

# Humidity Phidget

## Contents

### **1 GETTING STARTED**

### **2 USING THE HUM1000**

#### 2.1 Phidget Control Panel

##### 2.1.1 Windows

##### 2.1.2 macOS

#### 2.2 First Look

#### 2.3 Humidity

- 2.4 Temperature Sensor
- 2.5 Finding The Addressing Information
- 2.6 Using Your Own Program

### 3 TECHNICAL DETAILS

- 3.1 Current Consumption

### 4 WHAT TO DO NEXT

## Getting Started

Welcome to the HUM1000 user guide! In order to get started, make sure you have the following hardware on hand:

- HUM1000 - Humidity Phidget
- VINT Hub
- Phidget cable
- USB cable and computer

Next, you will need to connect the pieces:



1. Connect the HUM1000 to the VINT Hub using the Phidget cable.
2. Connect the VINT Hub to your computer with a USB cable.

Now that you have everything together, let's start using the HUM1000!

## Using the HUM1000

### Phidget Control Panel

In order to demonstrate the functionality of the HUM1000, the Phidget Control Panel running on a Windows machine will be used.

The Phidget Control Panel is available for use on both macOS and Windows machines.

### Windows

To open the Phidget Control Panel on Windows, find the  icon in the taskbar. If it is not there, open up the start menu and search for Phidget Control Panel

## macOS

To open the Phidget Control Panel on macOS, open Finder and navigate to the Phidget Control Panel in the Applications list. Double click on the  icon to bring up the Phidget Control Panel.

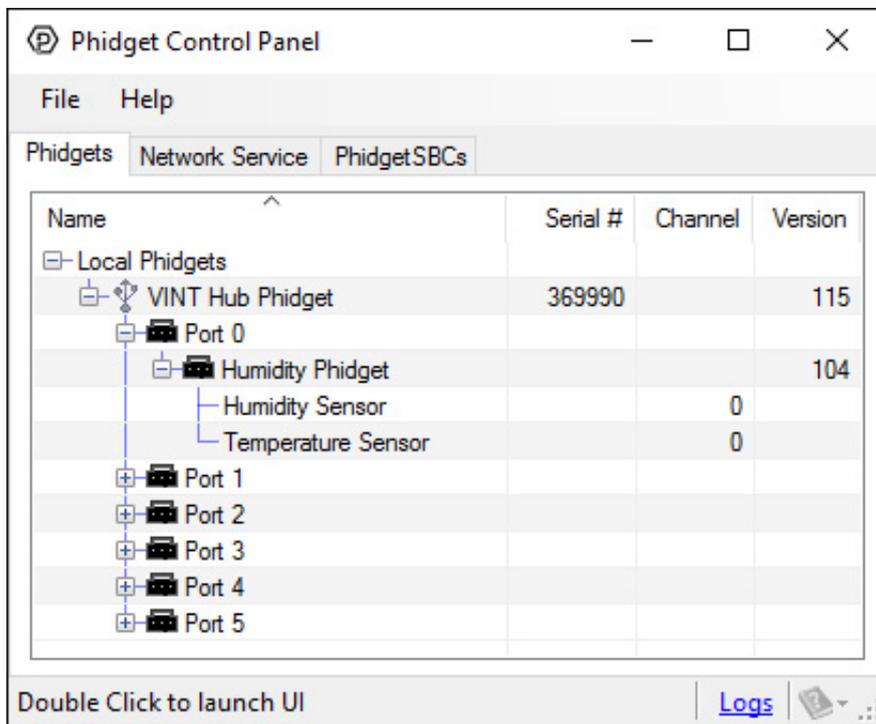
For more information, take a look at the getting started guide for your operating system:

- Getting started with Windows
- Getting started with macOS

Linux users can follow the getting started with Linux guide and continue reading here for more information about the HUM1000.

## First Look

After plugging the HUM1000 into your computer and opening the Phidget Control Panel, you will see something like this:



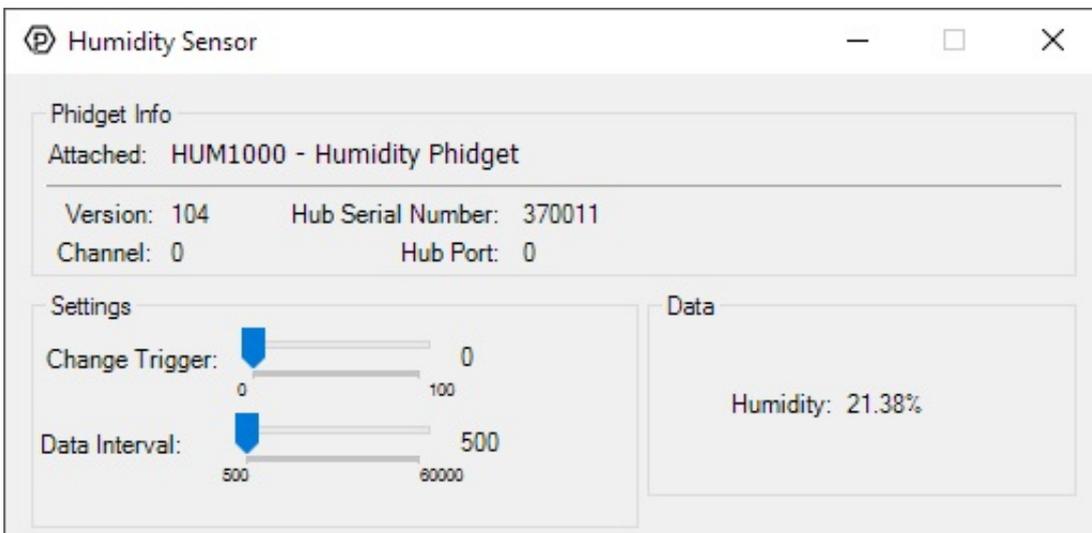
The Phidget Control Panel will list all connected Phidgets and associated objects, as well as the following information:

- **Serial number:** allows you to differentiate between similar Phidgets.
- **Channel:** allows you to differentiate between similar objects on a Phidget.
- **Version number:** corresponds to the firmware version your Phidget is running. If your Phidget is listed in red, your firmware is out of date. Update the firmware by double-clicking the entry.

The Phidget Control Panel can also be used to test your device. Double-clicking on an object will open an example.

## Humidity

Double-click on the Humidity Sensor object in order to run the example:

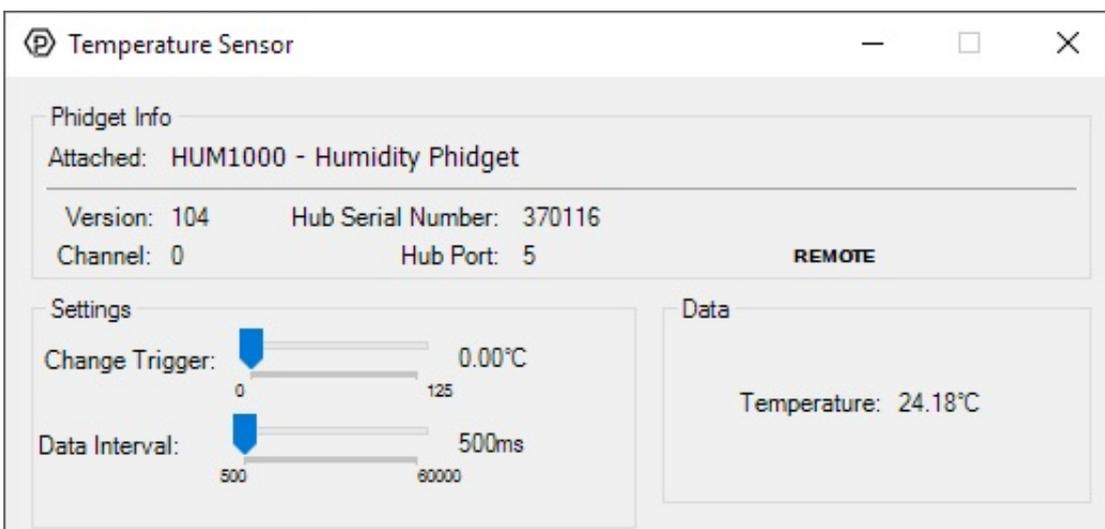


General information about the selected object will be displayed at the top of the window. You can also experiment with the following functionality:

- Modify the change trigger and/or data interval value by dragging the sliders. For more information on these settings, see the data interval/change trigger page.

## Temperature Sensor

Double-click on the Temperature Sensor object in order to run the example:

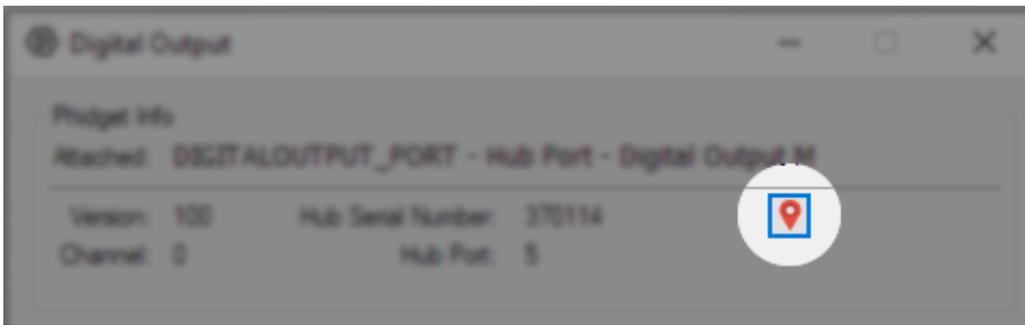


General information about the selected object will be displayed at the top of the window. You can also experiment with the following functionality:

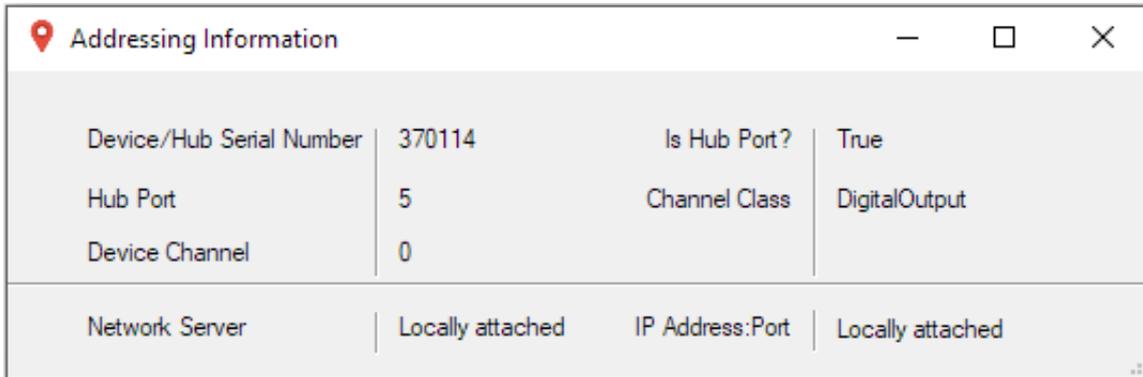
- Modify the change trigger and/or data interval value by dragging the sliders. For more information on these settings, see the data interval/change trigger page.
- The measured temperature can be seen next to the Temperature label. Cover the board with your hands to see the temperature quickly rise.

## Finding The Addressing Information

Before you can access the device in your own code, and from our examples, you'll need to take note of the addressing parameters for your Phidget. These will indicate how the Phidget is physically connected to your application. For simplicity, these parameters can be found by clicking the button at the top of the Control Panel example for that Phidget.



In the Addressing Information window, the section above the line displays information you will need to connect to your Phidget from any application. In particular, note the Channel Class field as this will be the API you will need to use with your Phidget, and the type of example you should use to get started with it. The section below the line provides information about the network the Phidget is connected on if it is attached remotely. Keep track of these parameters moving forward, as you will need them once you start running our examples or your own code.



## Using Your Own Program

You are now ready to start writing your own code for the device. The best way to do that is to start from our examples:

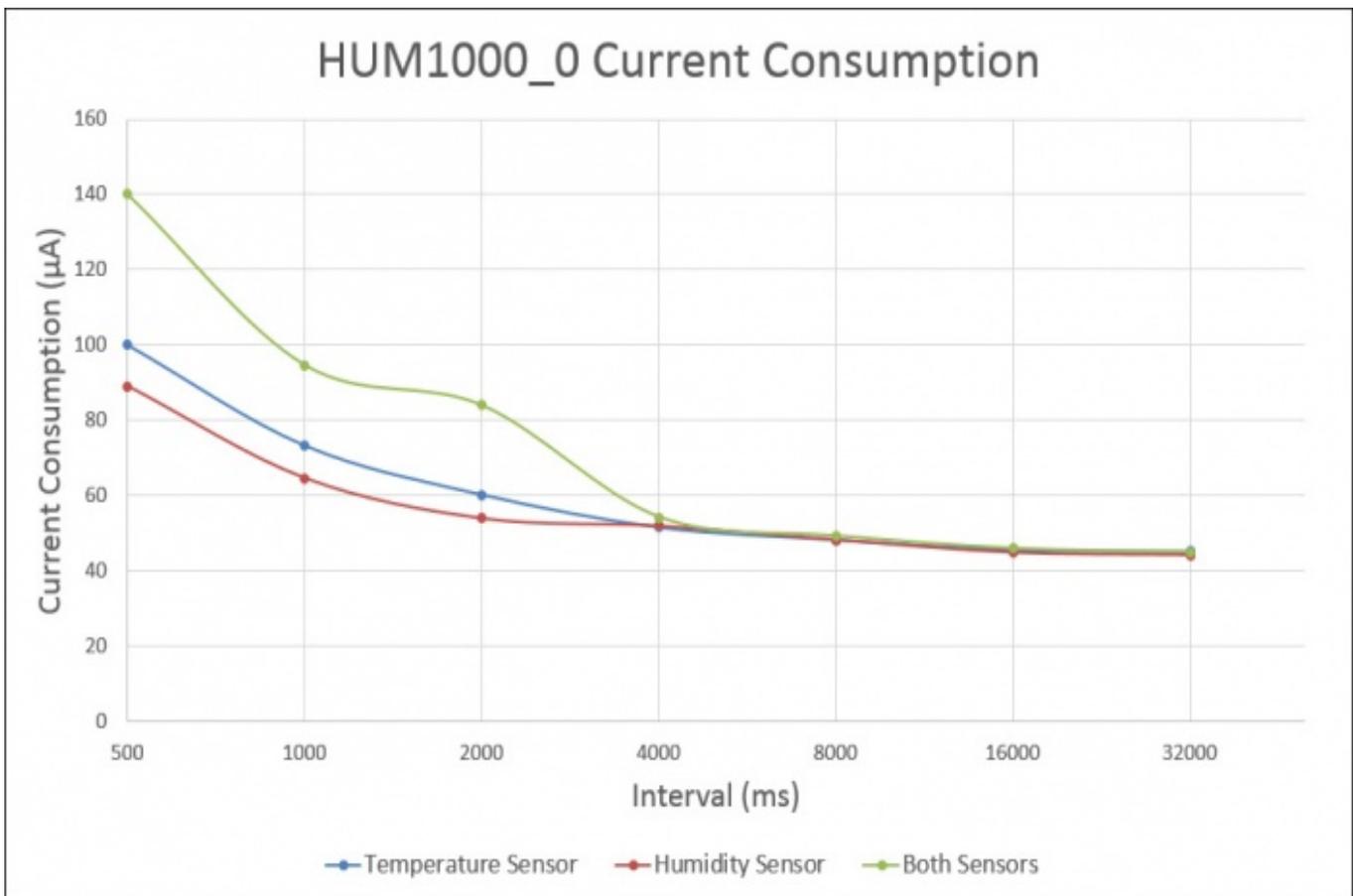
This Phidget is compatible with the HumiditySensor Examples and the TemperatureSensor Examples.

Once you have your example, you will need to follow the instructions on the page for your programming language to get it running. To find these instructions, select your programming language from the Software Overview page.

## Technical Details

### Current Consumption

The HUM1000's current consumption depends on which sensors are configured, and what the data interval is set to.



If neither of the sensors are configured or opened, the device will consume approximately 43µA.

## What to do Next

- Software Overview - Find your preferred programming language here to learn how to write your own code with Phidgets!
- General Phidget Programming - Read this general guide to the various aspects of programming with Phidgets. Learn how to log data into a spreadsheet, use Phidgets over the network, and much more.
- Phidget22 API - The API is a universal library of all functions and definitions for programming with Phidgets. Just select your language and device and it'll give you a complete list of all properties, methods, events, and enumerations that are at your disposal.