Getting Started Guide

Using MATLAB with LeCroy Oscilloscopes over a TCP/IP Connection

This guide shows you how to verify within 15 minutes that you can use MATLAB® with your LeCroy® oscilloscope over a LAN (TCP/IP) connection. MATLAB enables LeCroy oscilloscope users to acquire and analyze data, graphically visualize data, make custom measurements, generate reports, and develop automated applications.

If you have any questions or problems completing the steps in this guide, *contact MathWorks Technical Support.* MathWorks can also provide Getting Started Guides for using your instrument with a GPIB connection.

System Requirements for Completing This Guide

1. A LeCroy oscilloscope with a TCP/IP interface supporting remote instrument communication.

2. MATLAB and Instrument Control Toolbox™ installed on a computer.

Contact Sales at 508-647-7000 if you would like a trial of MATLAB and Instrument Control Toolbox, need assistance finding a MATLAB instrument driver for your instrument, or have any questions about the system requirements.

Prepare Your System

Step 1: Start Test & Measurement Tool

Start MATLAB and issue the following command to launch the Test & Measurement Tool:

```
tmtool
```

The Test & Measurement Tool (Figure 1) is the user interface for configuring and controlling instruments in MATLAB.
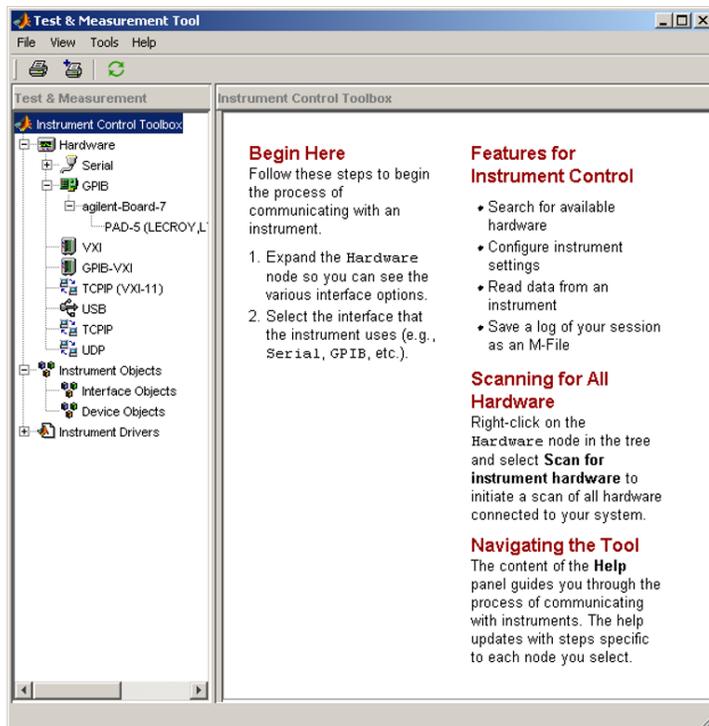

Figure 1. The Test & Measurement Tool from Instrument Control Toolbox.

Step 2: Connect your instrument

If you are using MATLAB and Instrument Control Toolbox to remotely control your instrument, connect both your instrument and computer to your TCP/IP network. Otherwise, if you are working locally on the instrument, your instrument does not need to be configured on the network.

In both cases, on your LeCroy instrument, select **Utilities > Utilities Setup**, and choose the **Remote** tab. Ensure that under **Control From** the **TCP/IP** button is selected. Your instrument's IP address is listed in the **TCP/IP** panel of the **Remote** tab. Note this IP address for later. If you are working locally on the instrument, the IP address you will use is 127.0.0.1.

You may verify connectivity to your instrument by issuing the following command from MATLAB:

```
!ping <instrument-IP>
```

where `<instrument-IP>` is the IP address of your instrument.

Step 3: Locate an instrument driver

MATLAB with Instrument Control Toolbox supports instrument drivers, which allow you to communicate with your device without knowing low-level instrument commands. To see if your instrument driver is already installed, right-click the `Instrument Drivers` node and select **Scan for Instrument Drivers**. Under the `Instrument Drivers` node, look for a driver that matches your device. If you do not have a matching driver installed, *more are available*. Note: The driver called "LeCroy Oscilloscopes – Basic Instrument Driver" is expected to work for most modern LeCroy oscilloscopes.

Downloaded drivers must be placed in a directory on the MATLAB path such as `<matlabroot>\work`. Right-click `Instrument Drivers` and choose **Scan for Instrument Drivers** to refresh the current list of drivers.

Connect to the Instrument

Step 4: Create the interface object

Navigate to the `Interface Object` node under the `Instrument Objects` node. Click **New Object** in the lower right. The resulting dialog box is shown in Figure 2. Choose **TCPIP** for the **Instrument object type**, enter the IP address (from Step 2) into the **Remote Host** field, and click **OK.**
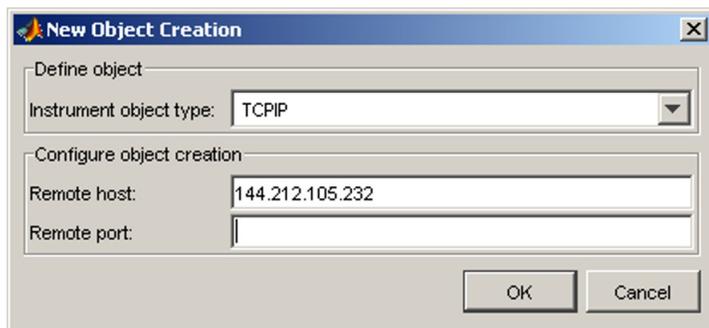
Figure 2. Interface creation dialog box.

Step 5: Create the device object

Under the `Instrument Drivers` node, right-click the instrument driver for your device and select **Create Device Object Using Driver** as in Figure 3. In this example, the instrument is a LeCroy WaveMaster 8600A. In the resulting dialog box, select the interface object you created in Step 4. Click **OK** to complete the creation of the device object.
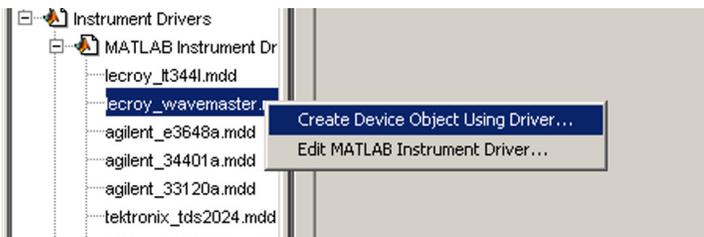

Figure 3. Creating an instrument object from a driver.

Step 6: Connect to the device object

Select the newly created instrument object, which in this example is labeled `scope-lecroy _ wavemaster`, under the `Device Objects` node. Click **Connect** in the upper right corner, and you are ready to start interacting with your device.

Interact with the Instrument

Step 7: Query a property

Select the instrument object, click the **Properties** tab, navigate to the `Timebase` property in the `Acquisition` group, and click **Get** as in Figure 4. The results appear in the **Response** pane on the right.
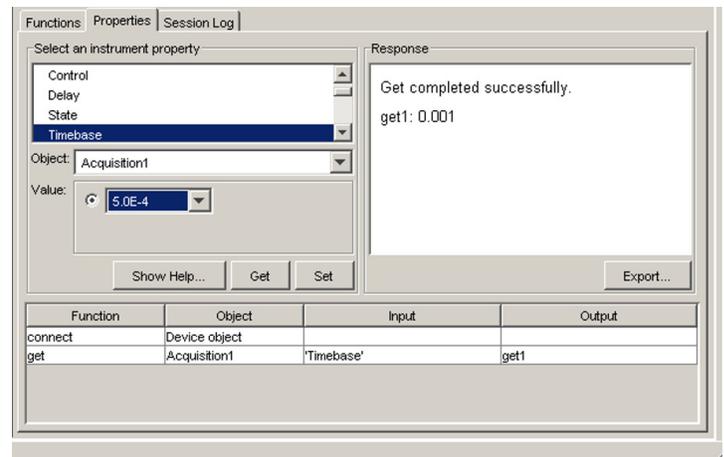
Figure 4. The result of getting the `Timebase` property.

Step 8: Configure a property

Navigate to the `Timebase` property, choose a value for the property, and click **Set**. The **Response** pane indicates if the set succeeded or displays an error message indicating the reason if it failed.

Step 9: Invoke a function

View the instrument's functions by choosing the **Functions** tab. To call the `readwaveform` function, first scroll to `readwaveform` under the Waveform group in the function list. Then enter input argument `'channel1'` (include the single quotes) to read the first channel. Next enter output arguments *Y, X, YUNIT, XUNIT, HEADER* as a comma-separated list. Finally, click **Execute** to read the waveform. The results appear in Figure 5.
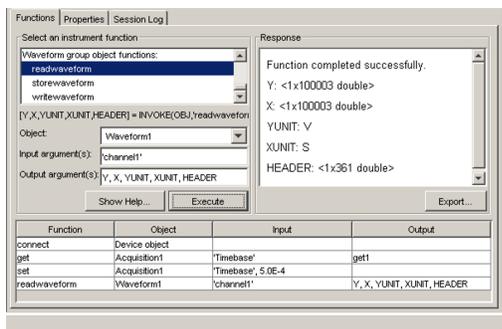
Figure 5. Invoking `readwaveform` of an instrument object.

Step 10: Visualize results

From the **File** menu, select **Export > Instrument Responses.** To plot the waveform (variable *Y*) from the resulting dialog box (Figure 6), check only the box to the left of variable name *Y* and uncheck all others.

In the **Data Destination** list, choose **MATLAB Figure Window** and click **Export**. This will create a MATLAB Figure window displaying the waveform you captured.
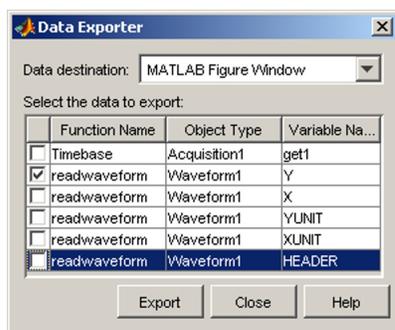
Figure 6. Data export dialog box.

Step 11: Analyze results

You may further analyze your data by exporting it to the MATLAB workspace. From the **Data Exporter** (Figure 6), choose **MATLAB Workspace** as the data destination, then click **Export**. Note: Once your values and waveforms are in the MATLAB workspace, you may use them with MATLAB functions or scripts including ones you create (not shown here).

Step 12: Reuse your session

The Test & Measurement Tool automatically generates MATLAB code for all of your interactions with your instrument. Select the **Session Log** tab of your instrument object to view the MATLAB code (Figure 7). Furthermore, using the **Save Session** button, you may export a code file that you can use to repeat your steps or develop a larger application.
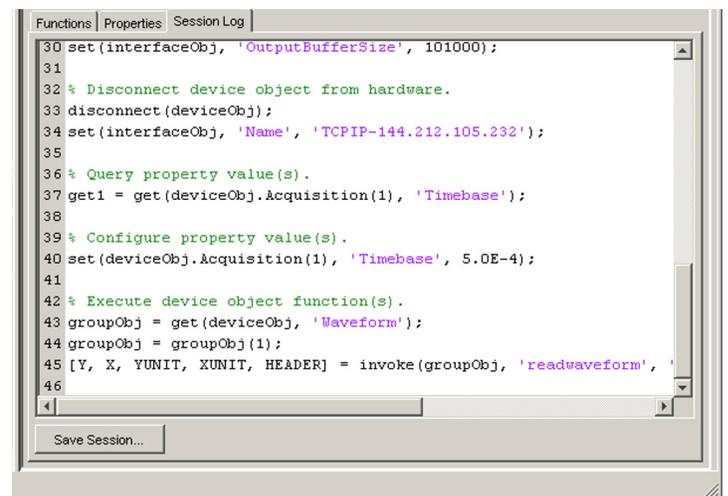

Figure 7. Viewing the code from your session.

If you have successfully completed all these steps, you have verified that you can use MATLAB and Instrument Control Toolbox with your instrument.