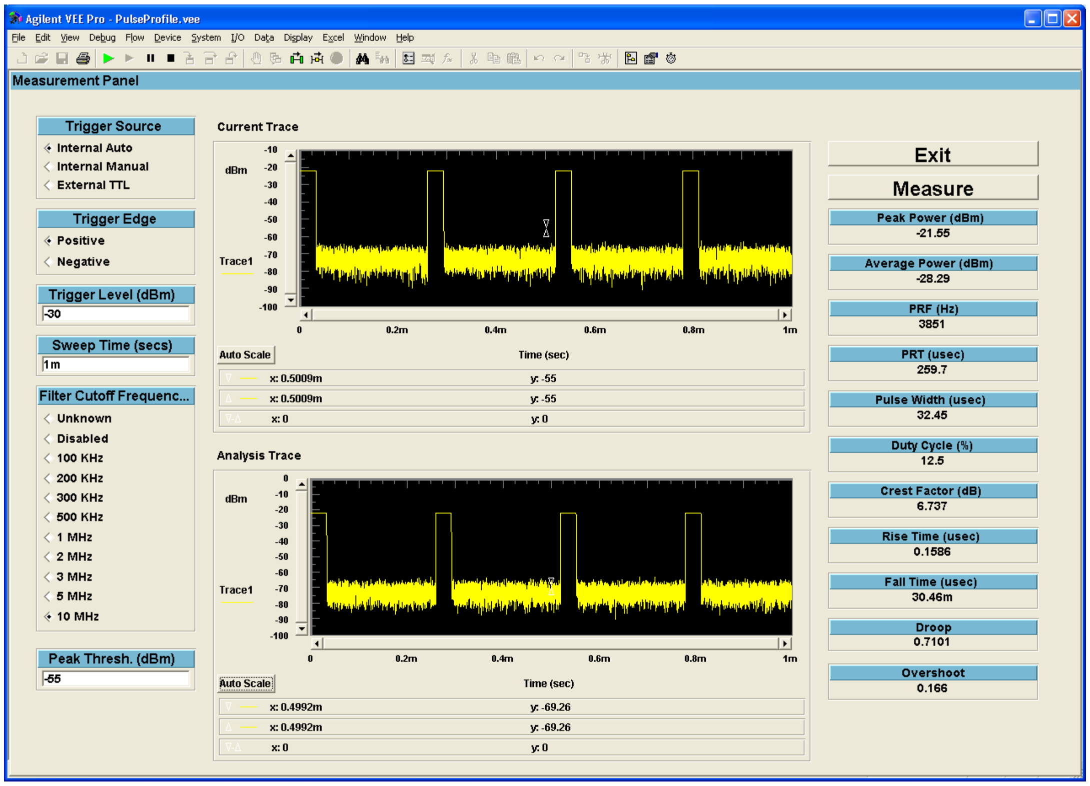
Ladybug VEE Pulse Profiling Sample Code

Overview

This document introduces the user to using Ladybug power sensors for pulse profiling in a VEE environment. It assumes familiarity with VEE but does not require expertise. It demonstrates key functions including:

- Sensor Initialization
- Sensor Setup
- Pulse Profile Measurements
- Pulse Profile Calculations

An example test program (with source code) is included. The user interface is shown in figure 1. It will work with model 480 sensors only. All VEE code is written in VEE version 7.5.



Agilent VEE Pro - PulseProfile.vee
File Edit View Debug Flow Device System I/O Data Display Excel Window Help
Measurement Panel
Current Trace
Analysis Trace
dBm
Trace1
Time (sec)
Auto Scale
Exit
Measure

Trigger Source: [✓ Internal Auto] Internal Manual External TTL

Trigger Edge: [✓ Positive] Negative

Trigger Level (dBm): -30

Sweep Time (secs): 1m

Filter Cutoff Frequenc...: Unknown Disabled 100 KHz 200 KHz 300 KHz 500 KHz 1 MHz 2 MHz 3 MHz 5 MHz [✓ 10 MHz]

Peak Thresh. (dBm): -55

Current Trace markers: x: 0.5009m y: -55; x: 0.5009m y: -55; x: 0 y: 0

Analysis Trace markers: x: 0.4992m y: -69.26; x: 0.4992m y: -69.26; x: 0 y: 0

Peak Power (dBm)	-21.55
Average Power (dBm)	-28.29
PRF (Hz)	3851
PRT (usec)	259.7
Pulse Width (usec)	32.45
Duty Cycle (%)	12.5
Crest Factor (dB)	6.737
Rise Time (usec)	0.1586
Fall Time (usec)	30.46m
Droop	0.7101
Overshoot	0.166

Figure 1: Run-time view of the Measurement Panel. Note that both the current trace and analysis trace are shown.

Exercising the Pulse Profiling Code

The pulse profiling code is an interactive application designed as an introduction to interfacing the Ladybug DLL with VEE. To start it, simply click on the 'Play' button or (if in Detail view), click on the 'Start' button of the main program.

The software has many capabilities and is easy to use. A few of the settings and techniques are shown here to assist you in getting started.

The overall code is shown in the following figure.

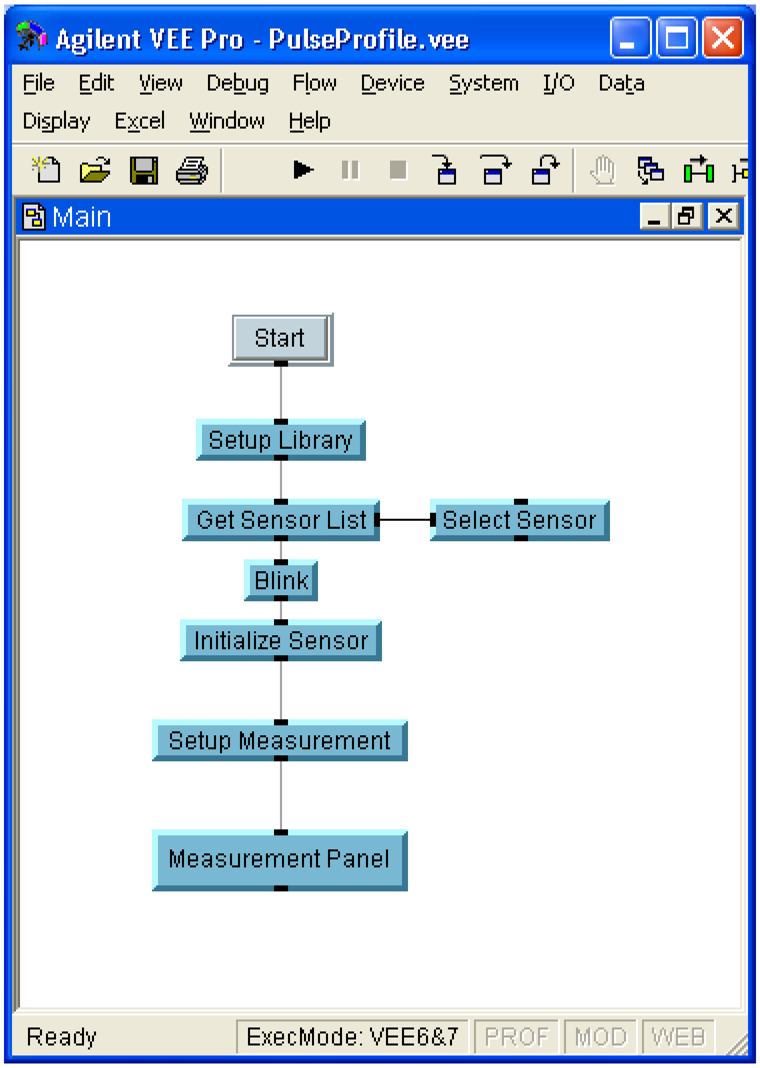


Figure 2: Detail view of the pulse profiling code. Note that Setup Library must be executed first.

Software Installation

To install the software, simply unzip the file to an appropriate folder on your hard drive. It will be ready to run immediately. To add to existing software, please refer to the section on adding to existing VEE files.

Setup Library

The first step in the application is to setup the Ladybug DLL library. This is shown in the figure below.

First, any existing library is deleted from memory. This ensures that the most current version of the library will be loaded and avoids certain problems.

Next, the software forms the DLL and declarations file. These are both done with similar formulas that will always work provided that both files are in the same folder as the VEE file. Unless you intend to move each of the files to different locations, it would be best to leave this part of the code alone.

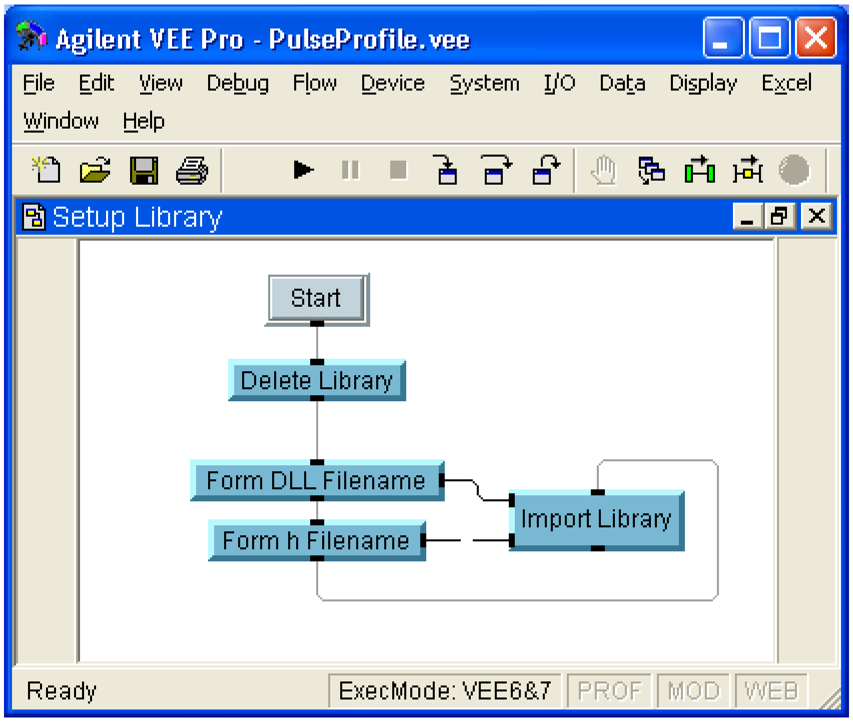


Figure 3: The Setup Library routine will work where ever the code is located due to the formulas that form the DLL and .h filenames.

Selecting the Power Sensor

The next two userobjects (Get Sensor List and Select Sensor) retrieve the list of all available sensors, filter out those that will not work (non-480 sensors), and allows the user to select the sensor.

The only item worth noting here that may be other wise confusing is that 480 type sensors have a ModelNumber of 3. In reality, this is the enumerated value of the model number.

Selecting the power sensor will set the variable Address_A.

Once a power sensor is selected, the LED on the power sensor will blink several times. This is controlled by the Blink userobject.

Initializing the Power Sensor

Prior to use, the power sensor must be initialized. This is performed in the "Initialize Sensor" UserObject, as shown in the following figure.

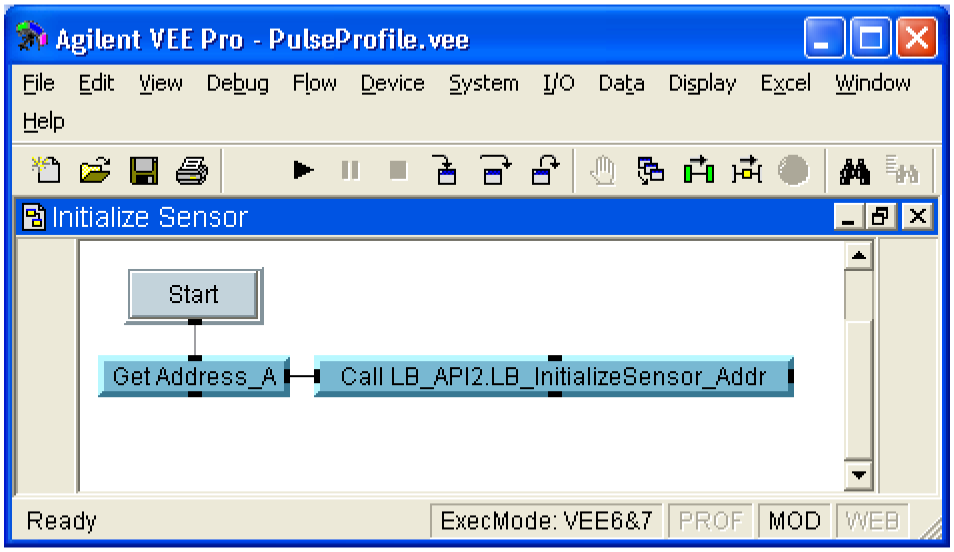


Figure 4: Sensor initialization using the address. Note that this is the typical format of the library calls, although a formula call could also be used.

Setting Up a Pulse Profile Measurement

Before taking measurements, certain default values are set and the relevant settings of the sensor are read. All of this occurs in the Setup Measurement UserObject, shown below.

First, the sweep time is set to 1 millisecond, the trigger source is set to internal continuous (0), and the gate mode is turned on.

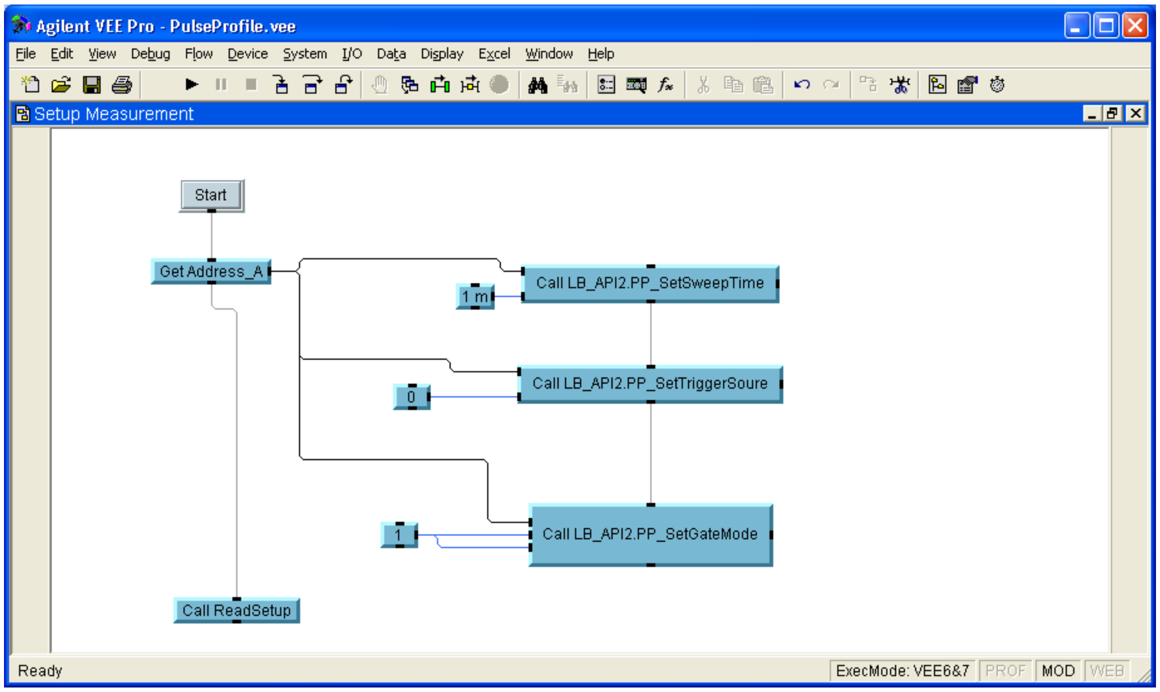

Agilent VEE Pro - PulseProfile.vee
File Edit View Debug Flow Device System I/O Data Display Excel Window Help
Setup Measurement
Start
Get Address_A
Call LB_API2.PP_SetSweepTime
1 m
Call LB_API2.PP_SetTriggerSoure
0
Call LB_API2.PP_SetGateMode
1
Call ReadSetup
Ready
ExecMode: VEE6&7
PROF
MOD
WEB

Figure 5: The Setup Measurement UserObject sets some of the default values of the sensor. Note that the sequence out pin of Get Address_A prevents the settings from being read until all three default values are set.

Next, the ReadSetup UserFunction is called. This UserFunction is used frequently to retrieve the sensor settings. It returns the trace length, the sweep time, the trigger edge, the trigger level, the trigger source, and the filter cutoff frequency. All of these are stored in variables.

The Measurement Panel

The panel view of the measurement panel UserObject was shown in figure 1. The detail view is shown below.

Essentially, there are two loops. The outer loop waits for any events to occur (such as a button being pressed). The inner loop retrieves the latest trace and displays it on the upper graph (Current Trace).

For example, if the user changes the trigger edge enum, the new trigger edge will be set and the loop will start again. Note that, at the beginning of the next iteration, the loop will read back the trigger edge and update the front panel.

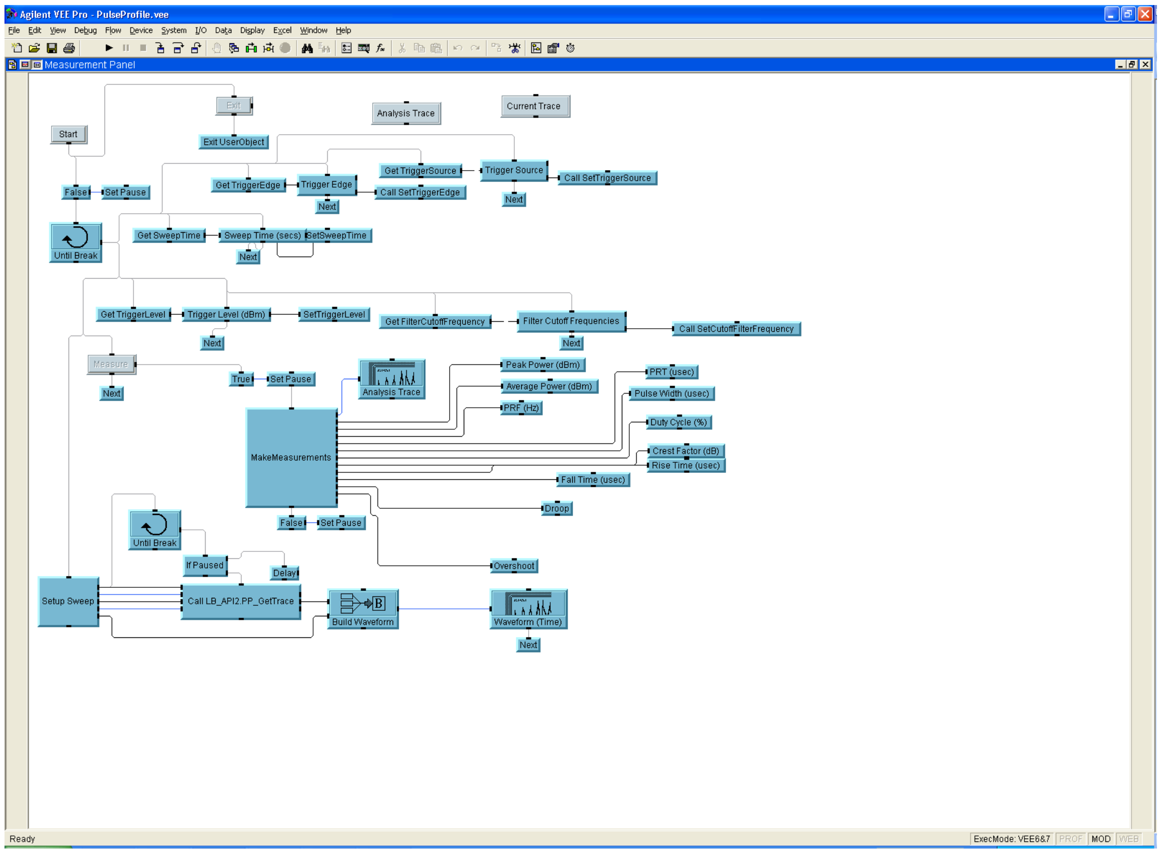


Figure 6: The detail view of the Measurement Panel demonstrates how measurements are made. Note that clicking on any control will start a new measurement loop execution.

Changing Settings

All of the settings are on the left side of the panel. Simply change the value or selection as desired and the power sensor will be updated immediately. Please note that some changes may have no effect – for example, changing the trigger level will have no effect if the trigger source is set to “Internal Auto”.

After each change, all of the sensor values are read (using the ReadSetup UserFunction) and the values on the front panel are updated.

Measuring Quantities

The most complex part of the Measurement Panel UserObject is the “Measure” execution. All of these calculations are instrument measurements are performed in the Measurements UserObject. The most important points concerning these measurements are:

1) The current trace is transferred to the analysis trace and all calculations are performed on the analysis trace.
2) Many of these calculations require two peaks. If there are less than two peaks, a warning is displayed and no measurements are made.
3) There are other warnings as well if the pulse size is too large or small relative to the sweep time.
4) All of these measurements are gated measurements. Gate 1 is always set prior to the measurements being made. There are extensive comments in the code concerning how the gate is set for each measurement.

Also, please note that the acquisition of new traces is paused while the calculations are made.

Using and Modifying the Code

There are approximately 250 functions available in the DLL. Each of these functions is documented in more detail in the LadyBug User’s Manual. Most are extremely straightforward and easy to use.

After the DLL is imported, they can be found in the Program Explorer under Compiled Functions → LB_API2. They can also be found in the function and object browser under Compile Functions in the LB_API2 category.

Sensor Communication

Communication with LadyBug power sensors is done through the index, the serial number, or the address of the sensor. In this example code, all communication is done through the address. This is because the address covers more functionality than the others and because it makes the code more flexible. For example, the user cannot change the serial number of a sensor but they can change the address.

Adding to Existing VEE Files

The UserObjects distributed with this code can be added directly to any existing VEE code. Be sure to import the DLL in the software before calling any of the functions. With a minimal amount of experience, you would probably prefer writing your own UserObjects and UserFunctions directly with the function calls.