

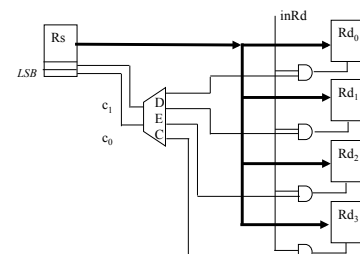
Esempi di Interconnessione tra registri

Prof. Daniele Gorla

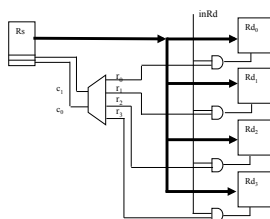
Esempio 1

Sia R_s un registro sorgente e siano Rd_0 , Rd_1 , Rd_2 ed Rd_3 quattro registri destinazione. Si progetti la rete di interconnessione tale che, quando in_Rd vale 1, il contenuto di R_s viene trasferito in Rd_j , dove j è codificato in binario dai due bit meno significativi di R_s . Si dettagli il progetto fino alle porte logiche.

SOLUZIONE:

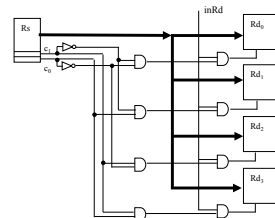


Soluzione (cont.)



c_2	c_1	r_0	r_1	r_2	r_3
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1

Dovendo dettagliare fino a livello di porte logiche, esplicitiamo il decodificatore. La tabella di verità del decodificatore si presenta nel seguente modo, dove r_0 , r_1 , r_2 ed r_3 sono le uscite del decodificatore, usate poi per costruire (in AND con il segnale in_Rd) il segnale di abilitazione dei registri Rd_0 , Rd_1 , Rd_2 ed Rd_3 , rispettivamente:



Esempio 2

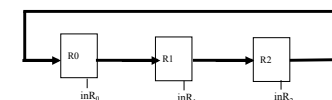
Si progetti un sistema di trasferimento fra i registri R_0 , R_1 ed R_2 tale che:

- R_0 viene portato in R_1 se $R_1 > R_2$;
- R_1 viene portato in R_2 se $R_0 < R_1$;
- R_2 viene portato in R_0 se $R_0 = R_1 \mid R_2$ (dove \mid indica l'OR bit a bit).

Si utilizzino nel progetto moduli combinatori noti (per esempio, ADD, CMP, ...)

SOLUZIONE:

La rete di interconnessione si realizza mettendo insieme tre schemi di trasferimento 1-a-1:

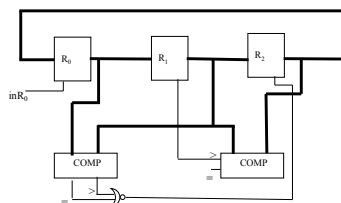


Soluzione (1)



Bisogna poi progettare la parte circuitale che permette di ottenere i valori dei segnali di controllo in R_0 , in R_1 e in R_2 . Le condizioni poste per ottenere i primi due trasferimenti suggeriscono di utilizzare le uscite di due comparatori:

- $R_0 \rightarrow R_1$ se $R_1 > R_2$;
- $R_1 \rightarrow R_2$ se $R_0 < R_1$;
- $R_2 \rightarrow R_0$ se $R_0 = R_1 \mid R_2$

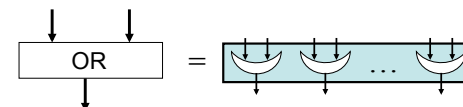


6

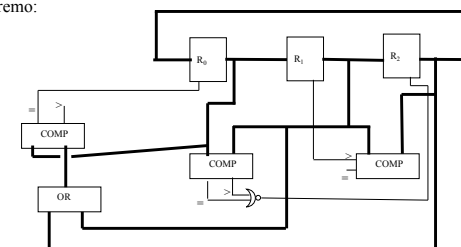
Soluzione (3)



Volendo usare dei moduli combinatori anche per la progettazione di in R_0 , definiamo il circuito OR:



Allora avremo:



6

Esempio 3



Progettare una rete di interconnessione "multi-multi" che consenta di caricare il contenuto di 2 fra N registri $R_1 \dots R_N$ da k bit su 1 fra M dispositivi di elaborazione $E_1 \dots E_M$ a due ingressi (la comunicazione avviene fra 2 su N sorgenti ed 1 su M destinazioni).

Disegnare lo schema a blocchi evidenziando tutti i segnali di controllo necessari per:

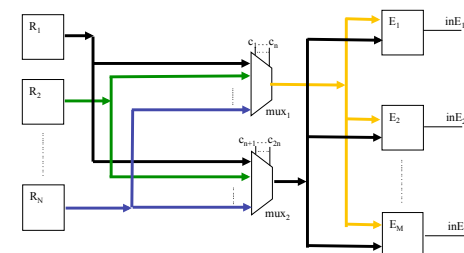
- selezionare 2 fra gli N registri sorgente (ovvero i due operandi);
- convogliare i 2 operandi in ingresso ad uno fra gli M dispositivi di elaborazione.

Disegnare poi lo schema circuitale (quindi con tutti i dettagli fino al livello di FF, porte logiche, numero e ruolo dei segnali di controllo necessari) per il caso di:

- 3 registri sorgente a due bit ($N=3$, $k=2$), con FF di tipo JK;
- 2 dispositivi destinazione ($M=2$), di cui uno sia un sommatore aritmetico e l'altro un circuito logico che esegua l'AND bit a bit fra i due registri sorgente selezionati.

7

Soluzione (1)



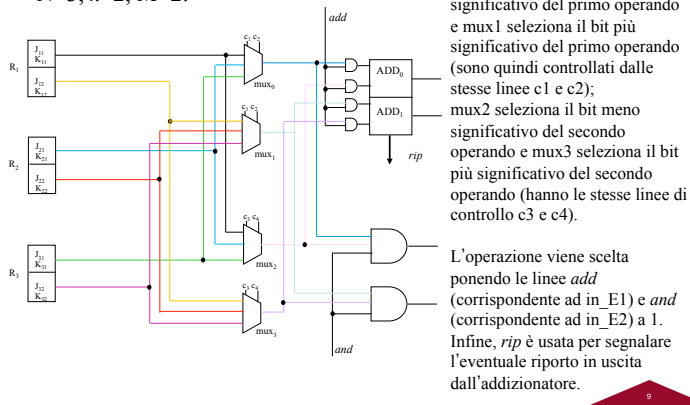
C'è un (insieme di k) MUX per ogni operando da inviare alle unità di elaborazione: MUX1 seleziona il primo operando tra gli N registri sorgente per mezzo dei segnali di controllo $c_1 \dots c_n$; MUX2 seleziona il secondo operando per mezzo dei segnali di controllo $c_{n+1} \dots c_{2n}$, dove, naturalmente, n è la parte intera superiore di $\log_2 N$.

Ogni unità di elaborazione è dotata di un segnale di controllo in E_j per abilitare la lettura e l'elaborazione degli operandi presenti sulle linee di ingresso all'unità j .

8

Soluzione (2)

$N=3, k=2, M=2$:



mux0 seleziona il bit meno significativo del primo operando e mux1 seleziona il bit più significativo del primo operando (sono quindi controllati dalle stesse linee c1 e c2); mux2 seleziona il bit meno significativo del secondo operando e mux3 seleziona il bit più significativo del secondo operando (hanno le stesse linee di controllo c3 e c4).

L'operazione viene scelta ponendo le linee *add* (corrispondente ad in_E1) e *and* (corrispondente ad in_E2) a 1. Infine, *rip* è usata per segnalare l'eventuale riporto in uscita dall'addizionatore.