ \text{Page } 64\text{KiB} &= 2^6 \cdot 2^{10}\text{B} = 2^{16}\text{B} \end{aligned}$$



$$\frac{46}{30} - \frac{16}{30} \quad \frac{37}{21}$$

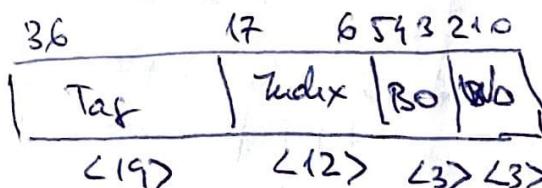
$$\begin{aligned} 1 \text{ cache block} &= 2^3 \text{ words} \\ 1 \text{ word} &= 2^3 \text{ B} \end{aligned}$$

Physical address



30 - block offset
16 - word offset

for cache



$$\frac{37}{19}$$

$$1 \text{ cache block} = 2^3 \text{ words} = 2^3 \cdot 2^3 \text{ B} = 2^6 \text{ B / block}$$

$$512 \text{ KiB data cache} = 2^9 \cdot 2^{10} \text{ B} = 2^{19} \text{ B} \Rightarrow \frac{2^{19} \text{ B}}{2^6 \text{ B / block}} = 2^{13} \text{ B / bank}$$

$$\frac{2^{18} \text{ B / bank}}{2^6 \text{ B / block}} = 2^{12} \xrightarrow{\text{Index size}} \text{blocks / bank}$$

b) Cache total size = $2^5 \times 2^{12}$ (1 bit + 19 bits + $2^6 B$) =
 s.a. blocks/ banks tag data

$$= 2^{19} B \text{ data} + 2^{12} \times 20 \text{ bits} \approx 512 \text{ KiB} + 3 \cdot 2^{12} \text{ B}$$

$$= (512 + 12) \text{ KiB} = 524 \text{ KiB}$$

TLB total size = $2^5 \times (1 \text{ bit valid} + 1 \text{ bit dirty} + 1 \text{ bit acc} +$
 $+ 30 \text{ bits tag} + 21 \text{ bits phys. addr}) \approx 2^5 \times 64 \text{ bits} =$
 $= 2^5 \times 2^3 \text{ B} = 2^8 \text{ B} = 0.25 \text{ KiB}$

