

## TRABAJO ESPECIAL DE GRADO

### **DISEÑO E IMPLEMENTACIÓN DE UN SERVIDOR WEB EMBEBIDO COMO SISTEMA DE CONTROL DE ACCESO AL LABORATORIO DE INVESTIGACIÓN Y DESARROLLO ELECTRÓNICO DE LA EIE EN LA UCV**

Presentado ante la Ilustre  
Universidad Central de Venezuela  
por el Br. Daniel Andrés González Mariño  
para optar al título de  
Ingeniero Electricista.

Caracas, 2014

## TRABAJO ESPECIAL DE GRADO

### **DISEÑO E IMPLEMENTACIÓN DE UN SERVIDOR WEB EMBEBIDO COMO SISTEMA DE CONTROL DE ACCESO AL LABORATORIO DE INVESTIGACIÓN Y DESARROLLO ELECTRÓNICO DE LA EIE EN LA UCV**

Tutor Académico: Ing. Servando Álvarez

Presentado ante la Ilustre  
Universidad Central de Venezuela  
por el Br. Daniel Andrés González Mariño  
para optar al título de  
Ingeniero Electricista.

Caracas, 2014

## CONSTANCIA DE APROBACIÓN

Caracas, 22 de octubre de 2014

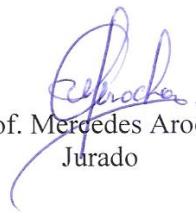
Los abajo firmantes, miembros del Jurado designado por el Consejo de Escuela de Ingeniería Eléctrica, para evaluar el Trabajo Especial de Grado presentado por el Bachiller Daniel A. González M., titulado:

**“DISEÑO E IMPLEMENTACION DE UN SERVIDOR WEB EMBEBIDO  
COMO SISTEMA DE CONTROL DE ACCESO  
AL LABORATORIO DE INVESTIGACION Y DESARROLLO  
ELECTRONICO DE LA EIE EN LA UCV”**

Consideran que el mismo cumple con los requisitos exigidos por el plan de estudios conducente al Título de Ingeniero Electricista en la mención de Electrónica, y sin que ello signifique que se hacen solidarios con las ideas expuestas por el autor, lo declaran APROBADO.



Prof. Ebert Brea  
Jurado



Prof. Mercedes Arocha  
Jurado



Prof. Servando Álvarez  
Tutor



UNIVERSIDAD CENTRAL DE VENEZUELA  
FACULTAD DE INGENIERÍA  
ESCUELA DE INGENIERÍA ELÉCTRICA  
DPTO. DE ELECTRÓNICA, COMPUTACIÓN Y CONTROL



*ACTA*

Quienes suscriben, Miembros del Jurado designado por el Consejo de Escuela de Ingeniería Eléctrica de la Facultad de Ingeniería, de la Universidad Central de Venezuela para examinar el Trabajo Especial de Grado presentado por el Bachiller **DANIEL A. GONZÁLEZ M.** cédula de identidad N° **V-18211493**, bajo el título de **“DISEÑO E IMPLEMENTACIÓN DE UN SERVIDOR WEB EMBEBIDO COMO SISTEMA DE CONTROL DE ACCESO AL LABORATORIO DE INVESTIGACIÓN Y DESARROLLO ELECTRÓNICO DE LA EIE EN LA UCV”**; a los fines de cumplir con el requisito legal para optar al grado de Ingeniero Electricista, dejan constancia de lo siguiente:

Considerando que el Trabajo Especial de Grado presentado y defendido por el Bachiller en referencia, presenta un grado de originalidad y excelencia tanto en su contenido como en su presentación formal que lo convierte en una fuente referencial útil, acuerda por unanimidad solicitarle a las autoridades correspondientes de la Facultad de Ingeniería, tengan a bien otorgar **MENCIÓN HONORÍFICA**, al Trabajo Especial de Grado antes mencionado.

En Caracas, Veintidós (22) de octubre de 2014.

PROF. EBERT BREA

Jurado

PROF. MERCEDES AROCHA

Jurado

PROF. SERVANDO ALVAREZ

Tutor Académico

*A mi Madre, Santa del Jesús Mariño. Sin ti nada de esto habría sido posible.*

*¡Te Amo!*

*A la memoria de mi Abuelo, Juan Ramón Figueroa*

## RECONOCIMIENTOS Y AGRADECIMIENTOS

A Dios Todopoderoso.

A mis Padres, por nunca desistir y siempre tener fe en mí.

A Mami y mi Abuelo, Olga Cuenca y Juan Ramón Figueroa, no puedo explicar en palabras todo lo agradecido que les estoy.

A mi madrina Andreina, por su amor incondicional y Jose por apoyarme siempre que lo necesitaba.

A Eduardo Andrés, Ana Lucia y Alejandro, por ser una gran felicidad en mi vida.

A mi madrina Adriana, por su gran amor y siempre estar presente en cada momento.

A Franco Javier y Dario Andrés, hermanos de la vida.

A Maru, Danilo y Félix Alejandro, su amistad es muy valiosa para mí.

A todos mis primos, tíos y tías, tenerlos en mi mente siempre ha sido razón de fortaleza.

A mi tutor, Servando Álvarez, sé que tenerme paciencia no es tarea fácil.

A Marcos Ariza, gracias por toda la ayuda brindada durante la realización de este trabajo.

A la Escuela de Ingeniería Eléctrica y sus profesores, Rafael Rivero, Pedro Pinto, Mercedes Arocha, Simón Morales, Tamara Pérez, William La Cruz, Joao Nunes, Napoleón Malpica y todos aquellos profesores de la Facultad que de una u otra manera me ayudaron a lo largo de este camino.

A mis amigos Richard Jiménez, Carlos Silva, Juan Aponte, César Gutiérrez y todos aquellos que siempre estuvieron allí brindándome su amistad.

Especialmente agradecido con mis amigos Jorge Díaz y Víctor Urbina, su amistad es invaluable.

*¡Gracias!*

**González M., Daniel A.**

**DISEÑO E IMPLEMENTACIÓN DE UN SERVIDOR WEB  
EMBEBIDO EN UN MICROCONTROLADOR COMO SISTEMA  
DE CONTROL DE ACCESO AL LABORATORIO DE  
INVESTIGACIÓN Y DESARROLLO ELECTRÓNICO DE LA EIE  
EN LA UCV**

**Tutor Académico: Ing. Servando Álvarez. Tesis. Caracas, U.C.V. Facultad de Ingeniería. Escuela de Ingeniería Eléctrica. Ingeniero Electricista. Opción: Electrónica. 2014, xiv, 83 + anexos.**

**Palabras Claves:** Control de acceso; Microcontroladores; RFID; Lector de Radiofrecuencia; Servidor Web; Sistema Embebido; Páginas Web; Servidor Web Apache; PHP; HTML5; CSS3; MySQL; Linux; pcDuino.

**Resumen.** En la actualidad, los sistemas de control de acceso son necesarios y tienen una importancia creciente, tanto para las empresas privadas como para los organismos públicos. En algunos casos es de vital importancia el conocimiento de las personas que acceden a dichos lugares de forma eficiente y rápida, desde cualquier lugar y en cualquier momento. Para cubrir esta necesidad, se diseñó e implementó un servidor web embebido en un microcontrolador como sistema de control de acceso a una instalación. Se realizó el diseño de una unidad de control la cual sería la encargada de manejar los datos de los elementos de identificación así como ejecutar acciones de autorización. Esta unidad fue diseñada con un microcontrolador PIC18F4550, regulador de tensión, relé para manejar las cerraduras y diversos conectores para elementos externos que se necesiten, todo esto en una tarjeta de circuito impreso. La comunicación de la controladora con el servidor se realiza utilizando comunicación serial TTL que se encarga de realizar las consultas en la base de datos del servidor y recibir respuesta de éste. Se realizó la implementación de un servidor web en un SBC (single board computer) llamado pcDuino en el cual se instalaron elementos capaces de manejar un servicio web como lo fue Apache junto con una base de datos en MySQL y para el manejo del servidor con PHP. Se realizó una interfaz gráfica de usuario (GUI) basada en páginas web diseñadas en HTML5 con el manejador de estilos CSS3 y funciones en Javascript así como también páginas dinámicas con PHP; en estas páginas el usuario puede observar la información pertinente al control de acceso así como ingresar nuevos datos al sistema.

# ÍNDICE GENERAL

<b>CONSTANCIA DE APROBACION</b>	iii
<b>ACTA</b>	iv
<b>DEDICATORIA</b>	v
<b>RECONOCIMIENTOS Y AGRADECIMIENTOS</b>	vi
<b>RESUMEN</b>	vii
<b>ÍNDICE GENERAL</b>	viii
<b>ÍNDICE DE FIGURAS</b>	xii
<b>ÍNDICE DE TABLAS</b>	xiv
<b>LISTA DE SIGLAS Y ACRÓNIMOS</b>	xv
<b>INTRODUCCIÓN</b>	1
<b>CAPITULO I. CONCEPTUALIZACIÓN DEL PROYECTO</b> .....	3
1.1 PLANTEAMIENTO DEL PROBLEMA.....	3
1.2 JUSTIFICACIÓN.....	3
1.3 OBJETIVO GENERAL.....	4
1.4 OBJETIVOS ESPECIFICOS.....	4
<b>CAPITULO II. MARCO REFERENCIAL</b> .....	5
2.1 Control de Acceso.....	5
2.1.1 Definición.....	5
2.1.1.1 Autenticación.....	5
2.1.1.2 Autorización.....	5
2.1.1.3 Administración de los datos.....	5
2.1.2 Esquema Básico de un Control de Acceso.....	6
2.1.2.1 Funcionamiento General.....	6
2.1.3 Esquema General de un control de Acceso.....	7
2.1.4 Elementos de Identificación Usados en un control de acceso.....	8
a. Teclados pin.....	8
b. Lectores de tarjetas magnéticas.....	8
c. Lectores de códigos de barra.....	9

d. Lectores biométricos.....	9
e. Lectoras de protocolo 1-Wire©.....	9
f. Lectores de proximidad.....	10
- Protocolo Wiegand.....	11
2.2 Sistemas Embebidos (SE).....	14
2.2.1 Definición.....	14
2.2.2 Características.....	14
2.2.3 Firmware.....	15
2.2.4 Exigencias.....	15
2.2.4.1 Confiabilidad.....	15
2.2.4.2 Disponibilidad.....	15
2.2.4.3 Seguridad.....	15
2.2.5 Fallas.....	15
2.2.6 Entradas y Salidas.....	16
2.2.6.1 Entradas y Salidas Analógicas.....	16
2.2.6.2 Entradas y Salidas Digitales.....	16
2.2.7 Lectura y Escritura.....	16
2.2.7.1 Polling.....	17
2.2.7.2 Interrupciones.....	17
2.2.8 Comunicación.....	17
2.2.9 Formas de Implementar SE.....	17
2.2.9.1 Microcontroladores.....	18
2.2.9.2 IO Boards.....	18
2.2.9.3 Sistemas sobre un chip (SOC) .....	19
2.2.9.4 Computadora embebida de una única tarjeta...	19
- Raspberry PI.....	20
- Intel® Galileo.....	21
- BeagleBoard.....	22
- pcDuino.....	23
2.2.10 GNU/Linux.....	24

2.3	Servidor Web.....	25
2.3.1	Definición de Servidor.....	25
2.3.2	Servidor Web.....	26
2.3.2.1	El servidor web Apache.....	26
<b>CAPITULO III. DEFINICIÓN Y DESCRIPCIÓN DEL HARDWARE..</b>		<b>27</b>
3.1	Definición del hardware.....	27
3.2	Descripción del hardware.....	28
3.2.1	Periféricos de Entrada y Salida.....	28
3.2.1.1	Periféricos de Entrada.....	28
3.2.2.2	Periféricos de Salida.....	29
3.2.2	Unidad Controladora.....	31
3.2.2.1	Alimentación.....	33
3.2.2.2	Microcontrolador PIC18F4550.....	33
3.2.2.3	Módulos de comunicación y conector de botón para salir.....	34
3.2.2.4	Conectores para entrada Wiegand.....	36
3.2.2.5	Relé.....	36
3.2.3	Servidor Web.....	39
<b>CAPITULO IV. DEFINICIÓN Y DESCRIPCIÓN DEL SOFTWARE...</b>		<b>43</b>
4.1	Definición del software.....	43
4.2	Descripción del software.....	44
4.2.1	Firmware de la unidad controladora.....	44
4.2.1.1	Rutina principal.....	45
4.2.1.2	Subrutina Agregar Bits.....	46
4.2.1.3	Subrutina Verificar Paridad.....	47
4.2.1.4	Subrutina Enviar Data Serial.....	48
4.2.1.5	Subrutina Abrir Puerta.....	49
4.2.2	Programa del pcDuino.....	49
4.2.2.1	Recepción de la data de la controladora.....	50
4.2.2.2	Programa del Servidor Web.....	53

-	Conexión Cliente - Servidor .....	53
-	Base de Datos.....	55
-	Interfaz gráfica de usuario (GUI).....	56
<b>CAPITULO V. PRUEBAS Y RESULTADOS.....</b>	<b>58</b>	
5.1	Pruebas Funcionales.....	58
5.1.1	Pruebas iniciales en la controladora.....	58
5.1.2	Pruebas iniciales en el servidor web.....	59
5.1.3	Pruebas de navegación en el servidor web.....	60
5.1.4	Pruebas de las alarmas del sistema.....	70
5.1.5	Pruebas completas del sistema.....	71
5.2	Especificaciones del equipo.....	74
5.2.1	Especificaciones de la unidad controladora.....	74
5.2.2	Especificaciones del servidor.....	75
5.3	Manual del equipo.....	75
<b>CONCLUSIONES.....</b>	<b>76</b>	
<b>RECOMENDACIONES.....</b>	<b>78</b>	
<b>REFERENCIAS BIBLIOGRÁFICAS.....</b>	<b>80</b>	
<b>BIBLIOGRAFÍA.....</b>	<b>81</b>	
<b>ANEXO.....</b>	<b>83</b>	

## ÍNDICE DE FIGURAS

<b>Figura 1-</b> Esquema de conexiones de un control de acceso básico .....	6
<b>Figura 2 –</b> Esquema General de un control de acceso .....	7
<b>Figura 3 –</b> Secuencia de lectura de la información de una tarjeta .....	11
<b>Figura 4 –</b> Ejemplo de transmisión de datos en el protocolo Wiegand .....	12
<b>Figura 5 –</b> Formato del Wiegand-26.....	13
<b>Figura 6- Diagrama de Bloques General del Equipo.....</b>	27
<b>Figura 7- Lector de tarjetas de proximidad KR101E a 125KHz.....</b>	29
<b>Figura 8- Lector en estado de espera (izquierda), autorizado (derecha) y beeper (abajo) .....</b>	30
<b>Figura 9- Cerradura eléctrica elegida.....</b>	30
<b>Figura 10- Tarjeta de leds indicadores.....</b>	31
<b>Figura 11- Unidad Controladora final en circuito impreso.....</b>	32
<b>Figura 12- Diagrama de Bloques de la Unidad Controladora.....</b>	32
<b>Figura 13- Bloque de alimentación.....</b>	33
<b>Figura 14- Microcontrolador PIC18F4550 .....</b>	34
<b>Figura 15- Módulos de comunicación y conexión RTE .....</b>	36
<b>Figura 16- Conectores para entrada Wiegand .....</b>	36
<b>Figura 17- Diagrama circuital del módulo del Relé .....</b>	37
<b>Figura 18- Disposición del módulo del Relé en la Tarjeta .....</b>	38
<b>Figura 19- El pcDuino .....</b>	41
<b>Figura 20 – Diagrama Circuital del módulo RTC .....</b>	42
<b>Figura 21 – Módulo RTC implementado .....</b>	42
<b>Figura 22- Diagrama de flujo, rutina principal.....</b>	45
<b>Figura 23- Diagrama de flujo, subrutina agregar bits .....</b>	46
<b>Figura 24- Diagrama de flujo firmware de recepción de datos en el servidor .....</b>	52
<b>Figura 25 - Diagrama de Interacción con el servidor web .....</b>	53
<b>Figura 26 - Diagrama de casos de niveles de accesos .....</b>	55

<b>Figura 27-</b> Página de Inicio del Servidor Web .....	57
<b>Figura 28 –</b> Vista de la página "Quienes Somos" .....	61
<b>Figura 29 –</b> Formulario de Solicitud en línea.....	62
<b>Figura 30 –</b> Página de Contacto.....	62
<b>Figura 31 -</b> Página “Acerca de...” (Información personal) .....	63
<b>Figura 32 –</b> Formulario en formato PDF .....	63
<b>Figura 33 -</b> Página de inicio de sesión .....	64
<b>Figura 34 -</b> Combinación de usuario y contraseñas incorrectos.....	64
<b>Figura 35 -</b> Respuesta de usuario no existe .....	64
<b>Figura 36 -</b> Respuesta de inicio de sesión satisfactorio .....	65
<b>Figura 37 -</b> Registros en la base de datos de inicios de sesión.....	65
<b>Figura 38 -</b> Visualización de personas autorizadas y habilitadas a acceder al laboratorio .....	66
<b>Figura 39 -</b> Visualización de solicitudes realizadas por personas externas al laboratorio .....	67
<b>Figura 40 -</b> Visualización de los últimos 30 ingresos realizados al laboratorio .....	67
<b>Figura 41 -</b> Ingreso de nueva autorización.....	68
<b>Figura 42 -</b> Nueva tarjeta ingresada satisfactoriamente .....	68
<b>Figura 43 -</b> Visualización de las alarmas generadas en el sistema .....	69
<b>Figura 44 -</b> Vista frontal de la maqueta .....	72
<b>Figura 45 -</b> Vista posterior de la maqueta .....	72
<b>Figura 46 -</b> Vista lateral de la maqueta .....	73
<b>Figura 47 -</b> Maqueta con la puerta abierta, indicando que fue autorizado el acceso.	73

## ÍNDICE DE TABLAS

<b>Tabla 1-</b> Comparación de las diferentes versiones del Raspberry Pi .....	20
<b>Tabla 2 -</b> Especificaciones de los Intel® Galileo .....	21
<b>Tabla 3-</b> Comparación de las diferentes versiones del BeagleBoard.....	22
<b>Tabla 4-</b> Comparación de las diferentes versiones del pcDuino .....	23
<b>Tabla 5-</b> Código de colores del lector de proximidad .....	29
<b>Tabla 6-</b> Especificaciones del hardware del pcDuino .....	40
<b>Tabla 7 -</b> Especificaciones de la unidad controladora.....	74
<b>Tabla 8 -</b> Especificaciones del Servidor.....	75

## LISTA DE SIGLAS Y ACRÓNIMOS

**RFID:** Dispositivo de Identificación por Radio Frecuencia

**SE:** Sistemas Embebidos

**ROM:** Memoria de Solo Lectura

**EEPROM:** Memoria de Solo Lectura con Borrado Eléctrico

**DAC:** Convertidor Digital Analógico

**ADC:** Convertidor Analógico Digital

**UART:** Receptor – Emisor Universal Asincrónico

**TTL:** Lógica Transistor-Transistor

**USB:** Bus Serial Universal

**SPI:** Interfaz de Periféricos Seriales

**I2C:** Circuito Inter-Integrados

**SOC:** Sistemas Sobre un Chip

**MMU:** Unidad de Manejo de Memoria

**SBC:** Computadora de Única Tarjeta

**GPU:** Unidad de Procesamiento Gráfico

**HTTP:** Protocolo de Transferencia de Hipertexto

**RTE:** Solicitud de Salida

**SPDT:** Un Polo Doble Tiro

**RTC:** Reloj en Tiempo Real

**GUI:** Interfaz Gráfica de Usuario

**LAMP:** Linux, Apache, MySQL, PHP

**GPL:** Licencia de Propósito General

**HTML:** Lenguaje de Etiquetas de Hipertexto

**OSI:** Interconexión de Sistemas Abiertos

**TCP:** Protocolo de Control de Transmisión

**IP:** Protocolo Internet

**DBA:** Administrador de Base de Datos

## INTRODUCCIÓN

Los sistemas embebidos han tomado mucha importancia en la actualidad ya que al ser sistemas que son controlados por un computador integrado, el cual puede ser un microcontrolador o microprocesador que permita estar dentro del sistema general, fuera del alcance y oculto a la vista para así entonces formar parte de un todo. En este caso el sistema general es un sistema de control de acceso.

Un adecuado sistema para controlar el acceso a una instalación es muy importante para la buena y correcta administración en una empresa o un negocio, se trata de tener el control de entrada y, en algunos casos de la salida, de las personas que hacen vida en el lugar para mejorar la productividad, con seguimiento de horarios, grupos de acceso así como la seguridad de los que allí se encuentren.

El servidor web es el programa informático encargado de procesar las aplicaciones del lado del servidor realizando conexiones bidireccionales y/o unidireccionales, síncronas o asíncronas con el cliente generando una respuesta en cualquier lenguaje o aplicación del lado del cliente. El código recibido por el cliente suele ser compilado y ejecutado por un navegador web. Para la transmisión de todos estos datos suele utilizarse algún protocolo de entre los muchos que se ofrecen, pertenecientes a las capas del modelo OSI.

Este trabajo especial de grado contiene en el Capítulo I toda la información del porqué se realizó esta investigación como el planteamiento del problema, la justificación y por último los objetivos planteados para la resolución del mismo.

En el Capítulo II, se desarrolló el marco teórico de la investigación, todo el conjunto de definiciones que se consideran pertinentes para el entendimiento del proyecto.

Seguidamente, el Capítulo III contiene la definición técnica del equipo, en el cual se realiza una descripción del hardware realizado para el funcionamiento del mismo y por qué se realizaron esas escogencias y en el Capítulo IV, la descripción

general del software realizado, partiendo desde el firmware del microcontrolador hasta el software implementado en el servidor web. Luego en el Capítulo V se muestran los resultados obtenidos luego de realizar la implementación del equipo y su respectivo análisis en el cual se indican todos los beneficios otorgados por el mismo así como también las mejoras que pudieran ser realizadas.

Por último, las conclusiones obtenidas al momento de finalizado el trabajo y las recomendaciones para posibles futuros proyectos basados en éste.

## **CAPITULO I. CONCEPTUALIZACIÓN DEL PROYECTO**

### **1.1 PLANTEAMIENTO DEL PROBLEMA**

Dentro de los laboratorios, se encuentran equipos y elementos de gran importancia para el desarrollo de investigaciones y prácticas. En caso de pérdida, se hace difícil el reponer estos equipos, y si se consigue reponerlo, no se hace inmediatamente, afectando directamente los estudios e investigaciones que para ese entonces se estén realizando y por ende, directamente a los estudiantes (Tesis en su mayoría) que hagan uso de los mismos.

En la actualidad, el uso de sistemas de control de acceso a instalaciones es necesario y tienen una importancia creciente, tanto para las empresas privadas como para los organismos públicos. En algunos casos es de vital importancia el conocimiento de las personas que acceden a dichos lugares de forma eficiente y rápida, desde cualquier lugar y en cualquier momento.

### **1.2 JUSTIFICACIÓN**

Actualmente en el Laboratorio de Investigación y Desarrollo Electrónico de la Escuela de Ingeniería Eléctrica se encuentran muchos materiales y equipos importantes que son de alto costo pertenecientes tanto a la Universidad como a los estudiantes tesistas que hacen su trabajo en el mismo, por lo que la implementación de un sistema de control de control de acceso es muy importante. A la hora de un problema que puede involucrar perdida de material y/o equipos no se sabe quiénes estuvieron en el laboratorio o quien tuvo acceso.

Un sistema de control de acceso permitiría monitorear estos accesos y así poder saber con certeza quienes han estado en el laboratorio; el ubicar esta información en una página web permite acceder a ella de manera remota, sin necesidad de estar en las

instalaciones para poder verificar.

### **1.3 OBJETIVO GENERAL**

Diseñar e implementar un servidor web embebido en un microcontrolador el cual sea capaz de monitorear a través de módulos adaptados para ello, el acceso al Laboratorio de Investigación y Desarrollo Electrónico de la Escuela de Ingeniería Eléctrica de la UCV y presentar la información mediante una página web.

### **1.4 OBJETIVOS ESPECÍFICOS**

- Definir la estructura del servidor embebido a implementar.
- Seleccionar los diversos componentes ya realizado el estudio y que se adapten mejor a los requerimientos del proyecto.
- Diseñar y/o utilizar módulos comerciales para la realización del equipo.
- Programar un firmware que permita el funcionamiento del servidor web.
- Desarrollar una página web en la que esté alojada la información pertinente al servidor.
- Realizar pruebas de funcionamiento del dispositivo implementado.
- Realizar un manual de uso del equipo ya implementado.

## **CAPITULO II. MARCO REFERENCIAL**

### **2.1 Control de Acceso**

#### **2.1.1 Definición**

Un control de acceso es un conjunto de instrumentos, equipamiento, normas y recursos asignados a la identificación de personas y a la restricción de sus atribuciones, destinados a preservar los activos y la confidencialidad dentro de una instalación. (Consentino, 2009a).

Ellos deben ser capaces de registrar fecha, hora y transacciones de los eventos que se realicen para así entonces proveer la posibilidad de una revisión posterior. Se deben cumplir tres características importantes: autenticación, autorización y administración de los datos manejados.

##### **2.1.1.1. Autenticación:**

Es referente a la acción del usuario. Asegurar que es la persona quien dice ser, mediante los métodos utilizados por el sistema (lectores de tarjetas magnéticas, tarjetas de radio frecuencia (RFID), botones magnéticos, lectores biométricos, entre otros).

##### **2.1.1.2. Autorización:**

Permite que el usuario, una vez autenticado, se le permita el acceso a la instalación o recursos.

##### **2.1.1.3. Administración de los datos:**

Permite el manejo de los datos del sistema: añadir usuarios, eliminarlos, modificar su nivel jerárquico, habilitarlos o deshabilitarlos, asignación de funciones específicas, entre otros.

## 2.1.2 Esquema básico de un Control de Acceso

### 2.1.2.1. Funcionamiento General

A través de un ejemplo se puede realizar una mejor explicación de cómo funcionan estos sistemas. Este ejemplo controlará el control de acceso de las personas a una habitación genérica.

El “corazón” de este sistema es el controlador que, por lo general, se encuentra oculto a la vista de los usuarios. A éste serán conectados tanto el elemento de adquisición de la data (proceso de autenticación) como el elemento de autorización, que en muchos casos puede ser una cerradura eléctrica.



*Figura 1- Esquema de conexiones de un control de acceso básico*

En la figura anterior se puede observar un esquema de conexiones sencillo. El funcionamiento de este sistema consiste en que cuando un usuario, que ha sido previamente habilitado en el sistema, desea ingresar, deberá acercar su elemento de identificación al lector para que se pueda realizar la lectura y esta sea transmitida al controlador.

El controlador al recibir la data, buscará en una lista de los usuarios que han sido habilitados; si la tarjeta es encontrada, habilitará la apertura de la puerta enviando una señal a la cerradura eléctrica por un lapso pequeño (por lo general unos pocos segundos), encendiendo una señal (en la mayoría de los casos un led de color verde) indicándole y dándole así el tiempo suficiente al usuario para la apertura de la misma.

En el caso contrario de que el usuario no se encuentre en la lista de tarjetas habilitadas, la cerradura no será habilitada y esto se le indica al usuario mediante una señal (por lo general un led de color rojo). En muchos sistemas también se estila el no

realizar ninguna acción, no encender leds, ni emitir sonidos, esto por motivos de seguridad adicional.

### 2.1.3 Esquema general de un Control de Acceso:

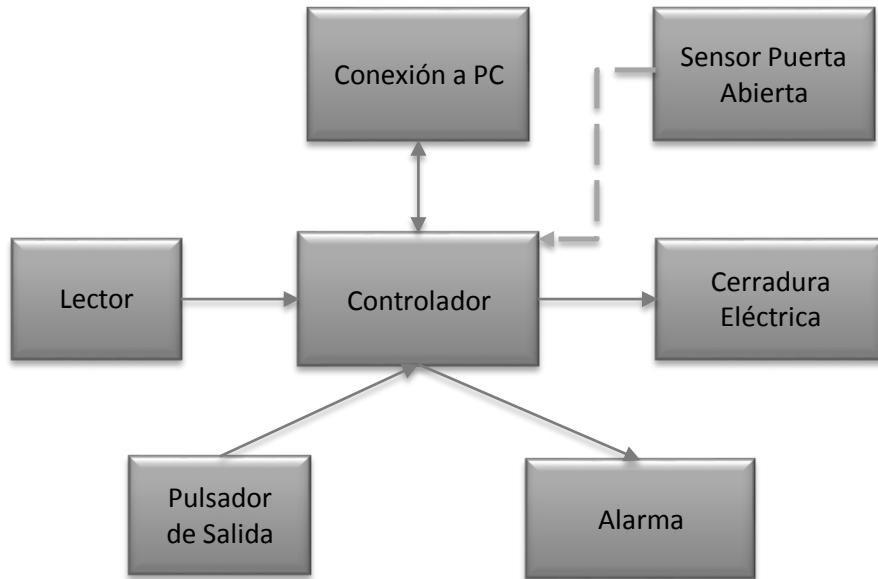


Figura 2 – Esquema General de un control de acceso

En la figura 2, se puede observar un sistema un poco más detallado, con algunos elementos incluidos, entre los que se tienen sistema de alarma, pulsador de salida y una conexión a un computador (PC), a través de la cual se puede establecer comunicación con la controladora y realizar configuraciones de acceso para los usuarios así como configuraciones generales de la controladora.

Las alarmas pueden ser configuradas en dado caso de que al sistema se le introduzca un sensor de puerta abierta, el cual le transmita una señal al controlador y sea indicado; esta información queda registrada en el sistema para una posterior revisión.

## **2.1.4 Elementos de identificación usados en un control de acceso:**

Estos elementos son aquellos dispositivos que, tal como su nombre lo indica, son capaces de identificar a su portador con mayor o menor precisión, esto dependiendo del dispositivo elegido para tal fin.

Entre estos se tienen: teclados pin, lectores de códigos de barra, lectores biométricos, lectores de tarjetas magnéticas, lectoras de 1-Wire® y lectoras de proximidad, dentro de estas últimas las lectoras en formato Wiegand.

### **a. Teclados pin:**

Ofrecen un nivel de seguridad de mediano a bajo. Esto debido a que son muy vulnerables ya que cualquier persona ajena puede observar a un usuario introducir la clave y de allí copiarla e introducirla el mismo.

Algunos de estos teclados tienen la capacidad de almacenar cierto número de claves programables que por lo general son de 4 números y pueden trabajar como dispositivos autónomos; la desventaja de esto es que no existe un registro de quien introdujo que clave, ni a qué hora se produjo un ingreso a la instalación.

Lo correcto es usarlos como dispositivos periféricos de controles de accesos. Para ello se debe estudiar principalmente la controladora a utilizar y buscar un teclado que se adapte a ella.

Actualmente existen controladoras que solo admiten datos en formato Wiegand. Ya con esto se limita también que el teclado deba transmitir usando este protocolo. Esto es una ventaja, ya que muchos elementos periféricos de controles de acceso transmiten utilizando este protocolo.

### **b. Lectores de tarjetas magnéticas:**

Son dispositivos que constan de una unidad lectora de banda magnética, la cual es capaz de reconocer un código (o datos) grabado dentro de la banda.

**c. Lectores de códigos de barra:**

Esta tecnología está adaptada desde la industria de alimentación e identificación. Posee varios formatos y códigos para ser utilizados, son de bajo costo y como ventaja se tiene que los códigos de barra pueden ser impresos en casi cualquier parte y con cualquier tipo de impresora. Entre las desventajas, se tiene que es muy vulnerable debido a que con una simple fotocopia del código de barras, se puede tener acceso.

Debido a esto, suelen ser utilizados en lugares donde se pueda aprovechar al máximo sus ventajas sin preocuparse por sus desventajas.

La mayoría de los lectores de códigos de barras son usados en supermercados o en la industria del comercio en general. Su protocolo de salida comúnmente es serial (RS-232). Las controladoras por lo general no poseen este tipo de comunicación, por lo que se debe hacer una etapa de conversión entre ellos o en su defecto conseguir lectoras de códigos de barras que son recomendadas por el fabricante de la controladora.

**d. Lectores biométricos**

Es una tecnología que se viene usando más frecuentemente en los últimos años. En empresas que requieran un alto nivel de seguridad, son una primera opción. Se basan en parámetros biométricos o biometrías y parten del principio de reconocer algún rasgo físico o comportamiento de la persona, por lo que es intransferible y el usuario siempre lo lleva consigo.

**e. Lectoras de protocolo 1-Wire®**

El protocolo 1-Wire es un sistema de buses de comunicación para dispositivos que provee alimentación, señalización y datos en un solo canal, de allí su nombre (1-Wire: una sola línea). En cuanto a lectoras para sistemas de control de acceso que usen este protocolo se tienen los iButton®, que son dispositivos que poseen un número serial único de 64-bits.

#### **f. Lectores de proximidad:**

Las tarjetas de proximidad fueron introducidas al mercado aproximadamente en la década de los noventas, momento en el que el control de accesos era dominado por las tecnologías de tarjetas magnéticas y de efecto Wiegand.

Ciertos fabricantes lanzaron productos de proximidad al mercado haciendo que sus protocolos de comunicación, de cierta manera, emularan los protocolos que se habían venido usando (lectoras de bandas magnéticas o de efecto Wiegand). Gracias a esto, solo era necesario reemplazar la lectora más no todo el sistema completo.

La tecnología de proximidad se basa en tener tarjetas pasivas, así como su nombre lo indica, que no posean alimentación propia. Las lectoras se encargan de emitir una señal para alimentar las tarjetas, las cuales poseen una bobina y un chip el cual contiene la información del número grabado en ella.

En el siguiente esquema se puede resaltar la secuencia de lectura de la información del chip de una tarjeta de proximidad:

- La lectora genera un campo electromagnético que le entrega energía.
- La tarjeta es colocada a la distancia adecuada para que reciba, por lo menos, la cantidad mínima necesaria de carga.
- Una vez alcanzado este umbral de energía comienza a transmitir la información que tiene programada en su chip.
- El lector decodifica el mensaje recibido de la tarjeta y de acuerdo a su configuración transmite el código recibido de la tarjeta al panel de control de accesos.

En la figura 3, se puede observar una imagen del esquema mencionado anteriormente.

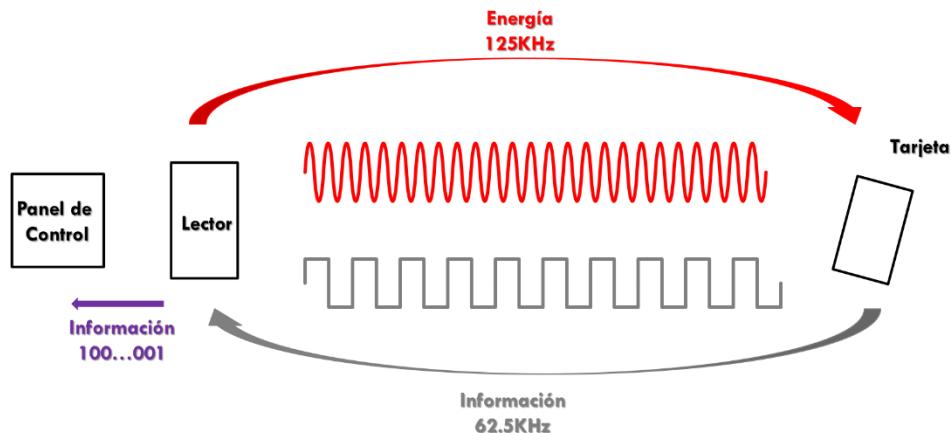


Figura 3 – Secuencia de lectura de la información de una tarjeta

Los fabricantes de lectores de proximidad ofrecen diferentes rangos de lecturas, generalmente se concentran en tres categorías:

- **Bajo rango (8 a 12cm):** son utilizados en el mayor de los casos, para aplicaciones de puertas residenciales u oficinas.
- **Rango medio (25 a 30cm):** se utilizan para aplicaciones en la que las tarjetas se buscan leer sin necesidad de sacarlas de carteras o maletines; son muy usadas en aplicaciones domiciliarias o estacionamientos de bajo costo.
- **Largo alcance (más de 50cm):** generalmente son usadas en estacionamientos.

Uno de los protocolos comúnmente usados para los elementos de proximidad, es el protocolo Wiegand. En la siguiente sección se explicará cómo es utilizado para lograr la comunicación entre el lector y la unidad controladora.

- **Protocolo Wiegand**

Este protocolo establece líneas de datos, alimentación y señalización. Estas últimas son las utilizadas para manejar los indicadores luminosos o sonoros (leds y beepers).

*“... es una forma de comunicación que fue definida e introducida al mercado por la empresa Sensor (hoy HID®), hace ya más de 15 años, es esencialmente unidireccional y permite el traspaso de datos entre una lectora y una controladora.”* (Consentino, 2009b).

Las líneas de datos se denominan Data0 y Data1, estas líneas siempre permanecen en estado lógico “1” y cuando se necesita enviar un bit de dato, se coloca la línea respectiva en “0” lógico (colocando a 0V dicha línea) de una duración mínima de 50us.

Por ejemplo, si se requiere enviar un “1” lógico, se lleva a cero (0V) la línea Data1 por una duración de 50us y si se requiere enviar un “0” lógico, se lleva a cero la línea Data0 por la misma duración. Cabe destacar que la duración entre pulsos o el tiempo de descanso mínimo entre un bit de dato y otro no debe ser menor a 1ms.

Un ejemplo de la transmisión en este protocolo se puede observar en la siguiente figura.

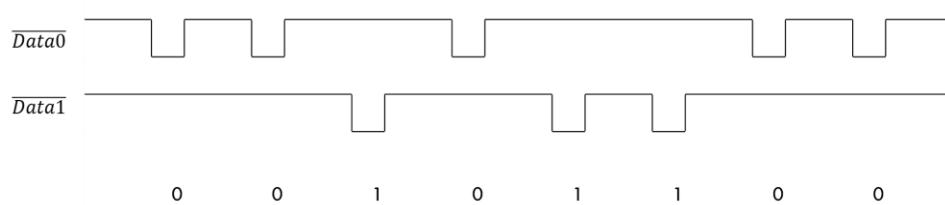


Figura 4 – Ejemplo de transmisión de datos en el protocolo Wiegand

Este ejemplo, en formato Wiegand, se denominaría Wiegand-8, ya que se están transmitiendo 8 bits de información. El estándar considerado para este protocolo es el Wiegand-26, se trata de una cadena de 26 bits en los cuales el primer bit transmitido es el bit de paridad par de los próximos 12 bits; después los siguientes 8 bits son los llamados “FacilityCode” y representan un número que abarca valores entre 0 y 255; los próximos 16 bits son llamados “Identificación” y abarca valores entre 0 y 65535 y el último bit transmitido es el bit de paridad impar de los últimos 12 bits.

En la siguiente figura se puede apreciar con mayor claridad lo mencionado.

<b>Formato General</b>	<b>PFFFFFFFDDDDDDDDDDDDDDDDDI</b>
<b>Paridad Par</b>	<b>PXXXXXXXXXXXXXDDDDDDDDDDDDDI</b>
<b>Facility Code</b>	<b>PFFFFFFFDDDDDDDDDDDDDDDDDI</b>
<b>Identificación</b>	<b>PFFFFFFFDDDDDDDDDDDDDDDDDI</b>
<b>Paridad Impar</b>	<b>PFFFFFFFDDDDDDXXXXXXDDDDDI</b>

*Figura 5 – Formato del Wiegand-26*

Realizando un ejemplo un poco más visual, suponiendo que una tarjeta contiene el código siguiente: 11011011011001110011001001

Separándolo en los campos mencionados anteriormente:

1**10110110**11001110011001001

Paridad par: 1

Facility code: **10110110** = 0xB6 (hexadecimal) = 182 (decimal)

Identificación: 1100111001100100 = 0xCE64 (hexadecimal) = 52836 (decimal)

Paridad impar: 1

Por lo que, el valor completo de esta tarjeta quedaría como: 182-52836.

Para entender el cálculo de la paridad, retomando el valor completo de la tarjeta:

1**1011011011001110011001001**, la paridad par se calcula sobre los 12 bits que están en color rojo, como este bit es uno (1), la cantidad de unos contenidos en esta cadena debe ser impar, lo cual es correcto ya que hay siete (7) y se debe agregar el primer bit para cumplir con la regla. Por otra parte, se puede observar que el bit de paridad impar es uno, por lo que la cuenta de unos en los últimos 12 bits debe ser par

y es así, ya que hay seis (6), por lo que se debe agregar el bit de paridad 1, para cumplir con la regla.

Cabe resaltar que el Wiegand es tanto un protocolo de comunicación como una forma de codificar una tarjeta. Las tarjetas pueden estar codificadas en el formato Wiegand y las lectoras se comunican con las unidades controladoras siguiendo el protocolo del mismo nombre. El detalle está en que si una tarjeta se encuentra codificada en formato Wiegand-XX bits, se está afirmando que, si el lector también se comunica con el protocolo Wiegand, esta comunicación será de los mismos XX bits.

## **2.2 Sistemas Embebidos (SE)**

### **2.2.1 Definición**

Un sistema embebido es un sistema computador destinado a una aplicación en particular. Siendo así una combinación de hardware y software específicos para una tarea por lo que son optimizados para la misma. (Perez, 2009).

### **2.2.2 Características**

- Tienen la capacidad de interactuar con su entorno bien sea de manera directa recibiendo valores de sensores y controlando señales o comunicándose con otros dispositivos.
- Poseen un funcionamiento específico ya que suelen ejecutar un programa para una tarea específica de forma repetitiva.
- Su métrica de diseño (medidas de características y/o implementación) requiere que sean poco costosos, posean un tamaño reducido y tengan un bajo consumo energético.
- Deben ser reactivos y en tiempo real ya que muchos de los sistemas embebidos se encuentran interactuando con el entorno que los rodea y deben tomar medidas o realizar cálculos para enviar respuestas rápidas.

### **2.2.3 Firmware**

Son todas aquellas rutinas de software que se encuentran almacenadas en una memoria no volátil (Flash, ROM, EEPROM, entre otros), muchas veces realizado para un hardware específico, por lo que se puede decir que se encuentra inmerso o muy acoplado a él.

### **2.2.4 Exigencias**

Entre las mayores exigencias de un SE, se tienen la confiabilidad, disponibilidad y seguridad.

#### **2.2.4.1. Confiabilidad**

Se refiere a la probabilidad de que un sistema cumpla con sus requerimientos siempre y cuando opere bajo sus condiciones establecidas de funcionamiento

#### **2.2.4.2. Disponibilidad**

Se refiere a la probabilidad de que el sistema esté operando correctamente en un instante de tiempo.

#### **2.2.4.3. Seguridad**

Se refiere a la probabilidad de que condiciones que puedan afectar el funcionamiento del sistema no ocurran, independientemente de si el sistema está o no, cumpliendo con su función. Especialmente hablando de daños, deterioro por el medio ambiente o el factor humano.

### **2.2.5 Fallas**

Aún bajo perfectos desarrollos de software/firmware, los sistemas embebidos se encuentran en ambientes externos lo cual puede afectar directamente el correcto funcionamiento. Entre estos factores pueden encontrarse altas temperaturas, interferencia, error humano, entre otros.

## 2.2.6 Entradas y Salidas

Debido a sus características, los sistemas embebidos deben interactuar con otros factores externos bien sea midiendo (o adquiriendo) señales del ambiente, leyendo datos de otras unidades o realizando acciones sobre el mismo.

Las entradas y salidas pueden ser manejadas de dos maneras: digitales o analógicas.

### 2.2.6.1. Entradas y Salidas Analógicas

Los valores intercambiados pueden tomar varios niveles, por lo que es necesario disponer de conversores Digital/Analógico (DAC) o Analógico/Digital (ADC). Un ADC convierte una tensión en un pin de entrada en su representación en una escala numérica o valor digital, la resolución del convertidor es el que ayudará a realizar una mejor representación del valor, por ejemplo un canal de 10 bits, permitirá representar valores desde 0 hasta 1023 ( $(2^{10}) - 1$ ).

Un DAC se encarga de convertir un valor digital a un valor de tensión o corriente en un terminal de salida, muchos sensores se manejan de esta manera (temperatura, humedad, luz, entre otros) así como también algunos actuadores (parlantes, motores).

### 2.2.6.2. Entradas y Salidas Digitales

Es el formato en el cual se intercambian unos (1) y ceros (0) lógicos, niveles lógicos de tensión, los cuales se interpretan como valores de referencia de niveles de tensión. Es el utilizado primordialmente para implementar protocolos de comunicación (UART, USB, Serials, entre otros). Estas E/S son muy útiles para controlar algunos dispositivos que requieran un solo nivel de referencia (encender o apagar un led) o para tomar algún valor o leer un estado (revisar si un botón ha sido presionado).

## 2.2.7 Lectura y Escritura

Existen ciertos paradigmas o modelos al momento de implementar la lectoescritura sobre las entradas y salidas, entre ellos se tienen:

### **2.2.7.1. Polling**

El polling se refiere a un uso exhaustivo de los ciclos de CPU para estar constantemente revisando o consultando el valor de alguna entrada específica. El principal problema de este método es que las constantes consultas degradarían el rendimiento del equipo ralentizándolo si no se tiene cuidado con la implementación.

### **2.2.7.2. Interrupciones**

Las interrupciones son aquellas en las que el CPU es notificado, de manera externa, que ha ocurrido algún cambio en un pin o variable de interés y, a nivel de software, se ejecuta una rutina de atención a la interrupción.

Cuando se trabaja con interrupciones se deben tomar en cuenta aspectos como la latencia de la interrupción, el cual es el tiempo entre la ocurrencia y la ejecución de la misma y si se está trabajando con sistemas de tiempo real reducir a lo mínimo este parámetro.

Existen otras formas de verificar las variables, la cual es a través de interrupciones por temporizadores, en las cuales cada cierto tiempo se realiza la consulta del estado de un periférico o variable.

## **2.2.8 Comunicación**

Existen diversas maneras en las cuales los sistemas embebidos se pueden comunicar, entre ellas se tienen la comunicación serial (RS-232/485, SPI, I2C, entre otros), puerto paralelo, actualmente se están utilizando protocolos como: USB, Ethernet, Wifi, redes de dispositivos, entre otros.

## **2.2.9 Formas de implementar SE**

Existen diversas maneras de implementar los SE, se debe realizar un estudio y determinar cuál implementación se adapta mejor al proyecto, entre estas

implementaciones se tienen aquellas basadas en microcontroladores, sistemas sobre un chip (SOC), computadoras de una única tarjeta, híbridos u otros.

### **2.2.9.1. Microcontroladores**

Para entender mejor lo que es un microcontrolador, se deben conocer conceptos como lo que es un microprocesador, el cual es un CPU de un solo circuito integrado; un computador es un CPU más una memoria y puertos de entrada y salida; un sistema computador es un computador más la conexión de periféricos.

Ahora, un microcontrolador es un sistema en el cual el microprocesador, el soporte, la memoria y las entradas y salidas se encuentran dentro de un mismo encapsulado.

Son dispositivos de fácil uso, su costo es bajo, flexibles en cuanto a implementación y de pequeño tamaño. Se pueden clasificar en tres grandes grupos, microcontroladores de 8 bits, 16-32 bits y procesadores de señales digitales (DSP).

Los microcontroladores poseen diversas características, entre las cuales resaltan la capacidad de poder utilizar interrupciones, temporizadores, varios protocolos de comunicación, algunos vienen con ADC y DAC, entre otros.

### **2.2.9.2. IO Boards**

Son placas que están orientadas a trabajar con entradas y salidas que permitan realizar diseños embebidos en ellas, básicamente se usan para realizar prototipos e incluyen todo lo necesario para comenzar a desarrollar en ellas (microcontrolador, conectores de E/S, memoria, entre otros).

Entre algunas de estas tarjetas destacan:

- Arduino®, en todas sus versiones
- Netduino®, igualmente en todas sus versiones

### **2.2.9.3. Sistemas sobre un chip (SOC)**

Se refiere a la integración de todos los componentes de un computador u otro dispositivo electrónico en un solo integrado; su principal diferencia con un microcontrolador es la memoria disponible y el soporte de una unidad para manejo de la memoria (MMU).

Permiten ejecutar sistemas operativos tradicionales y cada uno de los SOC está orientado a un tipo de aplicación determinado. Permiten implementar sistemas mucho más pequeños y de altas prestaciones.

Las desventajas al implementar un SE en un SOC son que se posee una cantidad de memoria RAM limitada, la velocidad del CPU es inferior a sistemas de escritorio pero por otra parte se tiene que poseen varios buses de interconexión (I2C, SPI, USB) así como comunicación serial y varios puertos de E/S.

Entre las arquitecturas más utilizados para la implementación de sistemas embebidos, de tipo SOC, se tienen los ARM y los x86.

### **2.2.9.4. Computadoras embebidas de una única tarjeta (SBC)**

Estas son computadoras que están completas y fabricadas en una única placa de circuito, el diseño está centrado en un microprocesador con RAM, almacenamiento, entradas y salidas y otras características que son necesarias para que sea una computadora completamente funcional en una sola placa.

En la actualidad, existen una gran variedad de SBC basadas en SOC, ya que estos últimos no pueden trabajar por si solos, requieren alimentación, conectores, y otros elementos.

Los SBC son más pequeños y energéticamente más eficientes, además de contar con conectores de entradas y salidas que un sistema tradicional no posee. Muchas veces

son utilizados para realizar prototipos rápidos de soluciones ya que existen SBC que se adaptan a las necesidades del sistema embebido a realizar.

Algunos de los SBC que existen en la actualidad son:

**- Raspberry Pi (posee varias versiones)**

El Raspberry Pi es una computadora del tamaño de una tarjeta de crédito la cual puede ser conectada a una televisión y un teclado. Es una pequeña computadora que puede ser usada para proyectos electrónicos y para muchas otras cosas que la computadora de escritorio puede hacer.

*Tabla 1- Comparación de las diferentes versiones del Raspberry Pi*

*Fuente: <http://downloads.element14.com/raspberryPi1.html> (traducida)*

Technical Features		
	Model A	Model B
<b>Chip</b>	Broadcom BCM2835 SoC full HD Procesador para Aplicaciones multimedia	Broadcom BCM2835 SoC full HD Procesador para Aplicaciones multimedia
<b>CPU</b>	Procesador para Aplicaciones ARM1176JZ-F a 700 MHz de Baja Potencia	Procesador para Aplicaciones ARM1176JZ-F a 700 MHz de Baja Potencia
<b>GPU</b>	Dual Core VideoCore IV® Multimedia Co-Processor	Dual Core VideoCore IV® Multimedia Co-Processor
<b>Memoria</b>	256MB SDRAM	512MB SDRAM
<b>Ethernet</b>	Ninguno	Conector RJ45 para Ethernet 10/100
<b>USB 2.0</b>	Un solo conector USB 2.0	Doble conector USB

---

<b>Salida de Video</b>	HDMI (rev 1.3 & 1.4) Composite RCA (PAL and NTSC)	HDMI (rev 1.3 & 1.4) Composite RCA (PAL and NTSC)
<b>Salida de Audio</b>	Conector de 3.5mm, HDMI	Conector de 3.5mm, HDMI
<b>Almacenamiento sobre la tarjeta</b>	Ranura para tarjetas SD, MMC, SDIO	Ranura para tarjetas SD, MMC, SDIO
<b>Sistema Operativo</b>	Linux	Linux
<b>Dimensiones</b>	8.6cm x 5.4cm x 1.5cm	8.6cm x 5.4cm x 1.7cm

### - Intel® Galileo

El Galileo es una tarjeta basada en el procesador Intel® Quark SoC X1000, un sistema sobre un chip de la clase de los Intel® Pentium®. Es la primera tarjeta basada en la arquitectura de Intel® diseñada para ser compatible tanto en software como en hardware a los “shields” de Arduino, con lo que realizar prototipos rápidos con esta tarjeta es muy factible.

*Tabla 2 - Especificaciones de los Intel® Galileo*

*Fuente: <http://www.intel.la/content/www/xl/es/do-it-yourself/galileo-maker-quark-board.html> (traducida)*

<b>Galileo v1</b>	
<b>Procesador</b>	Intel Quark X1000
<b>Velocidad</b>	
<b>Máxima de Procesamiento</b>	400MHz
<b>Pines</b>	6
<b>Analógicos</b>	
<b>Pines Digitales</b>	14
<b>Memoria</b>	256MB DDR3, , ranura para tarjeta microSD (hasta 32GB)
<b>USB</b>	HS USB 2.0 Client Port, LS/FS/HS USB 2.0 Host Port
<b>Interfaces Soportadas</b>	2x UART, 6x PWM, LCD, GPMC, MMC1, 2x SPI, 2x I2C,

Convertidor A/D, Red Ethernet RJ45 10/100, Puerto mini PCI-e
---

### - BeagleBoard (posee varias versiones)

Las “Beagles” son pequeñas computadoras (*open-hardware, open-software*) que se pueden conectar a cualquier cosa que se tenga en la casa, poseen alta funcionalidad en un pequeño tamaño ya que pueden ser usadas para cualquier aplicación pensada.

Pueden manejar tareas similares a las de una computadora de escritorio.

*Tabla 3- Comparación de las diferentes versiones del BeagleBoard*

*Fuente: <http://beagleboard.org/boards> (traducida)*

	BeagleBone Black	BeagleBone	BeagleBoard-xM	BeagleBoard
<b>Procesador</b>	AM3358 ARM Cortex-A8	AM3358 ARM Cortex-A8	DM3730 ARM Cortex-A8	OMAP3530 ARM Cortex-A8
<b>Velocidad Máxima de Procesamiento</b>	1GHz	720MHz	1GHz	720MHz
<b>Pines Analógicos</b>	7	7	0	0
<b>Pines Digitales</b>	65 (3.3V) 512MB DDR3 (800MHz x 16), 2GB (4GB on Rev C) almacenamiento interno con eMMC, ranura para tarjeta microSD	65 (3.3V) 256MB DDR2 (400MHz x 16), ranura para tarjeta microSD	53 (1.8V) 512MB LPDDR (333MHz x 32), ranura para tarjeta microSD	24 (1.8V) 256MB LPDDR (333MHz x 32), ranura para tarjeta microSD
<b>Memoria</b>	HS USB 2.0 Client Port, LS/FS/HS USB 2.0 Host Port	HS USB 2.0 Client Port, LS/FS/HS USB 2.0 Host Port	4 Port LS/FS/HS USB Hub, HS USB 2.0 OTG Port	USB HS Host Port, HS USB 2.0 OTG Port
<b>Video</b>	microHDMI, cape add-ons	cape add-ons	DVI-D (via HDMI connectors), S- Video	DVI-D (via HDMI connectors), S- Video
<b>Audio</b>	microHDMI, cape add-ons	cape add-ons	3.5mm stereo jack	3.5mm stereo jack
<b>Interfaces Soportadas</b>	4x UART, 8x PWM, LCD, GPMC, MMC1, 2x SPI, 2x I2C, A/D	4x UART, 8x PWM, LCD, GPMC, MMC1, 2x SPI, 2x I2C,	McBSP, DSS, I2C, UART, LCD, McSPI,	McBSP, DSS, I2C, UART, McSPI, PWM, JTAG

Converter, 2xCAN Bus, 4 Timers	A/D Converter, 2xCAN Bus, 4 Timers, FTDI USB to Serial, JTAG via USB	PWM, JTAG, Camera Interface
--------------------------------	--	-----------------------------

### - pcDuino (posee diferentes versiones)

Es una pequeña plataforma tipo computador, de alto desempeño y bajo costo que puede correr un sistema operativo como Ubuntu y Android ICS (ambos basados en GNU/Linux). Su pantalla puede ser visualizada a través de su conector de video HDMI, el cual puede ser conectado a cualquier televisor o monitor que posea esta interface.

Está pensado para la comunidad que desarrolla en código abierto, al ser basado en Linux así como también para proyectos electrónicos sencillos o que requieran un buen hardware ya que sus conectores permiten realizar la integración con los “*shields de Arduino*” (tarjetas diseñadas para ser apiladas sobre el Arduino) para la realización de diversas soluciones.

*Tabla 4- Comparación de las diferentes versiones del pcDuino*

**Fuente:** <http://www.pcdduino.com/pcduino-family/> (traducida)

	pcDuino v1	pcDuino v2	pcDuino v3
<b>CPU</b>	Allwinner A10 1GHz ARM Cortex A8	Allwinner A10 1GHz ARM Cortex A8	Allwinner A20 1GHz ARM Cortex A7 Dual Core
<b>GPU</b>	OpenGL ES2.0, OpenVG 1.1 Mali 400 core	OpenGL ES2.0, OpenVG 1.1 Mali 400 core	OpenGL ES2.0, OpenVG 1.1, Mali 400 Dual Core
<b>DRAM</b>	<b>1GB</b>	<b>1GB</b>	<b>1GB</b>
<b>Almacenamiento</b>	<b>2GB</b> Flash (4GB después del 2/1/2014), ranura para tarjeta microSD de hasta 32GB	<b>2GB</b> Flash (4GB after 2/1/2014), ranura para tarjeta microSD de hasta 32GB	<b>4GB</b> Flash, ranura para tarjeta microSD de hasta 32GB
<b>Video</b>	HDMI	HDMI	HDMI

SO Soportado	Lbuntu 12.04	Lbuntu 12.04	Lbuntu 12.04
	Android	Android	Android
Extensión de Interfaces	2.54mm headers	Arduino (TM) Headers	Arduino (TM) Headers
	10/100Mbps RJ45	10/100Mbps RJ45	10Mbps RJ45
Interfaz de Red	Extensión USB WiFi (no incluida)	WiFi	WiFi
Alimentación	5V, 2000mA	5V, 2000mA	5V, 2000mA

## 2.2.10 GNU/Linux

En la actualidad, debido a su principal característica de ser gratuito y *open-software*, el sistema operativo Linux es el más utilizado para los sistemas embebidos contra software de Microsoft® (Windows®), DOS, entre otros.

GNU/Linux está compuesto por el kernel (Linux) y herramientas del sistema GNU.

*“GNU es un sistema operativo similar a Unix que es software libre y respeta su libertad. Puede instalar versiones de GNU (más precisamente, sistemas GNU/Linux) que son completamente software libre”.* (Colaboradores, Consultado el 12-08-2014. <https://www.gnu.org/home.es.html>).

El kernel Linux es el mismo utilizado tanto para estaciones de trabajo como para sistemas embebidos la diferencia está en que al momento de compilar se incluye el soporte para el hardware a utilizar. Soporta numerosas arquitecturas, arm, arm64, avr32, x86, entre otros. Entre los diferentes sistemas embebidos se tienen Meego, Android, Angstrom, Ubuntu ARM, entre otros.

Las principales ventajas al usar GNU/Linux en sistemas embebidos son el poder reutilizar una gran cantidad de librerías de software que ya se encuentran diseñadas por colaboradores, uso de lenguajes de programación de mayor nivel de abstracción, al ser muchas veces código abierto, existe la posibilidad de editar y adaptarlo a los requerimientos.

Hoy en día los SE poseen un largo alcance, gracias al desarrollo de hardware con mayores atributos con lo que pueden ejecutar un SO de propósito general y no dejan de ser económicamente factibles. Existe un gran mercado con mayores requerimientos para los sistemas embebidos. Es por ello que existe un gran auge en cuanto al desarrollo de este tipo de plataforma y soluciones.

## 2.3 Servidor Web

### 2.3.1 Definición de Servidor

Un servidor es básicamente un computador en el que siempre se están ejecutando programas y se encuentra a la espera de solicitudes por parte de otros, bien sea otras computadoras o usuarios y les devuelve una respuesta acorde a lo solicitado.

*“...es un ordenador o máquina informática que está al “servicio” de otras máquinas, ordenadores o personas llamadas clientes y que le suministran a estos, todo tipo de información”.*

*“...en informática será un ordenador u otro tipo de dispositivo que suministra una información requerida por unos clientes (que pueden ser personas, o también pueden ser otros dispositivos como ordenadores, móviles, impresoras, etc.).”* (Sierra, M. 2013).

## 2.3.2 Servidor Web

Un servidor web, es un programa informático que siempre se encuentra en ejecución dentro de un **equipo servidor**. Se mantiene a la espera de peticiones de un cliente o usuario a través de un navegador web, y este se encarga de manejar estas peticiones y realizar respuestas de manera adecuada, devolviendo en este caso una página web con la información solicitada.

En la actualidad existen una gran cantidad de aplicaciones para servidores web, entre ellos se tienen IIS® que pertenece a Microsoft®, Cherokee, Monkey http, Lighttpd, Thttpd y Apache. Siendo este último el mayormente elegido debido a que cuenta con un gran grupo de voluntarios que trabaja en el mejoramiento del mismo, es software libre y cuenta con el apoyo de la fundación Apache. Es capaz de realizar implementación de protocolos actualizados y mostrar la información con gran rapidez.

### 2.3.2.1. El servidor web apache

Se denomina “El proyecto de servidor HTTP Apache”, el cual fue desarrollado como código fuente-aberto (*open-source*) para sistemas operativos modernos incluyendo UNIX y Windows®. El objetivo de este proyecto es el de proveer un servidor seguro, eficiente y extensible que a la vez provea de servicios HTTP que estén dentro de los lineamientos del estándar con el mismo nombre. Ha sido el servidor web más popular en el internet desde hace ya 17 años. (The Apache Software Foundation, 2012).

Al momento de realizar este trabajo la última versión disponible y estable de este software es la 2.4.10 en la cual se reparan algunos errores pequeños y se incrementa la seguridad del sistema.

A través de la adaptación de diversos módulos, se pueden elaborar páginas web dinámicas en el servidor, como lo son Python, Perl, Ruby, PHP y otros.

## CAPÍTULO III. DEFINICIÓN Y DESCRIPCIÓN DEL HARDWARE

### 3.1 Definición del hardware

Un servidor web como sistema de control de acceso, debe contar con varios elementos que, interconectados, lleven a un funcionamiento general. Entre estos se tienen la unidad que controla los dispositivos externos de identificación tales como lectoras de tarjetas, teclados pin, entre otros y el servidor que debe ser capaz de leer los datos provenientes de la unidad de control así como gestionarlos para que sean mostrados a través de una aplicación (bien sea de escritorio o web) y también dar respuesta para que la unidad de control pueda ejecutar alguna acción así como todos aquellos elementos que de una u otra manera ingresan datos al equipo o viceversa.

Básicamente consta de cuatro grandes bloques que a su vez se dividen en otros más pequeños, los cuales se explicarán con detenimiento más adelante indicando la selección, sus características principales y sus ventajas al ser usados.

En la siguiente figura se puede observar el diagrama de bloques general que conforma el equipo.

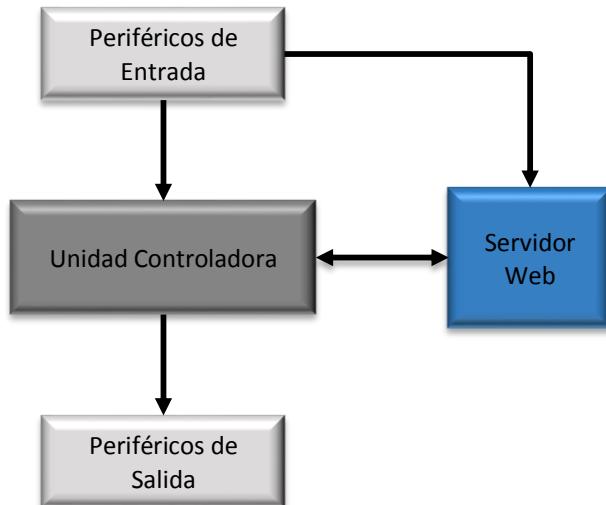


Figura 6- Diagrama de Bloques General del Equipo

El equipo diseñado es un servidor web como sistema de control de acceso, realizado mediante el uso de diversos módulos tanto diseñados y elaborados por cuenta propia, así como comerciales en casos donde se requiera minimizar errores y robustez.

A continuación se explicará detalladamente cada uno de los bloques mostrados en la figura anterior.

### **3.2 Descripción del Hardware**

#### **3.2.1 Periféricos de Entrada y Salida:**

##### **3.2.1.1 Periféricos de Entrada:**

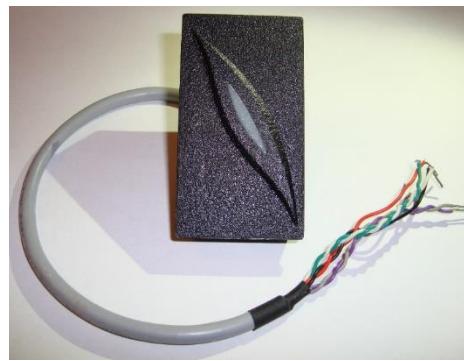
Se conocen como todos aquellos elementos que de una manera u otra ingresan datos al equipo.

El elemento principal que ingresa datos al sistema es el lector KR101E, de tarjetas de proximidad por radiofrecuencia (RFID, por sus siglas en inglés) que funciona a 125kHz, en formato Wiegand-26 (Capítulo II, sección 2.2.1 iterado f., para mayor información acerca de este formato). Posee características muy atractivas para el momento de su elección para realizar el proyecto, entre las cuales destacan que puede ser usado tanto en interiores como exteriores, fácil instalación, su rango de lectura es de hasta 10cm, posee indicadores luminoso y sonoro y una de sus grandes ventajas es que realiza la lectura a 125kHz que es una frecuencia bastante comercial para este tipo de aplicaciones.

Su rango de funcionamiento es desde 6V hasta 14V, pudiendo ser conectado con una fuente de 5V, con un consumo máximo de 70mA lo cual es totalmente favorable ya que la mayor parte de la electrónica se encuentra a este nivel de tensión.

Este lector es el encargado de obtener la información alojada en la tarjeta, la cual se encuentra codificada en formato binario y más adelante será traducida por la unidad controladora para que sea más sencillo trabajar con este valor.

En la figura 7, se puede observar el lector KR101E.



*Figura 7- Lector de tarjetas de proximidad KR101E a 125KHz*

El código de colores de los cables se encuentra indicado en la parte posterior del lector, los cuales indican:

*Tabla 5- Código de colores del lector de proximidad*

<b>Rojo</b>	DC+ (6-14V) Capaz a 5V
<b>Negro</b>	GND
<b>Gris</b>	Control del Led
<b>Morado</b>	Beeper
<b>Blanco</b>	Wiegand Data 0 (D0)
<b>Verde</b>	Wiegand Data 1 (D1)

Otro de los elementos que ingresa datos al equipo es el módulo de reloj en tiempo real (RTC, por sus siglas en inglés). Con el cual el servidor web se sincroniza para tomar la fecha y hora, para así poder realizar un registro válido en el sistema. Este módulo será mejor explicado más adelante, en la sección correspondiente al servidor web.

### **3.2.1.2 Periféricos de Salida:**

Son conocidos como los elementos externos capaces de emitir alguna señal fácilmente identifiable para el usuario.

El lector de tarjetas además de ser un periférico de entrada también forma parte de un periférico de salida, ya que posee un identificador luminoso (led) que le indica al usuario cuando está autorizado o no para ingresar a la instalación y posee a su vez un indicador sonoro (beeper) que emite un sonido en el momento que se está leyendo la tarjeta.

En la figura siguiente se pueden observar estos elementos, a la izquierda el estado en espera del lector (led rojo) y a la derecha acceso autorizado (led verde), abajo el beeper, que se encuentra en la parte posterior.



Figura 8- Lector en estado de espera (izquierda), autorizado (derecha) y beeper (abajo)

La hembrilla o cerradura eléctrica de la puerta que emite un sonido o zumbido cuando está habilitada para abrir la misma. A continuación se puede observar la hembrilla elegida para este proyecto.



Figura 9- Cerradura eléctrica elegida

Por último la tarjeta de leds indicadores, para mostrarle al usuario que el equipo se encuentra encendido y está habilitada la comunicación serial por los canales Rx y Tx. Este pequeño módulo se puede observar a continuación.

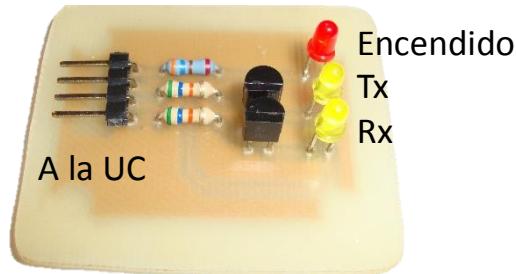


Figura 10- Tarjeta de leds indicadores

### 3.2.2 Unidad Controladora

La unidad controladora es la encargada de captar los datos provenientes del lector, traducirlos, convertirlos y enviarlos al servidor web y esperar respuesta de éste para tomar una decisión y realizar, o no, una acción que conlleve a la apertura de la puerta.

Está diseñada en una baquelita cuyo tamaño no excede los 14x12cm; el corazón de esta tarjeta es un microcontrolador PIC de la casa Microchip, 18F4550.

La tarjeta posee diversos conectores para elementos externos, entre ellos se tienen dos conectores para componentes que trabajen bajo los protocolos UART, I2C. Un conector para realizar la conexión hacia un puerto USB. Dos conectores para elementos que se comuniquen a través del protocolo Wiegand. Un conector para la conexión de un pulsador, que por lo general se denomina “botón de solicitud de salida” (RTE, por sus siglas en inglés). Y un conector para la hembrilla, el cual está siendo manejado por un relé de un solo polo y doble tiro (SPDT, por sus siglas en inglés) de 5VDC, este dato es importante para el momento de calcular la corriente que atravesie la bobina para realizar la activación del mismo; el relé posee las siguientes características para la conexión de elementos en sus terminales: 2A @ 30VDC y 1A @ 125VAC, es decir

que partiendo de estas especificaciones se debe seleccionar una hembrilla que cumpla con estas características. La elegida fue una que trabaja a 1.3A @ 12V.

Este conector posee dos modos de conexión los cuales son para elementos que requieran que el relé este normalmente abierto (NA) o normalmente cerrado (NC). A continuación se muestra, en la figura 12, el diagrama de bloques de la unidad controladora y en la figura 11 la unidad controladora ya realizada en baquelita.

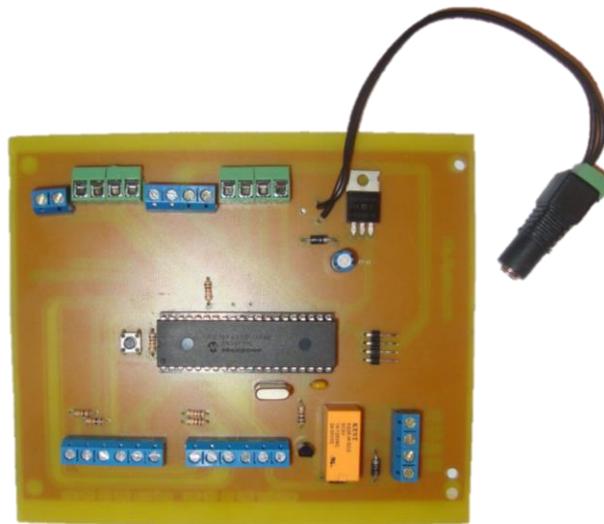


Figura 11- Unidad Controladora final en circuito impreso.

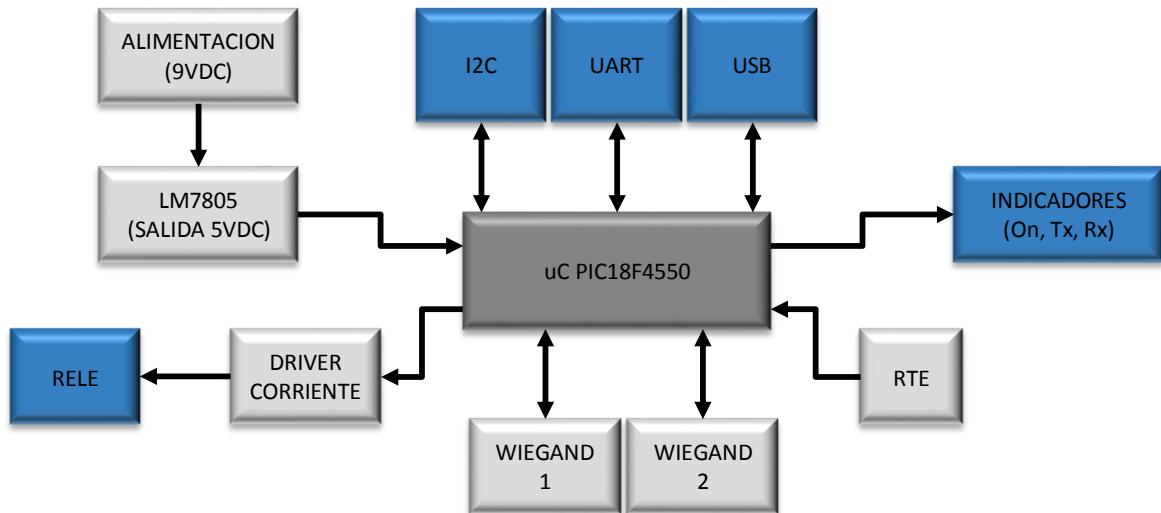


Figura 12- Diagrama de Bloques de la Unidad Controladora.

A continuación se explicará con mayor detalle cada una de las etapas de la tarjeta controladora.

### 3.2.2.1 Alimentación

La alimentación de la tarjeta se realiza a través de un transformador comercial que arroje 9VDC a la salida. Y una corriente mínima de 250mA para poder alimentar todos los componentes externos que se asocien a esta unidad.

Los 9V están directamente conectados a un convertidor de tensión LM7805, el cual es el encargado de convertir y otorgar a la tarjeta la tensión adecuada para su correcto funcionamiento (5VDC). Los elementos externos pueden tomar la alimentación de la tarjeta ya que cada conector, además de los canales de datos, poseen un puerto para alimentación (+5V y masa), o contar con alimentación independiente, siempre y cuando se conecten a la misma referencia de masa.

La figura 13 muestra el bloque de alimentación.

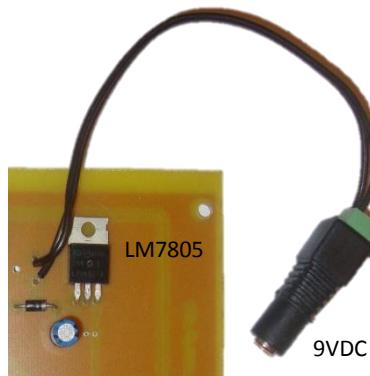


Figura 13- Bloque de alimentación

### 3.2.2.2 Microcontrolador PIC18F4550

El microcontrolador, el corazón de la unidad controladora, es el encargado de recolectar los datos provenientes de la lectora de proximidad, traducirlos y convertirlos en un formato entendible para el humano y que sea de fácil manipulación.

Se eligió el microcontrolador PIC18F4550 de la casa Microchip ya que es una unidad potente que puede responder rápido ante los requerimientos. Cuenta con 40 pines de los cuales 4 son de alimentación y el resto son configurables como entradas/salidas multipropósito.

Entre estos pines configurables se tienen puertos dedicados a la comunicación que lo hicieron aún más atractivo para su elección; entre ellos destacan el puerto UART, para establecer la comunicación serial asíncrona, el bus de datos de circuitos inter-integrados (I2C, por sus siglas en inglés), la comunicación por el protocolo USB, diversas entradas para establecer rutinas por cambios de estado (interrupciones externas) que eran necesarias para realizar el firmware y la rutina de lectura del protocolo Wiegand. Así como varios pines multipropósito de entrada/salida para los demás elementos que lo requieran.

A continuación se muestra el microcontrolador sobre la tarjeta controladora.

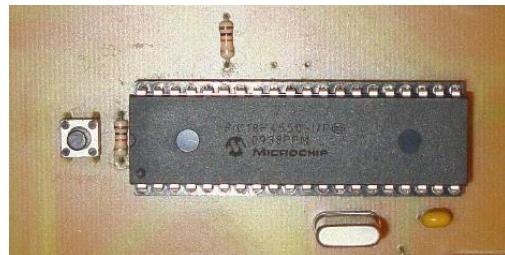


Figura 14- Microcontrolador PIC18F4550

### 3.2.2.3 Módulos de comunicación y conector de botón para salir (RTE)

Los módulos de comunicación presentes en la tarjeta son, como se mencionaron anteriormente, UART, I2C y USB.

Para la comunicación con el servidor, se consideró conveniente la utilización de la comunicación serial a través del módulo UART contenido en el microcontrolador, contando con los dos canales de este protocolo, Tx y Rx. Aprovechando este hecho y que más adelante permitiría la utilización de módulos para comunicarse a través de protocolos RS-232, RS-485, convertidores Seriales a USB, entre otros.

Este protocolo es sencillo de usar y bastante fiable para este tipo de requerimientos. Y cuando se hace uso de los protocolos RS-232 o RS-485 se pueden alcanzar grandes distancias entre la tarjeta y el elemento final de recepción de los datos.

El módulo del bus I2C, aunque para la elaboración de este proyecto no está siendo utilizado, se colocó pensando a futuro en aplicaciones de bajo costo, que no requieran un servidor web para almacenar la información ya que en este bus de comunicaciones, como su nombre lo indica, al ser un bus pueden ser conectados diversos elementos, entre los que destacarían un RTC y memorias EEPROM, que se comuniquen por este bus de datos. Solo bastaría realizar pequeñas modificaciones en el firmware del microcontrolador para realizar rutinas de lectura/escritura en la memoria tomando la fecha y hora del RTC.

De igual manera sucede con el módulo USB, se encuentra implementado por hardware y habilitado por software (las librerías se incluyen) mas no es utilizado para fines de este proyecto, fue colocado pensando en la extensibilidad del proyecto.

La conexión para el botón de la solicitud de salida o simplemente botón para apertura de puerta, es un sencillo circuito que al presionar un botón conectado a esta bornera envía una señal al microcontrolador, activando una interrupción y de esta manera se activa el relé, abriendo la puerta. Es importante recordar que siempre este botón debe encontrarse dentro de las instalaciones así como también, dependiendo del caso, oculto a la vista del público o tengan acceso a él pocas personas (en casos donde sea importante mantener un control del flujo del personal).

Cabe destacar que cada uno de los puertos de comunicación cuenta con borneras para la alimentación de los elementos externos que sean conectados, en la siguiente figura se pueden observar los módulos y sus especificaciones.

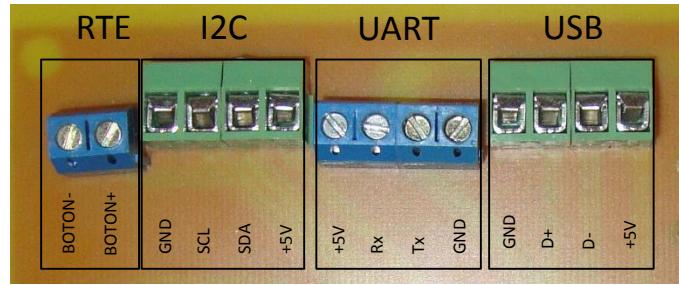


Figura 15- Módulos de comunicación y conexión RTE

### 3.2.2.4 Conectores para entrada Wiegand

La tarjeta controladora fue diseñada para que sea posible la conexión de dos elementos de entrada de datos con protocolo Wiegand. Esto pensado en el hecho de que se requiera un control de flujo de personal (lectores de tarjetas para entrar y salir de la instalación). Pero, para la realización de este proyecto solo se trabaja con uno de los conectores (un solo lector).

En la siguiente figura se pueden observar los conectores en la tarjeta, y a su vez, las indicaciones de cada uno de los bornes.

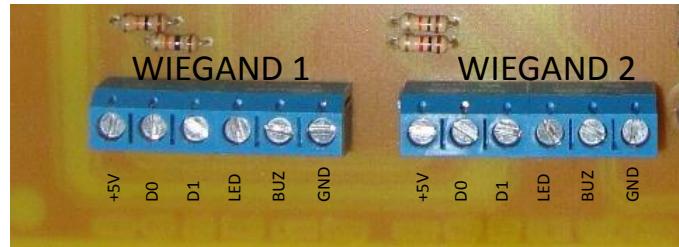


Figura 16- Conectores para entrada Wiegand

### 3.2.2.5 Relé

El modulo del relé, el encargado de activar la cerradura eléctrica consta de una resistencia de  $1\text{k}\Omega$ , un transistor NPN 2N3904, dos diodos, un relé KS2E-M-DC5, y los conectores para esto.

La resistencia y el transistor son los encargados de proporcionarle a la bobina interna del relé la corriente necesaria para que el campo magnético generado en esta sea suficiente como para activar el mecanismo interno.

Los diodos son ubicados para proteger a la tarjeta y la alimentación de las corrientes de retorno provenientes de la descarga de la bobina. Uno ubicado directamente entre la alimentación y la bobina del relé y el otro ubicado entre uno de los bornes del conector (HC) y el común (COM), esto está pensado en elementos que sean manejados por bobinas y puedan existir grandes picos en los cuales la corriente sea tan elevada y no dañe la alimentación o la cerradura colocada entre estos terminales.

El relé KS2E-M-DC5, cuyas especificaciones dicen que trabaja a 5VDC, la bobina disipa una potencia de 0.20W, con estos parámetros se realizó el cálculo aproximado para determinar la corriente de la bobina requerida y de esta manera la resistencia del transistor.

Por otra parte, entre los terminales de este relé se deben conectar elementos que cumplan con las especificaciones del fabricante, las cuales (según la hoja de datos) indican que se pueden conectar elementos cuyo valor máximo sea de 2A @ 30VDC y 1A @ 125VAC.

Para poder visualizar un poco mejor este bloque, en la figura 17 se tiene el diagrama circuital del mismo y en la figura 18 su disposición en la tarjeta.

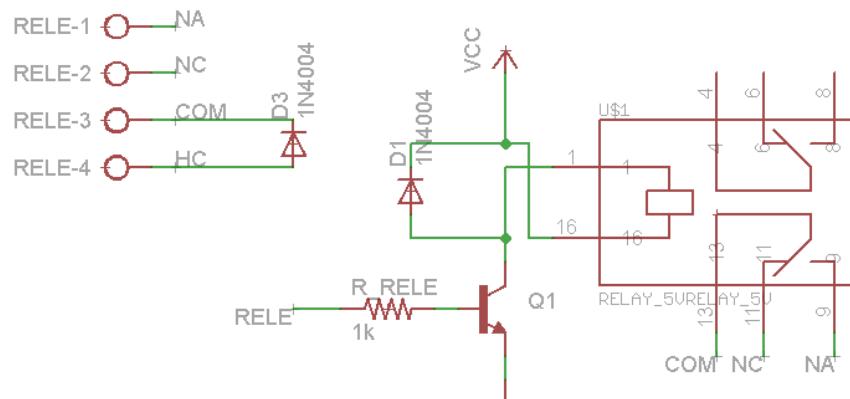


Figura 17- Diagrama circuital del módulo del Relé

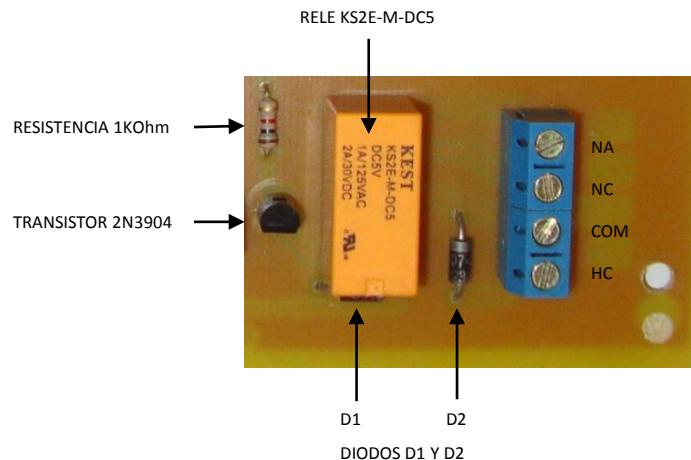


Figura 18- Disposición del módulo del Relé en la Tarjeta

La cerradura eléctrica elegida cumple con los requerimientos de este relé. Para realizar el cálculo de la misma, basta con medir la resistencia de la bobina y como trabaja a 12VDC, se hace un simple cálculo para determinar la máxima corriente que por ella pasa.

Teniendo 12VDC, una resistencia R de  $9\Omega$ , la corriente que atraviesa la cerradura al activarse será:

$$V = I * R$$

$$I = \frac{V}{R}$$

$$I = \frac{12V}{9\Omega}$$

$$I = 1.33A$$

Teniendo una corriente de 1.33A, se puede asegurar que el relé funcionará en condiciones normales ya que el máximo soportado en sus contactos al estar cerrados es de 2A@30VDC.

### 3.2.3 Servidor Web

En sus inicios, el módulo del servidor estaba pensado en realizarse con un microcontrolador PIC24FJ64GA002, de la casa Microchip, así como un controlador de red ENC28J610 de la misma casa, el cual sería el encargado de establecer la comunicación con la red, pero en vista de lo costoso y lo complicado de hacer funcionar estos elementos se optó por una solución mucho más robusta y con menor posibilidad de fallos.

Después de realizado un análisis y estudio de varias opciones (Raspberry PI, BeagleBone, Netduino, Arduino, pcDuino, entre otros), se optó por la tarjeta de desarrollo pcDuino.

El pcDuino es una pequeña plataforma de desarrollo tipo computador (SBC) que puede correr un sistema operativo como Ubuntu o Android. Posee salida de video HDMI así como puertos USB, Ethernet y otros que sirven para comunicación con dispositivos al estilo de un microcontrolador (UART, I2C, SPI, Entradas y Salidas digitales y analógicas, entre otros).

La tarjeta controladora se comunica con el pcDuino a través de la comunicación serial utilizando un módulo de conversión UART a USB a través de un circuito integrado FT232. La implementación del FT232 se debe a que para fines del proyecto se optó por aprovechar el módulo UART del microcontrolador y realizar la comunicación a través del mismo, ya que es un protocolo sencillo de manejar y de fácil implementación además de estar pensado en la posibilidad de conexión de otros módulos que utilicen dicho protocolo para una activación remota vía xbee, transmisión de los datos a través de módulos WiFi, RF24, Bluetooth, entre otros.

En el pcDuino se puede realizar la instalación de diversas aplicaciones para la gestión de servicios web; es de menor costo que una implementación con microcontroladores PIC y, como se ha mencionado anteriormente, mucho más sólida

y robusta, disminuyendo así el error humano al momento del diseño y realización de cualquier tarjeta que contenga una gran cantidad de componentes.

En la siguiente tabla se pueden detallar las especificaciones técnicas más relevantes del hardware del pcDuino.

*Tabla 6- Especificaciones del hardware del pcDuino*

*Fuente: <http://www.pcdduino.com/features/>*

Ítem	Detalles
<b>CPU</b>	1GHz ARM Cortex A8
<b>GPU</b>	OpenGL ES2.0, OpenVG 1.1 Mali 400 core
<b>DRAM</b>	1GB
<b>Almacenamiento en la Tarjeta</b>	2GB Flash, microSD (TF), hasta 32GB
<b>Salida de Video</b>	HDMI
<b>SO</b>	Linux3.0 + Ubuntu 12.04, Android ICS 4.0
<b>Interface de Extensión</b>	2.54mm Headers
<b>Interfaz de Red</b>	10/100Mbps RJ45 y extensión por USB WiFi
<b>Potencia</b>	5V, 2000mA
<b>Tamaño Total</b>	125mm X 52mm

Se puede observar que es un hardware bastante potente y cumple muy bien con lo requerido para el proyecto. Una rápida respuesta, conexión a la red así como una unidad de almacenamiento expandible, lo cual es favorable si se desea que la aplicación migre a una de mayor envergadura.

A continuación se puede observar el hardware del pcDuino en la figura 19.



Figura 19- El pcDuino

**Fuente:** [http://www.pcduino.com/wp-content/uploads/2012/12/pcduino\\_v1.jpg](http://www.pcduino.com/wp-content/uploads/2012/12/pcduino_v1.jpg)

El servidor web no estaría completo sin la incorporación de un elemento que sincronice fecha y hora, lo cual es importante ya que se está llevando a cabo el registro de los accesos realizados a una instalación (para este caso particular el LIDE), por lo que se optó por realizar un módulo de reloj en tiempo real (RTC) para llevar a cabo esta función.

Este módulo consta de un circuito integrado DS1307, tres resistencias de  $10\text{k}\Omega$ , un cristal de 32.768kHz y una batería CR2016 de 3V. Estos elementos, en conjunto son capaces de configurarse a un valor de fecha y hora establecido por el usuario y mantenerlo gracias al respaldo de la batería aun cuando el equipo se encuentre apagado.

La comunicación de este reloj con el pcDuino es a través del bus de datos I2C por los canales SCL y SDA y colocados a la misma referencia GND. En el momento de encender el servidor web, este realiza la solicitud al RTC y toma los datos para así tomar el tiempo correcto. En las figuras 20 y 21 se observan el diagrama circuital del RTC y el módulo implementado, respectivamente.

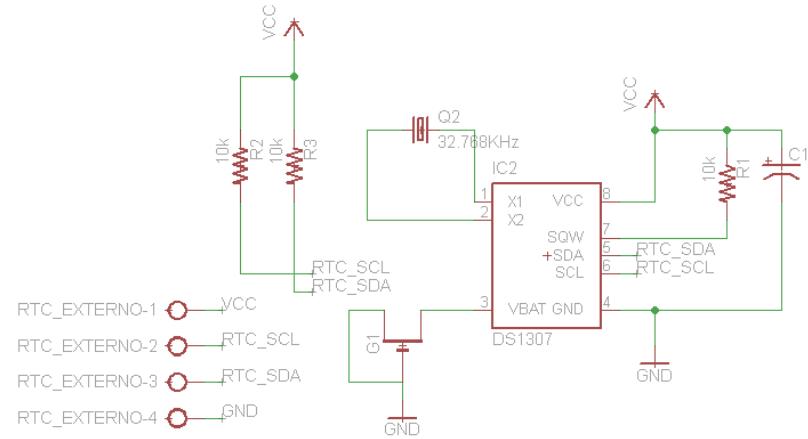


Figura 20 – Diagrama Circuital del módulo RTC



Figura 21 – Módulo RTC implementado

## CAPITULO IV. DEFINICIÓN Y DESCRIPCION DEL SOFTWARE

### 4.1 Definición del Software

Para el caso específico de este proyecto, en el cual se está trabajando con microcontroladores, se requiere la implementación de un programa o firmware para la lectura y traducción de los datos; además de una base de datos en la cual se almacene toda la información referente al servidor (nombres de usuarios, códigos de tarjetas, ingresos realizados, personas autorizadas, entre otros) y a su vez la programación de una página web en la cual se puedan mostrar los datos del servidor web, la cual funcionará como interfaz gráfica de usuarios (GUI).

El firmware del microcontrolador (PIC18F4550), será el encargado de obtener la data proveniente del lector de tarjetas de proximidad, convertirla en un valor decimal y enviarla al pcDuino, así como también esperar respuesta de este último para activar, o no, el circuito del relé. También debe ser capaz de entender cuando se presiona el botón para salir y activar directamente el relé.

En el caso del pcDuino, debe contener un firmware que siempre este esperando data proveniente de la unidad controladora, consultar la data recibida en la base de datos y si encuentra el valor consultado dar una respuesta inmediata a la unidad controladora.

También este firmware debe ser capaz de almacenar el valor consultado, si es encontrado, en alguna tabla de la base de datos con un registro de fecha y hora para indicar que persona está ingresando a la instalación.

Por último en el pcDuino también debe estar alojada una página web, en la cual se muestre la data pertinente al mismo. Esta página servirá como interfaz gráfica para los usuarios autorizados que deseen consultar alguna información.

## 4.2 Descripción del Software

El software del equipo está conformado por:

- El firmware de la unidad controladora, el cual es el responsable de obtener la data proveniente del lector de tarjetas de proximidad, enviar la consulta al servidor y esperar respuesta para realizar una acción, así como también estar en la capacidad de entender cuando se presiona el botón de salida y realizar la acción inmediata de abrir la puerta.
- El programa del pcDuino, que se encarga de obtener la data proveniente de la unidad controladora, realizar las consultas en el servidor de manera tal que le responda si logra encontrar la identificación en la base de datos.
- El servidor web, el cual consta de la base de datos, las funciones del servidor utilizadas para realizar la integración base de datos/página web, que sirve de interfaz gráfica de usuario.

### 4.2.1 Firmware de la unidad controladora

El firmware para el microcontrolador PIC18F4550, fue creado bajo una plataforma de desarrollo que permitiera realizar la compilación en lenguaje C, el cual incluye diversas librerías para el manejo de los periféricos del dispositivo.

Este firmware contiene las diversas rutinas que han sido comentadas anteriormente; partiendo desde la rutina principal en la cual se realiza la configuración automática del equipo (declaración de variables, definición de puertos y funciones), entrar en la rutina de recepción de datos de la lectora de tarjetas, enviar la data vía serial al servidor y realizar la activación del relé para abrir la puerta.

Al ser un software para un sistema de control de accesos, el usuario no puede realizar ninguna configuración personal, ya que esto podría implicar un ataque a la integridad de la seguridad. Quienes pueden realizar configuraciones son administradores, personas calificadas para ello.

#### 4.2.1.1 Rutina principal

Es la rutina principal que se ejecuta al encender el equipo, en la misma se ejecutan de manera automática la configuración del equipo. En esta rutina se espera siempre que suceda algún evento externo para realizar el llamado a otras rutinas.

En ella también es activado un temporizador, que se encarga de enviar, aproximadamente cada 20 segundos, un código único de la controladora para indicarle al servidor que la comunicación se encuentra activa.

Un diagrama de flujo de esta rutina se puede observar a continuación.

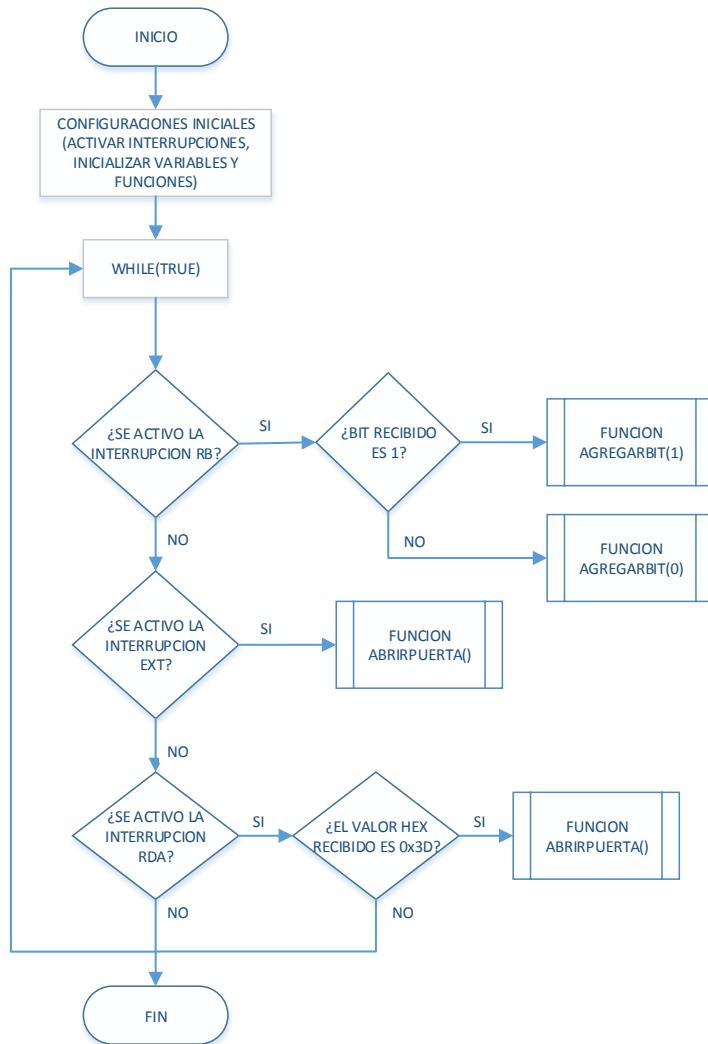


Figura 22- Diagrama de flujo, rutina principal

#### 4.2.1.2 Subrutina Agregar Bits

Es la rutina más delicada que posee el firmware del microcontrolador, debido a que se encarga de que el sistema pueda leer los datos correctos provenientes de la tarjeta de identificación al ser pasada por el lector de proximidad.

Esta se encarga de realizar el llamado de una subrutina, la cual es la encargada de contrastar la data recibida contra la paridad que también está integrada entre los bits de lectura. Para comenzar la explicación de esta rutina, se debe observar principalmente un diagrama de flujos de la misma, para poder entenderla un poco mejor.

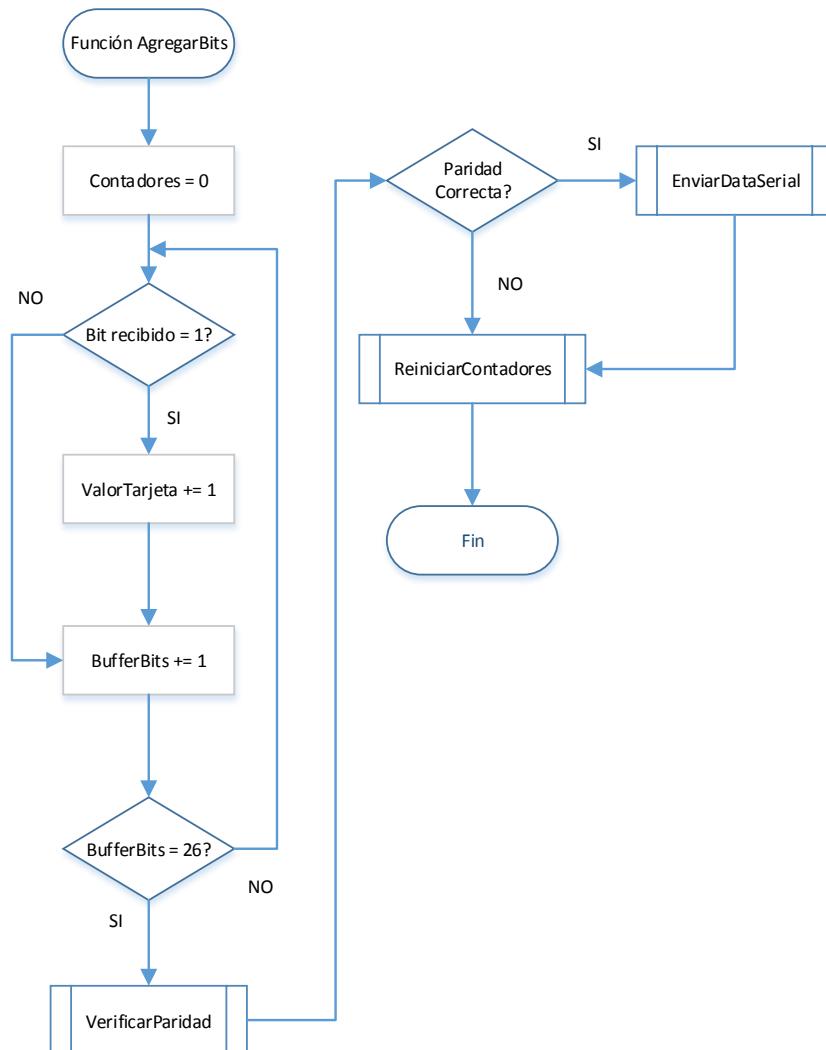


Figura 23- Diagrama de flujo, subrutina agregar bits

Esta rutina comienza teniendo todos los valores en cero, entre los cuales están el valor de la tarjeta, el conteo de bits leídos, entre otros, los cuales se van llenando conforme se vayan recibiendo los datos. Una vez que el conteo de bits leídos ha llegado al máximo (recordando que fue diseñada para un lector Wiegand-26, de los cuales 2 bits son de paridad y los 24 restantes son bits de datos), se procede a verificar la paridad, la cual es una subrutina en la cual se contrasta el primer bit de paridad contra los siguientes 12 bits y de la misma manera para los otros 12 bits.

Siendo verificada la paridad se procede al llamado de la subrutina encargada de acondicionar la data para ser enviada por vía serial al servidor y luego se reinician todos los valores utilizados para que estén disponibles para una nueva lectura.

En caso contrario, que la paridad calculada no coincida con la paridad de la tarjeta, se procede directamente a reiniciar todos los valores y esperar una nueva lectura.

#### **4.2.1.3 Subrutina Verificar Paridad**

Para la verificación de la paridad, se pasan los parámetros necesarios a esta subrutina: bit de paridad par, bit de paridad impar y el valor de la tarjeta (el compendio de los 24 bits de datos).

Recordando el formato de una tarjeta Wiegand-26 (Capítulo II, sección 2.1.4, iterado f.), la tarjeta posee dos bits de paridad, uno para la paridad par de los primeros 12 bits y el otro para la impar de los últimos 12 bits.

Se realiza el conteo de unos que hay en cada grupo de 12 bits. Para el caso de la paridad par, si el número de unos contados es par (o impar) y se verifica el bit de paridad par es cero (o uno), este segmento de la subrutina guarda un parámetro “verdadero”. De igual manera para la paridad impar, se realiza el conteo, si el número de unos es par (o impar) y el bit de paridad es uno (o cero), este segmento de la subrutina guarda un parámetro “verdadero”.

Si al finalizar la rutina ambos parámetros son “verdaderos”, esta devuelve que el valor leído de la tarjeta es correcto y se procede a realizar el llamado de la subrutina para enviar la data vía serial. En caso contrario, se realiza el llamado de la subrutina para reiniciar todos los valores, la cual se encarga de colocar todos los parámetros en cero (0) y esperar una nueva lectura de tarjeta.

#### 4.2.1.4 Subrutina Enviar Data Serial

Como su nombre lo indica, esta rutina se encarga de enviar el valor de la tarjeta vía serial al servidor. El puerto serial está configurado a **9600** baudios por segundo, **8** bits de datos, **sin paridad** y **1** bit de parada.

A esta rutina le llega el valor completo de la data del lector, sin los bits de paridad (después de ésta haber sido verificada), es decir los 24 bits de dato.

Como se tienen los 24 bits sin estar arreglados, hay que darle formato para que tome la representación de la tarjeta, esto es FacilityCode (8bits) – Identificación (16bits).

Luego de estar acondicionada, la data es enviada por vía serial al servidor web, byte a byte, de la siguiente manera:

- Primer byte 0x3C (<): indica el inicio de la trama.
- Segundo byte (FacilityCode): el byte del facility code.
- Tercer byte 0x0D (-): es el separador de los datos.
- Cuarto y quinto byte (Identificación): los dos bytes de la identificación.
- Sexto byte 0x3A (:): el separador de la data con la suma de verificación
- Séptimo byte (CheckSum): suma de verificación de la data
- Octavo byte 0x3E (>): indica el final de la trama.

Por lo que una cadena de caracteres se vería de la siguiente manera:

<( FacilityCode)-( Identificación):(CheckSum)>

Un ejemplo de la representación de una tarjeta transmitida desde la controladora, sería como se muestra a continuación:

<165-53860:207>

**Facility Code:** 165

**Identificación:** 53860

**CheckSum:** 207

#### **4.2.1.5 Subrutina Abrir Puerta**

Al recibir respuesta del servidor, se activa la interrupción serial del microcontrolador y se lee y guarda el byte recibido, si el mismo es igual al byte autorizado en el código, se realiza el llamado a esta subrutina.

Es una subrutina que coloca en estado lógico “1” un pin del microcontrolador, que se encuentra conectado al transistor que activa el relé y por tanto, habilita la hembrilla para realizar el ingreso a la instalación por unos pocos segundos (aproximadamente 2 segundos).

Otra de las maneras de activar esta subrutina es a través de la activación del botón para salir.

#### **4.2.2 Programa del pcDuino**

Al encender el pcDuino, se ejecutan los programas que permiten realizar la comunicación bidireccional con la unidad controladora. En estos programas se realiza la apertura de un puerto serial para poder escuchar por el toda la data entrante desde la controladora, de igual manera que en la controladora, éste puerto está configurado a **9600** baudios por segundo, **8** bits de datos, **sin paridad** y **1** bit de parada.

Las rutinas para la realización de esta tarea fueron desarrolladas en el programa Microsoft® Visual C# Express 2010, que cuenta con el licenciamiento gratuito de la misma plataforma. Así como el código de la página web fue diseñado con Visual Web Developer Express, igualmente de la casa Microsoft® y con mismo formato de licencia.

A nivel de servidor web, este se encuentra realizado con el paquete de programas LAMP, cuyas siglas significan Linux, Apache, MySQL y PHP. Son software abierto que cuentan con la licencia GNU GPL. Al estar desarrollando sobre el pcDuino, que cuenta con el kernel de Linux y como sistema operativo Lubuntu, el software se rige bajo esta licencia.

Explicando con un poco más de detalle, Apache es el programa que permite abrir las conexiones del servidor hacia las demás personas, se encarga de administrar las conexiones entrantes y salientes así como verificar la permisología que tiene cada usuario que intente realizar una conexión. MySQL es el gestor de la base de datos, en él se encuentra guardada toda la información del servidor web, tales como usuarios, administradores, ingresos realizados, personas autorizadas, solicitudes de contacto, entre otros. Por último, PHP, es un lenguaje especialmente adecuado para el desarrollo web y que puede ser incrustado en el código HTML. Se ejecuta a nivel del servidor y el cliente obtiene una respuesta en HTML. (Manual de PHP, 2014).

A continuación se explicarán con más detalles cada una de las partes que conforman el servidor web, desde la recepción de la data hasta la muestra de la misma en una página web para el cliente.

#### **4.2.2.1 Recepción de la data de la controladora**

Para poder ejecutar esta rutina, se tuvo que instalar en el pcDuino el compilador Mono, el cual es una implementación de código abierto para el Framework .NET de Microsoft®, basado en los estándares para la compilación con C#.

La rutina de recepción de la data de la controladora siempre se encuentra en la espera de un carácter de inicio por el puerto serial. El cual es un byte (0x3C en hexadecimal, “<” en ASCII), para indicarle que se comenzó la transmisión de los datos desde la controladora. Esta continúa recibiendo datos hasta que se llega al final de la cadena (0x3E en hexadecimal, “>” en ASCII). Al finalizar la recepción de la data, se realiza una separación de la cadena recibida en la ubicación del carácter (“:”), y se realiza el cálculo de la suma de verificación.

El resultado de esta suma se compara con la proveniente de la controladora, si estos datos coinciden quiere decir que hay una integridad en los datos recibidos y se realiza una consulta en la base de datos, si el código de la tarjeta es encontrado y se encuentra habilitado, se le envía a la unidad controlador un byte para indicarle que el usuario se encuentra habilitado y a la misma vez, se realiza un almacenamiento del nombre y apellido, fecha, hora y número de la puerta por la cual ingresó.

En caso contrario, que la suma de verificación no coincida, se dispara una alarma que indica que hubo un error en la recepción de los datos. Esta es almacenada en la base de datos para luego ser mostrada en la página web.

En esta rutina también se realiza la activación de unos temporizadores que se encargan de almacenar alarmas si existen problemas en la ejecución de los programas. Si se recibe un carácter de inicio (<), se inicia un temporizador de 2 segundos en el cual, si no se recibe data o un carácter de finalización (>), se agota el tiempo y se dispara el evento almacenando así la alarma en la base de datos.

De igual manera existe un temporizador que verifica que la conexión controladora – servidor se encuentre activa, la controladora se encarga de enviar cada 20 segundos, aproximadamente, un código único que la identifica, si el servidor recibe este código reinicia el temporizador, evitando así que se agote el tiempo, en caso contrario, si el tiempo se agota, se dispara un evento que almacena una alarma en la base de datos indicando que hay que revisar la conexión con la controladora ya que existe un problema.

En la siguiente figura se puede observar un diagrama de flujos del comportamiento de esta rutina.

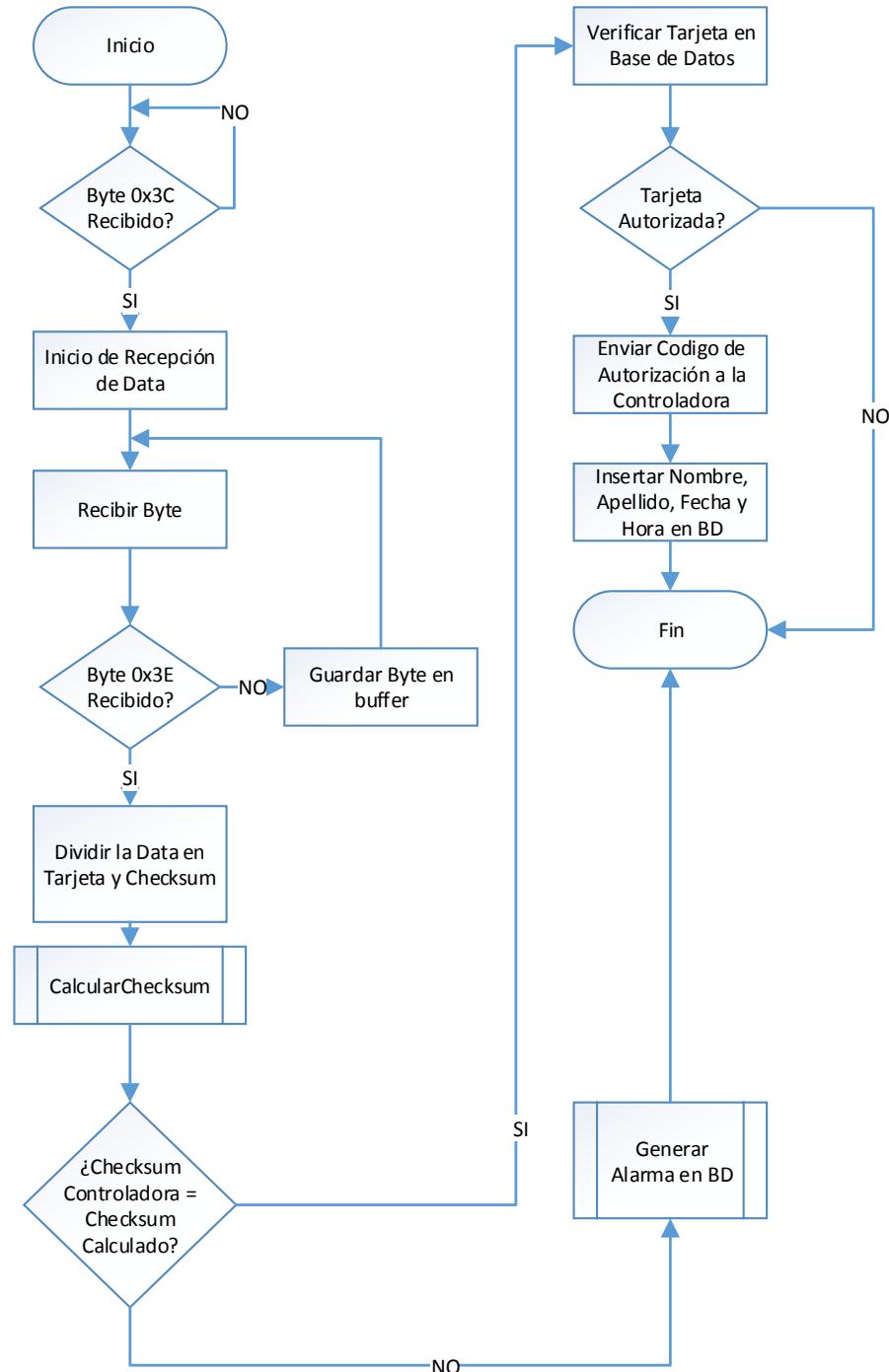


Figura 24- Diagrama de flujo firmware de recepción de datos en el servidor

#### 4.2.2.2 Programa del Servidor Web

En esta sección se explicarán las conexiones Cliente – Servidor, la realización de la GUI y la base de datos.

- **Conexión Cliente – Servidor**

Esta conexión se refiere al proceso de interacción entre la computadora o equipo local (cliente) y el remoto (servidor). El cliente realiza solicitudes y el servidor se encarga de responder a las mismas. Este modelo por lo general se basa en comunicaciones en una misma red.

En el caso de este proyecto, el modelo cliente – servidor esta realizado sobre un protocolo TCP/IP de la capa de conexión del modelo OSI. El software encargado de atender a estas solicitudes es Apache ejecutando un servicio que gestiona todas las solicitudes y realizar respuestas según sea el caso. En la siguiente figura se puede observar como el usuario puede interactuar con el servidor web.

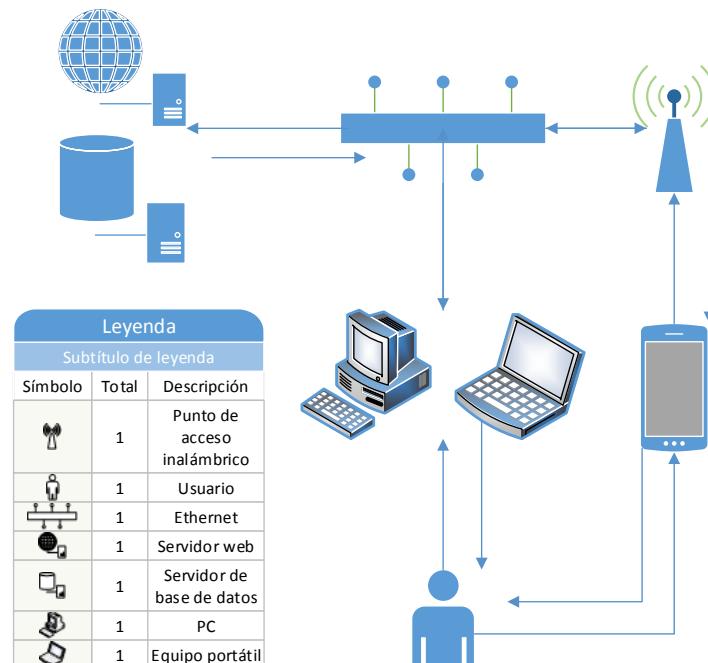


Figura 25 - Diagrama de Interacción con el servidor web

Para la interacción con estas solicitudes, se realizaron funciones en código PHP, las cuales son ejecutadas en el servidor y al cliente se le envía el resultado en forma de una respuesta, para este caso, en forma de página web. Muchas de estas funciones se encargan de mostrar datos, así como otras que se encargan de ingresar datos a la base de datos.

Para acceder a muchas de las funciones de este servidor web, el usuario deberá estar registrado como administrador en la base de datos ya que, como se está hablando de un sistema de control de acceso, la seguridad es fundamental en ello.

Las funciones que se realizan en el servidor y que requieren autenticación de administrador, son las consultas de ingresos realizados al laboratorio, verificar que personas están autorizadas, insertar nuevas tarjetas y usuarios, consultas de solicitudes para ingreso al laboratorio, entre otras.

Sin embargo hay otras funciones que no requieren un nivel de autenticación, como lo son la información de contacto, información de usuarios del laboratorio, entre otras.

En la figura 26, se puede observar un diagrama de como son los niveles de seguridad en el servidor web. Los usuarios administradores deben entrar a la función para iniciar sesión para poder acceder a las diferentes funciones que así lo soliciten. Mientras que un usuario anónimo solo puede observar e interactuar con un grupo pequeño de las mismas.

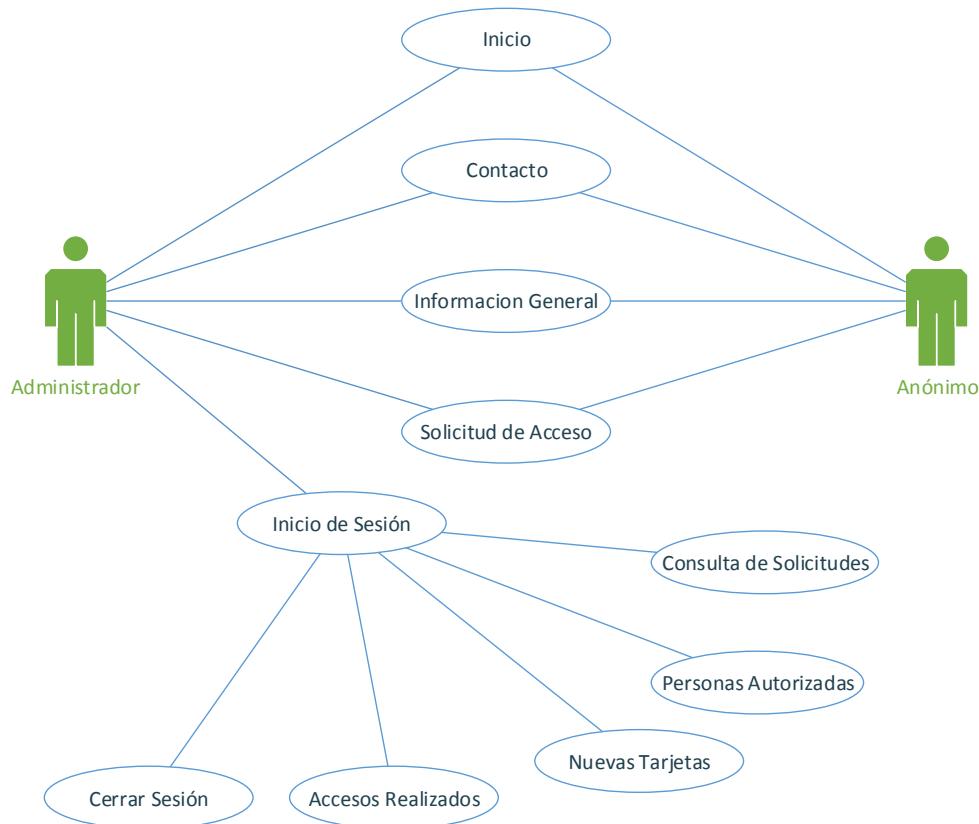


Figura 26 - Diagrama de casos de niveles de accesos

- **Base de datos**

La base de datos se encuentra realizada en MySQL, en ella se aloja toda la información pertinente al servidor. Usuarios, accesos realizados, tarjetas autorizadas, entre otros. Este entorno posee una interfaz gráfica para facilitar el manejo de la misma, phpMyAdmin, es una interfaz realizada bajo PHP que permite administrar la base de datos. Debe ser manejada por un administrador de base de datos, que puede ser un usuario administrador encargado de esta tarea u otro personal calificado que se denomina DBA (administrador de base de datos). El DBA puede activar o desactivar usuarios directamente, crear nuevos usuarios administradores, así como realizar todas las funciones importantes a nivel de almacenamiento.

Las funciones que se comunican con la base de datos desde el cliente lo hacen a través la ejecución de código PHP en el servidor y devuelven una respuesta en forma de página web con la información solicitada.

Solo se pueden registrar usuarios administradores desde el entorno de la base de datos, a nivel de cliente no existe ninguna función que lo realice.

Es importante que se realice un buen desarrollo de base de datos ya que de ella depende el correcto funcionamiento del equipo por completo.

Para realizar una consulta es importante conocer la “cadena de conexión”, que no es más que una combinación de nombre de usuario, contraseña, base de datos y servidor a conectarse (todos estos datos son propios de la base de datos y diferentes a los que un usuario administrador pueda tener, para garantizar la seguridad e integridad de la aplicación). Una vez realizada esta autenticación, se pueden realizar consultas de diferente índole, en nuestro caso específico, desde el firmware que recibe los datos de la unidad controladora y las consultas desde la página web del lado de los clientes.

- **Interfaz gráfica de usuario (GUI)**

Es la manera en la que el (o los) cliente(s), se pueden comunicar con el servidor. Son un conjunto de páginas web realizadas bajo código HTML5 con extensiones de JavaScript para diversas funciones y CSS3 para dar un estilo visual, de esta manera se conforman el entorno visual del usuario.

Fue diseñada en Visual Web Developer Express para realizar la disposición de los elementos y aplicarle estilos y funcionalidades (HTML5, JavaScript y CSS3) y las herramientas Geany (versión 1.24) y NetBeans (versión 8.0) para la realización del código PHP para las páginas dinámicas.

Al ingresar al servidor, vía cualquier programa para navegación web, se muestra de manera automática la página de inicio (index.php). Desde ella el usuario puede navegar a cualquiera de las otras páginas o funciones, siempre y cuando esté autorizado

(figura 26, diagrama de casos de niveles de acceso). A continuación se puede observar la página de inicio.

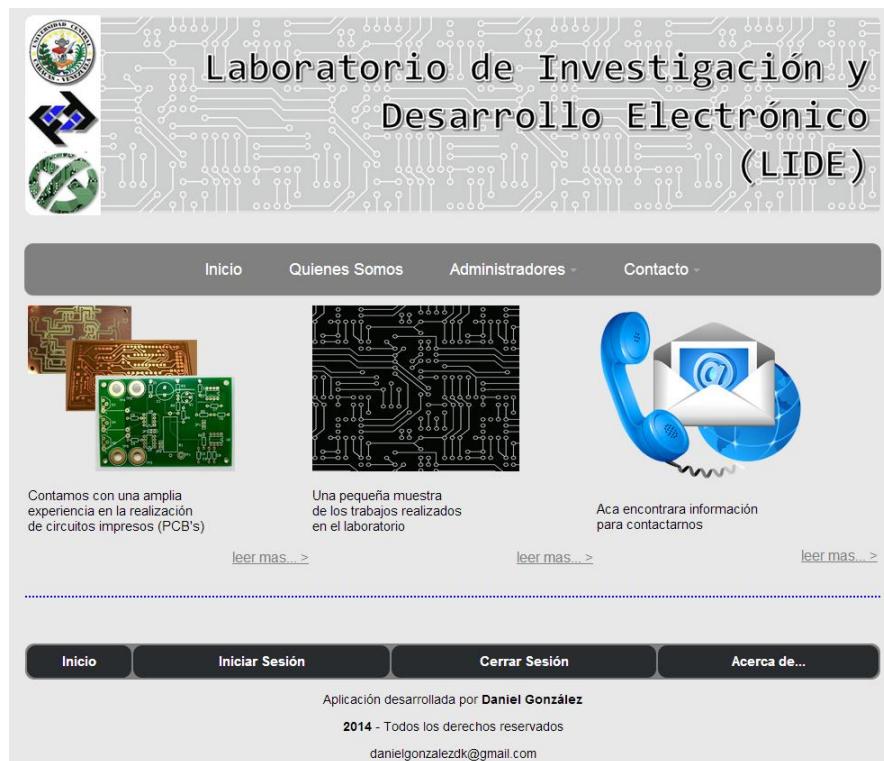


Figura 27- Página de Inicio del Servidor Web

Desde esta página el usuario puede iniciar sesión, verificar los ingresos realizados al laboratorio (si es usuario administrador) o solicitar el acceso al laboratorio o ver las maneras de contactar al mismo (en caso de ser usuario anónimo).

## CAPITULO V. PRUEBAS Y RESULTADOS

### 5.1 Pruebas Funcionales

Durante el proceso de diseño y construcción del prototipo funcional del servidor y la unidad controladora, así como de todos los módulos adicionales, se realizaron diversas pruebas para verificar el correcto funcionamiento de cada una de ellas y del conjunto.

#### 5.1.1 Pruebas Iniciales en la Controladora

Al realizar la construcción de la tarjeta de la unidad controladora, se procedió a verificar que estuviesen bien delimitadas sus pistas y no tuvieran cortocircuitos en ninguna de ellas antes de proceder a ensamblar y soldar cada uno de los componentes. Una vez soldados y antes de alimentar, se procedió nuevamente a verificar que no hubiesen cortocircuitos entre pistas.

Ya realizadas estas tareas, se alimentó la tarjeta con 9VDC y verificó que a la salida del regulador se tuviesen los 5VDC necesarios para la electrónica de la misma. Cabe destacar que este paso fue realizado sin tener nada conectado en los terminales externos ni haber colocado el microcontrolador.

Luego se procedió a colocar el microcontrolador, la tarjeta de leds indicadores y encender la tarjeta. Al cabo de unos segundos, esperar que inicialice el microcontrolador, se verificó que los leds estuviesen encendidos (led rojo de encendido y led amarillo de Tx, el led amarillo de Rx no se enciende debido a que no hay un elemento externo conectado todavía).

Se realizaron diversas pruebas en el momento para verificar el correcto funcionamiento de la tarjeta, como lo son activar el relé directamente y a través del botón RTE. Luego de esto, se conectaron los elementos externos que se necesitan en la controladora, el elemento de identificación (lector de radiofrecuencia) y el elemento para la comunicación vía serial. Una vez realizadas estas conexiones se realizaron

pruebas leyendo los datos de diversas tarjetas y enviándolos a una PC a través de un programa para la captura de los datos seriales (Terminal de Puerto Serial, diseñado en Visual C# Express 2010 ®), con el fin de leer y verificar que el firmware está realizando de manera correcta la tarea de traducción y envío de los datos así como el cálculo de la suma de verificación. De igual manera, desde este mismo terminal se realizó el envío de bytes para examinar el comportamiento de la controladora. Entre estos bytes se encontraba el que autoriza la activación del relé y por tanto, la apertura de la puerta. Mientras que con bytes diferentes, la controladora no realizaba acción alguna.

Se verificó que se estaba ejecutando la interrupción del temporizador, el cual es importante para la alarma de comunicación que se encuentra en el servidor. Cada 20 segundos, aproximadamente, se encarga de enviar un código único hacia el servidor.

Después de verificados todos los aspectos importantes en la controladora, se procedió a hacer lo propio en el servidor (pcDuino).

### **5.1.2 Pruebas iniciales en el servidor web**

Principalmente se verificó que el pcDuino iniciara de manera normal y corriera el sistema operativo, una vez realizado esto, se procedió a la configuración de los puertos y programas necesarios para el funcionamiento del servidor.

Se realizó la instalación del módulo RTC y se ejecutó el firmware diseñado para este, configurando fecha y hora. Se verificó que una vez realizado esto, el pcDuino tomara la fecha y hora correspondiente al mismo. Esto se realizó varias veces reiniciando el pcDuino, dejándolo apagado por unos minutos y volviéndolo a encender para constatar que si estaba tomando la hora del módulo.

Se configuró el puerto serial del servidor para indicarle que por el estaría recibiendo datos. Luego de configurado, se realizaron diversas tareas desde una PC

enviando datos y verificando que los mismos eran recibidos de manera correcta. Esto se realizó varias veces con el fin de detectar algún problema o error en la transmisión.

Luego se instaló el paquete LAMP, y verificó que el servidor estuviese corriendo ya que de manera automática este se activa. En un navegador web colocando en la barra de direcciones “localhost”, este mostró una respuesta de que funcionaba. Se comprobó la conexión a la base de datos ingresando a ella y realizando diversas consultas para confirmar su funcionamiento.

Se copió la página web al directorio destinado para ello y se ingresó a ella como “localhost” desde el pcDuino y a través de otras computadoras en la misma red utilizando la dirección IP, mostrando así como respuesta la página de inicio, lo que quería decir que estaba corriendo de manera correcta. Se navegó a través de ella para verificar que todos los elementos que la conforman se estuviesen ejecutando de manera correcta y, de existir algún error, repararlo.

Una vez tenido el pcDuino con todas las herramientas necesarias, se realizó la corrida del firmware para la adquisición de la data serial, indicándole puerto serial por el cual estaría escuchando y se emuló desde la PC como si una persona estuviese intentando acceder al laboratorio y las demás condiciones que podían activar las alarmas del sistema. Se observaron las respuestas obtenidas, lo cual coincidía con lo esperado si el usuario estaba o no autorizado y dependiendo de la simulación de las fallas y generación de alarmas.

Se realizaron pruebas para verificar que el cálculo de la suma de verificación de los datos era correcta y enviando datos erróneos (valor de tarjeta con suma de verificación incorrecto) para verificar que no realizara ninguna acción al respecto e insertara la alarma en la base de datos.

### **5.1.3 Pruebas de navegación en el servidor web**

En estas pruebas se realizaron diversas consultas a la página tanto como un usuario anónimo como un administrador.

El usuario anónimo solo pudo realizar navegación sencilla en la página. Visitar el inicio, la información de contacto e información de los trabajos que se realizan en el laboratorio, así como también pudo realizar solicitudes para acceder a las instalaciones.

Al momento de requerir otras páginas, al usuario se le solicita que ingrese nombre de usuario y contraseña para proseguir. Como no se encuentra registrado, el sistema no lo deja acceder.

Todos los usuarios son automáticamente dirigidos al inicio cuando se conectan al servidor; ahora al navegar con un usuario anónimo las respuestas del servidor se muestran en las siguientes figuras.

**--Profesores y estudiantes que hacen vida en el lide--**

**Rafel Rivero**  
Ingeniero Electricista (UCV)  
Mención Electrónica, Computación y Control  
Profesor de la Unidad Docente Electrónica  
Tesis: "[Tesis de Rivero](#)"

**Servando E. Álvarez C.**  
Ingeniero Electricista (UCV)  
Mención Electrónica, Computación y Control  
Profesor de la Unidad Docente Electrónica  
Tesis de Postgrado: "[Estación sismológica de bajo costo basado en acelerómetros de tecnología MEMS](#)"

**Daniel A. González M.**  
Estudiante de Ingeniería Eléctrica  
Mención Electrónica, Computación y Control  
Tesis de Pregrado  
Tesis: "[Diseño e Implementación de un Servidor Web Embebido en un Microcontrolador como Sistema de Control de Acceso al LIDE](#)"

*Figura 28 – Vista de la página "Quienes Somos"*

En esta página el usuario se pudo visualizar quienes son los profesores, tesis y pasantes que se encuentran trabajando en el laboratorio.

El siguiente elemento que se pudo observar con este usuario es el de “*Contacto*”, en este se encuentran todas las maneras de contactar al laboratorio.

Entre ellas están un formulario de solicitud en línea, un formulario en formato PDF para ser descargado y la información del laboratorio, como ubicación y teléfonos de contacto.

Figura 29 – Formulario de Solicitud en línea

Figura 30 – Página de Contacto

Adicional a estas páginas, el usuario también pudo visualizar la información personal del creador de la página web.

## Daniel Andrés González Mariño

Estudiante de Ingeniería Eléctrica  
Mención Electrónica, Computación y Control

Actualmente Tesista en el Laboratorio de Investigación y  
Desarrollo Electrónico  
--LIDE--

Tesis: "Diseño e Implementación de un Servidor Web Embebido en un Microcontrolador como Sistema de Control de Acceso al LIDE"

Universidad Central de Venezuela

Facultad de Ingeniería

Escuela de Ingeniería Eléctrica

Inicio

Aplicación desarrollada por Daniel González

2014 - Todos los derechos reservados

danielgonzalezdk@gmail.com

Figura 31 - Página “Acerca de...” (Información personal)

Sistema de Control de Acceso del Laboratorio de Investigación y Desarrollo Electrónico

**LIDE**

Facultad de Ingeniería - Escuela de Ingeniería Eléctrica

Universidad Central de Venezuela

Planilla para Solicituds del LIDE

Nombre y Apellido:

Cedula de Identidad:

e-mail:   Otro:

Mención a la que Pertecece

Acciones a Desarrollar en el LIDE:

Uso del Taladro  Soldadura  Solicitud de Material  Consulta

Fecha y Hora para la Solicitud:

Día  Mes  Año  Hora   am  pm

Imprimir Solicitud  Enviar Solicitud

Figura 32 – Formulario en formato PDF

Una vez visualizadas todas las páginas a las que se tiene acceso, se intentó ingresar a las de la sección “*Administradores*”, teniendo como respuesta en todos los caso un bloqueo y solicitando al usuario que inicie sesión, esta página se puede observar a continuación.



Figura 33 - Página de inicio de sesión

Una vez que el usuario ingresa sus datos y presiona aceptar, existen tres respuestas de la página, la primera es que el usuario no existe, la segunda es que el usuario existe pero la contraseña es incorrecta y la última es que el usuario es autenticado y nos redirige al inicio. Estas tres respuestas se pueden observar a continuación.

**El Usuario, "usuario" NO existe, por favor intente nuevamente**

Figura 35 - Respuesta de usuario no existe

**Combinación de usuario y contraseña incorrectos**

Figura 34 - Combinación de usuario y contraseñas incorrectos

Por último, si esta autenticado, nos redirige al inicio y para saber que el inicio de sesión fue satisfactorio, se observa en la parte superior izquierda, sobre la barra de navegación (así como también en todas las páginas que un usuario administrador navegue) el nombre de la persona que está ingresando.

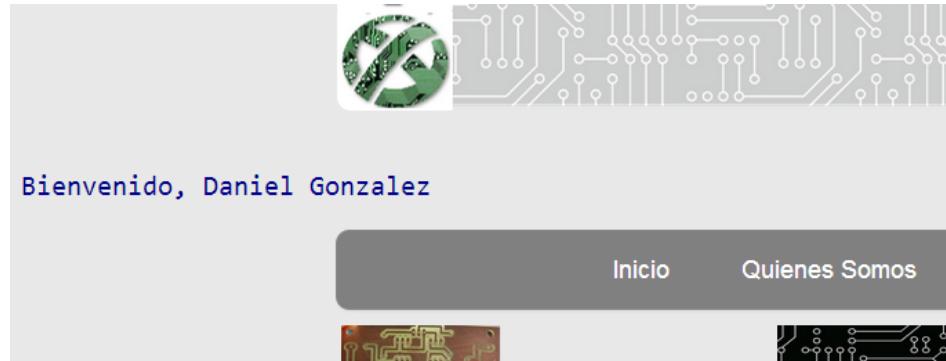


Figura 36 - Respuesta de inicio de sesión satisfactorio

Cabe destacar que todos los registros de inicio de sesión, bien sean satisfactorios o erróneos, quedan registrados en la base de datos, con el fin de más adelante, en caso de ser necesario, verificar si existe alguna persona intentando acceder al sistema que no esté autorizado, es decir auditar los datos.

Nickname	Nickname del usuario de inicio de sesión	Nombre	Nombre de la persona asociado al nickname	Apellido	Apellido de la persona asociado al Nickname	FechaHoraInicioSesion	Operacion realizada para el inicio de sesión	Operacion	Observacion
admin		Daniel		Gonzalez		2014-08-20 18:35:43	Realizar consultas varias de Administrador		Consulta realizada con éxito
admin		Daniel		Gonzalez		2014-08-21 07:37:55	Realizar consultas varias de Administrador		Consulta realizada con éxito
admin						2014-08-21 08:06:03	Realizar consultas varias de Administrador		Consulta realizada sin éxito, usuario sin privileg...
"usuario"						2014-08-21 08:06:23	Realizar consultas varias de Administrador		Consulta realizada sin éxito, usuario sin privileg...
admin		Daniel		Gonzalez		2014-08-21 08:12:26	Realizar consultas varias de Administrador		Consulta realizada con éxito

Figura 37 - Registros en la base de datos de inicio de sesión

Las páginas de la sección “Administradores” ahora se encuentran disponibles, por lo que se pueden visualizar. Entre ellas se tienen las personas autorizadas, consulta de solicitudes, verificación de accesos al laboratorio, ingresar nuevas tarjetas al sistema y verificar las alarmas generadas.

Al seleccionar el link de “Personas Autorizadas”, se muestra una ventana en la cual se indica nombre, apellido, descripción y estatus de la persona; con el estatus se quiere decir si está habilitado (1) o no lo está (0). En la siguiente figura se puede observar esta página.

Nombre	Apellido	Descripcion	Estatus
Daniel	Gonzalez	Tesista	1
Servando	Alvarez	Egresado	1
Rafael	Rivero	Egresado	1
Cesar	Gutierrez	Tesista	1
Juan	Aponte	Tesista	1
Daniel	Lourtec	Tesista	0

Figura 38 - Visualización de personas autorizadas y habilitadas a acceder al laboratorio

Luego se visitó la página de “Consulta de Solicitudes” en la que se pueden visualizar las solicitudes hechas por personas ajenas al laboratorio que necesiten acceder al mismo para realizar cualquier tarea, bien sea uso de las herramientas, realizar consultas a los profesores y/o tesistas que se encuentren en él, solicitar material de apoyo, entre otros.

Esta página se puede visualizar a continuación, con algunas solicitudes realizadas para probar su funcionamiento.

Nombre	Apellido	Cedula	Email	Mencion	Taladro	Soldadura	Material de Apoyo	Consulta	Dia Asistir	Observacion	Fecha de Solicitud
Pepe	Perez	4742163	pepeperez@pp.com	Industrial		X			2014-09-24		2014-08-21 14:31:30
Jorge	Diaz	12345678	jorlup@gmail.com	Electronica			X	X	2014-08-25	Otra prueba de solicitud de acceso	2014-08-21 14:30:06
Daniel	Gonzalez	18211493	danielgonzalezdk@gmail.com	Electronica	X	X	X	X	2014-08-29	Probar el sistema de control de acceso del lide	2014-08-21 14:28:50

Figura 39 - Visualización de solicitudes realizadas por personas externas al laboratorio

Luego de verificada esta página de solicitudes, se procedió a verificar los accesos realizados al laboratorio. Por defecto, solo se muestran los últimos 30 accesos realizados, pero esto es configurable. El hecho de que sea de esta manera, es para no sobrecargar la carga de la página en el navegador. A continuación se muestra el resultado obtenido de la visualización.

Fecha y Hora	Nombre	Apellido	Puerta
2014-08-21 13:57:59	Daniel	Gonzalez	1
2014-08-21 13:57:58	Daniel	Gonzalez	1
2014-08-21 13:57:44	Daniel	Gonzalez	1
2014-08-21 13:57:17	Daniel	Gonzalez	1
2014-08-21 13:57:13	Servando	Alvarez	1
2014-08-21 13:57:09	Juan	Aponte	1
2014-08-21 13:57:03	Rafael	Rivero	1
2014-08-21 13:56:59	Cesar	Gutierrez	1
2014-08-21 13:56:51	Daniel	Gonzalez	1
2014-08-20 22:03:22	Juan	Aponte	1

Figura 40 - Visualización de los últimos 30 ingresos realizados al laboratorio

Después de verificada la navegación en la página anterior, se procedió a realizar el ingreso de una nueva tarjeta al sistema, a través de la página “Nueva Tarjeta”, en ella se solicita nombre, apellido, si es tesista, egresado o pasante y el número de la tarjeta a ingresar. Una vez introducidos los datos, se procedió a presionar el botón de aceptar, y si los datos fueron ingresados correctamente, se nos indicó mediante un aviso debajo del formulario.



Este formulario se titula "Ingresar los datos de la nueva autorización". Contiene los siguientes campos:

- Nombre: Prueba
- Apellido: Ingreso
- Condición: Tesista
- Número de tarjeta: 123-45678

Al final del formulario se encuentran los botones "Aceptar" y "Limpiar".

Figura 41 - Ingreso de nueva autorización

Y la respuesta del sistema si el ingreso de los datos fue satisfactorio.



Este formulario muestra la respuesta del sistema:

Tesista

tarjeta #

Aceptar Limpiar

Última tarjeta insertada correctamente:  
Prueba Ingreso 123-45678

Figura 42 - Nueva tarjeta ingresada satisfactoriamente

Por último, se navegó hasta la sección de “Alarmas” para visualizar que se mostraran adecuadamente. La respuesta de esta página mostró una tabla con todas las alarmas que se pueden generar en el sistema. Estas se explicarán en la siguiente sección.

Alarmas Generadas en el Sistema		
IdAlarma	Descripcion	Fecha y Hora
21	Error - Intento de Ingreso no autorizado	2014-08-20 21:28:40
20	Error - Problema con la comunicacion con la Controladora	2014-08-20 14:47:58
19	Error - Intento de Ingreso no autorizado	2014-08-20 14:47:29
18	Error - Intento de Ingreso no autorizado	2014-08-20 14:47:25
17	Error - Problema con la comunicacion con la Controladora	2014-08-20 14:45:27
16	Error - Problema con la comunicacion con la Controladora	2014-08-20 14:44:57
15	Error - Problema en la recepcion de datos	2014-08-20 14:44:22
14	Error - Problema en la recepcion de datos	2014-08-20 14:44:22
13	Error - Problema en la recepcion de datos	2014-08-20 14:44:22
12	Error - Problema con la comunicacion con la Controladora	2014-08-20 12:55:02
11	Error - Problema con la comunicacion con la Controladora	2014-08-20 12:54:32
10	Error - Problema en la recepcion de datos	2014-08-20 12:54:02

Figura 43 - Visualización de las alarmas generadas en el sistema

Al finalizar la navegación en la página por la sección de administradores, se procedió a verificar que el cierre de sesión estuviese correcto y no dejara al usuario administrador presente. Cabe destacar que al cabo de unos minutos en los que un usuario administrador tenga la sesión iniciada pero se encuentre inactivo, la sesión se cerrará automáticamente con lo cual se mantiene la seguridad del sistema.

#### 5.1.4 Pruebas de las Alarmas del sistema

El sistema está en la capacidad de generar tres tipos de alarmas los cuales son:

- **Error - Problema con la comunicación con la Controladora**

Se trata básicamente de que el servidor no está recibiendo información de la controladora en un tiempo prudencial o detectó que no está presente. Para realizar estas pruebas, se realizó la corrida del firmware en el pcDuino y se esperó el tiempo de agotamiento para generar la alarma, el cual es cada 30s.

Al cabo de este tiempo, si no se ha recibido datos de ningún estilo, el servidor genera la alarma y la almacena en la base de datos para que después sea visualizada en la página web. Estas pruebas se realizaron primero emulando la conexión con una PC y enviándole señales al servidor para indicarle que hay una conexión activa y luego esperando varios minutos sin enviar señales de ningún tipo para verificar que se generara la alarma.

Luego se realizó la conexión de la controladora y se verificó que no se generara la alarma siempre y cuando estuviese la conexión activa o se estuviese enviando datos de la lectura de una tarjeta de identificación.

- **Error - Problema en la recepción de datos**

Esta alarma se genera cada vez que se ve interrumpida la transmisión de los datos, los mismos no llegan completos o la suma de verificación calculada no coincide con la suma de verificación proveniente de la controladora. Ya se sabe que los datos son de la forma < “*datos a enviar:checksum*” >, con un carácter de inicio (<), uno de finalización (>) y el separador (:). Si por alguna razón se comienza la transmisión de datos pero al cabo de unos pocos segundos esta no ha finalizado (>), se activa la alarma de que hubo un problema en la transmisión y hay que revisar el equipo.

Para realizar estas pruebas se emuló desde una PC el envío de datos hacia el servidor pero sin colocarle el carácter de finalización es decir <111-12341, y al cabo

de unos segundos, se verificó que la alarma se haya generado y almacenado en la base de datos; esto se hizo varias veces para verificar su correcto funcionamiento. Para realizar esta prueba con la controladora es muy complicado, ya que la velocidad de transmisión es muy elevada y no se puede interrumpir la misma.

Para verificar que la rutina de la suma de verificación igualmente estuviese funcionando correctamente, se emuló desde la computadora el envío de la data serial pero con un valor erróneo, por ejemplo <111-12341:187> sería el valor correcto a enviar y el firmware debería realizar el cálculo y permitir el ingreso si está autorizado; pero si el valor enviado fuese <111-12341:185>, el servidor calcularía como valor de verificación “187” al compararlo con “185” sabrá que hubo un problema en los datos y se procede a disparar la alarma.

- **Error - Intento de Ingreso no autorizado**

Si se da el caso de que una persona que no esté autorizada intenta ingresar al sistema, esta alarma se activa y se almacena en la base de datos. Esto es con el fin de que al momento de auditar los datos se pueda conocer que existen personas tratando de ingresar a la instalación y tomar medidas al respecto. Para la realización de estas pruebas se tomaron algunas de las tarjetas y se deshabilitaron en la base de datos. Teniendo así personas no autorizadas y generando la alarma.

### **5.1.5 Pruebas completas del sistema**

Para finalizar las pruebas, se realizó la conexión del sistema general (controladora, indicadores, cerradura eléctrica, botón de salida, transmisión serial, servidor, reloj del servidor) y se procedió a dejarlo encendido varios días seguidos, realizando pruebas ocasionales y verificando que se activaba la cerradura en caso de estar autorizado, verificando la generación de alarmas si se cortaba la transmisión controladora/servidor o el usuario no se encontraba habilitado, el correcto funcionamiento de la página web y el almacenamiento de la información en la base de datos.

Para realizar estas pruebas completas se armó una maqueta que se muestra en las siguientes figuras, consta de la puerta, la cerradura eléctrica, el pestillo, las conexiones a la controladora y al pcDuino, así como el resto de las conexiones pertinentes para el correcto funcionamiento.

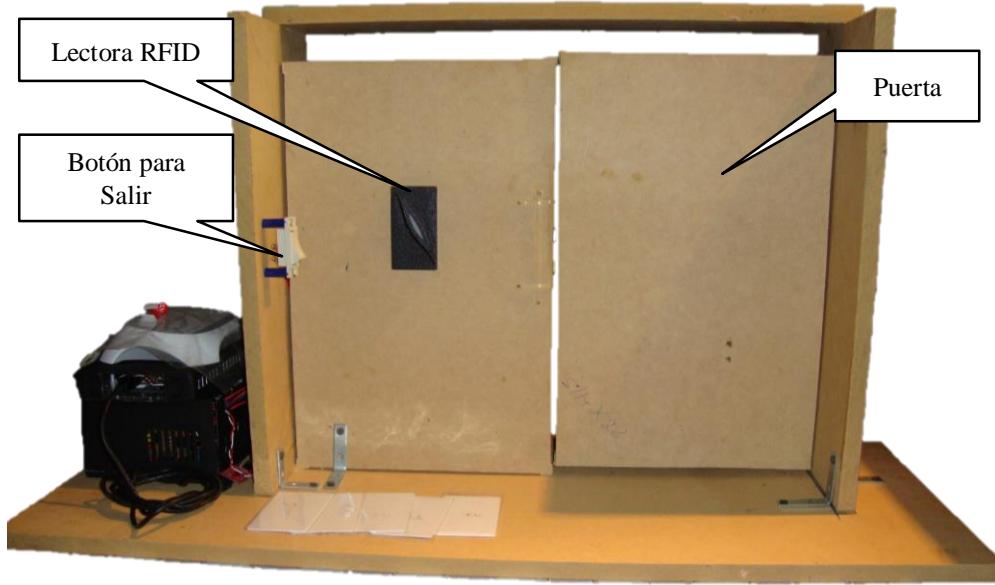


Figura 44 - Vista frontal de la maqueta

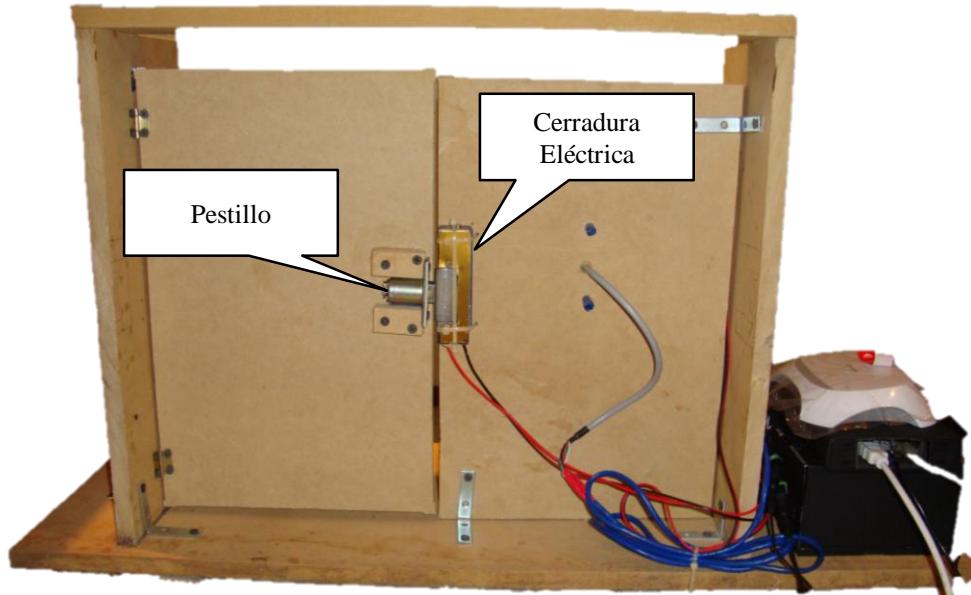


Figura 45 - Vista posterior de la maqueta

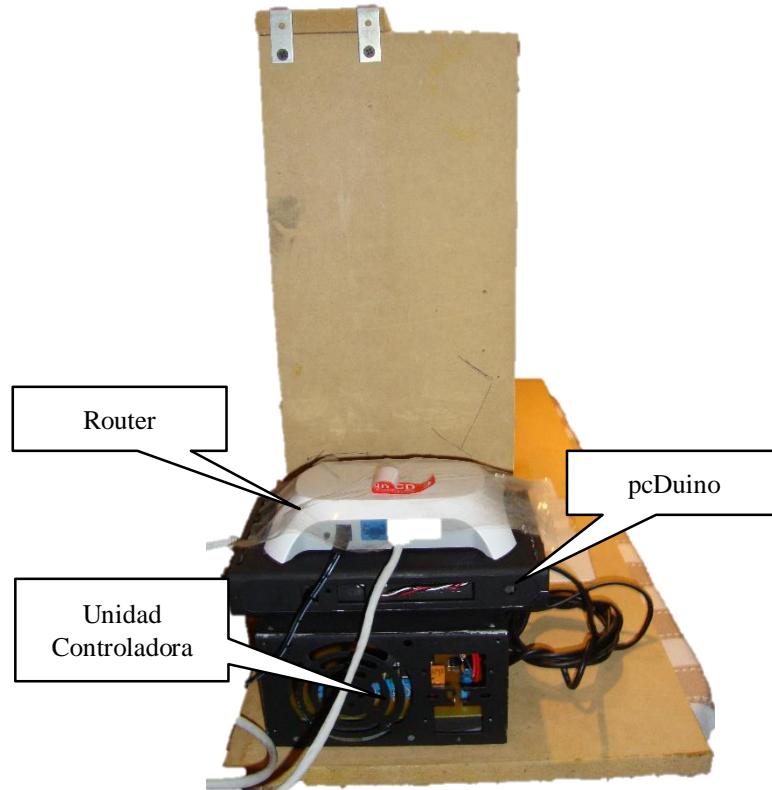


Figura 46 - Vista lateral de la maqueta

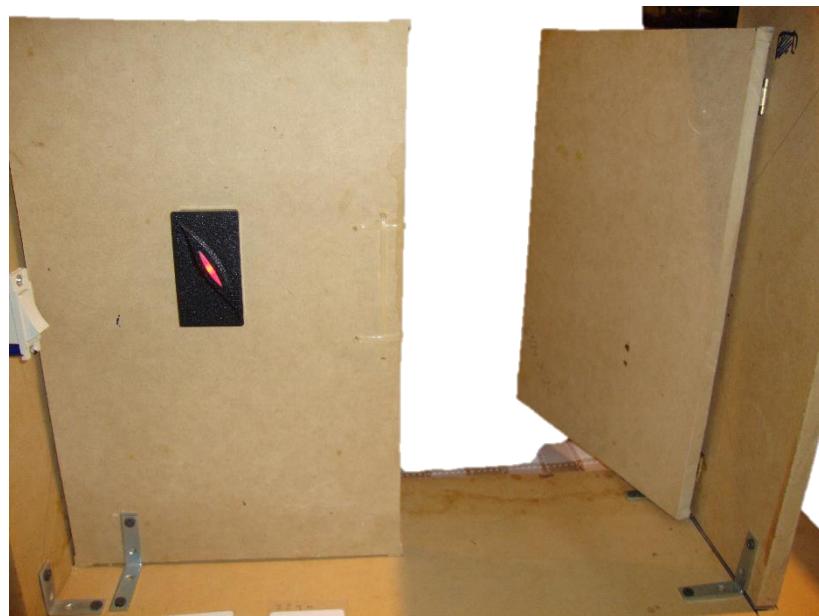


Figura 47 - Maqueta con la puerta abierta, indicando que fue autorizado el acceso

## 5.2 Especificaciones del equipo

### 5.2.1 Especificaciones de la Unidad Controladora

A continuación se muestra una tabla de las especificaciones de la unidad controladora.

*Tabla 7 - Especificaciones de la unidad controladora*

<b>Unidad Controladora - Especificaciones</b>		
	<b>Valor</b>	<b>Observación</b>
<b>Tensión de alimentación</b>	7.5 – 9VDC	
<b>Corriente mínima</b>	250 mA	
<b>Elementos de identificación que se pueden conectar</b>	2	Se deben comunicar en formato Wiegand-26
<b>CPU</b>	PIC18F4550	
<b>Conexiones Externas</b>	5	UART, I2C, USB, Botón de salida, Relé.
<b>UART</b>	9600 bps, sin paridad, 8 bits de datos y 1 bit de parada. (Rx, Tx, GND)	Para elementos que trabajen a 5VDC
<b>I2C</b>	SDA, SCL, GND	Para elementos que trabajen a 5VDC
<b>USB</b>	D+, D-, GND	Para elementos que trabajen a 5VDC
<b>Botón de Salida</b>	1	Conectar un botón entre estos terminales
<b>Relé KS2E-M-DC5</b>	<b>Bobina interna:</b>	
	5VDC, 0.20W	
	<b>Contactos:</b>	
	2A @ 30VDC 1A @ 125VAC	Potencia máxima que se puede disipar entre sus contactos 48W o 60VA

## 5.2.2 Especificaciones del Servidor

En la siguiente tabla se pueden observar las especificaciones del servidor.

*Tabla 8 - Especificaciones del Servidor*

<b>Servidor - Especificaciones</b>		
	<b>Detalles</b>	<b>Observaciones</b>
<b>CPU</b>	1GHz ARM Cortex A8	
<b>GPU</b>	OpenGL ES2.0, OpenVG 1.1 Mali 400 core	
<b>DRAM</b>	1GB	
<b>Almacenamiento en la Tarjeta</b>	8GB en una microSD	Expandible hasta 32GB
<b>Salida de Video</b>	HDMI	
<b>SO</b>	Linux3.0 + Ubuntu 12.04	
<b>Conexión con la controladora</b>	Conexión serial a través de un convertidor Serial TTL / USB	Puede utilizarse un puerto serial del pcDuino pero se debe realizar una conversión 5VDC – 3.3VDC en estos pines y viceversa
<b>Conexión de Red</b>	10/100Mbps RJ45	
<b>Alimentación</b>	5V, 2000mA	
<b>Tamaño Total</b>	125mm X 52mm	

## 5.3 Manual del Equipo

Este manual se encuentra ubicado en la sección “Anexo”.

## CONCLUSIONES

En el presente proyecto se diseñó e implementó un prototipo de un servidor web embebido en un microcontrolador para gestionar y controlar el acceso a una instalación, en este caso para el Laboratorio de Investigación y Desarrollo Electrónico de la Escuela de Ingeniería Eléctrica. A través del mismo se pudo monitorear y configurar el acceso de usuarios.

Con investigación y desarrollo del presente proyecto, se consolidó una buena base sobre los sistemas de control de acceso, su funcionamiento y elementos importantes que los conforman; sistemas embebidos y sus características más relevantes; los servidores web y su funcionamiento así como también el diseño de páginas web y el diseño y manejo de bases de datos para poder ser implementados.

Se implementó una unidad controladora que fuese capaz de recibir datos provenientes de elementos de identificación con protocolo Wiegand-26, traducirlos y convertirlos en el formato con el mismo nombre a través del uso de un microcontrolador PIC18F4550, el cual se encuentra dispuesto en una tarjeta de circuito impreso junto a los elementos necesarios para un control de acceso como lo son un relé para realizar la autorización, diversos conectores de los cuales unos fueron utilizados para implementar protocolos de comunicación, conexión de indicadores y botón para salida, todo ello haciendo uso de la gran cantidad de entradas y salidas digitales que el microcontrolador posee.

La lectura de los datos provenientes del elemento de identificación se implementó mediante las interrupciones externas que se ejecutan en el microcontrolador al detectar cambios en los terminales destinados para ello.

Aunque fue utilizado un RFID con protocolo Wiegand, también pueden ser conectados lectoras que transmitan en protocolo serial, I2C o OneWire (1-Wire), realizando pequeñas modificaciones en el firmware de la unidad controladora.

Se implementó un protocolo de comunicación serial TTL para que la unidad controladora enviara/recibiera datos hacia/desde el pcDuino.

El servidor web fue implementado en la tarjeta de desarrollo pcDuino, la cual es una computadora de única tarjeta (SBC) de bajo costo para un servidor sencillo.

El empleo de una plataforma estandarizada de hardware y software libre y que se puede ampliar, permite decir que el servidor puede ser extendido y replicado fácilmente con realizar pocos cambios.

Aunque el servidor está diseñado para implementarse en el pcDuino, el estudio realizado sobre los sistemas embebidos permite decir que puede ser migrado a otras tarjetas de desarrollo como el Raspberry PI, BeagleBoard, entre otras, sin realizar mayores cambios.

Se implementó un módulo de reloj en tiempo real, el cual sirve al pcDuino por comunicación en I2C, y al encenderlo se sincroniza con el mismo tomando fecha y hora que el dispositivo almacena.

Se implementó una base de datos en la que se almacenaron todos los registros provenientes de las consultas de la controladora, ingresos realizados, alarmas disparadas, personas autorizadas, entre otros.

El uso de una página web en el servidor, que sirve de interfaz gráfica de usuario (GUI), permitió la fácil supervisión y administración del sistema.

Con las pruebas realizadas al equipo se pudo verificar su correcto funcionamiento: permitiendo el acceso a aquellos usuarios que se encontraban habilitados o negándolo a aquellos que no, verificando la activación de las alarmas en cada caso respectivo y mostrando los datos en la página web de manera correcta.

El uso de sistemas embebidos tipo modular es una poderosa herramienta para generar aplicaciones de forma rápida y efectiva, minimizando el espacio y teniendo altas prestaciones, pudiendo ser escalables a futuro.

## RECOMENDACIONES

Para futuros proyectos se recomienda realizar la implementación del módulo USB que se encuentra en la tarjeta controladora a fin de eliminar el módulo convertidor Serial/USB FT232 y de esta manera poder utilizar el módulo UART con otra tarea, como por ejemplo conectar una lectora que trabaje bajo este protocolo.

Para aplicaciones de bajo costo, que no requieran un servidor web, se recomienda realizar la implementación del módulo del bus I2C, ya que a través del mismo pueden ser conectados diversos elementos, entre los que destacarían un RTC y memorias EEPROM, que se comuniquen por este bus de datos. Solo bastaría realizar pequeñas modificaciones en el firmware del microcontrolador para realizar rutinas de lectura/escritura en la memoria tomando la fecha y hora del RTC.

Para aplicaciones futuras, se recomienda tomar en cuenta que las entradas de las lectoras de formato Wiegand van conectadas directamente a los terminales del microcontrolador asociados a interrupciones externas, a través de estas pueden realizarse implementaciones con otros dispositivos como los iButton (que se manejan a través del protocolo 1-Wire<sup>©</sup>), pudiendo entonces ser conectados 4 dispositivos a la controladora.

Para una próxima versión de la unidad controladora, el relé presente en la tarjeta podría ser cambiado por uno que soporte mayor tensión y corriente entre sus terminales para aplicaciones que así lo requieran.

Otra mejora de la unidad controladora sería modificar el firmware del microcontrolador para colocarle un bootloader, para poder realizar actualizaciones utilizando el mismo puerto de comunicaciones (serial).

Aunque está diseñado para el pcDuino, se recomienda realizar pruebas del servidor web en otras plataformas o tarjetas de desarrollo como el Raspberry PI,

BeagleBoard, entre otros, en las que el mismo pueda correr y de esta manera determinar la capacidad de migración del proyecto.

Realizar la implementación de una página web que permita el registro de usuarios administradores desde la misma interfaz y así evitar el manejo desde la base de datos.

Realizar en el pcDuino rutinas que sean capaces de administrar las excepciones o errores causados por fallas (aparte de las alarmas) y que las mismas sean llevadas a un archivo de registro (*log file*) o sean almacenadas en la base de datos y se muestren en la interfaz de usuarios.

Para ampliar el sistema, se pueden implementar módulos convertidores Serial/Ethernet que se integren en cada unidad controladora, esto con el fin de poder realizar una red de control de accesos y los datos de las controladoras viajen a un servidor que siempre este escuchando a través de un puerto de red, que pudiese ser el mismo pcDuino con el mismo programa de control de acceso, realizando pocas modificaciones en el mismo, y en vez de escuchar un puerto serial, escuchar un puerto de red.

## REFERENCIAS BIBLIOGRÁFICAS

- Consentino, L. (2009a). *Data Técnica: Control de Accesos* [versión electrónica]. Revista Negocios de Seguridad Argentina. Vol. 45, p.p. 152, 156, 160.
- Consentino, L. (2009b). *Data Técnica: Control de Accesos* [versión electrónica]. Revista Negocios de Seguridad Argentina. Vol. 46, p.p. 156-176.
- Manual de PHP. (2014). [en línea] <<http://php.net/manual/es>> [consulta: 2014]
- MySQL 5.0 Reference Manual. (2013). [en línea] <<https://dev.mysql.com/doc/refman/5.0/es/>> [consulta: 2013].
- Perez A., David A. (2009). *Sistemas Embebidos y Sistemas Operativos Embebidos*. <[www.ciens.ucv.ve/escueladecomputacion/documentos/archivo/88+&cd=1&hl=es&ct=clnk&gl=ve](http://www.ciens.ucv.ve/escueladecomputacion/documentos/archivo/88+&cd=1&hl=es&ct=clnk&gl=ve)>. [Documento en formato pdf].
- Sierra, M. (2013). ¿Qué es un servidor y cuáles son los principales tipos de servidores. [en línea] <[http://aprenderaprogramar.com/index.php?option=com\\_content&view=article&id=542:que-es-un-servidor-y-cuales-son-los-principales-tipos-de-servidores-proxydns-webftpsmtp&catid=57:herramientas-informaticas&Itemid=179](http://aprenderaprogramar.com/index.php?option=com_content&view=article&id=542:que-es-un-servidor-y-cuales-son-los-principales-tipos-de-servidores-proxydns-webftpsmtp&catid=57:herramientas-informaticas&Itemid=179)> [consulta: 2014].
- Triviño M. y Buitrago, A. (2010). *Servidor Web Embebido en una FPGA con Codiseño como Metodología de Diseño*. <[www.laccei.org/LACCEI2010-Peru/published/ED079\\_Ariza.pdf](http://www.laccei.org/LACCEI2010-Peru/published/ED079_Ariza.pdf)>. [Documento en formato pdf].
- The Apache Software Foundation. (2012). [en línea] <<http://www.apache.org/>> [consulta: 2014].

## BIBLIOGRAFÍA

- Anónimo. C# Made Simple, ed. New Delhi, India: BPB Publications, 2001.
- Balena, F. Programación Avanzada con Microsoft® Visual Basic® .NET, ed. Madrid, España: McGraw-Hill, 2003.
- Carrasquel, J. Sistema de control maestro para el sistema de seguridad inalámbrico de la EIE de la UCV / José Carrasquel (Tesis). –Caracas: Universidad Central de Venezuela, 2006.
- Chávez, A. Red inalámbrica para aplicaciones en sistemas de seguridad / Albert Chávez (Tesis). –Caracas: Universidad Central de Venezuela, 2006.
- Crowder, D. Building a Web Site for Dummies, 4th Ed. Indiana, USA: Wiley Publishing Inc., 2010.
- Gaddis, T. Starting Out with C++, 3rd Ed. New Delhi, India: Haywood Community College, 2002.
- Gehrenbeck, E. y Balazas E. Diseño de un sistema de control horario y de acceso de personal / Ernst Gehrenbeck; Ervin Balazas (Tesis). –Caracas: Universidad Central de Venezuela, 1991.
- Gutierrez, I. Diseño de un sistema de control de acceso centralizado empleando una red distribuida / Iván Gutierrez (Tesis). –Caracas: Universidad Central de Venezuela, 1998.
- Guilarte, C. Diseño de un sistema de identificación automática con técnica de RFID / Carlos Guilarte (Tesis). –Caracas: Universidad Central de Venezuela, 1999.

Harris, A. HTML, XHTML, & CSS All-In-One for Dummies, 2nd Ed. Indiana, USA: Wiley Publishing Inc., 2011.

Mansfield, R. CSS Web Design for Dummies, ed. Indiana, USA: Wiley Publishing Inc., 2005.

McEwan, B. and Solomon D. Teach Yourself Transact-SQL in 21 Days, ed. New Delhi, India: Techmedia, 1998.

Mueller, J. HTML5 Programming with JavaScript for Dummies, ed. Indiana, USA: Wiley Publishing Inc., 2013.

Petersen, J. Absolute Beginner's guide to Databases, ed. New Delhi, India: Techmedia, 2002.

Petroutsos, E. and Ridgeway, M. Visual Basic® .NET Developer's Handbook, ed. New Delhi, India: BPB Publications, 2003.

Rodriguez, R. Implementación de un sistema de control, seguridad y acceso inalámbrico basado en estándar IEEE 802.15.4 y tecnología “ZigBee” / Rafael Rodriguez (Tesis). –Caracas: Universidad Central de Venezuela.

W3School, CSS Tutorials [en línea]. <<http://www.w3schools.com/css/>> [Consulta: 2014].

W3School, HTML Tutorials [en línea] <[http://www.w3schools.com/html/html5\\_intro.asp](http://www.w3schools.com/html/html5_intro.asp)> [Consulta: 2014].

W3School, Javascript Tutorials [en línea]. <<http://www.w3schools.com/js/>> [Consulta: 2014].

W3School, PHP Tutorials [en línea]. <<http://www.w3schools.com/php/>> [Consulta: 2014].

## **ANEXO**