Codici di correzione di errore

Codici di correzione di errore

- I computer possono, occasionalmente, commettere degli errori: errori, al fine di rilevare e correggere l'eventuale errore si possono aggiungere dei bit extra alla parola.

- Una **parola di codice** (o **codeword**) con n-bit è una parola che contiene **m** bit per i dati e **r** per il controllo dell'errore (n=m+r).

- La **distanza di Hamming** tra due parole di codice è data dal numero di differenze tra bit corrispondenti. Per il calcolo è sufficiente sommare il risultato dell'EXOR bit-a-bit delle due parole.

Codice di Hamming

- Ricordiamo che

Dimensione della parola	Bit di correzione	Posizioni individuate	Bit totali m+r	Spreco
m=4	r=3	$2^r \geq 4+r$	7	3/4
m=8	r=4	$2^r \geq 8+r$	12	4/8
m=16	r=5	$2^r \geq 16+r$	21	5/16

- Quindi con r bit possiamo indiduare la posizione nella parola (m+r)

Codice di Hamming: 4 bit dati e 3 di controllo

- Ricordando che per comporre i numeri da 1 a 7 utilizziamo le potenze di due:
  - 7 = 4 + 2 + 1
  - 6 = 4 + 2
  - 5 = 4 + 1
  - 4 = 4
  - 3 = 2 + 1
  - 2 = 2
  - 1 = 1

	4	2	1
7	1	1	1
6	1	1	
5	1		1
4	1		
3		1	1
2		1	
1			1

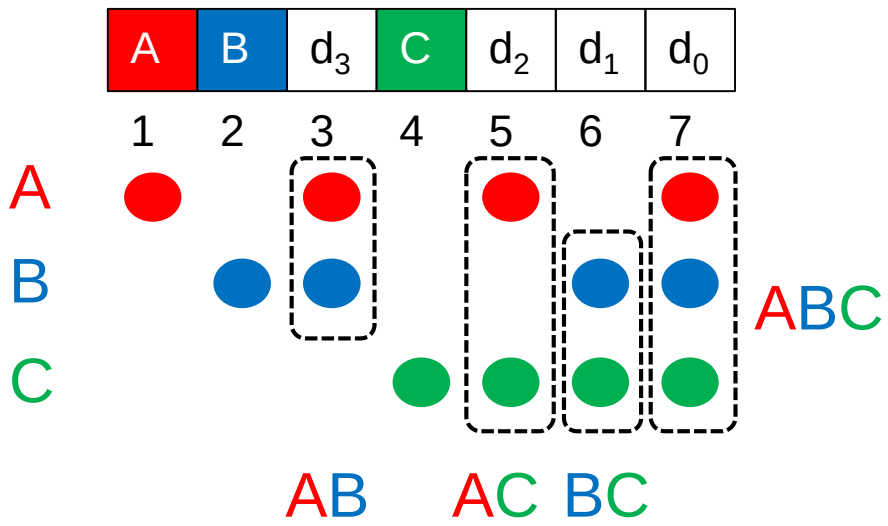
Codice di Hamming: 4 bit dati e 3 di controllo

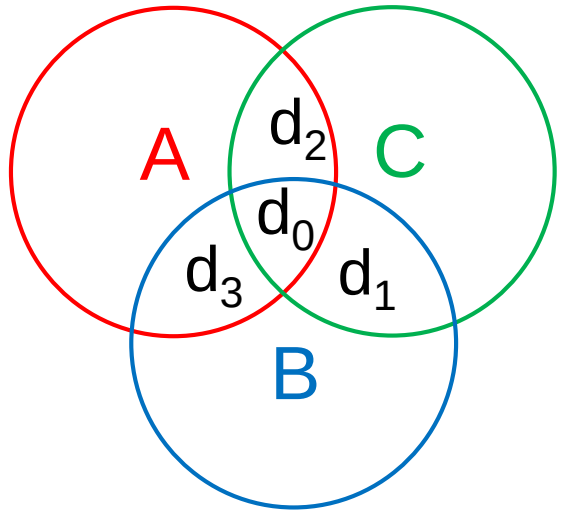
- Il valore 1 lo utilizziamo per comporre i numeri 1, 3, 5, 7
- Il valore 2 lo utilizziamo per comporre i numeri 2, 3, 6, 7
- Il valore 4 lo utilizziamo per comporre i numeri 4, 5, 6, 7
- Quindi potremmo utilizzare tre bit per indicare tutte le possibili combinazioni dei numeri componibili di una parola con 7 cifre…

	4	2	1
7	1	1	1
6	1	1	
5	1		1
4	1		
3		1	1
2		1	
1			1

Codice di Hamming: 4 bit dati e 3 di controllo

- Se A rappresenta il valore 1 controlla le posizioni 1, 3, 5, 7
- Se B rappresenta il valore 2 controlla le posizioni 2, 3, 6, 7
- Se C rappresenta il valore 4 controlla le posizioni 4, 5, 6, 7

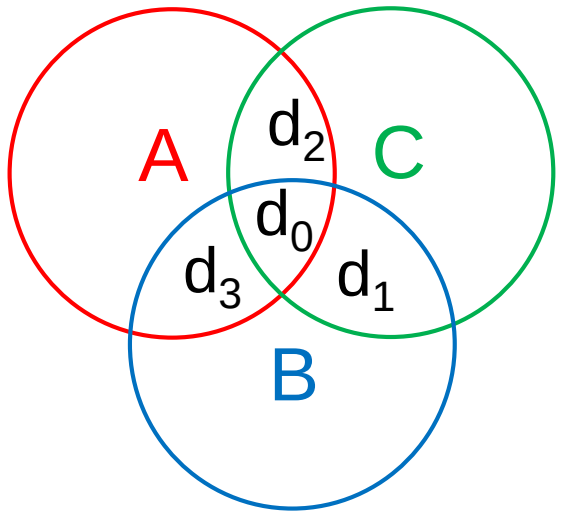




Esercizio

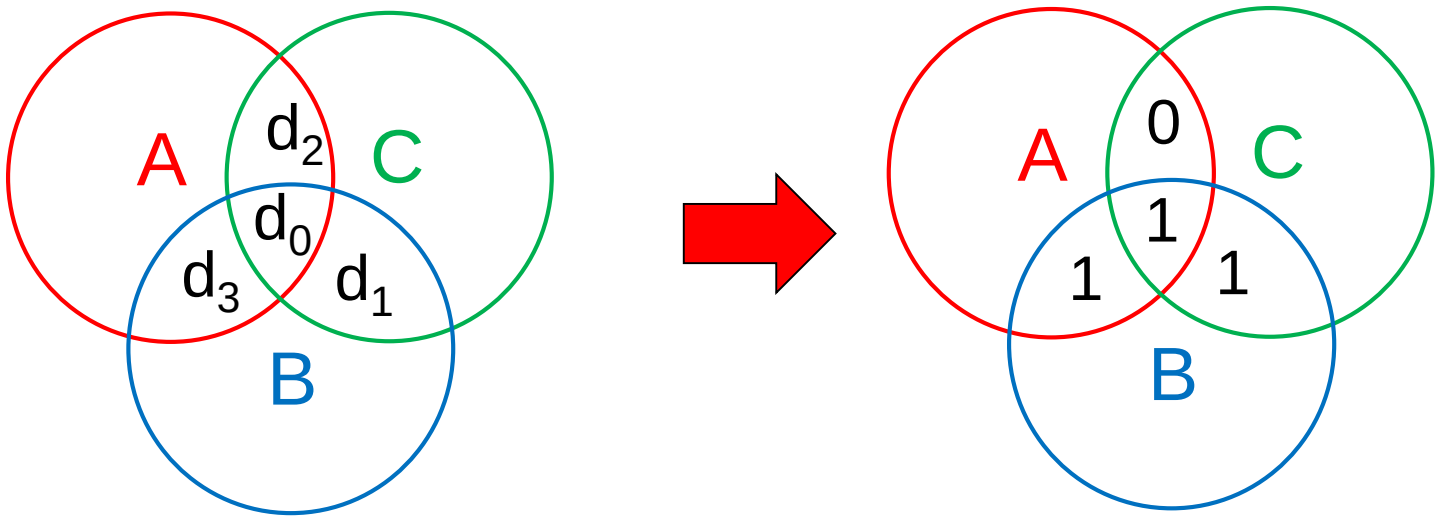
- Supponendo di disporre della parola binaria 1011 ($d_3d_2d_1d_0$).
- Trovare i bit del codice di correzione di Hamming

- *Suggerimento*
- Utilizzare questo schema per compilare le possibili intersezioni:



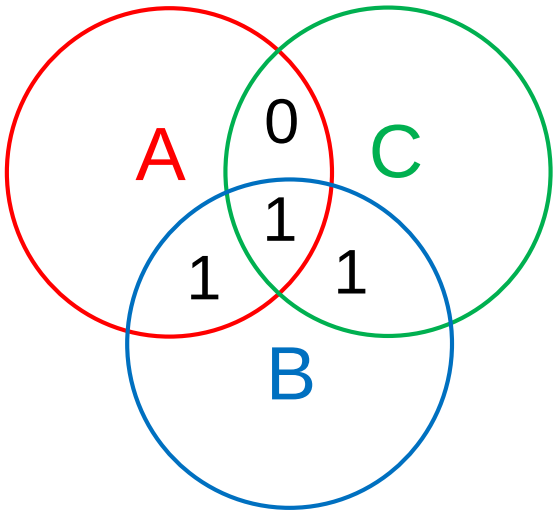
Soluzione

- Partendo dalla parola binaria parola binaria 1011 ($d_3d_2d_1d_0$), scrivere il valore che corrisponde nelle intersezioni:



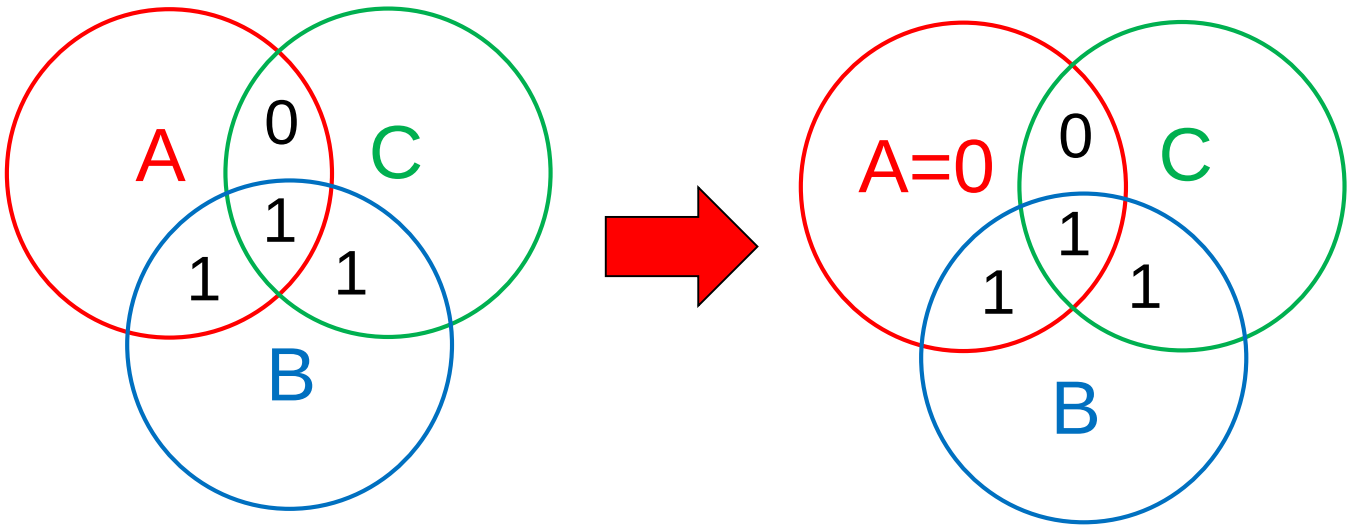
Calcolo dei bit di controllo

- Calcolare A, B e C in modo tale che il numero di 1 all'interno del cerchio sia pari:



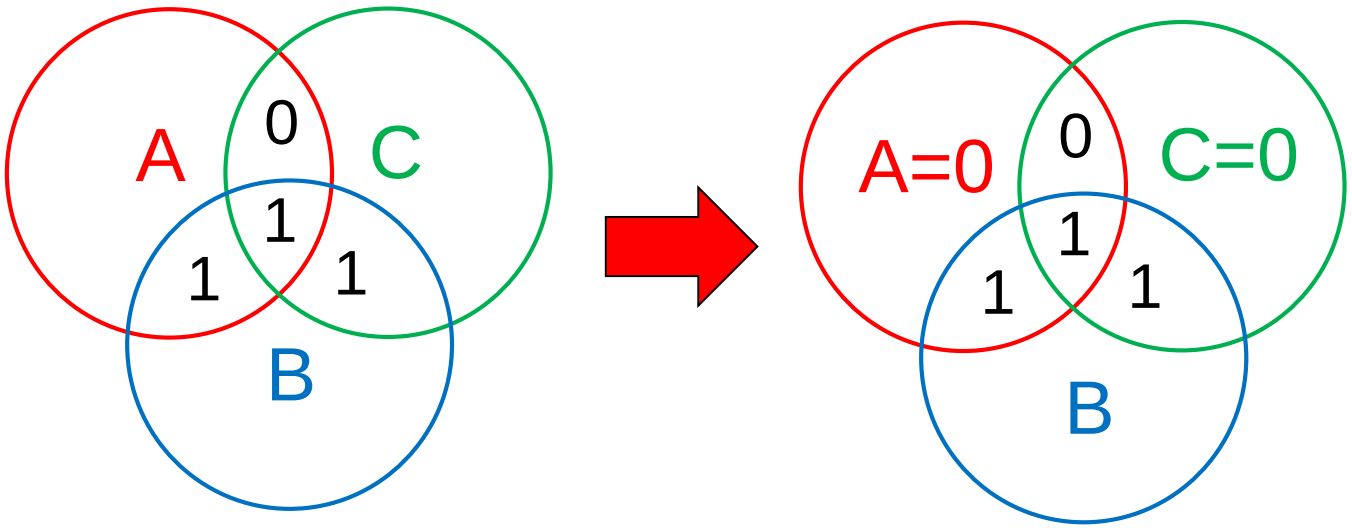
Esercizio – calcolo dei bit di controllo

- Calcolare A, B e C in modo tale che il numero di 1 all'interno del cerchio sia pari:



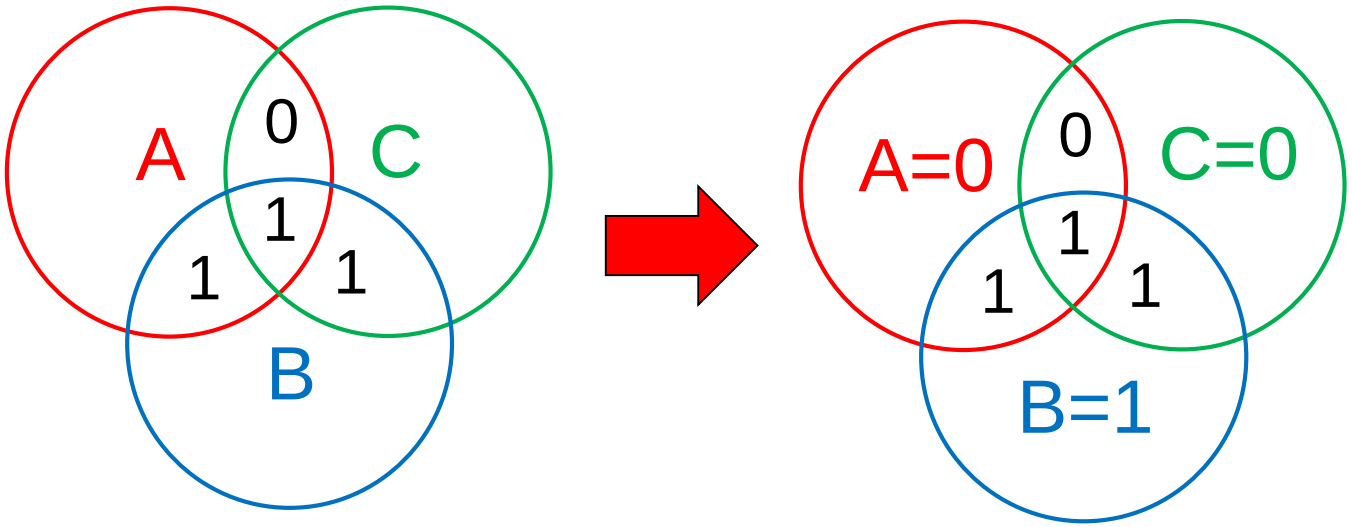
Esercizio – calcolo dei bit di controllo

- Calcolare A, B e C in modo tale che il numero di 1 all'interno del cerchio sia pari:



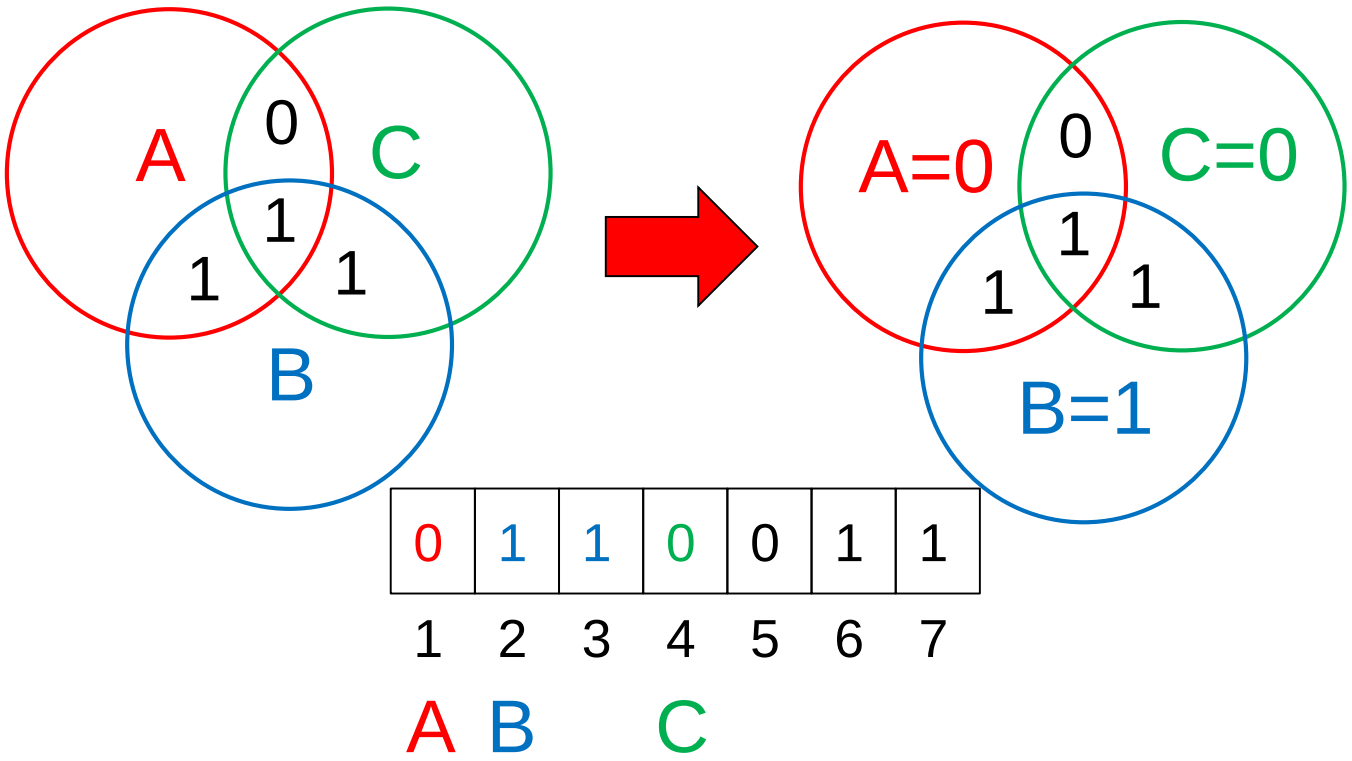
Esercizio – calcolo dei bit di controllo

- Calcolare A, B e C in modo tale che il numero di 1 all'interno del cerchio sia pari:



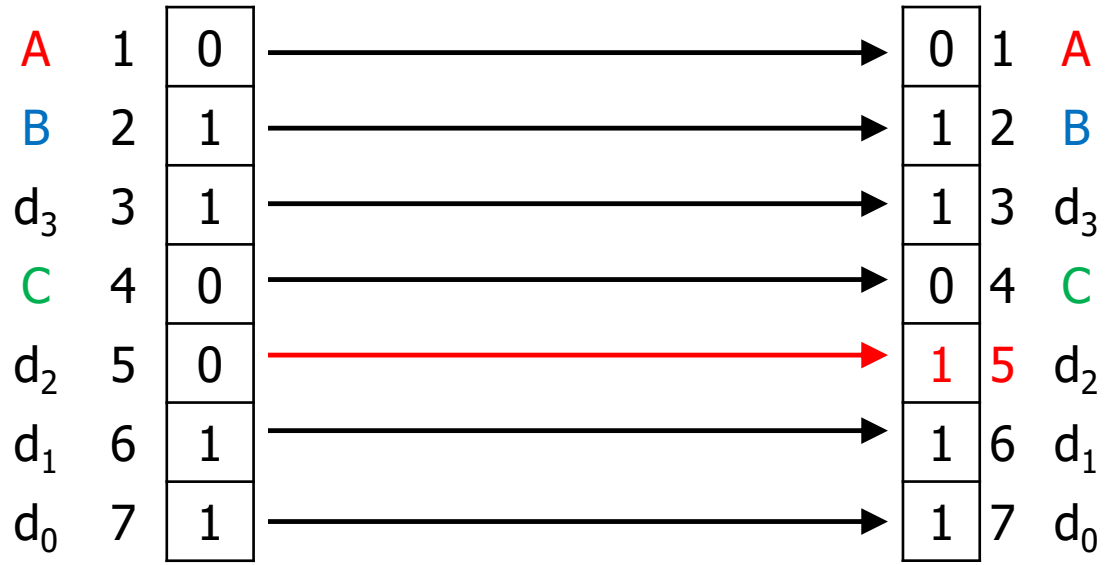
Esercizio – calcolo dei bit di controllo

- Calcolare A, B e C in modo tale che il numero di 1 all'interno del cerchio sia pari:



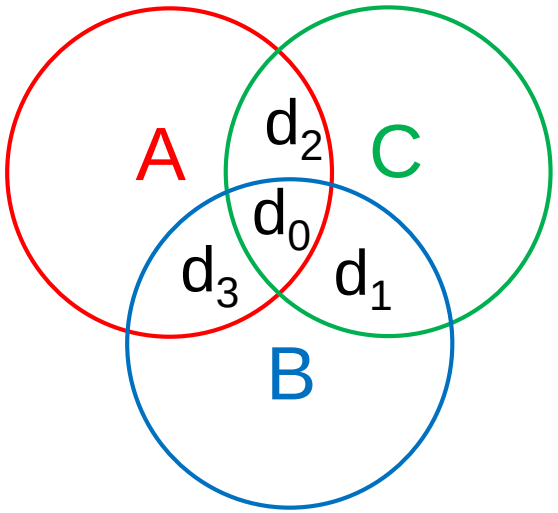
Simulazione di errore

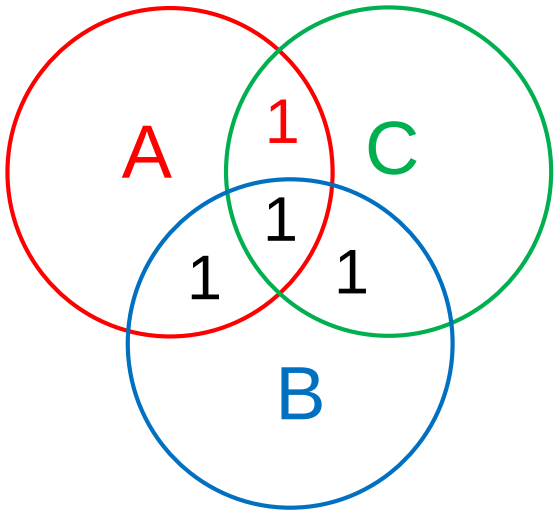
- Supponiamo che, a seguito di un errore di trasferimento, si modifichi il bit d_2
- Possiamo rilevarlo e recuperarlo grazie ai bit di controllo



Simulazione di errore

- Calcoliamo inizialmente i bit di parità

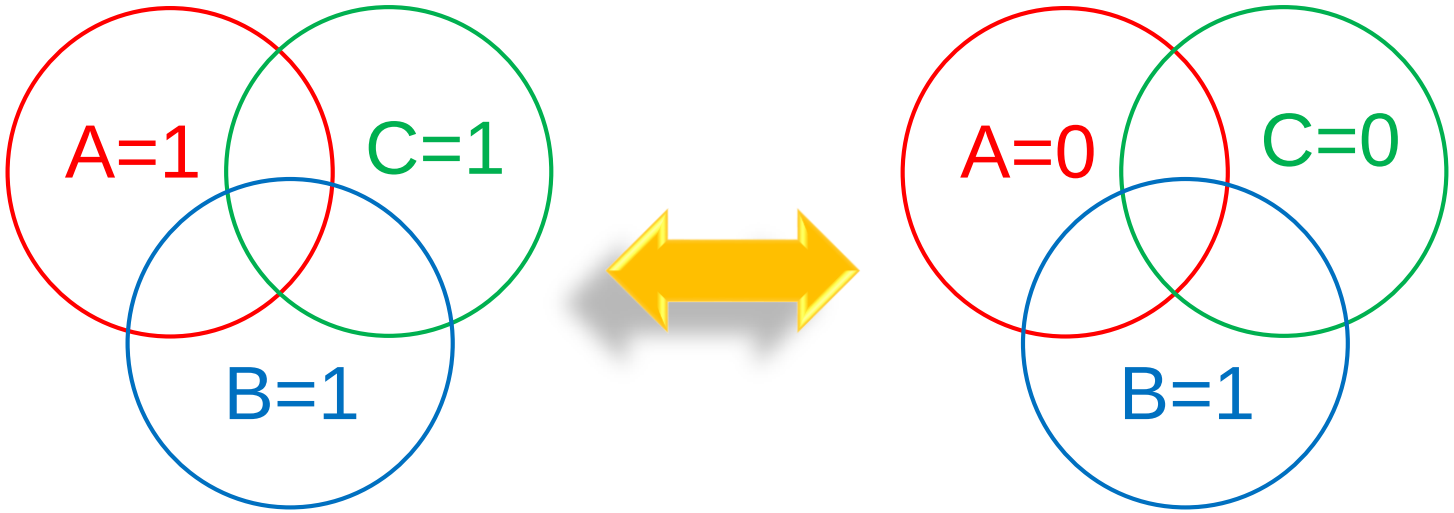




0	1	A
1	2	B
1	3	d_3
0	4	C
1	5	d_2
1	6	d_1
1	7	d_0

Simulazione di errore

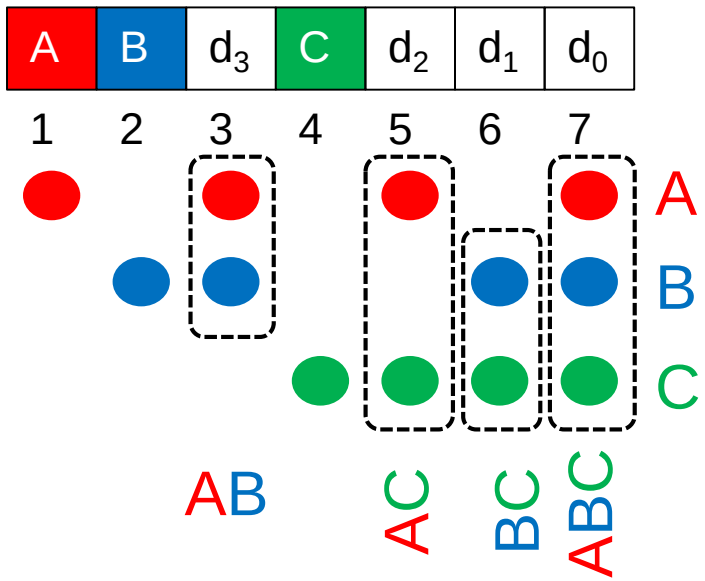
- A questo punto confrontiamoli con i vecchi valori



- Notiamo che A e C sono cambiati

Simulazione di errore

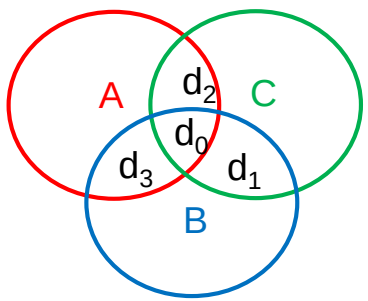
- I cerchi A e C hanno una parità errata rispetto a prima, l'unico indice in comune solo tra A e C è 5 cioè il bit dati d_2



Algoritmo per codice di Hamming con 4 bit dati

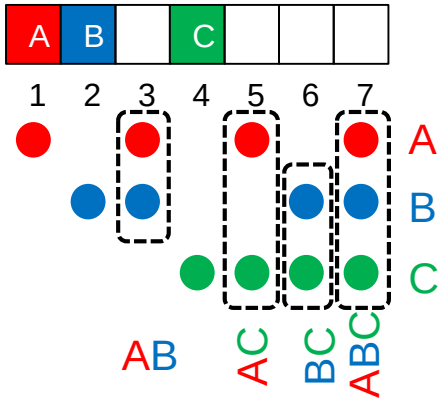
- È possibile calcolare i bit di controllo attraverso il calcolo matriciale, facendo attenzione di esprimere i valori della matrice risultante in binario

$$\begin{bmatrix} 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} d_3 \\ d_2 \\ d_1 \\ d_0 \end{bmatrix} = \begin{bmatrix} d_3+d_2+d_0 \\ d_3+d_1+d_0 \\ d_3 \\ d_1+d_2+d_0 \\ d_2 \\ d_1 \\ d_0 \end{bmatrix}_2 \begin{matrix} A \\ B \\ \\ C \\ \\ \\ \end{matrix} = P$$



Algoritmo per codice di Hamming con 4 bit dati

- A questo se non ci sono errori nel vettore P, il risultato del prodotto con la seguente matrice, espresso in base due, è un vettore con tutti valori zero:



$$\begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix} \times P = C = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}_2$$

- in caso contrario in C troviamo l'indicazione della posizione errata