Esercizio4: shift register

L’esercizio richiede la progettazione di uno shift register, mediante un approccio sia comportamentale che strutturale, con le seguenti caratteristiche:

* Modo variabile: lo shift register, sulla base di un segnale in ingresso, deve effettuare shift verso destra o verso sinistra.
* Shift variabile: sulla base di un segnale in ingresso, il registro deve poter variare il numero di posizioni di shift.

Immagine che contiene testo

Descrizione generata automaticamente

Approccio comportamentale:

per la progettazione di tipo comportamentale è stato utilizzato il costrutto if, dato che sia le opzioni di modo che di shift comprendevano solo due possibilità; in un caso più generale, il costrutto case sarebbe più appropriato.

Lo shift register realizzato prevede un ingresso seriale ed un output parallelo, gestiti nel seguente modo:

* Ingresso mode: tale segnale varia la modalità di shift: se pari ad 1, il valore viene inserito da destra, altrimenti da sinistra.
* Ingresso a: tale segnale funge da abilitazione, ovvero il registro effettua uno shift solo se, sul fronte di salita del clock, rileva un’abilitazione pari ad 1.
* Ingresso shift: è il segnale che gestisce il numero di posizioni dello shift: quando è pari a 0, avviene lo shift di una posizione, altrimenti di due.

Immagine che contiene testo

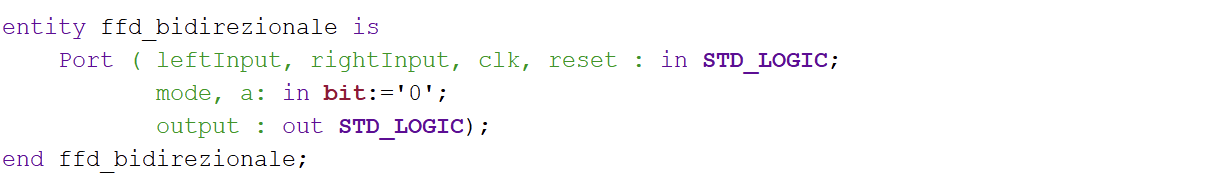
Descrizione generata automaticamente



Approccio strutturale:

per la realizzazione tramite un approccio strutturale, sono stati realizzati dei flip flop D bidirezionali.

Component Ffd\_bidirezionali:



La logica di tale componente è molto semplice:

prevede 2 ingressi (leftInput e RightInput) e, in base al segnale di mode in ingresso, decide quale riportare in uscita. Tale componente è sincrono e lavora solo con abilitazione pari ad 1.

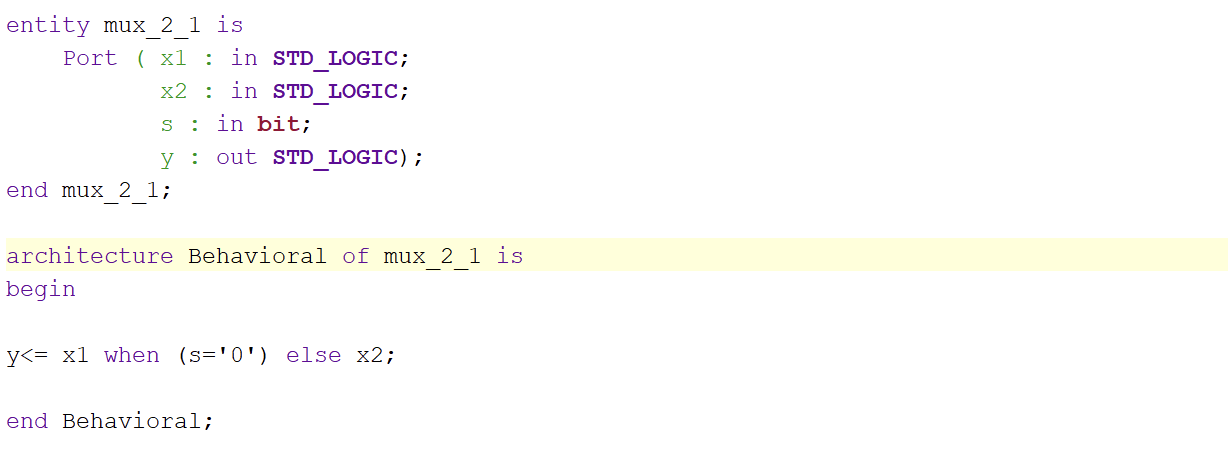
Immagine che contiene testo

Descrizione generata automaticamente

Component mux\_2\_1:

tale componente è necessario per selezionare quale valore deve essere riportato in ingresso ai flipflop, in base allo shift richiesto.

Dato che lo shift può variare al massimo di una posizione, un multiplexer 2:1 è sufficiente; in un caso più generale, si potrebbe ricorrere ad un multiplexer con un numero maggiore di ingressi.



Vediamo ora come tali componenti sono stati combinati:

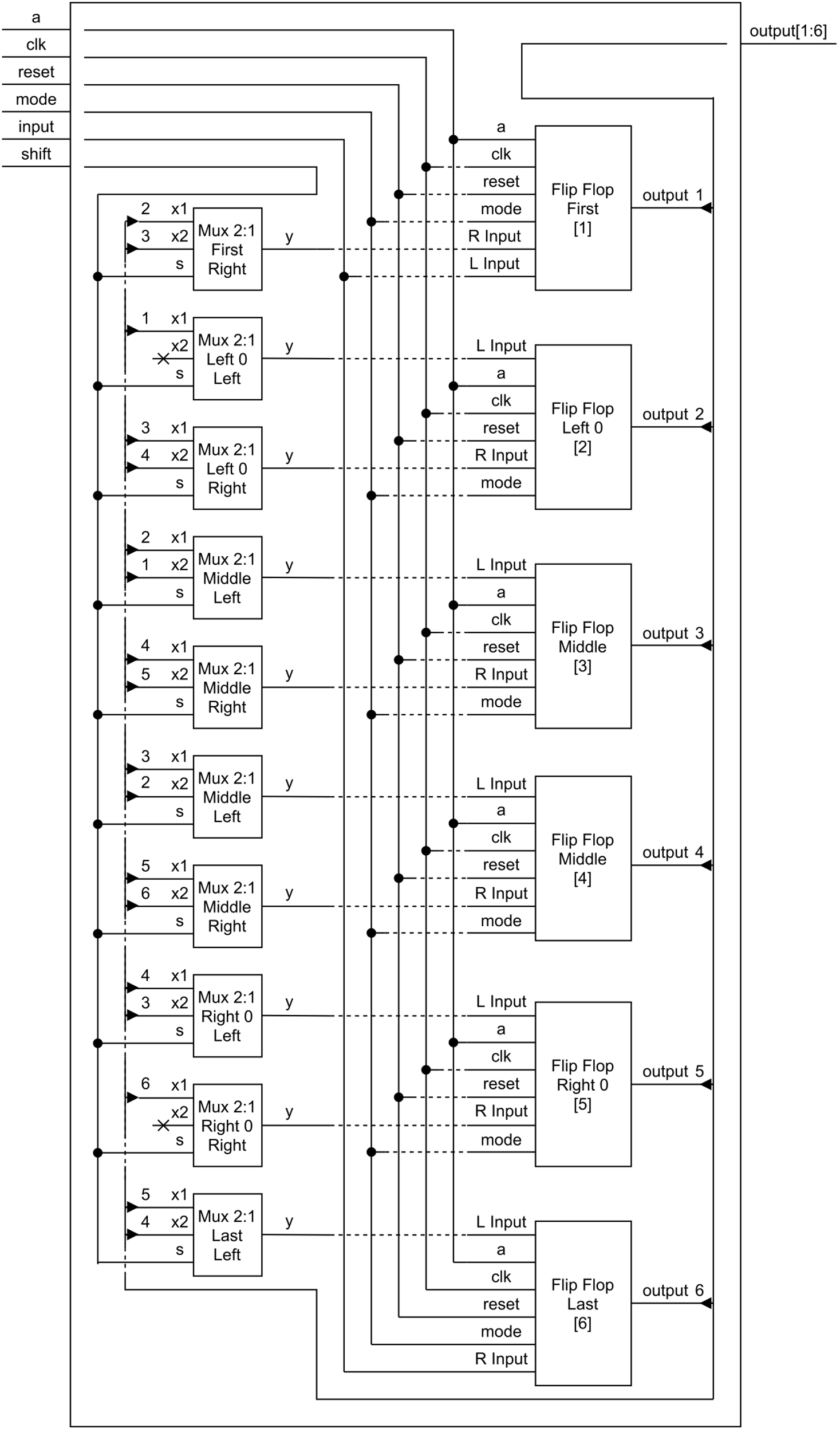
Immagine che contiene testo

Descrizione generata automaticamente

Immagine che contiene testo

Descrizione generata automaticamente

* Primoff: è il primo flip flop da sinistra, ovvero quello di posizione 1. È stato differenziato dagli altri poiché presenterà il multiplexer solo sull’ingresso destro, dato che il sinistro è legato all’input del registro.
* Ultimoff: è il primo flip flop da destra, ovvero quello di posizione N. È stato differenziato dagli altri poiché presenterà il multiplexer solo sull’ingresso sinistro, dato che il destro è legato all’input del registro.
* Ff\_with0: sono i flipflop che, nel multiplexer per la selezione, presentano degli 0, in quanto non hanno abbastanza flipflop che li precedono. Es: nel flipflop di posizione 2, per l’ingresso sinistro, la scelta ricadrà tra l’uscita del flipflop di posizione 1 ed uno 0, dato che non esiste un flipflop di posizione 0.
* Ffintermedi: sono quei flipflop che presentano multiplexer si per l’ingresso di destra che di sinistra, ed entrambi prendono gli ingressi da flipflop precedenti.



Simulazione:

verrà ora presentata una simulazione del funzionamento della macchina, verificando tutte le combinazioni dei segnali mode e shift. I risultati della simulazione sono analoghi per entrambe le architecture.

