



**Phidgets 1123 - Product Manual**

**For Board Revision 0**

**© Phidgets Inc. 2009**

# Contents

## **4 Product Features**

- 4 Connections
- 4 Type of Measurement

## **5 Getting Started**

- 5 Checking the Contents
- 5 Connecting all the pieces
- 5 Testing Using Windows 2000/XP/Vista
- 7 Testing Using Mac OS X

## **7 Programming a Phidget**

- 7 Code Samples
- 7 Coding for your Sensor

## **8 Technical Information**

- 8 Sensor Calibration
- 8 Measuring Voltage
- 8 Formulas
- 8 Other Interfacing Alternatives
- 9 Analog Input Cable Connectors
- 9 Mechanical Drawing
- 10 Device Specifications

## **10 Product History**

## **10 Support**

# Product Features

---

- Measures DC voltages from -30 to +30 Volts with a typical error of  $\pm 100$  mVolts.

## Connections

Designed to connect to a:

- 1018 - PhidgetInterfaceKit 8/8/8
- 1019 - PhidgetInterfaceKit 8/8/8 w/6 Port Hub
- 1070 - PhidgetSBC
- 1202 or 1203 - PhidgetTextLCD

## Type of Measurement

The sensor uses non-ratiometric measurement.

# Getting Started

---

## Checking the Contents

### You should have received:

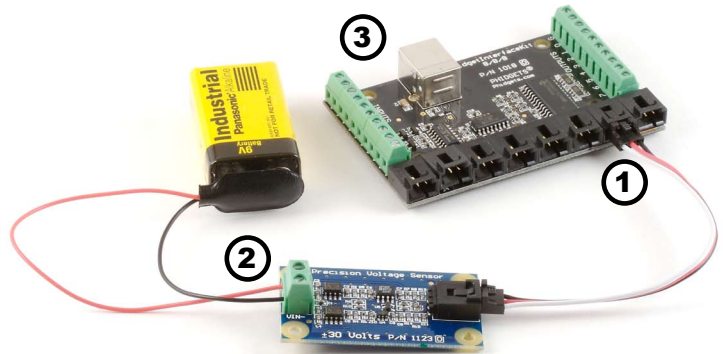
- A Precision Voltage Sensor
- A Sensor Cable

### In order to test your new Phidget you will also need:

- A PhidgetInterfaceKit 8/8/8 or a PhidgetTextLCD
- A USB Cable
- A 9V Battery, and a battery connector

## Connecting all the pieces

1. Connect the Precision Voltage Sensor to the Analog Input 6 on the PhidgetInterfaceKit 8/8/8 board using the sensor cable.
2. Connect the 9V battery to the terminal block.
3. Connect the PhidgetInterfaceKit to your PC using the USB cable.




## Testing Using Windows 2000/XP/Vista

### Downloading the Phidgets drivers


Make sure that you have the current version of the Phidget library installed on your PC. If you don't, do the following:

Go to [www.phidgets.com](http://www.phidgets.com) >> Drivers


Download and run Phidget21 Installer (32-bit, or 64-bit, depending on your PC)

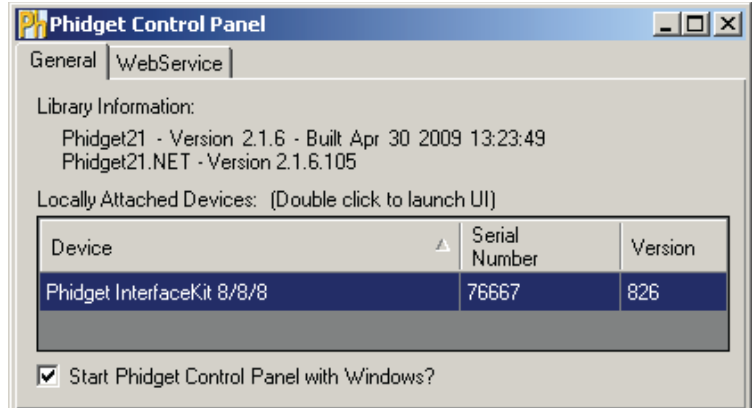
You should see the  icon on the right hand corner of the Task Bar.

### Running Phidgets Sample Program

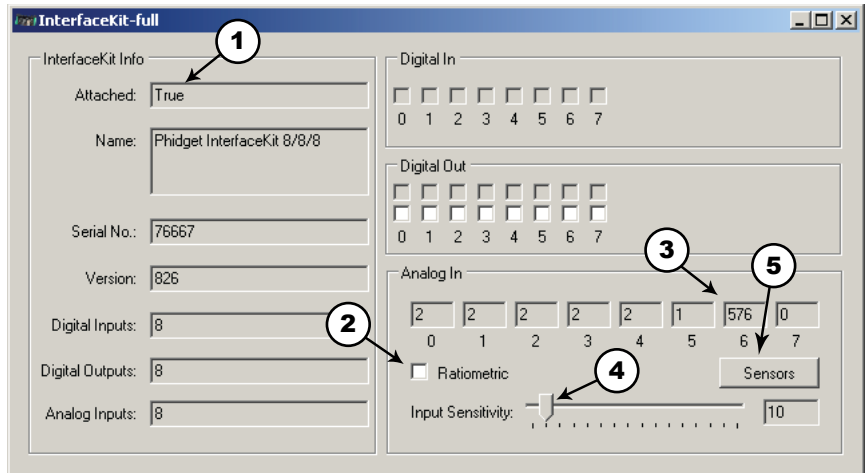
Double clicking on the  icon loads the Phidget Control Panel; we will use this program to make sure that your new Phidget works properly. Since the sensor is connected to a 1018, the computer will see only the 1018. The sensor is providing data through the Analog input it is connected to.

The source code for the InterfaceKit-full sample program can be found under C# by clicking on [www.phidgets.com](http://www.phidgets.com) >> Programming.

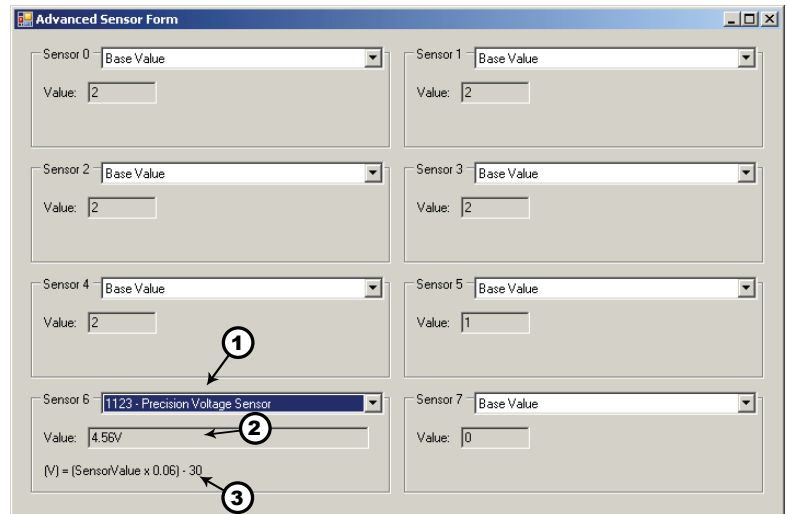
Double Click on the  icon to activate the Phidget Control Panel and make sure that the **Phidget InterfaceKit 8/8/8** is properly attached to your PC.



1. Double Click on **Phidget InterfaceKit 8/8/8** in the Phidget Control Panel to bring up InterfaceKit-full and check that the box labelled Attached contains the word True.
2. Make sure that the Ratiometric box is NOT Ticked.
3. The voltage of your power source is displayed in the Analog In box. The Formula to translate the value into Voltage is: Voltage (in volts) = (SensorValue x 0.06) - 30. The value of 653 in the example above translates into 9.18 Volts.
4. You can adjust the input sensitivity by moving the slider pointer.
5. Click on the Sensors button to bring up the Advanced Sensor Form.



1. In the Advanced Sensor Form, select the 1123 - Precision Voltage Sensor from the drop down menu.
2. The voltage measured across the sensor is shown here.
3. Formula used to convert the analog input SensorValue into Voltage.



## Testing Using Mac OS X

- Click on System Preferences >> Phidgets (under Other) to activate the Preference Pane
- Make sure that the Phidget InterfaceKit 8/8/8 is properly attached.
- Double Click on Phidget InterfaceKit 8/8/8 in the Phidget Preference Pane to bring up the InterfaceKit-Full example. This example will function in a similar way as the Windows version, but note that it does not include an Advanced Sensor Display.

## Programming a Phidget

---

Phidgets' philosophy is that you do not have to be an electrical engineer in order to do projects that use devices like sensors, motors, motor controllers, and interface boards. All you need to know is how to program. We have developed a complete set of Application Programming Interfaces (API) that are supported for Windows, Mac OS X, and Linux. When it comes to languages, we support VB6, VB.NET, C#.NET, C, C++, Flash 9, Flex, Java, LabVIEW, Python, Max/MSP, and Cocoa.

### Code Samples

We have written sample programs to illustrate how the APIs are used.

Due to the large number of languages and devices we support, we cannot provide examples in every language for every Phidget. Some of the examples are very minimal, and other examples will have a full-featured GUI allowing all the functionality of the device to be explored. Most developers start by modifying existing examples until they have an understanding of the architecture.

Go to [www.phidgets.com](http://www.phidgets.com) >> Programming to see if there are code samples written for your device. Find the language you want to use and click on the magnifying glass besides "Code Sample". You will get a list of all the devices for which we wrote code samples in that language.

If this is your first time writing a program to control a Phidget, you should read the Getting Started Guide for the language you are planning to use.

### Coding for your Sensor

Phidget analog sensors do not have their own API, but instead their output is a voltage that is converted to a digital value and accessed through the SensorValue properties and events on a PhidgetInterfaceKit. It is not possible to programmatically identify which sensor is attached to the Analog Input. Your application will need to apply any formulas from this manual to the SensorValue to translate it into usable data.

See the PhidgetInterfaceKit product manual for an overview of its API and a description of our architecture.

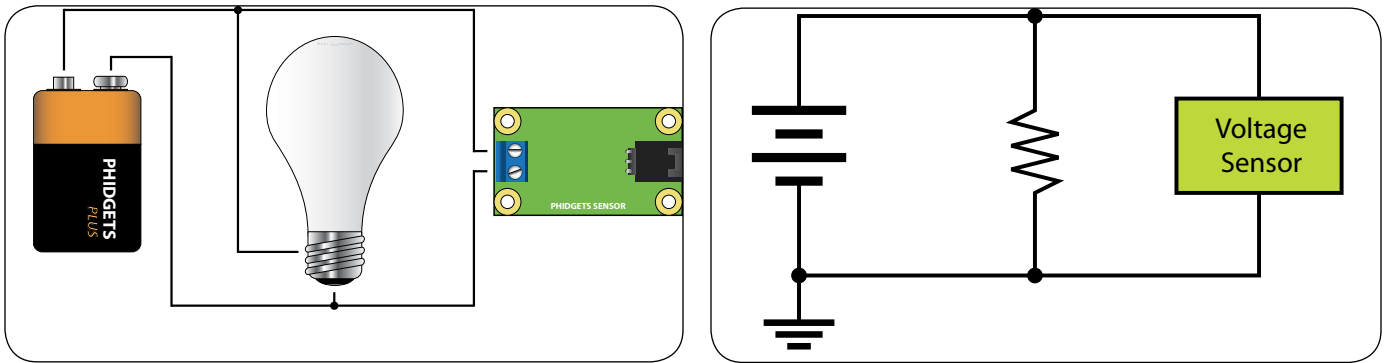
# Technical Information

Unlike a voltage measurement that depends on a fixed point (like ground/zero volts) a differential voltage sensor measures the voltage potential between any two points. One of these two points can be ground, but does not have to be. A differential voltage sensor is more flexible than a fixed-point voltage sensor, and has the ability to measure voltages across components or at other points in a circuit. This is the most common kind of voltage sensor and is the type commonly found in most bench-top and hand-help multimeters.

## Sensor Calibration

For the most accurate measurements it is recommended that the sensor be zeroed and span calibrated in software. Zero calibration should occur when the two inputs are either shorted together or are measuring the same physical point in a circuit. Span calibration should be done with a known voltage as close to 30VDC as possible.

## Measuring Voltage



The Precision Voltage Sensor should be wired in parallel to the circuit under test, as shown. In the diagrams above, the voltage source is represented by the battery symbol. The load is represented by a light bulb or schematic resistor symbol. The voltage sensor measures the differential voltage across the load.

## Formulas

The Formula to translate SensorValue into Voltage is:

$$\text{Voltage (in volts)} = (\text{SensorValue} \times 0.06) - 30$$

## Other Interfacing Alternatives

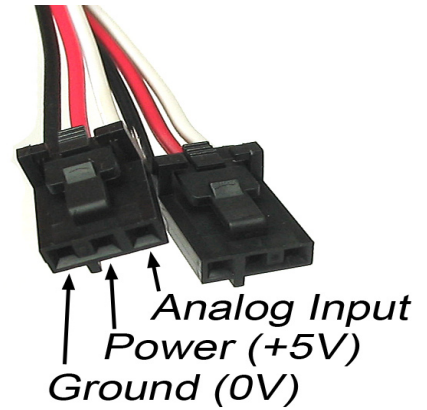
If you want maximum accuracy, you can use the RawSensorValue property from the PhidgetInterfaceKit. To adjust a formula, substitute (SensorValue) with (RawSensorValue / 4.095)

If the sensor is being interfaced to your own Analog to Digital Converter and not a Phidget device, our formulas can be modified by replacing (SensorValue) with (Vin \* 200). It is important to consider the voltage reference and input voltage range of your ADC for full accuracy and range.



# Analog Input Cable Connectors

Each Analog Input uses a 3-pin, 0.100 inch pitch locking connector. Pictured here is a plug with the connections labeled. The connectors are commonly available - refer to the Table below for manufacturer part numbers.

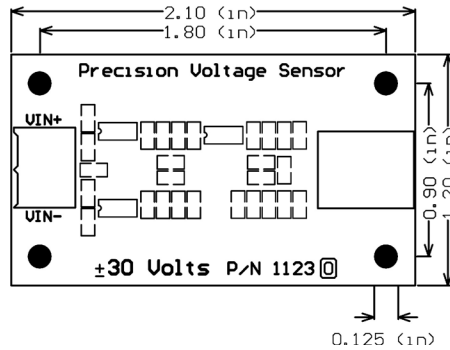


Cable Connectors		
Manufacturer	Part Number	Description
Molex	50-57-9403	3 Position Cable Connector
Molex	16-02-0102	Wire Crimp Insert for Cable Connector
Molex	70543-0002	3 Position Vertical PCB Connector
Molex	70553-0002	3 Position Right-Angle PCB Connector (Gold)
Molex	70553-0037	3 Position Right-Angle PCB Connector (Tin)
Molex	15-91-2035	3 Position Right-Angle PCB Connector - Surface Mount

Note: Most of the above components can be bought at [www.digikey.com](http://www.digikey.com)

## Mechanical Drawing

1:1 scale



**Note:** When printing the mechanical drawing, "Page Scaling" in the Print panel must be set to "None" to avoid re-sizing the image.

## Device Specifications

Characteristic	Value
Input Impedance	100K ohms
Absolute Maximum Positive Input Voltage (V+ = 5V)	+33 VDC
Absolute Maximum Negative Input Voltage (V+ = 5V)	-33 VDC
Maximum Differential Measurement	±30 VDC
Supply Voltage	4.75 to 5.25 VDC
Typical Error	±100mV
Maximum Error over ±30VDC	±500mV

## Product History

Date	Board Revision	Comment
December 2007	0	Product Release

## Support

Call the support desk at 1.403.282.7335 8:00 AM to 5:00 PM Mountain Time (US & Canada) - GMT-07:00

or

E-mail us at: [support@phidgets.com](mailto:support@phidgets.com)