ESRG
Embedded Systems
Research Group

CENTROALGORITMI

Project Status

PIEIC

Oscilloscope based in FPGA

Diogo Fernandes – PG47150
José Tomás Abreu – PG47386

Universidade do Minho
Escola de Engenharia
COMPETE
QUADRO DE REFERÊNCIA ESTRATÉGICO NACIONAL
União Europeia
Fundo Europeu de Desenvolvimento Regional
FCT Fundação para a Ciência e a Tecnologia
MINISTÉRIO DA CIÊNCIA, TECNOLOGIA E ENSINO SUPERIOR Portugal

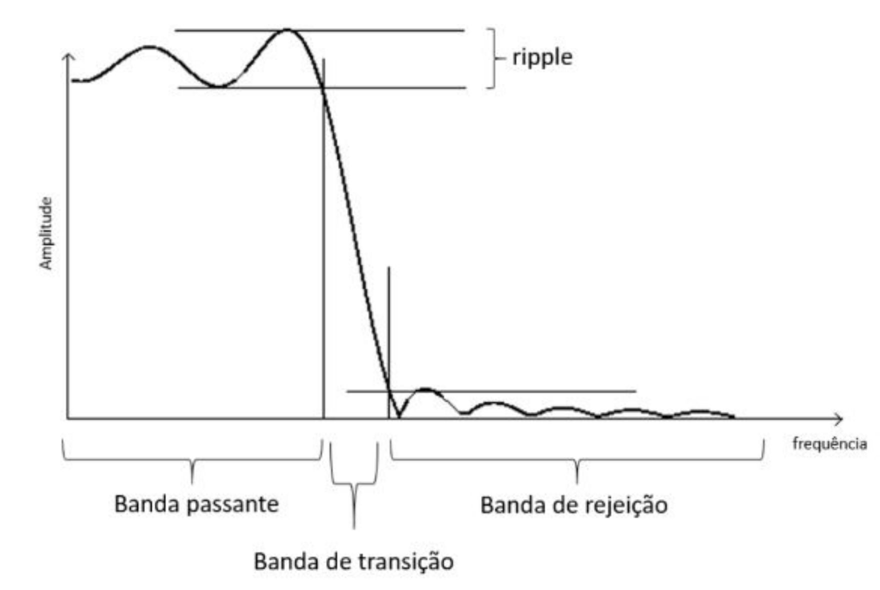
Agenda

- Digital Filters
  - FIR
  - IIR
- Sampling and Filtering
- Low Pass Filter
  - Filter Calculation
  - Frequency Response
  - Results
- High Pass Filter
  - Filter Calculation
  - Frequency Response
  - Results
- Band Pass Filter
  - Filter Calculation
  - Frequency Response
  - Results

Digital Filters

- IIR (Infinite Impulse Response)
  - Similar to analog filters
  - More efficient
  - Unstable

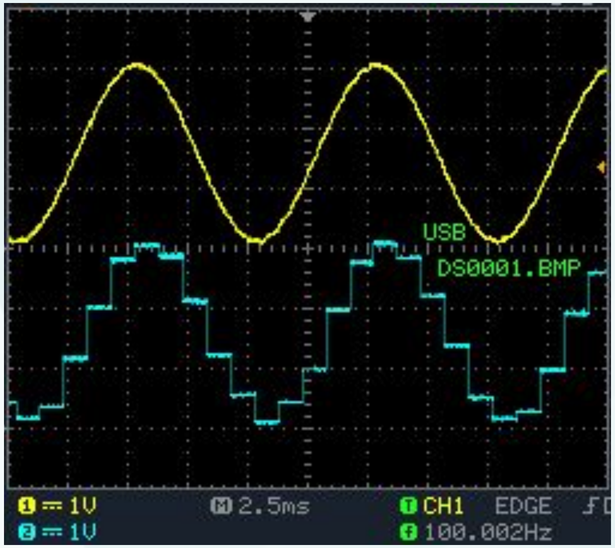
- FIR (Finite Impulse Response)
  - Better response
  - Stable
  - Robust
  - Linear phase



Sampling and Filtering

$$\begin{cases} t = nh \rightarrow x^*(t) = x(nh) \\ t \neq nh \rightarrow x^*(t) = 0 \end{cases}$$

$$f_S \geq 2.\, f_M$$



Sample Frequency: fs = 1 kHz

$$H(z) = \sum_{n=-\infty}^{+\infty} h(n)\, z^{-n}$$

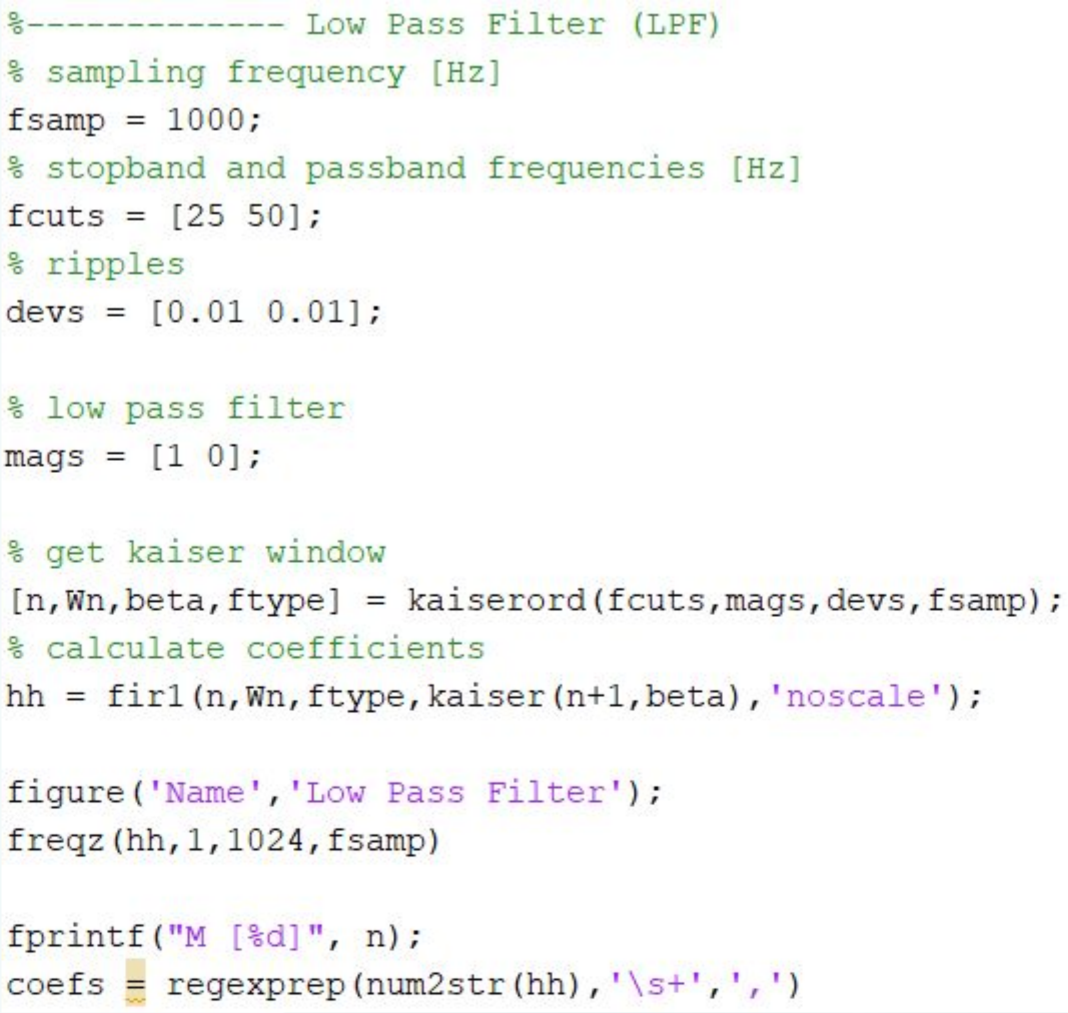
→

$$y(n) = h(n) * x(n) = \sum_{k=-\infty}^{+\infty} h(k)\, x(n-k)$$

$$y[n] = a_1 y[n-1] + \ldots + a_N y[n-N] + b_0 x[n] + b_1 x[n-1] + \ldots + b_M x[n-M]$$

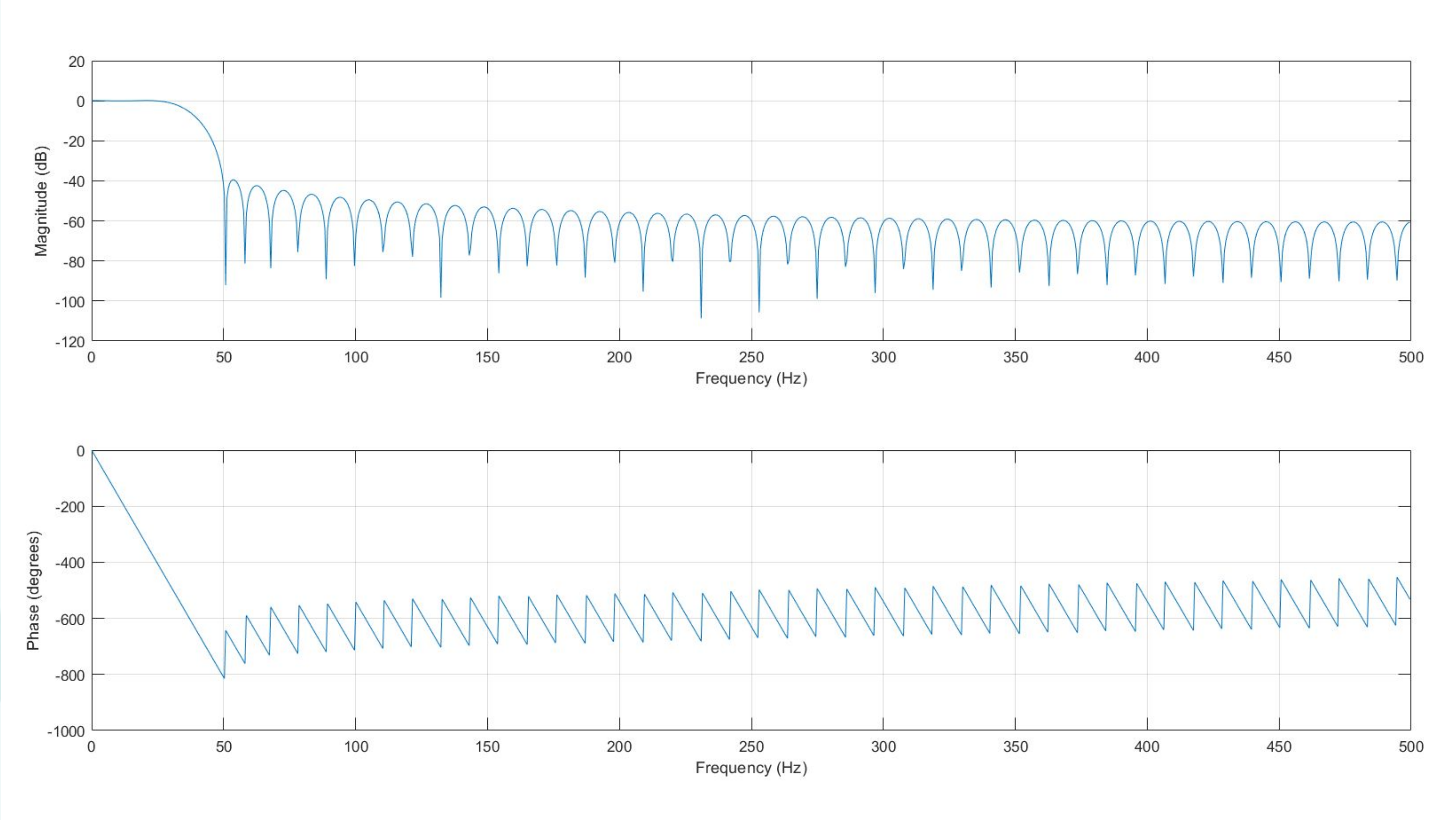
ak = 0 for FIR filter

Low Pass Filter

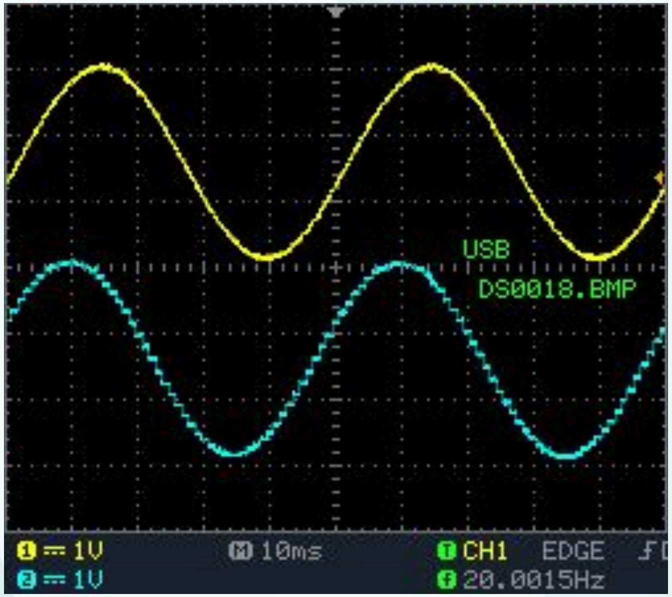


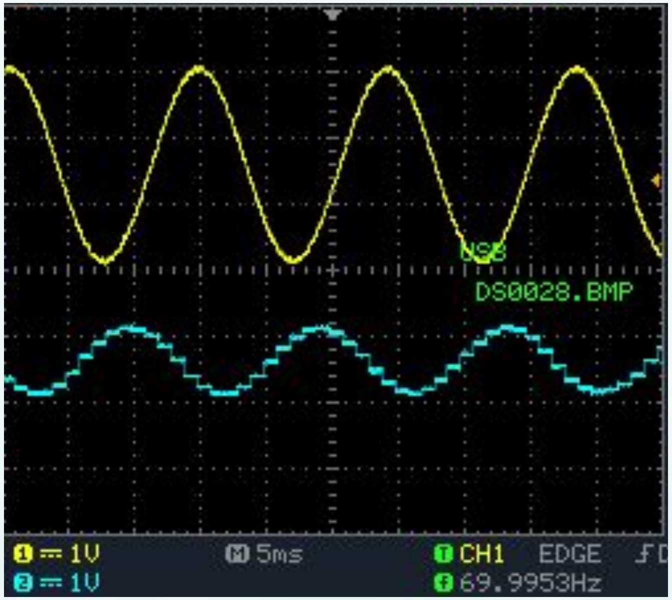
```
%------------- Low Pass Filter (LPF)
% sampling frequency [Hz]
fsamp = 1000;
% stopband and passband frequencies [Hz]
fcuts = [25 50];
% ripples
devs = [0.01 0.01];

% low pass filter
mags = [1 0];

% get kaiser window
[n,Wn,beta,ftype] = kaiserord(fcuts,mags,devs,fsamp);
% calculate coefficients
hh = fir1(n,Wn,ftype,kaiser(n+1,beta),'noscale');

figure('Name','Low Pass Filter');
freqz(hh,1,1024,fsamp)

fprintf("M [%d]", n);
coefs = regexprep(num2str(hh),'\s+',',')
```

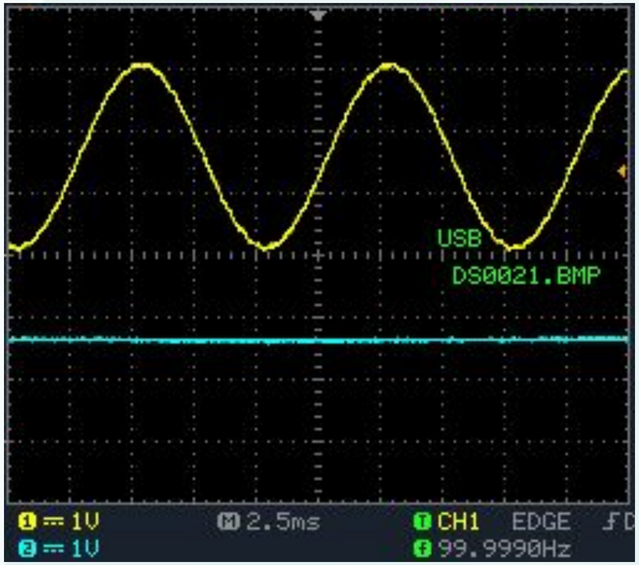
Low Pass Filter



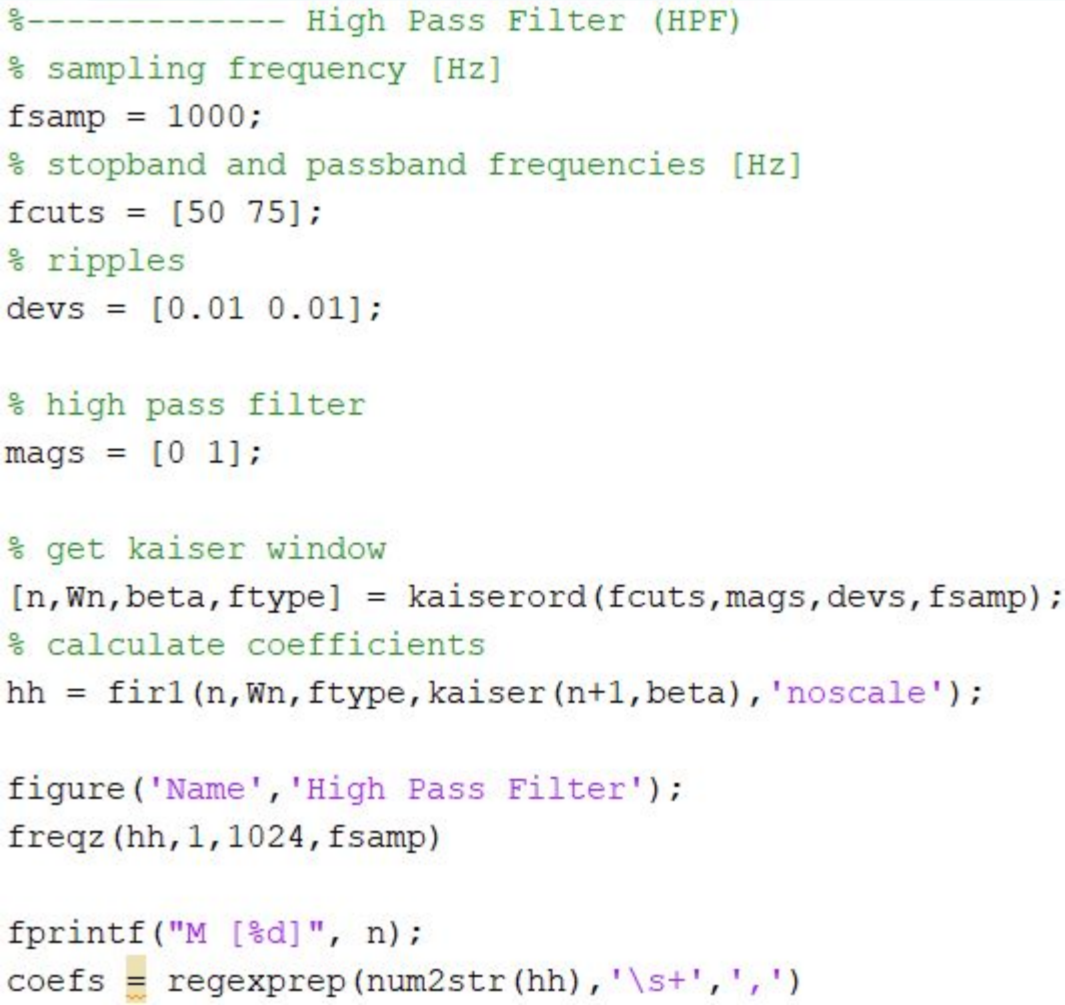
Low Pass Filter





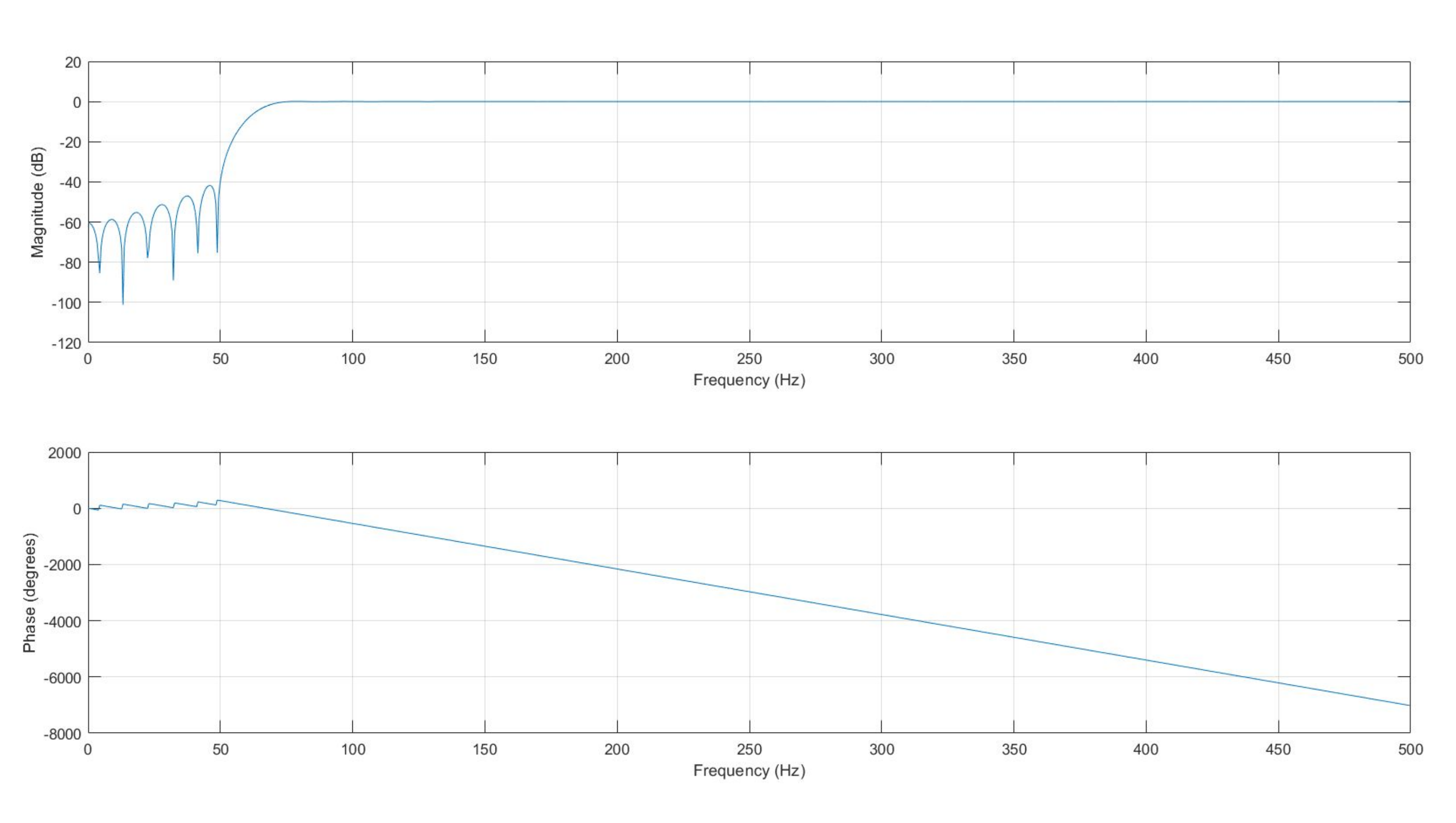


High Pass Filter

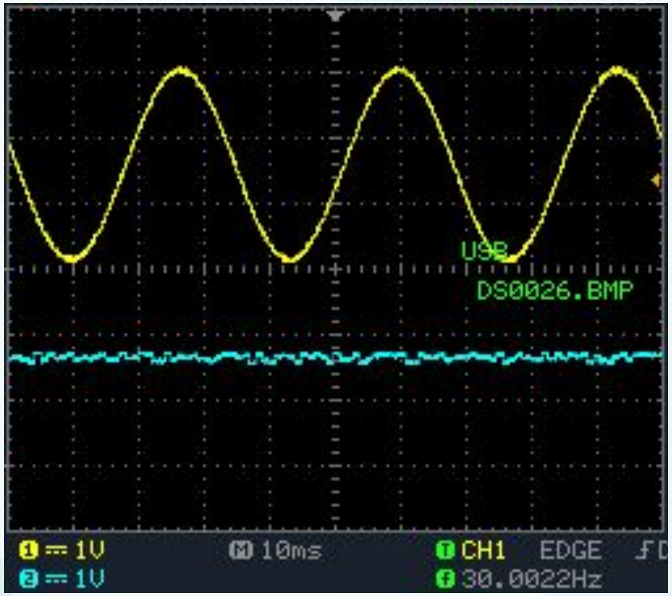


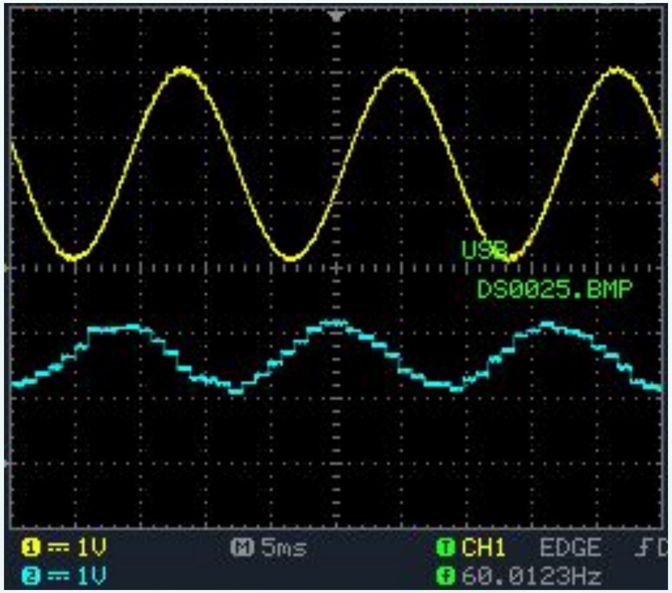
```
%------------- High Pass Filter (HPF)
% sampling frequency [Hz]
fsamp = 1000;
% stopband and passband frequencies [Hz]
fcuts = [50 75];
% ripples
devs = [0.01 0.01];

% high pass filter
mags = [0 1];

% get kaiser window
[n,Wn,beta,ftype] = kaiserord(fcuts,mags,devs,fsamp);
% calculate coefficients
hh = fir1(n,Wn,ftype,kaiser(n+1,beta),'noscale');

figure('Name','High Pass Filter');
freqz(hh,1,1024,fsamp)

fprintf("M [%d]", n);
coefs = regexprep(num2str(hh),'\s+',',')
```

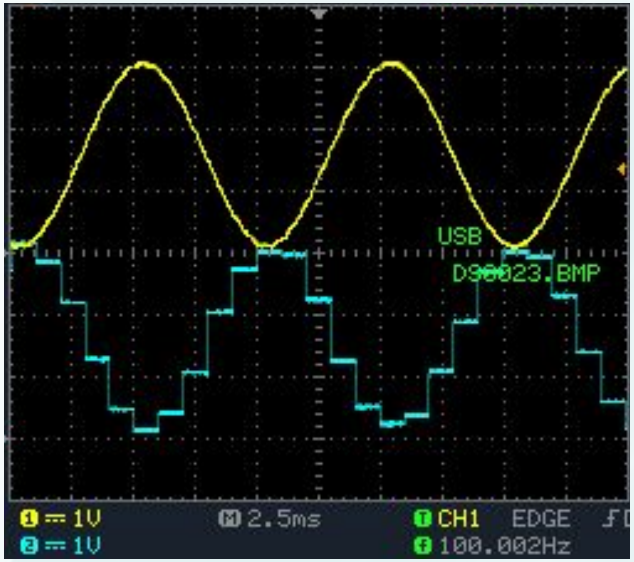
High Pass Filter



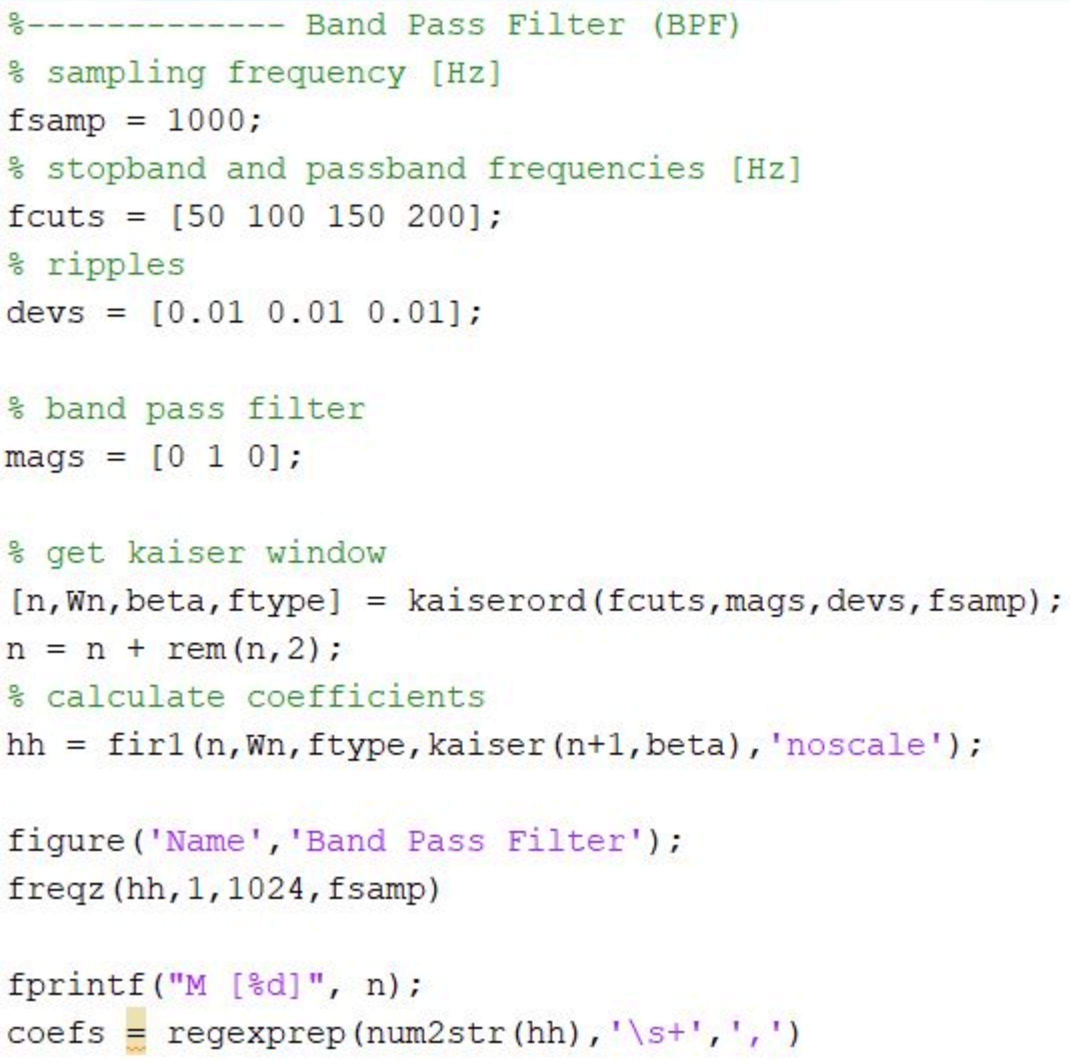
High Pass Filter





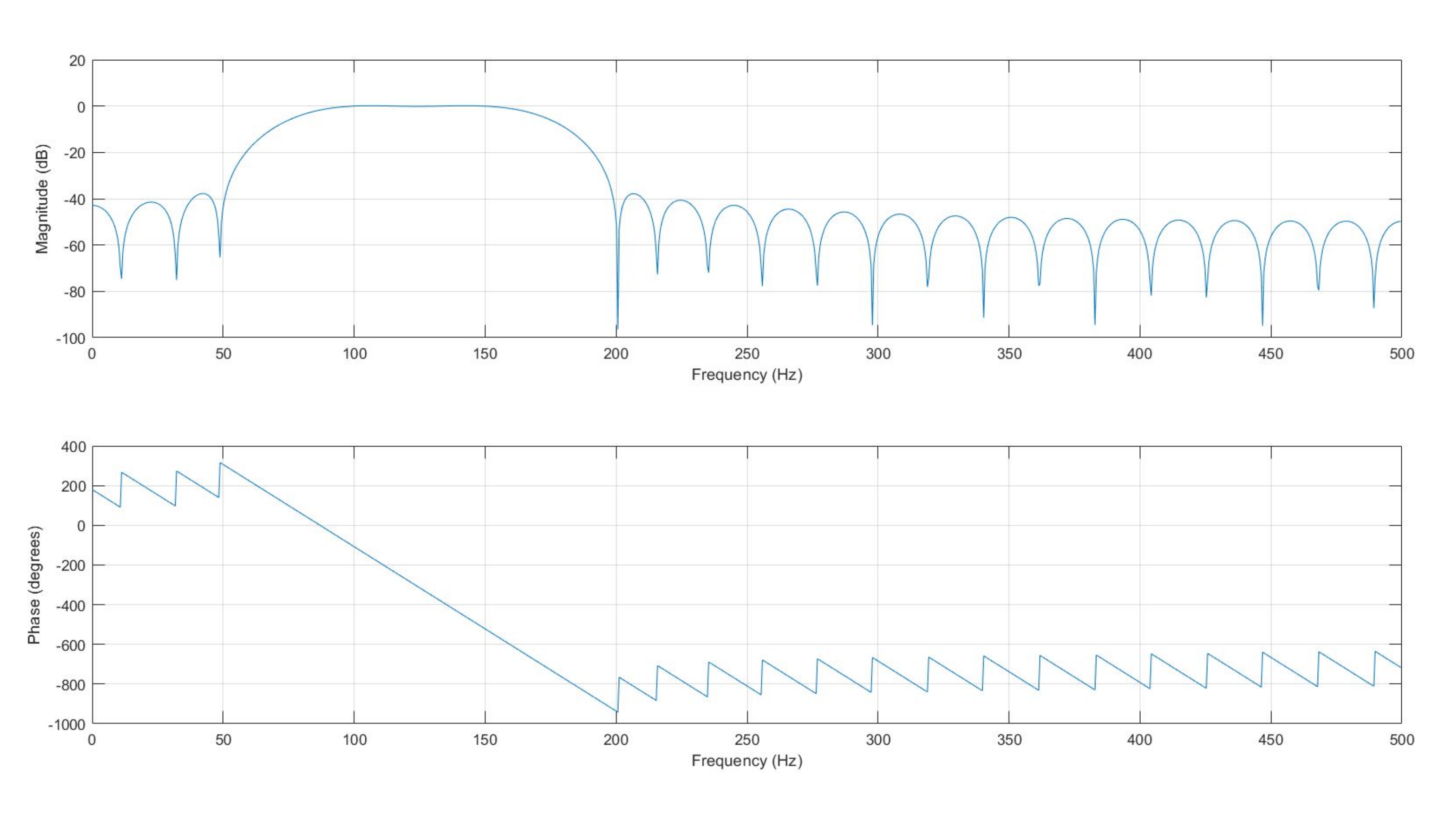


Band Pass Filter

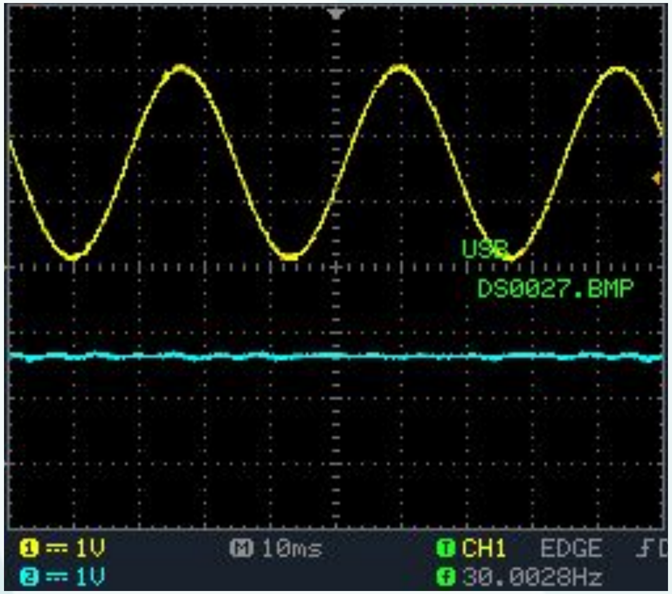


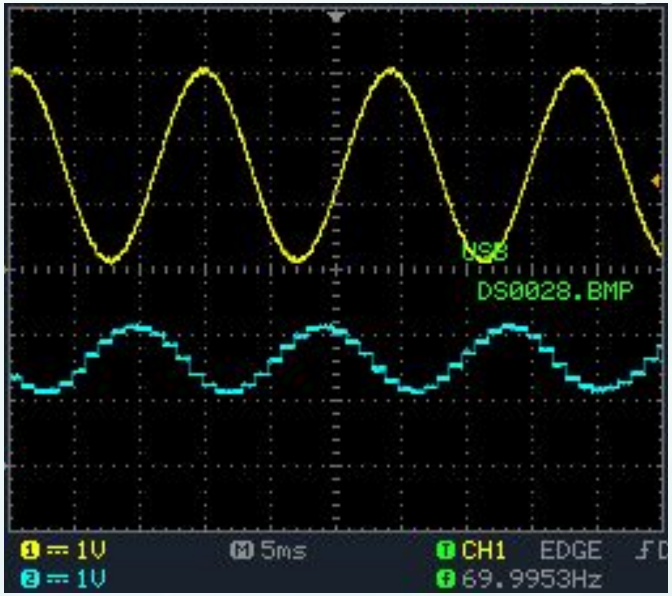
```
%------------- Band Pass Filter (BPF)
% sampling frequency [Hz]
fsamp = 1000;
% stopband and passband frequencies [Hz]
fcuts = [50 100 150 200];
% ripples
devs = [0.01 0.01 0.01];

% band pass filter
mags = [0 1 0];

% get kaiser window
[n,Wn,beta,ftype] = kaiserord(fcuts,mags,devs,fsamp);
n = n + rem(n,2);
% calculate coefficients
hh = fir1(n,Wn,ftype,kaiser(n+1,beta),'noscale');

figure('Name','Band Pass Filter');
freqz(hh,1,1024,fsamp)

fprintf("M [%d]", n);
coefs = regexprep(num2str(hh),'\s+',',')
```

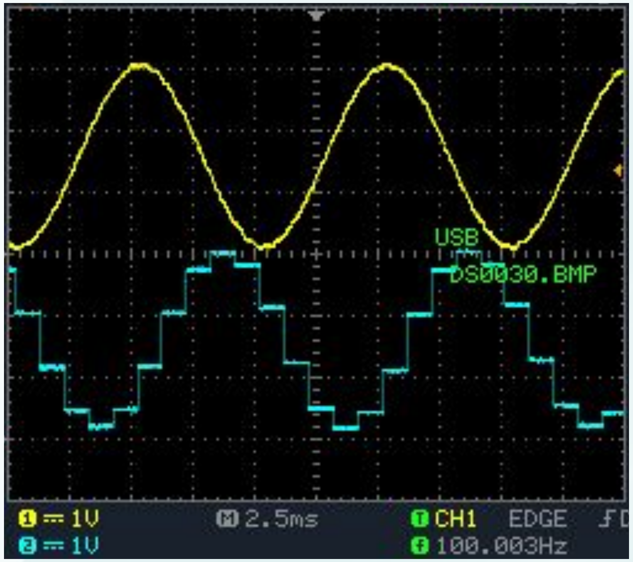
Band Pass Filter



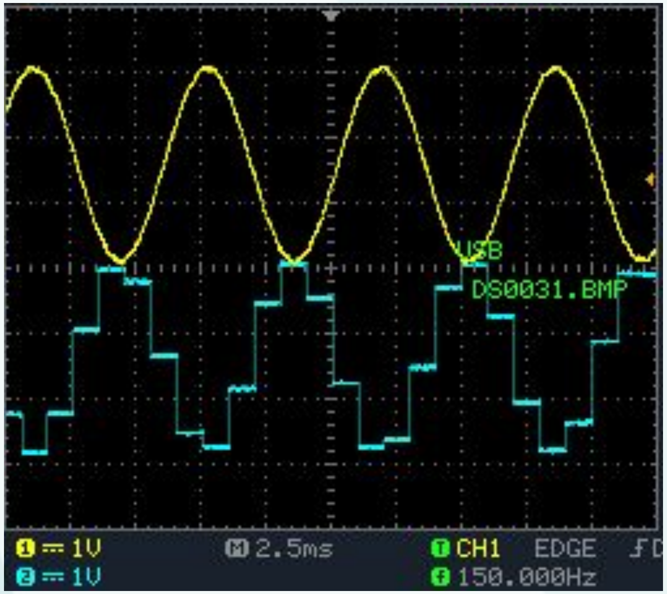
Band Pass Filter

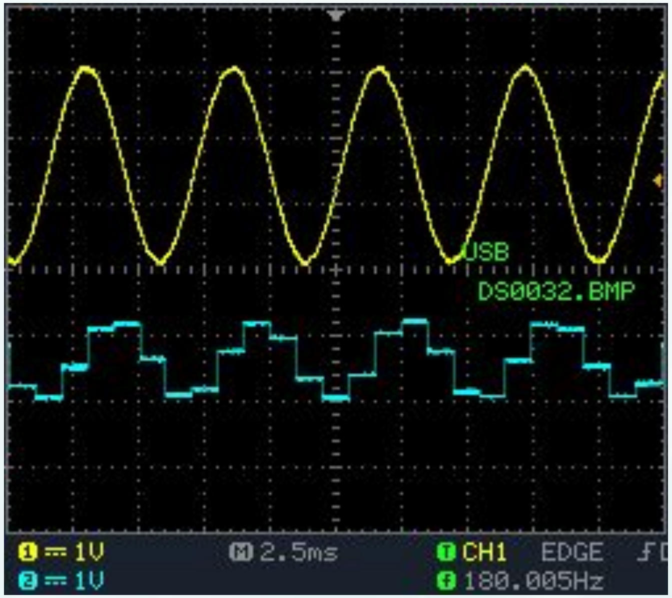


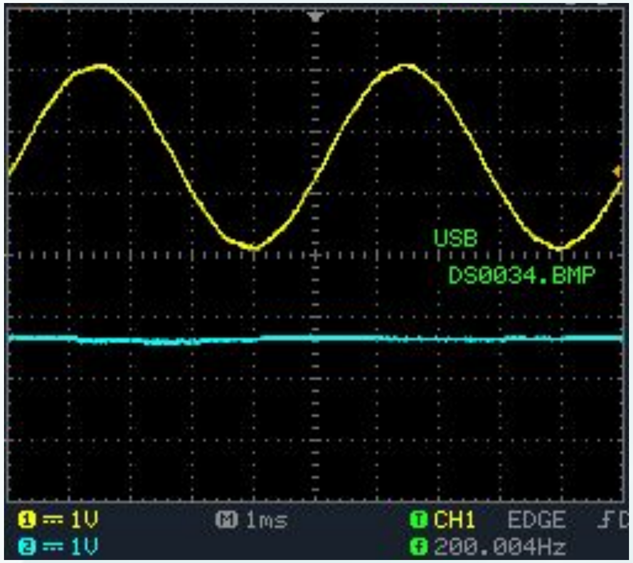




Band Pass Filter







Questions?