/\*

\* main.c

\*/

#include <stdio.h>

#include "usbstk5515.h"

#define IR\_length //order of the filter+1

#define In\_length //length of input signal

#define tmp\_l IR\_length+In\_length-1

Int16 x[tmp\_l];//Input-1: Input signal; actual length is In\_length but other values will be 0;basically we are padding zeroes

Int16 array[tmp\_l];

void linearbuff();

Int16 \*inPtr;

Int16 \*outPtr;

Int16 \*coeff;

Int16 coefs[IR\_length];//Input-2: Coefs signal(coefs of filter)

Int16 input;

Int16 output;

static Int16 buffer[IR\_length];

static void dataInput(); // dummy function to be used with ProbePoint

static void dataCoefs();

void linearbuff(void) // Commented by ketan

{

//int i;

//long int accumulator=0;

int n,k;

long int accumulator=0;

//static Int16 buffer[In\_length]={0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0};

// write convolution code here;

for (n = 0; n < tmp\_l; n++)

{

array[n] = 0;

for (k = 0; k < IR\_length; k++)

{

// To right shift the impulse

if ((n - k) >= 0

&& (n - k) < In\_length)

{

// Main calculation

array[n] = array[n] + x[k] \* coefs[n - k];

}

//printf("%d\t", y[n]);

}

}

//\*/

/\*

write convolution code here;

\*/

(\*outPtr)=(Int16)accumulator;

}

main(void)

{

int j=0,k=0;

for (k=0;k<IR\_length;k++)

{

buffer[k]=0;

}

for (k=0;k<tmp\_l;k++)

{

x[k]=0;

}

dataInput();

dataCoefs();

coeff = &coefs[0]; //coefs is coefficient of impulse response defined in fdacoefs\_int.h

inPtr = &input; //inPtr is a globally declared pointer to a Int16

outPtr = &output;

// SYS\_EXBUSSEL = 0x6100;

// USBSTK5515\_init( );

while(j<tmp\_l) // loop

{

/\* Read input data and also coefficients using a probe-point connected to a host file. \*/

//printf("x=%d\n",x[j]);

input=x[j];

//write your code for function call;

linearbuff();

array[j]=output;

j=j+1;

}

return 0;

}

/\*

\* FUNCTION: Read input signal and write processed output signal

\* using ProbePoints

\* PARAMETERS: none.

\* RETURN VALUE: none.

\*/

static void dataInput()

{

// do data I/O

return;

}

static void dataCoefs()

{

// do data I/O

return;

}