

Making the Most of ActiveDSO

Creating Command Buttons in Microsoft Excel

ActiveDSO is a LeCroy-designed ActiveX control for use with Waverunner, 93xx or LC series LeCroy Oscilloscopes and the LSA series of embedded system digitizers. The Microsoft Excel examples included with ActiveDSO contains command buttons that are used to call Visual Basic for Applications (VBA) subroutines. This Operating note provides a short tutorial on creating command buttons and accessing the associated VBA code from the spreadsheet.

Step 1: Creating the Command Button:

1A. Open the *Control Toolbox* toolbar (menu path: View → Toolbars → Control Toolbox). A toolbar with about 15 buttons should now be either "floating" over the spreadsheet, or anchored at the top, bottom, or side of you Excel application window, depending on where it was last placed. Move your mouse over each of the icons to see a description of each of the controls on the toolbar.

1B. Click on the button with the description *Command Button*. Move the mouse to where you would like to create the button, click and hold the left mouse button, drag the mouse to create a rectangular shape, and then release the mouse to create the button. (See figure 1)

(Note: The button is not a part of the ActiveDSO package, but is part of the standard set of user-interface controls included within the Microsoft Office suite of applications.)

Step 2: Changing the buttons properties:

At this point, you are in the *Design Mode*. The Design Mode button on the Control Toolbox toolbar has become depressed, and another small toolbar called *Exit Design Mode* has appeared.

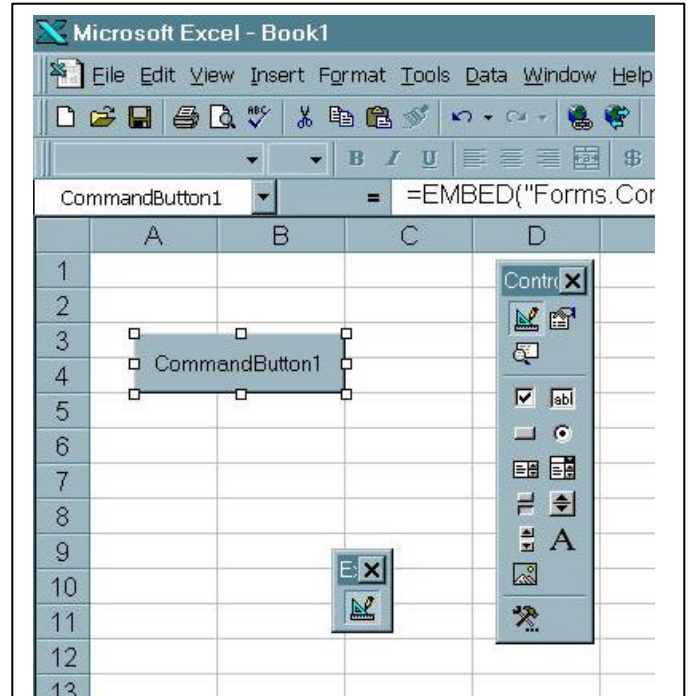


Figure 1 – The Control Toolbox is open, and a new Command Button has been created.

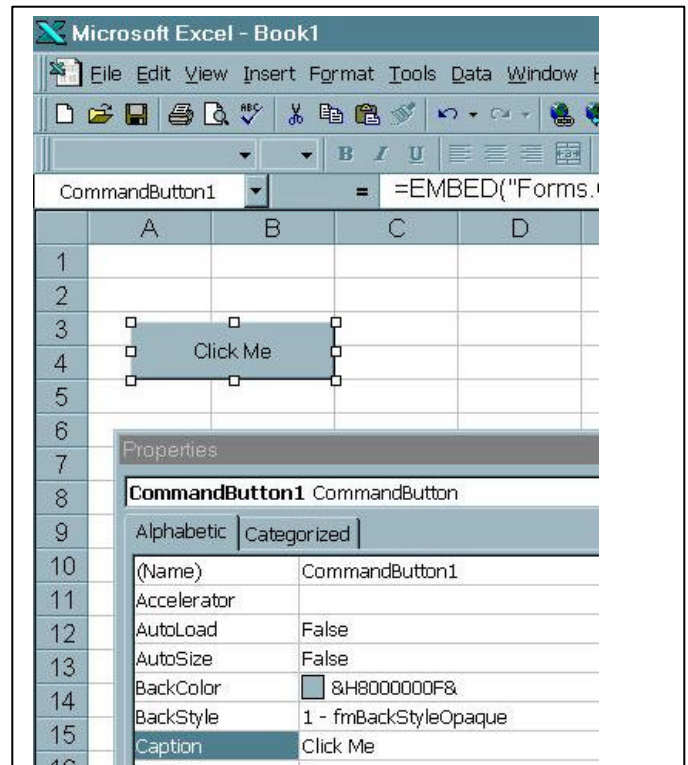


Figure 2 – The Properties window (cropped) is visible, and the button’s Caption property has been changed.

2A. While in Design Mode, controls on the spreadsheet can be moved, resized, deleted, etc. Right-click over the button you just created, and choose **Properties**. The Properties window is now visible, with the properties for the **CommandButton1** object displayed. Excel automatically names command buttons in this manner. (See figure 2)

2B. Change the text displayed on the button by changing the value of the **Caption** property from CommandButton1 to read "Click Me". Also change the **Name** property to read "MyFirstButton."

Step 3: Associate a subroutine with the button:

3A. Right-click on the button again, (now re-Named and re-Captioned). Choose **View Code** to open the **Microsoft Visual Basic** window. A subroutine called **MyFirstButton_Click** has been created. This subroutine will be executed when the button is clicked.

3B. Insert the text `MsgBox "Button Clicked!!!"` before the `End Sub` statement. (See figure 3) Switch back to the Excel window by using Alt-Tab.

3C. Exit Design Mode by clicking the Design Mode button on the Control Toolbox toolbar, and close the Properties window. Now click the "Click Me" button to execute the code. You should see a small dialog box with the text "Button Clicked!!!". At this point, you have successfully created a button and linked the Click event to a subroutine. (Figure 4)

A Few Notes:

1. The Visual Basic window does not need to be open in order for the code associated with the "Click Me" button to be enabled.
2. Subroutines can be executed from the Visual Basic window; this is useful when debugging code. (Note: it is necessary to exit design mode to run your VBA code.)
3. See the on-line help accessible from the Visual Basic window help menu to obtain

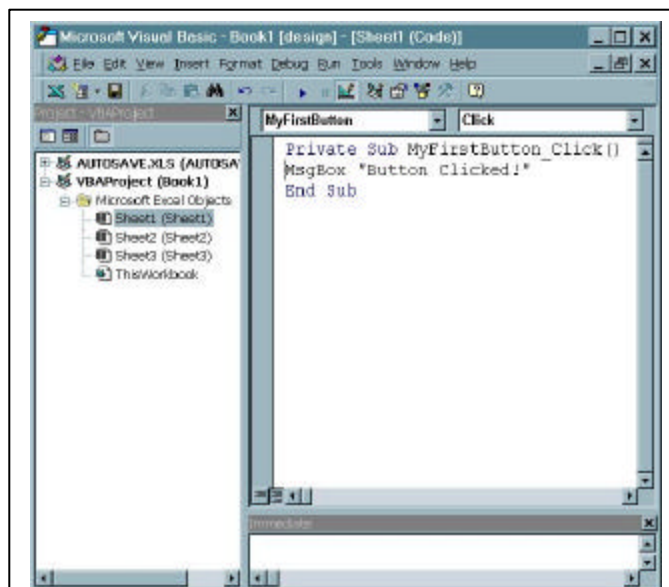


Figure 3 – The Visual Basic window. The Project, Code and Immediate windows are displayed in the default arrangement.

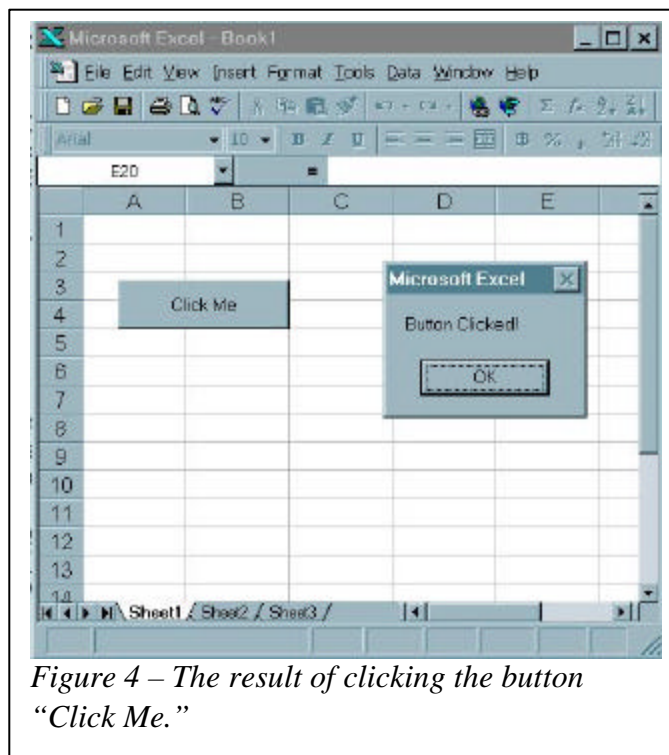


Figure 4 – The result of clicking the button "Click Me."

4. more information about writing and debugging VBA code.
4. For an introduction to ActiveDSO, please see LeCroy Application Brief L.A.B. 808.
5. It is useful to display the Visual Basic toolbar; it contains a button for switching to the Visual Basic window.

Filename: LAB809.doc
Directory: \\Apini\work\Lab809
Template: C:\Documents and Settings\kpazmino\Application
Data\Microsoft\Templates\Normal.dot
Title: Making the Most of ActiveDSO
Subject:
Author: LeCroy Corporation
Keywords:
Comments:
Creation Date: 4/28/1999 11:52 AM
Change Number: 3
Last Saved On: 5/4/1999 1:17 PM
Last Saved By: LeCroy Corporation
Total Editing Time: 5 Minutes
Last Printed On: 8/22/2001 2:59 PM
As of Last Complete Printing
Number of Pages: 2
Number of Words: 518 (approx.)
Number of Characters: 2,957 (approx.)