Correzione degli esercizi

•Scrivere la rappresentazione binaria dei numeri decimali:

- •30 → 11110
- •36 → 100100
- •15 → 1111

Correzione degli esercizi

•Scrivere la rappresentazione decimale dei numeri binari:

- •1000 → 8
- •1010 → 10
- •01011 → 11
- •10111 → 23

Somma binaria

- **Colonna per colonna**, da destra a sinistra
- **Riporto** se la somma su una colonna supera la base

$$\begin{array}{r}0\\+0\\\hline 0\end{array}\quad\begin{array}{r}1\\+0\\\hline 1\end{array}\quad\begin{array}{r}0\\+1\\\hline 1\end{array}\quad\begin{array}{r}1\\+1\\\hline 10\end{array}$$

- Tre cifre binarie (prima riga, seconda riga, riporto), **somma =1** se una o tre sono 1, **riporto = 1** se almeno due sono 1

Riporto: 1 1 1 1 0 0

$$\begin{array}{r}011100_2 +\\100111_2 =\\\text{-----------}\\1000011_2\end{array}$$

```
1    11     riporti
 1010011 +
 1100011 =
----------
10110110
```

Si vuole quindi **costruire un circuito** per **sommare due numeri binari**

```
10000110    riporti
 1010011 +
 1100011 =
----------
10110110
```

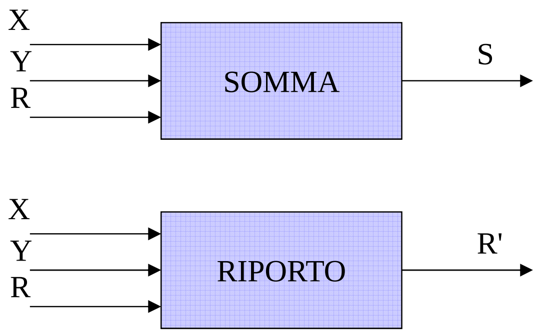
Iniziamo con un **circuito** che faccia la **somma su una colonna**

Abbiamo tre cifre binarie **X, Y, R** in **input** mentre in **output** vogliamo ottenere la **somma S** ed il **riporto R'**

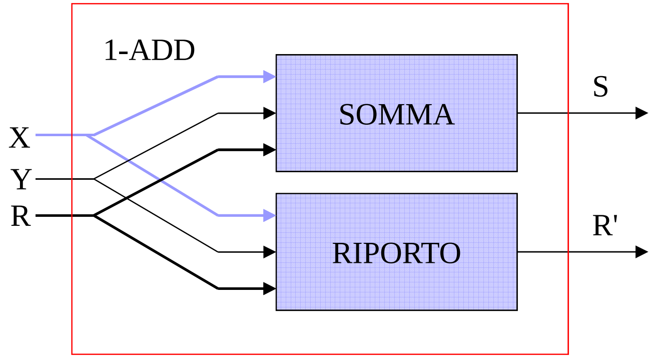
Tabella di verità

X	Y	R	S	R'
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

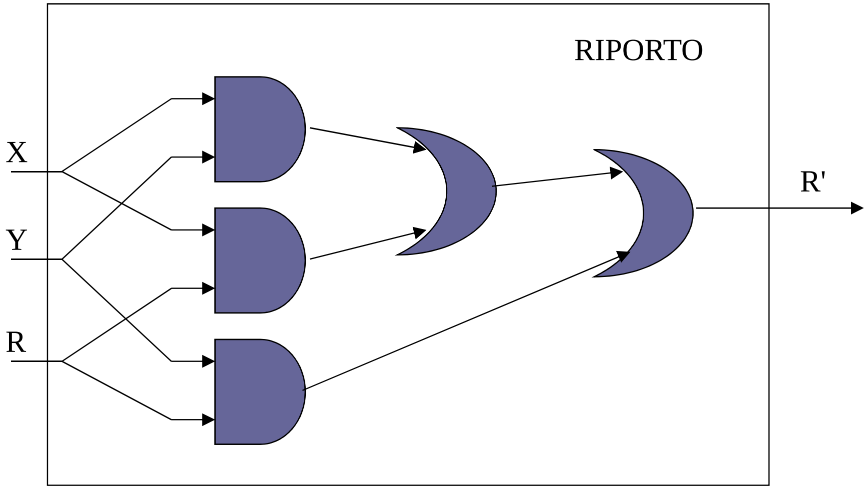
Supponiamo di avere i circuiti che calcolano **somma** e **riporto**



Possiamo allora **combinare** i circuiti **SOMMA** e **RIPORTO** per ottenere il seguente circuito **1-ADD**



Il circuito **RIPORTO** puo` essere realizzato nel seguente modo



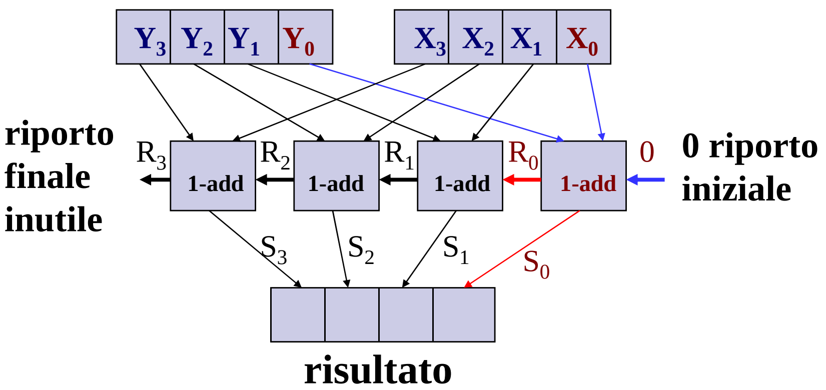
Basta infatti verificare la corrispondente tabella di verita’

Il circuito **SOMMA** naturalmente puo' pure essere realizzato (vedi dispensa).

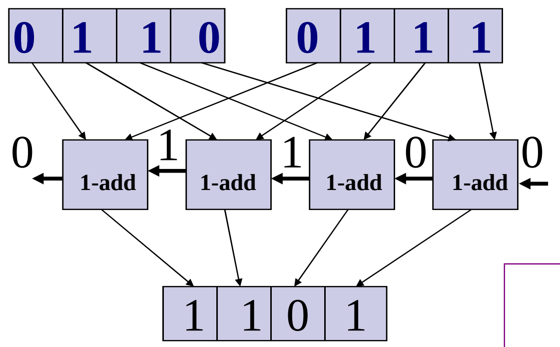
A questo punto **componendo K circuiti 1-ADD** e` possibile realizzare **un circuito K-ADD** che **somma due numeri binari di K cifre**.

Vediamo l'esempio della somma di due numeri binari di 4 cifre.

Somma di numeri di 4 bit



Esempio




```
0111 +
0110 =
------
1101
```

Attenzione

Si e` **trascurato** il problema del cosiddetto **overflow**, cioe' il risultato e' troppo grande per essere contenuto nei bit disponibili.

Per esempio:

```
  0111 +
  1110 =
 ------
 10101
```

Esercizi

```
11011+
 1100
---------
```

```
11111+
     1
---------
```

Correzioni

```
 1
 11011+
   1100
---------
100111
```

```
11111
 11111+
      1
---------
100000
```

Reali in notazione binaria

- $b_{k-1}\ b_{k-2} \ldots b_2\ b_1\ b_0\ ,\ b_{-1}\ b_{-2} \ldots$
- $b_{k-1} \times 2^{k-1} + b_{k-2} \times 2^{k-2} + \ldots + b_2 \times 2^2 + b_1 \times 2 + b_0 \times 2^0 + b_{-1} \times 2^{-1} + b_{-2} \times 2^{-2} + \ldots$
- Da decimale a binario:
  - Per la **parte intera**, **come sappiamo fare** (**metodo delle divisioni**)

REALE--> BINARIO

cosa significa una **parte frazionaria binaria**:

$$.1101001$$

$$2^{-1} + 2^{-2} + 2^{-4} + 2^{-7}$$

$$.1101001$$

$$2^{-1}\ 2^{-2}\ldots$$

moltiplicarlo per 2 significa spostare il punto di un posto a destra

$$1.101001$$

$$2^{0}\ 2^{-1}\ldots\ldots$$

Se abbiamo un valore decimale in base 10:

0.99 come troviamo la sua **rappresentazione in base 2**? Ragioniamo come segue:

Supponiamo che $.99 = .b_1b_2b_3\ldots b_k$ (binario)

Allora $2\times .99 = 1.98 = \mathbf{b_1}.b_2b_3\ldots b_k$

Quindi $\mathbf{b_1}$ è 1

e $.98$ è rappresentato da $.b_2b_3\ldots b_k$

Per trovare la **rappresentazione binaria** di **un decimale** lo moltiplichiamo per 2 ed osserviamo se **1** appare **nella parte intera:**

rappresentazione binaria di .59

.59×2= 1.18	**.72×2= 1.44**
.18×2= 0.36	**.44×2= 0.88**
.36×2= 0.72	**.88×2= 1.76**
	....

.100101.....

dipende da quanti bit abbiamo

Esempio

18.59

$18 \rightarrow 10010$ (metodo della divisione per 2)

$.59 \rightarrow .100101\ldots$(metodo della moltiplic. per 2)

10010.100101....

Esercizi

- Convertire i seguenti numeri binari in formato decimale:
  - 11,01
  - 101,111
  - 10,1
- Esprimere i seguenti valori in notazione binaria:
  - 4.5
  - 2.75
- Eseguire le seguenti somme binarie:
  - 1010,001+1,101
  - 111,11+0,01