

ESCUELA DE INGENIERÍA
MINERA E INDUSTRIAL
DE ALMADÉN



TRABAJO FIN DE GRADO

**CONTROL REMOTO DE
ILUMINACION CON
MICROCONTROLADORES Y
SERVICIO WEB**

UNIVERSIDAD DE CASTILLA-LA MANCHA
GRADO EN INGENIERÍA ELÉCTRICA

Autor: Rafael Calatayud González
Director: Dr. Javier A. Albusac Jiménez

2014/2015

AGRADECIMIENTOS Y DEDICATORIA

Me gustaría empezar este apartado de agradecimientos y dedicatorias, dándole las gracias a mi director de proyecto, al Doctor D. Javier Alonso Albusac Jiménez, por ofrecerme la posibilidad de hacer este trabajo fin de grado y por su confianza, colaboración y apoyo en mi proceso de realización puesto que no habría podido terminar el trabajo sin su ayuda. También agradecer al personal de la Escuela de Ingeniería Minera e Industrial de Almadén por el apoyo mostrado e interés de sus conocimientos para resolver cualquier duda durante la realización de este trabajo fin de grado.

Por último, agradecer a mi familia y amigos, por estar ahí en cada momento ayudándome y confiando en mí siempre. Cabe destacar la figura de mi novia Sonia, mi hermano Javier y mis padres Rafael y M^a del Carmen por apoyarme y aguantarme durante toda la carrera.

Dedicar este trabajo para mí es muy importante aquellas personas que me han apoyado en todo. Primero a mi padre por darme sus consejos y a mi madre por confiar en mí. A mi hermano por apoyarme en todo momento. También dedico este proyecto a mi novia por apoyarme y animarme en todo momento y soportarme durante la realización del mismo. A mi abuela Rosa por estar siempre cerca de mí y mis abuelos que ya no están pero siempre están ahí.

Muchas gracias a todos por hacer posible la realización de este proyecto.

RESUMEN

El objetivo de este proyecto está basado en domótica, ya que consiste en el diseño y construcción de un sistema de control de iluminación por internet.

Para ello se ha empleado Raspberry y los componentes principales para diseñar una casa domótica. Una vez construido el sistema se hizo necesario modelar su comportamiento mediante un lenguaje de programación, como es Python (lenguaje empleado en la programación Raspberry).

La comunicación entre la central de gestión (Raspberry) y PC (control inalámbrico mediante un navegador web) se hizo por un servidor sin necesidad de conectar ambos mediante cables. Para que la comunicación sea posible, es necesario programar las librerías de comunicación entre el servidor y Raspberry.

El comportamiento del sistema domótico permite la posibilidad de controlar las luces mediante dos formas, una sería mediante control remoto por la central de gestión que actúa sobre el módulo de relés y una segunda posibilidad por control manual de las llaves de luz; optando a un control simultáneo, además la central de gestión reconocerá el estado de la luz monitorizando los sensores de corriente utilizados en cada una de las aulas a controlar, dándole al usuario la posibilidad de elegir una decisión sobre el apagado o encendido de las luces sabiendo el estado de las luces.

El uso de estos sistemas domóticos puede ser de gran utilidad en tareas de ahorro energético y confort o lugares donde se evita, o es costosa o es peligrosa la presencia humana: instalaciones eléctricas, nucleares/radiactivas, instalaciones remotas con condiciones climáticas extremas, instalaciones submarinas.

Además, la realización de este proyecto abre una puerta interesante en el caso de una posible adaptación en la E.I.M.I.A, ya que todo el material desarrollado podrá ser empleado para un futuro para su adaptación o desarrollo. De esta forma, no necesitarán invertir tiempo en solucionar los aspectos básicos de control del sistema y podrán centrarse en las posibles mejoras.

Palabras clave: Domótica, Control Inalámbrico, Sensorización, Órdenes, Soporte de Comunicación.

ABSTRACT

The objective of this project is based on home automation, it consists on the design and construction of a wireless control system for a lighting installation controlled by the internet.

It has been done using Raspberry and the main hardware components to design a home automation system. After its assembly and using a programming language such as Python (language used in programming Raspberry), proceed to perform the computer modeling.

The communication between the central management unit (Raspberry) and PC (wireless control through a web browser) was carried out by a wireless server. In order to make possible the communication, it was necessary to program the libraries of communication between server and Raspberry.

The home automation system allows the possibility to control lighting with two forms: one of them would be remote controlled by the central management unit that acts on the relays module and the other one possibility would be manual controlled by light switches; enabling simultaneous control. Furthermore, the central management unit will recognize the status of each lighting device by monitoring current sensors used in each classroom to control, giving to the user the opportunity to make a decision of turn on/off each lightning device knowing its status.

The use of these automated systems can be useful in tasks of energy savings and comfort or in places where it is not able the human presence (for instance: remote places, places with extreme climate conditions...).

In addition, this project shows the potential opportunity of implementing this automation system in the E. I. M. I. A, due to all the developed material could be used for their implementation, adaptation or development in a really near future.

In this way, it will not be necessary to spend time making adjustment in basic aspects of the control system and could focus on making all possible improvements.

Key words: Home Automation, Wireless Control, Sensoring, Orders and Communication Support.

Contenido

RESUMEN	3
ABSTRACT	4
ILUSTRACIONES	8
TABLAS	9
1 INTRODUCCIÓN	11
2 OBJETIVOS	14
3.0 ESTADO DEL CONOCIMIENTO	17
3.1 Historia	17
3.2 Tipos de Microcontroladores	19
3.2.1 Arduino	19
3.2.3Vía Apc	20
3.2.2 BeagleBone Black	21
3.2.3 ODRUID-U3	21
3.2.4 Odroid-Xu+E	22
3.2.5 Nanode	22
3.2.6 HackBerry A10	23
3.2.7 A13-OLinuXino	23
3.2.7 Libelium Waspmote	24
3.3 Raspberry Pi	25
3.3.1 Hardware Raspberry	26
3.3.2 Especificaciones Técnicas Raspberry pi modelos A, B y B+	29
3.3.3 Usos principales que puedes darle en casa a una Raspberry pi	29
3.3.4 Software	30
3.3.4.1 Linux	33
3.3.4.2 La Terminal y la GUI	34
3.4 Sensores	42
3.4.1 Sensor convertidor analógico-digital	42
3.4.2 Sensor de corriente	43
3.4.3 Relés	46
3.4.3.1 Componentes	47
3.5 Comunicación Inalámbrica	50
3.5.1 Control Remoto Raspberry a PC: Mediante Ssh y Putty	50

3.5.2 SSh.....	50
3.5.3 Putty.....	51
3.5.4 Control Remoto VNC Server en Raspberry Pi con PC o Móvil.....	52
3.6 Lamp.....	54
3.6.1 Control Apache.....	54
3.6.2Lenguaje programación PHP.....	56
3.6.3 Control MYQSL.....	57
3.7 Seguridad Raspberry.....	59
3.8 Diseño de Circuitos con Fritzing.....	61
4 CONTROL DOMOTICO CON RASPBERRY B+.....	64
4.1 Introducción.....	64
4.1.1 Principales ventajas del proyecto.....	65
4.2 Arquitectura General del Sistema.....	66
4.2.1 Central de gestión.....	66
4.2.2 Control de activación/desactivación de luz.....	67
4.2.3 Detección de corriente.....	69
4.2.4 Convertidor de la señal analógica-digital del sensor de corriente.....	73
4.2.4.1 Convertidor analógico-digital Script en Python.....	77
4.2.5 Placa extensión de la central de gestión Pi Cobbler +.....	80
4.3 Conexión de los todos los componentes.....	81
4.3.1 Conexión de lámparas con relé e interruptor.....	83
4.3.2 Conexión Raspberry b+ con relés.....	84
4.3.3 Conexión Raspberry b+ con convertidor analógico-digital MCP3008.....	86
4.3.4 Conexión Raspberry b+ con sensor analógico y sensor de corriente.....	88
4.4 Instalación y Configuración de Raspbian.....	90
4.4.1 Configuración Raspbian.....	92
4.4.1.1 Expand root partition to fill SD card.....	93
4.4.1.2. Set keyboard layout.....	93
4.4.1.3. Change password for 'pi' user.....	93
4.4.1.4. Set locale y Set timezone.....	94
4.4.1.5 Configurar la pila TCP/IP manualmente.....	94
4.4.1.6. Actualizar sistema RASPBIAN.....	95
4.5 Raspberry como Servidor Web.....	96
4.5.1 Instalación de la Pila LAMP.....	97

CONTROL REMOTO DE ILUMINACION CON MICROCONTROLADORES Y SERVICIO WEB

4.5.2 Conexión ssh	99
4.6 Entorno Gráfico con VNC para Conexión Remota	103
4.6.1 Instalación y configuración Real VNC.....	103
4.7 Conociendo el GPIO de Raspberry Pi	104
4.7.1 Controlando GPIO de la Raspberry Pi con Python	106
4.7.2 Accediendo al GPIO de Raspberry Pi desde PHP.....	107
4.8 Creación Página web.....	110
5 ESTUDIO ECONÓMICO	115
6 CONCLUSIONES	121
7 BIBLIOGRAFÍA.....	125

ILUSTRACIONES

Ilustración 1: Ordenador BBC Micro	17
Ilustración 2: Microprocesador Arduino Uno	19
Ilustración 3: Microprocesador BeagleBone Black	21
Ilustración 4: Microprocesador Odroid-U3	21
Ilustración 5: Microprocesador Odroid-Xu+E	22
Ilustración 6: Microprocesador Nanode	22
Ilustración 7: Microprocesador HackBerry A10	23
Ilustración 8: Microprocesador A13-OLinuXino.....	23
Ilustración 9: Microprocesador Libelium Waspote.....	24
Ilustración 10: Logotipo Raspberry	25
Ilustración 11: Microprocesador Raspberry Pi A	26
Ilustración 12: Microprocesador Raspberry Pi B.....	26
Ilustración 13: Microprocesador Raspberry Pi A+	27
Ilustración 14: Microprocesador Raspberry Pi 2 B.....	27
Ilustración 15: Microprocesador Raspberry Pi B+.....	28
Ilustración 16: Logotipo OpenElec y Xbmc.....	30
Ilustración 17: Pantalla Inicio Raspberry.....	31
Ilustración 18: ADC MCP3008	42
Ilustración 19: Sensor de corriente ACS710.....	43
Ilustración 20: Sensor de corriente ACS712.....	44
Ilustración 21: Sensor corriente ACS 714.....	45
Ilustración 22: Placa de 4 Relés KEYES	47
Ilustración 23: Componentes de la placa de Relés	47
Ilustración 24: Modulo de relés alimentado por USB.....	49
Ilustración 25: Logotipo SSh.....	50
Ilustración 26: Pantalla configuración PUTYY	51
Ilustración 27: Logotipo REALVNC	52
Ilustración 28: Logotipo Pila LAMP	55
Ilustración 29: Pantalla de Inicio Frizting	61
Ilustración 30: Pantalla Protoboard Frizting	62
Ilustración 31: Pantalla Esquemas PCB Frizting	62
Ilustración 32: Partes de Raspberry B+	66
Ilustración 33: Placa de Relés KEYE.....	67
Ilustración 34: Vista inferior placa de relés KEYES.....	67
Ilustración 35: Sensor corriente ACS714.....	69
Ilustración 36: ADC MCP3008	73
Ilustración 37: Entradas y Salidas MCP3008	74
Ilustración 38: Placa de Extensión Pi Cobbler + Raspberry B+.....	80
Ilustración 39: Montaje del proyecto completo	81
Ilustración 40: Conmutada Interruptor y rele con Sensor de corriente ACS714	83
Ilustración 41: Montaje placa de relé con Raspberry	85
Ilustración 42: Montaje PCB placa de relé con Raspberry	85

Ilustración 43: Montaje ADC MCP3008 y Pi Cobbler +	87
Ilustración 44: Montaje PCB de Pi Cobbler + y ADC MCP 3008	87
Ilustración 45: Montaje Pi Cobbler + y ADC MCP 3008 con sensor ACS714	88
Ilustración 46: Montaje PCB de Pi Cobbler + y ADC MCP 3008 con sensor ACS714.....	89
Ilustración 47: Pagina Oficial Sistema Operativo Raspbian	90
Ilustración 48: Otros sistemas operativos para Raspberry	91
Ilustración 49: Primer Inicio en Raspberry	91
Ilustración 50: Menú de configuración de Raspberry	92
Ilustración 51: Configuración del teclado en Raspberry	93
Ilustración 52: Configuración del teclado en Raspberry	94
Ilustración 53: Configuración IP en Raspberry.....	95
Ilustración 54: Logotipo Pila LAMP	97
Ilustración 55: Instalación Apache en Raspberry.....	98
Ilustración 56: Configuración de PUTYY.....	99
Ilustración 57: Otorgamiento de permisos para PUTTY.....	100
Ilustración 58: Usuario y Contraseña de Raspberry para control remoto	100
Ilustración 59: Pantalla de la Terminal en Raspberry	101
Ilustración 60: Imagen del servidor Apache en funcionamiento	102
Ilustración 61: Acceso mediante VNC por Ip.....	103
Ilustración 62: Imagen del control remoto desde VNC.....	103
Ilustración 63: Puertos GPIO Raspberry B+ [7]	104
Ilustración 65: Programa para el control de luces mediante navegador WEB	109
Ilustración 66: Imagen de la web del control de luces.....	112
Ilustración 67: Maqueta iluminada por accionamiento web.....	113
Ilustración 68: Maqueta de la instalación de control remoto.	113

TABLAS

Tabla 1: Especificaciones Técnicas Raspberry pi modelos A, B y B+.....	29
Tabla 2: Glosario de Linux [4]	36
Tabla 3: Modelos Sensor de Corriente ACS.....	70
Tabla 4: Características de Funcionamiento ACS714	70
Tabla 5: Características del aislamiento ACS.....	71
Tabla 6: Esquema de Instalación.	71
Tabla 7: Terminales ACS	72
Tabla 8: Características Eléctricas MCP3008	75
Tabla 9: Especificaciones Eléctricas MCP3008	76
Tabla 10: Características de Temperatura MCP3008	76

CAPÍTULO 1: INTRODUCCIÓN

1 INTRODUCCIÓN

Desde mediados del siglo XX se han organizado varias exhibiciones para enseñarnos ideas de cómo las casas aparentarían y cómo funcionarían en un futuro lejano. La gente se imaginaba cómo se podría hacer más cómoda la estancia en casa, cómo se facilitarían las tareas domésticas, etc. Después de la aparición de dispositivos electrónicos inteligentes fue surgiendo el concepto de la automatización del hogar: la domótica.

Se denomina **domótica** al conjunto de sistemas electrónicos capaces de automatizar un hogar, aportando servicios de gestión energética, seguridad, bienestar y comunicación. Además tienen que estar integrados por medio de redes cableadas o inalámbricas, y cuyo control puede actuar, desde dentro y fuera del hogar.

Se podría definir como la *integración de la tecnología en un entorno cerrado para el diseño inteligente* [26].

El término *domótica* viene de la unión de las palabras *domo* (derivado de domus, que significa *casa* en latín) y *tica* (de *automática*, palabra en griego, ‘que funciona por sí sola’).

Los primeros dispositivos de automatización aparecieron en edificios de prueba a los comienzos de la década del 70. Pero realmente fue en la década del ‘80 cuando los sistemas integrados aparecieron a nivel comercial, para más tarde desarrollarse en aspectos domésticos de las casas urbanas. Es aquí donde empieza una nueva era en la domótica y su verdadero desarrollo al conseguir integrar dos sistemas (eléctrico y electrónico) en pos de la comunicación integral de los dispositivos del hogar.

El auge de la informática permite incorporar en el sistema eléctrico de los edificios el Sistema de Cableado de la Central de Gestión, que facilita la conexión de terminales y redes. Así, estos edificios reciben el nombre de “inteligentes”, por su automatismo al servicio del propietario, a poder transmitir datos por líneas de baja tensión. La domótica implantada desde hace más de treinta años, se ha desarrollado a gran escala desde que se crearon las redes informáticas de comunicación, ya sea por sistema cableado o vía Wi-Fi.

En la actualidad existen diversas ofertas para servicios de domótica, con nuevos protocolos que permiten adecuar unos parámetros únicos para cada usuario que en un principio era impensado. Al requerir una baja tasa de envío de datos, es en la actualidad

uno de los protocolos más requeridos para las casas “inteligentes”, ya sea en sensores de movimiento, detectores de humo y otras funciones de seguridad en el hogar.

Con la domótica aplicada a la automatización hogareña se mejora en seguridad, confort y ahorro energético, aspectos muy observados por los poseedores de estos sistemas.

Por lo tanto pueden ayudar a mejorar el confort del ser humano y el ahorro energético que es uno de los problemas más importantes en la actualidad como es el caso de la EIMIA de la Universidad de Castilla-la Mancha donde los problemas de gasto energético de luces pueden verse reducidos, mejorando así el ahorro energético en las instalaciones de iluminación en dicha universidad.

El principal problema existente en la EIMIA es el desgaste energético en la luz sin un posible control sobre ella, este problema se debe a la usencia de docentes en las aulas de la universidad una vez terminada la jornada de impartición de clases diaria, produciendo un desgaste energético y económico de la iluminación hasta el cierre de la escuela por el personal de conserjería.

Para evitar este consumo innecesario de energía se va a utilizar la domótica como solución.

Los elementos principales necesarios para domotizar un hogar son una central de gestión, sensores o detectores, actuadores y un soporte de comunicación

En los últimos años el mercado de ofertas se ha extendido, permitiendo encontrar diversas variantes de equipos domésticos de integración domótica como los micro controladores existentes en los últimos años como es el caso de Raspberry, el cual es apta para actuar como central de gestión, donde interactuará con los sensores o actuadores integrados a él; a través de un control remoto y un soporte de comunicación que monitorice los datos obtenidos de los sensores, por ello esta sigue siendo la tecnología más utilizada dentro de la domótica.

El objetivo de este proyecto es domotizar las luces de la EIMIA mediante un control remoto inalámbrico mediante un navegador web. Donde la central de gestión obtendrá los datos obtenidos por el sensor de corriente para la monitorización del estado actual paso/ausencia de corriente en las diferentes fases de la instalación eléctrica de iluminación. Siendo controladas estas fases por una placa de relés por control remoto a través de un navegador web.

.

CAPÍTULO 2: OBJETIVOS

2 OBJETIVOS

La realización de este proyecto está orientada a la aplicación de las competencias generales de la titulación de Ingeniería Eléctrica en la construcción de un control de luces domótico con aplicación de lenguajes de programación a casos de ingeniería.

Se trata de **construir un control de luces mediante un servidor web con una interfaz gráfica en HTML**. La central de gestión “Raspberry” obtendrá una monitorización de los datos obtenidos del paso/ausencia de corriente correspondientes de los sensores de corriente además del control de la iluminación mediante una placa de relés de corriente continua adaptada a 220V. El usuario controlará mediante una página web la placa de relés además de obtener el estado de la iluminación, este sistema de control luces puede ser controlado desde cualquier lugar del mundo a través de un navegador web. Lo que implica que cualquier persona autorizada tendría un control sobre el sistema simplemente con el uso de un navegador web.

Como alternativa a este modo también ofrece la posibilidad de ser controlado mediante los interruptores o llaves de la instalación de iluminación, para también tener el uso manual del control de las luces.

Debido a que el control de luces es remoto de manera inalámbrica se deberá de disponer de una conexión a internet para dicho control y para saber el estado de las luminarias en cada momento.

Para cumplir los objetivos generales expuestos anteriormente es necesario satisfacer los siguientes puntos.

1. **Diseño y construcción de una maqueta con la instalación del control de luces:**

Se deberá elegir la placa con las características necesarias para el control de micro controlador y componentes que sean más económicos pero que a su vez cumpla con su objetivo sin tener un coste total muy elevado y asequible.

2. **Adaptar la maqueta a voltajes de 220:**

Se ha adaptado a voltajes correspondientes a viviendas (220v) para hacerlo aún más real.

3. Uso manual mediante interruptores y control mediante página web:

También dispone del uso de sus respectivas llaves o interruptores de la vivienda y el control también mediante web desde móvil o PC.

El poder disponer de un control manual y un control mediante una web de internet. Hace que este proyecto cumpla con varios objetivos como el ahorro energético e económico un desarrollo a las nuevas tecnologías y el poder adaptarlo a la vida cotidiana es todo un éxito.

4. Estado del servidor con respecto a los sensores y relés:

Raspberry pi que actúa como servidor y está programada para detectar la señal analógica que manda el sensor de corriente y mostrarla en la web diseñada.

Esta señal será analizada y se mostrará en forma de color el estado de la luz en dicha web con los botones de apagado y encendido que controlan las luces mediante una placa de relés. Por tanto el sistema debe de reconocer el estado de la luz en cada momento para el usuario tome que decisión elegir si apagar o encender las luces. Para ello el servidor debe de comunicar a la web el estado de tales luces y responder la decisión del usuario para la que ha sido programada.

Además todos componentes y programas utilizados en dicho proyecto han sido detallados para sus posibles trabajos o prácticas en el diferente campo que abarca o una posible adaptación en una instalación real como puede ser la EIMIA.

CAPÍTULO 3: ESTADO DEL CONOCIMIENTO

3.0 ESTADO DEL CONOCIMIENTO

3.1 Historia

La computadora u ordenador no es un invento de una persona en particular, sino el resultado evolutivo de ideas y realizaciones de muchas personas relacionadas con el ámbito de la informática, la electrónica, la mecánica, la lógica, el álgebra y programación.

Los ordenadores están presentes desde el primer matemático e ingeniero persa Musa al-Juarismi (780-850), que inventó el algoritmo.

En 1971 John Blankenbaker creó el Kenbak-1 el primer ordenador personal educacional de la historia, sin un procesador, solo con el uso de puertas lógicas. Donde solo se vendieron muy pocas unidades (40) en centros de enseñanza. Con el paso del tiempo los ordenadores domésticos son mucho más complejos donde la programación ya no es cosa de niños y ya su uso se realiza con personal cualificado en programación. Estos ordenadores están instalados en escuelas, para el uso de programas y servicios, dejando a un lado más la programación pura y dura [27].

En 2006, un ingeniero informático de hardware de la compañía de chips Broadcom diseño un miniordenador de bajo coste que volviese a inspirar a los niños en el uso del ordenador y en el aprendizaje de la programación, como sucedió por primera vez en los años 80. Este ingeniero unió a un grupo de programadores e informáticos para crear la Fundación **Raspberry Pi** [3].



Ilustración 1: Ordenador BBC Micro

Esta fundación haciendo homenaje al BBC Micro, cuyos modelos se llamaban BBC Micro A, B y B+.

Estos modelos fueron viendo la luz poco a poco mejorando cada vez a su antecesor y así cubrir las carencias que van apareciendo con el tiempo, manteniendo un precio muy equitativo en todos los modelos.

A principios del 2012 vio la luz la primera **Raspberry Pi Modelo B**.

Un año más tarde salió a la venta el **Modelo A**, aún más económico. En la actualidad se han vendido más de cuatro millones de Raspberry Pi, y se han lanzado tres nuevas versiones diferentes.

Aunque el objetivo fundamental de **Raspberry Pi** es la educación, no se trata de un ordenador para niños pequeños. Es un ordenador con miles de aplicaciones diferentes en campos avanzados como las matemáticas la robótica o las monedas criptográficas, y otras más, como construir una mini consola o un robot.

Raspberry Pi ha tenido una gran aceptación entre los jóvenes ya que sus dimensiones tan pequeñas como el tamaño de una tarjeta de débito y su bajo precio de 30€, con la capacidad de actuar como un miniordenador y con infinidad de usos prácticos.

3.2 Tipos de Microcontroladores

En toda casa domótica [25] se necesita una central de gestión que controle los componentes electrónicos necesarios en la instalación, dicho procesador debe de disponer de unas prestaciones que vayan en correlación a los sensores y actuadores que se vayan a utilizar.

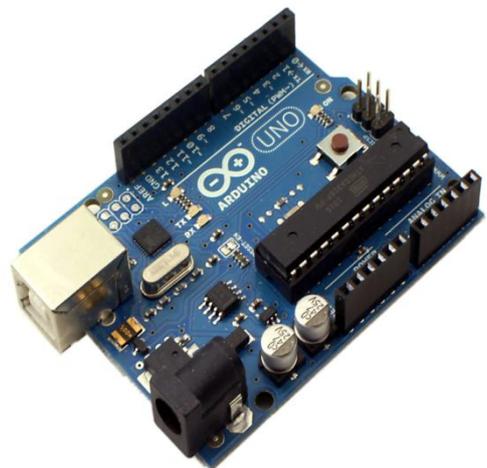
Para ello se requiere un micro procesador que controle sensores de corriente que monitoreen el estado de la luz y una placa de relés que será la encargada de realizar todas las operaciones programadas para el control del paso o corte de corriente en el sistema eléctrico de iluminación.

Entre todos los micro-controladores que existen en la actualidad en el cual existen diferentes modelos que podemos encontrar en el mercado europeo e internacional se optará por aquel que cumpla con todas las expectativas anteriormente citadas ya que cada micro-procesador ha sido diseñado con unas prestaciones y características diferentes, existiendo así un gran número de placas que pueden abarcar casi todos o todos los posibles proyectos como crear baterías, robots, servidores, detectores de humedad, presión, calor, luz, presencia etc.

Entre todas las posibles placas existentes están detalladas en este capítulo las más importantes, donde se elegirá aquella que cumpla con todas las expectativas.

3.2.1 Arduino

ARDUINO: es una plataforma de hardware libre, basada en una placa con un micro controlador y un entorno de desarrollo, diseñada para facilitar el uso de la electrónica en diferentes proyectos. El hardware de los diferentes modelos de placas Arduino están formados por un micro controlador Atmel AVR y puertos de entrada/salida analógica y digital. Estos micros controladores de bajo coste pueden permitir el desarrollo de múltiples diseños y proyectos. Otra parte fundamental es el software es una de las partes más importantes, que consiste en un entorno de desarrollo que implementa el lenguaje de programación Processing/Wiring y el cargador de arranque que es ejecutado en la



placa. En 2012 Arduino también empezó a usar micro controladoras CortexM3 de ARM de 32bits.

Por otro lado ARM y AVR no son compatibles entre sí a nivel binario, pero se pueden programar con el mismo IDE de Arduino [15]. La única diferencia clara en el micro controlador CortexM3 es que usan 3,3V, a diferencia de la mayoría de las placas AVR que usan 5V.

3.2.3 Via Apc

Esta placa ha sido diseñada pensando en Raspberry pi como podemos ver en la imagen. La mayoría de competidores de Raspberry Pi, buscan mejorar el Hardware sin excederse demasiado en el presupuesto, esto ha sido lo que ha marcado el paso a la hora de VIA desarrollar su propia placa.

Así nació esta tarjeta con un procesador **Wonder Media WM8750 ARM11** que consta de mono núcleo a 800Mhz, 512 MB de RAM, y 2GB de almacenamiento flash directamente integrado en la placa, además de poder permitirse instalar tarjetas de memoria extra (sd).

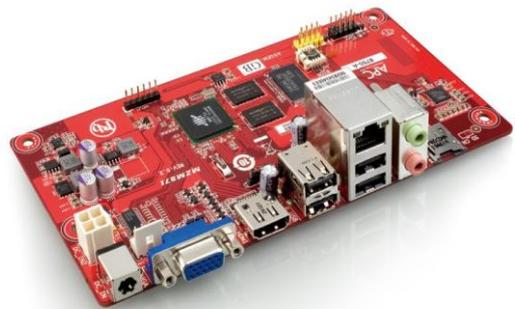


Ilustración 3: Microprocesador Via APC

VIA ha intentado mejorar a su competidor “Raspberry Pi” con bastante más opciones de conectividad, entre las que se incluyen un puerto USB extra para totalizar dos, puerto VGA y conectores de 3.5 mm para entrada y salida de audio, los mismos que siempre se han echado en falta dentro del Pi. Lamentablemente, a la hora de mejorar el hardware no han incluido un controlador para el WiFi de serie en la placa que intentarán mejorar en la siguiente versión.

3.2.2 BeagleBone Black

Es una alianza encargada de crear placas de desarrollo de código abierto cuyo principal contribuyente es **Texas Instruments**, podría considerarse más un precursor de Raspberry



Ilustración 3: Microprocesador BeagleBone Black

Pi que un competidor. Y es que ya había placas BeagleBoard [16] desde bastante antes que la Raspberry Pi llegara al mercado. Sin embargo, Raspberry fue la culpable de que Texas Instruments tuviera que mejorar sus placas ya que el presupuesto de una a otra era muy grande y solo les queda desarrollar algo más potente para hacerle frente en el mercado. BeagleBone Black [17] lo tiene todo para dejar en el terreno a Raspberry Pi.

El SoC tiene como corazón un procesador **AM335x Cortex-A8 a 1 GHz** firmado por la propia TI. El resto del conjunto es completado por 512 MB de RAM, 2GB de almacenamiento flash, puertos SD, HDMI, y el atractivo precio de 45€ que lo deja lo suficientemente cerca de su rival como para poder igualarlo en ventas alguno de estos días. Pero el nuevo modelo de Raspberry Pi 2 B volverá a torcer las expectativas de BeagleBone.

3.2.3 ODROID-U3

Se dedica a crear Hardware para desarrolladores, y la mayoría de sus productos son placas al estilo Raspberry Pi, como es el caso del **modelo U3** [18]. Esta pequeña placa muy económica con un precio de 65€ posee un gran chip **Exynos4412 Prime de Samsung**, con cuatro núcleos a 1.7GHz, a los que se suman 2 GB de RAM LPDDR2. Aunque de entrada la apuesta de ODROID nos puede parecer sustancialmente más atractiva más que la Raspberry Pi por tener cuatro veces su potencia con menos del doble del precio. Pero la realidad es que está bastante limitada como plataforma de desarrollo.



Ilustración 4: Microprocesador Odroid-U3

3.2.4 Odroid-Xu+E

Este modelo ha mejorado respecto al anterior con un mayor número de puertos de entrada/salida. Además muestra una gran potencia con 8 núcleos basados en la tecnología **big.LITTLE** de ARM, que son asistidos por 2 GB de memoria LPDDR3 RAM. Lamentablemente, dado su elevado precio, tiene algunos defectos como la falta de una unidad de almacenamiento flash o la oportunidad de poner la unidad para ampliarlo uno mismo, además de la falta también del módulo Wi-Fi.



Ilustración 5: Microprocesador Odroid-Xu+E

Por lo que Odroid-xu [18] sigue dejando muchos aspectos sin mejorar y por lo tanto un déficit de prestaciones que lo hacen ser un modelo muy simple para la competencia actual. Están diseñando nuevos modelos que saldrán a la luz próximamente intentando así cubrir y mejorar las necesidades de los usuarios.

3.2.5 Nanode

Nanode ha intentado evolucionar las placas, fue desarrollado por Ken Boak un Ingeniero Electrónico del Reino Unido del grupo Hackerspace. Nanode es una evolución de otras placas como es Arduino, que permite a este conectarse a Internet a través de un API aparte de utilizarlo como servidor de páginas web simples; al igual que Raspberry, donde el

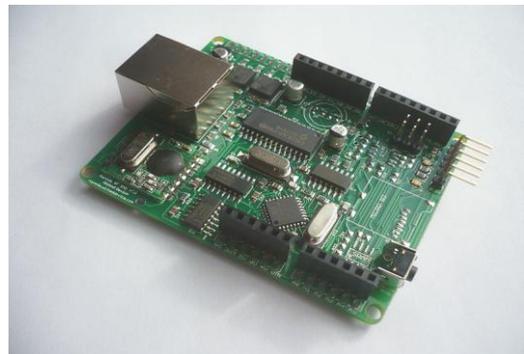


Ilustración 6: Microprocesador Nanode

usuario configura el dispositivo. A la hora de programar se utiliza el mismo entorno que aparte de que es abierto, estando disponible para que se pueda programar desde **cualquier sistema operativo (MAC, Linux y Windows)**.

Es la única diferencia clara que existe ella y Arduino es la diferencia de precio, vale menos con la misma calidad y misma funcionalidad.

3.2.6 HackBerry A10

Es el competidor con mejor precio que se presenta a sí mismo con un CPU ARM Cortex A8 con frecuencia de reloj de 1,2 GHz y 521 MB o 1 GB de RAM [19], el sistema es Android 4.0, lo que ha sido un error y no será la plataforma más *hackeable* (no haciendo honor al nombre), además de estar limitada en la entrada y salida de puertos.



Ilustración 7: Microprocesador HackBerry A10

3.2.7 A13-OLinuCino

Que con su procesador a 1 GHz y 512 MB de RAM [21] se pone a la altura del resto de competidores. Esta placa opta por un gran número de puertos estándar a los que se suman los medios de entrada/salida para desarrollo que en conjunto con su naturaleza de Linux. Por lo tanto es una de las mejores alternativas para los que estén en la búsqueda de la herramienta de desarrollo libre.

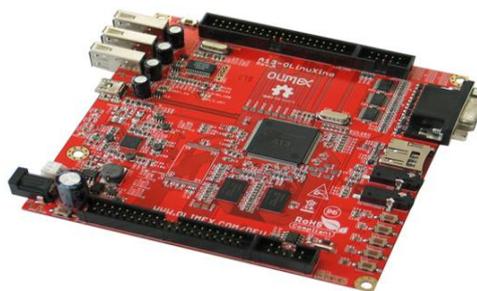


Ilustración 8: Microprocesador A13-OLinuCino

3.2.7 Libelium Wasmote

Este dispositivo fue diseñado para crear **redes inalámbricas de sensores** con unos requerimientos bastantes específicos y destinados a ser desplegados en un escenario real. Los anteriores dispositivos o placas se pueden considerar para aplicar en pequeñas maquetas y poder uno mismo desarrollar diferentes proyectos y sobre todo aprender a utilizar la electrónica. La mayoría de estas microcontroladores se utilizan en escuelas y universidades para el desarrollo a pequeña escala de redes o robots. Este dispositivo lo comercializa una empresa española reconocida como empresa innovadora por diferentes instituciones dentro de la Administración Española. En lo que si se parecen, es que Wasmote [20] y Arduino [15] utilizan el mismo entorno de desarrollo y el código que se desarrolla para Arduino se puede usar también en este dispositivo, simplemente ajustando pequeñas ajustes de configuración del sistema. El precio no está disponible.



Ilustración 9: Microprocesador Libelium Wasmote

3.3 Raspberry Pi

Raspberry Pi es un pequeño ordenador de placa reducida, fue creada y desarrollada en Reino Unido por la Fundación Raspberry Pi, con el objetivo de estimular la enseñanza de ciencias de la computación en las escuelas de primaria y secundaria de todas las escuelas de Europa debido a su bajo precio y sus grandes prestaciones.

La fundación Raspberry da soporte para las actualizaciones y descargas de las distribuciones para arquitectura ARM, Raspbian (derivada de Debian), RISC OS 5, Arch Linux ARM (derivado de Arch Linux) y Pidora (derivado de Fedora); y promueve principalmente el aprendizaje del

lenguaje de programación Python [7] y otros lenguajes también soportados como son Tiny BASIC, C++, Perl y Ruby

En 2006, los primeros diseños creados de Raspberry Pi se basaban en un pequeño micro controlador AVR Atmel ATmega644 con 4 a 256 KB de memoria flash programable, encapsulados de 28 a 100 pines.

En mayo de 2009, la Fundación Raspberry Pi se creó en Caldecote, Reino Unido donde Eben Upton era el administrador de la fundación, se reunió con un pequeño grupo de profesores académicos e ingenieros de la informática para crear un pequeño ordenador con la que puedan los niños a aprender e interactuar con la informática como se logró en 1981 con el primer ordenador Acorn BBC Micro.

El primer diseño basado en ARM se consiguió montar en un pequeño módulo del tamaño de una memoria USB, donde también disponía de un puerto USB y un puerto HDMI.

El 4 de febrero de 2013, salió a la venta el modelo A, pero debido a temas políticos los principales proveedores sólo lo pudieron poner a la venta en Europa ese día. Y este año acaban de sacar al mercado Raspberry Pi 2 B. Un modelo con grandes prestaciones.

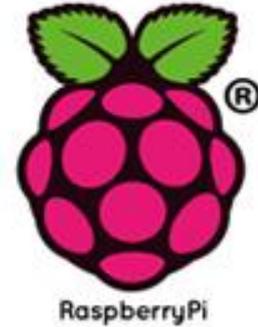


Ilustración 10: Logotipo Raspberry

3.3.1 Hardware Raspberry

Raspberry Pi A

Se trata de la segunda versión y la más básica. Con unas dimensiones de 85.6 x 56.5 mm, tiene un consumo de 1.5W y pesa alrededor de 45 gramos. Cuesta 25€.



Ilustración 11: Microprocesador Raspberry Pi A

Utiliza un chip BCM2835 de Broadcom con una CPU ARM a 700 MHz, 256 MB de memoria RAM, un procesador gráfico a 250 MHz, con conexiones HDMI, vídeo RCA, salida de auriculares 3.5 mm, un puerto USB, DSI, conector de cámara CSI, conector de tarjeta SD hasta 8Gb y 8 conectores GPIO. Las conexiones GPIO se pueden usar en diferentes interfaces para insertar diversos periféricos o sensores. El procesador gráfico es compatible con OpenGL 2.0 con una resolución de video de 1080p.

Raspberry Pi B

Fue el primer modelo que salió al mercado. Utiliza el mismo hardware que el modelo A, pero la diferencia es que esta incluye 512 MB de memoria RAM, así como un segundo puerto USB. Además, de conexión de red Ethernet 10/100 que permite conectar la Raspberry Pi a Internet e incorporarlo wifi mediante un USB externo. Las dimensiones y el peso son idénticas también al Modelo A (85.6 x 56.5 mm y 45 gramos), aunque el consumo si es mayor, 3.5 W. El precio cuesta alrededor de 30€.



Ilustración 12: Microprocesador Raspberry Pi B

Raspberry Pi A+

Este modelo, que vio la luz meses después del modelo A, también dispone del mismo hardware que el modelo A y su única diferencia es mayor número de pines GPIO [12] con un total de 17 pines, dispone de soporte de tarjetas microSD, un sistema de audio mejorado, además de unas dimensiones y consumo más pequeño. Mide 65 x 56.5 mm, y consume



Ilustración 13: Microprocesador Raspberry Pi A+

tan sólo 1W y pesa solamente 23 gramos. Su precio ronda los 20€.

Raspberry Pi 2 B

Broadcom ha lanzado un nuevo modelo con un potente chip que sustituye al actual BCM2835, utilizado en todos los modelos anteriores. El nuevo chip BCM2836 adquiere la misma arquitectura, pero añade una nueva CPU ARM Cortex-A7 de cuatro núcleos a 900 MHz, y 1 GB de memoria RAM a 450 MHz. Según sus creadores, es mucho más potente que los modelos anteriores, posibilitando la ejecución del sistema operativo Windows 10 en un futuro. El resto de hardware es el mismo que el modelo B+.



Ilustración 14: Microprocesador Raspberry Pi 2 B

También se ha anunciado la versión Raspberry Pi 2 A, pero no saldrá hasta finales de 2015. Y aún no está disponible la imagen de este modelo. El precio de este modelo será el mismo que modelos anteriores, alrededor de 35€.

Raspberry Pi Modelo B+

Las mejoras de esta placa fueron saliendo una vez iniciada su venta donde la aceleración por hardware para la codificación de vídeo se hizo disponible a partir del 24 de agosto de 2012, cuando la fundación Raspberry informó que la licencia sería de uso gratuito. Además de soportar CEC para poder ser controlado mediante un mando a distancia. También se optó por sacar a la venta el uso de codificación-descodificación de Microsoft vc1.

Meses después se anunció una mejora de la placa, que ofrecía un pequeño número de correcciones y mejoras, como unos agujeros de montaje, un circuito para hacer reset, soporte para depuración JTAG, etc.

La fundación anunció que todos los Raspberry Pi Modelo B [22] serían enviados a partir de ese momento con 512 MB de memoria RAM en vez de 256 MB.

En la actualidad consta de 512 MB con una posible mejora a 1 GB como es el caso del nuevo modelo Raspberry pi 2.



Ilustración 15: Microprocesador Raspberry Pi B+

3.3.2 Especificaciones Técnicas Raspberry pi modelos A, B y B+.

	Modelo A	Modelo B	Modelo B+
SoC:	Broadcom BCM2835 (CPU + GPU + DSP + SDRAM + puerto USB)		
CPU:	ARM 1176JZF-S a 700 MHz (familia ARM11)		
Juego de instrucciones:	RISC de 32 bits		
GPU:	Broadcom VideoCore IV, OpenGL ES 2.0, MPEG-2 y VC-1 (con licencia), 1080p30 H.264/MPEG-4 AVC		
Memoria (SDRAM):	256 MiB (compartidos octubre de 2012 con la GPU) el 15 de 512 MB (compartidos con la GPU) desde		
Puertos USB 2.0:	1	2 (vía hub USB integrado)	4
Entradas de vídeo:	Conector MIPI CSI que permite instalar un módulo de cámara desarrollado por la RPF		
Salidas de vídeo:	Conector RCA (PAL y NTSC), HDMI (rev1.3 y 1.4) ,Interfaz DSI para panel LCD		
Salidas de audio:	Conector de 3.5 mm, HDMI		
Almacenamiento integrado:	SD / MMC / ranura para SDIO	MicroSD	
Conectividad de red:	Ninguna	10/100 Ethernet (RJ-45) vía hub USB	
Periféricos de bajo nivel:	8 x GPIO, SPI, I ² C, UART		
Reloj en tiempo real:	Ninguno		
Consumo energético:	500 mA, (2.5 W)	700 mA, (3.5 W)	600 mA, (3.0 W)
Fuente de alimentación:	5 V vía Micro USB o GPIO header		
Dimensiones:	85.60mm × 53.98mm (3.370 × 2.125 inch)		

Tabla 1: Especificaciones Técnicas Raspberry pi modelos A, B y B+.

3.3.3 Usos principales que puedes darle en casa a una Raspberry pi

- Reproductor multimedia: Utilizarlo como centro multimedia para ver películas en el televisor hasta una calidad de 4K y formatos diferentes.
- Vigilancia: Conectarlo a una webcam wifi y sensores de movimiento y emplearlo como sistema de vigilancia.
- Control remoto de luces: Control de luces de una vivienda a distancia.
- Ordenador: Usarlo como PC de una vivienda.
- Servidor: Puede actuar como servidor para transferencias de archivos.
- Emulador de videoconsolas antiguas o máquinas recreativas.
- Dron volador.

3.3.4 Software

Raspberry Pi al igual que las diferentes placas anteriormente citadas tienen que estar provistas con uno o varios programas, en concreto usa mayoritariamente sistemas operativos basados en el núcleo Linux, como puede ser Raspbian, una distribución derivada de Debian [4] que está totalmente adaptada y optimizada para el hardware de Raspberry Pi, esta se lanzó a mediados de 2012 y es el sistema recomendado por la compañía.

El Slackware ARM arranca sin ninguna modificación. Los 128MB a 496 MB de memoria RAM incorporadas en las Raspberry Pi, soportan los 64 MB de RAM necesarios para arrancar dicha distribución en los sistemas ARM y i386 sin el uso de una interfaz gráfica. Por otro parte, se están desarrollando distribuciones las ligeras y concretas como es el caso de IPfire (distribución para el uso de firewall) o OpenElec y Raspbmc (distribuciones con el centro multimedia de XBMC).



Ilustración 16: Logotipo OpenElec y Xbmc.

Para poder arrancar la imagen desde la tarjeta SD solo se puede acceder a ella mediante el firmware del código cerrado de la GPU, llamado blob binario, que es el encargado de arrancar la tarjeta SD para posteriormente continuar con su configuración inicial recomendada.

Este blob binario está vinculado al Software Linux que también es de código cerrado.

Para ellos las aplicaciones hacen peticiones a las librerías de tiempo de ejecución donde estas si son de código abierto, y estas hacen el acceso a unos drivers de código abierto en el Kernel de Linux. La API del driver del kernel es específica para estas librerías. También existen aplicaciones que usan vídeo y hacen uso de OpenMAX, las aplicaciones 3D usan OpenGL ES y las aplicaciones 2D usan OpenVG; OpenGL ES y OpenVG hacen uso de EGL y éste último, del driver de código abierto del kernel.

En 2012 se lanzó el primera imagen de tarjeta SD que almacenaba un sistema operativo de nos mas de 1GB de memoria y que podía ser arrancada en una tarjeta SD. Dicha imagen

se basaba en Debian 6.0 que incorporaba el escritorio LXDE y un navegador web denominado Midori, aparte de algunas herramientas de programación.

La imagen que funcionaba bajo QEMU permitiendo que el Raspberry Pi pudiera ser emulado en diferentes sistemas.

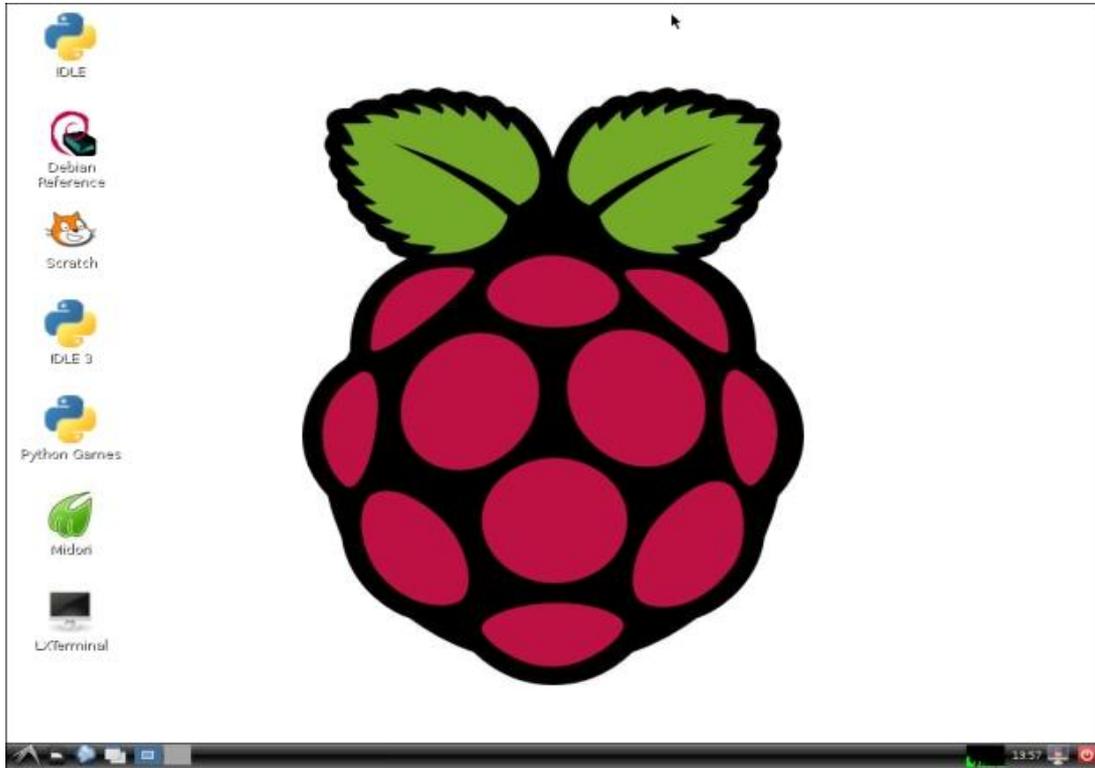


Ilustración 17: Pantalla Inicio Raspberry

El 8 de marzo de 2012, la fundación lanzó otro sistema operativo para Raspberry denominado Pi Fedora Remix [4] (actualmente llamada Pidora), era la distribución más recomendada por la fundación al ser desarrollada por una universidad prestigiosa de Canadá (Seneca). También se creó una tienda de aplicaciones para que la gente intercambiara programas o aplicaciones.

Seis meses después Alex Bradbury, director sobre el desarrollo Linux de la fundación, anunció que todo el código de la GPU Videocore para sistemas en ARM sería de código abierto, para poder disponer de este código estará disponible en un repositorio de la fundación en la web de GitHub.

El 5 de noviembre de 2012, Eben Upton fundador y ex administrador anunció a la comunidad el lanzamiento del sistema operativo RISC OS 5 para Raspberry Pi, pudiéndose descargar la imagen de forma gratuita desde la web de la fundación. Su vínculo con RISC OS se debe años atrás cuando se empezó a desarrollar una versión donde el mismo sistema

operativo incluiría dentro de su sistema una gran cantidad de programas y aplicaciones como es el caso de Maestro (editor de música), NetSurf (navegador web seguro), Packman (gestor de paquetes), StrongED (editor de texto), o una tienda de aplicaciones llamada Store donde se puede encontrar aplicaciones gratuitas o de pago. Además se puede implementar el sistema operativo con unos manuales en BASIC para su desarrollo. También se anunció el juego Minecraft: Pi Edition para Raspberry Pi, basado en la versión *Minecraft: Pocket Edition* para móviles y tablet. La descarga estaba disponible de forma oficial y gratuita por primera vez el 12 de febrero de 2013 desde página oficial del juego junto a las debidas instrucciones para ejecutarlo en Raspbian Wheezy [4]. Una de las características más importante de este lanzamiento es poder interaccionar con el juego mediante programación, con la intención de motivar y enseñar a los niños a aprender a programar.

El 25 de mayo de 2013, la fundación comunicó de que se estaba desarrollando una nueva versión del servidor gráfico para lograr mayor suavidad al usar la interfaz gráfica del escritorio de Raspberry, porque el procesamiento lo estaba realizando la GPU Videocore y no la CPU, impidiendo así en el renderizado 3D que era uno de los problemas más importantes sobre la interfaz gráfica.

Donde una semana después ya fue lanzada en la web de la fundación para su descarga la aplicación NOOBS (New Out of Box Software), que permitía facilitar la instalación de diferentes sistemas operativos para Raspberry Pi. NOOBS distribuye la imagen en su página oficial en forma de archivo Zip que se copia descomprimido directamente a una tarjeta SD de 4 GB o superior, y una vez arrancada la placa con su respectiva SD con la imagen, aparece un menú en que se da la opción de instalar una de las diferentes sistemas operativos o distribuciones en el espacio libre de la tarjeta SD de memoria.

Una vez iniciada es posible acceder al menú de distribuciones en el caso de instalar otro sistema.

Para ellos se entrará en menú apretando la tecla shift durante el arranque para reinstalar el sistema operativo o el poder elegir otro. Y la posibilidad de elegir diferentes distribuciones Linux de carácter general como Raspbian, Arch Linux ARM y Pidora; las distribuciones Linux para mediacenter con XBMC Openelec y RaspBMC; y el sistema operativo Risc OS5. En 2013, se implementó una nueva versión oficial de Oracle Java ARM.

Donde esta da soporte para el hardware, que ofrece la posibilidad de un mayor rendimiento que la versión OpenJDK ARM.

3.3.4.1 Linux

Es un proyecto de código abierto, que fue originalmente fundado para crear un núcleo que sería libre de utilizar por cualquier persona para cambiar o mejorar la distribución. El núcleo o “kernel” es el corazón del sistema operativo y se ocupa de la comunicación entre el usuario y el hardware.

Aunque el término “Linux” [23] es correctamente empleado para referirse al “núcleo” o “corazón”, es a menudo utilizado para referirse a una colección de diferentes proyectos de código abierto de distintas compañías. Esas colecciones de proyectos se unen para formar diferentes modelos de Linux, conocidas como distribuciones o “distros”.

La versión original de Linux fue combinada con una colección de herramientas creadas por un grupo denominado GNU. El sistema resultante fue conocido como GNU/Linux y era un sistema muy básico pero al mismo tiempo muy potente. A diferencia de los otros sistemas operativos de la época, como: múltiples cuentas de usuario, en donde varios usuarios podían compartir una misma computadora. Aquello fue algo que los otros sistemas operativos rivales de código cerrado tomaron abordo, tanto Windows como OS X hoy en día ya soportan múltiples cuentas de usuario sobre un mismo sistema. Esta característica continúa presente hoy en día en Linux y proporciona seguridad y protección al sistema operativo.

En Linux, pasará la mayor parte de su tiempo trabajando en su cuenta de usuario limitada o restringida. Esto no significa que se limiten las diferentes funciones en lo que puede hacer. En lugar de eso, trabajar con una cuenta de usuario limitada evita que accidentalmente estropee el software en la Raspberry Pi. También previene que los virus y otros malware infecten el sistema bloqueándoles el acceso a los archivos y directorios críticos del sistema. Antes de empezar, vamos a familiarizarse un poco con algunos de los términos y conceptos más importantes y utilizados en el mundo Linux, como se definen en la Tabla de más abajo.

3.3.4.2 La Terminal y la GUI

En todos los sistemas operativos como Windows y Mac OS, existen normalmente dos principales formas para ejecutar una determinada tarea: a través de la interfaz gráfica de usuario (GUI) y a través de la línea de comandos (conocida en la jerga Linux como “la consola” o “la terminal”).

Las distribuciones Linux pueden cambiar de apariencia y ser completamente diferentes en los aspectos de la imagen, dependiendo sobre el entorno de escritorio que utilicen. La distribución recomendada a utilizar es la Raspbian derivada de Debian, pero la mayoría de los comandos utilizados desde la terminal casi siempre son los mismos para todas las diferentes distribuciones existentes.

Linux está diseñado para ser lo más fácil e intuitivo posible para los nuevos usuarios, incluso, en la línea de comandos en la terminal. Aunque la forma más común de utilizar cada es mediante el comando “man”.

Este comando describirá la función de cada comando que se desee adquirir información. Cada aplicación Linux viene con un archivo de ayuda, conocido como “man page” (abreviatura en inglés para “página del manual”). Este archivo le proporciona un preámbulo sobre el software así como también los detalles sobre lo que hacen sus opciones y cómo emplearlas.

Para acceder a la página de manual de una determinada herramienta, simplemente escriba “man” seguido por el nombre del comando. Para ver la página del manual de ls (una herramienta que muestra un listado del contenido de los directorios) simplemente escriba `man ls`.

Tabla Glosario de Linux

Bash	El shell o intérprete de comandos más popular, utilizado en la mayoría de las distribuciones Linux.
Bootloader	Software responsable de cargar el núcleo Linux. El más común es GRUB.
La consola	Una versión de la terminal que siempre está disponible y es la primera imagen que ve en la Raspberry Pi.
Entorno de escritorio	Software que hace la GUI más bonita. GNOME y KDE son los entornos de escritorio muy populares.
Directorio	El término Linux para lo que Windows denomina como “carpetas” en donde los archivos se almacenan.
Distribución	Una versión particular de Linux. Fedora Remix, Arch y Debian son distribuciones.
Ejecutable	Un archivo que puede ejecutarse como un programa. Los archivos en Linux deben ser marcados a fin de que puedan ser ejecutados.
EXT2/3/4	El sistema de archivos EXTendido, el formato más común utilizado en Linux.
Sistema de archivos	La forma en que un disco duro o dispositivo de almacenamiento es formateado para quedar listo para almacenar archivos.
GNOME	. Uno de los entornos de escritorio Linux más comunes.
GNU	Un proyecto de software libre que proporciona muchas de las herramientas que se utilizan en las distribuciones Linux.
GRUB	El Gestor de Arranque Múltiple (GRan Unified Bootloader), creado por GNU y utilizado para cargar el núcleo Linux.
GUI	Una interfaz gráfica de usuario, con la que el usuario controla la computadora a través de un ratón o dispositivo táctil (touch).
KDE	Otro entorno de escritorio Linux muy popular.
Linux	Definido exactamente, es el núcleo utilizado por GNU/Linux. Definido popularmente, es un sistema operativo de código abierto.
Live CD (CD Vivo)	Una distribución Linux que se ofrece como un CD o DVD que no requiere de instalación.

Paquete	Una colección de archivos requeridos para la ejecución de una aplicación, comúnmente administrados por el gestor de paquetes.
Gestor de paquetes	Una herramienta que sigue la pista del software ya instalado y de las nuevas instalaciones.
Partición	Una sección de un disco duro que se encuentra lista para tener un sistema de archivos dedicado al almacenamiento de información.
Root	La cuenta de usuario principal en Linux, equivale a la cuenta de administrador en Windows.
Shell	Un intérprete de comandos basado en texto, cargado dentro de una terminal.
sudo	Un programa que permite a los usuarios limitados o restringidos ejecutar un comando como el usuario root.
SuperUsuario	Ver Root.
Terminal	Un intérprete de comandos basado en texto dentro del cual el usuario interactúa con un programa shell.
X11	El sistema de Ventanas X, es un paquete que proporciona una interfaz gráfica de usuario (GUI).

Tabla 2: Glosario de Linux [4]

Comandos Básicos de Linux

Aunque existen cientos de diferentes distribuciones Linux [23], todas ellas comparten un conjunto de herramientas en común. Esas herramientas son puestas en funcionamiento a través de la terminal y son análogas a herramientas similares presentes en Windows y OS X [28].

ls: Abreviación de “listing” (lista o listado), ls proporciona un lista de los contenidos del directorio actual. Opcionalmente puede ser invocado con un argumento que indica el directorio a ser listado. Como un ejemplo, al escribir ls /home se visualizará una lista de los contenidos del directorio /home, independientemente del directorio actual en el que se encuentre. El comando equivalente en Windows es dir.

cd: Las siglas para “change directory” (cambio de directorio), cd le permite navegar a través del sistema de archivos. Escribiendo únicamente cd lo colocará devuelta a su directorio inicial de partida. Si escribe el comando junto con el nombre del directorio al que desea moverse, por consecuencia, lo cambia hacia aquel directorio. Considere que los directorios pueden ser absolutos o relativos: cd boot lo llevará al directorio llamado boot bajo el directorio actual, pero cd /boot lo llevará directamente al directorio /boot desde cualquier lugar en donde usted se encuentre.

mv: El comando “move” (mover) tiene dos propósitos en Linux: permitir que un archivo sea trasladado de un directorio a otro, y también permitir que los archivos sean renombrados. Esta última característica puede parecer fuera de lugar, pero en términos de Linux, el archivo se ha movido de un nombre a otro. El comando es invocado en la siguiente forma: mv viejoarchivo nuevoarchivo.

rm: Abreviación para “remove” (remover), rm elimina archivos. Cualquier archivo (o lista de archivos) escrito después del nombre del comando será eliminado. Su equivalente en Windows es del, y los dos comandos comparten un mismo requisito, que es el cuidado que debe tomarse para asegurarse de eliminar el archivo correcto.

rmdir: Por regla general, el comando `rm` no puede eliminar directorios por sí mismo. Por consiguiente, el comando `rmdir` es proporcionado para borrar los directorios una vez que éstos han quedado vacíos de archivos debido al comando `rm`.

mkdir: Es el opuesto a `rmdir`. El comando `mkdir` crea nuevos directorios. Por ejemplo, escribir `mkdir mi carpeta` en la terminal, creará un nuevo directorio llamado `micarpeta` bajo el directorio actual de trabajo. Al igual que `cd`, los directorios que se proporcionen al comando pueden ser relativos o absolutos.

Paquete de accesorios del software Raspbian

Accesorios

Debian ref: Una guía de referencia integrada, ésta le ofrece una explicación detallada de la distribución Debian Linux y cómo los programadores pueden contribuir a su desarrollo.

Calculator: Una sencilla y práctica calculadora.

Gestor de archivos: El administrador de archivos `PCManFM` proporciona una interfaz gráfica para navegar en los archivos almacenados en la Raspberry Pi o en cualquier dispositivo de almacenamiento.

Leafpad: Este es un editor de texto muy sencillo, utilizado para crear rápidas anotaciones o escribir sencillos programas.

LXTerminal: La terminal de `LXDE` le permite utilizar la consola de línea de comandos dentro de una ventana sin tener que salir de la interfaz gráfica de usuario.

Terminal de =Root: Similar a `LXTerminal`, `Root Terminal` automáticamente lo inicia en la sesión con la cuenta de Super-usuario (`root`) para poder llevar a cabo tareas de mantenimiento en el sistema que son imposibles desde una cuenta de usuario normal.

Visor de Imágenes: `GPicView` le permite ver imágenes, tales como aquellas almacenadas en su cámara digital o en cualquier dispositivo de almacenamiento conectado.

Xarchiver: Si necesita crear o extraer archivos comprimidos, como los archivos Zip, ésta es la herramienta indicada.

Educación

Scratch: Un lenguaje de programación gráfico destinado a los niños. Aprenderá más sobre Scratch y sus capacidades en el Capítulo 10, “Una introducción a Scratch”.

Squeak: La plataforma sobre la cual Scratch se ejecuta. Raramente necesitará utilizar esta entrada del menú, y en su lugar deberá utilizar la entrada Scratch anterior.

Internet

Dillo: Un navegador web alternativo ligero y sencillo.

Midori: Navegador web rápido y ligero, es el equivalente a Internet Explorer en Windows o a Safari en OS X.

Midori Private Browsing (Midori Navegación Privada): Modo privado, por lo tanto los sitios que visite no serán guardados en el historial del navegador.

NetSuf Web Browser: Una alternativa a Midori, NetSuf puede tener mejor rendimiento en ciertos tipos de páginas web.

Wpa_gui: Una interfaz gráfica para la herramienta wpa_supplicant que le permite escoger la red ya configurada a la cual conectarse.

Programación

IDLE-Un entorno de desarrollo integrado (IDE) escrito solamente y específicamente para Python [10].

IDLE 3: IDLE configurado para utilizar el lenguaje de programación Python 3 más reciente, en vez del Python 2.7 por defecto. Ambos son compatibles entre sí, pero algunos programas pueden necesitar algunas características de Python 3. [6]

Scratch: Este acceso directo abre el lenguaje de programación educativo Scratch y es el mismo Scratch que se encuentra en la categoría Educación.

Squeak: Al igual que Scratch, este es un acceso directo en el que se encuentra en la categoría Educación.

Sonido y Video

Reproductor de Música: LXMusic es una interfaz simple y rápida para el software de reproducción de música.

Herramientas del Sistema

Administrador de tareas: Es una herramienta que le permite ver la cantidad de memoria libre disponible en la RasPi, Y la posibilidad de cerrar los programas que se hayan colgado o que de alguna forma no respondan.

Preferencias

Ajustes del monitor: La resolución en la que trabaja el monitor o televisor conectado a la Raspberry puede cambiarse, aunque las configuraciones avanzadas requerirán de la modificación de los archivos de configuración.

Aplicaciones preferidas: Una herramienta para cambiar las aplicaciones que abrirán determinados tipos de archivo una vez ejecutados.

Configuración de la sesión del escritorio: para la modificación del funcionamiento del sistema cuando el usuario inicia sesión, incluyendo qué programas serán cargados automáticamente y el tipo de gestor de ventanas que será utilizado.

Gestor de configuración de Openbox: La GUI LXDE utiliza un entorno de escritorio llamado Openbox, el cual puede configurarse desde aquí. Con esta herramienta, puede aplicar nuevos temas para cambiar la apariencia de la GUI, o alterar el funcionamiento de ciertos aspectos de la interfaz.

Personaliza apariencia y comportamiento: Un conjunto de herramientas para modificar la apariencia de la GUI, incluyendo el estilo y el color de las ventanas.

Teclado y Ratón: Una herramienta para configurar los dispositivos de entrada. Si el teclado está escribiendo caracteres incorrectos para ciertas teclas o el ratón es demasiado sensible, es aquí donde ese tipo de cosas pueden ser configurados.

3.4 Sensores

Los sensores son una de las partes o componentes del hardware más importantes del proyecto ya que sin ellos este trabajo no sería posible, son determinados para cada función ya que hay infinidad de sensores y funciones para cada uno de ellos [11].

Estos miden las características del ambiente, o sea, permiten medir la temperatura, distancia, posición, ruido, corriente; algunos casos se necesitarán que los sensores sean muy rápidos y que la precisión no sea muy importante para transmitir información y actuar rápidamente en cambio en otros casos será más importante la precisión que la rapidez, todo depende para que se pondrán sensores y de sus objetivos.

3.4.1 Sensor convertidor analógico-digital

Una de las carencias de Raspberry Pi son las entradas analógicas tal y como existen en otros dispositivos y placas como es Arduino, con lo cual muchos sensores analógicos no los podemos utilizar directamente. Para ello se precisa de un convertidor analógico-digital para poder dar uso a estos sensores tan necesarios para determinados proyectos.

El **MCP3008** es un convertidor de analógico-a-digital de 8 canales de 10 bits (ADC). Es barato, fácil de conectar y no requiere ningún componente adicional. Se utiliza el protocolo de **bus SPI**¹ que está soportado por el GPIO de la Raspberry Pi.

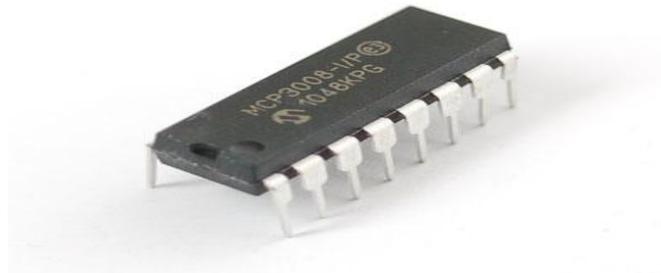


Ilustración 18: ADC MCP3008

¹ Bus SPI: (del inglés *Serial Peripheral Interface*) es un estándar de comunicaciones, usado principalmente para la transferencia de información entre circuitos integrados en equipos electrónicos.

3.4.2 Sensor de corriente

Hay diferentes tipos de sensores de corriente y de diferentes marcas. Entre esos tipos existen sensores con conexión Jack 3,5mm o con conexión pin o directamente con la protoboard.

La conexión pin será la que se realice en este proyecto por usar tantos sensores de corriente como queramos sin pasar de los pines GPIO [12] disponibles ya que Raspberry pi solo consta de una salida Jack al contrario de los pines que cuenta con 40.

Hay diferentes modelos de sensor de corriente con conexión pin según los Amperios a medir (10, 20 y 30 A). [24]

El chip acs710 consiste en un circuito integrado linear del sensor de pasillo de la precisión con una trayectoria de cobre de la conducción situada cerca de la superficie del dado del silicio. La corriente aplicada atraviesa la trayectoria de cobre de la conducción, y el voltaje de la salida análoga del sensor de pasillo sigue linear el campo magnético generado por la corriente aplicada. La exactitud del ACS710 se maximiza con esta configuración de empaquetado patentada porque el elemento de pasillo se sitúa en proximidad extremadamente cercana a la corriente que se medirá.

Características:

Medición desde -10 a 10 amperios.

Bajo nivel de ruido de las señales analógicas.

Tiempo de salida en respuesta de 5 μ s a la corriente de entrada.

1,5% de error de salida a 25°C.

Resistencia de 1.2mOhm.

Tensión mín de aislamiento 2.1 kVRMS de los pines de 1 4 a

los pines 5-8

5.0 VDC operación única.

66 a 185 mV / A sensibilidad de salida.

Salida de voltaje proporcional a la corriente alterna o continúa.

Fabricado con gran precisión.

Muy estable la salida de voltaje.

Ancho de banda del dispositivo 80 kHz se fija a través del pin FILT

Casi cero histéresis magnéticas.



Ilustración 19: Sensor de corriente ACS710

El chip ACS 712 es un sensor lineal de corriente ACS712 que integra de manera perfecta el efecto Hall. El sensor da una medida exacta a las señales AC y DC. Sus pistas gruesas permiten soportar hasta 5 veces las condiciones de sobre corriente dadas.

El ACS712 da una señal analógica de voltaje que varía linealmente con la corriente detectada. El dispositivo requiere 5 VDC para el pin marcado como VCC y un par de condensadores de filtro.

Características:

Medición desde -5 a 5 amperios.

Bajo nivel de ruido de las señales analógicas.

Tiempo de salida en respuesta de 5 μ s a la corriente de entrada.

1,5% de error de salida a 25°C.

Resistencia de 1.2mOhm.

Tensión mín de aislamiento 2.1 kVRMS de los pines de 1-4 a los pines 5-8

5.0 VDC operación única.

66 a 185 mV / A sensibilidad de salida.

Salida de voltaje proporcional a la corriente alterna o continúa.

Fabricado con gran precisión.

Muy estable la salida de voltaje.

Ancho de banda del dispositivo se fija a través del pin FILT.

Ancho de banda de 80 kHz.

Casi cero histéresis magnéticas.

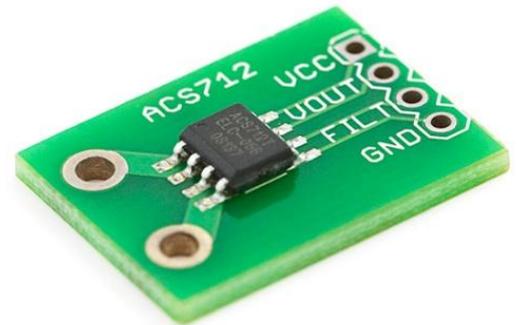


Ilustración 20: Sensor de corriente ACS712

El chip ACS714 es un sensor de corriente capaz de medir de forma lineal corrientes de hasta 30 amperios en ambos sentidos (+30A y -30A). Produce en su salida una tensión

lineal proporcional a la corriente que atraviesa sus pines de medición, lo que nos permite medir consumos o excesos de consumo en diferentes proyectos como en robótica cuando se utilizan grandes motores DC, en quadcopters o en circuitos que se necesite saber la corriente consumida por algún dispositivo o línea. Y este es el sensor que utilizaremos en dicho proyecto.

Características:

Medición desde -30 a 30 amperios.

Bajo nivel de ruido de las señales analógicas.

Tiempo de salida en respuesta de 5 μ s a la corriente de entrada.

1,5% de error de salida a 25°C.

Resistencia de 1.2mOhm.

Tensión mín de aislamiento 2.1 kVRMS de los pines de 1-4 a los pines 5-8

5.0 VDC operación única.

66 a 185 mV / A sensibilidad de salida.

Salida de voltaje proporcional a la corriente alterna o continúa.

Fabricado con gran precisión.

Muy estable la salida de voltaje.

Ancho de banda del dispositivo se fija a través del pin FILT.

Ancho de banda de 80 kHz.

Casi cero histéresis magnéticas.

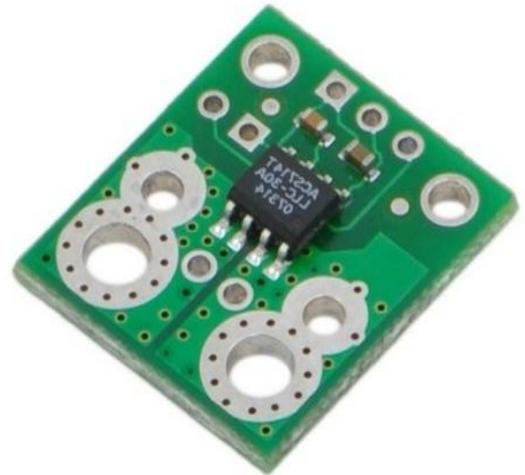


Ilustración 21: Sensor corriente ACS 714

3.4.3 Relés

El uso de relés supone controlar o interactuar con los aparatos eléctricos de todo tipo que operen con corriente alterna a **220 V**. Para el control de los aparatos eléctricos se necesita para ello unos relés que actúen de intermediarios entre Raspberry donde sus conexiones son de corriente continua, (**DC, Direct Current**) a **5 V** como máximo y los circuitos eléctricos que alimentan a lámparas y electrodomésticos que en general operan con corriente alterna, (**AC, Alternating Current**), a **220 V**, como es la parte alterna de los relés. Un relé es un dispositivo que trabaja a modo de interruptor, cerrando o abriendo un circuito eléctrico según la señal recibida en corriente continua con un pequeño voltaje para interactuar sobre el relé, en realidad no es más que un interruptor en el que el pulsador se ha sustituido por un electroimán que atrae una pieza metálica que abre o cierra el circuito según pase o no corriente por el electroimán. Cuando el electroimán es excitado por la corriente que circula por él, cambia de estado el interruptor, y cuando cesa la corriente vuelve a su estado original de reposo.

Lo más normal del relé es que este compuesto de tres posibles puntos de conexión a alto voltaje (220V), un punto denominado “c” o “com”, y dos puntos llamados NA, (normalmente abierto), y otra NC, (normalmente cerrado) que a la inversa permite el paso de la corriente alterna en tanto no haya voltaje en el circuito de corriente continua está actuando de forma NA (porque se mantendrá abierto en tanto no pase electricidad por el electroimán).

Las conexiones a realizar son normalmente conectar las salidas “C” y “NC,” para que actúen sincrónicamente el circuito de baja y de alta, si hay corriente en el circuito de baja (5V), lo hay en el de alta (220V) y si el de baja deja de tener corriente, se corta el circuito de alta, pero todo depende del usuario.

Relés hay de muchos modelos que pueden funcionar con distintos voltajes de entrada y salida así como entre dos corrientes continuas. Lo que caracteriza a un relé es; el voltaje adecuado que ha de tener la corriente que alimenta el electroimán y el máximo de amperios que puede llevar la corriente eléctrica que controla dicha placa de relés.

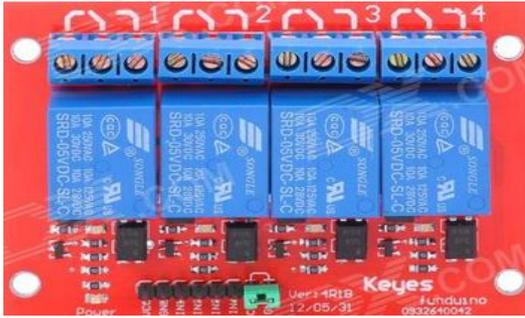


Ilustración 22: Placa de 4 Relés KEYES

En la imagen de la carcasa del relé, pone 5V y más abajo 10A /250V es decir que este relé funciona a 5 Voltios, por tanto puede utilizarse con Raspberry y controlar una red a 220 voltios y 10 amperios como máximo.

Raspberry suministra un máximo de **40 mA** por sus pines y un mínimo de **16 mA** por cada uno, pero para poder controlar con total seguridad la bobina de un relé de 5v se necesitan **80 mA**.

La solución recala entre la tarjeta **Raspberry** y el **relé**, en el cual un **transistor** actúa de amplificador de la señal, por lo que en definitiva será el transistor el que deje el paso de la corriente para el control de dicho relé. Se utiliza un **BD137** y un diodo de protección, en nuestro caso un **IN4001**, para eliminar la corriente parasita inducida que se produce al apagar la bobina y asegurar la Raspberry a sobretensiones.

3.4.3.1 Componentes

El relé está formado por los siguientes componentes ya integrados en él, ya que sin ellos habría que colocarlos en una placa casera y la fabricación del relé quizás no llegaría a sus mejores prestaciones.

Entre sus componentes vamos a detallar los más importantes.

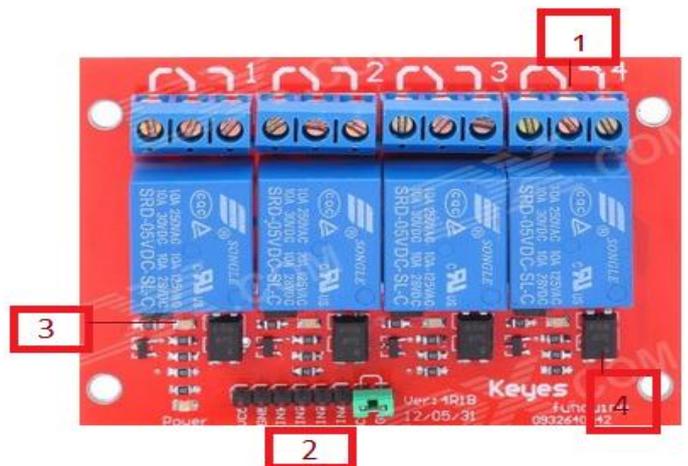


Ilustración 23: Componentes de la placa de Relés

1) Entradas y salidas: El diseño de los relés permite que trabajen con contactos **NA** (normalmente abiertos) o **NC** (normalmente cerrados). Y es en esta parte donde se realizan las conexiones de la fase con los interruptores en conmutada con el relé.

2) Pines: Esta placa de relé lleva incorporado los pines en corriente continua de VCC y GND con sus respectivos pines In1 al In4

3) Diodo: Un **diodo** solo permite la circulación de la corriente eléctrica a través de él **en una solo dirección**, lo que esto nos permite asegurar la Raspberry antes corrientes derivadas del relé cuando se produce la excitación de la bobina al actuar.

4) Transistor: Como siempre ocurre hay muchos modelos de transistores pero los que vamos a utilizar de momento con **Raspberry** son los pequeños chips, con tres largas patas llamadas **Colector, (C), Base, (B) y Emisor, (E)**.

El “**colector**”, (la C en el esquema), este se conecta a la salida del relé, que a su vez está conectado directamente al **PIN 5V** de la tarjeta.

Otra pata, el “**emisor**” (la E del esquema), se conecta a tierra, (**PIN GND**) mientras que la tercera pata la “**base**” se conecta a través de **una resistencia de 1000 ohm** a un **PIN digital de Raspberry**.

Por lo tanto el **relé** no abrirá la corriente eléctrica a 220 V y la bombilla permanecerá apagada. Cuando el **PIN** envíe **5 voltios**, circulara corriente entre las patas “b” y “E”, el relé entrara en funcionamiento, el circuito eléctrico se abrirá, permitiendo circular la corriente y lucirá la bombilla.

Para simplificar estos montajes sobre los componentes necesarios para fabricar un relé hay también tarjetas de expansión con relés que incorporan toda la circuitería necesaria para su correcto funcionamiento. Estas tarjetas pueden tener uno o varios relés, pero normalmente todos son iguales y actúan independientemente, por tanto si tienen cuatro relés como el de la figura por ejemplo permiten que una misma tarjeta Raspberry controle hasta cuatro circuitos o líneas diferentes conectando cuatro PINES diferentes a los puntos adecuados.

La tarjeta que se muestra en imagen es la que se utilizará en el proyecto y permite controlar cuatro circuitos eléctricos diferentes mediante una sola tarjeta Raspberry.

El esquema de la tarjeta para cuatro relés se puede ver en la figura de abajo.

El relé está formado normalmente de la entrada de 5 Volts y la de tierra (0V) suele ser común para todos los relés por tanto como se puede apreciar en la tarjeta con cuatro relés que presentamos solo hay cuatro entradas de continua la marcada como VCC, (5 voltios),

CONTROL REMOTO DE ILUMINACION CON MICROCONTROLADORES Y SERVICIO WEB

GND, (tierra), y cuatro entradas una por cada relé que conectaría con un PIN de Raspberry. Hay otras dos tarjetas protegidas por un puente de plástico verde que están señaladas como GND por tanto es una salida a tierra y COM por lo que en el caso especial de alimentar el relé de una fuente externa.

No obstante como hemos citado anteriormente hay una gran gama de relés con características y prestaciones diferentes como puede ser el caso de este módulo de relés.

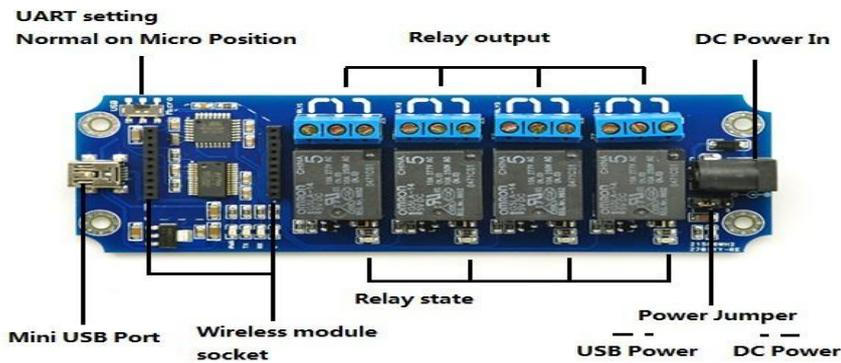


Ilustración 24: Modulo de relés alimentado por USB

3.5 Comunicación Inalámbrica

3.5.1 Control Remoto Raspberry a PC: Mediante Ssh y Putty

3.5.2 SSh

SSH o Secure Shell es un protocolo de red el cual permite el intercambio de datos a través de un canal seguro entre los dispositivos enlazados en red, como puede ser, por ejemplo, su ordenador y su servidor VPS [14].

Aunque la Raspberry es notablemente menos poderosa que la mayoría de los sistemas que se pueden encontrar en un centro de datos, esto no quiere decir que no puede actuar como un útil servidor dentro de un ambiente casero o de negocios. A pesar de contar con poca memoria y relativamente poco poder de procesamiento, su bajo consumo de energía y su ejecución silenciosa lo convierten en una gran opción para servir páginas sencillas de bajo tráfico a una red local o incluso afuera en la Internet.

Una gran parte de los modernos servidores web funcionan ejecutando una combinación de Linux, Apache, MySQL y PHP [8] (comúnmente conocidos como la pila LAMP).



Ilustración 25: Logotipo SSh

Linux es el encargado de proveer el sistema operativo: MySQL el soporte para la base de datos; y PHP el lenguaje de scripting para las páginas dinámicas. Con un servidor basado en LAMP, es posible la ejecución de paquetes extremadamente complejos que van desde sistemas de gestión de contenidos como WordPress hasta foros interactivos tales como phpBB. Todo esto es posible con la Raspberry Pi, siempre y cuando no espere un rendimiento similar al de un poderoso servidor comercial.

CONSEJO

Los servidores web trabajan mejor cuando disponen de mucha memoria. Para garantizar el máximo rendimiento, tendremos que cambiar el particionalmente la memoria de Raspberry a una división de 224/32 MB.

3.5.3 Putty

Putty [14] puede actuar en diferentes aplicaciones entre ellas cliente de SSH, Telnet, rlogin, y TCP raw todos estos protocolos de conexión disponen de licencia libre. Esta originalmente sólo es apta para Windows y ahora también está disponible en varias plataformas como Unix, y se está desarrollando una nueva versión para Mac OS las diferentes versiones. Otra gente ha contribuido con versiones no oficiales para otras plataformas modificando este código libre, tales como Symbian para teléfonos móviles ya que este software beta escrito y mantenido principalmente por Simón Tatham de la compañía, Open source y licenciado bajo la Licencia MIT.

PuTTY sirve como cliente SSH y Telnet es el enlace con el que podemos conectarnos a servidores remotos iniciando una sesión en ellos que nos permite ejecutar comandos. El ejemplo más claro es cuando empleamos PuTTY para ejecutar comandos en un servidor VPS y así poder instalar algún programa o configurar el servidor.

La parte de cliente Telnet sirve para conectarse a nuestro router doméstico. Para ellos hay que abrir los diferentes puertos para la conexión.

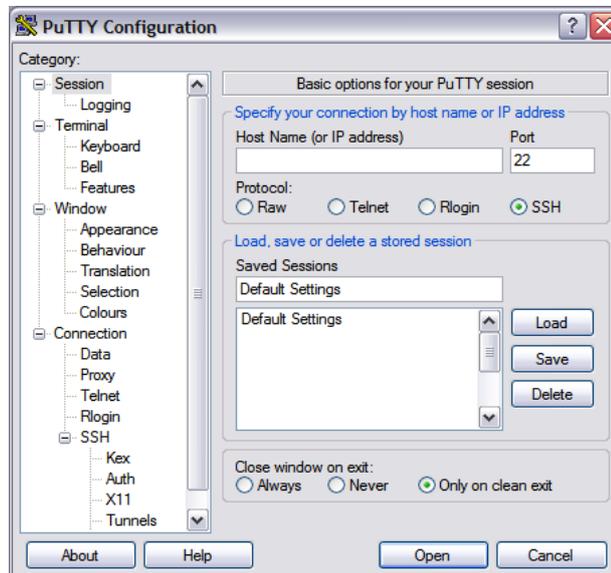


Ilustración 26: Pantalla configuración PUTY

3.5.4 Control Remoto VNC Server en Raspberry Pi con PC o Móvil

VNC con las siglas de *Virtual Network Computing* (**Computación Virtual en Red**). Es un programa de software de código libre basado en un vínculo entre cliente-servidor. El vínculo entre ellos permite tomar el control del ordenador servidor remotamente a través de un ordenador cliente, es decir un software de escritorio remoto. Este software de VNC no impide el acceso al sistema operativo del ordenador servidor con respecto al del cliente: donde es posible compartir la pantalla del PC de un usuario con respecto cualquier sistema operativo que soporte VNC, conectándose desde otro ordenador o



Ilustración 27: Logotipo REALVNC

dispositivo que disponga de un cliente VNC soportado.

La primera versión original del VNC se desarrolló en Reino Unido (Cambridge). El software era de código abierto para así poder realizar diferentes modificaciones en el sistema, donde hoy en día existen varios programas para el mismo uso pero con modificaciones gráficas o de desarrollo. Muchos software son derivados del código libre bajo licencia GNU General Public License.

En la parte de la educación y enseñanza VNC se puede utilizar para casos en que el profesor pueda impartir una clase, por ejemplo en un laboratorio de fuera del centro de enseñanza. También suele usarse para dar ayuda a un usuario inexperto, el técnico ve remotamente el problema y este informa el usuario.

El programa servidor también tiene la opción de funcionar como servidor HTTP en el cual se podrá mostrar la pantalla compartida en un navegador web con soporte para Java. En este caso el usuario remoto (cliente) no tiene que instalar un programa cliente de VNC, éste es descargado automáticamente por el navegador web.

VNC es independiente de la plataforma donde se vaya a utilizar ya que utiliza el navegador web para su uso.

Un cliente VNC instalado en un sistema operativo puede conectarse a un servidor VNC de la misma o de cualquier otro sistema operativo (Mac Os, Windows, Linux). Hay diferentes modelos de clientes y servidores para muchos sistemas operativos basados en GUI o Java. Donde uno o un grupo de clientes pueden conectarse a un servidor VNC al mismo tiempo, siempre y cuando se configure el servidor.

VNC ofrecen distintos modelos de conexión, aparte de las funciones típicas de VNC, funciones particulares optimizado para Microsoft Windows; otros disponen de transferencia de archivos, (no es el principal uso de VNC) si no de un gestor de archivos como puede ser Filezilla. VNC suele ser muy compatible en el sentido de que un usuario de VNC puede conectar con un servidor de otra, mientras que otros se basan en el código fuente de VNC, pero no son compatibles con el estándar VNC.

VNC se compone de 3 campos, cliente, servidor, y un protocolo de comunicación:

- ✓ VNC cliente es el usuario que controla e interactúa con el servidor. El cliente siempre controla al servidor.
- ✓ VNC servidor es el programa instalado en el equipo que permite compartir su pantalla con el cliente.
- ✓ VNC protocolo es muy simple, basado en una primitiva gráfica del servidor al cliente.

En el método normal de operación, un cliente se conecta a un puerto en el servidor (puerto por defecto 5900). Alternativamente, un navegador puede conectarse al servidor (puerto por defecto 5800). Además el servidor puede conectarse a un espectador en "modo de escucha" (puerto 5500). Una de las ventajas del modo de escucha es que el servidor no tiene que configurar su cortafuegos para permitir el acceso a este usando los puertos 5900 o 5800, por lo tanto toda la responsabilidad recae sobre el espectador, lo que es una ventaja muy importante si el sitio del servidor no tiene conocimientos informáticos, mientras que del usuario se espera que tengas más conocimientos en esta rama.

3.6 Lamp

LAMP es un conjunto de factores que lo componen y proporciona detalles de cómo configurar un Raspberry Pi como un servidor web.

Se denomina un servidor LAMP que es uno de la configuración más común para servidores web que estándar para:

Linux-sistemaoperativo

Un Pache - servidor web (http) de software

Mysql - servidor de base de datos

PHP o **P**erl - lenguajes de programación

Toda la configuración del servidor se realiza mediante líneas de comandos. El uso es un poco complejo, pero tiene muchas ventajas, incluyendo la capacidad de gestionar de forma remota e instalar el servidor. También significa que el equipo puede hacer uso del servidor en las páginas web en menos tiempo de procesador que drenan una interfaz gráfica de usuario, que es después de todo, el punto central de un servidor web.

3.6.1 Control Apache

El **servidor Apache** [9] como su nombre indica es un servidor web de código abierto, para muchas plataformas como Linux, Microsoft Windows o Mac Os.

En 1995 cuando comenzó su desarrollo se basó inicialmente en código NCSA HTTP pero más tarde fue reescrito por completo.

Su nombre se debe a la tribu Apache que fue la última en rendirse 1886 al que pronto se convertiría en gobierno de EEUU.

La preocupación de su Apache era que llegasen las empresas y particulares para dominar lo que habían creado los primeros ingenieros informáticos de Apache. Además Apache consistía solamente en un conjunto de parches a aplicar al servidor de NCSA.



Ilustración 28: Logotipo Pila LAMP

El servidor Apache se desarrolló dentro del proyecto HTTP Server de la Apache Software. Apache a diferencia de otros servidores presenta otras funciones como la bases de datos de autenticación y negociado de contenido, esta que fue muy criticada por la ausencia de una interfaz gráfica que ayudase en su configuración ya que sin ella es mucho más compleja redefinir algunas opciones.

Además tiene una gran aceptación en la red justo un año después de su creación. También es el servidor HTTP más usado, siendo 2005 el servidor utilizado hasta llegar al 70% de los sitios web en el mundo donde alcanzó su máxima cuota pero al contrario de esos años se ha sufrido un descenso en la cuota de mercado estos últimos años.

Hoy en día todos los sistema siempre tiene vulnerabilidades como Apache pero estas solo pueden ser aprovechadas por los usuarios locales del servidor y no remotamente desde otro lugar.

No obstante algunas pueden ser accionadas remotamente en ciertas situaciones críticas, o explotar por los usuarios locales más malévolos en las disposiciones de recibimiento compartidas que utilizan PHP como módulo de Apache.

Principalmente se usa este programa para crear y enviar páginas web en la Word Wide Web donde estas páginas pueden ser dinámicas o estáticas. Muchas de estas aplicaciones web están diseñadas asumiendo como ambiente de implantación a Apache, o el uso de algunas características propias de este servidor web.

Apache es el componente principal del servidor web en la popular plataforma como es LAMP, junto a MySQL y los lenguajes de programación PHP/Perl/Python más usados en este campo.

Este servidor web es redistribuido como un pack, donde el conjunto de varios paquetes propietarios de software, incluyendo la base de datos Oracle y el IBM Web Sphere

application, también Mac OS integra apache como parte de su propio servidor web y como soporte de su servidor de aplicaciones. Además es soportado por Borland en las herramientas de desarrollo como es el caso de Kylix y Delphi.

Apache no solo sirve para la creación de web es además usado para otras muchas tareas como el contenido que necesita ser expuesto en una forma segura y fiable. El ejemplo más claro es el momento de compartir archivos desde un ordenador personal hacia Internet.

Si el usuario tiene Apache instalado en su PC puede colocar archivos en la raíz de documentos de Apache, donde estos pueden ser compartidos.

Además existe un grupo de programadores de Apache con el fin de pre visualizar y probar código mientras éste es desarrollado.

3.6.2 Lenguaje programación PHP

PHP (acrónimo de *PHP: Hypertext Preprocessor*) es un lenguaje de programación de web de código abierto. Además está especialmente adecuado para el desarrollo web y una característica muy importante, el poder ser incrustado en HTML [13].

En vez de tener que usar muchos comandos para mostrar HTML (como en C++ o en Perl), las páginas de PHP contienen HTML con código implementado que muestran o ejecutan acciones desde un segunda plano (un ejemplo se va a mostrar "¡Esto es un script de PHP!").

El texto o código que se quiere mostrar de PHP está interpretado entre los símbolos de comienzo y final del código `<?php y ?>` que permiten entrar y salir del "modo PHP".

PHP se diferencia respecto a Java script en el código, donde es ejecutado en el servidor, generando HTML y este es re direccionado al cliente. Dicho cliente recibirá el resultado del código escrito en PHP después de ejecutar el script, aunque nunca se sabrá el código subyacente que era. No obstante también el servidor web puede ser configurado incluso para que procese solamente todos los ficheros HTML con PHP, por lo que no habría manera de que los usuarios que lean dicho script no puedan saber qué contiene dicho script.

La ventaja principal de utilizar PHP es su simple manejo para el principiante, pero también ofrece muchas otras características avanzadas relacionadas para los programadores más avanzados y profesionales. Aunque el desarrollo de PHP está solo centrado en el campo de la programación de scripts del lado del servidor, también es posible utilizarlo para otras funciones.

3.6.3 Control MYQSL

MySQL fue fundado por Michael Widenius , David Axmark y Allan Larsson y es un sistema de gestión de bases de datos relacional en red.

Este sistema permite bajo la GNU GPL para cualquier uso que sea compatible con esta licencia, pero las empresas que quieran obtenerlo deben comprar a la empresa una licencia específica que les permita este uso ya que sin él es imposible además está programado en su mayor parte en ANSI C.

Al contrario de Apache, donde el software es desarrollado por una comunidad pública y los derechos de autor del código son libres y están en poder del autor individual que lo requiera, MySQL es una empresa privada, con los derechos de copyright de la mayor parte del código y de la posterior venta de licencias privativas, la compañía ofrece soporte y servicios. Para sus operaciones de desarrollo contratan trabajadores de todo el mundo que colaboran vía Internet por MySQL.

Lenguajes de programación en MYQSL

Existen varias interfaces de programación de aplicaciones que permiten a las aplicaciones escritas en diversos lenguajes de programación, acceder a las bases de datos MySQL, incluyendo C, C++, C#, Python, PHP, Pascal, Delpi, Eiffel, Smalltalk, Lisp, etc.

Todos y cada uno de estos lenguajes de programación utiliza una interfaz de programación de aplicaciones específica.

Además de una interfaz llamada MyODBC que permite a cualquier lenguaje de programación que soporte ODBC comunicarse con las bases de datos MySQL.

Aplicaciones

MySQL suele ser muy utilizado en aplicaciones web php en plataformas (Windows o Linux y Mac Os) y diferentes tipos de servidor (Apache) y lenguaje de programación (PHP/Perl/Python), también opta por usar herramientas de seguimiento de errores como Bugzilla. Su éxito como aplicación web está muy unido a PHP, que a menudo aparece en combinación con MySQL.

Por lo tanto estamos hablando de una base de datos rápida y fluida en aplicaciones web donde no se suelen realizar modificaciones de datos y en cambio el entorno es completo e intuitivo en lectura de datos, lo que hace a MySQL uno de los sistemas de transferencia de

datos ideal para este tipo de aplicaciones. Dentro del entorno en el que se va a utilizar MySQL, es muy importante prever el rendimiento del sistema para detectar y corregir problemas tanto de SQL como de programación.

3.7 Seguridad Raspberry

La seguridad es otro de los apartados más importantes de la Raspberry, ya que pueden hacer daños en dicha placa si no se cuenta con la seguridad adecuada. Cuando se conecta cualquier dispositivo a INTERNET con una baja seguridad del mismo puede ser un problema porque cualquier persona con los conocimientos adecuados puede controlar el sistema a su placer. Por lo tanto si se utiliza Raspberry Pi como servidor web o FTP se deberá establecer una seguridad al dispositivo. Ya que sin seguridad podrían ser víctimas de su propio sistema. La mayoría de los ataques para obtener las contraseña se produce por fuerza bruta. Así que se endurecerá la contraseña contra esos ataques maliciosos.

Ataque de fuerza bruta es la forma de recuperar una clave probando todas las combinaciones posibles hasta encontrar aquella que permite el acceso.

LOGS: Lo denominamos como un registro donde se guardan todas las conexiones Ssh que se hacen desde cualquier dispositivo.

Para ellos se dispondrá de una contraseña de Raspberry para la seguridad.

CONTRASEÑA

Se deberá de elegir una contraseña segura para SSH Y VNC.

COMANDOS CAMBIAR CONTRASEÑA

```
# raspi-config # passwd root
# passwd pi
# passwd usuario
```

Desactivar login con root desde SSH

DESACTIVAMOS root en ssh ya que con root activado seriamos muy vulnerables.

```
# nano /etc/ssh/sshd_config
```

Límite de intentos por Ip

Para evitar los varios intentos de Ip, se bloqueará el número máximo de accesos fallidos desde una Ip y el número máximo de sesiones simultaneas que se puedan abrir.

```
MaxAuthTries 2
```

```
MaxStartups 5
```

Cambiar puerto de conexión

En el caso de cambiar de puerto se deberá elegir un puerto superior a 1024. Ya que los demás puertos hasta el 1024 tienen unas funciones ya aplicadas.

3.8 Diseño de Circuitos con Fritzing

Fritzing [2] es un software open-source que permite montar circuitos esquemáticos, crear PCB's al igual que con otros programas, y, aquí su especialidad, mostrar el montaje del circuito sobre una placa protoboard; es muy importante ese montaje antes de dar un proyecto por finalizado, lo que es la fase de prototipado. Esto es muy útil para montar circuitos electrónicos. Con Fritzing, se pueden montar circuitos en una protoboard, y crear a partir de él un esquema, viendo si se corresponde con el circuito, para asegurarse de la instalación.

Para descargar el programa Fritzing que es un software de código abierto, desde la página oficial (disponible para Windows y Linux).

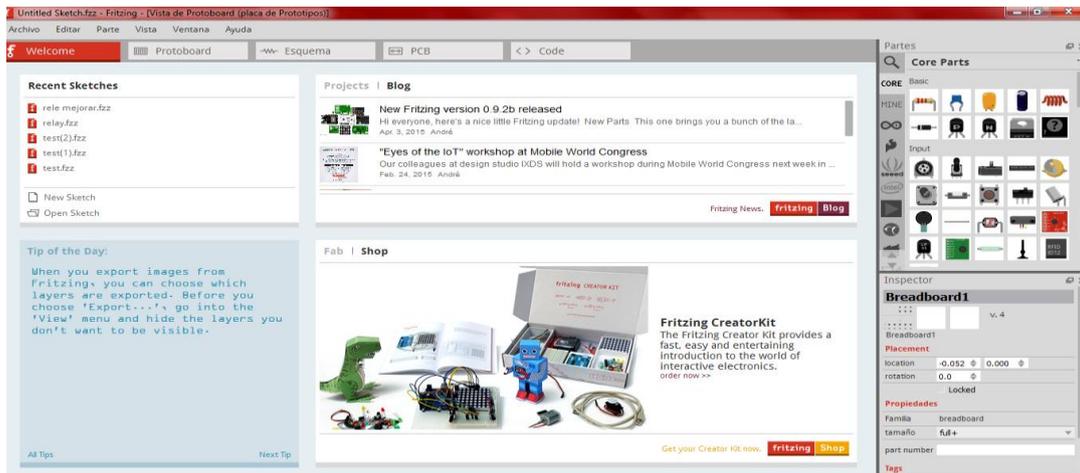


Ilustración 29: Pantalla de Inicio Fritzing

Una vez iniciado el programa (primera imagen), se mostrará las diferentes opciones de inicio.

En el modo protoboard (segunda imagen) se realizarán las conexiones correspondientes al proyecto en sí en una placa protoboard con una gran gama de componentes seleccionables que trae por defecto el programa.

CONTROL REMOTO DE ILUMINACION CON MICROCONTROLADORES Y SERVICIO WEB

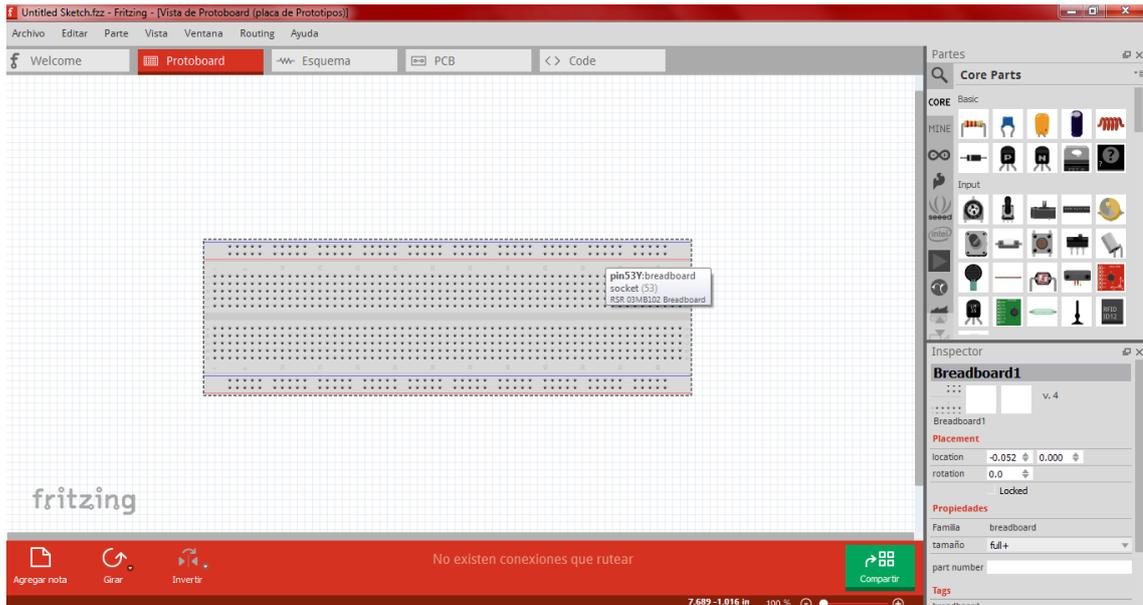


Ilustración 30: Pantalla Protoboard Fritzing

De ésta manera, tan solo introduciendo el montaje en la protoboard, se puede comprobar si el diseño es válido (comparando el esquema resultante con el que tenemos de antemano). Para diseñar una PCB para el circuito, simplemente vale con darle a mostrar PCB (igual que con el esquema, sólo que con PCB, y darle a ejecutar, al igual que con el esquemático. Nos ejecutará la PCB. Por defecto, Fritzing diseña las placas de doble cara, pero para cambiarla, basta con seleccionar las propiedades de la placa y cambiar "two layers" por "one layer" y volver a cambiar.

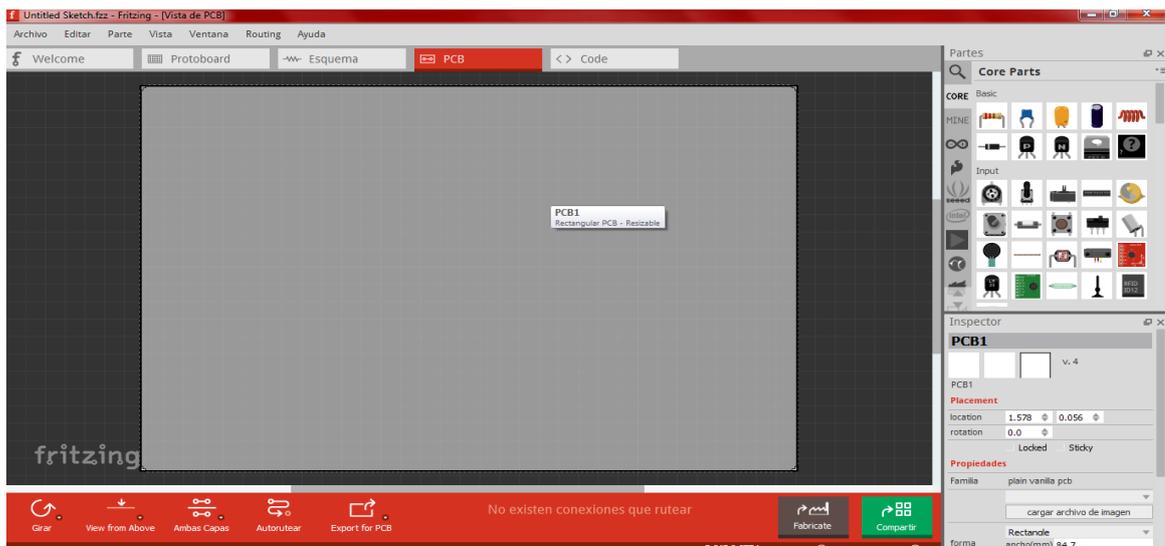


Ilustración 31: Pantalla Esquemas PCB Fritzing

CAPÍTULO 4: CONTROL DOMOTICO CON RASPBERRY B+



4 CONTROL DOMOTICO CON RASPBERRY B+

4.1 Introducción

Este trabajo de fin de grado consiste en la domotización de luz de una vivienda o local mediante un control remoto inalámbrico o las propias llaves de luz del hogar, optando así a un control simultáneo con la posibilidad de controlarlo desde ambas partes.

La constancia del desgaste energético de luz en las aulas de la EIMIA me llevó a la realización de dicho proyecto, además de mejorar el confort al personal de conserjería.

Los elementos necesarios para domotizar un hogar o local son una central de gestión, sensores o detectores, actuadores y un soporte de comunicación.

La central de gestión la formará el micro procesador Raspberry es la encargada de controlar y recibir la información de los actuadores y sensores como es el caso del sensor de corriente que informará a la central de gestión del estado de la luz en cada habitación. Para el apagado y encendido de la luz se realizará mediante una placa de relés (actuador) controlada por la central de gestión, donde todo este sistema es controlado por el usuario mediante el soporte de comunicación en el navegador web o por las llaves de luz del hogar.

Por lo tanto el sistema consta de un control automático controlado o dirigido por internet mediante el usuario y un control manual de luces mediante los interruptores del hogar o la vivienda.

Para la construcción de dicho proyecto los componentes necesarios son Raspberry como central de gestión y es la encargada de controlar un módulo de 4 relés para el control de luces, también son necesarios 4 sensores de corriente donde su función es leer el paso de corriente que circula por la fase de cada relé para saber el estado de la bombilla. Estos sensores de corriente solamente hacen lecturas en codificación analógica, por lo tanto esta señal no puede ser leída por nuestra central de gestión por lo que se utilizan convertidores analógico a digital (como es el caso del MCP3008) para que nuestra central Raspberry puede hacer la lectura de estos sensores correspondientes. Todo este sistema es controlado por el usuario a través de una página web que será el soporte de comunicación o las llaves de luz propias del hogar.

Dicha construcción debe ser aplicada o implementada a una vivienda u hogar.

Ya que sus buenas prestaciones y su bajo coste suponen una gran ventaja para la realización de este trabajo.

Todo esto ha sido comprado en unas tiendas especializadas en componentes electrónicos de Arduino en España.

4.1.1 Principales ventajas del proyecto

Las principales ventajas sobre otros proyectos de la misma índole son el bajo coste del proyecto con respecto al precio excesivo de las casas domóticas existentes hasta ahora.

La ventaja principal de este proyecto es la posibilidad de controlar la instalación de iluminación de 2 maneras simultáneas al mismo tiempo, la primera mediante los interruptores o llaves ya integradas en la instalación y la segunda posibilidad es el control remoto inalámbrico desde un navegador web; ambas posibilidades son funcionales en dicho proyecto con la posibilidad de controlar la iluminación desde cualquier parte del mundo simplemente con el uso de un navegador web, sin la necesidad de instalar ningún software adicional. Solo hace falta una conexión a internet desde cualquier PC o móvil.

4.2 Arquitectura General del Sistema

Para la realización de dicho proyecto con sus componentes necesarios citados anteriormente, se detallará la funcionalidad y el comportamiento de cada uno de ellos además de detallar las conexiones de todos y cada uno de los elementos existentes en la instalación.

4.2.1 Central de gestión

Raspberry pi cumple con las características necesarias para ser la central de gestión principal para controlar los relés y sensores de corriente aplicados a este proyecto de luces (ver sección 3.3).

Dicha central de gestión deberá soportar e interactuar con un módulo de 4 relés con los respectivos sensores de corriente en cada uno de ellos, también debe actuar como servidor para el soporte de comunicación con el usuario a través del navegador web.

A continuación se verá la placa Raspberry Pi b+ y sus respectivas partes.

- 1) Conexión de 40 pines GPIO
- 2) Procesador Broadcom BCM 2835 / 512 Mb Ram
- 3) 4 puertos USB 2.0
- 4) Conexión Ethernet RJ45
- 5) Conexión Jack 3.5 mm
- 6) Conector para cámara
- 7) Conexión 232 srs
- 8) Micro Sd
- 9) Alimentación micro USB 5v
- 10) HDMI

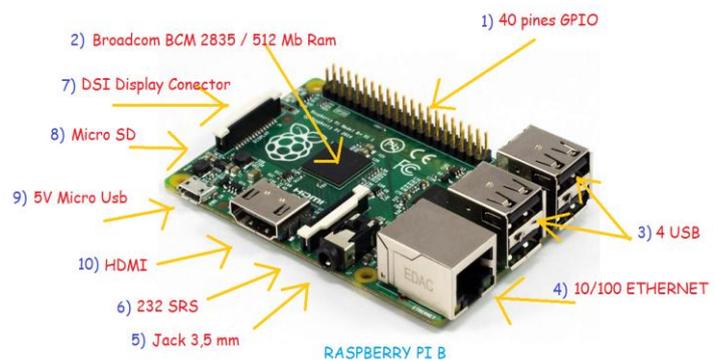


Ilustración 32: Partes de Raspberry B+

El bajo coste de esta placa con la gran potencia de su procesador además que consta de 40 pines de salidas y entradas son suficientes para el proyecto. Ya que solo se utilizaran 6 para el control del relé, 3 para el sensor de corriente ACS714 y 9 para el convertidor analógico a digital.

4.2.2 Control de activación/desactivación de luz

Para el control de activación y desactivación de la luz se eligió este módulo de relés que permite dicho control con las órdenes mandadas de la central de gestión gracias a su sistema de actuar con los 4 canales en corriente continua, además de contar con la posibilidad de controlar voltajes de hasta 220 V y 10 A. Ya que existen diferentes modelos en el mercado se utilizará este modelo en concreto porque se alimenta de 5V que es la suministrada por la placa Raspberry pi y está adaptado para la corriente eléctrica del hogar (220-230V).

Con este módulo de relé de 5V y 4 canales con opto acoplador, será capaz de controlar hasta 4 fases distintas, es decir para el control de luces de hasta 4 habitaciones.

También están acoplados una serie de componentes como son el diodo por cada relé ya que este impide sobretensiones derivadas del accionamiento de la bobina del relé y un transistor para asegurar un buen voltaje para mandar la señal al relé

Además su conexión mediante pin para su posterior control desde una web lo hacer ser una de los componentes más principales.

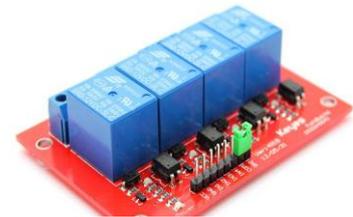


Ilustración 33: Placa de Relés KEYE

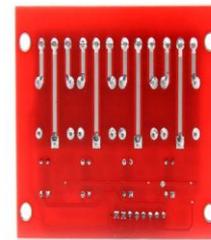


Ilustración 34: Vista inferior placa de relés KEYES

Características técnicas

- Alta calidad
- 5V 4-Canales placa interface relé, necesita 15-20mA de corriente para excitar el driver.
- Entrada protegida por opto acoplador.
- Equipado con un relé de alta corriente, AC250V 10A ; DC30V 10A
- Interface standard puede ser controlado directamente por un micro controlador (Arduino , 8051, AVR, PIC, DSP, ARM, ARM, MSP430, TTL logic)
- Contactos independientes para aislar y proteger el circuito.
- Dimensiones 7.5 x 5.5 x 1.8cm
- Peso 61 gr

Por lo tanto, este relé ha sido elegido por su prestación aunque se pueden fabricar mediante piezas por separado pero su precio tan bajo no compensa comprar las partes por separado y construirlo. Además, se debe tener en cuenta que se está trabajando con una corriente eléctrica y unos voltajes peligrosos.

Estos componentes necesitan actuar con unos mA adecuados para su correcto funcionamiento. Como podemos ver en las características del relé cada pin necesitar ser alimentado entre 15 y 20 mA para ser excitado.

Al tener 4 módulos relé necesitaríamos entre 60 y 80 mA que aportaría la Raspberry para alimentar esos pines. Al ser entre esos valores (60-80 mA) nos es suficiente con la Raspberry ya que esta puede aportar entre 62 y 80 mA en el caso de poner más módulos de relé necesitaríamos alimentar el relé con una fuente externa de 5V. Las fuentes externas pueden ser por USB o corriente dc. El tiempo de accionamiento dependerá de la rapidez con la que el servidor recibe y manda la señal con un tiempo mínimo de 1 segundo.

4.2.3 Detección de corriente.

Para la lectura de paso/ausencia de corriente en las luces del hogar se debe de elegir un sensor adecuado que soporte una intensidad de hasta 10 A que es lo que soportan los relés anteriormente citados y que pueda enviar dicha lectura a la central de gestión para saber el estado de la luz. Este sensor elegido puede soportar hasta un máximo de ± 30 A, además necesita ser alimentado por 5V de la central de gestión para su uso.

Este modelo al igual que otros con medias inferior permiten medir corriente en CA y CC. Lo que es una ventaja ya que los relés elegidos anteriormente soportan hasta 10 A.

Todos los tipos de sensores suelen mandar la señal de lectura en analógica. Por lo que la placa debe costar de entradas analógicas o como es el caso de la placa Raspberry se necesita un convertidor analógico a digital como es el caso del MCP3008 para que este pueda leer la lectura analógica que manda dicho sensor.

El paquete de dispositivo permite una fácil aplicación por el cliente. Las aplicaciones típicas incluyen el control motor, la carga detección y gestión, fuentes de alimentación conmutadas, y protección de fallo de sobre intensidad.

El dispositivo consta de un circuito preciso, bajo offset, lineal pasillo con una trayectoria de conducción de cobre ubicado cerca de la superficie de la morir. Corriente que fluye a través de esta conducción de cobre ruta genera un campo magnético que la Sala IC se convierte en una tensión proporcional. Exactitud de dispositivos se optimiza a través de la señal magnética al transductor Hall.

Una tensión precisa, proporcional es proporcionado por Low-compensar el, chopper-estabilizado BiCMOS IC Hall, que está programado la exactitud de su envasado.

La salida del dispositivo tiene una pendiente positiva ($> V_{IOUT} (Q)$)

Cuando una corriente fluye a través del aumento de cobre primario camino de conducción (de las patas 1 y 2, a los pines 3 y 4), la cual es la ruta utilizada para el muestreo actual. La resistencia interna de este camino conductor es de 1,2 m Ω típico, proporcionando baja



Ilustración 35: Sensor corriente ACS714

potencia pérdida. El espesor del conductor de cobre permite la supervivencia del dispositivo en un máximo de 5 × condiciones de sobre corriente.

Tipos ACS

Modelos	Rango: Int (A)	Sensibilidad (Typ) (mV/A)	T _A (°C)	Lectura
ACS714ELCTR-05B-T	±5	180	-40 a 85	3000 piezas/rell
ACS714ELCTR-20A-T	±20	100		
ACS714ELCTR-30A-T	±30	66		
ACS714ELCTR-05B-T	±5	185	-40 a 150	
ACS714ELCTR-20A-T	±20	100		
ACS714ELCTR-30A-T	±30	66		
ACS714ELCTR-50A-T	±50	40		

Tabla 3: Modelos Sensor de Corriente ACS

Rangos absolutos

Características	Símbolo	Notas	Clasificación	Unidad
Voltaje de alimentación	V _{CC}		8	V
Voltaje de alimentación invertido	V _{RCC}		-0,1	V
Voltaje de Salida	V _{IOUT}		8	V
Tensión inversa de salida	V _{RIOUT}		-0,1	V
Fuente de Salida actual	I _{IOUT} (Fuente)		3	mA
Salida Sink actual	I _{OUT(SKIN)}		10	mA
Tolerancia transitoria Sobre corriente	I _P	1PULSO 100MMS	100	A
Temperatura de ambiente de funcionamiento nominal	T _A	TIPO E	40 a 85	°C
		TIPOL	-40 a 150	°C
Temperatura Máxima de unión	T _j (máx.)		165	°C
Temp° de almacenamiento	T _{stg}		-65 a -170	°C

Tabla 4: Características de Funcionamiento ACS714

Características aislamiento

Características	Símbolo	Notas	Clasificación	Unidad
Prueba de rigidez dieléctrica	VISO	Homologada Agencia durante 60 segundos por UL norma 60950-1, primera edición	2100	V_{AC}
Tensión de trabajo para el aislamiento básico	VWFSI	Para (single) Aislamiento básico según la norma UL 60950-1, primera edición	354	V_{DC} a V_{pk}
Tensión de trabajo para el aislamiento reforzado	VWFRI	Para reforzado (doble) de aislamiento según la norma UL 60950-1, primera edición.	184	V_{DC} a V_{PK}

Tabla 5: Características del aislamiento ACS

Functional Block Diagram

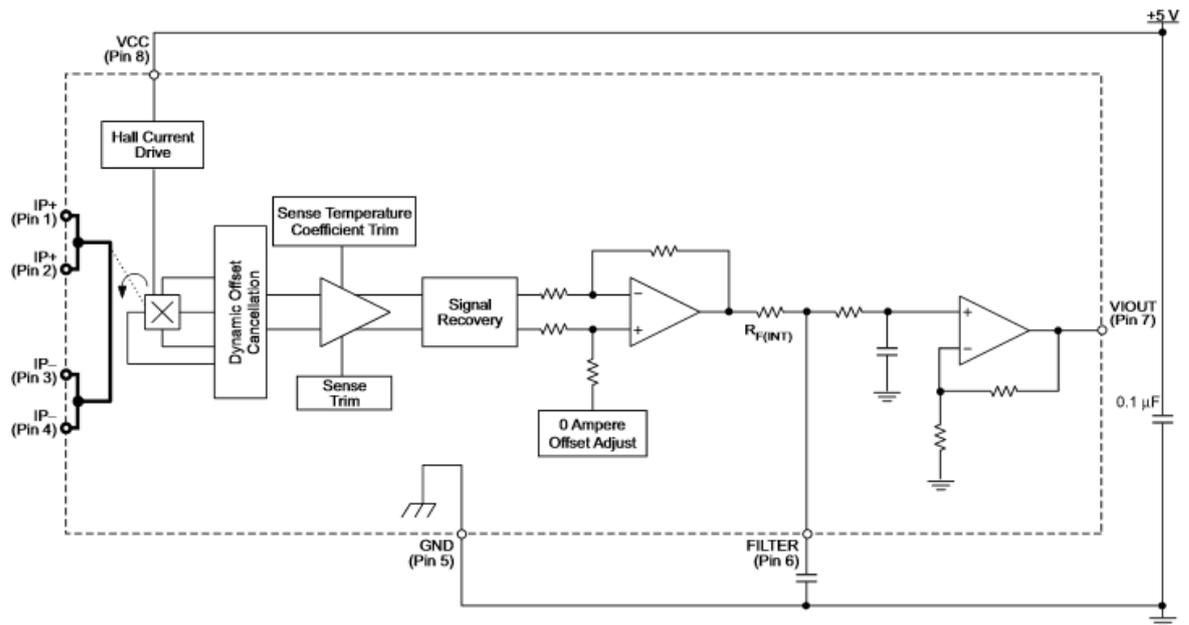


Tabla 6: Esquema de Instalación.

Terminales

Numero	Nombre	Descripción
1 y 2	IP+	Terminales para actuar tomando muestras; fusionado internamente
3 y 4	IP-	Terminales para actuar tomando muestras; fusionado internamente
5	GND	Terminal de tierra
6	FILTRO	Terminal para el condensador para del ancho de banda
7	Viout	Señal de salida analógica
8	VCC	Terminal de alimentación del dispositivo

Tabla 7: Terminales ACS

4.2.4 Convertidor de la señal analógica-digital del sensor de corriente

La central de gestión Raspberry necesita leer la lectura enviada por el sensor de corriente para conocer el estado de la luz en cada parte del hogar, el problema está en que la lectura enviada por el sensor de corriente es analógica lo que impide a la central hacer una lectura, ya que solo puede leer señales digitales, por ello dicha central necesita de un convertidor de analógica a digital. Dicho componente o chip es denominado MCP3008 de Microchip y es el encargado de convertir la lectura analógica del sensor de corriente ACS714 anteriormente citado en una señal digital para que la central Raspberry pueda hacer la lectura, este valor de medida está entre un rango de 0 y 1023 una vez convertida esta señal a digital, que determina el valor de la intensidad que circula por el sensor de corriente.

Este ADC es muy cómodo ya que proporciona 8 entradas analógicas las cuales se pueden controlar fácilmente hasta 8 sensores de corriente mediante un único bus SPI. Es especialmente útil con Raspberry Pi ya que no dispone de este tipo de entradas analógicas y con la ayuda de este chip y un pequeño script en Python se puede leer fácilmente los valores de sensores analógicos como es el caso del sensor de corriente ACS714

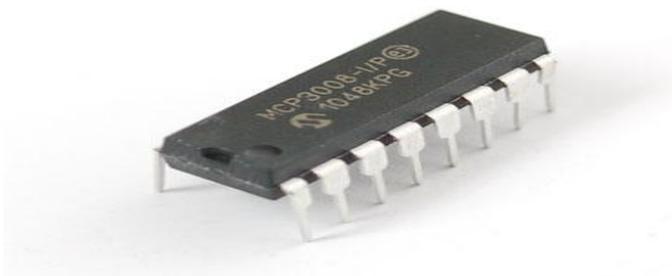


Ilustración 36: ADC MCP3008

Características:

- Interfaz de Datos: Serie, SPI
- Frecuencia de Muestreo: 200kSPS
- Temperatura de Trabajo Máx.: -40°C a 85°C
- Resolución: 10bit
- Tensión de Alimentación Máx.: 2.7V a 5.5V
- Dimensiones: 22x17 mm

En la siguiente imagen se puede ver las funciones que realiza cada pin para su conexión.

PDIP, SOIC

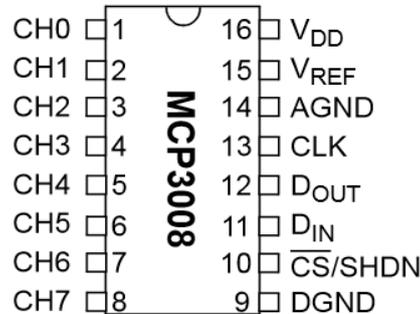


Ilustración 37: Entradas y Salidas MCP3008.

Características eléctricas: V_{dd}=5V V_{REF}=5V T_A=-40°C a +85°C f_{sample}=200ksps y f_{CLK}=18*f_{sample}

Parámetros	Sym	Min	Typ	Max	Unidad	Condición
Tiempo de conexión	T CONV			10	Reloj ciclo	
Entrada analógica					Reloj	
Tiempo de la muestra	T MUESTRA		1,5		Ciclo	
Throughput rate	F MUESTRA			200	Ksps	V _{DD} =V _{REF} =5V
				75	Ksps	V _{DD} =V _{REF} =2,7V
DC Precisión						
Resolución			10		bits	
Linealidad integrada	INL		±0,5	±1	LSB	
Linealidad diferencial	DNL		±0,25	±1	LSB	
Error desplazamiento				±1,5	LSB	
Ganancia de error				1	LSB	
Dynamic Performance						
Distorsión armónica total			-76		dB	V _{EN} = 0.1V a 4.9V@1 kHz

Señal-ruido y la distorsión (SINAD)	61		dB	VEN= 0.1V a 4.9V@1 kH
Rango dinámico libre de espurias	78		dB	VEN= 0.1V a 4.9V@1 kHz
Referencia de entrada			dB	
Rango de voltaje	0,25	VDD	V	Nota 2
Consumo de corriente	100 0,001	150 3	uA uA	CS = VDD= 5V
Entrada analógica				
Rango de voltaje de entrada para CH0 oCH1 en el modo de terminación única	VSS	VREF	V	
Rango de voltaje de entrada para IN + en modo pseudo-diferencia.	IN-	VREF	IN-	
Rango de voltaje de entrada para IN- en modo pseudo-diferencial	V SS -100	VSS+100	mV	

Tabla 8: Características Eléctricas MCP3008

ESPECIFICACIONES ELÉCTRICAS

Características eléctricas: **A menos que se indique lo contrario, todos los parámetros se aplican en VDD = 5V, V REF= 5V, T La = -40 ° C a + 85 ° C, f MUESTRA = 200 ksps y f CLK = 18 * f MUESTRA.** A menos que se indique lo contrario, se aplicarán los valores típicos para V DD = 5V, T La = + 25 ° C.

Parámetros	Sym	Min	Typ	Max	Unidad	Condición
Requisitos de energía						
Tensión de funcionamiento	VDD	2,7		5,5	V	
Corriente de funcionamiento	IDD		425 225	550	uA	V DD = V REF = 5V, D OUT descargadas V DD = V REF = 2.7V, D OUT descargada
Corriente espera	IDDS		0,005	2	uA	CS = VDD = 5.0V

Nota

- 1: Este parámetro se establece mediante la caracterización y no es 100% a prueba.
- 2: Ver gráficas que relacionan el rendimiento linealidad a VREF los niveles.
- 3: Debido a que la tapa de la muestra con el tiempo perder carga, velocidades de reloj efectivos inferiores a 10 kHz pueden afectar la linealidad rendimiento, especialmente a temperaturas elevadas. Consulte: Mantenimiento mínimo Velocidad del reloj ", para más información.

Tabla 9: Especificaciones Eléctricas MCP3008

CARACTERÍSTICAS DE TEMPERATURA

Especificaciones eléctricas: **A menos que se indique lo contrario, VDD = + 2.7V a + 5.5V, V SS = GND**

Parámetros	Sym	Min	Typ	Max	Unidad	Condición
Rangos de Temperaturas						
Rangos de Temperatura especificado	T _A	-40	-	85	°C	-
Rango de temperatura de funcionamiento T	T _A	-40	-	85	°C	-
Temperatura de almacenamiento	T _A	-65	-	150	°C	-
Resistencias paquete térmicas						
Resistencia térmica 14L-PDIP	Θ _{JA}	-	70	-	°C/W	-
Resistencia térmica 14L-SOIC	Θ _{JA}	-	108	-	°C/W	-
Resistencia térmica 14L-TSSOP	Θ _{JA}	-	100	-	°C/W	-
Resistencia térmica 14L-PDIP	Θ _{JA}	-	70	-	°C/W	-
Resistencia térmica 14L-SOIC	Θ _{JA}	-	90	-	°C/W	-

Tabla 10: Características de Temperatura MCP3008

4.2.4.1 Convertidor analógico-digital Script en Python

Para la lectura del sensor de corriente se necesita el siguiente programa en PYTHON el cual estará guardado en la SD de la Raspberry dentro de la carpeta /home

El Script que permitirá la lectura analógica del sensor de corriente es:

```
#!/usr/bin/env python

# just some bitbang code for testing all 8 channels

import RPi.GPIO as GPIO, time, os, sys

DEBUG = 1
GPIO.setmode(GPIO.BCM)

# read SPI data from MCP3008 chip, 8 possible adc's (0 thru
7)
```

La función readadc permite la lectura del sensor de corriente en uno de los ocho canales comprendidos entre 0 y 7 (el número de canal se pasa como primer parámetro de la función). Además, la propia función hace una conversión de la señal analógica a digital generando y devolviendo un valor discreto comprendido entre 0 y 1023. Cuanto menor sea el valor, menor es el valor de la intensidad.

```
def readadc(adcnum, clockpin, mosipin, misopin, cspin):
    if ((adcnum > 7) or (adcnum < 0)):
        return -1
    GPIO.output(cspin, True)

    GPIO.output(clockpin, False) # start clock low
    GPIO.output(cspin, False)   # bring CS low

    commandout = adcnum
    commandout |= 0x18 # start bit + single-ended bit
    commandout <<= 3  # we only need to send 5 bits here
    for i in range(5):
        if (commandout & 0x80):
            GPIO.output(mosipin, True)
        else:
            GPIO.output(mosipin, False)
            commandout <<= 1
```

CONTROL REMOTO DE ILUMINACION CON MICROCONTROLADORES Y SERVICIO WEB

```
GPIO.output(clockpin, True)
GPIO.output(clockpin, False)

adcout = 0
# read in one empty bit, one null bit and 10 ADC bits
for i in range(12):
    GPIO.output(clockpin, True)
    GPIO.output(clockpin, False)
    adcout <<= 1
    if (GPIO.input(misopin)):
        adcout |= 0x1

GPIO.output(cspin, True)

adcout /= 2      # first bit is 'null' so drop it
return adcout

if __name__ == '__main__':
    try:
        # change these as desired
        SPICLK = 11
        SPIMISO = 9
        SPIMOSI = 10
        SPICS = 7

        # set up the SPI interface pins
        GPIO.setup(SPICLK, GPIO.OUT)
        GPIO.setup(SPIMISO, GPIO.IN)
        GPIO.setup(SPIMOSI, GPIO.OUT)
        GPIO.setup(SPICS, GPIO.OUT)

        # Note that bitbanging SPI is incredibly slow on the
        Pi as its not
        # a RTOS - reading the ADC takes about 30 ms (~30
        samples per second)
        # which is awful for a microcontroller but better-
        than-nothing for Linux

        suma = 0
        for i in range(3):
```

En este bucle lo que se hacen son 3 medidas del sensor y se calcula la media aritmética.

```
        lectura = readadc(0, SPICLK, SPIMOSI, SPIMISO,
        SPICS)
        suma += lectura
        #print "lectura ", i+1, "en canal 0: ", lectura
```

CONTROL REMOTO DE ILUMINACION CON MICROCONTROLADORES Y SERVICIO WEB

```
media = suma/3
#print "LA MEDIA ES: ", media
GPIO.setmode(GPIO.BCM)
GPIO.setup(17, GPIO.OUT)
```

Si la media es un valor superior a 80, se considera que la luz está apagada y se enciende. Además, se devuelve el valor 1, a través de la entrada estándar al archivo .php donde se invocó este script. Todo ello para que el correspondiente botón se coloree de verde.

```
if media < 80:
    #esta apagado y he hecho click --> tengo que
encender
    GPIO.output(17, True)
    data = 1
```

Por el contrario, si la condición anterior no se cumple, el script devuelve el valor cero para que el botón dibujado en la interfaz web se coloree de color rojo.

```
else:
    #esta encendido y he hechoclick --> lo tengo que
apagar
    GPIO.output(17, False)
    data = 0

sys.stdout.write(str(data))

except KeyboardInterrupt:
    pass
```

4.2.5 Placa extensión de la central de gestión Pi Cobbler +

Para un mejor conexionado entre la central de gestión y el módulo de relés los sensores y el convertidor de analógico a digital, se utilizará de un extensor de pines de la central de gestión ya que sin él sería complicado acceder a los pines GPIO. Con esta placa de extensión se puede conectar cómodamente a Raspberry Pi a una placa de prototipado (breadboard) para luego conectar los diferentes sensores de corriente y actuador convertido Analógico-Digital.

Esta placa está diseñada solo para Raspberry Pi modelo B+.

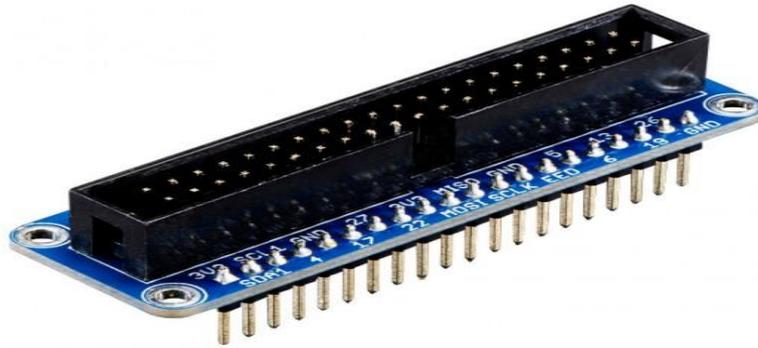


Ilustración 38: Placa de Extensión Pi Cobbler + Raspberry B+

4.3 Conexión de los todos los componentes

Las conexiones de circuitos esquemáticos han sido diseñadas con Fritzing [2] que es uno de los mejores programas para la creación de esquemas e instalaciones electrónicas como es el caso.

En los diferentes capítulos encontrados a continuación están detalladas las diferentes conexiones para el proyecto.

En cada apartado se reflejan la conexión de los circuitos esquemáticos mediante la conexión con Protoboard y también el esquema resultante.

Los componentes principales para el conexionado de la instalación la forman la central de gestión (Raspberry), el módulo de cuatro relés y las llaves de luz como actuadores, con los correspondientes sensores de corriente para cada relé y el convertidor de señal analógica a digital MCP3008 para la monitorización de corriente.

En la imagen posterior se muestra el sistema de luces completo con el correspondiente conexionado entre los diferentes componentes de la instalación y la central de gestión.

- Conexión total

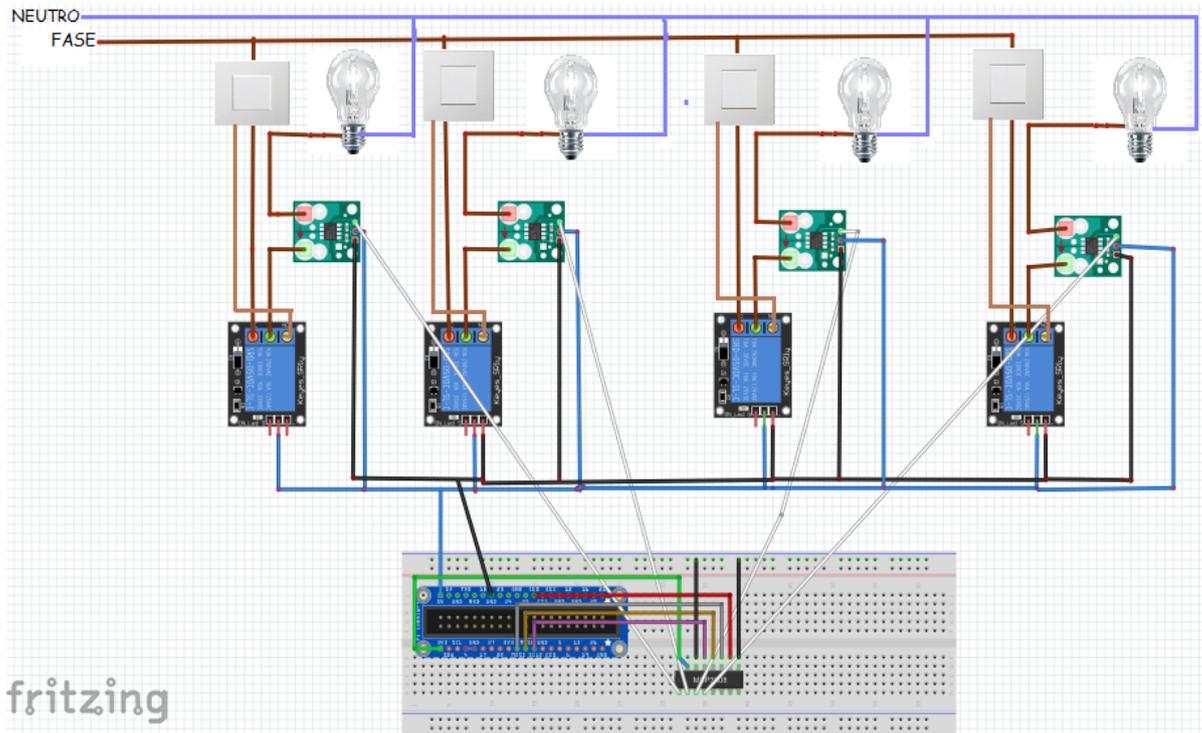


Ilustración 39: Montaje del proyecto completo

La imagen correspondiente a la conexión total muestra el conexionado completo para nuestro control remoto o manual de iluminación, la cual abarca los cuatro tipos de conexiones realizadas en los apartados siguientes, como puede ser el caso de la conmutada de las llaves o interruptores de luz con el módulo de relés para el control de iluminación desde ambas partes, donde también se detalla la conexión del módulo de cuatro relés y la central de gestión Raspberry. La tercera conexión es la relación entre el ADC (mcp3008) y la central de gestión, la cual permite el cambio de la señal analógica a digital para el monitorizado de los sensores de corriente. La conexión cuarta conexión entre los sensores de corriente y el ADC anteriormente citados son los encargados de producir las lecturas del estado de la luz para ser monitorizadas por la central.

Todas estas conexiones están detalladas minuciosamente en los apartados siguientes correspondientes a cada conexión citada.

4.3.1 Conexión de lámparas con relé e interruptor

El control de luces lo realizara la placa de 4 relés, esta placa consta de 2 partes una continua y otra alterna.

La parte de corriente alterna corresponde a la conexión del relé con la instalación eléctrica de luces de la vivienda y los interruptores de la vivienda.

Esta placa de relé estará situada justo después del cuadro eléctrico de luces entre las PIAs (Pequeños interruptores automáticos) y las luminarias del local.

En el caso de los interruptores estos deberán estar entre las PIAs y dicho relé, ya que este será el componente que nos permite accionar la luz desde 2 puntos diferentes.

El modo manual del interruptor y el modo automático del relé controlado vía WIFI.

La conexión entre los relés y los interruptores será en (conmutada) ya que es la única forma del sistema para controlar los elementos anterior nombrados.

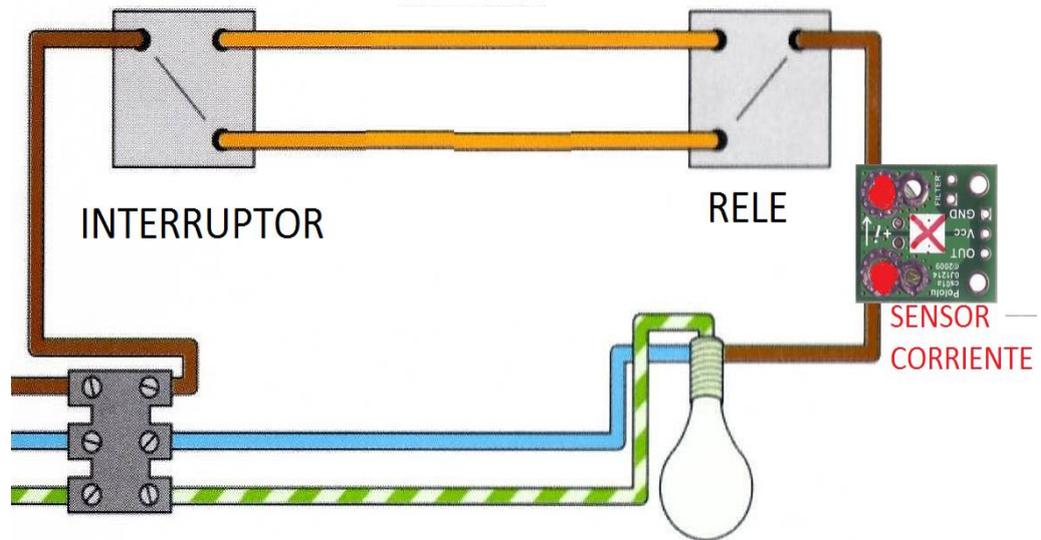


Ilustración 40: Conmutada Interruptor y rele con Sensor de corriente ACS714

La parte de continua corresponde con la instalación de la placa de relé con Raspberry que será la encargada de controlar las luces mediante el control remoto. Esta parte esta detallada a continuación.

4.3.2 Conexión Raspberry b+ con relés

Raspberry se encargará del modo Automático, donde la central de gestión será controlada por internet por el usuario.

Esta conexión nos permite mediante la web creada en el apartado (4.8) encender y apagar las luces mediante internet.

Esta conexión nos permite por un lado alimentar nuestro relé con 5v (proporcionados por Raspberry) que es la tensión mínima con la que podemos alimentar estos relés.

En nuestra placa de relé viene ya una solo GND para todos los relé pero para la creación de dicho conexionado con Frizting aún no están en la librería ese modelo de relé por lo que tendré que simplificar cada tierra (GND) de cada relé en una sola GND.

Cada Pin de señal de cada relé estará conectado a un pin diferente de la Raspberry que se encargara de recibir y enviar señales (de tensión) para accionar los respectivos relés para su funcionamiento.

La conexión entre Raspberry y módulo de 4 relés es la siguiente:

<u>Raspberry b+</u>		<u>Relé</u>
Pin 5v	-----	VCC
Pin GND	-----	GND
Gpio 27	-----	IN1
Gpio 22	-----	IN2
Gpio 6	-----	IN3
Gpio 19	-----	IN4

Como se puede ver en la imagen posterior el conexionado entre el módulo de cuatro relés con respecto la central de gestión se debe alimentar dicho módulo con 5V para su funcionamiento además de su respectiva tierra GND, los pines de control del módulo de relés son los que apagan o encienden las luces y tienen que estar conectados a los pines correspondientes de la central de gestión anteriormente nombrados.

CONTROL REMOTO DE ILUMINACION CON MICROCONTROLADORES Y SERVICIO WEB

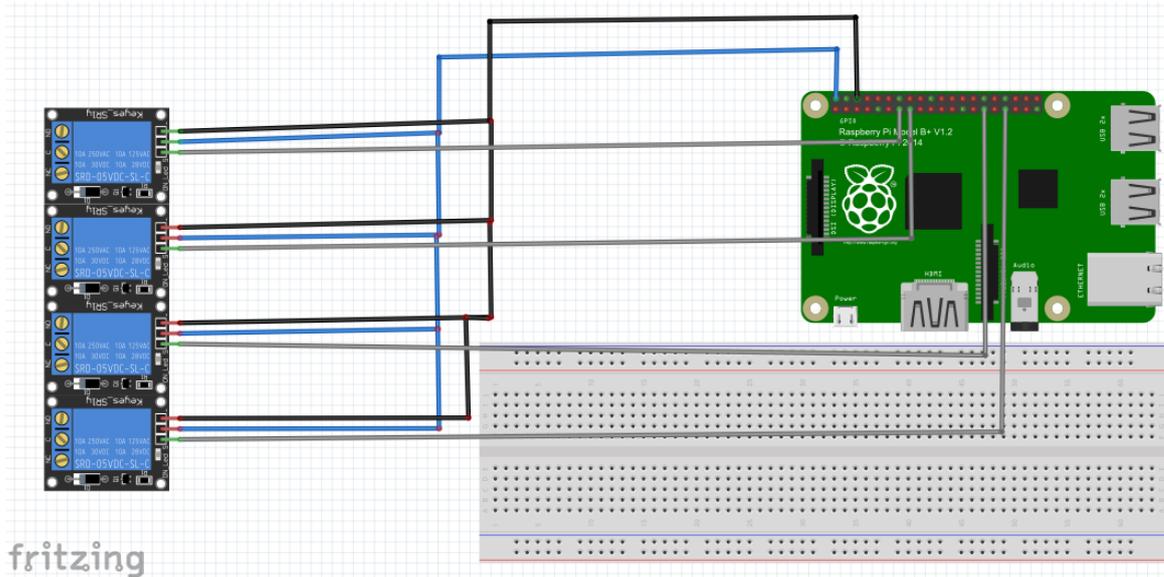


Ilustración 41: Montaje placa de relé con Raspberry

El esquema PCB correspondiente a la imagen anterior en la siguiente:

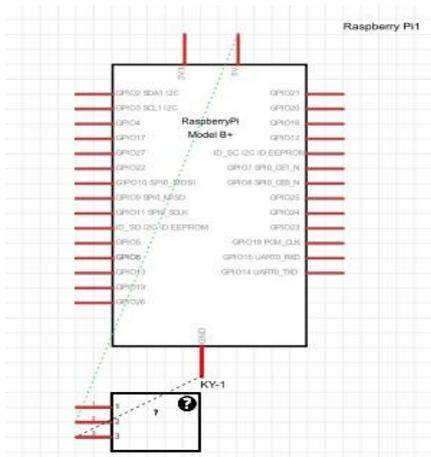


Ilustración 42: Montaje PCB placa de relé con Raspberry

4.3.3 Conexión Raspberry b+ con convertidor analógico-digital MCP3008

Como Raspberry no está provista de entradas ni salidas analógicas para el uso de estos sensores, este CHIP MCP3008 da la posibilidad de integrar entradas y salidas analógicas en dicha placa. Este chip ofrece 8 canales para el control de 8 salidas o entradas analógicas. En el cual solo 4 serán ocupadas por los sensores de corriente.

En la siguiente imagen se va a utilizar Pi Cobbler + (es una extensión de los pines de Raspberry para mejorar el conexionado en una *protoboard*²) ya que sin esta placa extensión de pines tendría que mantenerse en una forma inadecuada e inestable donde cada componente quedaría de forma libre como puede ser el caso del chip MCP 3008 que sin esta placa sería muy complicado realizar dichas conexiones.

<u>Raspberry b+</u>		<u>MCP 3008</u>
Pin 3,3V	-----	VDD
Pin 3,3V	-----	VREF
Pin GND	-----	AGND
Pin ce0	-----	CLK
Gpio MOSI	-----	DOUT
Gpio MISO	-----	DIN
Gpio SCLK	-----	CS
Pin GND	-----	DGND

Por lo tanto este chip ADC debe estar alimentado con 3.3V con su respectiva tierra GND. Además de la conexión de alimentación están conectados los pines SPI para el cambio de lectura de analógico a digital, esta relación entre los pines del MCP3008 permite transformar la señal analógica del sensor de corriente a digital para ser monitorizada por la central de gestión. Como este chip permite hasta 8 lecturas y estas están colocadas desde el CH0 al CH7 como se puede ver en la imagen posterior, los sensores de corriente ocuparán las plazas del CH0 al CH3.

² PROTOBOARD: es un tablero con orificios conectados eléctricamente entre sí, habitualmente siguiendo patrones de líneas, en el cual se pueden insertar componentes electrónicos y cables para el armado y prototipado de circuitos electrónicos y sistemas similares

CONTROL REMOTO DE ILUMINACION CON MICROCONTROLADORES Y SERVICIO WEB

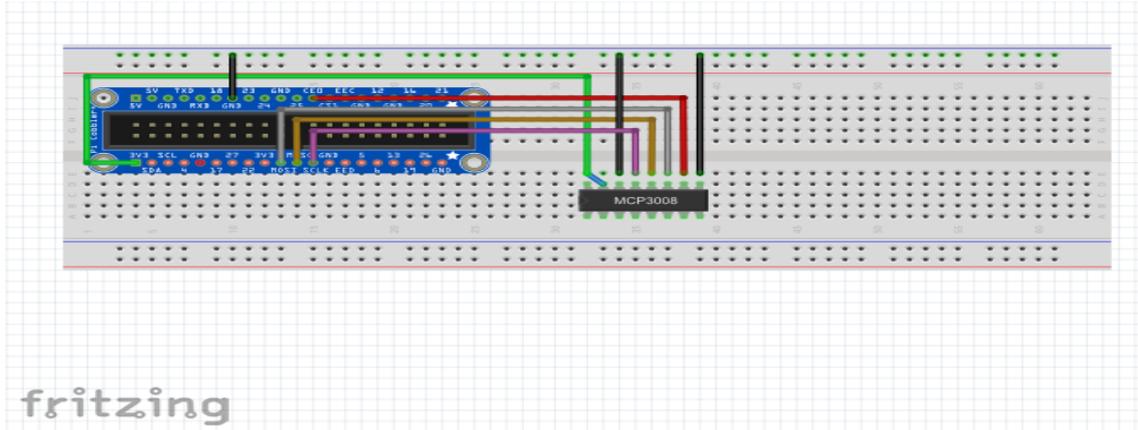


Ilustración 43: Montaje ADC MCP3008 y Pi Cobbler +

El esquema PCB correspondiente a Pi Cobbler + con el convertidor analógico digital MCP 3008. Con las respectivas conexiones de alimentación y conexión SPI para el cambio de analógico-digital.

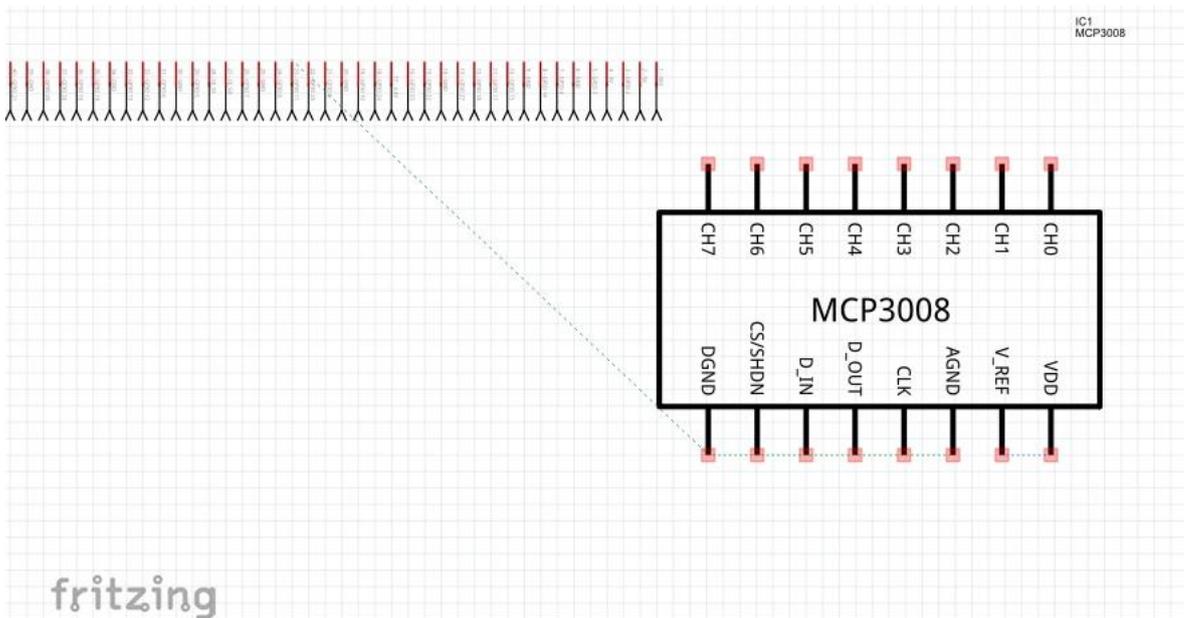


Ilustración 44: Montaje PCB de Pi Cobbler + y ADC MCP 3008

4.3.4 Conexión Raspberry b+ con sensor analógico y sensor de corriente

En el caso anterior del conexionado entre Pi Cobbler y el MCP 3008 es la unión entre Raspberry pi y el convertidor analógico-digital. Por lo tanto Raspberry ya dispone de entradas y salidas analógicas.

Al disponer de entradas analógicas el sensor de corriente ACS 714 enviará la lectura analógica de corriente al ordenador Raspberry y está al usuario del estado de la luz en cada momento.

Entre los diferentes sensores el ACS 714 permite reconocer hasta 30A en CA y CC en la fase de donde se encuentre dicho sensor en la instalación.

Raspberry b+

Pin 5V

ACS 714

VDD

Pin GND

GND

MCP 3008

Pin CH0

ACS 714

OUT

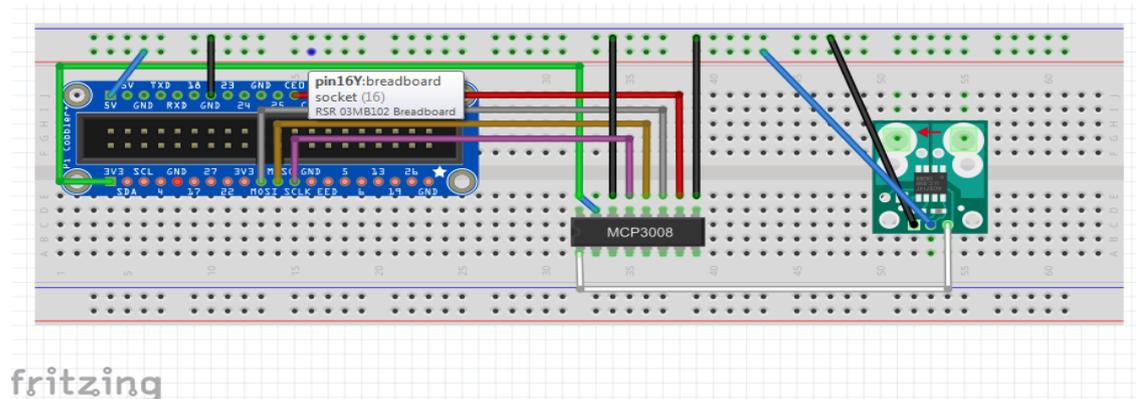


Ilustración 45: Montaje Pi Cobbler + y ADC MCP 3008 con sensor ACS714

En esta parte del circuito están colocados los diferentes sensores de corriente en los canales de lectura del MCP3008 del CH0 al CH3 al tener cuatro sensores de corriente para nuestro módulo de 4 relés. Dichos sensores de corriente tienen 3 pines de conexión de los cuales dos son para alimentarlo con 5V y su respectiva tierra GND, y el restante es el pin OUT que hace la lectura de corriente cuando se lo solicita la central de gestión.

En la siguiente imagen se pueden ver los diferentes canales de lectura del MCP3008 que corresponden a 8.

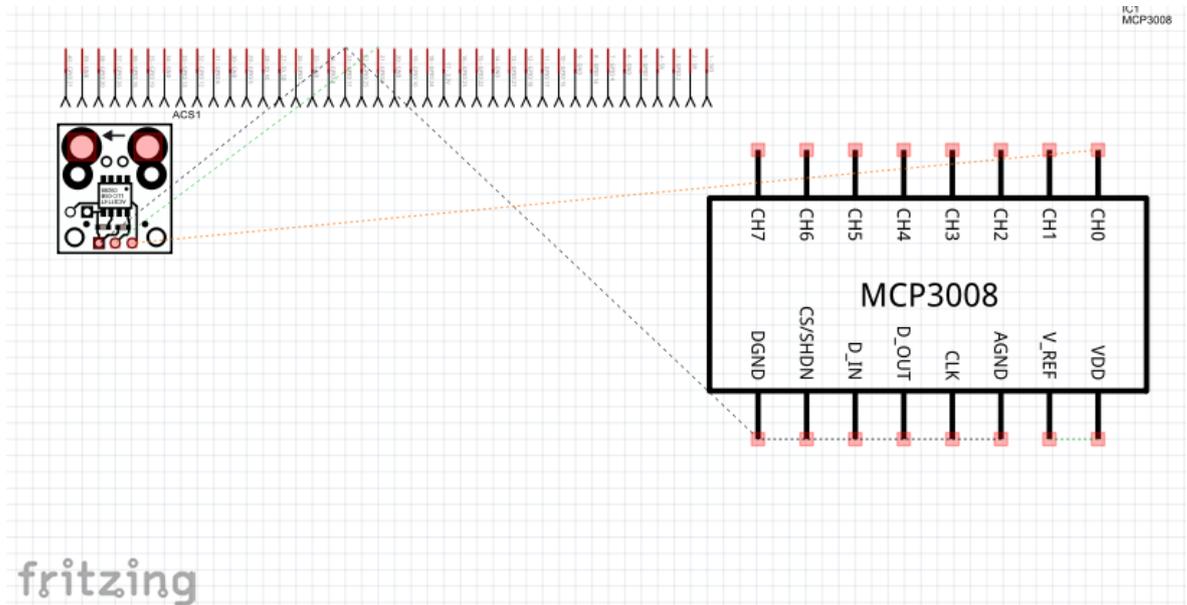


Ilustración 46: Montaje PCB de Pi Cobbler + y ADC MCP 3008 con sensor ACS714

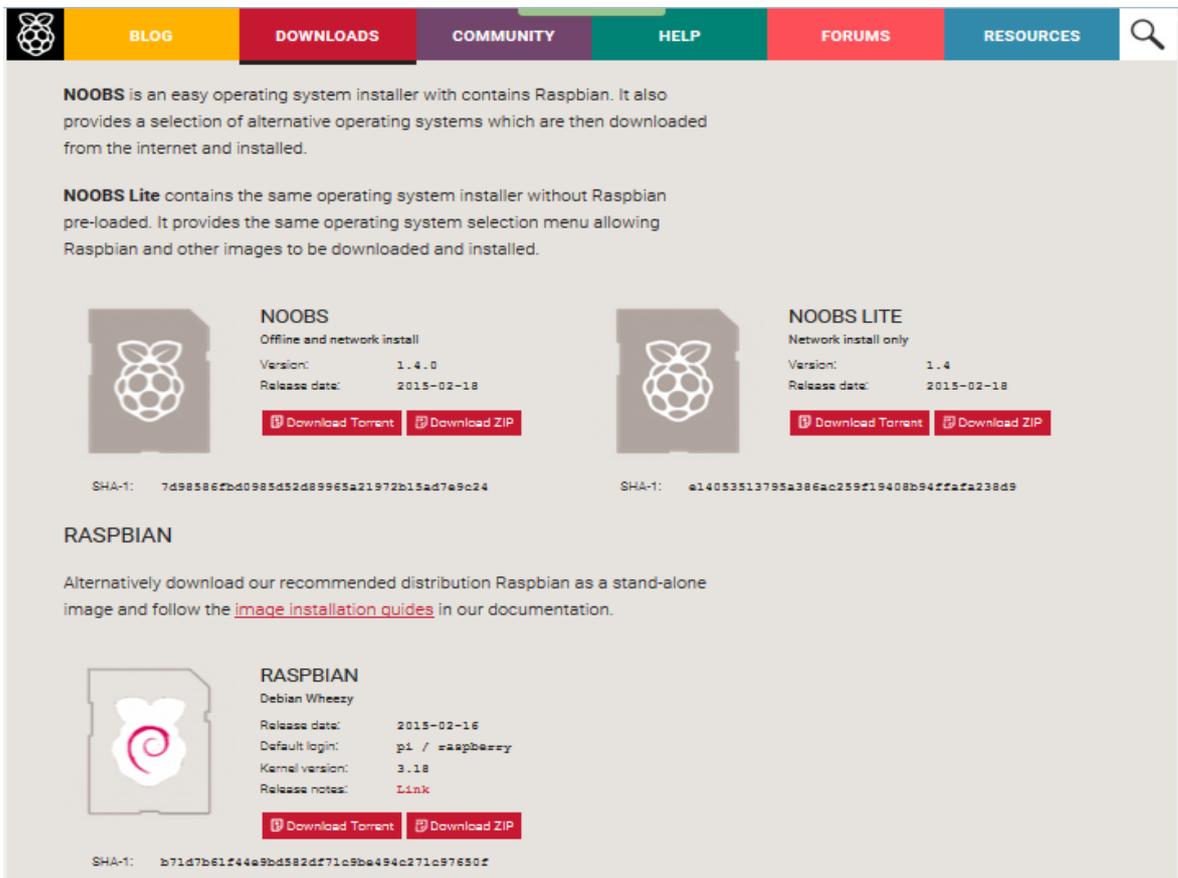
4.4 Instalación y Configuración de Raspbian

La configuración del sistema operativo, en el cual van a estar los diferentes programas que controlan el módulo de relés de la instalación.

Se puede instalar RASPBIAN de dos formas.

- 1) Mediante imagen en la SD.
- 2) Descargando la versión offline (actualización completa última versión).

El sistema operativo se encuentra en la página oficial de Raspberry.



The screenshot shows the 'Downloads' section of the Raspberry Pi website. It features a navigation bar with links for BLOG, DOWNLOADS, COMMUNITY, HELP, FORUMS, and RESOURCES. Below the navigation bar, there are three main download options:

- NOOBS**: Offline and network install. Version: 1.4.0, Release date: 2015-02-18. Includes 'Download Torrent' and 'Download ZIP' buttons. SHA-1: 7d98586fbd0985d52d89965a21972b15ad7a9c24.
- NOOBS LITE**: Network install only. Version: 1.4, Release date: 2015-02-18. Includes 'Download Torrent' and 'Download ZIP' buttons. SHA-1: e14053513795a386ac259f19408b94ffafa238d9.
- RASPBIAN**: Debian Wheezy. Release date: 2015-02-16, Default login: pi / raspberrypi, Kernel version: 3.18. Includes 'Download Torrent' and 'Download ZIP' buttons. SHA-1: b71d7b61f44e9bd582df71c9ba494c271c97650f.

Below the RASPBIAN section, there is a note: 'Alternatively download our recommended distribution Raspbian as a stand-alone image and follow the [image installation guides](#) in our documentation.'

Ilustración 47: Pagina Oficial Sistema Operativo Raspbian

De entre todos los sistemas operativos disponibles se utilizará Raspbian ya que cumple con las expectativas.

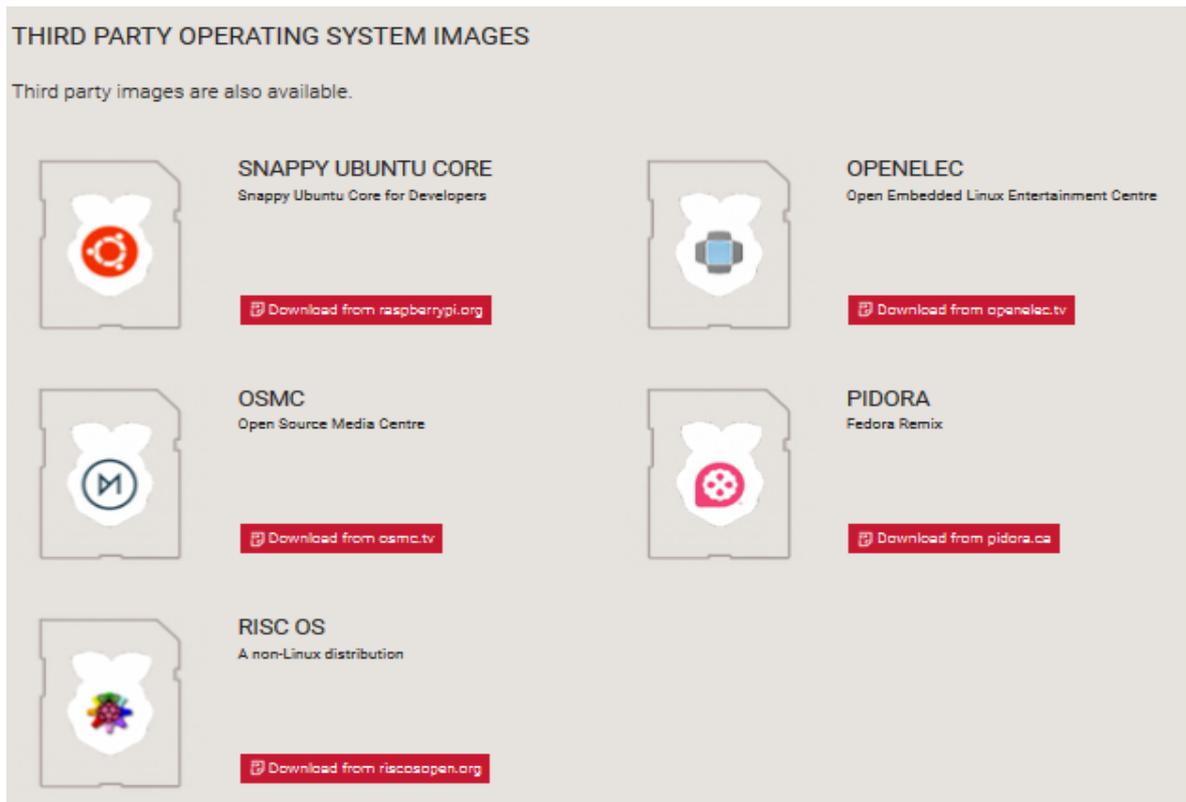


Ilustración 48: Otros sistemas operativos para Raspberry

La primera imagen de la instalación de Raspbian antes de la configuración del sistema.

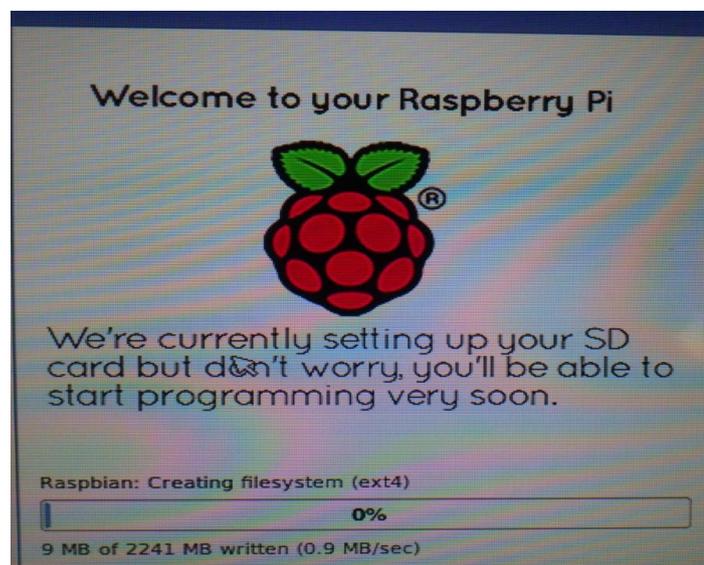


Ilustración 49: Primer Inicio en Raspberry

Una vez cargado el sistema y los primeros controladores del kernel, empezará con la configuración.

4.4.1 Configuración Raspbian

Se aplicará la configuración con respecto las necesidades del proyecto.

Aquí se ven detalladas las diferentes opciones para la configuración.

Los comandos para configurar el sistema son:

```
sudo raspi-config
```

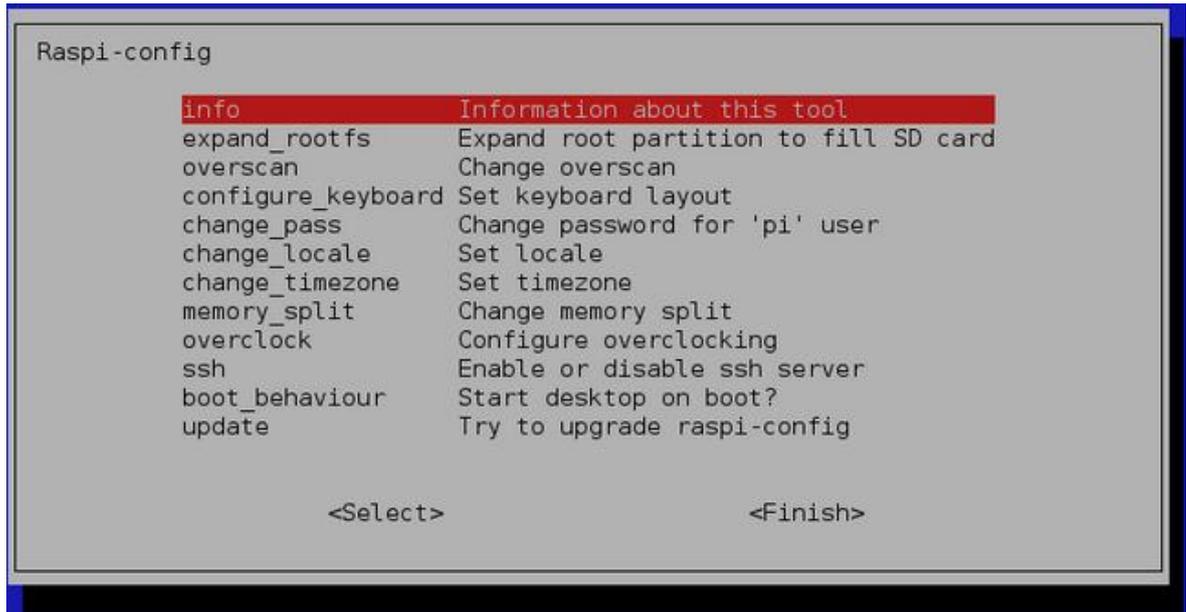


Ilustración 50: Menú de configuración de Raspberry

Una vez iniciada se mostrará una imagen como el menú correspondiente a la imagen de arriba.

Una vez iniciado el acceso a Raspberry se deberán configurar los diferentes campos según el usuario. A continuación en los siguientes apartados se detallan los diferentes campos de configuración del sistema en Raspberry:

4.4.1.1 Expand root partition to fill SD card

Se deberán utilizar tarjetas de 4Gb ya que el tamaño mínimo adecuado para dicho software como es Raspbian. En el caso de usar SD de mayor tamaño de deberá expandir la partición la imagen en la sd para ocupar todo su espacio.

4.4.1.2 Set keyboard layout

La configuración del teclado dependerá del modelo de teclado usado. Si se está usando control remoto en la terminal SSH no hace falta configurarlo.

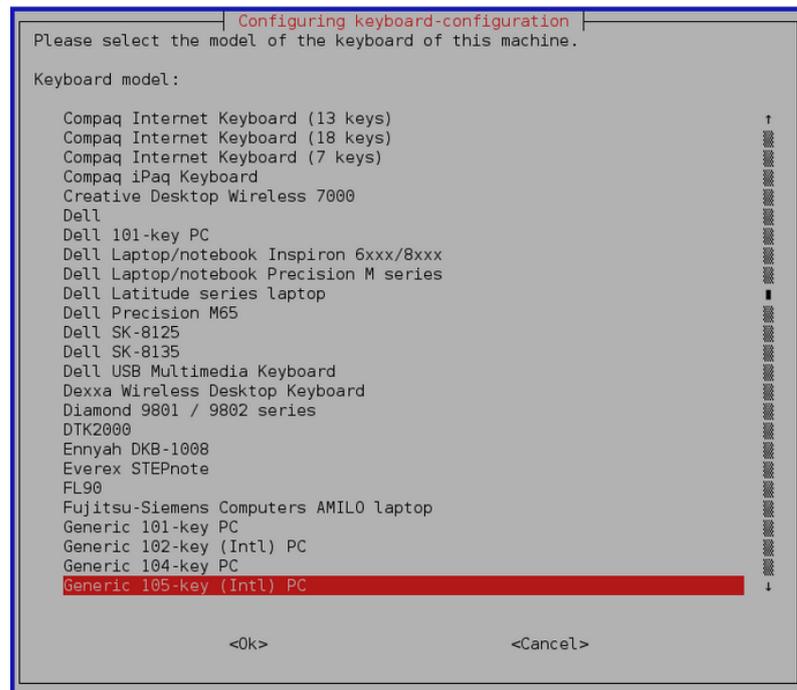


Ilustración 51: Configuración del teclado en Raspberry

4.4.1.3 Change password for 'pi' user

En este campo se puede modificar la contraseña y usuario de acceso a la placa.

```
Enter new UNIX password:  
Retype new UNIX password: █
```

Por seguridad se debe elegir una contraseña segura, para evitar posibles intrusos en la red.

4.4.1.4 Set locale y Set timezone

En este campo se debe configurar el sistema con el juego de caracteres del teclado y la zona horaria correspondiente.

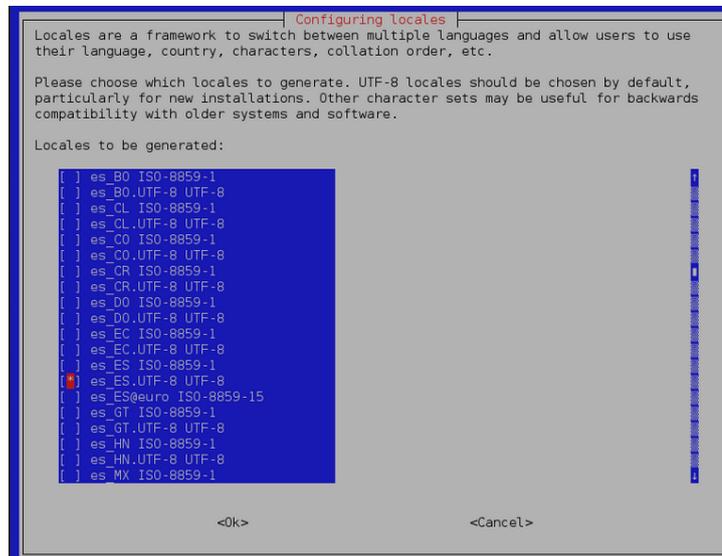


Ilustración 52: Configuración del teclado en Raspberry

En el caso de ser un teclado español se recomienda elegir `es_ES UTF-8`. También se deberá escoger la zona horaria donde se encuentre la placa.

4.4.1.5 Configurar la pila TCP/IP manualmente

Aunque no sea una de las opciones del configurador, puede resultar importante poder asignar una IP y otros parámetros de la red a Raspberry de forma manual. Para asignar estos valores, escribir estos comandos por consola:

```
sudo nano /etc/network/interfaces
```

Y modificar los siguientes apartados según la configuración que desee el usuario:

```
Auto eth0
iface eth0 inet static
address 192.168.0.***
netmask 255.255.255.0
gateway 192.168.0.1
```

```
GNU nano 2.2.6 Fichero: /etc/network/inte
auto lo
iface lo inet loopback

allow-hotplug wlan0
iface wlan0 inet manual
wpa-roam /etc/wpa_supplicant/wpa_supplicant.conf
iface default inet dhcp

auto eth0
iface eth0 inet static
address 192.168.0.18
netmask 255.255.255.0
gateway 192.168.0.1
```

Ilustración 53: Configuración IP en Raspberry

4.4.1.6 Actualizar sistema RASPBIAN

En este apartado se actualizarán y se reiniciara el sistema con los campos modificados anteriormente.

```
sudo apt-get update
sudo reboot
```

4.5 Raspberry como Servidor Web

En la actualidad, el desarrollo de aplicaciones web se debe contar con un servidor HTTP (Protocolo de Transferencia de Hipertexto) instalado en algún ordenador o PC, ya que éste es el encargado de enviar la información al cliente que la solicite, el cual a su vez se muestra dicha información como una página web. La mejor opción para contar con un servidor gratuito es utilizar Raspberry Pi, la cual se puede configurar para que actúe como servidor HTTP tanto para transferencia de archivos como para servidor de páginas web de la aplicación web.

Por lo tanto se creará un servidor que contendrá los archivos correspondientes a la web desarrollada para el control de luces conectadas a la placa que contendrá la información de los sensores de corriente y el uso del módulo de relés.

Un servidor web es un programa que se ejecuta continuamente en un segundo plano en un ordenador, manteniéndose a la espera de peticiones de ejecución que le hará un cliente para solicitar alguna información, donde el servidor se encarga de contestar a estas peticiones de los clientes de forma ordenada y adecuada, entregando como resultado una página o información dependiendo de los comandos solicitados por el cliente.

Aunque Raspberry Pi tiene menos prestaciones que un PC de escritorio, no significa que su rendimiento se vea afectado para utilizarse como un servidor web en un ámbito educativo o de negocios. A pesar de disponer de 512 MB de memoria RAM y la relativa poca potencia del procesador, el bajo consumo de energía y el bajo costo de la Raspberry Pi hacen que sea una herramienta de trabajo muy buena para el uso de páginas estáticas o dinámicas a nivel de red local o incluso a través de Internet. Existen diferentes variaciones para lograr que la Raspberry Pi funcione como un servidor web, la forma menos compleja consiste en instalar alguno de los servidores web más tradicionales, como Apache.

Una segunda opción es utilizar los diferentes lenguajes de programación como Python, Ruby o Java Script para crear nuevos servidores web, donde de una forma automática generen el HTML cuando reciben de los clientes en HTTP desde un navegador web. Para la creación del servidor web, se va a utilizar LAMP.



Ilustración 54: Logotipo Pila LAMP

LAMP está formado por:

- Linux
- Apache
- Mysql
- PHP

4.5.1 Instalación de la Pila LAMP

El siguiente paso es instalar los componentes restantes: Apache, MySQL y PHP. En la terminal o consola, para ellos se van a instalar los diferentes paquetes correspondientes:

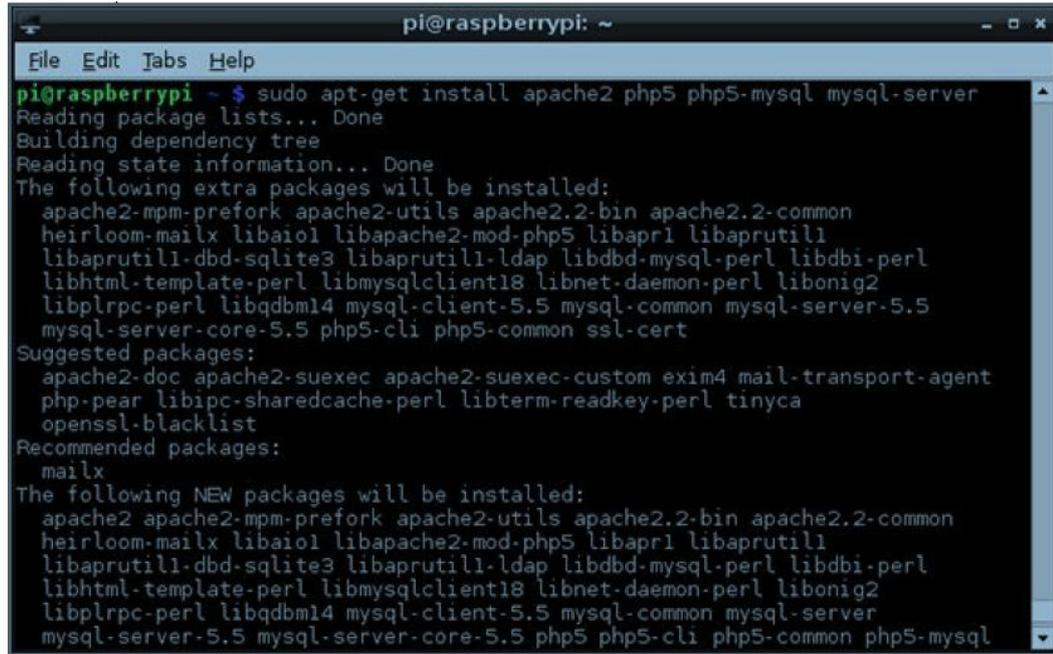
```
sudo apt-get update
```

```
sudo apt-get install apache2 php5 php5-mysql mysqlq-server
```

Estos comandos en la terminal consiguen que el gestor de paquetes apt obtenga una serie de dependencias necesarias para conseguir la ejecución totalmente funcional de la pila LAMP.

La instalación completa de la pila LAMP puede tardar bastante tiempo sobre la Raspberry durante el proceso de instalación, MySQL le solicitará escribir una contraseña.

Asegúrese de elegir una contraseña segura, ya que ésta protegerá a la base de datos MySQL, la cual, de acuerdo con servidor web que se vaya a utilizar, podrá almacenar nombres de usuario, detalles de pago y otra clase de información de identificación personal.



```
pi@raspberrypi: ~  
File Edit Tabs Help  
pi@raspberrypi ~ $ sudo apt-get install apache2 php5 php5-mysql mysql-server  
Reading package lists... Done  
Building dependency tree  
Reading state information... Done  
The following extra packages will be installed:  
  apache2-mpm-prefork apache2-utils apache2.2-bin apache2.2-common  
  heirloom-mailx libaio1 libapache2-mod-php5 libapr1 libaprutil1  
  libaprutil1-dbd-sqlite3 libaprutil1-ldap libdbd-mysql-perl libdbi-perl  
  libhtml-template-perl libmysqlclient18 libnet-daemon-perl libonig2  
  libplrpc-perl libqdbm14 mysql-client-5.5 mysql-common mysql-server-5.5  
  mysql-server-core-5.5 php5-cli php5-common ssl-cert  
Suggested packages:  
  apache2-doc apache2-suexec apache2-suexec-custom exim4 mail-transport-agent  
  php-pear libipc-sharedcache-perl libterm-readkey-perl tinycsa  
  openssl-blacklist  
Recommended packages:  
  mailx  
The following NEW packages will be installed:  
  apache2 apache2-mpm-prefork apache2-utils apache2.2-bin apache2.2-common  
  heirloom-mailx libaio1 libapache2-mod-php5 libapr1 libaprutil1  
  libaprutil1-dbd-sqlite3 libaprutil1-ldap libdbd-mysql-perl libdbi-perl  
  libhtml-template-perl libmysqlclient18 libnet-daemon-perl libonig2  
  libplrpc-perl libqdbm14 mysql-client-5.5 mysql-common mysql-server  
  mysql-server-5.5 mysql-server-core-5.5 php5 php5-cli php5-common php5-mysql
```

Ilustración 55: Instalación Apache en Raspberry

Una vez finalizada la instalación del software, tanto el servidor MySQL como el Apache se ejecutarán en segundo plano. Para comprobar que los servidores estén funcionando correctamente.

4.5.2 Conexión ssh

Poder controlar la Raspberry Pi desde cualquier lugar se utilizará a través de SSH [14], este programa facilita el trabajo, en el momento de desarrollar, instalar y configurar aplicaciones en dicha Raspberry, sin necesidad de tener conectado un TV o monitor.

Cuando se ejecute PuTTY con los parámetros necesarios se mostrará la pantalla del terminal.

Para descargar el programa PuTTY [14], que nos servirá para establecer una conexión entre Raspberry Pi, a través de SSH y el Pc o móvil.

El programa se puede descargar desde:

<http://www.putty.org>

Para la conexión de Ssh de utilizará el puerto 22 pero se puede utilizar otro a partir del puerto 1024.

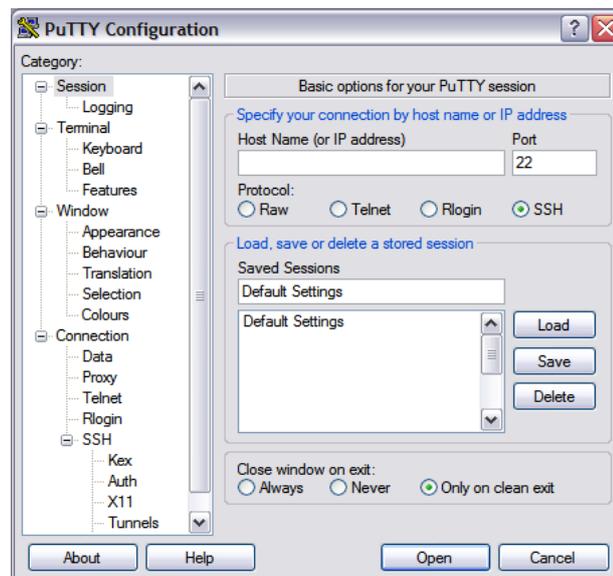


Ilustración 56: Configuración de PUTYY



Ilustración 57: Otorgamiento de permisos para PUTTY

Una vez iniciada la conexión Ssh se deberán utilizar el usuario y contraseña que por defecto lleva de serie Raspberry pi Usuario: pi y Password: raspberry .

El cambio de contraseña en Raspberry se puede realizar en el apartado de seguridad 4.4.1.3 Change password.

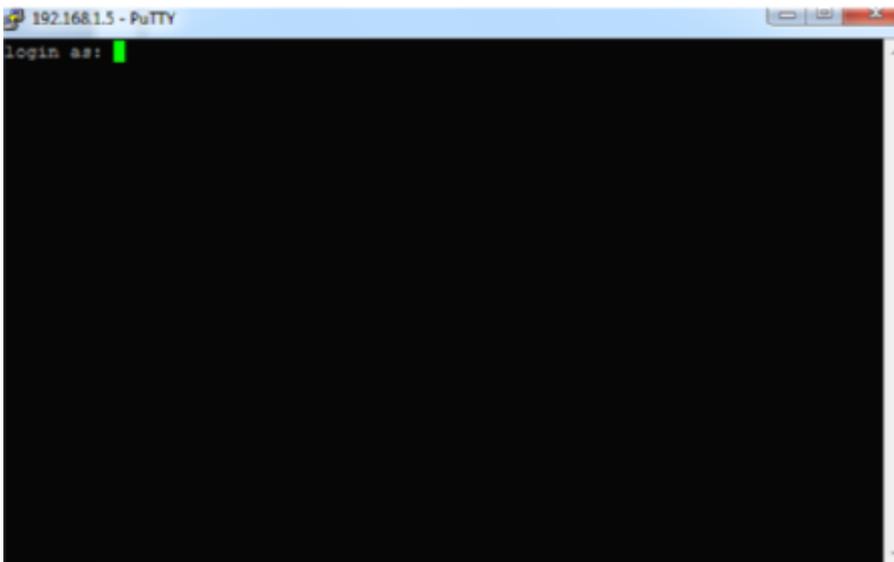
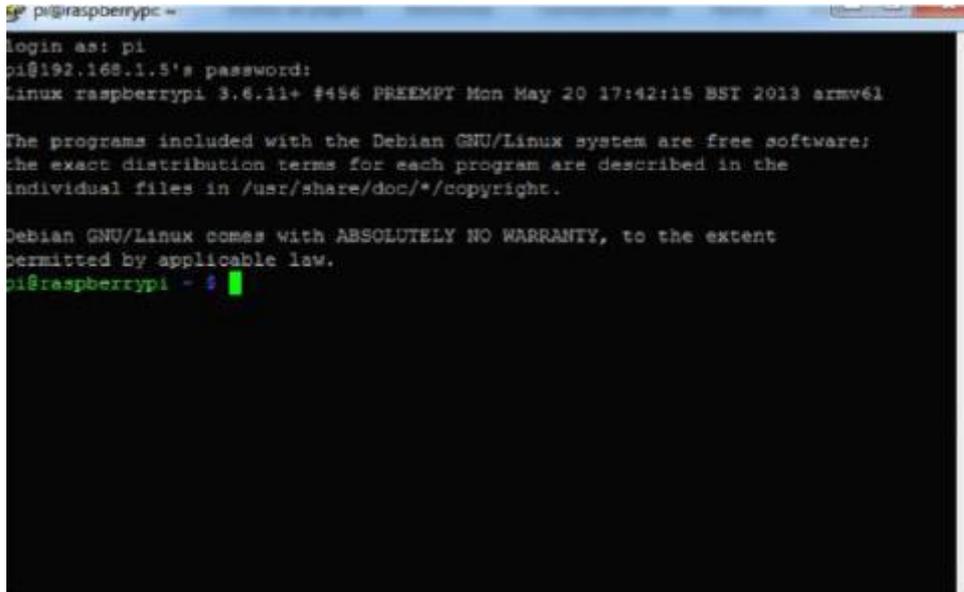


Ilustración 58: Usuario y Contraseña de Raspberry para control remoto



```
pi@raspberrypi ~$
login as: pi
pi@192.168.1.5's password:
Linux raspberrypi 3.6.11+ #456 PREEMPT Mon May 20 17:42:15 BST 2013 armv6l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
pi@raspberrypi ~$
```

Ilustración 59: Pantalla de la Terminal en Raspberry

Una vez iniciada la sesión con Ssh ya se procederá a la otorgación de permisos a las diferentes carpetas que se van a utilizar.

Otorgamiento de permisos

Los directorios típicamente utilizados por un servidor web en Linux se sitúan en `/var/www`, y el usuario típico para dicho entorno suele ser `www-data`. Ahora vamos a crear el grupo y usuario estándar para nuestro servidor, a la par que otorgamos los permisos pertinentes y añadimos a nuestro usuario por defecto (`pi`) al grupo comentado. De esta forma no será preciso que el usuario `root` (`su`) sea siempre el que pueda operar en `/var/www`.

Para otorgar permisos a la carpeta `/var/www` que es donde estará nuestro programa para raspberry escribiremos en el terminal:

```
sudo chown www-data:www-data /var/www/
```

Dar los permisos a la carpeta `www/`

```
sudo chmod 775 /var/www/
```

Añadir el usuario pi al grupo www-data

```
sudo usermod -a -G www-data pi
```

```
sudo visudo
```

Agregar el siguiente código al final.

```
www-data ALL=(root) NOPASSWD:ALL
```

Reiniciar el Servidor

```
sudo /etc/init.d/apache2 restart
```

Para saber si el servidor ya se encuentra preparado para funcionar, se le realizará una prueba.



Ilustración 60: Imagen del servidor Apache en funcionamiento

4.6 Entorno Gráfico con VNC para Conexión Remota

Para obtener un entorno gráfico en el PC o móvil para la modificación o control en Raspberry dependeremos de VNC.

Este permite el control remoto desde cualquier lugar con acceso a internet.

4.6.1 Instalación y configuración Real VNC

Se deberá de tener instalado dicho programa en los diferentes puntos desde donde se quiere tener acceso sea móvil o PC.

<http://www.realvnc.com/download/viewer>

Aquí se ve la imagen de acceso al control remoto desde VNC.

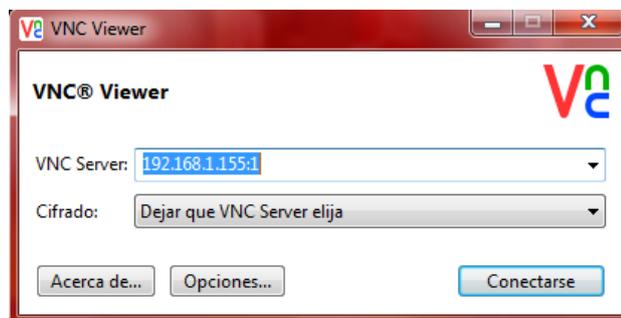


Ilustración 61: Acceso mediante VNC por Ip

Esta imagen se corresponde al acceso permitido en Raspberry.

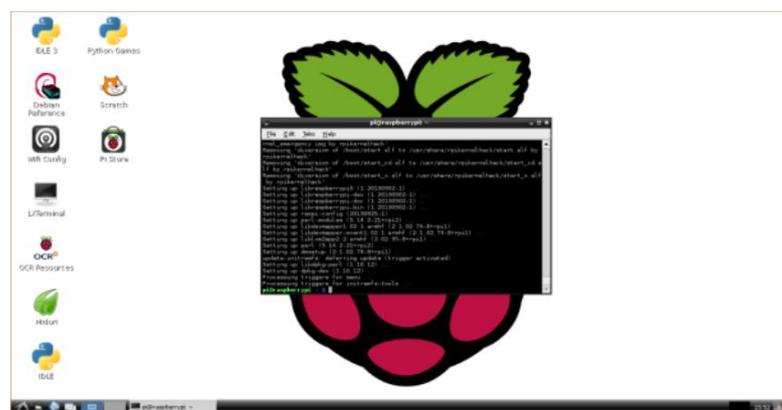


Ilustración 62: Imagen del control remoto desde VNC.

4.7 Conociendo el GPIO de Raspberry Pi

La Raspberry Pi, además de contar con puertos USB, HDMI y ETHERNET tiene a su disposición una serie de pines de propósito general, como son entradas y salidas digitales entre otras. Raspbian ofrece unas librerías que posibilitan la interface GPIO [12] en la raíz de la Sd `/sys/class/gpio`.

Este directorio contiene tres tipos de ficheros:

Ficheros de Control de Interface: Los GPIOs están disponibles en la memoria del núcleo y para poder usarlos, hay que pedir al núcleo que exporte a la memoria del pin que se requiere usar. Por lo tanto los ficheros de control sirven para pedir al núcleo el uso de cierto GPIO y también para liberar o limpiar el mismo una vez haya terminado.

Los GPIOs: Aparecen en forma de directorios. Donde existen diferentes tipos para controlarlos ya sea mediante librerías BOARD nombra la numeración del pin y BCM el nombre del pin.

Controlador GPIO: Actúa como directorio con diversos ficheros que contienen información sobre el chip GPIO de la placa.

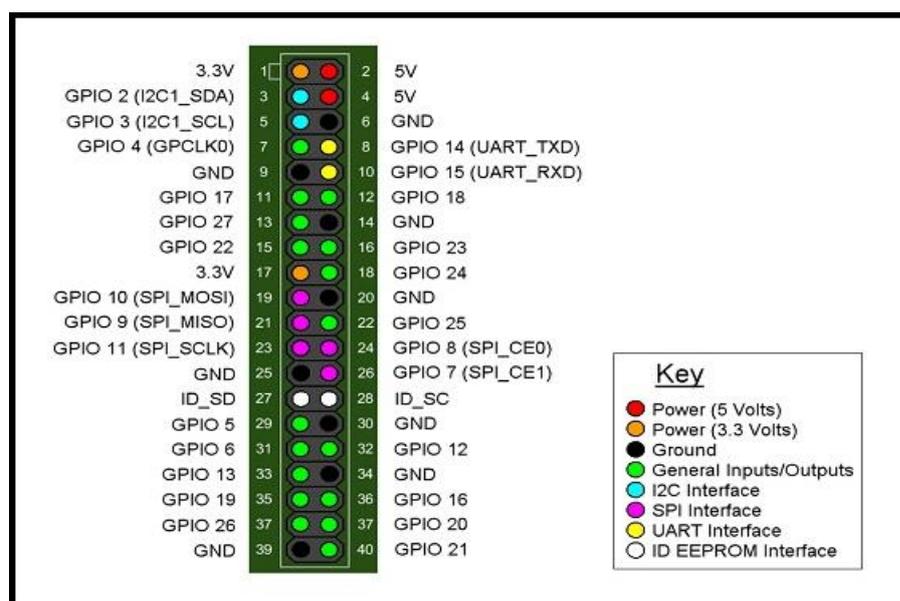


Ilustración 63: Puertos GPIO Raspberry B+ [7]

Conexión prueba de led

Para trabajar con el puerto GPIO, se utilizaran cables de pin donde en un extremo se conectara el GPIO correspondiente en Raspberry y del otro extremo se sacaran dos cables, uno sale del pin GPIO 4 para alimentar el led y una resistencia en serie de 220Ω y está a su vez va al ánodo del diodo led, y el otro del pin GND que va al cátodo del diodo led.

Para el control del led, desde la línea de comandos del terminal [5]; un ejemplo de uso sería este que pone el pin gpio 4 del bcm2835 en modo salida y luego le da el valor de 1 (HIGH) para encender y 0 (LOW) para apagar:

```
sudo -i  
echo "4" > /sys/class/gpio/export  
echo "out" > /sys/class/gpio/gpio4/direction  
echo "1" > /sys/class/gpio/gpio4/value  
echo "0" > /sys/class/gpio/gpio4/value
```

4.7.1 Controlando GPIO de la Raspberry Pi con Python

Python proporciona una librería con la que se puede interactuar con los pines del puerto GPIO, para disponer de esta librería, primero debemos instalarla para a continuación tener control sobre ella.

A continuación, se muestra el procedimiento que se debe llevar para su correcta instalación.

```
wgethttp://pypi.python.org/packages/source/R/RPi.GPIO/RPi.GPIO-0.1.0.tar.gz
```

```
tar xzf RPi.GPIO-0.1.0.tar.gz
```

```
cd RPi.GPIO-0.1.0
```

```
sudo python setup.py install
```

Para el control del pin GPIO4 que se encuentra en el pin 7 sudo python se debe aplicar dicho comando:

```
import RPi.GPIO as GPIO  
  
set mode(GPIO.BCM)  
  
GPIO.setup (7, GPIO.OUT)  
  
GPIO.output(7, False)  
  
GPIO.output(7, True)
```

4.7.2 Accediendo al GPIO de Raspberry Pi desde PHP

Para poder guardar dicho programa en PHP existe una función llamada *exec* la cual ejecuta un programa externo que ya tendremos programado anteriormente.

Ejemplo de uso.

El Script de Python necesario para prender o apagar los relés se guardarán con la función *exec* nombrada anteriormente.

El script para accionar procesa en PHP:

```
</script>  
<?php
```

Este script escrito en el lenguaje PHP (permite la creación de webs dinámicas, es decir, con un comportamiento variable de acuerdo a un algoritmo de programación), invoca un script escrito en el lenguaje Python que es el que permite manipular directamente los sensores y actuadores conectados a la placa Raspberry.

Además, es importante remarcar que el script `control_luz1.py` realizará medidas de la intensidad de corriente mediante el sensor (ACS 714), e informará sobre el estado actual: Paso o ausencia de corriente de la luz que se pretende controlar. Una problemática importante en este caso es que en este fragmento de código se necesita información generada por el script de Python invocado. Es decir, es necesario establecer un canal de comunicación entre ambos flujos de ejecución activos. A su vez, en función del estado aportado por el sensor, será necesario establecer una comunicación con la ventana anterior (la ventana principal) para decidir colorear de verde o rojo los botones y mostrar de esta forma al usuario el estado de la iluminación que se pretende controlar. Esto se hace mediante el código:

```
window.opener.document.getElementById("interruptor1").style.color="Green";
```

Con `window.opener` se hace referencia a la ventana desde donde se abrió la actual. Dese cuenta también, que en el resto de código de este archivo se combinan fragmentos de código en Javascript (para alterar el estado de la interfaz) y fragmentos de PHP para la evaluación de condiciones. Todo ello implica cierta complejidad.

Por último destacar la función `exec` de PHP que permite invocar y ejecutar cualquier script de Python, en este caso, uno alojado en el propio servidor. El segundo parámetro en este caso es una variable llamada `$output` que permite recoger valores generados por el script de Python, escritos directamente en la salida estándar. En este caso será la medida del sensor y el valor 1 en el caso de que se haya encendido la luz y 0 en el caso de que se haya apagado.

Por último, describiremos la última parte del proceso en el archivo `control_luz`

```
exec("sudo python /var/www/pi/control_luz1.py", $output, $v);

if ($output[0] == 1){
    //encendido
?>
    <script>

window.opener.document.getElementById("interruptor1").style.c
olor="Green";
        window.alert("SE ENCIENDE");
    </script>
<?php
}else{
    //apagado
?>
    <script>

window.opener.document.getElementById("interruptor1").style.c
olor="Red";
        window.alert("SE APAGA");
    </script>
<?php
}

?>
<script>
```

CONTROL REMOTO DE ILUMINACION CON MICROCONTROLADORES Y SERVICIO WEB

```
window.close();  
</script>
```

La placa consta de 4 relés por lo que deberá haber 8 script y 4 procesas php para dicho funcionamiento.

Tal y como se ve en la imagen.

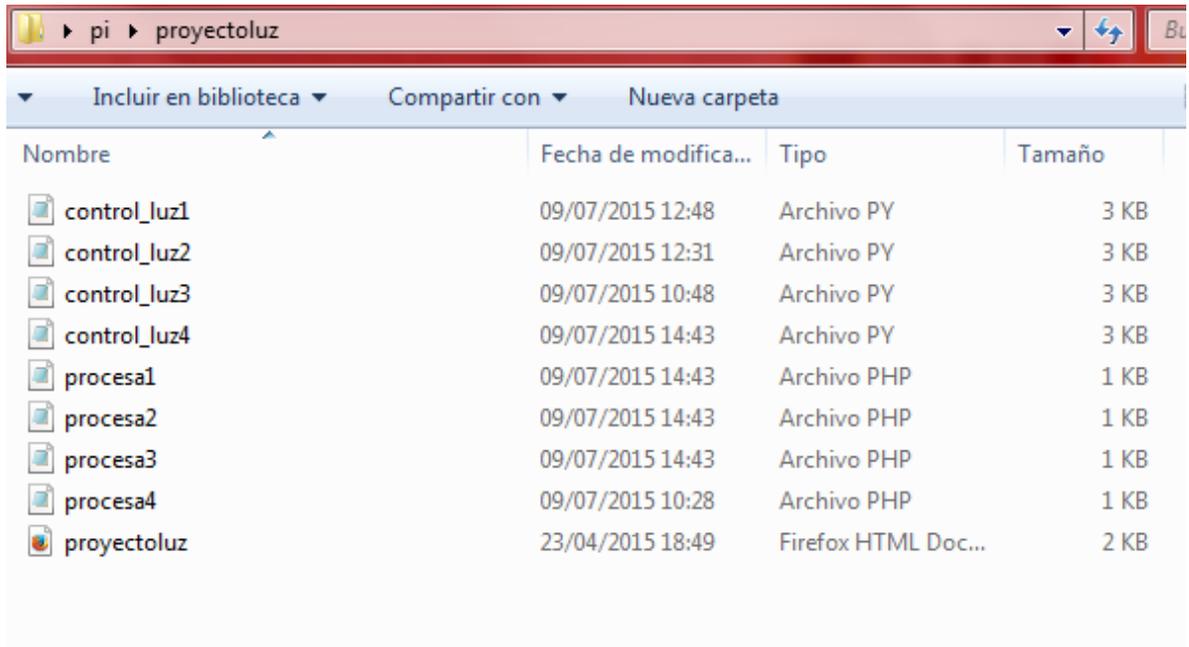


Ilustración 64: Programa para el control de luces mediante navegador WEB

4.8 Creación Página web

Estos diferentes programas se mostraran mediante una interfaz gráfica por web con java script [1].

La programación de la web es en HTML con funciones en PHP.

Esta interfaz gráfica está compuesta de los botones de apagado y encendido de cada relé.

Dentro de cada botón estará el script que hemos creado anteriormente tanto para encender como para apagar dichas luces.

Aquí esta detallada la programación en HTML de la página web que contendrá los script anteriormente citados para el uso de los relé.

```
<html>
<head>
<script>
//crear una variable por cada interruptor
//matriz de 2 interruptores (añadir tantos unos como interruptores haya)

//rojo: DC143C
//verde: 7FFF00

//función interruptor no hace falta tocar para nuevos interruptores
function interruptor(num) {

    window.open("procesa"+num+".php", "_blank");
    self.blur();
}
</script>
</head>
<body>
```

En el cuerpo de la página web se muestran cuatro botones, uno por cada una de las luces de la maqueta que se pretende controlar. El botón muestra de color verde la información textual para indicar que la luz, a la que pretende controlar, está encendida. Por el contrario, la fuente se mostrará de color rojo en el caso de que la luz esté apagada.

Cada vez que se presiona uno de los botones, se hace una llamada a la función interruptor situado al comienzo de este archivo. El problema es que la parte física del sistema con la placa Raspberry no se puede controlar directamente con un lenguaje de etiquetas como es HTML (lenguaje interpretable por los navegadores). Es necesario emplear otros lenguajes

de programación como python, que permite una ejecución de código local en el propio dispositivo. En este caso la placa Raspberry conectada a la red con el servidor WEB.

Por ello, la función interruptor de javascript, abre una nueva ventana con código PHP y este lenguaje ya sí que permite invocar código escrito en Python.

Por tanto y, en resumen, para alcanzar el objetivo de controlar un dispositivo físico a través de una página web, es necesario combinar cuatro tecnologías relativas a los lenguajes de programación: 1) HTML -> 2) Javascript -> 3) PHP y 4) Python. En los siguientes archivos (procesa y control luz) se explicará con más detalle este proceso.

```
<center><h1>Encendido y Apagado de interruptores</h1></center>

<!-- interruptor 1 en pin 7 e interruptor 2 en pin 11 -->
<input type="button" id="interruptor1" value="Interruptor 1: Pin
7" onClick="javascript:interruptor(1);" />
</br>
<input type="button" id="interruptor2" value="Interruptor 2: Pin
11" onClick="interruptor(2);" />
</br>
<input type="button" id="interruptor3" value="Interruptor 3: Pin
13" onClick="interruptor(3);" />
</br>
<input type="button" id="interruptor4" value="Interruptor 4: Pin
15" onClick="interruptor(4);" />
</br>
```

Inicialmente se muestran todos los botones de color rojo para indicar que todas las luces están apagadas.

```
<script>
document.getElementById("interruptor1").style.color="Red";
document.getElementById("interruptor2").style.color="Red";
document.getElementById("interruptor3").style.color="Red";
document.getElementById("interruptor4").style.color="Red";
</script>
</body>
</html>
```

El resultado de este programa en HTML visto desde un navegador web:



Encendido y Apagado de interruptores

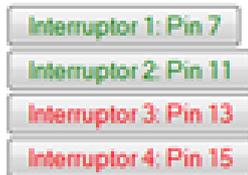


Ilustración 65: Imagen de la web del control de luces

La maqueta donde se ha instalado el sistema de control de iluminación con servicio web consta de 4 zonas, donde la central de gestión controlará cada zona que está compuesta del sistema de iluminación, con su correspondiente llave de luz conmutada con el relé seleccionado para esa zona, además del sensor de corriente que monitorizará el estado de la luz en cada zona de la maqueta.

Para emular la situación del cuadro de luz de la un hogar o local se optó por el uso de una regleta donde cada enchufe emulará las PIAS del cuadro de luces del hogar.

El materia con el que se ha construido la maqueta es DM un material más duradero que el aglomerado. El cual se eligió para unas posibles mejoras en un futuro o para diferentes proyectos que se puedan realizar.

Las herramientas utilizadas han sido una sierra caladora, una sierra ingletadora y una caja de tornillos de 0.3 cm de diámetro y 4 cm de largo.

En la ilustración 66 de las 4 zonas se han iluminado las dos primeras mediante control remoto por internet como se puede apreciar en la imagen, y en la ilustración 67 se puede ver el accionamiento de las dos zonas activadas citadas anteriormente por el control remoto mediante el servidor web, mientras que las dos zonas restantes no han sido activadas como se pueden ver en las ilustraciones 66 y 67.



Ilustración 66: Maqueta iluminada por accionamiento web.

En la imagen posterior se puede ver parte de la instalación del control remoto mediante el servidor web donde se pueden apreciar parte de los componentes principales.

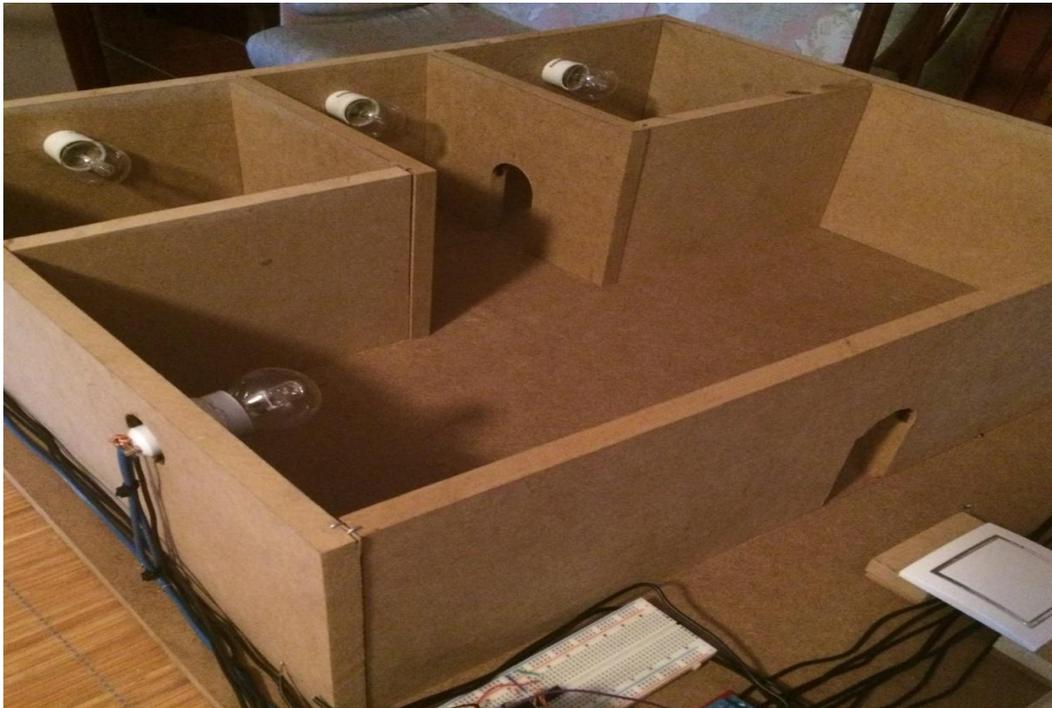


Ilustración 67: Maqueta de la instalación de control remoto.

CAPÍTULO 5: ESTUDIO ECONÓMICO

5 ESTUDIO ECONÓMICO

Los resultados obtenidos en la realización del proyecto vienen determinados por la rapidez del accionamiento del servidor. Es decir, se probó el tiempo de accionamiento del apagado y encendido de las luces por el control remoto mediante servidor web, obteniendo el tiempo que tarda la central de gestión en obedecer las órdenes del usuario. Para ello se ha calculado el tiempo de accionamiento que tarda desde que el usuario envía la orden hasta que la central de gestión actúa sobre el control de las luces. Obteniendo los resultados de distintas ciudades y pueblos de España.

CONEXIÓN RASPBERRY PUERTOLLANO		
	DISTANCIA KM	Tiempo de Accionamiento del servidor para apagado y encendido de las luces
CONEXIÓN PC	Madrid	196 < 3.400.000 microsegundos
	Ciudad Real	37 < 2.700.000 microsegundos
	Puertollano	0 < 800.000 microsegundos
	Almodovar	6 < 2.400.000 microsegundos
	Almadén	65 < 3.100.000 microsegundos
	Argamasilla De Calatarva	6 < 2.300.000 microsegundos

Para la elaboración de este proyecto se pretendió desde un primer momento hacerlo dentro de lo más real y más económico posible para usarlo en una vivienda o local de España. Se hizo un estudio de todos los componentes que deben componer la realización del control de luces y que cumplan las características deseadas. Una vez que se realizó la lista de los componentes necesarios para dicho proyecto se buscó en diferentes tiendas especializadas en estos campos los mejores precios para hacerlo lo más barato. A continuación se detalla una lista donde aparecen los diferentes componentes buscados en diferentes tiendas online de España.

Placa Raspberry Pi B+:

- ✓ Raspberry Pi B+ 40,66€ sin gastos de envío o 4€ envío exprés;

<http://tienda.bricogeek.com/placas-raspberry-pi/717-raspberry-pi-model-b.html>

- ✓ Raspberry Pi B+ 37.32€ sin gastos de envío o 9€ envío exprés;

http://www.ebay.es/itm/MB-Raspberry-Pi-Model-B-512MB-RAM-/191561432468?pt=LH_DefaultDomain_77&hash=item2c99f37d94

Sensor de corriente ACS:

- ✓ ACS714 10,89€ sin gastos de envío

<http://tienda.bricogeek.com/sensores/367-sensor-de-corriente-acs714-30a.html>

Convertidor Analógico-Digital MCP3008:

- ✓ **Convertidor Analógico-Digital MCP3008** 8.41€ sin gastos de envío

[http://tienda.bricogeek.com/bricogeek/678-conversor-adc-8-bits-mcp3008.html /](http://tienda.bricogeek.com/bricogeek/678-conversor-adc-8-bits-mcp3008.html/)

Módulo de 4 relés:

- ✓ **Módulo de 4 relés** □ □ KEYES 6.05€ sin gastos de envío

http://www.dx.com/es/p/4-ch-12v-power-relay-module-red-blue-173827?tc=EUR&gclid=CjwKEAajw96aqBRDNhM6MtJfE-wYSJADiMfggJY-1ziT-1vior0japelR6HOcMXNQ42Ii_8f6DWEw7BoCJGHw_wcB

- ✓ **Módulo de 4 relés** □ □ KEYES 14,97€ sin gastos de envío

<http://www.cetronic.es/sqlcommerce/disenos/plantilla1/seccion/producto/DetalleProducto.jsp?idIdioma=&idTienda=93&codProducto=888304018&cPath=1342>

WIFI USB:

- ✓ **WIFI USB APPUSB150DAV3 ADAPTADOR WIFI USB 150Mb WL 4dbi**

<http://www.cetronic.es/sqlcommerce/disenos/plantilla1/seccion/producto/DetalleProducto.jsp?idIdioma=&idTienda=93&codProducto=255191068&cPath=812>

Tarjeta d memoria SD Kingston SDHC 4GB Clase 4

- ✓ Tarjeta d memoria SD 6.25€ sin gastos de envío

http://www.pccomponentes.com/kingston_sdhc_8gb_clase_4.html

PLACA BREADBOARD:

- ✓ Placa de prototipo *BREADBOARD* 16x5cm 10,89€ **sin gastos de envío**

<http://tienda.bricogeek.com/herramientas-de-prototipado/239-placa-de-prototipo-16x5cm.html>

Precio control de luces con WIFI (para el control remoto):

Precio control de luces con WIFI

CONTROL REMOTO DE ILUMINACION CON
MICROCONTROLADORES Y SERVICIO WEB

Raspberry Pi	40,66€
ACS714 4 x	10,89€
MCP3008	9.19€
MODULO 4 RELE	14.93€
WIFI USB	16.65€
BREADBOARD	10,89€
SD	6.95€
Total:	
Aprox. 110.16€ en bricogeek.	
Aprox. 117.47€ en otras tiendas	

Precio control de luces con ETHERNET:

Raspberry Pi	40,66€
ACS714 4 x	10,89€
MCP3008	9.19€
MODULO 4 RELE	4.13€
BREADBOARD	10,89€
SD	6.95€
Total:	
Aprox. 93.51€ en bricogeek.	
Aprox. 98.78€ en otras tiendas	

Debido a la diferencia, casi 17 euros de un montaje a otro, se eligió por montar el control de luces con los módulos Ethernet, ya que este proyecto se realizara en la escuela en un área local donde tienen características más que suficientes para el proyecto.

Al final se llegó a una lista definitiva de compra:

Para la conexión de los componentes y construcción de soportes se han utilizando materiales nuevos como cables y terminales. Lo único que hay que sumar a los **86.56** euros son la madera e interruptores para la realización de la maqueta, haciendo un total de **86.56 euros** de la instalación del control luces remoto más **60 euros** del material necesario para la maqueta.

Las horas invertidas en la construcción del proyecto:

□□ En el mes de **octubre** y **noviembre** se investigó y estudió sobre placas como Arduino y Raspberry dedicando todas las mañanas de la semana durante cuatro horas al día, haciendo un total de aproximadamente **50 horas** en los dos meses.

□□ En **diciembre** se continuó con el estudio e investigación sobre control de luces mientras a la vez se empezó con la compra de los diferentes componentes que se necesitarían. En total en este mes se dedicaron alrededor de **50 horas**.

□□ En el mes de **enero, febrero y marzo** se siguió con la configuración del sistema, y se empezó a hacer las primeras pruebas con led. En estos dos meses se han hecho

aproximadamente **75 horas**.

□□ **En Abril** se instaló el módulo de 4 relés y la programación de los sensores de corriente dedicando tres días a la semana y cada día seis horas. En este mes se dedicó aproximadamente **25 horas**.

□□ El mes de **Mayo** se desarrolló la página web donde se dedicaron **50 horas**.

□□ El mes de **Junio** se terminó de construir la maqueta y la instalación domótica **con una duración de 50 horas**.

Sumando todas las horas dedicadas hacen un total de x horas, además de contar estas se le pueden sumar unas 25 horas en tutorías con el profesor de proyecto, sumando un total entre todo de **325 horas**.

Como conclusión se detallará todas las horas y el precio en la siguiente **tabla 21**:

Mes	Horas dedicadas	Precio * Hora (8.75€ la hora)
Octubre y Noviembre	50	437.5
Diciembre	50	437.5
Enero, Febrero y Marzo	75	656,25
Abril	25	218.75
Mayo	50	437.5
Junio	50	437.5
Tutorías	25	218.75
Total	325 horas	2730€

Sección	Horas trabajadas	Precio * Hora (8.75€ la hora)
Investigación sobre domótica	50	437.5
Accionamiento luz	50	437.5
Sensorización corriente	25	218.75
Servidor web	50	437.5
Montaje maqueta	25	218.75
Documento	62.5	542.5
Programación	50	437.5
Total	325 horas	2730€

CAPÍTULO 6: CONCLUSIONES

6 CONCLUSIONES

Para el desarrollo del presente trabajo fin de grado se han invertido muchas horas de investigación en búsqueda de información y desarrollo sobre casas domóticas en el control de luces. Muchas de estas horas de trabajo me han servido para ampliar muchos conocimientos sobre informática y electrónica que desconocía.

Además he podido comprobar la gran importancia de la programación tanto para la instalación de la placa Raspberry como la programación de los sensores utilizados, y también la práctica para la resolución de problemas de ingeniería en leyes de electricidad, electrónica y programación.

Todos estos campos de conocimientos han sido un complemento para mi formación en el grado de Ingeniería Eléctrica.

Para este trabajo fin de grado se propuso como objetivo fundamental construir desde un control de luces domótico para ser usado en la EIMIA de la Universidad de Castilla-la Mancha.

Una vez que se fue desarrollando el proyecto cada vez se hacía más posible adaptarlo a la EIMIA que era una decisión muy buena.

Dicho proyecto podría tener gran utilidad para acceder al control de luces de toda la planta del edificio nuevo. Donde estas zonas o áreas se encuentran lejos de conserjería y es un inconveniente para las personas encargadas de estas funciones recorrer un largo camino para apagar las luces donde se imparten clases sin tener constancia de que estén apagadas o encendidas. Otra gran utilidad es la verificación del estado de luz en las diferentes clases donde se imparten en la escuela.

La construcción de un sistema de control de luces mediante internet como la comunicación entre dispositivos diferentes (sensores y PC), interpretación de la información de sensores, tratamiento y procesamiento de señales digitales, entre otras partes.

Durante esta fase pudimos ver los posibles problemas que más adelante tendríamos que afrontar con algunas soluciones propuestas. Tras estas soluciones propuestas y el estudio propio realizado sirvió para plantear nuestras propias soluciones y cumplir con los objetivos marcados inicialmente.

Al final el proyecto se dividió en tres objetivos fundamentales:

- Diseño y construcción del control de luces
- Creación del servidor web para contener programa de luces
- Lectura de los sensores de corriente para complementar el programa de luces.

Los diferentes puntos propuestos se han cumplido.

En primer lugar como se indica en el capítulo 4, se diseñó un control de luces mediante internet haciendo uso de Raspberry con sus respectivos componentes.

En segundo lugar los script del programa, permiten la comunicación del usuario con el estado de luces en tiempo real. La complejidad del problema requirió muchas horas de estudio e investigación sobre este tema, aproximadamente 2 meses. Los lenguajes de programación que permiten procesamiento de señales son limitados. Se eligió PYTHON ya que es un programa muy potente y por tener librerías ya construidas sobre el tratamiento de señales analógicas como es este caso.

Para el control de luces desde una web hizo falta crear un servidor en Raspberry para que esta controlará y comunicará las diferentes señales analógicas como su control del módulo de relés ya que sin el servidor no sería posible tal proyecto.

Como en todo proyecto, se presentan problemas y antes estos se debe de anteponer una solución, como fue el caso del detectar la circulación de corriente por las diferentes fases de la instalación hasta conocer dicho sensor ACS.

Por último, una vez que se finalizó dicho proyecto se buscó mejorar algunos aspectos como las web.

La realización de estos objetivos ha permitido obtener unos resultados positivos, al conseguir controlar las luces mediante una conexión a internet dando lugar una posible adaptación a una vivienda, local o centro público lo que es un agrado personal.

A partir de este proyecto pueden surgir otros muchos en los que se presenten nuevas mejoras o el control de las diferentes partes de una casa como puede ser la calefacción.

Para finalizar, me gustaría resaltar que la realización de este proyecto ha sido una de las experiencias más memorables de mi carrera y no habría sido posible realizarlo sin mi director del proyecto.

CONTROL REMOTO DE ILUMINACION CON MICROCONTROLADORES Y SERVICIO WEB

Para cualquier estudiante de la universidad es muy importante el apoyo y la confianza del director del proyecto. Ya que nos motiva a seguir creciendo como estudiante y como personas. Y agradecerle todo su apoyo.

CAPÍTULO 7: BIBLIOGRAFÍA

7 BIBLIOGRAFÍA

- [1] **Harris Ray** JavaScript and DOM Scripting (Murach: Training & Reference) [Book]. - [s.l.] : Murach's, 2009.
- [2] **Monk Simon** Fritzing for Inventors: Take Your Electronics Project from Prototype [Book]. - [s.l.] : Mc Grill Hill.
- [3] **Monk Simon** Hacking Electronic An Illustrated DIY Guide for Makers and Hobbyists [Book]. - 22 abr 2013.
- [4] **Upton Eben and Halfacree Gareth** Raspberry Pi (Títulos Especiales) [Book]. - 22 oct 2013.
- [5] **Ibrahim Dogan** Raspberry Pi Advanced Programming [Book]. - 2014.
- [6] **Ronnis Donald** Raspberry_Pi_Projects_for_the_Evil_Genius [Book]. - [s.l.] : Mc Graw Hil, 2013. - Vols. ISBN: 978-0-07-181956-5.
- [7] **Shaw Zed A.** Aprenda A Programar Con Python [Book]. - 6 nov 2014.
- [8] **Truman Sean** PHP: Learn PHP Programming FAST!: The Ultimate PHP & mySQL Programming Crash Course For Beginners! [Book]. - 2014.
- [9] **J.Kula Piotr** Edicion Raspberry.Pi.Server.Essentials [Book]. - [s.l.] : Edit Open Source, 2014.
- [10] **Donat Wolfram** Learn Raspberry Pi Programming with Python [Book]. - [s.l.] : edition technology in action, 2014.
- [11] **Tero Karvinen Kimmo Karvinen , Ville Valtokari** Make:sensors:A Hands-On Primer for Monitoring the Real World with Arduino and Raspberry P [Book]. - 2 jun 2014.
- [12] **Segaran Seggy T** Raspberry pi with Custard: Interfacing to the Raspberry Pi GPIO [Book].
- [13] **Duckett Jon** HTML and CSS: Design and Build Websites (Inglés) [Book]. - nov 2011.
- [14] **Lucas Michael W** SSH Mastery: OpenSSH, PuTTY, Tunnels and Keys [Book]. - 17 feb 2012.
- [15] **Monk Simón** 30 Proyectos con Arduino [Book]. - [s.l.] : Estribor. - Vols. ISBN-10: 8494003003.
- [16] **Molloy Derek** Exploring Beaglebone [Book]. - 20 feb 2015.
- [17] **Rush Christopher** 30 BeagleBone Black Projects for the Evil Genius [Book]. - 1 sep 2014.
- [18] **Hall Robert** Odroid magazine [Book]. - feb 2014.
- [19] **A10 Pagina oficial Hackberry** <http://archive.miniand.com> [Online].
- [20] **Libelium Pagina oficial** www.libelium.com/es/ [Online].
- [21] **Olimex Pagina oficial** <https://www.olimex.com/> [Online].

[22] **Cook Andrew Robinson and Mike** Raspberry Pi Hardware Projects [Book]. - [s.l.] : Wiley. - Vol. 1.

[23] **Hows Peter Membrey and David** Learn Raspberry Pi Whith Linux [Book]. - [s.l.] : Technology In Action.

[24] **Grimmett Richard** Raspberry Pi Robotic Projects [Book]. - [s.l.] : BIRMINGHAM– MUMBAI, Febrero 2014.

[25] **Domínguez Fernando Sáez Vacas y Hugo Martín** [Book]. - [s.l.] : Fundación Rogelio Segovia, Junio 2006.

[26] **Domotica** <https://es.wikipedia.org/wiki/Dom%C3%B3tica> [Online].

[27] **kenbak** <https://es.wikipedia.org/wiki/Kenbak-1> [Online].

[28] **Halfacree Gareth** Raspberry Pi Guia del usuario [Book]. - sep 2012.