

ARCHITETTURA DEGLI ELABORATORI I
ESERCITAZIONE 5 - PROGETTAZIONE DI RETI DI
INTERCONNESSIONE FRA REGISTRI
ROBERTO NAVIGLI

1 Reti di Interconnessione fra Registri

In questa esercitazione vedremo i quattro tipi diversi di reti di interconnessione fra registri:

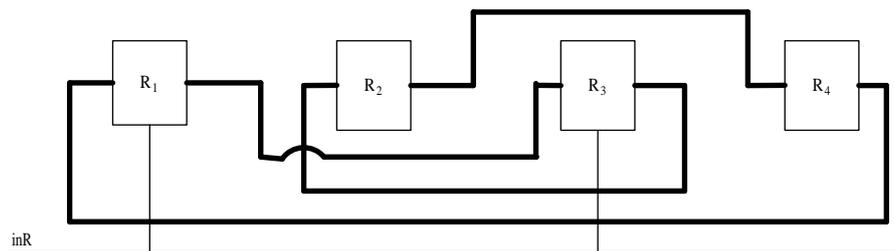
- registro sorgente e destinazione sono fissati (connessione punto-punto);
- registro sorgente variabile e registro destinazione prefissato (molti-a-uno: si usa il multiplexer);
- registro sorgente prefissato e registro destinazione variabile (uno-a-molti: si usa il demultiplexer);
- registri sorgente e destinazione variabili (molti-a-molti: mesh).

1.1 *Dati 4 registri R_1, R_2, R_3, R_4 da 2 bit ciascuno progettare la rete di interconnessione che renda possibili i seguenti trasferimenti:*

- Scrivi R_1 in R_3
- Scrivi R_2 in R_4
- Scrivi R_3 in R_2
- Scrivi R_4 in R_1

Un segnale di controllo inR attiva il trasferimento da R_1 a R_3 e da R_4 a R_1 .

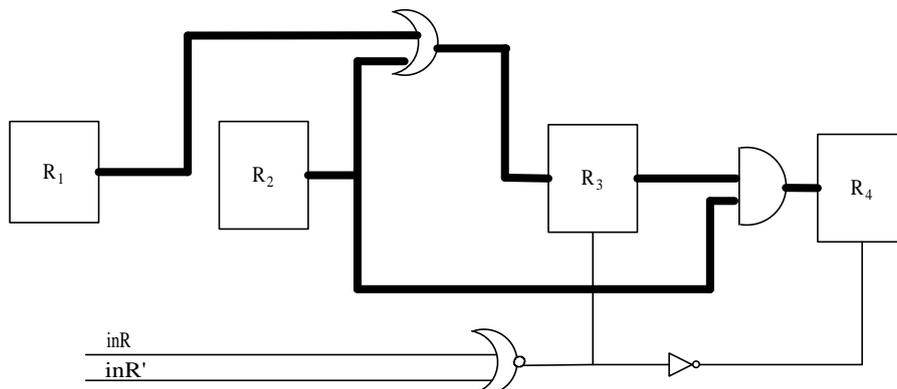
La rete di interconnessione richiesta è del tipo più semplice (punto-punto). E' sufficiente cioè collegare l'uscita di ciascun registro sorgente a quella del rispettivo registro destinazione. Per regolare i trasferimenti verso i registri R_1 e R_3 si utilizza un segnale inR , che viene fornito in ingresso al registro di destinazione come segnale di controllo. La rete è la seguente:



1.2 Dati 4 registri R_1, R_2, R_3, R_4 da 2 bit ciascuno progettare la rete di interconnessione regolata dai segnali di controllo inR e inR' che renda possibili i seguenti trasferimenti:

- se $inR \text{ XOR } inR' = 1$ scrive in R_3 l'OR tra R_1 e R_2
- se $inR \text{ XOR } inR' = 0$ scrive in R_4 l'AND tra R_2 e R_3
- lascia gli altri registri invariati.

La rete di interconnessione richiesta è del tipo punto-punto. La rete di interconnessione è la seguente:



Il circuito di controllo è banale: i due segnali di controllo, inR e inR' , vengono messi in XOR. L'uscita della porta XOR viene posta come segnale di controllo di R_3 , mentre il segnale negato viene fornito in ingresso come segnale di controllo di R_4 .

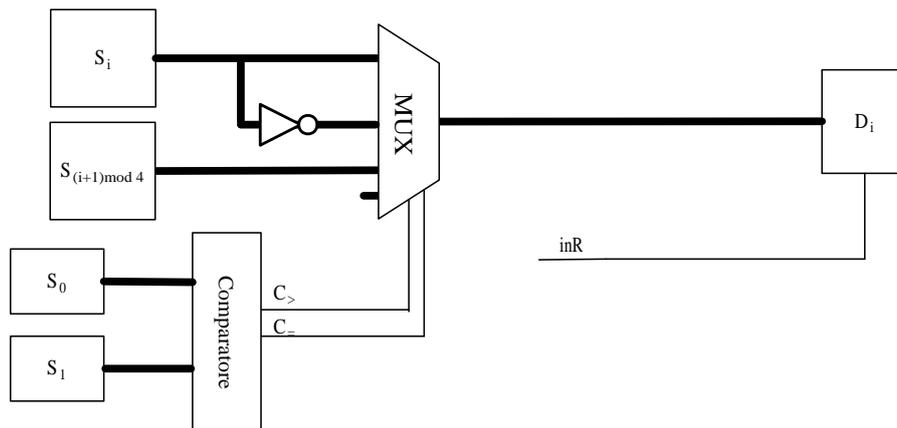
1.3 Sono dati 4 registri sorgente S_0, S_1, S_2 e S_3 e 4 registri destinazione D_0, D_1, D_2 e D_3 da 8 bit ciascuno. Si vogliono realizzare i seguenti trasferimenti:

- se $S_0 < S_1$ allora $D_i = S_i$ ($i = 0, \dots, 3$)
- se $S_0 > S_1$ allora $D_i = S_{(i+1) \bmod 4}$ ($i = 0, \dots, 3$)

- altrimenti $D_i = \overline{S_i}$.

I trasferimenti sono abilitati da un segnale di controllo inR . Si progetti il circuito in grado di realizzare il comportamento descritto (si può utilizzare il modulo predefinito per il confronto tra numeri visto a lezione).

La rete di interconnessione richiesta è del tipo multi-a-uno (multiplexer):



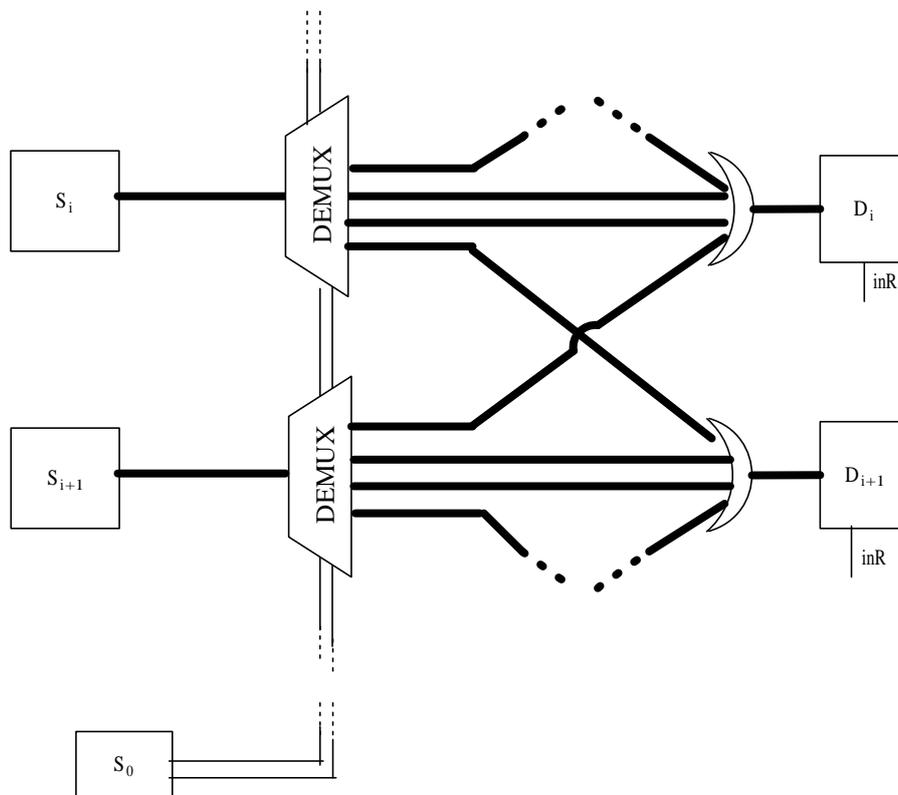
Il circuito di controllo è dato dal risultato del confronto tra S_0 e S_1 (l'esercizio non richiede il progetto del modulo di confronto). Il modulo di confronto restituisce due segnali, $C_>$ e $C_=$, che valgono 1 quando $S_0 > S_1$ e $S_0 = S_1$, rispettivamente. Utilizziamo questi due segnali come controllo del multiplexer. Infatti, i due segnali ci permettono di selezionare il caso appropriato tra i tre previsti dall'esercizio: quando $C_>C_= = 00$ vogliamo realizzare il primo caso, quando $C_>C_= = 01$ vogliamo realizzare il terzo caso e quando $C_>C_= = 10$ vogliamo realizzare il secondo caso (il caso $C_>C_= = 11$ non ci interessa, dato che i due segnali di confronto non possono assumere entrambi valore 1).

1.4 Sono dati 4 registri sorgente S_0, S_1, S_2 e S_3 e 4 registri destinazione D_0, D_1, D_2 e D_3 da 2 bit ciascuno. Si vogliono realizzare i seguenti trasferimenti:

- se $S_0 = 00$ allora $D_{(i-1) \bmod 4} = S_i$ ($i = 0, \dots, 3$)
- se $S_0 = 01$ oppure $S_0 = 10$ allora $D_i = S_i$ ($i = 0, \dots, 3$)
- se $S_0 = 11$ allora $D_{(i+1) \bmod 4} = S_i$ ($i = 0, \dots, 3$).

La rete di interconnessione è del tipo uno-a-molti (poiché possiamo trasferire il contenuto di un registro sorgente in uno tra tre registri destinazione). Per realizzare la rete, dobbiamo fare le seguenti osservazioni: (a) i due bit di S_0 costituiscono a tutti gli effetti i segnali di controllo per

la scelta dei tre casi previsti dall'esercizio; (b) utilizziamo un demultiplexer che abilita la linea in corrispondenza del registro di destinazione (tale scelta dipende, come detto, dai due segnali di controllo forniti da S_0); (c) dato S_i , impostiamo il valore del registro D_i mediante l'OR dei due segnali in uscita dal demultiplexer (casi $S_0 = 01$ e $S_0 = 10$), del segnale proveniente dal caso $S_0 = 11$ del demultiplexer associato a $S_{(i-1) \bmod 4}$ e di quello proveniente dal caso $S_0 = 00$ del demultiplexer associato a $S_{(i+1) \bmod 4}$. Notiamo che l'OR, a un dato istante, sarà sempre pari al valore di uno solo dei quattro collegamenti, poiché ciascuno di essi è associato a un caso di S_0 (se tale caso non è verificato, i bit sulla linea valgono 0). La rete è la seguente:



1.5 Sono dati 4 registri sorgente S_0, S_1, S_2 e S_3 e 4 registri destinazione D_0, D_1, D_2 e D_3 da 2 bit ciascuno. Si vuole trasferire nel registro il cui indice è rappresentato nel registro S_i (ad esempio, se $S_2 = 11$ vogliamo trasferire il contenuto di S_3 in D_2). Il trasferimento viene effettuato solo nei registri D_0 e D_2 se il bit più significativo di S_0 è 1 e solo nei registri D_1 e D_3 altrimenti.

Questo è un caso di rete mesh, in cui il numero di sorgenti e destinazioni è variabile. Dobbiamo quindi utilizzare tanti multiplexer quanti sono i

registri sorgenti. Impostiamo le linee di controllo del multiplexer associato al registro sorgente S_i come le linee di uscita di S_i stesso, poiché il suo contenuto determina l'indice del registro sorgente da trasferire in D_i . Inoltre impostiamo le linee di controllo di D_i pari al secondo bit di S_0 se i è pari e alla sua negazione se i è dispari:

