

**DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE CONTROL Y MONITOREO  
REMOTO PARA LA ILUMINACIÓN DE UNA VIVIENDA A TRAVÉS DE UN  
TELÉFONO MÓVIL CON SISTEMA OPERATIVO ANDROID Y BOARD ARM  
MINI2440**

**FERNANDO AUGUSTO ESPAÑA CANO  
ELIANA FERNANDA TACÁN CUELLAR**

**UNIVERSIDAD SURCOLOMBIANA  
FACULTAD DE INGENIERÍA  
PROGRAMA DE INGENIERÍA ELECTRÓNICA  
NEIVA – HUILA  
2014**

**DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE CONTROL Y MONITOREO  
REMOTO PARA LA ILUMINACIÓN DE UNA VIVIENDA A TRAVÉS DE UN  
TELÉFONO MÓVIL CON SISTEMA OPERATIVO ANDROID Y BOARD ARM  
MINI2440**

**FERNANDO AUGUSTO ESPAÑA CANO  
ELIANA FERNANDA TACÁN CUELLAR**

**Proyecto de grado presentado como requisito para optar  
el título de Ingeniero Electrónico**

**Director:  
JESÚS DAVID QUINTERO POLANCO  
Ingeniero Electrónico**

**UNIVERSIDAD SURCOLOMBIANA  
FACULTAD DE INGENIERÍA  
PROGRAMA DE INGENIERÍA ELECTRÓNICA  
NEIVA – HUILA  
2014**

**Nota de aceptación:**

---

---

---

---

---

---

---

**Firma director del proyecto**

---

**Firma jurado**

---

**Firma jurado**

**Neiva, 6 de Agosto de 2014**

*Dedicado a Dios, nuestro señor, y a nuestra Santa Madre María por haberme brindado vida, salud y alegría para seguir adelante. A la memoria de mamá Cecilia y papá Arturo, abuelos maternos, quienes me brindaron alegrías y un sinfín de enseñanzas. A mis padres y hermanas, por su apoyo incondicional, ejemplo y orientación en todo momento. A mi familia y amigos, por ofrecerme, de algún modo, su apoyo y llenarme de grandes momentos y días inolvidables. A mi novia Vanessa y su familia, por todo el soporte que en tan poco tiempo me han ofrecido.*

***-Fernando Augusto España Cano***

*Dedicado a Dios quién supo guiarme por el buen camino, darme fuerzas para seguir adelante y no desfallecer en los problemas que se me presentaban. A mis padres Carmen Cuellar y Jorge Tacan por su apoyo, consejos, comprensión, amor y ayuda en los momentos difíciles. A mis hermanos porque son un ejemplo de vida al luchar contra enfermedades y adversidades y han logrado salir adelante. A mis sobrinas porque son la luz de mi vida. A toda mi familia por brindarme su apoyo. A mis amigos por su apoyo incondicional y por llenar mi vida de momentos inolvidables. A Alfonso quien ha sido un apoyo incondicional. A todas y cada una de las personas que de una u otra forma ayudaron para lograr esta meta.*

***-Eliana Fernanda Tacán Cuellar***

## **AGRADECIMIENTOS**

A Dios padre todo poderoso por ayudarnos a ascender un escalón más en nuestras vidas al permitirnos graduar como profesionales, sin tu ayuda nada de esto sería posible.

A nuestras familias, por su apoyo incondicional durante todo nuestro proceso de formación, tanto profesional como personal.

A Luz Marina Cano Medina, quien con sus tantas noches de traspas hizo que le tomara cariño a las matemáticas y de ahí a mi carrera. Gracias a ella, y a todos sus conocimientos impartidos, es que hoy puedo decir que logré alcanzar una de mis metas, el ser profesional.

A nuestro director de proyecto, Jesús David Quintero Polanco, ingeniero electrónico, por sus orientaciones, asesorías, críticas constructivas y acompañamiento constante, por haber confiado en nosotros y haberse arriesgado a tomar dicha dirección.

A los profesores del programa de ingeniería electrónica de la Universidad Surcolombiana, quienes nos brindaron sus conocimientos y apoyo en nuestro proceso de formación profesional.

A nuestros amigos y compañeros, con quienes compartimos conocimientos y sobre todo experiencias inolvidables, en especial a Khristian Robayo, Fabián Orrego y Felipe Leyva.

## CONTENIDO

	pág.
INTRODUCCIÓN .....	19
1. PLANTEAMIENTO DEL PROBLEMA .....	21
1.1. DESCRIPCIÓN.....	21
2. JUSTIFICACIÓN .....	22
3. OBJETIVOS .....	24
3.1. OBJETIVO GENERAL.....	24
3.2. OBJETIVOS ESPECÍFICOS .....	24
4. MARCO TEÓRICO.....	25
4.1. EL CONCEPTO DE DOMÓTICA.....	25
4.1.1. Características de los sistemas domóticos .....	26
4.2. SISTEMAS EMBEBIDOS .....	26
4.2.1. Características de los sistemas embebidos .....	27
4.3. DISPOSITIVOS MÓVILES .....	28
4.3.1. Características de los dispositivos móviles .....	29
4.4. SISTEMAS OPERATIVOS .....	29
4.4.1. Sistemas operativos para dispositivos móviles .....	30
4.4.1.1. Windows Phone.....	30
4.4.1.2. BlackBerry OS .....	31
4.4.1.3. iOS.....	31
4.4.1.4. Android .....	32
4.4.2. Sistemas operativos embebidos .....	34
4.5. TARJETA DE DESARROLLO MINI2440.....	34
4.5.1. Hardware.....	35
4.5.2. Micro-controlador S3C2440A .....	40
4.5.3. Software.....	42
5. DESCRIPCION Y DESARROLLO DEL SISTEMA .....	43
5.1. COMUNICACIÓN BOARD ARM MINI2440 – PC .....	44
5.2. CIRCUITO DE ACOPLE.....	49

5.2.1.	Etapa de potencia para los bombillos.....	49
5.3.	DISEÑO Y DESARROLLO DE LA APLICACIÓN .....	50
5.3.1.	Herramientas de desarrollo empleadas: Software .....	51
5.3.1.1.	Sistema operativo software libre: OpenSUSE .....	51
5.3.1.2.	Compilador cruzado ARM LINUX GCC 4.3.2 .....	51
5.3.1.3.	Qtopia .....	54
5.3.1.4.	Base de datos SQLite3.....	54
5.3.1.5.	Eclipse IDE .....	56
5.3.2.	Proceso de desarrollo .....	56
5.3.3.	Aplicación web .....	60
6.	IMPLEMENTACIÓN .....	62
7.	RESULTADOS .....	65
8.	PROBLEMAS Y LIMITACIONES.....	72
9.	RECOMENDACIONES Y SUGERENCIAS .....	73
10.	CONCLUSIONES.....	74
	BIBLIOGRAFÍA.....	76

## LISTA DE TABLAS

	pág.
<b>Tabla 1.</b> Comparación sistemas operativos móviles.....	33
<b>Tabla 2.</b> Distribución de leds de usuario.....	93
<b>Tabla 3.</b> Distribución botones de usuario .....	93
<b>Tabla 4.</b> Distribución pines puerto GPIO .....	99



## LISTA DE FIGURAS

	pág.
<b>Figura 1.</b> Sistemas Operativos más conocidos para dispositivos móviles .....	30
<b>Figura 2.</b> Versiones sistema operativo Android.....	32
<b>Figura 3.</b> Kit tarjeta de desarrollo mini2440.....	35
<b>Figura 4.</b> Dimensiones board mini2440.....	37
<b>Figura 5.</b> Interfaces board mini2440.....	40
<b>Figura 6.</b> Arquitectura núcleo ARM920T .....	41
<b>Figura 7.</b> Esquema general del proyecto .....	44
<b>Figura 8.</b> Interfaz FileZilla para transferencia de datos entre la board – PC .....	45
<b>Figura 9.</b> Conexión con la board a través del protocolo de red TELNET desde la consola de Linux.....	46
<b>Figura 10.</b> Conexión con la board a través del protocolo de red OpenSSH desde la consola de Linux.....	47
<b>Figura 11.</b> Interfaz PuTTY para conexión con el servidor de la board a través del protocolo SSH desde Windows.....	48
<b>Figura 12.</b> Consola abierta con PuTTY y conectada al servidor de la board .....	49
<b>Figura 13.</b> Circuito esquemático de control de potencia para los bombillos.....	50
<b>Figura 14.</b> Sistema operativo OpenSUSE.....	51
<b>Figura 15.</b> Verificación de la instalación del compilador cruzado GCC .....	53
<b>Figura 16.</b> Entorno gráfico Qtopia 2.2.0 .....	54
<b>Figura 17.</b> Diagrama de arquitectura de desarrollo del sistema .....	57
<b>Figura 18.</b> Caso de usos del programa gpiosfinal.cgi .....	58
<b>Figura 19.</b> Diagrama de secuencias del programa start.cgi .....	59
<b>Figura 20.</b> Logo aplicación en dispositivo Android .....	60
<b>Figura 21.</b> Archivos carpeta “www” del servidor web .....	62
<b>Figura 22.</b> Nano router TP-Link TL-WR702N .....	64
<b>Figura 23.</b> Aplicación web Control Bombillos y Menú desplegable .....	65

<b>Figura 24.</b> Aplicación web Historial .....	66
<b>Figura 25.</b> Aplicación web Autores y mensaje Acerca de .....	67
<b>Figura 26.</b> Tabla de registros de encendido/apagado de los bombillos .....	68
<b>Figura 27.</b> Mensajes cuando se va la energía y al borrar bien el registro .....	69
<b>Figura 28.</b> Caja metálica para la board ARM mini2440 y la etapa de potencia .....	70
<b>Figura 29.</b> Circuito acoplado probando la etapa de potencia .....	70
<b>Figura 30.</b> Acople de todo el sistema .....	71
<b>Figura 31.</b> Elementos utilizados para la instalación de Qtopia .....	79
<b>Figura 32.</b> Identificación Supervivi .....	80
<b>Figura 33.</b> Interfaz DNW .....	82
<b>Figura 34.</b> Forma correcta de conexión .....	82
<b>Figura 35.</b> Conexión USB:OK .....	83
<b>Figura 36.</b> Forma de subir el Supervivi .....	84
<b>Figura 37.</b> Elección del kernel para el LCD .....	85
<b>Figura 38.</b> Elección e instalación SO Qtopia .....	86
<b>Figura 39.</b> Esquema de los chips SDRAM (K4S561632N-LC60).....	89
<b>Figura 40.</b> Modos de arranque y flash de almacenamiento .....	90
<b>Figura 41.</b> Reguladores de voltaje board mini2440.....	91
<b>Figura 42.</b> Conector para fuente de alimentación externa (CON8) .....	92
<b>Figura 43.</b> Circuito reset.....	92
<b>Figura 44.</b> Botones de usuario .....	94
<b>Figura 45.</b> Prueba del ADC con la resistencia variable W1 .....	94
<b>Figura 46.</b> Prueba de timbre PWM a través del puerto GPIO .....	95
<b>Figura 47.</b> Puertos seriales y RS-232 .....	95
<b>Figura 48.</b> Interfaces USB Host y esclavo.....	96
<b>Figura 49.</b> Interfaz LCD .....	97
<b>Figura 50.</b> Interfaz JTAG .....	98
<b>Figura 51.</b> Interfaz GPIO .....	98
<b>Figura 52.</b> Interfaz bus del sistema .....	100

## LISTA DE ANEXOS

	pág.
<b>Anexo A.</b> Manual de instalación Sistema Operativo Linux en Board ARM mini2440 .....	79
<b>Anexo B.</b> Manual de instalación del protocolo de red OpenSSH en la Board ARM mini2440 .....	87
<b>Anexo C.</b> Descripción partes Board ARM mini2440 .....	89

## GLOSARIO

**AJAX (Asynchronous JavaScript And XML):** JavaScript asíncrono y XML. Técnica de desarrollo que envía una solicitud de usuario al servidor, en segundo plano, invisible para el usuario, permitiendo realizar cambios a la página web sin necesidad de recargarla (actualizarla).

**ANDROID:** Sistema operativo de software libre y código abierto. Inicialmente, este sistema se usó para teléfonos móviles, pero hoy en día es usado en tabletas y computadoras (por medio de una máquina virtual). Tiene como logo oficial un androide de color verde y actualmente se encuentra disponible en el mercado la versión 4.4.2 conocido como Kitkat.

**API (Application Programming Interface):** Interfaz de programación de aplicaciones. Es una llave de acceso a funciones, que usan los programadores o desarrolladores (developers) de aplicaciones, que les permite hacer uso de un servicio web provisto por un tercero de manera segura. Esto evita que se conozca o acceda al código fuente de la aplicación original. Estas API´s son de gran ventaja, ya que proporcionan al programador un conjunto de funciones, permitiendo que éste no tenga que programar todo desde el principio.

**APK (Application Package File):** Archivo de paquete de aplicaciones. Es un tipo de formato con extensión .apk que contiene componentes empaquetados los cuales son usados tanto para teléfonos móviles como para tabletas que manejen sistema operativo Android. Este tipo de archivos solo pueden ser ejecutados desde la plataforma Android.

**APLICACIÓN WEB:** Programa que se ejecuta en un entorno web, es decir, a través de un navegador (Chrome, Explorer, Firefox, etc.). Este tipo de aplicaciones se puede ejecutar en diferentes sistemas operativos y puede ser vista desde diferentes dispositivos móviles.

**APN (Access Point Name):** Nombre de un punto de acceso. Nombre que se debe configurar en los teléfonos móviles para lograr tener acceso a internet usando la red de la operadora de telefonía móvil local. Estos ajustes también sirven para enviar y recibir mensajes multimedia.

**ARM (Advanced RISC Machines):** Máquinas RISC avanzadas. Tipo de procesadores que usa arquitectura RISC (Ordenador con Conjunto de Instrucciones Reducidas). Trabaja a 32 bits y son ideales para aplicaciones de baja potencia como PDA, tabletas, teléfonos móviles, videoconsolas, etc.

**BOA WEB SERVER:** Servidor web HTTP. Es ideal para aplicaciones de sistemas embebidos. Soporta programas CGI, genera directorios y descomprime el archivo automáticamente.

**BOARD:** Es conocido como 'placa base' o 'placa madre'. Es el elemento principal de todo ordenador y donde van conectadas todas y cada una de las partes y dispositivos que lo componen. Se pueden encontrar tanto en las CPU como de forma individual y menor nivel (no tan complejas), proporcionando una manera más sencilla a los desarrolladores en el momento de hacer diferentes tipos de programas.

**C++:** Lenguaje de programación que está orientado a objetos. Surge como mejora del lenguaje C, ya que este tipo de lenguaje solo permitía la programación estructurada. Una diferencia entre estos dos lenguajes es la extensión con la que se guardan sus archivos: ".c" para lenguaje C y ".cpp" para lenguaje C++.

**CGI (Common Gateway Interface):** Interfaz de entrada común. Método para la transmisión de información hacia un compilador instalado en el servidor. Cuando se hace la compilación en Linux del archivo creado en lenguaje C++, este crea un archivo de extensión ".cgi" el cual puede ser leído por la máquina. Si se abre un archivo con este tipo de extensión, se puede notar que el código está en hexadecimal. Este método es comúnmente usado para realizar bases de datos, formularios, etc.

**CÓDIGO ABIERTO (Open Source):** Fuente abierta. Es el término con el que se conoce al software que puede ser modificado y distribuido libremente, esto quiere decir que cualquier persona puede tener acceso al código fuente y modificarlo a su antojo.

**COMPILADOR:** Programa o aplicación que traduce un lenguaje de programación a otro de lenguaje de programación, generando un programa el cual puede ser entendido por la máquina y ejecutado.

**CONTROL DE POTENCIA (etapa de potencia):** Se usa para hacer el acople entre la tarjeta de desarrollo y la red eléctrica de la vivienda, esto con el fin de ajustar la señal digital a la red eléctrica y así lograr encender el bombillo.

**DESARROLLADOR (Developer):** Persona que crea, modifica o desarrolla programas y/o aplicaciones en un sistema operativo específico. Generalmente estas personas toman el código fuente, que ha sido liberado con anterioridad, y lo modifican a su antojo, dando así una mayor optimización a los dispositivos eléctricos con los que trabajan.

**DOMÓTICA:** Integración de diferentes sistemas mediante los cuales se puede automatizar una vivienda. El fin de la domótica es brindar al usuario un mejor

confort, seguridad y ahorro económico, aumentando la calidad de vida de las personas.

**FIRMWARE:** Programación en firme. Programa que controla todos los dispositivos electrónicos. Es el encargado de que todos los dispositivos funcionen correctamente. Dicho programa lleva un bloque de instrucciones de máquina los cuales tienen propósitos específicos. En otras palabras, es el software que maneja al hardware.

**GOOGLE PLAY:** Tienda donde se pueden encontrar y descargar diferentes tipos de aplicaciones para teléfonos con tecnología Android desarrolladas por distintos programadores a nivel mundial. Fue creada por Google y se encuentra en internet.

**GPIO (General Purpose Input/Output):** Entrada/Salida de Propósito General. Pin que no tiene ningún propósito en específico. Éste puede ser programado por el usuario para que actúe como entrada o salida de un sistema. Existen grupos de pines y se conocen como puerto GPIO. En este puerto, los pines pueden ser programados para actuar de diferente manera, dependiendo de la necesidad.

**HTML5 (HyperText Markup Language, Versión 5):** Lenguaje de marcado de hipertexto. Lenguaje de programación usado comúnmente para el desarrollo de páginas web. Este lenguaje puede ser interpretado por los navegadores de internet más recientes (Mozilla Firefox, Google Chrome, Safari, etc.) ya que los más antiguos no pueden descifrar las nuevas etiquetas.

**IDE (Integrated Development Environment):** Entorno de desarrollo integrado. Programa que está compuesto por un conjunto de herramientas útiles para un programador. Puede usarse ya sea para un lenguaje de programación específico o varios.

**INTERFAZ GRÁFICA:** Medio por el cual el usuario puede interactuar con la máquina, de una manera visual y agradable, su lenguaje a través de imágenes y objetos gráficos, los cuales representan la información y acciones disponibles. También puede ser conocida como GUI (Graphical User Interface, Interfaz de Usuario Gráfica).

**JAVA:** Lenguaje de programación orientado a objetos (usa objetos, entidades que tienen un estado, comportamiento e identidad, en sus interacciones). Es comúnmente usado tanto en Internet como en la informática en general.

**JAVASCRIPT:** Está enfocado en mejorar la interfaz de usuario, dándole ciertas propiedades dinámicas a una página web. Los scripts se ejecutan desde la parte del cliente, posibilitando que el código se altere sin necesidad de ser compilado para su ejecución.

**JQUERY:** Librería de JavaScript que facilita la programación de este lenguaje. Permite acceder a los recursos de HTML de una manera más cómoda. Puede interactuar con AJAX.

**JSON (JavaScript Object Notation):** Formato ligero de intercambio de datos. Cuando el usuario realiza una solicitud, el servidor devuelve una respuesta en formato JSON, en otras palabras, es una forma fácil y ágil de enviar y recibir datos entre diferentes lenguajes de programación. Puede interactuar con AJAX.

**KERNEL:** También llamado “núcleo del sistema”. Es el corazón de un sistema operativo. El Kernel es el conjunto de drivers necesarios para usar el equipo u ordenador. Es el encargado de que el software y hardware trabajen de la mano.

**LENGUAJE DE PROGRAMACIÓN:** Lenguaje que se usa para describir un conjunto de acciones, las cuales darán instrucciones a un equipo u ordenador determinado. Dichas acciones consisten en reglas sintácticas y semánticas las cuales definen su estructura.

**LINUX:** Sistema operativo de software libre y gratuito, usado para trabajar en computadores de mesa, portátiles y, hoy en día se puede encontrar, en tabletas y algunos teléfonos inteligentes (Smartphone). Al ser software libre, las personas pueden ver su código fuente, modificarlo, copiarlo y distribuirlo a su antojo.

**MODEM:** Derivado de la palabra MOdulador – DEModulador. Dispositivo usado para enviar señales moduladoras a través de otra señal llamada portadora. Permite la comunicación entre computadores a través de la línea telefónica.

**PROGRAMACIÓN ORIENTADA A OBJETOS (POO):** Tipo o clase de programación que se asemeja bastante a la forma de pensar del ser humano y no al de la máquina. A diferencia de la programación estructurada, que utiliza como elemento básico la función, este tipo de programación utiliza un objeto como su elemento básico. Se puede decir que el objeto, en este tipo de programación, son entidades que tienen un estado, comportamiento e identidad.

**RISC (Reduced Instruction Set Computer):** Computador con conjunto de instrucciones reducidas. Tipo de arquitectura computacional. Promueve conjuntos pequeños y simples de instrucciones las cuales pueden tomar poco tiempo para ejecutarse.

**ROUTER:** Enrutador. Sirve de interconexión entre computadoras que están conectadas a una red. Se encarga de establecer la ruta por la cual irá cada paquete de datos dentro de una red informática.

**SDK (Software Development Kit):** Kit de desarrollo de software. Paquete de software que contiene diferentes herramientas y programas los cuales ayudan al

programador en la creación de aplicaciones y/o programas para un sistema o plataforma concretos. Los SDK más conocidos son el de Iphone y el de Android.

**SERVIDOR:** Sistema que ofrece un servicio o recurso necesario para desempeñar una función o tarea utilizando una infraestructura de red para poder hacerlo.

**SMARTPHONE:** Teléfono inteligente. Teléfono móvil con capacidad de instalar programas que permiten aumentar el procesamiento de datos y la conectividad, pareciéndose a un computador.

**SOFTWARE LIBRE:** Se refiere a los tipos de software a los que los usuarios pueden ejecutar, copiar, estudiar, modificar y distribuir libremente. Dicho software debe ser comercial y estar a disposición del público en general.

**SQL (Structured Query Language):** Lenguaje de consulta estructurado. Tipo de lenguaje que está vinculado con la gestión de base de datos relacionales las cuales, permiten gran variedad de operaciones entre sí.

**SQLITE:** Pequeña librería programada en lenguaje C, la cual implementa un completo motor de base de datos relacional que no requiere configuración. Permite utilizar lenguaje estándar SQL, haciéndolo rápido y versátil. Dichas bases de datos son manejadas como un fichero plano, las cuales, el usuario puede manipular fácilmente.

**SUBLIME TEXT 2:** Editor de texto y código fuente capaz de soportar casi que cualquier tipo de lenguaje de programación para su creación o modificación. Está disponible para Windows, Apple y Linux.

**TABLET:** Tableta. Tiene gran parecido a los teléfonos táctiles pero de mayor tamaño. Posee pantalla táctil y no necesita de teclado físico ni mouse (ratón). Por ahora manejan sistema operativo Android (Google) e IOS (Apple), sin embargo, hay empresas que ya están sacando tabletas con su sistema operativo, es el caso de Firefox OS.



## RESUMEN

El presente proyecto se realizó con el fin de lograr un control y monitoreo remoto del sistema de luces de una vivienda, construida a escala (1:10), que funciona con la red eléctrica (110V), a través de un teléfono móvil que cuente con sistema operativo Android. Para llevar a cabo esto, se utilizó el puerto GPIO de un sistema embebido (Board ARM mini2440), el cual, recibe las órdenes que le sean enviadas desde el teléfono móvil (apagar/encender el bombillo), a través de una conexión WIFI, las procesa y envía a su destino. Se adecuó una etapa de potencia para cada bombillo, la cual ajusta la señal digital a la red eléctrica. Se creó una aplicación web móvil utilizando lenguaje de programación HTML5, dando la posibilidad de que dicha aplicación sea vista desde diferentes dispositivos móviles con sistema operativo Android (PC, tabletas, Smartphone, etc.) gracias a la clase WebView, adaptando la página web a un dispositivo. Con esto se consiguió tener comunicación entre la tarjeta y el teléfono móvil. Dicha aplicación posee una interfaz gráfica sencilla, en la que se muestra el nombre del lugar donde se encuentra cada bombillo, además, cuenta con una base de datos hecha en SQLite3, la cual, enseña el tiempo que éste lleva encendido y da la opción de introducir la potencia de cada bombillo, dando la posibilidad de conocer el valor, en pesos, que el usuario debe pagar por la energía consumida de cada bombillo en KW.h; esto se hizo ya que las especificaciones de cada bombillo son diferentes. Como resultado se obtuvo un sistema capaz de controlar las luces de una vivienda que puede variar de acuerdo a las especificaciones, para este caso, la aplicación web móvil sólo funciona con el sistema de luces que se creó.

**Palabras claves:** ARM mini2440, control de luces, monitoreo remoto, teléfono móvil, Android, base de datos, SQLite3, HTML5, aplicación web.

## ABSTRACT

The present project was performed in order to achieve a control and remote monitoring of the lights system of a housing, built to scale (1:10), which works with the mains (110V), through a mobile phone that has Android operating system. To carry out this, the GPIO port of an embedded system (Board ARM mini2440) was used, which, receives the orders that are sent from the mobile phone (turn off/turn on the light bulb), through a WIFI connection, it processes them and sends to its destination. A power stage was adapted for every light bulb, which it adjust the digital signal to the mains. A mobile web application was built using HTML5 programming language, giving the possibility that such implementation is be seen from different mobile devices with android operating system (PC, tablets, Smartphone, etc.) thanks to the WebView class, adapting the website to a device. With this was achieved to have communication between the board and the mobile phone. Such application has a simple graphical interface, in which shown the name of the place where it is each light bulb, also, it has a database made in SQLite3, which, teaches the time it takes on and gives the option of introduce the potency of each bulb, giving the possibility to know the value, in pesos, which the user must pay for the energy consumed by each bulb in KW.h; this was done as it specs of each light bulb are different. The result is a system capable of controlling the lights of a housing that can vary according to specifications, for this case, the web application works with the lighting system that was created.

**Keywords:** ARM mini2440, lighting control, remote monitoring, mobile phone, Android, database, SQLite3, HTML5, web application.

## INTRODUCCIÓN

La ciencia y la tecnología, dos palabras que aunque pase el tiempo, siempre la una tendrá que ver con la otra. En un mundo como el de hoy en día, en donde la ciencia y la tecnología avanzan a pasos agigantados, el ser humano debe proveerse y estar atento a estos cambios, los cuales, en su gran mayoría, se hacen para su bienestar y beneficio.

Hace poco más de 50 años, el ser humano se encontraba en un proceso de creación tecnológica sin precedentes. Transistores y microcontroladores fueron la base de la electrónica moderna, haciéndolos cada vez más pequeños pero con mayor capacidad y velocidad. Con la creación de los computadores, se dio un gran salto tecnológico hacia lo moderno. Tarjetas perforadas ahora podían ser interpretadas dentro de una interfaz gráfica la cual era visible por el ser humano. Se pasó de las máquinas de escribir a tener un entorno visual en donde el hombre y la máquina podían interactuar de una manera rápida y sencilla, permitiendo procesar y almacenar información sin temor a que ésta se perdiera. Se siguió avanzando, y todo era cada vez más pequeño, pero a la vez mejor que su antecesor. De los grandes computadores de mesa se pasaron a los computadores portátiles, pequeños aparatos electrónicos, capaces de cumplir con las funciones de un computador de mesa, pero con la diferencia de que se podía llevar a donde se quisiera.

Tener una rápida, sencilla y estable comunicación con un ser querido que estaba lejos era toda una odisea. Las cartas se podían perder en el camino y el uso de internet era muy reducido, casi que solo las grandes multinacionales podían acceder a éste. Gracias a ésta y muchas otras necesidades, nacen los teléfonos móviles, pequeños dispositivos capaces de comunicarse vocal e inalámbricamente con cualquier persona en el mundo sin importar donde ésta se encuentre.

Con la rápida evolución de la ciencia y la tecnología, es apenas obvio que hoy en día se encuentren gran variedad de dispositivos electrónicos. Computadores mini-portátiles, computadores de mesa táctil, Smartphone, tablets, etc. Todo esto conllevando a que haya una interacción más estrecha entre lo tecnológico y el ser humano.

La domótica o casas inteligentes, es una prueba de lo mucho que puede aportar la tecnología al bienestar del hombre. Desde el encendido y apagado de las luces, pasando por la apertura y cierre de puertas, hasta el control de la cocina, nevera, etc., la domótica se ha convertido hoy en día en una herramienta muy útil a la hora

de interactuar con los objetos que componen una casa. Todo esto puede ser logrado, ya sea desde un computador de mesa o computador portátil, incluso desde un Smartphone, gracias a la fácil interacción que tiene el ser humano con la tecnología.

En un mundo donde la innovación está a la vuelta de la esquina, no es raro que las personas vivan atrasadas de los adelantos tecnológicos, sin embargo, cada día se hace más sencillo lograr acceder a este tipo de información. Los teléfonos inteligentes logran tener aplicaciones web de todo tipo, desde noticias hasta juegos, pasando por las redes sociales y llegando hasta el control total de una residencia. A través de este tipo de dispositivo, es posible lograr el control de todas las cosas electrónicas que se encuentren dentro, dando a los usuarios la tranquilidad, confort y bienestar que se merecen.

Este proyecto está basado en un sistema capaz de controlar y monitorear las luces de una vivienda por medio de un teléfono móvil a través de un sistema embebido, el cual es el encargado, gracias a su programación, de interconectar las luces del prototipo de vivienda con el teléfono. Dicho control y monitoreo varía de acuerdo a las especificaciones de la vivienda, por consiguiente, esta aplicación es compatible únicamente con la maqueta en la que se prueba el sistema.

Ya que hay diferentes formas de lograr la comunicación entre el sistema embebido y el dispositivo móvil, se realizó un estudio, de tal forma que la persona que use la aplicación pueda encender o apagar las luces desde cualquier lugar, siempre y cuando el dispositivo móvil esté en red con el servidor de la board. A través de una aplicación web instalada en el teléfono móvil, la persona podrá controlar el encendido y apagado de las luces de la vivienda, y por medio de la base de datos, podrá monitorear el consumo de cada una, logrando así un ahorro económico.

## **1. PLANTEAMIENTO DEL PROBLEMA**

### **1.1. DESCRIPCIÓN**

Colombia es un país subdesarrollado que no cuenta con los últimos adelantos tecnológicos, sin embargo, desde hace algunos años ha venido creciendo en este ámbito. Ahora es posible encontrar casas y edificios automatizados, de tal forma que el usuario puede interactuar de manera rápida, confortable y eficiente con los diferentes dispositivos eléctricos y electrónicos ahí presentes. Estos avances han ido desencadenando, con el pasar de los días, un considerable aumento en la demanda de instalaciones eléctricas modernas, llevando así los términos de seguridad, confort y ahorro económico a otra dimensión. Debido a esta demanda, son muchas las empresas que ofrecen este tipo de servicio, algunas, brindando sus productos a precios muy elevados, impidiendo que gran parte de la población pueda acceder a este tipo de beneficio.

Una de las funciones del hogar con mayor demanda es el encendido y apagado de las luces a través de interruptores. Dado el constante movimiento laborioso de las personas, es fácil que olviden apagar algún bombillo de la casa. En base a lo anterior, existe la necesidad de crear un sistema de bajo costo que controle y monitoree de manera remota el sistema de luces de una casa, el cual pueda ser visualizado, por medio de una aplicación creada en HTML5, a través de un teléfono móvil con tecnología Android y desde cualquier dispositivo móvil con un navegador web.

## 2. JUSTIFICACIÓN

Colombia, específicamente el Ministerio de las TIC<sup>1</sup>, está desarrollando proyectos en diferentes partes del país, con el fin de que todas las personas tengan acceso a éstas tecnologías y vean el alcance que dichas herramientas pueden tener, reduciendo la brecha digital sobre la cual se tiene que construir una sociedad de la información y una economía del conocimiento. Los computadores portátiles, las tabletas y los teléfonos móviles inteligentes, son claros ejemplos de lo mucho que ha avanzado la tecnología y gracias a ello se pueden hacer interacciones entre éstos. Prueba de dicha integración puede ser vista mediante la domótica.

Hoy en día, hablar de domótica sin relacionarlo con las TIC es casi imposible, ya que éstas se complementan, permitiendo la innovación y el constante desarrollo de nuevos equipos y programas para su implementación en el hogar.

*“Las tecnologías de la información y la comunicación no son ninguna panacea ni fórmula mágica, pero pueden mejorar la vida de todos los habitantes del planeta. Se dispone de herramientas para llegar a los Objetivos de Desarrollo del Milenio, de instrumentos que harán avanzar la causa de la libertad y la democracia y de los medios necesarios para propagar los conocimientos y facilitar la comprensión mutua”.*<sup>2</sup>

Para que un sistema pueda ser considerado inteligente ha de incorporar elementos o sistemas basados en las Nuevas Tecnologías de la Información<sup>3</sup> (NTIC), esto con el fin de brindar a las personas comodidad y al mismo tiempo preservar el medio ambiente. Generalmente, un sistema domótico dispondrá de una red de comunicación y diálogo que permita la interconexión de una serie de equipos a fin de obtener información sobre el entorno doméstico y, basándose en dicha red, hará de la vivienda un entorno más funcional, inmediato y a la medida de cada usuario.

Este proyecto tiene como fin crear un sistema de bajo costo que permita, a los usuarios en general, controlar las luces de una vivienda a través de un sistema embebido (Board ARM mini2440), el cual debe ir integrado al prototipo de casa.

---

<sup>1</sup> Tecnologías de la Información y las Comunicaciones. Conjunto de herramientas computacionales e informáticas capaces de optimizar el manejo de la información y el desarrollo de la comunicación. Estas herramientas se encargan de estudiar, desarrollar, implementar, almacenar y distribuir la información mediante la utilización de hardware y software como medio de sistema informático.

<sup>2</sup> ANNAN, Kofi. *Discurso inaugural de la primera fase de la WSIS*. Ginebra, 2003.

<sup>3</sup> “Conjunto de aparatos, redes y servicios que se integran en un sistema de información interconectado y complementario. La innovación tecnológica consiste en que se pierden la frontera entre n medio de información y otro.” GÓMEZ MONT, Carmen.: *El desafío de los nuevos medios de comunicación en México*. AMIC-Diana, México. 1992. 180 p.

Este sistema podrá ser visualizado, por medio de una aplicación creada en HTML5, desde un teléfono móvil con tecnología Android y desde cualquier navegador web ingresando la dirección IP del servidor. Para que el usuario controle el sistema de luces desde cualquier lugar, el servidor del sistema embebido debe estar en red con la aplicación web. Además del control, podrá monitorear el tiempo de encendido y apagado de los bombillos y registrarlo en una base de datos, hecha en SQLite3, igualmente, visualizará el tiempo que éste lleva encendido y tendrá la opción para introducir la potencia de cada bombillo y, así saber el precio que debe pagar por el consumo de energía de estos. El control de la iluminación de la vivienda permitirá tener un ahorro económico, ya que, dado el caso de que el usuario salga y quede algún bombillo encendido lo pueda apagar, evitando así el uso excesivo de energía, además, monitorear constantemente el encendido de las luces de su vivienda.

Dado el riesgo que se puede presentar al realizar cambios en la conexión del sistema eléctrico de una vivienda, además de los costos que implica tener que romper paredes para introducir el cableado, se implementará una maqueta a escala (1:10) con un circuito base de cinco (5) bombillos, los cuales, estarán conectados a la red eléctrica y al sistema embebido para encender y apagar la iluminación; este circuito contará con una etapa de potencia para cada bombillo, la cual ajusta la señal digital a la red eléctrica. Debido a que se está automatizando el prototipo de casa, los interruptores de la maqueta son reemplazados por pulsadores, simulando un interruptor digital.

### **3. OBJETIVOS**

#### **3.1. OBJETIVO GENERAL**

Diseñar e implementar un sistema que controle y monitoree remotamente las luces de una vivienda, construida a escala (1:10), a través de la Board ARM mini2440, desde un teléfono móvil con sistema operativo Android o cualquier dispositivo móvil con navegador web, y, cuyo consumo y tiempo de encendido se almacene en una base de datos.

#### **3.2. OBJETIVOS ESPECÍFICOS**

- Realizar un estudio previo al diseño, con el fin de determinar la mejor forma para el desarrollo y culminación del sistema de control de los bombillos.
- Establecer la comunicación entre la Board ARM mini2440 y los diferentes dispositivos utilizados para el desarrollo del sistema.
- Determinar una forma de comunicación entre el computador y la board ARM mini2440 para tener acceso remoto a la tarjeta.
- Crear un programa que controle los bombillos a través de la Board ARM mini2440.
- Acoplar una etapa de potencia para ajustar la señal digital de la Board ARM mini2440 a la red eléctrica.
- Implementar una base de datos para almacenar el tiempo de encendido y apagado de los bombillos y mostrar en tiempo real el tiempo q éstos llevan encendidos.
- Diseñar una aplicación web que permita visualizar, desde diferentes dispositivos móviles, el programa de control y monitoreo creado.
- Diseñar e implementar una maqueta a escala (1:10) de una vivienda con su respectivo sistema de luces para así mostrar el funcionamiento de la aplicación.



## 4. MARCO TEÓRICO

### 4.1. EL CONCEPTO DE DOMÓTICA

Gracias a los numerosos avances, desarrollos y soluciones que se han venido creando con el pasar de los años, se ha llegado a una mayor integración entre todos los sistemas y equipos domésticos. Este tipo de integración tecnológica en el hogar es lo que se ha venido denominando como Domótica. Una definición estricta de vivienda domótica o inteligente, sería:

*“La Domótica puede definirse como la adopción, integración y aplicación de las nuevas tecnologías informáticas y comunicativas al hogar. Incluye principalmente el uso de electricidad, dispositivos electrónicos, sistemas informáticos y diferentes dispositivos de telecomunicaciones, incorporando la telefonía móvil e Internet. Algunas de sus principales características son: interacción, interrelación, facilidad de uso, tele-operación o manejo a distancia, fiabilidad, y capacidad de programación y actualización.”<sup>4</sup>*

Cabe resaltar que para que una vivienda sea considerada domótica, además de la inclusión de las TIC, debe disponer de sistemas integrados y que sean interactivos, es decir, que trabajen juntos de la mano.

Dentro de los equipos del sistema domótico se pueden encontrar los siguientes componentes:

- **Controlador:** Gestionan el sistema según la programación.
- **Actuador:** Ejecuta y/o recibe la orden del controlador y la ejecuta.
- **Sensor:** Monitoriza el entorno y capta información.
- **Bus:** Medio de transmisión por el cual se envía la información.
- **Interface:** Dispositivos o medios en donde se muestra la información del sistema.

---

<sup>4</sup> CHAPARRO. Jeffer. Domótica: La mutación de la vivienda. *Scripta Nova: Revista electrónica de geografía y ciencias sociales*. [En línea. Acceso libre]. Barcelona: Universidad de Barcelona, Geocrítica, 2003. <[http://www.ub.edu/geocrit/sn/sn-146\(136\).htm#\\_edn11](http://www.ub.edu/geocrit/sn/sn-146(136).htm#_edn11)> ISSN: 1138-9788. (A).

#### 4.1.1. Características de los sistemas domóticos

Los sistemas domóticos presentan una característica fundamental: debe haber comunicación entre ellos, aunque solo actúen los dispositivos a los cuales se dirigen las órdenes. Debido a esto, y para tener y garantizar un funcionamiento adecuado de los sistemas utilizados en la domótica, hay que tener en cuenta otros aspectos importantes como la simplicidad y facilidad de su utilización, portabilidad e interrelación, fiabilidad, flexibilidad y, por supuesto, el costo moderado.

- **Simplicidad y facilidad:** El sistema de control debe contar con una interfaz amigable, pensando en el usuario final. Ésta debe ser fácil de operar e intuitiva, permitiendo al usuario un aire de confort.
- **Portabilidad e interrelación:** La aplicación debe trabajar en diferentes dispositivos móviles, sin importar el sistema operativo que gobierne, dando así la opción de relacionarse y comunicarse con otras áreas de gestión de la edificación, obteniendo versatilidad y variedad en la ejecución de procesos requeridos.
- **Flexibilidad y costo moderado:** Se debe pensar en futuras modificaciones o ampliaciones que se le pueda hacer al hogar, para así tener en cuenta el costo y el esfuerzo requerido.
- **Fiabilidad y seguridad:** Todos estos sistemas tienen vulnerabilidades, por eso se debe tener buen conocimiento de los equipos que se manejan, cuidado con la información que se transmite y contar con sistemas que garanticen un funcionamiento ininterrumpido.

#### 4.2. SISTEMAS EMBEBIDOS

Los sistemas embebidos, algunas veces llamados sistemas empotrados (ocultos a la vista), a pesar de no ser conocidos por la mayoría de personas, actualmente se pueden encontrar en gran parte de dispositivos. Estos sistemas se identifican por tener una computadora con características especiales. Dichas computadora puede ser un micro-controlador o un micro-procesador, el cual, actúa como el cerebro del sistema y está diseñado para cumplir funciones específicas.

Una definición formal de lo que se entiende por sistema embebido sería:

*“Se entiende por sistemas embebidos a una combinación de hardware y software de computadora, sumado tal vez a algunas piezas mecánicas o de otro tipo, diseñado para tener una función específica”.*<sup>5</sup>

Según esta idea, un sistema embebido es diseñado específicamente para llevar a cabo de manera eficiente una tarea en particular, de tal forma que los recursos utilizados son optimizados para resolver dicho problema en concreto. Así, un sistema embebido se encontraría en un punto medio entre una computadora de propósito general (computadora personal o PC) y un sistema que es diseñado por hardware específicamente para una tarea o un conjunto reducido de ellas (PLD o FPGA).

#### **4.2.1. Características de los sistemas embebidos**

Las principales características de un sistema embebido son el bajo costo y consumo de potencia, debido a que estos emplean procesadores muy básicos, relativamente lentos y memorias pequeñas.

En estos sistemas, la velocidad no solo está dada por la velocidad del reloj del procesador, si no que el total de la arquitectura se simplifica con el fin de reducir costos. Estas restricciones de recursos hacen que se deba usar “sistemas operativos especiales, denominados de tiempo real (RTOS, Real Time Operating System). Los sistemas embebidos deben reaccionar a estímulos provenientes del ambiente, respondiendo con fuertes restricciones de tiempo en muchos casos, por lo tanto, se dice que un sistema trabaja en tiempo real si la información después de la adquisición y tratamiento está vigente”<sup>6</sup>.

Estos sistemas pueden poseer características cómo:<sup>7</sup>

- **Confiabledad:** Se debe tener un sistema capaz de minimizar al máximo las fallas ya que, en algunas aplicaciones, este aspecto puede ser primordial.
- **Seguridad personal:** Si hay una falla de seguridad, el usuario no se debe ver afectado.

---

<sup>5</sup> BARR, Michael. *Programming Embedded Ssystem in C and C++*. O'Reily, 1999.

<sup>6</sup> GALIANA LLINARES, Antonio Nadal. *Sistemas Embebidos*. Pág. 7.

<sup>7</sup> VALDERRAMA ARIAS, Oscar Iván. *Sistemas Embebidos*. Universidad Nacional Abierta y a Distancia UNAD. Programa de Ingeniería Electrónica. Sogamoso, Diciembre de 2009. Págs. 15 y 16.

- **Seguridad informática:** Debe haber una comunicación efectiva, confidencial y autenticada (encriptación).
- **Eficiencia:** Se determina la eficiencia de un sistema si cumple con diferentes aspectos como el manejo de la energía suministrada, tamaño reducido del dispositivo, tamaño reducido del código que gobierna el sistema, minimización del peso y, sobre todo, reducir los costos para su producción masiva.
- **Tiempo real:** Debe responder, en un lapso de tiempo lo suficientemente corto, a estímulos externos predefinidos, ya sea que provengan del objeto de control o mediante interfaces de usuario, para que se dé el correcto funcionamiento del sistema.
- **Interfaces de usuario:** La mayoría de sistemas embebido cuentan con algún medio por el cual interactuar con el usuario, ya sea a través de pantallas gráficas, botones, teclados, sensores, etc.

Actualmente en el mercado se pueden encontrar diferentes sistemas embebidos, cada uno condicionado por sus necesidades particulares, funcionamiento y, en algunos casos, con un sistema operativo. Entre los sistemas embebidos conocidos tenemos el Raspberry PI, la NI Single-Board RIO de National Instruments, la ECM-CDV de Data Modul, la mini2440, entre otras.

Los sistemas embebidos pueden tener gran variedad de subsistemas de entrada/salida. Uno de los más comunes es el llamado *entrada/salida de propósito general*, más conocido como GPIO (por sus siglas en inglés General Purpose Input-Output). Se suele utilizar para leer el estado de interruptores, botones y otros dispositivos externos, o bien para activar o desactivar acciones, procesos, etc.<sup>8</sup>

### 4.3. DISPOSITIVOS MÓVILES

Se puede considerar un dispositivo móvil cualquier dispositivo que se pueda utilizar mientras el usuario se encuentra en movimiento y está libre de cableado. Es común que hoy en día las personas utilicen diferentes dispositivos móviles, ya sean grandes o de bolsillo, que disponen de batería suficiente para poder funcionar de forma autónoma. Estos dispositivos o micro-ordenadores (algunos autores discrepan con que pueden ser considerados sistemas embebidos, aunque otros no) tienen capacidad de procesamiento, conexión permanente o intermitente a una red

---

<sup>8</sup> MARCOTEGUI ZABALZA, Fernando. *Representación de variables eléctricas en el sistema embebido mini2440*. Universidad Pública de Navarra. Escuela técnica superior de Ingenieros Industriales y de Telecomunicaciones. Pamplona, 2011. Pág. 10.

y presentan memoria limitada (es necesario aclarar que no todos los dispositivos inalámbricos son móviles y viceversa). Los ordenadores portátiles (laptop) no se consideran dispositivos móviles, ya que consumen más batería y suelen ser más pesados de lo que se espera de algo pensado para llevar siempre encima.

#### **4.3.1. Características de los dispositivos móviles**

Los dispositivos móviles, también conocidos como computadora de mano, son pequeños aparatos electrónicos capaces de moverse con el propio usuario, haciendo de su vida más sencilla y placentera. Dentro de las características más destacadas tenemos:

- Gran capacidad de comunicación, permitiendo tener acceso a información y servicios independiente del lugar y el momento en el que se encuentre el usuario.
- Reducido tamaño, permitiendo su fácil transporte.
- Recursos tanto a nivel personal como a nivel empresarial.

A grandes rasgos, y dependiendo del tamaño y funcionalidad de los dispositivos, se pueden clasificar en cinco (5) clases:

- Teléfono móvil
- Tabletas (tablets)
- PDA (Asistente personal digital)
- Reproductores y grabadores multimedia
- Videoconsolas

#### **4.4. SISTEMAS OPERATIVOS**

Un Sistema Operativo (SO) es un programa que actúa como intermediario entre el usuario y el hardware de un dispositivo. El propósito del SO es proporcionar un entorno en el cual el usuario pueda ejecutar los programas y verlos. Este programa se ejecuta continuamente en los dispositivos. El SO administra los recursos de la máquina, coordina el hardware y organiza archivos y directorios en dispositivos de almacenamiento. Los sistemas operativos se pueden encontrar tanto en dispositivos móviles como en sistemas embebidos.

#### 4.4.1. Sistemas operativos para dispositivos móviles

Los dispositivos móviles, con el avance de la tecnología, han logrado incorporar en su interior diferentes sistemas operativos (dependiendo de su fabricante) para lograr una completa interacción entre el usuario y el dispositivo. Se puede deducir que el uso de uno u otro SO determinan las capacidades multimedia de los dispositivos, y la forma de éstas interactuar con el usuario. Existen en el mercado una diversidad de opciones, si bien las más extendidas son Windows Phone (antes Windows Mobile), BlackBerry OS, iOS y Android; además, de los dispositivos con sistema operativo Linux puro.

**Figura 1.** Sistemas Operativos más conocidos para dispositivos móviles



##### 4.4.1.1. Windows Phone

Sistema operativo desarrollado por la empresa Microsoft<sup>9</sup>, sucediendo a Windows Mobile. Este sistema operativo es adoptado por la empresa fabricante Nokia<sup>10</sup>, luego de que en 2012 sus ventas netas bajaran un 22% con respecto al 2011 usando Symbian, aunque, en algunos móviles, y dependiendo del operador local, también

---

<sup>9</sup> Empresa dedicada al desarrollo, fabricación, licencia y producción de software y equipos electrónicos. Una de las más grandes y poderosas multinacionales a nivel mundial. Es propietaria de Windows y Office.

<sup>10</sup> Multinacional finlandesa dedicada a la tecnología y comunicaciones. Recientemente vendió la división de dispositivos móviles a Microsoft.

es usado por otros fabricantes como Acer, Toshiba, ZTE,<sup>11</sup> Huawei,<sup>12</sup> LG, Samsung, HTC<sup>13</sup> o Dell. Este sistema operativo está más enfocado al mercado de consumo en lugar del mercado empresarial, como lo hizo su antecesor. Cuenta con una interfaz de usuario llamada Modern UI<sup>14</sup> la cual integra mosaicos que son enlaces a diferentes aplicaciones. Dichas aplicaciones pueden ser descargadas desde la tienda oficial de Windows Phone.

#### 4.4.1.2. BlackBerry OS

Hasta hace algunos años fue el SO multitarea con mayor afluencia en la escena empresarial, en especial por sus servicios para correo y teclado QWERTY. En un dispositivo BlackBerry es posible redactar, enviar y recibir todo tipo de mensajes de correo electrónico, al igual que en el programa que se utiliza en un ordenador. Además, es posible realizar y contestar a las llamadas que se emitan a través de la red de telefonía móvil, lo que permite sustituir el teléfono móvil. También, como evolución lógica, los dispositivos de este fabricante permiten la navegación por internet en páginas HTML o WAP y tienen la capacidad de enviar o recibir mensajes SMS. Por lo demás, este sistema operativo incorpora múltiples aplicaciones y programas que convierten a los dispositivos en completos organizadores de bolsillo con funciones de calendario, libreta de direcciones, bloc de notas, lista de tareas, entre otras.<sup>15</sup>

#### 4.4.1.3. iOS

Inicialmente conocido como iPhone OS, iOS es el sistema operativo usado en los dispositivos móviles de la marca Apple. Originalmente fue diseñado para los teléfonos móviles iPhone, pero a medida que ha pasado el tiempo se ha ido incorporando en otros dispositivos como el iPad y el iPod Touch. Es una versión reducida de Mac OS X optimizada para los procesadores ARM. Aunque oficialmente no se puede instalar ninguna aplicación que no esté firmada por Apple (no es de código abierto) ya existen formas de hacerlo.<sup>16</sup> Anualmente este SO es actualizado

---

<sup>11</sup> Acrónimo de Zhong Xing Telecommunication Equipment Company Limited. Proveedor global de equipos de telecomunicaciones y soluciones de redes. Cuenta con gran variedad de productos.

<sup>12</sup> Mayor fabricante de redes y equipo de telecomunicaciones de china y el mundo.

<sup>13</sup> Fabricante de Smartphone taiwanés.

<sup>14</sup> Interfaz de usuario la cual desliza una pestaña en donde aparecen las aplicaciones en pequeños mosaicos. Se asemeja con la pestaña que desliza el Windows 8.

<sup>15</sup> BAZ ALONSO, Arturo, FERREIRA ARTIME, Irene, ÁLVAREZ RODRIGUEZ, María, GARCÍA BANIELLO, Rosana. *Dispositivos móviles*. Universidad de Oviedo. Ingeniería de telecomunicaciones. Pág. 7.

<sup>16</sup> *Dispositivos móviles*. Pág. 7.

y a la fecha está en su 7 versión, incorporando características y mejoras exclusivas para los dispositivos. Este sistema no soporta Java ni Adobe Flash, por lo que los sitios web con dichas tecnología no pueden ser vistos en este sistema operativo. Dada su poca libertad de código y restricciones, no muchas personas pueden desarrollar para esta plataforma. Actualmente es el segundo sistema operativo más utilizado.

#### 4.4.1.4. Android

Actualmente es el sistema operativo líder en el mercado mundial de los dispositivos móviles. Su logo está representado por un androide verde apodado “Andy”. Se trata de un sistema operativo de software libre y código abierto (puede ser estudiado, modificado, mejorado y redistribuido, mediante la disponibilidad de su código fuente, por cualquier persona), basado en Linux. Las versiones de este sistema tienen una peculiaridad, ya que, además de ir en orden alfabético, toman nombres de postres. En la actualidad, la versión más reciente es la 4.4.4 denominada Kitkat. Sin embargo, en la última Conferencia de Desarrolladores llamada Google I/O 2014, esta empresa dio a conocer lo que será su nueva versión de Android apodada “Android L”.

**Figura 2.** Versiones sistema operativo Android







**Fuente:** LOS POSTRES DE ANDROID. La esquina positiva. Cristiiniita. Noviembre 6 de 2013. [En línea. Acceso libre] <<http://laesquinapositiva.wordpress.com/2013/11/06/los-postres-de-android/>>



De forma nativa, éste sistema es programado en lenguaje Java, sin embargo hay quienes dicen que este sistema fue programado de forma nativa en lenguaje C<sup>17</sup>.

**Tabla 1.** Comparación sistemas operativos móviles

				
<b>Compañía propietaria</b>	Open Hanset Alliance	Apple	Microsoft	Research In Motion (RIM)
<b>Año de lanzamiento</b>	Septiembre de 2008 Android 1.0 Apple Pie	Junio de 2007 iPhone OS 1.0	Noviembre de 2010 Photon 7.0	Marzo de 2003 RIM OS 3.0
<b>Núcleo del SO</b>	Linux	XNU híbrido Darwin & Mac OS X	Windows CE (WP7) Windows NT (WP8)	Basada en Java
<b>Licencia de software</b>	Libre y abierto	Propietario	Propietario	Cerrado
<b>Familia CPU soportada</b>	ARM, x86, MIPS, IBM, POWER	ARM	Silverlight Microsoft XNA	Línea de smartphones de BlacBerry
<b>Máquina virtual</b>	Dalvik	No	.net	Java
<b>Lenguaje de programación</b>	Java, C, C++	C, C++, Objective-C	C#. VB.NET, otros	Java, C++
<b>Plataforma de desarrollo</b>	Windows, MAC, Linux	Mac	Windows	Windows, Mac
<b>Kit de desarrollo SDK</b>	Android SDK	Xcode	Microsoft Windows SDK	BlackBerry Java SDK
<b>Motor del navegador web</b>	WebKit	Safari	Internet Explorer	WebKit
<b>Soporte Flash</b>	Si	No	Si	Si
<b>Soporte HTML5</b>	Si	Si	Si	Si
<b>Tienda de aplicaciones</b>	Play Store	App Store	Windows Phone Market	BlackBerry App World
<b>Sistema de gestión de paquetes</b>	.apk	.app	.xap	.cod
<b>Última versión estable</b>	Android 4.4.3 Kitkat	iOS 7.1	Windows Phone 8	BlackBerry 7.1

Para descargar aplicaciones de forma gratuita Android cuenta con su propia tienda on-line llamada Google Play (anteriormente conocida como Play Store). Esta

<sup>17</sup> Según Wikipedia: La enciclopedia libre, “Este sistema operativo está compuesto por 12 millones de líneas de código, incluyendo 3 millones de líneas de XML, 2.8 millones de líneas de lenguaje C, 2.1 millones de líneas de Java y 1.75 millones de líneas de C++”.

aplicación también se puede encontrar en todos los teléfonos con este sistema operativo.

#### **4.4.2. Sistemas operativos embebidos**

Los sistemas operativos embebidos son aquellos que fueron creados especialmente para que funcionen en un sistema embebido, como un cajero automático, lavadora, horno microondas, etc. Estos sistemas operativos presentan características de tiempo real, pero a su vez presentan limitaciones de tamaño, memoria y energía.

Así como hay gran variedad de sistemas operativos para teléfonos móviles, se pueden encontrar diferentes sistemas operativos embebidos, estos son:

- VxWorks
- Windows XP para sistemas embebidos
- QNX
- Sistema operativo Linux

#### **4.5. TARJETA DE DESARROLLO MINI2440**

La tarjeta de desarrollo (Board) mini2440 representa la herramienta central de este proyecto, ya que ésta será la encargada de ejecutar la aplicación, cuyo diseño es el objeto del presente trabajo.

La Board mini2440 es un dispositivo de origen Chino, el cual es distribuido, a nivel internacional, por compañías como FriendlyARM e Industrial ARMWorks. A nivel nacional se pueden encontrar compañías que se encargan de traer el dispositivo, es el caso de tdrobótica, SigmaElectrónica, DynamoElectronics o DidácticasElectrónicas, aunque se debe hacer sobre pedido. Se podría decir que se trata de un sistema de computación versátil y de propósito múltiple, pero con notables limitaciones de recursos con respecto a un PC.

Sorprende la arquitectura con la que viene la board, pues para ser un componente de baja potencia, trabaja con tecnología RISC (ARM920T de 32 bits), siendo los ciclos de reloj mucho más rápidos que su oponente CISC. Utiliza un micro-controlador fabricado por la empresa surcoreana Samsung (S3C2440A).

Esta tarjeta viene en distintas versiones, presentando cada una pequeñas diferencias en cuanto a su configuración y prestaciones. Se puede adquirir con o sin pantalla LCD táctil, ya sea de 3,5" o 7", dependiendo de su referencia (ubicada en la parte trasera de la pantalla, W35i, X35, N35 o T35) puede ser diseñada por SONY, Toppoly, NEC o Sharp, de mayor memoria NAND Flash (64, 128, 256 o 1024 Mb).

Para este proyecto se utilizó una tarjeta mini2440 de 1024 Mb NAND Flash con pantalla LCD táctil de 3,5" referencia W35i, fabricada por Sharp, con una resolución de 320x240. El kit también incluía un adaptador de corriente eléctrica, un cable de conexión serie RS232 y USB, un cable de red Ethernet y un adaptador JTAG.

Al ser una tarjeta de desarrollo, son múltiples las tareas que programadores o desarrolladores pueden hacer, teniendo la posibilidad de probar ahí mismo sus aplicaciones, además, "su robustez y su precio reducido hacen que esta tarjeta sea de gran utilidad en el campo de la docencia y la investigación".<sup>18</sup>

**Figura 3.** Kit tarjeta de desarrollo mini2440



Para poder describir la Board mini2440 es necesario mirar tanto su hardware como su software.

#### **4.5.1. Hardware**

La board mini2440 está construida sobre una tarjeta de 10 x 10 cm, y cuenta con una gran variedad de interfaces y sistemas de almacenamiento, los cuales, están

---

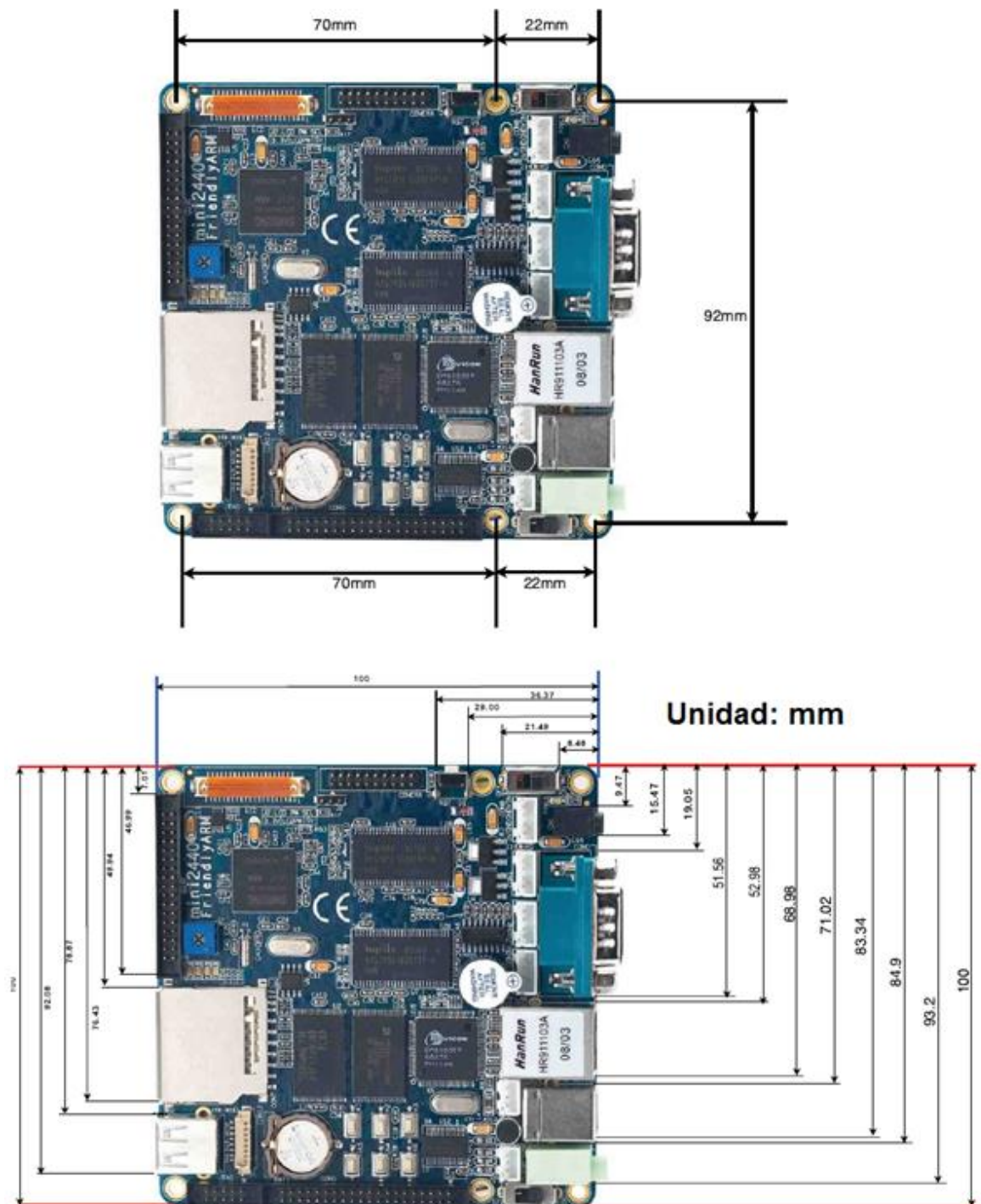
<sup>18</sup> MARCOTEGUI ZABALZA, Fernando. *Representación de variables eléctricas en el sistema embebido mini2440*. Universidad Pública de Navarra. Escuela técnica superior de Ingenieros Industriales y de Telecomunicaciones. Pamplona, 2011. Pág. 17.

montados sobre la placa. Incluye una pantalla LCD de 3.5", la cual está sobre la tarjeta y es sujeta a través de tornillos, en caso de que el usuario quiera quitarla para, ya sea ver sus referencias o trabajar, no tenga ningún inconveniente y su interacción sea mucho más sencilla. El PCB (tarjeta de circuito impreso) de la board está diseñado para ser de 4 capas, adoptando tecnología ENIG<sup>19</sup> y cableado profesional de igual longitud para asegurar la integridad de las señales del cable de señal clave.

---

<sup>19</sup> Níquel electrolítico de inmersión de oro (ENIG por sus siglas en inglés) es un tipo de siembre superficial utilizado para tarjetas de circuito impreso (PCB).

Figura 4. Dimensiones board mini2440



Fuente: <[http://www.friendlyarm.net/dl.php?file=mini2440\\_dimension.pdf](http://www.friendlyarm.net/dl.php?file=mini2440_dimension.pdf)> FriendlyARM.

A continuación se mencionan todos los componentes hardware de la board mini2440:<sup>20</sup>

- **Micro-controlador:**

- S3C2440A de arquitectura ARM920T CMOS de 32 bits
- Frecuencia principal de 400 MHz (Max. 533 MHz)
- 200 MIPS efectivas a 180 MHz

- **Memoria RAM:**

- SDRAM de 64 MB montada sobre la placa
- Bus de datos de 32 bits
- Frecuencia de reloj de la SDRAM de hasta 100 MHz

- **Memoria flash:**

- NAND Flash de 1024 MB (chip K9K8G08), no volátil
- NOR Flash de 2 MB con BIOS instalada, no volátil

- **Interfaz LCD:**

- Interfaz táctil resistiva de 4 canales conectada al ADC de 10 bits de resolución
- Soporte para displays LCD STN escala de grises de 4 y 16 niveles, 256 y 4096 colores
- Soporte para displays LCD TFT escala de grises de 4 y 16 niveles, 256, 64k colores y true color
- Tamaño de pantalla 3.5". Existen versiones de hasta 12.1"
- Resolución máxima de pantalla de 1024 x 768 píxeles
- Configuración estándar para el LCD táctil NEC 240 x 320 / 3.5" TFT True color
- Conectada a una interfaz de alimentación sobre la placa de 12V, para la alimentación del módulo backlight LCD TFT 12V CCFL<sup>21</sup>

---

<sup>20</sup> Industrial ARMWorks. *Mini2440 Hardware Essentials*. Agosto de 2009.

< <http://www.andahammer.com/assets/Uploads/All2440/Firmware/Mini2440Essentials.pdf> >

<sup>21</sup> Lámparas fluorescentes de cátodo frío (CCFL, Cold Cathode Fluorescent Lamp). No tienen filamentos. Usan alta tensión para encender. Su consumo es muy bajo en comparación con una lámpara fluorescente normal, pero su rendimiento luminoso es más bajo.

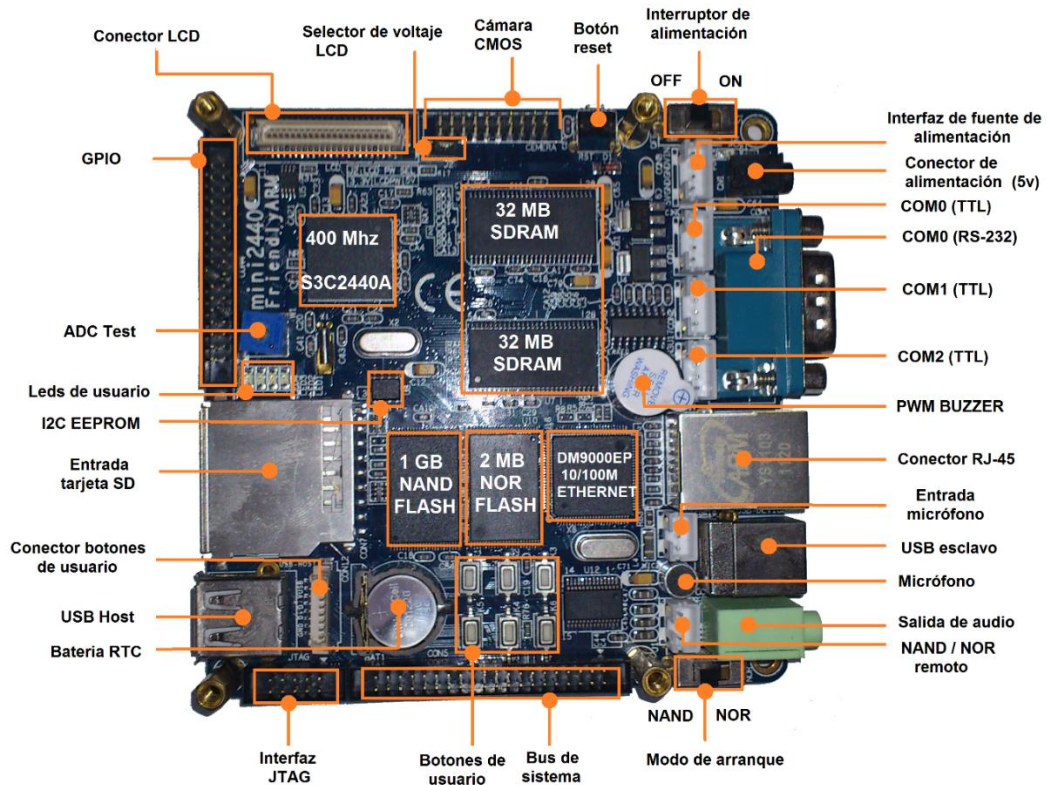
- **Otras interfaces y accesorios externos:**
  - 1 interfaz 10/100M Ethernet RJ-45 (chip de red DM9000)
  - 3 puertos serial, uno configurado para RS-232 (COM0)
  - 1 USB host
  - 1 interfaz USB esclavo tipo B hembra
  - 1 interfaz para tarjetas de almacenamiento SD, sin límite de tamaño
  - 1 interfaz de salida de audio estéreo
  - 1 micrófono integrado
  - 1 entrada de micrófono
  - 1 interfaz JTAG de 10 pines de 2.0 mm de espacio
  - 4 LEDs de usuario
  - 6 botones de usuario, con conexión a conectores GPIO y 8 pines de conector de usuario
  - 1 botón de reinicio (reset), conectado a un chip de reset
  - 1 interruptor de selección de arranque desde Flash NAND/NOR
  - 1 potenciómetro (resistencia variable) para test del ADC (A/D test)
  - 1 timbre PWM (buzzer)
  - 1 chip de bus I2C EEPROM para datos de configuración o prueba, abarca 256 bytes
  - 1 interfaz de cámara de 20 pines de 2.0 mm de espacio
  - 1 interfaz de alimentación (5V), con interruptor de encendido (power switch) y luz indicadora
  
- **Fuente de reloj del sistema:**
  - De cristal pasivo
  - Reloj interno de tiempo real RTC(con batería de litio de respaldo)
  
- **Interfaz de expansión:**
  - 1 interfaz GPIO de 34 pines de 2.0 mm de espacio
  - 1 interfaz de bus del sistema de 40 pines de 2.0 mm de espacio

La board mini2440 usa algunos recursos de hardware, los cuales son:

- **Puentes:** Hay un solo puente (J2). Éste se utiliza para seleccionar el voltaje de entrada de la tarjeta para el manejo del LCD (3.3 o 5V).
- **Interfaces:** Las interfaces se mencionaron anteriormente y se pueden observar en la Figura 5.



**Figura 5.** Interfaces board mini2440



#### 4.5.2. Micro-controlador S3C2440A

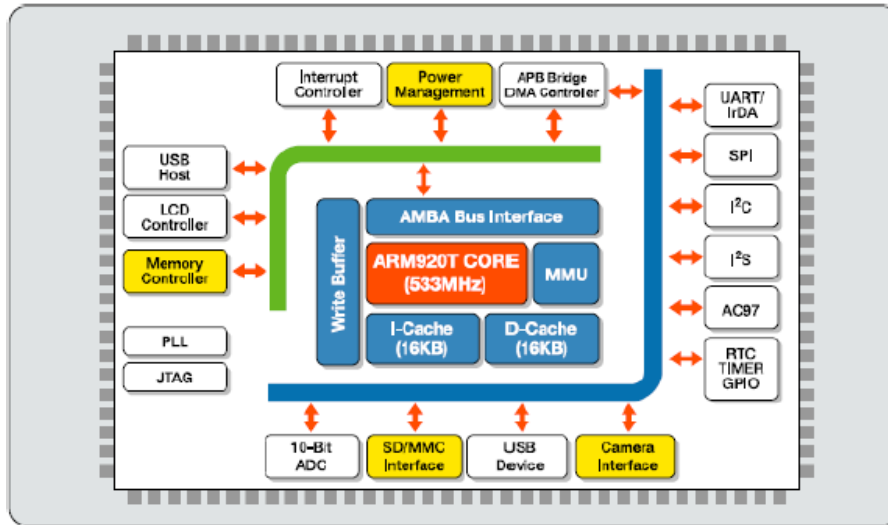
Este micro-controlador CMOS de 32 bits, fabricado por la empresa surcoreana Samsung, pertenece a la familia ARM9TDMI de la arquitectura ARM. La versión de la arquitectura es la ARMv4T y está desarrollado con el núcleo ARM920T. El S3C2440 está diseñado para dispositivos portátiles y aplicaciones en general de baja potencia.

Este micro-controlador se basa en la arquitectura Harvard<sup>22</sup>, al separar los caches de instrucciones y datos, que está dirigido a aplicaciones multi-programa de alto rendimiento y bajo consumo de energía. Implemente una arquitectura mejorada MMU (Unidad de Gestión de Memoria) para proporcionar servicios de traducción y control de permisos de acceso para las instrucciones.

<sup>22</sup> Este término proviene de la computadora Harvard Mark I, la cual almacenaba instrucciones en cintas perforadas y los datos en interruptores. Esta arquitectura lo que hace es separar las instrucciones y los datos y almacenarlos en cachés separadas para mejorar el rendimiento (en oposición a la arquitectura de Von Neumann).



**Figura 6.** Arquitectura núcleo ARM920T



**Fuente:** [En línea. Acceso libre]. <<http://www.datacheetking.com/Indexdl/Scans-007/Scans-00142900.pdf/>>

La interfaz del ARM920T con el resto del sistema se hace sobre buses unificados de datos y direcciones. Esta interfaz habilita la implementación de un esquema de bus tipo AMBA<sup>23</sup>, ASB<sup>24</sup>, o AHB<sup>25</sup>. Soporta macro-celda ETM<sup>26</sup> para el rastreo en tiempo real de datos e instrucciones.

En la arquitectura ARM920T, el procesador dispone de 37 registros de 32 bits, de los cuales:<sup>27</sup>

- 6 son registros de estado, aunque no se usan todos los bits.
- 31 son de propósito general, incluyendo el contador del programa.

Este tipo de arquitectura posee siete (7) tipo de excepción, cada una de las cuales se procesa mediante un modo privilegiado de ejecución. Estas son:<sup>28</sup>

<sup>23</sup> Arquitectura de bus de micro-controladores avanzada. Se utiliza cuando son diseños SoC.

<sup>24</sup> Sistema avanzado de bus.

<sup>25</sup> Bus avanzado de alto rendimiento.

<sup>26</sup> Macro-celda de trazado embebido. Tecnología de depuración de bajo nivel.

<sup>27</sup> MARCOTEGUI ZABALZA, Fernando. *Representación de variables eléctricas en el sistema embebido mini2440*. Universidad Pública de Navarra. Escuela técnica superior de Ingenieros Industriales y de Telecomunicaciones. Pamplona, 2011. Pág. 23.

<sup>28</sup> *Representación de variables eléctricas en el sistema embebido mini2440*. Pág. 25.

- Reinicio
- Intento de ejecutar instrucción indefinida
- Interrupciones software (SWI), pueden usarse para implementar llamadas al sistema operativo.
- Error al buscar una interrupción en la memoria.
- Error al acceder a datos en memoria.
- IRQ, interrupciones normales
- FIQ, interrupciones rápidas.

#### 4.5.3. Software<sup>29</sup>

Dada la complejidad de la configuración del hardware de esta board, está diseñada para ser dominada por un sistema operativo que gestione los procesos básicos del sistema. El sistema tiene suficiente capacidad de procesamiento y memoria para albergar y ejecutar un sistema operativo multitarea/multiusuario con una interfaz gráfica de usuario completa.

Esto significa que la persona que utilice esta tarjeta de desarrollo deberá primero instalar un sistema operativo previamente compilado para esta arquitectura, junto con los controladores de los subsistemas. Luego de esto, podrá programar y desarrollar sus aplicaciones haciendo uso de las plataformas y librerías disponibles que le puedan dar soporte.

Al día de hoy, existen varias opciones de sistemas operativos para la ARM mini2440, dentro de los cuales se encuentran:

- Linux
- Android
- Windows CE

Para el desarrollo del presente proyecto se escogió por sistema operativo una distribución de Linux llamada Qtopia. Sobre este sistema operativo se hablará más adelante. Para saber cómo instalarlo ver **Anexo A**.

La descripción detallada de las interfaces de la board ARM mini2440 se puede ver en el **Anexo C**.

---

<sup>29</sup> MARCOTEGUI ZABALZA, Fernando. *Representación de variables eléctricas en el sistema embebido mini2440*. Universidad Pública de Navarra. Escuela técnica superior de Ingenieros Industriales y de Telecomunicaciones. Pamplona, 2011. Pág. 28.

## 5. DESCRIPCION Y DESARROLLO DEL SISTEMA

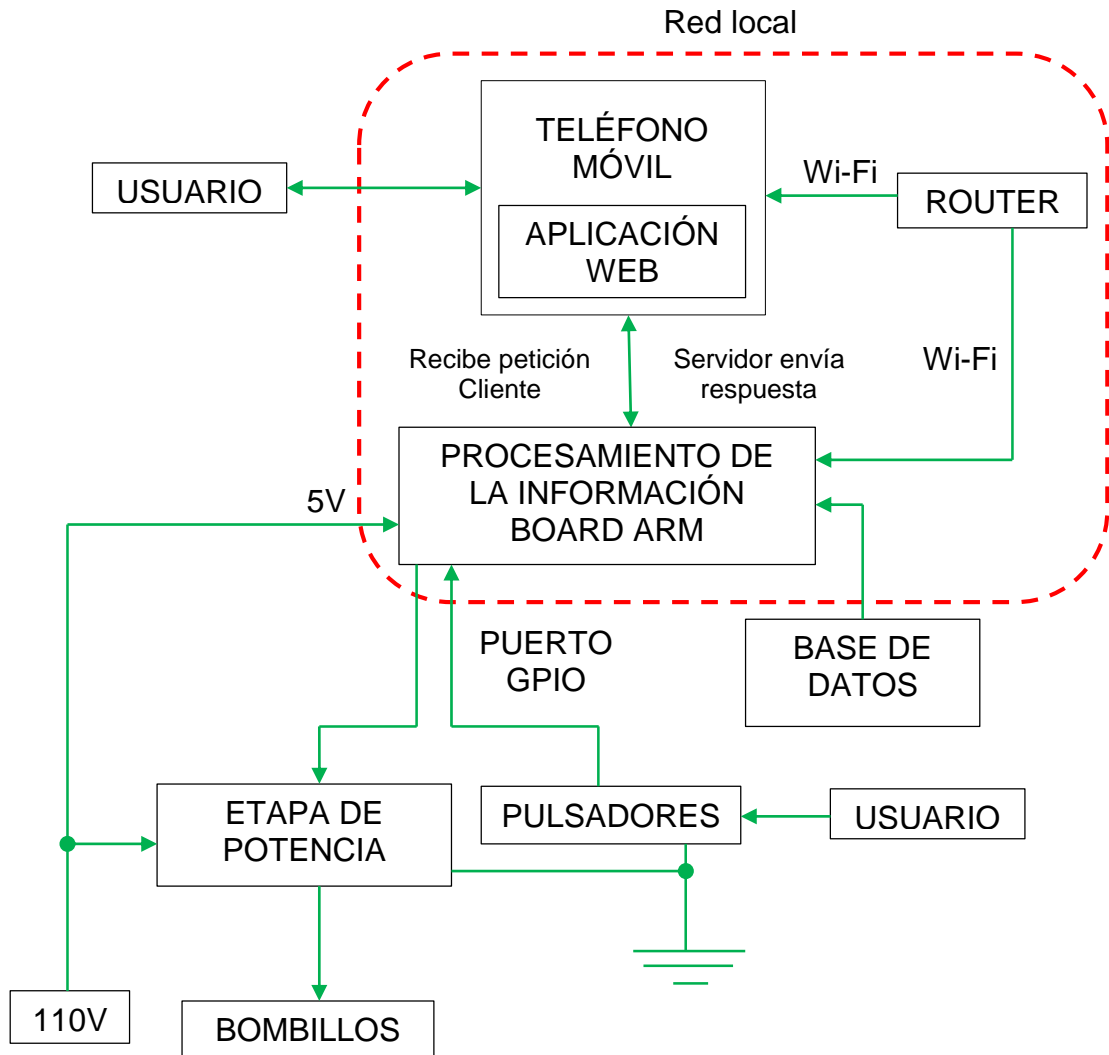
El sistema de control y monitoreo remoto de los bombillos se realizó mediante el uso del puerto GPIO de la board ARM mini2440, la cual, utiliza un circuito de potencia para acoplar los elementos a la red eléctrica. Esta board hará las veces de servidor, y será a través de ella que se logre enviar y recibir las peticiones del cliente. En el puerto GPIO se seleccionaron 5 pines como entrada y 5 pines como salida. En los pines de entrada se conectaron los pulsadores y en los de salida los bombillos, los cuales van anteceditos por el circuito de potencia. Se hace uso de un nano Router de marca TP-Link referencia TL-WR702N para que todo el circuito esté en la misma red, permitiendo la interacción del sistema con el usuario.

Dado que es un sistema pensado para controlar las luces de una casa remotamente (hogar domótico), se cambian los tradicionales interruptores por pulsadores, simulando interruptores digitales, los cuales al pulsarse, permitan al usuario encender o apagar el bombillo manualmente. Todos los datos serán registrados en una base de datos hecha en SQLite3, en dónde se podrá observar la hora y la fecha de encendido y apagado de los bombillos, mostrar el bombillo que se encendió y el tiempo que éste lleva encendido. Esta base de datos estará alojada en el servidor.

Para el diseño de control se trabajó desde un computador de escritorio que contara con sistema operativo Linux (en este caso se utilizó la distribución OpenSuse en su versión 12.3), ya que la compilación del código sólo se puede hacer a través de la consola de este sistema.

La aplicación web se encargará de la interfaz gráfica para que el usuario pueda interactuar cómodamente con el control de los bombillos, ya sea accediendo a la aplicación desde un teléfono móvil con tecnología Android o desde cualquier dispositivo móvil con navegador web ingresando la dirección IP del servidor (192.168.1.230). Desde ahí se podrá observar el tiempo real que lleva encendido cada bombillo, y, al introducir la potencia, en wattios, el usuario podrá saber cuánto dinero debe pagar hasta ese momento por el consumo de los bombillos. Se crearon dos botones: uno para eliminar la tabla de registros, pero esto sólo eliminaría la tabla, ya que los registros siguen almacenados, si no, las consultas no se podrían realizar, y otro, para ver los registros que están siendo almacenados en la base de datos. Al haber varios dispositivos conectados al servidor, se puede ver el cambio casi que instantáneo (ya que tiene un pequeño retardo) del estado de los bombillos, además, el tiempo que éstos llevan encendidos sólo será sincronizado cuando el bombillo respectivo se apague.

Figura 7. Esquema general del proyecto



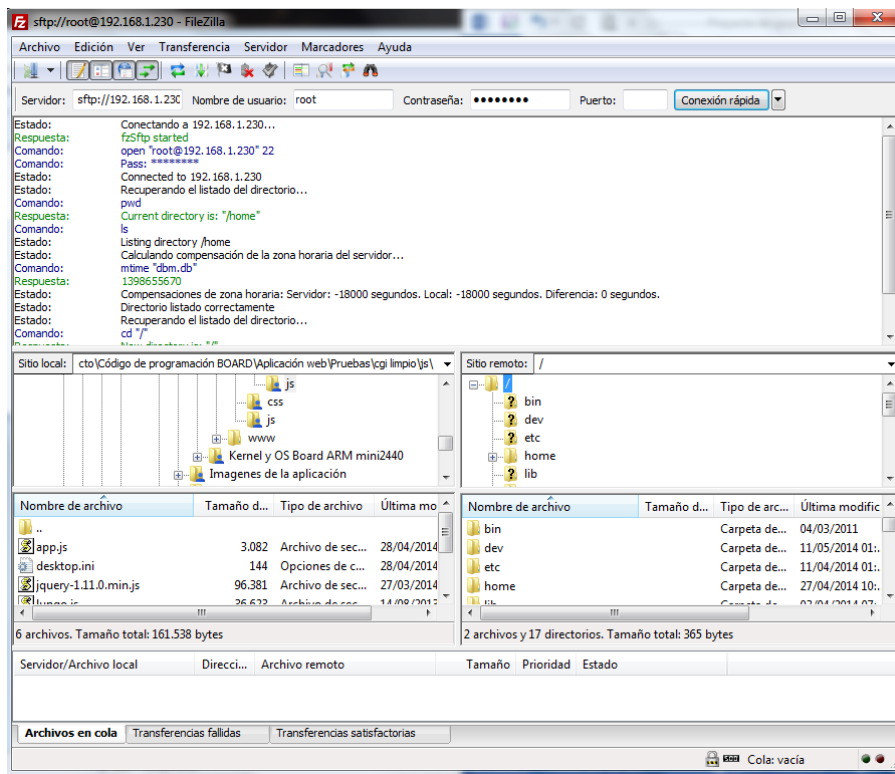
### 5.1. COMUNICACIÓN BOARD ARM MINI2440 – PC

Luego de cambiar el sistema operativo por defecto que trae la board ARM mini2440, se estableció una comunicación entre ésta y el PC, para mayor comodidad de trabajo. Por defecto, la tarjeta se puede trabajar desde la consola que se encuentra ubicada dentro de la board, pero este trabajo es muy tedioso, ya que, al no tener teclado físico, toca que el usuario escriba, por medio del lápiz óptico, letra por letra en la consola, haciendo el trabajo más largo y dispendioso.

La board ARM mini2440 cuenta con una dirección IP la cual se utiliza para acceder a ésta y funciona como servidor. La dirección para tener acceso es 192.168.1.230. Se debe aclarar que para lograr tener acceso se debe estar en red con el servidor de la tarjeta.

Para llevar a cabo la transferencia de archivos se utilizó un programa llamado FileZilla<sup>30</sup>. Por medio de éste es posible llevar a cabo una conexión con la máquina a través de diferentes protocolos (FTP, SFTP, FTPS, FTPES). Este cliente multiplataforma sirve para ser usado tanto en versiones de Windows como en versiones de Linux.

**Figura 8.** Interfaz FileZilla para transferencia de datos entre la board – PC




Para solucionar este pequeño inconveniente, la board ARM mini2440 cuenta con un protocolo de red por defecto llamado TELNET<sup>31</sup>, el cual permite la gestión y

<sup>30</sup> Cliente FTP multiplataforma de código abierto y software libre. Soporta diferentes protocolos de transferencia, además del FTP.

<sup>31</sup> Protocolo de red que permite viajar a otra máquina para manejarla remotamente.

transferencia de archivos entre la board y el PC pero únicamente a través del protocolo FTP<sup>32</sup>. El mayor problema que posee al conectarse a través de este protocolo se debe a la seguridad, ya que toda la información que se envíe viaja por la red como texto plano (cadena de texto sin cifrar), facilitando que personas ajenas puedan acceder a la información. Además de este problema de seguridad, para este caso, al pasar archivos de la board al PC y viceversa, había pérdida de datos, dando como resultado que el archivo enviado o recibido no pasaba completo, provocando errores al intentar abrirlos.

**Figura 9.** Conexión con la board a través del protocolo de red TELNET desde la consola de Linux



```
fernando@linux-29fk:~$ su -
Contraseña:
linux-29fk:~ # telnet 192.168.1.230
Trying 192.168.1.230...
Connected to 192.168.1.230.
Escape character is '^]'.

Kernel 2.6.32.2-FriendlyARM on (/dev/pts/1)
FriendlyARM login: root
Password:
[root@FriendlyARM /home]#
```

Dado que el protocolo multiplataforma TELNET no cumplió con las expectativas esperadas, se decidió utilizar otro protocolo de red el cual fuera compatible con la board y permitiera la transmisión y recepción de datos completa y segura. Para esto, se utilizó el protocolo de red OpenSSH<sup>33</sup> (Open Secure Shell, Abrir intérprete de órdenes segura), el cual es soportado por la board mini2440 (para saber cómo instalarlo ver **Anexo B**). Para que el protocolo de red funcionara, se debía hacer la activación manual (*/usr/sbin/sshd*), pero, para evitar este tedioso trabajo, se realizó


<sup>32</sup> Protocolo de transferencia de archivos (File Transfer Protocol).

<sup>33</sup> Conjunto de aplicaciones que permiten realizar comunicaciones cifradas a través de una red, usando el protocolo SSH. Permite controlar completamente ordenadores mediante un intérprete de comandos (consola).

un script el cual se configuró para que al encender la board se active automáticamente, permitiendo instantáneamente al usuario tener comunicación a través de éste. El script tiene por nombre “Servicios” y se encuentra ubicado en la dirección `/opt/Qtopia/bin`. Para que se ejecute de forma automática, se debe seleccionar el script “Servicios” en la ventana que dice “Autostart”. El servidor `sshd` es el demonio<sup>34</sup> responsable de gestionar las conexiones remotas SSH al equipo. Éste es el responsable de que se pueda lograr acceder remotamente a la board para realizar distintas tareas a través de la consola.

Con cualquiera de las dos (2) formas de comunicación remota con la board, Telnet u OpenSSH, el usuario deberá acceder con privilegios de `root`<sup>35</sup>, esto garantizará al usuario tener acceso total a las diferentes carpetas de la board, podrá proporcionar o quitar permisos de lectura o escritura en los distintos archivos, entre otras actividades. De no hacerlo, ni siquiera podrá ingresar el comando para acceder a la board.

**Figura 10.** Conexión con la board a través del protocolo de red OpenSSH desde la consola de Linux



```
(root) 192.168.1.230 - Konsole
Archivo Editar Ver Marcadores Preferencias Ayuda
fernando@linux-29fk:~> su -
Contraseña:
linux-29fk:~ # ssh root@192.168.1.230
root@192.168.1.230's password:
[root@FriendlyARM /home]#
```

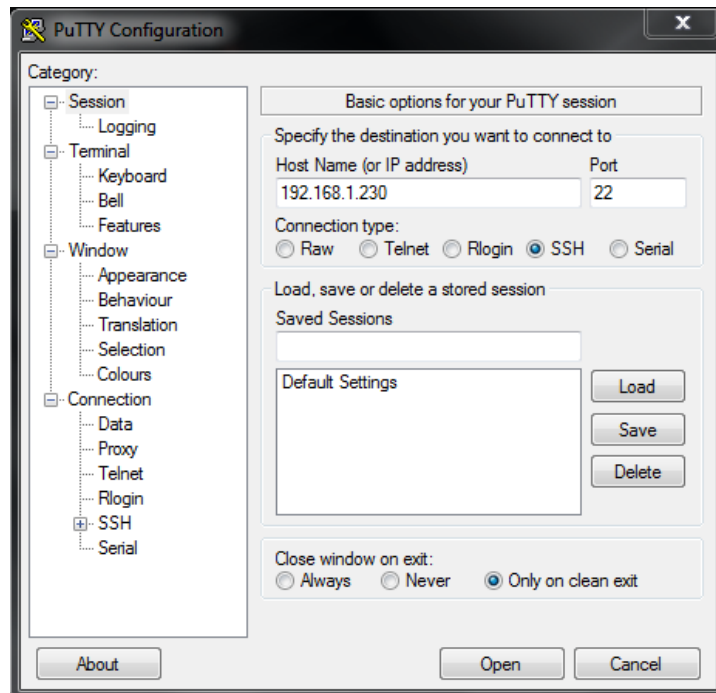
---

<sup>34</sup> Proceso informático especial no interactivo, es decir, se ejecuta en segundo plano en vez de ser controlado directamente por el usuario.

<sup>35</sup> También es conocido como superusuario. El usuario `root` es lo que normalmente se conoce como administrador, ya que posee todos los derechos sobre la cuenta a la que está accediendo.

Ya que el control remoto de la board se hace a través de consola, el usuario que use Windows deberá tener una instalada en su equipo, debido a que Windows no cuenta con consola. Linux trae por defecto una consola para realizar las diferentes tareas desde ahí (esto dado por que algunas personas no les gusta manejar interfaz gráfica). Dado que la mayoría del tiempo se manejó el SO Windows, se utilizó PuTTY para lograr la comunicación a través de consola. Este programa se usa como cliente para hacer varios tipos de conexiones a distintos servidores y protocolos como SSH y Telnet. Para lograr tener conexión con el servidor de la board, PuTTY pide introducir el nombre del Host o la dirección IP, que para este caso la dirección IP del servidor por defecto será 192.168.1.230, a la cual se quiere acceder y el tipo de protocolo que desea utilizar para la comunicación (RAW, TELNET, RLOGIN, SSH o SERIAL). Al seleccionar el tipo de conexión, automáticamente se escoge el puerto por el cual se realizará la transmisión entre el cliente y el servidor, y, una vez hecho esto, se presiona el botón “Open” e inmediatamente se abre una ventana en forma de consola desde donde se controlará la board ARM mini2440.

**Figura 11.** Interfaz PuTTY para conexión con el servidor de la board a través del protocolo SSH desde Windows





**Figura 12.** Consola abierta con PuTTY y conectada al servidor de la board



```
192.168.1.230 - PuTTY
login as: root
root@192.168.1.230's password:
[root@FriendlyARM /home]#
```

Al querer ingresar al servidor de la board, ya sea para intercambiar archivos a través de FileZilla o manejar la board por consola, pedirá un USUARIO y CONTRASEÑA. Para este caso el usuario será “root” y la contraseña será “1234abcd”. Cabe aclarar que por defecto la board NO trae contraseña, así que se puede poner “root” o simplemente se da “enter”. El proceso para cambiar la contraseña es algo complejo, así que se recomienda no intentar cambiarla, ya que puede traer problemas futuros para el control de la board remotamente. También se debe tener en cuenta que para realizar la conexión con el servidor de la board ARM mini2440 es necesario estar en red, ya sea tener la board conectada a la red wifi por medio de un cable Ethernet, o, a través de un router inalámbrico, formando así una red LAN.

## **5.2. CIRCUITO DE ACOPLE**

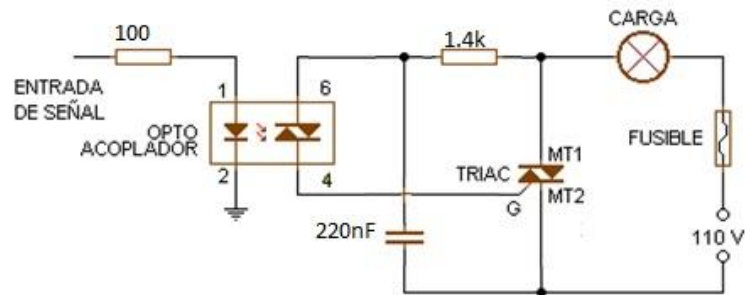
Dado que la board es un elemento de baja potencia, se diseñó un circuito de acople, de tal forma que los circuitos que se vayan a integrar con la board ARM mini2440 no sufran daño alguno, y menos la propia tarjeta de desarrollo.

### **5.2.1. Etapa de potencia para los bombillos**

Después de generar la señal utilizando la board ARM mini2440, es necesario construir un circuito que permita acoplar la señal digital a la red eléctrica domiciliaria. Para el diseño del circuito de acoplamiento se usó el opto-acoplador MOC3010. El MOC controla la compuerta del Triac, y así, mediante la supresión de la corriente, se altera la señal característica del voltaje. El Triac interno del MOC es

controlado por un fotodiodo, que a su vez es controlado por pulsos provenientes del micro-controlador.

**Figura 13.** Circuito esquemático de control de potencia para los bombillos



El MOC se activa y se desactiva cuando la board lo ordena, ya sea debido a la acción del pulsador o debido a la aplicación desde el teléfono móvil, haciendo que la corriente circule y se complete el circuito alimentando los bombillos y permitiendo su iluminación.

Cabe aclarar que se diseñó una (1) etapa de potencia para cada bombillo, lo que significa que se hace uso de cinco etapas de potencias, todas separadas para que trabajen de manera independiente.

### 5.3. DISEÑO Y DESARROLLO DE LA APLICACIÓN

Para obtener el funcionamiento correcto del sistema, se crearon algunos archivos separados, tales que, cada uno sea el que controle una función en específico, por ejemplo, un archivo para el control y muestreo de la base de datos, un archivo para especificar los puertos a utilizar en la board, ya sea como entrada o salida, un archivo para la parte gráfica de la aplicación web y un archivo el cual se debe estar ejecutando todo el tiempo, ya que éste será el encargado de recibir las peticiones hechas por el usuario y enviar su respuesta. Para lograr controlar los bombillos remotamente, se creó una aplicación web móvil, la cual funciona desde un teléfono móvil con sistema operativo Android, además, al ser una aplicación web, puede ser maniobrada desde un PC. Dicha aplicación permite encender o apagar los bombillos, ver la base de datos registrada y borrar su contenido. Además de la aplicación web se creó la aplicación que controlará el servidor de la board ARM mini2440.

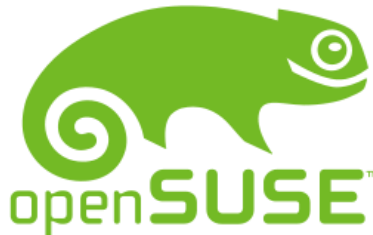
### 5.3.1. Herramientas de desarrollo empleadas: Software

Para el desarrollo de la aplicación, tanto para el servidor como para la aplicación web, se utilizó una serie de programas los cuales permitieron el total desarrollo de ésta.

#### 5.3.1.1. Sistema operativo software libre: OpenSUSE

Dado que la tarjeta solo se puede compilar desde un sistema operativo con Linux, en este caso se utilizó la distribución OpenSUSE en su versión 12.3. Se eligió este SO ya que su interfaz gráfica tiene bastante similitud con Windows, haciendo su manejo mucho más sencillo, además de que tiene la opción de que el usuario pueda elegir entre dos (2) entornos de escritorio: Gnome o KDE. Es gratuito, pues se puede descargar desde su página oficial en internet, totalmente estable y de fácil instalación (se graba la imagen obtenida en un CD dejándolo auto-arrancable y siguiendo los pasos de instalación), además, cuenta con una amplia comunidad de usuarios los cuales brindan soporte.

**Figura 14.** Sistema operativo OpenSUSE



Luego de instalar esta distribución de Linux, el sistema automáticamente se actualiza (si se cuenta con conexión a internet) y descarga paquetes necesarios para el correcto funcionamiento a través de YaST. Todo esto lo hace de forma autónoma y el usuario sólo debe aceptar las descargas propuestas.

#### 5.3.1.2. Compilador cruzado ARM LINUX GCC 4.3.2

Normalmente la board no tiene la misma arquitectura que una computadora de propósito general. El código fuente debe ser compilado para ser ejecutado en una plataforma distinta a aquella en la que éste se ejecuta. Debido a que en los sistemas embebidos es imposible compilar archivos (dada la limitación de recursos), se utilizó un compilador cruzado o cross compiler.

GCC (GNU Compiler Collection, colección de compiladores GNU) es un conjunto de compiladores desarrollados por el proyecto GNU. Estos compiladores se consideran estándar para los sistemas operativos de UNIX, de código abierto y algunos sistemas propietarios como MAC OS X. GCC requiere el conjunto de aplicaciones binutils<sup>36</sup> para realizar tareas como identificar archivos objeto u obtener su tamaño para copiarlos, traducirlos, enlazarlos o quitarles símbolos innecesarios. En su versión 4.3 incluye compiladores para C (gcc), C++ (g++), Java (gcj), Ada (GNAT), Objective-C (gobjc), Objective-C++ (gobjc++) y Fortran (gfortran).

ARM GCC es la extensión de GCC para la compilación cruzada de las arquitecturas ARM. Tanto GCC como ARM GCC son software libre y son distribuidos por la FSF bajo la licencia GPL<sup>37</sup>.

Para este caso se utilizó el compilador cruzado ARM LINUX GCC 4.3.2, debido a que es el único que soporta la tarjeta embebida. Existen varias versiones de este compilador cruzado, pero la versión 4.3.2 fue la única soportada por la board ARM mini2440, las otras versiones presentaron fallos a la hora de la compilación. Este compilador recibe un programa fuente de cualquier de los lenguajes de programación anteriormente mencionados y genera un programa ejecutable binario en el lenguaje de máquina donde ha de correr.

Para la instalación de este compilador cruzado se deben seguir los siguientes pasos (cabe aclarar que estos pasos están hechos para ser ejecutados desde consola en la distribución de Linux):

- Se descarga el archivo `arm-linux-gcc-4.3.2.tgz` desde el CD de instalación (el CD se puede descargar desde el servidor de friendlyarm), se crea una carpeta cualquiera y se copia ahí (`/home/usuario/compilador`).
- Se extrae el archivo descargado con el siguiente comando:

```
# tar xvzf arm-linux-gcc-4.3.2.tgz
```

Para llevar a cabo esta acción es necesario estar ubicados en la carpeta dónde está copiado el archivo.

- Una vez extraído se crea el directorio `/home/usuario/usr/local/arm/4.3.2/`

---

<sup>36</sup> GNU Binary Utilities es una colección de herramientas de programación para la manipulación de código objeto en varios formatos de archivos objeto.

<sup>37</sup> General Public License. Licencia que garantiza a los usuarios la libertad de usar, compartir (copiar) y modificar el software. Cualquier software que posea esta licencia es software libre.

- Se exporta el PATH cada vez que se vaya a ejecutar el compilador con el siguiente comando (aunque también se puede hacer modificando la opción bashrc):

```
# export PATH=$PATH:/home/"usuario"/compilador/usr/local/arm/4.3.2/bin
```

- Hecho esto, se puede comprobar la versión del compilador instalado. Si la instalación está hecha correctamente el usuario verá la versión utilizada, si no, es porque algo quedó mal hecho. El siguiente comando muestra la versión del compilador:

```
# arm-linux-gcc -v
```

Hecho los pasos anteriores, ya será posible compilar el programa.

**Figura 15.** Verificación de la instalación del compilador cruzado GCC

```

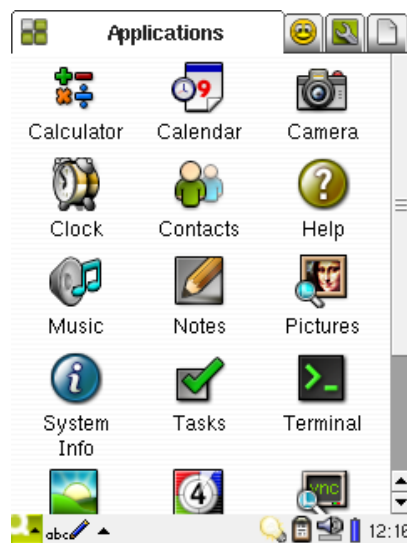
fernando@linux-29fk:~> su -l
Contraseña:
linux-29fk:~ # export PATH=$PATH:/home/fernando/Descargas/Compilador/usr/local/arm/4.3.2/bin
linux-29fk:~ # arm-linux-gcc -v
Using built-in specs.
Target: arm-none-linux-gnueabi
Configured with: /scratch/julian/lite-respin/linux/src/gcc-4.3/configure --build=i686-pc-linux-gnu --host=i686-pc-linux-gnu --target=arm-none-linux-gnueabi --enable-threads --disable-libmudflap --disable-libssp --disable-libstdcxx-pch --with-gnu-as --with-gnu-ld --enable-languages=c,c++ --enable-shared --enable-symvers=gnu --enable-cxa_atexit --with-pkgversion='Sourcery G++ Lite 2008q3-72' --with-bugurl=https://support.codesourcery.com/GNUToolchain/ --disable-nls --prefix=/opt/codesourcery --with-sysroot=/opt/codesourcery/arm-none-linux-gnueabi/libc --with-build-sysroot=/scratch/julian/lite-respin/linux/install/arm-none-linux-gnueabi/libc --with-gmp=/scratch/julian/lite-respin/linux/obj/host-libs-2008q3-72-arm-none-linux-gnueabi-i686-pc-linux-gnu/usr --with-mpfr=/scratch/julian/lite-respin/linux/obj/host-libs-2008q3-72-arm-none-linux-gnueabi-i686-pc-linux-gnu/usr --disable-libgomp --enable-poison-system-directories --with-build-time-tools=/scratch/julian/lite-respin/linux/install/arm-none-linux-gnueabi/bin --with-build-time-tools=/scratch/julian/lite-respin/linux/install/arm-none-linux-gnueabi/bin
Thread model: posix
gcc version 4.3.2 (Sourcery G++ Lite 2008q3-72)
linux-29fk:~ #

```

### 5.3.1.3. Qtopia

La tarjeta de desarrollo ARM mini2440 es capaz de soportar tres (3) sistemas operativos diferentes: Linux, Android y Windows CE. Para este caso se instaló una distribución de Linux llamada Qtopia en su versión 2.2.0 (hoy en día se conoce como Qt Extended), ya que se sabe que Linux es un sistema operativo muy potente, seguro y gratuito. Este sistema operativo es una plataforma de aplicaciones para dispositivos móviles y está desarrollado bajo la licencia GPL (Opie<sup>38</sup>).

**Figura 16.** Entorno gráfico Qtopia 2.2.0



Qtopia forma una capa de abstracción que trabaja por encima del kernel de Linux, proporcionando servicios como entorno gráfico basado en ventanas, pantalla de escritura manual, multimedia, videojuegos, sincronización con el PC, entre otros. El conjunto “kernel de Linux + Qtopia” se puede ver como un sistema operativo embebido con entorno gráfico.

### 5.3.1.4. Base de datos SQLite3

SQLite es una base de datos muy similar a la conocida Access en el sistema Windows, pero a diferencia de ésta, posee una serie de ventajas que la hacen interesante de aplicar:

---

<sup>38</sup> Open Palmtop Integrated Environment. Entorno de desarrollo creado a partir de Qt Extended, soportado en diferentes dispositivos móviles en los que se ejecute Linux.

- Multiplataforma, cumple con los estándares (en su mayoría) SQL92.
- Se puede usar en modo ventana de comandos (Shell) o embebido en aplicaciones de código.
- Motor que trabaja embebido con la aplicación que lo use (en el servidor, en programas propios escritos, etc.).
- Portable, la base de datos es el fichero y nada más.

El código para SQLite es de dominio público y, por lo tanto, libre de uso para cualquier propósito, ya sea comercial o privado. Actualmente SQLite se encuentra en gran variedad de aplicaciones, incluyendo proyectos de alto perfil. A diferencia de la mayoría de las otras bases de datos SQL, SQLite no tiene un proceso de servidor independiente.

SQLite lee y escribe directamente en archivos de disco ordinarios. Una base de datos completa de SQL con varias tablas, índices y vistas, está contenida en un archivo de disco único. Los programas que se enlacen con la librería SQLite pueden tener acceso a la base de datos SQL, sin tener que ejecutar un programa de sistema de gestión de base de datos relacional separado.

La base de datos SQL soportada por la board es la SQLite3. Este es un sistema de gestión de bases de datos relacionales. Es de dominio público, escrito en lenguaje C y muy ligero. Soporta diferentes plataformas como Windows, Linux y Mac OS. Este tipo de base de datos se le puede llamar empotrada, ya que no inician un servicio en una máquina independientemente de la aplicación, pudiéndose enlazar directamente al código fuente o bien utilizarse en forma de librería (`#include`), que es la manera como se utilizó en este caso.

Para llevar a cabo la instalación de SQLite3 en la board ARM mini2440, es necesario que se siga al pie de la letra los siguientes pasos:

- Descargamos el archivo `SQLITE3` del siguiente enlace: [http://www.modbus.pl/download/zxy66/v19/sqlite\\_fa.zip](http://www.modbus.pl/download/zxy66/v19/sqlite_fa.zip).
- El archivo descargado será un archivo comprimido, el cual contendrá las siguientes carpetas: `bin`, `include`, `lib`, `share`.
- Copiamos los archivos de la carpeta `"lib"` a la carpeta `"/usr/lib"` de la board ARM mini2440.
- Copiamos el archivo `'sqlite3'` de la carpeta `"bin"` a la carpeta `"/usr/bin"` de la board ARM mini2440.
- Se crean enlaces del archivo `'libsqlite3.so.0.8.6'` con el siguiente comando:

```
In -s /usr/lib/libsqlite3.so.0.8.6 /lib/libsqlite3.so.0
In -s /usr/lib/libsqlite3.so.0.8.6 /lib/libsqlite3.so
```

Si se quiere usar la base de datos SQLite para la board ARM en aplicaciones con lenguaje C/C++ se debe incluir la librería en la cabecera del proyecto (`#include sqlite3.h`) y en la línea de comandos, cuando se haga la compilación del archivo, se debe agregar la opción `-lsqlite3`. Esta instalación se debe realizar desde un sistema operativo Linux.

### 5.3.1.5. Eclipse IDE

Entorno de desarrollo integrado (IDE), de código abierto y multiplataforma. Es una completa plataforma de programación, desarrollo y compilación de elementos tan variados como sitios web, programas en C++ o aplicaciones Java. Hasta hace poco era la plataforma ideal para los desarrolladores de Android, ya que hoy en día se pueden encontrar entornos de desarrollo especializados como Android Studio o Netbeans, sin embargo, Eclipse continúa siendo el IDE más completo.

En esta ocasión, Eclipse se utilizó para acoplar la aplicación web en un teléfono con tecnología Android insertando un widget en el linear layout, proporcionado por el SDK de Android, llamado clase WebView. Esta clase lo que hace es insertar un widget en el espacio de trabajo, permitiendo desplegar el navegador dentro de la aplicación. Para hacer uso de esta clase es necesario usar el paquete del navegador llamado webkit, ya que se van a importar algunos paquetes, aunque éste viene incluido en el SDK de Android.

### 5.3.2. Proceso de desarrollo

Una vez instalado el software necesario, se está en condiciones de llevar a cabo el desarrollo de la aplicación propiamente dicho. La aplicación consta de varios archivos, los cuales son compilados a una extensión ejecutable válida por la arquitectura ARM. Estos archivos estarán alojados en la carpeta del servidor web de la mini2440, que para este caso será la carpeta “www”. Esta carpeta deberá ser creada por el usuario y debe, obligatoriamente, tener ese nombre. El servidor web por defecto de la board es BOA<sup>39</sup> server.

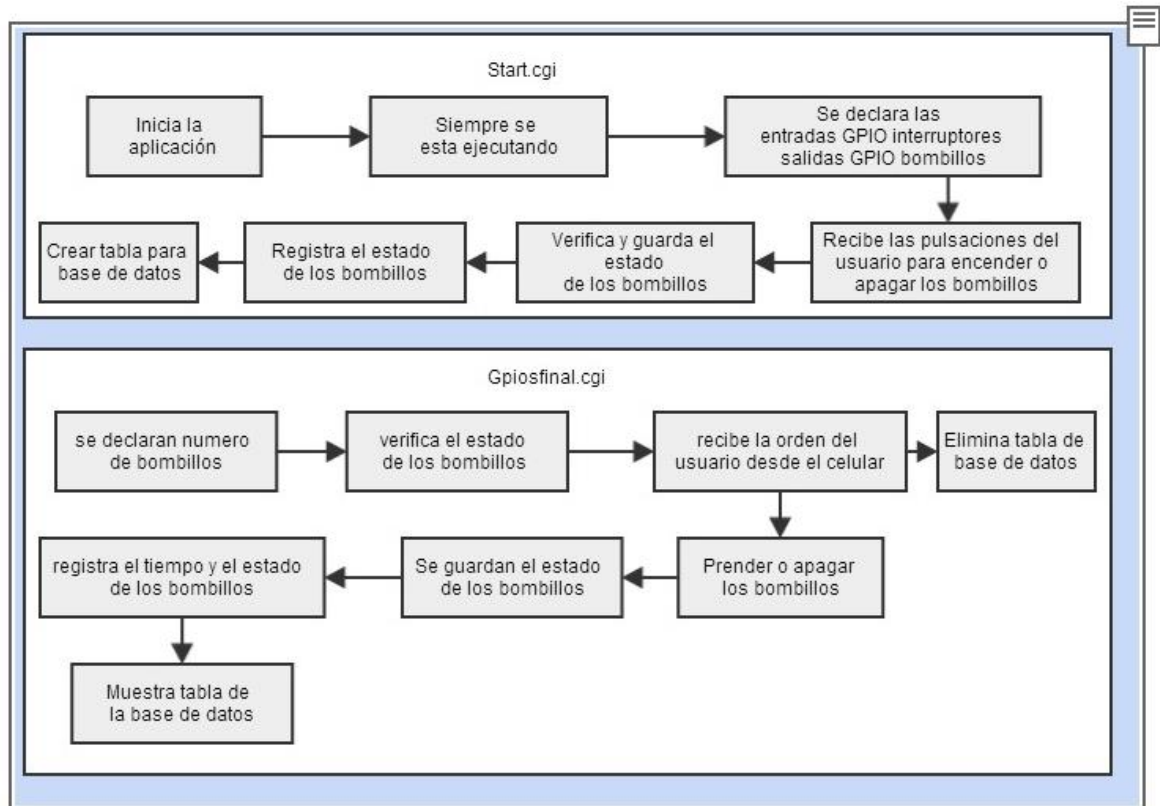
Debido a que se desea controlar los bombillos de una casa y al mismo tiempo llevar los registros de encendido y apagado de éstos en una base de datos, se realizó código por separado, generando archivos para cada una de las funciones.

---

<sup>39</sup> Servidor web de código abierto. Ocupa poco espacio haciéndolo ideal para aplicaciones embebidas. Está bajo la licencia GPL.



**Figura 17.** Diagrama de arquitectura de desarrollo del sistema

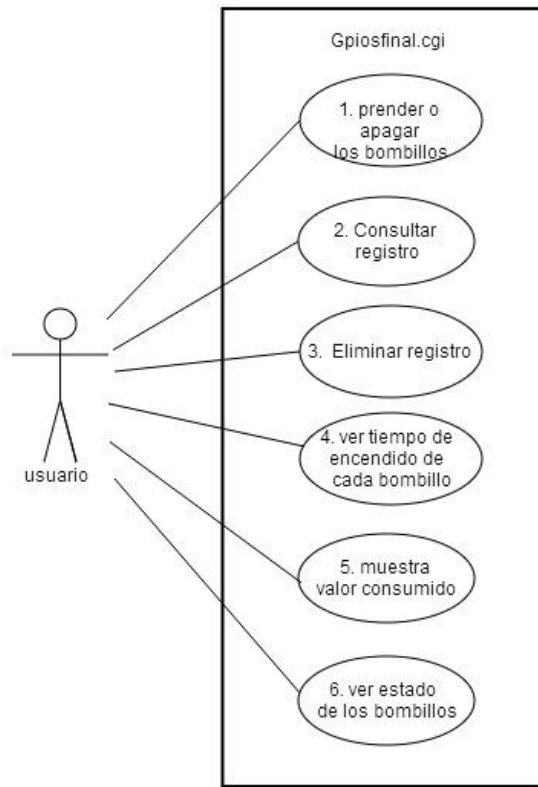


La función `gpiosfinal.cgi` consulta el estado de cada bombillo y recibe las ordenes que el usuario desea realizar desde el celular, si desea prender o apagar un bombillo este verifica el estado y realiza la acción correspondiente guardando el estado de cada bombillo, esta función también se encarga de mostrar la tabla que contiene la base de datos de la aplicación y de eliminar la misma.

El programa `gpiosfinal.cgi` es el encargado de manejar las acciones que realiza el usuario con la aplicación desde el celular.

Para que el servidor pueda recibir solicitudes y enviar su respectiva respuesta, se hace uso de una técnica de desarrollo web llamada AJAX. Esta técnica lo que hace es enviar una solicitud de usuario al servidor en segundo plano, siendo invisible para el usuario, mientras se mantiene la comunicación asíncrona con el servidor. Además del AJAX, se hace uso de un formato ligero para el intercambio de datos llamado JSON.

**Figura 18.** Caso de usos del programa gpiosfinal.cgi



AJAX envía una solicitud, debido a que si el cliente hace una petición el servidor debe devolver una respuesta, para este caso se solicita el tiempo de encendido de todos los bombillos. El servidor debe dar una respuesta a dicha solicitud y la respuesta la envía en formato JSON. Este envío y recepción de datos entre diferentes lenguajes de programación se hace más sencillo gracias a JSON.

Además del AJAX y JSON, se utilizó JQUERY. Este es un plugin JavaScript que facilita algunas funciones JavaScript, dando la posibilidad de escribir menos código pero programar más. Anteriormente para hacer algo se necesitaban de muchas líneas de código JavaScript, pero con JQUERY esto se reduce.

Resumiendo, JQUERY facilita la programación JavaScript y permite acceder a los recursos de html de una manera más cómoda. El JQUERY se utilizó para hacer uso de AJAX y poder administrar fácilmente la información JSON que devuelve el servidor. En los archivos JavaScript, donde aparezca el símbolo "\$" es porque se está haciendo uso de JQUERY.

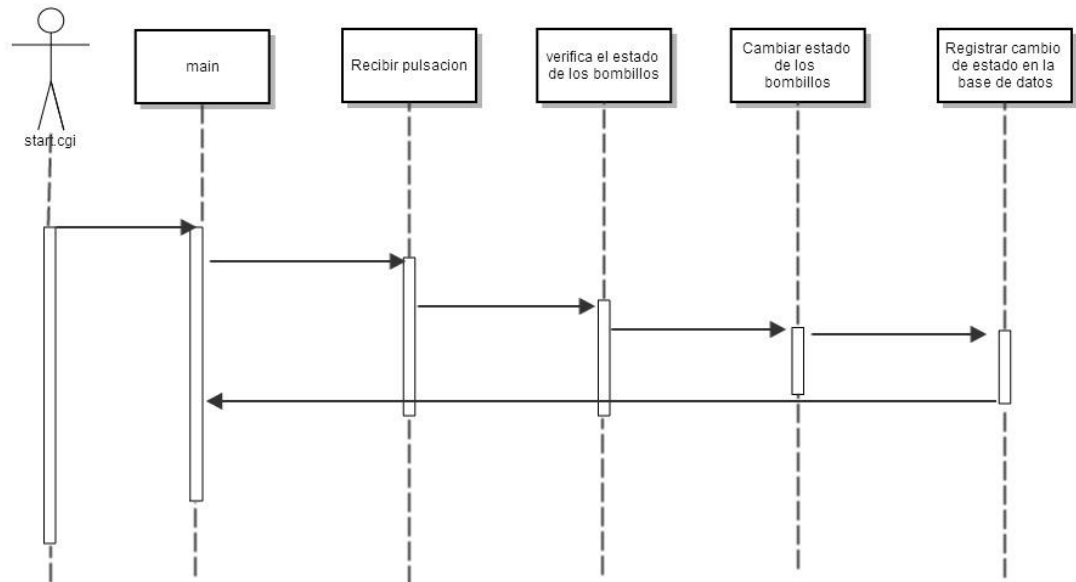
La función start.cgi se encarga de iniciar la aplicación. Esta función siempre está en ejecución, además, recibe las pulsaciones que realiza el usuario. Verifica en qué

estado se encuentran los bombillos y realiza la acción correspondiente, seguidamente, guarda el estado de los bombillos y los registra en una tabla. Dicha tabla será la base de datos almacenada en el servidor.

Este programa, start.cgi, se ejecuta desde el momento en que se inicia la tarjeta y es el encargado de iniciar la aplicación. Si este archivo no es ejecutado, el sistema no funcionará. Para entender el funcionamiento de este programa se realizó el siguiente diagrama de secuencias.

Dicho archivo está en el mismo script en donde se guardó el protocolo de red OpenSSH, ya que ambos se necesitan ejecutar al momento de encender la tarjeta.

**Figura 19.** Diagrama de secuencias del programa start.cgi



Estas funciones están acompañadas por librerías, las cuales se agregan a la cabecera del programa para que se tengan en cuenta al momento de su ejecución. Dichas librerías han sido creadas por desarrolladores a nivel mundial y lo único que se hizo fue hacer un llamado a la librería ya creada. Aunque no todas las librerías han sido descargas, pues existen unas librerías JavaScript, las cuales se crearon para manejar diferentes funciones y simplemente hacer un llamado de éstas.

### 5.3.3. Aplicación web

Las aplicaciones web son aplicaciones que corren sobre un navegador web, en este caso, sobre el navegador web del dispositivo móvil.

Para que el usuario visualice los bombillos que se encienden y apagan remotamente, además de la base de datos, se creó una aplicación web con lenguaje HTML5, la cual cuenta con una interfaz gráfica sencilla y de fácil uso. Dicha aplicación web se creó a partir del framework de LungoJS. Este framework proporciona las herramientas necesarias para la creación de páginas web para que sean visibles desde los diferentes dispositivos móviles (teléfonos, tabletas y PC). Actualmente existen variedad de framework para la creación de aplicaciones web, entre los más populares encontramos a Bootstrap, el cual es el framework con el que trabaja la famosa aplicación Twitter<sup>40</sup>.

Además de utilizar el framework de Lungo, se utilizó una micro librería JavaScript llamada QuoJS, hecha por el mismo autor de Lungo, Javier Jiménez. Esta librería se encarga, a través de lenguaje de programación JavaScript, de enviar diferentes eventos al dispositivo móvil, ya sea para controlar el menú desplegable por medio de eventos de gestos con los dedos, hasta utilizar AJAX.

Como ya se mencionó, la aplicación web fue escrita en HTML5, pero dado que se debe visualizar en un teléfono móvil con sistema operativo Android, fue necesario utilizar el entorno de desarrollo Eclipse para acoplar la aplicación y generar una aplicación ejecutable para Android. Para Lograr esto, el SDK de Android proporciona una clase denominada WebView para acoplar páginas web, ya sea obteniéndolas a través de una URL o bien recibiendo el html directamente desde una Activity.

**Figura 20.** Logo aplicación en dispositivo Android



---

<sup>40</sup> Aplicación web gratuita mediante la cual se pueden enviar mensajes, no mayor a 140 caracteres, en tiempo real a personas de su interés.

Para acoplar la aplicación creada en HTML5, se debe crear una nueva aplicación para Android. Se le da el nombre a la aplicación y se define la API sobre la cual se desea que funcione la aplicación, para este caso se decidió que la aplicación funcione mínimamente con la versión 4.0 denominada Ice Cream Sandwich (API 14); esto ya que según un estudio, los usuarios, en su gran mayoría, ya utilizan esta versión de Android y es la que menos errores de compilación genera, pudiendo la aplicación llegar a más usuarios sin casi ningún tipo de problema.

Una vez seleccionado esto y seguido los pasos, aparecerá una ventana donde se observa cómo va quedando la aplicación y un pequeño letrero donde dice "Hello World". Se elimina dicho letrero, dejando el espacio limpio y se selecciona el widget de WebView para que ocupe toda la ventana donde se está creando la aplicación.

Luego de esto, se va al manifiesto y se seleccionan los permisos que se le quiere dar a la aplicación, para este caso se eligió solo permiso de internet. En el manifiesto de la aplicación se introduce la URL a la cual se desea acceder, en este caso será la del servidor de la board ARM mini2440: 192.168.1.230. Se escogen algunos parámetros de configuración de la WebView, como por ejemplo no dejar que se haga zoom a la aplicación, evitar los botones de navegación, etc. Y finalmente se compila y se ejecuta para crear la aplicación web.

## 6. IMPLEMENTACIÓN

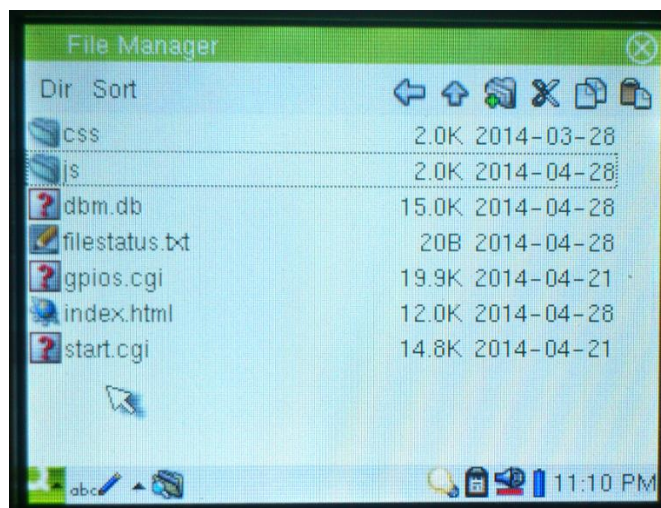
Ya habiendo hecho el diseño y dejando claro las características a tener en cuenta, se procede a hacer su respectiva implementación. Se debe tener en cuenta que la implementación del sistema se hizo en una casa a escala (1:10), usando la red eléctrica de 110V.

La carpeta del servidor web, denominada “www” deberá contener los siguientes archivos y librerías:

- Carpeta CSS de Lungo
  - unido.css → Librerías de Lungo unida en una sola
- Carpeta JS librerías JavaScript
  - app.js → Uso QuoJS y mensajes alerta
  - jquery-1.11.0.min.js → Facilita funciones de JavaScript
  - lungo.js → Librería Lungo
  - quo.js → Librería Quo
  - tiempo.js → Comprueba cada “x” tiempo estado de bombillos
- dbm.db → Base de datos de SQLite
- filestatus.txt → Pruebas de estado
- gpios.cgi → Consultado estado bombillos y envía ordenes
- index.html → Interfaz gráfica aplicación web
- start.cgi → Verifica estado actual bombillos y registra en tabla

El código donde se crearon los archivos se puede encontrar en el CD.

**Figura 21.** Archivos carpeta “www” del servidor web



Dado que los archivos en su mayoría fueron escritos en lenguaje de programación C++, se debe utilizar el compilador cruzado GCC para convertir dichos archivos a una extensión entendible por la máquina. Para ello se utilizó el siguiente comando:

```
arm-linux-g++ gpio.cpp gpiosfinal.cpp -l sqlite3 -o gpios.cgi  
arm-linux-g++ gpio.cpp start.cpp -l sqlite3 -o start.cgi
```

Estos comandos se deben ejecutar desde la consola de Linux, pero anteriormente el compilador cruzado debe haber exportado el PATH, si no dará error.

Lo que están diciendo estos archivos básicamente es que se van a tomar unas funciones escritas en C++, se definen qué funciones son, se agrega el sqlite3 para que se pueda ejecutar la base de datos y todo eso lo convierte a un archivo de salida con extensión .cgi.

Cabe aclarar que para que funcione el sqlite3 se deben instalar las carpetas del sqlite3 en su respectiva posición dentro de las carpetas donde se encuentra el compilador cruzado. Estos pasos no son específicos ya que toca copiar los archivos hasta que el compilador funcione correctamente, si no se realiza esta acción no compilará bien el archivo y se tendrán problemas.

Gracias a la extensión .cgi el usuario puede solicitar los datos que se están ejecutando en el servidor web. Éste define un estándar para transferir datos entre el cliente y el programa. Toda aplicación que se ejecute en el servidor debe contener la extensión .cgi, a menos que sea una librería.

Compilado los archivos y almacenados en el servidor, se realiza la red entre la board y el teléfono móvil, para que el cliente pueda enviar las peticiones y lograr encender los bombillos.

Los pines del puerto GPIO pueden ser usados tanto de entrada como de salida. Para este caso se definió unos pines de salida (para encender y apagar los bombillos a través del dispositivo móvil) y otros de entrada (para encender y apagar los bombillos a través del pulsador).

Para ser usado como salida se definió los siguientes pines:

- CON4\_PIN9
- CON4\_PIN13
- CON4\_PIN17
- CON4\_PIN21
- CON4\_PIN25

Para ser usado como entrada se definió los siguientes pines:

- CON4\_PIN12
- CON4\_PIN16
- CON4\_PIN20
- CON4\_PIN24
- CON4\_PIN28

Se dispuso así, de forma intercalada, dado que no se consiguió un bus para acoplar los pines del puerto GPIO. En vez de eso, a cada pin se le adecuó un pin de una regleta para lograr tener un poco más largo los pines del puerto GPIO. El pin 3 será la tierra común para los pulsadores.

Para realizar la red se utilizó un nano router inalámbrico de la familia TP-Link de referencia TL-WR702N. Se eligió este router dada su facilidad de uso. Puede alcanzar velocidades de datos inalámbricos de hasta 150Mbps y su conexión a la energía se puede dar ya sea por conexión USB hasta un ordenador o por medio de un adaptador de energía externo.

**Figura 22.** Nano router TP-Link TL-WR702N



Estando los archivos compilados y resguardados en el servidor de la board, y poniendo en red el teléfono móvil con la board vía wifi, se ejecutan las órdenes por parte del cliente, ya sea para encender o apagar los bombillos, de modo manual a través del pulsador o de la aplicación móvil.

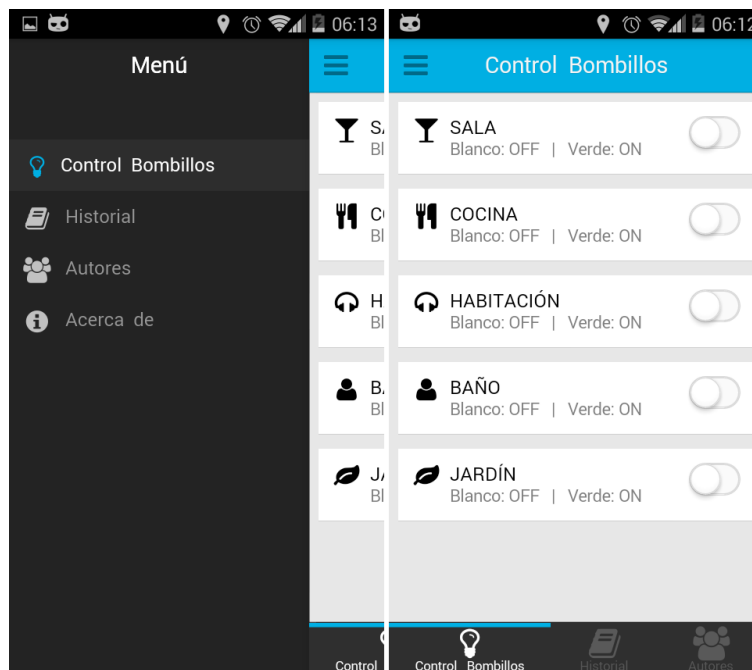


## 7. RESULTADOS

Recopilando todo lo anteriormente mencionado, se consiguió construir un sistema capaz de controlar y monitorear remotamente el sistema de luces de una vivienda, a través de la board ARM mini2440, y cuya interfaz gráfica puede ser vista desde cualquier dispositivo móvil que cuente con un navegador web.

El usuario puede tener un control absoluto sobre los bombillos, pudiendo encenderlos o apagarlos ya sea de forma manual o remotamente, de todos modos, de cualquiera de las dos formas el cambio será registrado en la base de datos y podrá consultarla cada vez que quiera hacerlo. Además, la aplicación web cuenta con un botón de reset, el cual permite al usuario eliminar la tabla de la base de datos y empezar nuevamente a tomar el control del consumo de energía.

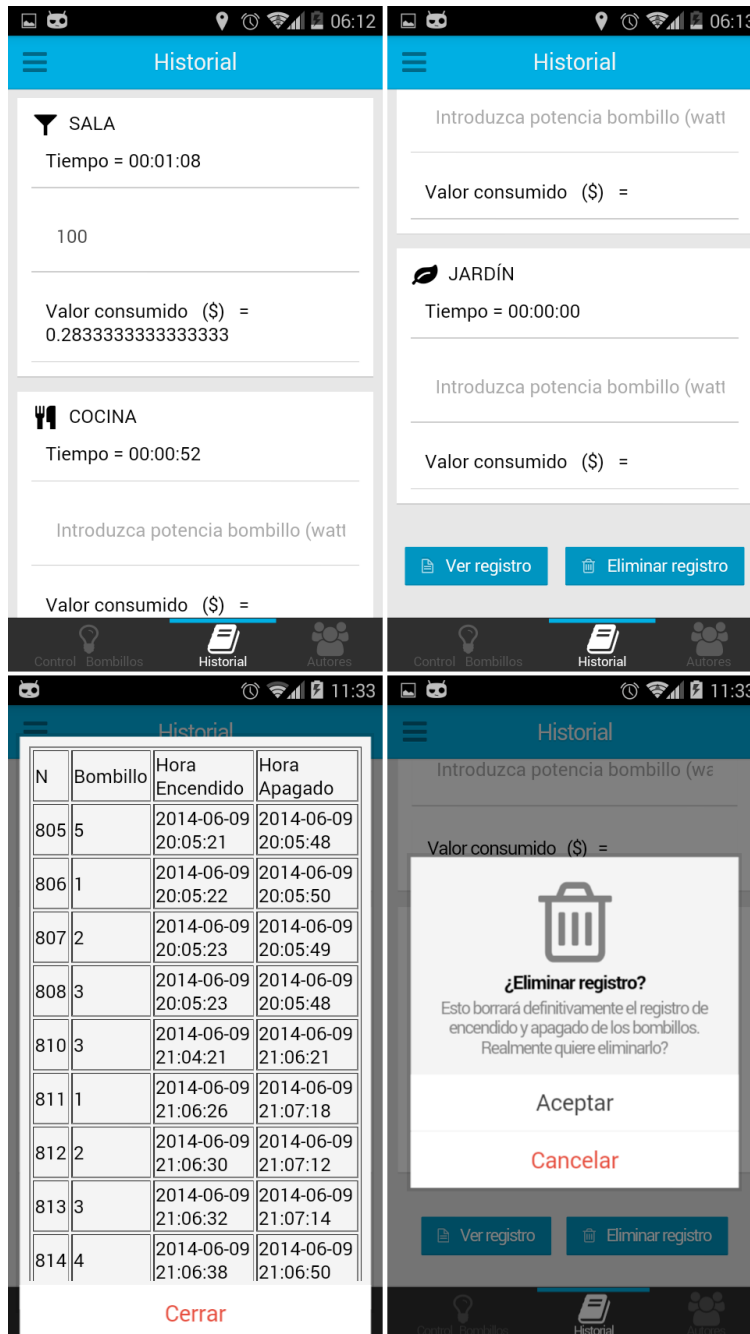
**Figura 23.** Aplicación web Control Bombillos y Menú desplegable



La aplicación web consta de tres (3) opciones de menú: Control Bombillos, el cual muestra el lugar dónde se llevará a cabo el control de encendido/apagado de los bombillos, haciendo dicho control a través de un botón deslizable de derecha a izquierda y viceversa; Historial, el cual es dónde se muestra el tiempo real que lleva encendido cada bombillo, da la opción de introducir la potencia del bombillo para arrojar el precio que se debe pagar hasta el momento por el consumo de ese

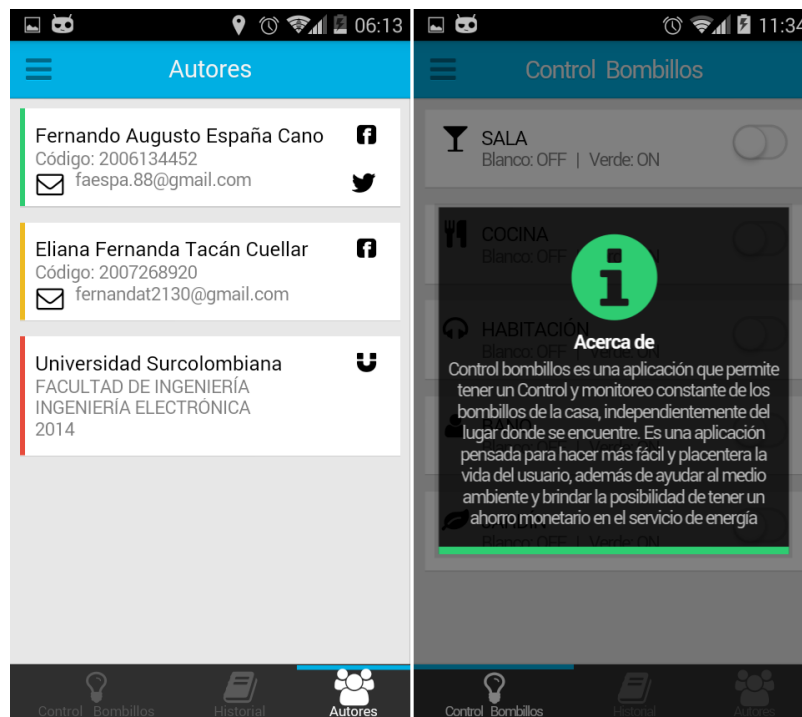
bombillo y aloja dos (2) botones para ver o eliminar la tabla de la base de datos; y Autores, que es donde se muestra la información de los desarrolladores del proyecto, incluyendo su página oficial a las redes sociales como Facebook y Twitter.

**Figura 24.** Aplicación web Historial



Además de contar con el menú en la parte inferior, la aplicación utiliza un menú desplegable hacia la derecha, brindando una opción más que las anteriormente mencionadas: Acerca de. En esta opción aparece una ventana en donde se habla acerca de la aplicación. Se puede abrir en cualquier momento. El menú desplegable puede ser abierto de dos maneras: pulsando el ícono en la parte superior izquierda o deslizando el dedo hacia la derecha. Se cierra deslizando el dedo a la izquierda o presionando de nuevo el ícono.

**Figura 25.** Aplicación web Autores y mensaje Acerca de



Dado que se realizó el trabajo en una board embebida de baja potencia, es necesario y de vital importancia no llenar el servidor con tanta información para no llegar a sufrir colapsos. Por ello, se creó un botón de “Eliminar Registro” el cual está ubicado en la parte inferior de la sección “Historial”. Esto limpia completamente la tabla de registros, pero el contador sigue llevando su cuenta, lo que no permitirá que se pierda el tiempo que lleva encendido cada bombillo.

Una de las ventajas que tiene esta aplicación es que puede ser abierta al tiempo por cinco (5) teléfonos diferentes y en todos se verá el cambio casi que instantáneo de cuando alguna persona encienda o apague un bombillo. Se dice “casi” ya que, como no es un sistema que funciona en tiempo real, tiene un pequeño retardo de 3 segundos. Esto no se vio como una desventaja, debido a que se necesita de ese retardo para que las peticiones sean bien analizadas por parte del servidor. El conteo del tiempo también se verá en todos los equipos que estén conectados. Se recomienda que no sean más de cinco (5) ya que si se conectan más se puede llegar a tener un colapso del sistema por parte del servidor.

Cada vez que se enciende un bombillo, éste es registrado en una tabla, la cual muestra el número del bombillo que se ha encendido (1 es el de arriba, 5 es el de más abajo), la hora de encendido y hora de apagado, junto con la fecha actual. Ya que en la sección Historial se muestra el tiempo “real” que lleva encendido cada bombillo, este tiempo “real” solo es mostrado realmente cuando el bombillo es apagado, haciendo una resta entre el tiempo de apagado del bombillo contra el tiempo de inicio y arrojando un valor real. Pueda que en algunas ocasiones el tiempo se vea corriendo muy rápido, esto es debido al sistema de trabajo de la board, y es precisamente por esto que se creó esta regla, para evitar este tipo de inconvenientes.

**Figura 26.** Tabla de registros de encendido/apagado de los bombillos

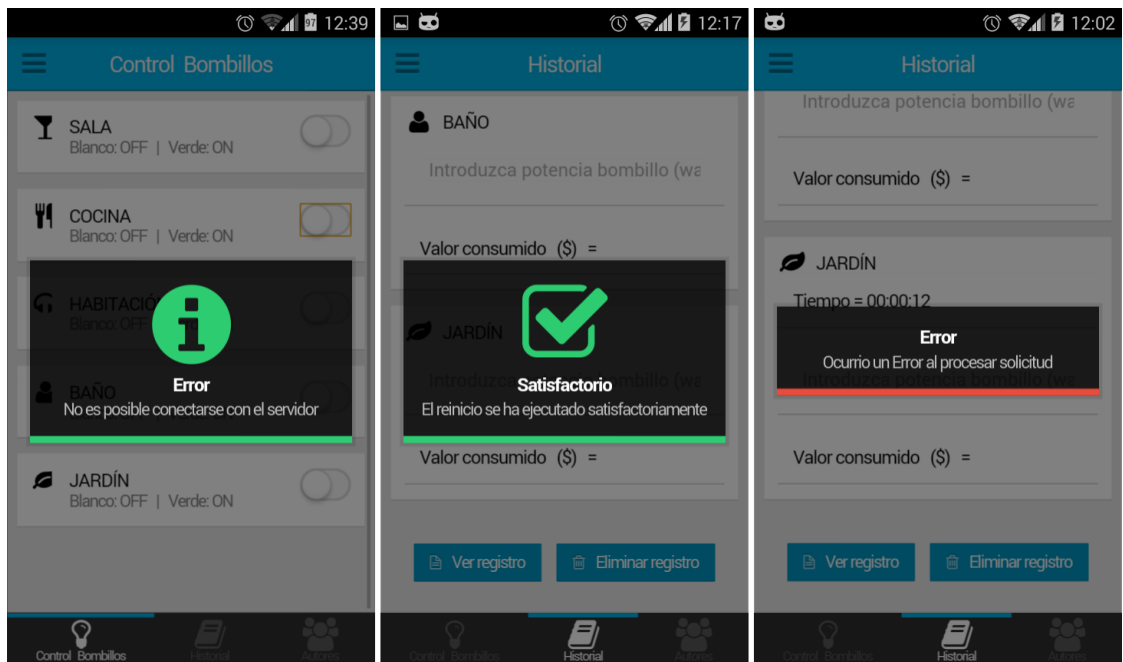
N	Bombillo	Hora Encendido	Hora Apagado
816	3	2014-06-10 20:14:23	(null)
817	5	2014-06-10 20:14:24	(null)

N	Bombillo	Hora Encendido	Hora Apagado
816	3	2014-06-10 20:14:23	2014-06-10 20:14:40
817	5	2014-06-10 20:14:24	2014-06-10 20:14:40

En el momento en que la tabla muestre “null” significa que el bombillo aún no se ha apagado, por consiguiente no mostrará la hora de apagado ni el tiempo real que éste lleva encendido. Solo hasta que éste se apague se registrará en la tabla. En caso de que la energía se vaya, el sistema apagará automáticamente los bombillos y al momento en que el usuario intente encender uno de estos, se desplegará un mensaje diciendo que no es posible conectar con el servidor. Si el usuario desea eliminar la tabla de registros, tampoco podrá, y en vez de que le salga un mensaje satisfactorio, aparecerá un mensaje diciendo que no es posible procesar dicha solicitud. Al momento en que la energía regrese, los bombillos volverán a su estado anterior, dejando encendidos los que estaban encendidos.

**Figura 27.** Mensajes cuando se va la energía y al borrar bien el registro



Se construyó una caja metálica en donde se introdujo la board ARM mini2440 y la etapa de potencia. Esto con el fin de facilitar las conexiones, tanto de los bombillos como entre la etapa de potencia y la tarjeta. Esta caja metálica tiene unas medidas de 15x15x10 Cm. Se adecuó un ventilador de 12V el cual va conectado directamente a la board a través de un cable USB. En la parte de arriba se encuentra la pantalla táctil de la board, la cual es protegida por un acrílico transparente, esto con el fin de que el usuario o cualquier otra persona no pueda manipularla directamente. En el frente de la caja se encuentran los conectores, tanto de los bombillos como el de la corriente para 110V.

**Figura 28.** Caja metálica para la board ARM mini2440 y la etapa de potencia



Antes de acoplar el circuito completo, se creó una etapa de potencia de prueba para observar si el sistema lograba adaptarse perfectamente a ésta. Los resultados fueron satisfactorios y se implementó la etapa de potencia probada para los cinco (5) bombillos, una para cada bombillo, trabajando de forma independiente.

**Figura 29.** Circuito acoplado probando la etapa de potencia

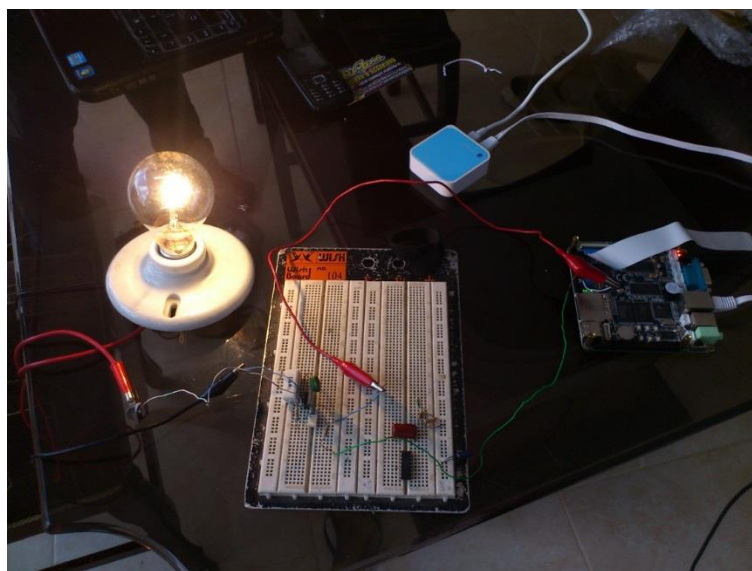




Figura 30. Acople de todo el sistema



## 8. PROBLEMAS Y LIMITACIONES

Uno de los grandes problemas presentes es la poca información que se tiene con respecto a la board ARM mini2440. La escases de manuales y material explicativo hace bastante difícil la tarea de trabajar con un elemento tan completo como los es esta tarjeta embebida. Los pocos manuales que hay dicen muy poco de la board y las pocas personas que se han atrevido a desarrollar en este sistema embebido están en el viejo continente, haciendo bastante difícil la traducción de manuales.

Por alguna razón, dado que no se pudo identificar el problema, la hora de la board no coincide con la hora que se registra en la base de datos, teniendo un adelanto de 6 horas aproximadamente.

El sistema sólo funciona mientras este en red entre la board y el teléfono móvil. Por ello no será posible encender o apagar los bombillos desde otra parte distinta a la red a la que estén conectados los dispositivos.

En caso de que se vaya la energía eléctrica, habrá pérdida de información, debido a que el sistema no alcanza a registrar los cambios en el servidor.

El framework LungoJS utilizado para crear la parte gráfica de la aplicación web aún está muy reciente y presenta constantes fallos.

Solo es posible instalar el kernel y sistema operativo de la board en un computador con sistema operativo Windows XP. Si se utiliza máquina virtual no va a funcionar, ya que la board utiliza los controladores de Windows XP. Hay una opción un poco más sencilla pero sería a través del sistema operativo Linux, para las personas que se les facilite manejar este tipo de SO.



## 9. RECOMENDACIONES Y SUGERENCIAS

Dado que la board ARM mini2440 es un sistema embebido de baja potencia puede llegar a presentar temperaturas altas. Por eso es recomendable que se tenga en un lugar ventilado, libre de objetos que le puedan poner alrededor y lejos de paredes caliente o los rayos del sol.

El lugar de instalación debe contar con un sistema de puesta a tierra, capaz de garantizar la integridad de los dispositivos presentes.

Se recomienda no instalar más de 5 bombillos a la board, pues puede llegar a provocar saturación de información, dado que es un sistema con muy poca memoria de almacenamiento.

No instalar la aplicación a niños. Esto debido a que pueden empezar a jugar a encender y apagar los bombillos, volviendo inestable el sistema y presentando errores a la hora de leer las solicitudes del cliente. Además de que pueden llegar a fundir los bombillos.

Para el trabajo y desarrollo de aplicaciones con la tarjeta embebida se recomienda que las funciones se hagan en archivos aparte. Esto permitirá tener un mejor entendimiento del código que se está desarrollando, y en caso de fallas, poder detectarlas de manera más rápida y eficiente.

Para tener un total control remoto sobre los bombillos, se recomienda que el usuario que instale este sistema en su vivienda pida a la empresa de telefonía móvil a la que está afiliado una IP fija, esto con el fin de que la IP que utiliza el servidor se logre acoplar a la IP fija y pueda acceder desde cualquier parte del mundo sin ninguna restricción.

Para una futura mejora del sistema, la aplicación web deberá incluir el número de bombillos que el usuario desea controlar, además, dado que la board tiene bastantes interfaces para su uso, se logre implementar un sistema total de seguridad, teniendo sensores en puertas y ventanas, brindando seguridad total a la casa.

Se sugiere un sistema de respaldo en caso de ausencia de energía eléctrica.

## 10. CONCLUSIONES

- Se implementó satisfactoriamente el sistema de control y monitoreo remoto de los bombillos de una casa, gracias al uso de un sistema embebido como unidad central de procesamiento. La importancia de éste radica en la gran variedad de posibilidades que se pueden realizar utilizando una tarjeta de desarrollo como esta.
- Al cambiar el sistema operativo de la tarjeta ARM mini2440 por una distribución válida de Linux como lo es Qtopia, se logró que la tarjeta presentara las aplicaciones necesarias para que el usuario desarrollara de forma adecuada la aplicación que controla el sistema de luces del prototipo de vivienda.
- Se comprobó, mediante la instalación de OpenSSH, que sí es posible utilizar un protocolo distinto de comunicación de red al que viene en la board por defecto, permitiendo transmitir y recibir datos de manera segura.
- La creación de aplicaciones web con lenguaje de programación HTML5 facilita el acoplamiento con cualquier dispositivo móvil, dando la posibilidad de que la aplicación pueda ser vista desde diferentes dispositivos a través de su navegador web. Además, con variar algunos parámetros en el archivo hecho en HTML5 se puede construir una aplicación capaz de soportar diferentes sistemas operativos.
- La clase proporcionada por el SDK de Android denominada WebView facilita la creación de aplicaciones web a partir de una página web, dándole al usuario la posibilidad de programar en un lenguaje diferente al nativo (Java), brindando la gran ventaja de que pueda ser vista en diferentes teléfonos móviles con sistema operativo Android sin la necesidad de crear un código para cada dispositivo y habilitar, con variar algunos parámetros, el uso de JavaScript dentro de la página web.
- La base de datos admitida por la tarjeta embebida es SQLite3. A pesar de que se supone que es de bajo rendimiento, sirve perfectamente para aplicaciones que no tengan mucho manejo de datos.
- Los circuitos que ajustan las señales digitales a la red eléctrica permiten el desarrollo de proyectos con sistemas embebidos y dispositivos móviles aplicados a la automatización de elementos que funcionen con la red eléctrica.
- La utilidad del trabajo realizado no se restringe al uso único del control y monitoreo remoto de las luces de una vivienda, si no a un amplio rango de aplicaciones, ya que es adaptable a cualquier plataforma en la que sea

necesario el uso de técnicas como el encendido remoto de un automóvil, control de puertas de una vivienda, entre otros.

- La creación de aplicaciones capaces de ser vistas desde cualquier dispositivo móvil hace que las personas se interesen cada día más y más por el mundo tecnológico. Ahora es posible controlar cualquier dispositivo a distancia, siempre y cuando se tenga conexión remota a su servidor.
- El manejo de funciones en un mismo archivo hace de la programación un proceso más complejo debido a los diferentes lenguajes que se pueden llegar a utilizar. JQUERY agiliza bastante las sentencias utilizadas en JavaScript y permite el uso sencillo de AJAX, dando la posibilidad de que el servidor pueda dar respuestas rápidas y certeras.

## BIBLIOGRAFÍA

- [1] F. M. Zabalza, «Representación de variables eléctricas en el sistema embebido mini2440,» Pamplona, 2011, p. 9 y 10.
- [2] K. ANNAN, «Discurso inaugural de la primera fase de la WSIS,» Ginebra, 2003.
- [3] S. JUNESTRAND, X. PASSARET y D. VÁZQUEZ, de *Domótica y hogar digital*, Madrid, Paraninfo, S.A., 2005, pp. Págs. 4-7.
- [4] C. GÓMEZ MONT, *El desafío de los nuevos medios de comunicación en México*, México: AMIC-Diana, 1992.
- [5] J. CHAPARRO MENDIVELSO, *Domótica: La mutación de la vivienda*, Universidad de Barcelona: Scripta Nova - Revista electrónica de geografía y ciencias sociales, 2003.
- [6] O. I. Valderrama y J. C. Vesga, «Universidad Nacional Abierta y a Distancia UNAD,» Diciembre 2009. [En línea]. [Último acceso: 29 Abril 2014].
- [7] L. M. G. N. B. Betancourt, «IMPLEMENTACIÓN DE UN SISTEMA TELEOPERADO CON REALIMENTACIÓN VISUAL PARA EVASIÓN DE OBSTÁCULOS DE UN ROBOT MÓVIL,» Bogotá D.C., 2009.
- [8] L. A. Y. S. S. G. Sanpedro Herrera, «ANÁLISIS, DISEÑO E IMPLEMENTACIÓN DE UNA TARJETA DE DESARROLLO UTILIZANDO EL MÓDULO MINI2440,» Quito, 2012.
- [9] J. Donelson, *Building and Installing Software for the Mini2440*, 2008.
- [10] FriendlyARM, «FriendlyARM,» [En línea]. Available: <http://www.friendlyarm.net/>. [Último acceso: Abril 2014].
- [11] I. ARMWorks, «ARMWork,» [En línea]. Available: <http://andahammer.com/>. [Último acceso: Abril 2014].
- [12] I. ARMWorks, *Mini2440 Hardware Essentials*, United States, 2009.
- [13] I. ARMWorks, *Compiling and Running Programs With Linux and Qtopia 2.2.0*.

- [14] E. Aguilar, Pruebas de Sqlite en un sistema Linux - guía para iniciarse, 2006.
- [15] «Room Technology,» 5 Diciembre 2012. [En línea]. Available:  
<http://theroomtech.blogspot.com/>. [Último acceso: Febrero 2014].

# **ANEXOS**

## **Anexo A.** Manual de instalación Sistema Operativo Linux en Board ARM mini2440

Anteriormente se dijo que la board ARMI mini2440 soporta tres (3) SO diferentes: Linux, Android y Windows CE. En este caso se instaló una distribución de Linux, debido a que es un sistema operativo potente, seguro y gratuito. La distribución que se instalará en la mini2440 será QTOPIA 2.2.0

Los elementos necesarios para instalar el SO Qtopia en la mini2440 son:

- Un cable serial (USB-SERIAL).
- Una extensión de cable serial de conector tipo Hembra-Hembra.
- Un cable USB.
- Board ARM mini2440 conectado a su respectiva fuente de voltaje.

**Figura 31.** Elementos utilizados para la instalación de Qtopia



Es recomendable que el computador que se vaya a utilizar tenga instalado el sistema operativo Windows XP, puesto que los controladores vienen para ser trabajados desde este sistema operativo. No se recomienda hacer la instalación desde una máquina virtual, pues aunque está utilizando virtualmente un sistema operativo XP, internamente trabaja con los mismos controladores.

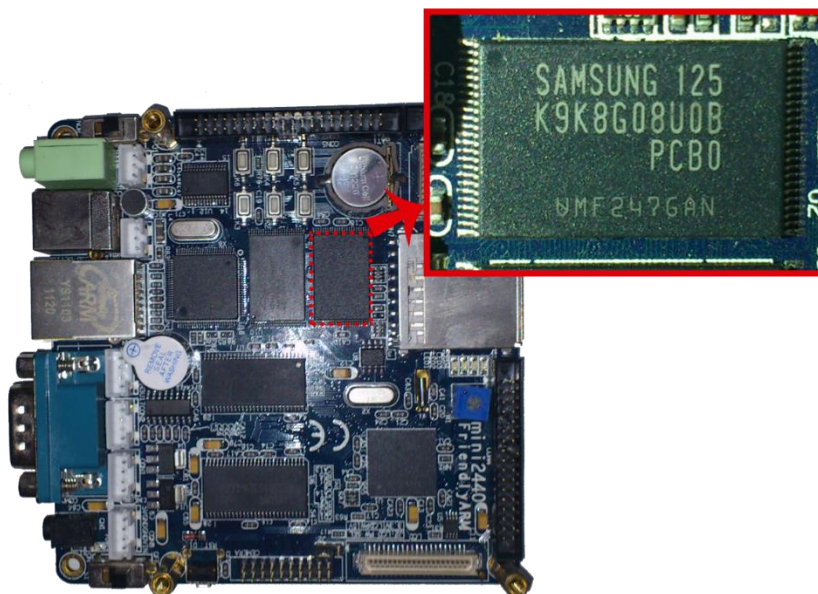
Ahora, se descargan los controladores para XP:

- Cable USB – SERIAL
- USB para la mini2440

Después de instalar los controladores es necesario tener los archivos DNW para XP. Este DNW es el programa que permite la transferencia de ficheros del PC a la mini2440.

Hay un archivo que no es el mismo para todas la board mini2440. Ese es el Supervivi. Para identificar cuál se tiene se debe mirar el chip de la memoria NAND y según su serie se identifica el Supervivi correspondiente:

**Figura 32.** Identificación Supervivi



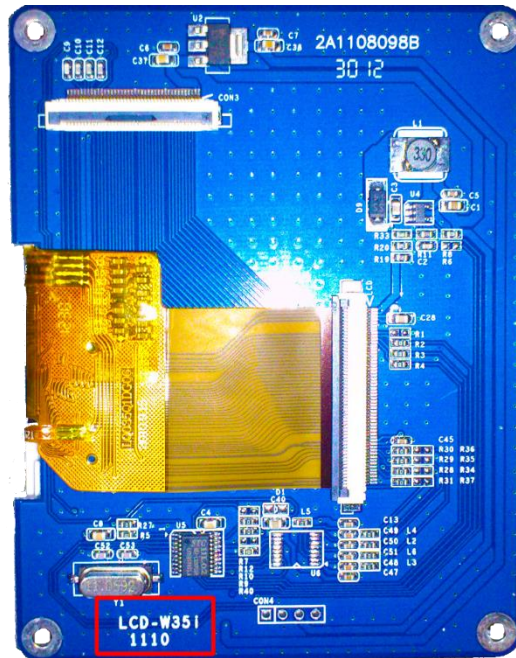
Los posibles números serie son:

- K9F1208 → 64MB NAND Flash
- K9F1G08 → 128MB NAND Flash
- K9F2G8 → 256MB NAND Flash
- K9k8G08 → 1024MB NAND Flash

El Supervivi de 64M es para el de 64MB NAND y el de 128M es para 128MB, 256MB y 1024MB.



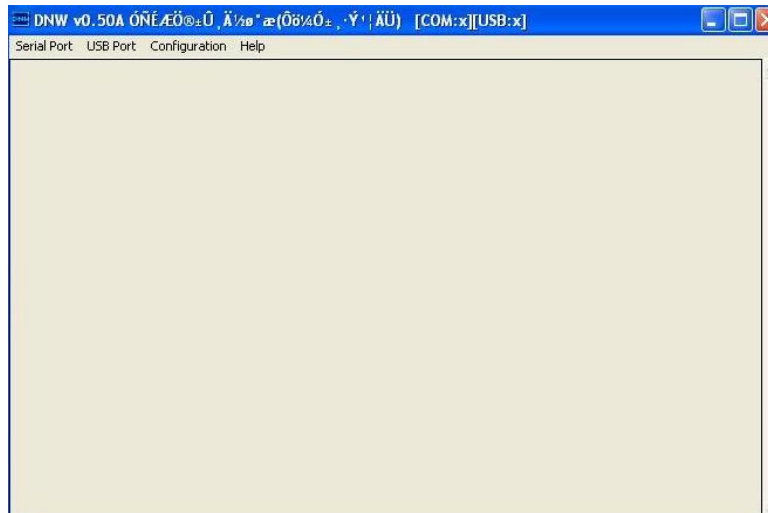
Para descargar el kernel de Linux se debe tener en cuenta el tipo de pantalla LCD que dispone la board ya que no todos tienen la misma pantalla. Para nuestro caso se dispone de una pantalla LCD-W35i.



Finalmente se descarga la imagen del sistema operativo Qtopia.

Se conecta los cables SERIAL y USB del computador a la ARM mini2440, para el cable serial se recomienda la extensión para que los conectores se correspondan sin problemas. Se ejecuta el archivo DNW.

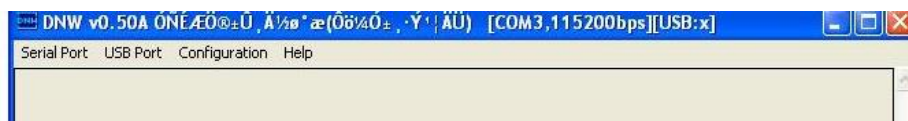
**Figura 33.** Interfaz DNW



Se va a CONFIGURATION → OPTIONS y se elige el puerto serial correspondiente, un Baud Rate=115200 y un Address del USB=0x30000000.

Se elige SERIAL PORT →CONNECT si todo está bien aparecerá en la barra de la ventana lo siguiente: [COM3, 115200].

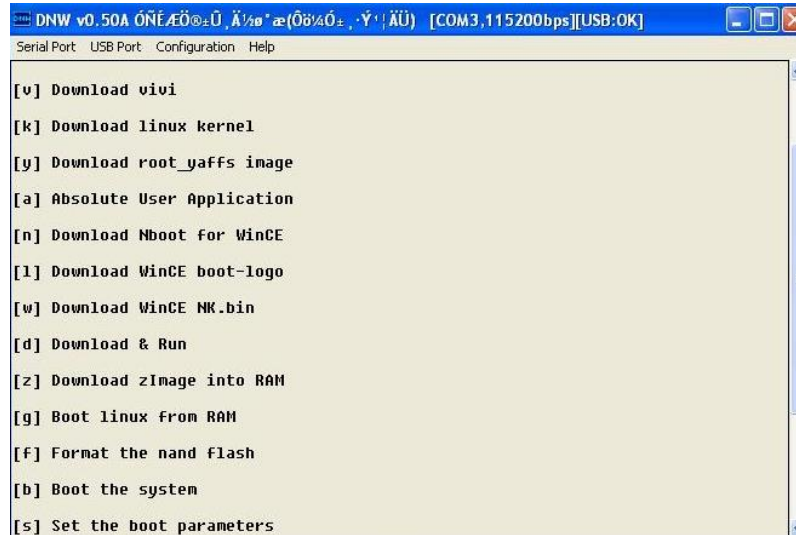
**Figura 34.** Forma correcta de conexión



Se ubica en NOR el switch para el modo del boot, esto mientras esta en OFF el switch de energía.

Se enciende la board mini2440 poniendo en ON el switch de energía. En el modo NOR el LCD está apagado, pero en la barra de estado del DNW, si todo está bien aparecerá [USB:OK]. Además se desplegara un menú con opciones, en algunas ocasiones este menú no se despliega pero no hay problema mientras la barra de estado muestre lo dicho anteriormente todo estará bien.

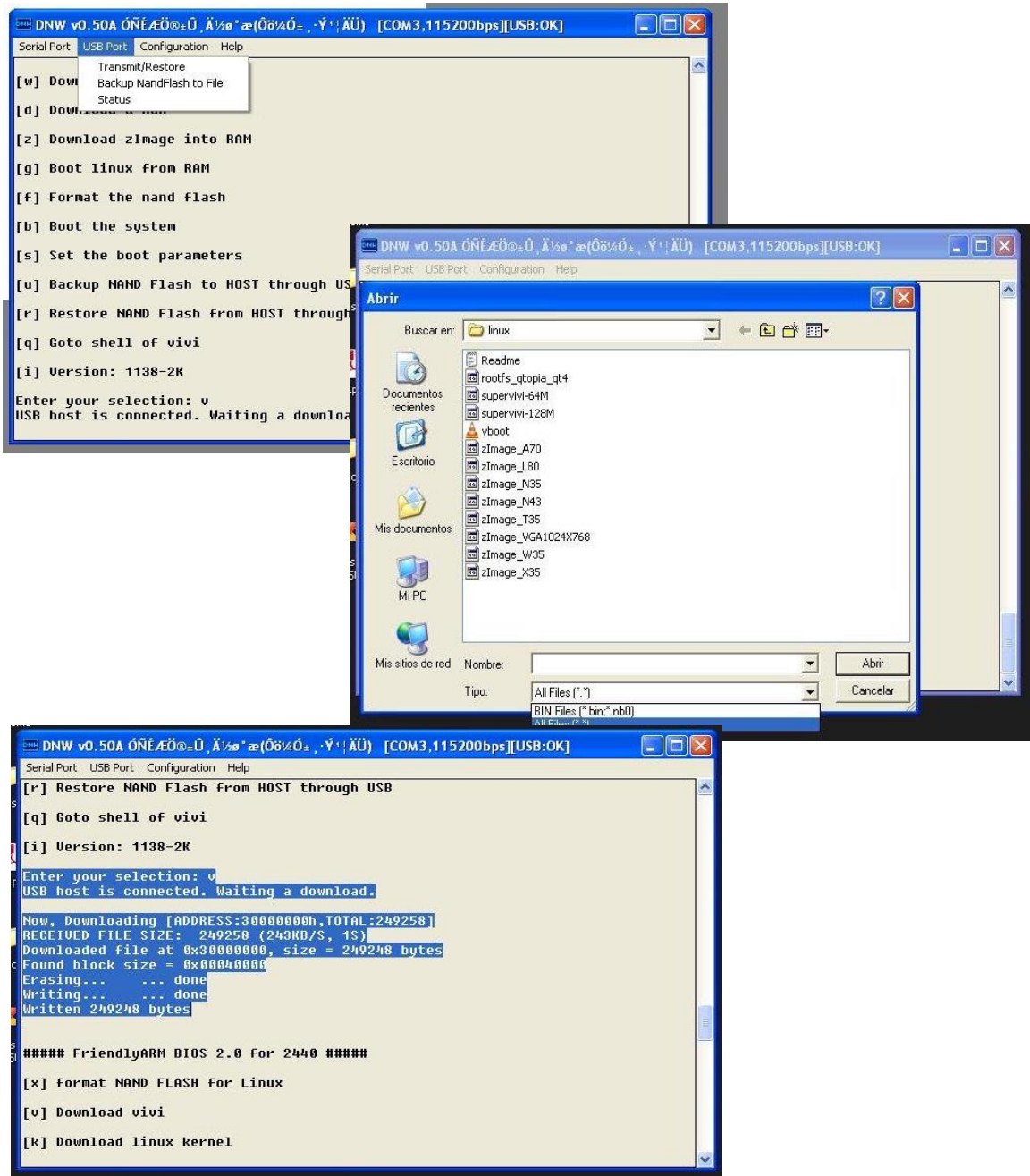
**Figura 35.** Conexión USB:OK



Se formatea la memoria NAND FLASH donde está el boot inicial para ello se presiona x.

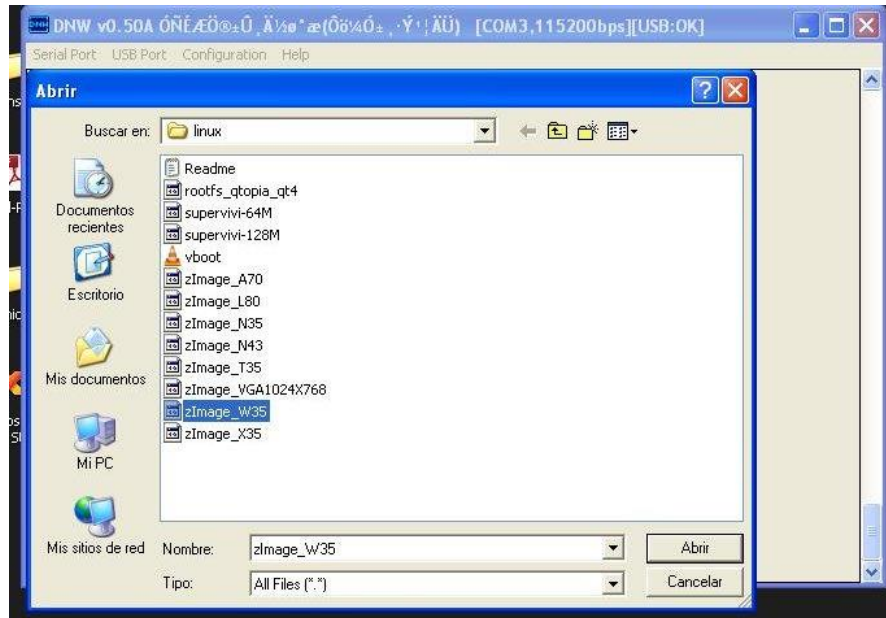
Se presiona v para subir a la ARM mini2440 el archivo Supervivi, después de esto nos vamos a USB Port → Transmit y se elige el Supevivi correspondiente; en este caso se utilizó supervivi-128M. En algunas ocasiones en el cuadro de dialogo no aparece el archivo que se necesita, para solucionarlo solo se selecciona ALL FILES en donde dice TYPE.

Figura 36. Forma de subir el Supervivi



Es hora de utilizar el kernel, se presiona *k* e inmediatamente USB Port → Transmit y elegimos el archivo imagen correspondiente a la pantalla que se tiene, en este caso LCD-W35i.

**Figura 37.** Elección del kernel para el LCD



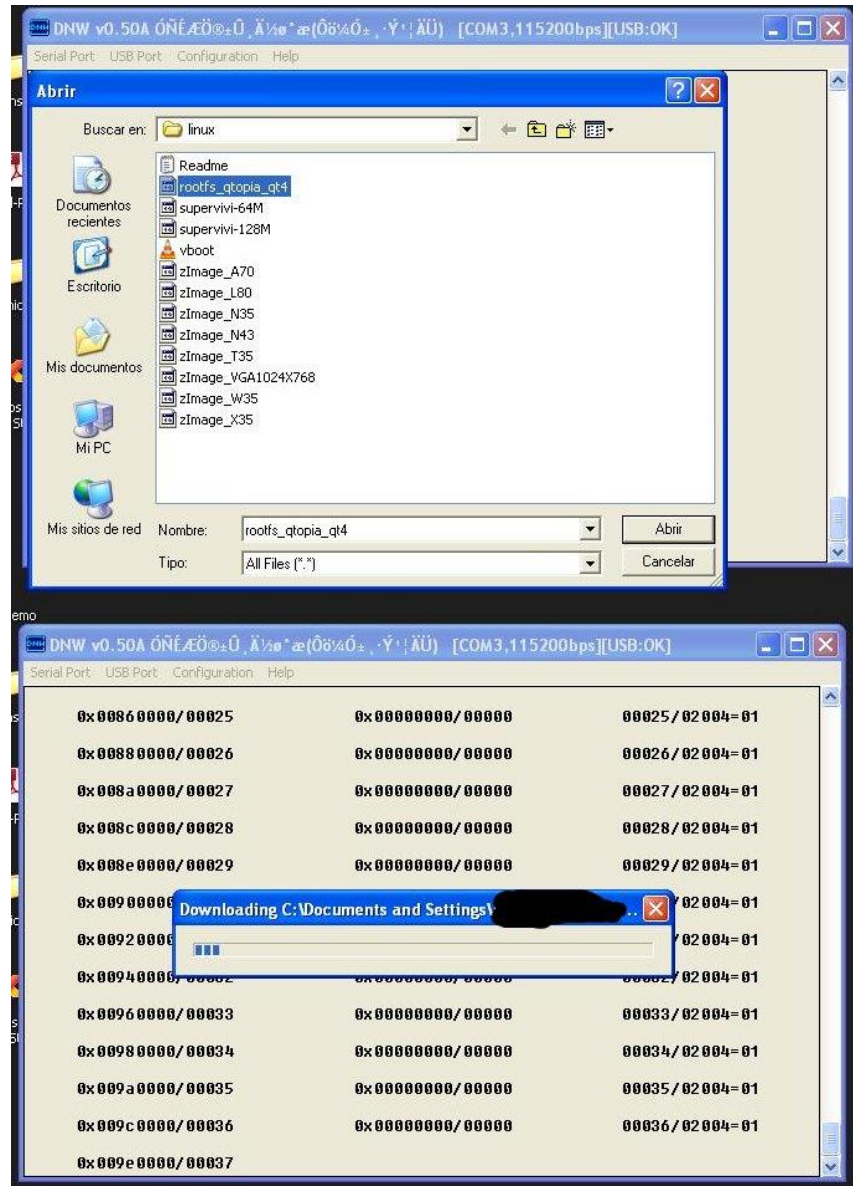
En caso de elegir una opción que no corresponde con la del LCD el DNW no dará ningún error y parecería que todo estuviera bien, pero cuando se finaliza y se enciende la ARM mini2440 la pantalla presentara errores, esto porque el kernel está compilado con otro tipo de resolución para la pantalla LCD.

Finalmente se carga la imagen del sistema operativo QTOPIA, se presiona y. De la misma manera USB Port → Transmit y se elige la imagen.

El proceso de finalización tardara un tiempo considerable. Después de que finalice se apaga la ARM Mini2440 y se regresa el switch de BOOT a NAND para que arranque desde ese sector de memoria, y se enciende nuevamente la ARM mini2440 con el sistema operativo QTOPIA.

Para las configuraciones iniciales (pedirá calibrar la pantalla táctil, configurar la Zona Horaria, fecha, etc.) y en general para utilizar la pantalla LCD de manera efectiva se sugiere utilizar un lápiz óptico. El idioma está en chino, así que se abre una aplicación de diccionario que es visible por que el icono son dos banderas: China y Británica, y elegir el idioma Ingles.

Figura 38. Elección e instalación SO Qtopia



## **Anexo B.** Manual de instalación del protocolo de red OpenSSH en la Board ARM mini2440

Para instalar OpenSSH, se deben descargar algunos archivos desde aquí.

Una vez descargado, se corre el programa build.sh con el siguiente comando:

```
sh build.sh
```

Instalamos el OpenSSH dentro de la mini2440 con SO Linux, así:

- Copiamos los siguientes archivos a la board en la dirección /lib:

```
~/openssh/zlib/lib/libz.so.1.2.5  
~/openssh/openssl/lib/libcrypto.so.1.0.0  
~/openssh/openssl/lib/libssl.so.1.0.0
```

Corremos los siguientes comandos dentro de la board:

```
>ln -s /lib/libz.so.1.2.5 /lib/libz.so.1  
>ln -s /lib/libcrypto.so.1.0.0 /lib/libcrypto.so  
>ln -s /lib/libssl.so.1.0.0 /lib/libssl.so
```

- Copiamos el siguiente archivo a la board en la dirección /usr/sbin:

```
~/openssh-5.6p1/sshd
```

Copiamos los siguientes archivos a la board en la dirección /usr/local/bin:

```
~/openssh-5.6p1/"scp sftp ssh ssh-add ssh-agent ssh-keygen ssh-keyscan"
```

Copiamos los siguientes archivos a la board en la dirección /usr/libexec:

```
~/openssh-5.6p1/"sftp-server ssh-keysign"
```

Copiamos el siguiente archivo a la board en la dirección /usr/local/etc:

```
~/openssh-5.6p1/sshd_config,ssh_config
```

- En la board corremos los siguientes comandos:

```
> mkdir -p /usr/local/etc/  
> mkdir -p /var/run
```

```
> mkdir -p /var/empty/sshd  
> chmod 755 /var/empty
```

- Copiamos los archivos que contengan /ssh\_\* a la board en la dirección /usr/local/etc
- Agregamos una línea de código al archivo /etc/passwd:

```
sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin
```

Se debe tener mucho cuidado al agregar la línea de código, ya que si se elimina algo en ese archivo, todo puede dejar de funcionar y el usuario deberá instalar de nuevo el SO operativo, visto en el Anexo A.

- Probamos que haya quedado correctamente instalado:

Iniciamos el demonio sshd en la board desde la consola: # /usr/sbin/sshd

Luego en el PC se pone \$ ssh root@192.168.1.230 y si conecta quiere decir que la instalación se realizó correctamente.



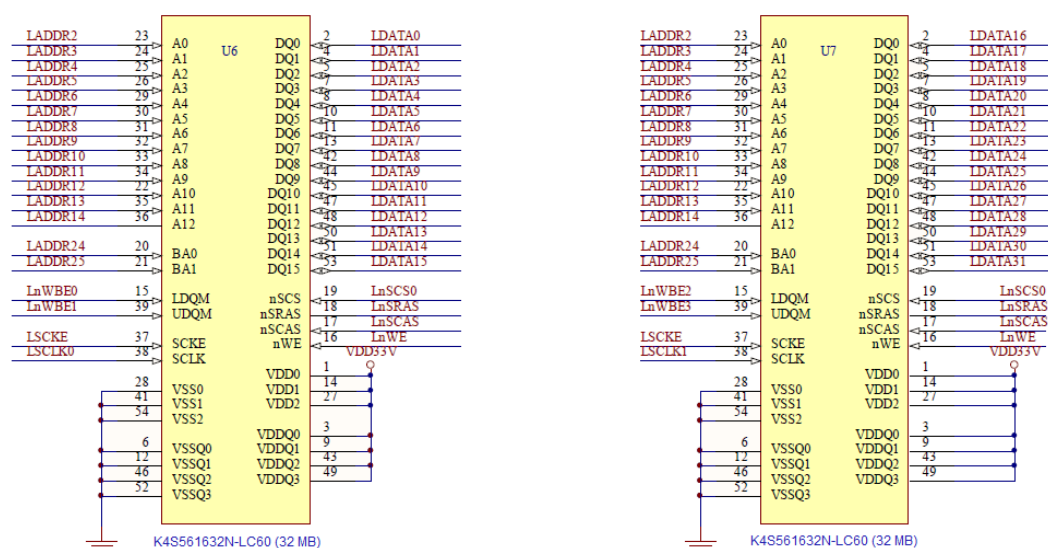
## Anexo C. Descripción partes Board ARM mini2440<sup>41</sup>

A continuación se describen detalladamente cada una de las partes de la Board ARM mini2440 y diferentes formas de conexión y utilización.

### SDRAM

La board mini2440 utiliza dos (2) chips SDRAM de 32 MB cada uno (referencia: K4S561632N-LC60) o 64 MB por ambos. Los dos chips se unen en paralelo para formar un bus de datos de 32 bits, con el fin de aumentar la velocidad de acceso.

**Figura 39.** Esquema de los chips SDRAM (K4S561632N-LC60)



**Fuente:** <<http://www.andahammer.com/assets/Uploads/All2440/Firmware/Mini2440Essentials.pdf>> Pág. 4

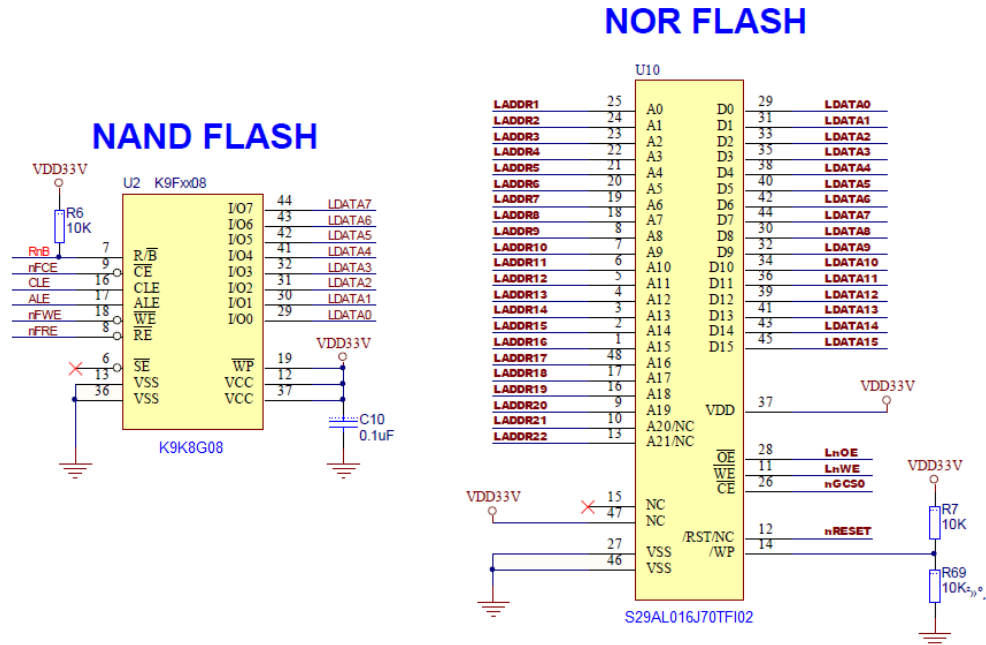
### Flash de almacenamiento y modo de selección de inicio

La board mini2440 utiliza dos (2) memorias flash de almacenamiento. Una es la NOR Flash de 2 MB (referencia: S29AL016J70TFI02) y la otra es la NAND Flash, que para este caso será de 1 GB (referencia: K9K8G08). Ambos tipos de almacenamiento se pueden utilizar como fuente de arranque, ya que la mini2440

<sup>41</sup> Mini2440 Hardware Essentials. Industrial ARMWorks. [En línea. Acceso libre] <[www.friendlyarm.us](http://www.friendlyarm.us)> Agosto 2009.

cuenta con dos tipos de arranque: NAND y NOR flash. Con el switch 'Modo de arranque' se puede elegir el modo en el que se quiere iniciar la board.

**Figura 40.** Modos de arranque y flash de almacenamiento



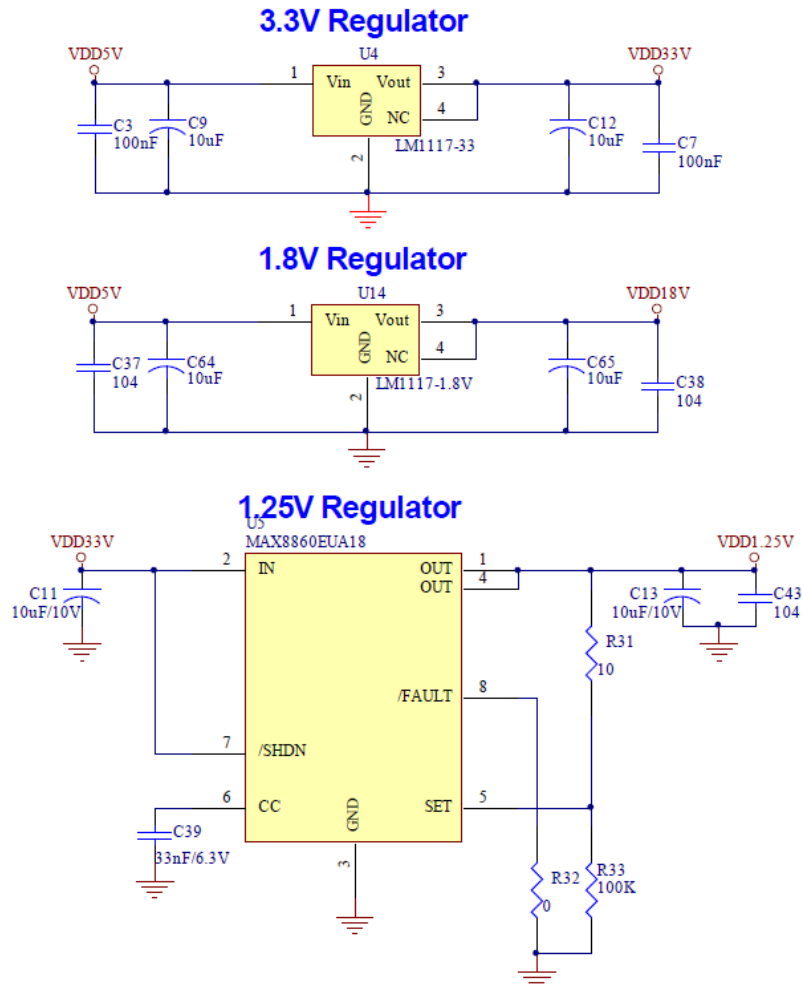
**Fuente:** <<http://www.andahammer.com/assets/Uploads/All2440/Firmware/Mini2440Essentials.pdf>> Pág. 5

## Sistema de alimentación

La board necesita una fuente de alimentación externa de 5V, la cual es proporcionada por el adaptador de voltaje que se conecta a la red eléctrica. A través de diferentes reguladores de voltaje se reduce el voltaje para generar: 3.3V, 1.8V y 1.25V.

Debido a que la mini2440 no está diseñada especialmente para los dispositivos móviles de mano, no incorpora un circuito completo de gestión de potencia. La fuente de alimentación es controlada por el interruptor de alimentación (S1).

Figura 41. Reguladores de voltaje board mini2440

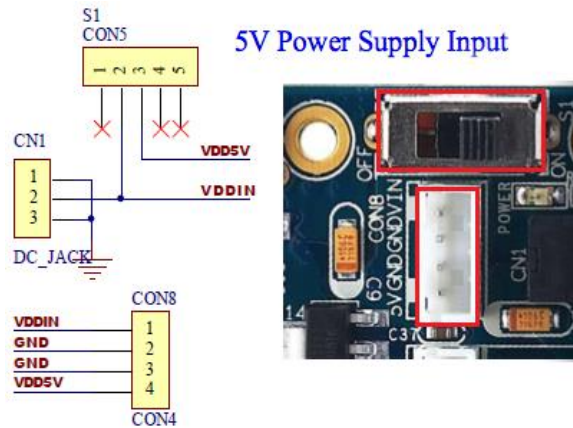


**Fuente:**

<<http://www.andahammer.com/assets/Mini2440Schematic.pdf?PHPSESSID=f170285010a9474671202e28a3fdd173>>

Para la comodidad de conectar otra fuente de alimentación, la board cuenta con un conector (CON8). Es un conector de 4 pines, con espaciado de 2.0 mm entre ellos. El primer y último pin son 5V, y los dos pines entre éstos será tierra (GND).

**Figura 42.** Conector para fuente de alimentación externa (CON8)

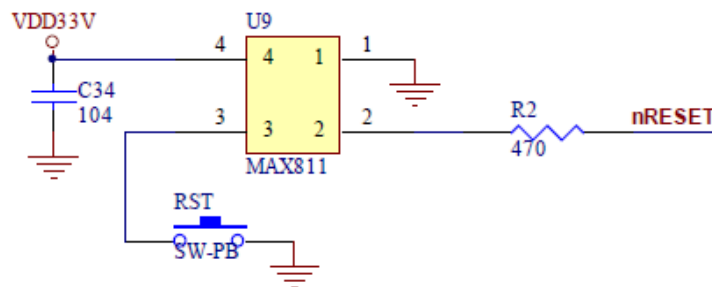


**Fuente:** mini2440\_schematic\_20100127.pdf

### Circuito reset

Un sencillo circuito integrado (MAX811) se utiliza para restablecer el funcionamiento de la board.

**Figura 43.** Circuito reset



**Fuente:** <<http://www.andahammer.com/assets/Uploads/All2440/Firmware/Mini2440Essentials.pdf>> Pág. 6

### Leds de usuario

La mini2440 cuenta con 4 leds programables, éstos están conectados al puerto GPIO de la CPU. Los leds son un tipo de indicador de estado comúnmente usado en este tipo de tarjetas de desarrollo. Éstos se activan en bajo (alumbran cuando la señal es baja).

**Tabla 2.** Distribución de leds de usuario

Led	GPIO	Nombre en el esquemático
1	GPB5	nLED_1
2	GPB6	nLED_2
3	GPB7	nLED_3
4	GPB8	nLED_4

### Botones de usuario

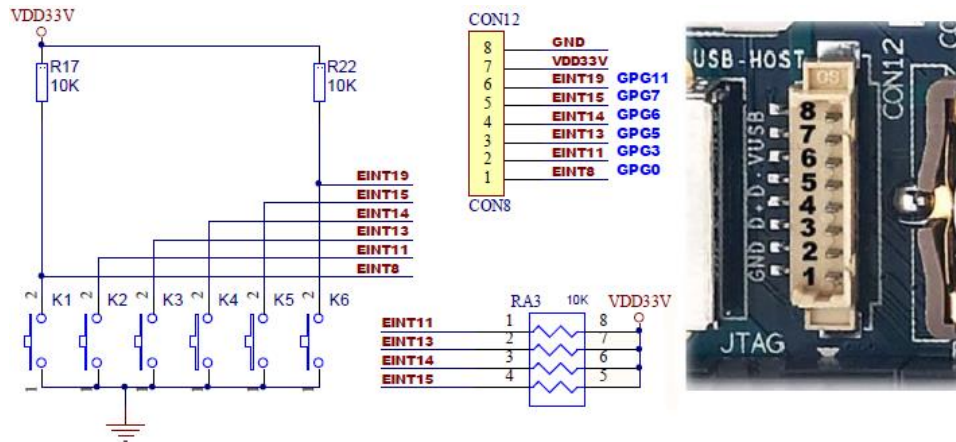
La board cuenta con 6 botones de usuario. Estos botones están conectados directamente a la salida de los leds, haciendo posible el control de leds a través de éstos. También se pueden utilizar para alguna función especial en alguna otra aplicación, por lo que también están conectados al puerto GPIO y al conector CON12.

**Tabla 3.** Distribución botones de usuario

Botón	GPIO	Nombre en el esquemático	CON12
1	GPG0	EINT8	CON12.1
2	GPG3	EINT11	CON12.2
3	GPG5	EINT13	CON12.3
4	GPG6	EINT14	CON12.4
5	GPG7	EINT15	CON12.5
6	GPG11	EINT19	CON12.6

CON12.7 es fuente de alimentación de 3.3V y CON12.8 es tierra (GND)

**Figura 44. Botones de usuario**

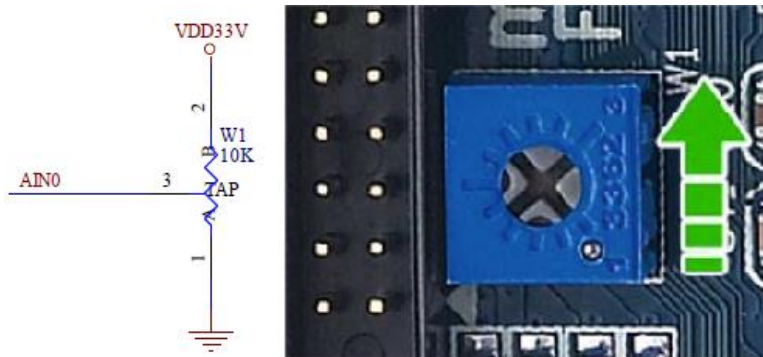


**Fuente:** <<http://www.andahammer.com/assets/Uploads/All2440/Firmware/Mini2440Essentials.pdf>>  
 Pág. 8

**Salida Análogo/Digital (ADC)**

La mini2440 tiene 4 canales de conversión ADC, los cuales pueden estar conectados a la salida de los leds de usuario; los canales se encuentran en la interfaz CON4 (puerto GPIO). Para las pruebas del ADC, la entrada AIN0 ha sido relacionada con la resistencia variable W1.

**Figura 45. Prueba del ADC con la resistencia variable W1**

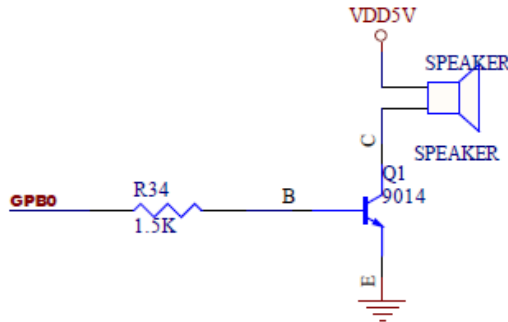


**Fuente:** <<http://www.andahammer.com/assets/Uploads/All2440/Firmware/Mini2440Essentials.pdf>>  
 Pág. 8

## PWM Buzzer

Este pequeño timbre PWM se utiliza para realizar pruebas de timbre sobre la mini2440. La señal es programada a través del pin 31 del puerto GPIO (GPB0).

**Figura 46.** Prueba de timbre PWM a través del puerto GPIO

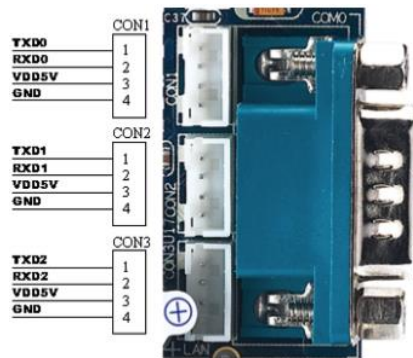


**Fuente:** <<http://www.andahammer.com/assets/Uploads/All2440/Firmware/Mini2440Essentials.pdf>> Pág. 8

## Puerto serial

La board mini2440 soporta 3 puertos seriales. Cada uno tiene un conector con señales de nivel TTL y todos están conectados a CON4, el puerto GPIO. Los puertos CON1 y CON2 pueden ser combinados como un puerto serie completamente funcional. En la mayoría de las prácticas los puertos se utilizan para enviar y recibir datos. Para comodidad del usuario, CON1 se ha convertido en RS-232.

**Figura 47.** Puertos seriales y RS-232

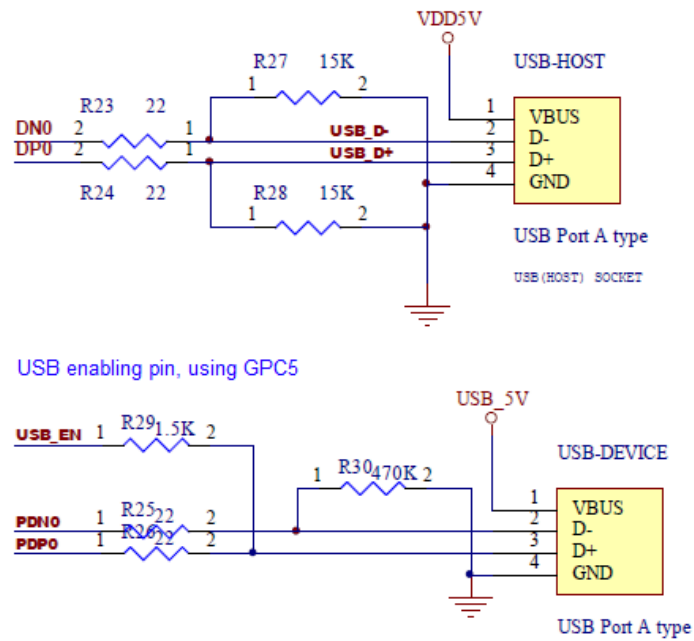


**Fuente:** <<http://www.grupoitech.com.br/docs/componentes/FriendlyARM/MBD-mini2440+7in-LCD-manual.pdf>> Pág. 18

## Interfaz USB

La board cuenta con dos (2) interfaces USB: USB host y USB esclavo. La primera es como la de un computador ordinario, en donde se puede conectar un teclado, mouse, cámara, etc. El USB esclavo está dedicado, generalmente, para la descarga de programas a la tarjeta. El USB esclavo se puede controlar con un bit de registro GPC5, ya sea para activarlo o desactivarlo. Se puede desactivar para ahorrar recursos de la CPU.

**Figura 48.** Interfaces USB Host y esclavo



**Fuente:** <<http://www.grupoitech.com.br/docs/componentes/FriendlyARM/MBD-mini2440+7in-LCD-manual.pdf>> Pág. 19

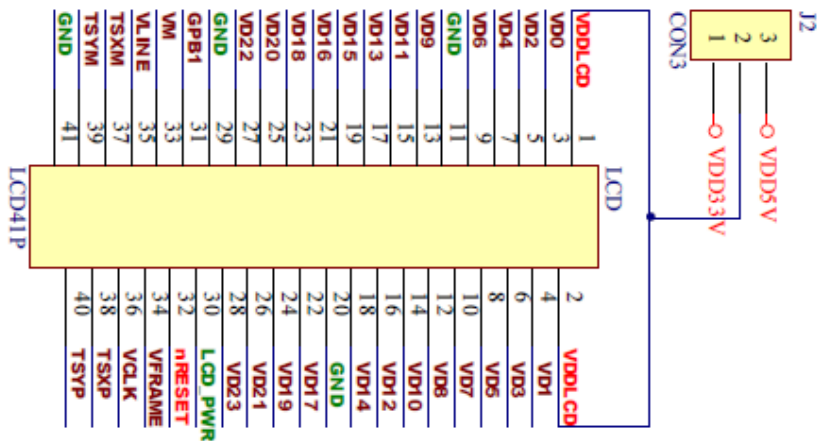
## Interfaz LCD

La interfaz LCD en la mini2440 tiene un conector blanco de 41 pines con un espacio entre conectores de 0.5 mm. Los pines 37, 38, 39 y 40 son la interfaz de la pantalla táctil.

El puente J2 es el conductor de voltaje para la alimentación de la pantalla LCD. El voltaje de alimentación varía entre 5V o 3.3V. Generalmente todas las pantallas LCD manejan un voltaje de alimentación de 5V.



**Figura 49.** Interfaz LCD



**Fuente:** <<http://www.grupoitech.com.br/docs/componentes/FriendlyARM/MBD-mini2440+7in-LCD-manual.pdf>> Pág. 20

## EEPROM

Este chip está conectado a los pines de la señal I2C en la CPU. Se utiliza principalmente para probar el bus I2C.

## Interfaz Ethernet

Para la conexión Ethernet, la board utiliza un chip DM9000 autoadaptable con conector Ethernet RJ-45. El conector RJ-45 contiene un inductor acoplado para conectar el módulo mini2440 a un router mediante cable Ethernet, por lo tanto no es necesario hacer una adaptación para tener conexión a la red.

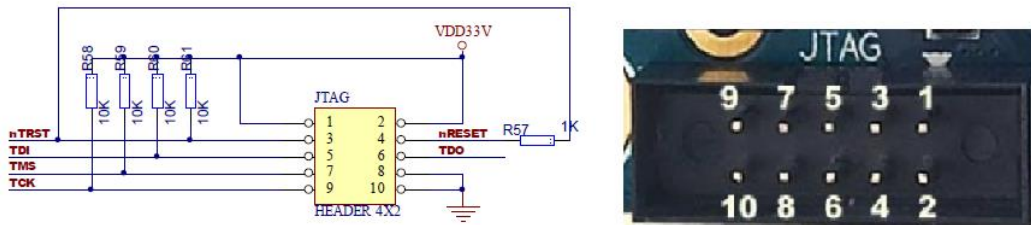
## Interfaz de audio

Utiliza un chip UDA1341, el cual es conectado vía I2S. Se utiliza como códec estéreo para realizar un sistema de códec de audio.

## Interfaz JTAG

Esta interfaz por lo general es utilizada para depuración. Consta de 10 pines y tiene cuatro señales: TMS (Selector de Modo de Testeo), TCK (Reloj de Testeo), TDI (Entrada de Datos de Testeo) y TDO (Salida de Datos de Testeo).

**Figura 50.** Interfaz JTAG

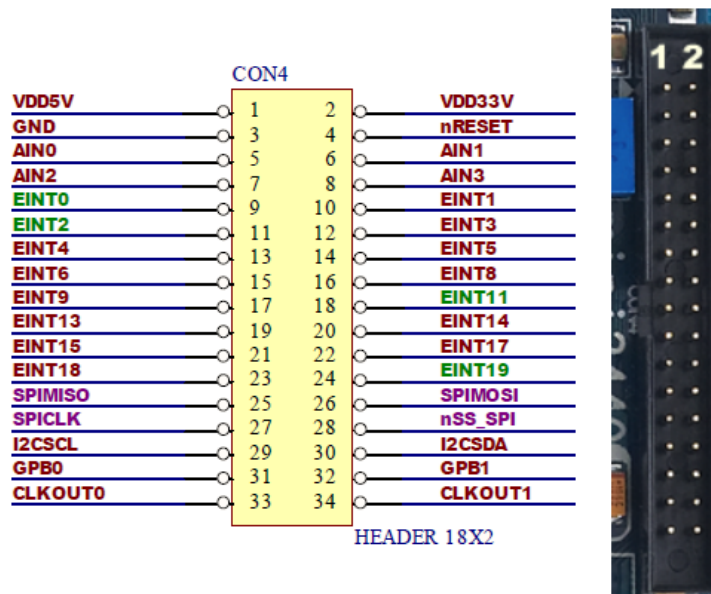


**Fuente:** <<http://www.grupoitech.com.br/docs/componentes/FriendlyARM/MBD-mini2440+7in-LCD-manual.pdf>> Pág. 24

### Interfaz GPIO

La palabra GPIO es la abreviación de Interfaz de Entrada/Salida de Propósito General (General Purpose Input/Output interfaces). Este puerto cuenta con 34 pines separados 2.0 mm entre sí, y se conoce como CON4. Contiene otros pines, con los cuales se pueden controlar distintas interfaces de la board. Sus funciones pueden variar dependiendo de la aplicación para la que se le esté utilizando. Así mismo, los pines pueden ser usados como entradas o como salidas.

**Figura 51.** Interfaz GPIO



**Fuente:** <<http://www.grupoitech.com.br/docs/componentes/FriendlyARM/MBD-mini2440+7in-LCD-manual.pdf>> Pág. 24

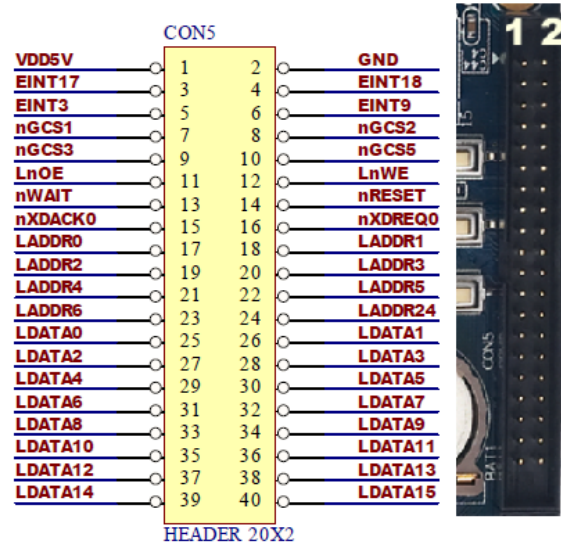
**Tabla 4.** Distribución pines puerto GPIO

PIN	Nombre	Descripción	PIN	Nombre	Descripción
1	VDD5V	5V alimentación (In/Out)	2	VDD3.3V	3.3V alimentación (output)
3	GND	Tierra	4	NRESET	Señal de reset (output)
5	AIN0	Entrada canal 0	6	AIN1	Entrada canal 1
7	AIN2	Entrada canal 2	8	AIN3	Entrada canal 3
9	EINT0	EINT0/GPF0	10	EINT1	EINT1/GPF1
11	EINT2	EINT2/GPF2	12	EINT3	EINT3/GPF3
13	EINT4	EINT4/GPF4	14	EINT5	EINT5/GPF5
15	EINT6	EINT6/GPF6	16	EINT8	EINT8/GPG0
17	EINT9	EINT9/GPG1	18	EINT11	EINT11/GPG3/nSS1
19	EINT13	EINT13/GPG5/SPIMISO1	20	EINT14	EINT14/GPG6/SPIMOSI1
21	EINT15	EINT15/GPG7/SPICLK1	22	EINT17	EINT17/GPG9/nRST1
23	EINT18	EINT18/GPG10/nCTS1	24	EINT19	EINT19/GPG11
25	SPIMISO	SPIMISO/GPE11	26	SPIMOSI	SPIMOSI/EINT14/GPG6
27	SPICLK	SPICLK/GPE13	28	nSS_SPI	nSS_SPI/EINT10/GPG2
29	I2CSCL	I2CSCL/GPE14	30	I2CSDA	I2CSDA/GPE15
31	GPB0	TOUT0/GPB0	32	GPB1	TOUT1/GPB1
33	CLKOUT0	CLKOUT0/GPH19	34	CLKOUT1	CLKOUT/GPH10

### Interfaz Bus del sistema

El bus del sistema, llamado como CON5 en la mini2440, contiene 16 cables de datos (D0-D15), 8 cables de direcciones (A0-A6, A24) y algunos cables de control de señales (selección del chip, R/W, reset, etc.).

**Figura 52.** Interfaz bus del sistema



**Fuente:** <<http://www.grupoitech.com.br/docs/componentes/FriendlyARM/MBD-mini2440+7in-LCD-manual.pdf>> Pág. 27