

UNIVERSIDAD NACIONAL DEL ALTIPLANO

FACULTAD DE INGENIERÍA MECÁNICA ELÉCTRICA, ELECTRÓNICA Y SISTEMAS
ESCUELA PROFESIONAL DE INGENIERÍA ELECTRÓNICA



DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DOMÓTICO UTILIZANDO
REDUCED INSTRUCTION SET COMPUTING Y SERVIDOR DE APLICACIONES

TESIS

PRESENTADO POR:

SAUL HUAQUIPACO ENCINAS
RICHARD JOHN CONTRERAS MAMANI

PARA OPTAR EL TÍTULO PROFESIONAL DE
INGENIERO ELECTRÓNICO

PUNO

PERÚ

2013

UNIVERSIDAD NACIONAL DEL ALTIPLANO
FACULTAD DE INGENIERÍA MECÁNICA ELÉCTRICA,
ELECTRÓNICA Y SISTEMAS
ESCUELA PROFESIONAL DE INGENIERÍA ELECTRÓNICA

DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DOMÓTICO UTILIZANDO
REDUCED INSTRUCTION SET COMPUTING Y SERVIDOR DE APLICACIONES

TESIS PRESENTADA POR
SAÚL HUAQUIPACO ENCINAS
RICHARD JOHN CONTRERAS MAMANI
PARA OPTAR EL TÍTULO DE
INGENIERO ELECTRÓNICO

NACIONAL DEL
APROBADA POR EL JURADO REVISOR. CONFORMADO POR

PRESIDENTE	:	 _____
PRIMER MIEMBRO	:	ING. FERDINAND LOGARDO PINEDA ANCCO _____
SEGUNDO MIEMBRO	:	 _____
DIRECTOR DE TESIS	:	 _____
ASESOR DE TESIS	:	 _____

PUNO PERÚ


2013

ÁREA: Automatización e instrumentación

TEMA: Control y automatización

DEDICATORIAS:

A mi familia a ellos que me han visto crecer y acompañado siempre; A Ti A.D.R.A que me has tenido paciencia y has sabido esperar hasta este momento.

Saúl Huaquipaco Encinas

A la Universidad Nacional del Altiplano, específicamente a la Escuela Profesional de Ingeniería Electrónica, docentes, estudiantes (compañeros), personal administrativo y todo lo que la compone. Los que fueron parte de mi formación profesional, donde estos años me

Permitieron aprender mucho, no sólo referido en cuanto a la carrera sino también en general a cómo afrontar la vida, esta tesis está dedicada a todos ellos.

Richard John Contreras Mamani

AGRADECIMIENTOS:

A mi alma mater que me ha enseñado que el esfuerzo y la determinación son siempre la mejor manera de conseguir los objetivos.

A mis docentes los que siempre aprendí lo mejor dentro y fuera de aulas.

Saúl Huaquipaco Encinas

A mis seres queridos, mi madre, padre, hermana, quienes con su apoyo incondicional, estuvieron permanentemente apoyándome para poder concluir la carrera profesional; y siguen ayudándome en mi formación personal.

A la organización Cultural Nueva Acrópolis, parte importante para aclarar mis profundas inquietudes y mostrarme como encontrar y vivir un ideal además de conocer personas maravillosas en esta extraordinaria aventura.

A mis amigos por su compañía, gracias a ellos pude y puedo ir descubriendo cada vez más sobre el misterio que llamamos vida.

Richard John Contreras Mamani

ÍNDICE

ÍNDICE

ÍNDICE DE IMÁGENES Y TABLAS

RESUMEN 12

ABSTRACT 13

INTRODUCCIÓN 14

CAPÍTULO I:

PLANTEAMIENTO DEL PROBLEMA, ANTECEDENTES Y OBJETIVOS DE LA INVESTIGACIÓN

1.1. PLANTEAMIENTO DEL PROBLEMA 17

1.1.1. Definición del problema 18

1.1.1.1. Caracterización del problema 18

1.1.1.2. Sistematización del problema 18

1.1.2. Delimitación del problema 19

1.1.2.1. Delimitación espacial 19

1.1.2.2. Delimitación temporal 19

1.1.2.3. Delimitación social 20

1.2. ANTECEDENTES DE LA INVESTIGACIÓN 20

1.3. OBJETIVOS DE LA INVESTIGACIÓN 21

1.3.1. Objetivo general 21

1.3.2. Objetivos específicos 21

CAPÍTULO II:**MARCO TEÓRICO, MARCO CONCEPTUAL E HIPÓTESIS DE LA INVESTIGACIÓN**

2.1. MARCO TEÓRICO	23
2.1.1. Domótica e inmótica	23
2.1.2. El hogar digital	25
2.1.3. Climatización y consumo energético	26
2.1.4. Entretenimiento y confort	26
2.1.5. Seguridad	26
2.1.6. Servicios comunitarios	26
2.1.7. La vivienda bioclimática	27
2.1.8. El hogar seguro	27
2.1.9. Aplicaciones	28
2.1.10. Placa de control (ATmeg1280)	28
2.1.11. Servidores	29
2.1.11.1. Tipos de servidores	29
2.1.12. Actuadores electrónicos	30
2.1.13. Linux	31
2.1.14. Sistemas existentes	31
2.1.14.1. X10	31
2.1.14.2. HEYU	32
2.1.14.3. HCS	33
2.1.14.4. ExDomus	33

2.2. MARCO CONCEPTUAL	34
2.2.1. Software propietario	34
2.2.2. Sistema operativo	34
2.2.3. Unix	34
2.2.4. Software libre	34
2.2.5. Hardware libre	35
2.3. HIPÓTESIS DE LA INVESTIGACIÓN	36
2.3.1. Hipótesis general	36
2.3.2. Hipótesis específicas	36
2.3.3. Variables	36
2.3.3.1. Variable independiente	36
2.3.3.2. Variable dependientes	36
 CAPÍTULO III:	
MÉTODO DE INVESTIGACIÓN	
3. MÉTODO DE INVESTIGACIÓN	38
3.1. TIPO DE INVESTIGACIÓN	38
3.2. NIVEL DE INVESTIGACIÓN	38
3.3. ENFOQUE DE LA INVESTIGACIÓN	38

CAPÍTULO IV:**CARACTERIZACIÓN DE LA INVESTIGACIÓN**

4.1. CARACTERÍSTICAS GEOGRÁFICAS	40
4.2. CLASIFICACIÓN TIPOLOGICA	41
4.3. CLASIFICACIÓN TOPOLOGICA	42
4.4. TABLA COMPARATIVA SISTEMAS DOMÓTICOS EXISTENTES	44

CAPÍTULO V:**INGENIERÍA DEL PROYECTO**

5.1. HARDWARE	46
5.1.1. Servidor	46
5.1.2. Tarjeta de control	51
5.1.3. Sensores	54
5.1.3.1. Sensor de temperatura	54
5.1.3.2. Sensor de movimiento	56
5.1.3.3. Sensor de humo	57
5.1.3.4. Sensor de gas	57
5.1.3.5. Sensor de apertura	58
5.1.3.6. Sensor de luminosidad	58
5.1.4. Etapa de potencia	59
5.2. SOFTWARE	62
5.2.1. Sistema operativo	62
5.2.2. Apache	62

5.2.3. Mysql	63
5.2.4. Php	63
5.3. ESQUEMA GENERAL DEL SISTEMA	64
5.4. IMPLEMENTACIÓN	65
5.4.1. Descripción del domicilio	65
5.4.2. Instalación del Software en el servidor	67
5.4.2.1. Software necesario	67
5.4.2.2. Descarga de software de instalación	67
5.4.2.3. Instalación del sistema operativo	69
5.4.2.4. Primer arranque	70
5.4.2.5. Instalación de Apache2	71
5.4.2.6. Instalación de Mysql	72
5.4.2.7. Instalación de Php	73
5.4.2.8. Configuración de la web	73
5.5. ANÁLISIS DE RESULTADOS	107
5.5.1. Consumo energético de servidor	107
5.5.2. Costo de servidores	109
CONCLUSIONES	110
RECOMENDACIONES	111
BIBLIOGRAFÍA	112
WEBGRAFÍA	117

INDICE DE IMÁGENES Y TABLAS

Imagen 01. Esquema de una instalación domótica	27
Imagen 02: ubicación del área de investigación	40
Imagen 03: Arquitectura centralizada del sistema domótico	41
Imagen 04: topología estrella	43
Tabla 01: tabla comparativa entre los sistemas domóticos	44
Imagen 05: Arquitectura del software del Broadcom BCM2835	47
Imagen 06: Diagrama de bloques fuente	49
Imagen 07: Broadcom BCM 835 y periféricos	50
Imagen 08: Diagrama esquemático de la tarjeta de control del sistema domótico.	52
Imagen 09: Diagrama PCB de la placa de control	53
Imagen 10: Sensor de temperatura LM35	54
Tabla 02: Tabla de comparación entre valores simulados y reales	55
Imagen 11: Conexión sensor de Movimiento	56
Imagen 12: Sensor de Movimiento	56
Imagen 13: Conexión del sensor de Humo	57
Imagen 14: Sensor de gas	57
Imagen 15: interruptor de contacto magnético	58
Tabla 03: Pulso del Triac según luminosidad	59

Imagen 16: Diagrama esquemático etapa de potencia	60
Imagen 17: Placa de etapa de potencia	61
Imagen 18: Esquema de funcionamiento servidor web	62
Imagen 19: diagrama de bloques del sistema domótico	64
Imagen 20: Plano de la vivienda	65
Imagen 21: Maqueta de la vivienda	66
Imagen 22: Etapa de potencia de maqueta de la vivienda	66
Imagen 23: MD5 & SHA-1 Checksum Utility	68
Imagen 24: MD5 & SHA-1 Checksum Utility	69
Imagen 25: Entorno gráfico Raspbian Wheezy	71
Imagen 26: página de muestra del local host	72
Imagen 27: interfaz web del sistema domótico	102
Imagen 28: Consumo energético de servidores	108
Imagen 29: Comparativa del consumo unitario entre servidores	109

RESUMEN

La automatización de las viviendas proponen mejorar cuatro aspectos fundamentales de las mismas, la seguridad, el ahorro energético, la accesibilidad y el confort; nosotros hemos puesto principal importancia en la seguridad y el ahorro energético.

En la actualidad existen una gran variedad de sistemas y tecnologías disponibles que pueden ser utilizados y permiten obtener las prestaciones antes mencionadas tales como: HEYU, HCS, ExDomus, X10, etc. Sin embargo, se decidió trabajar con la tecnología web, teniendo como base del hardware del servidor un computador con arquitectura Reduced instruction Set Computing y como software de este un servidor LAMP.

El sistema consta de actuadores y sensores que se comunican con el servidor a través de una tarjeta de control utilizando la tipología centraliza y la topología de red estrella que hacen que el servidor reciba toda la información proveniente de los sensores para que este las procese y envíe las ordenes correspondientes a los actuadores.

Se hicieron análisis comparativos en cuanto al consumo energético de nuestro servidor con respecto a servidores tradicionales logrando obtener resultados exitosos.

La reducción en el consumo energético respecto a uso de un servidor tradicional con procesador Xeon E3-1200 fue considerable, lográndose en una reducción de hasta 445 watts.

ABSTRACT

Automating aim to improve housing four fundamental aspects of them , safety , energy conservation , accessibility and comfort , we have placed great importance on safety and energy savings .

Currently there are a variety of systems and technologies available that can be used and lead to the aforementioned benefits such as: Heyu , HCS , ExDomus , X10 , etc. . However, it was decided to work with web technology , based server hardware with a computer architecture Reduced instruction Set Computing software as this a LAMP server.

The system consists of sensors and actuators that communicate with the server through a control card using the typology and centralized star network topology that makes the server receives all the information from the sensors to the process and send this the corresponding commands to the actuators.

Comparative analysis was made in terms of energy consumption with respect to our traditional server servers achieving successful outcomes .

The reduction in energy consumption compared to a traditional server using Xeon processor E3 -1200 was considerable , achieving a reduction of up to 445 watts.

INTRODUCCIÓN

Lo sistema domóticos tradicionales se caracterizan por no usar servidores dentro de su diseño, es por ello que el presente trabajo de investigación trata sobre el uso de un servidor web usando un computador con arquitectura Reduced Instruction Set Computing (RISC) para controlar y administrar un sistema domótico.

El presente trabajo de investigación utilizada un servidor web con arquitectura RISC para resolver de manera eficiente los cuatro pilares de la domótica: eficiencia energética, seguridad, , accesibilidad y confort; poniendo más énfasis en la primera puesto que una de las características de usar un servidor con arquitectura RISC y su muy bajo consumo energético.

Es muy habitual encontrar la arquitectura RISC aplicado teléfonos inteligentes o tabletas pero no lo hemos encontrado aplicado en sistemas domóticos por lo que este trabajo pretende contribuir brindando información sobre el uso y aplicación de esta arquitectura a los sistemas domoticos.

En el capítulo uno el planteamiento del problema encontraremos información relacionada a nuestra propuesta de implementar este tipo de tecnología, así como los antecedentes de esta investigación y los objetivos que perseguidos.

En el capítulo dos se da un panorama en cuanto a lo que a domótica se refiere, sus características, alcances, bondades y se menciona los elementos utilizados en el presente trabajo; los conceptos referidos y tecnologías empleadas, para que pueda ser comprendido aún para los no entendidos en el tema.

En el capítulo tres mencionamos que el tipo de investigación es aplicada y el nivel de la misma exploratoria, descriptiva y explicativa.

En el capítulo cuatro se encontrará una comparación entre el uso de la tecnología que se estudia y utiliza en la presente tesis y las tecnologías ya existentes.

En el capítulo cinco hacemos una descripción de los circuitos utilizados además de la instalación y configuración del sistema operativo en el servidor y la instalación de las aplicaciones web. Finalmente encontraremos una muestra en el análisis de los resultados comparando el consumo energético y los costos entre servidores.



CAPÍTULO I

PLANTEAMIENTO DEL PROBLEMA, ANTECEDENTES Y OBJETIVOS DE LA INVESTIGACIÓN

1.1. PLANTEAMIENTO DEL PROBLEMA

Uno de los aspectos fundamentales en cuanto al tema que nos ocupa es de la seguridad, tanto en bienes patrimoniales, como la seguridad personal.

La inseguridad siempre ha sido un tema de preocupación en los hogares. Más aun cuando el 86% de la población dijo sentirse inseguro ante la ola de crímenes, asaltos, asesinatos y otros hechos delictivos que suceden en cualquier parte del Perú, este estudio presentado por el INEI correspondiente al último semestre del 2012 muestra la precariedad de la seguridad en las viviendas como en las calles del país.

El Programa de las Naciones Unidas para el Desarrollo (PNUD) elaboró el “Informe regional de desarrollo humano 2013-2014, seguridad ciudadana con rostro humano: diagnóstico y propuestas para América Latina” En el documento se señala que el Perú tiene la percepción más alta de inseguridad en el barrio, con 50% (esto según una encuesta hecha en el país a 1.500 adultos por país).

La inseguridad no solo tiene agentes externos sino también posibles agentes internos, por ejemplo; las fugas de gas, peligros de incendio entre otras eventualidades que generalmente ocurren cuando no estamos presentes dentro del domicilio.

En la actualidad existen varios sistemas domóticos que resuelven estos problemas, pero acarrear un alto costo, esto debido a que la mayoría de ellos utilizan software propietario para su funcionamiento y muchas veces solo están diseñados para usos en dispositivos limitados.

Otro de los aspectos fundamentales es el consumo energético, y es que los sistemas domóticos que existe actualmente tienen aparte de un alto consumo energético, la presente tesis titulada: “Diseño e implementación de un sistema domótico utilizando Reduced Instrucción Set Computing” buscar dar solución a estos problemas citados, brindando una mayor seguridad en los hogares, a través de la tele presencia, sistemas de cámaras, encendido de luces remoto o programado, reducción de costos en cuanto a los elementos utilizados y el consumo energético.

Otra de las bondades de estos sistemas es que brindan una experiencia de confort, en cuanto a la regulación de la iluminación, regulación de la temperatura y automatización de algunas necesidades.

Además de un control de usuario desde donde se encuentre el mismo, brindando una ubicuidad al sistema domótico.

1.1.1. Definición del problema.

1.1.1.1. Caracterización del problema.

¿Cómo mejorar los niveles de seguridad, optimizar el consumo energético y la accesibilidad en los hogares?

1.1.1.2. Sistematización del problema.

- a) ¿La arquitectura Reduced instruction set computing y servidor de aplicaciones podrá ayudar a mejorar los sistemas domóticos?

- b) ¿Cuáles serán las ventajas y desventajas de usar la arquitectura Reduced instruction set computing y servidor de aplicaciones en los sistemas domóticos?
- c) ¿Cuál será la factibilidad de usar la arquitectura Reduced instruction set computing y servidor de aplicaciones en sistemas domóticos?
- d) ¿Será posible Implementar el diseño domótico que cumpla con lo cuatro pilares de la domótica haciendo uso de la arquitectura RISC y servidor de aplicaciones?

1.1.2. Delimitación del problema

Una de las principales limitantes para desarrollar el proyecto es el factor económico, otro factor que limita el desarrollo del prototipo es la adquisición de algunos tipos de sensores, a causa de no contar en el mercado local y nacional.

1.1.2.1. Delimitación Espacial

El proyecto de investigación se desarrolla generalmente en el domicilio de uno de los tesisistas, y para el desarrollo de la maqueta se requirió los servicios de un técnico de la ciudad de Puno del departamento de Puno.

1.1.2.2. Delimitación temporal

El proyecto de investigación se inició en ejecución a partir de febrero del año 2013, posteriormente finalizando en noviembre del 2011.

1.1.2.3. Delimitación social

La limitación social, es la resistencia al cambio tecnológico que tienen las empresas o ciudadanos de la ciudad de Puno, como también la desconfianza de adquirir el proyecto.

1.2. ANTECEDENTES DE LA INVESTIGACIÓN

Podemos encontrar los orígenes de la domótica la cual se remonta aproximadamente a los años setenta, cuando, tras muchas investigaciones aparecieron los primeros dispositivos de automatización de edificios basados en la aún exitosa tecnología X-10 (protocolo de comunicaciones para el control remoto de dispositivos eléctricos, que utiliza la línea eléctrica (220V o 110V). Durante los años siguientes la comunidad internacional mostró un creciente interés por la búsqueda de la casa ideal, comenzando diversos ensayos con avanzados electrodomésticos y dispositivos automáticos para el hogar. Los primeros sistemas comerciales fueron instalados, sobre todo, en Estados Unidos y se limitaban a la regulación de la temperatura ambiente en las oficinas.

En nuestro país, en la Pontificia Universidad Católica del Perú, Diego Armando Lozada Díaz, presenta la tesis titulada: Diseño y desarrollo de un transmisor y receptor de un sistema domótico utilizando tecnología de corrientes portadoras y el protocolo x-10, para optar el Título de Ingeniero Electrónico. También en la Universidad Católica de Santa María, para la facultad de Ciencias e Ingenierías físicas y formales; Torres Cárdenas Daniel Stevens, presenta la tesis titulada: Diseño de un Sistema Domótico para Seguridad Utilizando Tecnología WAP.

En la región tenemos la tesis presentada en la escuela profesional de ingeniería Electrónica de la Universidad Nacional del Altiplano, por David Salinas Mendoza titulada “Diseño y construcción de un sistema de control distribuido para aplicaciones en domótica basado en microcontroladores de microchip”

1.3. OBJETIVOS DE LA INVESTIGACIÓN

1.3.1. Objetivo General

Diseñar e implementar un sistema domótico utilizando reduced instruction set computing y servidor de aplicaciones.

1.3.2. Objetivos específicos

- Estudiar el sistema domótico utilizando reduced instruction set computing y servidor de aplicaciones.
- Diseñar el sistema domótico utilizando reduced instruction set computing y servidor de aplicaciones.
- Implementar un sistema domótico utilizando reduced instruction set computing y servidor de aplicaciones.



CAPÍTULO II

MARCO TEÓRICO, MARCO CONCEPTUAL E HIPÓTESIS DE LA
INVESTIGACIÓN

2.1. MARCO TEÓRICO

2.1.1. Domótica e Inmótica

Los diccionarios franceses incorporaron el término domotique a partir de 1998. Esta palabra se introdujo en España por los Pirineos como Domótica, que procede del latín domus (casa, domicilio) y del griego αὐτόματοϛ, automática (aunque existen autores que opinan que deriva de informática, como defiende el Diccionario de la RAE, o incluso de robótica).

Huidobro J.M. y Millán R. (2004) recogen que el origen de la Domótica se remonta a los años setenta, cuando en Estados Unidos aparecieron los primeros dispositivos de automatización de edificios basados en la aún hoy exitosa tecnología X-10. Estas incursiones primerizas se alternaron con la llegada de nuevos sistemas de calefacción y climatización orientados al ahorro de energía, en clara sintonía con la crisis del petróleo. Los primeros equipos comerciales se limitaban a la colocación de sensores y termostatos que regulaban la temperatura ambiente. La disponibilidad y proliferación de la electrónica de bajo coste favoreció la expansión de este tipo de sistemas, despertando así el interés de la comunidad internacional por la búsqueda de la casa ideal. Los ensayos con electrodomésticos avanzados y otros dispositivos automáticos condujeron a comienzos de los años noventa, junto con el desarrollo de los PC y los sistemas de cableado estructurado, al nacimiento de aplicaciones de control, seguridad, comunicaciones que son el germen de la Domótica actual¹.

¹ Domótica: Un enfoque sociotécnico, Hugo Martín Domínguez, Fernando Sáez Vacas, Junio 2006

Para entender mejor este tema conviene delimitar el significado del término Domótica, si bien resulta altamente complicado en tanto que recoge conceptos en cambios permanentes y bajo discusiones continuas. Para Huidobro J.M. y Millán R. (2004), la Domótica se aplica a los sistemas y dispositivos que proporcionan algún nivel de automatización dentro de la casa, pudiendo ser desde un simple temporizador para encender y apagar una luz o aparato a una hora determinada, hasta los más complejos sistemas capaces de interactuar con cualquier elemento eléctrico del hogar. La vivienda domótica es por tanto "aquella que integra un conjunto de automatismos en materia de electricidad, electrónica, robótica, informática y telecomunicaciones, con el objetivo de asegurar al usuario un aumento del confort, la seguridad, el ahorro energético, las facilidades de comunicación y las posibilidades de entretenimiento". Se pretende con ello integrar todos los aparatos del hogar a fin de que funcionen de la forma más eficaz posible y con la necesidad de una intervención mínima o inexistente por parte del usuario.

Por otra parte, se viene hablando de Inmótica para referirse a la automatización de edificios terciarios o de servicios (hoteles, oficinas, hospitales, plantas industriales, universidades), como combinación de la voz latina immobilis, aquello que está fijo, de donde deriva el término castellano inmueble, y de la ya vista 'automática'. Este concepto se identifica habitualmente también como building management system, en referencia a la coordinación y gestión de las instalaciones con que se encuentran equipadas las edificaciones, así como a su capacidad de comunicación, regulación y control. El origen del término

Inmótica es también francés y, aunque es de uso bastante común en España, todavía no ha sido recogido por el diccionario de la RAE.

Es común la confusión de no distinguir entre ambas disciplinas, Domótica e Inmótica. Recurriendo a un ejemplo propuesto por Lorente S. (2004), el parecido entre ambas disciplinas es similar al existente entre los turismos y los camiones de la marca Mercedes. Si bien los elementos y componentes utilizados en viviendas y en edificios terciarios son fuertemente similares, como lo son en un camión Mercedes y en un turismo Mercedes, en cuanto a aplicaciones y funcionalidad, que están relacionadas directamente con el mercado, apenas existen intersecciones entre los conjuntos de viviendas y edificios terciarios, como no las hay entre camiones y turismos.

2.1.2. El hogar digital

Como se tiene dicho, la domótica también se asocia, sobre todo en ámbitos de telecomunicaciones, al denominado hogar digital. El gran progreso tecnológico sufrido por los sistemas de telecomunicación y el desarrollo y proliferación de Internet, han incrementado exponencialmente nuestra capacidad para crear información, almacenarla, transmitirla, recibirla y procesarla. El mayor acceso a la información, ha venido además asociado a una mayor facilidad para comunicarnos, para establecer nuevas vías de diálogo con el resto del mundo, en cualquier momento y desde cualquier lugar. Tras una etapa de introducción lenta de la tecnología digital, ahora estamos en los comienzos de una revolución de servicios para el hogar, donde las pasarelas residenciales,

apoyadas con conexiones de banda ancha, conectarán inteligentemente todos los dispositivos del hogar, soportando servicios interactivos y de valor añadido de diversa índole.

2.1.3. Climatización y consumo energético.

- Programación del encendido y apagado de todo tipo de aparatos (calderas, aire acondicionado, toldos, luces, etc.), según las condiciones ambientales.
- Acomodación a los planes de tarifas reducidas (tarifa nocturna).
- Contadores electrónicos que informan del consumo energético.

2.1.4. Entretenimiento y confort.

- Conexión a Internet desde cualquier punto.
- Visión de canales de TV (televisión) en cualquier habitación.
- Control de los dispositivos eléctricos/electrónicos

2.1.5. Seguridad.

- Configuración de procedimientos de avisos en caso de intrusión o avería (alarma técnica).
- Instalación de cámaras y micrófonos para ver lo que ocurre.
- Control del acceso a la vivienda.

2.1.6. Servicios comunitarios.

- Control de la iluminación de las zonas comunes.
- Manejo de alarmas de seguridad y alarmas técnicas.
- Servicios Web para la comunidad de propietarios.



Imagen 01. Esquema de una instalación domótica.

2.1.7. La vivienda bioclimática

En realidad, no existen marcadas diferencias entre una vivienda tradicional y otra con equipamiento domótico. Se trata de la misma vivienda, con equipamiento semejante, y con el mismo diseño arquitectónico. La diferencia sólo estriba en la incorporación de una mínima tecnología, que permita gestionar de forma más eficiente los distintos equipos e instalaciones domésticas que conforman la vivienda (es decir, la calefacción, el aire acondicionado, la iluminación, etc.). No obstante, últimamente la vivienda domótica también se asocia mucho al concepto de vivienda bioclimática, es decir, a aquella vivienda con un nuevo diseño arquitectónico que se adapta mejor al medio ambiente, reduciendo el consumo energético y contribuyendo a mantener nuestra salud y de la planeta Tierra.

2.1.8. El hogar seguro

La vivienda domótica también se suele asociar al hogar seguro (vivienda bunker), ya que una gran parte de los dispositivos que se utilizan para proteger una vivienda frente a intrusiones (si exceptuamos las rejas, puertas blindadas y

otros dispositivos de protección pasivos) tienen mucho que ver con lo que la domótica implica, en cuanto se refiere a los aspectos de seguridad y comunicación.

Así, cada vez es más común, dotar a las nuevas viviendas de un pre - instalación de alarma y son numerosas las empresas que ofrecen sus servicios para la recepción de las mismas y actuación en caso de conflicto. La conexión a redes externas y, en concreto a Internet de banda ancha, es cada día más común y ello facilita el control de la vivienda por medio de la pasarela residencial, en cualquier momento y desde cualquier lugar desde donde se disponga de una conexión, vía la red fija o la red móvil.

2.1.9. Aplicaciones

- Ahorro energético
- Confort
- Seguridad
- Accesibilidad

2.1.10. Placa de control (ATmeg1280)

Para tener un control estable, pensamos en utilizar una interfaz probada, una plataforma de hardware libre, basada en una placa con un microcontrolador y un entorno de desarrollo, diseñada para facilitar el uso de la electrónica en proyectos multidisciplinarios.

El hardware consiste en una placa con un microcontrolador Atmel AVR y puertos de entrada/salida. Los microcontroladores más usados son el Atmega168, Atmega328, Atmega1280 que esta es la que utilizaremos para nuestro proyecto, ATmega8 por su sencillez y bajo coste que permiten el desarrollo de múltiples diseños. Por otro lado el software consiste en un

entorno de desarrollo que implementa el lenguaje de programación Processing/Wiring y el cargador de arranque (boot loader) que corre en la placa. Por ello utilizaremos arduino, desde octubre de 2012, Arduino se usa también con microcontroladoras CortexM3 de ARM de 32 bits⁵, que coexistirán con las más limitadas, pero también económicas AVR de 8 bits. ARM y AVR no son plataformas compatibles a nivel binario, pero se pueden programar con el mismo IDE de Arduino y hacerse programas que compilen sin cambios en las dos plataformas. Eso sí, las microcontroladoras CortexM3 usan 3.3V, a diferencia de la mayoría de las placas con AVR que usan mayoritariamente 5V. Sin embargo ya anteriormente se lanzaron placas Arduino con Atmel AVR a 3.3V como la Arduino Fio y existen clónicos de Arduino Nano y Pro como Meduino en que se puede conmutar el voltaje.

2.1.11. Servidores

En informática, un servidor es un nodo que forma parte de una red, provee servicios a otros nodos denominados clientes.

2.1.11.1. Tipos de servidores

En la siguiente lista hay algunos tipos comunes de servidores: es el que almacena varios tipos de archivos y los distribuye a otros clientes en la red.

- Servidor de impresiones.
- Servidor de fax.
- Servidor de la telefonía.
- Servidor proxy
- Servidor del acceso remoto (RAS).
- Servidor de uso.
- Servidor web.

- Servidor de base de datos.
- Servidor de reserva.
- Servidor de seguridad
- Servidor dedicado
- Servidor no dedicado

2.1.12. Actuadores electrónicos

Un actuador es un dispositivo capaz de transformar energía hidráulica, neumática o eléctrica en la activación de un proceso con la finalidad de generar un efecto sobre un proceso automatizado. Este recibe la orden de un regulador o controlador y en función a ella genera la orden para activar un elemento final de control como, por ejemplo, una válvula.

Existen varios tipos de actuadores como son:

- Electrónicos
- Hidráulicos
- Neumáticos
- Eléctricos

Los actuadores electrónicos también son muy utilizados en los aparatos mecatrónicos, como por ejemplo, en los robots. Los servomotores CA sin escobillas se utilizarán en el futuro como actuadores de posicionamiento preciso debido a la demanda de funcionamiento sin tantas horas de mantenimiento.

Los actuadores electrónicos también son muy utilizados en los aparatos mecatrónicos, como por ejemplo, en los robots. Los servomotores CA sin escobillas se utilizarán en el futuro como actuadores de posicionamiento

preciso debido a la demanda de funcionamiento sin tantas horas de mantenimiento.

2.1.13. Linux

Linux es un núcleo libre de sistema operativo (también suele referirse al núcleo como kernel) basado en Unix. Es uno de los principales ejemplos de software libre y de código abierto. Linux está licenciado bajo la GPL v2 y está desarrollado por colaboradores de todo el mundo. El desarrollo del día a día tiene lugar en la Linux Kernel Mailing List Archive

El núcleo Linux fue concebido por el entonces estudiante de ciencias de la computación finlandés, Linus Torvalds, en 1991. Linux consiguió rápidamente desarrolladores y usuarios que adoptaron códigos de otros proyectos de software libre para usarlo con el nuevo sistema operativo. El núcleo Linux ha recibido contribuciones de miles de programadores de todo el mundo. Normalmente Linux se utiliza junto a un empaquetado de software, llamado distribución Linux y servidores.

2.1.14. Sistema existentes

2.1.14.1. X10:

Estándar internacional abierto para la comunicación entre dispositivos, utilizado en domótica. Utiliza fundamentalmente las líneas eléctricas para señalización y control, aunque define también un protocolo de transporte inalámbrico. Fue desarrollado en 1975 por Pico Electronics, Escocia para permitir el control remoto de artefactos en el hogar. Fue la primera tecnología de propósito general para redes domóticas y aún sigue siendo la más utilizada.

Pueden conectarse hasta 256 dispositivos, aprovechando un espacio de direcciones de 8 bits. Este estándar puede considerarse un método de conexión y comunicación de equipos, que aprovecha la red eléctrica para sus comunicaciones, por lo cual no requiere la instalación de un cableado original, reduciendo costos pero también al mismo tiempo podría ser considerado un método de control básico de equipos: por ejemplo, si se pretende encender y apagar una luz en forma remota, basta con conectar a la lámpara a un dispositivo X10 y a éste último a la red eléctrica, otorgándole un código único dentro de la casa y del otro lado, se conecta otro artefacto compatible con X10, el cual pueda enviar señales para el código de la lámpara. Desde este punto de vista, puede considerarse que X10 otorga una forma de programación básica y de fácil instalación pero al mismo tiempo, bastante limitada.

2.1.14.2. HEYU:

Es un programa de consola para Linux y otras Sistemas Operativos derivados de UNIX (Mac OS X, FreeBSD, Solaris, entre otros) para el control de luces y dispositivos en la oficina y el hogar. El sistema es capaz de interactuar con equipos X10, temporizar acciones y de monitorear/registrar los eventos generados. El software permite asimismo generar scripts de acciones para su posterior ejecución.

En resumen, HEYU actúa como controlador a los sensores y actuadores basados en el protocolo X10, brindando cierto grado de automatización pero teniendo la limitación de ser un producto para línea de comandos con lo cual el espectro de posibles usuarios queda restringido de gran manera. Del mismo

modo, cuenta con el inconveniente que es difícil de operar y monitorear en forma remota, cuando el usuario se encuentra fuera de su casa.

Home Control System (HCS) de Home Automation, Inc (HAI): este sistema propietario puede funcionar como sistema centralizado (comprando un equipo de hardware que actúa como módulo central) o como sistema distribuido (ciertos equipos pueden actuar de forma localizada y autónoma).

2.1.14.3. HCS:

Permite la temporización de acciones, el envío de notificaciones y el control de funciones de seguridad, ahorro de energía y entretenimiento. Puede ser operado mediante una interfaz gráfica táctil (que se compra por separado) o vía telefónica. En forma opcional puede ser controlado con control remoto a través de una PC que corra Windows Media Center o vía internet mediante software adicional.

En pocas palabras, HCS es un sistema altamente configurable y escalable, a costa de requerir equipo específico de hardware para poder funcionar y software adicional para poder realizar muchas de las tareas que HEYU no permite.

2.1.14.4. ExDomus:

Es un software para el Media Center de Windows XP/Vista que permite controlar luces, cortinas, calentadores, cámaras, sistemas de seguridad, etc. usando como protocolo de conexión X10 o KNX, según la versión. Permite la temporización de acciones así como también la ejecución de acciones disparadas por un evento externo, tal como la activación de la alarma. Se trata de un software pago (aunque cuenta con una versión básica gratuita) que permite automatizar ciertas tareas pero dentro del ámbito del hogar (no cuenta con herramientas para la programación o visualización de eventos en forma remota)

2.2. MARCO CONCEPTUAL

2.2.1. Software propietario:

El software propietario se refiere a cualquier programa informático en el que los usuarios tienen limitadas las posibilidades de usarlo, modificarlo o redistribuirlo (con o sin modificaciones), o cuyo código fuente no está disponible o el acceso a éste se encuentra restringido.

2.2.2. Sistema operativo:

Un sistema operativo (SO, frecuentemente OS, del inglés Operating System) es un programa o conjunto de programas que en un sistema informático gestiona los recursos de hardware y provee servicios a los programas de aplicación, ejecutándose en modo privilegiado respecto de los restantes y anteriores próximos y viceversa.

2.2.3. Unix:

Es un sistema operativo portable, multitarea y multiusuario; desarrollado, en principio, en 1969, por un grupo de empleados de los laboratorios Bell de AT&T, entre los que figuran Ken Thompson, Dennis Ritchie y Douglas McIlroy.

2.2.4. software libre:

El software libre (en inglés "free software", aunque esta denominación a veces se confunde con "gratis" por la ambigüedad del término "free" en el idioma inglés, por lo que también se usa "libre software") es la denominación del software que respeta la libertad de todos los usuarios que adquirieron el

producto y, por tanto, una vez obtenido el mismo puede ser usado, copiado, estudiado, modificado, y redistribuido libremente de varias formas. Según la Free Software Foundation, el software libre se refiere a la libertad de los usuarios para ejecutar, copiar, distribuir, y estudiar el mismo, e incluso modificar el software y distribuirlo modificado.

2.2.5. Hardware libre:

Hardware libre a aquellos dispositivos de hardware cuyas especificaciones y diagramas esquemáticos son de acceso público, ya sea bajo algún tipo de pago o de forma gratuita. La filosofía del software libre es aplicable a la del hardware libre y por eso forma parte de la cultura libre. Algo que tiene en común el hardware con el software es que ambos corresponden a las partes tangibles de un sistema informático.

2.3. HIPÓTESIS DE LA INVESTIGACIÓN

2.3.1. Hipótesis general

El diseño de sistema domótico basado en arquitectura RISC y tecnologías web usando servidor LAMP puede proporcionar la misma fiabilidad que un sistema domótico comercial a un menor costo y con mayor accesibilidad a través de computadoras, tablets y celulares.

2.3.2. Hipótesis específicas.

- Se podrá realizar el estudio de los distintos sistemas domótico que pueden proporcionar la misma fiabilidad que un sistema domótico comercial a un costo económico más bajo.
- Se podrá diseñar un sistema domótico basado en arquitectura RISC y servidor LAMP.
- Se podrá implementar un sistema domótico basado en arquitectura RISC y servidor LAMP

2.3.3. Variables.

2.3.3.1. Variable independiente.

- Sistema domótico.

2.3.3.2. Variables dependientes.

- Arquitectura RISC
- Consumo energético



CAPÍTULO III

MÉTODO DE INVESTIGACIÓN

3. MÉTODO DE INVESTIGACIÓN

3.1. Tipo de investigación

La presente investigación es de tipo aplicada, pues consiste en el empleo práctico del conocimiento o teorías de las redes y comunicaciones, servidores y actuadores electrónicos.

3.2. Nivel de Investigación

El nivel de investigación utilizado es el Cuasi-experimental

Dado que explicamos los cambios producidos en el consumo energético del sistema domótico manipulando las variables independientes.

3.3. Enfoque de la investigación.

- cuantitativa

CAPÍTULO IV

CARACTERIZACIÓN DEL ÁREA DE INVESTIGACIÓN



4.1. CARACTERÍSTICAS GEOGRÁFICAS

La ciudad de Puno, capital de la Región del mismo nombre, está ubicada a orillas del lago Titicaca a 3827 m.s.n.m., entre las coordenadas geográficas 15°50'15" latitud sur y 770°01'18" longitud oeste del meridiano de Greenwich. Cuenta con una extensión aproximada de 460.63Km² y tiene los siguientes límites geográficos:

Norte : Provincia de San Román, Huancané y parte del lago Titicaca

Sur : Provincia de el Collao y Departamento Moquegua

Este : Provincia de el Collao y el lago Titicaca

Oeste : Departamento de Moquegua y provincia de San Román

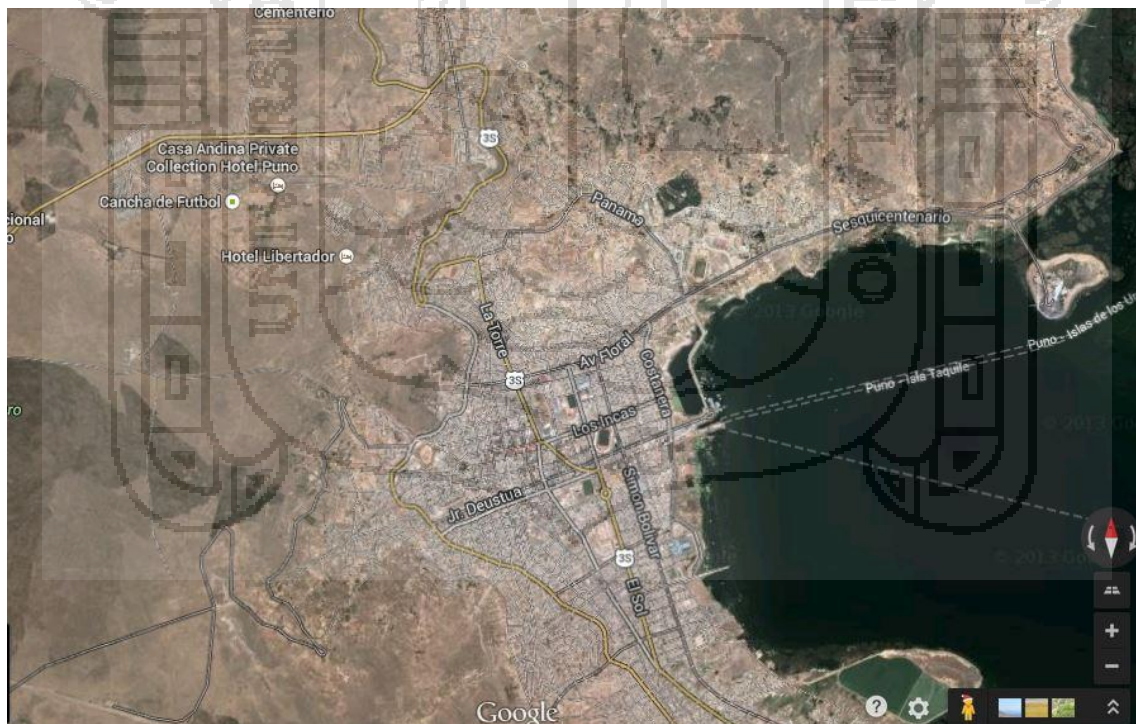


Imagen 02: ubicación del área de investigación

4.2. CLASIFICACIÓN TIPOLOGICA

El sistema Domótico tiene una arquitectura centralizada se caracteriza por tener un único servidor que recibe toda la información de las entradas, que las procesa y envía a las salidas las ordenes de acción correspondientes y tiene las siguiente características:

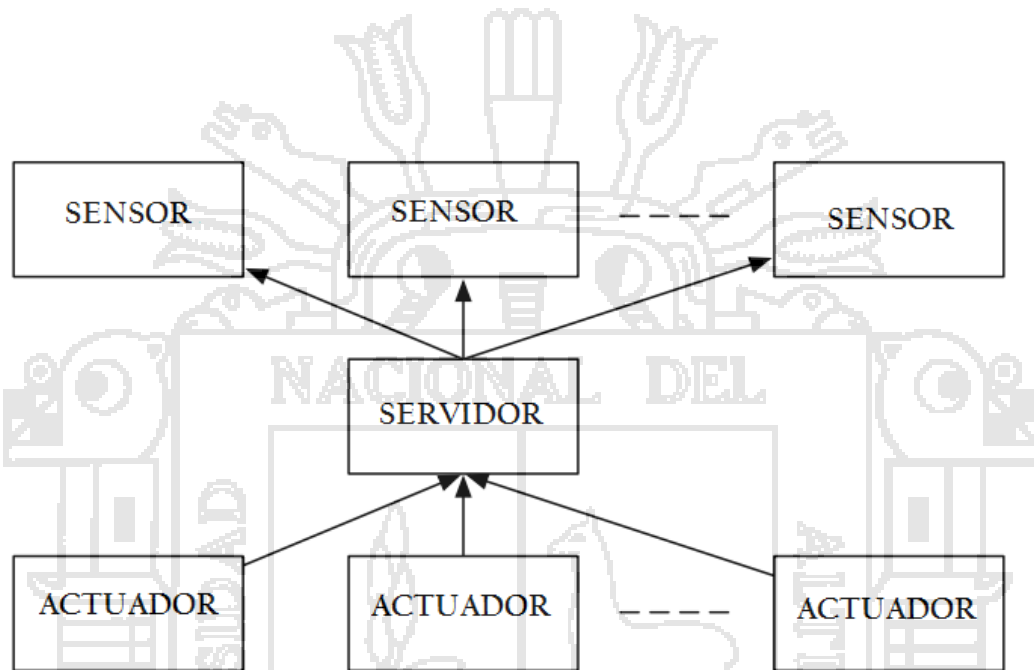


Imagen 03: Arquitectura centralizada del sistema domótico

- Los elementos (sensores y actuadores) son de tipo universal
 - Coste reducido o moderado
 - Fácil uso y formación
 - Instalación sencilla
- Los inconvenientes son:
 - Cableado significativo
 - Sistemas dependientes del funcionamiento óptimo de la central

- Modularidad difícil
- Reducida ampliabilidad
- Capacidad del sistema (canales o puntos)

4.3. CLASIFICACIÓN TOPOLÓGICA

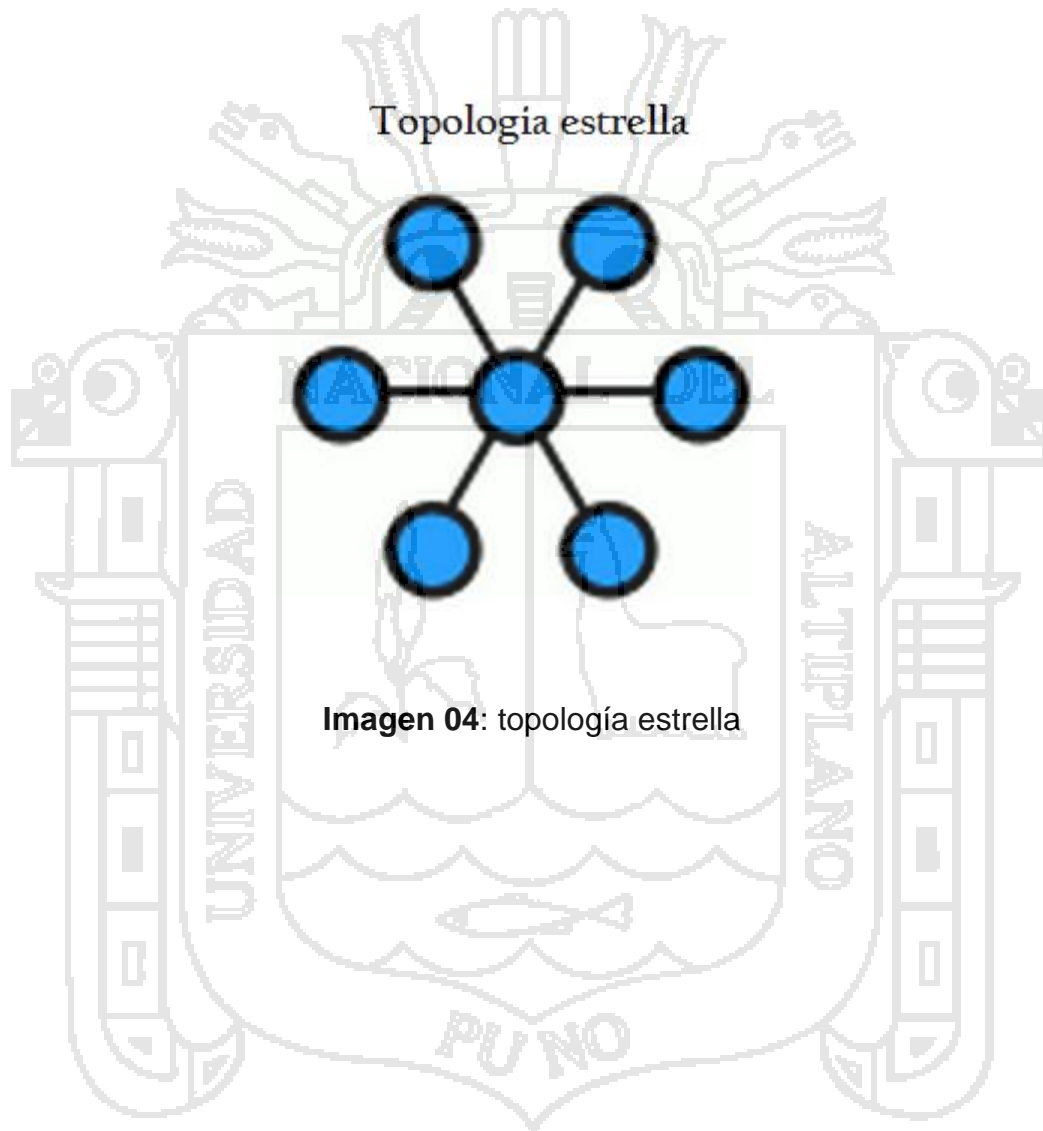
Otro aspecto importante a un sistema es su topología, es decir, la organización física y lógica de los nodos de la red nuestro sistema a ser de arquitectura centralizada utiliza el modo estrella en el que los dispositivos de entrada (sensores) y los de salida (actuadores) van conectados hasta la central (servidor) desde donde se efectúa el tratamiento de los datos conjunto y tienen los siguientes características:

Ventajas de la Topología estrella

- La topología estrella tiene dos ventajas grandes a diferencia de la topología Bus y Ring.
- Es más tolerante, esto quiere decir que si una computadora se desconecta o si se le rompe el cable solo esa computadora es afectada y el resto de la red mantiene su comunicación normalmente.
- Es fácil de reconfigurar, añadir o remover una computadora es tan simple como conectar o desconectar el cable.

Desventajas de la Topología Estrella

- Es costosa ya que requiere más cable que la topología Bus y Ring.
- Si una computadora se cae, no puede enviar ni recibir mensajes.



4.4. TABLA COMPARATIVA DE SISTEMAS DOMÓTICOS EXISTENTES

	X10	HEYU	HCS	ExDomus	Nuestro sistema
Arquitectura	Distribuida (instalación individual de equipos)	Centralizada	Centralizado o Distribuido, si se dispone de pocos dispositivos a automatizar	Centralizada	Centralizada
Facilidad de instalación	Alta (con sólo conectar los equipos a la red eléctrica y asignarles un código único salen funcionando)	Baja. Requiere experiencia en entornos UNIX (manejo de Shell scripting y compilación a partir del código fuente)	Media	Media	Media
Facilidad de uso	Media. A través de Múltiples controles remotos	Baja	Alta	Media Alta	Alta
Método de control	Controles remotos X10	No posee interfaz gráfica, es un programa de línea de comandos	Se utiliza un control remoto que se comunica con una PC con Windows Media Center	Se utiliza un control emotoque se comunica con una PC con Windows Media Center	Web
Control remoto fuera del hogar	No disponible sin software adicional	No disponible en forma nativa	Vía telefónica, celulares o, con software adicional, vía Internet	No disponible	Disponible
Facilidad de programación	Alta	Media. Requiere generar scripts	Alto	Media	Media
Poder de programación	Bajo	Medio alto	Alto	Media	Medio
Temporización de acciones	Sólo utilizando equipos adicionales	Si	Si	Si (en las versiones pagas del software)	Si
Registro de eventos	No disponible sin software adicional	Si	Si	No disponible	Si

Tabla 01: tabla comparativa entre los sistemas domóticos.

CAPÍTULO V

EXPOSICIÓN Y ANÁLISIS DE RESULTADOS



INGENIERÍA DEL PROYECTO

El diseño del sistema, consta de 2 partes fundamentales, Hardware y software. Luego se procede a la implementación del sistema.

5.1. HARDWARE

La etapa del hardware, en la cual describiremos los circuitos utilizados para el funcionamiento del sistema domótico.

5.1.1. Servidor

El hardware del servidor es un ordenador de tamaño reducido que está basado en la arquitectura ADVANCED REDUCED INSTRUCTION SET COMPUTING MACHINE lo que le hace tener un bajo consumo energético trabaja con un procesador Broadcom BCM2835 .

El BCM2835 es un procesador de aplicaciones multimedia con un coste optimizado para aplicaciones móviles y embebidos avanzados que requieren los más altos niveles de rendimiento multimedia. Diseñado y optimizado para la eficiencia energética, BCM2835 utiliza tecnología VideoCore 4 de Broadcom para habilitar las aplicaciones en la reproducción multimedia, imágenes, cámara de vídeo, streaming media, gráficos y juegos en 3D. En la siguiente imagen podemos apreciar la arquitectura del software para el Broadcom BCM2835.

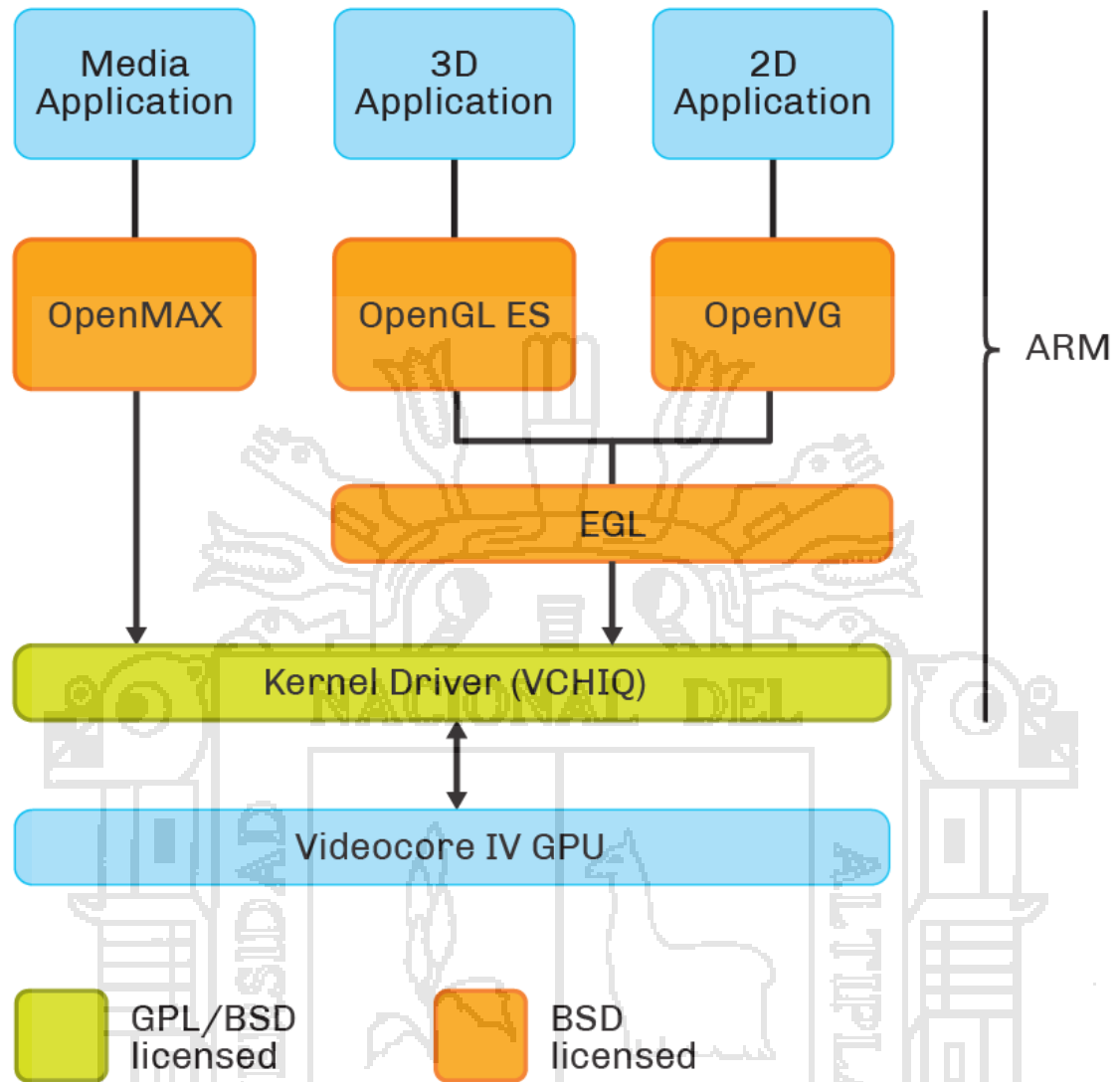


Imagen 05: Arquitectura del software del Broadcom BCM2835

Además del procesador tiene las siguientes especificaciones:

- CPU: 700 MHz ARM11 ARM1176JZF-S core
- GPU: Broadcom VideoCore IV, OpenGL ES 2.0, OpenVG 1080p30 H.264 de alto perfil de codificación / decodificación
- Memoria (SDRAM) 512 MB
- Puertos USB 2.0: 1
- Salidas de vídeo: vídeo compuesto RCA, HDMI (no al mismo tiempo)
- Salidas de audio: Conector 3.5 mm jack, HDMI
- Entradas de audio: ninguno, pero un micrófono USB o tarjeta de sonido podrían añadirse
- Almacenamiento: Secure Digital | ranura para tarjetas SD / MMC / SDIO
- Red: 10/100 MBits Ethernet con cable RJ45
- Periféricos de bajo nivel: Pines (Entradas y salidas de propósito general), Serial
- Reloj en tiempo real: Ninguno
- Potencia de consumo: de 700 mA, (3,5 W) a 1500 mA, (5,25 W)
- Fuente de energía: 5 V (DC) a través de Micro USB de tipo B o encabezado GPIO
- Tamaño: 85,0 x 56,0 mm x 15 mm, 85,0 x 56,0 mm x 17 mm
- Peso: 31g 40g

Como se puede apreciar en el siguiente diagrama de bloques:

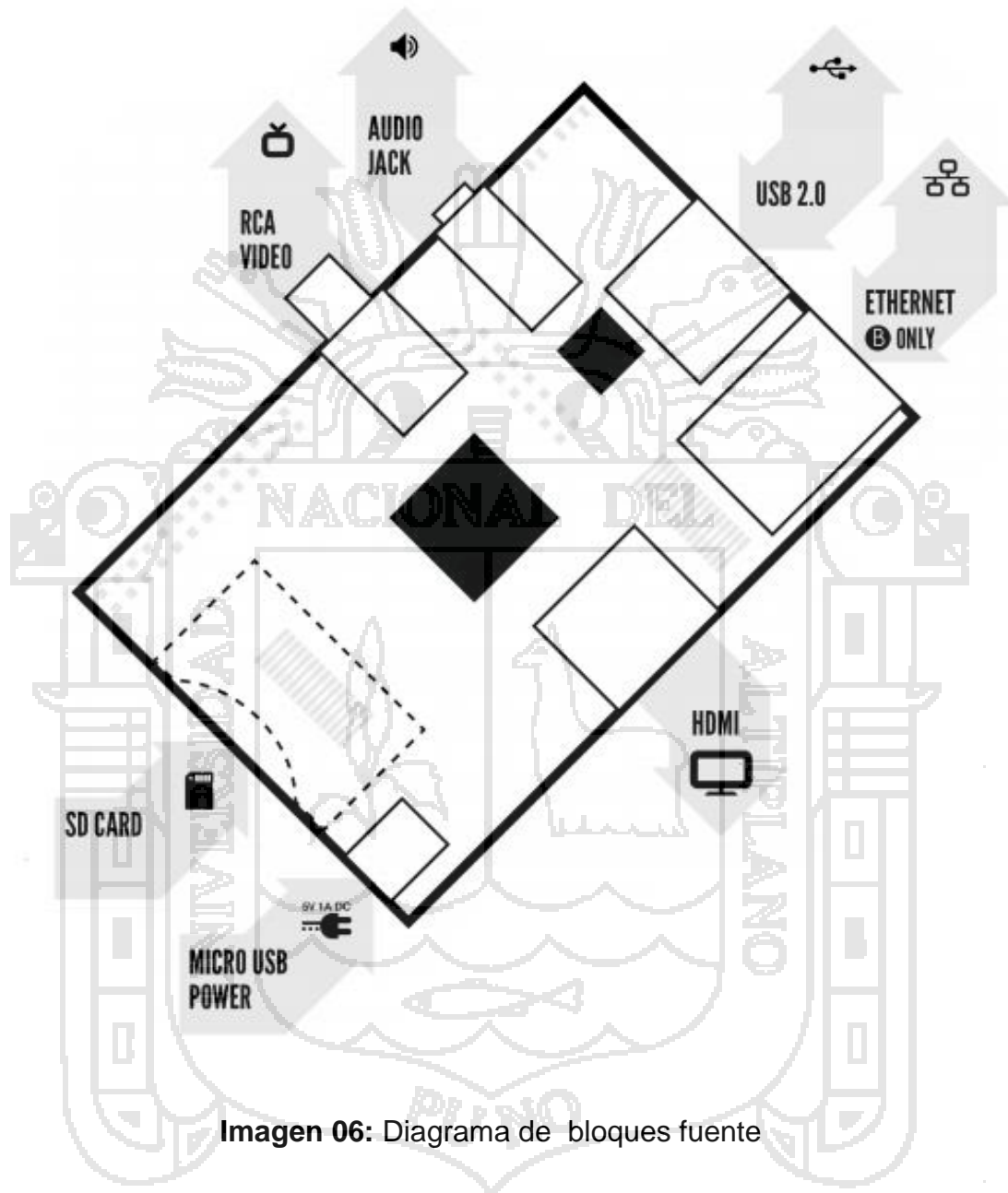
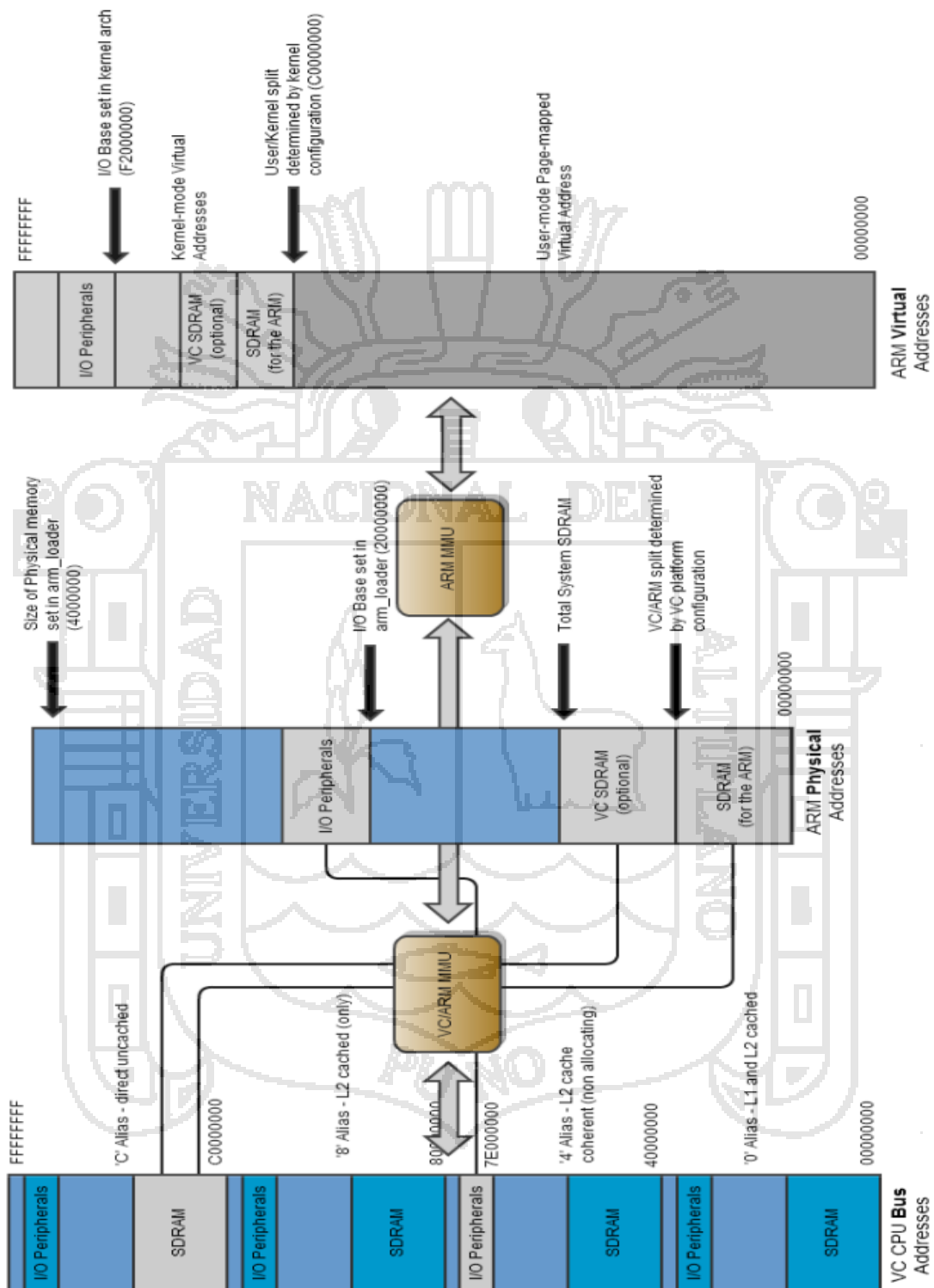


Imagen 06: Diagrama de bloques fuente

La comunicación del procesador Broadcom BCM835 con los periféricos se



realiza de acuerdo a la siguiente imagen:

Imagen 07: Broadcom BCM 835 y periféricos.

5.1.2. Tarjeta de control

Es una placa microcontrolador basada ATmeg1280. Tiene 54 entradas/salidas digitales (de las cuales 14 proporcionan salida PWM), 16 entradas digitales, 4 UARTS (puertos serie por hardware), un cristal oscilador de 16MHz, conexión USB, entrada de corriente, conector ICSP y botón de reset.

Cuenta con las siguientes características:

- Microcontrolador ATmega1280 128KB de memoria flash para almacenar código (4KB son usados para el arranque del sistema (bootloader).El ATmega1280 tiene 8 KB de memoria SRAM. El ATmega1280 tiene 4KB de EEPROM
- Voltaje de funcionamiento 5V
- Voltaje de entrada (recomendado) 7-12V
- Voltaje de entrada (limite) 6-20V
- Pines E/S digitales 54 (14 proporcionan salida PWM)
- Pines de entrada analógica 16
- Intensidad por pin 40 mA
- Intensidad en pin 3.3V 50 mA
- Memoria Flash 128 KB de las cuales 4 KB las usa el gestor de arranque (bootloader)
- SRAM 8 KB
- EEPROM 4 KB
- Velocidad de reloj 16 MHz

En el siguiente grafico podemos apreciar el circuito esquemático de la tarjeta de control.

TARJETA DE CONTROL SISTEMA DOMOTICO CON ARQUITECTURA RISC Y SERVIDOR DE APLICACIONES

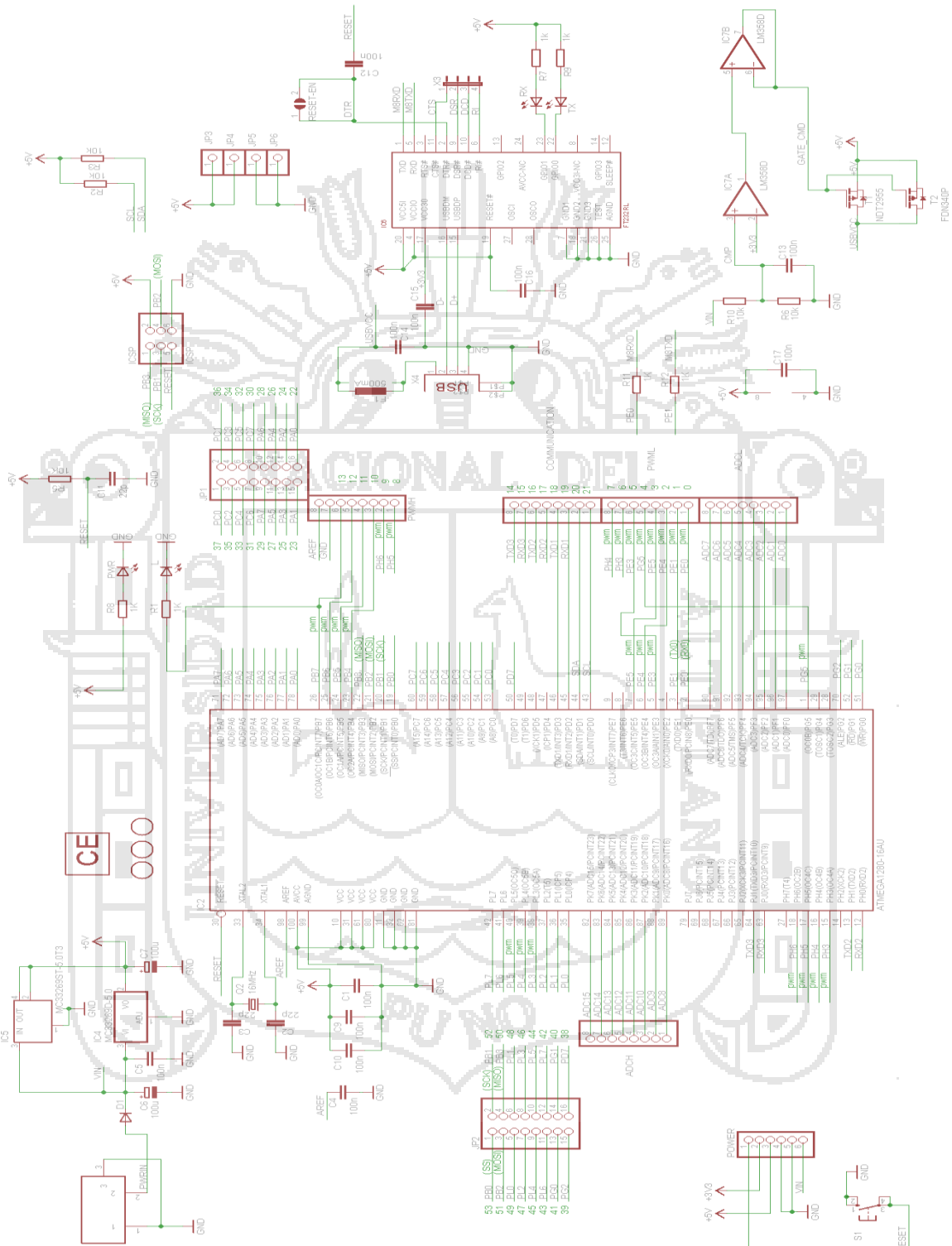


Imagen 08: Diagrama esquemático de la tarjeta de control del sistema domótico.

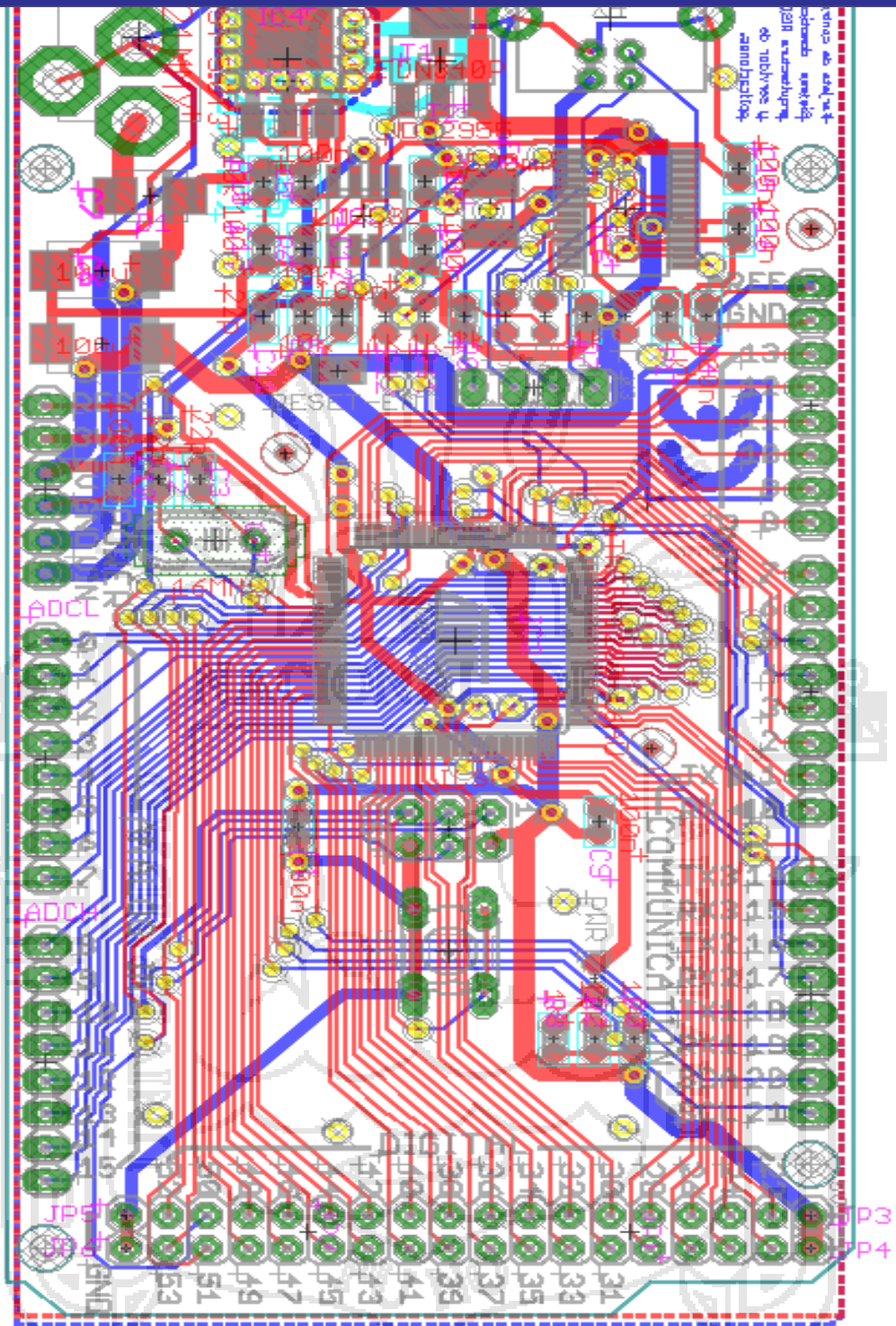


Imagen 09: Diagrama PCB de la placa de control.

Temperatura C°	Simulacion (mV)	Real (mV)
15	150	151.47
16	160	161.48
17	170	171.51
18	180	181.55
19	190	191.57
20	200	201.58
21	210	211.62
22	220	221.65
23	230	231.67
24	240	241.68
25	250	251.71
26	260	261.73
27	270	271.77
28	280	281.77
29	290	291.79
30	300	301.83

Tabla 02: Tabla de comparación entre valores simulados y reales

5.1.3.3. Sensor de humo:

Estos sensores detectan partículas en el aire, calor o humo, posibilitando la activación de señales sonoras y luminosas.

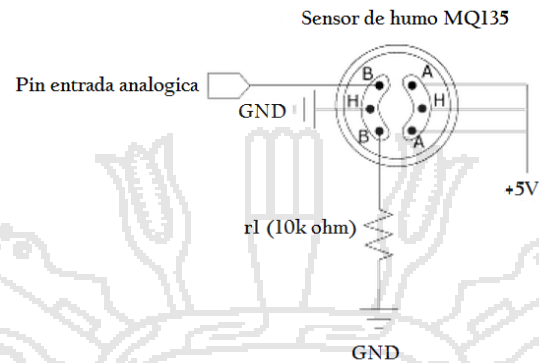


Imagen 13: Conexión del sensor de Humo

5.1.3.4. Sensor de gas:

Detectan gases tóxicos y explosivos como butano, propano, gas natural, gas ciudad, etc. Se sitúan en diferentes alturas en función del gas a detectar; para gases como el butano o propano, el detector se ha de colocar a unos 20 cm del suelo, y para gas natural o gas ciudad, cuya densidad es menor, se colocan a 20 cm del techo

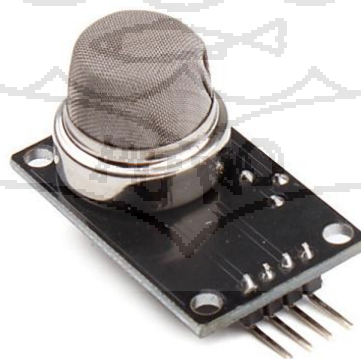


Imagen 14: Sensor de gas

5.1.3.5. Sensor de apertura:

En el caso de detectores perimetrales (contactos magnéticos), se instalará la parte imantada en la puerta o ventana, mientras que la parte cableada se colocará en el marco de ésta. Deben estar en la parte de la puerta o ventana contraria a las bisagras.



Imagen 15: interruptor de contacto magnético

5.1.3.6. Sensor de luminosidad:

Como la librería de Proteus no posee nuestro modelo de LDR hemos simulado su funcionamiento con una resistencia variable, la cual podemos modificar su resistencia al igual que se modificará realmente al variar la luminosidad que recibe. Este valor lo introducimos por el conversor del PIC para realizar una lectura continua del estado de luminosidad que tenemos, es aquí donde a través de programa se realiza el control de la luz, dependiendo del valor que tengamos se disparara en un cierto tiempo la salida hacia el Triac.

Luminosidad	Valor conversor	Pulso Triac
0%	208	No hay pulso
25%	262	2.5ms
50%	352	5ms
75%	536	7.5ms
100%	1023	-

Tabla 03: Pulso del Triac según luminosidad.

En la tabla podemos observar, los porcentajes de luminosidad, siendo 0% cuando la LDR está tapada y no recibe luz, y 100% cuando la LDR recibe máxima luz natural. Como vemos al 0% no hay pulso ya que queremos que la bombilla de su máxima luz, por lo que el triac no ha de actuar, en cambio al 100% no enviaremos ningún tipo de pulso, sino que simplemente apagaremos la bombilla ya que significará que tenemos la luz adecuada.

5.1.4. Etapa de potencia

Reciben las órdenes y las transforman en señales de aviso, regulación o conmutación.

Un relé es un interruptor eléctrico que utiliza un electroimán para mover el interruptor de la posición de apagado a encendido en lugar de una persona que mueve el interruptor. Se necesita una relativamente pequeña cantidad de energía para activar un relé, pero el relé puede controlar algo que llama mucho más poder.

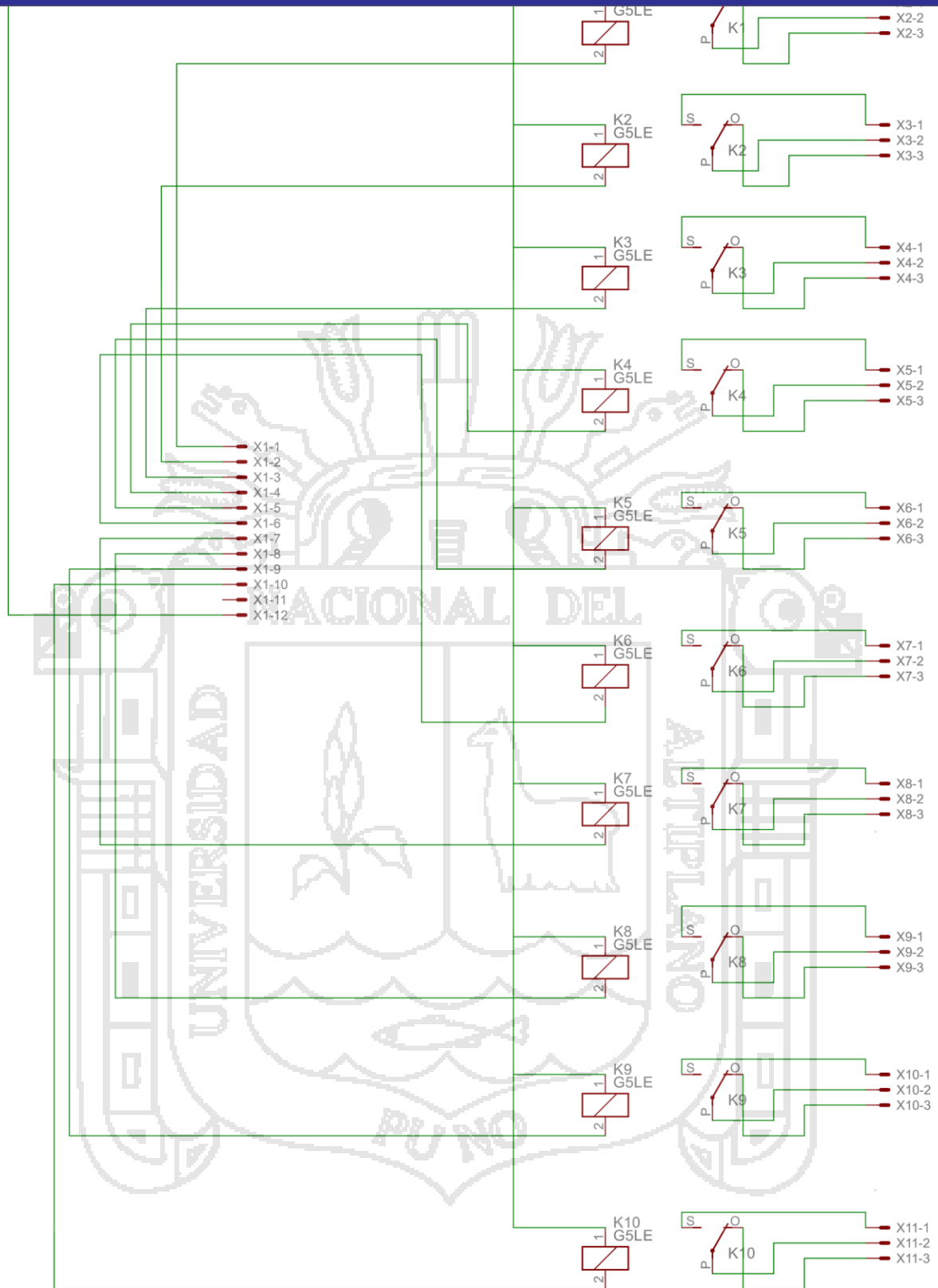


Imagen 16: Diagrama esquemático etapa de potencia.

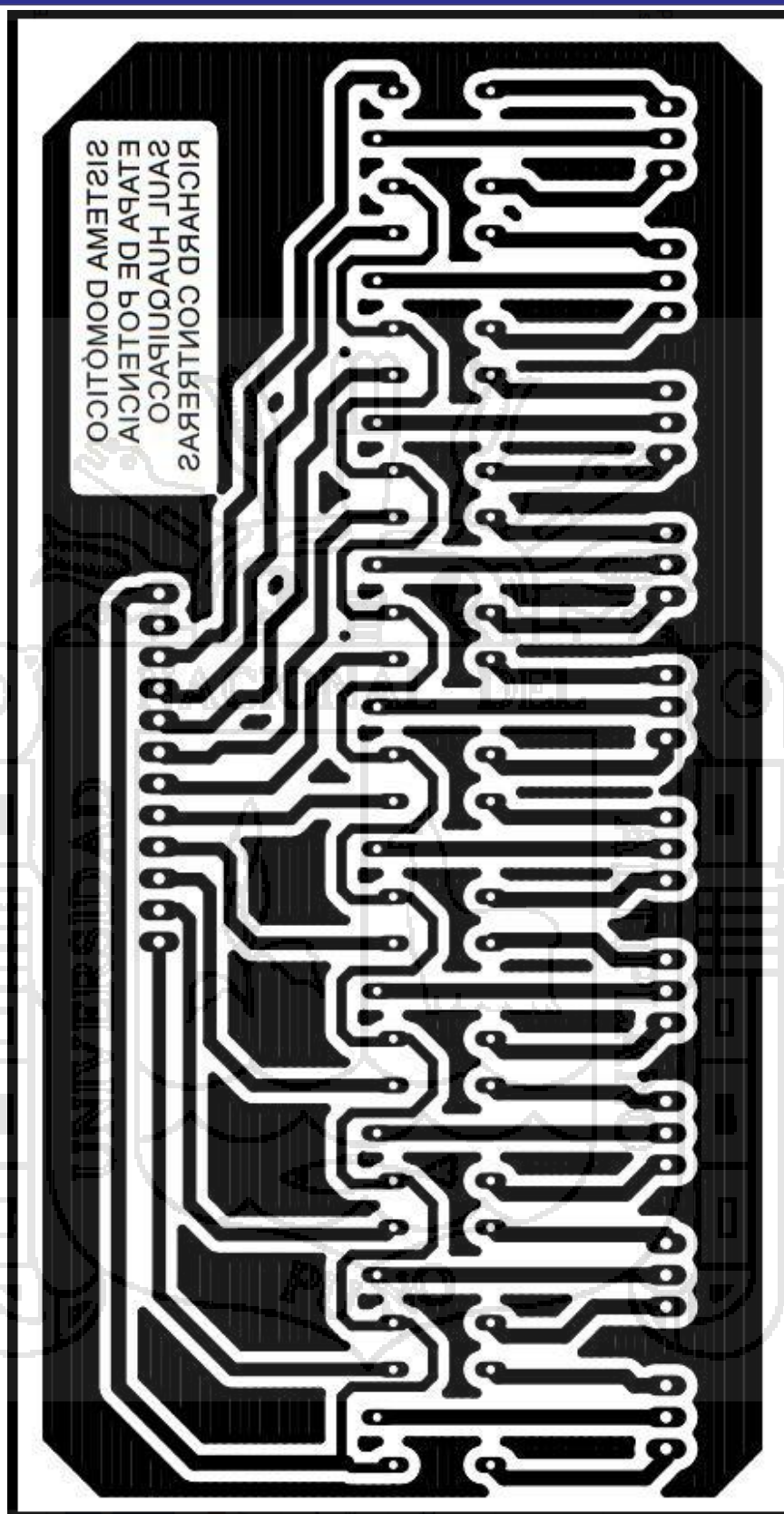


Imagen 17: Placa de etapa de potencia

5.2. SOFTWARE

5.2.1. Sistema operativo:

El sistema operativo que se instaló en el servidor está basado en Linux, es una variante del S.O. Debían ya que son lo que mejor se adaptan a la arquitectura del procesador (Reduced Instuction Set Computing), y con compatibles con las aplicaciones a ser instaladas en el servidor.

5.2.2. Apache:

El servidor HTTP Apache es un servidor web libre y de código abierto, el más popular en cuanto a uso, sirviendo de facto como plataforma de referencia para el diseño y evaluación de otros servidores web.

La versión APACHE que hemos usado es 2.4.7 y tiene el siguiente esquema de funcionamiento.

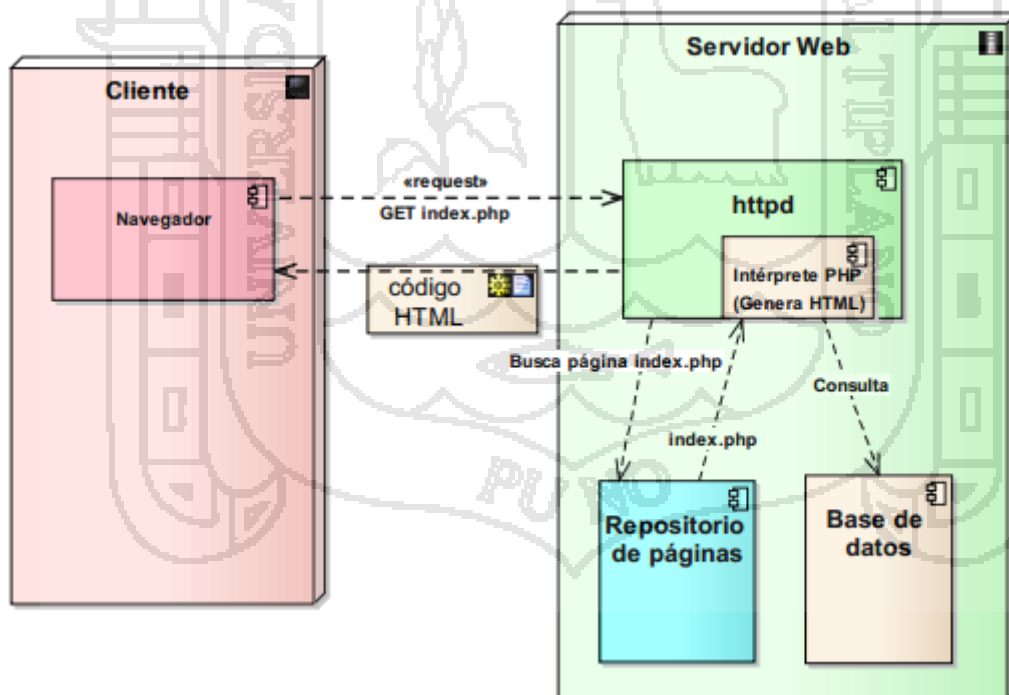


Imagen 18: Esquema de funcionamiento servidor web

5.2.3. MySQL: Es un sistema de administración de bases de datos (Database Management System, DBMS) para bases de datos relacionales. Así, MySQL no es más que una aplicación que permite gestionar archivos llamados de bases de datos.

Existen muchos tipos de bases de datos, desde un simple archivo hasta sistemas relacionales orientados a objetos. MySQL, como base de datos relacional, utiliza múltiples tablas para almacenar y organizar la información. MySQL fue escrito en C y C++ y destaca por su gran adaptación a diferentes entornos de desarrollo, permitiendo su interacción con los lenguajes de programación más utilizados como PHP, Perl y Java y su integración en distintos sistemas operativos.

También es muy destacable, la condición de open source de MySQL, que hace que su utilización sea gratuita e incluso se pueda modificar con total libertad, pudiendo descargar su código fuente. Esto ha favorecido muy positivamente en su desarrollo y continuas actualizaciones, para hacer de MySQL una de las herramientas más utilizadas por los programadores orientados a Internet.

La versión de MySQL usada en nuestro servidor es la 5.6.11.

5.2.4. PHP:

Es un lenguaje de código abierto muy popular, adecuado para desarrollo web y que puede ser incrustado en HTML. Es popular porque un gran número de páginas y portales web están creadas con PHP. Código abierto significa que es de uso libre y gratuito para todos los programadores que quieran usarlo. Incrustado en HTML significa que en un mismo archivo vamos a poder combinar código PHP con código HTML, siguiendo unas reglas.

La versión utilizada en nuestro servidor es 5.5.5

5.3. ESQUEMA GENERAL DEL SISTEMA

Como se muestra en la siguiente imagen el sistema domótico es un sistema de tipo centralizado con topología estrella.

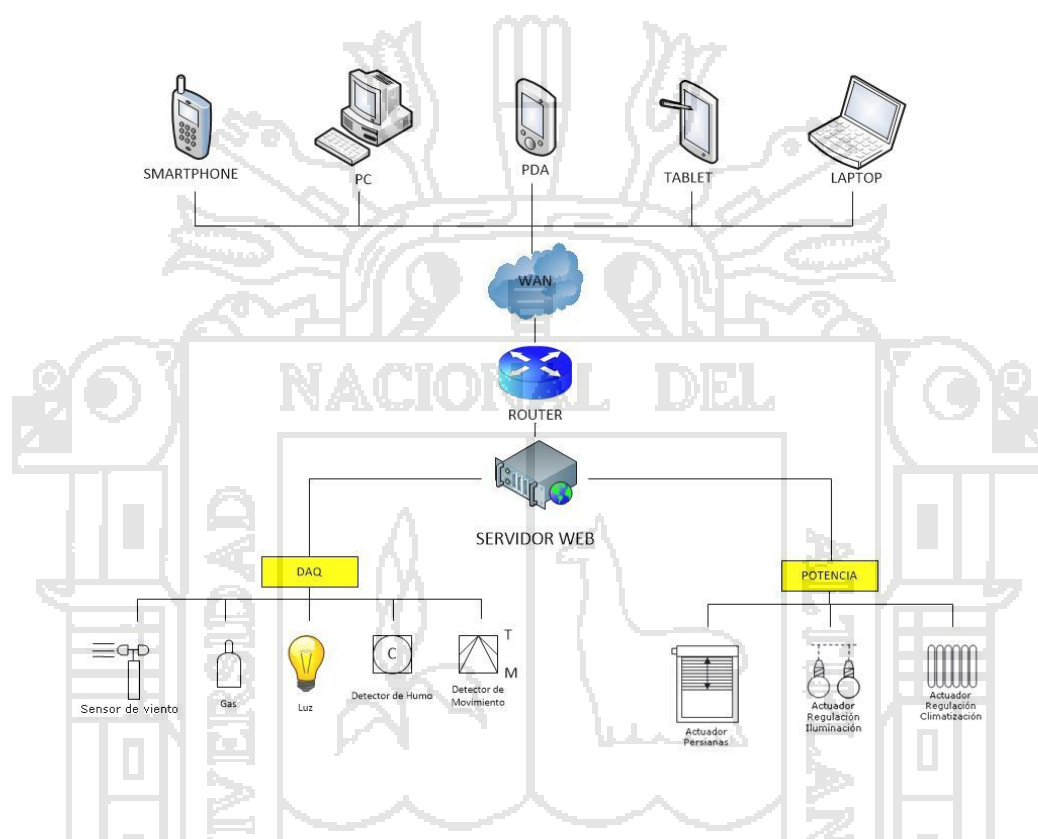


Imagen 19: diagrama de bloques del sistema domótico

Para lo cual tenemos un servidor central con un hardware basado en la arquitectura advanced reduced instruction set computing machine con un servidor LAMP (Linux Apache MySQL PHP) que está conectado a una tarjeta de control que recoge los datos provenientes de los sensores para ser procesados y a su vez envía las ordenes a los actuadores (circuito de potencia)

5.4. IMPLEMENTACION

5.4.1. Descripción del domicilio

Como se puede apreciar en el siguiente plano tenemos casa con una distribución 3 dormitorios, una sala de estudio y una sala comedor.

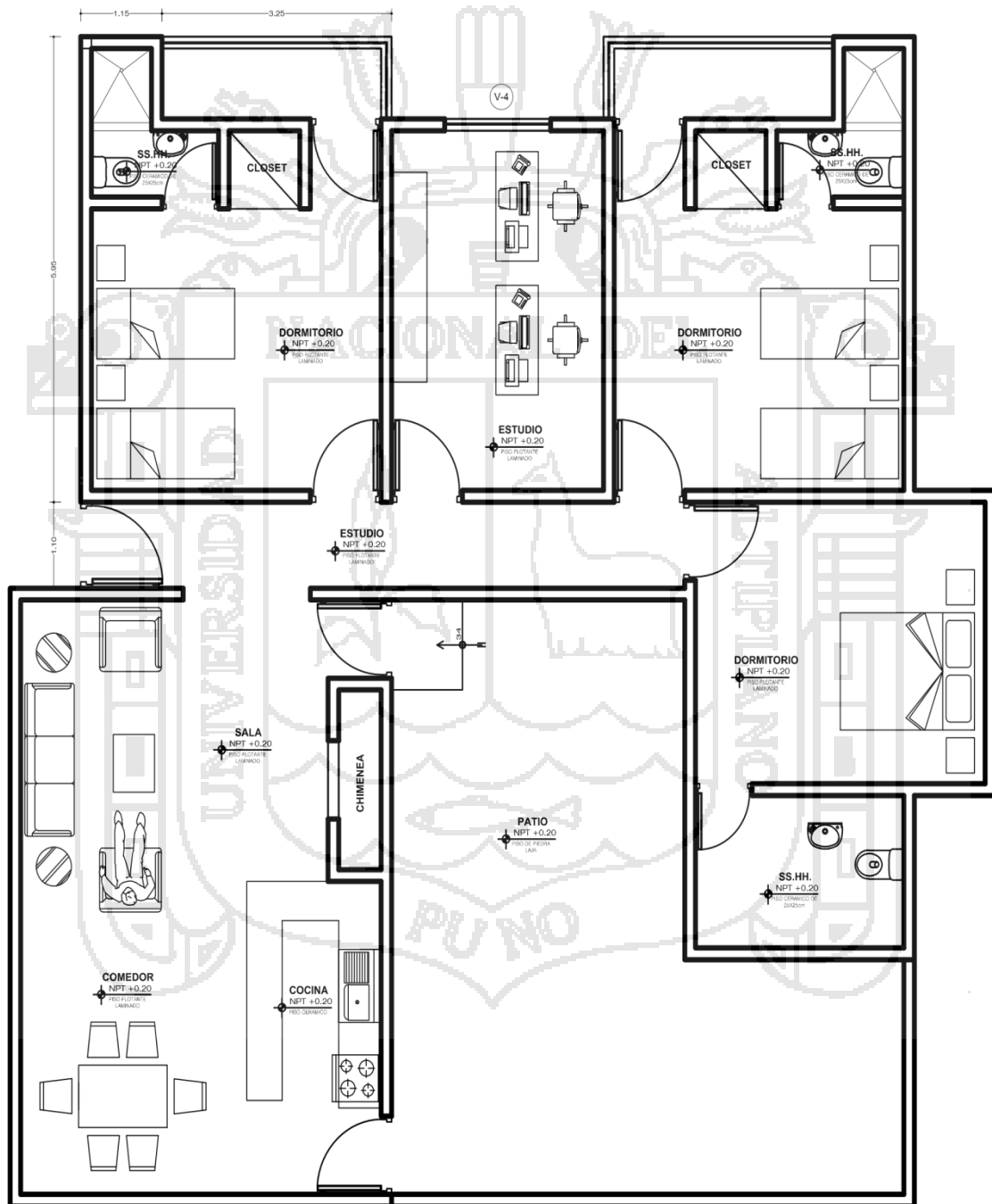


Imagen 20: Plano de la vivienda



Imagen 21: Maqueta de la vivienda

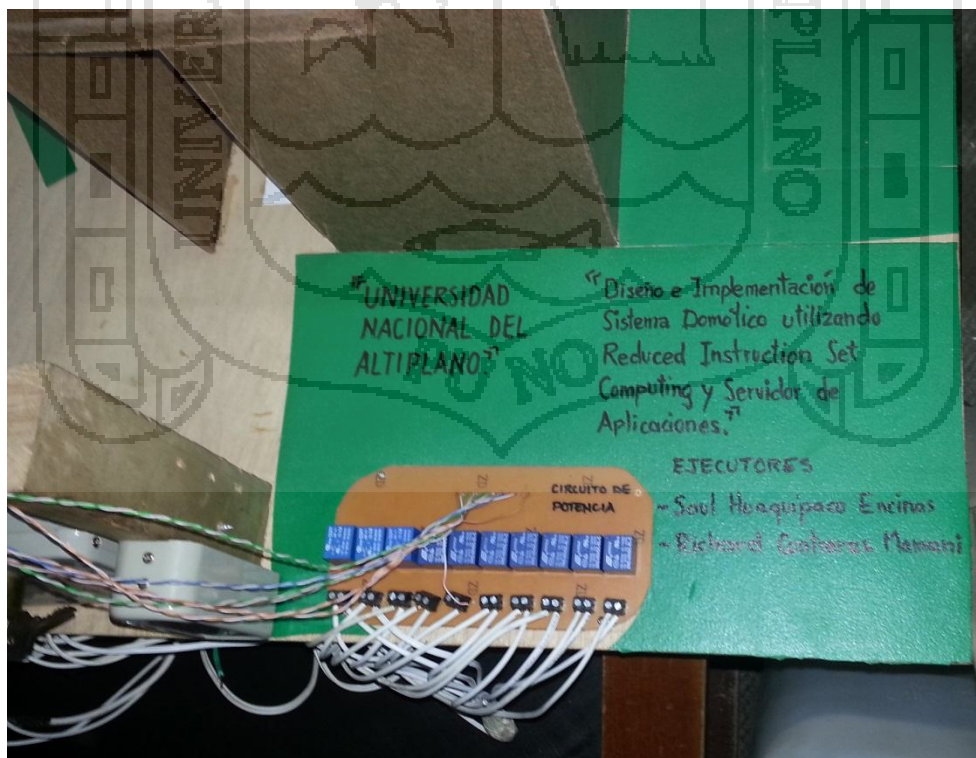


Imagen 22: Etapa de potencia de maqueta de la vivienda

5.4.2. *Instalación* del Software en el servidor

5.4.2.1. Software necesario

- Win32 Disk imager,
- MD5 & SHA-1 Checksum Utility,
- Raspbian “Wheezy”,

5.4.2.2. Descarga de software de instalación

1. Ir a la página <<http://www.raspberrypi.org>>.
2. Ir a la sección de descargas – downloads- y bajarnos el archivo .zip.

Se nos abrirá una página nueva al mismo tiempo que empezará la descarga. No cerrar la página ya que necesitaremos de su contenido para verificar la correcta descarga del archivo.

Descargar MD5 & SHA-1 Checksum utility

3. Ir al siguiente link. De la pantalla que aparece, darle a “Descargar ahora”.

Descarga del Sistema Operativo.

4. Ejecutar el MD5 & SHA-1 Checksum Utility:

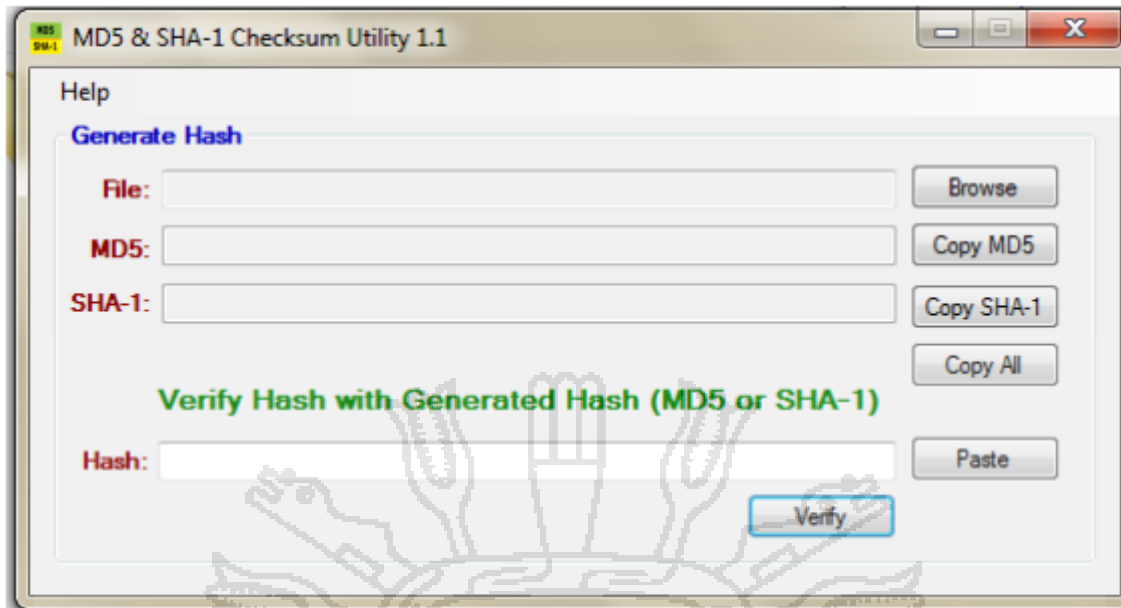


Imagen 23: MD5 & SHA-1 Checksum Utility.

5. Le damos a Browse y vamos a buscar el archivo .zip que contiene Raspbian.
6. Volvemos al navegador y recuperamos la página que se nos había abierto cuando habíamos empezado a descargar el archivo. Copiamos y pegamos el SHA-1 Checksum:
7. Finalmente para comparar el hash le damos al botón Verify:

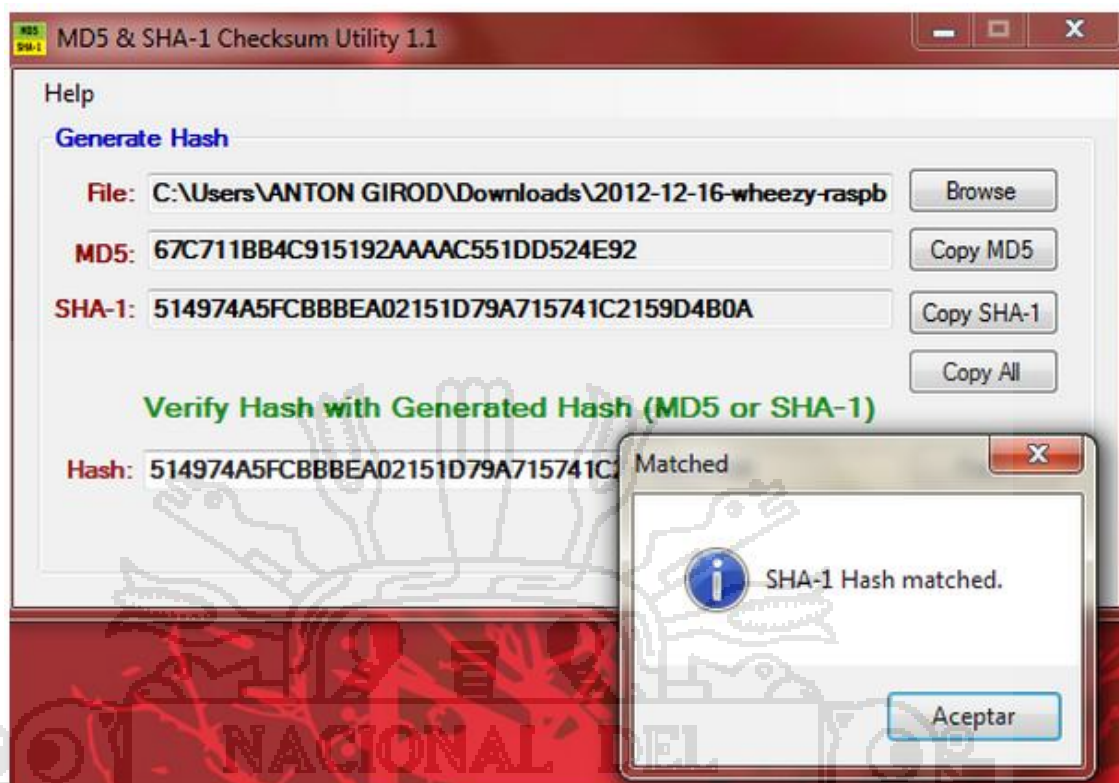


Imagen 24: MD5 & SHA-1 Checksum Utility

En el ejemplo de arriba vemos dos Hash que no se corresponden, si esto sucede tenemos que volver a descargar el sistema operativo. En cambio, en la siguiente imagen vemos como se corresponden y que por lo tanto tenemos un S.O. correcto:

Si esta imagen aparece, la descarga habrá sido exitosa.

8. Descarga e instalación de Win32DiskImager
9. Para instalarlo, simplemente descomprimir el archivo que obtengamos donde queramos.

5.4.2.3. Instalación del sistema operativo

1. Descomprime el archivo .zip que contiene Raspbian.
2. Inserta la SD al PC.

3. Abre el Win32DiskImager como administrador, donde pone device seleccionamos la SD y donde aparece el icono de la carpeta seleccionamos el archivo Raspbian.
4. Le damos a Write y esperamos a que escriba en la SD el Sistema Operativo.
5. Si todo ha ido bien, una vez se termine el proceso de escribir tendremos el Sistema Operativo instalado en la SD.
6. Finalmente, ya podemos insertar nuestra tarjeta SD en Raspberry Pi.

5.4.2.4. Primer arranque

1. En el primer arranque se nos llevará a la ventana de configuración de Raspi-config, donde podremos cambiar parámetros como la hora local. Si no, lo podemos hacer más adelante. Seleccionar la opción `expand_rootfs` y decir 'yes' para resetear.
2. La raspberry pi se reseteará y tendremos que logearnos; ponemos: Type: 'pi' y password: 'raspberrry'.
3. En ese momento las líneas de comando aparecerán como `pi@raspberrry ~ $`.
4. Ahora podemos arrancar la interfaz gráfica, escribimos en la línea de comandos: 'startx'.
5. Arrancará un entorno gráfico

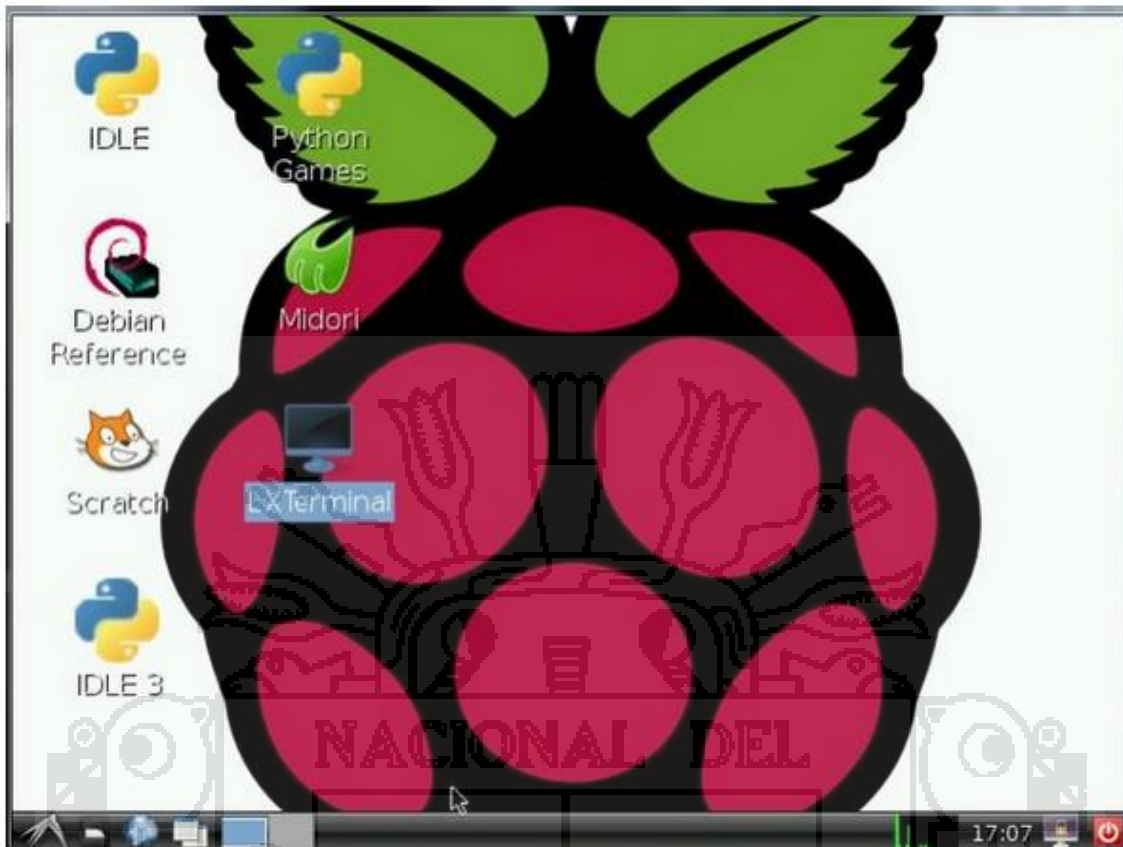


Imagen 25: Entorno gráfico Raspbian Wheezy

5.4.2.5. Instalación de Apache2

Apache2 está disponible como paquete Debian, por lo tanto podemos instalarlo así:

```
apt-get install apache2
```

Para verificar su instalación nos dijimos a <http://192.168.0.100>,

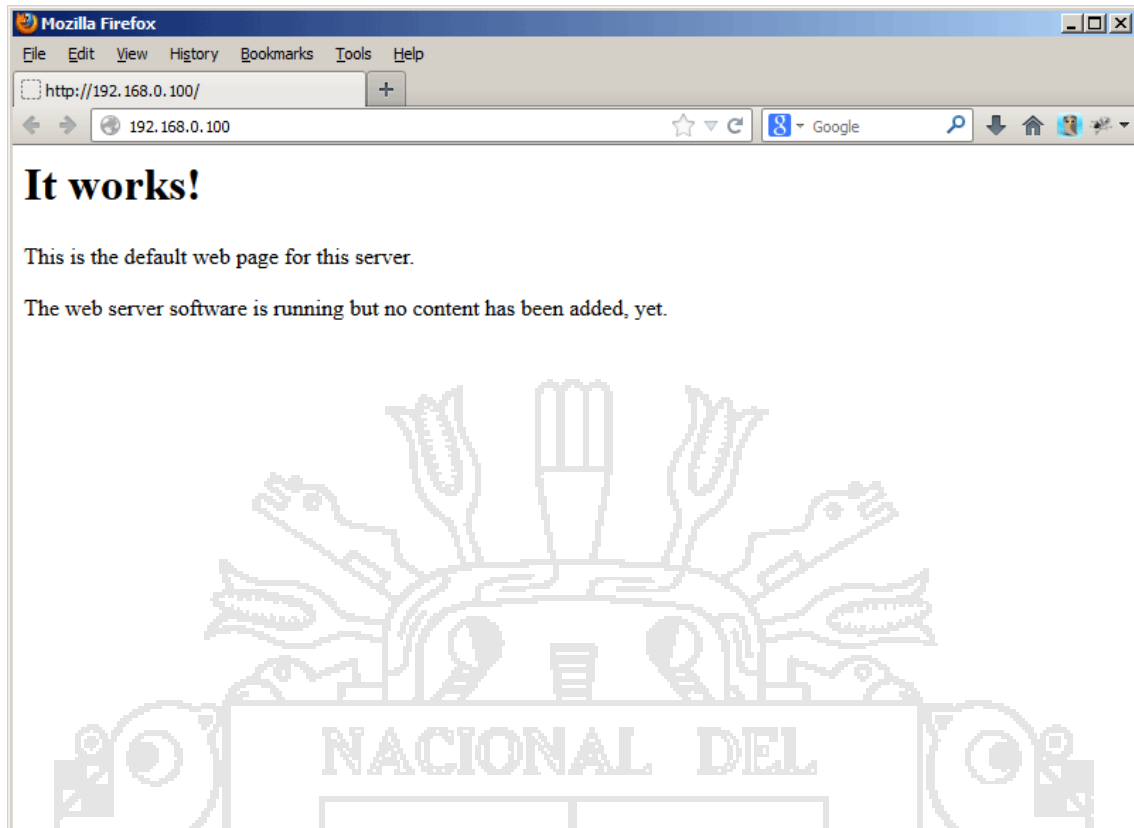


Imagen 26: página de muestra del local host

5.4.2.6. Instalación de Mysql

Primero instalamos MySQL como esto:

```
apt-get install mysql-server mysql-client
```

Se le pedirá que proporcione una contraseña para el usuario root de MySQL - esta contraseña es válida para el usuario root @ localhost, así como root@server1.example.com, por lo que no tiene que especificar una contraseña de root de MySQL manualmente más tarde.

5.4.2.7. Instalación de Php

Podemos instalar PHP5 y el módulo de Apache PHP5 como sigue:

```
apt-get install php5 libapache2-mod-php5
```

Debemos reiniciar Apache después:

```
/ Etc/init.d/apache2 restart
```

5.4.2.8. Configuración de la Web.

Para el funcionamiento de la web se necesita de los siguientes archivos en la carpeta del host de la página.

a) Índice.php

```
<?php
session_start();
if(!isset($_SESSION['user'])||empty($_SESSION['user'])){
    header('location:login.php');
}
?>
<!DOCTYPE HTML>
<html lang="en-US">
<head>
    <meta charset="UTF-8">
    <title></title>
    <script src="jquery-1.6.2.min.js"></script>
    <script type="text/javascript">
        function cambiarLed(ledButton, ledLetra){//
```

```
var texto=$(ledButton).val();

if(texto=='On'){

    estadoLed=ledLetra.toLowerCase();//

    $(ledButton).val('Off');

}else{

    estadoLed=ledLetra.toUpperCase();//

    $(ledButton).val('On');

}

$.ajax({

    url:'arduino2.php?led='+estadoLed,

});

</script>

</head>

<body>

<div>

<h1>SERVIDOR DOMOTICO "TESIS GRADO 2013"</h1>

</div>

<div>

<a href="logout.php">Salir</a>

</div>

<div>

<label>Tomacorrientes</label>

<input type="button" onclick="cambiarLed(this, 'a')

value="Off" />
```

```
</div>

<div>

    <label>Dormitorio I </label>

    <input type="button" onclick="cambiarLed(this, 'b')
value="Off" />

</div>

<div>

    <label>Sala</label>

    <input type="button" onclick="cambiarLed(this, 'c')
value="Off" />

</div>

<div>

    <label>Aire Acondicionado</label>

    <input type="button" onclick="cambiarLed(this, 'd')
value="Off" />

</div>

<div>

    <label>Cocina</label>

    <input type="button" onclick="cambiarLed(this, 'e')
value="Off" />

</div>

<div>

    <label>Almacen</label>

    <input type="button" onclick="cambiarLed(this, 'f')
value="Off" />
```

```
</div>

<div>

    <label>Dormitorio III</label>

    <input type="button" onclick="cambiarLed(this, 'g')
value="Off" />

</div>

<div>

    <label>Dormitorio II</label>

    <input type="button" onclick="cambiarLed(this, 'h')
value="Off" />

</div>

<div>

    <label>Estudio</label>

    <input type="button" onclick="cambiarLed(this, 'i')
value="Off" />

</div>

<div>

    <label>Puerta</label>

    <input type="button" onclick="cambiarLed(this, 'j')
value="Off" />

</div>

<div>

    <label>Calefaccion</label>

    <input type="button" onclick="cambiarLed(this, 'k')
value="Off" />
```

```

</div>

<div>

    <label>Jardin</label>

    <input type="button" onclick="cambiarLed(this, 'k'"
value="Off" />

</div>

<div>

    <label>Genegal</label>

    <input type="button" onclick="cambiarLed(this,
'abcdefghijkl)' value="Off" />

</div>
</body>
</html>

```

b) Conexión.php

```

<?php

    $datos=$_GET['led'];

    ini_set('display_errors', '1');

    include ("php_serial.class.php");

    $serial = new phpSerial;

    $serial->deviceSet('/dev/ttyACM0');

    $serial->confBaudRate(9600);

    $serial->confCharacterLength(8);

    $serial->confParity("none");

    $serial->confStopBits(1);

```

```

$serial->confFlowControl("none");

$serial->deviceOpen();

    $serial->sendMessage($datos."\r");

        $serial->deviceClose();

?>

<html></html>

```

c) Php_serial.class.php

```

<?php
define ("SERIAL_DEVICE_NOTSET", 0);
define ("SERIAL_DEVICE_SET", 1);
define ("SERIAL_DEVICE_OPENED", 2);
class phpSerial
{
    var $_device = null;
    var $_windevice = null;
    var $_dHandle = null;
    var $_dState = SERIAL_DEVICE_NOTSET;
    var $_buffer = "";
    var $_os = "";

    /**
     * This var says if buffer should be flushed by sendMessage (true)
     or manually (false)
     *

```

```
* @var bool
*/
var $autoflush = true;
/**
 * Constructor. Perform some checks about the OS and setserial
 *
 * @return phpSerial
 */
function phpSerial ()
{
    setlocale(LC_ALL, "en_US");
    $sysname = php_uname();
    if (substr($sysname, 0, 5) === "Linux")
    {
        $this->_os = "linux";
        if($this->_exec("stty --version") === 0)
        {
            register_shutdown_function(array($this,
"deviceClose"));
        }
    }
    else
    {
        trigger_error("No stty available, unable to run.",
E_USER_ERROR);
    }
}
```

```

    }

    elseif(substr($sysname, 0, 7) === "Windows")
    {
        $this->_os = "windows";

        register_shutdown_function(array($this,
            "deviceClose"));
    }
    else
    {
        trigger_error("Host OS is neither linux nor windows,
unable tu run.", E_USER_ERROR);

        exit();
    }
}
//
// OPEN/CLOSE DEVICE SECTION -- {START}
//
/**
 * Device set function : used to set the device name/address.
 * -> linux : use the device address, like /dev/ttyS0
 * -> windows : use the COMxx device name, like COM1 (can also
be used
 * with linux)
 *
 * @param string $device the name of the device to be used

```



```

* @return bool

*/

function deviceSet ($device)
{
    if ($this->_dState !== SERIAL_DEVICE_OPENED)
    {
        if ($this->_os === "linux")
        {
            if (preg_match("@^COM(\d+):?@$i", $device,
                $matches))
            {
                $device = "/dev/ttyS" . ($matches[1] -
                    1);
            }
            if ($this->_exec("stty -F " . $device) === 0)
            {
                $this->_device = $device;
                $this->_dState =
                    SERIAL_DEVICE_SET;
                return true;
            }
        }
        elseif ($this->_os === "windows")
        {

```

```

        if (preg_match("@^COM(\d+):?$@i", $device,
$matches) and $this->_exec(exec("mode " . $device)) === 0)
        {
            $this->_windevice    =    "COM"    .
$matches[1];
            $this->_device      =    "\\.\com"  .
$matches[1];
            $this->_dState      =
SERIAL_DEVICE_SET;
            return true;
        }
        trigger_error("Specified serial port is not valid",
E_USER_WARNING);
        return false;
    }
    else
    {
        trigger_error("You must close your device before to
set an other one", E_USER_WARNING);
        return false;
    }
}
}

/**

```

```

* Opens the device for reading and/or writing.
*
* @param string $mode Opening mode : same parameter as
fopen()
* @return bool
*/
function deviceOpen ($mode = "r+b")
{
    if ($this->_dState === SERIAL_DEVICE_OPENED)
    {
        trigger_error("The device is already opened",
E_USER_NOTICE);
        return true;
    }
    if ($this->_dState === SERIAL_DEVICE_NOTSET)
    {
        trigger_error("The device must be set before to be
open", E_USER_WARNING);
        return false;
    }

    if (!preg_match("@^[raw]\+?b?$@", $mode))
    {

```

```

        trigger_error("Invalid opening mode : ".$mode.". Use
fopen() modes.", E_USER_WARNING);

        return false;

    }

    $this->_dHandle = fopen($this->_device, $mode);

    if ($this->_dHandle !== false)
    {
        stream_set_blocking($this->_dHandle, 0);
        $this->_dState = SERIAL_DEVICE_OPENED;
        return true;
    }
    $this->_dHandle = null;
    trigger_error("Unable to open the device",
E_USER_WARNING);
    return false;
}
/**
 * Closes the device
 *
 * @return bool
 */

function deviceClose ()
{
    if ($this->_dState !== SERIAL_DEVICE_OPENED)

```

```
{  
    return true;  
}  
if (fclose($this->_dHandle))  
{  
    $this->_dHandle = null;  
    $this->_dState = SERIAL_DEVICE_SET;  
    return true;  
}  
  
    trigger_error("Unable to close the device",  
E_USER_ERROR);  
    return false;  
}  
//  
// OPEN/CLOSE DEVICE SECTION -- {STOP}  
//  
//  
// CONFIGURE SECTION -- {START}  
//  
  
/**  
 * Configure the Baud Rate  
 * Possible rates : 110, 150, 300, 600, 1200, 2400, 4800, 9600,  
38400,
```

```
* 57600 and 115200.  
  
*  
  
* @param int $rate the rate to set the port in  
  
* @return bool  
  
*/  
  
function confBaudRate ($rate)  
{  
    if ($this->_dState !== SERIAL_DEVICE_SET)  
    {  
        trigger_error("Unable to set the baud rate : the  
device is either not set or opened", E_USER_WARNING);  
        return false;  
    }  
    $validBauds = array (  
        110 => 11,  
        150 => 15,  
        300 => 30,  
        600 => 60,  
        1200 => 12,  
        2400 => 24,  
        4800 => 48,  
        9600 => 96,  
        19200 => 19,  
        38400 => 38400,  
        57600 => 57600,  
    );  
}
```

```
115200 => 115200

);
if (isset($validBauds[$rate]))
{
    if ($this->_os === "linux")
    {
        $ret = $this->_exec("stty -F " . $this->_device .
" " . (int) $rate, $out);
    }
    elseif ($this->_os === "windows")
    {
        $ret = $this->_exec("mode " . $this->_windevice . " BAUD=" . $validBauds[$rate], $out);
    }
    else return false;

    if ($ret !== 0)
    {
        trigger_error ("Unable to set baud rate: " .
$out[1], E_USER_WARNING);
        return false;
    }
}
}
```

```
/**
 * Configure parity.
 * Modes : odd, even, none
 *
 * @param string $parity one of the modes
 * @return bool
 */
function confParity ($parity)
{
    if ($this->_dState !== SERIAL_DEVICE_SET)
    {
        trigger_error("Unable to set parity : the device is
either not set or opened", E_USER_WARNING);
        return false;
    }

    $args = array(
        "none" => "-parenb",
        "odd" => "parenb parodd",
        "even" => "parenb -parodd",
    );

    if (!isset($args[$parity]))
    {
```



```

        trigger_error("Parity mode not supported",
E_USER_WARNING);

        return false;

    }

    if ($this->_os === "linux")
    {
        $ret = $this->_exec("stty -F " . $this->_device . " " .
    $args[$parity], $out);
    }
    else
    {
        $ret = $this->_exec("mode " . $this->_windevice . "
    PARITY=" . $parity{0}, $out);
    }
    if ($ret === 0)
    {
        return true;
    }

    trigger_error("Unable to set parity : " . $out[1],
E_USER_WARNING);

    return false;

}

/**

```

```

* Sets the length of a character.
*
* @param int $int length of a character (5 <= length <= 8)
* @return bool
*/
function confCharacterLength ($int)
{
    if ($this->_dState !== SERIAL_DEVICE_SET)
    {
        trigger_error("Unable to set length of a character :
the device is either not set or opened", E_USER_WARNING);
        return false;
    }
    $int = (int) $int;
    if ($int < 5) $int = 5;
    elseif ($int > 8) $int = 8;
    if ($this->_os === "linux")
    {
        $ret = $this->_exec("stty -F " . $this->_device . " cs" .
$int, $out);
    }
    else
    {
        $ret = $this->_exec("mode " . $this->_windevice . "
DATA=" . $int, $out);
    }
}

```

```
    }  
  
    if ($ret === 0)  
    {  
  
        return true;  
  
    }  
  
    trigger_error("Unable to set character length : " . $out[1],  
E_USER_WARNING);  
    return false;  
}  
  
/**  
 * Sets the length of stop bits.  
 *  
 * @param float $length the length of a stop bit. It must be either 1,  
 * 1.5 or 2. 1.5 is not supported under linux and on some  
computers.  
 * @return bool  
 */  
function confStopBits ($length)  
{  
  
    if ($this->_dState !== SERIAL_DEVICE_SET)  
    {  
  
        trigger_error("Unable to set the length of a stop bit :  
the device is either not set or opened", E_USER_WARNING);  
  
        return false;  
  
    }  
  
}
```

```

        if ($length != 1 and $length != 2 and $length != 1.5 and
!($length == 1.5 and $this->_os === "linux"))
        {
            trigger_error("Specified stop bit length is invalid",
E_USER_WARNING);

            return false;
        }
        if ($this->_os === "linux")
        {
            $ret = $this->_exec("stty -F " . $this->_device . " " .
(($length == 1) ? "-" : "") . "cstopb", $out);
        }
        else
        {
            $ret = $this->_exec("mode " . $this->_windevice . "
STOP=" . $length, $out);
        }
        if ($ret === 0)
        {
            return true;
        }

        trigger_error("Unable to set stop bit length : " . $out[1],
E_USER_WARNING);

        return false;
    }

```

```

/**
 * Configures the flow control
 *
 * @param string $mode Set the flow control mode. Available
modes :
 * -> "none" : no flow control
 * -> "rts/cts" : use RTS/CTS handshaking
 * -> "xon/xoff" : use XON/XOFF protocol
 * @return bool
 */
function confFlowControl ($mode)
{
    if ($this->_dState !== SERIAL_DEVICE_SET)
    {
        trigger_error("Unable to set flow control mode : the
device is either not set or opened", E_USER_WARNING);
        return false;
    }
    $linuxModes = array(
        "none" => "clocal -crtscts -ixon -ixoff",
        "rts/cts" => "-clocal crtscts -ixon -ixoff",
        "xon/xoff" => "-clocal -crtscts ixon ixoff"
    );
    $windowsModes = array(
        "none" => "xon=off octs=off rts=on",

```

```

"rts/cts" => "xon=off octs=on rts=hs",
"xon/xoff" => "xon=on octs=off rts=on",
);

if ($mode != "none" and $mode != "rts/cts" and $mode
!= "xon/xoff") {
    trigger_error("Invalid flow control mode specified",
E_USER_ERROR);
    return false;
}

if ($this->_os === "linux")
    $ret = $this->_exec("stty -F " . $this->_device . " " .
$linuxModes[$mode], $out);
else
    $ret = $this->_exec("mode " . $this->_windevice . " " .
$windowsModes[$mode], $out);
if ($ret === 0) return true;
else {
    trigger_error("Unable to set flow control : " . $out[1],
E_USER_ERROR);
    return false;
}
}

/**
* Sets a setserial parameter (cf man setserial)

```

```

* NO MORE USEFUL !

*   -> No longer supported

*   -> Only use it if you need it

*

* @param string $param parameter name

* @param string $arg parameter value

* @return bool

*/

function setSetserialFlag ($param, $arg = "")
{
    if (!$this->_ckOpened()) return false;

    $return = exec ("setserial " . $this->_device . " " . $param . "
" . $arg . " 2>&1");

    if ($return{0} === "I")
    {
        trigger_error("setserial: Invalid flag",
E_USER_WARNING);

        return false;
    }

    elseif ($return{0} === "/")
    {
        trigger_error("setserial: Error with device file",
E_USER_WARNING);

        return false;
    }
}

```

```
        else
        {
            return true;
        }
    }

    //
    // CONFIGURE SECTION -- {STOP}
    //
    //
    // I/O SECTION -- {START}
    //
    /**
     * Sends a string to the device
     *
     * @param string $str string to be sent to the device
     * @param float $waitForReply time to wait for the reply (in
seconds)
     */
    function sendMessage ($str, $waitForReply = 0.1)
    {
        $this->_buffer .= $str;

        if ($this->autoflush === true) $this->flush();

        usleep((int) ($waitForReply * 1000000));
    }
}
/**
```


* Reads the port until no new datas are available, then return the content.

*

* @param int \$count number of characters to be read (will stop before

* if less characters are in the buffer)

* @return string

*/

function readPort (\$count = 0)

{

if (\$this->_dState !== SERIAL_DEVICE_OPENED)

{

trigger_error("Device must be opened to read it",
E_USER_WARNING);

return false;

}

if (\$this->_os === "linux")

{

\$content = ""; \$i = 0;

if (\$count !== 0)

{

do {

if (\$i > \$count) \$content .= fread(\$this->

>_dHandle, (\$count - \$i));

```

else $content .= fread($this->_dHandle,

128);

    } while (($i += 128) === strlen($content));

    }

    else

    {

        do {

            $content .= fread($this->_dHandle,

128);

        } while (($i += 128) === strlen($content));

    }

    return $content;

}

elseif ($this->_os === "windows")

{

    /* Do nothing : not implented yet */

}

trigger_error("Reading serial port is not implemented for

Windows", E_USER_WARNING);

return false;

}

/**

 * Flushes the output buffer

 *

 * @return bool

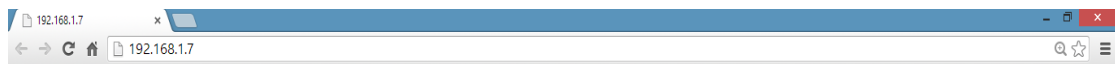
```

```
*/  
  
function flush ()  
{  
    if (!$this->_ckOpened()) return false;  
  
    if (fwrite($this->_dHandle, $this->_buffer) !== false)  
    {  
        $this->_buffer = "";  
        return true;  
    }  
    else  
    {  
        $this->_buffer = "";  
        trigger_error("Error while sending message",  
E_USER_WARNING);  
        return false;  
    }  
}  
//  
// I/O SECTION -- {STOP}  
  
//  
  
//  
  
// INTERNAL TOOLKIT -- {START}
```

```
//  
  
function _ckOpened()  
{  
    if ($this->_dState !== SERIAL_DEVICE_OPENED)  
    {  
        trigger_error("Device must be opened",  
E_USER_WARNING);  
        return false;  
    }  
    return true;  
}  
  
function _ckClosed()  
{  
    if ($this->_dState !== SERIAL_DEVICE_CLOSED)  
    {  
        trigger_error("Device must be closed",  
E_USER_WARNING);  
        return false;  
    }  
    return true;  
}  
  
function _exec($cmd, &$out = null)  
{  
    $desc = array(  
        1 => array("pipe", "w"),
```

```
        2 => array("pipe", "w")
    );
    $proc = proc_open($cmd, $desc, $pipes);
    $ret = stream_get_contents($pipes[1]);
    $err = stream_get_contents($pipes[2]);
    fclose($pipes[1]);
    fclose($pipes[2]);
    $retVal = proc_close($proc);
    if (func_num_args() == 2) $out = array($ret, $err);
    return $retVal;
}
//
// INTERNAL TOOLKIT -- {STOP}
//
?>
```

5.4.2.9. Interfaz Web del sistema.



SERVIDOR DOMOTICO "TESIS GRADO 2013"

Tomacorrientes On
Dormitorio I On
Sala On
Aire Acondicionado On
Cocina On
Almacen On
Dormitorio III On
Dormitorio II On
Estudio On
Puerta Off
CalefaccionLed k Off
Jardin Off
General On

Imagen 27: interfaz web del sistema domótico

5.4.2.10. Programación del microcontrolador

```
int led1 = 2;
```

```
int led2 = 3;
```

```
int led3 = 4;
```

```
int led4 = 5;
```

```
int led5 = 6;
```

```
int led6 = 7;
```

```
int led7 = 8;
```

```
int led8 = 9;
```

```
int led9 = 10;

int led10 = 11;

int led11 = 12;

int incomingbyte;

void setup()
{
  Serial.begin(9600);

  pinMode(led1,OUTPUT);

  pinMode(led2,OUTPUT);

  pinMode(led3,OUTPUT);

  pinMode(led4,OUTPUT);

  pinMode(led5,OUTPUT);

  pinMode(led6,OUTPUT);

  pinMode(led7,OUTPUT);

  pinMode(led8,OUTPUT);

  pinMode(led9,OUTPUT);

  pinMode(led10,OUTPUT);

  pinMode(led11,OUTPUT);

}

void loop()
```

```
{  
  
    if(Serial.available()>0)  
  
    {  
  
        incomingbyte = Serial.read();  
  
    }  
  
    if(incomingbyte == 'A')  
    {digitalWrite(led1,HIGH);}  
  
    if(incomingbyte == 'a')  
    {digitalWrite(led1,LOW);}  
  
        if(incomingbyte == 'B')  
    {digitalWrite(led2,HIGH);}  
  
    if(incomingbyte == 'b')  
    {digitalWrite(led2,LOW);}  
  
    if(incomingbyte == 'C')  
    {digitalWrite(led3,HIGH);}  
  
    if(incomingbyte == 'c')  
  
    {digitalWrite(led3,LOW);}  
  
    if(incomingbyte == 'D')  
  
    {digitalWrite(led4,HIGH);}  
  
    if(incomingbyte == 'd')
```



```
{digitalWrite(led4,LOW);}
```

```
if(incomingbyte == 'E')
```

```
{digitalWrite(led5,HIGH);}
```

```
if(incomingbyte == 'e')
```

```
{digitalWrite(led5,LOW);}
```

```
if(incomingbyte == 'F')
```

```
{digitalWrite(led6,HIGH);}
```

```
if(incomingbyte == 'f')
```

```
{digitalWrite(led6,LOW);}
```

```
if(incomingbyte == 'G')
```

```
{digitalWrite(led7,HIGH);}
```

```
if(incomingbyte == 'g')
```

```
{digitalWrite(led7,LOW);}
```

```
if(incomingbyte == 'H')
```

```
{digitalWrite(led8,HIGH);}
```

```
if(incomingbyte == 'h')
```

```
{digitalWrite(led8,LOW);}
```

```
if(incomingbyte == 'I')
```

```
{digitalWrite(led9,HIGH);}
```

```
if(incomingbyte == 'i')
```

```
{digitalWrite(led9,LOW);}
```

```
if(incomingbyte == 'J')
```

```
{digitalWrite(led10,HIGH);}
```

```
if(incomingbyte == 'j')
```

```
{digitalWrite(led10,LOW);}
```

```
if(incomingbyte == 'K')
```

```
{digitalWrite(led11,HIGH);}
```

```
if(incomingbyte == 'k')
```

```
{digitalWrite(led11,LOW);}
```



5.5. ANÁLISIS DE RESULTADOS

5.5.1. Consumo energético de servidor

Mediante las pruebas realizadas pudimos comprobar que el consumo total de energía eléctrica del servidor es muy baja

La fuente a alimentación del servidor es de 5 voltios con 1 amperio de corriente y por cuanto la energía consumida por un dispositivo eléctrico se mide en vatios-hora (Wh) entonces tenemos:

Consumo energético por hora (Wh) = Voltage X Corriente

- $Wh = 5 \times 1$

Por lo tanto nuestro servidor consume 5 watts hora lo que quiere decir que en 24 horas consume:

- $5 \text{ Watts} \times 24 = 120 \text{ Watts}$

Por lo tanto en un mes consume:

- $120 \text{ watts} \times 30 = 3600 \text{ Watts}$ o 3.6 kilowatts

Considerando que el costo promedio del Kilowatt es de 0.65 céntimos de sol tenemos.

Costo en soles consumo energético

3.6 kilowatts X 0.65 céntimos = 2.34 soles

En una Comparación en consumo energético entre nuestro servidor y otros servidores:

a) Servidor con arquitectura RISC, 5 watts hora

Watts por hora X 24 Horas X 30 días X 0.00065 nuevos soles = costo mensual de uso energía.

○ $5 \times 24 \times 30 \times 0.00065 = 2.34$ nuevos soles

b) Servidor Intel i3 arquitectura CISC, 250 watts hora

○ $250 \times 24 \times 30 \times 0.00065 = 117$ nuevos soles

c) Servidor Profesional Xenon E3-1200, 450 watts hora.

○ $450 \times 24 \times 30 \times 0.00065 = 210.6$ nuevos soles

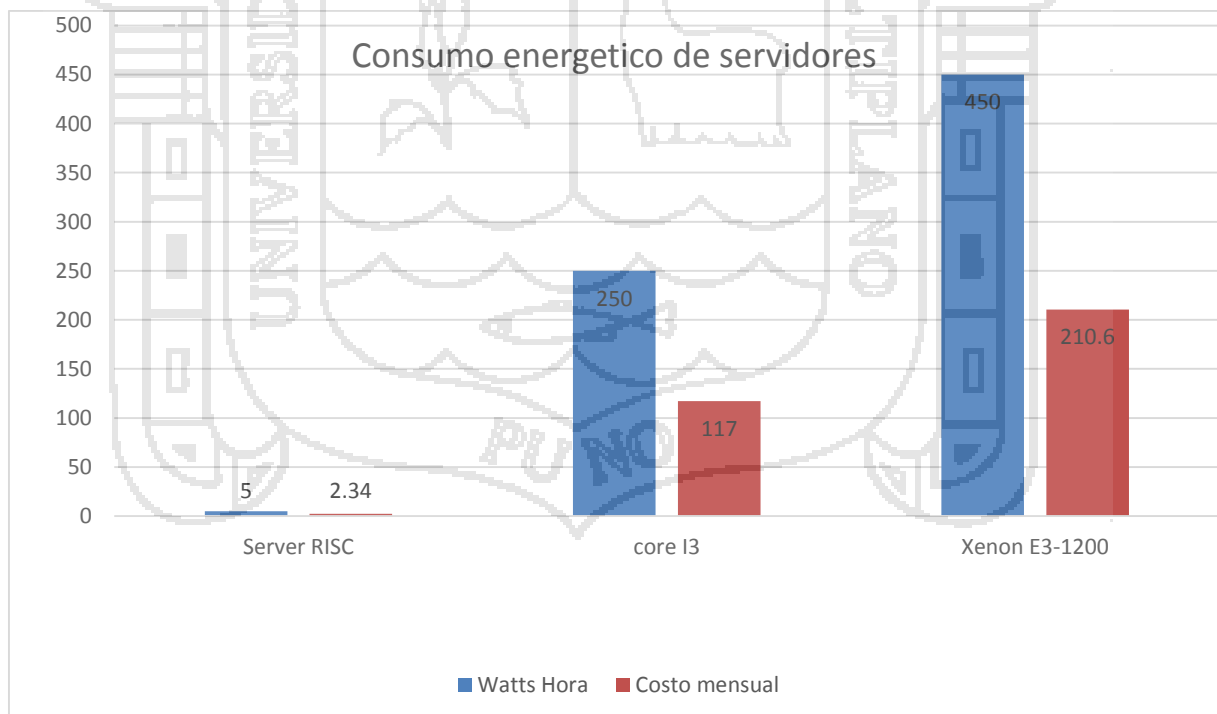


Imagen 28: Consumo energético de servidores

5.5.2. Costo de servidores

El costo de los servidores varía dependiendo de las características y de las aplicaciones que uno necesite para el caso de este estudio se tomaron en cuenta tres tipos de servidores.

- Servidor con arquitectura advanced reduced instruction set computing machine.
- Servidor con arquitectura CISC con procesador Intel core i3.
- Servidor con arquitectura CISC con procesador Intel xenón E3-1200

Los costos de servidores con los siguientes:

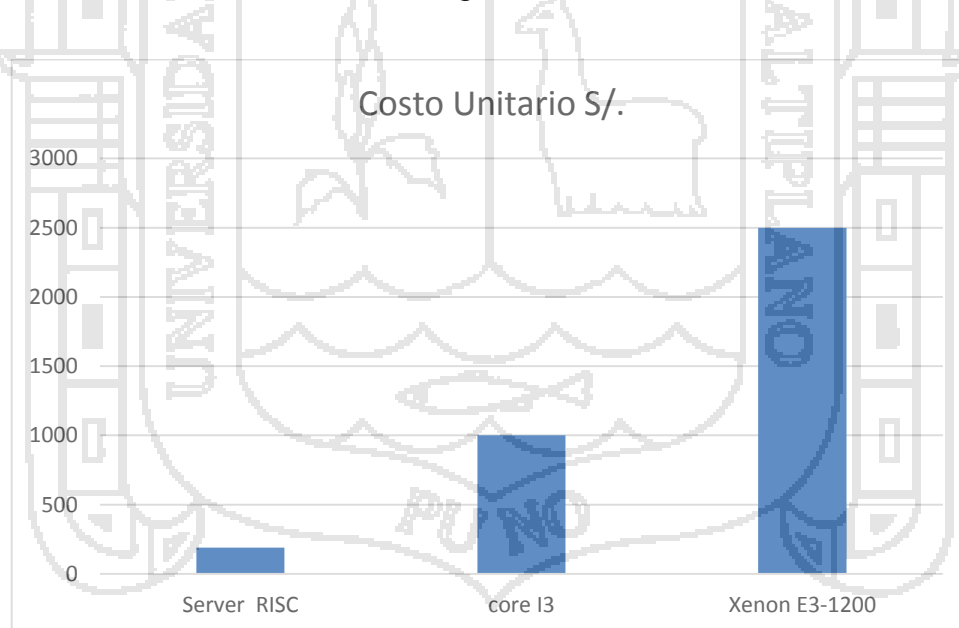


Imagen 29: Comparativa del consumo unitario entre servidores.

CONCLUSIONES

Primera

Se realizó un estudio comparativo entre los diversos sistemas domóticos mostrando las ventajas y desventajas de los mismos en cuanto a costos hasta y consumo energético hasta 445 watt hora menos.

Segunda

Se diseñó un sistema domótico usando como hardware del servidor un computador de bajo costo (190 nuevos soles) y consumo energético (5 watts hora) con arquitectura Reducer instruction set computing y usando como software un servidor LAMP.

Tercera

Se implementó el sistema domótico con bajo costo económico ya que la mayor parte del diseño del prototipo está basado en software y hardware libre. Lo que permite además seguir investigando sobre el proyecto ya que no tiene restricciones de tipo propietario.

RECOMENDACIONES

Primera

Seguir investigación sobre aplicaciones web en la domótica ya que estas permiten gran compatibilidad con los diversos tipos de dispositivos móviles.

Segunda

Usar siempre las alternativas de software y hardware libre ya que estas no acarrear costos ni restricciones.

Tercera

Estudiar más configuración de seguridad en servidores linux ya que esto suele dificultar la configuración y puesta en marcha de los servidores.

BIBLIOGRAFÍA

- HERNANDEZ Sampieri, Douglas; Fernández Collado, Carlos y Baptista Pilar, Lucio. • Metodología de la Investigación. 3^a Ed. México, McGraw-Hill, 2003. 5p-111p. •
- O'DRISCOLL, G. The essential guide to Home Networking Technologies. Edit: • Prentice Hall, 2000. 125p.
- SÁNCHEZ Arias, D. Domótica: diseño de una casa inteligente basado en la tecnología • Jini. Tesis de Licenciatura. Ingeniería en Sistemas Computacionales, Universidad de las • Américas. 2004.
- D. Bonino, E. Castellina, and F. Corno. The DOG gateway: Enabling ontology-based intelligent domotic environments. Consumer Electronics, IEEE Transactions on, 54(4):1656 – 1664, November 2008.
- M. Celio, W. Laiolo, and J. Stirling. Estado del Arte, Domótica-Proyecto de grado PGSISMO.
- H. M. Domínguez and F. S. Vacas. Domótica: Un enfoque sociotécnico.
- Fundación Rogelio Segovia para el Desarrollo de las Telecomunicaciones, 2006.

- W. R. Escobar, Z. Pinzón, and P. Naranjo. Domótica: Ciencia ficción hecha realidad. *Ciencia e Ingeniería Neogranadina*, (011):59 – 65, 2001.
- F. M. S. Fiuza, K. A. O. Duarte, D. G. Vidaurrázaga, and J. D. de la Calle. Domótica: La Tecnología al servicio de la Autonomía Personal.
- Yregoi, S. R. Castañeda, and F. J. B. Rodríguez. Solucion de tecnologias de información para el ahorro de energia y agua en el hogar mediante domotica.
- S. P. Hernández. Consideraciones para la aplicación de la domótica desde la concepción del diseño arquitectónico. *Arquiteturarevista*, (1):63 – 75, 2010.
- S. Junstrand, X. Passaret, and D. Vázquez. Domótica y hogar digital.
- G. V. Madrid. Tema 5 aplicaciones. Soluciones comerciales.
- N. G. D. I. T. Martínez and M. C. J. M. R. Alcaraz. Desarrollo e implementación de un sistema domótico en un hogar del estado de colima.
- J. Munoz, J. Fons, V. Pelechano, and O. Pastor. Hacia el modelado conceptual de sistemas domóticos. Dpto. De Sistemas Informáticos y Computación, Universidad Politécnica de Valencia, Valencia, 2003.

- R. J. M. Tejedor. Domótica: Edificios inteligentes. Creaciones Copy-right, 2004.
- S. S. R. Valenzuela, M. D. S. Laguna, and J. A. Holgado. Control Domótico del Hogar a través de una Plataforma de Servicios Distribuida basada en JXTA.
- M. A. Zamora-Izquierdo, J. Santa, and A. F. Gomez-Skarmeta. An Integral and Networked Home Automation Solution for Indoor Ambient Intelligence. Pervasive Computing, IEEE, 9(4):66 – 77, October 2010.
- D. Bonino F., Corno F., Razzak. 2010. Mobile Interaction with Smart Environments through Linked Data
- Mafalda Seixas, João Palma. Remote Alarm And Command System For residential Domotics Trough Gsm – Sms
- J. Picerno, K. Tenzer. 2010. API de alto nivel genérico para desarrollo de aplicaciones de domótica.
- E. Martínez, R. Cabello, F. J. Gómez, J. Martínez. 2003. A Solution for the Integration of Domestic Devices on Network Management Platforms
- S. Soucek, G. Russ, C. Tamarit. 2001. The Smart Kitchen Project -An Application of Fieldbus Technology to Domotics.

- J. V. Meggelen, J. Smith, L. Madsen. 2007. Asterisk: The Future of Telephony. Segunda Edición. O'Reilly Media
- P. Pellegrino, D. Bonino, and F. Corno. 2006. Domotic house gateway. Marco Aiello. 2006. The role of web services at home.
- Edgar Landívar. 2008. Comunicaciones Unificadas con Elastix Volumen 1. Primera Edición. GNU Free Documentation License, Versión 1.3.
- Alfredo Certain. Asterisk. 2005 Comunicaciones de Código Abierto. Gecko Networks.
- Asociación 4G Américas. 2010. Textos al 9-1-1: Análisis del diseño y limitaciones de SMS
- Diego Javier Silva Abello. SIMULACIÓN DE UN AMBIENTE DOMÓTICO IMPLEMENTANDO TECNOLOGÍA JINI.
- Carlos Patricio Álvarez Barreno, Danilo Orlando Holguin Cabezas, and Ernesto Bolivar Serrano Guevara. Diseño de una instalación domótica en un condominio para el control de seguridad e iluminación mediante la tecnología

- LONWORKS. page 257, 2007. • María del Carmen Bueno Hernández. SISTEMA DOMÓTICO ORIENTADO a USABILIDAD.
- DISEÑO DE PLATAFORMA DE CONTROL VOCAL. Juan Carlos Vesga Ferreira. SISTEMA DE CONTROL DOMÓTICO UTILIZANDO LA RED ELÉCTRICA COMO MÉDIO FÍSICO DE TRANSMISIÓN.
- Juan Carlos Martín. Instalaciones domóticas. Editex, January 2009. • Jorge roberto Alvarado Cadena and Cesar Oswaldo Arevalo Coronel. DISEÑO e IMPLEMENTACIÓN DE UN SISTEMA DOMÓTICO PARA EL CONTROL y SEGURIDAD EN TIEMPO REAL VÍA TELÉFONO CELULAR.
- Wayne Tomásí. Sistemas de Comunicaciones Electrónicas. Pearson Educación, January 2003
- ROMERO, M. A.; Domótica – Edificios Inteligentes, 2003

WEBGRAFÍA

- DOMÓTICA SOLUCIONES Integrales SL, Introducción y tecnologías de la Domótica, 2002. <http://www.domotica.net>
- JUNESTRAND.S. El proyecto Hábitat 2010 2002 •
http://www.casadomo.com/revista_domotica_articles.asp
- Cisco – Voice Over IP: Per Call Bandwidth Consumption. Consulta 20 Mayo 2012.
- Atmel – Atmega8L Datasheet Download. Consulta 16 Noviembre 2011.
http://www.atmel.com/dyn/resources/prod_documents/doc2486.pdf
- Webestilo - Introducción a MySQL. Consulta 15 Febrero 2012
•<http://www.webestilo.com/mysql/intro.phtml>
- <http://arduino.cc/forum/index.php?PHPSESSID=1465a4f969ff5c751442f1266c938413&topic=32397.15>
- Conceptos basicos. Manual de MySQL. Tutorial de MySQL. WebEstilo.
www.webestilo.com
- Domodesk - A fondo: Zigbee. Consulta 8 Junio 2013.
<http://www.domodesk.com/content.aspx?co=97&t=21&c=47>

- Webestilo - Introducción a MySQL. Consulta 15 Agosto 2013
<http://www.webestilo.com/mysql/intro.phtml>
- W. Chamorro, D. Guerrón. 2008. Diseño e implemetación del control de acceso y seguridad del laboratorio de instrumentación utilizando el protocolo X-10
- 3CX – 10 reasons to switch to an IP PBX. Consulta 20 de Julio 2013
<http://www.3cx.com/PBX/pbx-benefits-wp.html>
- Wikipedia- Centos. Consulta 20 Julio 2013 <
<http://es.wikipedia.org/wiki/CentOS>>
- Savannah - Download. Consulta 1 Julio 2013.
<http://savannah.nongnu.org/projects/avr-libc/>
- Ethershield - Download. Consulta 1 Julio 2013. <
<http://ethershield.thiseldo.co.uk/index.html>>
- Microchip – Enc28j60 Datasheet Download. Consulta 16 Julio 2013.
<http://ww1.microchip.com/downloads/en/devicedoc/39662a.pdf>

- Midcom – Download. Consulta 16 Julio 2013. <http://products.midcom-inc.com/Products/Catalog/LANCatalog.pdf>
- Domodesk - DD-5230 SMS. Consulta 20 Julio 2013. [http://www.domodesk.com/product/9/14/12/1/SMS_Control_ADVANCE_US_B_SOFTWARE_\(ENCENDIDO_ALERTA_REMOTO_por_M%C3%93VIL\).htm](http://www.domodesk.com/product/9/14/12/1/SMS_Control_ADVANCE_US_B_SOFTWARE_(ENCENDIDO_ALERTA_REMOTO_por_M%C3%93VIL).htm)
- Domodesk - DD-6321. Consulta 20 Julio 2013. http://www.domodesk.com/product/58/14/52/1/CONTROLADOR_IP_POWER_ON
- http://www.domotia.com/tienda/product.php?id_product=19 Domodesk - DD-6390. Consulta 20 Julio 2013.
- http://www.domodesk.com/product/22/14/37/1/CONTROL_IP_por_INTERNET_RACK_8_entadas_Digitales_8_salidas_Rel%C3%A9.htm
- Detector PIR LC -100 PI – Download. Consulta 18 Julio de 2012 < http://www.hestia-france.com/downloads/Notice_LC100PI-1.pdf >
- Contacto magnetic SM-217Q- Download. Consulta 18 Julio de 2012
http://www.seco-larm.com/pdfs/SM-x17Q_PI_P0908.pdf