

ALU (unità aritmetiche logiche)

Definizione: l'ALU (unità aritmetico-logica) è la parte del processore che svolge le operazioni aritmetico-logiche. È un insieme di circuiti combinatori.

Blocchi di base per costruire l'ALU

AND gate 

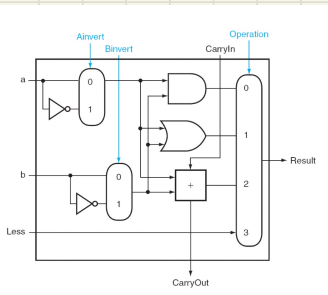
OR gate 

Inverter 

Multiplexor 

ALU a 32 bit

ALU 0 - ALU 30



Tutte le possibili combinazioni di Ainvert, Binvert e Operation, generano un output diverso

AND → Ainvert 0, Binvert 0, Operation 0

OR → Ainvert 0, Binvert 0, Operation 1

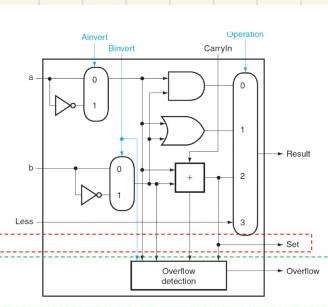
NAND → NOT(a AND b) = NOT a OR NOT b Ainvert 1, Binvert 1, Operation 1

NOR → NOT(a OR b) = NOT a AND NOT b Ainvert 1, Binvert 1, Operation 0

Somma → Ainvert 0, Binvert 0, Operation 2 CarryIn (01) → CarryOut (01)

Sottr → Ainvert 0, Binvert 1, Operation 2 CarryIn (01) → CarryOut (01)

ALU 31



Il Set ritorna il suo valore entrando

nella ALU 0 come Less, il multiplexor Operator

sarà 3 e ritorna il risultato di a < b.

SLT (Set on Less Than) → $a < b$

$(a-b)$ è negativo

bit 31 di $(a-b)$ è 1

→ ALU0 - ALU31 fa sottrazione ($\text{Less} = 0$) ⇒ Set ritorna a ALU0
e controlla bit di segno e ritorna se $a < b$ in Op. = 3

OSS: attenzione all'overflow → viene fatto un controllo nel bit 31

bit-31(A) = 0(+), bit-31(B) = 1(-), bit-31(A-B) = 1(-)

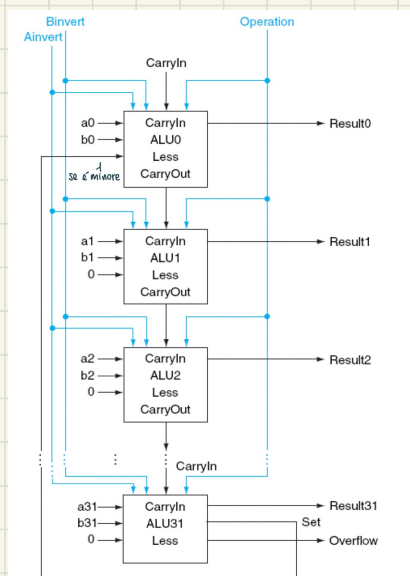
bit-31(A) = 1(-), bit-31(B) = 0(+), bit-31(A-B) = 0(+)

BEQ (Branch-on-equal) → $a = b$

$a - b = 0$

→ ALU0 - ALU31 fa sottrazione ⇒ tutte le operazioni bit a bit entrano in un NOR. Se sono tutti 0, allora NOR = 1
⇒ $a = b$

ripple carry ~ una organizzazione delle ALU



Definizione: Connessione a 1-bit ALU adiacenti.

Questa organizzazione chiamata **ripple carry** permette di realizzare una ALU a 32 bit.

→ se è minore, lo sappiamo solo qui perché deve fare la sottrazione di tutti i bit! per il resto il Less è sempre 0. (SLT)