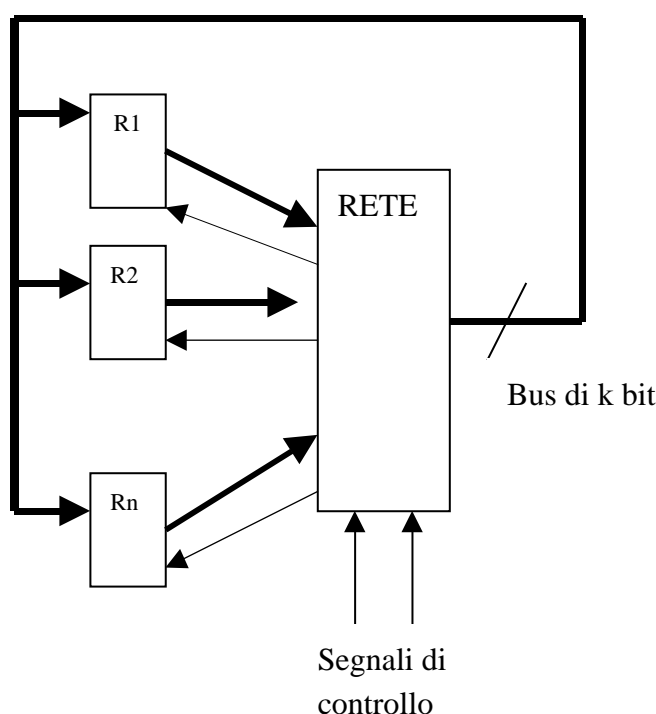


## Reti di interconnessione fra registri



**Figura 1**

Questo schema illustra una rete di connessione fra n registri di k bit ciascuno. La linea in grassetto indica un bus di k linee. Le linee sottili sono segnali di controllo.

I segnali di controllo in uscita dalla rete abilitano un'operazione di Write sui vari registri. Quelli in ingresso selezionano l'opportuna operazione di trasferimento.

La rete in figura consente il trasferimento del contenuto di **un registro sorgente** (Transmitter) su **uno o più registri destinazione** (Receivers).

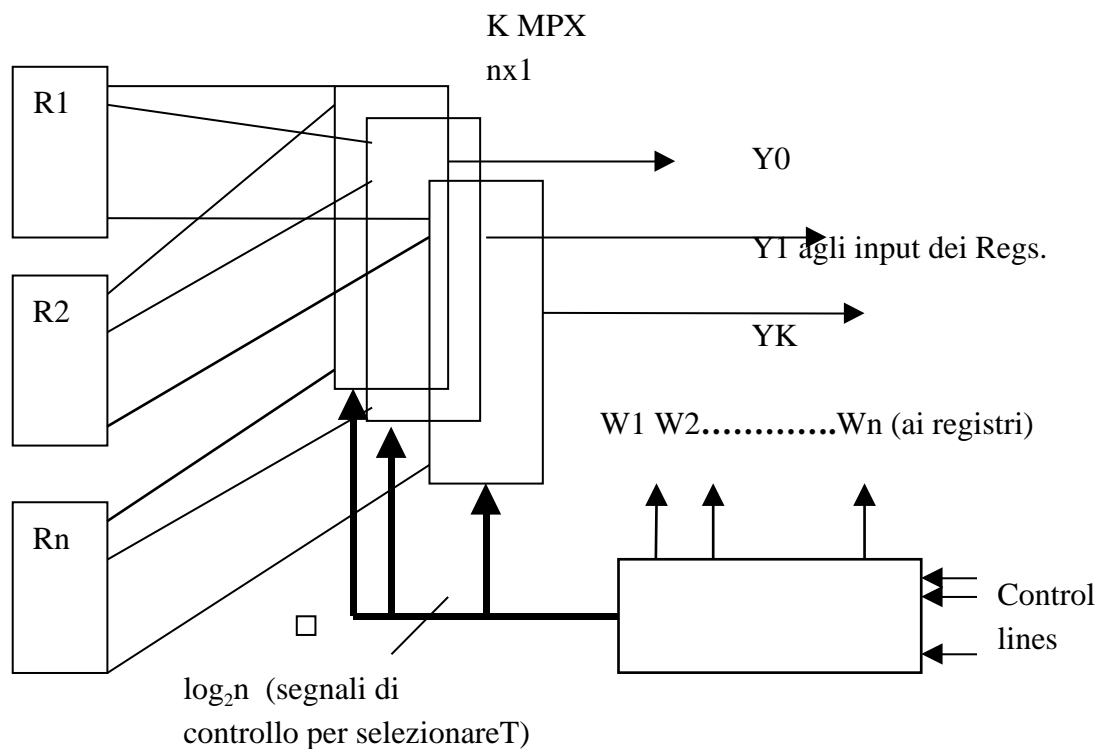
Come mostrato nella figura successiva, la rete include k multiplexres  $n \times 1$ . I k MPX sono tutti controllati dagli stessi  $\log_2 n$  segnali di controllo. Il j-esimo MPX prende in ingresso i bit j-esimi di ciascuno degli n registri. L'input i-esimo ( $i=1..n$ ) del j-esimo MPX è collegato al bit j-esimo ( $j=1..k$ ) del registro  $R_i$ .

Quando i segnali di controllo in ingresso ai k MPX contengono il valore i ( $i=1..n$ ), il contenuto dei k bit del registro  $R_i$  sarà disponibile sulle uscite  $Y_1..Y_k$  dei k MPX.

Il circuito combinatorio in basso a destra della figura 2 è un circuito che riceve in ingresso un codice indicante il tipo di trasferimento da effettuare, e produce in uscita:

- la selezione del registro Trasmittente (ottenuta agendo sui segnali di controllo dei MPX) e
- la selezione di uno o più Ricevitori, ottenuta portando al valore attivo i corrispondenti segnali di Write.

Il progetto di tale circuito dipende dal numero  $r$  tipo di trasferimenti che si vuole sia possibile effettuare.



**Figura 2**

## Esempio 1

Supponiamo di avere 4 registri da 2 bit ciascuno,  $R1..R4$ . Per selezionare uno fra 4 Transmitters occorrono 2 MPX a 4 ingressi. MPX1 riceverà in ingresso i bit  $Q_0^i$  di ciascun registro ( $i=1..4$ ), mentre MPX2 riceverà i bit  $Q_1^i$ .

Supponiamo che si desideri rendere possibili i seguenti trasferimenti:

1.  $R1 \rightarrow R2$
2.  $R4 \rightarrow R3, R1$
3.  $R2 \rightarrow R3$
4.  $R3 \rightarrow R4, R1$

Trattandosi di 4 possibilità, bastano due segnali di controllo,  $c0$  e  $c1$ .

Assegniamo la seguente codifica:

Trasferimento desiderato	codice identificativo
$R1 \rightarrow R2$	$c0=0 \ c1=0$
$R4 \rightarrow R3, R1$	$c0=1 \ c1=1$

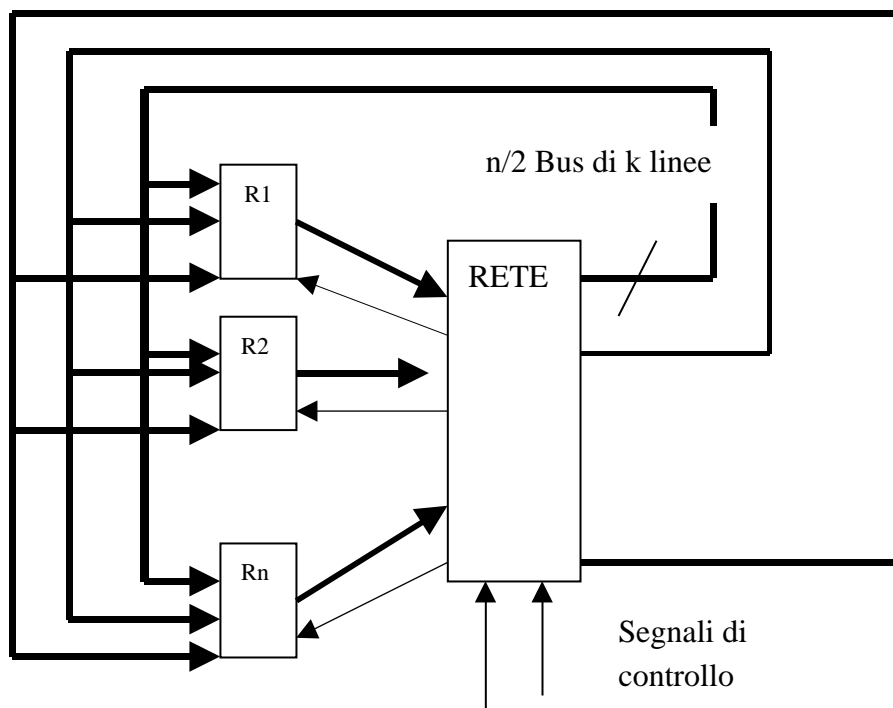
$R2 \rightarrow R3$        $c0=0 \ c1=1$   
 $R3 \rightarrow R4, R1$     $c0=1 \ c1=1$

Possiamo progettare il circuito sulla base della seguente tabella di verità:

Codice trasferimento $c1c0$	Selezione del registro trasmettitore $s1s0$ (ai MPX)	Selezione dei registri riceventi $W4W3W2W1$ (segnali di write)
00	00 (R1)	0010 (write on R2)
01	11 (R4)	0101 (write on R3 and R1)
10	01 (R2)	0100 (write on R3)
11	10 (R3)	1001 (write on R4 and R1)

Il passaggio dalla tabella di verità alle espressioni booleane (ed i circuiti) per  $s0, s1, W1, W2, W3$  e  $W4$  avviene secondo le modalità viste a lezione per il progetto di circuiti combinatori.

## Trasferimento parallelo



**Figura 3**

La rete in figura tre consente fino a  $n/2$  trasferimenti in parallelo, ad esempio:  $R1 \rightarrow R2$ ,  $R3 \rightarrow R4 \dots R_{n-1} \rightarrow Rn$ .

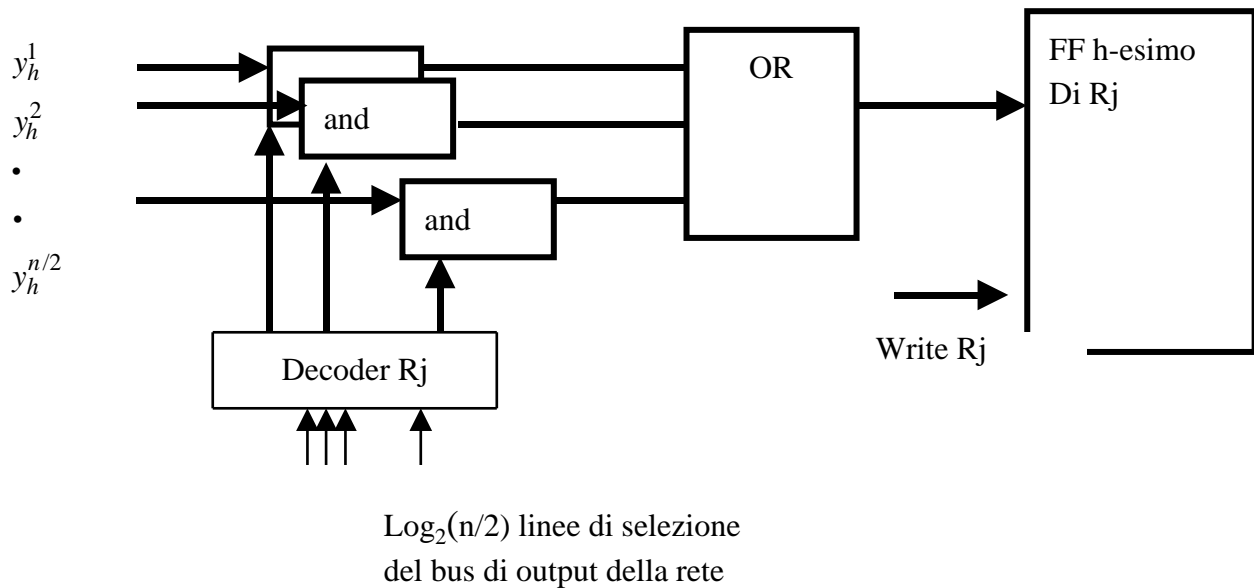
In questo caso in uscita troviamo  $n/2$  bus di  $k$  linee ciascuno (se i registri memorizzano  $k$  bit).

La rete sarà costituita da:

- $n/2$  gruppi  $G_j$  ( $j=1..n/2$ ) di  $k$  MPX  $n \times 1$ .
- Ogni gruppo  $G_j$  è controllato dal medesimo insieme di  $\log_2 n$  linee di controllo  $(s_1^j s_2^j \dots s_{\log_2 n}^j)$ , ma la selezione del trasmitter è indipendente per ogni gruppo  $G_j$ , ovvero in generale  $(s_1^j s_2^j \dots s_{\log_2 n}^j) \quad (s_1^k s_2^k \dots s_{\log_2 n}^k)$  per  $i, j$

Ogni gruppo  $G_j$  è connesso come nella precedente figura 2: il MPX h-esimo del gruppo  $G_j$  riceve in ingresso gli h-esimi bit di tutti gli  $n$  registri.

Ogni FF h-esimo di ogni registro j-esimo avrà un ingresso siffatto:



**Figura 4 Circuiti di selezione dell'input del FF h-esimo del registro j-esimo**

Come si vede il FF h-esimo potrà ricevere un input proveniente da una fra le uscite h-esime  $y_h^k$ ,  $k = 1..n/2$  di ciascuno degli  $n/2$  bus di uscita della rete. Ovviamente, per ogni operazione di

trasferimento, al più un input può essere selezionato, sulla base di segnali di controllo in ingresso al decoder, prodotti dal circuito combinatorio di controllo che, per ogni codifica di ingresso, deve selezionare al più  $n/2$  coppie di Transmitters e Receivers.

In questo caso il progetto del circuito combinatorio è più complesso. Il circuito riceverà in ingresso un certo numero di segnali di controllo  $c_1..c_n$ , che codificano  $2^n$  operazioni di trasferimento diverse. Ogni operazione può coinvolgere fino a  $n/2$  trasferimenti paralleli. Le uscite prodotte dal circuito combinatorio devono selezionare, per ogni singolo trasferimento, l'opportuno trasmettitore e l'opportuno ricevitore.

## Esempio 2

Consideriamo ancora 4 registri da 2 bit ciascuno. La rete avrà due gruppi di MPX ( $n/2=2$ ), ogni gruppo conterrà 2 MPX ( $k=2$ ) avente 4 ingressi ( $n=4$ ). Ogni gruppo di MPX sarà controllato da due linee di controllo,  $\{s_1^1 s_2^1\}, \{s_1^2 s_2^2\}$ . Ci saranno due bus di uscita  $\{y_0^1 y_1^1\}, \{y_0^2 y_1^2\}$ . Infine indichiamo con  $\{r^1 r^2 r^3 r^4\}$ , i segnali di selezione dell'input per ciascuno dei 4 registri. Occorre un solo segnale di controllo per ciascun registro, che ha valore zero se si vuole selezionare dal primo bus, valore 1 se si vuole selezionare dal secondo bus. Nel caso il registro non debba ricevere alcun input, il valore di  $r^i$  non avrà importanza, poiché il segnale di Write per quel registro sarà posto a zero.

Supponiamo si debbano abilitare i seguenti trasferimenti:

- $R1 \rightarrow R2, R3 \rightarrow R4$
- $R1 \rightarrow R4, R2 \rightarrow R3$

Dunque basta 1 bit di controllo, C, per codificare uno di questi due comandi.

Trasferimenti da abilitare	Selezione Transmitters	Selezione input registri	Segnali di scrittura
C	$\{s_1^1 s_2^1\}, \{s_1^2 s_2^2\}$	$\{r^1 r^2 r^3 r^4\}$	W1 W2 W3 W4
C=0 ( $R1 \rightarrow R2, R3 \rightarrow R4$ )	00 10 (R1 e R3)	X 0 X 1	0 1 0 1
C=1 ( $R1 \rightarrow R4, R2 \rightarrow R3$ )	00 01 (R1 e R2)	X X 1 0	0 0 1 1

Naturalmente, un progetto "ad-hoc" della rete di selezione degli input per i vari registri avrebbe semplificato molto il circuito. Infatti, R1 non è mai Receiver, R2 riceve solo da R1 se C=0, R3 solo da R2 se C=1, R4 da R3 se C=0 e da R1 se C=1.