GPIO Control from Command Line

Jan 2017

When creating projects using the GPIO pins on a Raspberry Pi, you can control all of the GPIO functions from Python. However, after attaching some hardware to the Raspberry Pi, it can be nice to toggle the GPIO pins to check functionality (especially handy if you are controlling something visual like an LED or a motor).

Install WiringPi

WiringPi contains the command line utility gpio that we can use to control our GPIO pins. To install WiringPi, use the following command.

```
sudo apt-get install wiringpi
```

Once installed, you will have access to the gpio utility.

As a quick test, from the command line type:

```
gpio -v
```

This will display the version of WiringPi you have installed.

```
gpio version: 2.32
Copyright (c) 2012-2015 Gordon Henderson
This is free software with ABSOLUTELY NO WARRANTY.
For details type: gpio -warranty

Unable to determine hardware version. I see: Hardware   : BCM2835
,
 - expecting BCM2708 or BCM2709.
If this is a genuine Raspberry Pi then please report this
to projects@drogon.net. If this is not a Raspberry Pi then you
are on your own as wiringPi is designed to support the
Raspberry Pi ONLY.
```

Another test, and something you can use a lot to check the status of the gpio is the </color red>readall</color> switch.

```
gpio readall
```

This will give an output similar to below (This output is from a Pi Zero v1.3)

```
 +-----+-----+---------+------+---+-Pi Zero--+---+------+---------+-----+-----+
 | BCM | wPi |   Name  | Mode | V | Physical | V | Mode | Name    | wPi | BCM |
 +-----+-----+---------+------+---+----++----+---+------+---------+-----+-----+
 |     |     |    3.3v |      |   |  1 || 2  |   |      | 5v      |     |     |
 |   2 |   8 |   SDA.1 |   IN | 1 |  3 || 4  |   |      | 5V      |     |     |
 |   3 |   9 |   SCL.1 |   IN | 1 |  5 || 6  |   |      | 0v      |     |     |
 |   4 |   7 | GPIO. 7 |   IN | 1 |  7 || 8  | 1 | ALT0 | TxD     | 15  | 14  |
 |     |     |      0v |      |   |  9 || 10 | 1 | ALT0 | RxD     | 16  | 15  |
 |  17 |   0 | GPIO. 0 |   IN | 0 | 11 || 12 | 0 | OUT  | GPIO. 1 | 1   | 18  |
 |  27 |   2 | GPIO. 2 |   IN | 0 | 13 || 14 |   |      | 0v      |     |     |
 |  22 |   3 | GPIO. 3 |   IN | 0 | 15 || 16 | 0 | OUT  | GPIO. 4 | 4   | 23  |
 |     |     |    3.3v |      |   | 17 || 18 | 1 | OUT  | GPIO. 5 | 5   | 24  |
 |  10 |  12 |    MOSI |   IN | 0 | 19 || 20 |   |      | 0v      |     |     |
 |   9 |  13 |    MISO |   IN | 0 | 21 || 22 | 0 | OUT  | GPIO. 6 | 6   | 25  |
 |  11 |  14 |    SCLK |   IN | 0 | 23 || 24 | 1 | IN   | CE0     | 10  | 8   |
 |     |     |      0v |      |   | 25 || 26 | 1 | IN   | CE1     | 11  | 7   |
 |   0 |  30 |   SDA.0 |   IN | 1 | 27 || 28 | 1 | IN   | SCL.0   | 31  | 1   |
 |   5 |  21 | GPIO.21 |  OUT | 1 | 29 || 30 |   |      | 0v      |     |     |
 |   6 |  22 | GPIO.22 |  OUT | 0 | 31 || 32 | 0 | IN   | GPIO.26 | 26  | 12  |
 |  13 |  23 | GPIO.23 |  OUT | 1 | 33 || 34 |   |      | 0v      |     |     |
 |  19 |  24 | GPIO.24 |  OUT | 0 | 35 || 36 | 0 | OUT  | GPIO.27 | 27  | 16  |
 |  26 |  25 | GPIO.25 |  OUT | 0 | 37 || 38 | 0 | OUT  | GPIO.28 | 28  | 20  |
 |     |     |      0v |      |   | 39 || 40 | 1 | OUT  | GPIO.29 | 29  | 21  |
 +-----+-----+---------+------+---+----++----+---+------+---------+-----+-----+
 | BCM | wPi |   Name  | Mode | V | Physical | V | Mode | Name    | wPi | BCM |
 +-----+-----+---------+------+---+-Pi Zero--+---+------+---------+-----+-----+
```

It can be seen that this Pi has GPIO set to OUT and some to IN. I am using the Pi as a clock, so these are the lines going to my LED display.

Using gpio command

Before looking at the commands, there is a switch I want to talk about, this is -g. If you use the -g switch you are operating on BCM numbers, not PIN numbers. I always use BCM numbers, but some people prefer pin numbers.

To set the function of the GPIO pin, use gpio -g mode

```
using BCM number
gpio -g mode 18 out - Sets GPIO18 to OUTput mode (GPIO18 is physical pin 12)
using pin instead of bcm
gpio mode 12 out - Sets GPIO18 to OUTput mode (GPIO18 is physical pin 12)
Modes are In, Out, PWM, Up, Down and Tri.
```

Set GPIO pin to 1 or 0

```
gpio -g write 18 0 - sets GPIO18 to a zero.
gpio -g write 18 1 - sets GPIO18 to a one.
```