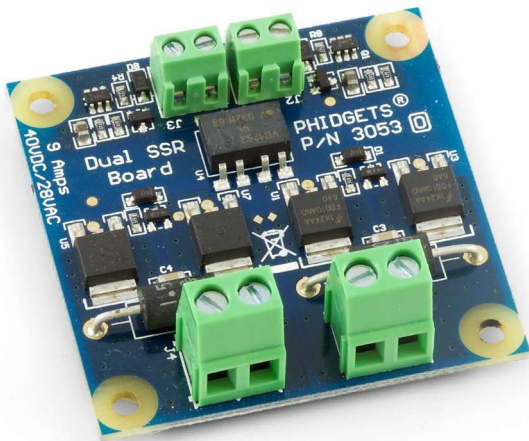


Product Manual

3053 - Dual SSR Relay Board



Phidgets 3053 - Product Manual

For Board Revision 0

© Phidgets Inc. 2010

Contents

4 Product Features

4 Connections

5 Getting Started

5 Checking the Contents

5 Connecting all the pieces

5 Testing Using Windows 2000/XP/Vista

6 Testing Using Mac OS X

7 Programming a Phidget

7 Code Samples

7 Coding for your Relay

8 Technical Information

8 Solid State Relays

8 Using the SSR Board

8 Isolation and Protection Devices

9 Other Interfacing Alternatives

9 Device Specifications

9 Product History

9 Support

Product Features

- The Solid State Relay is used to switch up to 9 Amps at up to 40 Volts DC, or 28 Volts AC.
- Allows any Phidget device with digital outputs to control larger loads and devices like AC or DC motors, electromagnets, solenoids, and incandescent light bulbs.
- Opto-coupled isolation between the control inputs and the outputs
- Includes protective measures against inductive transients and ESD

Connections

Designed to connect to a:

- 1010 - PhidgetInterfaceKit 8/8/8 Mini-Format
- 1011 - PhidgetInterfaceKit 2/2/2
- 1018 - PhidgetInterfaceKit 8/8/8
- 1019 - PhidgetInterfaceKit 8/8/8 w/6 Port Hub
- 1072 - PhidgetSBC2
- 1203 - PhidgetTextLCD

Getting Started

Checking the Contents

You should have received:

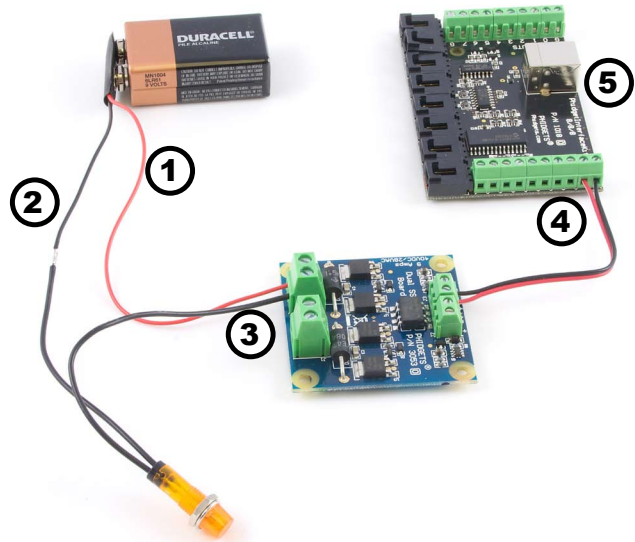
- A Dual SSR Board

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, a battery connector
- An incandescent bulb
- 2 pieces of wire

Connecting all the pieces

1. Connect the positive/red wire from the battery to one output terminal of the SSR Board
2. Connect negative/black wire from the battery to one of the bulb wire.
3. Connect the bulb wire to the other output terminal of the SSR board.
4. Attach the SSR Board to your PhidgetInterfaceKit by connecting a wire from the terminal block marked with a + to a digital output and another wire from the terminal block marked with a - to any ground terminal on the InterfaceKit.
5. Connect the PhidgetInterfaceKit Board to your computer using the USB cable.



Testing Using Windows 2000/XP/Vista/7

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 >> Drivers


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

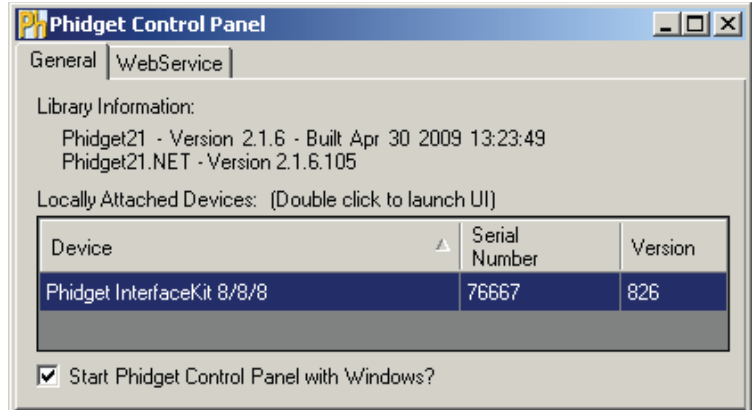
You should see the  icon on the right hand corner of the Task Bar.

Running Phidgets Sample Program

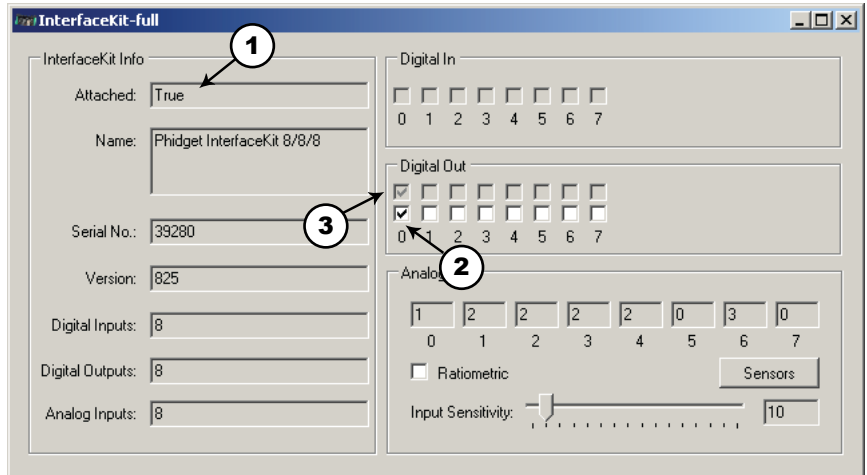
Double clicking on the  icon loads the Phidget Control Panel; we will use this program to make sure that your new Phidget works properly. Since the sensor is connected to a 1018, the computer will see only the 1018. The sensor is providing data through the Analog input it is connected to.

The source code for the InterfaceKit-full sample program can be found under C# by clicking on www.phidgets.com >> 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. Turn the bulb on and off by clicking on the Digital output box.
3. The bottom row of the digital out shows the status of the request, while the top row displays the status of the digital output as reported by the 3053 Relay.



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 >> 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 Relay

Relays do not have their own API, but instead they are controlled by a digital output on a PhidgetInterfaceKit. The PhidgetInterfaceKit API is used to set the state of the digital output. It is not possible to programmatically identify if a relay is attached to the digital output.

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

Technical Information

Solid State Relays

Solid State Relays, or SSRs, are devices designed to operate like standard relays but without mechanical motion. Built instead out of silicon transistors, SSRs allow currents and voltages to be switched by simple digital signals from a microprocessor or any other device that can supply the small amount of current needed to activate the SSR's internal switching mechanism.

Using the SSR Board

Using the SSR Relay Board in your application is typically done according to the diagram on the right, though there are other implementations for it as well.

The SSR control inputs can be controlled with a wide range of voltage levels between 3 and 30V. Be sure to match the polarity of the control signal with the +/- labels on the input terminal blocks.

One concern with any type of relay is the turn-on and turn-off times. Since the SSR Board uses optoisolation, the turn-on delay depends on how strong the control signal is, while the turn-off delay is quite constant, and quick. The 3053 uses internal circuitry to keep the current of the control signal around 10mA, no matter how what voltage level is used. To improve the turn-on time of the SSR, a short current spike (approximately 35mA for 300µs) at turn-on is possible if your control signal can provide it. This allows the SSR to turn on quicker, reducing heating during the turn-on transition. If your control signal is not able to provide the 35mA, the SSR will turn on anyway, but will take up to 50% longer.

If the Phidget InterfaceKit 8/8/8 is being used to drive the input, the control signal has an output resistance of about 250 Ohms, and the SSR has a turn-on time of 1.90ms. The turn-off time is approximately 160µs.

Isolation and Protection Devices

The SSR Board is safe to use with sensitive control devices like microprocessors, and will not damage a Phidget device or your PC. This is achieved through the use of an optoisolation chip - creating a physical barrier between the SSR control inputs and the potentially high voltage and high current outputs.

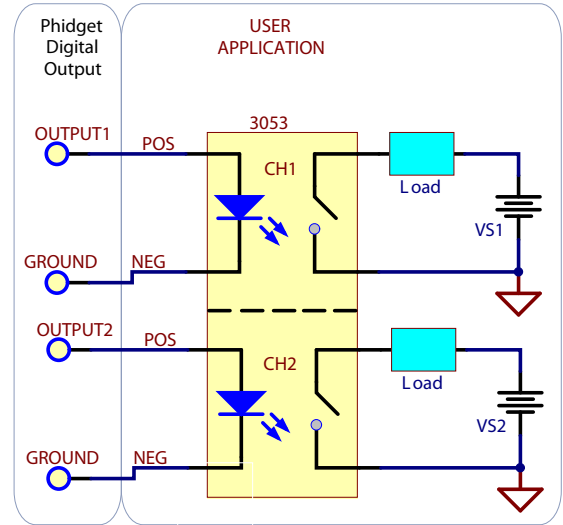
In addition to the optoisolation between the inputs and outputs, there is an on-board 47V voltage suppression device across the relay output that protects the board from static electricity and surges from inductive loads.

To protect the optoisolation LEDs from potential overcurrent on the inputs, a transistor-based current limiting circuit has been added. The circuit limits the constant current travelling through the LEDs to 10mA, regardless of the input voltage. The transistors dissipate what would have been extra current in the form of heat.

The 3053 Dual SSR Board comprises of two channels. Both channels are electrically and physically isolated from each other, and are controlled individually.

Warning

The SSR Board is able to safely switch at a speed of 20Hz. This board is not intended as a high frequency switch due to the extreme heating that will occur. Although it is possible to switch the SSR Board at up to 300Hz, it easily produces enough heat to liquefy the solder underneath the transistors (which requires at least 240°C) at a 3A load.



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.

Device Specifications

Characteristic	Value
Output Contact On Resistance	0.05 ohms
Isolation Voltage (Input to Output), time = 60s minimum	5300 Vrms
Working Voltage for Isolation	2500 Vrms
Maximum DC Switching Voltage	40 VDC
Maximum AC Switching Voltage	28 VAC
Maximum Switching Current (AC or DC)	9A per channel
Maximum Continuous Current @ 25°C (AC or DC)	9A per channel
Min/Max Input Voltage (Control)	3 to 30V
Maximum Switching Speed at Zero load	300 Hz
Turn-On Time at Zero load with 50 Ohm input impedance	900 µs
Turn-On Time switching 9A with 50 Ohm input impedance	1.4 ms
Turn-Off Time at Zero load	160 µs
Turn-Off Time switching 9A	160 µs
Turn-On Time at Zero load using 8/8/8 on inputs	1.90 ms
Turn-Off Time at Zero load using 8/8/8 on inputs	130 µs
Input Wire Gauge	16 to 26 AWG
Output Wire Gauge	12 to 24 AWG

Product History

Date	Board Revision	Comment
May 2010	0	Product Release

Support

Call the support desk at 1.403.282.7335 9:00 AM to 5:00 PM Mountain Time (US & Canada) - GMT-07:00

or

E-mail us at: support@phidgets.com