

# RASPBERRY PI, CONTROL DE INTERFACES MEDIANTE APLICACIONES WEB.

Medina Alfonso\*, Rodríguez Juan, Maldonado Christian,  
Díaz Arnoldo, Quintero-Rosas Verónica  
Instituto Tecnológico de Mexicali, México  
Dpto. De Sistemas, ITMexicali, México  
{veroquintero@yandex.ru,  
adiaz@itmexicali.edu.mx }

**Abstract.** Deseando demostrar la capacidad del dispositivo Raspberry pi en el entorno de desarrollo de aplicaciones, así como aprovechar su capacidad para controlar distintos dispositivos de hardware pudimos desarrollar aplicaciones web las cuales manipulan hardware de manera remota con la implementación de este dispositivo como intermediario de control entre la aplicación y el hardware.

**Keywords:** Raspberry pi, aplicaciones web, manipulacion de hadware, control.

## I Introducción

Raspberry Pi [3] un ordenador de placa reducida con un costo muy bajo, desarrollado en Reino Unido por la fundación Raspberry Pi, con el objetivo de estimular la enseñanza de ciencias de la computación en las escuelas. Raspberry Pi puede interpretar lenguajes de programación comunes como C, Python, PHP, Java y además de poder gestionar bases de datos, dispone también de un puerto de expansión de entradas y salidas de propósito general conocido como GPIO (General Purpose Input/Output) y que se accede a él, a través de un conector de cabecera rotulado en la placa.

## 2 Proyectos realizados

### 2.1Cubo de Leds

Matriz de leds de 3 dimensiones Fig. 1, el cual por medio de un algoritmo se controla encendido y apagado de cada uno de los leds del cubo, en cada posición se otorgaron diversas secuencias de encendido creando efectos visuales muy llamativos, esto lo desarrollamos en lenguaje C [2].



**Fig. 1:** Cubo de Leds



**Fig. 2:** Seguidor de línea

Su pseudocódigo es el siguiente.

```
//polarizamos las 3 capas
digitalWrite (9, 0) ;
digitalWrite (10, 0) ;
digitalWrite (11, 0) ;
//ciclo para definir la secuencia de las columnas
int x=0;
for( x ; x < 9 ; x++ )
{
digitalWrite (x, 1) ; //encendemos el pin x
delay(1000);
digitalWrite (x, 0) ; // y apagamos
```

## 2.2 Seguidor línea negra.

El seguidor de línea es un robot formado con sensores, motores y un cerebro, es aquí donde implementamos el uso de Raspberry Pi (como cerebro), el robot [4] sigue una línea negra en un fondo blanco, dicha línea es detectada a través de dos sensores de infrarrojos, los cuales son colocados sobre la línea negra esto con el fin de detectar cuál de los dos sensores fue el que salió de la línea y así corregir la dirección del robot. El funcionamiento es sencillo; si el sensor de la derecha sale de la línea, cambia la dirección hacia la izquierda y viceversa; en cambio si ambos sensores están en la línea, la dirección sería recta Fig. 2.

## 2.3 Encendido de una lámpara de 60 watts a distancia.

El proyecto consiste en lograr encender una lámpara remotamente en un ambiente web usando como servidor [5] la Raspberry, el cual a su vez se encarga de encender una lámpara en el momento en que el cliente lo solicite a través de una página web. Desarrollamos dos aplicaciones en lenguaje c una de ellas pone en alto unos de los pines de Raspberry[6] y la otra lo pone en bajo a ese mismo pin, se podría decir que una de ellas enciende la lámpara mientras la otra aplicación la apaga. La aplicación pone en alto el pin 0 pero el voltaje que este provee es insuficiente para encender una lámpara de 30w, para esto se utilizó un relevador el cual en nuestro caso se encarga de bloquear un voltaje de 120v hasta que reciba una señal de 5v

```
//librerías utilizadas
#include<stdio.h>
#include<WiringPi.h>
main()
{
if (wiringPiSetup () == -1) return 1 ;
pinMode (0, OUTPUT) ;
digitalWrite (0, 1) ; // pin 0 de GPIO ALTO
return 0 ;
} Encender.c
```

La interfaz con el usuario es de lo más simple, al encontrar a la dirección en el navegador podremos observar el estado del foco con un imagen sencilla de un foco encendido, al tocar la pantalla utilizando las propiedades touch de nuestro Smartphone podremos cambiar el estado del foco de encendido a apagado y de apagado a encendido respectivamente, cambiando entre una vista con un foco apagado a una vista con un foco encendido de una manera muy amigable.



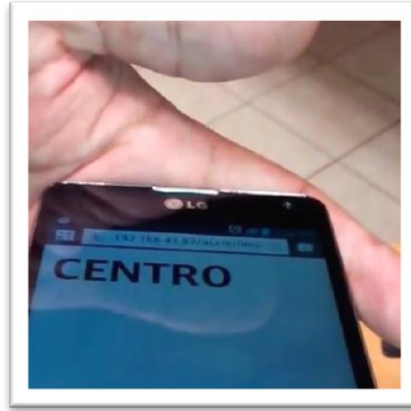
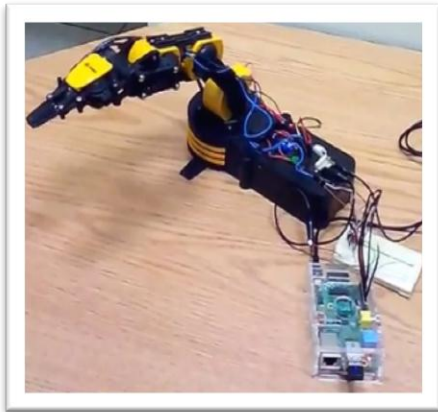
Fig. 3: Vista “on.php”



Fig. 4: Vista “off.php”

## 2.4 Control de brazo robótico, mediante el acelerómetro de un celular.

Al momento de ejecutar la aplicación con un número como parámetro, la aplicación se encarga de que Raspberry Pi active uno de los cuatro motores del brazo robótico a controlar [5]. Cada uno de los pines esta físicamente conectado a un puente H.



**Fig. 5:** Brazo robótico conectado a Raspberry Pi. **Fig. 6:** Dispositivo móvil ejecutando la aplicación web.

Con esto lo que se tiene es, una aplicación que dependiendo del parámetro que reciba activara un motor, para interpretar el acelerómetro: se desarrolló una aplicación web donde se utilizó Raspberry pi como servidor instalando apache y PHP.

```
include<stdio.h>
include<WritingPi.h> //librería de writingPi utilizar puerto GPIO
main(parametro){
if (parametro==1) //arriba
digitalwrite(pin1, 1) //pin1 en alto
if (parametro==2) //abajo
digitalwrite(pin2, 1) //pin2 en alto
if (parametro==3) //derecha
digitalwrite(pin3, 1) //pin3 en alto
if (parametro==4) //izquierda
digitalwrite(pin4, 1) //pin4 en alto
if (parametro==0) // causa que se detenga el brazo
{ digitalwrite(pin1, 0)
digitalwrite(pin2, 0)
digitalwrite(pin3, 0)
digitalwrite(pin4, 0) }
```

De esta forma se logró controlar a distancia con el acelerómetro de cualquier celular o con cualquier dispositivo, ya sea celular o tableta que cuente con acelerómetro y sin importar el sistema operativo (android, IOS, BlackBerry) mientras este cuente con explorador web. Las herramientas de desarrollo web utilizadas fueron PHP, JQuery y Ajax. JQuery [3] nos permitió obtener los datos del acelerómetro del dispositivo que accediera a la aplicación web, y por medio de un script en JavaScript logramos establecer condiciones en los valores de los ejes x, y.

```
$(window).bind("devicemotion", function(e){
    var movitBaby = e.originalEvent,
        accelera = movitBaby.accelerationIncludingGravity,
        //valores de acelerometro
        x = accelera.x,
        y = accelera.y;
        direccion(x,y); //método que manda las variables a un php
    mediante ajax
});
```

El script que diseñamos se encarga de que cuando el dispositivo este en una posición deseada mande una variable x con la dirección de movimiento mediante Ajax se ejecuta en el servidor un documento PHP el cual a su vez se encarga de ejecutar y mandar los parámetros a la aplicación anteriormente desarrollada en c.

### 3. Conclusiones

La creciente popularidad de Raspberry Pi, supone la eliminación definitiva del gigantesco paso que existe entre una buena idea y un prototipo. Raspberry es un dispositivo capaz de controlar, automatizar y definir sistemas de todo tipo, incluyendo los nuevos sistemas cyberfísicos.

Raspberry Pi es el nuevo horizonte para la materialización de lo que antes solo existiría en proyectos en el mundo, cuenta con potencial para cambiar la forma en la que nos relacionamos con la electrónica de consumo. Se ha comenzado a plantear antes de comprar el nuevo “gadget” que se encuentra actualmente de moda, que si tal vez podríamos crearlo nosotros mismos con nuestras necesidades personalizadas.

[1] Charles Bell. Beginning Sensor Networks with Arduino and Raspberry Pi. Apress; Edition: 1 (21 de noviembre de 2013).

[2] Eben Upton and Gareth Halfacree. Raspberry Pi User Guide. Wiley John + Sons (1 de Septiembre de 2012).

[3] Brendan Horan. Practical Raspberry Pi. APRESS; Edición: 2013. (27 de marzo de 2013).

[4] Chala, G., & Xavier, D. (2014). DISEÑO Y CONSTRUCCIÓN DE UN SISTEMA DE CONTROL DE LUCES PARA UNA CASA UTILIZANDO LA PLACA RASPBERRY (Doctoral dissertation).

[5] K. Dennis, Andrew. Raspberry Pi Home Automation with Arduino. Packt Publishing (5 de febrero de 2013).

[6] Dr. Simon Monk. Raspberry pi Cookbook. O'Reilly Media; Edición: 1 (10 de enero de 2014)