1024 User Guide

Go to this device's product page [1]

Getting Started

Checking the Contents

You should have received:	In order to test your new Phidget you will also need:
• A PhidgetRFID • A USB Cable	• A compatible RFID tag

Connecting the Pieces

Connect the PhidgetRFID board to the computer 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

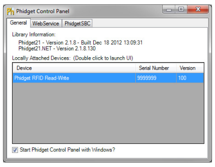
Double clicking on the Ph icon loads the Phidget Control Panel; we will use this program to ensure that your new Phidget works properly.

The source code for the **PhidgetRFID-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If an entry in this list is red, it means the firmware for that device is out of date.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 **Phidget RFID** is properly attached to your PC.



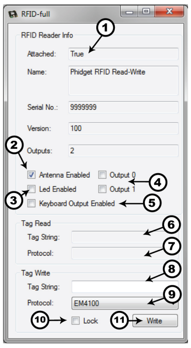
Library Information:

Phidget21 - Version 2.1.8 - Built Dec 18 2012 13:09:31
Phidget21.NET - Version 2.1.8.130

Locally Attached Devices: (Double click to launch UI)

Device	Serial Number	Version
Phidget RFID Read-Write	9999999	100

☑ Start Phidget Control Panel with Windows?



RFID Reader Info

Attached: True
Name: Phidget RFID Read-Write
Serial No.: 9999999
Version: 100
Outputs: 2

☑ Antenna Enabled ☐ Output 0
☐ Led Enabled ☐ Output 1
☐ Keyboard Output Enabled

Tag Read
Tag String:
Protocol:

Tag Write
Tag String:
Protocol: EM4100
☐ Lock

1. Double Click on **Phidget RFID Read-Write** in the Phidget Control Panel to bring up RFID-full and check that the box labelled Attached contains the word True.
2. Click on the Antenna Enabled box to enable the antenna.
3. Click on the LED Enabled box to turn the on-board LED on or off.
4. The Output 0 box controls the +5V digital output and the Output 1 box controls the external LED digital output.
5. When the Keyboard Output is Enabled, the identification string will be written at the cursor point in an application running in active window on the PC (such as a word document, for example).
6. Bring an RFID tag close to the PhidgetRFID board and check that the identification string is displayed. Make sure that the Antenna is enabled for this step.
7. When a tag is read, its protocol is displayed here.
8. Type the tag string here when writing to a writeable tag. You must use the proper string length and characters according to your chosen protocol.
9. When writing, choose your protocol from this drop-down list. For more information on supported protocols, see the Tag Types section.
10. When checked, this box will permanently write to a tag. **Once a tag is written with the lock on, it can not be re-written.**
11. Once you've selected the protocol and string, press this button to write to the tag currently within range.

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 RFID is properly attached.
5. Double Click on Phidget RFID in the Phidget Preference Pane to bring up the RFID-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

Tag Types

We support read-only tags that have been programmed with a supported protocol, as well as T5577 type tags for writing.

T5577

T5577 tags can be written with any of the supported protocols. Fresh T5577 tags that have never been programmed may show up as an EM4100 tag, or may not show up at all, depending on how the factory programmed them. After writing, they will always show up as the written tag. We also support a lock function which prevents a tag from ever being re-written.

Protocols

A protocol is a way of encoding data on an RFID tag. This is not the same as the tag type. For example, we support the T5577 tag type, which can be programmed with any of the protocols which we support. We also support read-only tags which have been programmed in any of these protocols.

We support reading and writing in 3 protocols with the 1024:

EM4100

EM4100 (also known as EM4102) is the protocol that all previous PhidgetRFID readers have supported. Therefore, if you want to use the 1024 to write to writable tags to be read with previous versions of the PhidgetRFID, you need to write them in this protocol first. This protocol encodes 40 bits of arbitrary data. Read-only tags that are factory programmed with this protocol are supposed to be unique.

Phidgets represents this protocol as a 10-digit hex string (include leading 0's) ie. "0087f3bc91". This is the format to use for writing new tags, and to expect from the tag events.

ISO11785 FDX-B

ISO11785 defines tags used for animal ID. If you have a pet cat / dog, chances are fairly high that they have one of these tags implanted. FDX-B refers to the way that the ISO11785 data is encoded on the RFID tag, and is the industry-standard encoding scheme.

This tag consists of a 10-bit country code and a 38-bit unique ID.

The country code is ISO 3166 [2]. The '999' code is set aside for testing.

The unique ID is 38-bit unsigned, so that's a range of 0 - 274,877,906,943.

Phidgets represents this protocol as a 15-digit decimal number string - concatenated 3-digit country code and 12-digit id. ie. "999000000000123" would represent the testing country code and an id of 123. please note that the 12-digit id part cannot exceed the 38-bit maximum integer value of 274,877,906,943.

Note that Animal tags with a valid country code are supposed to be unique. Of course, with the 1024 you can freely copy an existing Animal Tag.

PhidgetsTAG

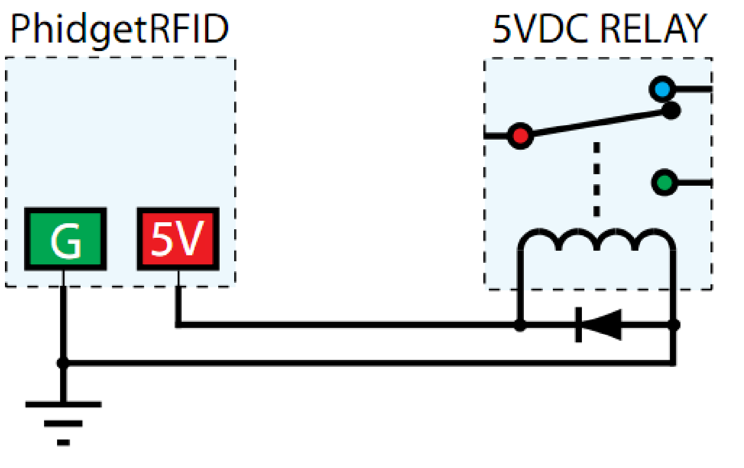
The PhidgetsTAG protocol is an internal protocol only supported by the PhidgetRFID 1024.

This protocol allows storing an ASCII string, up to 24 characters. ie. "I am a Phidgets Tag!"

The ASCII data must be 7-bit, so no extended ASCII support, but standard text is all supported (as well as control codes).

Controlled Outputs

The PhidgetRFID has four outputs - two of which are available to the user, and two of which are for internal control of the Phidget board only. Output 0 is a +5V source from the USB bus through a P-Channel MOSFET with less than one ohm impedance. This can be used to switch a TTL or CMOS device, or it can be used to drive a 5VDC relay such as the Aromat JS1-5V. Output 1 is an LED drive output at 5VDC with maximum 15mA of available current (250 ohm CMOS output). Both Output 0 and 1 are available in hardware at the terminal blocks on the PhidgetRFID board. If Output 0 is used to drive a relay, a fast clamping diode must be placed across the relay drive pins as shown in the diagram on the right. Not doing so can result in permanent damage to the PhidgetRFID board.



Make sure to attach a diode across the terminals of the relay if you are using output 0 on the RFID board to drive it.

1024 Outputs

Output	Function	Connection
0	+5VDC Source	Terminal Block
1	External LED Drive	Terminal Block
LED	Internal LED Drive	Internal Only
RF Enable	RF Antenna Enable	Internal Only

Interfering Signals

If you are using multiple RFID readers, placing them too close together will cause interference when reading tags. You could work around this problem by rapidly "polling" each 1024 by turning the antenna on, checking for tags, and then turning it off in sequence. Of course, this will lengthen the amount of time it takes for your system to read a tag, since you may have to wait for the nearest reader to become active.

Object Speed

When trying to read tags, you should allow the tag to remain within detection range for at least 50ms. Tags moving through the detection area faster than this may not register at all.

Further Reading

For more information on RFID readers and tags, visit the RFID Primer.

API

We document API Calls specific to this product in this section. Functions common to all Phidgets and functions not applicable to this device are not covered here. This section is deliberately generic. For calling conventions under a specific language, refer to the associated API manual in the Quick Downloads section for that language. For exact values, refer to the device specifications.

Functions

int OutputCount () [get] : Constant

Returns the number of digital outputs available on this PhidgetRFID. These are the outputs provided by the terminal block.

bool OutputState (int OutputIndex) [get,set]

Sets/Returns the state of an output. True indicates activated, False deactivated. False is the default state.

bool AntennaOn() [get,set]

Sets/Returns the state of the antenna. True turns the antenna on, False turns it off. The antenna if by default turned off, and needs to be explicitely activated before tags can be read.

bool LEDOn() [get,set]

Sets/Returns the state of the onboard LED. True turns the LED on, False turns it off. The LED is by default turned off.

string LastTag () [get]

Returns the last tag read. This method will only return a valid tag after a tag has been seen. This method can be used even after a tag has been removed from range of the reader.

int LastTagProtocol () [get]

Returns the protocol of the last tag read. This method will only return a valid protocol after a tag has been seen. This method can be used even after a tag has been removed from range of the reader.

bool TagStatus() [get]

Returns the state of whether or not a tag is being read by the reader. True indicates that a tag is on (or near) the reader.

void Write(string tag, int protocol, boolean lock)() [get]

Writes new data to a tag, using the specified protocol. Lock will lock the tag from further writes. See above for string formats according to protocol.

Events

OnOutputChange(int OutputIndex, bool State) [event]

An event issued when an output has changed.

OnTag(string tagString, int protocol) [event]

An event issued when a new tag is seen by the reader. The event is only fired one time for a new tag, so the tag has to be removed and then replaced before another OnTag event will fire.

Note: in some languages this event is called Tag2

Note: it is very important not to block in this event, or you will receive extra attach / detach events.

OnTagLost(string tagString, int protocol) [event]

An event issued when a tag is removed from the reader.

Note: in some languages this event is called TagLost2

Compliance

Phidgets Inc

1024_0

FCC ID: SUT1024-0

This device complies with Part 15 of the FCC Rules. Operation is subject to the following two conditions:

(1) This device may not cause harmful interference, and

(2) This device must accept any interference received, including interference that may cause undesired operation.

Note: The manufacturer is not responsible for any radio or TV interference caused by unauthorized modifications to this equipment. Such modifications could void the user’s authority to operate the equipment.

The user is cautioned that any changes or modifications not expressly approved by the party responsible for compliance could void the user’s authority to operate the equipment.

This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/TV technician for help.

This unit was tested with shielded cables on the peripheral devices. Shielded cables must be used with the unit to ensure compliance.

Product History

The 1024 - PhidgetRFID Read/Write replaced our previous RFID Reader, the 1023 - PhidgetRFID [3].

Date	Board Revision	Device Version	Comment
February 2013	0	100	Product Release

References

[1] http://www.phidgets.com/products.php?product_id=1024
[2] http://en.wikipedia.org/wiki/ISO_3166-1_numeric
[3] http://www.phidgets.com/products.php?product_id=1023

Article Sources and Contributors

1024 User Guide *Source*: http://www.phidgets.com/wiki/index.php?title=1024_User_Guide *Contributors*: Mparadis

Image Sources, Licenses and Contributors

Image:1024.jpg *Source*: http://www.phidgets.com/wiki/index.php?title=File:1024.jpg *License*: unknown *Contributors*: Mparadis

File:1024_0_Connecting_The_Hardware.jpg *Source*: http://www.phidgets.com/wiki/index.php?title=File:1024_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:1024_0_Control_Panel_Screen.jpg *Source*: http://www.phidgets.com/wiki/index.php?title=File:1024_0_Control_Panel_Screen.jpg *License*: unknown *Contributors*: Mparadis

File:1024_0_RFID_Screen.jpg *Source*: http://www.phidgets.com/wiki/index.php?title=File:1024_0_RFID_Screen.jpg *License*: unknown *Contributors*: Mparadis

image:Rfid_relay.png *Source*: http://www.phidgets.com/wiki/index.php?title=File:Rfid_relay.png *License*: unknown *Contributors*: Burley

File:Fcc-logo.png *Source*: http://www.phidgets.com/wiki/index.php?title=File:Fcc-logo.png *License*: unknown *Contributors*: Mparadis