

# Digital Input Primer

---

## Introduction

Digital Inputs can be used to convey the state of various devices such as push buttons, limit switches, relays, or logic level outputs. They have two states: Low (or TRUE), and High (or FALSE). Any signal that is expected to interact with more than just these 2 states is inappropriate for this type of input.

Digital inputs are one of the easiest components to work with, since all that is required is a simple check to see which state they are in when an event triggers.

**Note:** The digital inputs on the 1012 - PhidgetInterfaceKit 0/16/16 are active high rather than active low, so the instructions on this page do not apply. For information on the inputs on the 1012 please refer to the technical section of it's User Guide.

## Specifications

### Digital Input Hardware Filter

There is built-in filtering on the digital input, to eliminate false triggering from electrical noise. The digital input is first RC filtered by a 15K/100nF node, which will reject noise of higher frequency than 1kHz. This filter generally eliminates the need to shield the digital input from inductive and capacitive coupling likely to occur in wiring harnesses. You can further reduce noise by externally filtering the input signal, but you will lose sensitivity in the process.

### Digital Input Hysteresis

The digital input has hysteresis - that is, it will hold its current state (false or true), unless a large change occurs. To guarantee FALSE, the digital input must be at least 3.75V, and to guarantee TRUE, the digital input must be less than 1.25V.

## Digital Input Sampling Characteristics

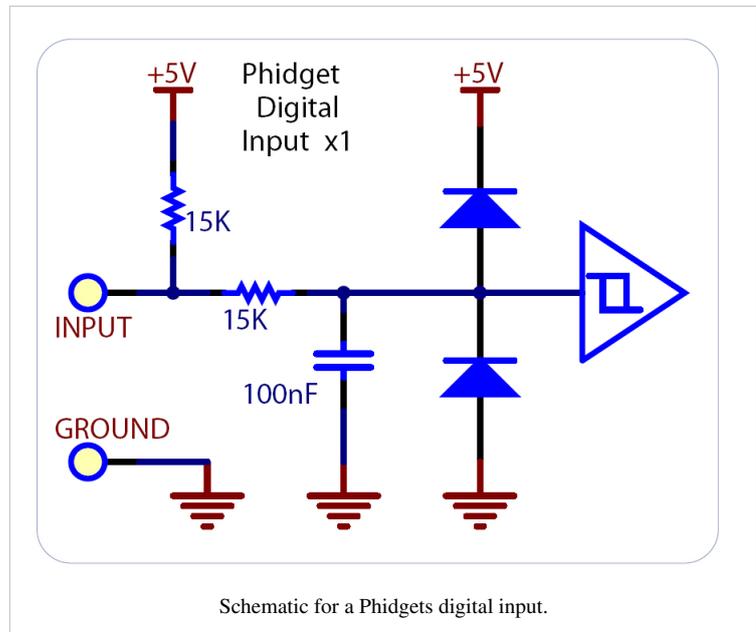
The state of the digital inputs are reported back to the PC periodically. During this sampling period, if a digital input was true for greater than 4.0ms, the digital input is guaranteed to be reported as true in software. This makes the digital input much more sensitive to reporting TRUE state, and makes it useful to watch for short events. Any Digital Input True events of less than 1.5ms are never reported.

## Electrical Specifications

The digital inputs have a built in 15K pull-up resistor. By connecting external circuitry, and forcing the input to Ground, the Digital Input in software will read as TRUE. The default state is FALSE - when you have nothing connected, or your circuitry (switch, etc) is not pulling the input to ground.

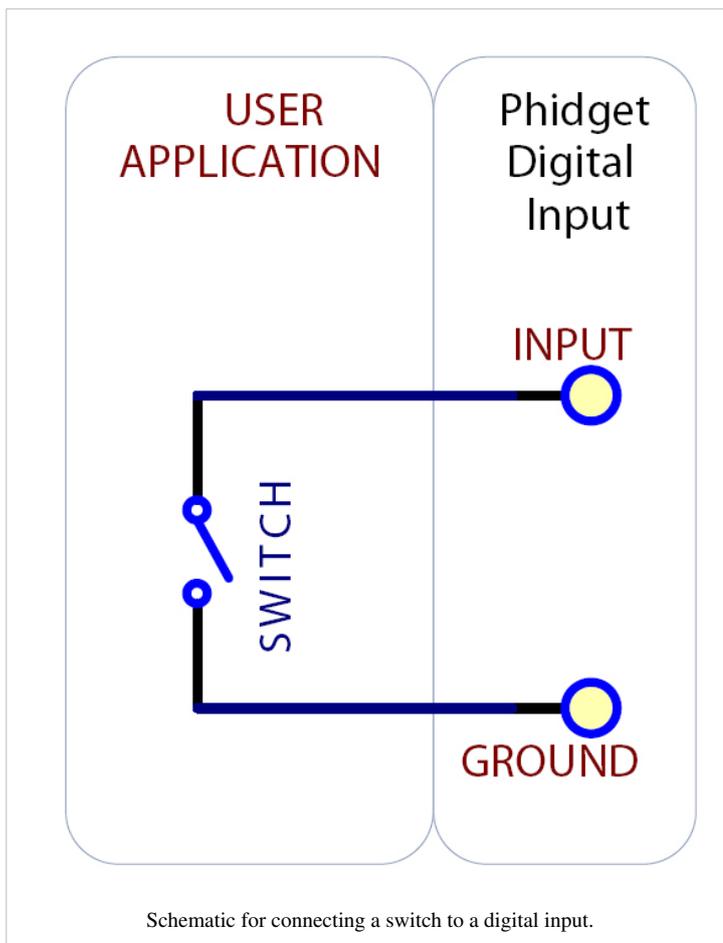
## Using the Digital Inputs

Here are some circuit diagrams that illustrate how to connect various devices to the digital inputs on your Phidget.



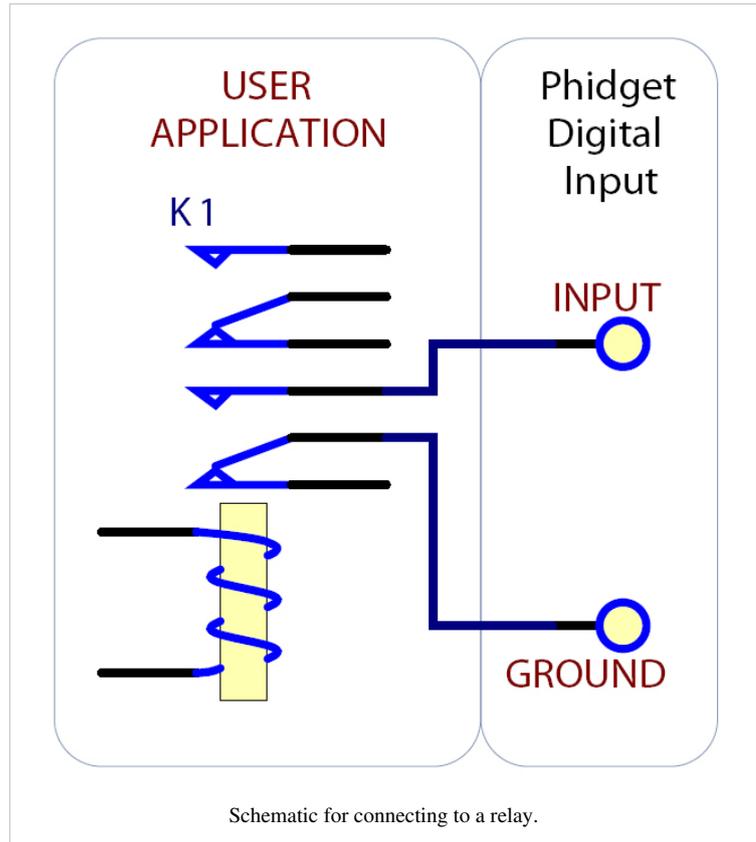
## Wiring a switch to a Digital Input

Closing the switch causes the digital input to report TRUE.



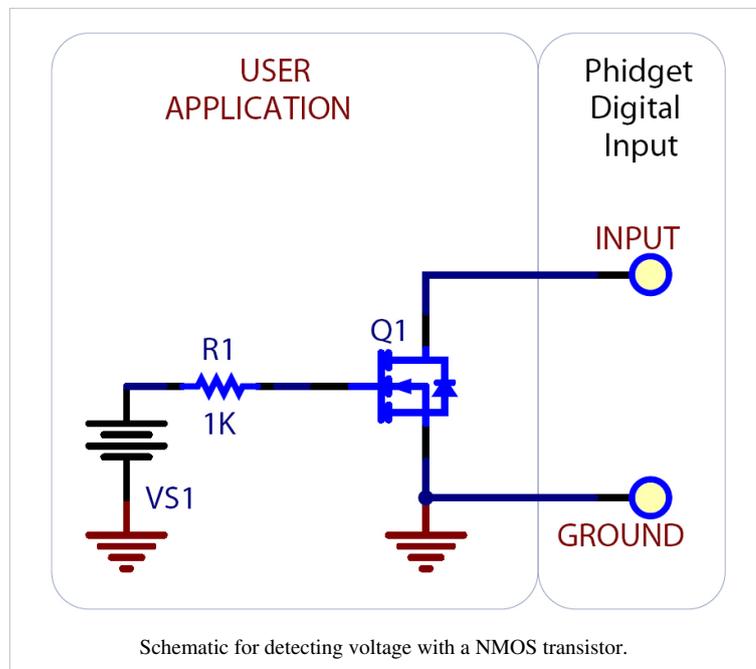
## Monitoring the Position of a Relay

The relay contact can be treated as a switch, and wired up similarly. When the relay contact is closed, the Digital Input will report TRUE.



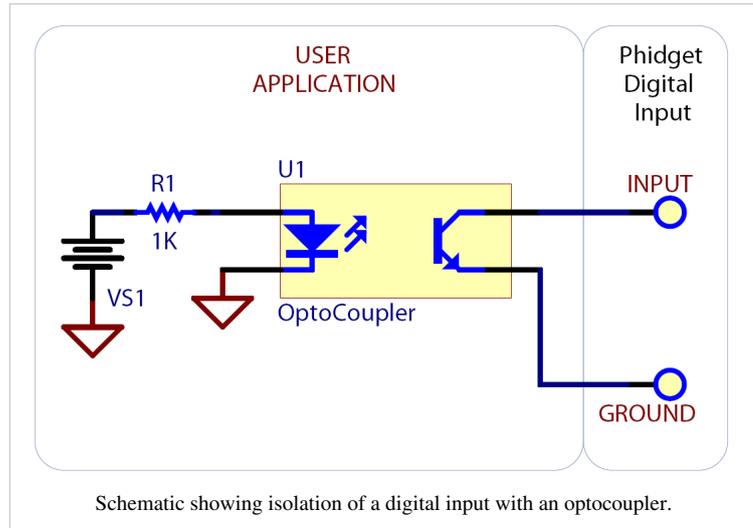
## Detecting an External Voltage with an N-Channel MOSFET

A MOSFET can be used to detect the presence of an external voltage. The external voltage will turn on the MOSFET, causing it to short the Digital Input to Ground. If the MOSFET is conducting  $> 270\mu\text{A}$ , the Digital Input is guaranteed to report TRUE. If the MOSFET is conducting  $< 67\mu\text{A}$ , the Digital Input is guaranteed to report FALSE. The voltage level required to turn on the MOSFET depends on the make of MOSFET you are using. Typical values are 2V-6V.



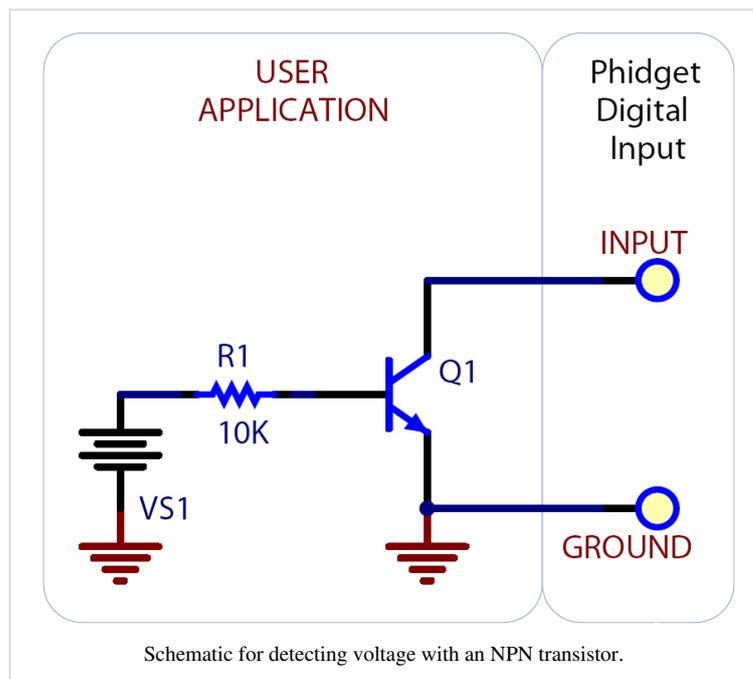
## Isolating a Digital Input with an Optocoupler

When driving current through the LED, the Digital Input will report TRUE. The amount of current required will depend on the optocoupler used. Design to sink at least 270uA to cause the digital input to report TRUE, and less than 67uA to report FALSE.



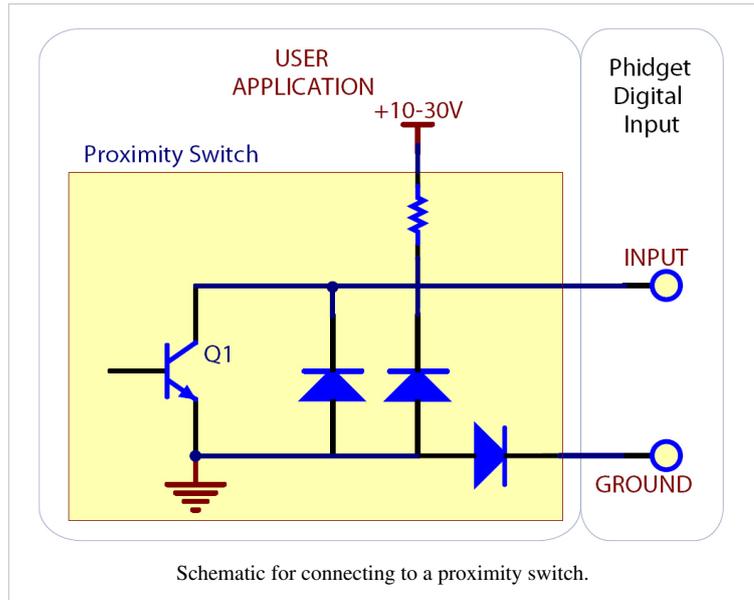
## Detecting an External Voltage with an NPN Transistor

This circuit can be used to measure if a battery is connected, or if 12V (for example) is on a wire. By designing to have Collector-Emitter current  $> 270\mu\text{A}$ , the digital input will report TRUE.



### Using a Capacitive or Inductive Proximity Switch

Capacitive proximity switches can detect the presence of nearby non-metallic objects, whereas inductive proximity switches can detect only the presence of metallic objects. To properly interface one of these proximity switches to the digital inputs, a 3-wire proximity switch is required, as well as an external power supply. We have checked the following switch from Automation Direct to verify that it works with the Digital Inputs. Similar capacitive or inductive proximity switches from other manufacturers should work just as well.

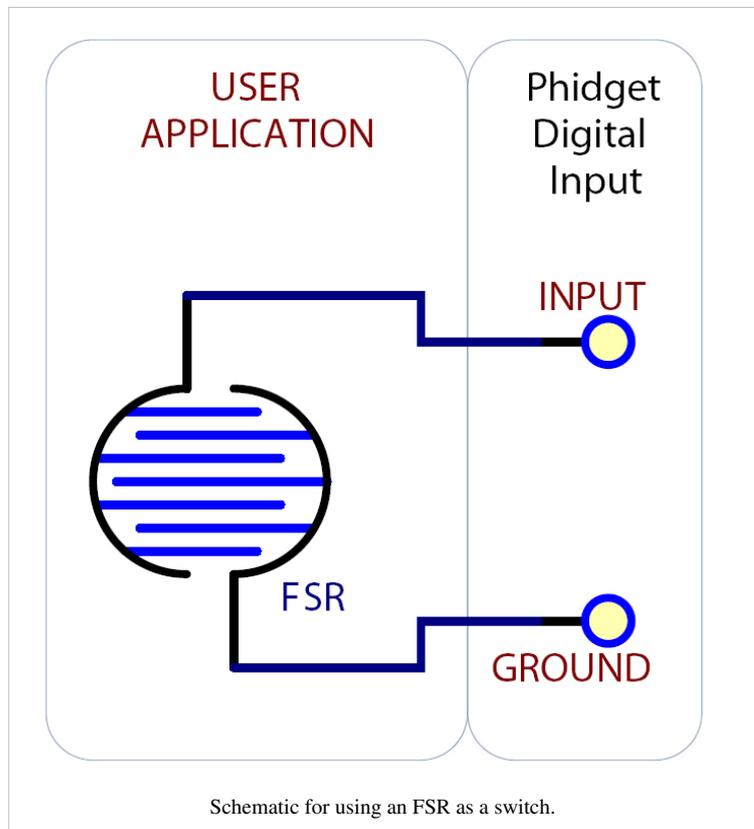


### Switches

Manufacturer	Web Page	Capacitive Part No.	Inductive Part No.
Automation Direct	<a href="http://www.automationdirect.com">www.automationdirect.com</a>	CT1 Series	AM1 Series

### Using an FSR or Variable Resistor as a Switch

The digital inputs can be easily wired to use many variable resistors as switches. If the resistance falls below 3.75k Ohms, the Digital Input will go TRUE. If the resistance rises above 75k Ohms, the Digital Input will go FALSE.



# Article Sources and Contributors

**Digital Input Primer** *Source:* [http://www.phidgets.com/wiki/index.php?title=Digital\\_Input\\_Primer](http://www.phidgets.com/wiki/index.php?title=Digital_Input_Primer) *Contributors:* Burley, Cora, Mparadis

## Image Sources, Licenses and Contributors

**Image:digital\_input.jpg** *Source:* [http://www.phidgets.com/wiki/index.php?title=File:Digital\\_input.jpg](http://www.phidgets.com/wiki/index.php?title=File:Digital_input.jpg) *License:* unknown *Contributors:* Mparadis

**Image:switch\_digital\_input.jpg** *Source:* [http://www.phidgets.com/wiki/index.php?title=File:Switch\\_digital\\_input.jpg](http://www.phidgets.com/wiki/index.php?title=File:Switch_digital_input.jpg) *License:* unknown *Contributors:* Mparadis

**Image:relay\_position.jpg** *Source:* [http://www.phidgets.com/wiki/index.php?title=File:Relay\\_position.jpg](http://www.phidgets.com/wiki/index.php?title=File:Relay_position.jpg) *License:* unknown *Contributors:* Mparadis

**Image:external\_voltage\_mosfet.jpg** *Source:* [http://www.phidgets.com/wiki/index.php?title=File:External\\_voltage\\_mosfet.jpg](http://www.phidgets.com/wiki/index.php?title=File:External_voltage_mosfet.jpg) *License:* unknown *Contributors:* Mparadis

**Image:isolating\_digital\_input\_optocoupler.jpg** *Source:* [http://www.phidgets.com/wiki/index.php?title=File:Isolating\\_digital\\_input\\_optocoupler.jpg](http://www.phidgets.com/wiki/index.php?title=File:Isolating_digital_input_optocoupler.jpg) *License:* unknown *Contributors:* Mparadis

**Image:external\_voltage\_npn.jpg** *Source:* [http://www.phidgets.com/wiki/index.php?title=File:External\\_voltage\\_npn.jpg](http://www.phidgets.com/wiki/index.php?title=File:External_voltage_npn.jpg) *License:* unknown *Contributors:* Mparadis

**Image:proximity\_switch.jpg** *Source:* [http://www.phidgets.com/wiki/index.php?title=File:Proximity\\_switch.jpg](http://www.phidgets.com/wiki/index.php?title=File:Proximity_switch.jpg) *License:* unknown *Contributors:* Mparadis

**Image:fsr\_switch.jpg** *Source:* [http://www.phidgets.com/wiki/index.php?title=File:Fsr\\_switch.jpg](http://www.phidgets.com/wiki/index.php?title=File:Fsr_switch.jpg) *License:* unknown *Contributors:* Mparadis