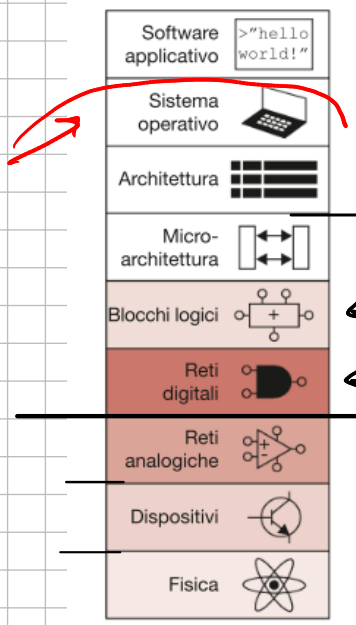
ASTRAZIONE



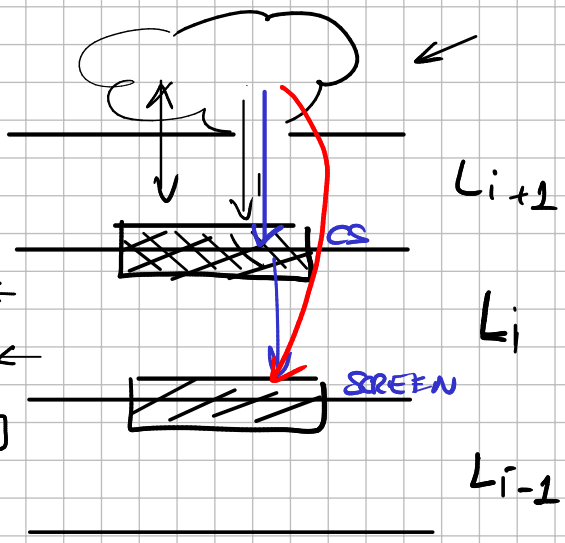
LINUX

x86 64b

i5 - 8ª 1,6GHz

R M

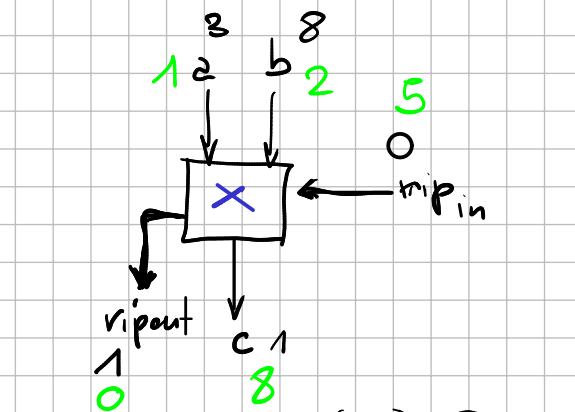
D

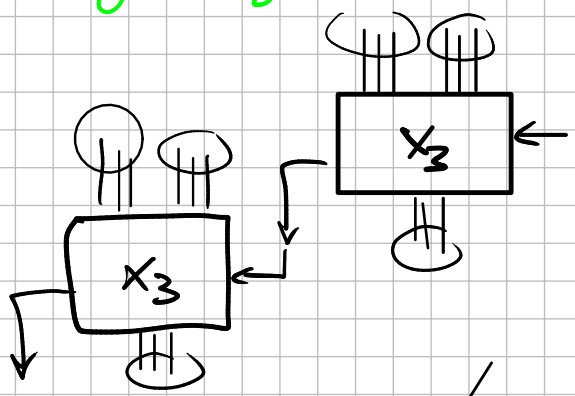


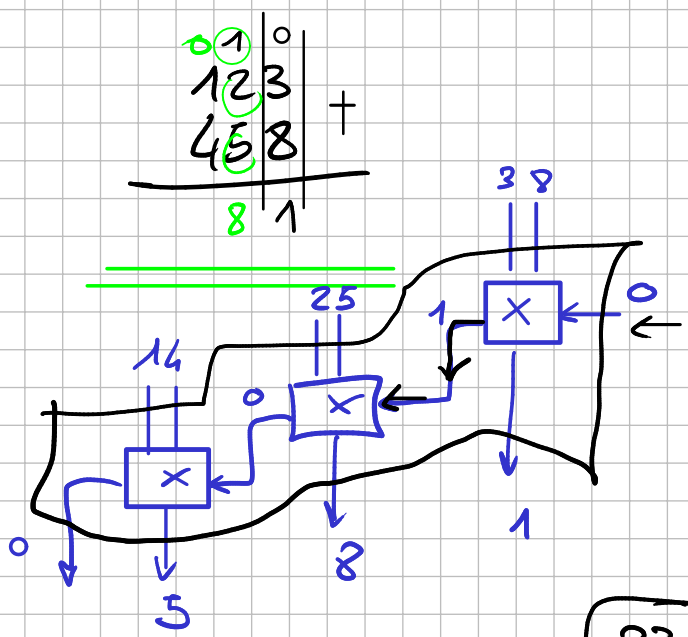
Gerarchia

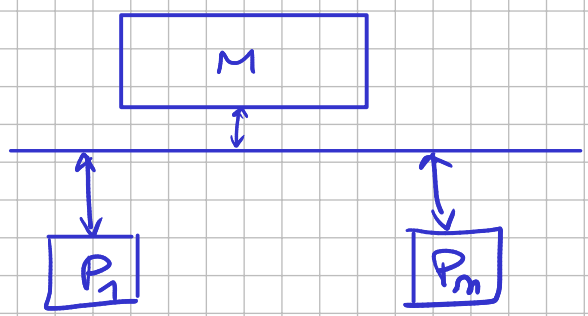
Modularità

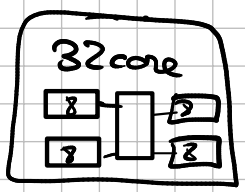
Regolarità

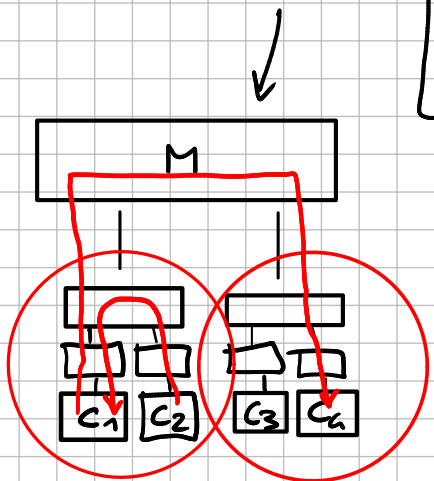












NUMA

DECIMALE POSIZIONALE

base = 10

alfabeto 0 1 2 --- 9

123_{10}

↑

213

↑

pos 2 1 0

pesi $10^2\ 10^1\ 10^0$

BINARIO

base = 2

alfabeto 0 1

F V

0V 5V

bit bit bit

101_2 → $?_{10}$ $1\times2^2 + 0\times2^1 + 1\times2^0$ = 4 + 1 → 5_{10}

pos 2 1 0

pesi $2^2\ 2^1\ 2^0$

$5_{10} \rightarrow xxx_2$

101

5 | 2
1 | 2 | 2
0 | 1

5_{10} + 3_{10} = 8_{10}

101_2 + 011_2 = 1000 → 8_{10} (riporti: 1 1)

7 + 5 = 2 (riporto 1)

15 − 7

5_{10} − 3_{10} = 2_{10}

101 − 011 = 010

Ottale

b = 8

0 ÷ 7

1 cifra ottale

3 bit

Hex

b = 16

0 ÷ 9 A ÷ F

1 cifra hex

4 bit

0	000
1	001
2	010
3	011
4	100
5	101
6	110
7	111

1111 0011 0010 0001 → F321 xxxx

1110 0011 0010 0001 → E321

Relativi +x −y 0

Modulo e segno

+0 / −1 — bit segno

4 bit

+0	+7
−0	−7

1000
0000

+3 − +2

+3 + +2

2 circuiti differenti × fare { + / −

Complemento a 2

a) interi positivi → rappresentazione "naturale"

+3 ⟶ 00…0101

b) interi negativi → 1) n_2 (n = −n)

2) inverte tutti i bit

3) +1

−3 0000 0011

1111 1100
0000 0001

1111 1101

n bit 2^n

−x −(x−1) … 0 1 2 … y

2^{n-1} 2^{n-1}

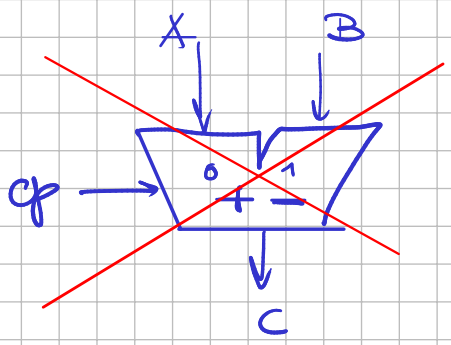
$-(2^{n-1}) \dots (2^{n-1})-1$

−3 1111 1101 +
0000 0101

1 0000 0010

2

127 + 1




```
  1
  1 1 1 1 1 1 1
  0 1 1 1 1 1 1 1
  7 6 5 4 3 2 1 0
  0 0 0 0 0 0 0 1
-------------------
 (1)0 0 0 0 0 0 0 0
```

A - B