if ~any(size(Wind) <= 1
    ;'msg = 'The window specified must be a vector
    return
else
    Wind = Wind(:).'; % Make it a row vector
end

function [nbands,ff,Ftype,msg] = desiredfreq(Wn,Ftype
    % DESIREDFREQ  Compute the vector of frequencies to pass to FIRLS%
    % Inputs
    %  Wn    - vector of cutoff frequencies
    %  Ftype - string with desired response ('low','high
    % Outputs
    %  nbands - number of frequency bands
    %  ff     - vector of frequencies to pass to FIRLS
    %  Ftype  - converted filter type (if it's necessary to
    % Initialize output args
    ;[ ] = nbands
    ;[ ] = ff
    ;' ' = msg

    ,( if any( Wn<0 | Wn>1
        ;'msg = 'Frequencies must fall in range between 0 and 1
        return
    end
    ,(if any(diff(Wn)<0
        ;'msg = 'Frequencies must be increasing
        return
    end
    ;'( :)Wn = Wn
    ;nbands = length(Wn) + 1
    ,(if (nbands > 2) && strcmpi(Ftype,'bandpass
        Ftype = 'DC-0'; % make sure default 3 band filter is bandpass
    end
    ;[ff = [0,Wn(1:nbands-1); Wn(1:nbands-1),1
    ;( :)ff = ff

    % DESIREDMAG  Compute the magnitude vector to pass to FIRLS%
    % Inputs
    %  Ftype - string with desired response ('low','high
    % Outputs
    %  aa     - filter coefficients
    %  First_Band - first band of the filter
    (function [aa,First_Band] = desiredmag(Ftype,nbands

5-1