

SHELVING DESIGN

FILTRO A(z)

Il filtro A(z) è un filtro del secondo ordine di tipo IIR, riceve quindi 5 coefficienti in ingresso (a_1, a_2, b_0, b_1, b_2) che definiscono la funzione di trasferimento e quindi il suo comportamento.

Questo filtro è già stato realizzato e lo stesso vale per la CU, vanno solo fatte alcune modifiche minime per togliere alcuni componenti non necessari.

REGISTRI RIN/ROUT

Il registro RIN serve per memorizzare il dato all'interno dell'audio processor quando è valido (il segnale VIN serve proprio per comunicare alla CU quando un dato è disponibile), mentre il registro ROUT fornisce al blocco di trasmissione della UART il campione filtrato da trasmettere (il segnale VOUT indica alla UART che un dato è disponibile).

REGISTRI RSW e MULTIPLEXER MUX0

Il registro RSW salva il valore dei due switch utilizzati per:

- definire il comportamento filtrante (LF/HF)
- mandare il dato in loop-back per test

Il multiplexer MUX0 ha come selettore proprio il valore di uno dei due switch, in particolare dello switch che determina il loop-back o l'esecuzione del filtro.

MEMORIZZAZIONE DEI COEFFICIENTI

I coefficienti che determinano il funzionamento del filtro ($a_1, a_2, b_0, b_1, b_2, H_0/2$) vengono salvati all'interno dell'audio processor tramite opportuni registri prima che il filtro inizi ad operare, ci sarà uno stato apposito (INIT) nella CU che permette di inizializzare il filtro con i coefficienti subito dopo lo stato di RESET, ma prima che il sistema diventi operativo.

PARALLELISMO INTERNO

I dati in ingresso sono in CA2 su 8 bit, da intendersi nel range [-0.5;0.5], estendendo ciascun dato di 4 bit le operazioni vengono fatte con dati su 12 bit.

Estendere di 4 bit i dati in arrivo significa anche estendere il range rappresentabile di un fattore 2^4 , di conseguenza è possibile rappresentare dati senza overflow su un intervallo pari a [-8;8].

SATURAZIONE E SCALAMENTO IN USCITA

Il blocco di saturazione satura il campione filtrato se al di fuori del range [-4;4]; questa è una scelta data da due fattori cioè i bit da scalare e la perdita in risoluzione, infatti:

- se in uscita si scala di tre bit verso destra il campione saturato (divido per 8), ritorno nel range rappresentabile su 8 bit cioè [-0.5;0.5]: per ottenere nuovamente il dato corretto durante la simulazione, è sufficiente scalare verso sinistra di 3 bit (moltiplico per 8) per ottenere nuovamente il dato iniziale.

- lo scalamento verso destra si traduce in una risoluzione minore (scalando di tre bit perdo il valore dei tre LSB, anche se durante la simulazione riscalo verso sinistra), tuttavia la precisione che si ottiene a fronte dello scalamento rimane comunque ragionevole (pari a 1/32).

ALLINEAMENTO DEGLI OPERATORI

Per i sommatori non è necessaria alcuna operazione di allineamento degli addendi.

Per i moltiplicatori, dato che i due operandi sono su 12 bit (con i primi 8 dopo la virgola), l'uscita sarà su 24 bit di cui i primi 16 dopo la virgola: questi 16 bit non servono tutti ma solo i primi 8 dopo la virgola, gli altri 8 si possono eliminare scalando verso destra di 8 posizioni, e poi mandare in uscita i 12 LSB del prodotto scalato.