arm Education

L.EEC025 - Fundamentals of Signal Processing (FunSP)

2023/2024 – 1st semester

Week07, 23 Oct 2023

Objectives:

-measuring the frequency response of comb filters running in real-time on the STM32F746G Discovery board:

- **an FIR comb filter**
- **an IIR comb filter**

DSP Education Kit

LAB 6

Measuring the frequency response of an FIR and IIR comb filter

Issue 1.0

Contents

1 Introduction 1

1.1 Lab overview 1

2 Requirements 1

3 The Comb Filter 1

4 Preliminary (mandatory) analysis before the current lab 3

5 Observation of Frequency Response Using a Sinusoidal Input Signal 4

5.1 Setting-up for this lab experiment 6

5.2 Sketching the frequency response of comb filters 7

5.3 Testing another filter (extra, only if time permits) 9

6 Conclusions 10

7 Additional References 10

1 Introduction

1.1 Lab overview

The examples in this lab motivate the analysis and experimentation with comb filters. A simple experimental method is explored of estimating the frequency response magnitude of an FIR comb filter, and of an IIR comb filter.

2 Requirements

To carry out this lab, you will need:

- An STM32F746G Discovery board
- A PC running Keil MDK-Arm
- An oscilloscope
- 3.5 mm audio jack cables + BNC-BNC cables
- An audio frequency signal generator

3 The Comb Filter

As in the case of the moving average filter, the comb filter is a very simple type of filter that has peculiar characteristics, which makes it very convenient and useful in a number of applications including special audio effects, digital interpolation and digital decimation. The basic structure of a comb filter consists of the simple combination of an input signal ($x[n]$) and a delayed and scaled version of the input signal, or of the output signal ($y[n]$), or both. Thus, comb filters have three types of structure: an FIR structure, an IIR structure, or a combined FIR-IIR structure that we call general structure. These are illustrated in Figure 1. The examples that are illustrated in this figure assume that the delays affecting either the input sequence, or the output sequence, involve 6 samples.

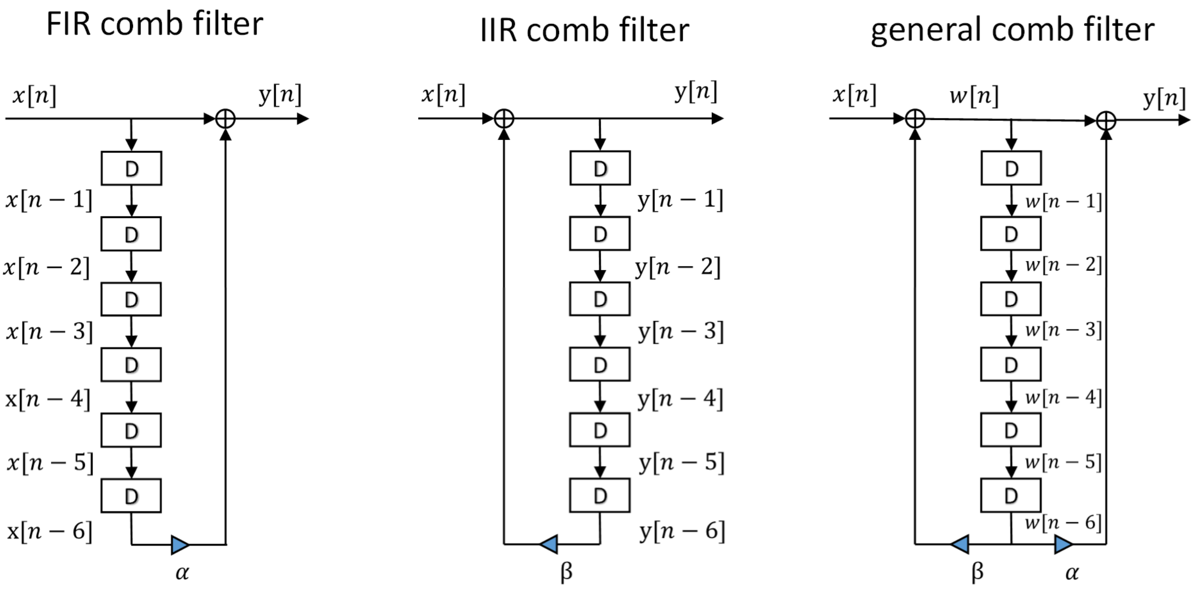


Figure 1: FIR comb filter structure (on the left), IIR comb filter structure (on the center), and general comb filter structure (on the right). The illustrated examples assume that the delay chain in each case involves 6 samples.

Using the Z-Transform, and assuming causality, it can be easily shown that the transfer function of the FIR comb filter is

$$\frac{Y(z)}{X(z)} = 1 + \alpha z^{-6}, \ |z| > 0, \tag{1}$$

the transfer function of the IIR comb filter is

$$\frac{Y(z)}{X(z)} = \frac{1}{1-\beta z^{-6}}, \ |z| > \sqrt[6]{|\beta|}, \tag{2}$$

and that the transfer function of the general comb filter is

$$\frac{Y(z)}{X(z)} = \frac{1+\alpha z^{-6}}{1-\beta z^{-6}}, \ |z| > \sqrt[6]{|\beta|}. \tag{3}$$

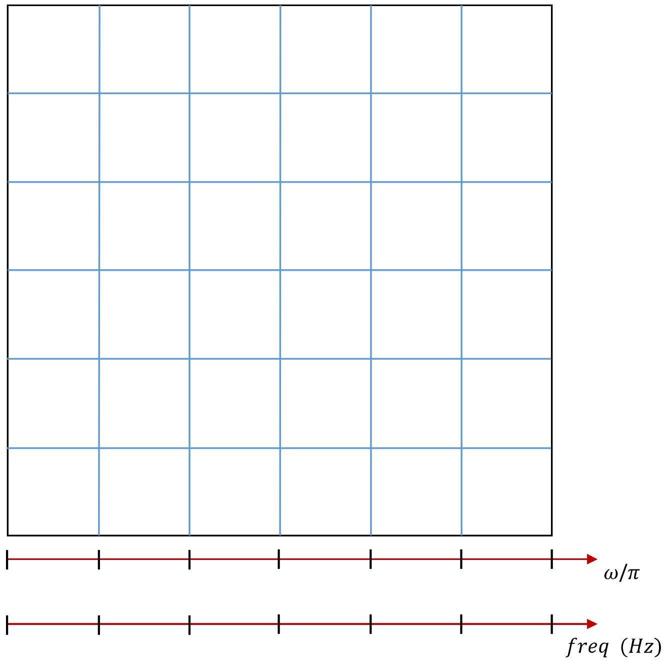
The peculiar characteristics of comb filters that justify their designation are manifested in the frequency domain, specifically, by the shape of the frequency response magnitude. In this lab (PL) class, we will measure the frequency response magnitude of an FIR comb filter according to Figure 1, where $\alpha = -0.95$, and the frequency response magnitude of an IIR comb filter according to Figure 1, where $\beta = -0.6$.

4 Preliminary (mandatory) analysis before the current lab

Before the lab (PL) class, you should answer the following questions. Assume that the sampling frequency is 8 kHz.

Question 1: find a compact (real-valued) expression that characterizes the frequency response magnitude of the FIR comb filter according to Equation (1).

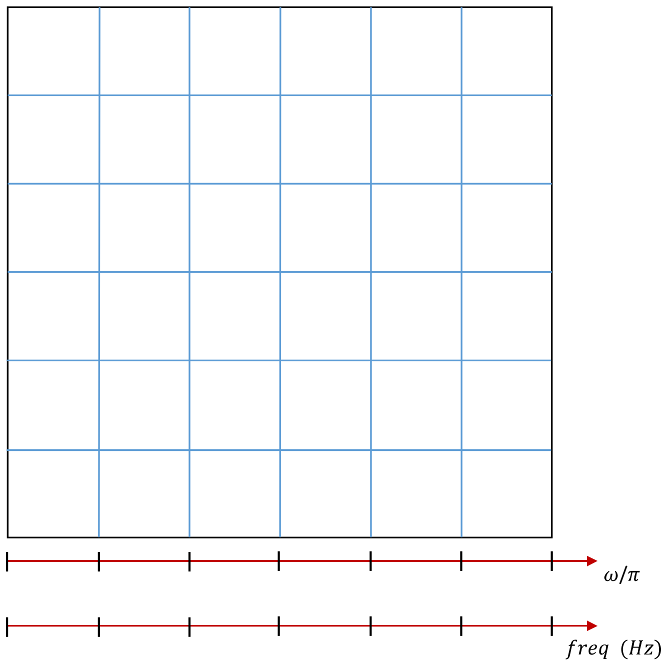
Question 2 [2pt / 10]: represent graphically the frequency response magnitude of the FIR comb filter using your answer to Question 1. Represent frequency using two different but equivalent frequency axes: ω/π (in the range [0, 1]) and `freq` (in Hz, in the range [0, F_N] where F_N is the Nyquist frequency). In particular, identify the maximum and minimum gain values in the frequency response magnitude.



Question 3: find a compact (real-valued) expression that characterizes the frequency response magnitude of the IIR comb filter according to Equation (2).

Question 4 [2pt / 10]: represent graphically the frequency response magnitude of the IIR comb filter using your answer to Question 3. As in the previous case, represent frequency using two different but equivalent frequency axes: ω/π (in the range [0, 1]), and `freq` (in Hz, in the range [0,

F_N] where F_N is the Nyquist frequency). In particular, identify the maximum and minimum gain values in the frequency response magnitude.



5 Observation of Frequency Response Using a Sinusoidal Input Signal

In this lab experiment, we will use the modified `main()` project file that is named **`stm32f7_average_intr_COMB.c`** and that is available on the Moodle platform. Its C code is listed next.

```
// stm32f7_average_intr_COMB.c

#include "stm32f7_wm8994_init.h"
#include "stm32f7_display.h"

#define SOURCE_FILE_NAME "stm32f7_average_intr_COMB.c"
#define N 6
#define ALFA -0.95
#define BETA -0.6

extern int16_t rx_sample_L;
extern int16_t rx_sample_R;
extern int16_t tx_sample_L;
extern int16_t tx_sample_R;

float32_t w[N+1] = {0, 0, 0, 0, 0, 0, 0};

enum filtertype{FIR,IIR};
```

```
void BSP_AUDIO_SAI_Interrupt_CallBack()
{
// when we arrive at this interrupt service routine (callback)
// the most recent input sample values are (already) in global variables
// rx_sample_L and rx_sample_R
// this routine should write new output sample values in
// global variables tx_sample_L and tx_sample_R
  int16_t i;
  float32_t w0, yn;

  // uncomment just one of the following two lines
   enum filtertype myfilter=FIR;
  // enum filtertype myfilter=IIR;


  w0  = (float32_t)(rx_sample_L);

  switch (myfilter)
  {
       case FIR:
          yn = w0 + (float32_t)(ALFA) * w[N];
          break;
       case IIR:
          w0 = w0 + (float32_t)(BETA) * w[N];
               yn = w0;
               // yn += (float32_t)(ALFA) * w[N]; // don't uncomment this line
          break;
       default:
          yn = w0;
  }

  tx_sample_L = (int16_t)(yn);
  tx_sample_R = (int16_t)(yn);

  w[0] = w0;
  for (i=N ; i>0 ; i--) w[i] = w[i-1];

  BSP_LED_Toggle(LED1);

  return;
}

int main(void)
{

  stm32f7_wm8994_init(AUDIO_FREQUENCY_8K,
                      IO_METHOD_INTR,
                      INPUT_DEVICE_INPUT_LINE_1,
                      OUTPUT_DEVICE_HEADPHONE,
                      WM8994_HP_OUT_ANALOG_GAIN_0DB,
                      WM8994_LINE_IN_GAIN_0DB,
                      WM8994_DMIC_GAIN_9DB,
                      SOURCE_FILE_NAME,
                     NOGRAPH);

  while(1){}
}
```

Take a moment to analyze this C code in order to conclude on:

- what lines of the code implement Equation (1)
- what lines of the code implement Equation (2)
- what filter is implemented by the code line:
  // yn += (float32_t)(ALFA) * w[N]; // don't uncomment this line

Question 5: what is the sampling frequency that this code specifies ? What is the Nyquist frequency ?

Question 6: What type (FIR, IIR) of comb filter is implemented by the above version of the C code ?

After unzipping it, take the `stm32f7_average_intr_COMB.c` file to the "src" directory that is located under folder:

C:\uvision\Keil\STM32F7xx_DFP\2.9.0\Projects\STM32746G-Discovery\Examples\DSP Education Kit\

Now, proceed as usual to start the Keil MDK-Arm development environment (µVision) and to replace the existing `main()` file in that project by the new `main()` that is `stm32f7_average_intr_COMB.c` .

Remember that the directory where you can find the **`DSP_Education_Kit.uvprojx`** project file is:

C:\uvision\Keil\STM32F7xx_DFP\2.9.0\Projects\STM32746G-Discovery\Examples\DSP Education Kit\MDK-ARM

You can copy-paste this link directly to File Explorer in Windows for a quick and easy access. For your convenience, this link is also available on a TXT file on Moodle.

Now, proceed as usual to compile the code, downloading it to the STM32F746G board (by starting the debugger), and then to run the code.

5.1 Setting-up for this lab experiment

Set the function generator to generate a sine wave having 5 Vpp and 2000 Hz. Using a "T" and a BNC-BNC cable, take the output of the function generator to CHAN1 of the oscilloscope.

As shown in Figure 2, connect the output of the sinusoidal signal generator (i.e. the function generator) to the (left channel of the) LINE IN socket on the Discovery board (Remember: make sure that you use the adapter with the **blue mini-jack** whose interface board has a resistor divider. It is meant to protect the analog input of the ***kit*** against excessive voltage levels).

Then, using another BNC-BNC cable, take the LEFT channel of the STM32F746G LINE OUT output to the CHAN2 input of the oscilloscope.

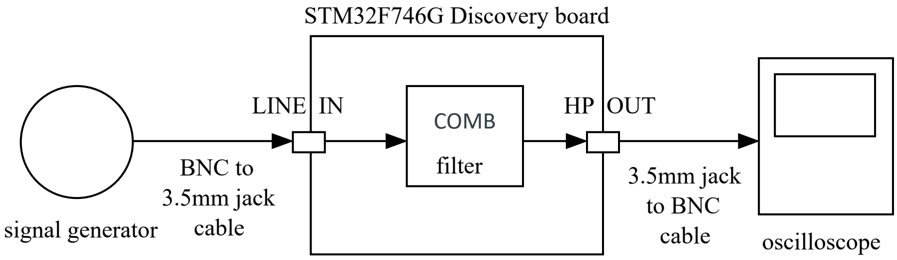


Figure 2: Connection diagram for measuring the frequency response magnitude of the comb filter implemented by program `stm32_average_intr_COMB.c` using a signal generator and an oscilloscope.

Using the oscilloscope SETTINGS button and menu, make sure that the Vpp and frequency of both input and output signals are being measured in real-time. You should obtain a graphical representation similar to that illustrated in Figure 3.

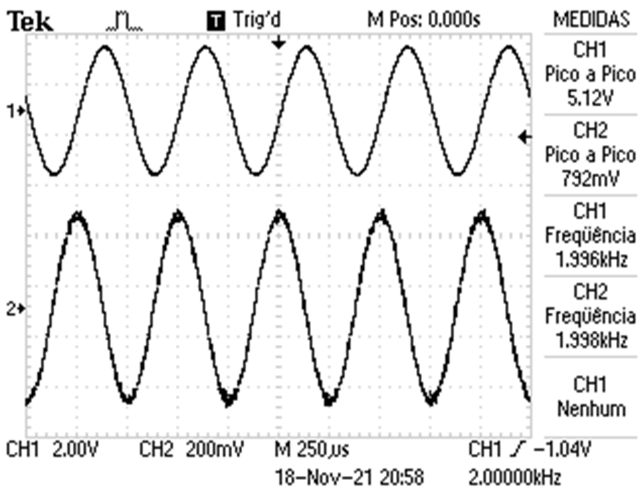


Figure 3: input and output signals from program `stm32f7_average_intr_COMB.c` on the oscilloscope.

5.2 Sketching the frequency response of comb filters

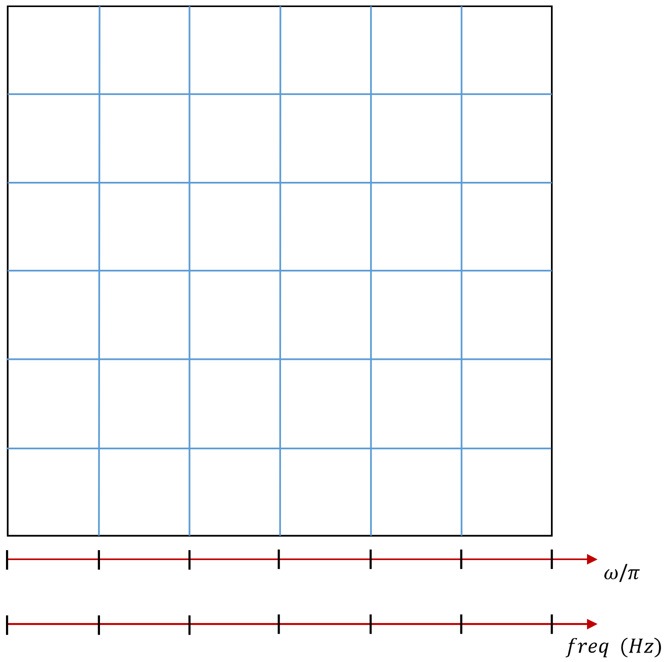
The frequency response of a filter tells us its gain at different frequencies, and hence one way of assessing the frequency response of the filter is simply to measure its gain using a sinusoidal input signal at a number of different frequencies.

As the frequency of the inputs signal is varied, the amplitude of the output signal should change. The gain of the comb filter is higher at certain frequencies, and lower at other frequencies.

In this experiment, you will vary the frequency from 100 Hz up to 4 kHz and take note of the Vpp and frequency of the output wave represented on the oscilloscope, every time it reaches a maximum Vpp, and every time it reaches a minimum Vpp.

Question 8: How many maxima and minima do you observe in the frequency response magnitude ? Is that consistent with your answers to Questions 2, or 4 above ?

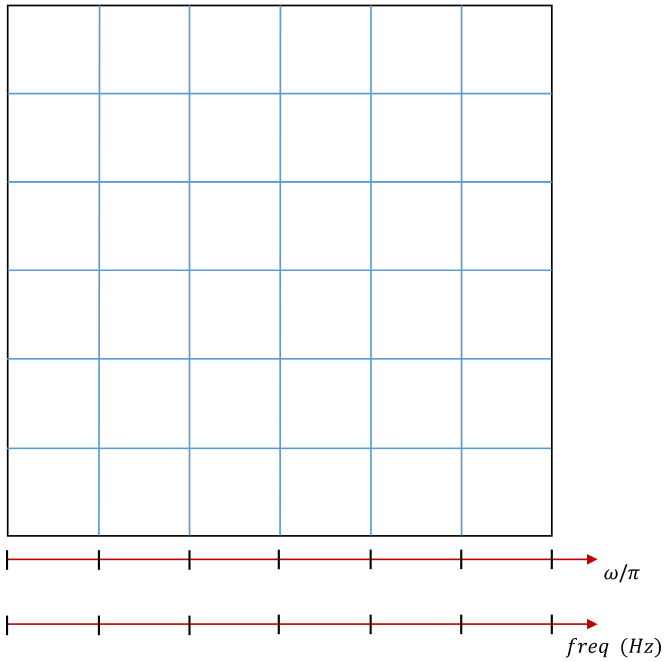
Sketch in the following plot the approximate frequency response of the comb filter that is implemented by the above C code when you vary the frequency from 100 Hz up to 4 kHz (suggestion: use as reference points the Vpp and corresponding frequency every time the output sinusoid reaches a maximum Vpp, and every time it reaches a minimum Vpp).



Question 8 [3pt / 10]: Is this sketch consistent with that of either Question 2, or Question 4 ? Are the ratios between the maximum Vpp and minimum Vpp consistent with the ratios between maximum gain and minimum gain in one of the theoretical cases, as per Question 2, or Question 4 ?

Now, modify the `stm32f7_average_intr_COMB.c` file such that it implements the other type of comb filter (i.e. if it is configured as an FIR, change it to an IIR, and if it is configured as an IIR filter, change it to an FIR filter). Now, compile the modified code, download it to the STM32F746G board and run it.

Sketch in the following plot the approximate frequency response of the new comb filter when you vary the frequency from 100 Hz up to 4 kHz (suggestion: use as reference points the Vpp and corresponding frequency every time the output sinusoid reaches a maximum Vpp, and every time it reaches a minimum Vpp).



Question 9 [3pt / 10]: Is this sketch consistent with that of either Question 2, or Question 4 ? Are the ratios between the maximum Vpp and minimum Vpp consistent with the ratios between maximum gain and minimum gain in one of the theoretical cases, as per Question 2, or Question 4 ?

5.3 Testing another filter (extra, only if time permits)

The `stm32f7_average_intr_COMB.c` file includes the following commented line:

```
                // yn += (float32_t)(ALFA) * w[N]; // don't uncomment this line
```

If you uncomment this line, what filter is being implemented ?

Now, compile the modified code, download it to the STM32F746G board and run it.

If you test the frequency response of this new filter, how does it compare that of the previous tests (Question 8 and Question 9) ? How do you explain possible differences ?

6 Conclusions

This laboratory exercise motivated the implementation and real-time test of different comb filters. Students should know why comb filter are called so, how their frequency response magnitude looks like, and how it changes as a function of the length of the delay line and of the ALFA and BETA coefficients.

7 Additional References

Comb filters:

https://en.wikipedia.org/wiki/Comb_filter