

Отчет по лабораторной работе №3

Головизнин Э. О.

25 мая 2018 г.

1 Цель работы

Изучить воздействие ФНЧ на тестовый сигнал с шумом.

2 Постановка задачи

Сгенерировать гармонический сигнал с шумом и синтезировать ФНЧ. Получить сигнал во временной и частотной областях до и после фильтрации. Сделать выводы о воздействии ФНЧ на спектр сигнала.

3 Теоретический раздел

3.1 Общие сведения о линейной фильтрации

Линейный фильтр — динамическая система, применяющая некий линейный оператор ко входному сигналу для выделения или подавления определённых частот сигнала и других функций по обработке входного сигнала. Линейные фильтры широко применяются в электронике, цифровой обработке сигналов и изображений, в оптике, теории управления и других областях.

Наиболее часто они используются для того, чтобы подавить нежелательные частоты входного сигнала или для того чтобы выделить нужную полосу частот в сигнале. Существует большое количество различных типов и модификаций линейных фильтров, в статье описаны наиболее распространённые.

Несмотря на природу фильтра — механическую, оптическую, электронную, программную или электрическую, а также на частотный диапазон, в котором они работают, математическая теория линейных фильтров универсальна и может быть применена к любому из них.

4 Ход работы

4.1 Генерация гармонического сигнала с шумом

Синусоидальный сигнал:

```
function func = my_sin(t)
    A = 4; % амплитуда
    f = 12; % частота
    ph = 2; % сдвиг фазы

    func = A * sin(2 * pi * f * t + ph);
end
```

Построим исходный сигнал и зашумленный сигнал:

```
% Частота сигнала:
Fn = 4;
% Частота дискретизации:
Fd = 2000;

t = 0:1/Fd:1;

x = my_sin(t);

y = awgn(x, 15);
```

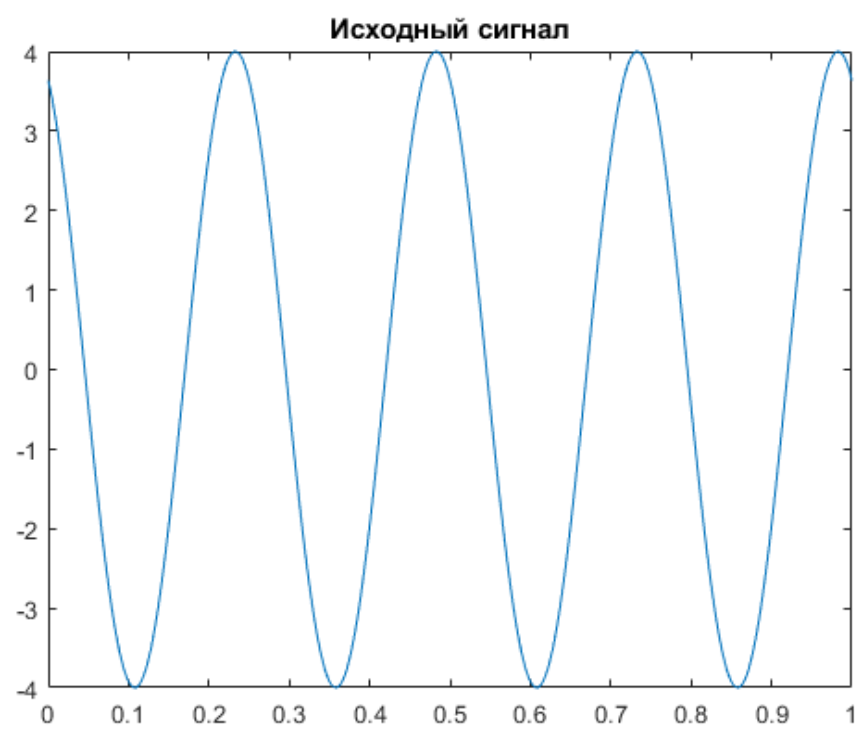


Рис. 1: Исходный сигнал

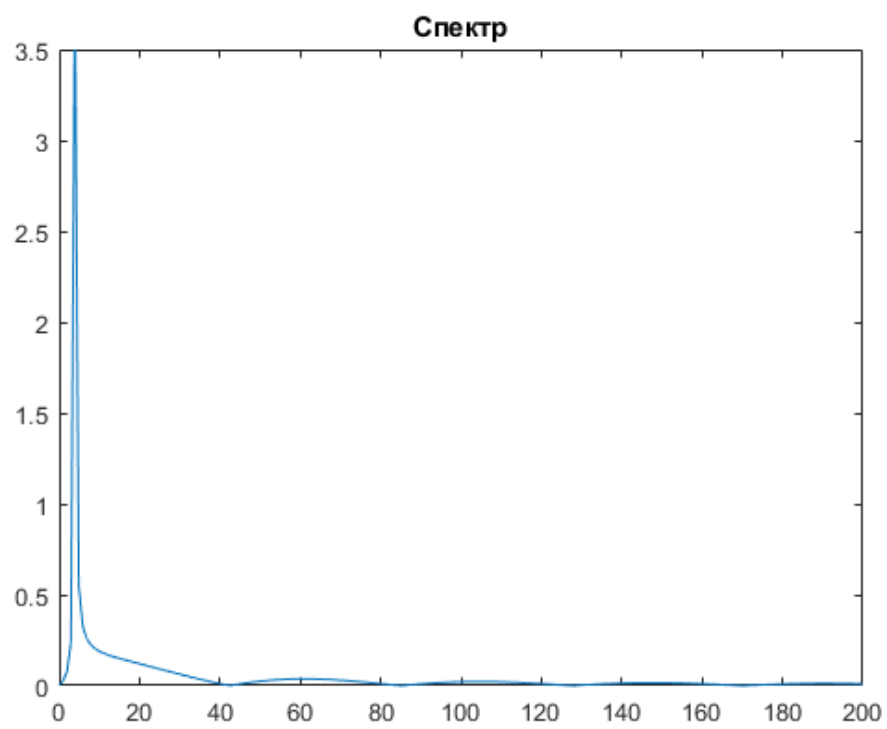


Рис. 2: Спектр исходного сигнала

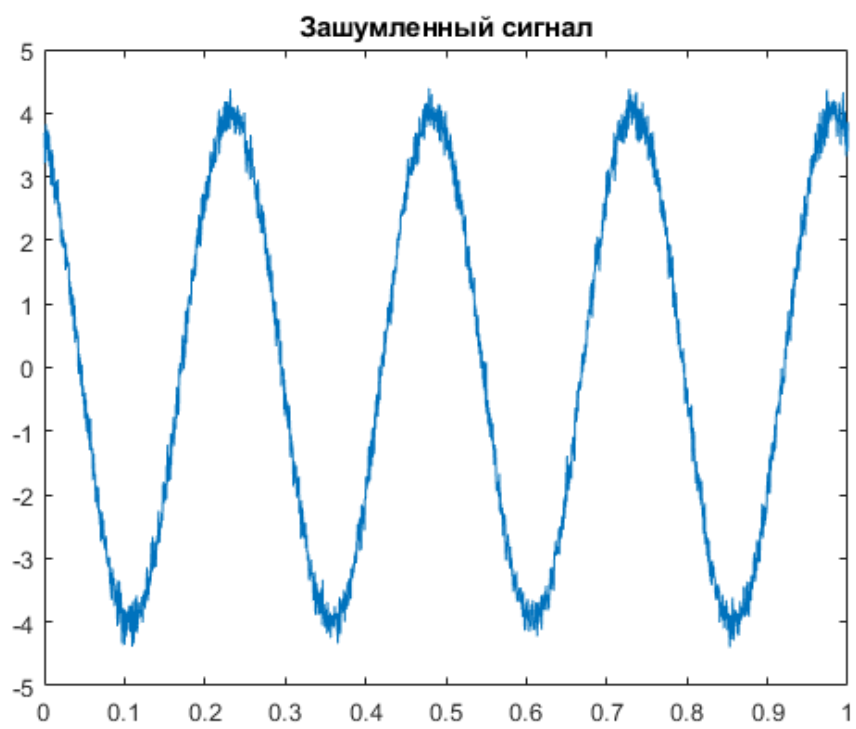


Рис. 3: Зашумленный сигнал.

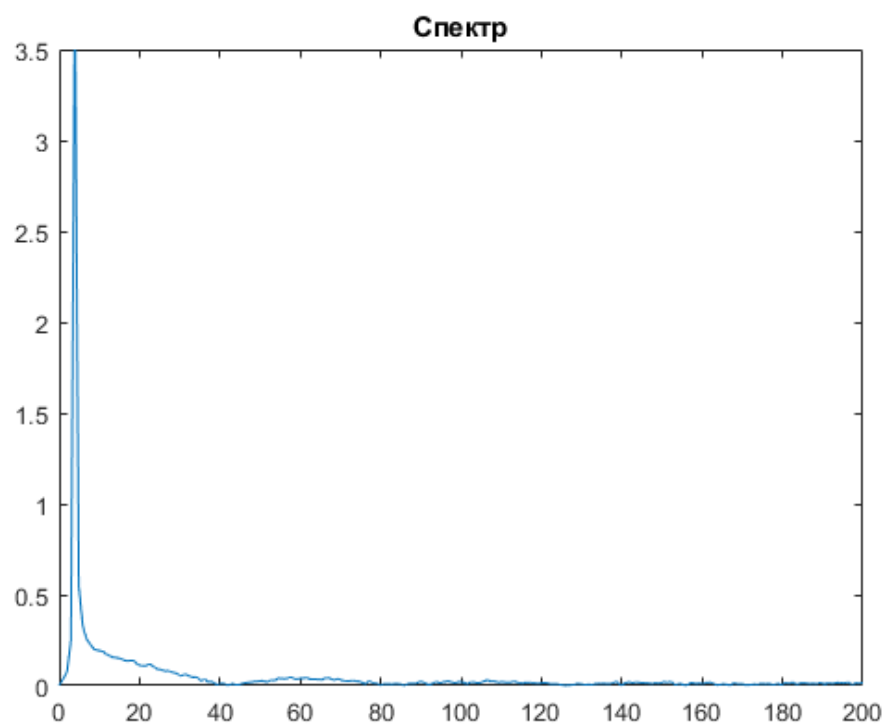


Рис. 4: Спектр зашумленного сигнала

4.2 Фильтрация сигнала с шумом

Синтезируем фильтр с помощью Filter Design:

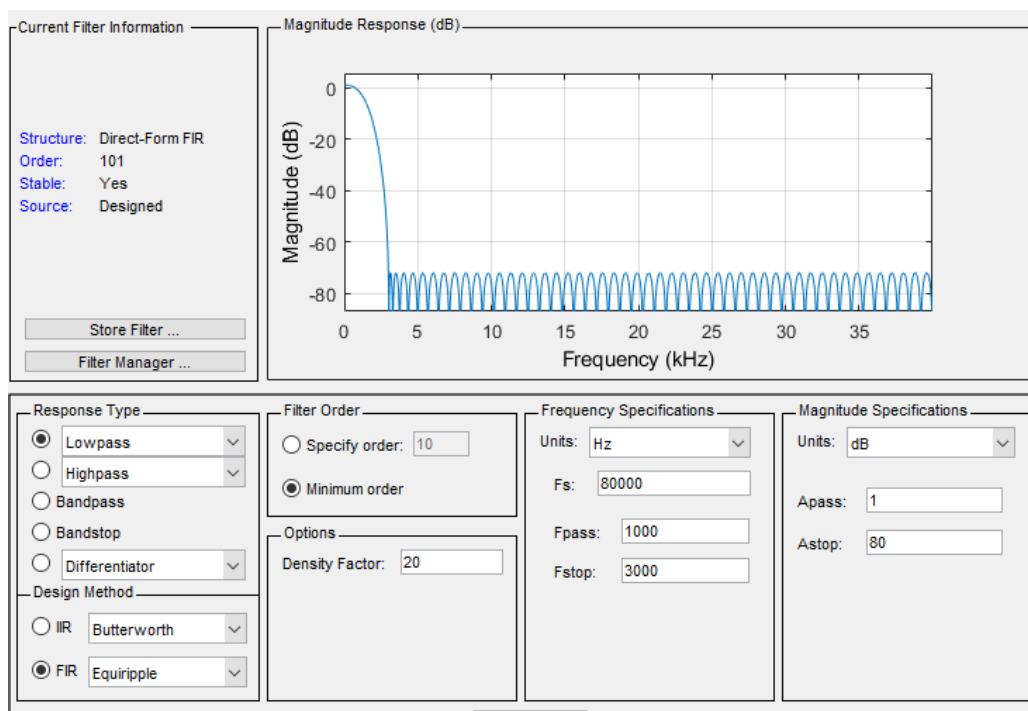


Рис. 5: Настройки фильтра в Filter Designer

Сгенерированный код:

```
function Hd = ffilter
%FFILTER Returns a discrete-time filter object.

% MATLAB Code
% Generated by MATLAB(R) 9.4 and DSP System Toolbox 9.6.
% Generated on: 25-May-2018 12:10:18

% Equiripple Lowpass filter designed using the FIRPM function.

% All frequency values are in Hz.
Fs = 80000; % Sampling Frequency

Fpass = 1000; % Passband Frequency
Fstop = 3000; % Stopband Frequency
Dpass = 0.057501127785; % Passband Ripple
Dstop = 0.0001; % Stopband Attenuation
dens = 20; % Density Factor
```



```

% Calculate the order from the parameters using FIRPMORD.
[N, Fo, Ao, W] = firpmord([Fpass, Fstop]/(Fs/2), [1 0], [Dpass, Dstop]);

% Calculate the coefficients using the FIRPM function.
b = firpm(N, Fo, Ao, W, {dens});
Hd = dfilt.dfir(b);

% [EOF]

```

Получим отфильтрованный сигнал:

```

f = ffilter;
filtered = filter(ffilter, y);

```

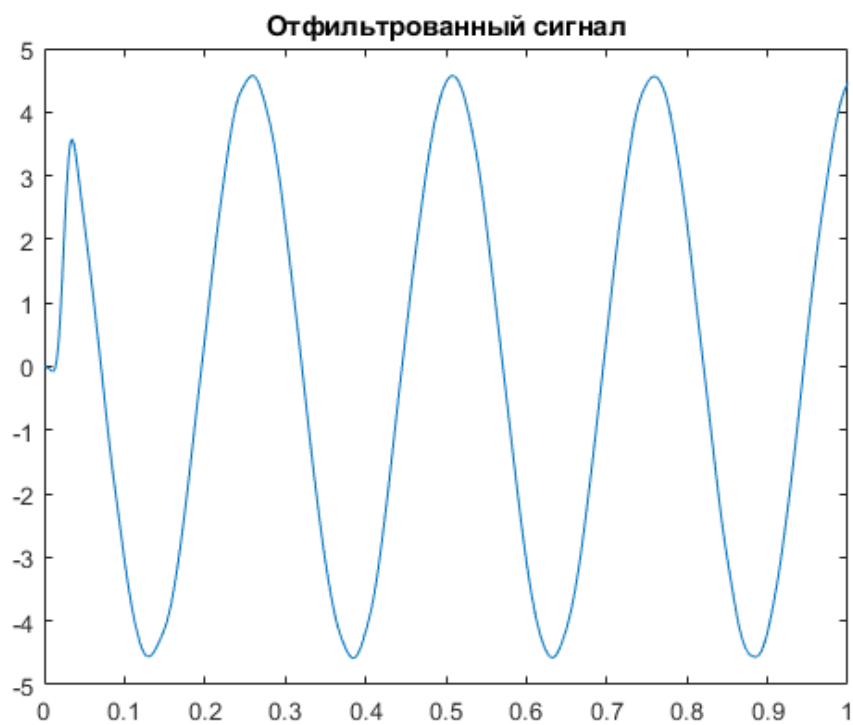


Рис. 6: Отфильтрованный сигнал.

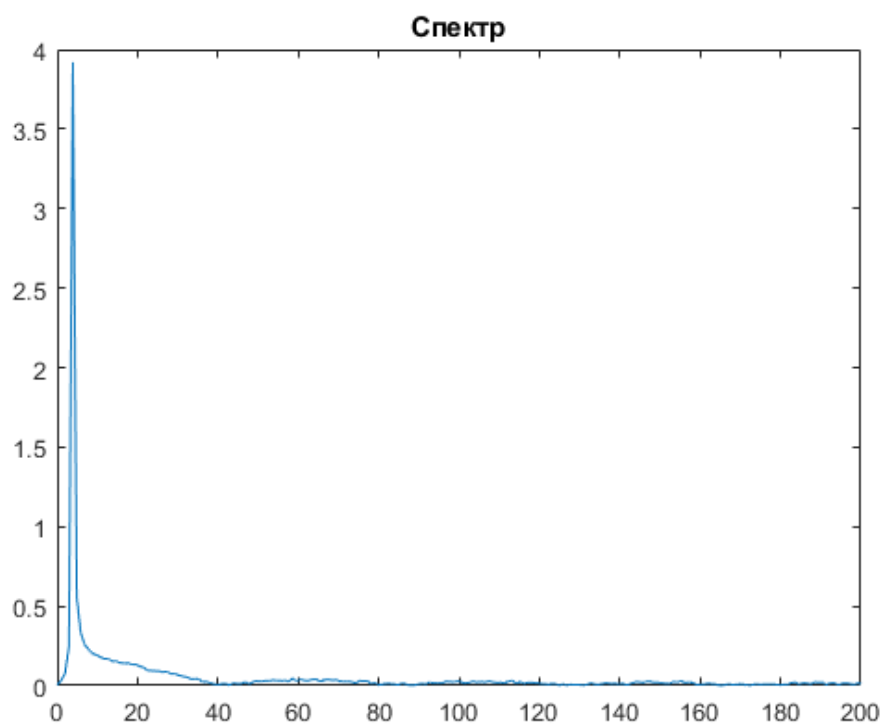


Рис. 7: Спектр отфильтрованного сигнала.

5 Выводы

В ходе работы была исследована работа фильтра нижних частот с помощью инструмента Filter Design. Отфильтрованный сигнал практически соответствует исходному, значит, фильтр работает правильно. Но имеются некоторые различия с исходным сигналом.