



FONDAMENTI DI INFORMATICA

Lezione n. 8

- DESCRIZIONE LIVELLO REGISTRO
- REGISTER TRANSFER LEVEL (RTL)
- I MODULI BASE RTL
- STRUTTURE DI INTERCONNESSIONE
- DESCRIZIONE E PROGETTO A LIVELLO RTL

In questa lezione sono presentate le caratteristiche e i moduli primitivi del livello a trasferimento fra registri (RTL). Questi moduli corrispondono a elementi fisici disponibili sia nella progettazione discreta sia in quella VLSI.

Si discutono le strutture di interconnessione che hanno un ruolo estremamente importante nelle prestazioni dei sistemi di elaborazione



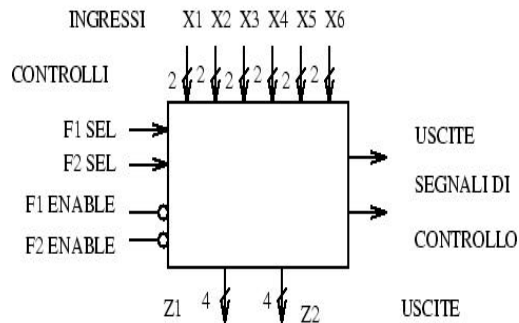
LIVELLO RTL

- E' immediatamente superiore al livello logico nella gerarchia di analisi di un sistema di elaborazione.
- Le informazioni binarie sono raggruppate in modo da formare parole o word o vettori.
- I componenti primitivi sono circuiti sequenziali o combinatori progettati per elaborare o immagazzinare parole.
- I componenti a livello registro sono (o sono stati) i blocchi elementari integrati MSI.



LIVELLO RTL

- Non esistono simboli adottati universalmente per i circuiti a livello di registro.
- Lo schema in figura rappresenta un modulo generico a livello RTL.



- Questa rappresentazione non è codificata, ma quando si rappresenta un modulo RTL, si fa riferimento a modelli di questo tipo.
- Negli ultimi anni sono stati introdotti linguaggi di descrizione dell'HW a livello logico, RTL e funzionale.



LIVELLO RTL

MODULO	FUNZIONE
Porte generalizzate	Operazioni booleane
Multiplexers	Instradamento dati
Codificatori e decodificatori	Verifica e conversione di codici
Matrici logiche	Funzioni booleane complesse
Moduli aritmetici	Operazioni numeriche (sommatori, ALU,)
Registri a scorrimento	Conversione serie-parallelo o parallelo-serie. Memorizzazione
Contatori	Controllo e generazione di temporizzazioni

c
o
m
b
i
n
a
t
o
r
i

M
o
d
u
l
i

s
e
q
u
e
n
z
i
a
l
i



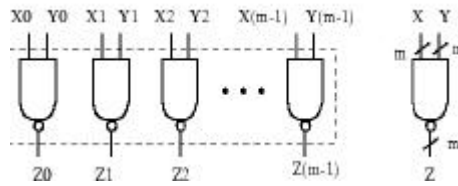
OPERAZIONI SU PAROLE

- **Operazione booleana su parole (vettori di bit):**

$$z(X_0, X_1, \dots, X_{n-1}) =$$

$$z(x_{0,0}, \dots, x_{n-1,0}), \dots, z(x_{0,m-1}, \dots, x_{n-1,m-1}).$$

- **Con $n=2$, z NAND, OR ...**



- **Operazioni con scalari**

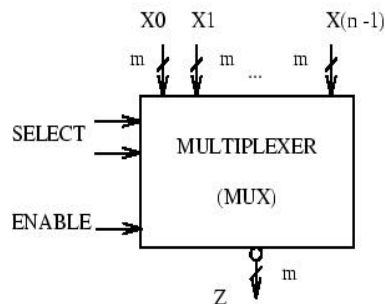
$$yX = (yx_0, yx_1, \dots, yx_{m-1})$$

$$y + X = (y + x_0, y + x_1, \dots, y + x_{m-1})$$



MULTIPLEXER

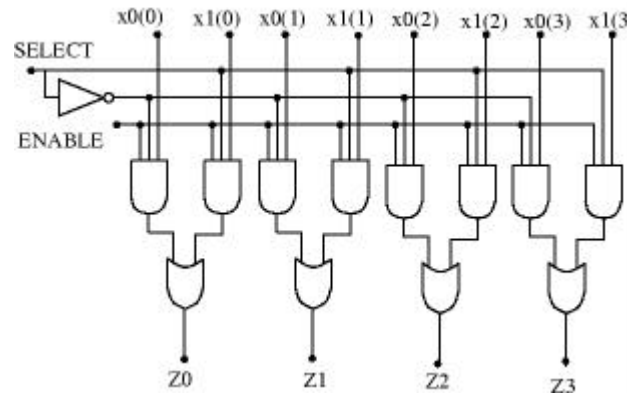
- **Seleziona uno fra più dati di ingresso e li invia verso una destinazione comune.**
- **I dati in ingresso sono vettori di bit.**



Multiplexer a n-ingressi di m-bit



REALIZZAZIONE DI MULTIPLEXER



**Espansione a livello logico di
un multiplexer a 2 ingressi e 4 bit.**



USO DEI MULTIPLEXER

Con soli multiplexer è possibile realizzare una qualunque funzione booleana.

Per una funzione di n variabili è necessario un multiplexer con:

- 2^n ingressi da 1 bit.
- n ingressi per la selezione codificata.

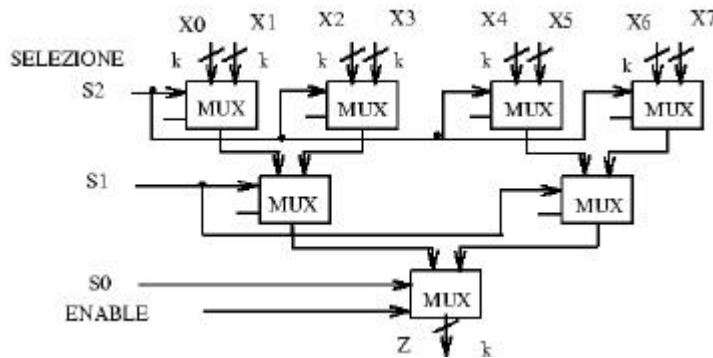
Agli ingressi del multiplexer vengono posti i valori assunti dalla funzione per le varie combinazioni delle variabili di ingresso della funzione stessa.

Le variabili della funzione sono posti agli ingressi di selezione.



USO DEI MULTIPLEXER

E' possibile realizzare k funzioni di 3 variabili con 7 multiplexer a 2 ingressi di k bit.

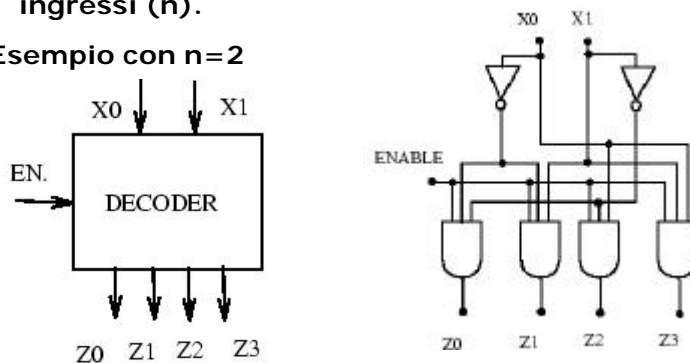


DECODIFICATORE

Un decodificatore o decoder o demultiplexer è un circuito combinatorio che:

- pone ad 1 una e solo una delle 2^n variabili di uscita.
- la variabile è scelta in funzione del valore degli ingressi (n).

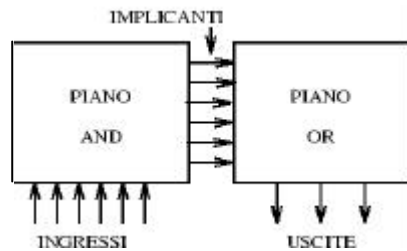
Esempio con $n=2$





MATRICI LOGICHE

Circuiti combinatori a due livelli con struttura topologica ordinata.



Le funzioni logiche specifiche sono programmate a partire da una struttura Hardware preesistente.

Matrici logiche programmabili (PLA) dal costruttore o dall'utilizzatore.

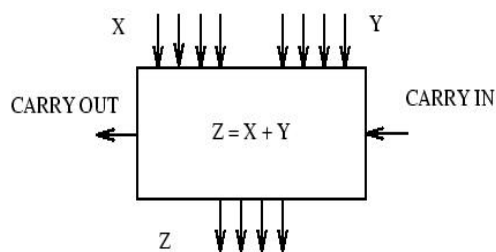


ELEMENTI ARITMETICI

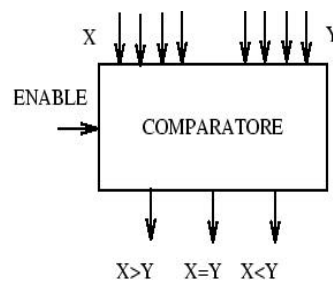
Sono gli elementi combinatori più complessi.

Esempi:

Sommatore parallelo



Comparatore





LINGUAGGI DI DESCRIZIONE DELLO HW

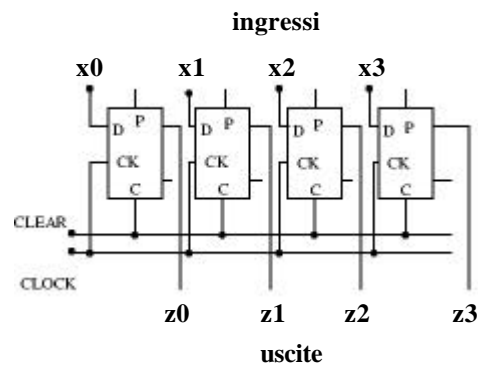
- La descrizione dell'HW mediante schemi non ha prodotto tecniche formali standardizzate di uso generalizzato.
- Si sono invece affermati linguaggi di descrizione dell'HW che derivano da linguaggi di programmazione ad alto livello quali ADA o C.
- I più affermati linguaggi di questo tipo sono:
 - VERILOG - origini industriali, deriva da C.
 - VHDL - VHSIC (Very High Speed IC) Hardware Description Language, deriva da ADA, promosso dal DoD.
- Questi linguaggi sono tipicamente utilizzati nei processi di progettazione che utilizzano tecniche automatiche (CAD).



ELEMENTI DI MEMORIA

- Gli elementi di memoria a livello RTL sono i *registri*.
- I registri sono costituiti da elementi di memoria (FLIP-FLOP) collegati per memorizzare e operare su insiemi di bit.

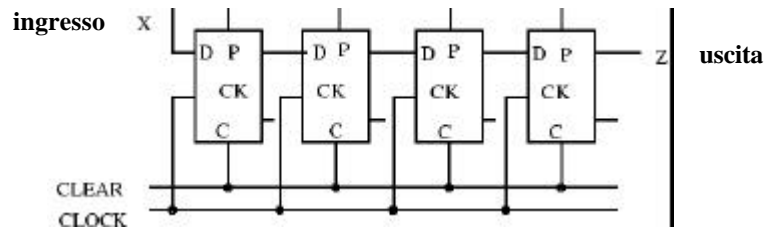
Insieme di Flip-Flop di tipo D che realizza un registro con ingresso e uscita paralleli.





REGISTRI A SCORRIMENTO

Ingresso seriale e uscita seriale.

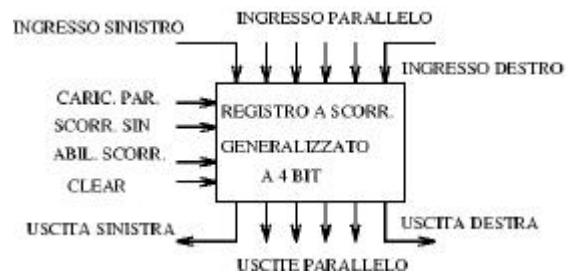


I Flip-Flop sono connessi in modo che i bit possano trasferirsi ordinatamente da un elemento a quello adiacente.

Il trasferimento avviene ad ogni colpo di clock.



REGISTRO GENERALIZZATO

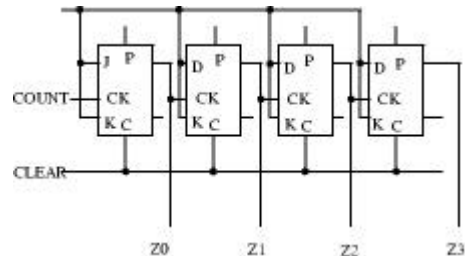


Svolge le seguenti funzioni:

- Ingresso parallelo
- Ingresso seriale
- Uscita parallela
- Uscita seriale
- Abilitazione scorrimento
- Scorrimento a destra
- Scorrimento a sinistra



CONTATORI



Contatore
asincrono

- I contatori sono moduli che hanno come solo ingresso il segnale di clock.
- Lo stato interno del sistema evolve ponendo sulle uscite la sequenza dei codici desiderata.
- Utilizzo: Generatori di sequenze, Divisori di frequenza.



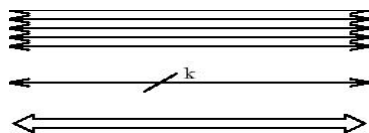
STRUTTURE DI INTERCONNESSIONE

- Le strutture di interconnessione collegano i moduli che devono scambiarsi informazioni.
- In un sistema di calcolo le prestazioni, l'affidabilità e il costo delle interconnessioni hanno un ruolo fondamentale.
 - *Tipo*: la connessione può essere di tipo serie (un solo filo) o parallelo (più connessioni fisiche).
 - *Prestazioni*: si misurano in (M,K)bytes o (M,K)bit al secondo trasferiti.
 - *Affidabilità*: dipende principalmente dal connettore.
 - *Costo*: legato al tipo di connessione e alla standardizzazione.

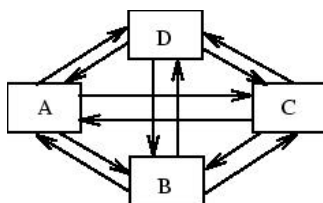


BUS

- Internamente ad un sistema digitale la principale struttura di interconnessione è il *BUS*.
- Il *BUS* è un insieme di conduttori che trasferiscono le informazioni da un elemento sorgente a un elemento destinazione.
 - **BUS dedicato.** Collega in modo esclusivo due moduli.
 - **BUS condiviso.** Collega tra loro più moduli. Collegamenti fra elementi diversi avvengono in tempi diversi (*time multiplexing*).



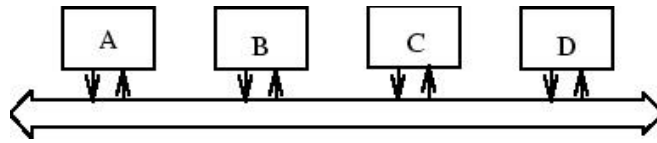
BUS DEDICATI



- La connessione completa di N moduli richiede $N(N-1)$ bus dedicati.
- Il sistema di interconnessione consente di effettuare contemporaneamente più operazioni di trasferimento di dati.



BUS CONDIVISI



- Minori costi, struttura modulare.
- Prestazioni inferiori rispetto ai bus dedicati.

L'uso di una struttura di interconnessione condivisa e standardizzata è stato un aspetto cruciale nella evoluzione dei moderni sistemi di elaborazione.

La realizzazione di sistemi modulari in grado di essere configurati a piacere a partire da elementi disponibili sul mercato consente di ridurre di molto i costi dei sistemi completi.