IoT

Hardware Control using PHP

Sarwan Singh

Assistant Director(S)

NIELIT Chandigarh

Installing PHP and web server

- sudo apt-get install apache2 php5

or

Instead of Apache, you can install a more lightweight server like lighttpd

- sudo apt-get install lighttpd php5

Working thru Php code

- Executing applications with a PHP code can be done with two different functions: *exec* (for execute) and *system*.

system (string $command, int $return_var)

first parameter is the command to execute and the second one is the returned status of the executed command. The second parameter is optional.

Using system

```
<?php
   system ( "gpio mode 0 out" );
   system ( "gpio write 0 1" );
?>
```

Using exec

- it reads and stores what the command printed

```
<?php
   exec ( "gpio read 0", $status );
   print_r ( $status );
 ?>
```

exec (string $command, array $output, int $return_var)

the only difference with systems is the $output array, which will store the command's output

```
<?php
    $status = array ( 0 );
    //set pins mode to output
    for ($i = 0; $i <= 7; $i++ ) {
        system ( "gpio mode ".$i." out" );
    }
    //turns on the LEDs
    for ($i = 0; $i <= 7; $i++ ) {
        system ( "gpio write ".$i." 1" );
    }
```

```
    //reads and prints the LEDs status
    for ($i = 0; $i <= 7; $i++ ) {
        exec ( "gpio read ".$i, $status );
        echo ( $status[0] );
    }
    //waits 2 seconds
    sleep ( 2 );
    //turns off the LEDs
    for ($i = 0; $i <= 7; $i++ ) {
        system ( "gpio write ".$i." 0" );
}
?>
```

Building Bridges : shell_exec() function

- Easiest way to use PHP with Raspberry Pi
- shell_exec() can call Python scripts that perform certain tasks and control GPIO pins.
- Another approach is to deploy the Wiring Pi library for working with GPIO pins and then use the library with PHP via the shell_exec() function

Install Wiring Pi

- The library is not available as a binary package, so you need to compile and install it from the source
- Start with installing the Git software using:

```
sudo apt-get install git-core
```

- Then, clone the Wiring Pi Git repository by running

```
git clone git://git.drogon.net/wiringPi
```

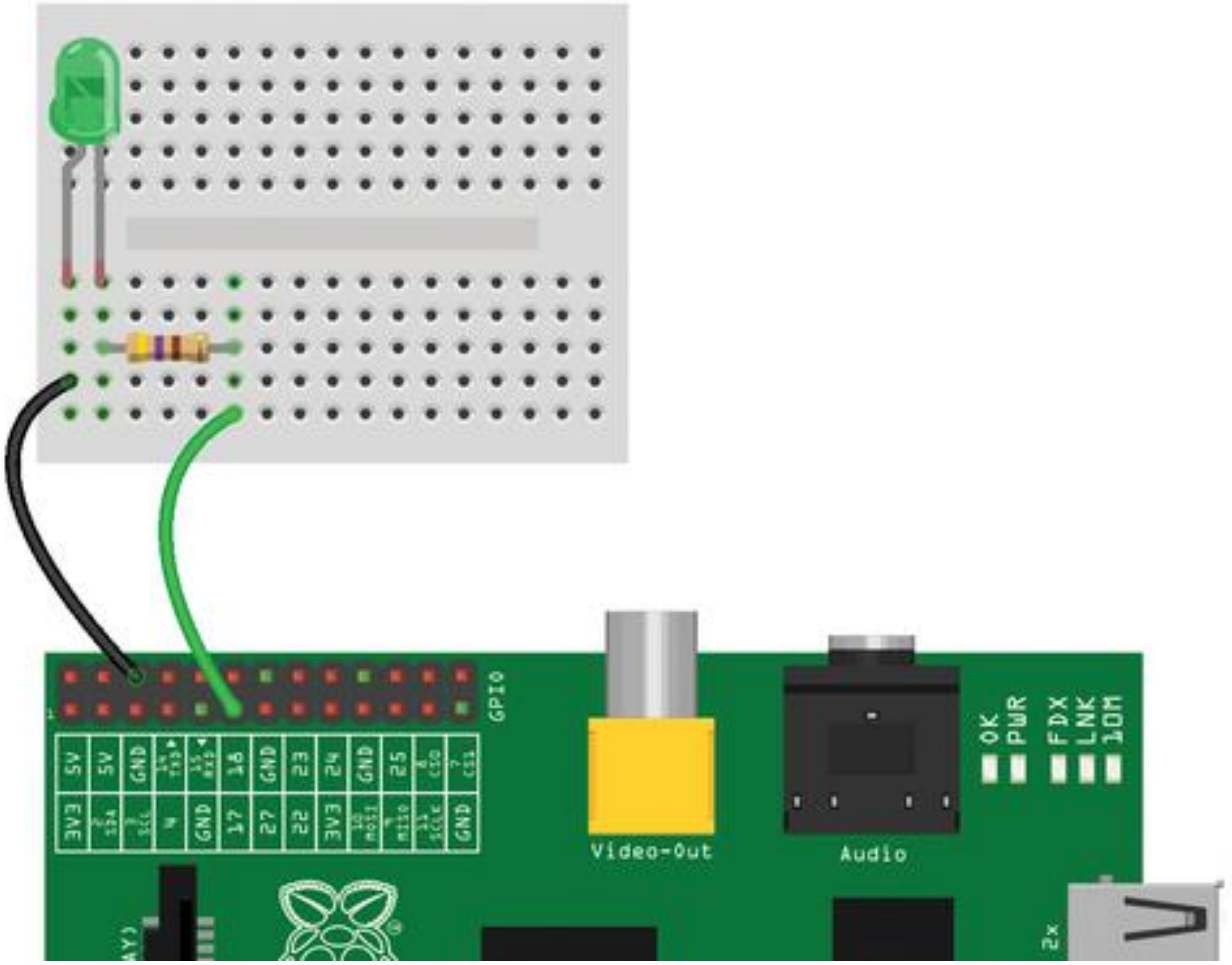
- Switch to the resulting wiringPi directory and use the ./build command to compile and install Wiring Pi:

```
cd wiringPi
./build
```

Install Wiring Pi

- Testing wiring Pi is installed and works properly, run the command gpio -v it should return current version
- gpio readall command to view a detailed GPIO layout diagram.

Circuit



Simple PHP App to Control an LED

```
01 <html>
02 <head>
03 <meta name="viewport" content="width=device-width" />
04 <title>LED Control</title>
05 </head>
06      <body>
07      LED Control:
08      <form method="get" action="gpio.php">
09          <input type="submit" value="ON" name="on">
10          <input type="submit" value="OFF" name="off">
11      </form>
```

Simple PHP App to Control an LED

```
12      <?php
13      $setmode17 = shell_exec("/usr/local/bin/gpio -g mode 17 out");
14      if(isset($_GET['on'])){
15          $gpio_on = shell_exec("/usr/local/bin/gpio -g write 17 1");
16          echo "LED is on";
17      }
18      else if(isset($_GET['off'])){
19          $gpio_off = shell_exec("/usr/local/bin/gpio -g write 17 0");
20          echo "LED is off";
21      }
22      ?>
23      </body>
24 </html>
```

http://127.0.0.1/gpio.php
http://localhost/gpio.php

- When you press one of the buttons, its value is passed as a part of the URL

(gpio.php?on=ON and gpio.php?off=OFF).

- GPIO pin 17 is controlled by PHP code that uses the shell_exec() function.

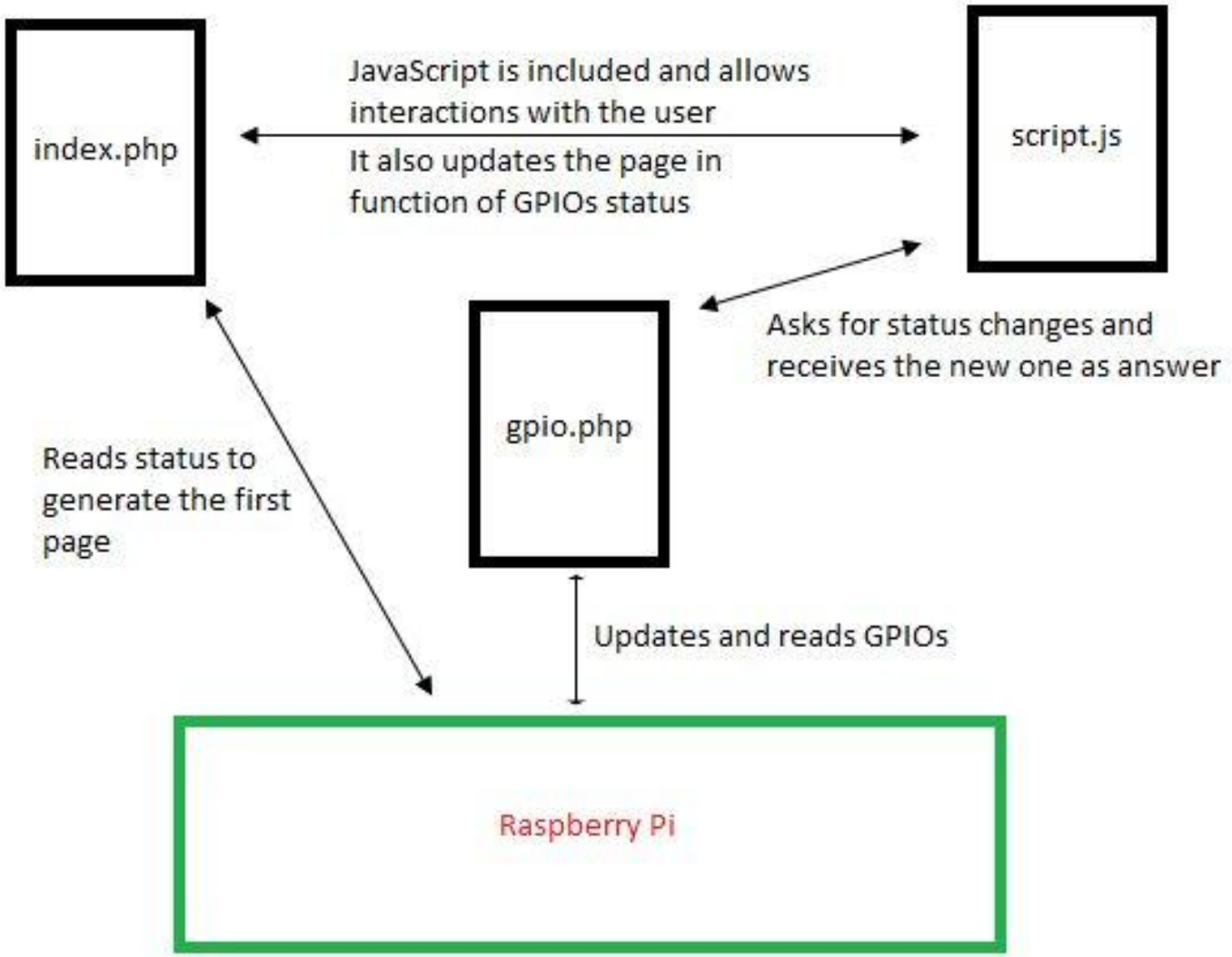
Problems

- If the buttons don't work, most likely the web server doesn't have appropriate rights to execute shell commands. To fix this, run the sudo visudo command and add the following line to the sudoers file:
- www-data ALL=NOPASSWD: ALLAlso, make sure that the /var/www directory belongs to the www-data user and group (use sudo chown -R www-data:www-data /var/www to set the correct owner).

Switch

Trigger switch:

```
08      <form method="get" action="switch.php">
09          <input type="submit" value="Trigger" name="switch">
10      </form>
11      <?php
12      $setmode17 = shell_exec("/usr/local/bin/gpio -g mode 17 out");
13      if(isset($_GET['switch'])){
14          $gpio_off = shell_exec("/usr/local/bin/gpio -g write 17 1");
15          sleep (0.5);
16          $gpio_on = shell_exec("/usr/local/bin/gpio -g write 17 0");
17          echo "Done!";
18      }
19      ?>
```


index.php
JavaScript is included and allows interactions with the user
It also updates the page in function of GPIOs status
script.js
gpio.php
Asks for status changes and receives the new one as answer
Reads status to generate the first page
Updates and reads GPIOs
Raspberry Pi

More Easy route thru PHP -gpio-php

- php-gpio project, provides a dedicated PHP library for accessing GPIO pin on Raspberry Pi.

- install the library and the accompanying files into the /home/pi directory

wget http://getcomposer.org/composer.phar

php composer.phar create-project --stability='dev'ronanguilloux/php-gpio

- Php script must contain the following code

```
require 'vendor/autoload.php';
use PhpGpio\Gpio;
$gpio = new GPIO();
$gpio->setup(17, "out");
```

- Php code to switch LED on/off

```
$gpio->output(17, 1)
$gpio->output(17, 0)
```

- $gpio->unexportAll() command resets all pins.

```
01 <?php
02 require 'vendor/autoload.php';
03 use PhpGpio\Gpio;
04 $gpio = new GPIO();
05 $gpio->setup(17, "out");
06 while (true){
07         $gpio->output(17, 1);
08         sleep(1);
09         $gpio->output(17, 0);
10         sleep(1);
11 }
```

- gpio-php cannot be used directly in PHP pages served by a web server.
- Soln :

```
shell_exec('sudo php path/to/php-gpio/\
blink_led.php > /dev/null 2> /dev/null &');
```

Calling python code from web page

```
<?php
   if (isset($_POST['on']))
   {  exec("sudo killall python");
      exec("sudo python /var/www/mystuff/ledON.py");
   }else if (isset($_POST['off'])){
      exec("sudo killall python");
      exec("sudo python /var/www/mystuff/ledOFF.py");
   }else if (isset($_POST['blink'])){
     exec("sudo python /var/www/mystuff/ledBLINK.py");
   }
?>
```

- http://www.instructables.com/id/Simple-and-intuitive-web-interface-for-your-Raspbe/step5/Making-the-interface/