3.1.1. Direct mapping

Fast MATH → Intensity

Address Word → 32 bits
Data Word → 32 bits – 4 B

split Cache → data 16 KiB
→ instr. 16 KiB

1 block = 16 words = 2^4 words = $2^4 \cdot 2^2$ B = 2^6 B → Block Offset
(byte addressing)
2^9 bits = 512 bits

total memory 2^{32} B = $2^{30} \cdot 2^2$ B = 4 GiB

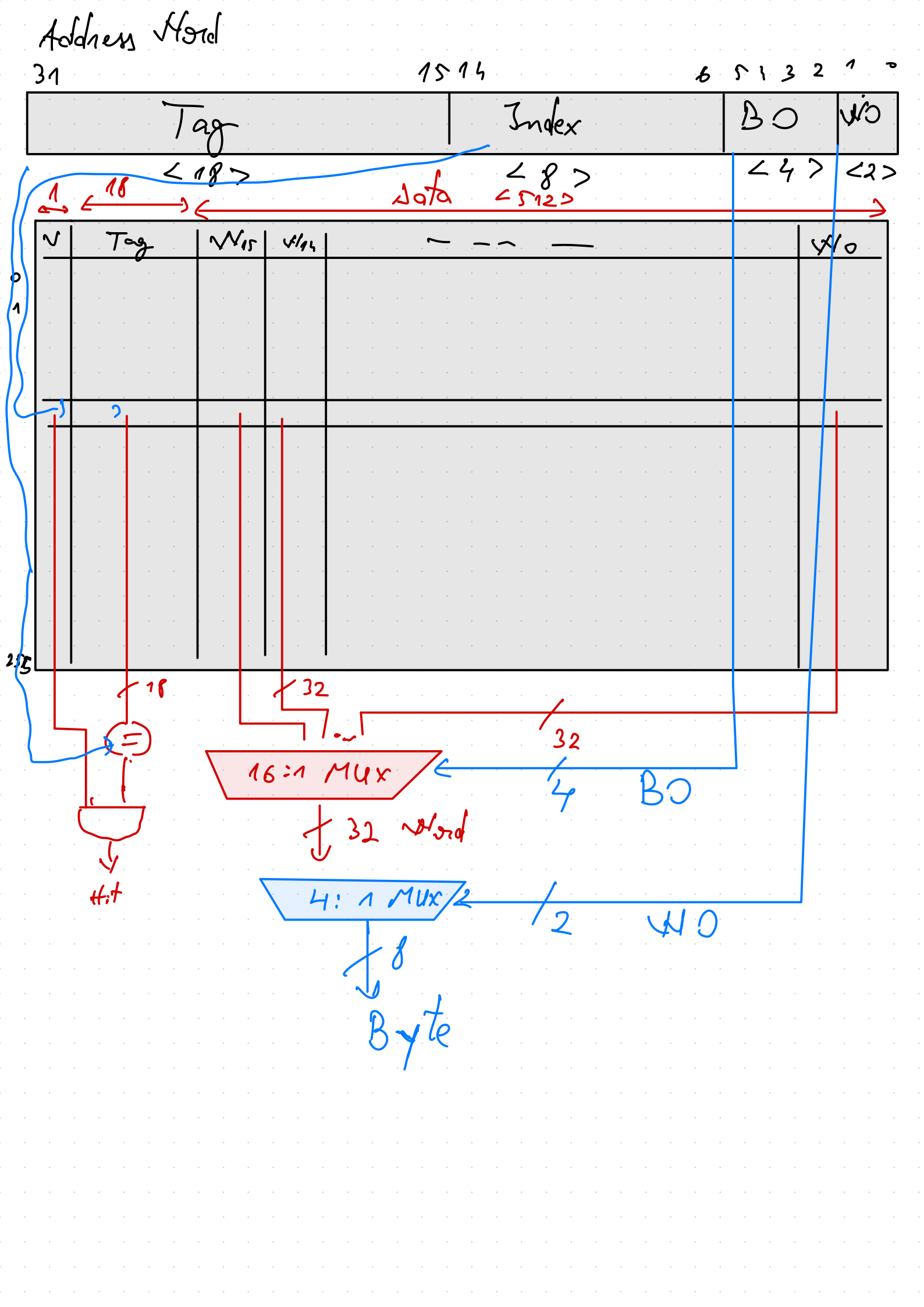
Cache Size = 16 KiB = $2^4 \cdot 2^{10}$ B = 2^{14} B

= $2^{14} / 2^6$ blocks = 2^8 blocks = 256 blocks

index = 8

Main Memory: 4 GiB = $\frac{2^{30} \cdot 4\text{ B}}{2^6\text{ B}}$ blocks = 2^{26} blocks

tag size = m − m = 26 − 8 = 18

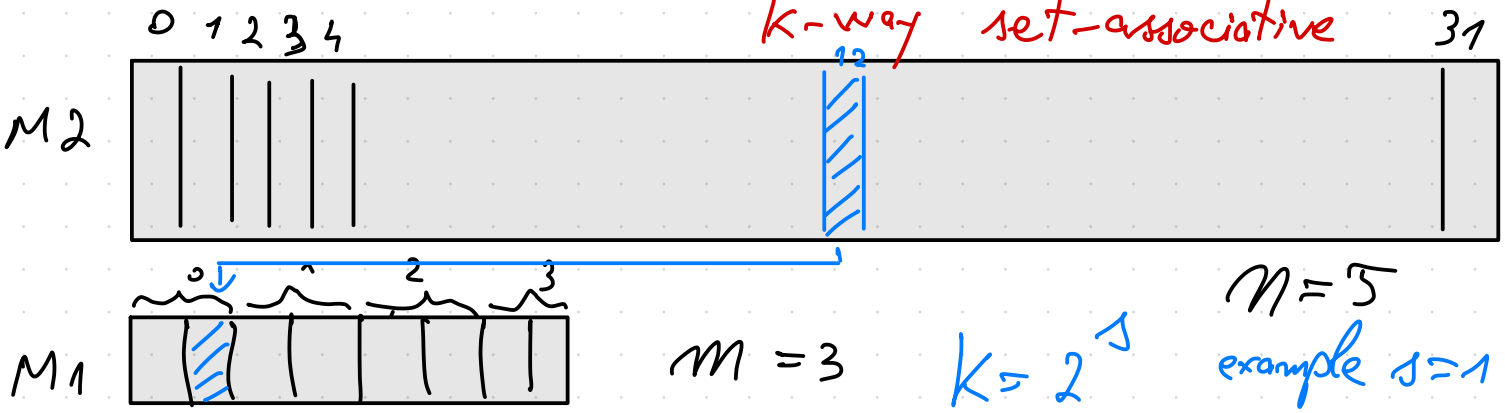

Address Word
31
15 14
6 5 4 3 2 1 0
Tag
Index
BO
WO
<18>
<8>
<4>
<2>
1
18
Data
<512>
V
Tag
W15
W14
W0
0
1
255
18
32
=
16:1 MUX
32
4
BO
32 Word
Hit
4:1 MUX
2
WO
8
Byte

Cache Performance Benchmarks:

Mem Access < Hit / Miss

Hit Rate = 1 - Miss Rate

3.1.2 Set-associative mapping

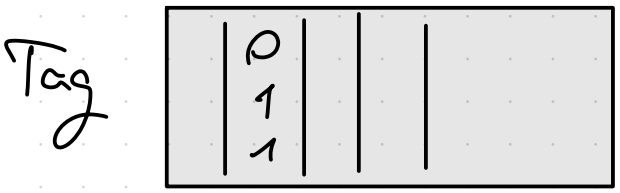


$i = j \bmod 2^{m-s}$

$12 \bmod 2^2 = 0$

Block Index 0 11 00

Tag



"Câte linii băgați acolo, profu?"

Mihai "Uncancellable" Udrescu

$MM = 2^{24}$ words

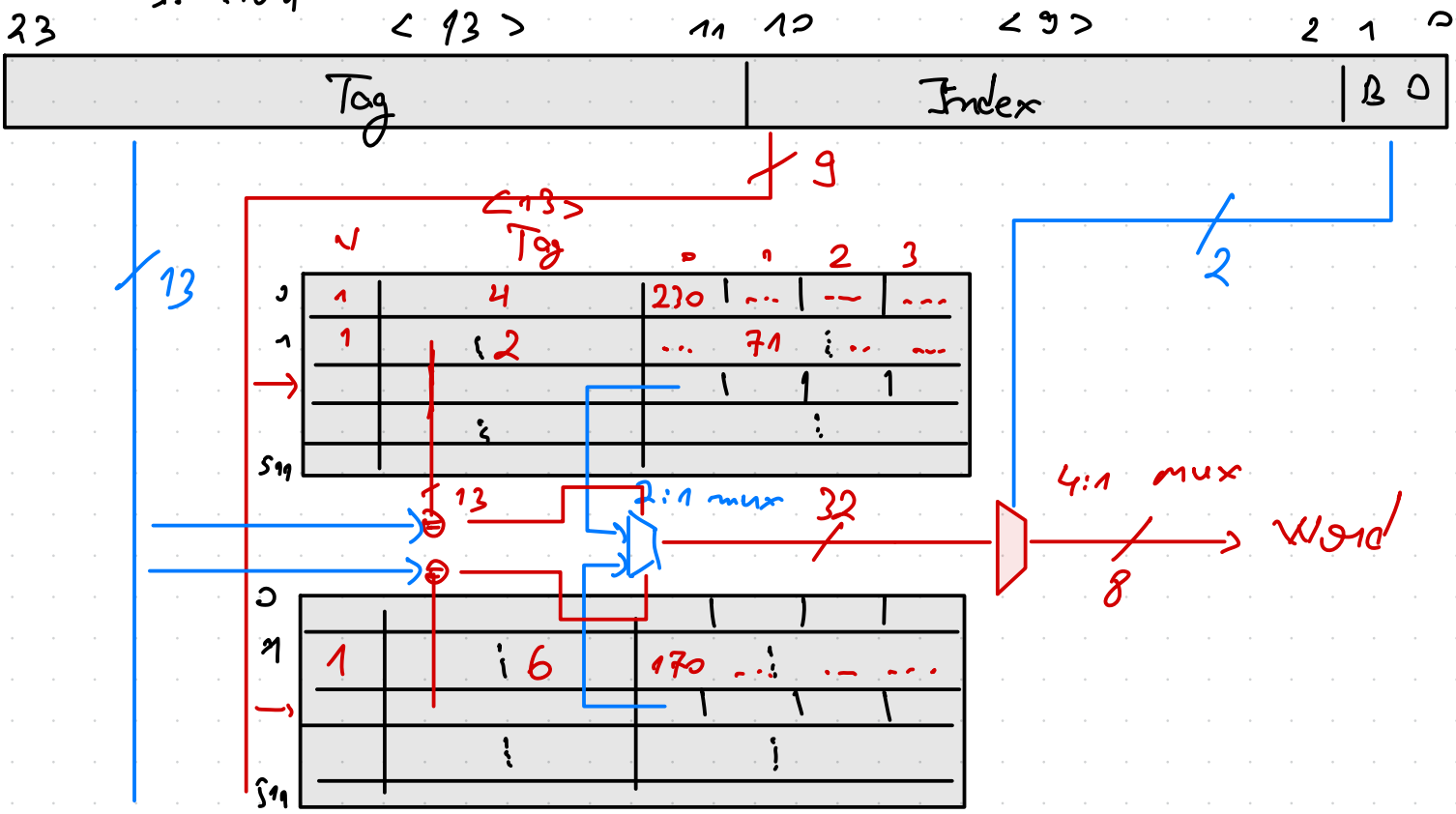
1 word = 1 byte

1 block = 4 words

Size M_1 = 1k: Block

Address	Code
8192	230
4102	71
12292	170

Address Word



$4102 = 2^{12} + 2^2 + 2^1$

$8192 = 2^{13}$

$12292 = 2^{13} + 2^{12} + 2^2$

	Tag	Index	B_0
8192	0 _ _ _ _	0100	0000000000	00
4102	_ _ _ _ _ _	0010	0000000001	10
12292	0 _ _ _ _	0110	0000000001	00

"Nu există fairness pe lumea asta. Wake up!"

M. Udrescu

VAX 11/780

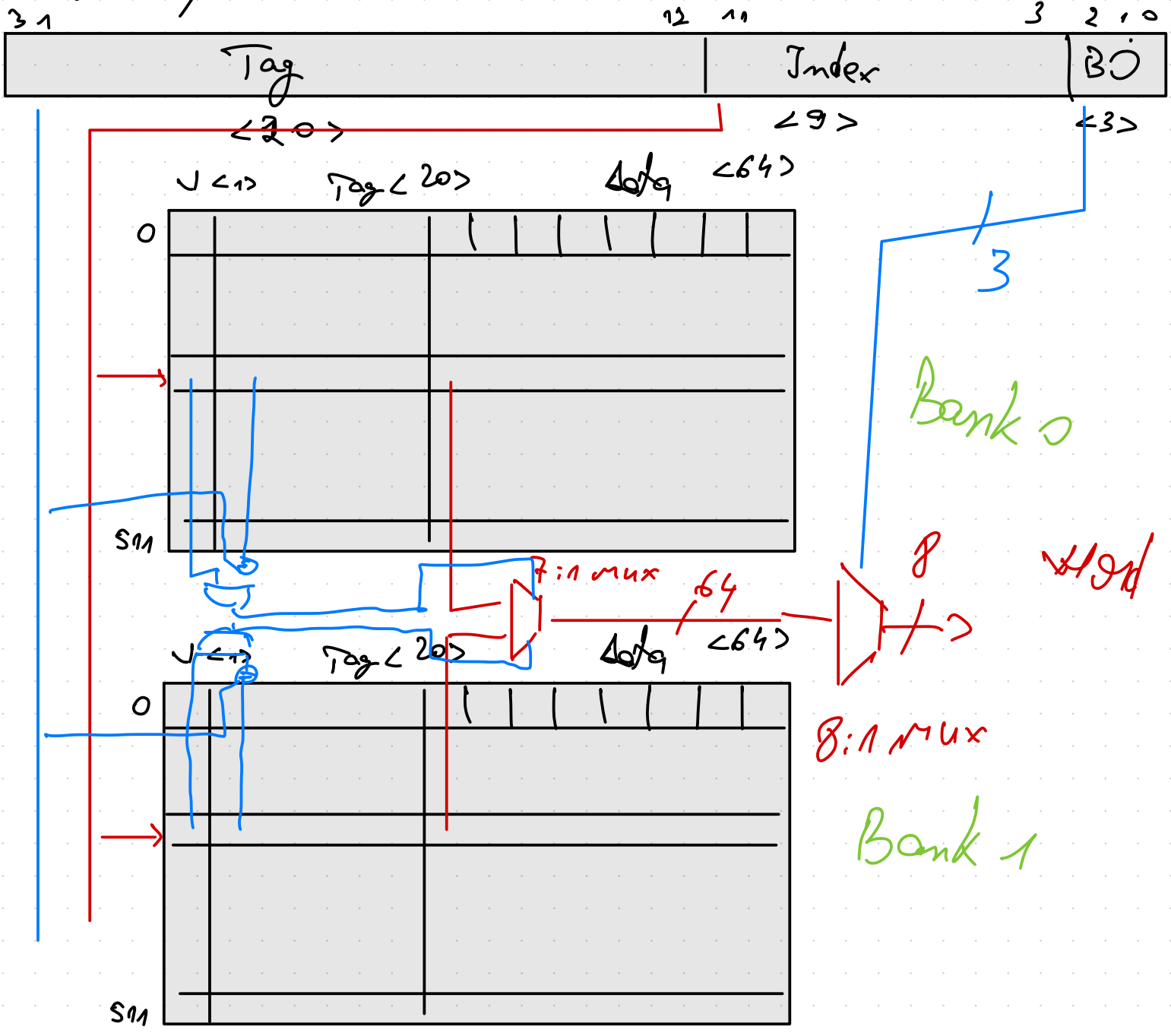
Size (MM) = 4 GiB

1 word = 1 B

1 block: 8 words = 2^3 B = 64 bits

Cache data size = 1 KiBlock: $\frac{2^{10}}{2} = 2^9$ sets → index

2-way set associative cache



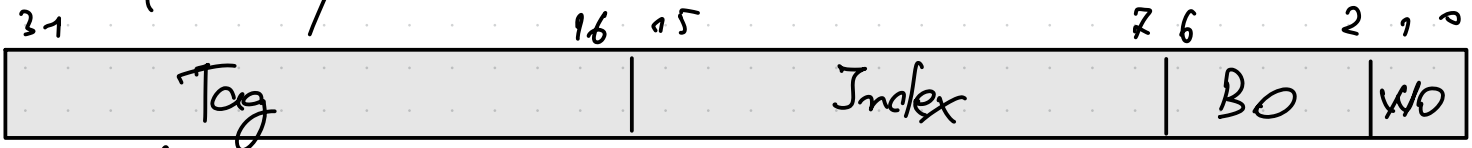
Size MM $= 4\,GiB = 2^{30} \cdot 2^2\,B = 2^{32}\,B$

1 word $= 32$ bits $= 4\,B$ (2^2) → ×10

1 block $= 32$ words $= 2^5 \cdot 2^3\,B = 2^8\,B$

Size Cache $= 256\,kiB = 2^{18}\,B$ → BO

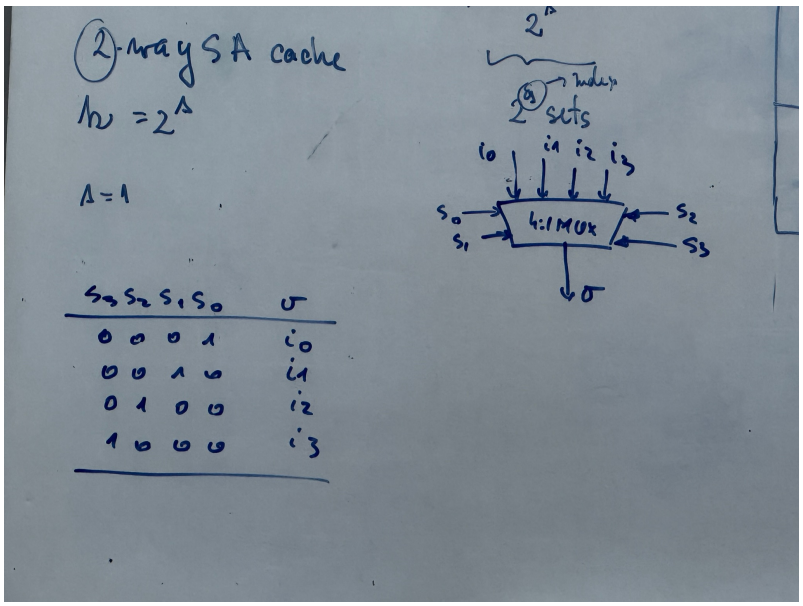
4-way SA cache $k = 4$ $k = 2^2 \rightarrow s = 2$

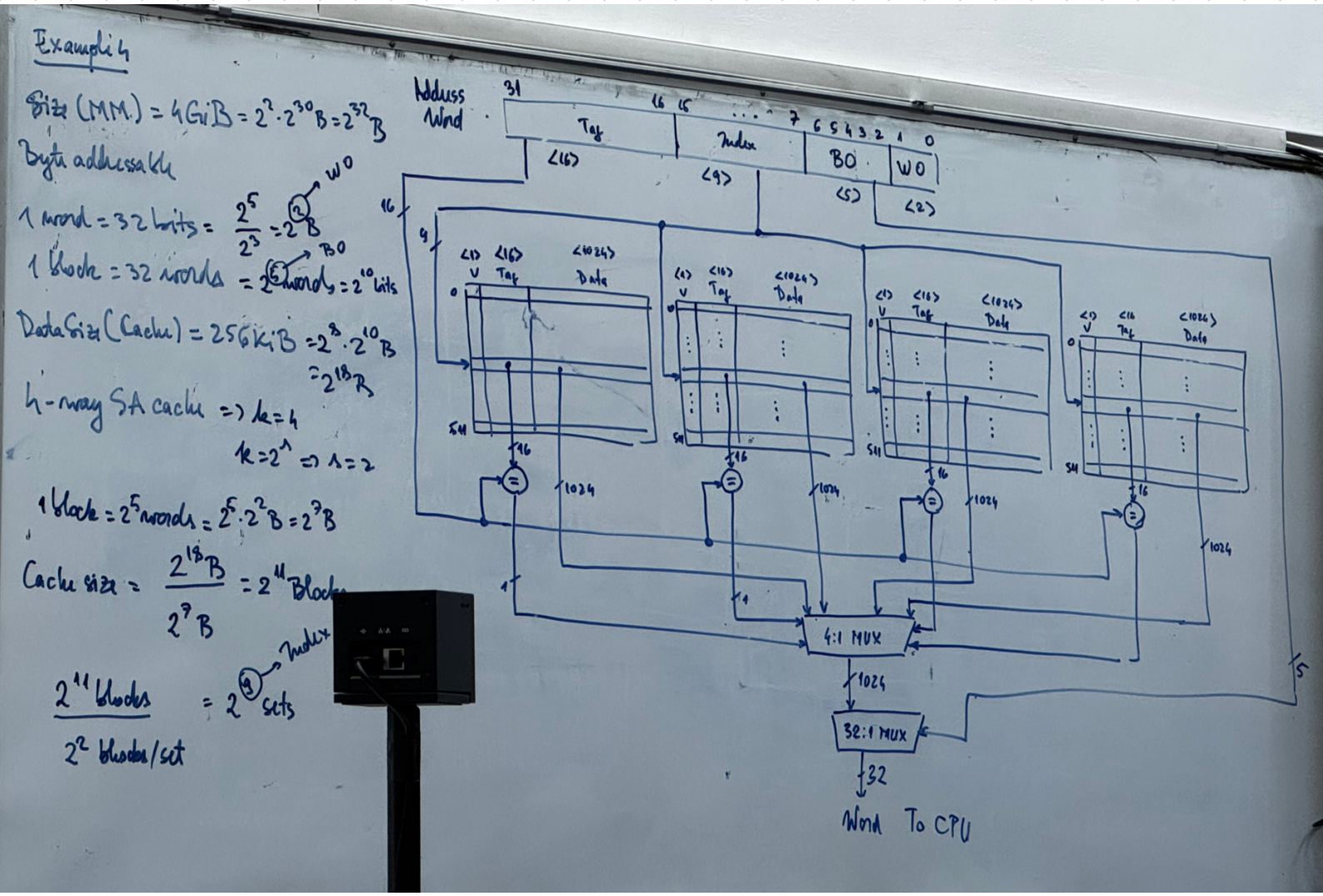


1 block $= 2^5$ words $= 2^7\,B$

Cache Size $= \frac{2^{18}\,B}{2^7\,B} = 2^{11}$ blocks

$\frac{2^{11}\text{ blocks}}{2^2\text{ blocks/set}} = 2^9$ sets ← Index





"N-aveți nicium drept!" M. Udrescu

4-way SA cache

8 blocks / bank 2^3

4 words / block

2^2

MM → 2^{16} words 2^{14} blocks

Word addressing

Initial cache empty

→ Write back

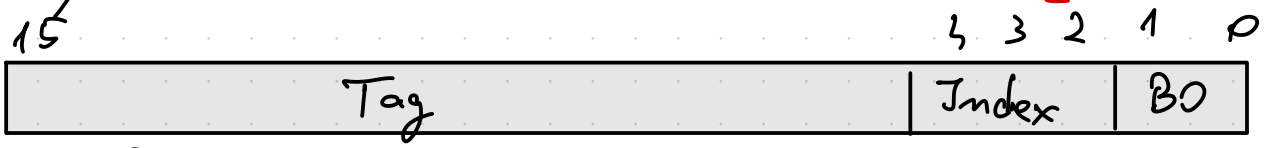
→ LRU replacement

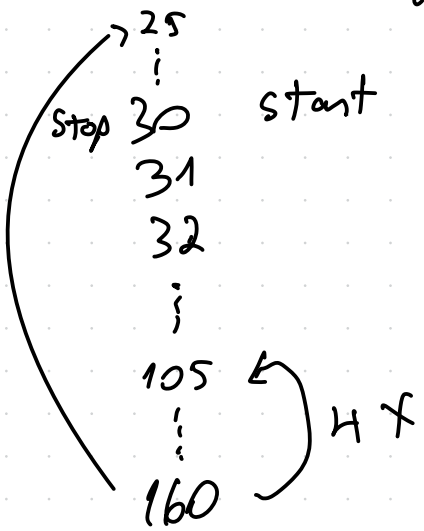
∴ Cache access time => t_{ac} = 2ns

Miss Penalty = t_{mp} = 20ns

a) address format

b) Hit Ratio + Average Memory Access Time





Example 5

4-way SA

8 blocks / bank = 2^3

4 words / block = 2^2

MM → 2^{16} words

Word addresses

Initially empty

Write back

LRU replacement

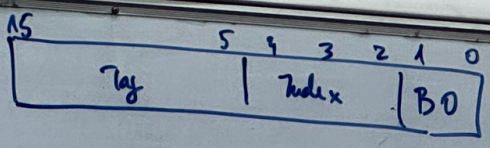
Cache access time = t_{ac} = 2 ns

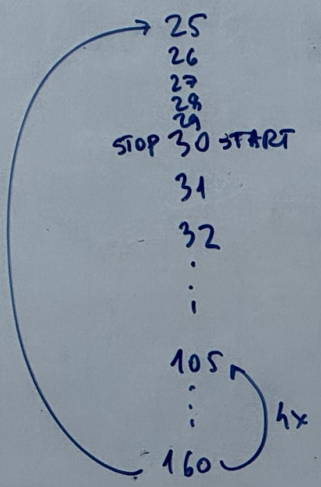
Miss Penalty = 20 ns

a) address format

b) Hit Ratio + AMAT

Average Memory Access Time.





MM