



Codici di Correzione degli Errori

Architettura degli elaboratori

Laurea in Informatica

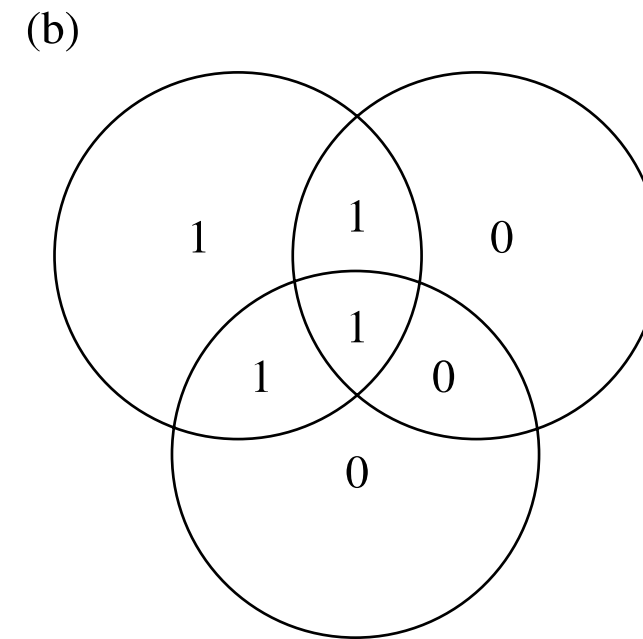
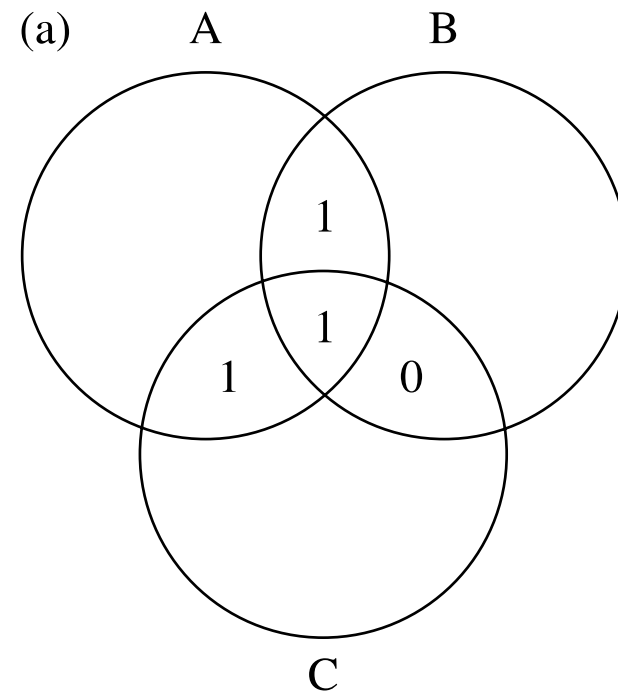
Docente: Federico Corò

ECC – Error Correction Code

- Implementato nella memoria RAM di server e workstation
- Implementato in Cache in molte CPU (e.g. AMD ZEN, Intel Xeon/Core)
- Single-Error-Correcting (SEC) code
 - Hamming Error Correcting Code
- SEC-DED: Single-Error-Correcting, Double Error Detecting

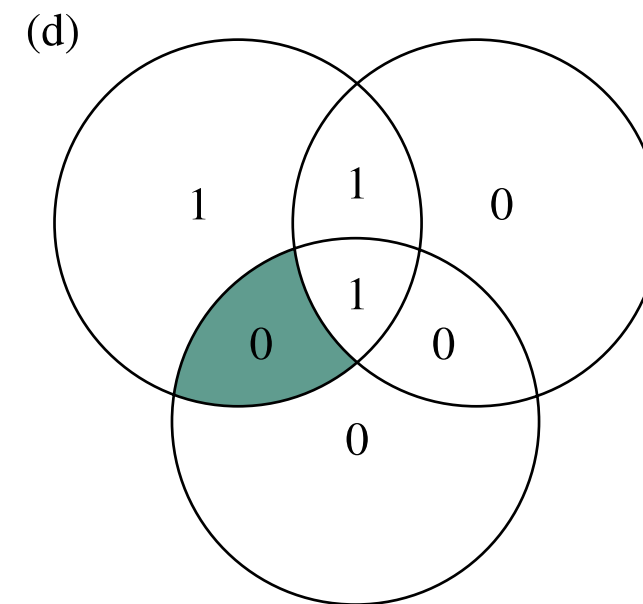
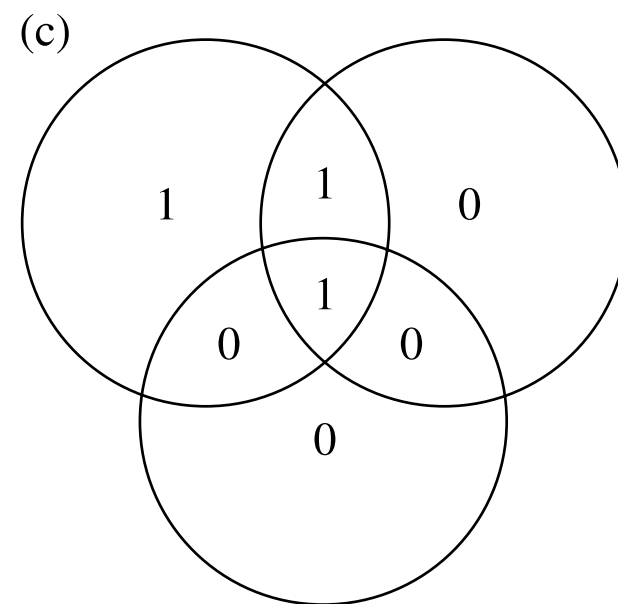
Codice di Hamming

a) 4 bit originali da salvare



b) **Bit di controllo**
per la correzione
errori
Parity bit: la somma
degli elementi di
ogni insieme deve
essere pari

c) Quando i bit
vengono letti, può
capitare un errore
(switch da 0 a 1 o
vice-versa) **sia nei bit
del messaggio
originale che nei bit
di controllo**



c) Algoritmo di
correzione errori
permette di
individuare quale bit
è affetto da errore e
correggerlo

Figure 5.8 Hamming Error-Correcting Code

Codice di Hamming

- Permette di **correggere** errori di un singolo bit
- Quanti **bit di controllo** servono per un input di M bit?
Chiamiamo tale numero K
- Idea:
 - Ognuno degli $M+K$ bit (dati o di controllo) totale possono essere affetti da errore
 - per la correzione, ho bisogno di poter indicare quale degli $M+k$ bit è affetto da errore, + il caso in cui non ci sono errori
 - Totale: $M+K+1$ combinazioni
- Ci serve il minimo K tale che $2^K \geq M + K + 1$

Codice di Hamming

- Ci serve il minimo K tale che $2^K \geq M + K + 1$
- Ad esempio, per $M=8$
 - Servono $K=4$ bit, infatti $2^4 \geq 8 + 4 + 1$

Codice di Hamming: come posizionare i bit di controllo?

- **Sindrome:** XOR (bit a bit) tra i bit di controllo di un messaggio letti dalla RAM e quelli calcolati a partire dal corpo del messaggio
- **Desiderata:**
 - se la sindrome contiene tutti 0, non ci sono stati errori
 - Se contiene un singolo bit a 1, l'errore è in un bit di controllo, quindi il messaggio rimane corretto
 - Se contiene più bit a 1, indicano la posizione del bit con errore (di cui deve essere eseguito lo switch)
- **Soluzione:** bit di controllo in posizioni potenza di 2

Codice di Hamming: come posizionare i bit di controllo?

Bit position	12	11	10	9	8	7	6	5	4	3	2	1
Position number	1100	1011	1010	1001	1000	0111	0110	0101	0100	0011	0010	0001
Data bit	D8	D7	D6	D5	D4	D3	D2	D1				
Check bit					C8				C4		C2	C1
Word stored as	0	0	1	1	0	1	0	0	1	1	1	1
Word fetched as	0	0	1	1	0	1	1	0	1	1	1	1
Position number	1100	1011	1010	1001	1000	0111	0110	0101	0100	0011	0010	0001
Check bit					0				0		0	1

Figure 5.10 Check Bit Calculation

0 1 1 0

- Ogni bit di controllo calcola la parità (corrispondente allo XOR) tra i bit il cui position number contiene un 1 nella stessa posizione del bit di controllo.
- Esempio: store e fetch della parola, c'è un errore al bit 6

ESERCIZIO 1 Codice di correzione di Hamming

Si supponga che una parola di dati da 8 bit memorizzata sia

11001010

Adottando l'algoritmo di **Hamming**, determinare quanti e quali bit di controllo verrebbero immagazzinati in memoria insieme alla parola di dati, ed in quale posizione.

SOLUZIONE ESERCIZIO 1 Codice di correzione di Hamming

Posizione	12	11	10	9	8	7	6	5	4	3	2	1
Pos. bin	1100	1011	1010	1001	1000	0111	0110	0101	0100	0011	0010	0001
Data bit	D8	D7	D6	D5		D4	D3	D2		D1		
Check bit					C8				C4		C2	C1
	1	1	0	0	0	1	0	1	1	0	0	1

Sapendo che:

$$C1 = D1 \oplus D2 \oplus D4 \oplus D5 \oplus D7$$

$$C2 = D1 \oplus D3 \oplus D4 \oplus D6 \oplus D7$$

$$C4 = D2 \oplus D3 \oplus D4 \oplus D8$$

$$C8 = D5 \oplus D6 \oplus D7 \oplus D8$$

Otteniamo che:

- $C1 = 0 \oplus 1 \oplus 1 \oplus 0 \oplus 1 = 1$
- $C2 = 0 \oplus 0 \oplus 1 \oplus 0 \oplus 1 = 0$
- $C4 = 1 \oplus 0 \oplus 1 \oplus 1 = 1$
- $C8 = 0 \oplus 0 \oplus 1 \oplus 1 = 0$

ESERCIZIO 2 Codice di correzione di Hamming

Per la parola

00111001

i bit di controllo memorizzati sono 0111.

Si supponga che, quando la parola viene letta dalla memoria, i bit di controllo siano calcolati per essere 1101.

Quale parola di dati è letta dalla memoria?

SOLUZIONE ESERCIZIO 1 Codice di correzione di Hamming

Posizione	12	11	10	9	8	7	6	5	4	3	2	1
Pos. bin	1100	1011	1010	1001	1000	0111	0110	0101	0100	0011	0010	0001
Data bit	D8	D7	D6	D5		D4	D3	D2		D1		
Check bit					C8				C4		C2	C1
	0	0	1	1	0	1	0	0	1	1	1	1

Calcoliamo la sindrome, pertanto la parola sindrome è:

$$(0\ 1\ 1\ 1) \oplus (1\ 1\ 0\ 1) = 1\ 0\ 1\ 0$$

quindi il bit errato è quello in posizione (in base 10) 10 (bit dati D6). Ne consegue che la parola letta dalla memoria è

00011001

ESERCIZIO 3 Codice di correzione di Hamming

Quanti bit di controllo sono necessari se il codice a correzione di errore di Hamming viene usato per rilevare errori di bit singoli in una parola di dati a 1024 bit?

SOLUZIONE ESERCIZIO 3 Codice di correzione di Hamming

Dobbiamo trovare il valore minimo di K tale che:

$$2^K \geq M + K + 1$$

Se $M = 1024 = 2^{10}$

$$2^K \geq 2^{10} + K + 1$$

Sicuramente abbiamo che $K > 10$. Verifichiamo se $K = 11$ soddisfa la disuguaglianza

$$2^{11} \geq 2^{10} + 11 + 1?$$

Vero! Quindi la risposta è $K = 11$.

ESERCIZIO 4 Codice di correzione di Hamming

Si consideri il codice di correzione di Hamming (SEC) per una parola di dati a 16 bit.

Data la seguente parola dati

0101000000111001

riportare tutti i bit scritti in memoria.

SOLUZIONE ESERCIZIO 4 Codice di correzione di Hamming

Poichè $M = 16 = 2^4$, sappiamo che $K > 4$. $K=5$ va bene perchè

$$32 = 2^5 \geq 2^4 + 5 + 1 = 22$$

Pertanto dobbiamo avere $M + K = 16 + 5 = 21$ bit disposti secondo lo schema che assegna ai bit di controllo le posizioni che sono potenze di 2 (C1, C2, C4, C8, C16) e alle altre posizioni i bit dati (da D1 a D16):

pos	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
Data bit	D16	D15	D14	D13	D12		D11	D10	D9	D8	D7	D6	D5		D4	D3	D2		D1		
Check bit						C16								C8				C4		C2	C1
	0	1	0	1	0		0	0	0	0	0	1	1		1	0	0		1		

Per capire quali bit dati contribuiscono a formare il valore per i vari bit di controllo bisogna rappresentare tutte le posizioni tramite i 5 bit di controllo e selezionare per ogni bit di controllo i bit dati che corrispondono ad un valore di 1 per quel bit

SOLUZIONE ESERCIZIO 4 Codice di correzione di Hamming

Per calcolare il valore per i vari bit di controllo, bisogna rappresentare tutte le posizioni tramite i 5 bit di controllo e selezionare per ogni bit di controllo i bit dati che corrispondono ad un valore di 1 per quel bit

C16	C8	C4	C2	C1	pos	Bit dati	Bit controllo
0	0	0	0	1	1		C1
0	0	0	1	0	2		C2
0	0	0	1	1	3	D1	
0	0	1	0	0	4		C4
0	0	1	0	1	5	D2	
0	0	1	1	0	6	D3	
0	0	1	1	1	7	D4	
0	1	0	0	0	8		C8
0	1	0	0	1	9	D5	
				
						D16	

Risultato Finale:

$$C1 = D1 \oplus D2 \oplus D4 \oplus D5 \oplus D7 \oplus D9 \oplus D11 \oplus D12 \oplus D14 \oplus D16$$

$$C2 = D1 \oplus D3 \oplus D4 \oplus D6 \oplus D7 \oplus D10 \oplus D11 \oplus D13 \oplus D14$$

$$C4 = D2 \oplus D3 \oplus D4 \oplus D8 \oplus D9 \oplus D10 \oplus D11 \oplus D15 \oplus D16$$

$$C8 = D5 \oplus D6 \oplus D7 \oplus D8 \oplus D9 \oplus D10 \oplus D11$$

$$C16 = D12 \oplus D13 \oplus D14 \oplus D15 \oplus D16$$

SOLUZIONE ESERCIZIO 4 Codice di correzione di Hamming

Possiamo finalmente calcolare I valori per i bit di controllo

pos	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
Data bit	D16	D15	D14	D13	D12		D11	D10	D9	D8	D7	D6	D5		D4	D3	D2		D1		
Check bit						C16								C8				C4		C2	C1
	0	1	0	1	0	0	0	0	0	0	0	1	1	0	1	0	0	0	1	0	1