

*Instituto Tecnológico de Costa Rica*

*Escuela de Ingeniería en Electrónica*



*"Monitoreo y Control de Dispositivos Residenciales vía Internet"*

*Realizado para Motorola de Costa Rica*

*Informe de Proyecto de Graduación para optar por el grado de Bachiller en  
Ingeniería en Electrónica*

*Juan Diego Calvo Chaves*

*Cartago, 2001*

*A mis padres, cuyo esfuerzo de muchos años  
esta empezando a rendir frutos.*

*A Carolina, cuyo apoyo durante mi carrera  
universitaria ha sido fundamental.*

### AGRADECIMIENTO

*Al Ing. Carlos Badilla, cuyo interés y seguimiento a este proyecto son una muestra de su alto compromiso y dedicación a sus estudiantes.*

*A los señores Alexander Saborío y Eduardo Figueroa, cuyo apoyo incondicional ha sido fundamental para el desarrollo de este proyecto.*

*Al señor Gerardo Castillo, cuyo soporte ha sido pieza clave para la correcta culminación de este proyecto.*

## ÍNDICE GENERAL

<b>INTRODUCCIÓN</b> .....	10
1.1 Descripción de la empresa .....	10
1.1.1 Descripción general .....	10
1.1.2 Descripción del departamento donde se realizó el proyecto de graduación. ....	11
1.2 Contexto en el que se inscribe el problema a resolver .....	11
<b>DELIMITACIÓN DEL PROBLEMA</b> .....	13
2.1 Definición del problema .....	13
2.2 Especificaciones del sistema .....	13
2.3 Estructura general del sistema .....	14
2.3.1 Estructura general de la unidad de monitoreo e instrucción .....	14
2.3.2 Estructura general del acceso a Internet .....	14
2.4 Solución propuesta .....	15
2.4.1 Solución propuesta para la unidad de monitoreo e instrucción.....	15
2.4.2 Solución propuesta para el acceso vía Internet .....	16
<b>OBJETIVOS</b> .....	18
3.1 Objetivo General.....	18
3.2 Objetivos Específicos.....	18
<b>PROCEDIMIENTO METODOLÓGICO</b> .....	20
4.1 Desarrollo de la unidad de monitoreo e instrucción .....	20
4.2 Hardware externo a la unidad de monitoreo e instrucción .....	21
4.3 Desarrollo de interfaz gráfica para el usuario remoto .....	21
4.4 Interfase de comunicaciones .....	22
<b>DESCRIPCIÓN DEL HARDWARE DEL SISTEMA</b> .....	23
5.1 Hardware de la unidad de monitoreo e instrucción.....	23
5.1.1 Hardware de la unidad de control .....	24
5.1.2 Hardware del sistema de seguridad.....	28
5.1.3 Hardware del sistema de luces .....	30
5.1.4 Hardware del sistema de aire acondicionado .....	32
<b>DESCRIPCIÓN DEL SOFTWARE DEL SISTEMA</b> .....	36
6.1 Descripción del software de la unidad de monitoreo e instrucción .....	36
6.2 Rutina principal de la unidad de control .....	38
6.3 Software del sistema de seguridad.....	38
6.3.1 Subrutina de inicialización del sistema de seguridad.....	39
6.3.2 Subrutina de control del sistema de seguridad .....	40
6.4 Software del sistema de luces .....	42
6.4.1 Subrutina de inicialización del control de luces.....	43
6.4.2 Subrutina de control del sistema de luces .....	45
6.5 Software del sistema de aire acondicionado.....	49
6.5.1 Subrutina de inicialización del sistema de aire acondicionado .....	51

6.5.2	Subrutina de control del sistema de aire acondicionado.....	53
6.6	Interfaz gráfica .....	54
<b>ANÁLISIS Y RESULTADOS .....</b>		<b>57</b>
7.1	Explicación del diseño .....	57
7.2	Alcances y limitaciones.....	58
<b>CONCLUSIONES Y RECOMENDACIONES.....</b>		<b>60</b>
8.1	Conclusiones .....	60
8.2	Recomendaciones .....	61
<b>BIBLIOGRAFÍA .....</b>		<b>63</b>
<b>APÉNDICES .....</b>		<b>65</b>
Apéndice 1:	Tabla de referencia rápida de información sobre el proyecto ..	65
Apéndice 2:	Descripción de términos y de herramientas utilizadas .....	66
Apéndice 3:	Programación del microcontrolador en circuito y en aplicación. .....	67

## ÍNDICE DE FIGURAS

Figura 1. Diagrama de entradas y salidas de la unidad de monitoreo e instrucción.....	15
Figura 2. Diagrama de funcionamiento del sistema con la herramienta EMIT ..	17
Figura 3. Diagrama general de la unidad de monitoreo e instrucción .....	23
Figura 4. Diagrama de bloques de la unidad de control.....	24
Figura 5. Diagrama de conexión del cristal y de la red de filtrado.....	25
Figura 6. Diagrama de circuito para el interruptor de reset .....	26
Figura 7. Diagrama de conexión de capacitores en los pines de alimentación .	28
Figura 8. Diagrama de conexión de sensores de estado del sistema de seguridad .....	29
Figura 9. Diagrama de conexión de interruptores de estado del sistema de luces .....	31
Figura 10. Hardware del modulo de entrada del sistema de aire acondicionado .....	33
Figura 11. Diagrama de flujo de la rutina principal de la unidad de control.....	37
Figura 12. Diagrama de flujo de la subrutina de inicialización del sistema de alarma .....	40
Figura 13. Subrutina de control del sistema de seguridad .....	41
Figura 14. Registro Switches en la subrutina de control de luces .....	42
Figura 15. Diagrama de flujo de la subrutina de inicialización del control de luces .....	44
Figura 16. Diagrama de flujo de la subrutina de control de luces.....	46
Figura 17. “Detección de cambios” en la subrutina de control de luces .....	47
Figura 18. “Accionamiento de luces” en la subrutina de control de luces .....	48
Figura 19. Operación del software del sistema de aire acondicionado .....	50
Figura 20. Subrutina de inicialización del control del aire acondicionado .....	52
Figura 21. Subrutina de control del sistema de aire acondicionado .....	54
Figura 22. Diagrama de la interfaz gráfica desarrollada.....	55
Figura 23. Diagrama de circuito para funcionamiento del microcontrolador en modo monitor .....	68

## ÍNDICE DE TABLAS

Tabla 1. Valores de conversión del ADC con base en temperaturas sensadas	34
Tabla 2. Variables del sistema de seguridad.....	38
Tabla 3. Variables del sistema de luces.....	43
Tabla 4. Variables del sistema de aire acondicionado .....	51

## RESUMEN

*Hoy día, cada vez más productos se están convirtiendo en productos inteligentes gracias a la incorporación de soluciones electrónicas basadas en sistemas digitales. Los productos inteligentes le permiten a las personas mantenerse en constante comunicación y con acceso a la información en todo momento y lugar.*

*Dentro de este contexto, surge la posibilidad de desarrollar una solución que le permita a las personas, desde cualquier lugar y en cualquier momento, monitorear y controlar ciertos elementos importantes de su hogar.*

*Para que la información de cada uno de los dispositivos pueda ser recolectada, y para que esos mismos dispositivos puedan ser controlados, es necesario contar en el hogar con un dispositivo electrónico que se encargue de estas tareas, y que además tenga la capacidad de reportar su estado y de recibir órdenes desde fuentes externas. En este sentido, la solución ideal para este dispositivo electrónico es la basada en un microcontrolador, el cual le brinda la versatilidad de poderse programar para la aplicación específica, un bajo costo, y un nivel de funcionalidad muy alto.*

*Para el acceso remoto, la forma mas útil y económica es hoy en día la Internet, por esto, el sistema se ha desarrollado tendiendo hacia una solución que se pueda controlar a través de Internet, mediante el desarrollo de un java applet para la interfaz de usuario, y mediante el uso de herramientas diseñadas específicamente para aplicaciones de conectividad de sistemas electrónicos con Internet.*

*Mediante el proyecto desarrollado es posible, a través de una interfaz gráfica basada en un java applet, monitorear y controlar las luces, sistema de seguridad y sistema de aire acondicionado de una residencia.*

*Sin embargo, es importante rescatar el hecho de que la solución desarrollada es fácilmente aplicable en otras áreas, tales como el control de sistemas industriales, o de sistemas en establecimientos comerciales.*

**Palabras clave:** Microcontrolador, Internet, interfaz gráfica, java applet, control residencial.



## ABSTRACT

*Today, many products are becoming intelligent devices thanks to electronic solutions based in digital systems. Intelligent systems allow full access to all the information available from a system via a constant communication with it.*

*Within these parameters, surges the possibility of developing solutions that let end users access and control, from any place at any time, certain important elements of their houses.*

*For the tasks of collecting data and controlling different devices from a remote area, a house needs to be equipped with an electronic device that has to have the function of realizing such tasks. It also has to have the ability to report its condition and to take orders from remote sources. Because of this, the ideal solution for such a device has to be based in a microcontroller, which has the versatility of letting it be programmed for an specific application, has a low cost, and a real high functionality level.*

*Today the cheapest option for remote access is through the Internet, because of this; the system has been developed towards a solution that can be controlled through the Internet by a, java applet, user interface and tools developed specifically to connect electronic devices with the Internet.*

*Thanks to the development of this project, and through a graphic user interface based on a java applet, it is possible to monitor and control the lights, security, and air conditioning systems of a household.*

*Non-the less, it is important to rescue the fact that the developed solution can be easily applied to other areas, such as the control of industrial or commercial systems.*

Keywords: Microcontroller, Internet, graphic interface, java applet, household control.

# CAPÍTULO 1

## INTRODUCCIÓN

---

### 1.1 Descripción de la empresa

#### 1.1.1 Descripción general

Fundada en 1928, Motorola es líder global en soluciones integradas de comunicaciones y soluciones electrónicas embebidas. Estos incluyen:

- a. Productos y sistemas reforzados por software para telefonía inalámbrica, radios de dos vías, mensajería y comunicaciones satelitales, así como redes y productos de acceso a Internet para consumidores, operadores de redes, y clientes comerciales, industriales y de gobierno.
- b. Soluciones de semiconductores embebidos para clientes en los mercados de redes, transportes, comunicaciones inalámbricas, imagen y entretenimiento.
- c. Sistemas electrónicos embebidos para los mercados automotriz, comunicaciones, imagen, sistemas de manufactura, computadores e industrial.
- d. Sistemas digitales y analógicos, y terminales para televisión, para operadores de televisión por cable en banda ancha.

Actualmente, Motorola se encuentra en la categoría de Industria Electrónica, entre las tres primeras de los Estados Unidos, y entre las 13 primeras del mundo. En la lista de Fortune 500 ocupa el puesto 24 en los Estados Unidos, y el 102 en el mundo. Es el sexto exportador en los Estados Unidos, y la sexta empresa en los Estados Unidos en inversión en investigación y desarrollo. Las ventas de Motorola durante 1999 ascendieron a 30.9 billones de dólares, con una inversión de 3.2 billones en investigación y desarrollo. En la actualidad Motorola cuenta con aproximadamente 148 mil empleados en 1100 oficinas distribuidas en 45 países, y 65 plantas de manufactura en 17 países.<sup>1</sup>

---

<sup>1</sup> Tomado de: <http://www.mot.com/General/inside.html>

### **1.1.2 Descripción del departamento donde se realizó el proyecto de graduación.**

El proyecto de graduación se realizó con el Sector de Productos Semiconductores de Motorola, en las oficinas de Motorola de Costa Rica.

El Sector de Productos Semiconductores es el productor número uno de procesadores embebidos, ofreciendo múltiples soluciones que permiten a sus clientes crear nuevas oportunidades de negocios en los mercados de consumo, redes y computación, transportes y comunicaciones inalámbricas.

Las oficinas locales de Motorola de Costa Rica, que sirven a toda la región centroamericana, cuentan actualmente con 30 personas representando los diferentes grupos de productos de Motorola. Cuenta con personal de ventas, soporte técnico, relaciones con el gobierno y representación corporativa de la compañía.

El Sector de Productos Semiconductores cuenta con representación en la región centroamericana a través del practicante de Ingeniero de Aplicaciones Juan Diego Calvo, persona quien desarrolló este proyecto, y quien se reporta al señor Eduardo Figueroa, Gerente Regional de Ventas basado en las oficinas de Motorola de Puerto Rico.

Como asesor local en las oficinas de Motorola de Costa Rica se contó con el apoyo del Ing. Alex Saborío, quien cuenta con amplia experiencia en el área de Internet y redes. Asimismo, se contó con el apoyo del Lic. Eduardo Figueroa en la asignación de recursos para el proyecto, así como la orientación del proyecto en términos de potenciales negocios o comercialización.

### **1.2 Contexto en el que se inscribe el problema a resolver**

Hoy en día, mientras la tecnología se vuelve cada vez más sofisticada, más productos se están convirtiendo en productos inteligentes. Los productos son cada vez más intuitivos y fáciles de utilizar para la gente, y con niveles de funcionalidad sin igual. Son más pequeños en tamaño y consumen menos potencia. Nos encontramos ante un verdadero apogeo de los sistemas inteligentes, en particular de los sistemas embebidos, que permiten utilizar la computación para gobernar todos los productos que utilizamos diariamente.

Una solución embebida se entiende como la mezcla de hardware y software que se combinan en una solución que irá dentro de un sistema mayor, y al cual le proveerá ese componente de inteligencia necesario para que el sistema sea más fácil de usar, más intuitivo, y con capacidad de tomar ciertas decisiones por sí mismo, todo esto con el fin de hacer más fácil la vida del usuario.

Actualmente, el gran impulsor del mercado embebido es la conectividad. Los dispositivos inteligentes le permiten a las personas mantenerse conectadas: el acceso a Internet nos seguirá donde quiera que vayamos al ritmo en el que los teléfonos celulares adquieren capacidades para navegar por Internet, además los automóviles se convertirán en extensiones de nuestra oficina y de nuestro hogar con los sistemas de comunicaciones y entretenimiento facilitados por la tecnología GPS y celular.

## **CAPÍTULO 2**

### **DELIMITACIÓN DEL PROBLEMA**

---

#### **2.1 Definición del problema**

Dentro del marco de los sistemas inteligentes incorporados en todos los aparatos de uso diario, incluidos los del hogar, y de la conectividad y fácil acceso a la información que nos permiten estos sistemas, surge una importante necesidad: poder monitorear y operar, desde cualquier parte, los diferentes dispositivos del hogar, siendo éstos por ejemplo sistemas de seguridad, sistemas de acondicionamiento del aire, y luces. Para ello es necesario tener una forma de monitorear y operar los dispositivos y accesorios instalados en el hogar desde donde quiera que estemos, y a la hora que sea. Por el hecho de ser posible de acceder desde cualquier parte del mundo, la Internet es el mejor medio para satisfacer esta necesidad.

#### **2.2 Especificaciones del sistema**

Para satisfacer la necesidad del monitoreo y control de dispositivos y accesorios residenciales vía Internet, es necesario desarrollar un sistema que permita, a través de Internet, conocer el estado de los dispositivos y a la vez controlarlos. Este sistema debe presentar al menos las siguientes características:

- a. Poder instalarse en el hogar.
- b. Contar con comunicación con todos los diferentes dispositivos y accesorios del hogar, susceptibles de ser gobernados computacionalmente. Esta comunicación puede ser bidireccional o unidireccional según sea el caso.
- c. Tener capacidad de monitorear la operación de los diferentes dispositivos conectados, tanto en forma sincrónica como asincrónica.
- d. Estar en capacidad de enviar instrucciones de operación a los dispositivos, ya sea en forma temporizada, o asincrónicamente por solicitud del usuario.
- e. Contar con una interfase de conexión a Internet, mediante la cual el usuario pueda recibir información acerca del estado de operación de

los dispositivos, así como enviar instrucciones de operación a los mismos.

- f. Poder construirse al costo más bajo posible, con el fin de que este sistema sea accesible a la mayoría de la población.

## **2.3 Estructura general del sistema**

Para alcanzar un correcto cumplimiento de las especificaciones del sistema, en términos generales éste debe contar con dos etapas, las cuales se describen a continuación.

### **2.3.1 Estructura general de la unidad de monitoreo e instrucción**

Es necesario desarrollar una unidad capaz de comunicarse con los diferentes dispositivos, y que a su vez permita controlar su operación.

Para el sistema de seguridad del hogar, la unidad debe ser capaz de detectar violaciones en zonas establecidas del hogar, y esta funcionalidad debe poderse habilitar o deshabilitar, es decir, el sistema de seguridad puede activarse o desactivarse.

Para el sistema de aire acondicionado, la unidad debe ser capaz de obtener los datos de temperatura en diferentes zonas del hogar, y ser capaz de accionar el sistema de aire acondicionado durante el tiempo necesario con el fin de mantener el hogar en la temperatura deseada. De igual forma, esta funcionalidad debe poderse activar o desactivar.

Por último, para las luces, la unidad debe ser capaz de conocer cuáles luces se encuentran encendidas o apagadas en el momento, y debe ser capaz de encender y apagar las diferentes luces.

### **2.3.2 Estructura general del acceso a Internet**

La unidad de monitoreo e instrucción debe ser capaz de comunicarse vía Internet cuando ésta es accedida desde un sitio remoto. Para esto, es necesario contar con una interfase de software que conecte a esta unidad con Internet, y que le permita enviar los datos acerca del estado actual de cada uno de los dispositivos del hogar, y le permita también recibir instrucciones desde el sitio remoto, con el fin de poder controlar los diferentes dispositivos del hogar.

Adicional a esto, es necesario contar con una interfaz de usuario en el sitio remoto, la cual le permita al usuario interactuar y ver desplegado el estado

actual de cada uno de los dispositivos de su hogar en una forma sencilla y clara, y que además le permita dar las órdenes de control necesarias para el funcionamiento de los diferentes dispositivos.

## 2.4 Solución propuesta

Tal como se mencionó anteriormente, el sistema está dividido, en términos generales, en dos partes: la unidad de monitoreo e instrucción, y el acceso a Internet.

### 2.4.1 Solución propuesta para la unidad de monitoreo e instrucción

Según lo planteado en las especificaciones del sistema, es necesario obtener una solución lo más barata posible. Por esto, la mejor opción es el desarrollo de una unidad hecha específicamente para esta aplicación, y que contenga únicamente lo necesario para esto. De esta forma se obtendrá la solución más efectiva en costo.

Es necesario desarrollar una unidad capaz de comunicarse con los diferentes dispositivos, y capaz de controlar su operación.

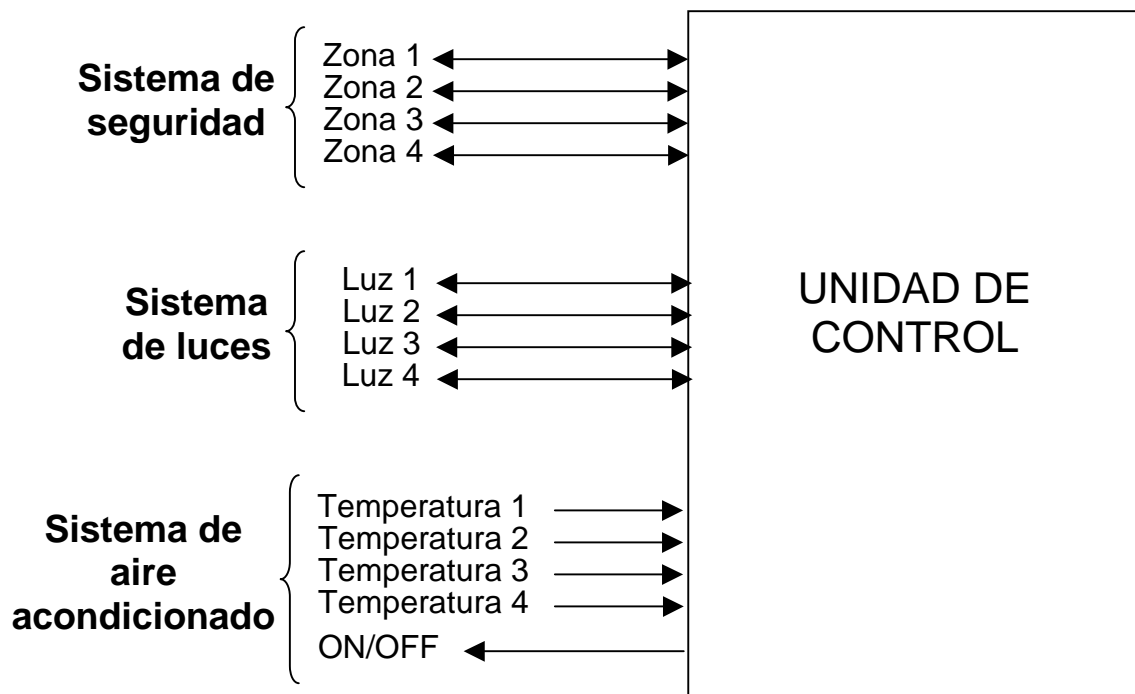


Figura 1. Diagrama de entradas y salidas de la unidad de monitoreo e instrucción

En la figura 1 se presenta el diagrama del funcionamiento de la unidad de control. Este diagrama está basado en la definición de las funciones que debe tener la unidad de monitoreo e instrucción, dadas en la primera parte de la estructura general del sistema, y basadas también en las especificaciones del sistema.

La solución proyectada para el desarrollo de la unidad de monitoreo e instrucción, es una que tenga como componente fundamental un microcontrolador, el cual tendrá básicamente las mismas entradas y salidas definidas en el diagrama de la figura 1.

Tal como puede apreciarse en la figura 1, entre otras cosas este microcontrolador debe tener puertos de conversión analógica a digital, por los cuales recibirá los valores de temperatura del sistema de aire acondicionado. De igual forma, deberá poseer cierta cantidad de pines de entrada/salida de propósito general, con los cuales pueda realizarse el control de las luces, y el chequeo de cada una de las zonas del sistema de seguridad.

#### **2.4.2 Solución propuesta para el acceso vía Internet**

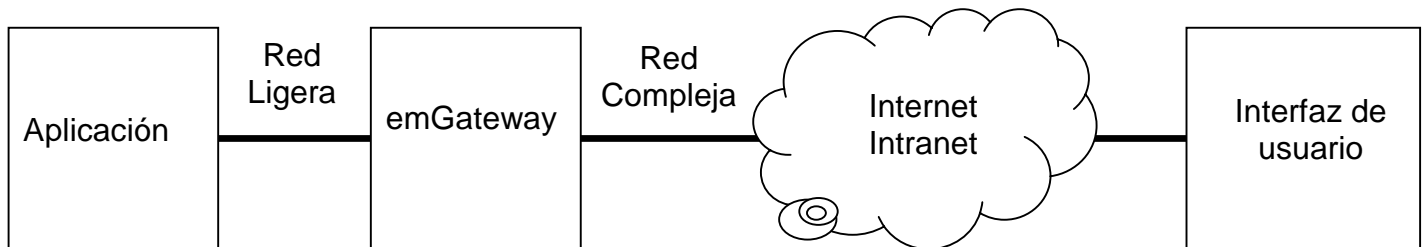
Para el acceso vía Internet a la unidad de monitoreo e instrucción, es necesario contar con una interfase entre los protocolos de transmisión de datos entre computadoras a través de Internet, y los protocolos de comunicaciones ligeros implementables y manejables con el microcontrolador de la unidad de monitoreo e instrucción, tal como el RS232.

Para esto la solución proyectada utiliza la herramienta EMIT desarrollada por la compañía emWare. La herramienta EMIT provee una interfase, conocida como emGateway, la cual funciona como puente entre redes ligeras (comunicaciones manejadas por un microcontrolador de 8 bits), y redes complejas (comunicaciones entre computadoras), facilitando así tanto el acceso a la información provista por la unidad de monitoreo e instrucción, como la transmisión de esa misma información.

En la figura 2 se presenta el diagrama del funcionamiento del sistema utilizando la herramienta EMIT. Puede verse el papel y el lugar que ocupa la interfase emGateway, sirviendo como puente entre la red ligera e Internet. Por otro lado, el bloque correspondiente a la aplicación corresponde en este caso a la unidad de monitoreo e instrucción, junto con algunas adiciones necesarias, tales como la creación de tablas de datos, así como las rutinas de comunicaciones e inicialización del sistema de comunicaciones de la herramienta EMIT.



Además de la interfase entre protocolos de transmisión de datos que debe tenerse y que es facilitada por emGateway, es necesario crear una interfaz mediante la cual el usuario pueda acceder a la información suministrada por la unidad de monitoreo e instrucción. Esta interfaz será creada a través de una página Web, lo cual permitirá darle un carácter grafico, intuitivo y amigable, a la interfaz de usuario.



**Figura 2.** Diagrama de funcionamiento del sistema con la herramienta EMIT

Normalmente las páginas de Internet están basadas en código HTML. El código HTML es un código estático, es decir, luego de una solicitud de transmisión de una página, ésta es transmitida y no habrá acción alguna por parte del navegador ni del servidor mientras no se dé otra solicitud del usuario.

Mientras tanto, en el caso del monitoreo de dispositivos que se desea efectuar, es necesario poder detectar y ver los cambios en tiempo real, es decir, se debe estar transmitiendo la información en forma continua, y por esto debe crearse lo que se conoce como una Java Runtime Interface, la cual se ejecuta en el navegador de Internet que provee la interfaz, y proveerá esta funcionalidad necesaria para el monitoreo en tiempo real.

## CAPÍTULO 3

### OBJETIVOS

---

#### 3.1 Objetivo General

Diseñar y construir un sistema que permita monitorear y controlar las luces, aire acondicionado y sistema de seguridad de una residencia, desde una interfaz gráfica basada en una página web.

#### 3.2 Objetivos Específicos

1. Diseñar y construir una etapa de hardware y software para el control de luces, la cual permita encender o apagar ciertas luces predeterminadas del hogar. Esta etapa debe ser capaz de funcionar sin alterar el control normal, basado en interruptores, de la residencia, por lo cual debe ser capaz de reconocer cuando se da un cambio en una luz debido a un interruptor, y conocer en todo momento el estado de las luces.
2. Diseñar y construir una etapa de hardware y software para el control del sistema de aire acondicionado. Este control debe ser capaz de realizar las siguientes funciones:
  - a. Conocer la temperatura ambiental de ciertas zonas predefinidas del hogar.
  - b. Ser capaz de accionar el aire acondicionado de cada zona específica cuando la temperatura de esa zona está sobre la temperatura deseada, y mantenerlo apagado cuando la temperatura de la zona es igual o menor a la temperatura deseada.
  - c. Debe poderse activar o desactivar cuando el usuario lo decida.
3. Diseñar y construir una etapa de hardware y software para el control del sistema de seguridad del hogar. Esta etapa debe realizar las siguientes funciones:
  - a. Detectar violaciones en zonas predefinidas del hogar.

- b. Poderse activar o desactivar a solicitud del usuario.
4. Desarrollar una interfaz gráfica basada en código HTML, la cual se cargue en un navegador de Internet, y permita al usuario realizar las siguientes funciones:
- a. Conocer el estado de las luces de la residencia conectadas a la etapa de control de luces, y encender o apagar cualquiera de estas luces.
  - b. Activar o desactivar el sistema de aire acondicionado.
  - c. Conocer la temperatura actual de cualquiera de las zonas predefinidas del sistema de aire acondicionado.
  - d. Aumentar o disminuir la temperatura deseada para las zonas predefinidas del sistema de aire acondicionado
  - e. Activar o desactivar el sistema de seguridad del hogar.
  - f. Conocer cuando una de las zonas predefinidas del sistema de seguridad es violada.
5. Desarrollar una etapa intermedia, basada en la herramienta EMIT, la cual permita a la interfaz gráfica cargada en el navegador de Internet de una PC, y al sistema de control instalado en el hogar, comunicarse entre sí con el fin de que el usuario pueda, desde la interfaz gráfica, realizar las funciones antes descritas.

## **CAPÍTULO 4**

### **PROCEDIMIENTO METODOLÓGICO**

---

La metodología para el desarrollo del proyecto se divide en cuatro etapas fundamentales, las cuales en términos generales deben resolverse en forma secuencial, existiendo también tareas específicas de cada etapa que pueden realizarse en paralelo.

#### **4.1 Desarrollo de la unidad de monitoreo e instrucción**

La unidad de monitoreo e instrucción está compuesta por dos partes fundamentales: hardware y software.

Dentro del hardware, ha sido necesaria la escogencia del microcontrolador a utilizar para la unidad, así como los componentes externos al microcontrolador que son indispensables para su correcto funcionamiento, como por ejemplo osciladores, protección contra ruido y otros.

En cuanto al software, es necesaria la creación de las rutinas del microcontrolador que definen la operación de la unidad de monitoreo e instrucción. Para alcanzar una mayor facilidad de creación y migración, así como la compatibilidad con las rutinas utilizadas como interfase de comunicaciones, ciertas partes del software de la unidad se desarrollaron en lenguaje C.

Para la etapa de investigación previa al desarrollo de la unidad, fue necesaria la lectura de los manuales del microcontrolador y del compilador de lenguaje C, así como el estudio de la teoría sobre lenguaje C y sobre la programación de microcontroladores en lenguaje C.

Entre los materiales requeridos para el desarrollo de la unidad de monitoreo e instrucción se encontraron entre otros el microcontrolador, componentes eléctricos varios como condensadores, resistencias, diodos, etc., placas para el montaje de los componentes. Asimismo, dentro del equipo se encuentra una computadora para la escritura y simulación del software y un sistema de desarrollo para la prueba del software en el microcontrolador.

## **4.2 Hardware externo a la unidad de monitoreo e instrucción**

El hardware externo a la unidad de monitoreo e instrucción, el cual permite realizar la emulación del hogar en pequeña escala, incluye entre otras cosas: dispositivos de potencia que servirán para accionar las luces del sistema de iluminación, bombillos, interruptores que servirán para emular los sensores de las diferentes zonas del hogar protegidas por el sistema de seguridad, sensores de temperatura que servirán para la entrada de datos de temperaturas de las diferentes zonas del hogar sensadas por el sistema de aire acondicionado, y otros componentes que servirán para desplegar cualquier variable necesaria en la presentación del modelo a pequeña escala.

Durante esta fase se realizó la escogencia de los componentes a utilizar, estudio de especificaciones de los mismos, montaje del hardware, acoplamiento con la unidad de monitoreo e instrucción, y pruebas del funcionamiento de ambos módulos en conjunto.

Entre el equipo requerido para completar esta fase se encuentran: equipo de medición de variables eléctricas, computador y sistema de desarrollo del microcontrolador, tarjeta para el montaje de los componentes, entre otros.

## **4.3 Desarrollo de interfaz gráfica para el usuario remoto**

La primera etapa de esta fase del proyecto es la del estudio sobre la programación del aplicativo Java necesario para el monitoreo en tiempo real, así como el estudio del funcionamiento del software a utilizar en el desarrollo de esta fase (Symantec Visual Café). La etapa siguiente es el desarrollo del aplicativo que se carga en el navegador de Internet, y que funciona como interfaz para el usuario remoto.

Durante esta fase del proyecto el equipo fundamental es una computadora, en la cual se carga el software de desarrollo, y se desarrolla el programa de aplicación.

Debe notarse que la interfaz gráfica va directamente relacionada con lo desarrollado en las fases anteriores del proyecto, pues las variables que puedan ser sensadas y controladas por la unidad de monitoreo e instrucción, serán las mismas que puedan ser observadas y controladas a través de la interfaz de usuario.

#### **4.4 Interfase de comunicaciones**

Entre otras cosas, durante esta fase se realizan ciertas modificaciones al software de la unidad de monitoreo e instrucción que permiten efectuar las comunicaciones, y como etapa final se realiza el montaje de todo el sistema con el fin de realizar las pruebas y la posterior presentación del mismo.

Para esta fase del proyecto es necesario el uso del sistema de desarrollo NET.08, el cual está basado en la arquitectura de microcontroladores de la familia HC08 de Motorola, específicamente el microcontrolador MC68HC908GP32. Este sistema de desarrollo incluye lo necesario para realizar la adaptación de la interfase emGateway entre la aplicación (unidad de monitoreo e instrucción ya desarrollada), y la interfaz de usuario basada en HTML ya desarrollada. Asimismo, la computadora sigue siendo una parte fundamental del equipo pues en ésta se realizan los cambios de software necesarios, así como la prueba final del sistema.

Previa al inicio de esta fase es necesaria la lectura en forma específica de todo lo referente a la interfase, así como al sistema de desarrollo NET.08.

## CAPÍTULO 5

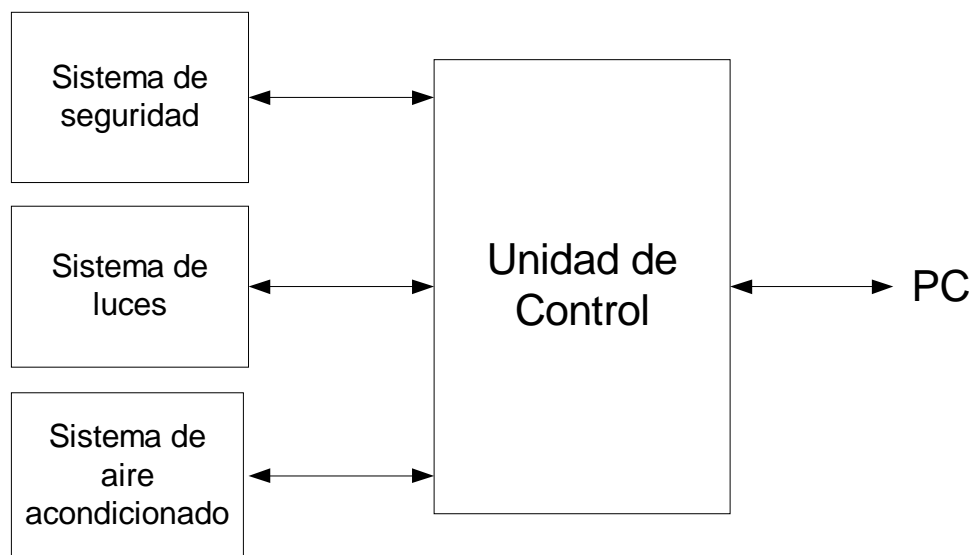
### DESCRIPCIÓN DEL HARDWARE DEL SISTEMA

---

#### 5.1 Hardware de la unidad de monitoreo e instrucción

La unidad de monitoreo e instrucción cuenta con un módulo principal y tres módulos secundarios. El módulo principal consiste en el microcontrolador de la unidad de control, junto con la circuitería externa necesaria para el correcto funcionamiento del mismo.

Los tres módulos secundarios de la unidad de monitoreo e instrucción corresponden al hardware necesario para controlar cada una de las funciones principales del sistema: sistema de seguridad, sistema de luces, y sistema de aire acondicionado. La figura 3 muestra la estructura general de la unidad de monitoreo e instrucción.



**Figura 3.** Diagrama general de la unidad de monitoreo e instrucción

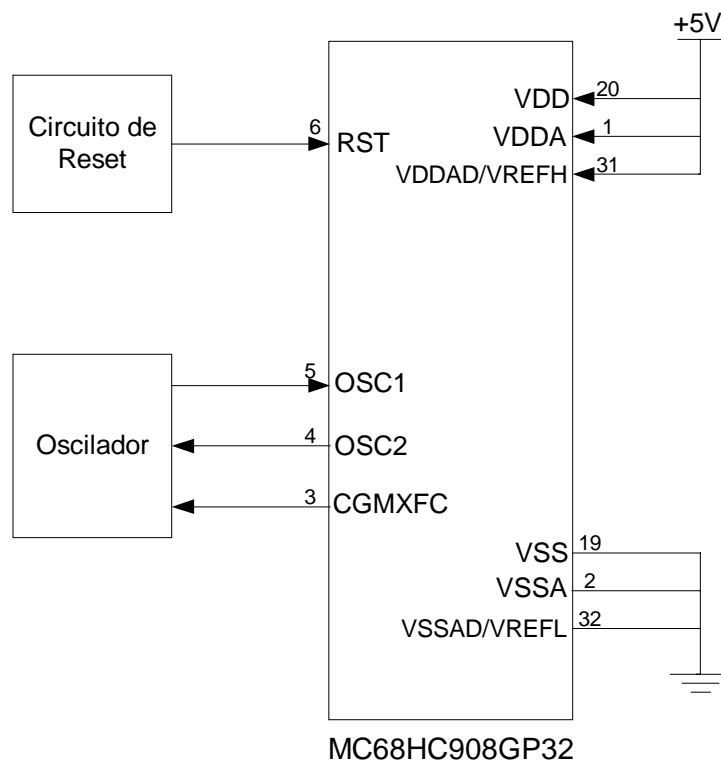
### 5.1.1 Hardware de la unidad de control

Este módulo consta del microcontrolador de la unidad, que corresponde al microcontrolador de Motorola MC68HC908GP32. Asimismo, este módulo contiene toda la circuitería externa al microcontrolador, y que es indispensable para que éste funcione correctamente.

Entre otras cosas, la circuitería externa al microcontrolador está constituida por los siguientes elementos:

- a. Oscilador de cuarzo que provee la señal de reloj al microcontrolador.
- b. Circuito de reset del microcontrolador.
- c. Voltajes de alimentación del microcontrolador.
- d. Capacitores para filtración en pines de alimentación del microcontrolador.

La figura 4 muestra un diagrama general de la conexión de esta circuitería externa, y provee una referencia acerca de los pines del microcontrolador utilizados para este fin.



**Figura 4.** Diagrama de bloques de la unidad de control

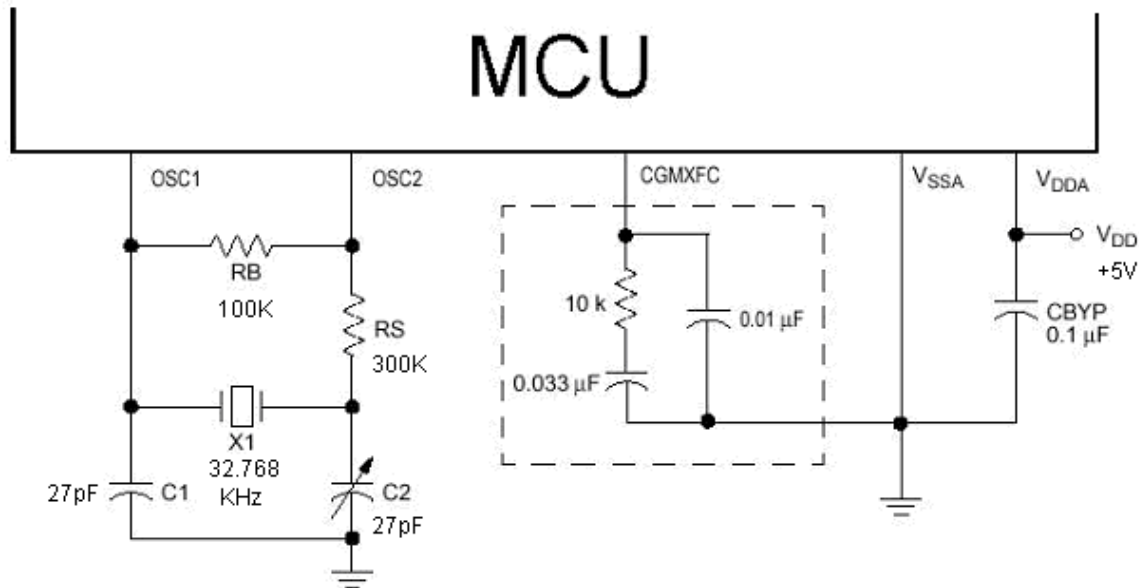


A continuación se detalla cada uno de los elementos periféricos mostrados en la figura 4, y que son indispensables para el correcto funcionamiento del microcontrolador.

*a. Módulo del oscilador*

El módulo del oscilador está constituido por un cristal de cuarzo como componente principal, y rodeado por los capacitores y resistencias indispensables para su correcta oscilación. Forma también parte de este módulo la red que sirve al PLL (Phase Locked Loop), del microcontrolador para filtrar las correcciones de fase a través del pin CGMXFC.

En la figura 5 se muestran los componentes de este módulo, así como su interconexión, tanto para los componentes que rodean al cristal, como para la red de filtrado. El circuito mostrado en la figura 5 es el recomendado por el fabricante en el manual del microcontrolador.



**Figura 5.** Diagrama de conexión del cristal y de la red de filtrado

Los capacitores C1 y C2 externos al cristal son ambos de 27pF, mientras que las resistencias RB y RS son de 10MΩ y 300KΩ respectivamente.

El cristal X1 utilizado es de 32.768KHz. A través de la programación del PLL (Phased Locked Loop) del microcontrolador, esta frecuencia de cristal externo de 32.768KHz, es convertida en una frecuencia de bus interna de 7.3728MHz. La razón para la escogencia de esta frecuencia de operación se

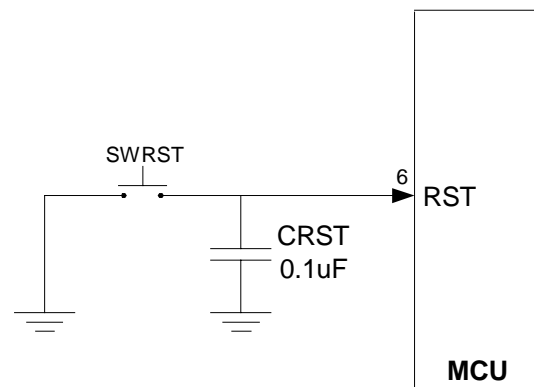
debe principalmente a la programación del SCI (Serial Communications Interface), del microcontrolador, pues esta frecuencia dividida entre 64 da exactamente 115200, que es en baudios la velocidad máxima para comunicación serial con el PC.

A través de la red de filtrado, mostrada dentro del recuadro con líneas punteadas en la figura 5, el generador de reloj del microcontrolador filtra las correcciones de fase realizadas por el PLL. Esta red podría ser reemplazada por un solo capacitor de  $0.47\mu\text{F}$ , sin embargo la red mostrada en el recuadro es la que se utiliza para obtener la mayor estabilidad en el oscilador.

El capacitor CBYP de  $0.1\mu\text{F}$  es uno de los capacitores, también de filtrado, que deben conectarse entre VDD y VSS en distintos puntos del microcontrolador. Esto se verá con más detalle en el punto “d” de la presente sección.

#### *b. Circuito de reset*

El circuito de reset consiste en el botón que permite realizar el “hardware reset” del microcontrolador. En la figura 6 se muestra el diagrama de este circuito.



**Figura 6.** Diagrama de circuito para el interruptor de reset

El pin de reset del microcontrolador (RST), es activo en bajo. Debido a que el microcontrolador cuenta con pull-up interno, no es necesario colocar una resistencia a VDD para mantener la entrada RST en alto mientras no se presione el botón.

#### *c. Voltajes de alimentación del microcontrolador*

En la figura 4 es posible apreciar los pines del microcontrolador conectados a un potencial de 5V, y los pines conectados a tierra.

Los pines VDD y VSS corresponden a la alimentación misma del microcontrolador. El pin VDD corresponde al voltaje de alimentación alto (+5V), mientras que el pin VSS corresponde a la tierra del microcontrolador.

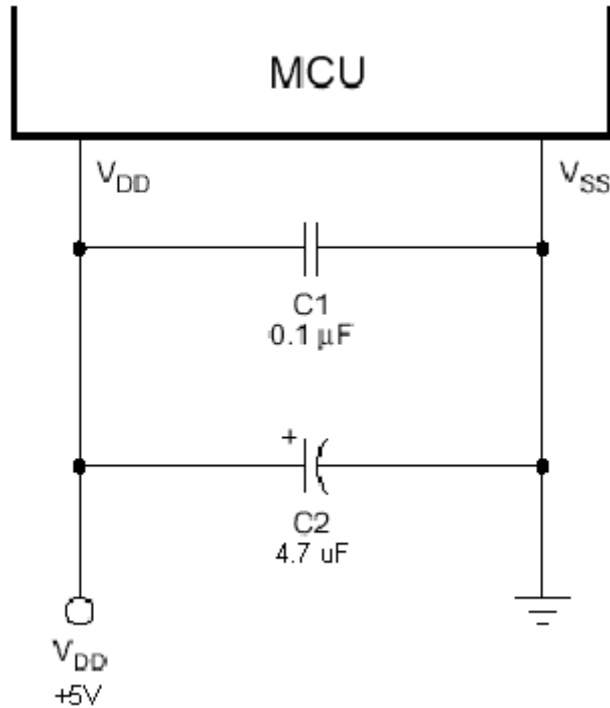
Por otro lado, tal como se muestra en la figura 4, existen otros pines también conectados a los mismos niveles de voltaje. Estos corresponden a VDDA y VSSA, y a VDDAD/VREFH y VSSAD/VREFL. VDDA y VSSA corresponden a los niveles de referencia del oscilador del microcontrolador, y son los límites alto y bajo utilizados por el oscilador interno para generar la señal de reloj para el microcontrolador. Por otro lado, los pines VDDAD/VREFH y VSSAD/VREFL corresponden al convertidor analógico-digital (ADC), y éstos son los voltajes de referencia utilizados por el ADC para realizar las conversiones, es decir, cuando el voltaje de entrada del ADC es igual a VREFH, el resultado de la conversión será el valor hexadecimal FF, que corresponde al valor máximo por tratarse de un convertidor de 8 bits; por otro lado, si el voltaje de entrada del ADC es igual a VREFL, el resultado de la conversión será igual a 0 (valor mínimo).

#### *d. Capacitores en pines de alimentación del microcontrolador*

Debido a las rápidas transiciones de señales en los pines del microcontrolador, se dan altas demandas de corriente de corta duración. Con el fin de filtrar el ruido que podría generarse por esto y por otras fuentes externas, el fabricante recomienda colocar capacitores entre los pines de voltaje alto y bajo (VDD y VSS). En la figura 7 se ilustra con un diagrama de circuito la conexión de estos capacitores.

La conexión de estos capacitores se realiza tanto entre los pines VDD y VSS, como entre el resto de pines conectados a +5V y a tierra (VDDA-VSSA, y VDDAD-VSSAD).

Con el fin de hacer el circuito lo más inmune al ruido como sea posible, el fabricante recomienda colocar estos capacitores lo más cerca posible al encapsulado del microcontrolador.



**Figura 7.** Diagrama de conexión de capacitores en los pines de alimentación

El capacitor C2 mostrado en el figura 7, es recomendado en aplicaciones en que se requiera que los pines de los puertos entreguen cantidades relativamente grandes de corriente. En esta aplicación se ha utilizado un capacitor con un valor de  $4.7 \mu F$ .

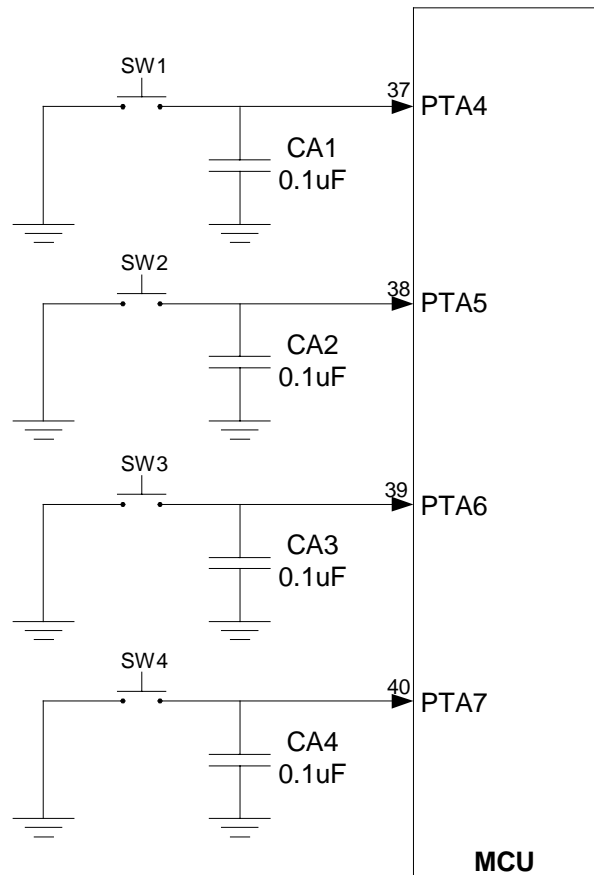
### 5.1.2 Hardware del sistema de seguridad

El sistema de seguridad tiene como salidas las señales de alarma que van al microcontrolador. Estas señales son generadas por interruptores que, dependiendo de la aplicación, pueden ser de muy diversos tipos, tales como sensores de puertas, de ventanas, sensores infrarrojos, sensores de movimiento, e incluso sensores de humo si el propósito fuera detectar incendio. Todos estos sensores tienen como característica común el hecho de que funcionan con dos estados, encendido y apagado.

Para los efectos del sistema de control, todos estos sensores representan interruptores que hacen que las correspondientes entradas del microcontrolador oscilen entre los valores lógicos de cero y uno.

Para los efectos de este proyecto y del software desarrollado para el control del sistema de seguridad, un cero lógico en uno de los pines de entrada del microcontrolador representa que la zona correspondiente ha sido violada, mientras que un uno lógico representa que la zona se encuentra segura. Esto

se diseñó de esta forma con el fin de ahorrar en componentes externos, debido a que es posible habilitar, por software, resistencias de pull-up internas en los pines de entrada del microcontrolador, lo cual representa un ahorro de una resistencia por cada entrada, que se traduce en ahorro de espacio en un eventual circuito impreso, y el consiguiente ahorro en costo que esto representa.



**Figura 8.** Diagrama de conexión de sensores de estado del sistema de seguridad

En la figura 8 se presenta el diagrama del módulo del sistema de seguridad. En la figura se presentan los posibles sensores como simples interruptores de dos estados. Asimismo, puede apreciarse que no existen resistencias de pull-up externas pues éstas las provee el microcontrolador internamente y son habilitadas por software.

Tal como se muestra en la figura 8, los sensores del sistema de seguridad, que para el microcontrolador son vistos como simples interruptores, están conectados a los pines PTA[4..7], es decir, están conectados a cuatro pines del puerto A del microcontrolador, dejando disponibles los otros cuatro pines del puerto A, los cuales en este caso son usados para el sistema de luces.

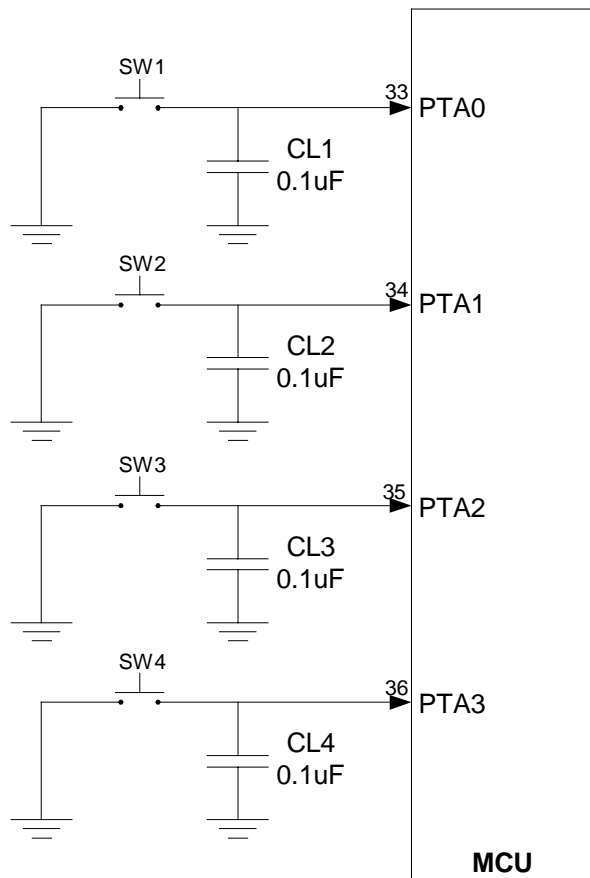
### 5.1.3 Hardware del sistema de luces

El hardware del sistema de luces esta constituido de manera principal por dos partes: módulo de entrada y módulo de salida.

#### *a. Módulo de entrada del sistema de luces*

Este módulo lo constituyen los interruptores de luces que forman parte de la instalación eléctrica del hogar. Para los efectos de la unidad de control, estos interruptores son entradas, que a través de los pines PTA[0..3] son sensados por el microcontrolador, con el fin de conocer su estado y poder luego accionar los controles de las luces dependiendo de los estados en que se encuentren dichos interruptores.

Al igual que con los interruptores del sistema de seguridad, los interruptores de las luces van a acompañados de otros componentes que permiten su correcto funcionamiento. De igual forma, se habilitan, por software, resistencias de pull-up internas en el microcontrolador, con lo cual no es necesario colocar resistencias de pull-up externas. En la figura 9 se ilustra la conexión de estos interruptores.



**Figura 9.** Diagrama de conexión de interruptores de estado del sistema de luces

*b. Módulo de salida del sistema de luces*

El segundo módulo del sistema corresponde a las luces que son accionadas por el microcontrolador, es decir, a la salida del microcontrolador que se encarga de controlar las luces. Como se ha descrito anteriormente, las luces pueden ser controladas de dos formas distintas: con los interruptores del sistema de luces, o desde el PC. Por lo tanto, el microcontrolador es capaz de accionar las luces cuando sea necesario, indistintamente de si este accionamiento se da por cambio en un interruptor o por una orden del PC.

Es importante resaltar el hecho de que no hay ningún tipo de accionamiento directo de las luces en esta aplicación, es decir, todo interruptor funciona como entrada del microcontrolador, y éste es el encargado de accionar las luces a través de su salida. La razón de esto es que es necesario que el microcontrolador conozca en todo momento el estado de cada una de las luces, con el fin de que pueda enviar esta información al PC cuando le sea solicitada.

Las salidas del microcontrolador utilizadas para el accionamiento de las cuatro luces son los pines PTD[2..5]. El resto de pines del puerto D (PTD[0..1]), no son utilizados en este sistema.

*c. Coexistencia con el sistema de luces convencional*

Debido a la fuerte necesidad de poder controlar las luces del hogar a través de los interruptores convencionales, el sistema se ha diseñado de forma tal que permita la utilización de los interruptores de luces convencionales como parte del control de las luces del hogar.

Para el logro de esto, los interruptores del hogar actúan como entradas del sistema de control, siendo éstos los representados en la figura 9 como interruptores del sistema de luces.

Sin embargo, el funcionamiento de estos interruptores al incorporar el sistema de control es diferente al convencional, pues lo leído por el sistema de control son los cambios de estado de los interruptores, y no su estado, por lo cual con un simple cambio de estado es posible encender o apagar una luz, independientemente del cambio que se haya dado, o del estado inicial y final del interruptor.

Esta forma de operación brinda la flexibilidad necesaria para que las luces puedan ser controladas tanto a través de la interfaz grafica como a través de los interruptores convencionales, y sin que ninguna de las dos formas represente un obstáculo para el correcto funcionamiento de la otra.

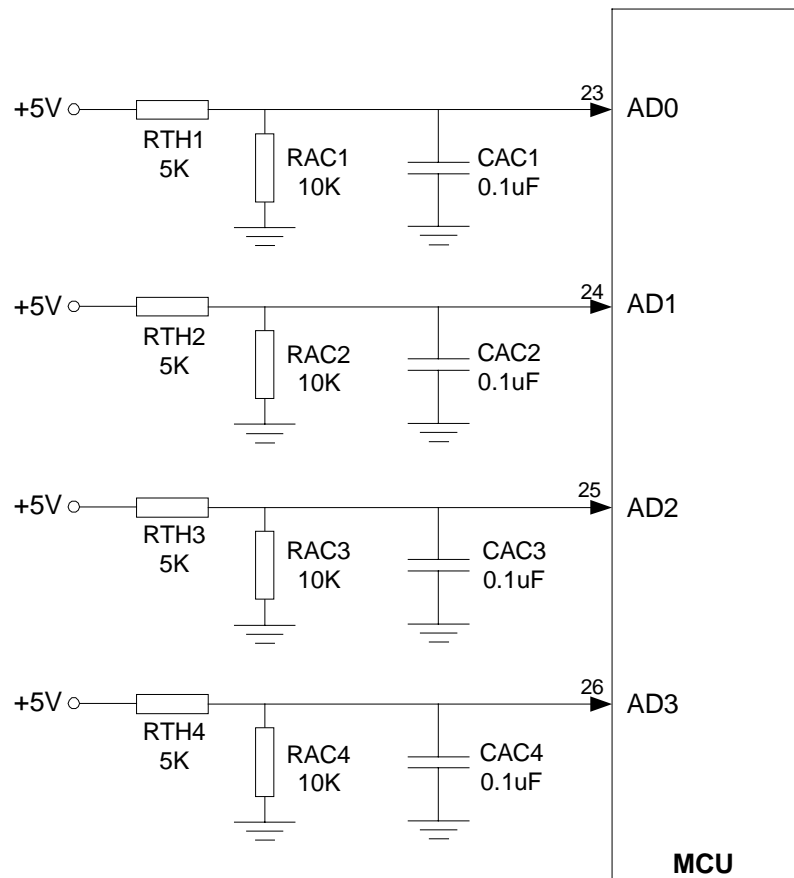
#### **5.1.4 Hardware del sistema de aire acondicionado**

Al igual que el sistema de luces, el hardware del sistema de aire acondicionado está constituido por dos módulos, uno de entrada y otro de salida. A continuación se brinda un detalle de cada uno de estos módulos.

*a. Módulo de entrada del sistema de aire acondicionado*

Este módulo está compuesto por el circuito sensor de temperatura, el cual es constituido por los termistores, que funcionan como sensores, junto con todos los componentes necesarios para la generación de voltajes adecuados en los pines del ADC, a partir de los niveles de temperatura sensados por los termistores. En la figura 10 se presenta el diagrama de este módulo de entrada del sistema de aire acondicionado.





**Figura 10.** Hardware del modulo de entrada del sistema de aire acondicionado

En la figura 10, las resistencias RTH[1..4] representan a los termistores de 5K $\Omega$  utilizados como sensores de temperatura. Se han utilizado termistores marca Siemens con coeficiente de temperatura negativo y con encapsulado para ambientes altamente húmedos.

En esta aplicación se aprovecha el cambio de voltaje en los terminales de los termistores, el cual es causado por el cambio de resistencia debido a la temperatura (característica de estos componentes). Este voltaje en los terminales de los termistores es el que se envía como entrada del convertidor analógico-digital del microcontrolador, para la lectura de los valores de temperatura en cada una de las zonas.

Con base en los datos de los termistores y del ADC, se llegó a una tabla que contiene los valores de temperatura posibles, con los valores de conversión del ADC asociados a ellos. En la tabla 1 se presentan estos valores. Los valores de esta tabla son los utilizados por el software para la lectura y manejo de temperaturas en el módulo de control del sistema de aire acondicionado.

**Tabla 1.** Valores de conversión del ADC con base en temperaturas sensadas

Temperatura en el sensor (°C)	Voltaje en el pin del ADC (V)	Conversión En el ADC (HEX)
30	3.6402	186
29	3.5807	183
28	3.5212	180
27	3.4617	177
26	3.4022	174
25	3.3427	170
24	3.2832	167
23	3.2237	164
22	3.1642	161
21	3.1047	158
20	3.0452	155
19	2.9857	152
18	2.9262	149
17	2.8667	146
16	2.8072	143
15	2.7477	140
14	2.6882	137
13	2.6287	134
12	2.5692	131
11	2.5097	128
10	2.4502	125
9	2.3907	122
8	2.3312	119
7	2.2717	116
6	2.2122	113
5	2.1527	110
4	2.0932	107
3	2.0337	104
2	1.9742	101
1	1.9147	98
0	1.8552	95

*b. Módulo de salida del sistema de aire acondicionado*

El segundo módulo del sistema de aire acondicionado es el correspondiente a las salidas del microcontrolador que activan y desactivan el aire acondicionado de cada una de las zonas.

Para cada una de las zonas existe un valor deseado de temperatura, y con base en este valor y el valor de temperatura leído, el microcontrolador toma la decisión de encender o apagar el aire acondicionado en cada zona. Es decir,

si para una determinada zona la temperatura leída es menor o igual a la temperatura deseada, el microcontrolador mantiene apagado el aire acondicionado en esa zona, mientras que si la temperatura leída es mayor que la temperatura deseada, el microcontrolador enciende el aire en esa zona, y lo mantendrá encendido hasta que la temperatura baje al valor deseado.

## CAPÍTULO 6

### DESCRIPCIÓN DEL SOFTWARE DEL SISTEMA

---

#### 6.1 Descripción del software de la unidad de monitoreo e instrucción

El software de esta unidad es el que se ejecuta en el microcontrolador, y mediante el cual se controlan en el hogar las luces, el sistema de aire acondicionado, y el sistema de seguridad.

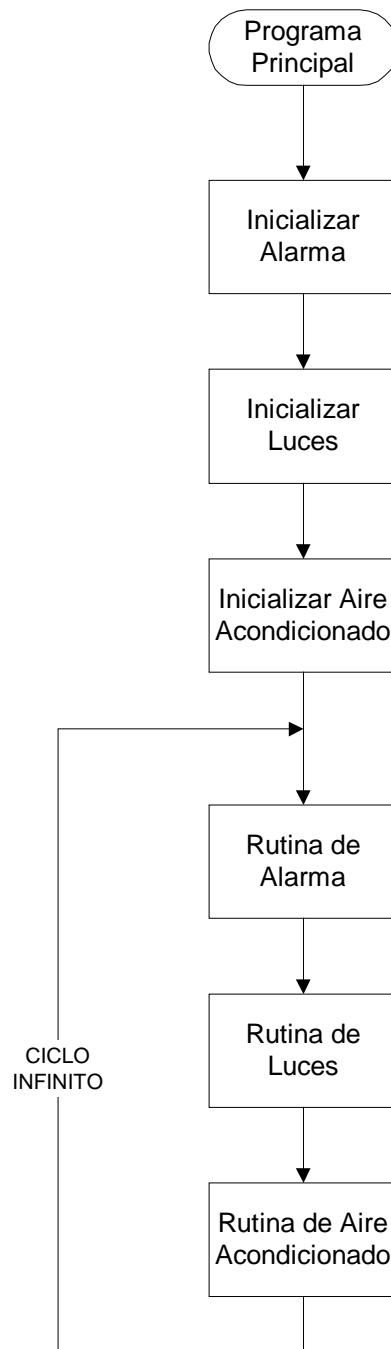
Entre otras cosas, el software con que se programa el microcontrolador de la unidad de monitoreo e instrucción la hace capaz de encender y apagar las luces, conocer el estado actual de las luces, sensar la temperatura de las diferentes zonas del hogar, activar o desactivar los sistemas de aire acondicionado dependiendo de los valores de dichas temperaturas, así como detectar violación de zonas del sistema de seguridad cuando éste se encuentra activado.

Al software de la unidad de monitoreo e instrucción, una vez que se encuentra funcionando, se le deben agregar algunas rutinas que se encargan de realizar la comunicación con la interfaz gráfica y el almacenamiento de las variables en cierta forma, con el fin de poder accederlas desde la computadora.

Con el fin de hacer este proceso de agregar rutinas lo más suave posible, y de que una vez hecho no haya conflicto entre las diferentes partes del software, existen ciertos requerimientos planteados para el software desarrollado para la unidad de monitoreo e instrucción. El principal de estos requerimientos es que la unidad debe programarse bajo un sistema de “multitasking cooperativo”, esto significa que el mismo debe tener un programa principal que se dedique únicamente a hacer llamados a subrutinas para cada una de las tareas, y que al finalizar la última subrutina, regrese a llamar nuevamente a la primera, y se mantenga de esta manera en este ciclo infinito.

Dentro del contexto del “multitasking cooperativo”, para las subrutinas se tiene como principal requerimiento el que éstas no mantengan ocupado al microcontrolador por un período muy grande de tiempo. Por ejemplo, que no haya partes en las subrutinas en donde el microcontrolador deba quedarse sin realizar operación alguna esperando a que suceda un evento, como podría ser una conversión del convertidor analógico/digital, o un conteo del temporizador del microcontrolador. Asimismo, no debe haber en el software interrupciones que saquen al microcontrolador, por un intervalo prolongado de tiempo, de la tarea que se encontraba realizando.

A continuación se describirá la rutina principal de la unidad de control, que es la encargada de hacer los llamados a cada una de las subrutinas. Posteriormente, se describirán los módulos de software desarrollados para cada una de las funciones de la unidad de control: sistema de seguridad, sistema de luces y sistema de aire acondicionado.



**Figura 11.** Diagrama de flujo de la rutina principal de la unidad de control

## 6.2 Rutina principal de la unidad de control

En la figura 11 es posible apreciar el diagrama de flujo de la rutina principal del software de la unidad de monitoreo e instrucción. Esta rutina comienza con la inicialización de cada una de las funciones de la unidad, la cual se realiza una sola vez, para luego entrar a un ciclo infinito (loop), en el que secuencialmente se hacen llamados a las rutinas que controlan cada una de las funciones de la unidad (sistema de seguridad, sistema de luces y sistema de aire acondicionado).

## 6.3 Software del sistema de seguridad

Este módulo del software de la unidad de control, es el encargado de controlar todos los procesos que tienen que ver con la operación del sistema de seguridad.

A continuación se enumeran algunos puntos que sirven como descripción acerca del funcionamiento de este módulo y del sistema de seguridad en general:

- Se monitorean cuatro zonas diferentes.
- A partir de que se activa el sistema, cualquier cambio en una zona genera alarma.
- Si al activarse el sistema una de las zonas se encuentra abierta, se generará de inmediato la respectiva alarma.
- La lectura de zonas se realiza a través de los pines PTA[4..7] del microcontrolador.
- Existen 4 registros, Alarm[1..4], que indican el estado de cada zona (si Alarmx=1 entonces la zona ha sido violada).
- El registro AlarmON indica si el sistema está activado o desactivado (AlarmON=1: sistema activado).

En la tabla 2 se brinda una descripción detallada de las variables que constituyen este módulo.

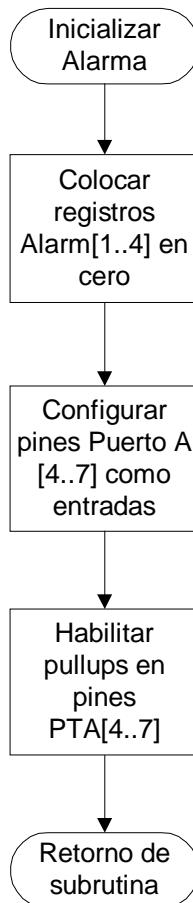
**Tabla 2.** Variables del sistema de seguridad

Variable	Tipo	Descripción	Posibles Valores	Ubicación
AlarmON	Byte	Indica si el sistema de alarma se encuentra encendido o apagado.	Binario, 0 y 1	RAM
Alarm1	Byte	Indica si la zona 1 ha sido violada.	Binario, 0 y 1	RAM

<b>Alarm2</b>	Byte	Indica si la zona 2 ha sido violada.	Binario, 0 y 1	RAM
<b>Alarm3</b>	Byte	Indica si la zona 3 ha sido violada.	Binario, 0 y 1	RAM
<b>Alarm4</b>	Byte	Indica si la zona 4 ha sido violada.	Binario, 0 y 1	RAM
<b>Temporal</b>	Byte	Almacena valores a comparar en las estructuras de decisión de la subrutina.	00h a FFh	RAM

### 6.3.1 Subrutina de inicialización del sistema de seguridad

Como se dijo anteriormente, la rutina principal incluye llamados a tres subrutinas que realizan la inicialización de cada una de las funciones de la unidad de monitoreo e instrucción. En primer lugar se encuentra la subrutina de inicialización del sistema de alarma, la cual básicamente inicializa en cero los registros de estado de las alarmas, configura como entradas los pines del puerto de entrada/salida del microcontrolador utilizados para el sistema de alarma (PTA[4..7]), y habilita las resistencias internas de pull-up en los pines del puerto A utilizados por el sistema de seguridad. En la figura 12 es posible apreciar el diagrama de flujo de esta subrutina.



**Figura 12.** Diagrama de flujo de la subrutina de inicialización del sistema de alarma

### 6.3.2 Subrutina de control del sistema de seguridad

Seguidamente se tiene la subrutina que se encarga propiamente del control de las funciones del sistema de seguridad. Esta subrutina es invocada por la rutina principal en el ciclo infinito (loop), durante la ejecución del programa de la unidad.

En la figura 13 es posible apreciar el diagrama de flujo de esta subrutina. Esta subrutina primero verifica si el sistema de alarma está encendido, en caso de no estar encendido, sale de la subrutina; y en caso que esté encendido, realiza un sondeo de cada una de las zonas, verificando si ha sido violada o no. Cuando una zona ha sido violada, coloca un uno en el registro "Alarm" correspondiente a esa zona. Después de examinar todas las zonas, sale de la subrutina.



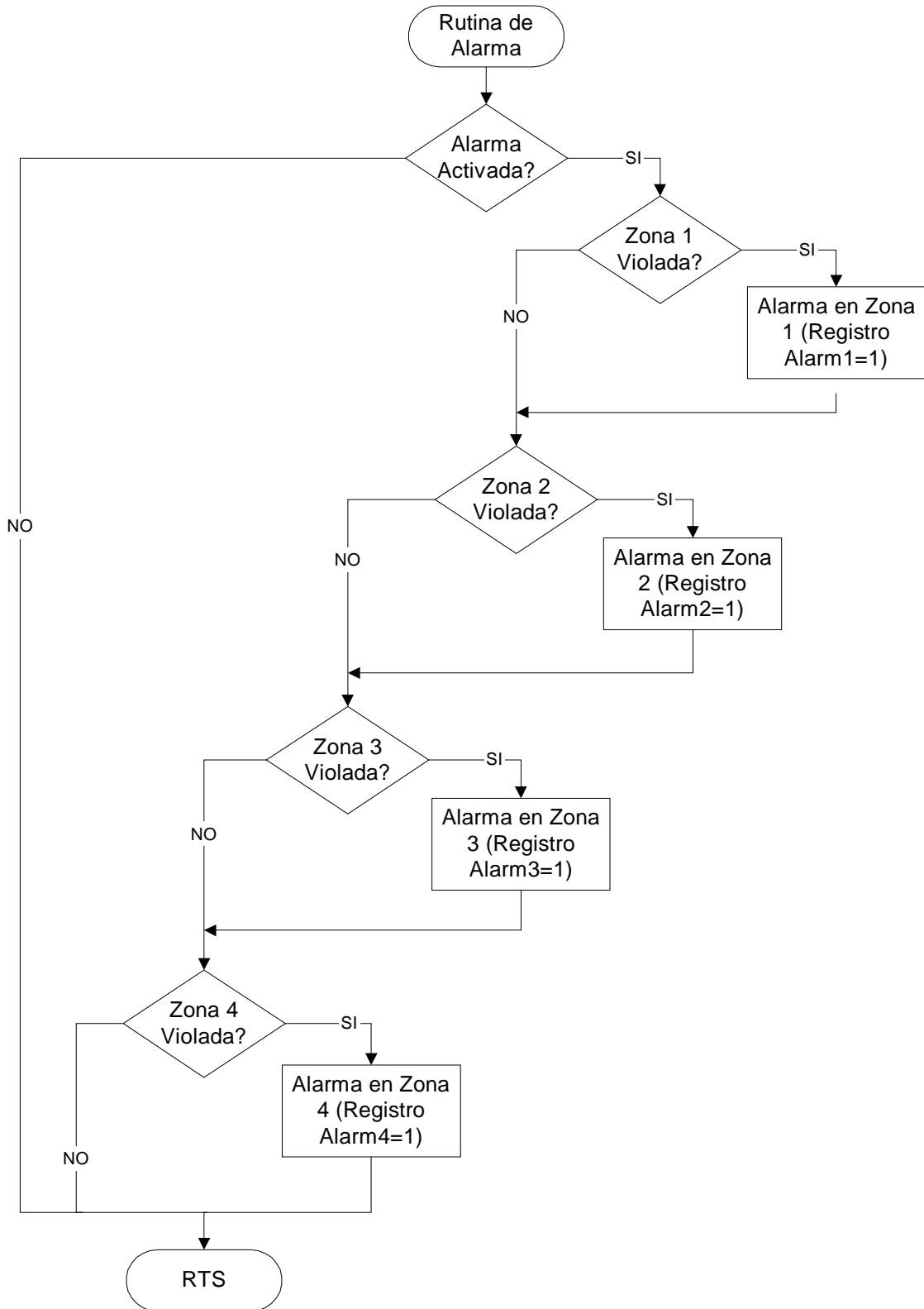


Figura 13. Subrutina de control del sistema de seguridad

## 6.4 Software del sistema de luces

Tal como su nombre lo indica, corresponde a los módulos de software encargados de asegurar la correcta operación del sistema de luces. Al igual que con el sistema de seguridad, para el control de este sistema existen dos módulos de software, uno correspondiente al de inicialización del sistema, y otro correspondiente al módulo principal de control. Algunas de las características principales son:

- Se controlan cuatro luces diferentes, y son monitoreados sus respectivos interruptores.
- Todas las luces arrancan apagadas independientemente del estado inicial de los interruptores, y luego el encendido o apagado de las luces se da con simples cambios con respecto al estado inicial.
- Las cuatro luces son controladas a través de los pines PTD[2..5] del microcontrolador.
- Los interruptores de luces son monitoreados a través de los pines PTA[0..3] del microcontrolador.
- Existen cuatro registros Light[1..4] que indican el estado actual de cada una de las luces (encendida o apagada). Estos registros son actualizados cada vez que el software entra a la subrutina de control del sistema de luces, siempre y cuando haya habido cambio en al menos uno de los interruptores.
- Antes de salir de la subrutina de luces, se da el accionamiento de las mismas dependiendo del estado de los registros Light[1..4].
- El registro Switches es de doble propósito. Se utiliza para almacenar los valores de los interruptores de luces leídos la última vez que el software entró a la subrutina de luces. Asimismo, durante la subrutina de luces, este registro se utiliza para indicar los interruptores que han cambiado de estado desde la última vez que se entró a la subrutina. En la siguiente figura se ilustra el funcionamiento de este registro.

X	X	X	X	SW4	SW3	SW2	SW1
7	6	5	4	3	2	1	0

**Figura 14.** Registro Switches en la subrutina de control de luces

Como se aprecia en la figura, los bits 0 a 3 del registro Switches indican el estado de los interruptores 1 a 4 del sistema de luces. Asimismo, durante la ejecución de la subrutina de control de luces, estos mismos bits son utilizados para representar los interruptores que han cambiado desde la última vez que el software entró a la subrutina (SWx=1: interruptor x cambió; SWx=0: interruptor x no cambió).

A continuación se detallan las variables del sistema utilizadas por los módulos de software del sistema de luces.

**Tabla 3.** Variables del sistema de luces

<b>Variable</b>	<b>Tipo</b>	<b>Descripción</b>	<b>Posibles Valores</b>	<b>Ubicación</b>
<b>Light1</b>	Byte	Indica el estado actual de la luz número 1.	Binario, 0 y 1	RAM
<b>Light2</b>	Byte	Indica el estado actual de la luz número 2.	Binario, 0 y 1	RAM
<b>Light3</b>	Byte	Indica el estado actual de la luz número 3.	Binario, 0 y 1	RAM
<b>Light4</b>	Byte	Indica el estado actual de la luz número 4.	Binario, 0 y 1	RAM
<b>Switches</b>	Byte	1. Almacena estado de los interruptores; 2. Indica interruptores cambiados	00h a 01h	RAM
<b>Temporal</b>	Byte	Almacena valores a comparar en las estructuras de decisión de la subrutina.	00h a FFh	RAM

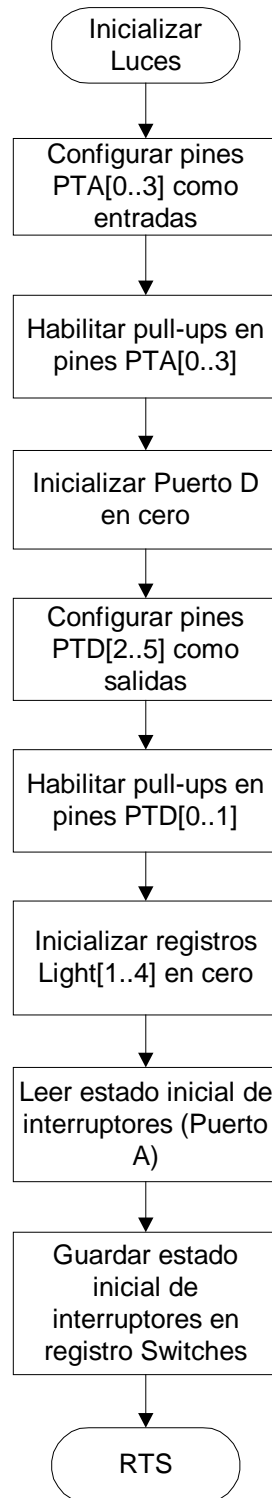
A continuación se detallan los dos módulos que componen el software de control del sistema de luces.

#### **6.4.1 Subrutina de inicialización del control de luces**

Esta subrutina se encarga de realizar la inicialización necesaria para el correcto funcionamiento posterior de la subrutina de control dentro del ciclo infinito. Esta subrutina es invocada una sola vez luego del reset del microcontrolador. Seguidamente se enumeran las tareas realizadas por esta subrutina:

- Configurar como entradas los pines utilizados para el monitoreo de los interruptores, habilitando las correspondientes resistencias de pull-up internas.
- Configurar como salidas los pines utilizados para controlar las luces, inicializando antes en cero el puerto como medida de protección, y habilitando las resistencias de pull-up internas en los pines restantes del puerto, también como medida de protección.
- Inicializar registros Light[1..4] en cero (todas las luces inician apagadas).

- Leer estado inicial de los interruptores y almacenarlo en el registro Switches.



**Figura 15.** Diagrama de flujo de la subrutina de inicialización del control de luces

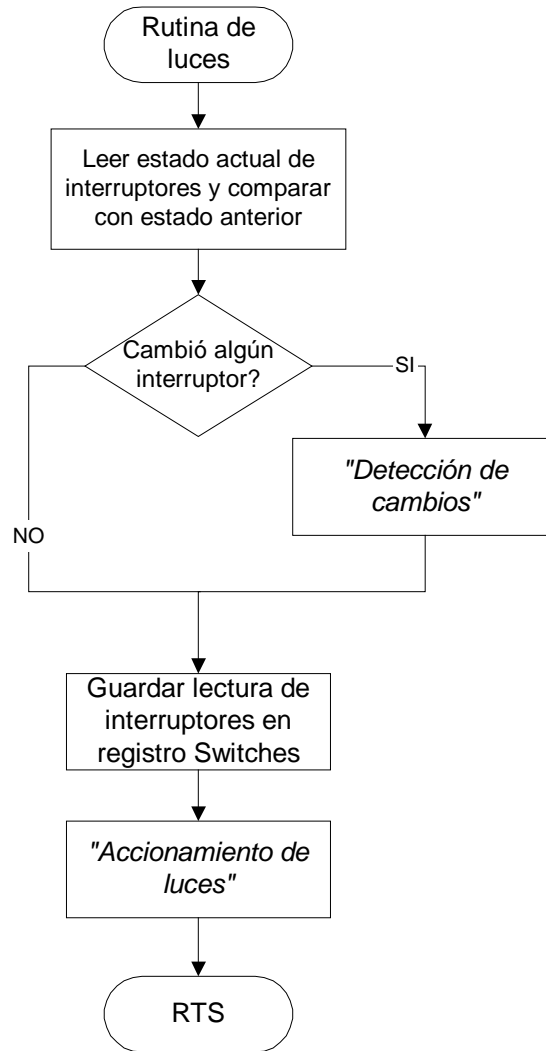
En la figura 15 es posible apreciar el diagrama de flujo de la subrutina de inicialización del sistema de luces. En el diagrama es posible apreciar las tareas realizadas por la subrutina, y los registros y puertos específicos manipulados por ésta, según la descripción de funcionamiento del sistema de luces dado al inicio de la sección.

#### **6.4.2 Subrutina de control del sistema de luces**

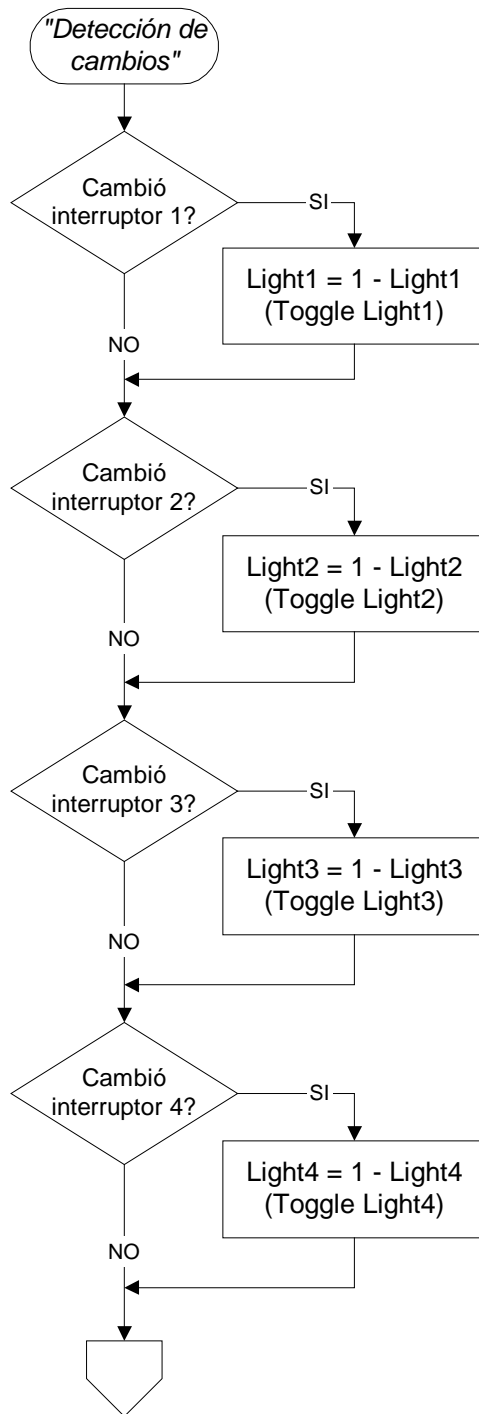
Esta subrutina se ejecuta una vez por cada entrada del software en el ciclo infinito. Las principales tareas realizadas por esta subrutina son:

- Lectura del estado actual de los interruptores.
- Comparación del estado actual de los interruptores con el estado anterior. En caso de haber cambios, realiza los cambios correspondientes en los registros Light[1..4].
- Guarda la lectura del estado actual de los interruptores, la cual será utilizada como estado anterior la siguiente vez que el software entre a la subrutina.
- Actualiza las luces con base en el estado de los registros Light[1..4], los cuales han sido actualizados anteriormente si se detectaron cambios en los interruptores.

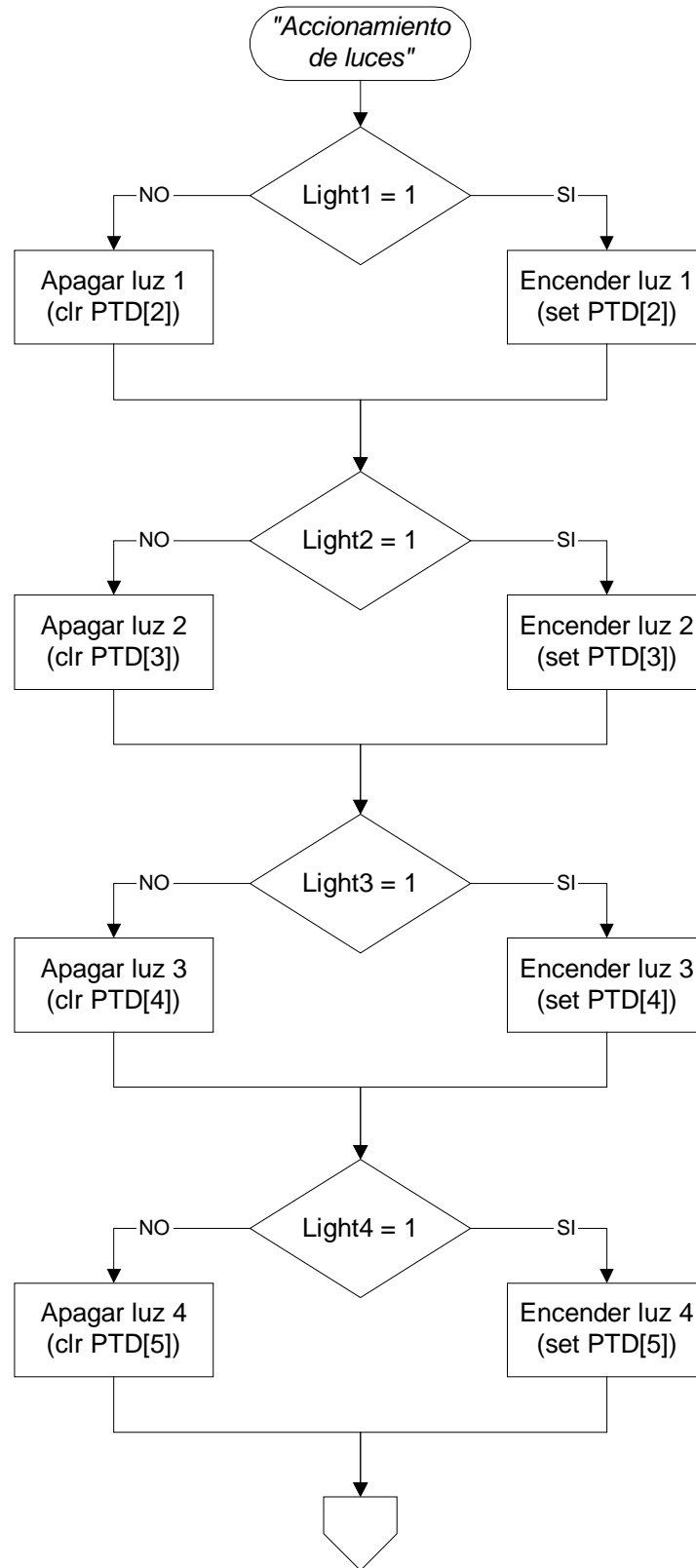
En la figura 16 se presenta el diagrama de flujo de la subrutina de control de luces. Para mayor facilidad en la lectura y comprensión del diagrama, se presenta inicialmente en la figura 16 la estructura general de este módulo de software, en donde se hace referencia a dos procesos denominados “Detección de cambios” y “Accionamiento de luces”, cuyos diagramas de flujo se presentan en las figuras 17 y 18 respectivamente.



**Figura 16.** Diagrama de flujo de la subrutina de control de luces



**Figura 17.** “Detección de cambios” en la subrutina de control de luces



**Figura 18.** “Accionamiento de luces” en la subrutina de control de luces

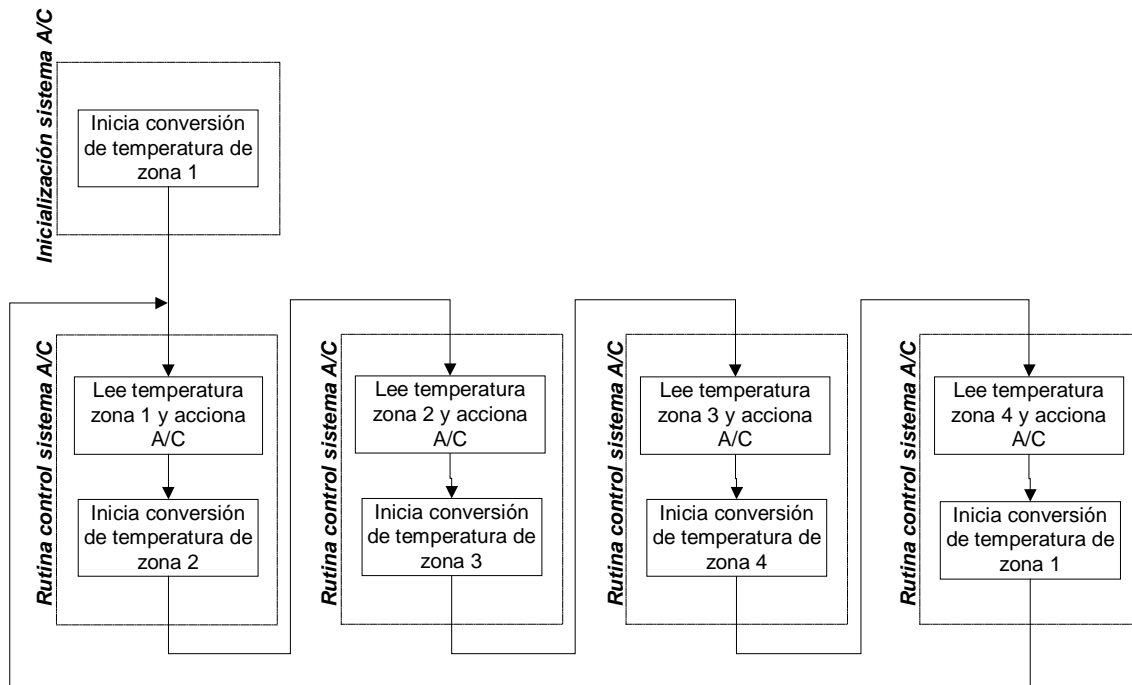


## 6.5 Software del sistema de aire acondicionado

El último de los tres módulos principales de software de la unidad de control es el correspondiente al del aire acondicionado. Al igual que los dos módulos anteriores, éste está constituido por dos subrutinas, una de inicialización que se ejecuta al inicio del programa, y la otra correspondiente al control en sí, que es ejecutada dentro del ciclo infinito del programa principal.

Las características principales del software de control y del funcionamiento del sistema de aire acondicionado son las siguientes:

- Monitorea y controla la temperatura de cuatro zonas diferentes.
- Se activa o desactiva cada zona por separado.
- Todas las zonas inician desactivadas. Al activarse una zona debe indicarse la temperatura solicitada para esa zona.
- Existen cuatro registros AirCondON[1..4] para indicar si cada zona se encuentra activada o desactivada (AirCondONx=1: zona x activada; AirCondONx=0: zona x desactivada).
- El monitoreo de temperatura de las cuatro zonas se realiza a través de los pines PTB/AD[0..3], que son cuatro de los pines de lectura del convertidor analógico-digital (ADC).
- El accionamiento del aire acondicionado se realiza a través de los pines PTB[4..7].
- El registro ContZonas se utiliza durante la ejecución para saber cuál temperatura está convirtiendo el ADC en ese momento.
- Existen cuatro registros TempFinal[1..4] que indican el valor de la temperatura deseada en cada zona.
- Debido al tiempo que toma la conversión por parte del ADC, y dentro del contexto del “multitasking cooperativo”, no es posible examinar la temperatura de las cuatro zonas cada vez que se entra a la rutina de control del sistema de aire acondicionado. Por tanto, durante la subrutina de inicialización se configura el ADC para que realice la conversión de la primera zona, y luego durante la subrutina de control, se lee del ADC la temperatura convertida de esa zona y se toma la decisión de encender o apagar el aire en esa zona. Asimismo, se inicia la conversión de la temperatura de la siguiente zona, la cual será leída la siguiente vez que se entra a la subrutina de control. En la siguiente figura se ilustra esta forma de operación del sistema de aire acondicionado (A/C).



**Figura 19.** Operación del software del sistema de aire acondicionado

En la figura 19 puede apreciarse la forma de operación del software de control del sistema de aire acondicionado. Esta forma de operación se adopta con el fin de que mientras el ADC se encuentra convirtiendo un valor de temperatura, sea posible aprovechar ese tiempo para que el microcontrolador siga adelante con el resto de procesos.

Al finalizar la conversión, el dato queda almacenado en un registro del ADC, por lo que al entrar nuevamente a la subrutina del aire acondicionado, este dato puede ser leído para tomar acciones específicas con respecto al mismo (encendido o apagado del aire acondicionado), y seguidamente iniciar la conversión de la temperatura siguiente, cuyo valor será leído del ADC la siguiente vez que se ejecute la subrutina de control del aire acondicionado.

A continuación se detallan las características de cada una de las variables utilizadas por el software de control del sistema de aire acondicionado.

**Tabla 4.** Variables del sistema de aire acondicionado

Variable	Tipo	Descripción	Posibles Valores	Ubicación
<b>AirCondON1</b>	Byte	Indica si está activado o desactivado el A/C de zona 1	Binario, 0 y 1	RAM
<b>AirCondON2</b>	Byte	Indica si está activado o desactivado el A/C de zona 2	Binario, 0 y 1	RAM
<b>AirCondON3</b>	Byte	Indica si está activado o desactivado el A/C de zona 3	Binario, 0 y 1	RAM
<b>AirCondON4</b>	Byte	Indica si está activado o desactivado el A/C de zona 4	Binario, 0 y 1	RAM
<b>TempFinal1</b>	Byte	Registra el valor deseado de temperatura para la zona 1	00h a FFh	RAM
<b>TempFinal2</b>	Byte	Registra el valor deseado de temperatura para la zona 2	00h a FFh	RAM
<b>TempFinal3</b>	Byte	Registra el valor deseado de temperatura para la zona 3	00h a FFh	RAM
<b>TempFinal4</b>	Byte	Registra el valor deseado de temperatura para la zona 4	00h a FFh	RAM
<b>TempActual1</b>	Byte	Almacena el valor actual de temperatura en la zona 1	00h a FFh	RAM
<b>TempActual2</b>	Byte	Almacena el valor actual de temperatura en la zona 2	00h a FFh	RAM
<b>TempActual3</b>	Byte	Almacena el valor actual de temperatura en la zona 3	00h a FFh	RAM
<b>TempActual4</b>	Byte	Almacena el valor actual de temperatura en la zona 4	00h a FFh	RAM
<b>ContZonas</b>	Byte	Registra el número de zona cuya temperatura está siendo convertida por el ADC.	0 a 4	RAM
<b>Temporal</b>	Byte	Durante la subrutina de control del A/C, almacena el valor leído del ADC.	00h a FFh	RAM

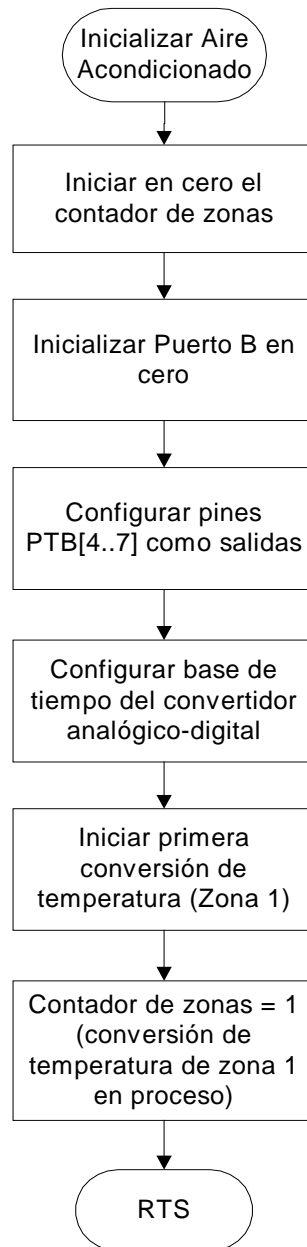
Para la lectura y el manejo de valores de temperatura dentro del software del sistema de aire acondicionado, se utilizan los datos mostrados en la tabla 1, en el capítulo 5.

### 6.5.1 Subrutina de inicialización del sistema de aire acondicionado

Entre las rutinas de inicialización se tiene por último la que se encarga del sistema de aire acondicionado, la cual tiene como función principal la configuración del convertidor analógico/digital para que funcione tal como se

especifica en esta aplicación, en donde el mismo debe convertir los valores de temperatura, obtenidos a través de un termistor, que corresponden a las temperaturas de las diferentes zonas del hogar, y con base en las cuales el controlador tomará decisiones sobre el encendido y apagado del aire acondicionado.

En la figura 20 es posible apreciar el diagrama de flujo de la subrutina de inicialización del control del aire acondicionado.



**Figura 20.** Subrutina de inicialización del control del aire acondicionado

En lo que al ADC se refiere, en la configuración del mismo se define la base de tiempo con que trabajará, así como el tipo de conversión (continua o simple), que realizará (en este sistema la conversión es simple pues solo se da cuando la subrutina lo ordena, y realiza una sola conversión a la vez). Por otro lado, con el fin de no interrumpir otras subrutinas y trabajar bajo el esquema de “multitasking cooperativo” del que se ha hablado anteriormente, para este sistema es deshabilitada la opción de generación de interrupción por conversión completa, pues la lectura de los valores se realiza por encuesta (polling).

Además de la configuración del convertidor analógico/digital, la rutina de inicialización del sistema de control del aire acondicionado se encarga de comenzar la conversión del valor actual de temperatura de la zona 1 del hogar, esto con el fin de ganar tiempo y lograr que cuando el microcontrolador entre por primera vez a la rutina de control del aire acondicionado, este valor ya pueda ser leído y se pueda, basado en el valor, tomar decisiones sobre el encendido o apagado del aire acondicionado de la zona 1.

### **6.5.2 Subrutina de control del sistema de aire acondicionado**

Por último se encuentra la subrutina de control del sistema de aire acondicionado. Tal como se ilustró anteriormente, esta subrutina lee el valor de temperatura de una zona cada vez que el programa entra a la misma, y comienza la conversión del valor de temperatura de la zona siguiente, el cual será leído la siguiente vez que el programa entre a la subrutina.

Por lo tanto, debido a que durante la rutina de inicialización del control de aire acondicionado la conversión del valor de temperatura de la zona 1 es iniciada, entonces la primera vez que el programa entre a esta subrutina, el valor de temperatura de la zona 1 podrá ser leído, y se tomarán decisiones sobre el estado del aire acondicionado para la zona 1 basadas en el valor leído, a la vez que se inicia la conversión del valor de temperatura de la zona 2, el cual será leído la segunda vez que el programa entre a esta subrutina, y así sucesivamente.

En la figura 21 se presenta el diagrama de flujo de esta subrutina de control del aire acondicionado.

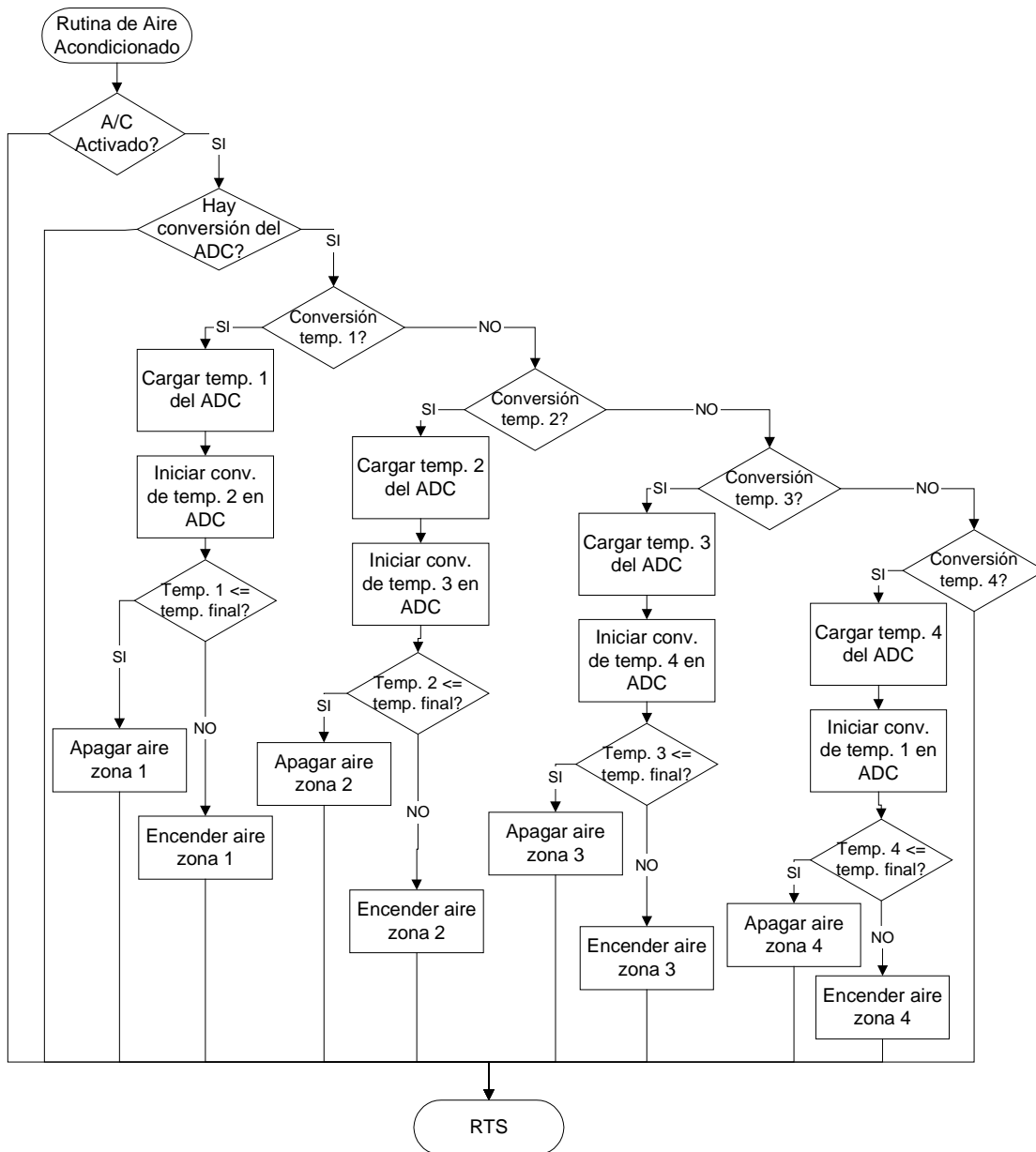


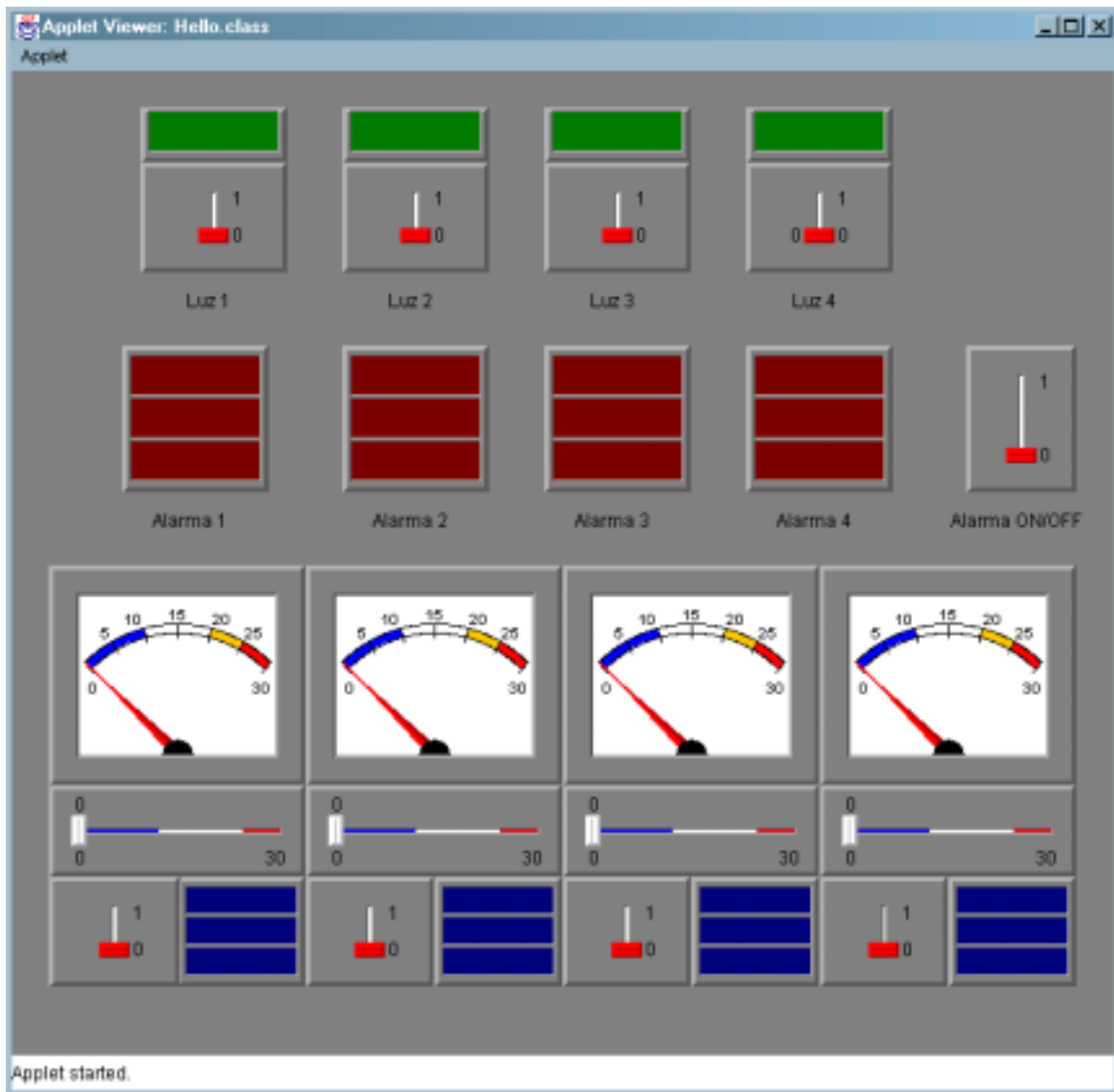
Figura 21. Subrutina de control del sistema de aire acondicionado

## 6.6 Interfaz gráfica

Para el desarrollo de la interfaz gráfica se utilizó la herramienta de software Visual Café de Symantec. Esta es una herramienta para programación visual, mediante la cual es posible desarrollar aplicaciones gráficas en una forma relativamente sencilla, basadas en Java.

Existen dos tipos de aplicaciones Java: las aplicaciones que pueden ejecutarse y funcionar por autónomamente en un PC mediante un archivo ejecutable (.exe); y las aplicaciones que se incluyen en una página web, y que se ejecutan sobre el navegador de Internet que accede a esa página. Estas últimas son llamadas “java applets”.

En el caso de la interfaz gráfica desarrollada para el control de los dispositivos del hogar, se trata de una java applet desarrollada en Visual Café. En la figura 22 se presenta esta interfaz gráfica.



**Figura 22.** Diagrama de la interfaz gráfica desarrollada

En la figura 22 se muestra la interfaz gráfica provista a través de una java applet desarrollada en Visual Café. Para el caso mostrado en la figura, el sistema se encuentra apagado pues todas las variables se encuentran en cero.

Los objetos utilizados en Visual Café para conformar esta interfaz, tales como los interruptores y los medidores analógicos, son provistos por la herramienta EMIT y fueron agregados al Visual Café. Estos objetos permiten el desarrollo de interfases, muy intuitivas, para el monitoreo y control de dispositivos electrónicos.



## CAPÍTULO 7

# ANÁLISIS Y RESULTADOS

---

### 7.1 Explicación del diseño

Como se ha podido observar a lo largo del documento, el diseño realizado para la consecución de los objetivos planteados se divide en dos partes principales: software y hardware. Cada una de estas partes representa desarrollos distintos, con un cierto grado de independencia que ha permitido el desarrollo de ambos en forma paralela.

Tal como se explicó en el capítulo 5, el hardware del sistema está constituido en forma central por el microcontrolador de Motorola MC68HC908GP32, alrededor del cual se han diseñado los módulos de hardware necesarios para su correcto funcionamiento, además de los módulos específicos de cada una de las funciones controladas: alarma, luces y aire acondicionado.

Entre los módulos de hardware indispensables para el funcionamiento del microcontrolador se encuentran como principales las conexiones de alimentación, las cuales deben estar debidamente protegidas contra ruido mediante el uso de los capacitores recomendados por el fabricante y descritos en el capítulo 5; y el módulo del oscilador, el cual provee una señal de baja frecuencia al microcontrolador, que es utilizada por el generador de reloj interno para la generación de la señal de reloj, indispensable para el funcionamiento del microcontrolador.

Con respecto a los módulos de hardware correspondientes a cada una de las funciones del sistema, estos están diseñados de forma tal que sea posible, en conjunto con el software, satisfacer las especificaciones dadas para el funcionamiento del sistema.

Otro módulo de hardware desarrollado, y que cuenta con una gran importancia dentro del sistema, es el correspondiente a la interfase de comunicaciones RS232/TTL, la cual cuenta, como componente principal, con el circuito integrado MAX232 de Maxim. Este módulo brinda la interfase de hardware indispensable para la correcta comunicación del sistema con una PC, desde la cual es posible, entre otras funciones, realizar el monitoreo y control de las variables del sistema.

En forma análoga al hardware, el software del sistema está constituido por ciertos módulos de configuración indispensables para el correcto

funcionamiento del microcontrolador, tales como la configuración del generador de reloj, de los puertos de entrada y salida, el ADC, puerto serial, entre otros.

Adicional a estas rutinas de configuración se encuentran las rutinas específicas para el control de cada una de las funciones del sistema de acuerdo a las especificaciones del sistema, y cuyos detalles de diseño fueron abordados en el capítulo 6.

## **7.2 Alcances y limitaciones**

Tomando como referencia los objetivos general y específicos planteados para este proyecto en el capítulo 3 del presente documento, es posible ahora realizar un análisis de los alcances, limitaciones y puntos de posible mejoramiento del proyecto desarrollado.

Según lo enumerado en el capítulo 3, y las funciones que el sistema desarrollado es capaz de realizar, es posible concluir que el sistema cumple en forma rigurosa con los objetivos planteados para el desarrollo de este proyecto. Por lo tanto, los alcances de este proyecto son los mismos vislumbrados al momento de establecer los objetivos para el mismo.

Mediante el sistema desarrollado es posible controlar las luces del hogar en una forma que no impide el uso de los interruptores de luces manuales instalados en todo hogar. El sistema detecta violaciones o disparos de alarmas en las zonas conectadas como entradas del mismo. Además, es capaz de controlar el sistema de aire acondicionado, apagándolo o encendiéndolo de acuerdo a la temperatura ambiente sensada y a la temperatura deseada.

Adicionalmente, es posible realizar todas estas funciones desde una interfaz grafica que se ejecuta mediante un java applet que podría, en forma sencilla, ser cargada en un servidor web que permita el acceso al sistema desarrollado desde puntos lejanos al mismo, a través de la Internet.

Sin embargo, a pesar de las funcionalidades alcanzadas con el sistema desarrollado, éste es aún un sistema prototipo que para su funcionamiento en la vida real necesita ciertos elementos importantes. El más destacable de estos elementos es el hecho de que la interfaz gráfica debe cargarse en un servidor web para que pueda ser accesada remotamente, y puedan realmente controlarse, a través de Internet, los dispositivos del hogar.

Otro elemento importante que debería desarrollarse para el funcionamiento del sistema en la vida real, son las etapas de potencia necesarias para el accionamiento de los bombillos, los sistemas de aire acondicionado, y el sensado de interruptores y sensores de alarma de un hogar real, los cuales en este proyecto, para fines demostrativos, se han simulado a través de LED's e interruptores tipo DIP, pero que en la vida real debería

contarse con optoacopladores y dispositivos electrónicos de potencia tales como TRIAC's, u otros accionadores tales como relés, que permitan la entrada de las señales al sistema y el accionamiento de los dispositivos en forma segura.

Con respecto al sistema de aire acondicionado, su módulo de entrada ha sido diseñado con los sensores de temperatura conectados debidamente, y con el software necesario para la lectura de las temperaturas sensadas y su correspondiente manipulación para que el sistema cumpla con la funcionalidad requerida. Sin embargo, dentro del planteamiento de los objetivos para este módulo del sistema, no se ha tomado en cuenta su interacción con posibles sistemas de aire acondicionado preexistentes, es decir, el sistema desarrollado controla por sí solo el aire acondicionado de diferentes zonas, pero no contempla un módulo mediante el cual sea posible controlar el aire acondicionado en forma manual dentro del hogar. Por lo tanto, como posible mejora al sistema desarrollado, podría desarrollarse una interfaz de usuario para el hogar, mediante la cual sea posible observar los valores de temperatura actual, modificar las temperaturas deseadas, y activar o desactivar el sistema de aire acondicionado, todo en forma manual, y que todos estos cambios puedan observarse mediante la interfaz gráfica.

## **CAPÍTULO 8**

### **CONCLUSIONES Y RECOMENDACIONES**

---

#### **8.1 Conclusiones**

1. Mediante el sistema desarrollado, es posible monitorear y controlar, desde una interfaz gráfica basada en un Java Applet, las luces, aire acondicionado y sistema de seguridad de una casa.
2. Con el control desarrollado para el sistema de luces, es posible conocer y modificar, desde la interfaz gráfica, el estado de las luces, e interactuando con el proceso normal de encendido y apagado manual a través de interruptores.
3. Mediante el control del sistema de seguridad, es posible observar, a través de la interfaz gráfica, violaciones en las zonas de seguridad preestablecidas. Además es posible activar o desactivar el sistema de seguridad desde la misma interfaz.
4. Con el sistema de control de aire acondicionado, es posible observar en la interfaz gráfica la temperatura sensada en cada una de las zonas preestablecidas, y modificar la temperatura deseada para cada zona con el fin de que el sistema de control accione los mecanismos de aire acondicionado de cada zona. Además es posible activar o desactivar el sistema en cada zona, pudiéndose ver en todo momento la temperatura sensada, sin importar si el sistema se encuentra activado o no.

## 8.2 Recomendaciones

1. Para un funcionamiento del sistema bajo condiciones reales, existen otros elementos que deben agregarse o desarrollarse con el fin de cumplir con ciertos requerimientos propios de la vida real. Estos elementos, los cuales pueden ser parte de un futuro proyecto a realizar, son los siguientes:
  - a. Configuración de un servidor web en el cual se pueda cargar la java applet desarrollada, para que esta pueda ser accesada remotamente a través de Internet. Para esto, debe contarse con ciertos elementos de hardware y software que permitan llevar esto a cabo, así como con una dirección IP fija para el servidor web, por lo tanto, idealmente el acceso del servidor a la Internet debe ser a través de una conexión dedicada con un proveedor de servicios de Internet (ISP), o a través de otras tecnologías en las que la conexión permanente con el ISP no represente un incremento importante en el costo de estos servicios, tales como Internet por cable, o conexiones conmutadas sin restricción en el tiempo de acceso, utilizando cualquiera de la gama de tecnologías existentes dependiendo de la disponibilidad local y de la velocidad de transmisión requerida: telefonía convencional, ISDN, ADSL, etc. Debe tomarse en cuenta que para las conexiones conmutadas y de Internet por cable, no se cuenta con una dirección IP fija, lo cual puede ser una limitante para el acceso remoto.
  - b. Desarrollo de las etapas de potencia necesarias para la interacción entre el sistema digital desarrollado, y los demás elementos involucrados en el sistema, tales como bombillos, interruptores de luz, sensores de alarma, sistemas de aire acondicionado, etc. Existen diversas opciones en cuanto a los elementos que pueden utilizarse para este fin, pero en forma general siempre deberá contarse con dispositivos que provean un aislamiento eléctrico al microcontrolador (optoacopladores), y dispositivos que a partir de las señales de baja potencia que salen del microcontrolador, sean capaces de manejar los elementos reales a niveles de potencia mayores (TRIAC's, relés).
2. El microcontrolador utilizado para el desarrollo de este proyecto brinda otras opciones de funcionamiento y programación que no han sido explotadas en este proyecto. Una de estas opciones es que gracias a la tecnología de memoria Flash con que cuenta el microcontrolador, es posible realizar lectura y escritura de la memoria de programa sin

necesidad de sacar el microcontrolador de la tarjeta electrónica en la que se encuentre, lo cual provee una ventaja muy importante para la realización de actualizaciones y mejoras al software. Existen dos formas de realizar esto, denominadas programación en circuito y programación en aplicación. En el Apéndice 3 del presente documento se presenta una explicación más detallada acerca del funcionamiento de ambos modos de programación del microcontrolador.

3. Con el fin de hacer al módulo de aire acondicionado más funcional y práctico para su uso en el hogar, es recomendable incluir una interfase de usuario local que permita conocer las temperaturas actuales y fijar las temperaturas deseadas, así como activar o desactivar el sistema. Esto podría lograrse mediante el desarrollo de una interfase controlada por el microcontrolador, o mediante la interacción del mismo con un sistema de aire acondicionado preexistente, en donde las entradas y salidas del mismo puedan funcionar como salidas y entradas del microcontrolador, en un funcionamiento similar al desarrollado para el sistema de luces.

## BIBLIOGRAFÍA

---

Acosta, Maurizio. Programming Microcontrollers in C. Guadalajara, México: Motorola Inc., 2000.

Cosmic Software Inc. C Cross Compiler User's Guide for Motorola MC68HC08. San Jose, California, 1998.

EmWare Inc. EMIT Developer's Manual. Salt Lake City, Utah, 1999.

EmWare Inc. EMIT User Interface Guide. Salt Lake City, Utah, 1999.

EmWare Inc. EMIT User's Manual. Salt Lake City, Utah, 1999.

EmWare Inc. Net.08 Developer's Guide. Salt Lake City, Utah, 1999.

EmWare Inc. Net.08 Reference Board Manual. Salt Lake City, Utah, 1999.

Lun, T.C. In-Circuit Programming of FLASH Memory in the MC68HC908GP32. Hong Kong: Motorola Inc., 2000.

Maxim Integrated Products. +5V-Powered, Multichannel RS-232 Drivers/Receivers. Sunnyvale, California, 2000.

Motorola Inc. CPU08 Central Processor Unit Reference Manual. Phoenix, Arizona, 1996.

Motorola Inc. M68ICS08GP In-Circuit Simulator Operator's Manual. Phoenix, Arizona, 1999.

Motorola Inc. MC68HC908GP32 HCMOS Microcontroller Unit Technical Data. Phoenix, Arizona, 2000.

Motorola Inc. Understanding Small Microcontrollers. Phoenix, Arizona, 1998.

Ng, Y.T. Power-On, Clock Selection, and Noise Reduction Techniques for the Motorola MC68HC908GP32. Hong Kong: Motorola Inc., 2001.

Robb, Stuart. Creating Efficient C Code for the MC68HC08. East Kilbride, Scotland: Motorola Inc., 2000.

Siemens Matsushita Components GmbH & Co. NTC Thermistors.

Trenado, Rene. Connecting the MC68HC908GP32 to an Internet Service Provider (ISP) using the Point-to-Point Protocol (PPP). Tijuana, Baja California, México: Motorola Inc., 2001.

Van Sickle, Ted. Programming Microcontrollers in C. Eagle Rock, Virginia: LLH Technology Publishing, 1994. (Motorola Series in Solid State Electronics).



## APENDICES

---

### Apéndice 1: Tabla de referencia rápida de información sobre el proyecto

#### Información del estudiante:

**Nombre:** Juan Diego Calvo Chaves  
**Cédula:** 1-1009-620                      **Carné ITCR:** 9607813  
**Dirección en época lectiva:** San Rafael de Escazú, de Banco San José 500 Norte y 150 Oeste.  
**Dirección en época no lectiva:** Misma a la anterior.  
**Teléfono en época lectiva:** 228-4494  
**Teléfono época no lectiva:** 228-4494  
**Email:** [jdcalvo@hotmail.com](mailto:jdcalvo@hotmail.com)

#### Información del proyecto:

**Nombre del proyecto:** Monitoreo y control de dispositivos residenciales vía Internet.  
**Área del Proyecto:** Sistemas Digitales y Redes.

#### Información de la empresa:

**Nombre:** Motorola de Costa Rica  
**Zona:** San José, Cantón Central, Pavas, Rohrmoser.  
**Dirección:** Oficentro Plaza Mayor, Etapa 2, Piso 3, Boulevard Rohrmoser.  
**Teléfono:** 296-5385      **Fax:** 296-5387      **Apartado:** 672-1007 Centro Colón  
**Actividad Principal:** Representación de Motorola Inc., ventas, soporte técnico.

#### Información del encargado en la empresa:

**Nombre:** Alex Saborío  
**Puesto que ocupa:** Administrador de Telecomunicaciones  
**Departamento:** Infraestructura y Telecomunicaciones  
**Profesión:** Ingeniero en Electrónica      **Grado Académico:** Bachiller  
**Teléfono:** 296-5385      **Ext.:** 115  
**Email:** [AAS033@email.mot.com](mailto:AAS033@email.mot.com)

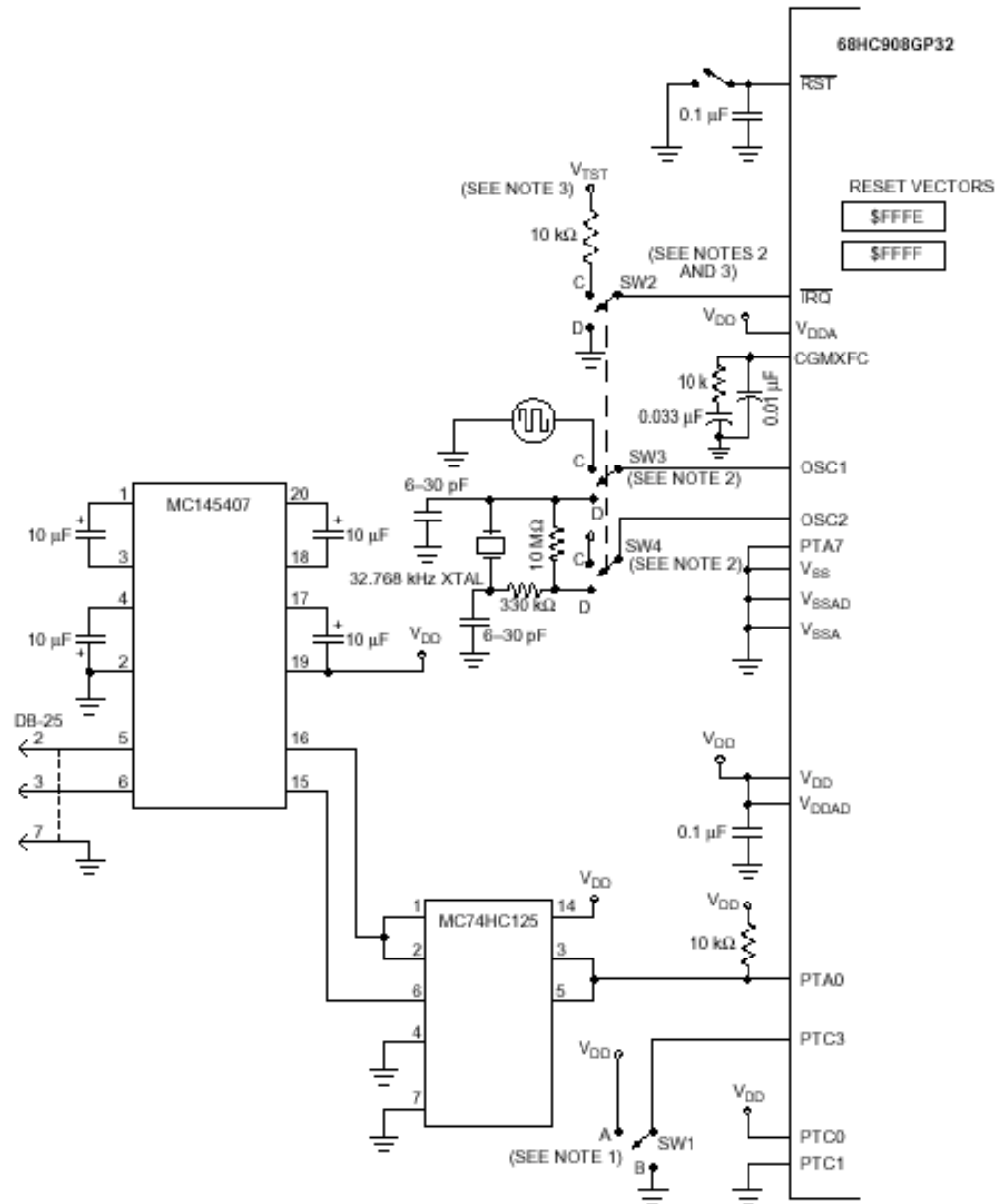
## **Apéndice 2: Descripción de términos y de herramientas utilizadas**

1. *Herramienta EMIT*: Herramienta de software provista por emWare dentro del sistema de desarrollo NET.08. Esta herramienta contiene rutinas de software para la configuración de las tablas de datos dentro del microcontrolador, en las cuales se basa el acceso a las variables del sistema desde la interfaz gráfica. Asimismo, esta herramienta contiene, con fines didácticos, ejemplos de programación de diferentes aplicaciones, y versiones de evaluación del compilador de lenguaje C Cosmic para microcontroladores Motorola HC08, y del Visual Café de Symantec para el desarrollo de java applets.
2. *Interfase emGateway*: Software provisto por emWare como uno de los componentes de la herramienta EMIT, el cual se instala en un PC y funciona como puerta de enlace hacia el sistema microcontrolado. Instalado en un servidor web, el emGateway funciona como enlace entre la Internet y el sistema microcontrolado.
3. *Java Runtime Interface*: Interfase para monitoreo y acceso a variables en tiempo real, desarrollada a través del java applet diseñado para la interfaz gráfica.
4. *Sistema de desarrollo NET.08*: Sistema de desarrollo provisto por emWare para el desarrollo de soluciones de conectividad de dispositivos electrónicos a Internet. Entre otras cosas, este sistema de desarrollo contiene la herramienta EMIT, junto con una tarjeta de referencia provista para la prueba y estudio de las aplicaciones de ejemplo provistas en la herramienta EMIT.

### **Apéndice 3: Programación del microcontrolador en circuito y en aplicación.**

Existen dos formas de realizar la programación del microcontrolador sin necesidad de ser removido de la tarjeta electrónica: programación en circuito, y programación en aplicación. La programación en circuito es en la que con el uso de hardware adicional, y con el microcontrolador funcionando en un modo especial llamado modo monitor, es posible reprogramar la memoria de programa con datos transmitidos al microcontrolador a través del puerto serial. En la figura 23 se muestra el diagrama de circuito recomendado por el fabricante, tomado del manual del microcontrolador, para el funcionamiento del mismo en modo monitor, en el cual se puede realizar, entre otras cosas, la programación del microcontrolador en circuito.

Además de la programación en circuito, es posible realizar la denominada programación en aplicación, en la cual mediante una rutina del software que se ejecuta en el microcontrolador, es posible manipular los datos de la memoria Flash con base en datos generados por el mismo programa, o recibidos por el puerto serial o por otros medios. Este modo de programación es especialmente útil para la manipulación de tablas de datos estáticos que deben renovarse cada cierto tiempo. No es posible leer o escribir en la memoria Flash con programa ejecutándose desde la misma memoria Flash, por lo que para realizar esto, es necesario cargar en la memoria RAM la rutina que realiza este procedimiento, trasladar la ejecución del programa a la memoria RAM mientras se ejecuta esta rutina, y luego retornar la ejecución a la memoria Flash una vez que sus datos han sido manipulados.



Notes:

1. For monitor mode entry when  $\overline{IRQ} = V_{TST}$ :  
 SW1: Position A — Bus clock = CGMXCLK + 4 or CGMVCLK + 4  
 SW1: Position B — Bus clock = CGMXCLK + 2
2. SW2, SW3, and SW4: Position C — Enter monitor mode using external oscillator.  
 SW2, SW3, and SW4: Position D — Enter monitor mode using external XTAL and internal PLL.

**Figura 23.** Diagrama de circuito para funcionamiento del microcontrolador en modo monitor