

The structure of the installation files is shown below:

Name	Date modified
WIN_32	10/5/2021 10:36 AM
WIN_64	10/5/2021 10:36 AM
WINUSB_32	10/15/2021 10:34 AM
WINUSB_64	10/15/2021 10:34 AM
Explantions of Instalation Files.pdf	10/15/2021 10:46 AM
LB_INST.INI	10/6/2021 11:33 AM
ReadMe.txt	10/5/2021 10:58 AM
Setup.exe	10/12/2021 9:32 AM

Most users will run setup.exe on the root as shown above. When executed, this application interrogates the system to obtain the operating system name (e.g., Win 7 or Win 10) and bitness (32 bit or 64 bit). Once this information is known, it offers to execute one of the setups located in the subdirectories. You will first be offered the most likely or preferred installation (located in one of the four subdirectories) based on this information. In addition, you may be offered custom or alternative installations.

So, the Setup.exe selects and runs one of the four installations (Setup.exe) in one of the four subdirectories based on your operating situation and your preferences.

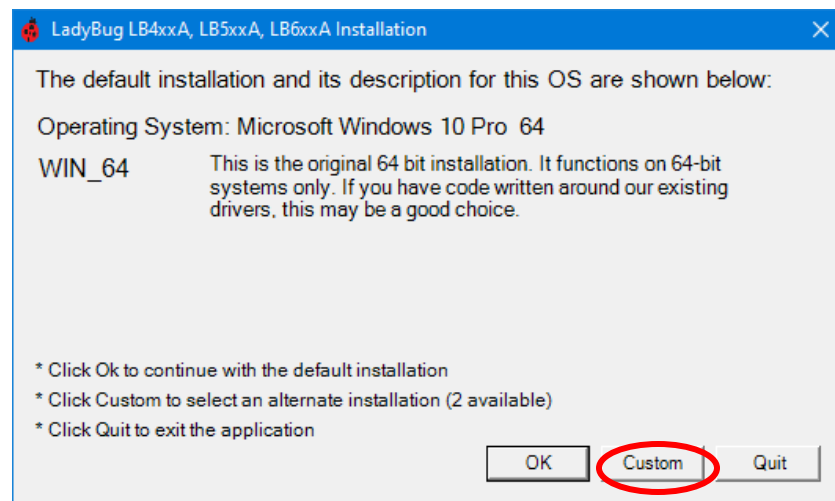
Each installation directories contain a distinct setup file. These files are distinguished by their original issue date, bitness and the instrument driver they employ. A person may have cause to choose one installation over another including an alternate. This document is intended to help in making that decision. Note to test system developers, the drivers retain the same LB\_API2 interface.

As noted above, there are four files and four subdirectories. Each subdirectory contains a unique setup.exe. These directories and their setup.exe are described in the table below:

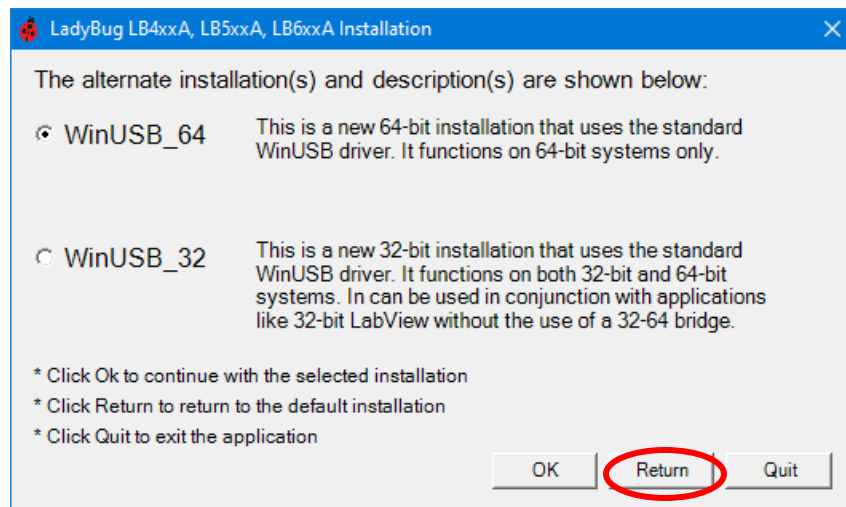
Item	Type	Description
WIN_32	Subdirectory	Contains the original 32-bit installation package. It uses a custom bulk mode USB driver. It runs on Windows XP, Windows 7 and Windows 8.
WIN_64	Subdirectory	Contains the first 64-bit installation package. This uses a customized bulk mode USB driver. It is for 64-bit Windows 7 and Windows 8 and Windows 10.
WINUSB_32	Subdirectory	This installation is being introduced with this package in 2022. It makes use of Microsoft's WinUSB driver. It runs on Microsoft Windows 32-bit and 64-bit systems starting with Windows 7 through Windows 10.
WINUSB_64	Subdirectory	This installation is being introduced with this package in 2022. It makes use of Microsoft's WinUSB driver. It runs on Microsoft Windows 64-bit systems starting with Windows 7 through Windows 10.

Some may need help answering the question for some will be, "which installation do I choose?" We've included descriptions of the various setups and an additional brief ReadMe.txt file to help make this choice. If this is insufficient, please don't hesitate to call our support line.

In addition, we've included an explanation of each choice directly in the Setup.exe GUI. The screen below shows Setup.exe offering a WIN\_64 installation. Note the description in this example. It also has a "Custom" button. The "Custom" button allows you to select an alternate or custom installation.



If you click the "Custom" button, you'll see a screen similar to the one below. Your screen may differ depending on your operating system. This is how Setup offers alternate or custom installations. The custom offerings also have descriptions to help in your decision. This example was taken during a Windows 10 Pro, 64-bit installation. You can see that, in addition to the WIN\_64 installation above, there are two custom or alternative installations or setups available as shown below.



Finally, clicking the Return button, goes back to the previous screen.

Users may find the table below useful in selecting an appropriate installation. For programmers and system developers, it is important to note that LB\_API2 has retained the same programming interface (API). This was done to ease the transition between systems regardless operating system or bitness.

Select a situation	Considerations	Recommendations
You only use the LadyBug applications that ship with the product (e.g. Pulse Profiling)	Ease of use and getting up and running quickly.	Run the default installation
You've been using LadyBug sensors on your <b>Windows XP, 32 bit or Windows 7, 32 bit</b> systems and you intend to continue to use these systems	Recompiling, rebuilding your test system code is not an option at this time	Install the <b>Win_32</b> setup. This is the installation you've been using all along and you should continue to use it. You can do this through Setup.exe on the root or the Win32 setup.exe in the Win32 directory. Or continue using the setup you've been using.
You've been using LadyBug sensors on your <b>64 bit Windows 7, 8 or 10</b> systems and intend to continue to use these systems	Rebuilding your systems is not an option at this time	Install the <b>Win_64</b> setup. This is the same installation you've been using. You can do this through our Setup.exe on the root or go directly to the Win64 setup.exe in the Win64 subdirectory.
You've been using LadyBug sensors on your <b>64 bit Windows 7, 8 or 10</b> systems with <b>LabView 32 bit</b> along with a <b>32-64 bit bridge</b> .	<b>You want to get rid of the 32-64 bridge</b> and you're willing to invest a modest amount of effort to do so.	We recommend switching to the <b>WinUSB_32</b> package. <b><u>The LB_API2 calls in your code, in the bridge and the WinUSB LB_API2.dll are all the same.</u></b> With a modest effort you can switch to the WinUSB_32 installation and eliminate the 32-64 bridge. It may also help performance.
You have <b>older 32-bit systems (XP)</b> and would like to upgrade to <b>Windows 10 32-bit</b> for your test systems.	You are willing to do a modest amount of work to do this.	We recommend the <b>WinUSB_32</b> installation. <b><u>The WinUSB 32 LB_API2 interface is identical.</u></b> The WinUSB 32-bit drivers work on all Windows 32-bit and 64-bit systems except for Windows XP, 32 bit.
You are developing on a <b>mix of new 32-bit and 64-bit systems</b> and using <b>NI LabView 32-bit</b> for at least for part of this effort.	Look at our sample code, test harnesses and programming guides for assistance	In this case we recommend the <b>WinUSB_32</b> installation. It allows you to communicate with our sensors using the WinUSB driver directly. <b>However, if you intend to make calls to LB_API2 from your 64 bit applications</b> you may need to also use the 64 bit version of the driver.
You are developing on <b>new 64-bit systems (e.g. Win 10 64-bit)</b> using <b>NI LabView 32-bit</b>	Look at our sample code, test harnesses and programming guides for assistance	In this case we recommend the <b>WinUSB_32</b> installation. It allows you to communicate with our sensors on 64-bit system using the WinUSB driver and WoW64.
You're developing <b>new systems</b> and you're using all <b>64 bit</b> programming components and you <b>have no need for 32-bit</b> components in your future.	Look at our sample code, test harnesses and programming guides for assistance	In this case we recommend the <b>WinUSB_64</b> installation. It allows you to communicate with our sensors using the WinUSB driver and WoW64.

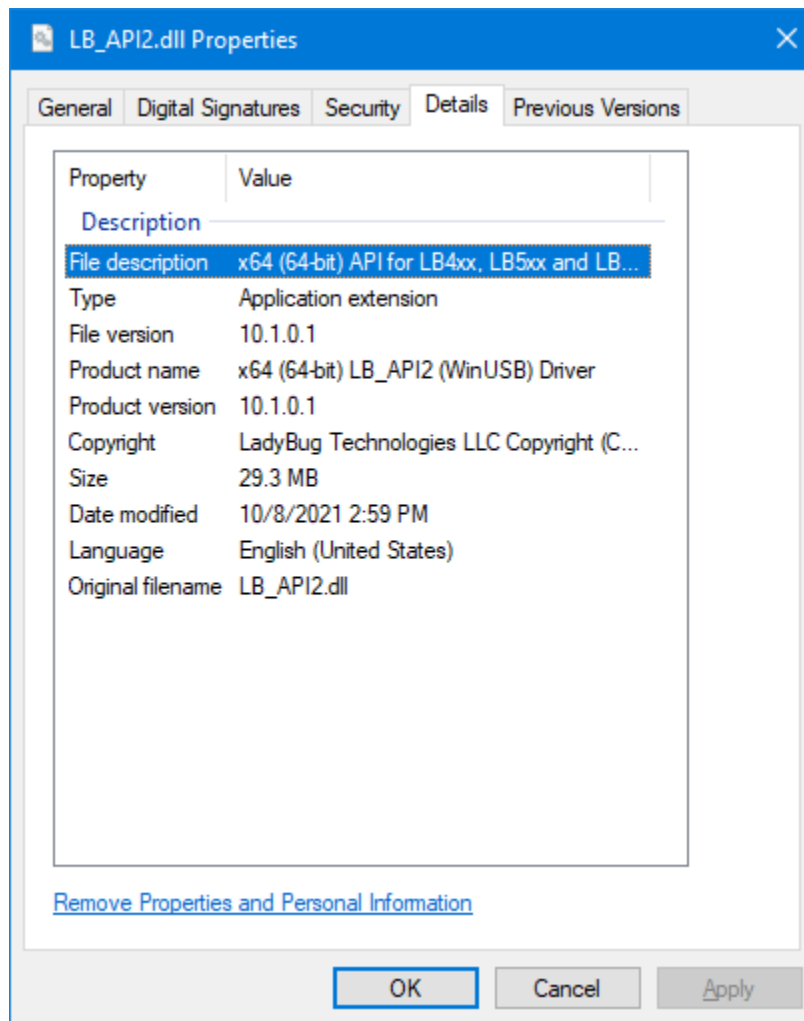
There are a few remaining points.

#### Additional Notes:

1. The new installations, WinUS\_32 and WinUSB\_64, have sample code for the following languages:
  - a. C++ (Visual Studio 2017)
  - b. C# (Visual Studio 2017)
  - c. Python (note, the version considerations)
  - d. LabView
  - e. VB.Net (Visual Studio 2017)
2. The WinUSB\_32 and WinUSB\_64 setups provide an independent means of installing the driver (without installing all the applications). These files are contained in the installation. So, you will have to do a complete installation on one machine to get access to these driver installation files. Using these files to install the WinUSB\_32 and WinUSB\_64 drivers is straight forward.
3. One constraint of the WinUSB\_32 and WinUSB\_64 drivers that is managed by the operating system so only one process (exe) can access our sensors at any one time. It is important to note that we've constructed the new WinUSB 32-bit and 64-bit power meter and pulse profiling applications to support multiple sensors. To be specific:
  - a. One application may access many (up to twelve) LB4xx/LB5xx/LB6xx devices
  - b. If one application is running and accessing any LB4xx/LB5xx/LB6xx device, any additional application that attempts to access these same devices will be disallowed access by the operating system.
4. The Setup.EXE is on the root is provided as a convenience. It is not necessary to use this although we highly recommend it. You may copy and use any one of the four setup packages independently for use with LadyBug devices.

To use these setups independently, copy the files from the appropriate subdirectory to a directory of your choosing. And then run the copied Setup directly. For instance, to run the original 32-bit installation, copy the files in WIN\_32. This should include an \*.msi and setup.exe. Then run setup.exe. The same is true for other options. There will be one or two files depending on the installation you select.
5. For those using Win 10, 32-bit, you will only be offered the WinUSB\_32 installation. This installation has the added benefit of being able to run on both 32- and 64-bit systems. Specifically, this setup can be used with 32-bit LabView (or any 32-bit application) on a Windows, 64-bit operating system.
6. For those using LabView 32-bit, we recommend the WinUSB\_32 for future development. This is because it runs on both the 32-bit and 64-bit systems (except Windows XP). The WinUSB driver is supported through the WoW64 system. A Windows XP WinUSB upgrade or extension appears to be available currently. But we did not test it nor are we sure how long this will be available.
7. For those that have used the 32-64 bridge in past development, it may be worth considering moving to WinUSB\_32. The LB\_API2 dll interface is the same as before so the transition should be relatively easy (just pointing to a new dll). You should see improved performance and stability.

8. The LB\_API2.dll on the WinUSB\_32 and WinUSB\_64 dll must be accompanied by a lower level WinUSB dll of the same bitness. In other words, they must be copied and used as a pair. For ease of identification these files, the WinUSB\_32 and WinUSB\_64 LB\_API2.dll drivers and lower file named LBUSBDRVD.dll are tagged (with bitness and driver) because the files names are identical. To see this information, open the windows File Explorer, right click on the file (LB\_API2.dll or LBUSBDRVD.dll) and select properties. Then select the “Details” tab. You should see something like the screen below. Note that the first line (File Description) starts off as “x64 (64-bit) API...”. And the product name is “x64 (64-bit) LB\_API2 (WinUSB) Driver.” Similar details accompany the 32 bit version of the WinUSB\_32 bit driver files. These details allow you to independently determine the type and bitness of these drivers.



9. As stated throughout this document, every effort has been made to retain the original programming interface and the LB\_API2 file name. So, moving between platforms and changing bitness should cause minimal disruption. The most likely change that you will need to address is the change in pointer size (32-bit vs. 64-bit) between the various operating systems. This difference is understandably unavoidable, but your programming language of choice may make this a non-issue.