

How to Control Tabor AWGs with MATLAB

Example2: Generate a sequence from 3 waveform files

In this tutorial, we will give an example on how to load 3 waveform files and a sequence table into the Tabor AWG memory, to generate a sequence. This example is done using the IVI driver.

If you haven't been acquainted with controlling Tabor AWGs using MATLAB's Instrument Control Toolbox, please refer to [Tabor's website](#) for the previous tutorials in the series "How to Control Tabor AWGs with MATLAB":

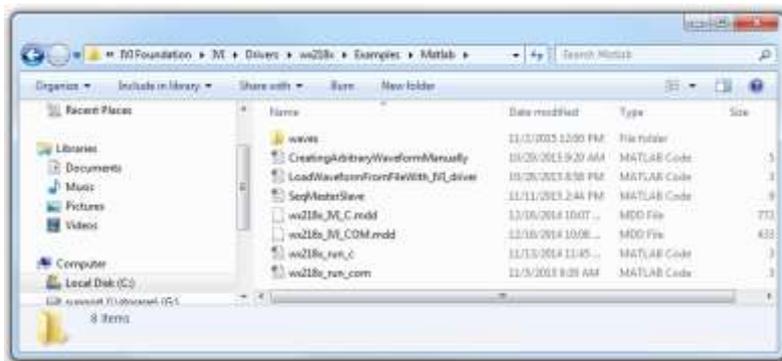
- 'Getting started'.
- 'Using SCPI commands'.
- 'Using the IVI driver'.

In order to control instruments using MATLAB, the instrument control toolbox is required. Please note that the Instrument Control Toolbox is an additional application that needs to be added. For more information you can visit the [Mathworks](#) website.

For this tutorial, we will use MATLAB version R2014a 32bit and a USB interface. To ensure you successfully established all the necessary settings for remote control over the Tabor instrument using LAN/USB/GPIB, please go over the [connectivity tutorials](#) on the Tabor's website.

The 'wx218x_run_com.m' itself as well as the waveforms (in the 'waves' folder), can be found after installation of the latest Tabor IVI driver through this path on your computer:

'C:\Program Files (x86)\IVI Foundation\IVI\Drivers\wx218x\Examples\Matlab'.



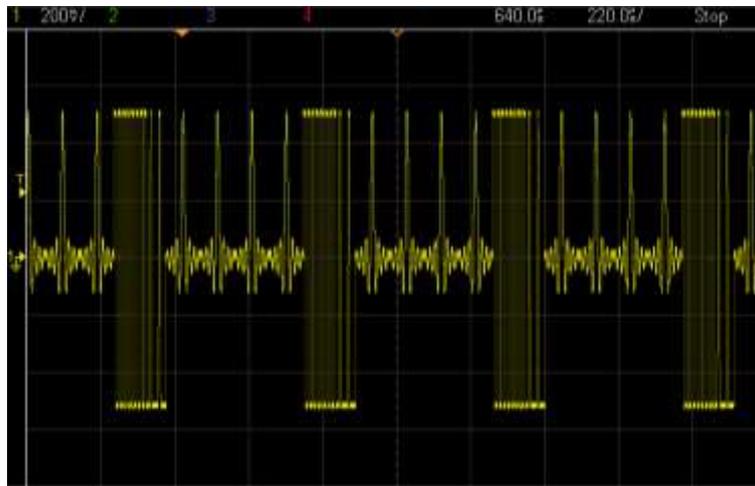
For this example, we won't go over the steps one by one. For information regarding the steps, please read the detailed comments in the code itself.

```

%% The following script 'wx218x_run_com.m' demonstrates how to create a sequence
% with a wx2184C waveform generator using USB connection,
% using the wx218x IVI-COM driver.
% Author: Irina Tseitlin
% email: support@taborelec.com
% Copyright (C) 2011-2015 Tabor Electronics Ltd
ACTIVE_CHANNEL = 'Channel1';
% Open connection to instrument with IP address or USB address
% (uncomment your connection type and change the address to your address. Use arbconnection or NI-MAX
%to verify your USB address)
%dev = icdevice('wx218x_IVI_COM.mdd',
%'TCPIP0::192.168.0.175::5025::SOCKET','optionstring','simulate=false,DriverSetup=Trace=True,TraceArr
%ay=true');
dev = icdevice('wx218x_IVI_COM.mdd',
'USB0::0x168C::0x2184::0000215470','optionstring','simulate=false');
try
connect(dev);
groupId = get(dev, 'Identity');
InstrModel = groupId.InstrumentModel;
% Reset device
groupCnf = get(dev, 'Utility');
invoke(groupCnf, 'Reset')
% Set the sample clock
groupArb = get(dev, 'Arbitrary');
groupArb.SampleRate = 80E6;
% Create three segments in the channel A and load waves in them
groupArb = get(dev, 'ArbitraryWaveform');
% Doing this only for WX 4 channels
if ((strcmp(InstrModel, 'WX2184')) == 1) || ((strcmp(InstrModel, 'WX1284')) == 1) ||
(strcmp(InstrModel, 'WX2184C')) == 1) || ((strcmp(InstrModel, 'WX1284C')) == 1))
groupArb.TraceMode = 'WX218xTraceModeDuplicate';
end
% Downloading the waveforms
wavesdir = [pwd '\waves\'];
wfmHandle1 = invoke(groupArb, 'LoadArbWfmFromFile', ACTIVE_CHANNEL, [wavesdir 'pulse_2048.wav']);
wfmHandle2 = invoke(groupArb, 'LoadArbWfmFromFile', ACTIVE_CHANNEL, [wavesdir 'sinc_8192.wav']);
wfmHandle3 = invoke(groupArb, 'LoadArbWfmFromFile', ACTIVE_CHANNEL, [wavesdir 'square_1024.wav']);
% Create the sequence in the active channel, currently the active
% channel is Out1 because the 'LoadArbWfmFromFile' function was used with the 'Channel1'
% parameter
% Sequence Description:
% Step #          Segment #          Repeats Count  jump flag
% [ 1 ]          [ Segment 1 ( pulse_2048.wav ) ]          [ 2 ]          [ 0 ]
% [ 2 ]          [ Segment 2 ( sinc_8192.wav ) ]          [ 4 ]          [ 0 ]
% [ 3 ]          [ Segment 3 ( square_1024.wav) ]          [ 8 ]          [ 0 ]
wfmHandles = [wfmHandle1; wfmHandle2; wfmHandle3];
loop = [2; 4; 8];
flag = [0; 0; 0];
% Create the sequence in the active channel.
groupSeq = get(dev, 'Arbitrarysequence');
seqHandle = invoke(groupSeq, 'CreateSequenceAdv', int32(wfmHandles), int32(loop), uint8(flag));
% set the Output Mode to Sequence
groupOutput = get(dev, 'Output');
groupOutput.OutputMode = 'WX218xOutputModeSequence';
% Output enabled
invoke(groupOutput, 'Enabled', ACTIVE_CHANNEL, 1)
% Output SYNC signal enable
invoke(get(dev, 'Outputsync'), 'Enabled', ACTIVE_CHANNEL, 1)
% Close the connection with instrument
disconnect(dev);
delete(dev);
catch aException
disp(aException.message);
% Close the connection with instrument
disconnect(dev);
delete(dev);
end

```

The resulting sequence should appear on scope as shown below:



The outputted sequence.



The outputted sequence – zoom in on one cycle.

In the next tutorials, we will show 2 more examples:

- Example #3: Synchronize Between Two Tabor AWGs Using The IVI Driver Functions.
- Example #4: Create an arbitrary waveform from binary data using SCPI commands.

For More Information

To learn more about how to use MATLAB with Tabor instruments, visit our website Support & Tutorials zone. For more of Tabor's solutions or to schedule a demo, please contact your local Tabor representative or email your request to info@tabor.co.il. More information can be found at our website at www.taborelec.com

© Proprietary of Tabor Electronics Ltd.