

Read Only Memory (ROM)

- Memorizzazione permanente
 - Non volatili
- Usate per memorizzare:
 - microprogrammi
 - subroutine di libreria
 - programmi di sistema (BIOS)
 - funzioni tabulate

238

Tipi di ROM

- Scritte in produzione
 - Molto costoso per pochi “pezzi”
- Programmabili (una sola volta)
 - PROM
 - Necessitano di strumentazione speciale per la programmazione
- Principalmente di lettura (Read “mostly”)
 - Erasable Programmable (EPROM)
 - Si cancellano (per intero) tramite raggi ultravioletti
 - Electrically Erasable (EEPROM)
 - Impiegano molto più tempo per la scrittura che per la lettura
 - Memorie Flash
 - Cancellazione elettrica di blocchi di memoria

239

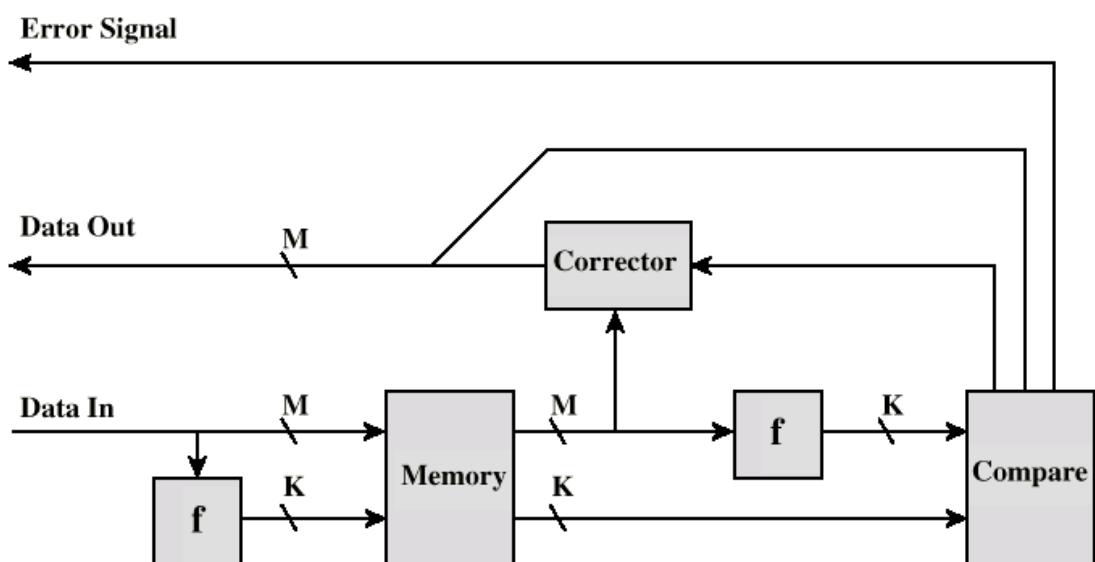
Correzione Errori



- Guasti Hardware (Hard Failure)
 - Guasti permanenti
- Errori Software (Soft Error)
 - Random, non-distruttivi
 - Danni alla memoria non permanenti
- Errori rilevati ed eventualmente corretti usando, ad esempio, codici correttori di Hamming

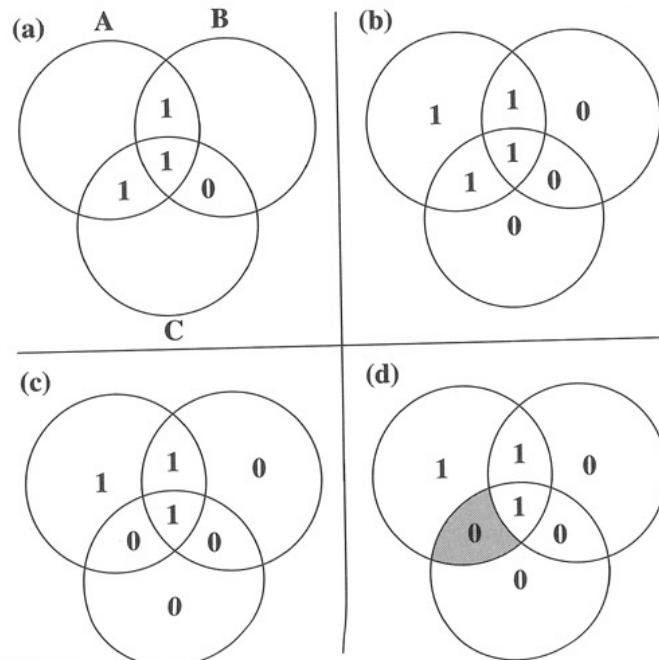
246

Schema di funzionamento del codice a correzione di errore



247

Esempio di codice a correzione di errore di Hamming



248

Correzione degli errori: disposizione bit

Quanti bit di controllo servono ? $2^K-1 \geq M + K$

Bit di dati	Bit di controllo	% incremento
8	4	50
16	5	31,25
32	6	18,75
64	7	10,94

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

249

Codice correzione Hamming

Esempio generazione bit di controllo

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

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

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

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

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

si ha

$$C1 = 0 \oplus 1 \oplus 0 \oplus 0 \oplus 1 = 0$$

$$C2 = 0 \oplus 0 \oplus 0 \oplus 0 \oplus 1 = 1$$

$$C4 = 1 \oplus 0 \oplus 0 \oplus 1 = 0$$

$$C8 = 0 \oplus 0 \oplus 1 \oplus 1 = 0$$

OR esclusivo (XOR)

$$0 \oplus 0 = 1 \oplus 1 = 0$$

$$1 \oplus 0 = 0 \oplus 1 = 1$$

250

Correzione degli errori: disposizione bit

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	0	1	1	0	1	1	1
Position Number	1100	1011	1010	1001	1000	0111	0110	0101	0100	0011	0010	0001
Check Bit					0				0		0	1
risultato XOR												
0 1 1 0												

251

Correzione degli errori: SEC-DED

correzione di singolo errore, rilevazione di 2 errori

