1. Introduction

The aim of the laboratory was to familiarize with two types of digital filers – FIR and IIR. The main difference between is that Finite Impulse Response has a response that settle down to zero in some time, hence finite, while Infinite Impulse Response can last to infinity due to its feedback loop. Both types have its advantages and disadvantages:

- FIR:

  + easily designed to have linear phase and avoid phase distortion

  + as there is no feedback, there is no possibility of instability

  + better fixed point performance than IIR

  - more complex, requires higher computational power and more memory

  - FIR needs more coefficients so it introduces bigger latency

  - there is no analog equivalent

- IIR:

  + easier to implement, less complex

  + lower latency

  + analog equivalent can be achieved

  - phase characteristic are non-linear

  - lower fixed point performance

  - positive feedback can cause instability

2. Results

All coefficients were generated using [online generator](). Used code:

```
//3rd low pass
#define NZEROS 3
#define NPOLES 3
#define GAIN   (float)3.430944333e+004

//2nd bandpass
/*#define NZEROS 4
#define NPOLES 4
#define GAIN   (float)4.787177042e+002*/

void HAL_TIM_PeriodElapsedCallback(TIM_HandleTypeDef *htim)
{
  if(htim == &htim2)
  {
    //3rd low pass
    xv[0] = xv[1]; xv[1] = xv[2]; xv[2] = xv[3];
            xv[3] = input / GAIN;
            yv[0] = yv[1]; yv[1] = yv[2]; yv[2] = yv[3];
            yv[3] =  (xv[0] + xv[3]) + 3 * (xv[1] + xv[2])
                     + (  0.8818931306f * yv[0]) + ( -2.7564831952f * yv[1])
                     + (  2.8743568927f * yv[2]);
            output = yv[3];

    //2nd bandpass
    /*xv[0] = xv[1]; xv[1] = xv[2]; xv[2] = xv[3]; xv[3] = xv[4];
            xv[4] = input / GAIN;
            yv[0] = yv[1]; yv[1] = yv[2]; yv[2] = yv[3]; yv[3] = yv[4];
            yv[4] =   (xv[0] + xv[4]) - 2 * xv[2]
                     + ( -0.8752145483f * yv[0]) + (  3.6101781308f * yv[1])
                     + ( -5.5942343734f * yv[2]) + (  3.8592561983f * yv[3]);
            output = yv[4] + 2048;*/

    HAL_DAC_SetValue(&hdac, DAC1_CHANNEL_2, DAC_ALIGN_12B_R, output);
  }
}

void HAL_ADC_ConvCpltCallback(ADC_HandleTypeDef* hadc )
{
  input = HAL_ADC_GetValue(&hadc1);
  //amplitude = modulation / 4096;
}
```

2.1. 3rd order low pass IIR filter

Coefficients (in the snippet above) were generated to obtain 3rd order low pass IIR filter (Butterworth) for f_c = 1kHz. Sample rate was equal to 200 kHz (as set in CubeMX). Next signal was outputted by Analog Discovery and fed to STM32 for filtration.

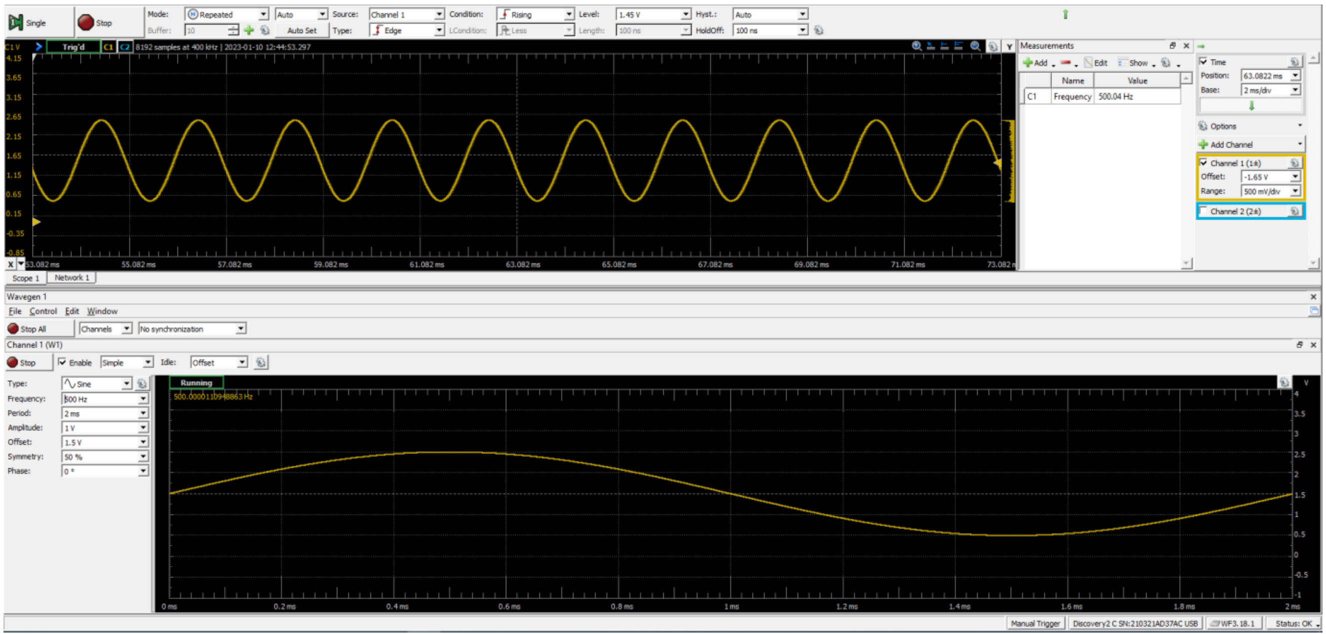

Figure 1 Signal filtration for f = 500 Hz

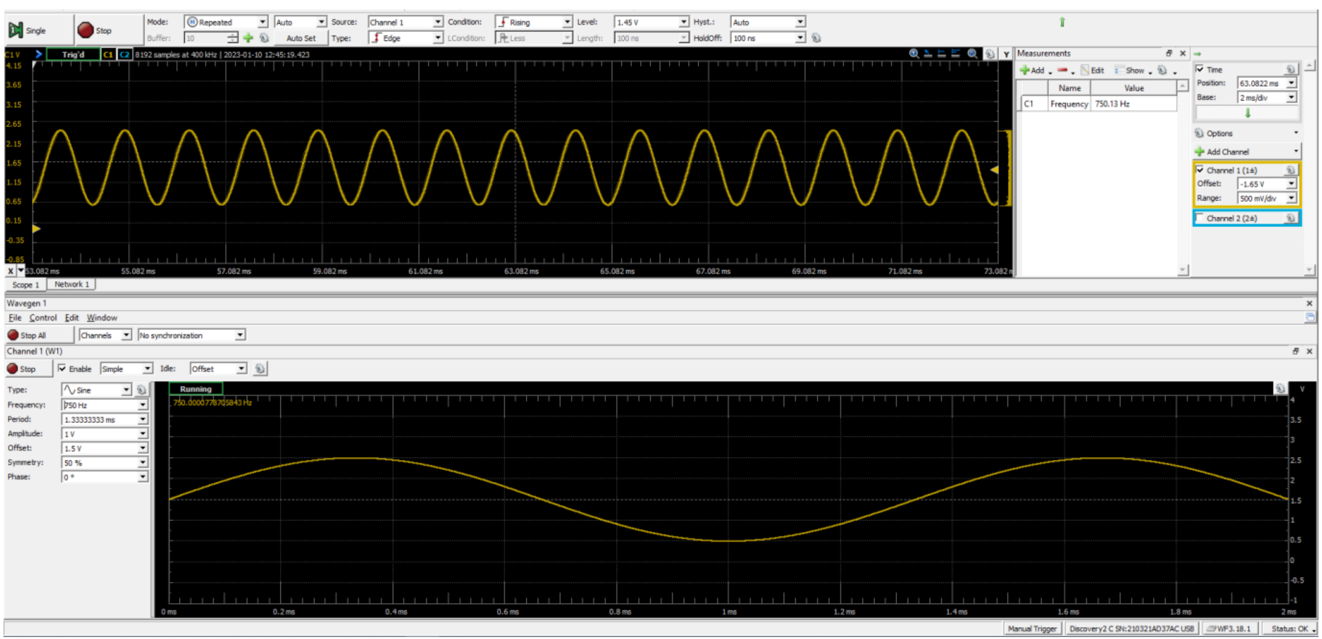

Figure 2 Signal filtration for f = 750 Hz

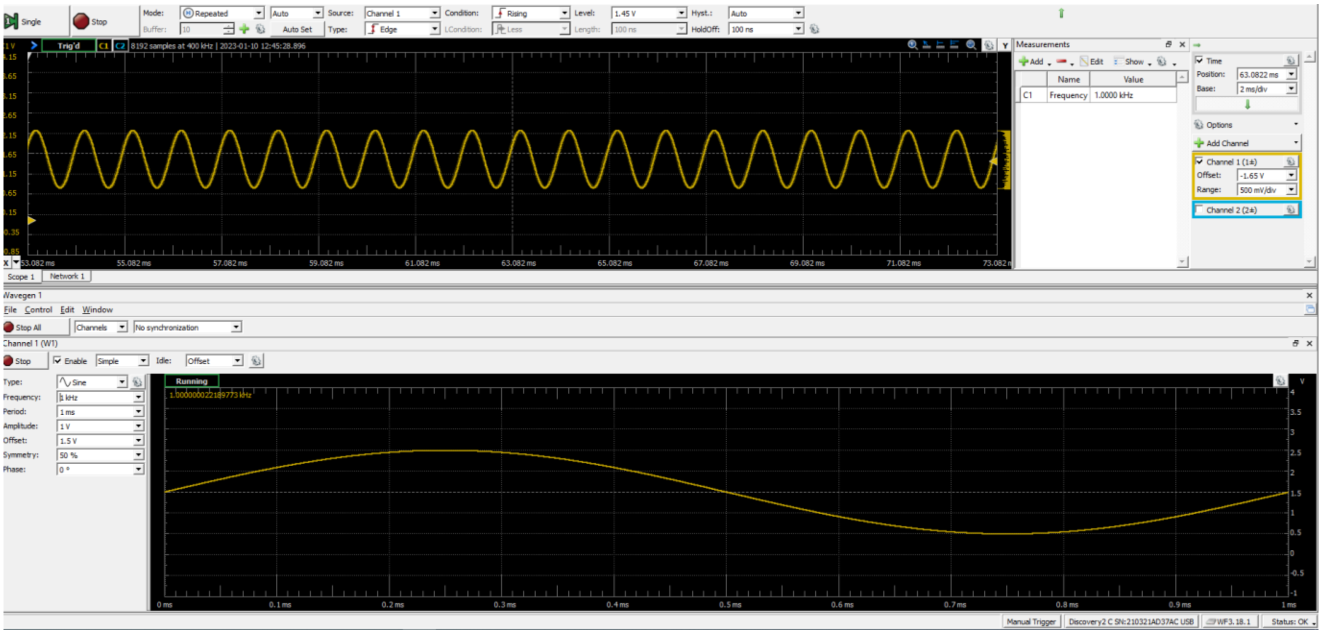

Figure 3Signal filtration for f = 1 kHz

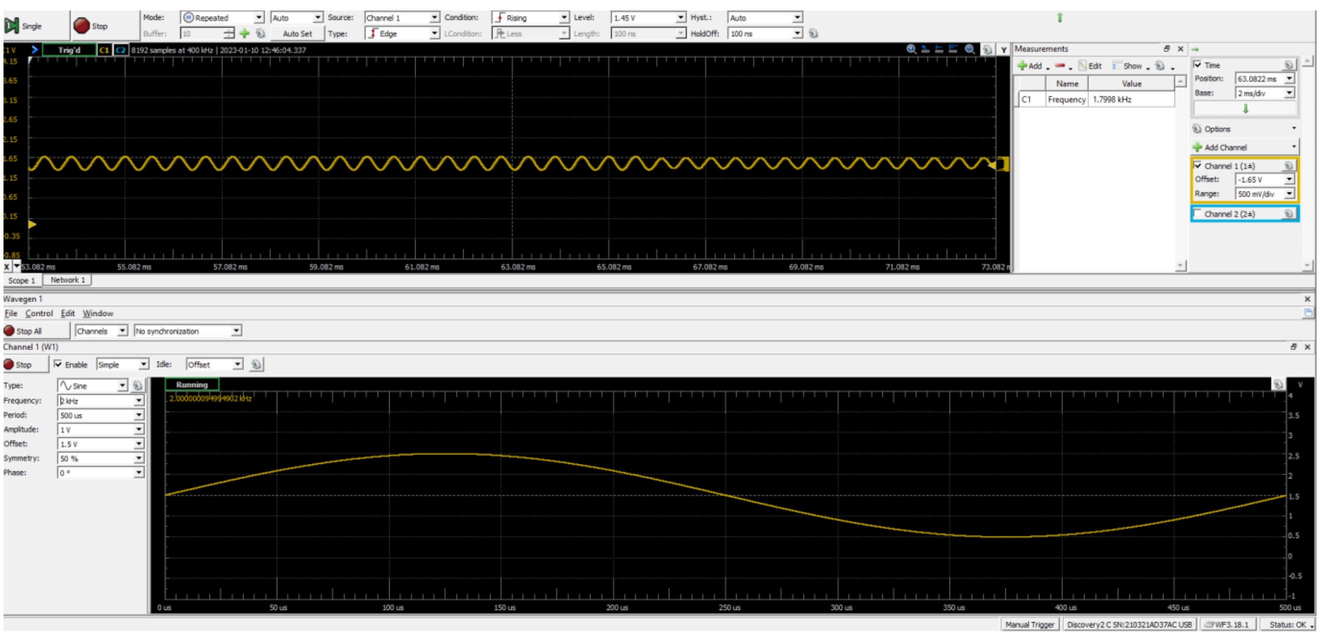

Figure 4 Signal filtration for f = 2 kHz

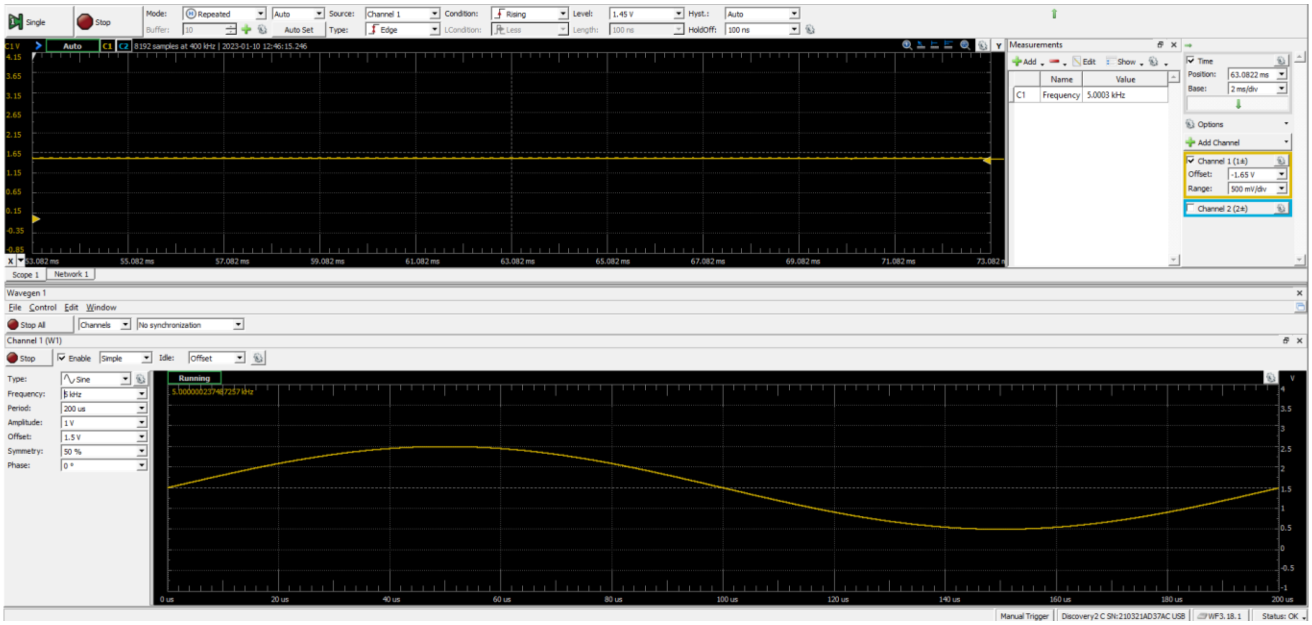

Figure 5 Signal filtration for f = 5 kHz

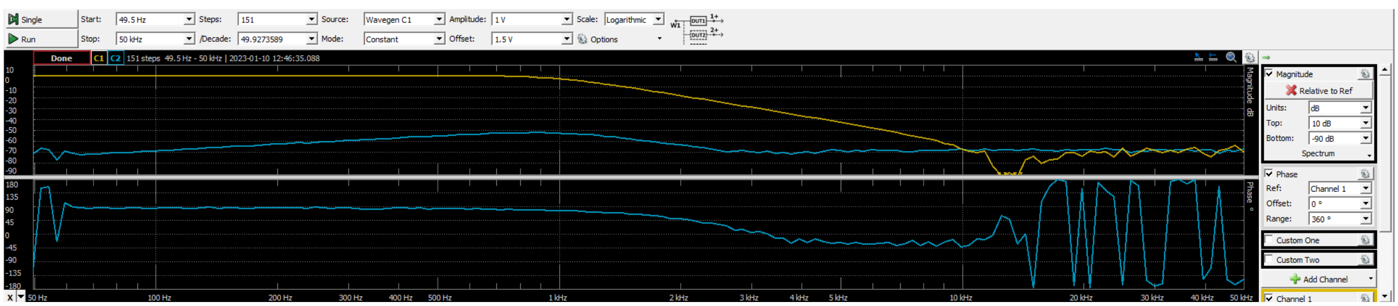

Figure 6 Bode chart

As seen in Fig. 1 – 5 filter works properly. Under the cut-off frequency (set to 1 kHz) amplitude stays (around) the same and after the f_c signal starts to be attenuated. The characteristic of the implemented filter are more easily seen on Bode graph. Top part shows magnitude and bottom shows phase. As expected, on level of f_c magnitude drops by ~3dB, then it continues to decrease. Increasing order of filter would increase the slope.

2.2. 2nd order band pass IIR filter

Coefficients were generated to obtain 2nd band pass IIR filter (Butterworth) for f_{c1} = 500 Hz and f_{c2} = 2 kHz. Next, signal was outputted by Analog Discovery and fed to STM32 for filtration.

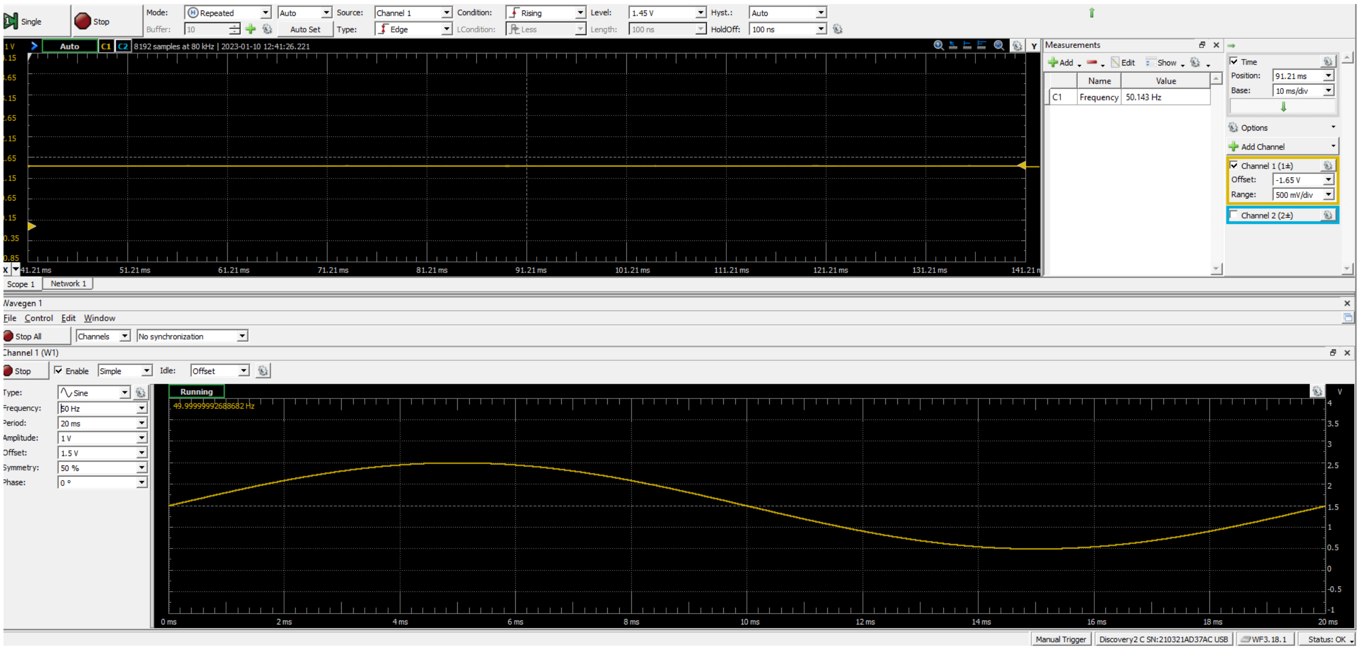

Figure 7 Signal filtration for f = 50 Hz

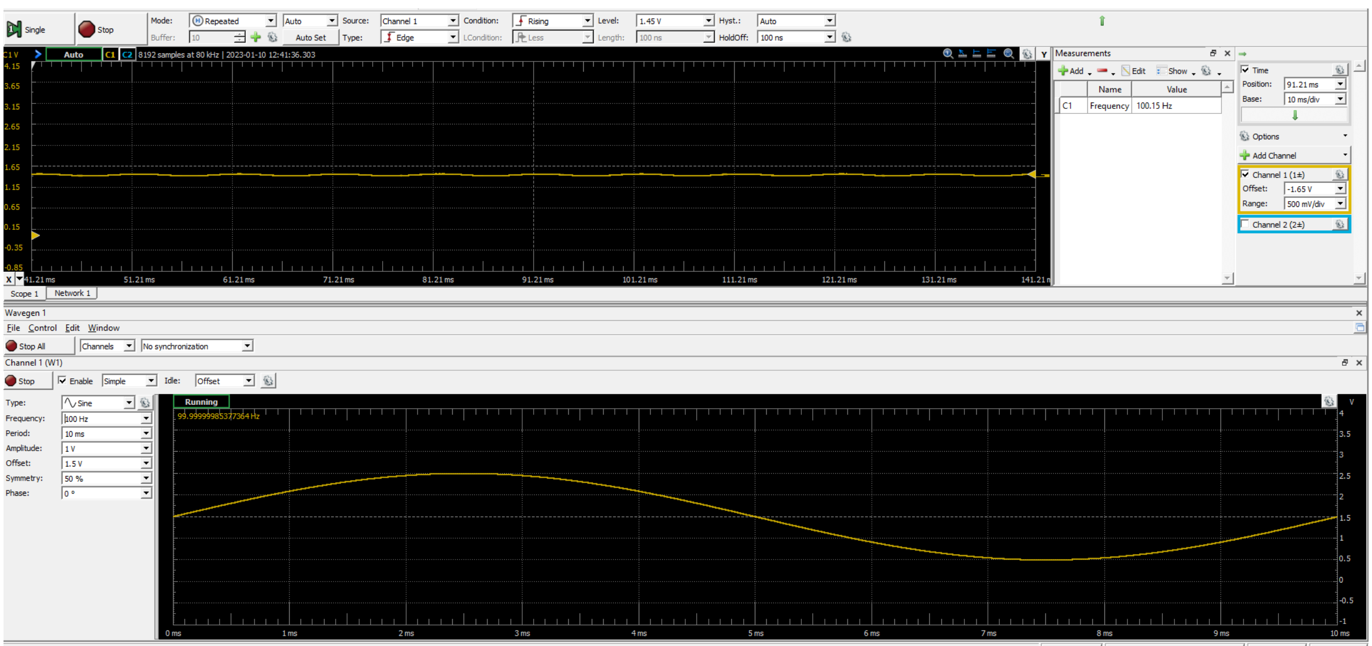

Figure 8 Signal filtration for f = 100 Hz

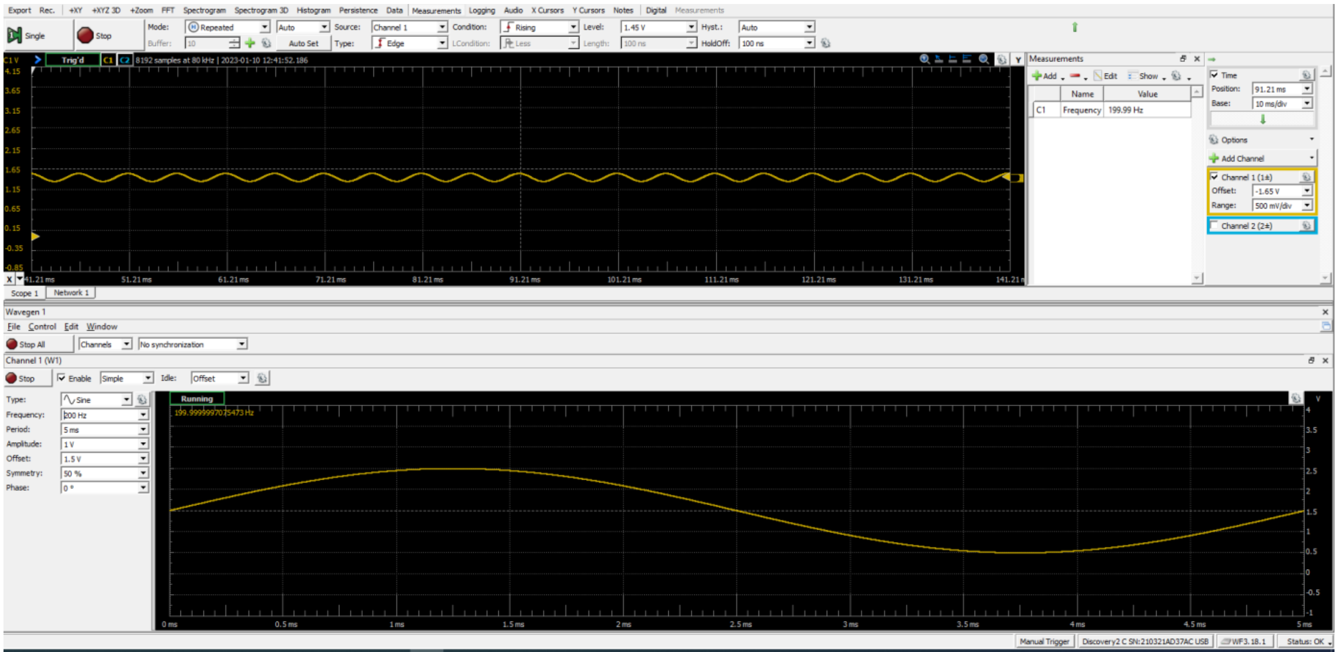

Figure 9 Signal filtration for f = 200 Hz

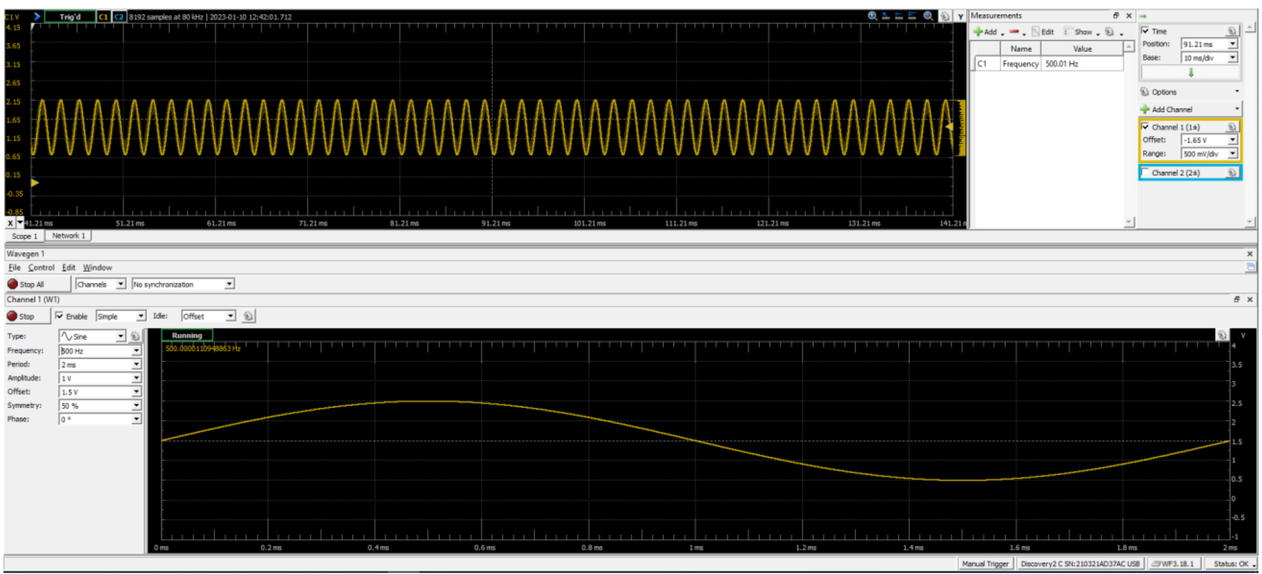

Figure 10 Signal filtration for f = 500 Hz

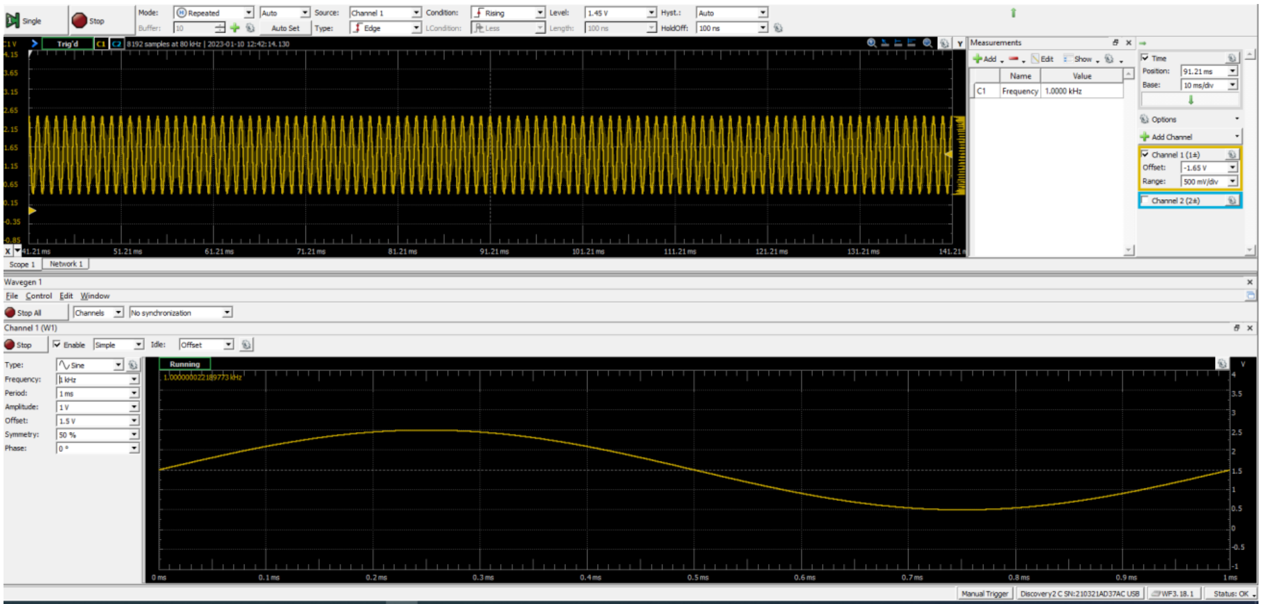

Figure 11 Signal filtration for f = 1 kHz

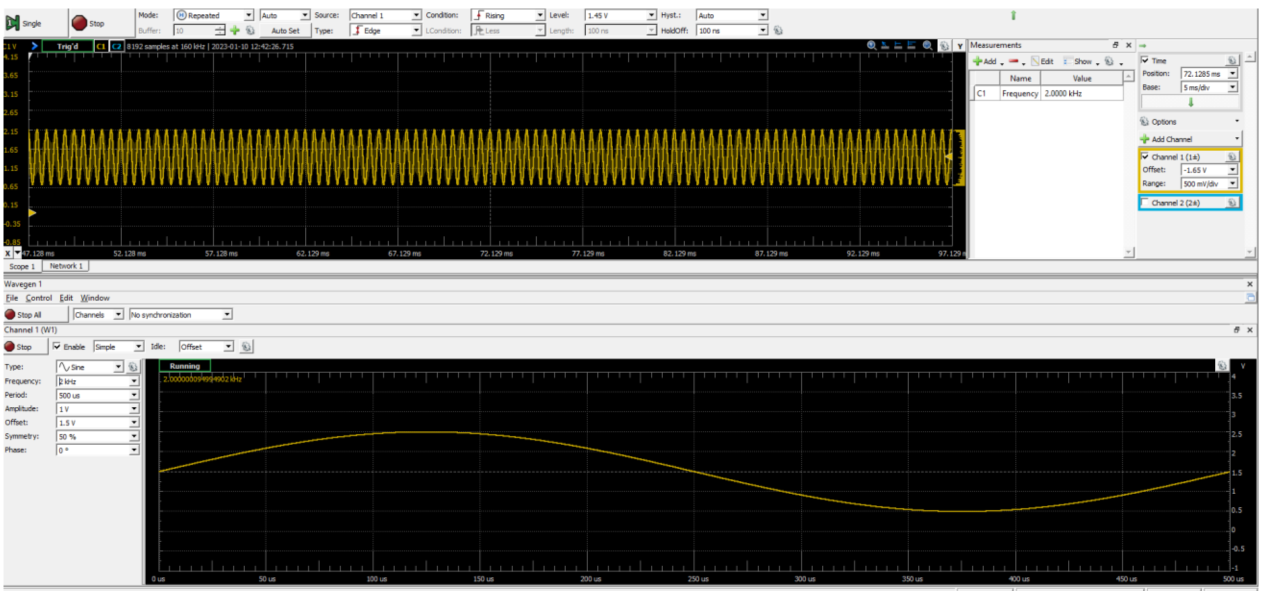

Figure 12 Signal filtration for f = 2 kHz

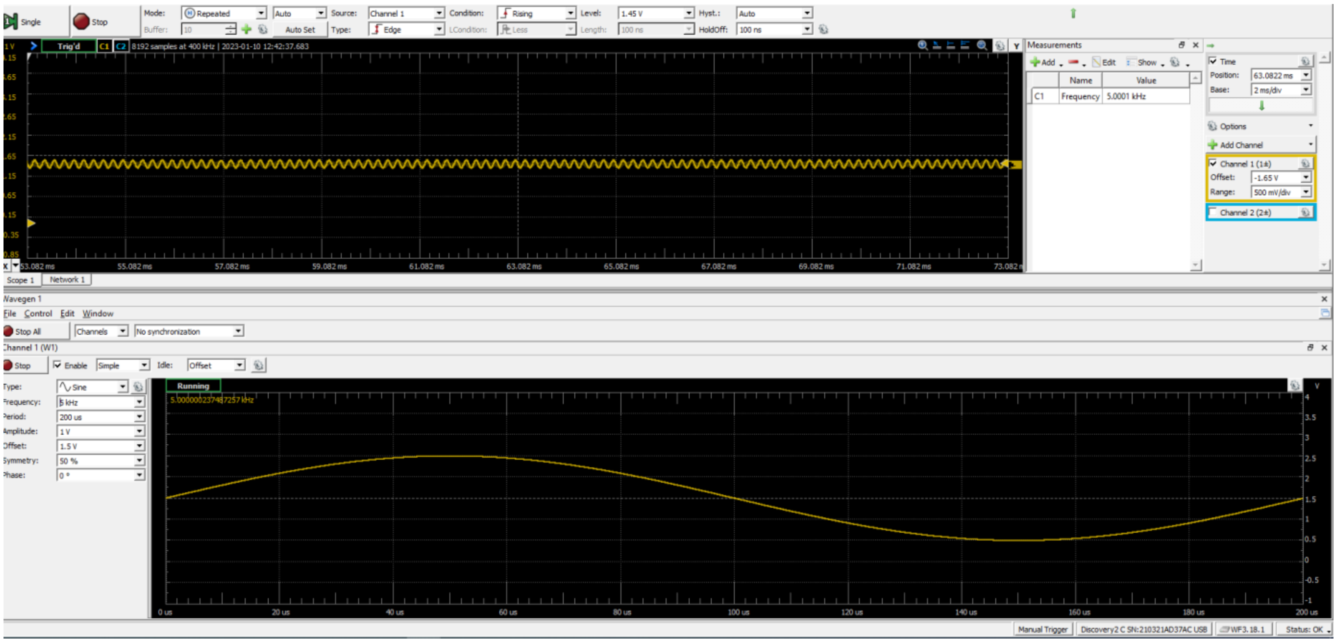

Figure 13 Signal filtration for f = 5 kHz

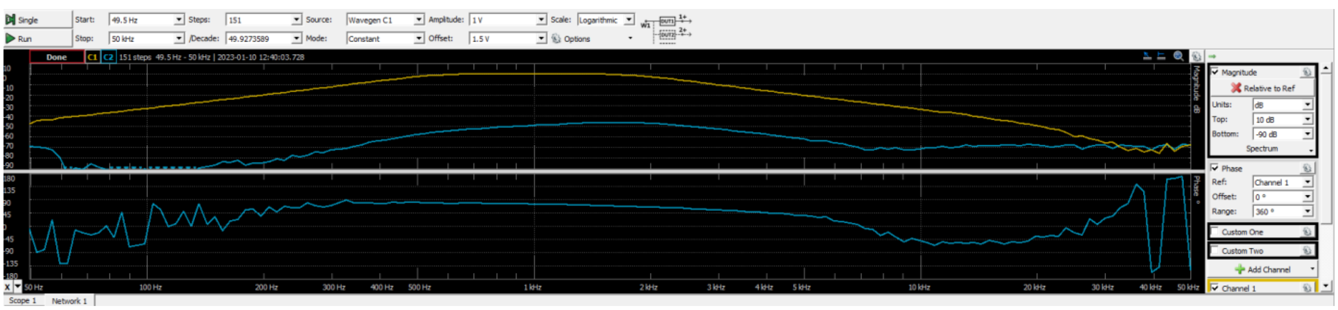

Figure 14 Bode graph

As shown on Fig. 7 – 13, signal seems to be attenuated below 500 Hz and above 2 kHz as expected. Again, Bode graph shows a better view. One can see near 0 dB magnitude between f_{c1} and f_{c2}. In those points magnitude drop by ~3 dB. Beyond corner frequencies magnitude decreases.