



Sommatori e Moltiplicatori

Prof. Alberto Borghese
Dipartimento di Scienze dell'Informazione

borgnese@dsi.unimi.it

Università degli Studi di Milano

Riferimenti: B.5 sul Patterson, per i moltiplicatori HW, capitolo 9 fino a 9.3.3.2 sul Fummi,
Per gli algoritmi di moltiplicazione: capitolo 3 (fino a 3.4) del Patterson.



Sommario

Sommatori

Moltiplicatori

Algoritmi per la moltiplicazione



(Half) Adder ad 1 bit

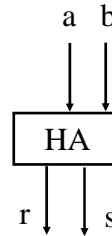
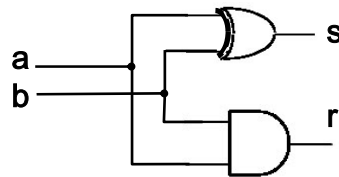


Tabella della verità della somma:

| a | b | somma | riporto |
|---|---|-------|---------|
| 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 |

$$s = a \oplus b$$

$$r = ab$$



La somma è diventata un'operazione logica!

Cammini critici:

Somma = 1;

Riporto = 1;

Complessità

Somma = 1 porta;

Riporto = 1 porta;



Full Adder ad 1 bit



Tabella della verità della somma completa:

| a | b | r_{in} | somma | riporto |
|---|---|----------|-------|---------|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 1 | 0 | 0 | 1 | 0 |
| 1 | 1 | 0 | 0 | 1 |
| 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 1 | 0 | 1 |
| 1 | 0 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 |

$$s = m_1 + m_2 + m_4 + m_7$$

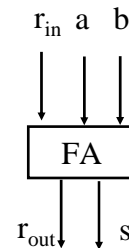
$$r = m_3 + m_5 + m_6 + m_7$$

$$s = \overline{a} \overline{b} r_{in} + \overline{a} b \overline{r_{in}} + a \overline{b} \overline{r_{in}} + a b r_{in} =$$

$$= (a \oplus b) \overline{r_{in}} + (\overline{a} b + a \overline{b}) r_{in} =$$

$$= (a \oplus b) \overline{r_{in}} + (a \oplus b) r_{in}$$

$$r_{out} = \overline{a} \overline{b} r_{in} + \overline{a} b r_{in} + a \overline{b} r_{in} + a b r_{in} = ab + (a \oplus b) r_{in}$$



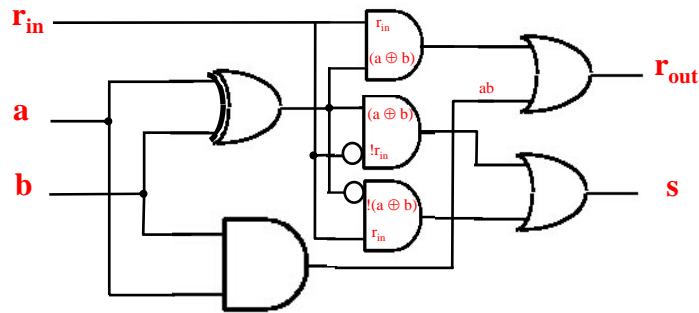


Implementazione circuitale



$$s = (a \oplus b) \bar{r}_{in} + (a \oplus b) r_{in}$$

$$r_{out} = ab + (a \oplus b) r_{in}$$



7 porte logiche.
Cammini critici: $s \rightarrow 3$; $r_{out} \rightarrow 3$



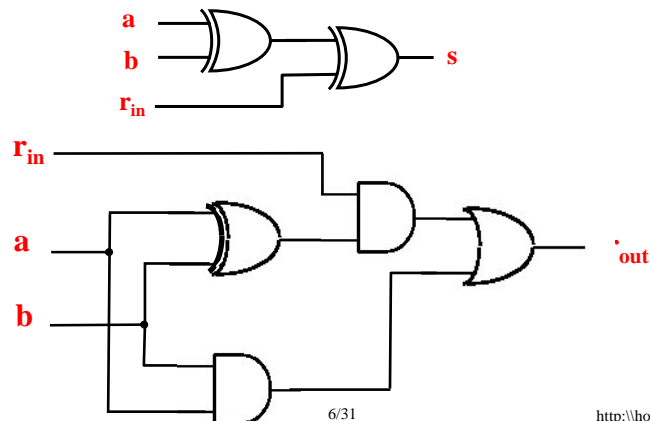
Semplificazione circuitale



$$s = (a \oplus b) \bar{r}_{in} + \overline{(a \oplus b)} r_{in} = (a \oplus b) \oplus r_{in}$$

$$r_{out} = ab + (a \oplus b) r_{in}$$

5 porte logiche.
Cammini critici: $s \rightarrow 2$; $r_{out} \rightarrow 3$

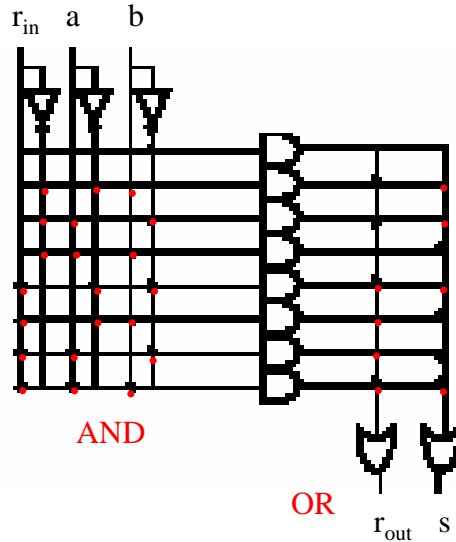




Implementazione mediante PLA



| a | b | r _{in} | somma | r _{out} |
|---|---|-----------------|-------|------------------|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 1 | 0 | 0 | 1 | 0 |
| 1 | 1 | 0 | 0 | 1 |
| 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 1 | 0 | 1 |
| 1 | 0 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 |



SOP: costruisco i mintermini e li sommo.



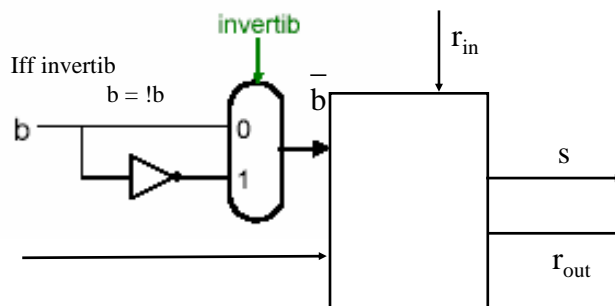
Sottrazione



In complemento a 2 diventa un'addizione: $a - b = a + !b + 1$

Serve:

- un inverter (NOT).
- la costante 1 da collegare al riporto in ingresso (occorre utilizzare un full adder anche per il bit meno significativo con $r_{in0} = 1$).





Esercizi



Scrivere il circuito che esegue la somma di: $3 + 4$ in base 2.
Riportare tutte le uscite delle porte logiche.

Scrivere il circuito che esegue la seguente sottrazione: $5 - 2$ in base 2.
Riportare tutte le uscite delle porte logiche.



Sommario



Addizionatori

Moltiplicatori

Algoritmi per la moltiplicazione



Moltiplicazione binaria



Moltiplicando \longrightarrow 1 1 0 1 1 x
 Moltiplicatore \longrightarrow 1 1 1 =

| | |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------|
| $ \begin{array}{r} 11011 \times 27_{10} \\ 111 = 7_{10} \\ \hline 111111 \\ 11011+ \\ 11011- \\ 11011- - \\ \hline 10111101 \quad 189_{10} \end{array} $ | $ \begin{array}{r} \hline 11111 \\ 11011+ \\ 11011- \\ \hline 1 \\ 1010001+ \\ 11011- - \\ \hline \end{array} $ |
| <p>Prodotto \longrightarrow 1 0 1 1 1 1 0 1</p> | |



La moltiplicazione binaria



Possiamo vederla come:

Un primo stadio in cui si mette in AND ciascun bit del moltiplicatore con il moltiplicando.

Un secondo stadio in cui si effettuano le somme (full adder) dei bit sulle righe contenenti i prodotti parziali.



Moltiplicazione binaria



Moltiplicando \longrightarrow 1 1 0 1 1 x 27_x
 Moltiplicatore \longrightarrow 1 0 1 1 = 11 =

Prodotti parziali \longrightarrow 1 1 1 1 1
 1 1 0 1 1 + 27 +
 1 1 0 1 1 - 27 - =

Riporto \longrightarrow 0 0 0 0 0 297
 1 0 1 0 0 0 1 +
 0 0 0 0 0 - -

Somma parziale \longrightarrow 1 1 0 1 0
 1 0 1 0 0 0 1 +
 1 1 0 1 1 - - - =

Prodotto \longrightarrow 1 0 0 1 0 1 0 0 1 $\rightarrow 297_{10}$



Moltiplicazione binaria (su 4 bit)



Moltiplicando \longrightarrow 1 0 1 1 x 11₁₀
 Moltiplicatore \longrightarrow 1 0 1 = 5₁₀

Prodotti parziali \longrightarrow 0 0 1 0 0
 1 0 1 1 + 11 +
 0 0 0 0 - 0 +
 1 0 1 1 - - 44 +

Prodotto \longrightarrow 0 1 1 0 1 1 1 55₁₀

Il prodotto parziale è = $\begin{cases} \text{Moltiplicando incolonnato opportunamente} \\ 0 \end{cases}$



La matrice dei prodotti parziali



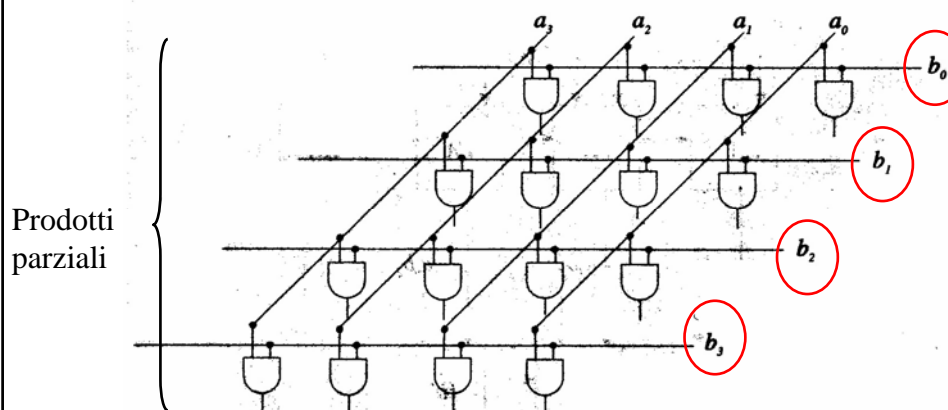
Prodotti parziali

| | | a_3 | a_2 | a_1 | a_0 | |
|--|-----------|-----------|-----------|-----------|-----------|-------|
| | | $a_3 b_0$ | $a_2 b_0$ | $a_1 b_0$ | $a_0 b_0$ | b_0 |
| | | $a_3 b_1$ | $a_2 b_1$ | $a_1 b_1$ | $a_0 b_1$ | b_1 |
| | $a_3 b_2$ | $a_2 b_2$ | $a_1 b_2$ | $a_0 b_2$ | | b_2 |
| | $a_3 b_3$ | $a_2 b_3$ | $a_1 b_3$ | $a_0 b_3$ | | b_3 |

In binario i prodotti parziali sono degli AND.



Il circuito che effettua i prodotti





La matrice dei prodotti parziali



| | | | | | | | |
|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-------|
| | | | a_3 | a_2 | a_1 | a_0 | |
| | | | $a_3 b_0$ | $a_2 b_0$ | $a_1 b_0$ | $a_0 b_0$ | b_0 |
| | | $a_3 b_1$ | $a_2 b_1$ | $a_1 b_1$ | $a_0 b_1$ | | b_1 |
| | $a_3 b_2$ | $a_2 b_2$ | $a_1 b_2$ | $a_0 b_2$ | | | b_2 |
| $a_3 b_3$ | $a_2 b_3$ | $a_1 b_3$ | $a_0 b_3$ | | | | b_3 |

In binario i prodotti parziali sono degli AND.



Somma delle prime 2 righe dei prodotti parziali



| | | | | | | | |
|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-------|
| | | | a_3 | a_2 | a_1 | a_0 | |
| | | | $a_3 b_0$ | $a_2 b_0$ | $a_1 b_0$ | $a_0 b_0$ | b_0 |
| | | $a_3 b_1$ | $a_2 b_1$ | $a_1 b_1$ | $a_0 b_1$ | | b_1 |
| | $a_3 b_2$ | $a_2 b_2$ | $a_1 b_2$ | $a_0 b_2$ | | | b_2 |
| $a_3 b_3$ | $a_2 b_3$ | $a_1 b_3$ | $a_0 b_3$ | | | | b_3 |

$a_3 b_1$

$a_2 b_1$

$a_3 b_0$

$a_2 b_0$

$a_1 b_1$

$a_0 b_1$

$a_0 b_0$

| | |
|-------------------|----------------------|
| 1 1 0 1 1 x | 27 x |
| 1 0 1 1 = | 11 = |
| ----- | |
| 1 1 1 1 1 | |
| 1 1 0 1 1 + | 27 + |
| 1 1 0 1 1 - | 27 - = |
| ----- | |
| 0 0 0 0 0 | 297 |
| 1 0 1 0 0 0 1 + | |
| 0 0 0 0 - - | |
| ----- | |
| 1 1 0 1 0 | |
| 1 0 1 0 0 0 1 + | |
| 1 1 0 1 1 - - - = | |
| ----- | |
| 1 0 0 1 0 1 0 0 1 | -> 297 ₁₀ |

Somma dei primi 2 prodotti parziali:
Aggiunge il terzo prodotto parziale:

HA e FA non sono equivalenti
per i diversi cammini critici.

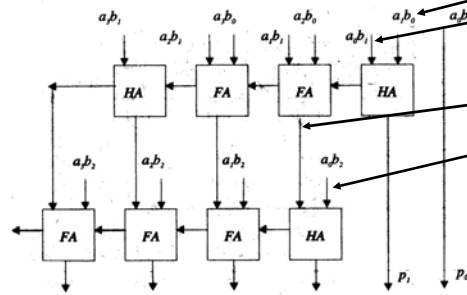


Somma della terza riga



I primi due prodotti parziali sono sommati dalla prima batteria di sommatori.

Ogni altro prodotto parziale è sommato da un'ulteriore batteria di sommatori.



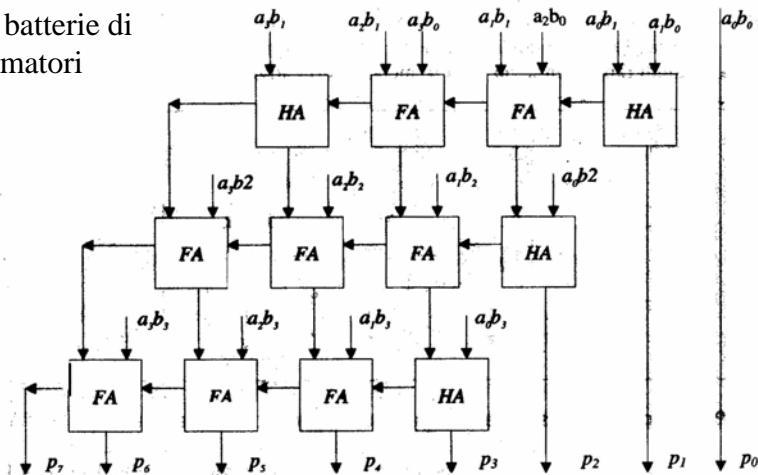
$$\begin{array}{r}
 11011 \times 27 \times \\
 1011 = 11 = \\
 \hline
 11111 \\
 11011 + 27 + \\
 11011 - 27 - = \\
 \hline
 00000 \quad 297 \\
 1010001 + \\
 00000 - = \\
 \hline
 11010 \\
 1010001 + \\
 11011 - - - = \\
 \hline
 100101001 \rightarrow 297|_{10}
 \end{array}$$



Circuito completo della somma dei prodotti parziali



N-1 batterie di sommatori



Problema: overflow: A e B su 32 bit \Rightarrow P su 64 bit.



Valutazione del cammino critico



Cammini critici:

Half Adder:

Somma - 1 porta

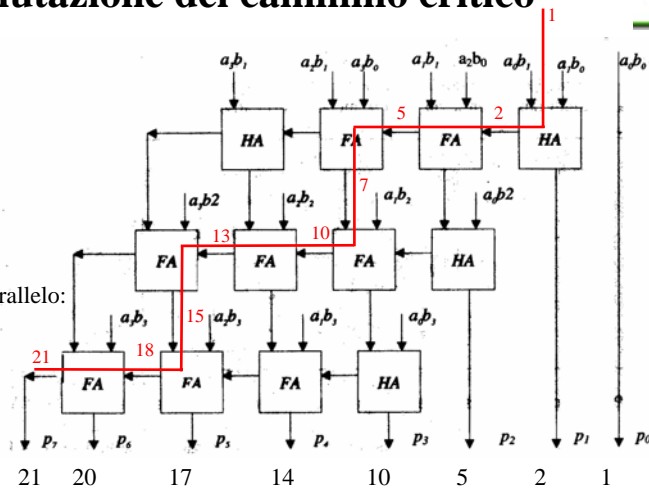
Riporto - 1 porta

Full Adder:

Somma - 2 porte

Riporto - 3 porte

Gli AND operano in parallelo:
ritardo 1.



Cammino critico: 19



Sommario



Addizionatori

Moltiplicatori

Algoritmi per la moltiplicazione

Diapositiva 21

b2 prova lalfafdafdakmn adf
borghese; 14/03/2005



L'approccio firmware



Nell'approccio firmware, viene inserita nella ALU una unità di controllo e dei registri. L'unità di controllo attiva opportunamente le unità aritmetiche ed il trasferimento da/verso i registri. Approccio "controllore-datapath".

Viene inserito un microcalcolatore dentro la ALU.

Il primo microprogramma era presente nell'IBM 360 (1964).



Shift (scalamento)

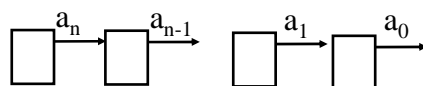


Dato A su 32 bit: $a_j = a_{j-k}$ k shift amount ($>$, $=$, $<$ 0).

Effettuato al di fuori delle operazioni selezionate dal Mux della ALU, da un circuito denominato *Barrel shifter*.

Tempo comparabile con quello della somma.

Operazioni codificate in modo specifico nell'ISA.



Shift dx 1



Il bit a_0 si "perde".
Il bit $a_n = 0$.



Algoritmi per la moltiplicazione



Il razionale degli algoritmi firmware della moltiplicazione è il seguente.

$$\begin{array}{r}
 \text{Moltiplicando} \quad 11011 \times \\
 \text{Moltiplicatore} \quad 101 =
 \end{array}$$

Si analizzano sequenzialmente i bit del moltiplicatore e:

- 1) Si mette 0 nella posizione opportuna (se il bit analizzato del moltiplicatore = 0).
- 2) Si mette una copia del moltiplicando nella posizione opportuna (se il bit analizzato del moltiplicatore è = 1).

$$\begin{array}{r}
 \text{-----} \\
 11011 + \\
 00000 - \\
 11011 - - \\
 \text{-----} \\
 \text{Prodotto} \quad 10000111
 \end{array}$$



Moltiplicazione utilizzando somma e shift



Utilizzo un registro prodotto da 64 bit, inizializzato a 0.

$$\begin{array}{r}
 11011 \times \quad A \\
 111 = \quad B
 \end{array}$$

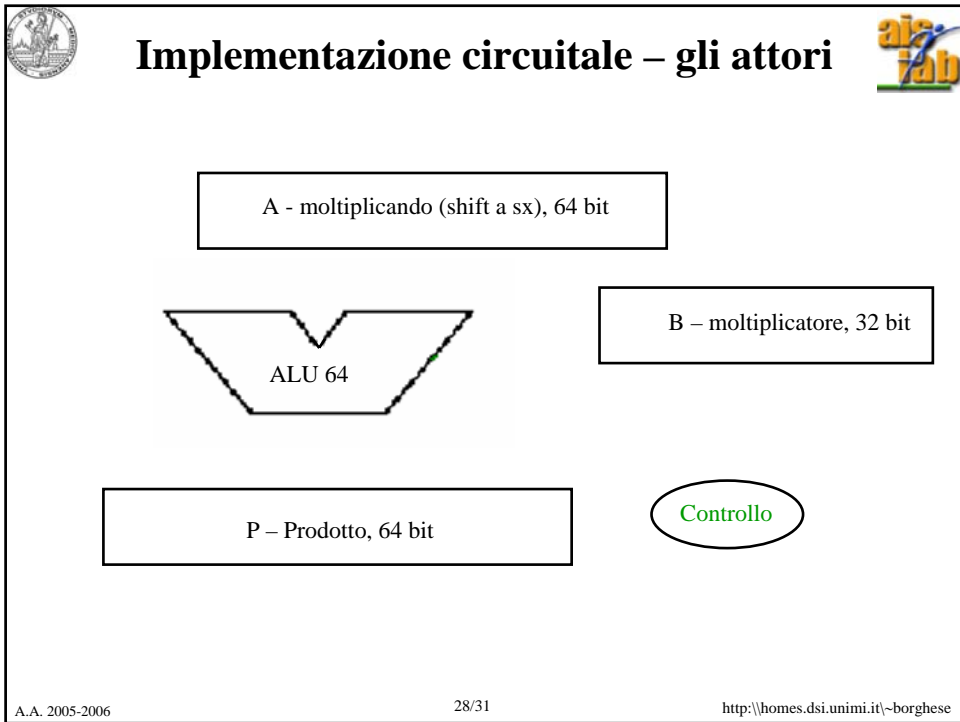
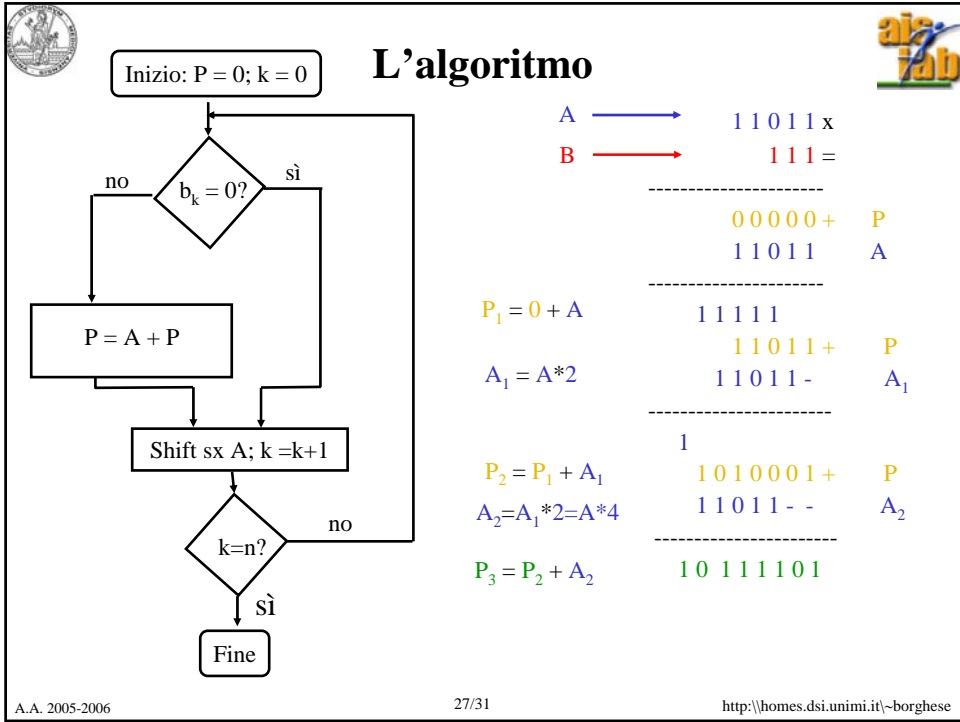
Itero per ogni bit del moltiplicatore:

A) Sommo il moltiplicando al prodotto se il bit = 1.

B) Shift a sx di un bit il moltiplicando

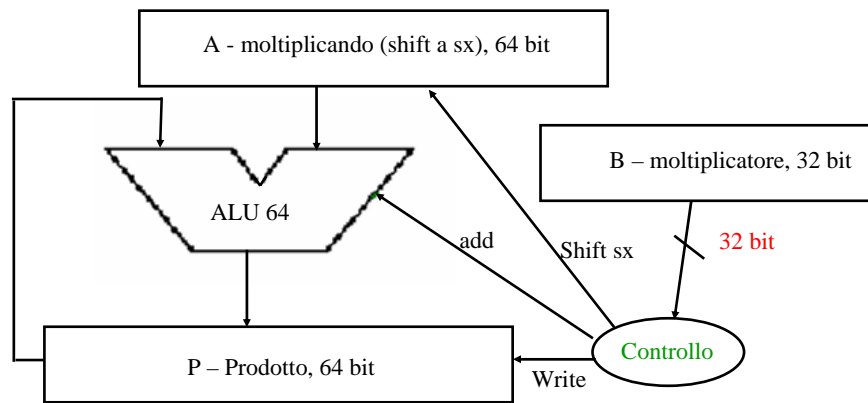
$$(A' = A * \text{base}).$$

$$\begin{array}{r}
 \text{-----} \\
 00000 + \\
 11011 \\
 \text{-----} \\
 11111 \\
 11011 + \quad P \\
 11011 - \quad A \\
 \text{-----} \\
 1 \\
 1010001 + \\
 11011 - - \\
 \text{-----} \\
 10111101
 \end{array}$$





Implementazione circuitale



Qual'è il problema?



Esercizi



Costruire il circuito HW che esegui la moltiplicazione 7×9 in base 2.

Eeguire la stessa moltiplicazione secondo l'algoritmo visto, indicando passo per passo il contenuto dei 3 componenti: A che contiene il moltiplicando, B che contiene il moltiplicatore e P che contiene somme parziali ed il risultato finale.



Sommario



Addizionatori

Moltiplicatori

Algoritmi per la moltiplicazione