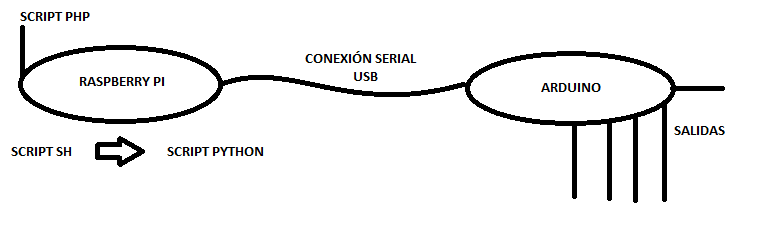
**9.5 CONTROL DE LUZ, VENTILACIÓN Y MOTOR POR SERIAL**

Después de haber efectuado pruebas y haber perfeccionado la transmisión de órdenes por serial, he intentado centralizar todas las conexiones de cableado en torno a la Arduino, que hará de interpretador de ordenes emitidas por la Raspberry.

Para lograr esta tarea, tendremos que comunicarnos con la arduino mediante serial por el cable usb. Según la orden mandada a la arduino, esta ejecutará una u otra tarea:



Mediante una página web accederemos al script php que nos recogerá la acción a ejecutar, el php ejecutará el script sh correspondiente, y ese a su vez ejecutará el script Python de comunicación con la arduino, pasándole el parámetro establecido. La arduino recepciona la orden y lleva a cabo la tarea correspondiente.

**PARTE SOFTWARE**

A continuación muestro el árbol de scripts utilizados:

**Scripts web**

/var/www/raspton 🡪 apagar.php boton\_arriba.jpg luces.php

boton\_abajo.jpg botonreset.jpg luzapagada.bmp stop.png

botonapagado.jpg controladores.php luzencendida.bmp

**Scripts del sistema**

/home/pi/gpioscripts 🡪 arduino.py

/home/pi/gpioscripts**/luces=** arduino\_exterior.sh arduino\_salon.sh estado\_garaje exterior.sh prueba.sh

arduino\_garaje.sh estado\_exterior estado\_salon garaje.sh salon.sh

/home/pi/gpioscripts**/motor=** arduino\_motor.sh motor.sh

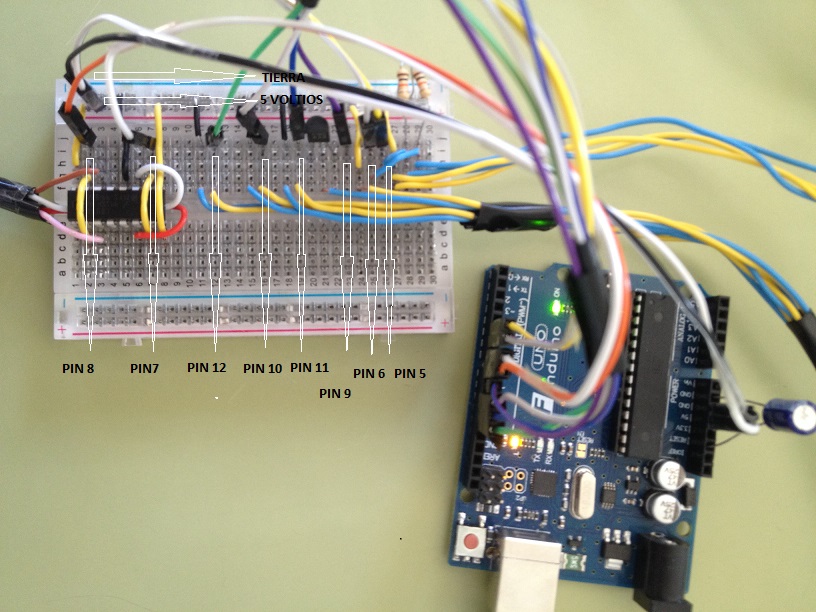
/home/pi/gpioscripts**/ventilación=** arduino\_ventilador.sh estado\_ventilador ventilador.sh

**Nota=** Los scripts del sistema que empiezan por arduino\_ son los que controlan el hardware mediante conexión serial, los demás lo controlan mediante los pin gpio.

**Script en arduino**

Este script estará cargado en la arduino: raspton\_arduino

**PARTE HARDWARE**



Pin 8 🡪Motor dirección arriba

Pin 7 🡪 Motor dirección abajo

Pin 12 🡪 luz salón

Pin 10 🡪 Luz garaje

Pin 11 🡪 Luz exterior

Pin 9 🡪 Salida ventilador

Pin 6 🡪 Final de carrera up

Pin 5 🡪 Final de carrera down

**FUNCIONAMIENTO**

El funcionamiento es prácticamente igual que el del punto 9.2, pero en esta ocasión hemos incluido el control del motor. Para mandarle ordenes a la arduino utilizaremos el script /home/pi/gpioscripts/arduino.py.

Su uso es el siguiente: Python /home/pi/gpioscripts/arduino.py “parámetro”

Los diferentes parámetros pueden ser:

H12🡪 encender luz salón.

L12 🡪 apagar luz salón.

H11 🡪 Encender luz exterior.

L11 🡪 Apagar luz exterior.

H10 🡪 Encender luz garaje.

L10 🡪 Apagar luz garaje.

H09 🡪 Encender ventilador.

L09 🡪 Apagar ventilador.

MUP 🡪 Abrir puerta.

MDO 🡪 Cerrar puerta.

MOF 🡪 Parar puerta.

Sabiendo esto y de forma idéntica al documento 9.2, creamos diferentes script sh descritos al comienzo del documento, que no harán más que ejecutar el script Python mandando el argumento que corresponda a la funcionalidad que precisemos.

Por ejemplo el script arduino\_salon.sh:

*#!/bin/bash*

*if [ "$1" = "encender" ]*

*then*

*python /home/pi/gpioscripts/arduino.py* ***H12***

*echo 1 > /home/pi/gpioscripts/luces/estado\_salon*

*elif [ "$1" = "apagar" ]*

*then*

*python /home/pi/gpioscripts/arduino.py* ***L12***

*echo 0 > /home/pi/gpioscripts/luces/estado\_salon*

*fi*

Siguiendo la estructura de este script, cuando ejecutamos **sh /home/pi/gpioscripts/luces/arduino\_salon.sh encender** , la arduino activará el pin 12 como salida y se encenderá la luz, en el caso de poner apagar, la luz dejará de estar encendida. También se actualiza el estado del pin mediante un archivo llamado estado\_salon en el si se ha escrito 1, está encendido y 0 si está apagado, esto nos servirá para mostrar el estado de los pines mediante php.

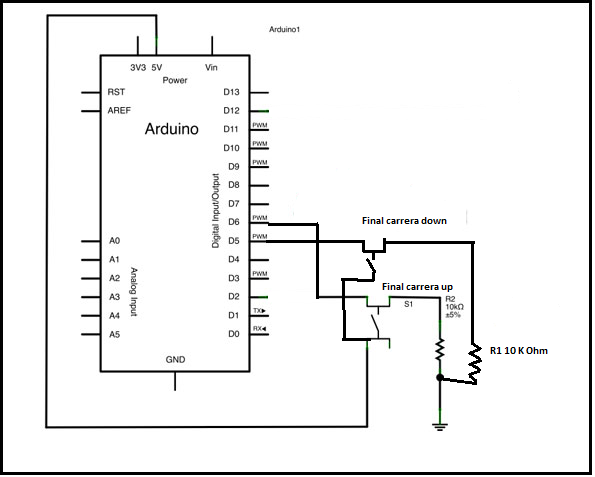
Ya por último la interfaz web:

 La pantalla principal de controles corresponde al script 🡪controladores.php

Como vimos en el documento 9.2, cada botón comprueba el fichero de estado del pin, si se encuentra el cero muestra el botón apagado, si por el contrario se encuentra en 1 muestra el botón encendido, a su vez cuando se pulsa el botón ejecuta los script pertinentes con la función exec(); Por ejemplo para el botón salón, se ejecutará exec(“**sh /home/pi/gpioscripts/luces/arduino\_salon.sh encender**”); el parámetro variará según el estado del pin, si se encuentra encendido el parámetro será apagar y viceversa.

Los finales de carrera son unos interruptores que se pulsan cuando el recorrido de la puerta tanto para arriba como para abajo llegan a su final, en este caso el interruptor deja pasar la señal, que llega a la arduino y apagar el motor.

En este esquema de ejemplo podemos ver cómo están montado los finales de carrera, en este proyecto se utilizan los pines 6 y 5 como entrada y el voltaje que utilizamos es el proveniente del pin de 5 v.



Fuentes: <http://www.arduteka.com/2011/11/tutorial-0002-controlando-entradas-y-salidas-digitales/>