

Using Simulink in Signal Processing Applications

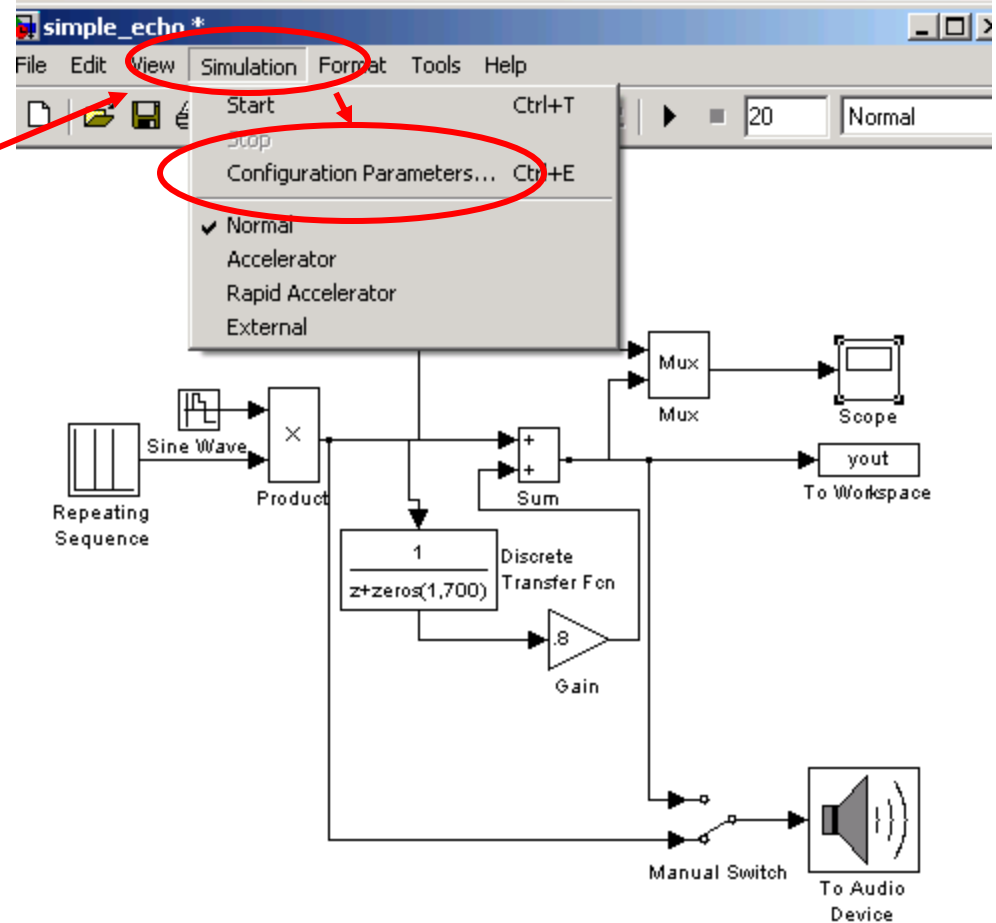
Basic Simulink blocks discussed

- 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

Configuration Parameters: ale1/Configuration (Active)

Select:

- Solver
- Data Import/Export
- Optimization
- Diagnosics
 - Sample Time
 - Data Validity
 - Type Conversion
 - Connectivity
 - Compatibility
 - Model Referencing
 - Saving
- Hardware Implementati...
- Model Referencing
- Simulation Target
 - Symbols
 - Custom Code
- Real-Time Workshop
 - Report
 - Comments
 - Symbols
 - Custom Code
 - Debug
 - Interface

Simulation time

Start time: 0.0 Stop time: inf

Solver options

Type: Fixed-step Solver: discrete (no continuous states)

Fixed-step size (fundamental sample time): 1

Tasking and sample time options

Periodic sample time constraint: Unconstrained

Tasking mode for periodic sample times: Auto

Automatically handle rate transition for data transfer

Higher priority value indicates higher task priority

Specify sample time used for simulation

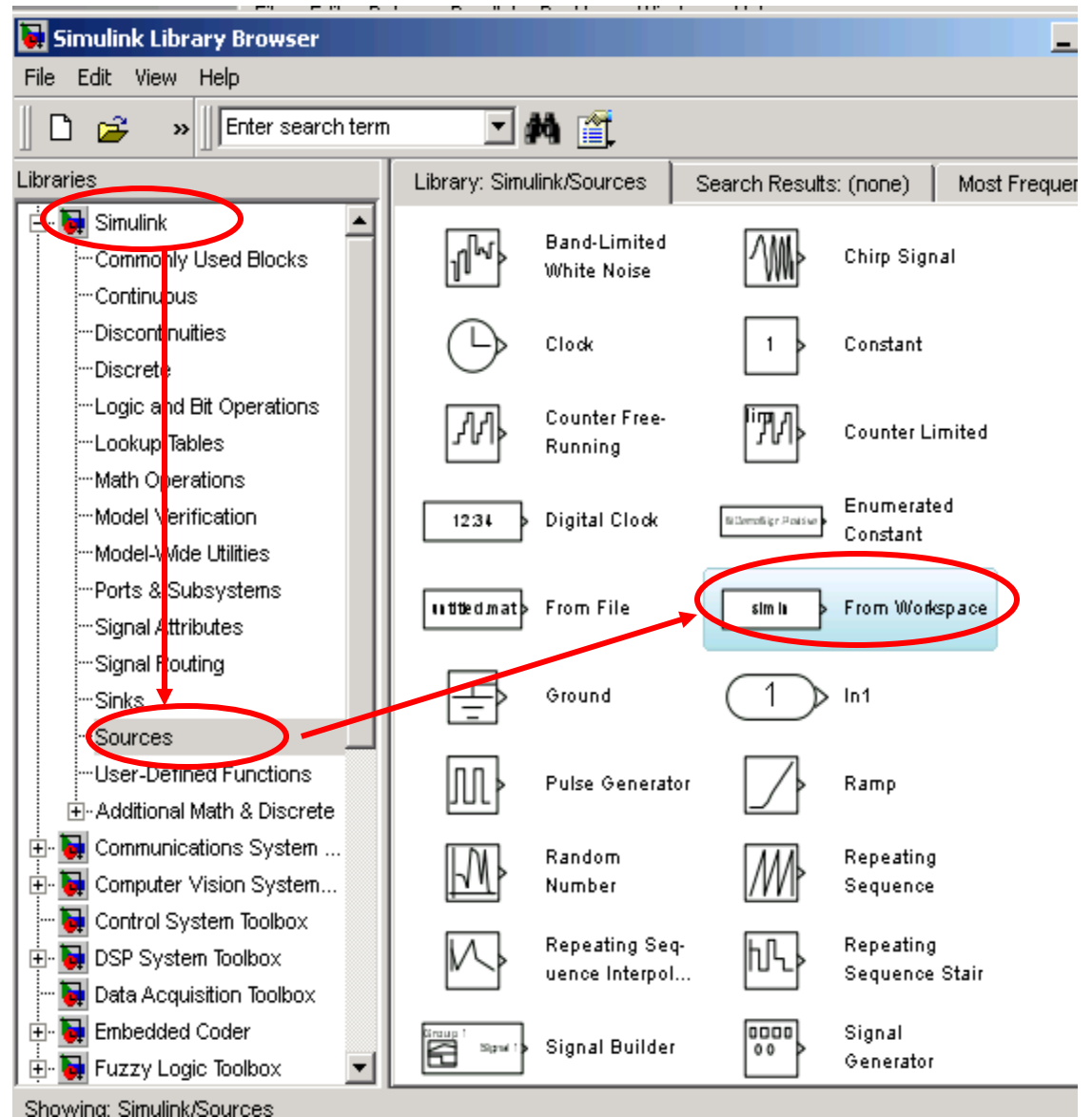
2) How to read data from the workspace

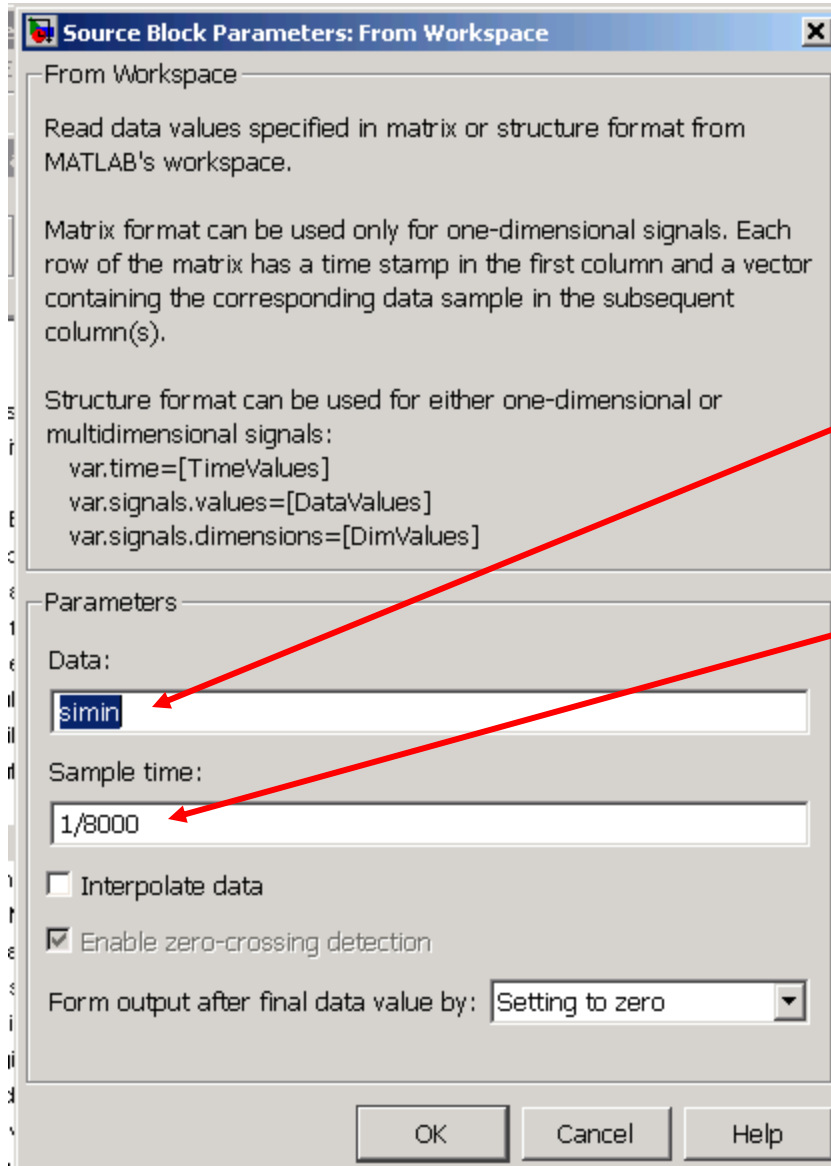
Select

Simulink

→ Sources

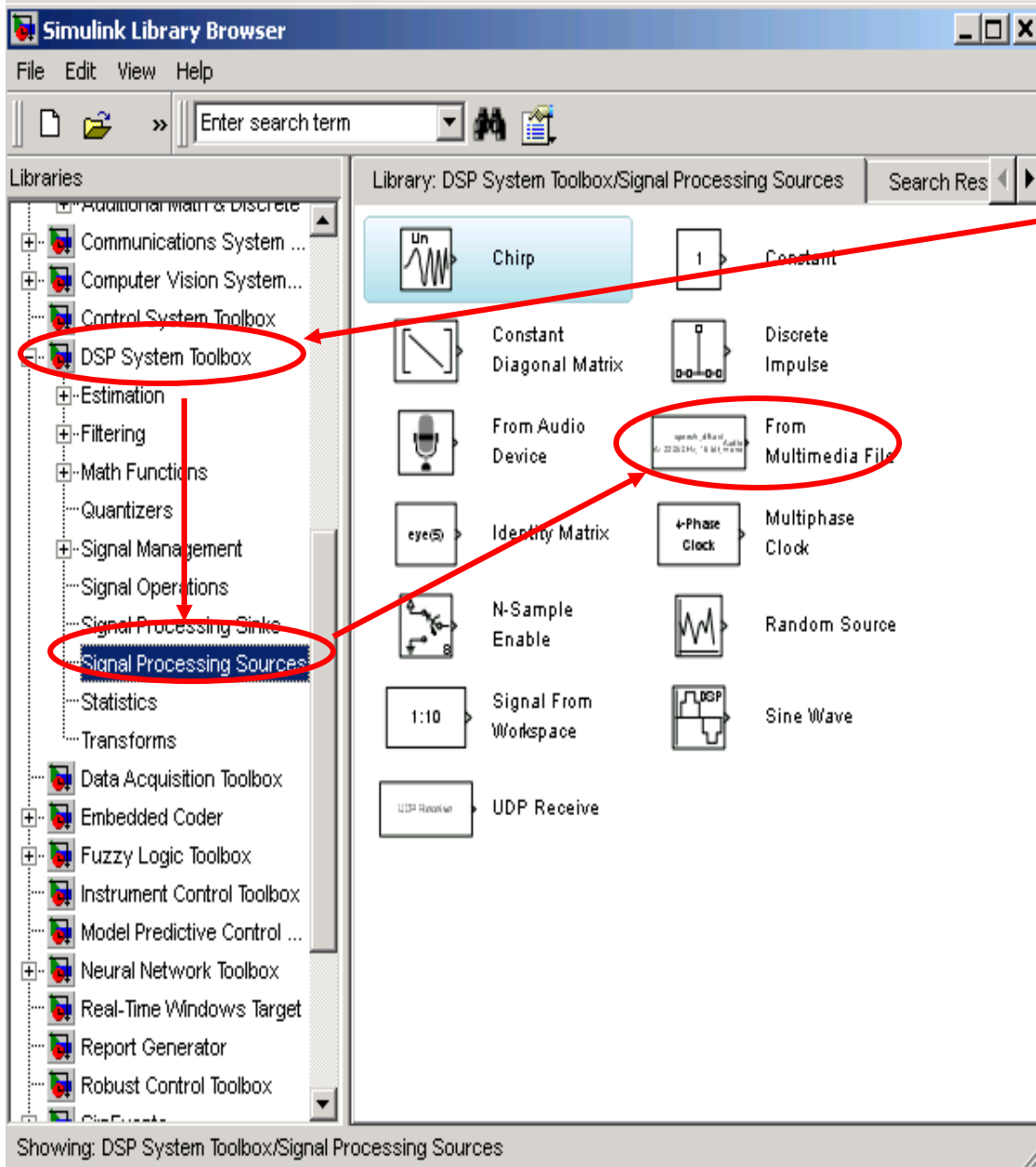
→ From Workspace





INPUT DATA FORMAT

- 1) Data must be formatted as $y_{nn2} = [\text{timesample}, \text{datasample}]$, format: $N \times 2$
- 2) Need to define `-timesample-` with the correct sampling frequency



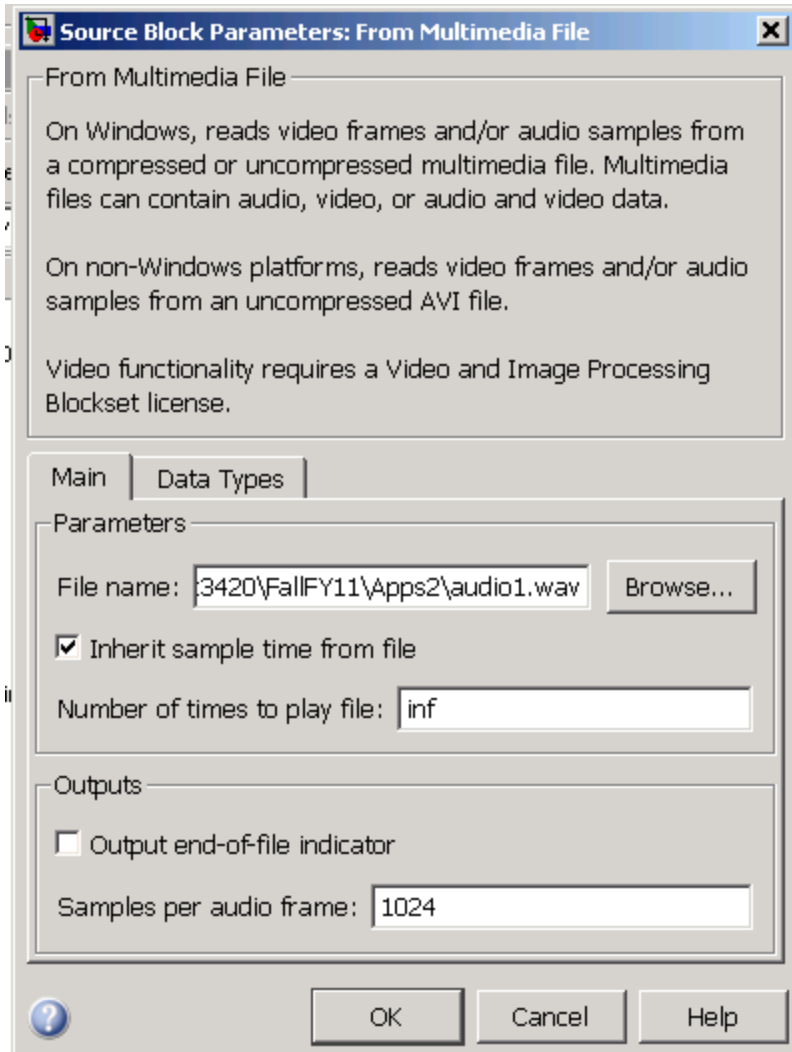
3) How to read .wav file

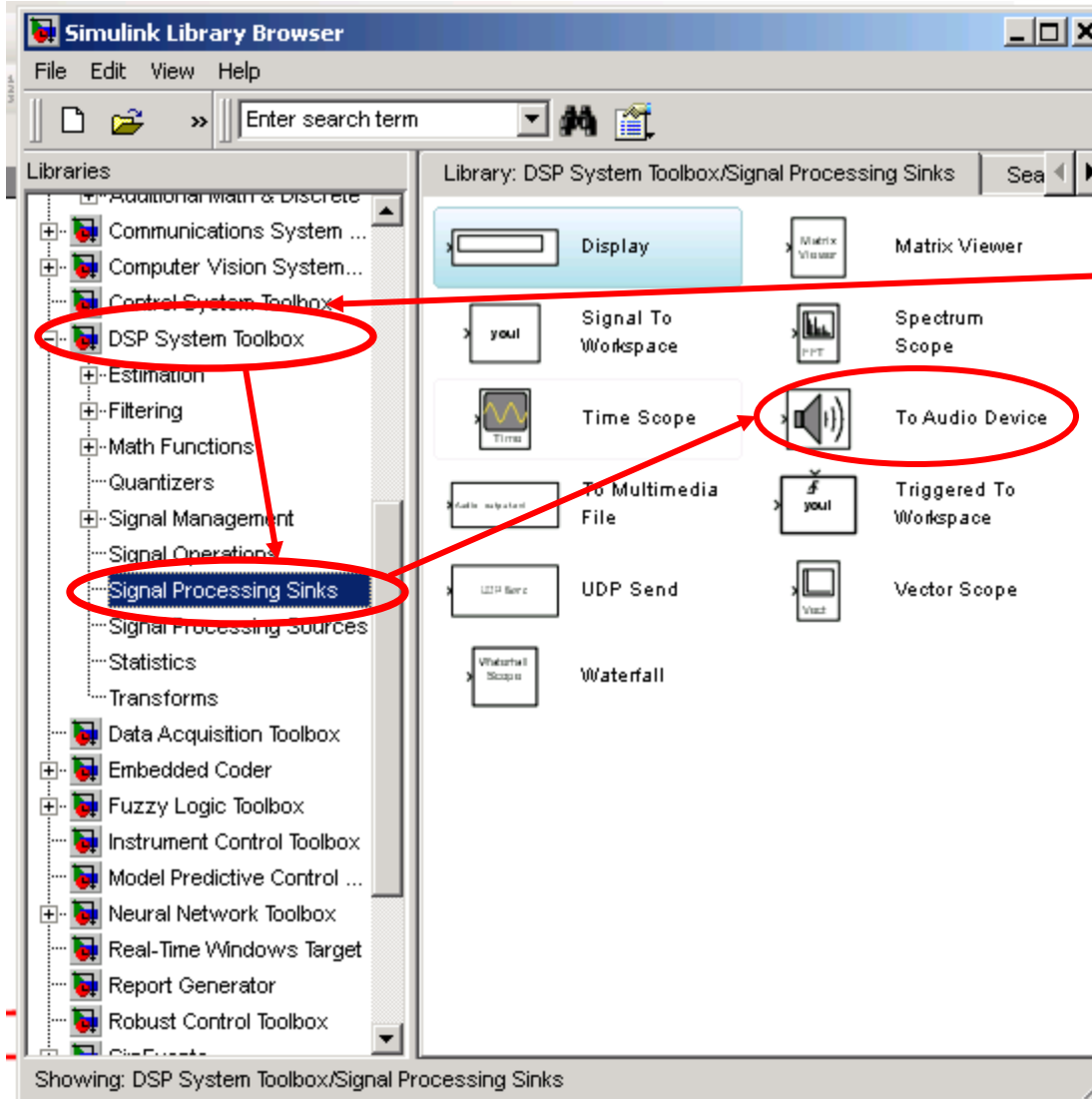
Select

DSP System Toolbox

→ Signal Processing Sources

→ From Multimedia File





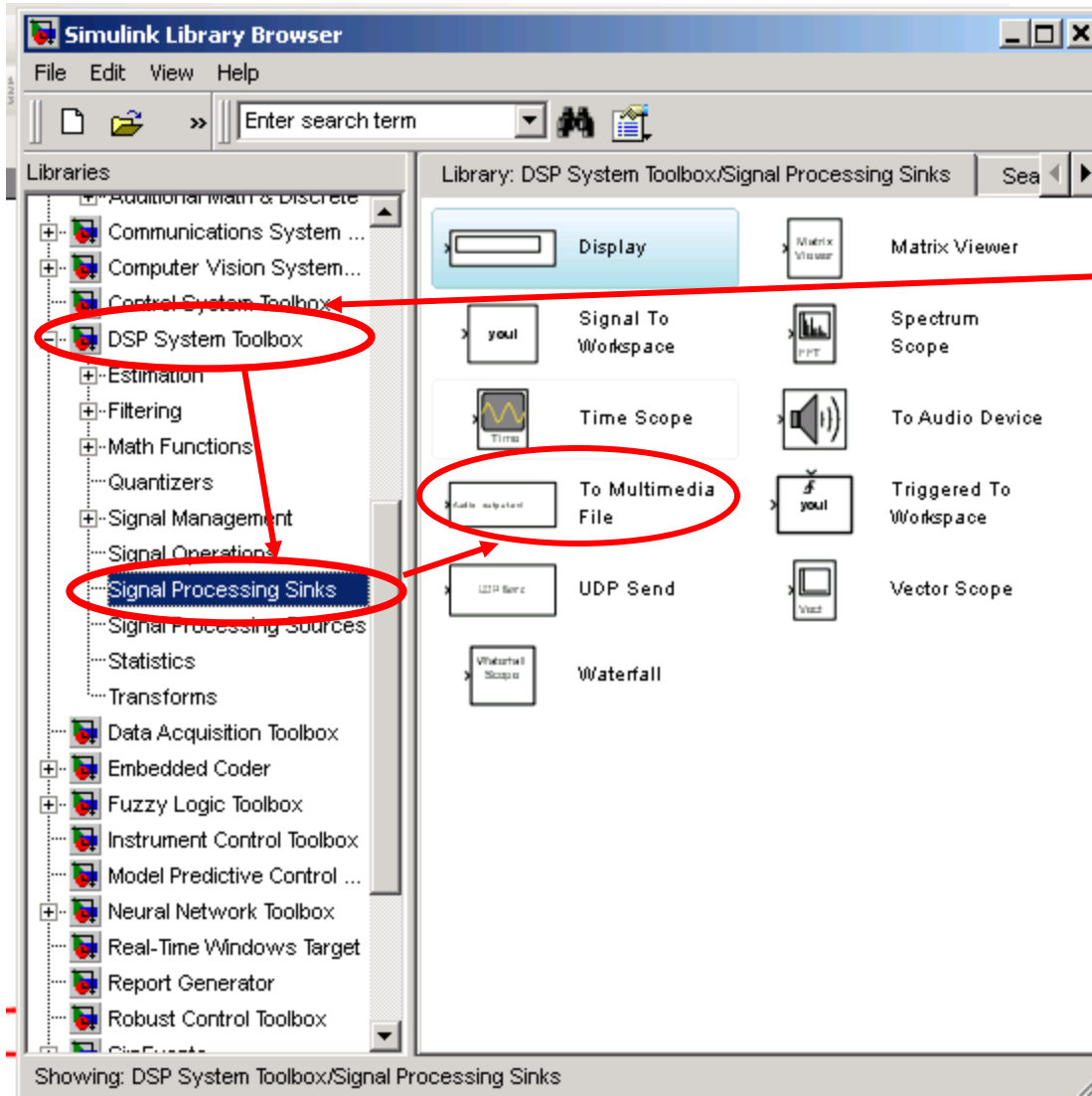
4) How to listen to a sound file

Select

DSP System Toolbox

→ Signal Processing Sinks

→ To Audio Device



5) Save data to a multimedia file

Select

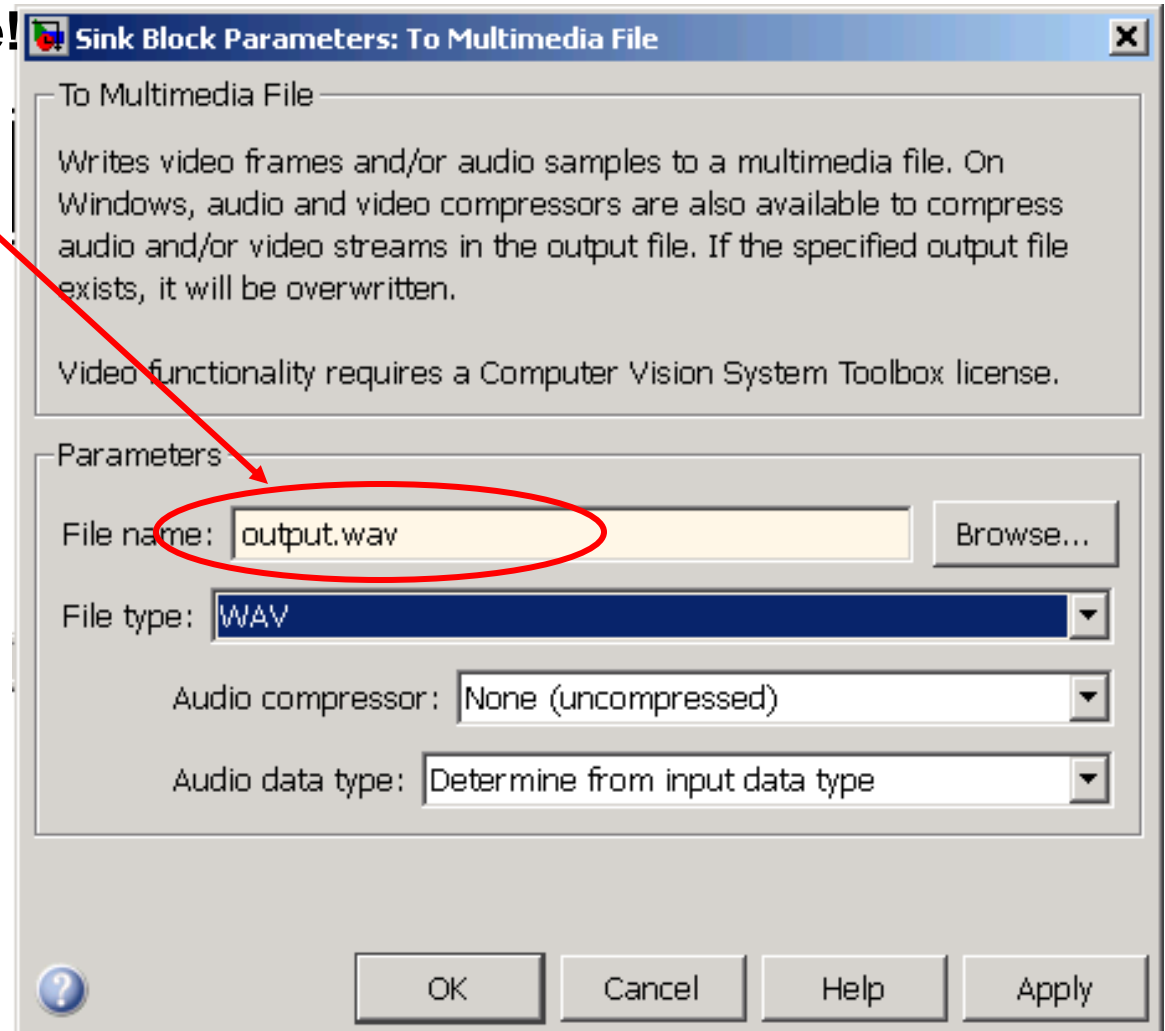
DSP System Toolbox

→ Signal Processing Sinks

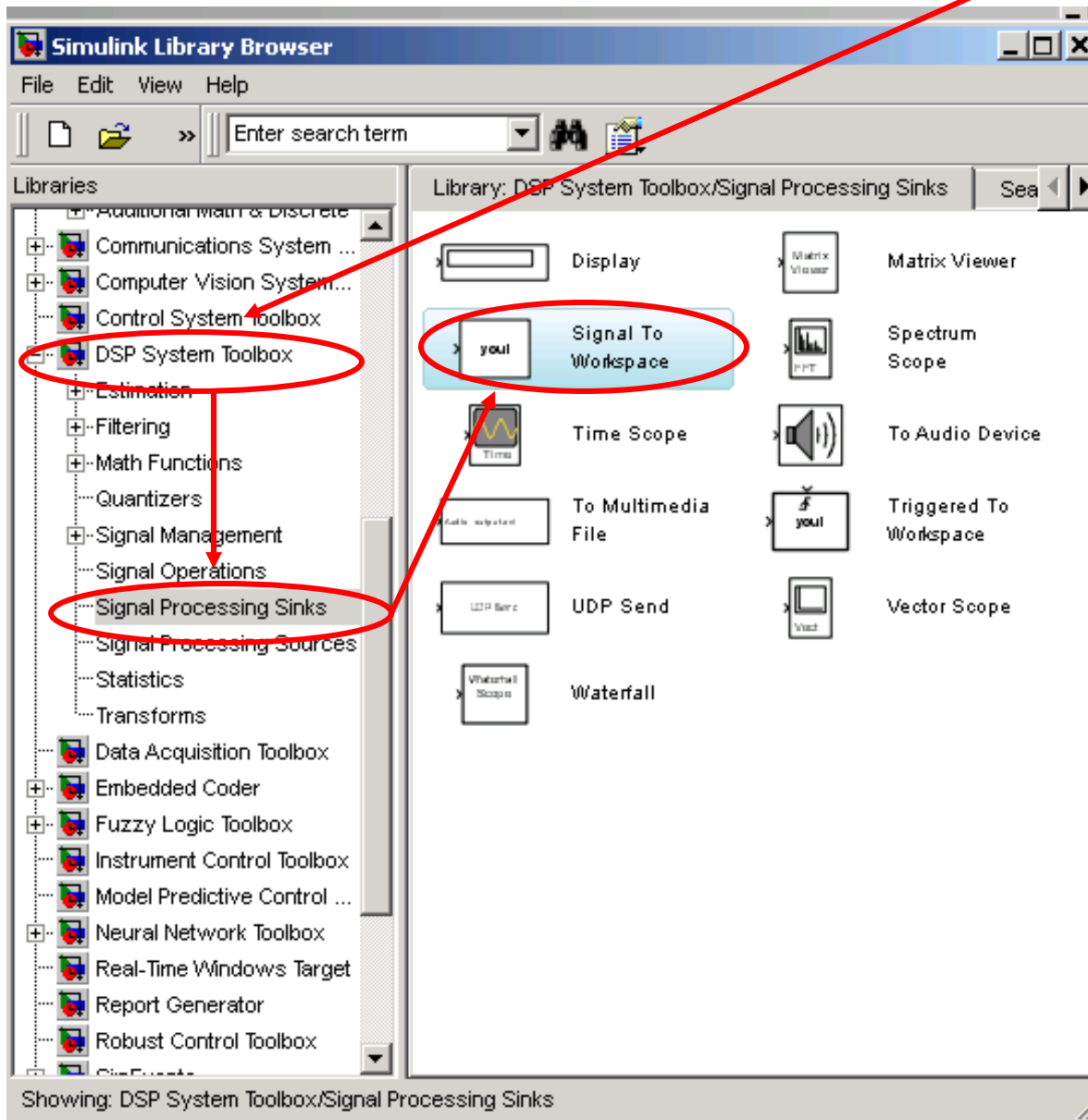
→ To Multimedia File

Output to Wave device

Need to define file name!



6) Save data to workspace



Select

DSP System Toolbox

→ Signal Processing Sinks

→ Signal To Workspace

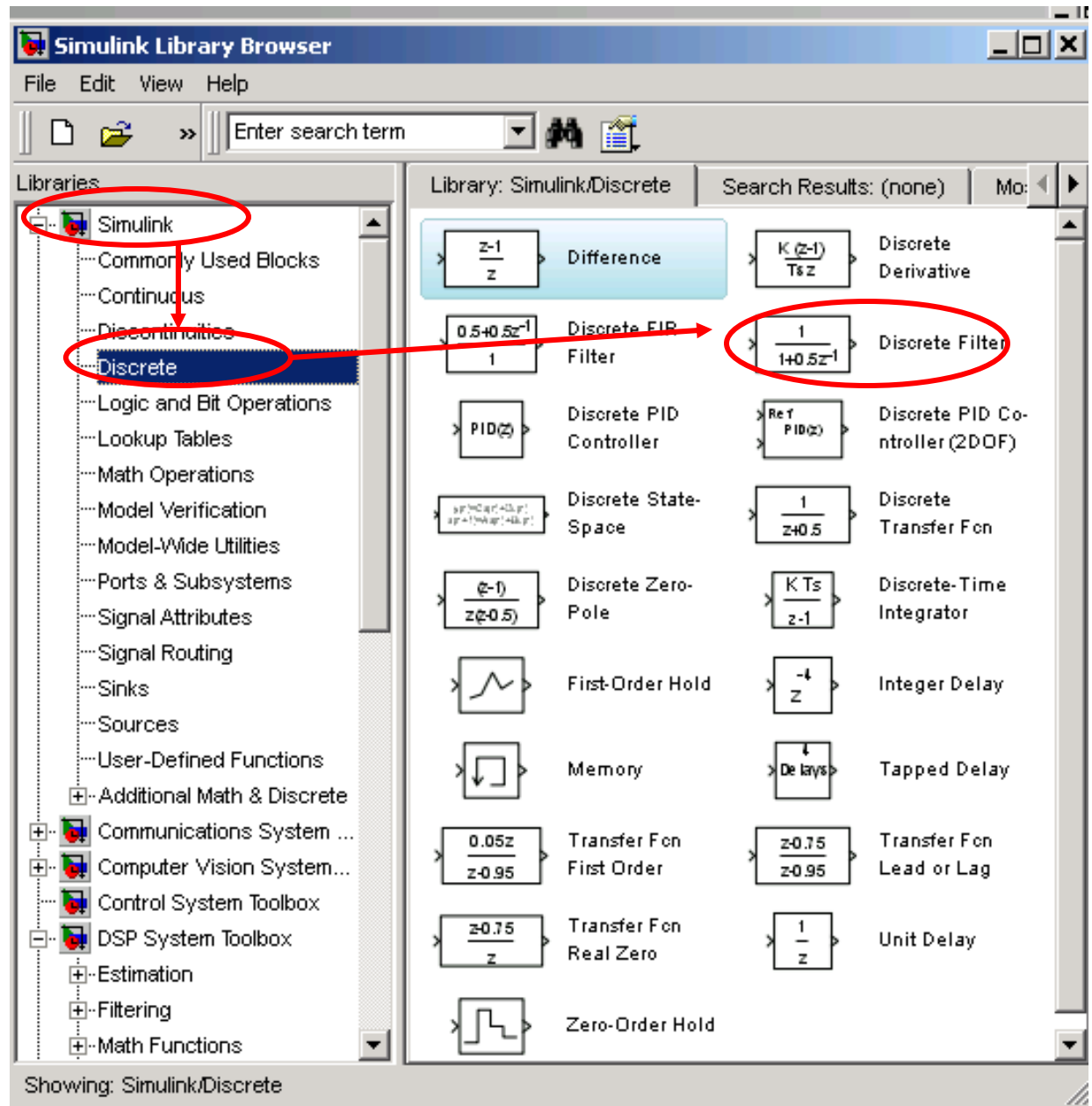
7) Specify IIR/FIR Filter characteristics

Select

Simulink

→ Discrete

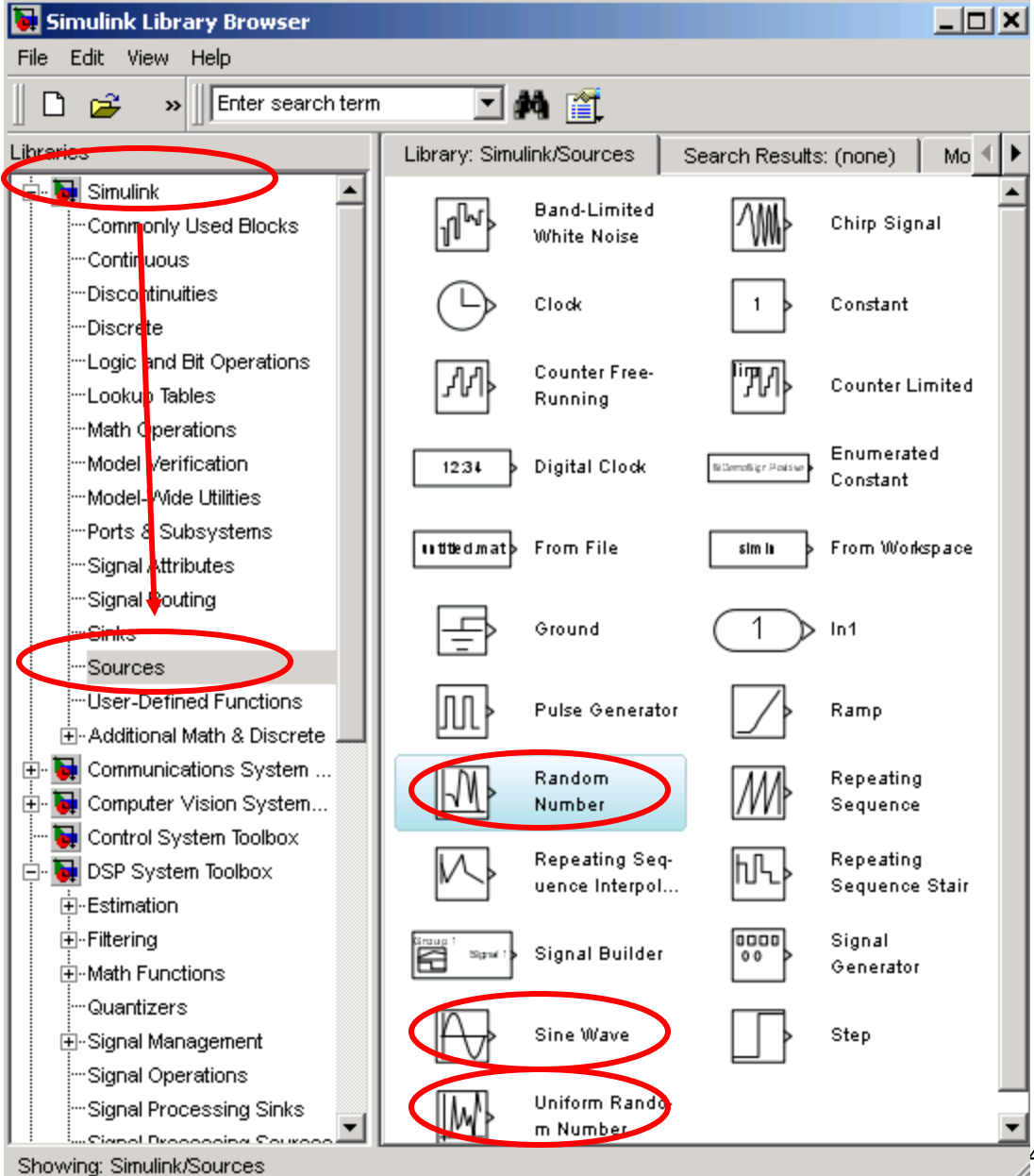
→ Discrete Filter



8) Specify internal input data

Specific parameters specified within each block

Select
Simulink
→ Sources



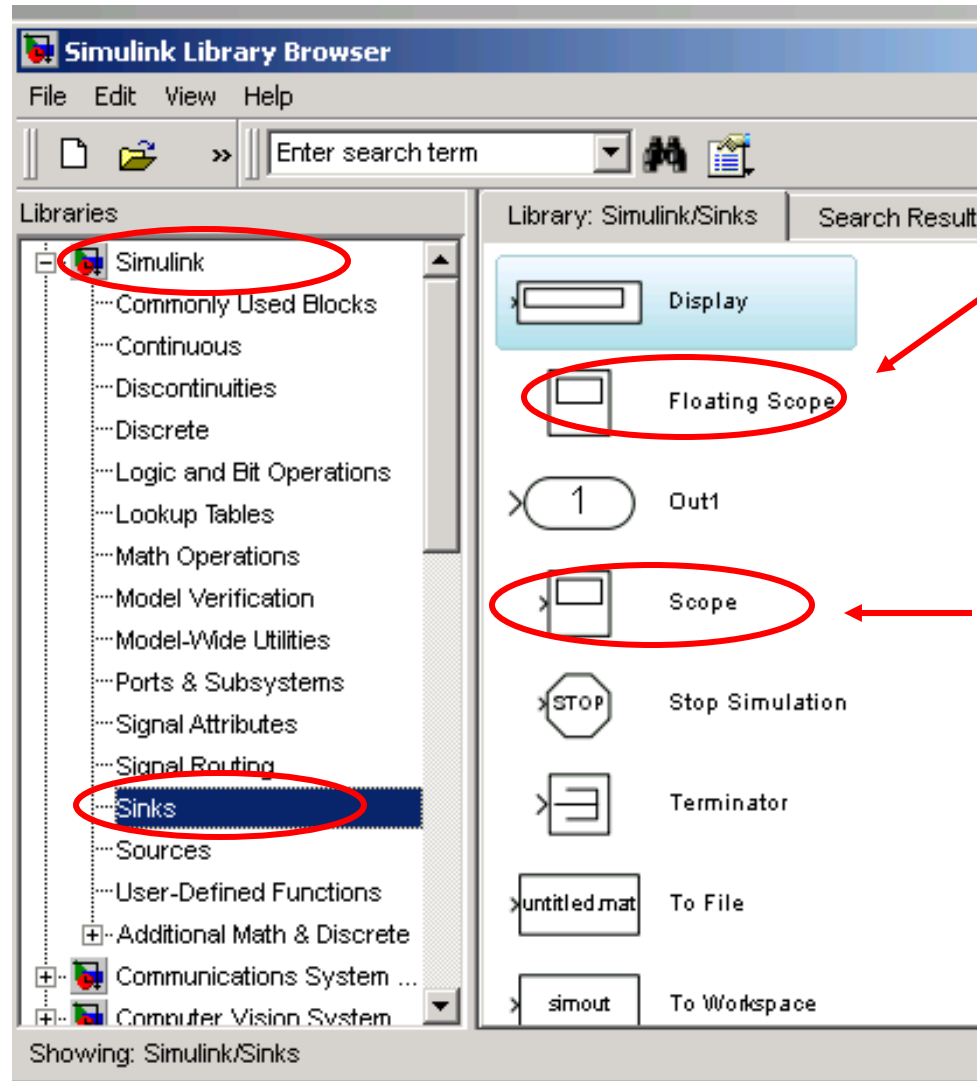
→ White Gaussian Noise

→ Sinewave

→ Uniform Random Noise

9) Plot data using Scope blocks

Select
Simulink
→ Sinks



2 options:

a) Floating Scope
(need to specify
inputs)

b) Scope
(need to
connect inputs)

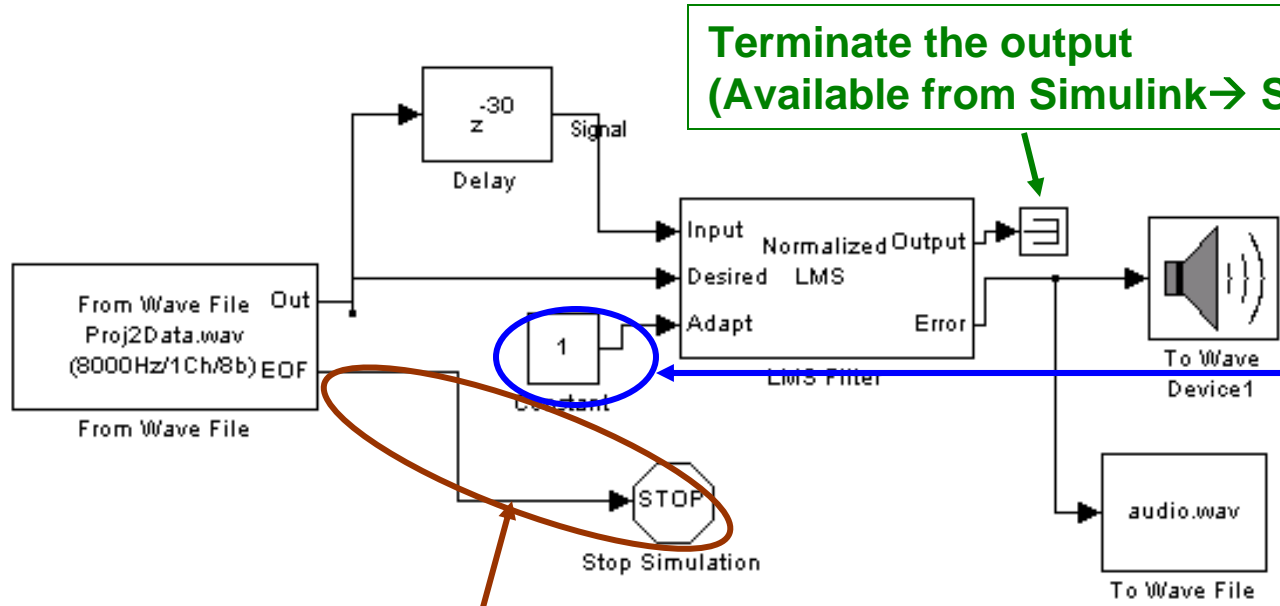
10) Implement the LMS algorithm (adaptive noise canceller application shown)

Call the LMS algorithm from:

DSP System Toolbox
→ **Filtering**
→ **Adaptive Filters**
→ **LMS**



LMS Adaptive Noise Canceller



Terminate the output
(Available from Simulink → Sinks)

To allow for filter coefs updating based on external non-zero input value

To allow for automatic termination of the simulation

Normalized LMS configuration parameters

Function Block Parameters: LMS Filter

LMS Filter

Adapts the filter weights based on the chosen algorithm for filtering of the input signal.

Select the Adapt port check box to create an Adapt port on the block. When the input to this port is nonzero, the block continuously updates the filter weights. When the input to this port is zero, the filter weights remain constant.

If the Reset port is enabled and a reset event occurs, the block resets the filter weights to their initial values.

Main | Fixed-point

Parameters

Algorithm: Normalized LMS

Filter length: 50

Specify step size via: Dialog

Step size (mu): 0.7

Leakage factor (0 to 1): 1

Initial value of filter weights: 0

Adapt port

Reset port: None

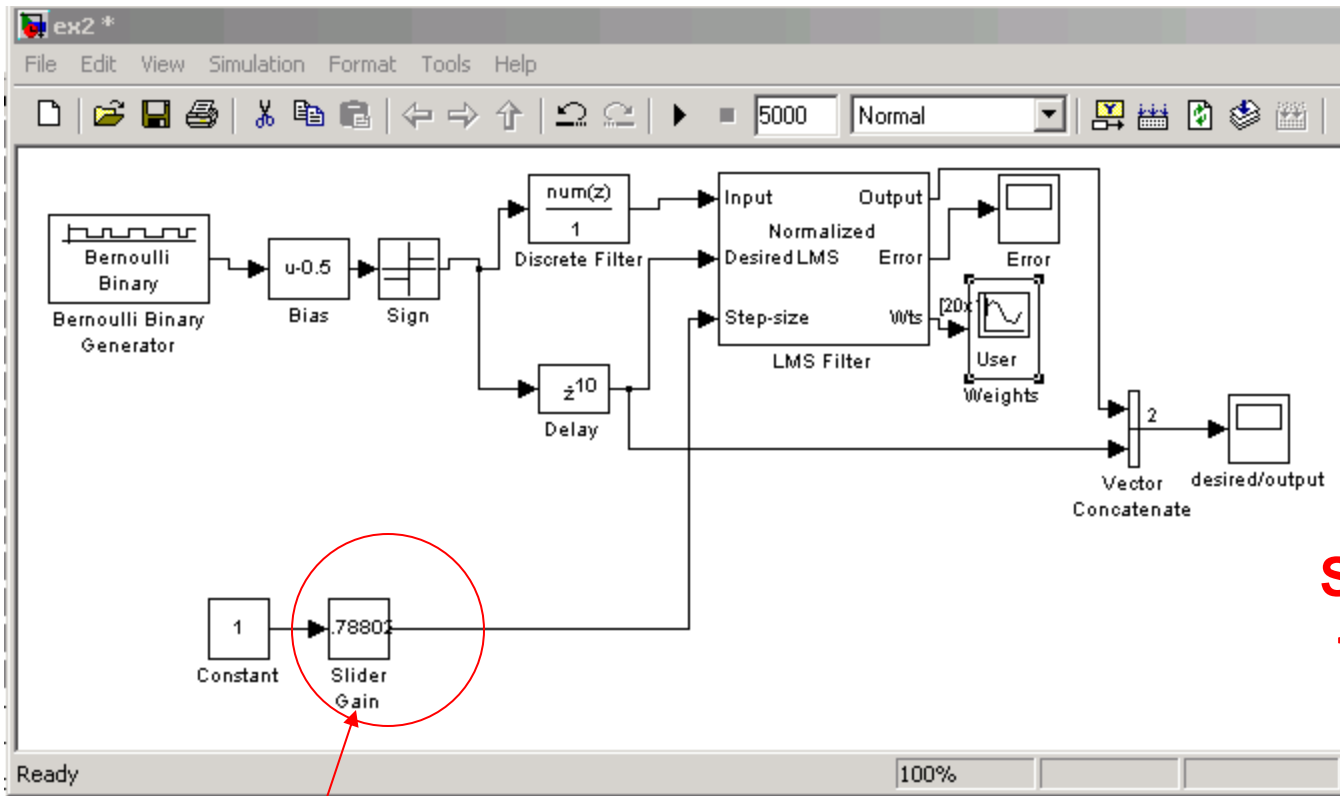
Output filter weights

OK Cancel Help Apply

Leakage=1 \Leftrightarrow no leakage

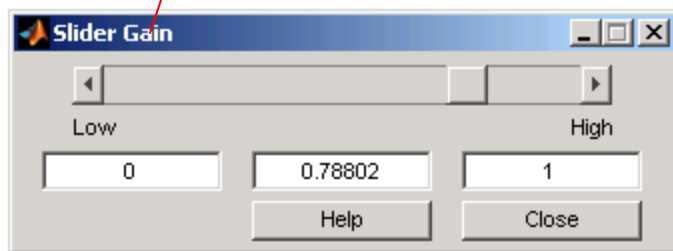
Check to allow filter coef adaptation based on external non-zero value

Check if you want to get the filter coefficient values out



LMS step size can be varied using the “slider gain”

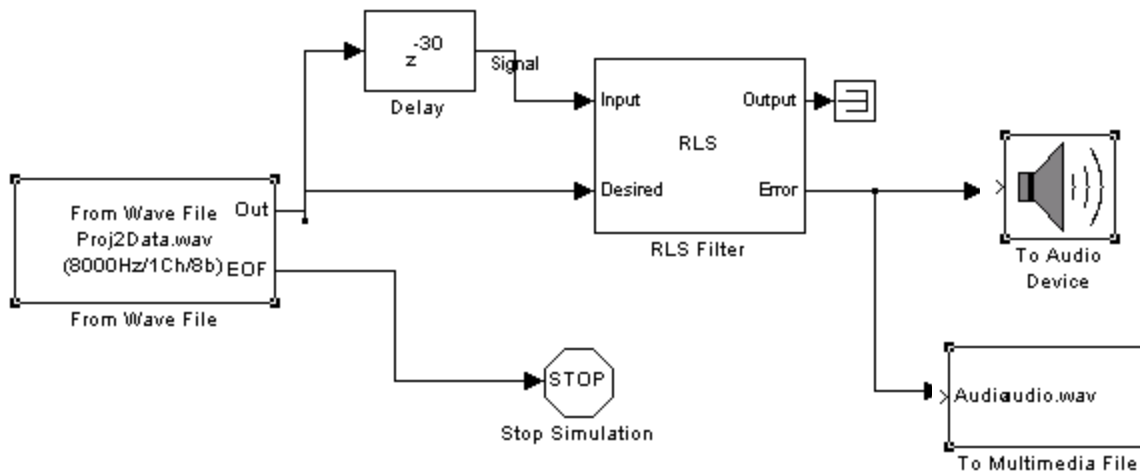
**Simulink
→ Math Operations**



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



LMS Adaptive Equalization



Call the RLS algorithm from:

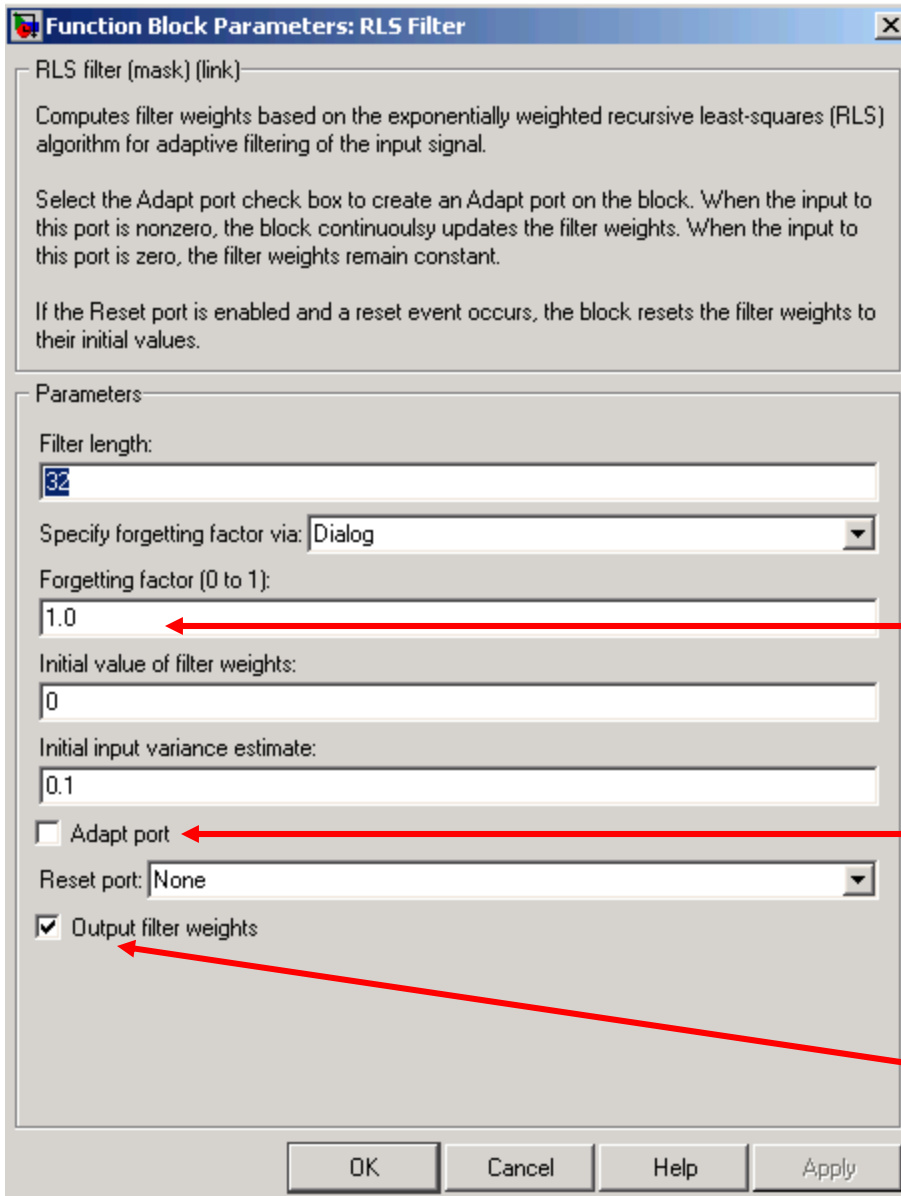
DSP System Toolbox

→ **Filtering**

→ **Adaptive Filters**

→ **RLS**

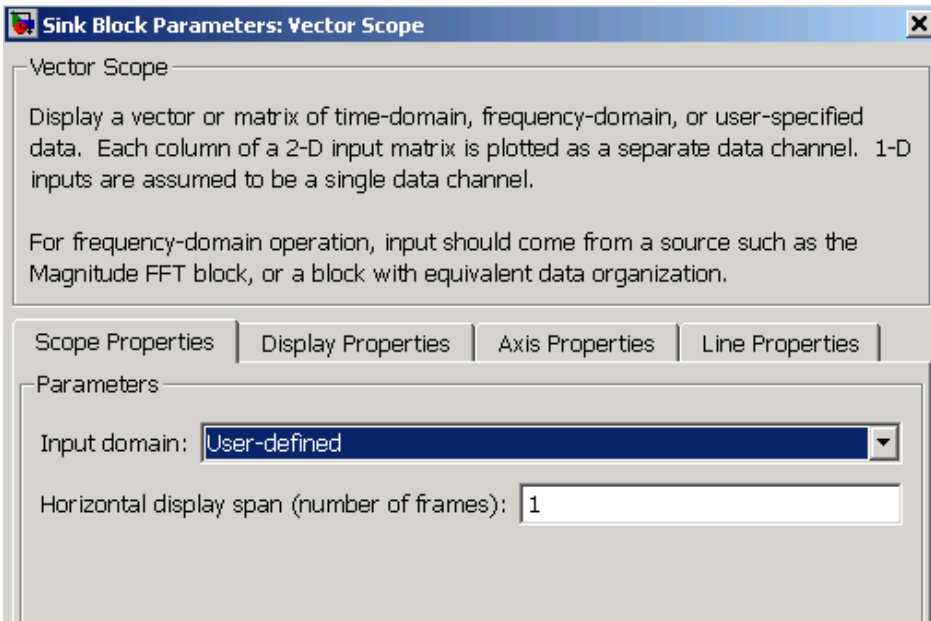
RLS configuration parameters



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

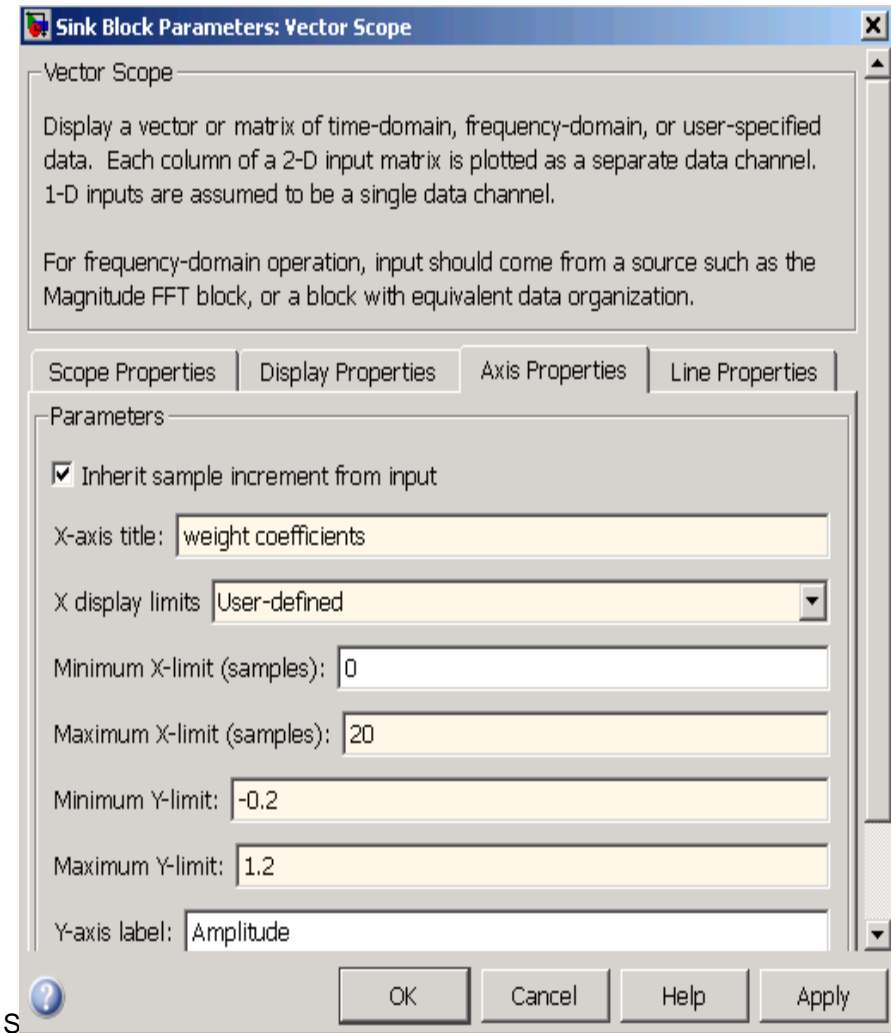
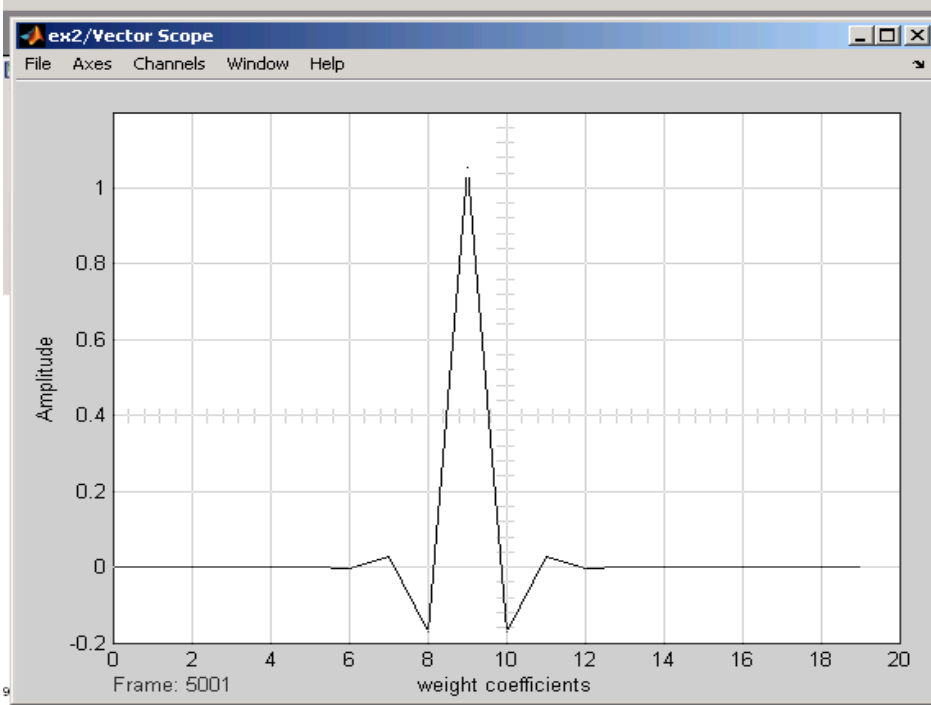


12) Plot filter coefficients using the vector scope

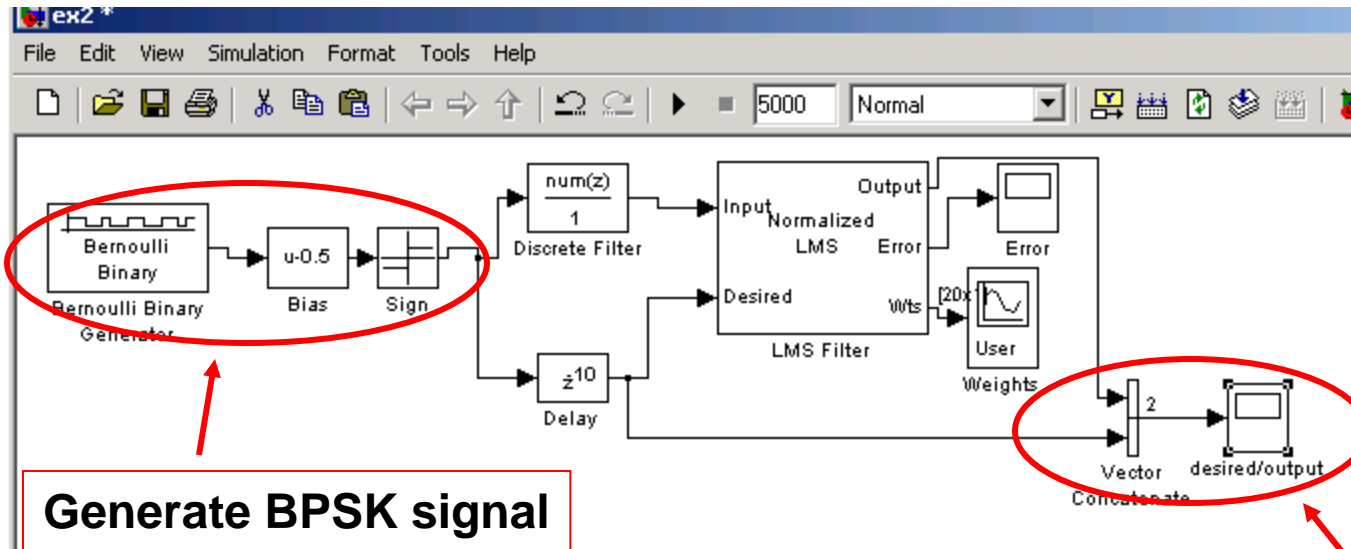
→ DSP System Toolbox

→ Signal Processing Sinks

→ Vector Scope



13) Plot multiple data streams on the same figure

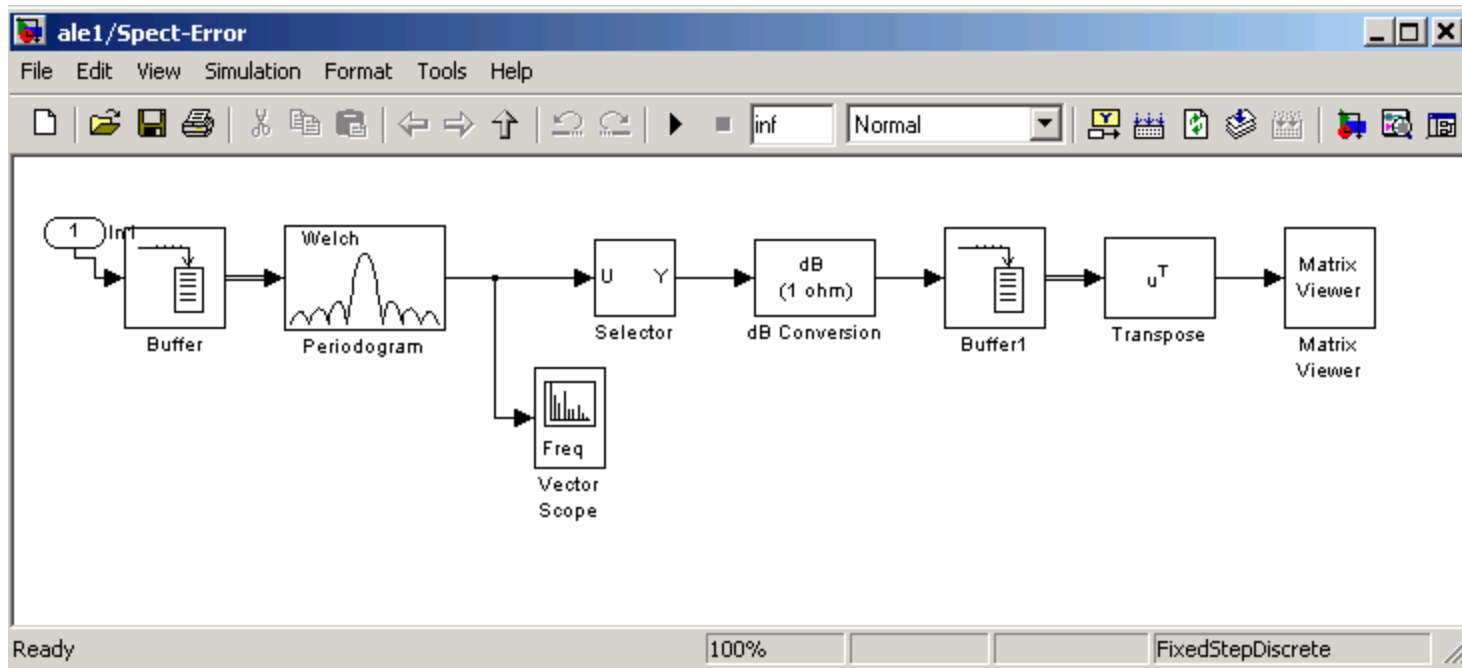


**Use “vector concatenate”
+ Regular scope**

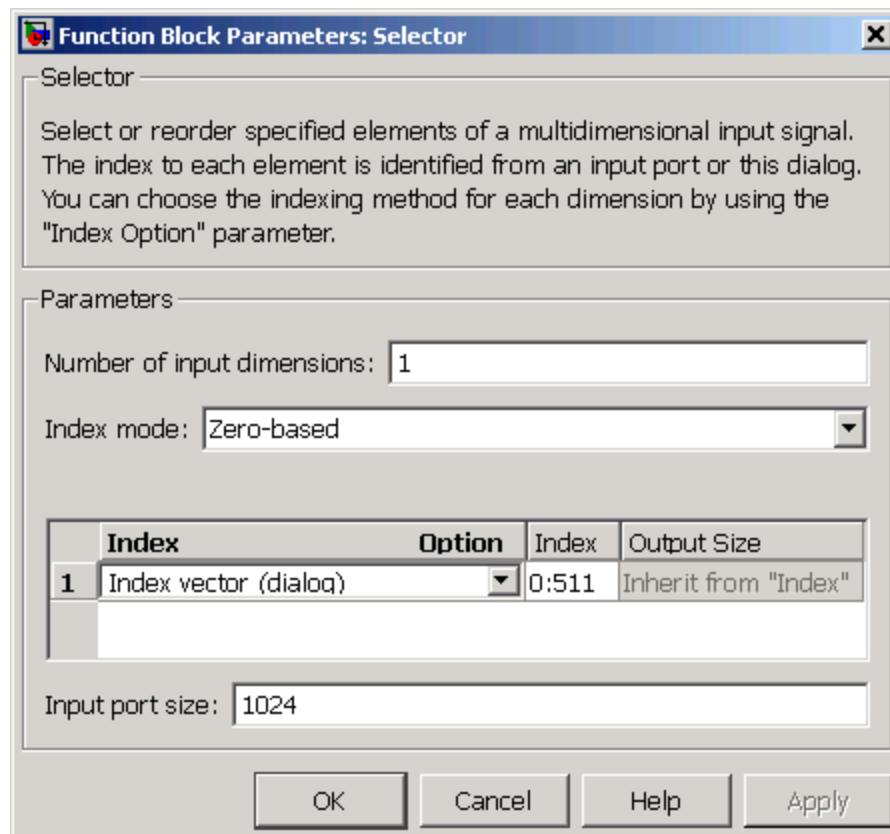
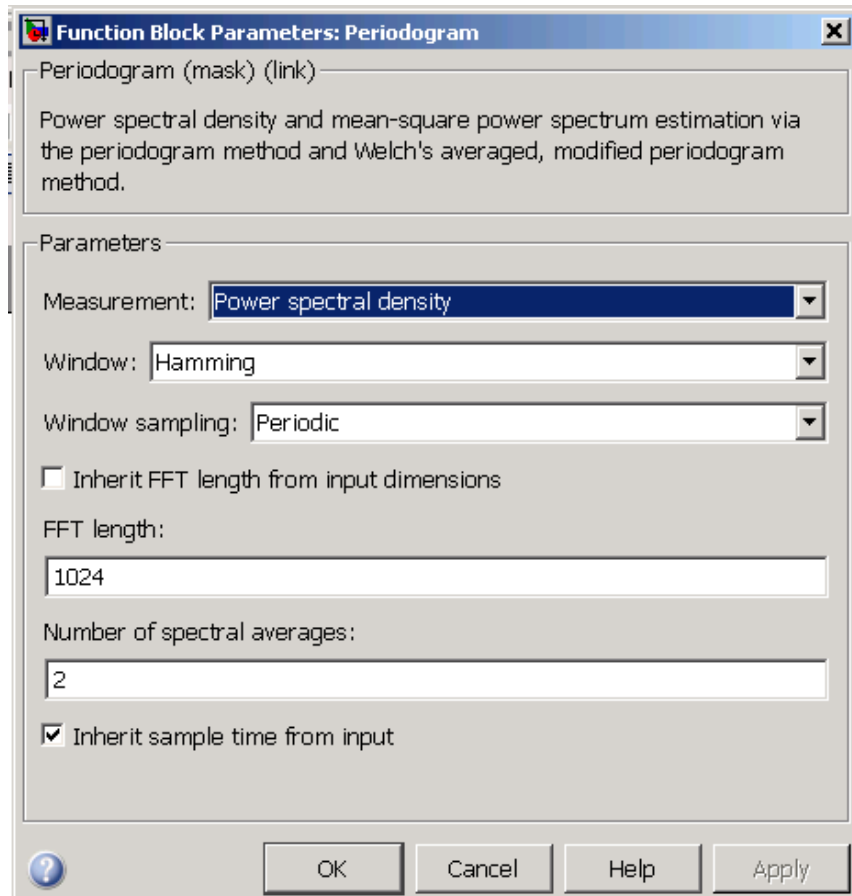
**Simulink → Commonly used Blocks
→ Vector Concatenate
→ Scope**

14) Generate spectrum and spectrogram plots

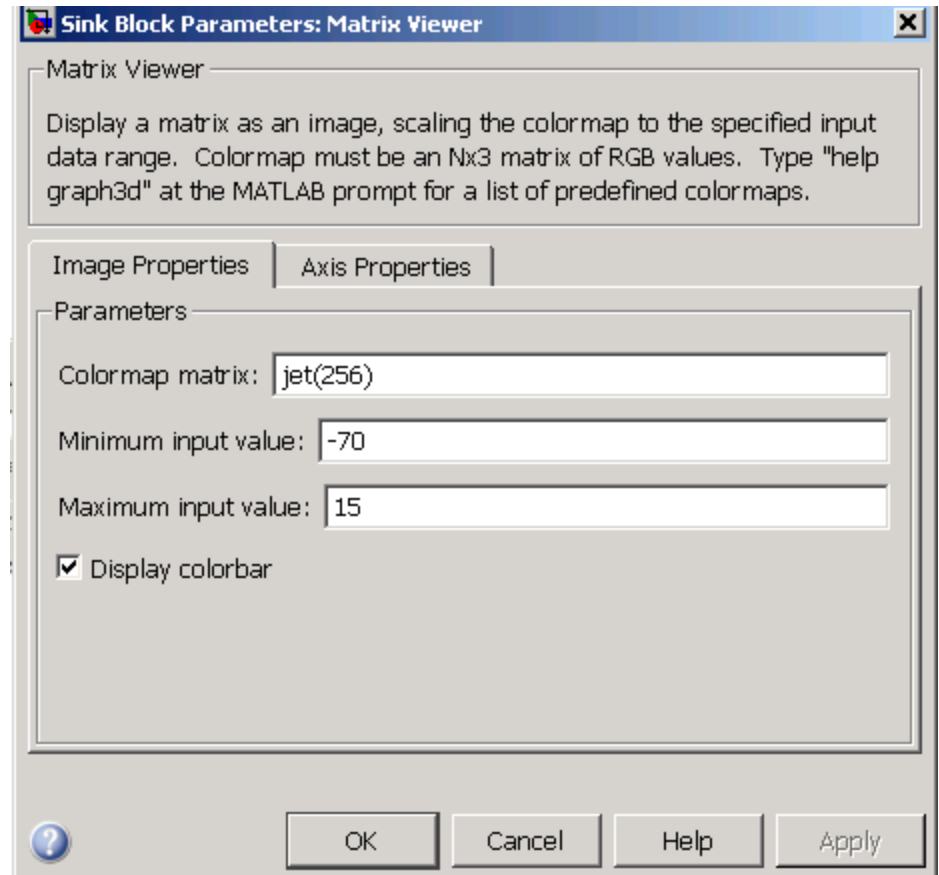
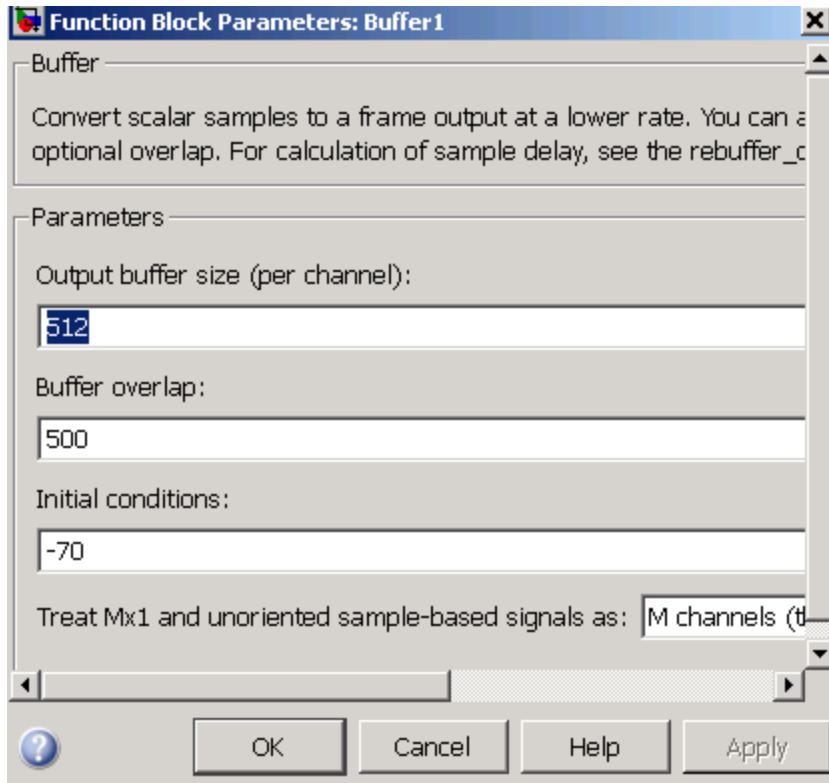
→ Specta.mdl (provided in course material)



Blocks used in spectra.mdl



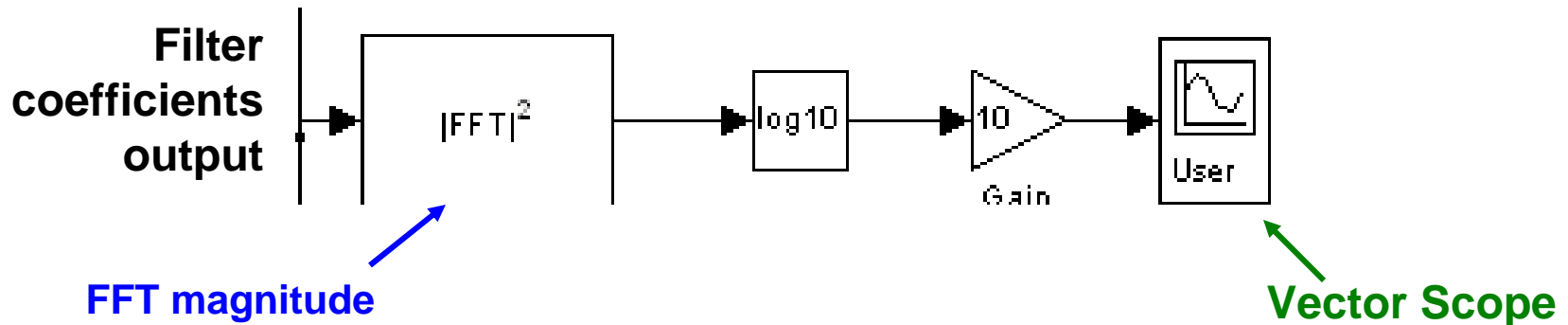
Blocks used in spectra.mdl, cont'



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.



FFT magnitude

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

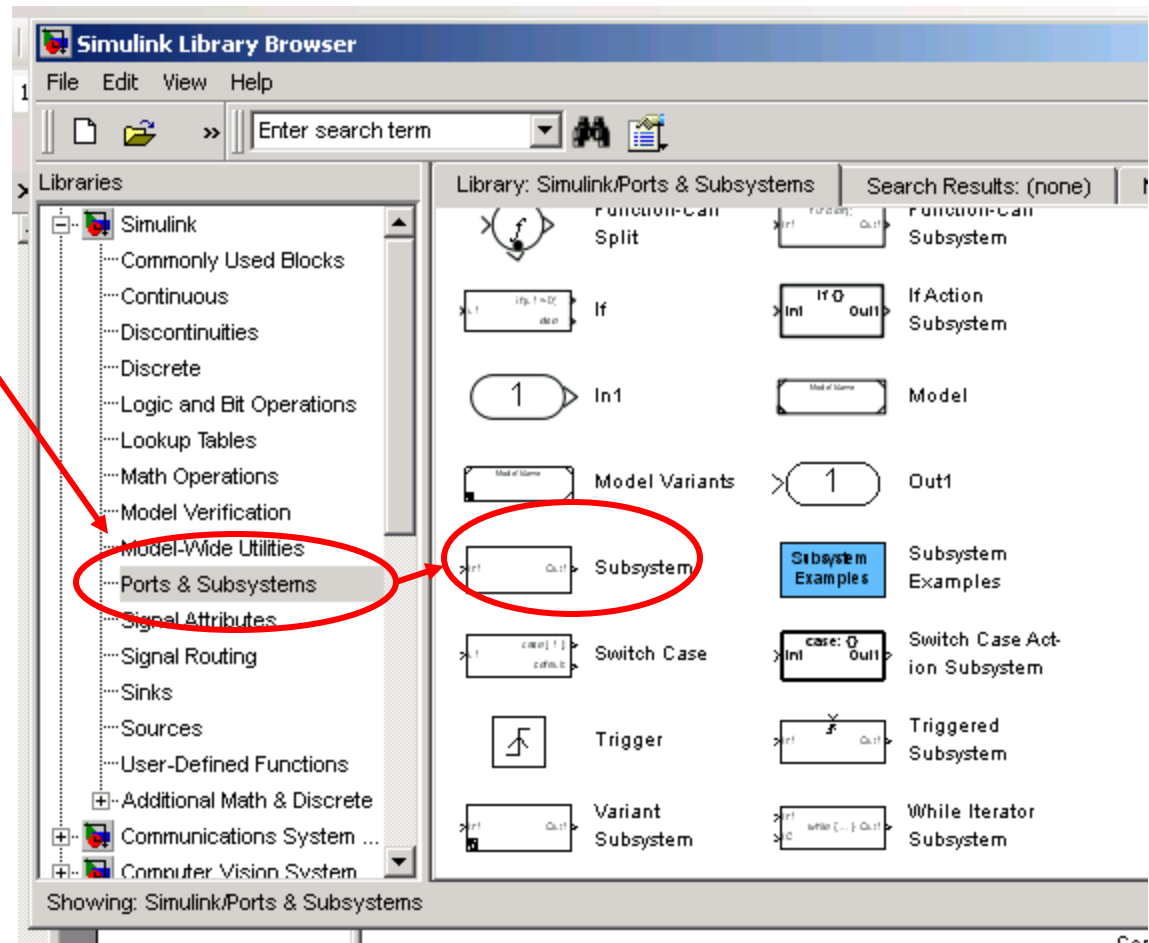
16) Listen to audio signals (Batch mode from Simulink)

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

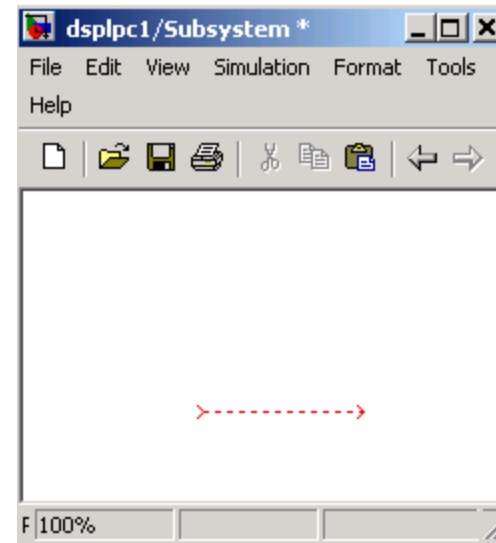
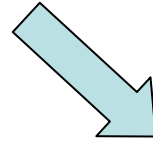
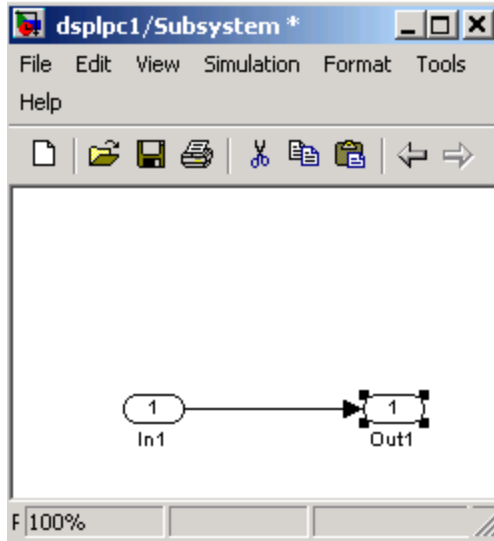
Simulink

→ Ports & Subsystems

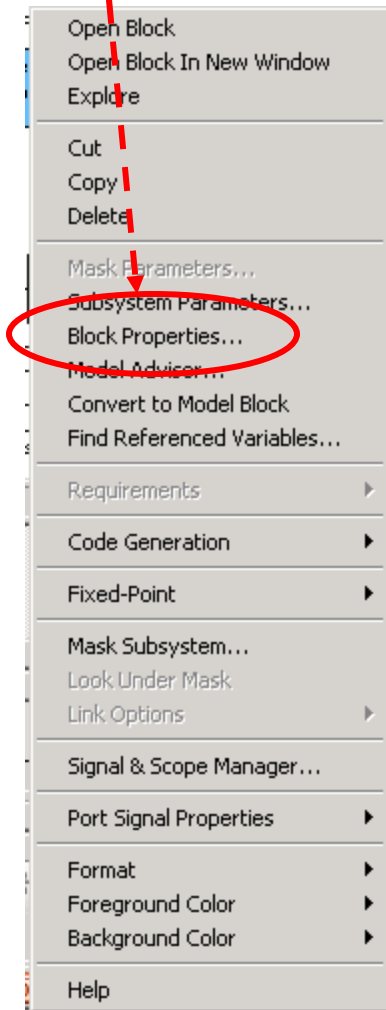
→ Subsystem



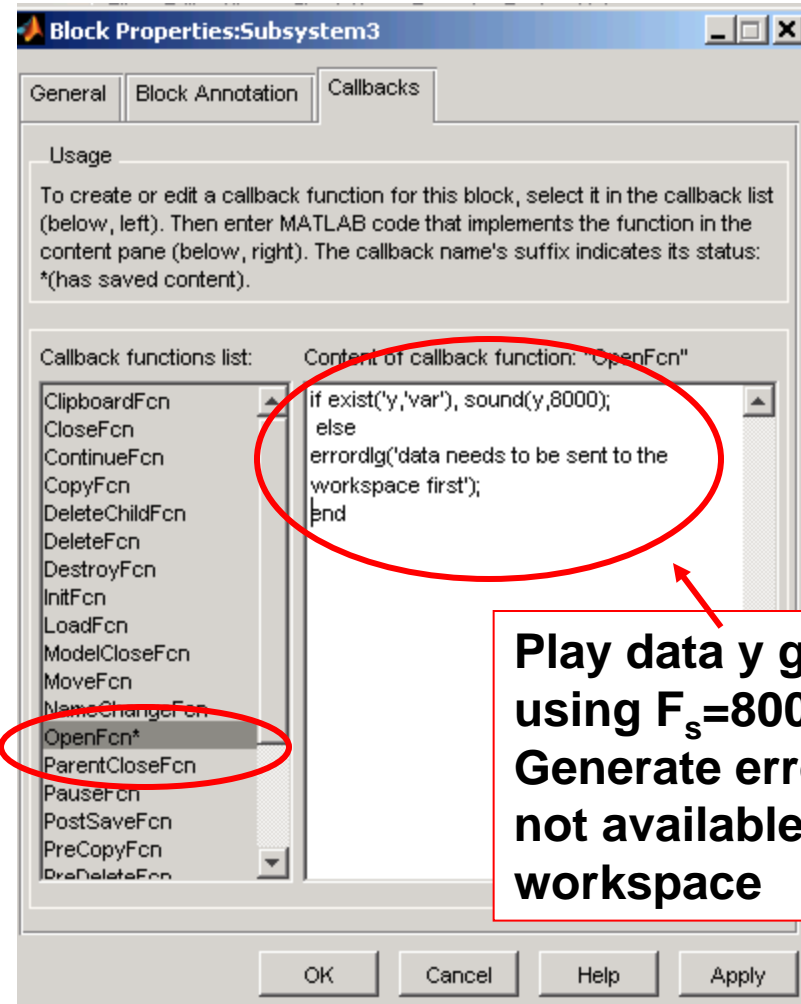
c) Remove subsystem input/output ports



d) Code audio play action by accessing system block properties



e) In-code audio play commands



Play data y generated using $F_s=8000\text{Hz}$ & Generate error if data is not available in the workspace

