Assignment-3

Fast Convolution

To perform Fast Convolution we define the class CFastConv, the members and functions within which are given below:

**Public Members:**

**The constructor destructor pair:**

CFastConv(void);

virtual ~CFastConv(void);

**Initializer:**

Error\_t init (float \*pfImpulseResponse, int iLengthOfIr, int iBlockLength = 8192);

The init function initializes the class with the impulse response and the block length, with the parameters:

pfImpulseResponse -impulse response samples (mono only)

param iLengthOfIr -length of impulse response

param iBlockLength -processing block size

**Create and Destroy Instance** :

static Error\_t createInstance (CFastConv\*& pCFastConv);

static Error\_t destroyInstance (CFastConv\*& pCFastConv);

Creates and destroys respectively an instance of class CFastConv.

**Process and ProcessTimeDomain:**

process (float \*pfInputBuffer, float \*pfOutputBuffer, int iLengthOfBuffers ); processTimeDomain (float \*pfInputBuffer, float \*pfOutputBuffer, int iLengthOfBuffers );

This accepts the following paramaters:

parampfInputBuffer (mono)

param pfOutputBuffer (mono)

param iLengthOfBuffers can be anything from 1 sample to 10000000 samples.

This is the process function wherein the fast convolution is performed.

Also , the processTimeDomain is the function wherein blocked Fast Convolution is performed in Time Domain.This is commented in the Class definition, and can be used if needed by uncommenting.

**Flush Buffer:**

Error\_t flushBuffer(float \*pfOutputBuffer, int iLengthOfBuffer);

This is the function in which the values that are out of range of the length of the input signal (also knnown as the Tail) is stored in the given output Buffer.The length of this buffer has atleast the length of the Impulse response.

**Private Members:**

int m\_iLengthOfIr- Length of Impulse Response

float \*m\_pfImpulseResponse- The impulse response

int m\_iBlockLength- Block Length

**Ring Buffers**

CRingBuffer<float> \*m\_pCRingBuffCurr;

CRingBuffer<float> \*m\_pCRingBuffPrev;

To store the current and previous values within the process.

int m\_iNumIrBlcks- No of IR blocks

CFft \*m\_pCFFT- FFT Object

float \*m\_pfIRfft- value of the FFT of the Impulse Response

int m\_iNxtPow2BlkLen- the next power of two to the block length.

**IMPLEMENTATION NOTE:**

It is seen that the Time Domain process function takes more time to compute as compared to the FFT based process function.