

March 21, 2012

### Summary

*Visual Basic scripting is used to internally program the scope and automate selected scope operations. Visual Basic scripts, when executed, can control and automate scope operations*

### Overview of Automation

In addition to supporting the familiar ASCII-based remote commands that have been used to control all LeCroy DSOs for many years, all of the Windows-based “X-Stream” instruments fully support control by Automation interfaces based on Microsoft’s Component Object Model (COM). Using COM, the controlling application can run directly on the instrument without requiring an external controller.

Scope setup files, in X-Stream DSOs, are ASCII text files that contain a complete Visual Basic Script “program” that, when “executed,” will restore the instrument to a previously recorded state. In effect, each time a panel is saved, the instrument effectively writes you a program that, when executed, returns the instrument to the saved state.

This tutorial presents a procedure of how to create a simple remote control application, which will run on the instrument, from scratch. It doesn’t rely upon any third-party development tools, since it uses Windows’ built-in text editor (Notepad) and the Visual Basic Script interpreter (VBScript), which is also installed on all LeCroy XStream oscilloscopes.

Users are strongly urged to consult LeCroy’s Automation manual on the LeCroy Website at:

[http://cdn.lecroy.com/files/manuals/entire\\_x-tream\\_automation\\_manual.pdf](http://cdn.lecroy.com/files/manuals/entire_x-tream_automation_manual.pdf)

This manual is a comprehensive resource on all the aspects of automation programming. Among the topics discussed is the use of the XStream Browser. This application connects to the scope and exposes the automation items. It will let you make changes to the scope from it and you can also make a change on the scope and then refresh the browser to read the new value. (This is a nice way to find and test commands). The XStream browser introduction can be found in the remote manual on page 72 and in the Automation Manual on page 1-5.

---

## Equipment Required

WaveSurfer MXs series oscilloscope

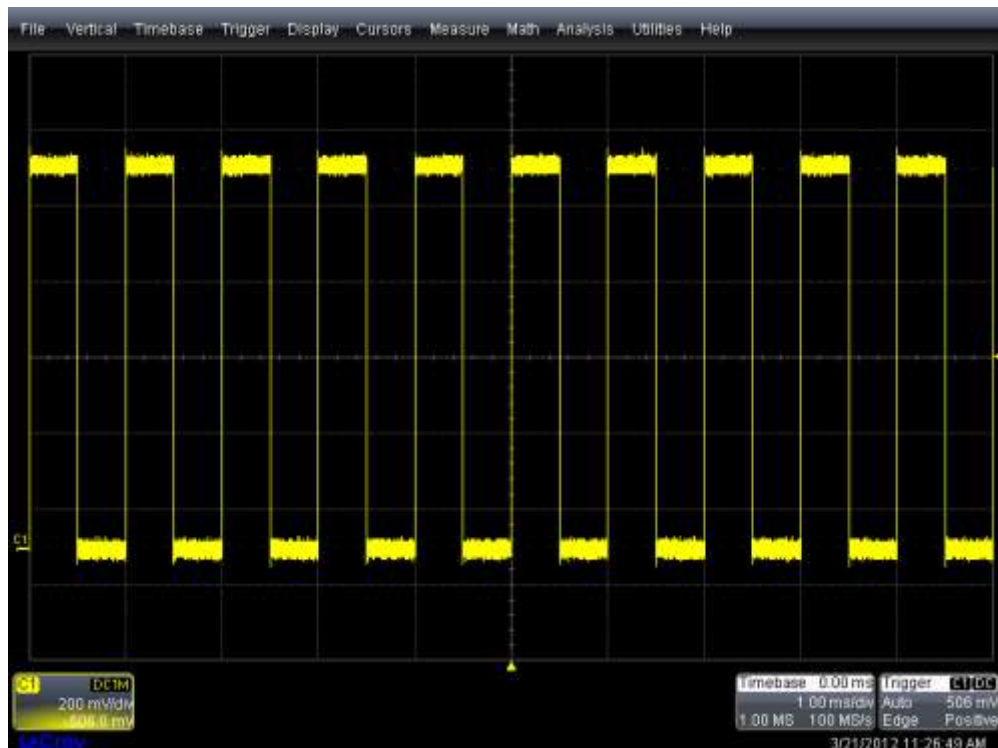
Passive probe

## Initial Setup

Displays shown in the tutorial are based on the following initial setup on a WaveSurfer MXs scope:

1. Connect the passive probe from the channel 1 input to the Cal test point on the front panel of the oscilloscope.
2. Recall the default setup: File pull down > Recall Setup> Recall Default.
3. Turn off channel 2.
4. Auto Setup the scope: Press AutoSetup twice.

This completes the initial setup. The scope display should be similar to Figure 1.



**Figure 1:** The initial setup for this tutorial, using the 1 kHz calibrator square wave

As mentioned in the introduction, LeCroy XStream oscilloscopes create a Visual Basic program when storing panel setups. Figure 2 shows the first few lines of such a panel setup.

```
On Error Resume Next
set XStreamDSO = CreateObject("LeCroy.XStreamDSO")
set RecallSetupLocker = XStreamDSO.RecallSetupLock

' AladdinPersona ...
XStreamDSO.HideClock = False

Set ProbesCal = XStreamDSO.ProbesCal
' ProbesCal ...
ProbesCal.CalSource = "FastEdge"
ProbesCal.DeskewCalRef = "C2"
ProbesCal.DeskewAll = 0
ProbesCal.CommonMode = 0
ProbesCal.Gain1 = 1
ProbesCal.Offset1 = 0
ProbesCal.Gain2 = 1
ProbesCal.Offset2 = 0
ProbesCal.Gain3 = 1
ProbesCal.Offset3 = 0
ProbesCal.Gain4 = 1
ProbesCal.Offset4 = 0
ProbesCal.SkewCal1 = 0
ProbesCal.SkewCal2 = 0
ProbesCal.SkewCal3 = 0
ProbesCal.SkewCal4 = 0
ProbesCal.SkewCal5 = 0
ProbesCal.AdvancedMode = False

Set CustomDSO = XStreamDSO.CustomDSO
' CustomDSO ...
CustomDSO.ActionScript1 = "D:\CustomDSO\custom 1.lss"
CustomDSO.ActionEnable1 = True
CustomDSO.ActionScript2 = "D:\CustomDSO\custom 2.lss"
CustomDSO.ActionEnable2 = True
```

**Figure 2:** Part of a setup (.lss) file showing Visual Basic commands

The second line of the setup file contains a Create Object Statement

## The CreateObject Statement

```
set XStreamDSO = CreateObject("LeCroy.XStreamDSO")
```

CreateObject is the Visual Basic function that creates an instance of a COM Server (a.k.a. ActiveX Control). The argument "LeCroy.XStreamDSO" refers to our DSO application. Once it has instantiated (connected to) our DSO application we need some kind of 'handle' (pointer) to it so that we can use it later to communicate with the instrument. CreateObject returns a handle to us, which we store in the XStreamDSO variable.

We will use a similar approach in developing a Visual Basic script to automate the scope operation.

Let's start by creating a Visual Basic script that will use the X1horizontal relative cursor location to control the center location of a zoom trace. So as you move the cursor the zoom trace will follow the X1 cursor location.

Here is the script created in the NotePad Windows application:

```
set app = CreateObject("LeCroy.XStreamDSO")

' Display a message on the display
app.SystemControl.PersistentMessage = "Script running; turn off cursor to stop."

' Change the trigger mode to stopped
app.Acquisition.TriggerMode = "Stopped"

' set the cursor type to horizontal absolute
app.Cursors.Type ="HorizRel"

' Turn the cursor on
app.Cursors.View = True

'set the zoom 1 trace horizontal expansion to 10:1
app.Zoom.Z1.Zoom.HorZoom = 10

' Turn on the zoom 1 trace
app.Zoom.Z1.View = True

' Loop to have zoom center track cursor horizontal location, exit when cursor is turned off
While app.Cursors.View = True

    ' force trigger immediately
    'set arguments to 0,False to wait for a triggerable event.
    app.Acquisition.Acquire -1,True

    'Read cursor horizontal location
    curtime=app.Cursors.XPos1

    'Set Zoom Z1 center to cursor location
    app.Zoom.Z1.Zoom.HorCenter=curtime
Wend

' Clear the message on nthe screen
app.SystemControl.PersistentMessage = ""

' Disconnect the automation link
Set app = Nothing

' Display a Windows pop up message
MsgBox "Script Complete!"
```

First we use the creatobject statement to link to the scope using the variable "app".

Display a message that a script is running.

Set the cursor type to Horizontal Relative and turn it on

Zoom trace Z1 is set to a 10:1 horizontal expansion and turned on.

Start a While Loop to run until the cursor is turned off.  
Force a trigger

Read the horizontal cursor location

Pass the horizontal X1 cursor location to the zoom center location.

loop back to the start of the While loop.

If the cursor is turned off exit the While loop, clear the 'script running' message

Clear the create object variable.

Display a message that the script is complete.

### **Creating the Script**

Open Notepad and copy the script into Notepad

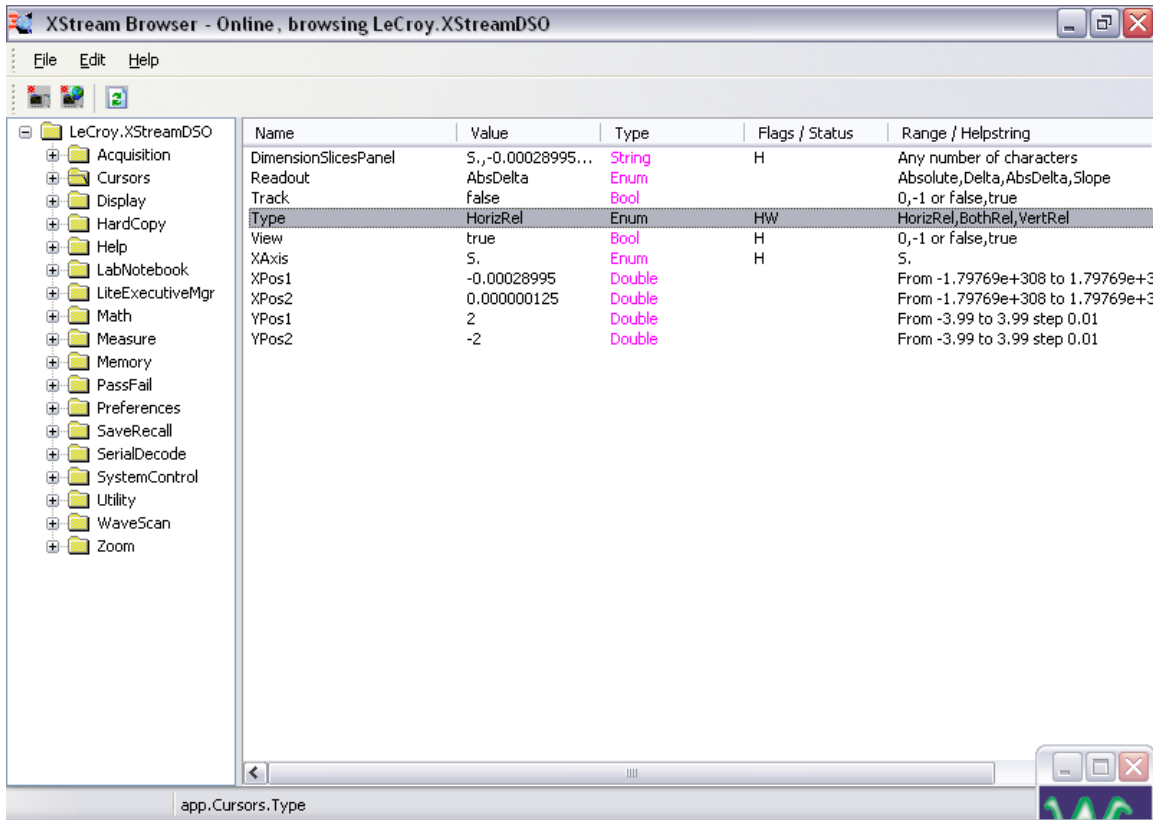
Save the script using any name with a LeCroy Setup Script.lss file extension.

## The XStream Browser

The context of each of the automation commands used above can be found using the XStream Browser or from the Automation Command Manual:

([http://cdn.lecroy.com/files/manuals/automation\\_command\\_ref\\_manual\\_wm-wp.pdf](http://cdn.lecroy.com/files/manuals/automation_command_ref_manual_wm-wp.pdf))

Figure 3 shows the XStream Browser with the Cursor Type command highlighted. The right hand column shows the possible arguments. By right clicking on the browser entry you can change or copy the current setting.

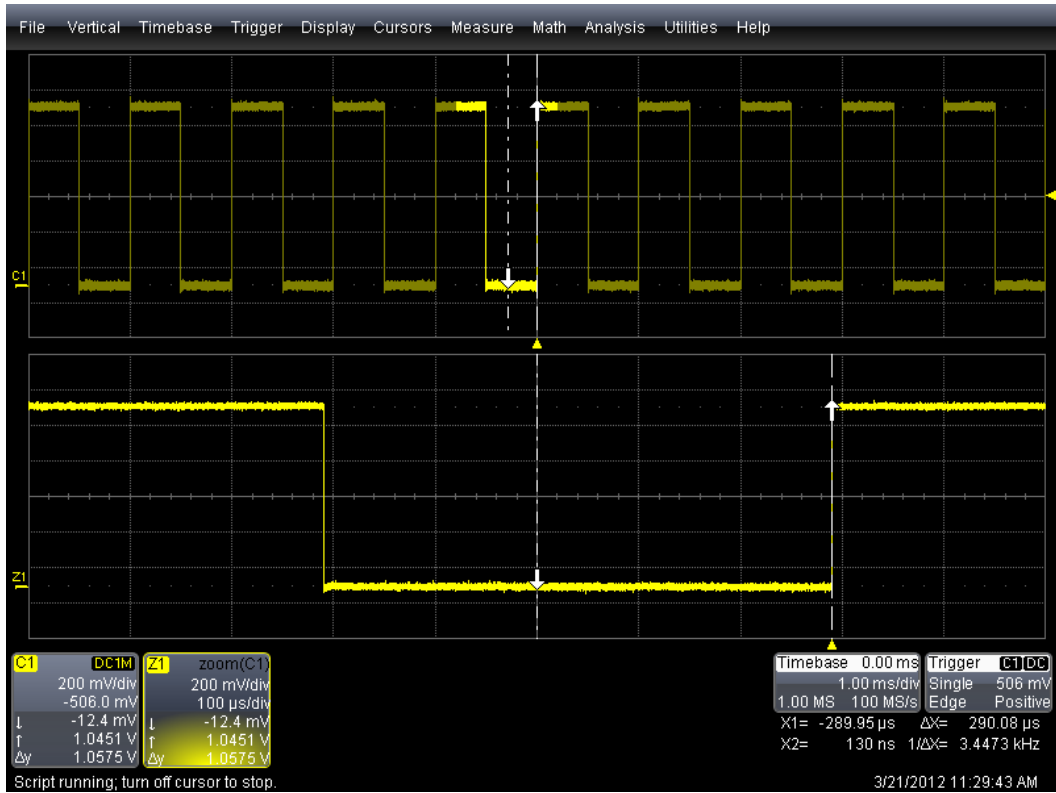


**Figure 3:** The XStream Browser highlighting the Cursor Type command. Note the body of the command appears at the bottom of the browser display. You can also copy and paste the command into your script from the browser

## Running the Script

The completed script file can be copied to the scope. The D:\Scripts folder is a convenient location. The script can be executed directly from the Windows Explorer by navigating to it and double clicking.

Figure 4 shows the result of running the script, script message in lower left corner and the center of zoom Z1 tracks the cursor location.



**Figure 4:** The result of running the script. The center of the zoom trace Z1 tracks the cursor location

## Additional Script Examples

The LeCroy Website has a group of example scripts at:

<http://www.lecroy.com/support/techlib/programmingexamples.aspx?type=8&capid=106&mid=528&smid=852>

Study these scripts and become familiar with the oscilloscope automation commands. This will make it easy for you to write your own scripts.

This completes the tutorial.