Laboratorio de DSP y FPGA

Trabajo Práctico N° 2

Grupo 2

KAMMANN, Lucas Agustín

FARALL, Facundo David

DAVIDOV, Gonzalo Joaquín

TROZZO, Nicolás Rafael

# Ejercicio 1

Item b

Utilizando la herramienta FDATool de Matlab se diseña un filtro FIR pasa banda de orden 20 que cumpla la siguiente plantilla para una frecuencia de muestreo fs=48kHz y se obtienen los correspondientes coeficientes.Graphical user interface, diagram

Description automatically generated with medium confidence

Al implementar el filtro en el DSP se obtuvo la siguiente respuesta en frecuencia:Graphical user interface

Description automatically generated

En donde como se puede ver la amplitud de la banda de paso está por debajo de los 0dB debido a alguna atenuación presente en el circuito. Para solucionar esto y poder ver los resultados con mayor claridad se multiplica por un factor de compensación de 2,2 para que la banda de paso esté por encima de los 0dB.A picture containing diagram

Description automatically generated

Finalmente, con el filtro ya diseñado y compensado, se procede a medir el tiempo de procesamiento de la interrupción resultando en tint=324.39ns.A screenshot of a computer

Description automatically generated with medium confidence

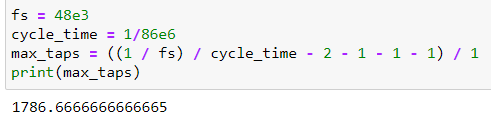
Item c

Se busca diseñar un filtro FIR del tipo sinc inverso pasa altos del mayor orden posible teniendo en cuenta la frecuencia de muestreo y la cantidad de memoria a utilizar.

Se usa una fs = 48kHz. Sabiendo que con el clock a 86MHz, y que las instrucciones que corren en la rutina de interrupción son:Text

Description automatically generated

Se obtuvo que el orden máximo implementable en la DSP es 1786 utilizando el 100% del duty cycle, teniendo en cuenta cuántos ciclos de máquina toma cada instrucción. Se decidió dejar un margen y realizar un filtro de orden 1720 que utiliza el 96%.



Orden máximo debido a fs.

Graphical user interface, diagram

Description automatically generated with medium confidence

Duty cycle con filtro de orden 1720.

Teniendo en cuenta el mapa de memoria a continuación, y sabiendo que los coeficientes en el programa actual comienzan a cargarse desde la posición Y:0, se tiene espacio de memoria interna (con la cual es posible el parallel move) hasta Y:$00FFFF, es decir aproximadamente 65.500 palabras de 24 bits de los cuales solo utilizaremos 1720 espacios en la memoria Y, y luego la misma cantidad de memoria utilizada para los valores n-i (i=0,…,1719) que utiliza el filtro en el espacio de memoria X.

Table

Description automatically generated

Con estos valores en mente se diseñó nuevamente utilizando FDATool en MatLab el siguiente filtro:

A picture containing text, screenshot, indoor

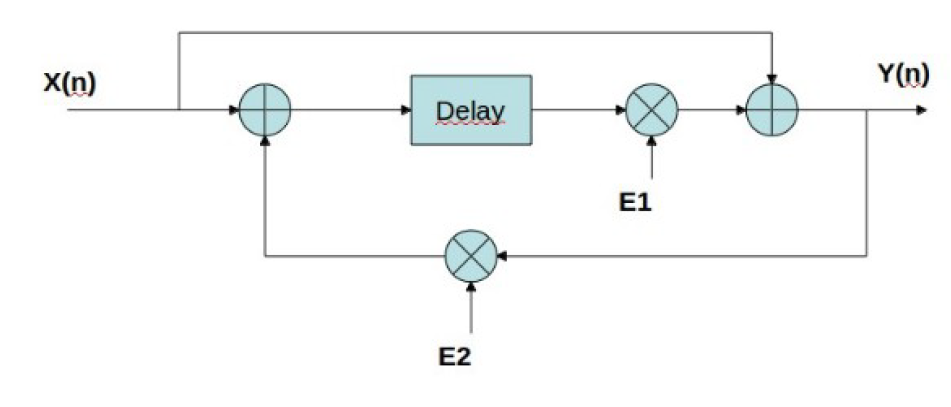
Description automatically generated

A continuación, se puede observar el funcionamiento y la respuesta en frecuencia del mismo al implementarlo en la DSP. A picture containing text, computer

Description automatically generated

# Ejercicio 2

Se realizó un reverberador que cumple el esquema de la siguiente imagen:



En donde E1 representa la atenuación de los ecos reflejados y E2 una reflexión de estos ecos.

El programa consiste en implementar la ecuación que se obtiene del diagrama en bloques:

Ecuación 1: Cálculo de la salida del reverberador

Definiendo:

Ecuación 2

El bloque de retardo temporal de la señal W es implementado con un buffer en memoria, almacenando las muestras [W(n-1), …, W(n-N)]. El buffer es configurado como un buffer circular al cual se accede por un registro R que siempre apunta al final del mismo, es decir, a la muestra más antigua.

Para un instante , en primer lugar se toma la muestra del buffer y se calcula la salida mediante la Ecuación 1. Luego se computa el valor de mediante la Ecuación 2 para ser guardado en el buffer, valor que será utilizado N muestras más adelante.

Vale mencionar que fue necesario agregar instrucciones NOP para evitar problemas de dependencias entre instrucciones dentro del pipeline a la hora de leer y cargar en el acumulador A.

Con respecto al funcionamiento del algoritmo de reverberación, se obtiene el resultado esperado. Variando los coeficientes, que representan \_\_\_\_, se puede obtener más o menos reverberación. Además, se debe ajustar el valor del delay N para que la reverberación suene real. Al variar este valor se tiene la sensación de estar cambiando el tamaño de la habitación reverberante.

Para lograr obtener una buena percepción de la reverberación, empíricamente se determinó un delay de 4096 taps y los valores E1=0.7 y E2=0.4.

Se corrió el reverberador en el DSP y se midió la respuesta impulsiva del mismo obteniéndose así el siguiente gráfico:A screenshot of a computer

Description automatically generated with medium confidence

Se puede observar como a medida que pasa el tiempo la amplitud va decayendo, logrando el conocido perfil de retardos de la reverberación.