LB3264 - LadyBug 32 / 64 Bit Bridge
Excel Spreadsheet with VBA Example

This document is used with LB3264AccessV3.xls and LadyBug's LB3264 Bridge software driver package. The spreadsheet and VBA programs are designed to decrease the time required to become familiar with LadyBug programmatic sensor control when using LadyBug's sensors from 32 bit software on 64 bit operating systems.

Background:

The recent migration to 64 bit operating systems has created IO issues for users running 32 bit applications in 64 bit operating systems. Windows 64 systems will not allow 32 bit kernel level dll's to load; as a result WoW64 will not load many dll's for Test and Measurement industry software. LadyBug provides 32 and 64 bit drivers, however as mentioned above LadyBug's 64 bit dll must be used on 64 bit systems and 32 bit applications cannot directly access the dll. LadyBug provides its LB3264 driver package to bridge the operating system bit-sizes.

It is important to note that Windows XP, 7 and 8 versions have all been sold in 64 and 32 bit word sizes. This issue relates to the Word Bit Size and not to the Windows version.

This Spreadsheet package requires a computer running 64 bit Windows, an installed 32 bit version of excel and an installed LadyBug sensor. The sensor operation should be verified by running LadyBug's power meter application.

To get the system operating quickly, install the entire package into a newly created directory, "C:\3264 Excel". The VBA program references this directory, the pointers may be moved, however it is recommended that the example be tested and fully functional first.

How it works:

In order to convert the data word-size, LadyBug has created a TCP/IP Socket Server. The server receives data from 32 bit applications via LadyBug's 32 bit special dll (LB_API2_32_64.dll). The server passes the newly converted 64 bit wide data stream to LadyBugs's 64 bit dll (LB_API2.dll) which passes it on to the sensor. Data flows both ways of course. The Socket Server is an executable that must be launched prior to functioning. The Socket Server is launched and controlled by LB_API2_32_64.dll using the StartSocket and StopSocket instructions. Visibility and certain other items are controlled by data in the file lb_api_32_64.ini.

Included Files:

lb_api_32_64.ini	Control file for the Socket Server
LB_SocketServer.exe	Socket Server
LB_API2.dll	LadyBug's 64 Bit dll
LB_API2_32_64.dll	LadyBug's 32 bit dll that accesses the Socket Server

Recommended additional Documentation:

LadyBug *Power Sensor Programming Manual* from LadyBug's site or the Power Meter Directory
LB-32-64 Solution.pdf from the Socket Server download on LadyBug's web site

The Example

The example was written for use in Excel 2007. To protect computers from viruses, Microsoft has disabled VBA functionality by default in newer versions of Excel. This is controlled by macro security, the trust center or other locations on various versions of Excel. Security **MUST** be disabled prior to operating this example successfully.

Important note: Excel will NOT allow dll access if it (Excel) was started by launching the xls or xlsm file from explorer or from its recent files list in the dropdown start menu. Excel should be started with administrator rights and LB3264AccessV3.xls MUST be opened from the file open menu in Excel for a successful launch of the Socket Server.

There are two forms and two modules with the example. The first form, *Sensor Control* will launch when the sheet is loaded. The Socket Server can be seen launching at the same time. If the Socket Server is not seen loading, then there is an error of some sort. The error likely relates to Excel rights, security within Excel or system use of Sockets & TCP/IP. The program will stop at the Socket Start line. Note the x=SocketStart() instruction that starts the Socket. Below is the form initialize code.

```
Private Sub UserForm_Initialize()
ActiveWorkbook.Sheets(1).Activate 'Make sure the worksheet is active and displayed
x = StartSocket()
If x <> 1 Then Stop 'Error here if socket is not started
GetSensor
ActiveWorkbook.Sheets(1).Activate
'SensorCon.Show
End Sub
```

When the program closes, the socket MUST be stopped. This will not happen if the program is stopped by a means other than the exit button on the form or the red x on the form. The socket has been left visible so that it can be seen and closed manually if necessary.

The *SensorFunctions* module contains all of the dll calls. A few examples are shown below. Note the address to the dll.

```
Public Declare Function LB_SensorCnt Lib "c:/3264 Excel/LB_API2_32_64.dll" () As Long
Public Declare Function StartSocket Lib "c:/3264 Excel/LB_API2_32_64.dll" () As Long
Public Declare Function StopSocket Lib "c:/3264 Excel/LB_API2_32_64.dll" () As Long
```

The form *Power Measure Timer Control* is used to set time based power measurements. These can be made and the data placed in the spreadsheet. The system timer calls are in the *TimeController* module.

This VBA program and spreadsheet have been provided for example purposes and have not been fully error checked.

Further information is available at www.LadyBug-tech.com or contact LadyBug at 707-645-1050