

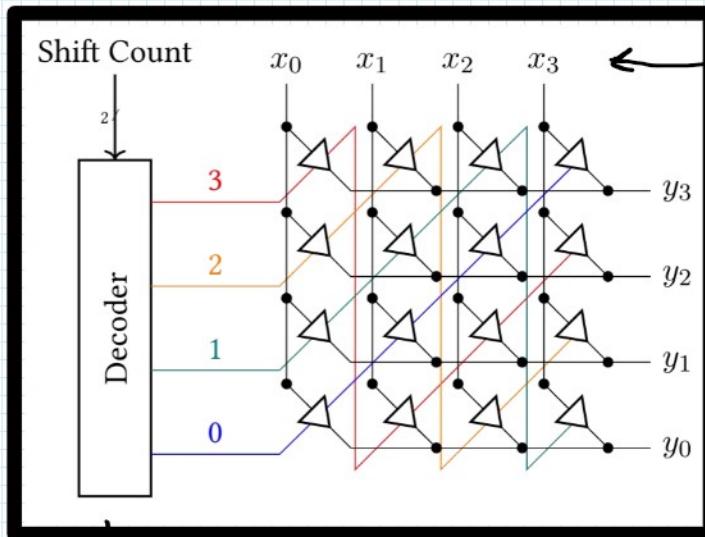
BARREL SHIFTER

martedì 1 novembre 2022 12:48

DATO UN INPUT, SE VUOI FORNIRE IN OUTPUT UNO SHIFT di ZERO POSIZIONI, I SINGOLI BIT IN INGRESSO VERRANNO CORTOCIRCUITATI CON i SINGOLI BIT IN USCITA.

$$SX: 1101 \xrightarrow{1} 1010 \xrightarrow{2} 0100 \xrightarrow{3} 1000$$

SE voglio fare uno shift a sinistra di una posizione i miei interruttori creeranno dei corto circuiti tra le linee. e così per due posizioni.



- questo è un Barrel Shifter a 4 bit.

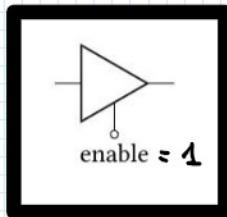
→ quante posizioni di shift.
Di quante posizioni devo effettuare lo shift?

IL DECODER ABILITA UNA SOLA LINEA DI USCITA, IN FUNZIONE DEL VALORE CHE RICEVE.

SE ABILITO LA LINEA NUMERO 0, QUELLA BLU, ATTIVERÀ QUEI A BUFFER THREE STATE.

IL BUFFER THREE STATE

- FA PASSARE il valore di ingresso, al valore di uscita, solo se il segnale di controllo che chiamiamo "ENABLE" vale 1.



COSÌ IL BIT X0 TROVA IL BUFFER THREE-STATE ABILITATO ed esce su y0.
IL BIT X1 TROVA IL TERZO BUFFER THREE-STATE ABILITATO ed esce su y1.
Lo stesso vale per x2 e x3, su y2 e y3.

STO EFFETTUANDO UNA TRASLAZIONE DI ZERO POSIZIONI.

SE ALL'INGRESSO dello SHIF COUNT MANDO 01 come segnale, attiverò la linea 1,

SE ALL'INGRESSO dello SHIFT COUNT MANO 01 come segnale, attiverò la linea 1, quella verde, X0 troverà (a cascata) il terzo buffer three-state abilitato e X0 uscirà su y1.

X1 troverà abilitato il secondo buffer three-state e uscirà su y2.

X2 uscirà su y3.

e X3 uscirà su y0.

HO EFFETTUATO UNO SHIFT A DESTRA DI UNA POSIZIONE, MA È UNA ROTAZIONE A DESTRA.

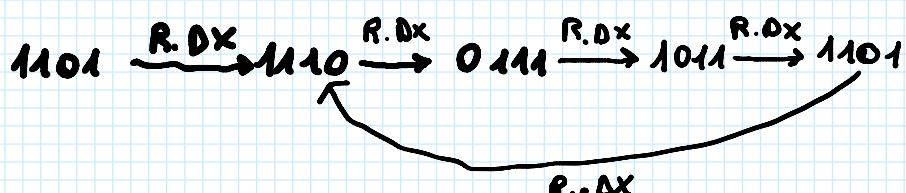


SE ATTIVO LA LINEA 2, $X_0 \rightarrow y_2$, $X_1 \rightarrow y_3$, $X_2 \rightarrow y_0$, $X_3 \rightarrow y_1$. QUINDI È UNA ROTAZIONE A DESTRA DI DUE POSIZIONI.



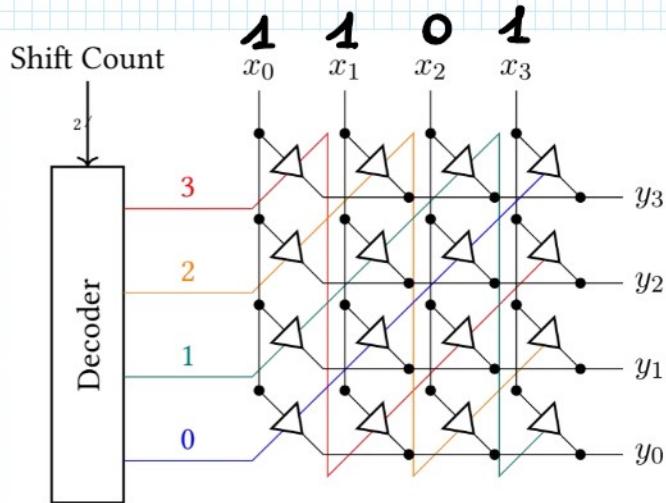
FACCIO GIRARE I BIT.

SE EFFETTUO PIÙ di $\frac{m}{2}$ SPOSTAMENTI STO FAENDO UNO SHIFT A SINISTRA.



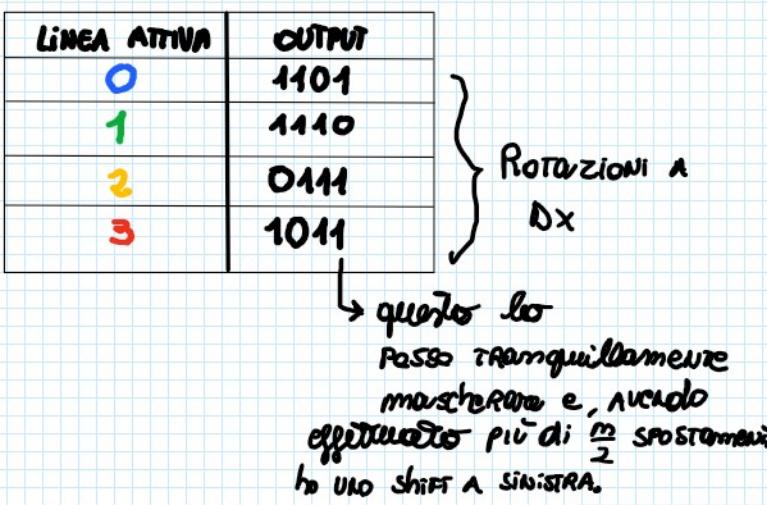
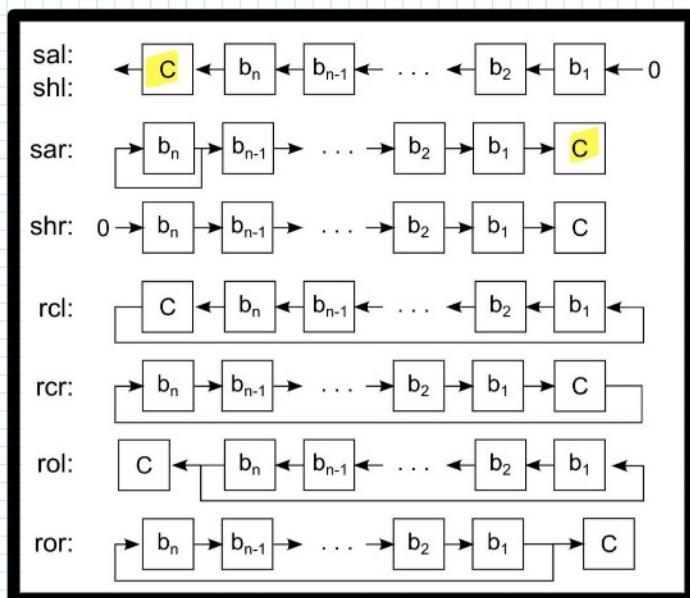
E SE MASCHERO SPECIFICATAMENTE ALCUNI BIT, IN PARTICOLARE IL BIT DI FATTORE, NON IMPLEMENTO UNA ROTAZIONE, MA LO SHIFT COME LO ABBIANO GIÀ VISTO.

ESEMPIO: 1101

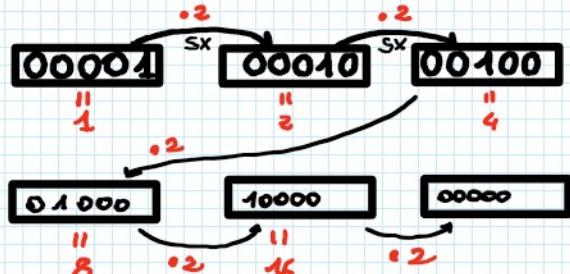


Come esempio, si consideri un barrel shifter che opera su una word di 4 bit ABCD: il circuito può ruotare l'ordine dei bit ABCD effettuando uno o più spostamenti verso destra senza alcuna perdita di bit, in modo da ottenere in uscita DABC, CDAB, BCDA o ABCD. Il barrel shifter viene impiegato in diverse applicazioni, in particolare come componente nei microprocessori (insieme all'ALU).

OPERAZIONI TIPICHE:



→ SPOSTA VERSO SINISTRA TUTTI I BIT. È quello che abbiamo visto in precedenza. Shift a sx.
LA C FINISCE LA SOTTO, UN CARRY.
SE Moltiplico Riferitamente per due:



$$16 \cdot 2 \neq 0 \quad 16 \cdot 2 = 32$$

E quindi ci serve il carry!

SAL = SHIFT ARITMETICO A SINISTRA.

SE HO UNA PAROLA BINARIA, IL MIO PROCESSORE NON HA IDEA DI QUALE RAPPRESENTAZIONE STAVO USANDO.

IMMAGINIAMO DI AVERE RAPPRESENTATO UN NUMERO IN COMPLEMENTO A 2.

SUPPONIAMO SIA UN NUMERO IN COMPLEMENTO A 2 FATTO COSÌ: 1100 (neg)

SE LO SHIFTO A DESTRA VENNE: 0110 (ha cambiato di segno).

NON HO DAVERO DIVISO PER 2.

COME ALLORA OTTO 000 MAINTENUTO 1100 SEGUONO?

NON HO DAVVERO DIVISO PER 2.

~~COME DOVEVO FARLO PER MANTENERE IL SEGNO?~~

ANZICHÉ farlo entrare uno zero qui, avrei dovuto fare entrare un uno;

IL PROCESSORE NON HA IDEA DEL CONTESTO. GLIELO DOBBIANO DIRE NOI. LUI NON PUÒ FARLO DA SOLO.

SE STO UTILIZZANDO UN NUMERO NEGATIVO IN COMPLEMENTO A DUE E LO VOGLIO

DIVIDERE PER DUE, devo effettuare uno shift aritmetico a destra, ossia il bit che entra come bit più significativo è il bit più significativo.

$$\begin{array}{r} 1100 \text{ (Neg)} \xrightarrow[\text{DX ARIT}]{} 1110 \\ 0101 \text{ (Pos)} \xrightarrow[\text{DX ARIT}]{} 0010 \end{array}$$

} Deve prendere il bit più significativo e proponerlo

Lo shift non aritmetico a destra :

SHR

$$1100 \xrightarrow[\substack{\text{VISTO COME} \\ \text{NUMERO} \\ \text{POSITIVO}}]{\text{dx}} 0110 \quad (\text{ho diviso per 2})$$

||
6
||
12

SHR VS SAR, lo dico alla CPU se so utilizzando dati segnati o non segnati, quando voglio di dividere per 2.

IL PROCESSORE PIOTERÀ I DATI IN MAMERA DIVERSE,

FORNIREMO ALLA CPU UN CONTESTO!

LA CARATTERISTICA COMUNE DI TUTTO QUELLO VISTO FINORA È:

Dobbiamo ottimizzare i fili oltre che i circuiti.

SE HO TANTI CIRCUITI, PER RIDURRRE I FILI POSSO PROLUDER AD AVERE UN UNICO CIRCUITO PROGRAMMABILE, CHE IMPLEMENTA TUTTE LE OPERAZIONI DI CUI HO BISOGNO.

QUESTO CIRCUITO È LA **ALU**

