function [coeff\_mat,freq\_resp\_filt\_mat,fgrid] = third\_octave\_filters(f0,fs)

% ALMOST IDENTICAL TO A CONST Q FAM SYSTEM AND octave\_filters.m

% THIS SYSTEM IS IMPLEMENTING A MATRIX OF THIRD OF AN OCTAVE FILTERS

if(nargin == 0)

fs = 48000; % function sets fs = 48kHz as default value (in case wasn't passed)

fc = 27.5; % function sets fc = 27.5 (the frequency corresponding to musical note A0) if wasn't passed

elseif(nargin == 1)

fs = 48000; % sampeling freq

fc = f0; % fc0

else

fc = f0;

end

f\_bottom = 20; % bottom of spec

f\_top = 20\*10^3; % top of spec

fc\_vec = fc\*2.^([0:24]./3)

Q = 1/((2)^(1/2)); % butterworth quality factor;

fb\_vec = fc\_vec./Q;

N = fs; % number of evaluation points

freq\_resp\_filt\_mat = []; % a matrix to hold frequency response of each filter

coeff\_mat = []; % a matrix to hold each filter's coeffs: h0: b00 | b10 | b20...

clf; figure(1); % ----------------

fgrid = fs\*(0:(N-1))/(N); % a00 | a10 | a20...

hold on; % ----------------

xlabel("f[Hz] {\copyright}ROT"); ylabel("|H(f)|"); grid on; % h1: b01 | b11 | b21...

title("One Third of an Octave-spread Butterworth Band Pass Filters"); axis([f\_bottom,f\_top,0,1]); % a01 | a11 | a21...etc.

for i = [1:length(fc\_vec)]

[b,a] = butter(2,[fc\_vec(i) - fb\_vec(i)/2, fc\_vec(i) + fb\_vec(i)/2]./(fs/2),"bandpass");

coeff\_mat = [coeff\_mat;b;a]; % add vectors b,a to the coeff matrix

[h,~] = freqz(b,a,fs); % calc frequency response of filter

freq\_resp\_filt\_mat = [freq\_resp\_filt\_mat;h']; % add frequency resp vector to the filter matrix

plot(fgrid,abs(h),'black');

end

xline(fc\_vec,':'); yline(0.707, ':'); % mark center frequencies and -3db

hold off;

end