

MAPD: FIR filter

Studente: Agostini Federico

Gruppo di lavoro: Agostini Federico
Bottaro Federico
Pompeo Gianmarco

1 Premessa

La scrittura del codice è stata frutto di un lavoro di gruppo; i singoli membri hanno poi eseguito l'analisi per i diversi tipi di filtri proposti. Dal momento che il codice utilizzato è lo stesso, si è deciso di proporre nella relazione tutti i risultati ottenuti.

2 Schema del filtro

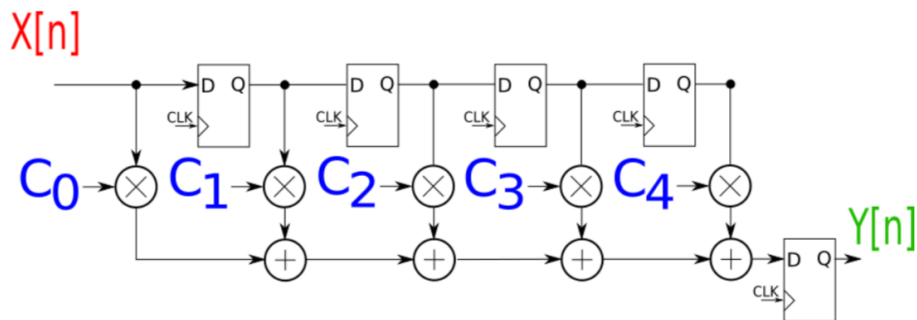


Figura 1: *Rappresentazione schematica del filtro*

3 Schema logico

Si presenta in Figura 2 lo schema generato dalla sintesi del codice. Si possono distinguere i 5 flip flop (in azzurro) e i blocchi algebrici (in giallo). Lo schema è stato realizzato supponendo che i segnali fossero a 16 bit.

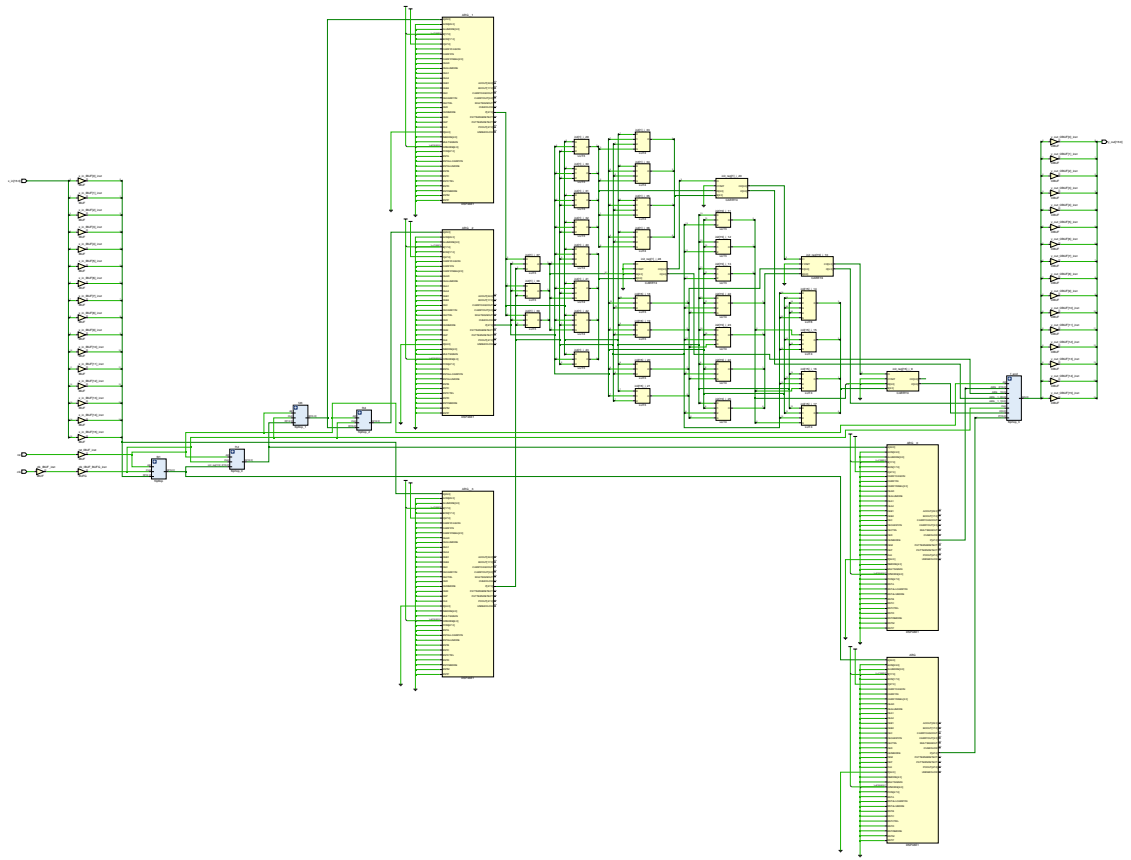


Figura 2: *Rappresentazione del circuito logico generato da Vivado*

4 Filtro scelto

Nella Tabella 1 sono riportati i valori dei coefficienti per i vari tipi di filtro ricavati utilizzando l'apposito script fornito. Per ricavare la frequenza di cut off bisogna moltiplicare il valore presente in tabella per $f_s/2$, dove $f_s = 11025$ Hz nel nostro caso.

Tabella 1: *Tabella dei Coefficienti*

	Passa Basso	Passa Alto	Elimina Banda	Passa Banda
C_0	0.1933	0.1933	0.3258	- 0.3258
C_1	0.2033	- 0.2033	0	0
C_2	0.2066	0.2066	0.3483	0.3483
C_3	0.2033	- 0.2033	0	0
C_4	0.1933	0.1933	0.3258	- 0.3258
Cut Off	0.1	0.9	0.1 - 0.9	0.4 - 0.6

Per testare il codice si sono utilizzati i valori dei coefficienti suggeriti ($C_0 = 1$, $C_1 = 2$, $C_2 = 3$, $C_3 = 4$, $C_4 = 5$), mentre il segnale in ingresso è stato preso costante ($x_{in} = 1$). Una volta appurata la correttezza del valore dell'output ($y_{out} = 15$), si è quindi proceduto all'assegnazione dei coefficienti elencati in Tabella 1.

5 Conversione dei coefficienti in VHDL

Per gestire numeri compresi tra 0 e 1 in VHDL, è necessario eseguire prima moltiplicarli per una costante k in modo da renderli maggiori di 1, e poi, finite le operazioni, riscalarli i risultati per lo stesso fattore k . Per realizzare ciò, si è scelto di shiftare a sinistra i valori binari dei C_i di 10 posizioni, ed infine utilizzare la funzione SHIF_RIGHT per riottenere i risultati corretti.

6 Presentazione output

Si presentano nella seguente sezione gli output ottenuti per i diversi filtri. Il segnale di ingresso è stato preso costante e pari a $x_{in} = 5000$, valore maggiore di 2^{10} come suggerito. Per tali simulazioni si è dovuto ricorrere a segnali a 32 bit in quanto i valori ottenuti eccedevano il valore massimo raggiungibile con 16 bit (pari a $2^{15} - 1$).

6.1 Filtro passa basso

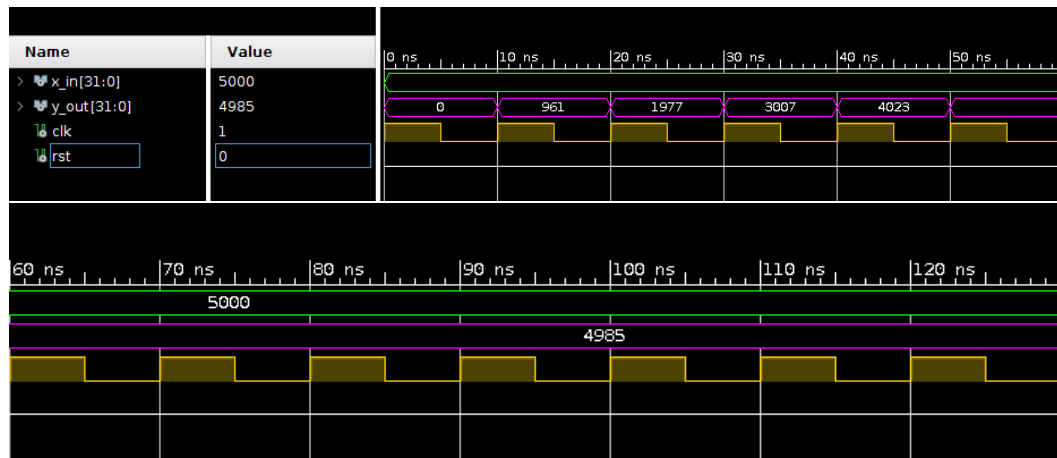


Figura 3: Risultato del testbench per il filtro passa basso

6.2 Filtro passa alto

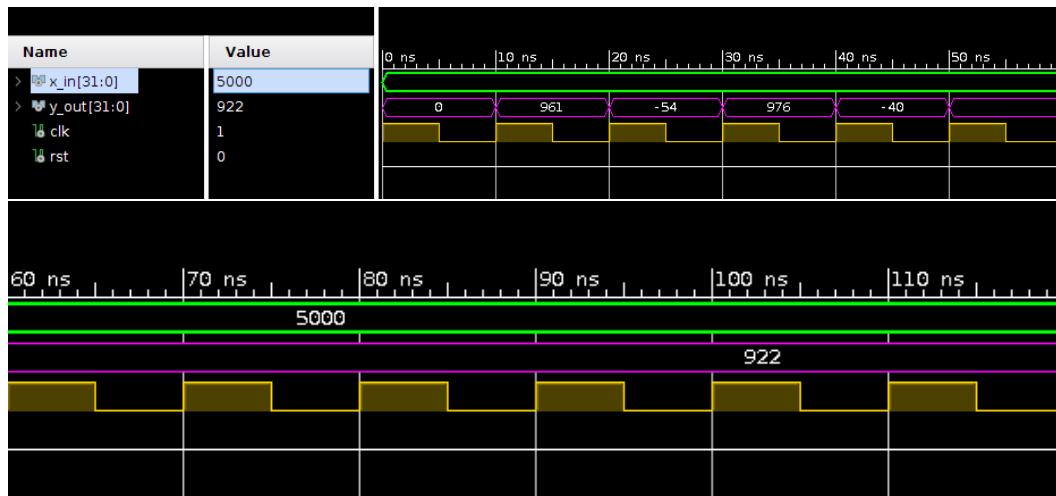


Figura 4: Risultato del testbench per il filtro passa alto

6.3 Filtro elimina banda

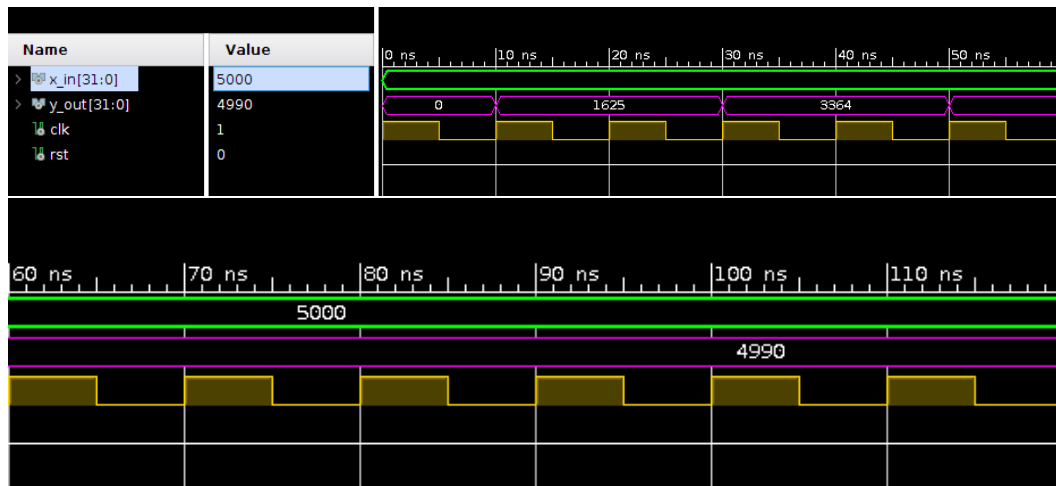


Figura 5: Risultato del testbench per il filtro elimina banda

6.4 Filtro passa banda

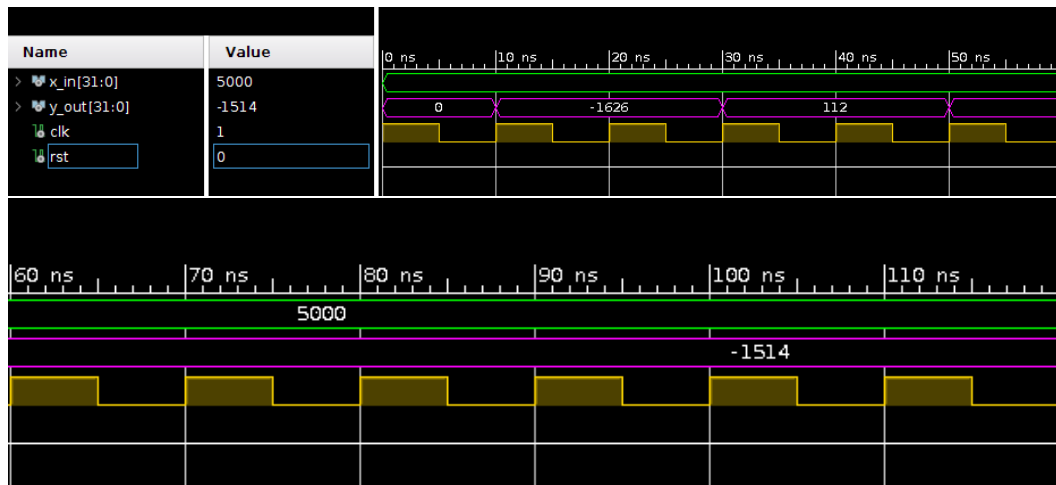


Figura 6: Risultato del testbench per il filtro passa banda

6.5 Discussione risultati

Tutti i risultati ottenuti sono stati confrontati con i valori attesi, ricavati tramite la:

$$y_{n+1} = \sum_{i=1}^N x[n-1]C_i$$
Tale confronto ha confermato la previsione per ogni filtro analizzato.