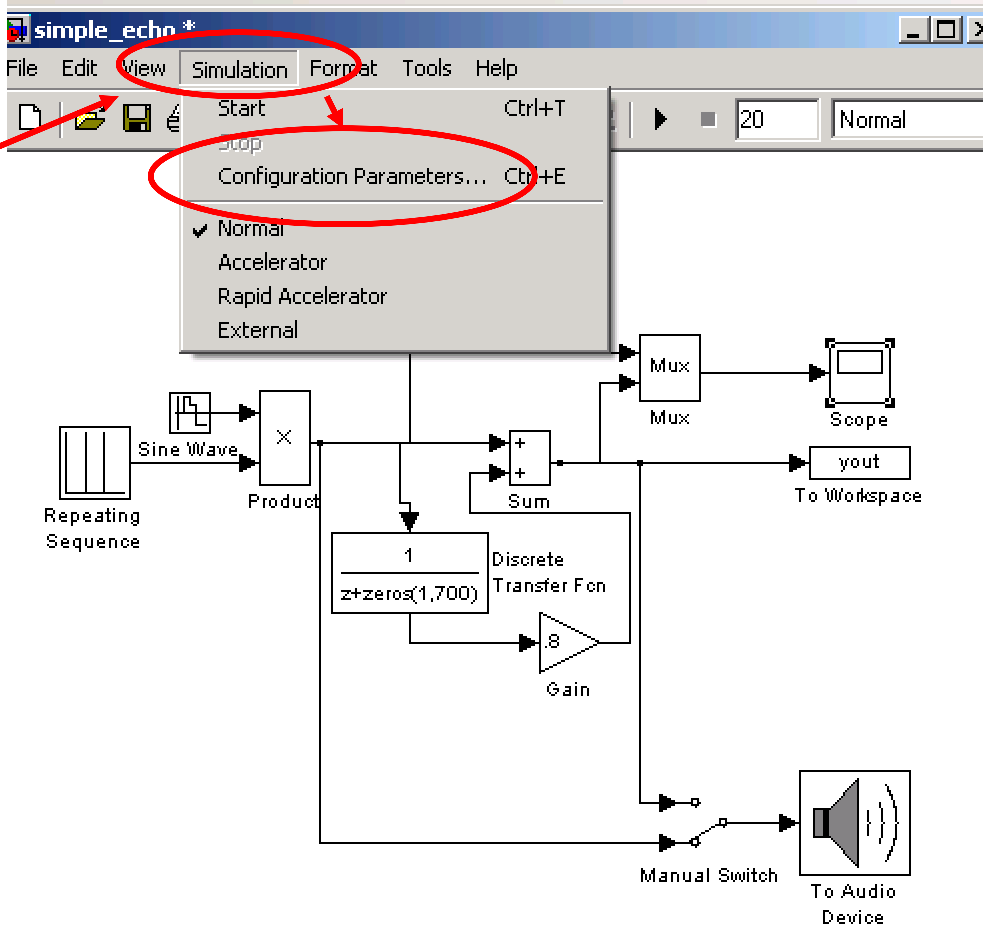
Using Simulink in Signal Processing Applications

Basic Simulink blocks discussed

- How to:
  - 1) Specify configuration parameters
  - 2) Read data in from workspace
  - 3) Read data in from multimedia file
  - 4) Listen to a sound file
  - 5) Save data to multimedia file
  - 6) Save data to workspace
  - 7) Specify IIR/FIR discrete filter characteristics
  - 8) Specify internal input data
  - 9) Plot using Scope blocks
  - 10) Implement the LMS algorithm in Simulink
  - 11) Implement the RLS algorithm in Simulink
  - 12) Plot the filter coefficients using the vector scope
  - 13) Plot multiple data streams on the same figure
  - 14) Generate spectrum and spectrogram plots
  - 15) Generate frequency response plot from filter coefficients
  - 16) Listen to processed audio signals

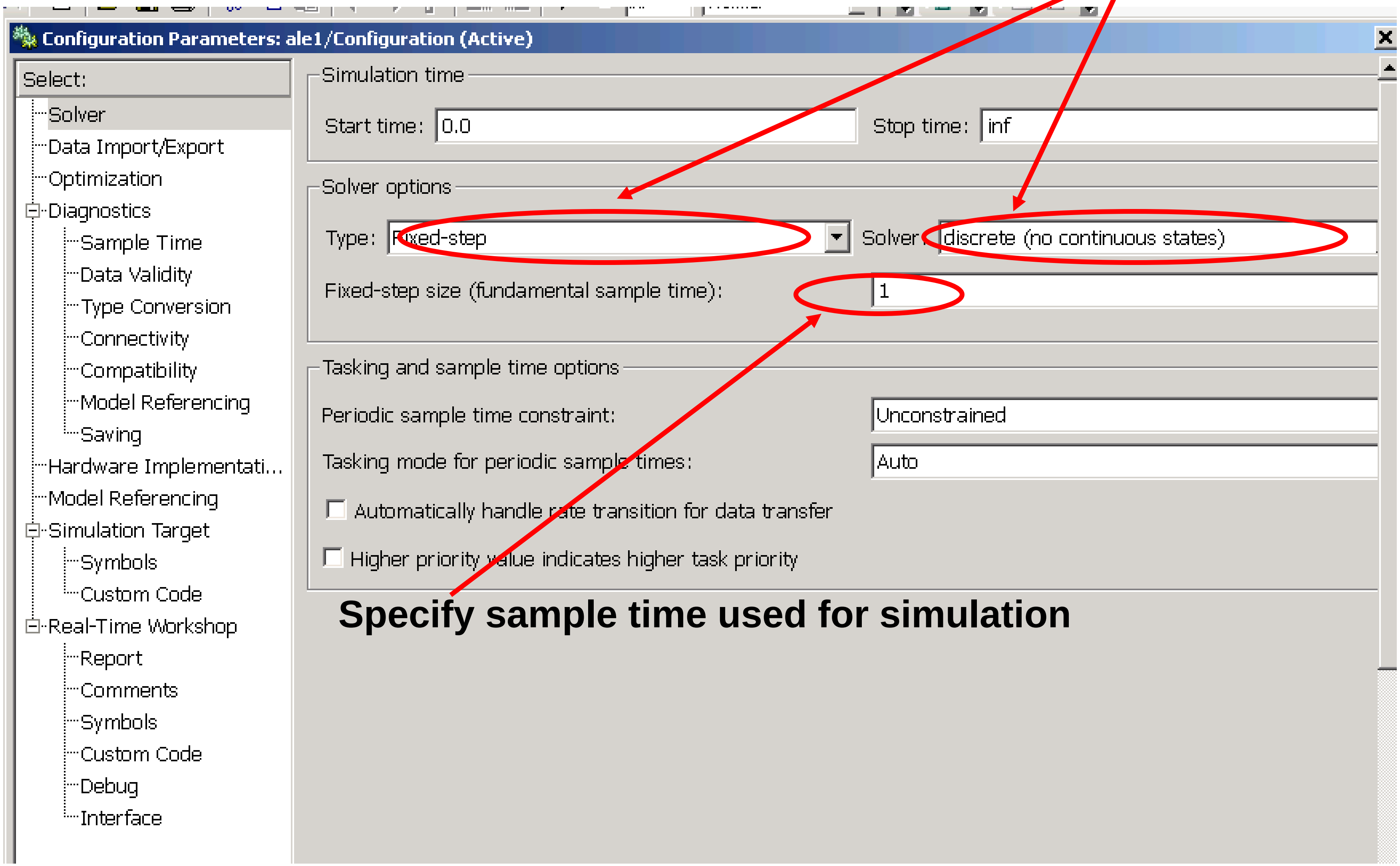
1) How to set-up configuration parameters



Check/specify configuration parameters

Select
Simulation
→ Configuration Parameters

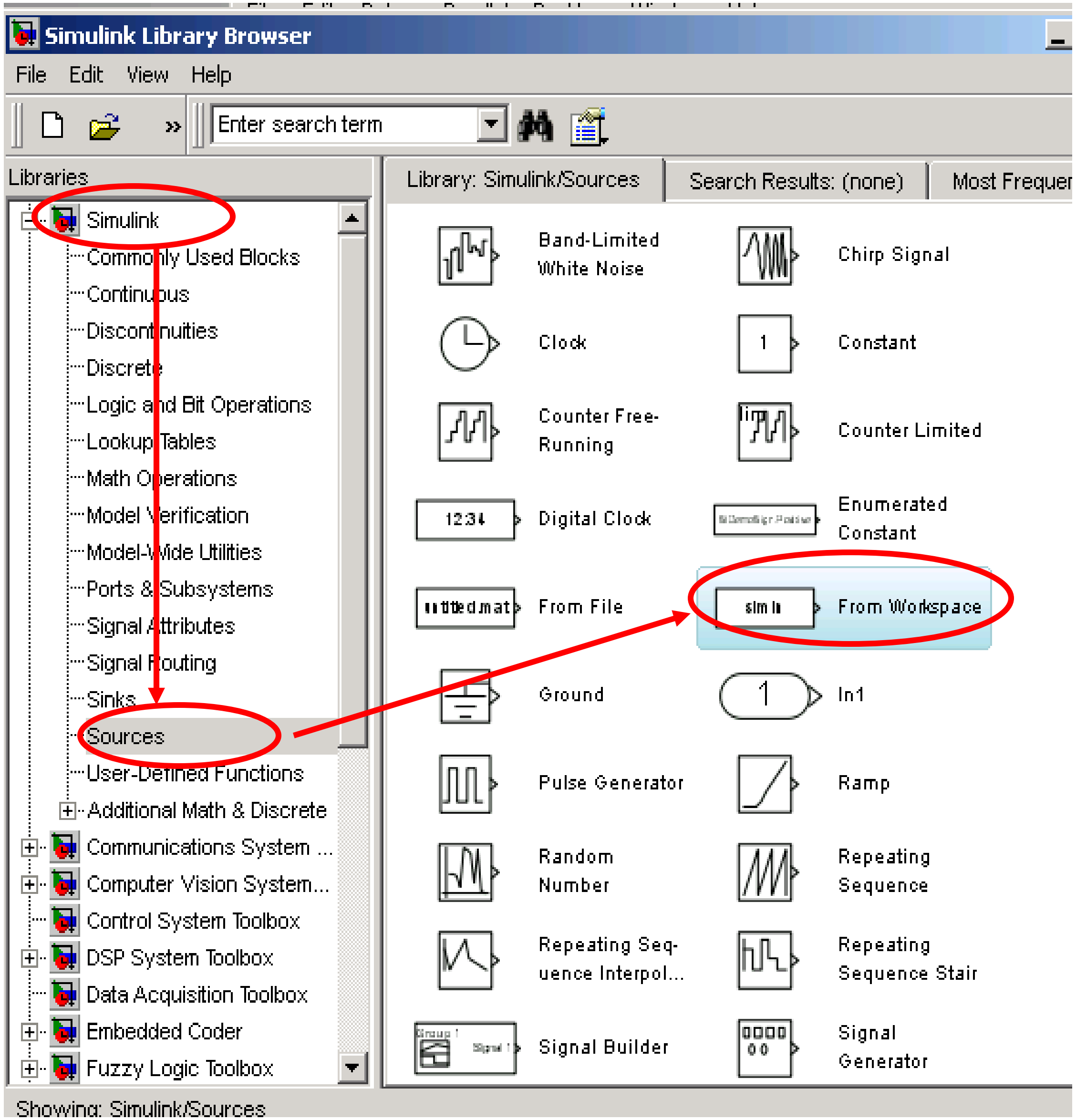
Required choice for discrete implementation

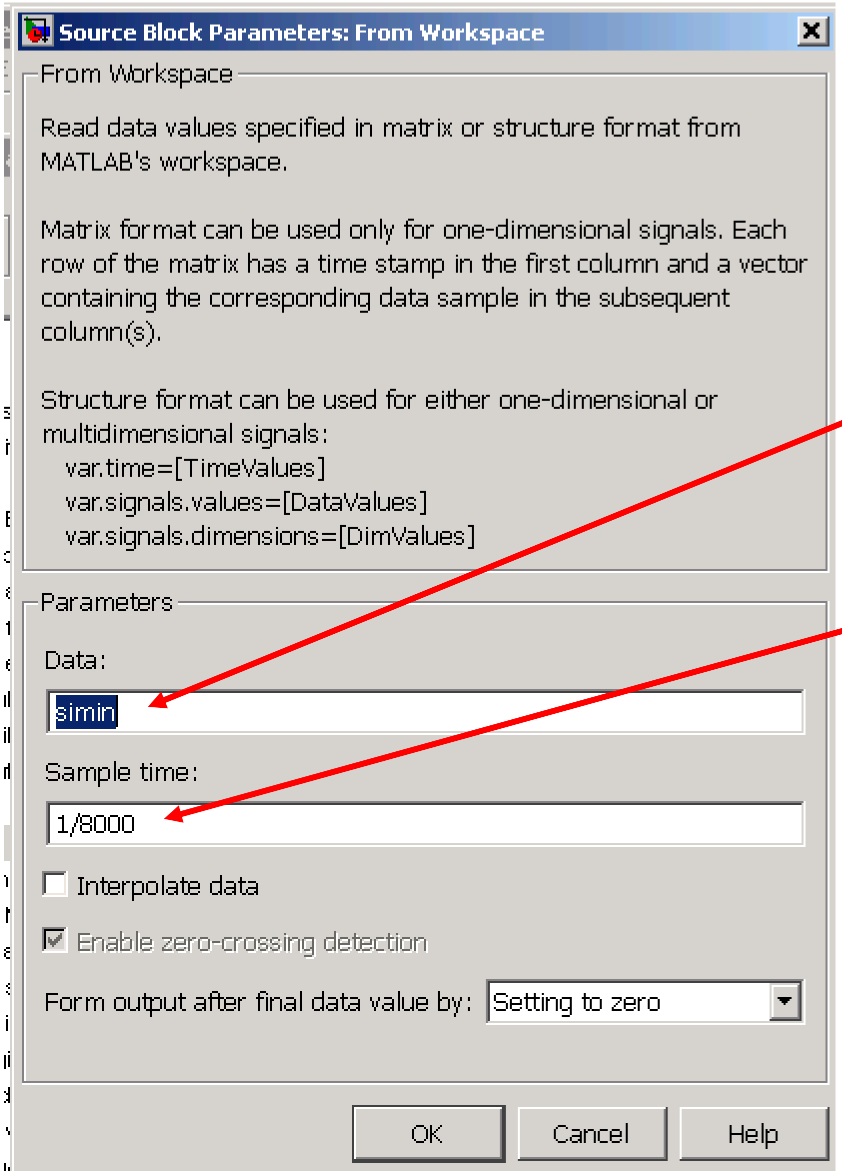


Specify sample time used for simulation

2) How to read data from the workspace

Select
Simulink
→Sources
→From Workspace

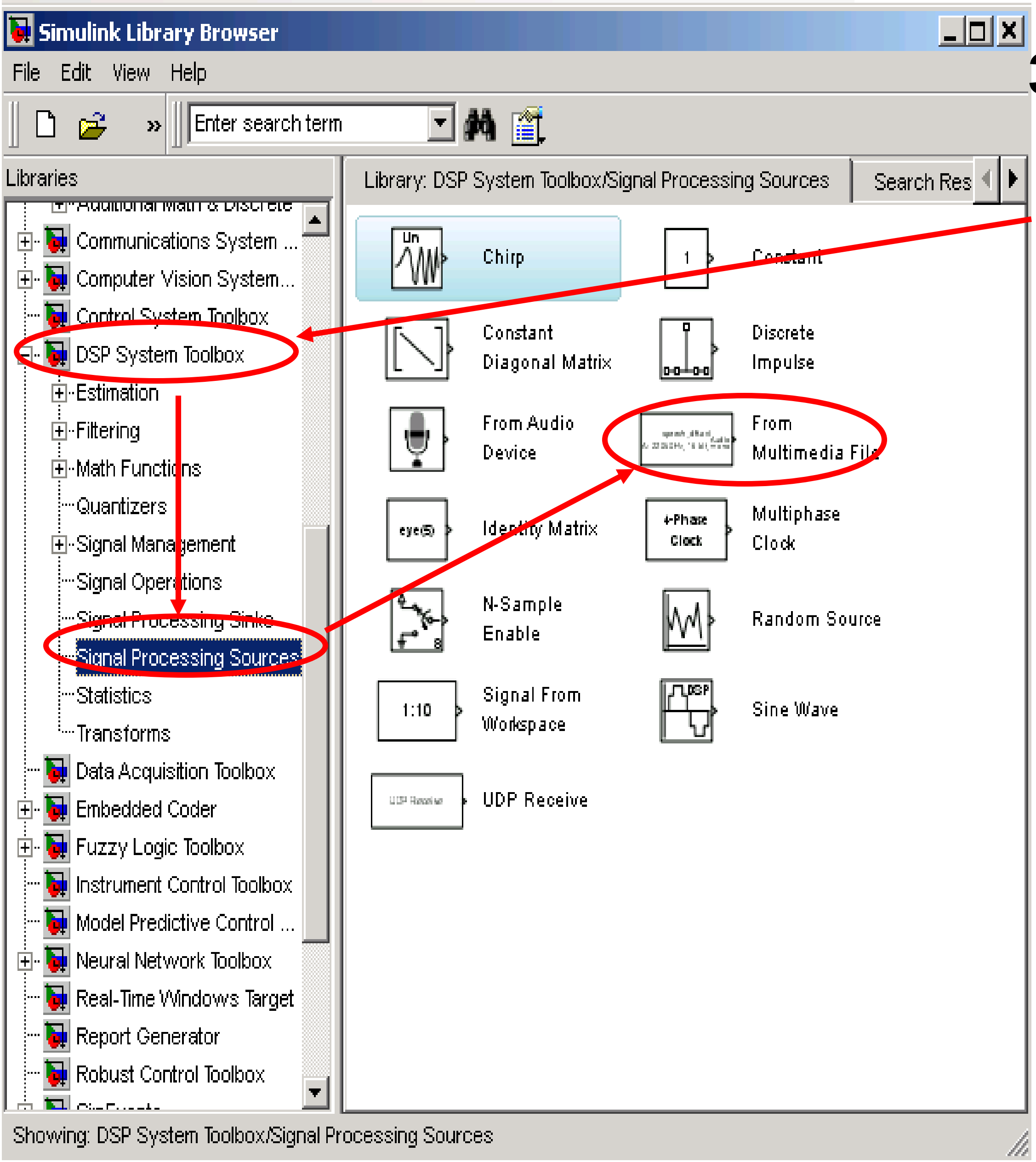




INPUT DATA FORMAT

1) **Data must be formatted as ynn2=[timesample,datasample], format: N×2**
2) **Need to define –timesample- with the correct sampling frequency**

3) How to read .wav file

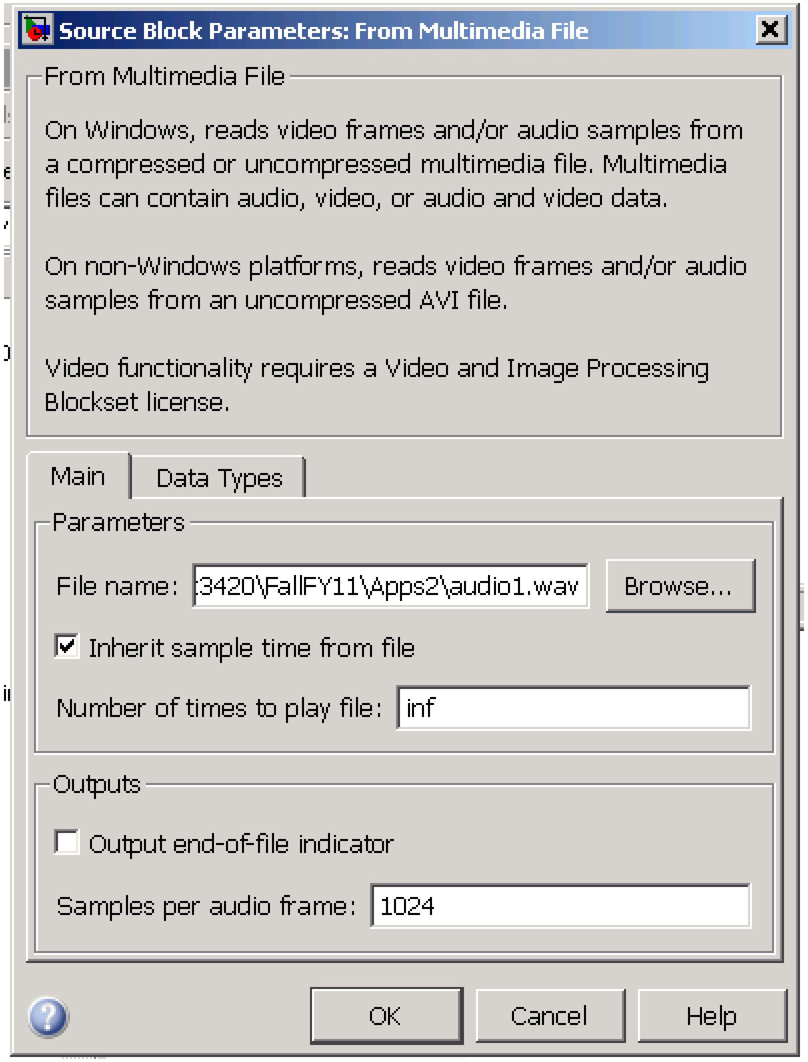


Select

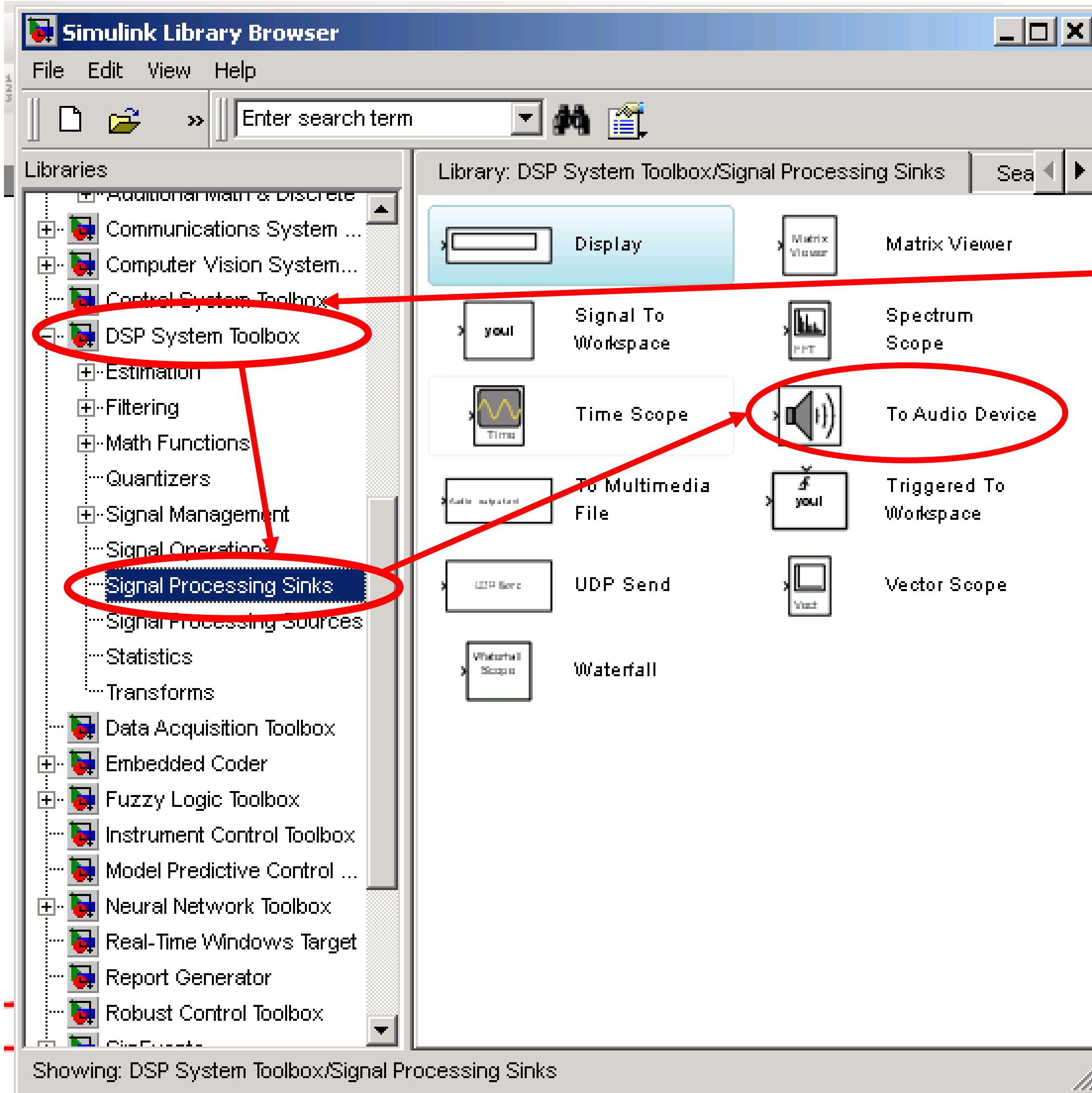
DSP System Toolbox

→ Signal Processing Sources

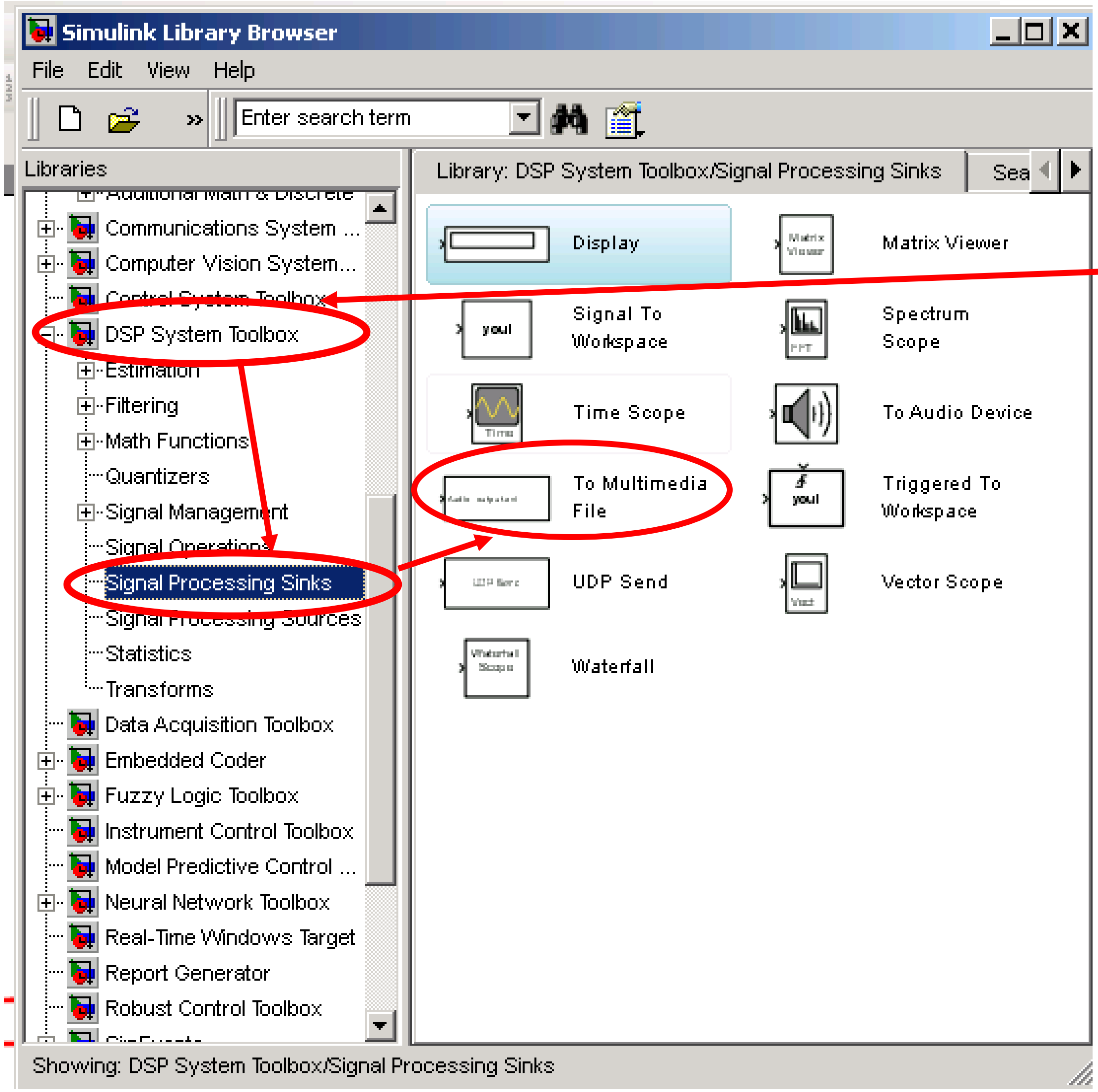
→ From Multimedia File


Source Block Parameters: From Multimedia File
From Multimedia File
On Windows, reads video frames and/or audio samples from a compressed or uncompressed multimedia file. Multimedia files can contain audio, video, or audio and video data.
On non-Windows platforms, reads video frames and/or audio samples from an uncompressed AVI file.
Video functionality requires a Video and Image Processing Blockset license.
Main
Data Types
Parameters
File name:
:3420\FallFY11\Apps2\audio1.wav
Browse...
Inherit sample time from file
Number of times to play file:
inf
Outputs
Output end-of-file indicator
Samples per audio frame:
1024
OK
Cancel
Help

4) How to listen to a sound file



Select
DSP System Toolbox
→ Signal Processing Sinks
→ To Audio Device



5) Save data to a multimedia file

Select

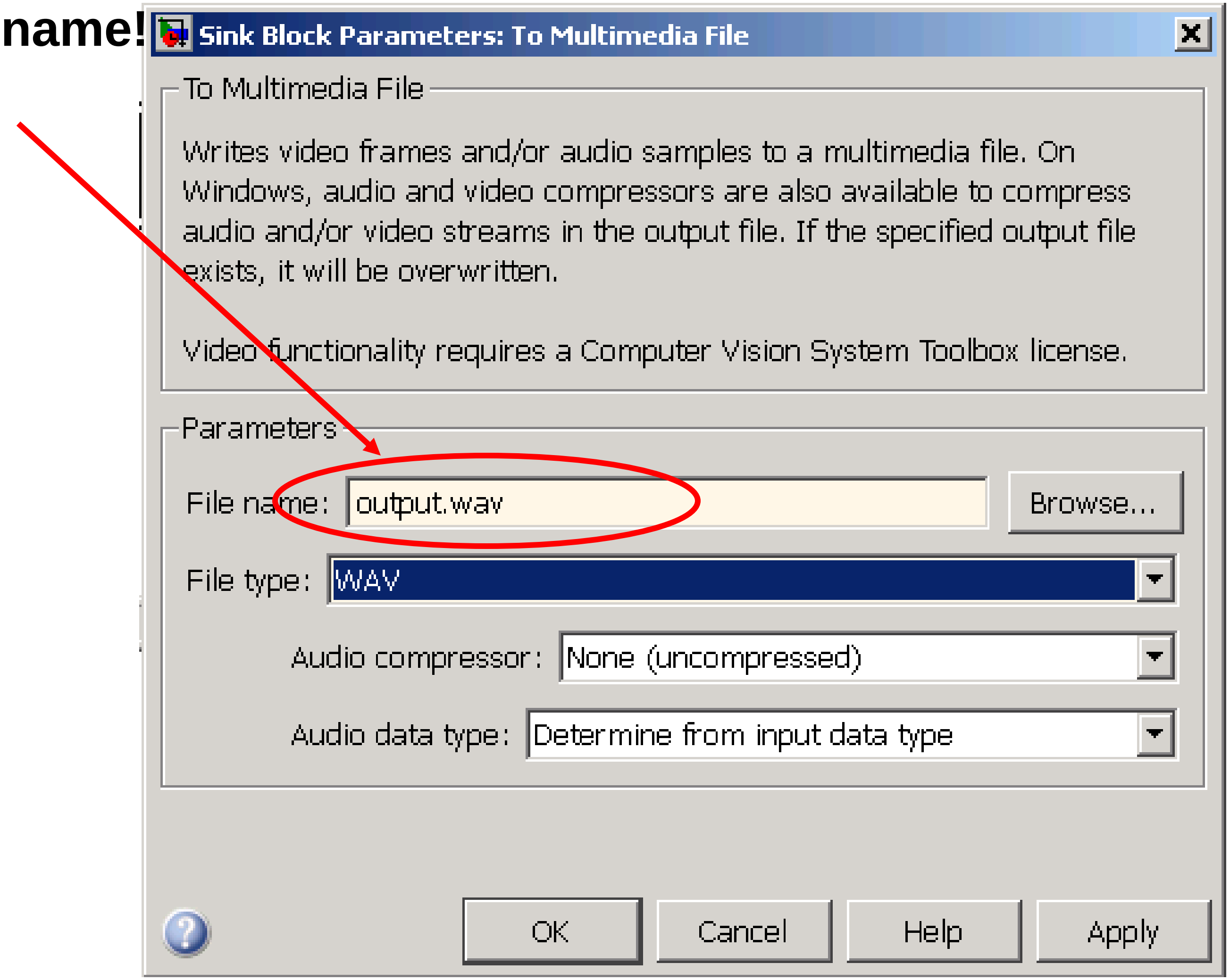
DSP System Toolbox

→ Signal Processing Sinks

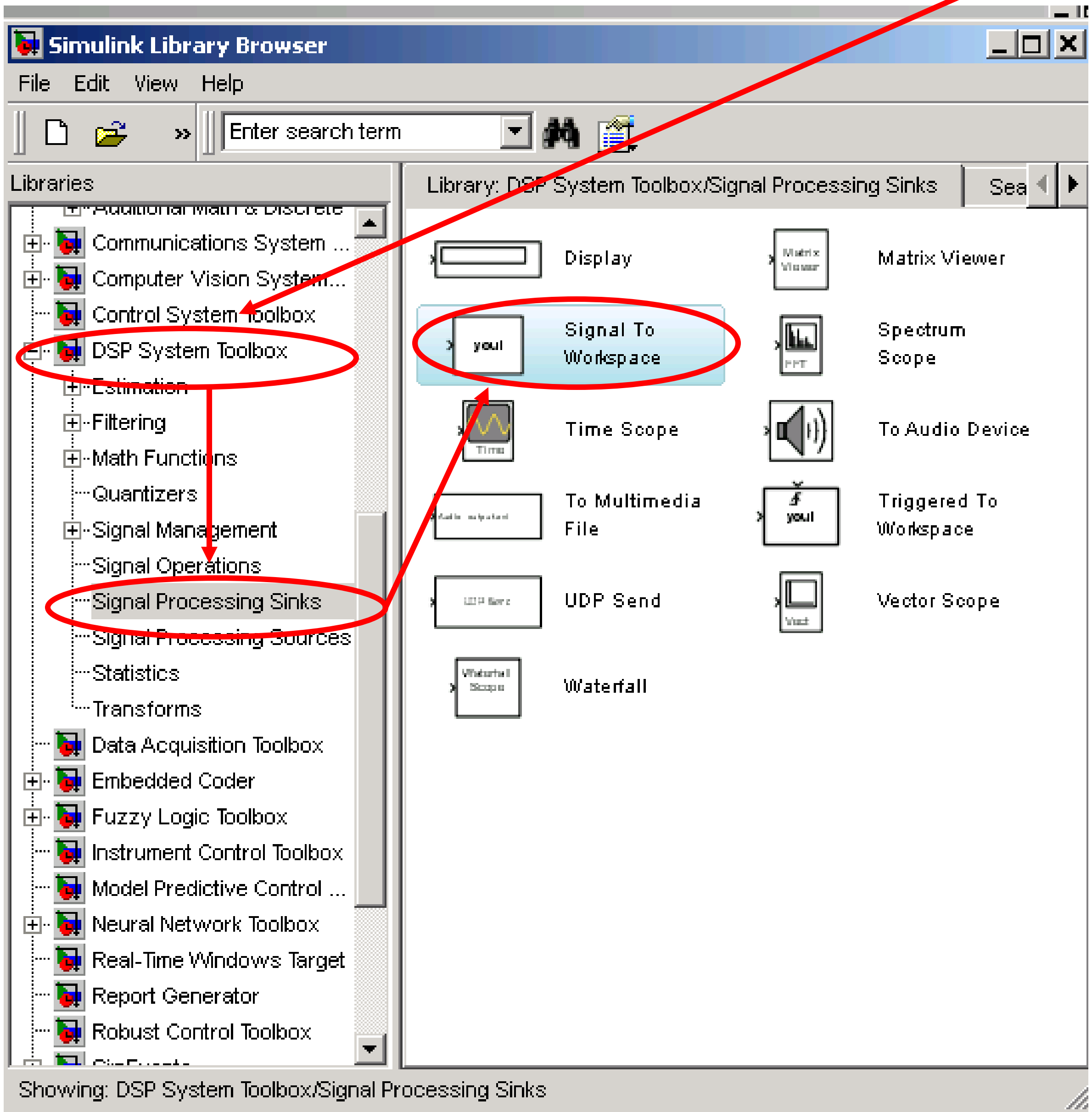
→ To Multimedia File

Output to Wave device

Need to define file name!



6) Save data to workspace



Select

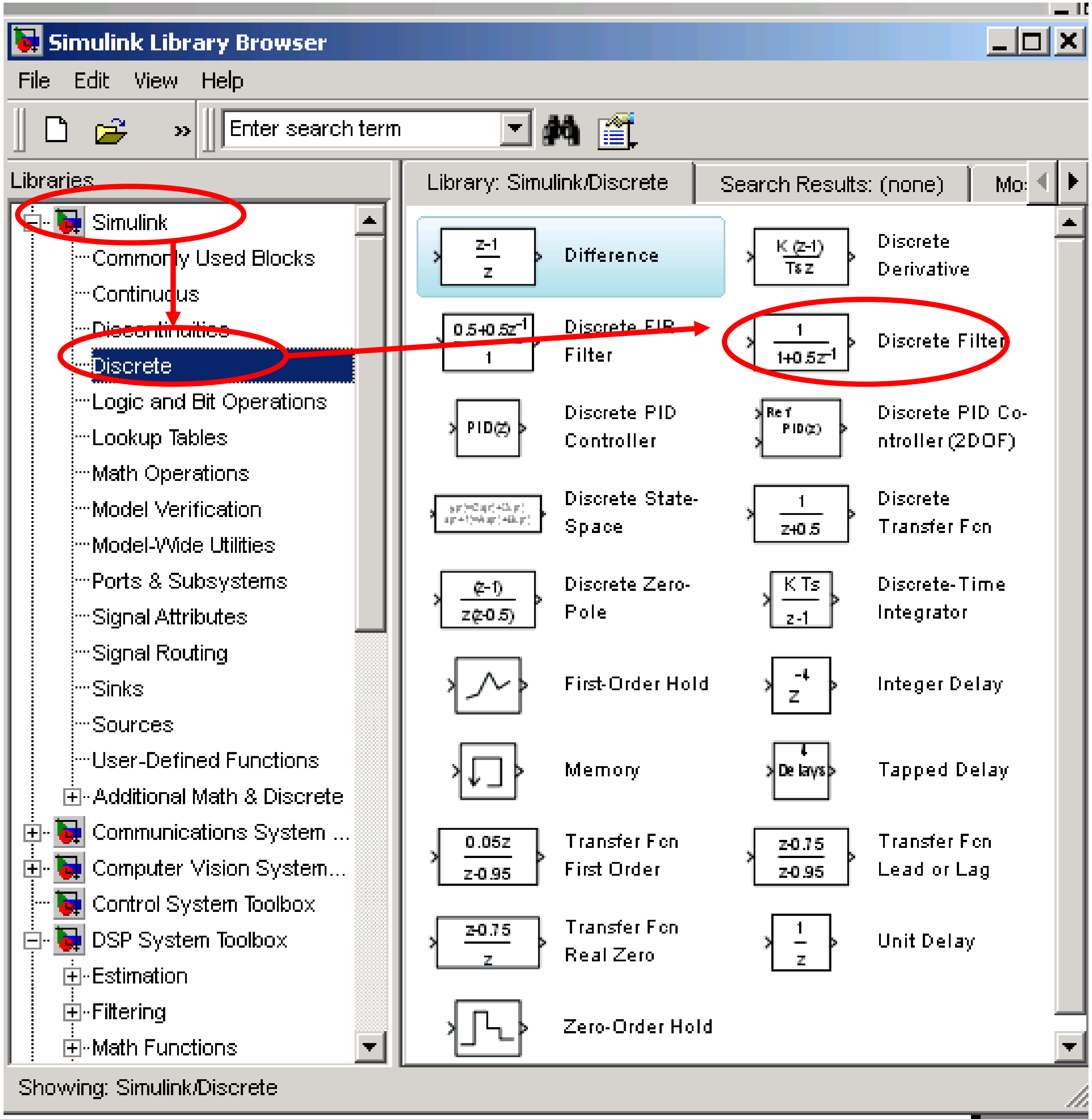
DSP System Toolbox

→ Signal Processing Sinks

→ Signal To Workspace

7) Specify IIR/FIR Filter characteristics

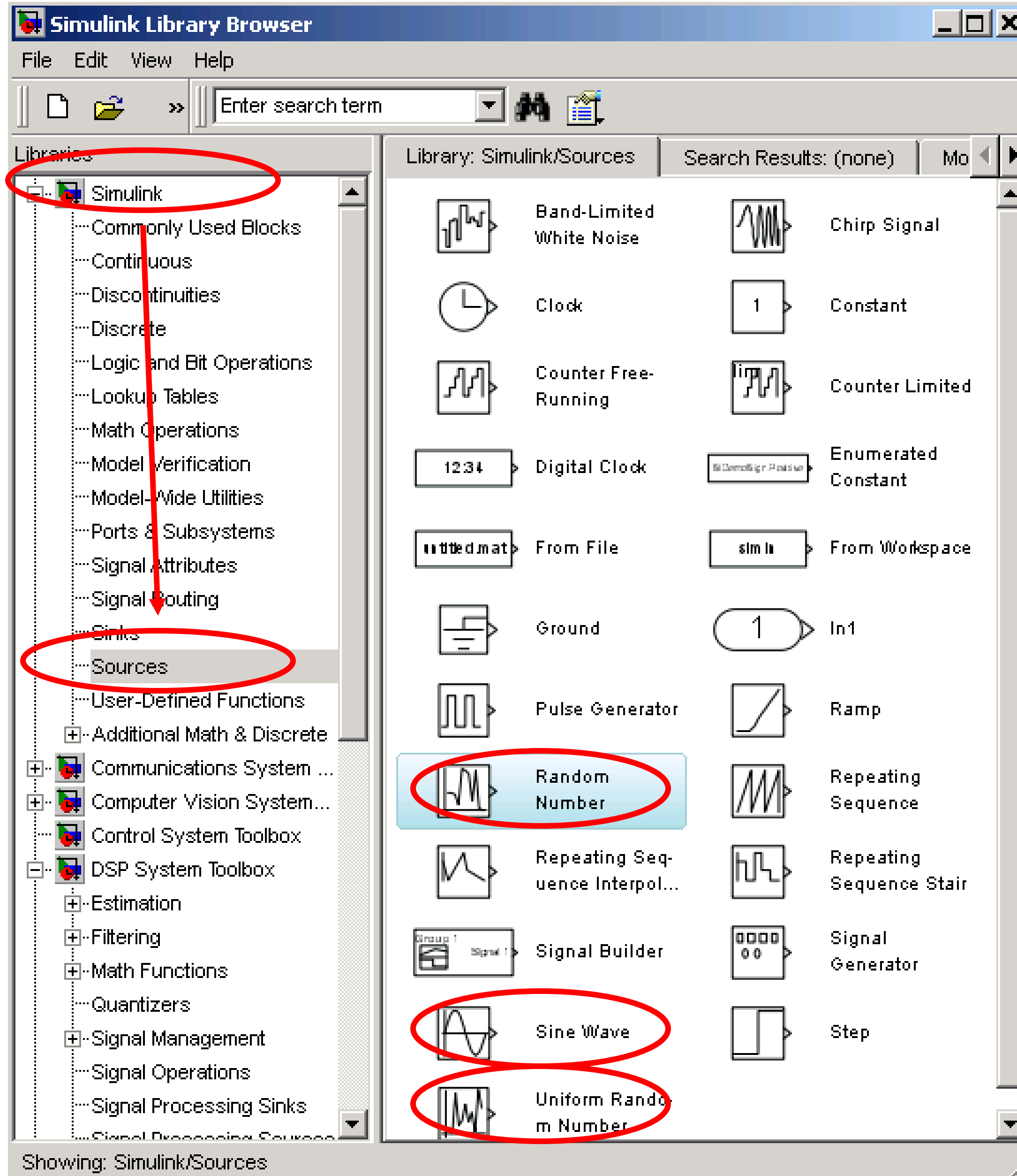
Select
Simulink
→ Discrete
→ Discrete Filter



8) Specify internal input data

Specific parameters specified within each block

Select
Simulink
→ Sources



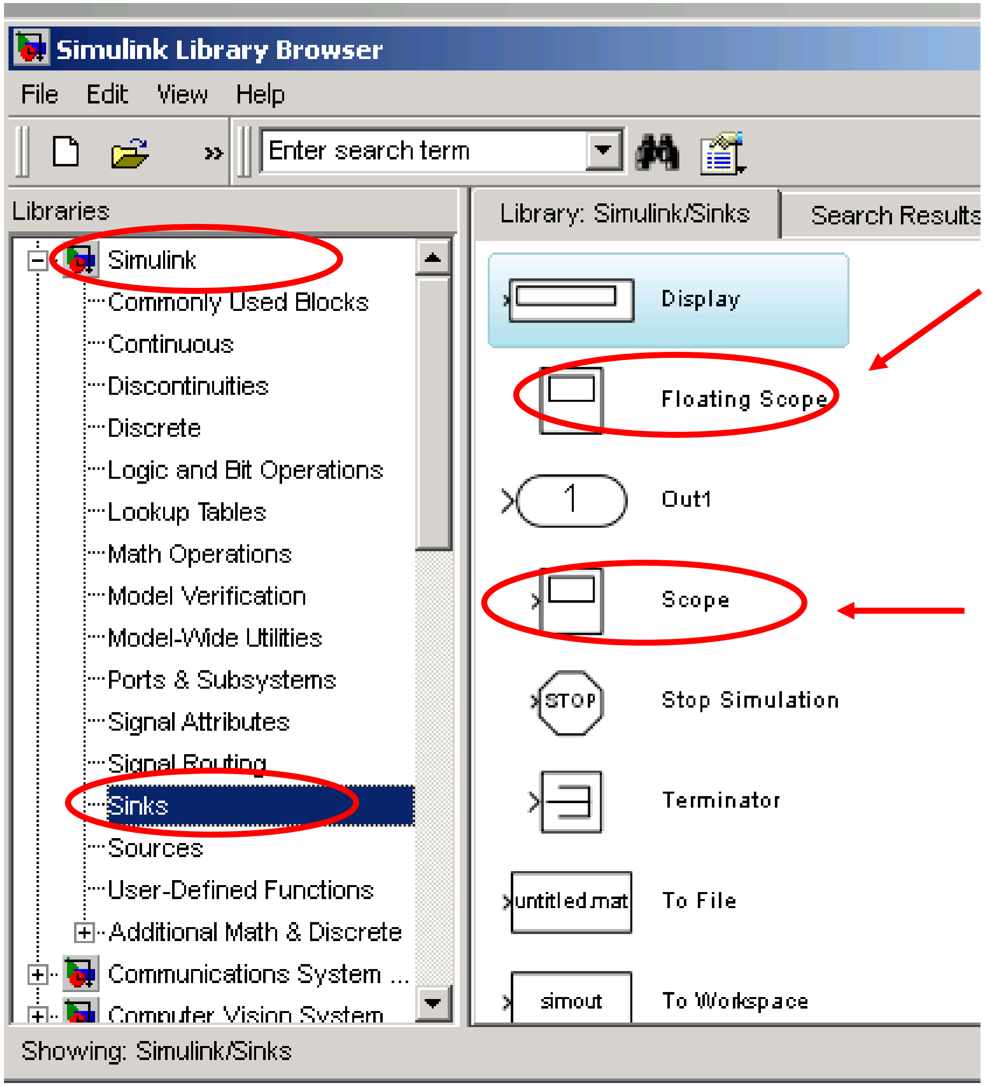
→ White Gaussian Noise

→ Sinewave

→ Uniform Random Noise

9) Plot data using Scope blocks

Select
Simulink
→ Sinks

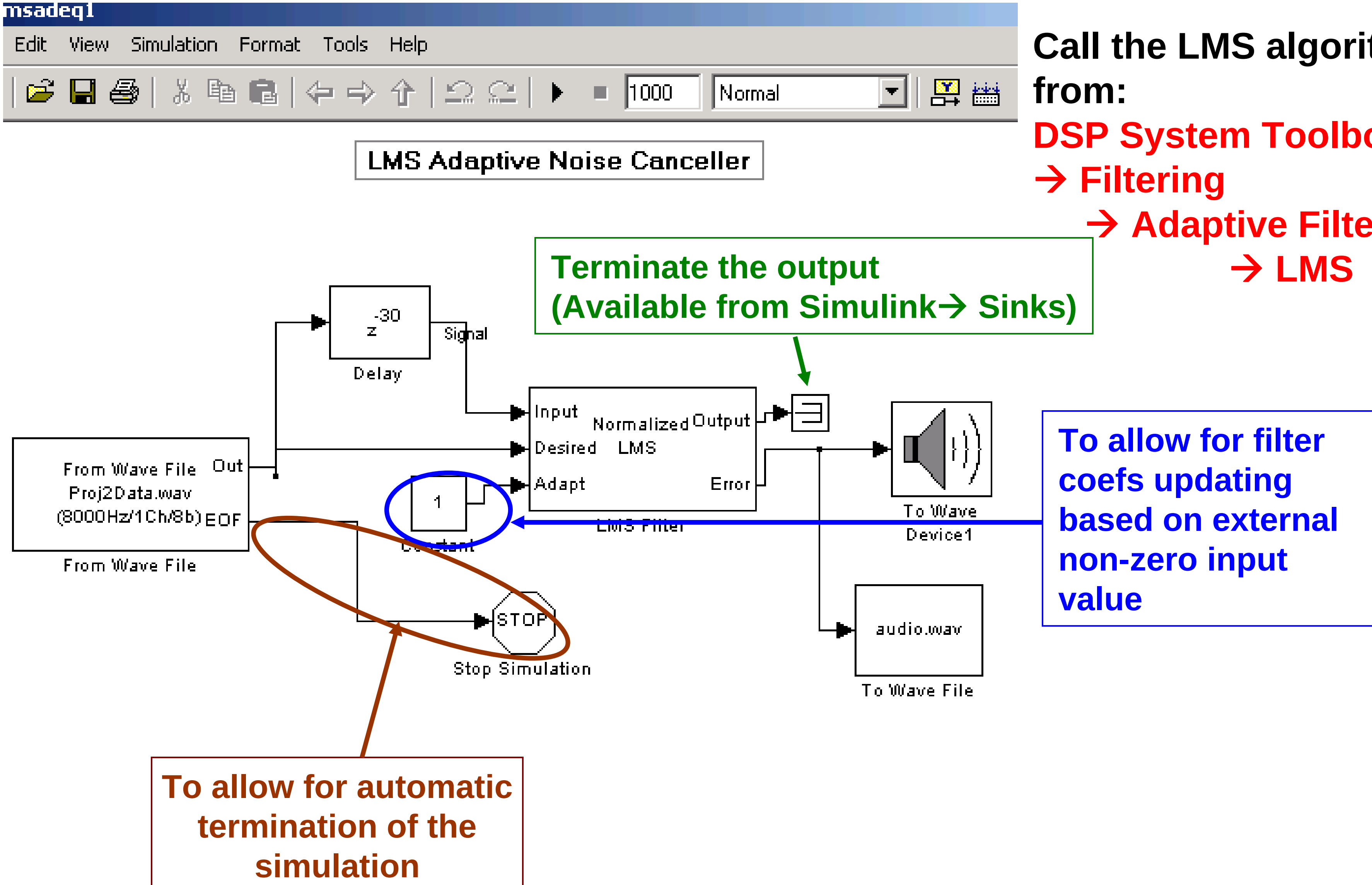


<u>2 options</u>:

a) Floating Scope (need to specify inputs)

b) Scope (need to connect inputs)

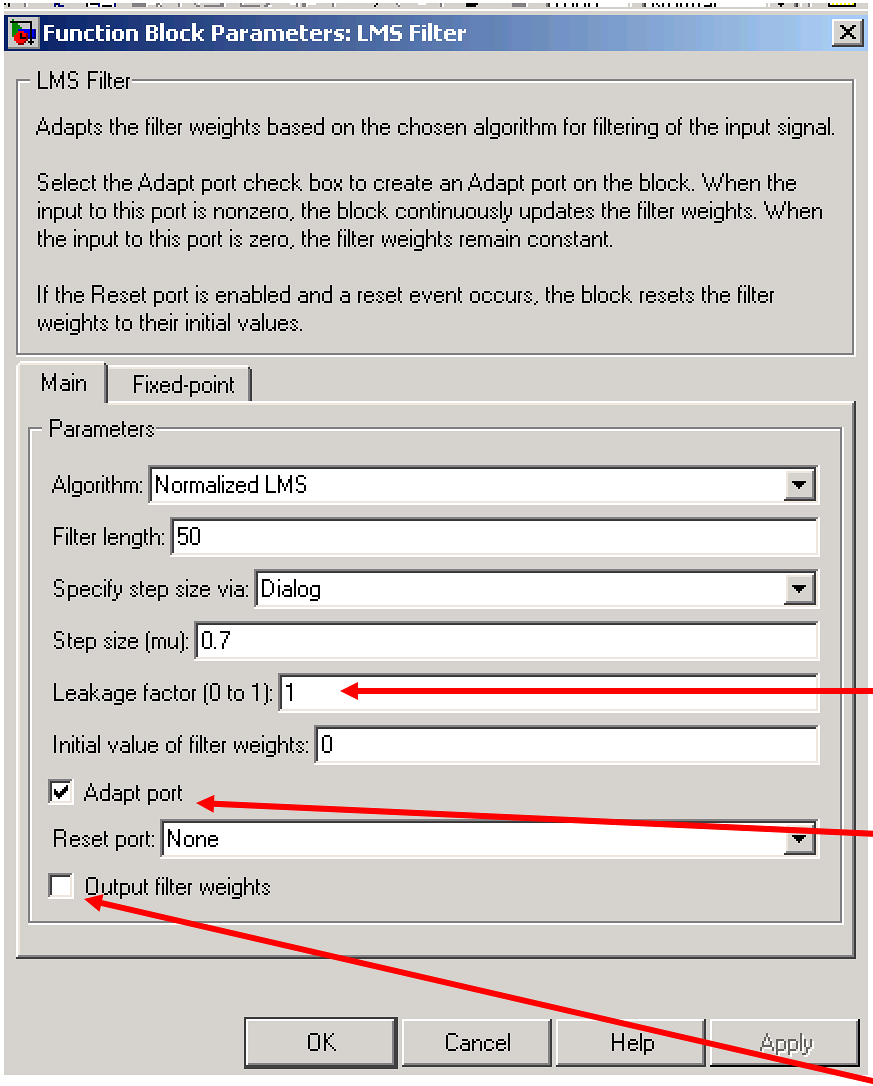
10) Implement the LMS algorithm (adaptive noise canceller application shown)



Call the LMS algorithm from:

DSP System Toolbox → Filtering → Adaptive Filters → LMS

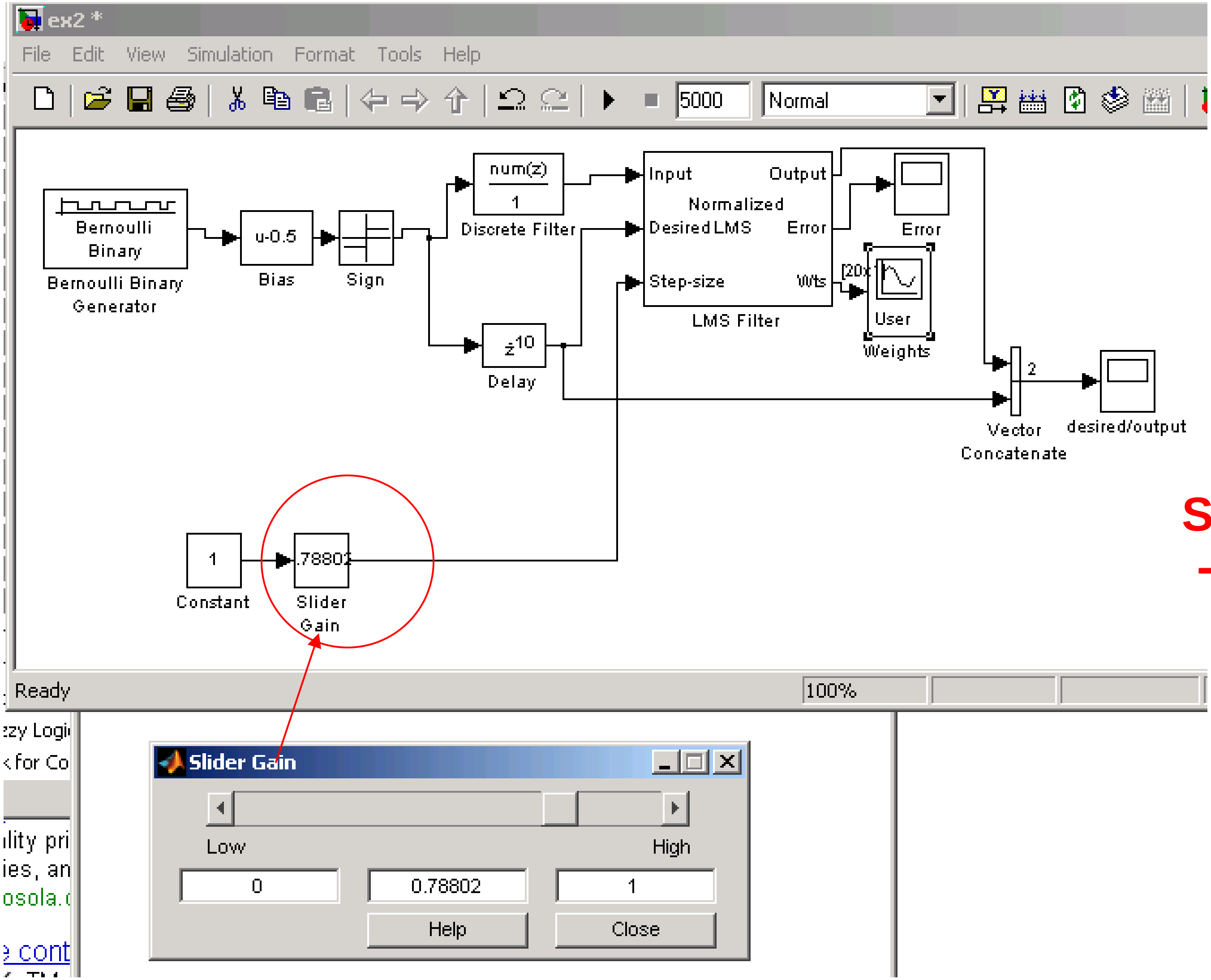
Normalized LMS configuration parameters



Leakage=1 ⇔ no leakage

Check to allow filter coef adaptation based on external non-zero value

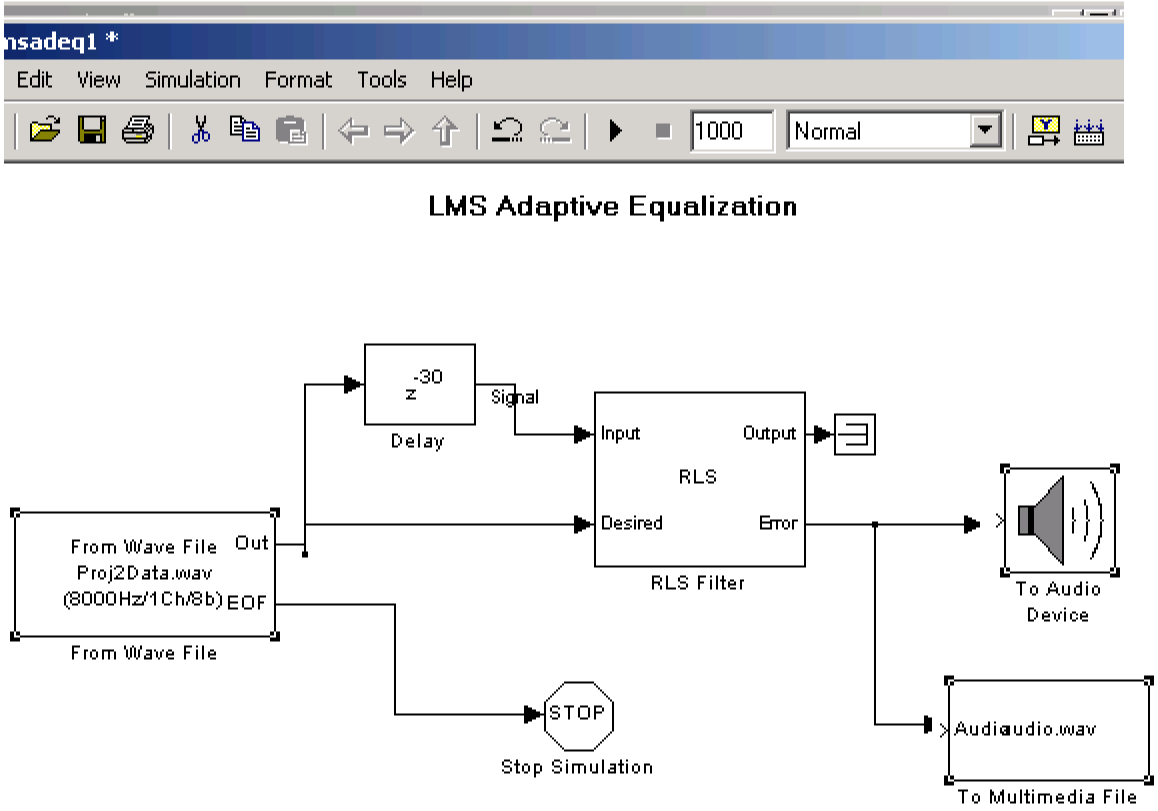
Check if you want to get the filter coefficient values out



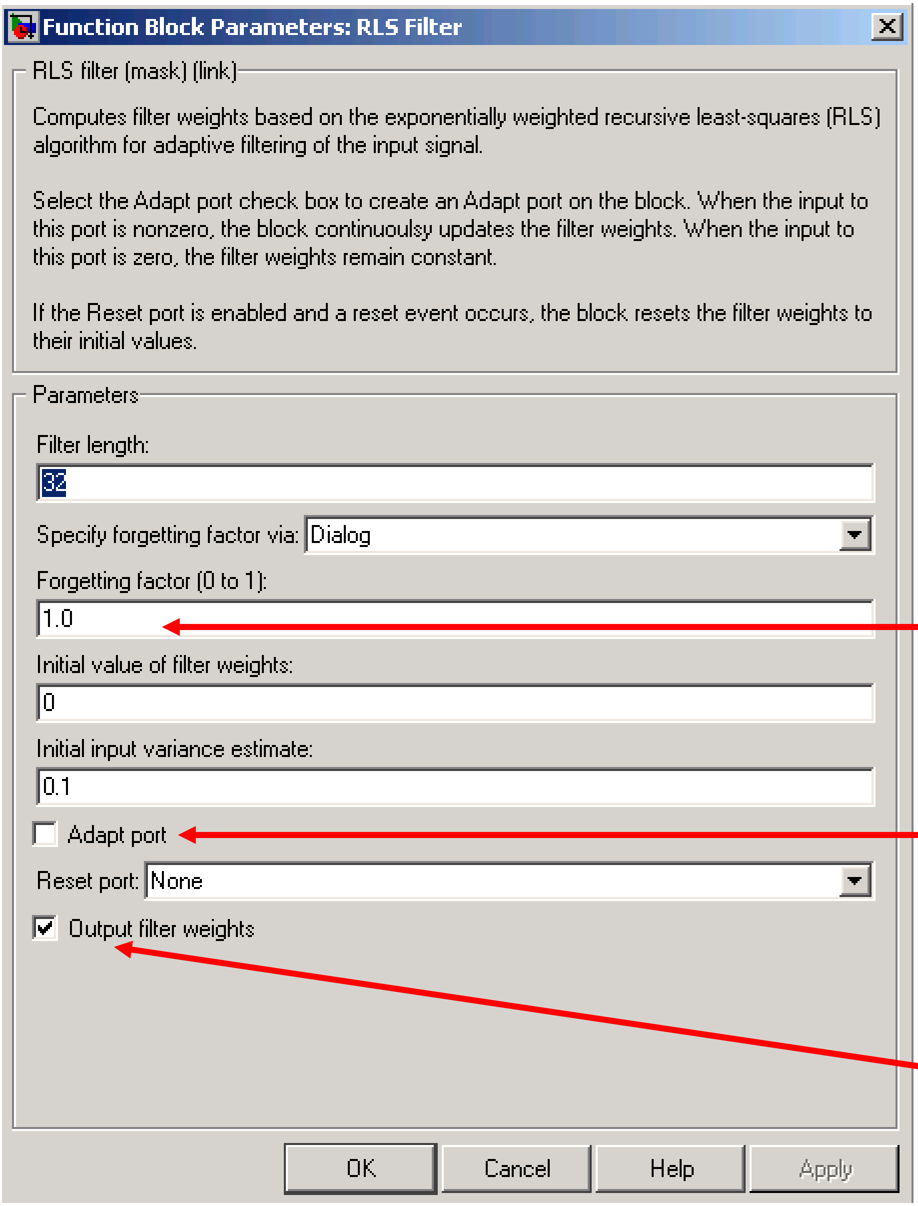
LMS step size can be varied using the "slider gain"

Simulink → Math Operations

11) Implement the RLS algorithm (adaptive noise canceller application shown)



Call the RLS algorithm from:
DSP System Toolbox
→ Filtering
→ Adaptive Filters
→ RLS



RLS configuration parameters

A value of 1 specifies an infinite memory.

Check to allow filter coef adaptation based on external non zero input value

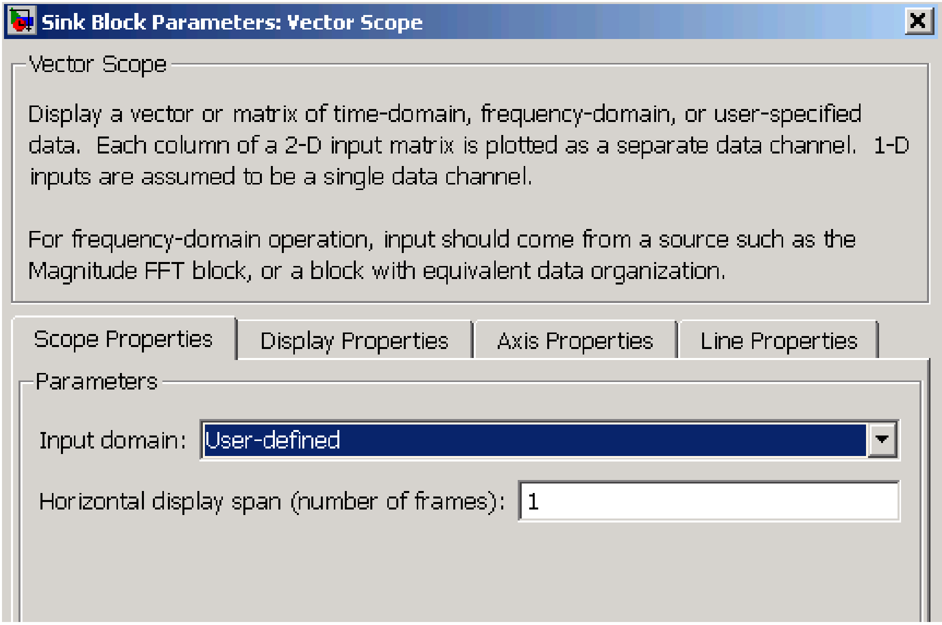
Check if you want to get the filter coefficient values out

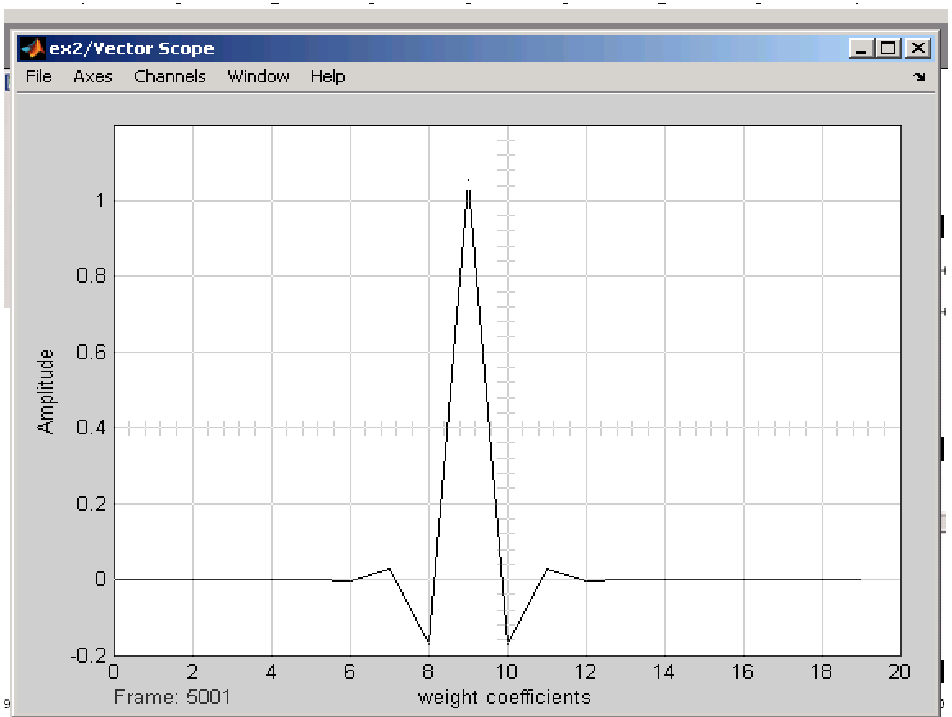
12) Plot filter coefficients using the vector scope

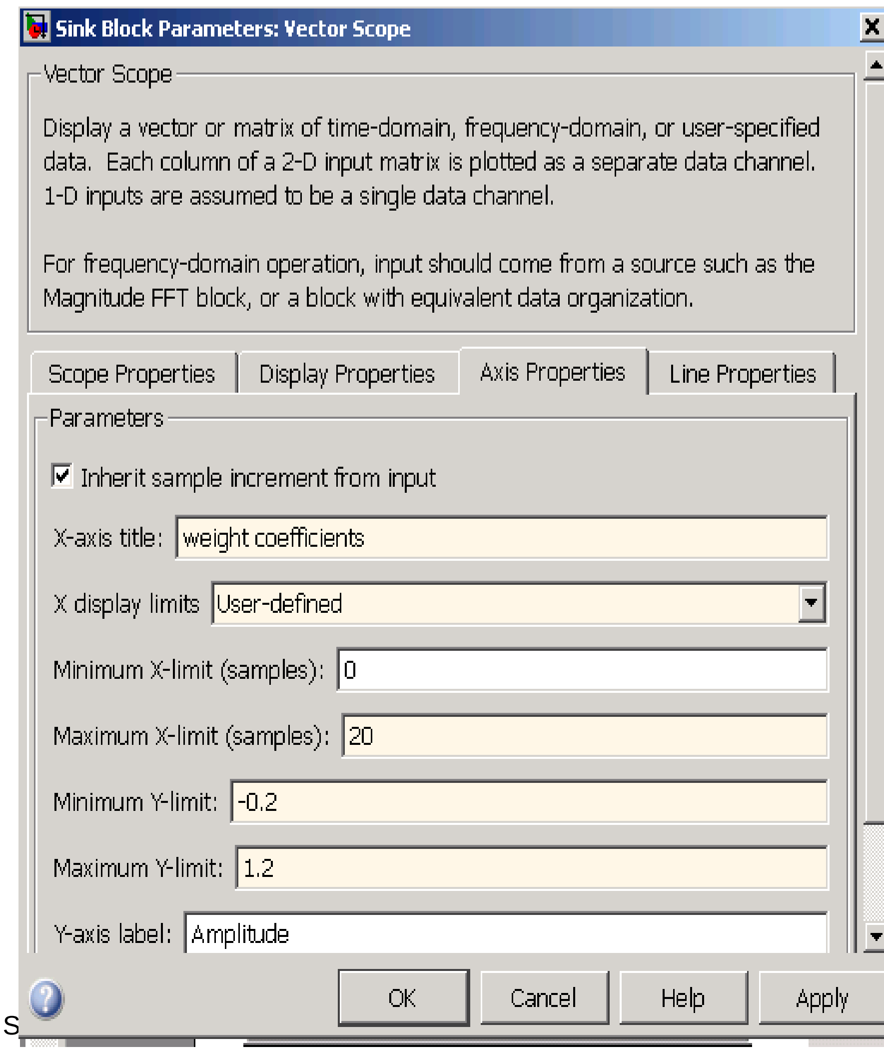
→ DSP System Toolbox

→ Signal Processing Sinks

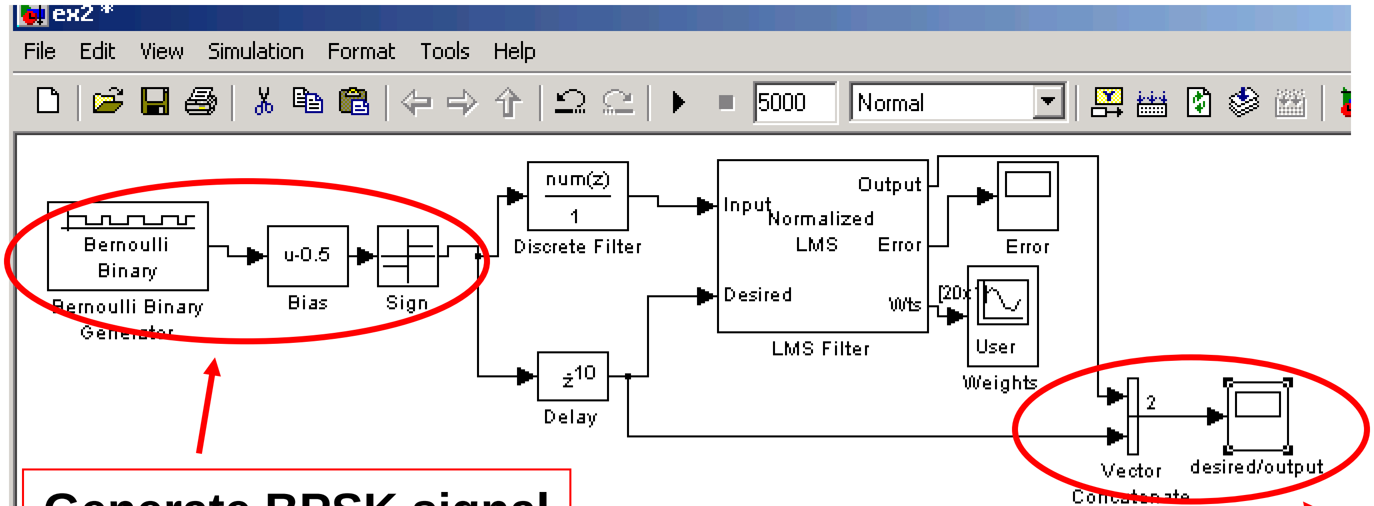
→ Vector Scope







13) Plot multiple data streams on the same figure

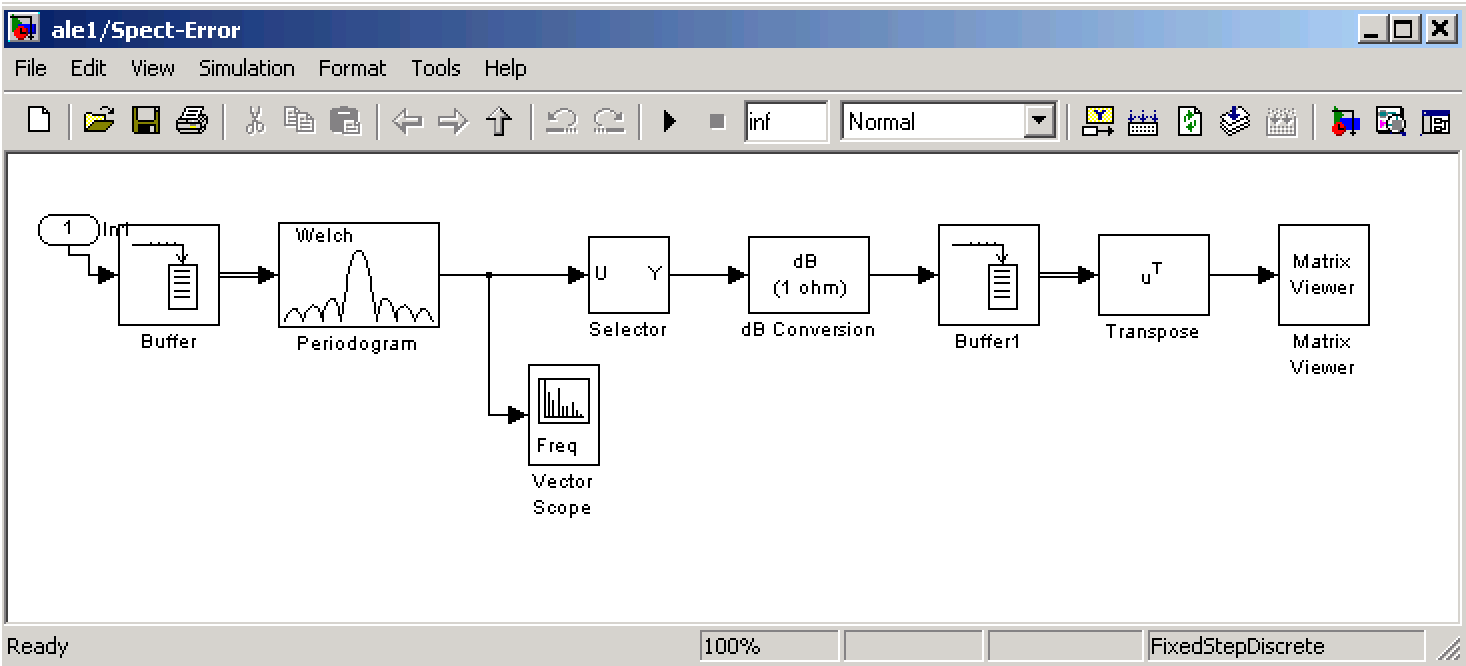


Generate BPSK signal

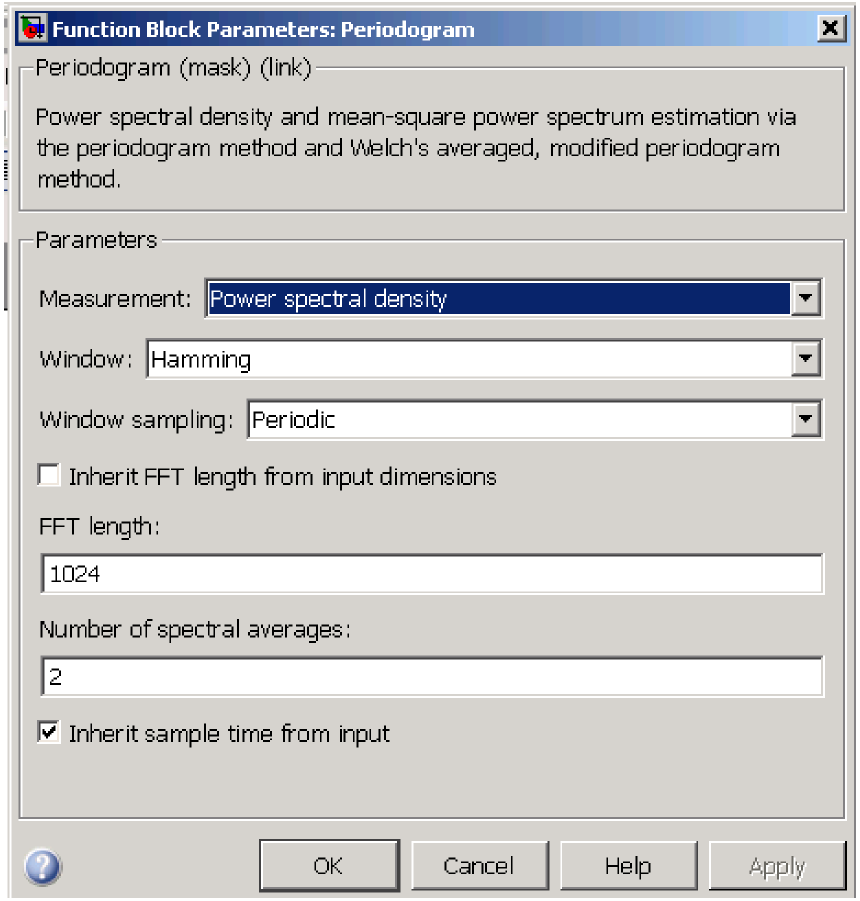
Use "vector concatenate" + Regular scope

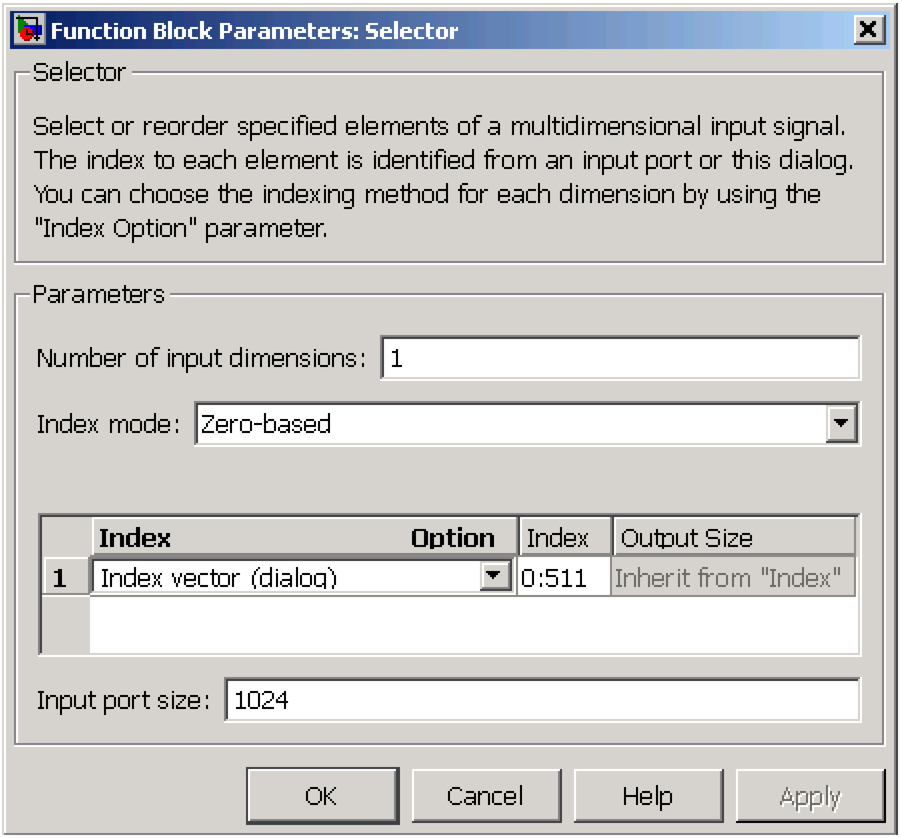
Simulink → Commonly used Blocks
→ Vector Concatenate
→ Scope

14) Generate spectrum and spectrogram plots
→ Specta.mdl (provided in course material)

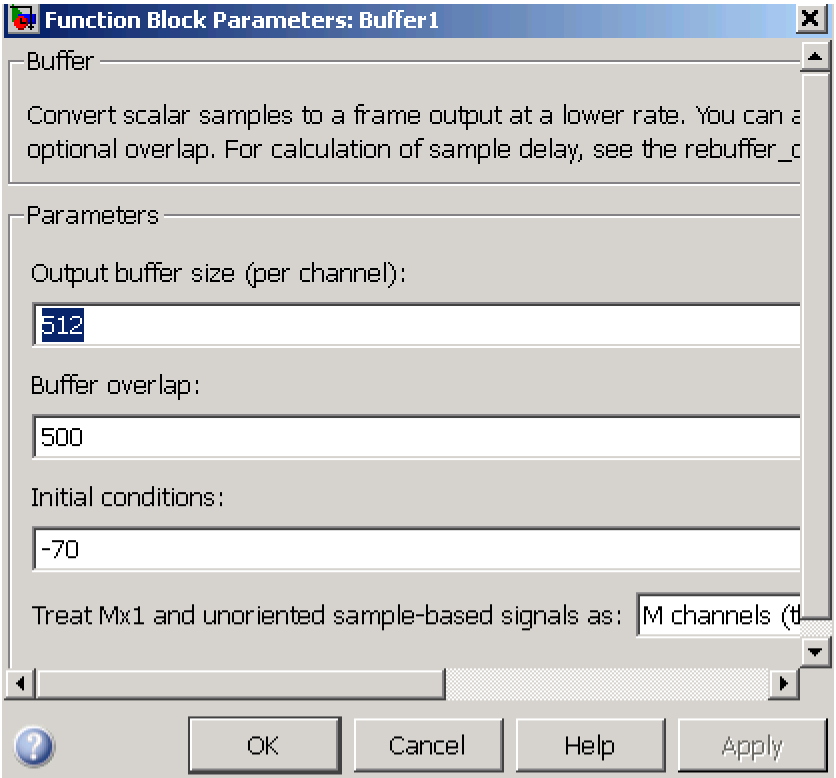


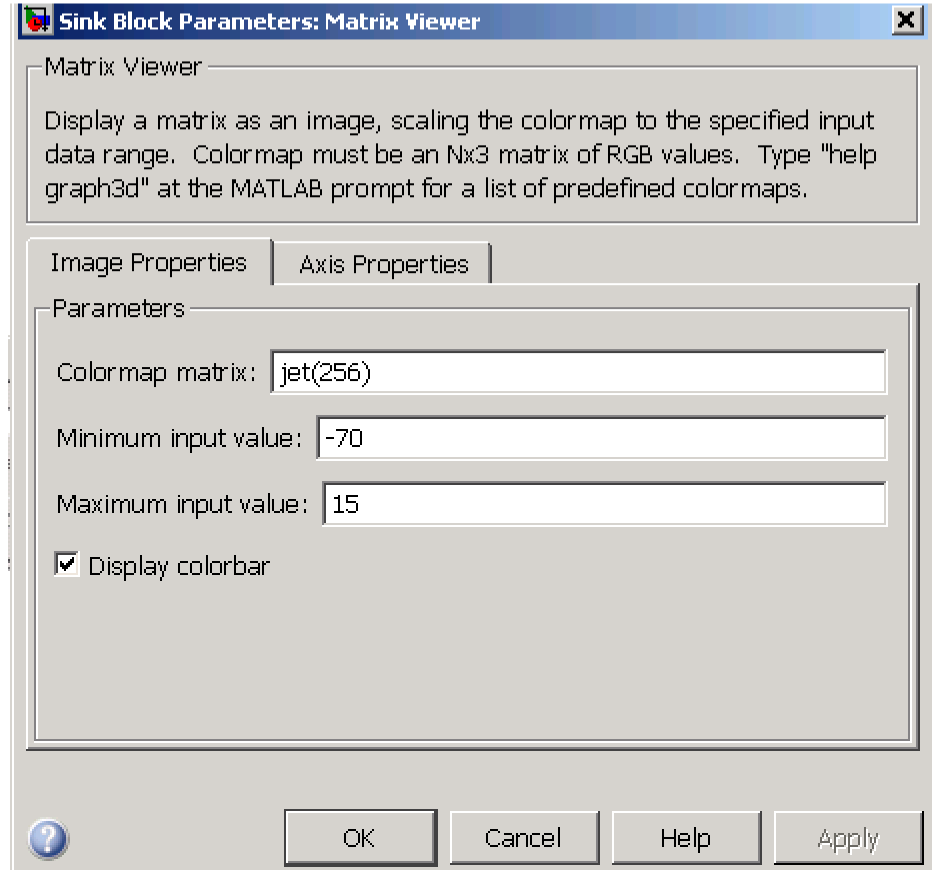
Blocks used in specta.mdl





Blocks used in specta.mdl, cont'



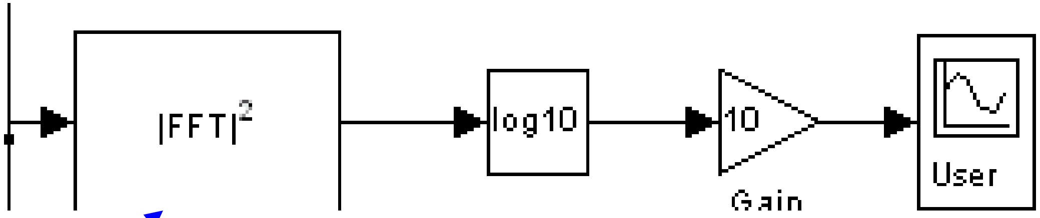


15) Frequency response plot generated from filter coefficients

The frequency response for the model $|1/A(e^{j\omega})|^2$ can be computed in dB from the filter coefficients by using the following blocks (this implementation leads to a frequency response plot identical to that given by *freqz.m*).

Note: The spectrum scope uses the periodogram to compute the spectrum expression which results in a discrepancy between simulink & freqz.m results.

Filter coefficients output



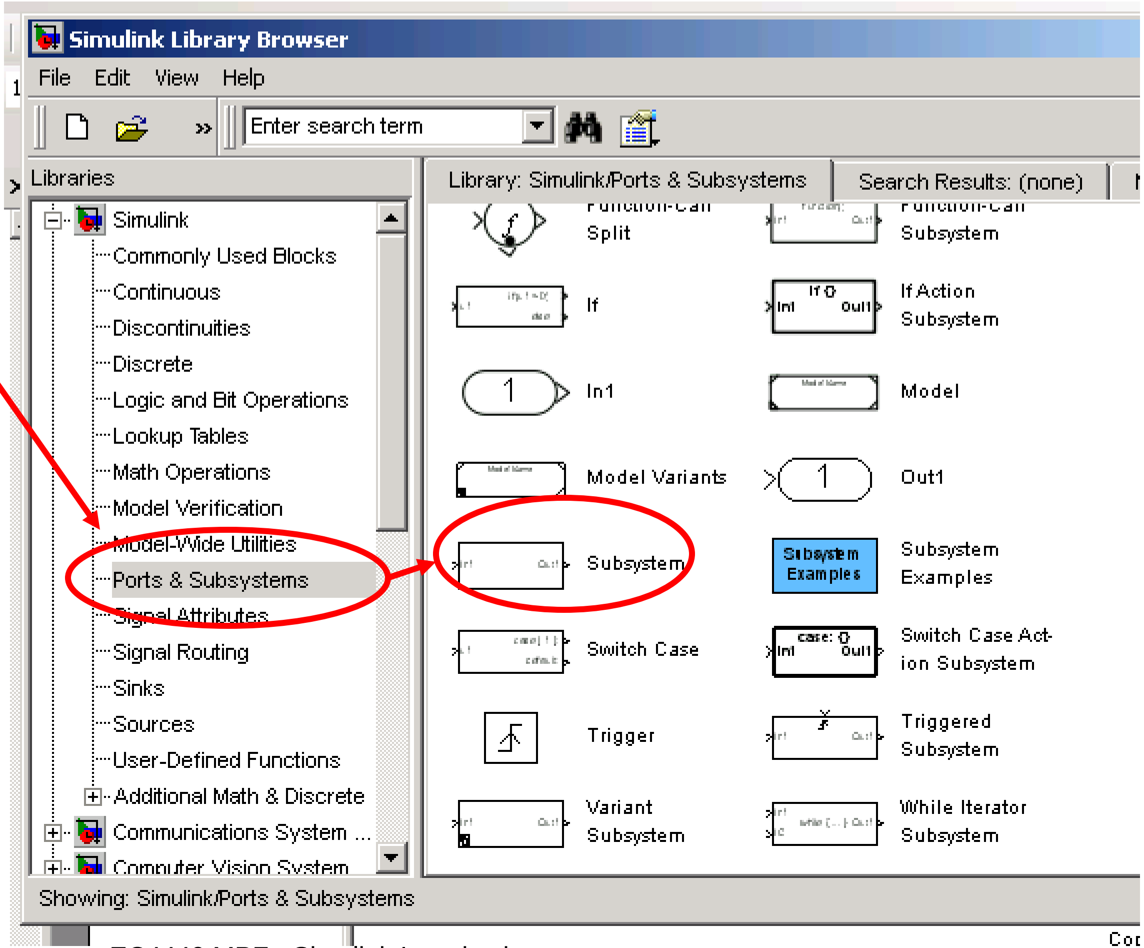
FFT magnitude
Note: FFT must be zero-padded sufficiently ~1024 or above to insure good visual quality of the frequency response

Vector Scope

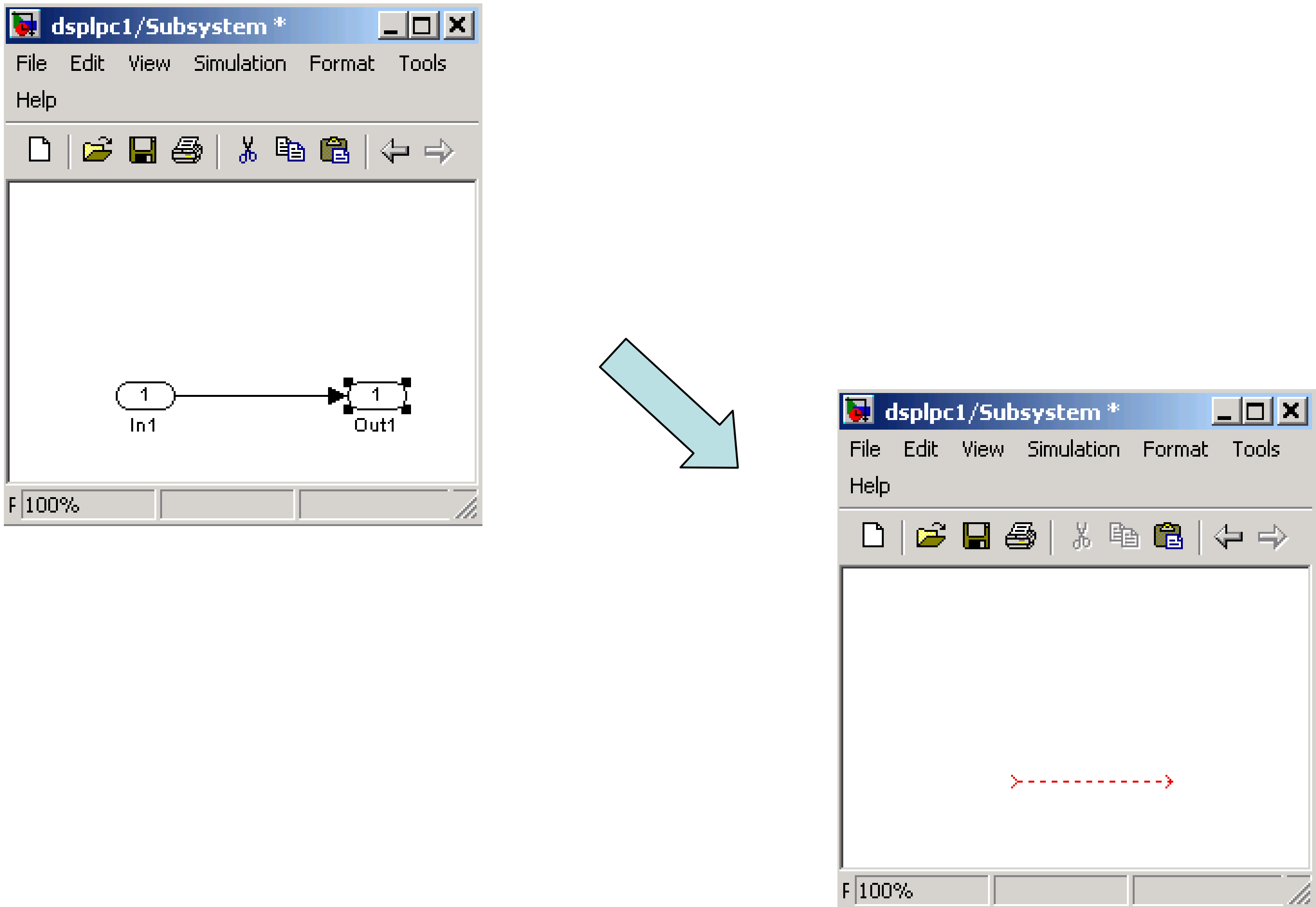
16) Listen to audio signals (Batch mode from Simulink)

a) send data to workspace
b) create a subsystem which plays the data

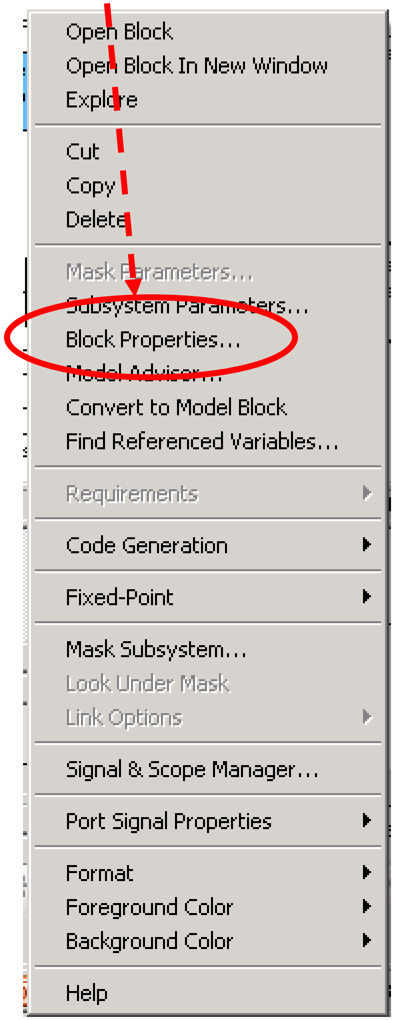
Simulink
→ Ports & Subsystems
→ Subsystem



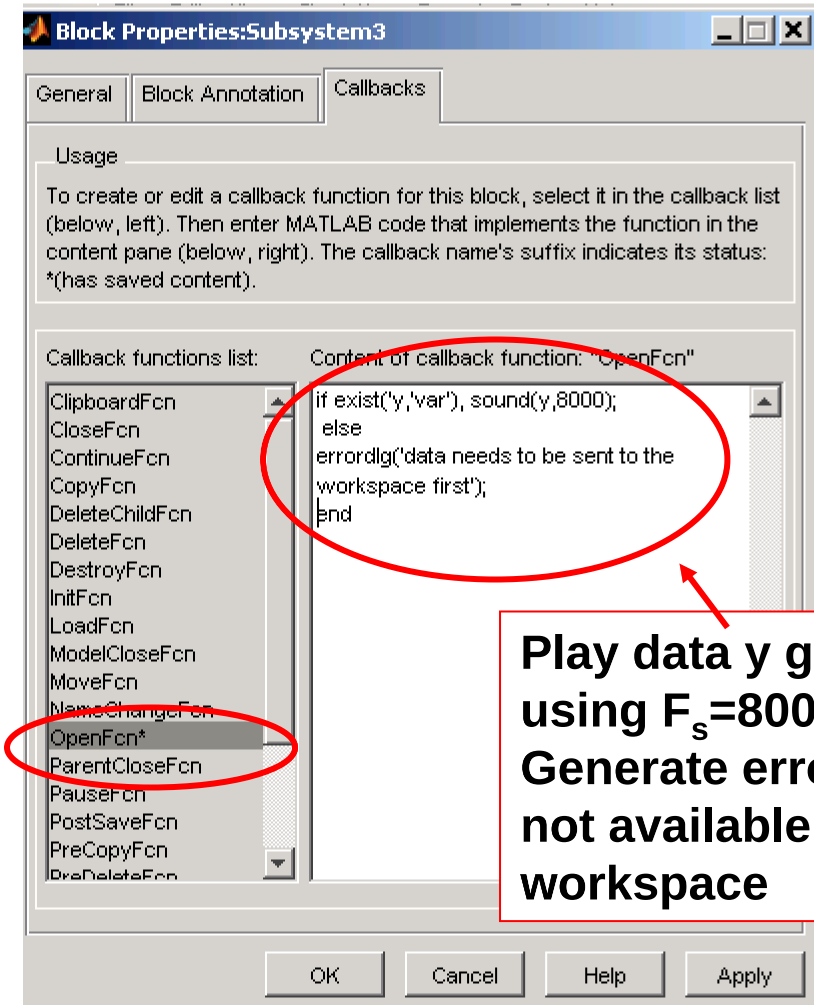
c) Remove subsystem input/output ports



d) Code audio play action by accessing system block properties



e) In-code audio play commands



Play data y generated using F_s=8000Hz & Generate error if data is not available in the workspace