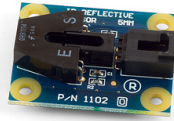


1102 User Guide



Go to this device's product page ^[1]

Getting Started

Checking the Contents

You should have received:

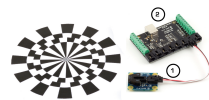
- An IR Reflective Sensor 5mm
- A Sensor Cable

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

- A PhidgetInterfaceKit 8/8/8
- A USB Cable
- A reflective surface of some kind (just about anything will do).

Connecting the Pieces

1. Connect the IR Reflective Sensor 5mm to the Analog Input 6 on the PhidgetInterfaceKit 8/8/8 board using the sensor cable.
2. Connect the PhidgetInterfaceKit to your PC using the USB cable.



Testing Using Windows 2000 / XP / Vista / 7

Make sure you have the current version of the Phidget library installed on your PC. If you don't, follow these steps:

1. Go to the Quick Downloads section on the Windows page
2. Download and run the Phidget21 Installer (32-bit, or 64-bit, depending on your system)
3. You should see the **Ph** icon on the right hand corner of the Task Bar.

Running Phidgets Sample Program

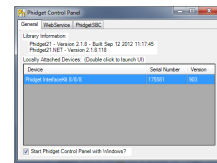
Double clicking on the **Ph** icon loads the Phidget Control Panel; we will use this program to ensure that your new Phidget works properly.

The source code for the **InterfaceKit-full** sample program can be found in the quick downloads section on the C# Language Page. If you'd like to see examples in other languages, you can visit our Languages page.

Updating Device Firmware

If an entry in this list is red, it means the firmware for that device is out of date. Double click on the entry to be given the option of updating the firmware. If you choose not to update the firmware, you can still run the example for that device after refusing.

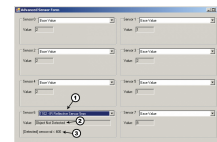
Double Click on the  icon to activate the Phidget Control Panel and make sure that the **Phidget InterfaceKit 8/8/8** is properly attached to your PC.



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 Ticked.
3. Place a piece of reflective material close (5mm) to the sensor and check the value in the Analog In box. Any value less than 400 means that the sensor has detected the object. If you want to use a pattern like the one we used in the Connecting the Hardware picture, do a search on the Web for "optical encoder pattern" and specify images.
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 Sensor 6 box, select the 1102 - IR Reflective Sensor 5mm from the drop down menu.
2. The sensor's detection state for an object will be shown here.
3. Formula used to convert the analog input SensorValue into a detection state.



Testing Using Mac OS X

1. Go to the Quick Downloads section on the Mac OS X page
2. Download and run the Phidget OS X Installer
3. Click on System Preferences >> Phidgets (under Other) to activate the Preference Pane
4. Make sure that the Phidget InterfaceKit 8/8/8 is properly attached.
5. Double Click on Phidget InterfaceKit 8/8/8 in the Phidget Preference Pane to bring up the InterfaceKit-full Sample program. This program will function in a similar way as the Windows version.

Using Linux

For a step-by-step guide on getting Phidgets running on Linux, check the Linux page.

Using Windows Mobile / CE 5.0 / CE 6.0

For a step-by-step guide on getting Phidgets running on Windows CE, check the Windows CE page.

Technical Details

The Infrared sensor can detect an object at 5mm. It measures the amount of energy from the object and returns a value between 0 and 1000. A returned value between 0 and 400 signifies that the object has been detected. Values over 400 indicate that there is no reflective object within range, or that an object is too close.

The 1102 sensor consists of an infrared emitting diode and an NPN silicon phototransistor mounted side by side on a converging optical axis in a black plastic housing. The phototransistor responds to radiation from the emitting diode only when a reflective object passes within its field of view. The area of the optimum response approximates a circle 5mm in diameter.

The amount of reflectivity is measured by Infrared and some materials that look very reflective to the human eye might not be as reflective in the infrared spectrum. The sensor can only detect objects that are between 3 to 7 mm

away; it cannot see any objects outside that range.

The 1102 is not a digital sensor. The returned value is inversely proportional to the amount of reflectivity of the object (0 being more reflective, and 400 being less reflective), but in practice the variation between sensors is broad enough that the 1102 should not be used to measure the reflectivity of an object - only to detect if the object exists. You may have trouble using this sensor through a pane of glass, since the IR light can easily reflect off of the surface of the glass.

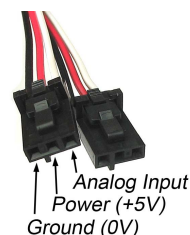
When working with sensors that are not digital, it is helpful to filter out noise by implementing a simple hysteresis in your code. By interpreting any `SensorValue < 400` as a detection, and not releasing the detection until `SensorValue` goes above some higher threshold, such as 500, multiple triggering can often be avoided.

Finally, the 1102 works best in a constrained environment, where objects can be mechanically guaranteed to be within the 3-7mm range, or not present at all.

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.

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



API

Phidget analog sensors do not have their own API- they simply output a voltage that is converted to a digital value and accessed through the "Sensor" properties and events on the `PhidgetInterfaceKit` API. It is not possible to programmatically identify which sensor is attached to the Analog Input. To an `InterfaceKit`, every sensor looks the same. Your application will need to apply formulas from this manual to the **SensorValue** (an integer that ranges from 0 to 1000) to convert it into the units of the quantity being measured. For example, this is how you would use a temperature sensor in a C# program:

```
// set up the interfacekit object
InterfaceKit IFK = new InterfaceKit();

// link the new interfacekit object to the connected board
IFK.open("localhost", 5001);

// Get sensorvalue from analog input zero
int sensorvalue = IFK.sensors[0].Value;

// Convert sensorvalue into temperature in degrees Celsius
double roomtemp = Math.Round(((sensorvalue * 0.22222) - 61.11), 1);
```

See the `PhidgetInterfaceKit` User Guide for more information on the API and a description of our architecture.

For more code samples, find your preferred language on the Languages page.

Product History

Date	Board Revision	Device Version	Comment
October 2005	0	N/A	Product Release

References

[1] http://www.phidgets.com/products.php?product_id=1102

Article Sources and Contributors

1102 User Guide *Source:* http://www.phidgets.com/wiki/index.php?title=1102_User_Guide *Contributors:* Burley, Mparadis

Image Sources, Licenses and Contributors

Image:1102.jpg *Source:* <http://www.phidgets.com/wiki/index.php?title=File:1102.jpg> *License:* unknown *Contributors:* Mparadis

File:1102_0_Connecting_The_Hardware.jpg *Source:* http://www.phidgets.com/wiki/index.php?title=File:1102_0_Connecting_The_Hardware.jpg *License:* unknown *Contributors:* Mparadis

File:Ph.jpg *Source:* <http://www.phidgets.com/wiki/index.php?title=File:Ph.jpg> *License:* unknown *Contributors:* Mparadis

File:1018_2_Control_Panel_Screen.jpg *Source:* http://www.phidgets.com/wiki/index.php?title=File:1018_2_Control_Panel_Screen.jpg *License:* unknown *Contributors:* Mparadis

File:Sensor_InterfaceKit_Screen_997.jpg *Source:* http://www.phidgets.com/wiki/index.php?title=File:Sensor_InterfaceKit_Screen_997.jpg *License:* unknown *Contributors:* Mparadis

File:1102_0_Advanced_Sensor_Form_Screen.jpg *Source:* http://www.phidgets.com/wiki/index.php?title=File:1102_0_Advanced_Sensor_Form_Screen.jpg *License:* unknown *Contributors:* Mparadis

File:Analoginput.jpg *Source:* <http://www.phidgets.com/wiki/index.php?title=File:Analoginput.jpg> *License:* unknown *Contributors:* Mparadis