



**UNIVERSIDAD AUTÓNOMA DEL
ESTADO DE HIDALGO**

**ESCUELA SUPERIOR DE
TIZAYUCA**

**“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA ELECTRÓNICO DE
REGISTRO DE ACCESO PARA EL ÁREA DE ALMACÉN EN LA
EMPRESA MAQUINADOS E INYECCIONES AUTOMOTRICES”**

TESIS

**QUE PARA OBTENER EL TÍTULO DE INGENIERO EN ELECTRÓNICA
Y TELECOMUNICACIONES**

PRESENTA

MIRIAM ESTEVEZ CABRERA

ASESOR: ING. HUGO RUIZ GONZALEZ

TIZAYUCA, HIDALGO. NOVIEMBRE DE 2008

AGRADECIMIENTOS

La elaboración de esta tesis es parte fundamental en una etapa muy importante de mi vida y el comienzo de otras más.

Si bien ha requerido de esfuerzo, fe y mucha dedicación de mi parte, quiero expresar mi agradecimiento a Dios que me bendice permitiéndome mirar el sol de frente.

En el trayecto universitario que el tiempo obliga y en la culminación de este proyecto agradezco la valiosa aportación de las siguientes personas, además de quienes merecen ser mencionadas por haber plasmado su huella en mi camino.

A la empresa “Maquinados e Inyecciones Automotrices S.A. de C.V.” por haberme dado la oportunidad de implementar este sistema.

A mis padres Antonia y José Genaro, quienes inculcaron en mi el rigor que guía mi transitar por la vida, gracias por haber hecho posible mi estancia en la universidad y por su apoyo en la realización de esta tesis.

A mis hermanitos: Isaac, Abel, Ismael y Pepe por su compañía y confianza.

A Refugio, Marcos, Teresa, Julio, Mario, Elena, Luis y Alfredo por su amistad y ese apoyo incondicional que tan importante fue para mí en los días de realización de este proyecto.

A Héctor Castro A. por su predisposición permanente e incondicional en aclarar mis dudas, por su comprensión y asesoría.

Al Ing. Hugo Ruiz G. por su asesoría y dirección en el trabajo.

Al Ing. Asdrubal L. Chau por sus substanciales sugerencias y su valiosa colaboración en el desarrollo del *firmware*.

Gracias a todas aquellas personas que de una u otra manera, participaron en la realización de esta tesis, hago extensivo mi más sincero agradecimiento.

ÍNDICE

CAPÍTULO I

Objetivo general.....	1
Objetivos específicos.....	2
Planteamiento del problema.....	3
Justificación.....	4
Organización del trabajo.....	5

CAPÍTULO II: Antecedentes

2.1.- Orígenes de los sistemas electrónicos en edificios.....	6
2.2.- Características del área industrial con acceso restringido.....	9
2.3.- Propuesta de Diseño del Sistema de Control de Acceso.....	10

CAPÍTULO III: Descripción y Selección de los Módulos del Sistema de Control de Acceso

3.1.- Módulo de detección de presencia (barrera infrarroja).....	12
3.1.1.- Principio básico de funcionamiento de una barrera infrarroja.....	12
3.1.2.- Características de la barrera infrarroja.....	14
3.2.- Módulo de identificación y control.....	15
3.2.1.- Características de la base de datos.....	16
3.3.- Módulo de alarma.....	18
3.4.- Módulo de Interfaz.....	19
3.4.1.- Funcionamiento básico de un microcontrolador.....	20
3.4.2.- Características generales del microcontrolador ATmega8535.....	22
3.4.3.- Características específicas del microcontrolador para la comunicación serial.....	23
3.4.4.- Transmisión de datos.....	26

CAPÍTULO IV: Implementación del sistema

4.1.- Visión global de funcionamiento de la interconexión de los módulos.....	30
4.2.- Descripción del hardware.....	33
4.2.1.- Desarrollo del módulo de detección de presencia (barreras infrarrojas).....	33
4.2.1.1.- Modos de operación.....	33
4.2.2.- Desarrollo del módulo de alarma.....	35
4.2.3.- Terminal de identificación.....	36
4.2.4.- Diagramas de conexión.....	37
4.2.5.- Placa de circuito impreso (PCB) de interfaz.....	39
4.3.- Descripción del software.....	43
4.3.1.- Software de la PC.....	43
4.3.1.1.- Declaración de funciones.....	43
4.3.1.2.- Base de datos.....	45
4.3.1.3.- Comunicación serial.....	49
4.3.2.- Firmware del microcontrolador.....	54

CAPITULO V: Conclusiones

5.1.- Pruebas y resultados.....	63
5.2.- Estudio económico.....	68
5.3.- Futuros desarrollos.....	69
5.4.- Conclusiones.....	70
Apéndice A Características específicas del microcontrolador ATmega 8535 y MAX232....	71
Apéndice B Fuente de alimentación.....	73
Apéndice C Herramienta de programación AVRISP mkII.....	74
Apéndice D Código fuente del firmware.....	75
Bibliografía y fuentes consultadas	79

ÍNDICE DE TABLAS

Tabla 4.1 Entradas puerto A.....	56
Tabla 4.2 Salidas puerto C	56
Tabla 5.1 Costos.....	68

ÍNDICE DE FIGURAS

Figura 1.1 Módulos de funcionamiento del sistema propuesto.....	3
Figura 2.1 Dimensiones del área restringida.....	9
Figura 2.2 Elementos de los módulos del sistema.....	10
Figura 3.1 Barrera infrarroja propuesta.....	14
Figura 3.2 Encapsulado metálico.....	15
Figura 3.3 Relación de tablas y campos de la BDD.....	17
Figura 3.4 Módulo de alarma	19
Figura 3.5 Microcontrolador.....	20
Figura 3.6 Estructura básica de un microcontrolador.....	21
Figura 3.7 Microcontrolador AVR ATmega8535.....	23
Figura 3.8 Registro de datos.....	24
Figura 3.9 Registro A de control y estado USART.....	24
Figura 3.10 Registro B de control y estado USART.....	25
Figura 3.11 Registro C de control y estado USART.....	25
Figura 3.12 Registro de selección de baud rate.....	25
Figura 3.13 Conjunto de códigos de ASCII.....	26
Figura 3.14 Transmisión de datos en modo serie	28
Figura 4.1 Diagrama de flujo del funcionamiento del sistema.....	30
Figura 4.2 Colocación de las Barreras Infrarrojas.....	33
Figura 4.3 Modo 1 entrada.	34
Figura 4.4 Modo 2 salida.....	35
Figura 4.5 Colocación de la sirena de alarma.....	36
Figura 4.6 Colocación de la terminal de identificación.....	36
Figura 4.7 Diagrama de conexión de las barreras.....	37
Figura 4.8 Diagrama de conexión de la alarma.....	38

Figura 4.9	Diagrama de conexión MAX232.....	39
Figura 4.10	Placa PCB cara superior (<i>top layer</i>).....	40
Figura 4.11	Placa PCB cara inferior (<i>bottom layer</i>).....	41
Figura 4.12	Placa PCB en tercera dimensión.....	41
Figura 4.13	Placa PCB con componentes físicos.....	42
Figura 4.14	Comando timer.....	43
Figura 4.15	Comando multimedia.....	44
Figura 4.16	Tabla movimiento_entrada.....	47
Figura 4.17	Registro entrada.....	47
Figura 4.18	Tabla movimiento_salida.....	48
Figura 4.19	Registro salida.....	49
Figura 4.20	Comando MScomm.....	49
Figura 4.21	Comando DAO.....	53
Figura 4.22	InputBox.....	53
Figura 4.23	Active Report.....	54
Figura 5.1	Placa PCB final (<i>bottom layer</i>).....	63
Figura 5.2	Placa PCB final (<i>top layer</i>).....	64
Figura 5.3	Interconexión física de los módulos.....	64
Figura 5.4	Plano del área de registro.....	67
Figura.	Fuente de alimentación.....	73

RESUMEN

En el presente trabajo se expone el desarrollo y la elaboración de un sistema electrónico de registro aplicado a un área de guarda de materiales y productos con acceso restringido, ubicada en una nave industrial de la empresa “Maquinados e Inyecciones Automotrices S.A. de C.V.”, en Tizayuca, Hgo., para proporcionar seguridad en el área.

En los siguientes capítulos se describen las características y requerimientos del área restringida, los fundamentos teóricos para comprender las funciones que realiza este sistema, cuya característica primordial es la interconexión de un microcontrolador con circuitos externos como alarmas, sensores, entre otros, y al mismo tiempo realizar una comunicación serial con la PC con la finalidad de proporcionar una base de datos con los movimientos de entradas y salidas que se llevan a cabo en el área restringida. Se proporciona una descripción detallada del funcionamiento de los componentes y dispositivos utilizados en la elaboración de este sistema que combina tanto *software* y *hardware*, así como la implementación del sistema electrónico que controla el acceso del personal de esta empresa, por medio de módulos de detección de presencia (sensores infrarrojos), una terminal de identificación contenida en la PC que además proporcionará la base de datos con los registros del personal autorizado y finalmente un modulo de alarma para alertar al encargado del área que se presentó un acceso no autorizado; todos estos periféricos conectados a un microcontrolador el cual actúa como una interfaz hacia la PC cumpliendo así con el objetivo general que tiene como premisa brindar una mayor seguridad al área en cuestión.

ABSTRACT

In the present project is carried out the development, elaboration and application of an electronic system of access for an industrial area warehouse of materials and products with restricted access, located in an industrial ship of Maquinados e Inyecciones Automotrices S.A. de C.V. Company in Tizayuca, Hgo., in order to satisfy the necessity with a greater security in the area indicated above.

In the following chapters the characteristics and requirements of the restricted area are expressed, and the theoretical foundations are included to understand the functions that this project will do, whose fundamental characteristic of this system is the interconnection of a microcontroller with peripherals such as alarms, sensors, among others, and at the same time to establish a serial communication with the PC in order to provide a database with the movements of entrances and exits that are carried out in the restricted area. One provides a detailed description of the operation of the components and devices used in the elaboration of this system that combines software and hardware, as well as the implementation of the electronic system that will be able to control the access of the personnel of this company, by means of using presence detection modules (sensorial infrared), a identification terminal contained in the PC that will provide the database with the registries of the authorized personnel, and finally, an alarm unit which alerts the one who is in charge that appeared in the area a nonauthorized access, all these peripherals connected through a microcontroller which will act like an interface towards the PC, it will fulfill the general mission that it has the objective of offering a greater security to the specified area.

CAPITULO I

OBJETIVO GENERAL

Diseñar e implementar un sistema electrónico que permita el control de acceso del personal en un área restringida de guarda de materiales y productos, ubicada dentro de la nave industrial “Maquinados e Inyecciones Automotrices S.A. de C.V.” en Tizayuca, Hgo., mediante módulos de detección de personal, identificación, interfaz y alarma.

OBJETIVOS ESPECÍFICOS

- Establecer los módulos y dispositivos electrónicos accesibles para el funcionamiento del sistema electrónico de seguridad.
- Hacer uso de sensores infrarrojos para la detección del personal que a su vez deberá registrarse en una terminal de identificación, de no ser así el sistema posee un módulo de alarma para de este modo informar un acceso no autorizado.
- Diseñar el programa que actuará como terminal de identificación, implementado en una PC que para los fines de la empresa es obsoleta, este software será capaz de soportar comunicación vía puerto serie y deberá mantener un registro permanente de las entradas y salidas del área restringida mediante una base de datos.
- Mediante un microcontrolador realizar la interconexión de los módulos, y al mismo tiempo transferir en modo serial esta información a la PC, y con base en esta información realizar las funciones correspondientes a la detección, identificación del personal y activación o desactivación de alarmas.
- Con base en los requerimientos del sistema, diseñar la placa de circuito impreso (PCB) que funcionará como interfaz entre los módulos de entrada-salida y la PC.
- Realizar pruebas de campo para comprobar el desempeño del sistema y determinar el tiempo óptimo que los usuarios tendrán para registrarse en la terminal de identificación.

PLANTEAMIENTO DEL PROBLEMA

La demanda de seguridad que existe principalmente en el sector industrial para controlar áreas con acceso restringido ha ido en aumento en los últimos años debido a fenómenos económicos y sociales que enfrenta nuestro país, tal es el caso de la empresa “Maquinados e Inyecciones Automotrices S.A. de C.V.”, la cual requiere un sistema electrónico que permita el control de registro de acceso a un área restringida de guarda de materiales y productos, ubicada dentro de una nave industrial. Por ello, se propone implementar el siguiente sistema electrónico.

En la Figura 1.1 se muestra el diagrama del sistema propuesto para llevar a cabo el control de registro de acceso.

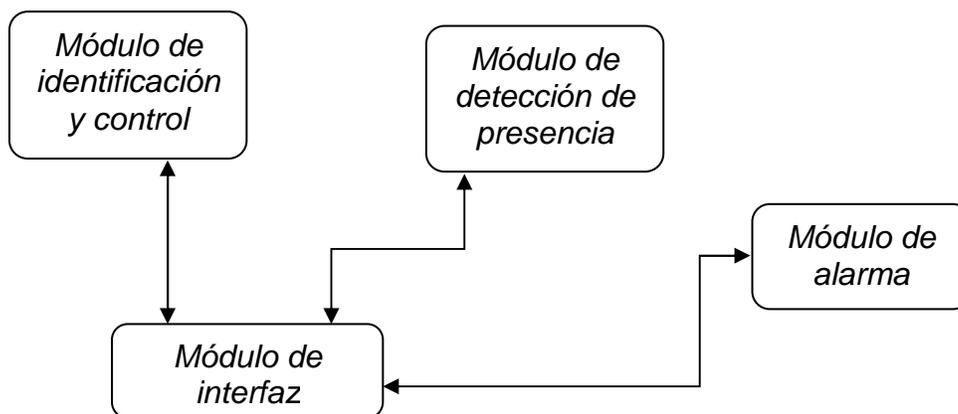


Figura 1.1 Módulos de funcionamiento del sistema propuesto

El funcionamiento del sistema se basa en la implementación e instalación de cuatro módulos, los cuales contienen dispositivos electrónicos que permiten el control de registro de acceso del personal que labora en esas Instalaciones, de tal forma que cuando el personal ingrese al área será detectado por unos sensores infrarrojos y estará obligado a registrarse por medio de una terminal de Identificación instalada en una PC, la cual contendrá una base de datos con los registros de las personas que estarán autorizadas para ingresar a dicha área. El módulo de interfaz estará interconectado a la terminal de identificación, detección de presencia y alarma, como lo muestra la Figura 1.1. Cuando ocurra un acceso no autorizado, la interfaz indicará al módulo de alarma que active la sirena electrónica para informar al encargado del área almacén.

JUSTIFICACIÓN

Las demandas de seguridad en términos de la administración de una empresa están evolucionando constantemente, específicamente cuando se habla de la seguridad del personal, visitantes, contratistas y por supuesto de los bienes e información confidencial. Por lo que las aportaciones de la electrónica han hecho posible el desarrollo de una organización, automatización y control en sistemas de seguridad aplicados en áreas industriales con acceso restringido. Debido a la centralización de los datos de la empresa es posible supervisar y controlar confortablemente desde una PC, los estados de funcionamiento y alarmas de los sistemas que componen las instalaciones.

Los sistemas electrónicos combinados con terminales de identificación se convierten en una ayuda inestimable para la administración responsable de la seguridad de la empresa, por lo tanto la electrónica nos brinda una herramienta muy eficaz en esta área, por lo que hacer uso de esta, es un buen aliciente para llevar a cabo el desarrollo de este sistema de seguridad.

ORGANIZACIÓN DEL TRABAJO

La forma en la que se lleva a cabo el desarrollo del sistema se plantea de la siguiente manera y consta de varias partes; la primera parte comprende la búsqueda de información, con la que se recopila datos útiles para la investigación y proporciona los fundamentos para comprender el funcionamiento de los dispositivos que integrarán los módulos del sistema; con base en el marco teórico se seleccionan los dispositivos electrónicos que forman parte de los módulos del sistema; como segunda parte de este trabajo se tiene el desarrollo del *software* que permite el control de acceso al área restringida, mediante una base de datos de usuarios autorizados y que también mantiene un registro permanente de entradas y salidas; una vez establecido el *software* y la forma en la que interactúa con los módulos, es necesario realizar una plataforma que soporte la interfaz así como la conexión de los diversos módulos a lo que llamaremos *hardware*; y finalmente se comprueba el correcto funcionamiento del sistema por medio de pruebas físicas en el área para obtener las conclusiones de este trabajo.

CAPITULO II: ANTECEDENTES

2.1.- Orígenes de los sistemas electrónicos en edificios

Históricamente el hombre ha construido edificios para crear un entorno controlado para poder vivir y para poder trabajar. Pero a lo largo de las últimas décadas han cambiado las prioridades en el diseño y la organización de edificios según las necesidades del hombre, mismas que dependen de su hábitat y de su *modus vivendi*.

Bajo este concepto surgen los "Edificios Inteligentes" a mediados de los años 80, cuando se instalaron los primeros controles que realizaban la regulación de la temperatura ambiente (1), atrayendo la atención al ofrecer un nuevo concepto para el diseño y la construcción de edificios. La propuesta de los Edificios Inteligentes mencionó por primera vez la integración de todos los aspectos de comunicación dentro del edificio, tales como teléfono, comunicaciones por computadora, seguridad, control de todos los subsistemas del edificio (calefacción, ventilación y aire-acondicionado) y todas las formas de administración de energía[...] (2).

Al principio, el calificativo "inteligente" era simplemente una referencia al alto grado de automatización, obtenido gracias a la integración de todos los sistemas. Pero no fue hasta 1984 que se tuvo noticia del primer edificio "inteligente": el Lloyds Building, construido en Londres y proyectado por Richard Rogers (1).

Desde entonces los edificios actuales se han sometido a intensos estudios con el objetivo de cumplir con las siguientes características que califican a un edificio inteligente en nuestro país:

- a) Máxima economía. Eficiencia en el uso de energéticos.
- b) Máxima flexibilidad. Adaptabilidad a un bajo costo a los continuos cambios tecnológicos requeridos por sus ocupantes y su entorno.
- c) Máxima seguridad (Entorno, Usuario y Patrimonio). Capacidad de proveer un entorno ecológico interior y exterior respectivamente habitable y sustentable, altamente seguro que maximice la eficiencia en el trabajo a los niveles óptimos de confort de sus ocupantes.

- d) Máxima automatización de la actividad. Eficazmente comunicativo en su operación y mantenimiento.
- e) Máxima predicción y prevención. Refaccionamiento Virtual. Operación y mantenimiento bajo estrictos métodos de optimización (3).

La creatividad para el desarrollo de tecnologías que satisfagan las necesidades que tienen los usuarios de controlar los diversos sistemas de un edificio, es una actividad cotidiana para la electrónica. Basta observar el impresionante avance desde hace más de 20 años que se tiene del *hardware* y *software*, cuyo objetivo principal es la automatización integral del edificio, reduciendo el consumo de energía, aumentando el confort (incorporando sensores y actuadores por todo el edificio de tal forma que se permita disfrutar de aplicaciones como la automatización del apagado o encendido de energía eléctrica, la zonificación de la calefacción, controlar el cierre y apertura de persianas, etc.) y seguridad, todo lo anterior puede ser definido como inmótica.

El incremento de la seguridad no sólo repercute en la protección de la mayor parte de nuestros bienes particulares, sino también de nuestra protección personal. La implementación de detectores de presencia y cámaras distribuidas en puntos clave del edificio que permiten avisar, además al usuario de posibles catástrofes (incendios, escapes de agua, fugas de gas, cortes de electricidad, etc.), permiten también generar una acción, como controlar una llave de paso para detener una fuga de agua.

Las últimas innovaciones en cuanto a control de intrusión ofrecen alarmas de seguridad inalámbricas que son capaces, por ejemplo, de funcionar si el ladrón ha cortado la línea telefónica (4).

En materia de seguridad de control de presencia y acceso a ciertas zonas específicas se han diseñado diversos sistemas de control electrónico aplicados principalmente en áreas industriales; debido a la gran demanda por parte de este sector, la administración electrónica de los sistemas que conforman la inmótica de una empresa están en constante evolución, principalmente los sistemas de seguridad.

Dichos sistemas de seguridad han ido evolucionando de lo digital a lo biométrico. Por lo que la creación de las técnicas de identificación a través de las huellas dactilares,

reconocimiento de voz, reconocimiento facial y el iris en el ojo han sido las herramientas más eficaces para la identificación de personas, ya que estos métodos de identificación biométricos tienen el 98% de efectividad [...] (5) .

Por lo tanto la biometría se considera como el método más apropiado, ya que ciertos rasgos de cada persona, son inherentes a ella y sólo a ella. La biometría permite una autenticación segura, al contrario que el empleo de contraseñas o tarjetas, ya que estos últimos pueden ser clonados o utilizados por personas no autorizadas.

Para el presente trabajo la mejor opción de terminal de identificación sería por sistemas biométricos, el más accesible es el reconocimiento de huellas dactilares aunque éste elevaría el costo del sistema, sin tomar en cuenta que por la naturaleza del área de trabajo no es recomendable.

El sistema que se propone para el control de acceso es el punto de partida y puede expandirse hasta llevar a cabo la automatización integral de los sistemas que conforman las instalaciones industriales, a tal punto de que el sistema propuesto pueda convertirse en inmótica en un futuro.

2.2.- Características del área industrial con acceso restringido

Se necesita implementar un sistema electrónico que permita el control de registro de acceso de personas a un área restringida de guarda de materiales industriales (lingotes de aluminio, acero, herramientas de corte, etc.), productos manufacturados, herramientas de precisión, ubicada dentro de una nave industrial donde se inyecta y maquina refacciones automotrices. En la Figura 2.1 se describen las dimensiones del área de almacén.

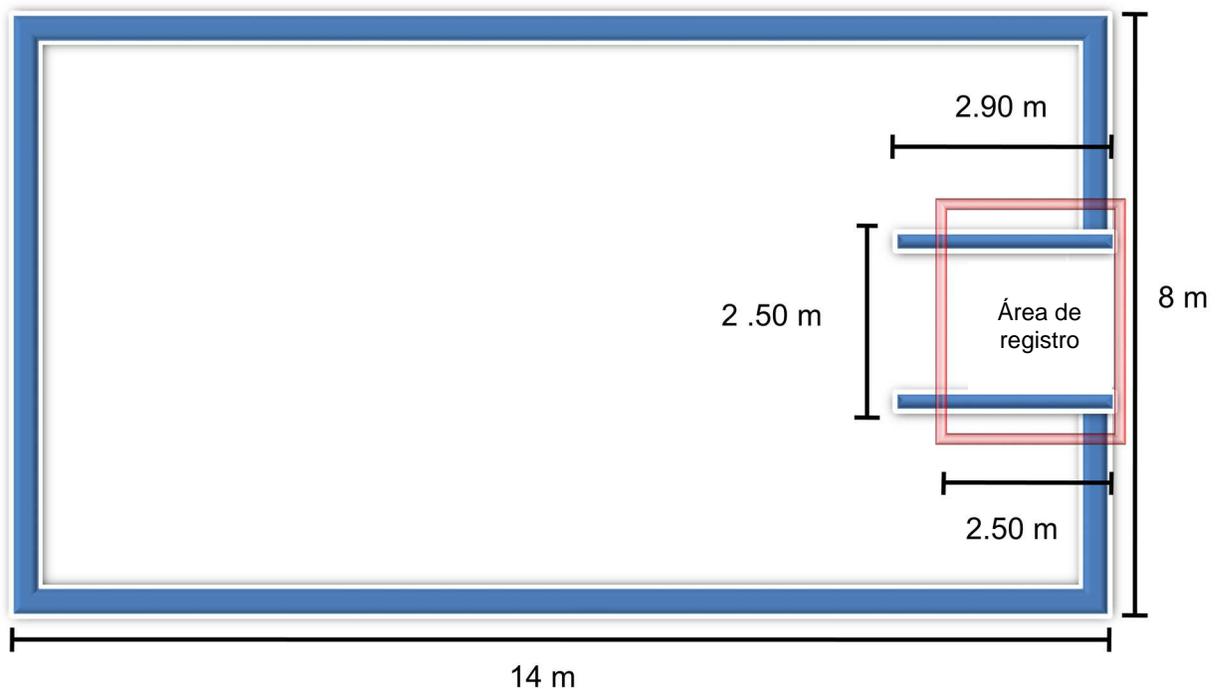


Figura 2.1 Dimensiones del área restringida

El área de registro es de 2.50 m de largo por 2.50 m de ancho, la cual representa la entrada al almacén con dimensiones de 14 m de largo por 8 m de ancho.

2.3.- Propuesta de diseño del sistema de control de acceso

Debido a los requerimientos y características del área restringida se propone implementar el siguiente sistema electrónico.

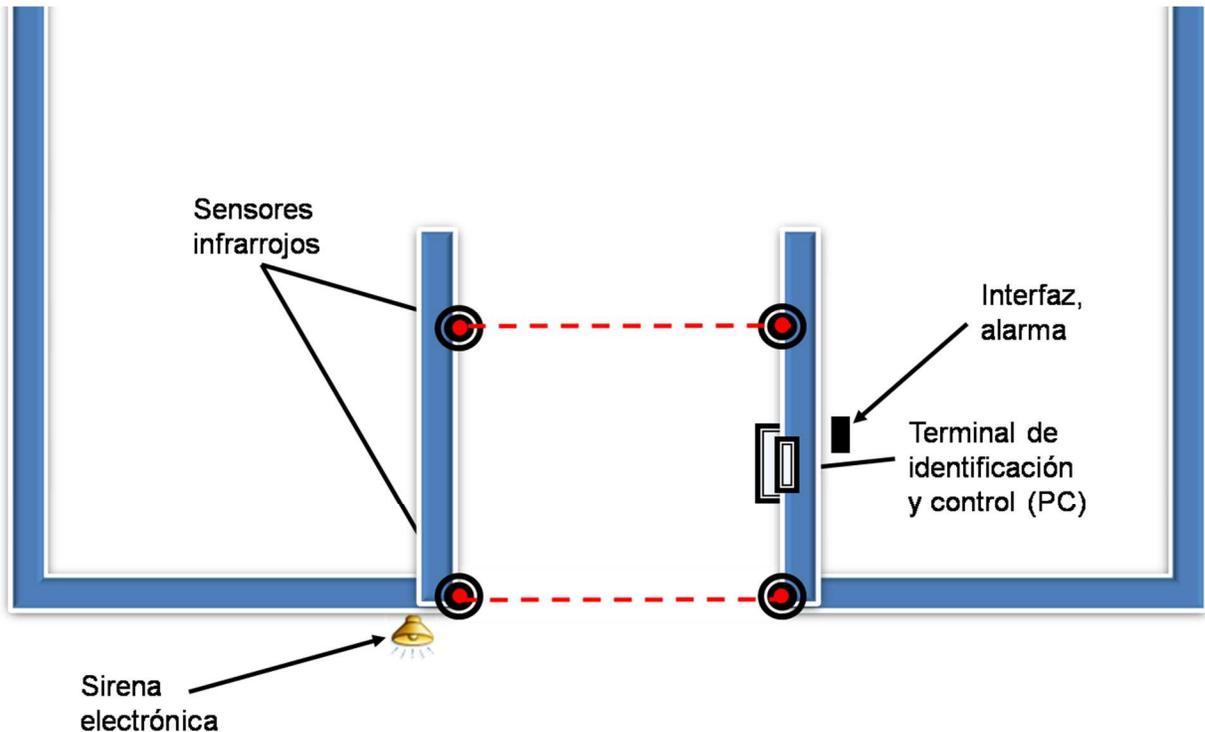


Figura 2.2 Elementos de los módulos del sistema

En primera instancia, cuando el personal desee ingresar al almacén, será detectado por unos sensores que se colocarán en la entrada del área de registro. Los sensores de presencia estarán conectados a una interfaz junto con una la alarma, la cual se activará en caso de un acceso no autorizado.

El objetivo de interconectar estos dispositivos es para que puedan ser programados o reprogramados según los parámetros de acceso que se deseen controlar. Esta interfaz estará conectada a una PC; ésta, además de controlar la interfaz será la terminal de identificación, la cual tendrá las siguientes características:

Después de que el usuario sea detectado por los sensores inmediatamente escuchará la instrucción que le pida identificarse por medio de la terminal, en la pantalla del monitor se visualizará sólo una ventana en la que deberá introducir su número de usuario y una contraseña; al capturar los datos en el sistema, este procesará la información en su base

de datos para verificar si el acceso es o no permitido, se sincronizará con la interfaz para que ésta indique al módulo de alarma que active o desactive la sirena electrónica. Con esta interconexión de los elementos antes mencionados se puede lograr el control de registros de accesos al almacén.

Los módulos que integran el sistema se alimentarán con 6 volts de corriente directa que serán suministrados mediante una fuente regulada de voltaje (ver apéndice B), dado que la terminal de identificación se encuentra en la PC para garantizar el suministro de energía lo más conveniente será el uso de un *no-break*, el cual es opcional.

CAPÍTULO III: DESCRIPCIÓN Y SELECCIÓN DE LOS MÓDULOS DEL SISTEMA DE CONTROL DE ACCESO

3.1.- Módulo de detección de presencia (barrera infrarroja)

Para detectar la presencia de las personas se propone la implementación de sensores volumétricos de tecnología infrarrojos, los cuales son los más adecuados para la detección de movimiento. Dadas las condiciones de iluminación, temperatura y dimensiones del área de registro, es conveniente la instalación de barreras infrarrojas en la entrada del almacén.

3.1.1.- Principio básico de funcionamiento de una barrera infrarroja

Los haces fotoeléctricos son una de las primeras formas de detección electrónica de intrusos. Su simplicidad de operación, acoplado con numerosos componentes y adelantos en el diseño, aseguran que ellas aún sean populares. No obstante, cuando se les usa en forma exterior o interior, su efectividad depende enormemente de cómo estas barreras de haces, superan condiciones ambientales, tales como lluvia, neblina, polvo y nieve. Es importante entender la naturaleza de la luz infrarroja y las características de los dispositivos en sí mismos, para obtener los mejores resultados de los haces fotoeléctricos en el sistema de control electrónico que se desea implementar. Algunas de las características fundamentales de los haces fotoeléctricos en las barreras infrarrojas se mencionan a continuación.

Una barrera fotoelectrónica de haz está compuesta por dos elementos principales: un transmisor de luz infrarroja y un receptor (6). Las ondas infrarrojas viajan en línea recta y no pueden atravesar paredes; como resultado, el transmisor y el receptor tienen que estar a la vista uno del otro. Cuando un intruso interrumpe la señal entre el transmisor y el receptor, se inicia una señal de salida de alarma.

La región de infrarrojas está inserta entre las frecuencias de radio más altas y la porción visible del espectro electromagnético. El haz infrarrojo es producido por un LED infrarrojo pulsante, que puede tener una longitud de onda de 0.7 a 100 μm (micrómetros) [...] (7). En

los 0.890 μm ya no es visible para el ojo humano por lo que a partir de este rango puede emplearse para barreras infrarrojas de control de acceso.

Las ventajas de utilizar una luz con esta longitud de onda son las siguientes:

- a) Invisibilidad (incluso durante la noche).
- b) Pequeña absorción o distorsión de la luz en el aire y en otros medios de densidad similar.
- c) La expectativa de vida útil de un LED transmisor está estimada en 100,000 horas (11 años aproximadamente).

Una de las desventajas de utilizar luz con esta longitud de onda puede estar dada por los efectos de un ambiente atmosférico hostil, es decir, los cambios ambientales circundantes pueden afectar un haz infrarrojo pulsante. Esto debe tenerse en cuenta especialmente en instalaciones exteriores donde las condiciones del tiempo pueden afectar a los infrarrojos a menudo más que a la luz visible. Los haces fotoeléctricos son afectados por las condiciones meteorológicas, del siguiente modo:

- a) **Turbulencia del aire originado por el calor.** Cuando se produce una turbulencia del aire causada por viento o calor, el factor de deflexión cambia y crea problemas potenciales a los haces fotoeléctricos, particularmente cuando se les utiliza en líneas largas. El fenómeno de la turbulencia del aire se destaca con mayor importancia a las horas de fuerte luz solar. Los efectos son los mismos que se originan cuando uno ve una imagen distorsionada sobre una autopista recalentada. Es importante considerar este efecto en los sistemas de haces infrarrojos durante la etapa de diseño del sistema.
- b) **Partículas acarreadas por el aire.** El polvo, los cristales de sal, el agua, el vapor, y la polución, pueden ser hallados y considerados como partículas acarreadas en el aire. Estas partículas, pueden afectar la dispersión y absorción del infrarrojo.
- c) **Absorción a través del aire.** El uso de una longitud de onda de 0.890 micrómetros, en los haces, tendrá una pérdida pequeña o nula a través del aire, donde, una longitud de onda de 1.8 micrómetros de infrarrojo experimentará un corte total del

haz. Esto es, por lo tanto, un dato crítico al seleccionar la longitud de onda apropiada para utilizar haces fotoeléctricos en exteriores (6). Pero debido a que la ubicación del área de registro es en el interior de una nave industrial, ésta característica no es tomada en cuenta, sin embargo, se propone la utilización de los siguientes sensores con longitud de onda de .940 micrómetros.

3.1.2.- Características de la barrera infrarroja

La Figura 3.1 ilustra la barrera que se implementará en el módulo de detección del personal. Es una barrera infrarroja de marca Infrarrojos.net que consta de un emisor y un receptor con filtro de luz de día, por lo que es inmune a los cambios de temperatura y a la luz del medio ambiente, además tiene un ángulo de captación de 32° y un alcance máximo de 6 m en interiores.

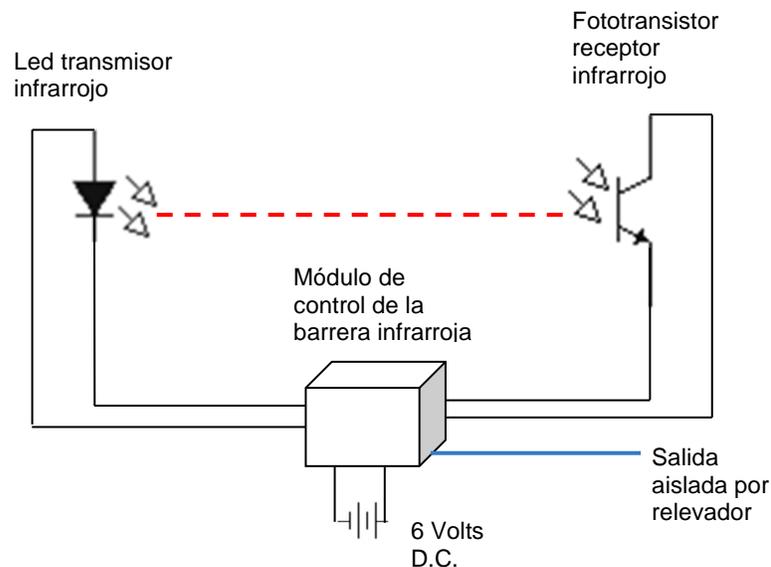


Figura 3.1 Barrera infrarroja propuesta

El módulo de control de la barrera basa su funcionamiento en una etapa codificadora y transmisora de pulsos. Este sistema utiliza un microcontrolador tipo 12F629, el cual contribuye a la estabilidad del mismo y a la inmunidad contra falsas alarmas. Además de poseer una salida aislada por relevador de contactos de aleación de plata que soporta 120 Volts de corriente alterna a 10 Amperes.

Esta barrera se alimenta con 4 baterías de 1.5 Volts tipo AA, batería de 6 Volts o una fuente de voltaje que suministre 6 volts de corriente continua.

Los sensores de las barreras trabajan a .940 micrómetros que es considerado como infrarrojo cercano (8).

La Figura 3.2 muestra el encapsulado metálico con rosca para simular un tornillo, del que está hecho el receptor y transmisor infrarrojo. Estas características de camuflaje servirán para que al personal le sea difícil traspasar los sensores sin que estos detecten su presencia (8).



Figura 3.2 Encapsulado metálico

3.2.- Módulo de identificación y control

La Terminal de Identificación será una PC de uso interno de la empresa por lo que no generará costo alguno. La PC desempeñará las funciones de control al procesar la información de la interfaz, y como terminal de identificación a través de un programa hecho en la plataforma del software orientado a objetos *Visual Basic 6.0*, con una base de datos elaborada en *Microsoft Access*.

Esta computadora es tipo Pentium III, con una memoria RAM de 256 MB y con una capacidad de disco duro de 40 GB y cuenta con un procesador de 1.6 GHz.

Se sugiere implementar este tipo de terminal de identificación ya que los materiales con los que se trabaja y los procesos de elaboración de productos llevados a cabo en esa nave industrial, suelen desgastar las huellas dactilares de los empleados, por lo que no es recomendable instalar una terminal biométrica de huellas dactilares, porque habría problemas de reconocimiento de las mismas, además de que los costos serían mayores.

3.2.1.- Características de la base de datos

Como primer paso para la elaboración del programa *ODBMS (Object Database Management System)* en el *software* orientado a objetos *Visual Basic 6.0* se necesitará la elaboración de una base de datos cuya finalidad es almacenar la información relativa a cada persona que labora en la empresa y que podría tener acceso al área restringida.

El sistema de administración de base de datos deberá ser capaz de representar relaciones de los datos almacenados y crear los archivos lógicos que se requieran para tener una capacidad de explorar los datos rápidamente y con diferentes criterios de búsqueda, además de garantizar la integridad, privacidad y seguridad de la información almacenada.

La base de datos tendrá dos tipos de usuarios:

- Usuario Administrador: tendrá acceso a toda la BDD (base de datos) con la facultad de recuperar, crear, supervisar y modificar los datos contenidos en la misma.
- Usuario Final: no tendrá acceso a la BDD, por lo que sus funciones se limitan a ingresar un número de usuario y una contraseña cuando desee ingresar al área restringida.

La BDD es dinámica y de tipo relacional, ya que se tiene planeado convertirla en una BDD orientada a objetos para que el usuario administrador pueda manipular desde el *ODBMS (Object Date Base Manager System)*, es decir desde el *software* elaborado en *Visual Basic 6.0* las siguientes características.

- Tablas para almacenar datos correspondientes a las sesiones efectuadas por el personal.
- Consultas para buscar y recuperar únicamente los datos que se necesitaran.
- Ver, agregar y actualizar los datos de las tablas.
- Crear informes para analizar o imprimir los datos.

Esta base de datos deberá contener, entre otros, los siguientes campos generales:

- a) Número de usuario. Son dígitos de identificación personal para cada empleado cuya función, además de identificar al usuario, es ser la clave primaria en la base de datos.

- b) Contraseña de usuario. Contraseña única e intransferible para cada empleado para que éste pueda ingresar al área.
- c) Nombre, apellidos.
- d) Cargo. Describe la labor del personal dentro de la empresa.
- e) Fecha. Fecha correspondiente a cada sesión del usuario.
- f) Hora de entrada. Es la hora de inicio de sesión del usuario.
- g) Hora de salida. Es la hora del término de sesión del usuario.
- h) Tiempo total de permanencia. Este campo contabilizara el tiempo total de sesión.

La Figura 3.3 muestra la relación de las tablas así como los campos contenidos en la BDD

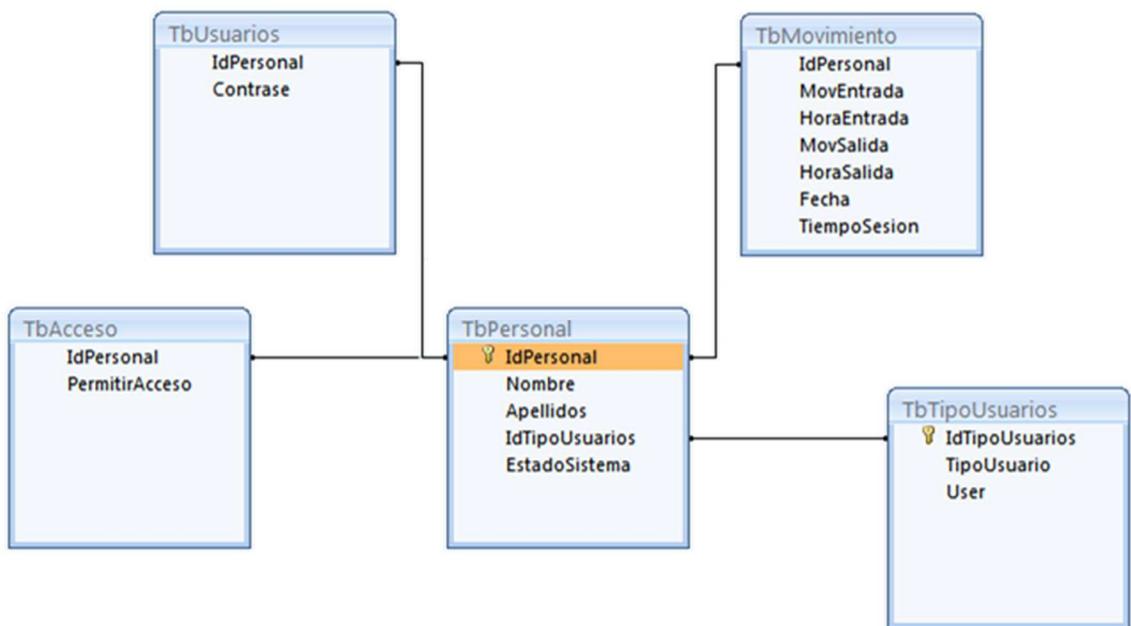


Figura 3.3 Relación de tablas y campos de la BDD

La tabla TbPersonal es la principal, cuya función es relacionar las tablas: TbAcceso, TbUsuarios, y TbMovimiento, por medio de una llave principal que representa el campo IdPersonal. La tabla TbTipoUsuarios se relaciona con la TbPersonal a través del campo IdTipoUsuarios como llave.

3.3.- Módulo de alarma

El objetivo de instalar una alarma en el sistema propuesto es establecer un indicador del estado de seguridad de acceso al almacén, es decir, será la encargada de informar por medio de la emisión de un sonido si las reglas de seguridad en el área restringida fueron irrumpidas.

En el presente trabajo se propone el siguiente módulo de Alarma de marca Infrarrojos.net, el cual posee las siguientes características:

- a) El estado de activación y desactivación es por medio de un control remoto infrarrojo *Sony* o tipo universal programado como *Sony* que envía un código de 4 dígitos, este código es programable y guardado en una memoria no volátil. Alcance máximo 10 m.
- b) El módulo de control funciona con 4 baterías de 1.5 V tipo AA, batería de 6V o una fuente que suministre 6 volts de corriente continua.
- c) Led de Activación: Indica que el sistema está activado.
- d) Se le puede conectar cualquier tipo de sensores tales como: sensores detectores de ruptura de vidrios, sensores de movimiento, sensores magnéticos, sensor de proximidad, sensor infrarrojo de barrera.
- e) Salida controlada por relevador: ésta permanece activa y desactivada, el tiempo que se le programe. Los contactos del relevador con aleación de plata soportan 10 amperes, incluso se pueden conectar varios dispositivos simultáneamente el único límite es de 10 A el total de la carga.
- f) Salida para marcador telefónico, sirena, luces, motor, etc.
- g) Protección de cableado: si se corta el cableado del *switch* de activación o de cualquier sensor se activa automáticamente.
- h) *Reset* automático de activación: el sistema se reactiva automáticamente después de haber pasado cierto tiempo pre-programado de que la sirena o luces estuvieron activas.
- i) Zona de activación retardada: esta zona queda habilitada después de cierto tiempo programado, de haber sido activado el sistema.

- j) Zona de activación Instantánea e interruptor de pánico: se activa inmediatamente la salida de alarma independientemente de que esté o no activado el sistema (8).

En la Figura 3.4 se muestran las conexiones de este módulo a implementar.

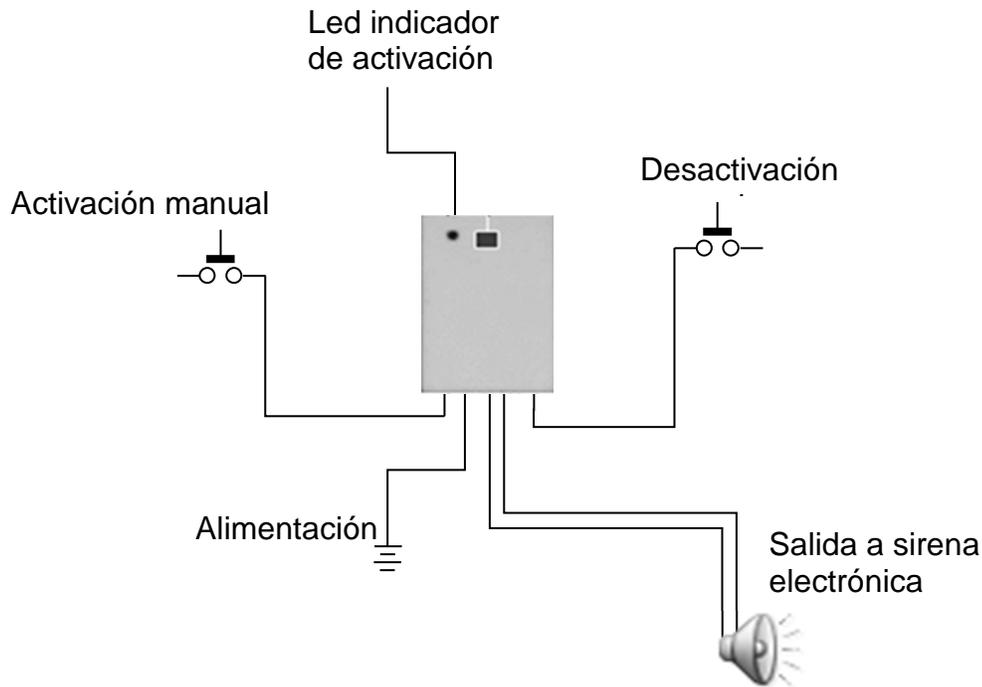


Figura 3.4 Módulo de alarma

3.4.- Módulo de interfaz

Para la interconexión de los demás elementos del sistema se propone la implementación de una interfaz.

Esta interfaz será el *hardware* y *software* que permitirá la conexión de diferentes procesos de los dispositivos propuestos para el control de acceso.

La interfaz basa su funcionamiento en un microcontrolador el cual estará colocado sobre una placa de circuito impreso, ésta se conectará a la Terminal de Identificación (PC) por medio del puerto serie.

3.4.1.- Funcionamiento básico de un microcontrolador

Un microcontrolador es un circuito integrado programable de alta escala de integración que incorpora en su interior las tres unidades funcionales: unidad central de proceso (CPU), memoria y entrada/salida (E/S).

Es considerado un sistema cerrado porque todos sus elementos están contenidos en su interior y sólo salen al exterior las líneas que gobiernan los periféricos como lo muestra la Figura 3.5 (9).

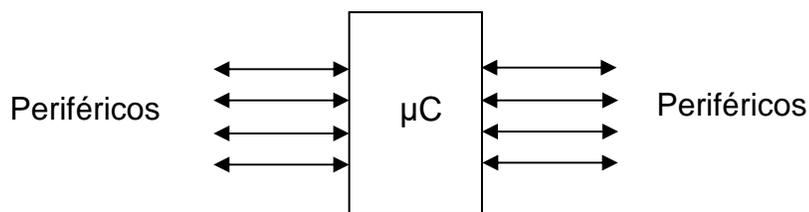


Figura 3.5 Microcontrolador

Los elementos principales que componen a un microcontrolador son los siguientes:

- a) CPU (Unidad central de proceso).
- b) Memoria RAM (para contener los datos).
- c) Memoria para el programa tipo: ROM/PROM/EEPROM.
- d) Líneas de E/S (entrada/salida) para comunicarse con el exterior.
- e) Diversos módulos para el control de periféricos (Temporizadores, puertos serie y paralelo, etc.)
- f) Generador de pulsos de reloj que sincronizan el funcionamiento de todo el sistema.

La estructura básica de un microcontrolador se ilustra en la Figura 3.6 (10):

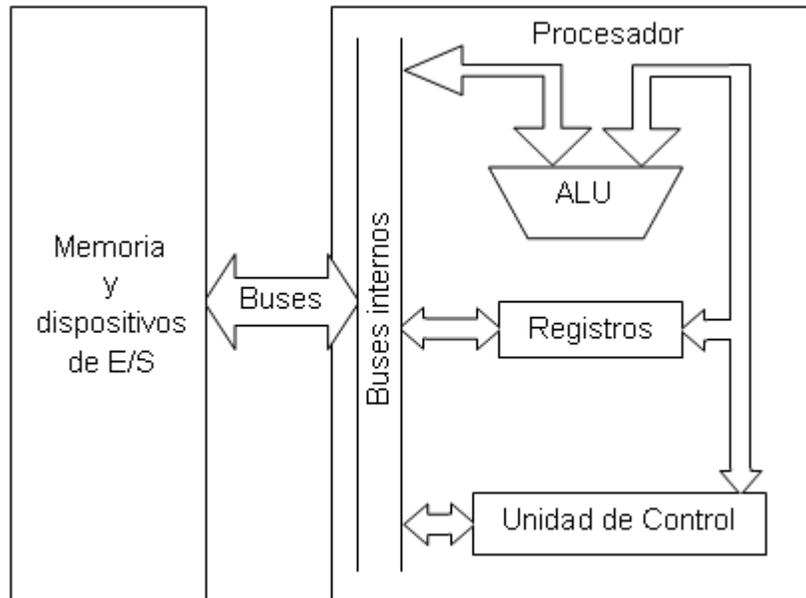


Figura 3.6 Estructura básica de un microcontrolador

Registros. Conjunto de bits que son considerados como una sola entidad [...] (11), sirven para almacenar los resultados de la ejecución de instrucciones, cargar datos desde la memoria externa o almacenarlos en ella.

Existen dos tipos de registros:

- a) [Registro de estado y control. En este registro, que se encuentra alojado en la SRAM, se mantiene el estado actual del microcontrolador: algunos bits llamados banderas, indican hechos tales como el signo de la última operación realizada, si hubo acarreo en la última suma o resta, etc.][...] (11)
- b) [Registro de propósito general. Son registros de 8 bits que pueden ser utilizados como operandos en las operaciones aritméticas y/o lógicas, como fuentes o destinos de datos en operaciones de transferencia de datos dentro del microcontrolador y también permiten el acceso a memoria SRAM, EEPROM y FLASH.][...] (11)

Unidad aritmética y lógica (ALU). Es parte de la unidad central de proceso (CPU), proporciona capacidades para realizar cálculos: computacionales (operaciones aritméticas simples), lógicos, decisiones lógicas, transferencia de datos.

Unidad de control. En ella recae la lógica necesaria para la decodificación y ejecución de las instrucciones, el control de los registros, la ALU y los buses, porque se encarga de agrupar componentes tales como la unidad de decodificación, unidad de ejecución, controladores de memoria cache, controladores de buses, controladores de interrupción, *pipelines*, entre otros elementos, dependiendo siempre del tipo de microcontrolador.

Buses. Son el medio de comunicación que utilizan los diferentes componentes del microcontrolador para intercambiar información entre sí y hacia el exterior. Existen tres tipos de buses:

- **Dirección:** Conjunto de líneas que conforman la capacidad de direccionamiento de los dispositivos con los cuales se requiere trabajar.
- **Datos:** Conjunto de líneas que conforman la capacidad de transmisión de datos del sistema.
- **Control:** Conjunto de líneas utilizadas para gestionar los distintos procesos de escritura y lectura; y también empleadas para controlar la operación de los dispositivos del sistema. (10)

3.4.2.- Características generales del microcontrolador ATmega8535

Los AVR son una familia de microcontroladores del Corporativo *ATMEL*, con procesadores tipo *RISC* (Computadora con conjunto de instrucciones reducido) lo que significa que cuentan con un código eficiente y poderoso para poder ser programados y aplicados en diversas áreas. Por ejemplo, en la industria automotriz, en la cual *ATMEL* ha diseñado sus microcontroladores para brindar un mejor desempeño y un costo efectivo en soluciones automotrices para control de sensores y actuadores, además de una red interna del vehículo con *CAN* (*Controller Area Network*) y *LIN* (*Local Interconnect Network*).

Los microcontroladores AVR cuentan con una tecnología en memoria flash de baja potencia, proporcionando capacidades auto-programables sumamente flexibles y una alta integración de sistema; ejecutando poderosas instrucciones en un solo ciclo de reloj, los microcontroladores AVR logran un rendimiento aproximado de alrededor de un millón de instrucciones por segundo (MIPS) por MHz, lo cual permite optimizar sistemas.

La arquitectura de AVR proporciona un alto nivel en densidad y compatibilidad de código a través de toda la familia. A diferencia de otras arquitecturas de 8 bits, no hay necesidad de requerir otro código, ni convertir el mismo, ya sea cuando emigra a un dispositivo con una densidad diferente de programa o a opciones periféricas.

Debido a las grandes ventajas de implementación de microcontroladores de la Familia AVR, el microcontrolador que se propone es el ATmega8535L de la Figura 3.7, de baja potencia, tecnología CMOS de 8 bits, basado en arquitectura AVR, con procesador tipo RISC el cual le permite: una codificación uniforme de instrucciones (ejemplo: el código de operación se encuentra siempre en la misma posición de bit en cada instrucción, la cual es siempre una palabra), lo que permite una decodificación más rápida; un conjunto de registros homogéneo, permitiendo que cualquier registro sea utilizado en cualquier contexto y así simplificar el diseño del compilador; modos de direccionamiento simple con modos más complejos reemplazados por secuencias de instrucciones aritméticas simples. Tiene 32 registros de propósito general, trabaja con 8 Kbytes de memoria *FLASH*, 512 Kbytes en memoria *EEPROM* y 512 Kbytes en memoria *SRAM*. El rango de velocidad es de 0-8MHz y su voltaje de alimentación es de 6.0 Volts. (10).



Figura 3.7 Microcontrolador AVR ATmega8535L

3.4.3.- Características específicas del microcontrolador para la comunicación serial

Para llevar a cabo la comunicación serial es necesario establecer la configuración y las funciones necesarias para esto. La *USART (Universal Synchronous and Asynchronous serial Receiver and Transmitter)* permite al microcontrolador la comunicación serial

RS-232 con otros dispositivos, es un recurso muy flexible de comunicación ya que permite comunicación síncrona o asíncrona, múltiples “*baud rates*”, ancho de caracter configurable y detección de errores de transmisión.

Cada caracter de transmisión consta de:

- 1 bit de comienzo.
- 5, 6, 7, 8 ó 9 bits de datos.
- 1 bit de paridad.
- 2 bits de stop (nivel alto).

Los registros de configuración y funcionamiento del USART son los siguientes:

Registro de datos Figura 3.8:

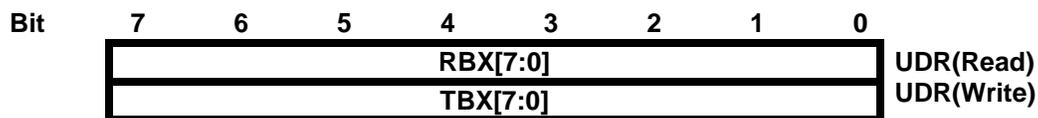


Figura 3.8 Registro de datos

RXB = Buffer de lectura.

TXB = Buffer de escritura.

Registro A de control y estado USART Figura 3.9:



Figura 3.9 Registro A de control y estado USART

RXC = A '1' si hay datos sin leer del RXB.

TXC = A '1' si la transmisión se ha completado.

UDRE = Este bit indica si la UDR (*USART Data Register*) está lista para recibir un nuevo dato, es decir si UDRE (*USART Data Register Empty*) es igual a uno, UDR está vacío y entonces el buffer está listo para ser escrito.

FE, DOR y PE = Control de errores de trama, sobrecarga y paridad, respectivamente.

U2X = A '1' duplica la velocidad de transmisión en modo asíncrono.

MPCM = Habilita el modo de comunicación del multi-procesador.

Registro B de control y estado USART Figura 3.10:

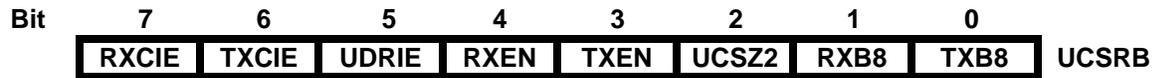


Figura 3.10 Registro B de control y estado USART

RXCIE, TXCIE y UDRIE = Habilita interrupciones en RXC, TXC y UDRE.

RXEN y TXEN = Habilita el receptor y transmisión.

UCSZ2 = Tamaño del caracter.

RXB8 y TXB8 = El noveno caracter en recepción o transmisión.

Registro C de control y estado USART Figura 3.11:



Figura 3.11 Registro C de control y estado USART

UMSEL = Selecciona entre modo síncrono (1) o asíncrono (0).

UPM1-0 = Modo de paridad

UCSZ1-0 = Tamaño de caracter.

Registro de selección de *baud rate* Figura 3.12:

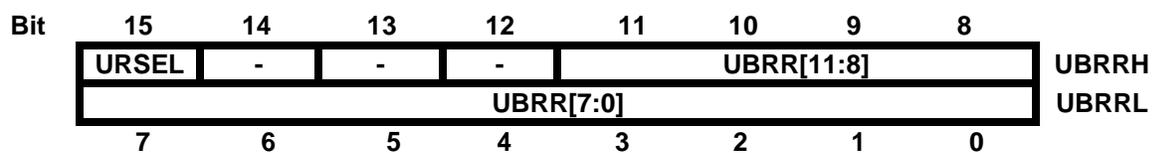


Figura 3.12 Registro de selección de baud rate

URSEL(bit 15) = Selecciona entre acceder a UBRRH ('0') o UCSRC ('1'), es leído como cero cuando se escribe en UBRRH.

Bit 14:12 = Reservado para usos futuros.

Bit 11:0 - UBRR11:0 = Este registro contiene la selección de baud rate según el cristal que genera el reloj. UBRRH contiene los cuatro bits más significativos y UBRL los ocho bits menos significativos (12).

3.4.4.- Transmisión de datos

La transmisión de datos entre la PC y el microcontrolador será a través de señales eléctricas que representarán el movimiento de información binaria.

Los caracteres se convierten en corriente eléctrica. El estándar utilizado para representar estos caracteres es el código ASCII, acrónimo de American Standard Code for Information Interchange.

Este esquema de codificación asigna valores numéricos a las letras, números, signos de puntuación y algunos otros caracteres. Al normalizar los valores utilizados para dichos caracteres, ASCII permite que las computadoras y programas informáticos intercambien información. ASCII incluye 256 códigos divididos en dos conjuntos, estándar y extendido, de 128 códigos cada uno. Estos conjuntos representan todas las combinaciones posibles de 7 u 8 bits, siendo esta última el número de bits en un byte, como lo muestra la Figura 3.13.

				6,5,4	000	001	010	011	100	101	110	111
3	2	1	0	Hex	0	1	2	3	4	5	6	7
0	0	0	0	0	NUL	DLE	SP	0		P		p
0	0	0	1	1	SOH	SBA	!	1	A	Q	a	q
0	0	1	0	2	STX	EUA	"	2	B	R	b	r
0	0	1	1	3	ETX	IC	#	3	C	S	c	s
0	1	0	0	4	EOT	RA	\$	4	D	T	d	t
0	1	0	1	5	ENQ	NAK	%	5	E	U	e	u
0	1	1	0	6		SYN	&	6	F	V	f	v
0	1	1	1	7		ETB	'	7	G	W	g	w
1	0	0	0	8			(8	H	X	h	x
1	0	0	1	9	PT	EM)	9	I	Y	i	y
1	0	1	0	A	NL	SUB	*	:	J	Z	j	z
1	0	1	1	B		ESC	+	;	K	[k	
1	1	0	0	C	FF	DUP	,	<	L	\	l	
1	1	0	1	D		SF	-	=	M]	m	
1	1	1	0	E		FM	.	>	N	^	n	
1	1	1	1	F		ITB	/	?	O	_	o	

Figura 3.13 Conjunto de códigos de ASCII

El conjunto ASCII básico, o estándar, utiliza 7 bits para cada código, lo que da como resultado 128 códigos de caracteres desde 0 hasta 127 (00H hasta 7FH hexadecimal). El conjunto ASCII extendido utiliza 8 bits para cada código, dando como resultado 128 códigos adicionales, numerados desde el 128 hasta el 255 (80H hasta FFH extendido). En el conjunto de caracteres ASCII básico, los primeros 32 valores están asignados a los códigos de control de comunicaciones y de impresora -caracteres no imprimibles, como retroceso, retorno de carro y tabulación- empleados para controlar la forma en que la información es transferida desde una computadora a otra o desde una computadora a una impresora. Los 96 códigos restantes se asignan a los signos de puntuación corrientes, a los dígitos del 0 al 9 y a las letras mayúsculas y minúsculas del alfabeto latino.

Los códigos de ASCII extendido, del 128 al 255, se asignan a conjuntos de caracteres que varían según los fabricantes de computadoras y programadores de software. Estos códigos no son intercambiables entre los diferentes programas y computadoras como los caracteres ASCII estándar. Por ejemplo, IBM utiliza un grupo de caracteres ASCII extendido que suele denominarse conjunto de caracteres IBM extendido para sus computadoras personales. Apple Computer utiliza un grupo similar de caracteres ASCII extendido para su línea de computadoras Macintosh. Por ello, mientras que el conjunto de caracteres ASCII estándar es universal en el hardware y el software, los caracteres ASCII extendido pueden interpretarse correctamente sólo si un programa, computadora o impresora han sido diseñados para ello.

El Conjunto de caracteres ASCII, en informática es un código estándar de 7 bits para la representación de caracteres - letras, dígitos, signos de puntuación e instrucciones de control- con valores binarios. El intervalo de los valores del código es de 0 a 127. Aunque ASCII carece tanto de los acentos como de los caracteres especiales utilizados en diversos idiomas europeos, y no es capaz de representar caracteres en los alfabetos no latinos, reviste importancia por ser el sistema de codificación de caracteres más internacional. Muchos conjuntos de caracteres latinos no ingleses son extensiones o modificaciones del sistema de codificación ASCII. La mayoría de los sistemas de computadoras personales utilizan un código ASCII extendido o modificado de 8 bits, con

Así mismo, la transmisión de datos se realizará mediante el procedimiento asíncrono, cada carácter (byte) a ser transmitido es delimitado por un bit denominado de cabecera o arranque, uno o dos bits denominados de terminación o parada. El bit de arranque tiene las funciones de sincronización de los relojes del transmisor y del receptor. El bit de parada, se usa para separar un carácter del siguiente.

También se empleará **RS-232** como una interfaz de bajo nivel en la capa física para la comunicación serial, en **9 Pines**, la cual define una interfaz de tipo no balanceada empleando un intercambio en serie de datos binarios a velocidades de transmisión superiores a los 20,000 bps, opera con datos síncronos pero está limitada por una longitud de cable de aproximadamente de 15.24 m. La norma correspondiente a esta interfaz es la siguiente.

Norma RS-232: es la más popular y a la vez es una de las más antiguas de todas las normas empleadas en las comunicaciones seriales; esta norma cubre los tres aspectos siguientes, entre el equipo terminal de datos (DTE), y el equipo de comunicación de datos (DCE):

- a) Las características eléctricas de las señales.
- b) Las conexiones mecánicas de los conectores.
- c) La descripción funcional de las señales usadas.

La norma RS-232 fue desarrollada en la década de los 60 para gobernar la interconexión de equipos terminales con MODEMS; una desventaja para esta norma RS-232 está en que sus niveles lógicos no son compatibles TTL (Transistor-Transistor Logic), esto es porque con niveles lógicos TTL no se puede transmitir mas allá de los 15 metros de longitud y esto a muy bajas velocidades, los niveles de voltaje para la norma RS-232 son:

- a) Para un uno lógico desde -3.0 volts hasta -15.0 volts
- b) Para un cero lógico desde +3.0 volts hasta +15.0 volts

La longitud máxima no debe sobrepasar los 15 metros, esto depende de las velocidades de transmisión empleadas, aunque es posible generar los niveles de voltaje de la norma RS-232C con componentes discretos, existen en la actualidad circuitos integrados excitadores de línea y receptores integrados que satisfacen estas especificaciones, como por ejemplo los circuitos MC1488, MC1489 y el MAX232.

CAPÍTULO IV: IMPLEMENTACIÓN DEL SISTEMA

4.1.- Visión global de funcionamiento de la interconexión de los módulos.

La siguiente parte de la elaboración de este sistema es el desarrollo de la implementación, punto en el que se unen el *software* correspondiente al *firmware* del microcontrolador del módulo de interfaz y el *software ODBMS* del módulo de identificación y control, así como el *hardware* del sistema.

Los procesos de interconexión de los dispositivos de los diferentes módulos propuestos para el sistema de registro de acceso se muestran a través del diagrama de flujo de la Figura 4.1. Dicho diagrama está dividido en dos secciones: en la primera (Figura 4.1 A) se describen los procesos que realiza el *firmware* del microcontrolador, los cuales llevan a cabo la comunicación con el módulo de detección (barreras infrarrojas), módulo de alarma y el establecimiento de una comunicación serie asíncrona con el módulo de identificación y control (PC/*ODBMS*) mediante el envío y recepción de caracteres. La segunda sección (Figura 4.1B) corresponde a los procesos que realiza el módulo de identificación y control, el cual como ya se mencionó es el encargado de mantener el control de entradas y salidas de los usuarios mediante registros almacenados en una BDD; de establecer la comunicación con la interfaz (microcontrolador) y brindar instrucciones al usuario para su correcto registro por medio del audio de la PC.

Con base en el diagrama de flujo se realizaron las simulaciones correspondientes del sistema, tomando en cuenta los siguientes casos:

- Caso 1: cuando una sola persona intenta ingresar al área.
- Caso 2: cuando esta misma persona desea salir.
- Caso 3: cuando 2 ó más personas pretendan entrar y/o salir.
- Caso 4: cuando una persona entra y otra desea salir.

De los cuales todos los casos cumplieron con el objetivo de llevar a cabo el registro del personal, mediante las simulaciones, mismas que fueron hechas en el *software Programming Editor*.

Debido a que durante la prueba de simulación los resultados fueron satisfactorios, el desarrollo de la interfaz del sistema puede pasar a la siguiente etapa de programación en lenguaje C.

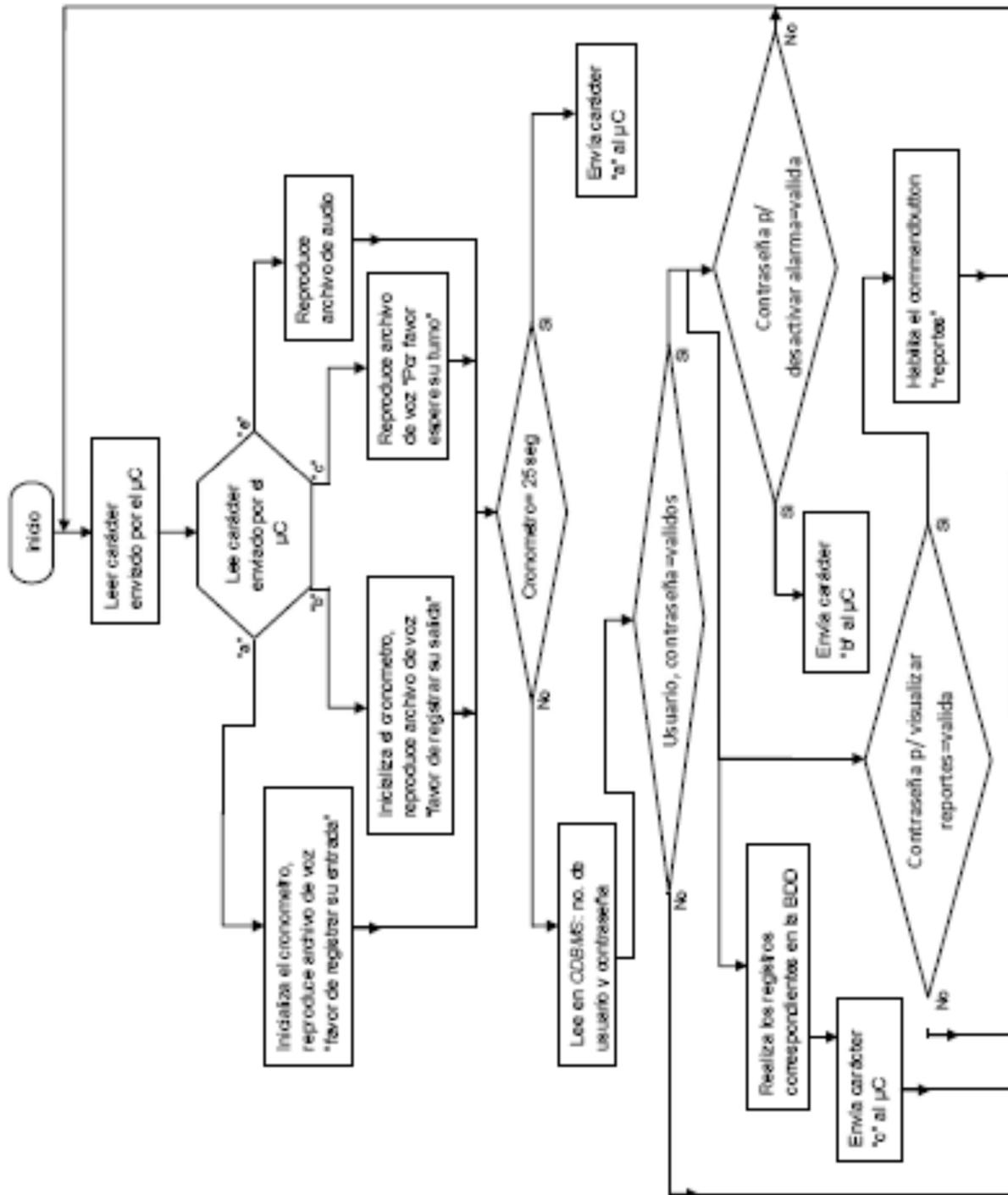


Figura 4.1 B Diagrama de flujo. (A) módulo de interfaz (µC).
 (B) módulo de identificación y control (PC/DBMS)

4.2.- Descripción del hardware

En esta sección se dará una breve descripción de los diversos circuitos físicos, así como de su funcionamiento individual y características de conexión.

4.2.1.- Desarrollo del módulo de detección de presencia (barreras infrarrojas)

Tomando en cuenta las características de la entrada al almacén, y la posición donde se requiere colocar las barreras para la detección eficiente del personal se propone la siguiente ubicación para las barreras infrarrojas:

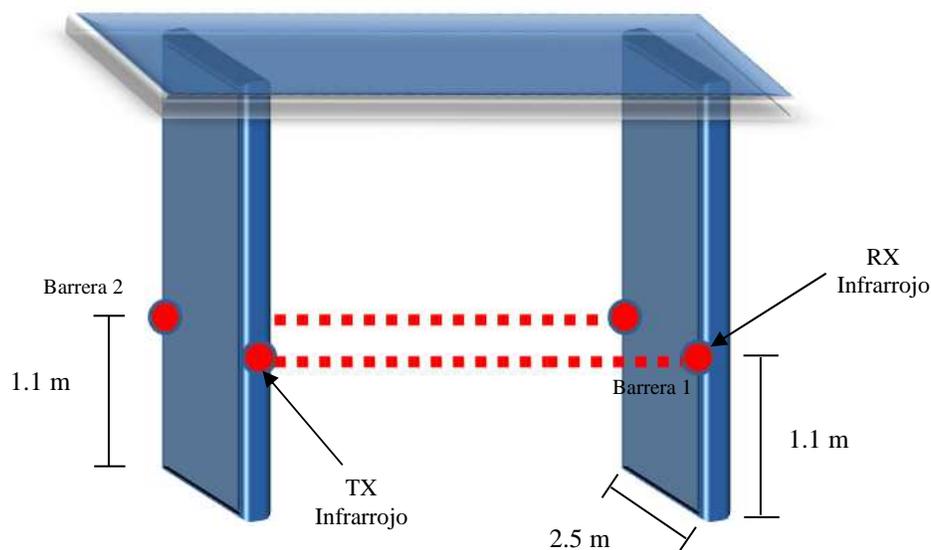


Figura 4.2 Colocación de las barreras infrarrojas

La colocación de las barreras infrarrojas en el área de registro será como lo muestra la Figura 4.2. La primera barrera estará posicionada a 1.10 m del nivel del suelo, del lado izquierdo se colocará el transmisor (LED infrarrojo) y del lado derecho el receptor (fototransistor); y a 2.50 m medidos de manera horizontal desde donde se encuentra la primera barrera, se instalará la segunda barrera.

4.2.1.1.- Modos de operación

Debido a la frecuencia en la que el personal ingresa y egresa del almacén, las barreras infrarrojas en conjunto con los demás dispositivos electrónicos operarán de los siguientes modos:

- a) **Modo 1 entrada al área de almacén.** Cuando una persona ingresa al área atravesando la barrera 1 como lo muestra la Figura 4.3, se le enviará una señal lógica al microcontrolador, activada por la salida del relevador del módulo de control de la barrera, al recibir esta señal el microcontrolador le enviará a la computadora a través del puerto serie la indicación de que alguien ha cruzado la barrera 1 e inmediatamente se escuchará la frase “Favor de registrar su entrada”, si la barrera 1 vuelve a activarse, antes de que termine de registrarse la persona que ya se encuentra en el área, el microcontrolador le indicará a la computadora que alguien más está intentando entrar y la segunda persona escuchará la frase “Por favor espere su turno”. El usuario tendrá 25 segundos para introducir su contraseña en la terminal de identificación, si no lo hace se accionará la alarma, sí se identifica a tiempo y tiene acceso denegado tendrá que abandonar el área, pero si tiene acceso permitido podrá cruzar la barrera 2 sin ningún problema.

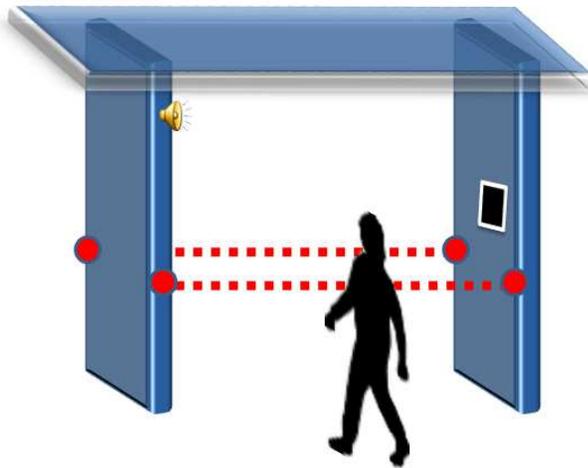


Figura 4.3 Modo 1 entrada

- b) **Modo 2 salida.** La Figura 4.4 hace referencia a la acción de la persona al salir del área (Figura 4.4 A) y a su paso atraviesa la barrera 2 (Figura 4.4 B), en ese instante el módulo de control de la barrera 2 enviará al microcontrolador una señal lógica, este a su vez le indicará a la computadora que alguien está a punto de salir por lo que se escuchará la frase: “Favor de registrar su salida”. El usuario tendrá 25 segundos para registrar su salida introduciendo nuevamente su contraseña en la PC, sí no lo hace se accionará la alarma. De igual manera que en el modo 1, cuando hay una persona

registrándose y otra persona cruza alguna de las barreras, se le dará la instrucción de “Por favor espere su turno”.

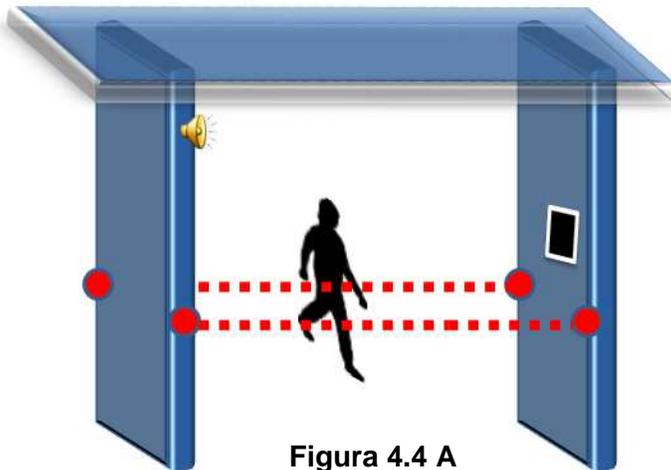


Figura 4.4 A

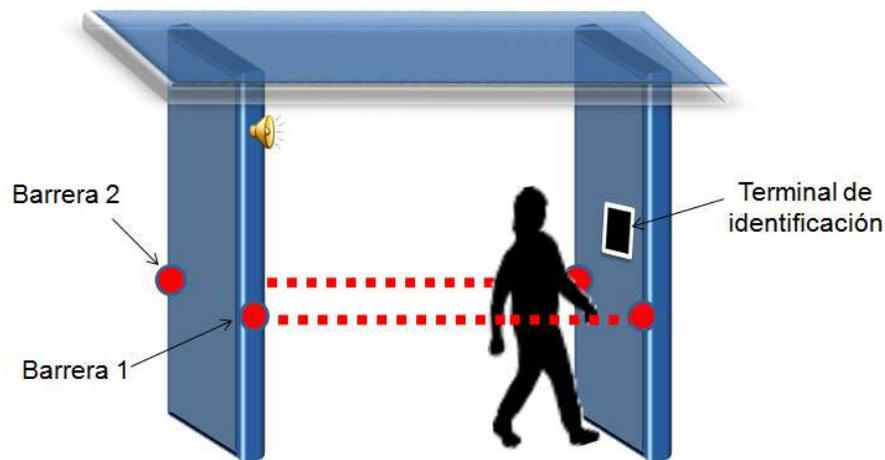


Figura 4.4 B

Figura 4.4 Modo 2 salida. (A) usuario atraviesa barrera2, (B) usuario efectúa registro

4.2.2.- Desarrollo del módulo de alarma

El modo en que opera la alarma es en la zona de activación instantánea, es decir, se activa inmediatamente la salida de alarma, mediante un nivel bajo lógico que recibe del microcontrolador, y es desactivada por medio de otro nivel bajo lógico aplicado a sus terminales de activación y desactivación manual.

La colocación de la sirena de la alarma en el área de almacén será como lo ilustra la Figura 4.5, a 2.10 metros del nivel del piso, en el lado superior derecho de la entrada.

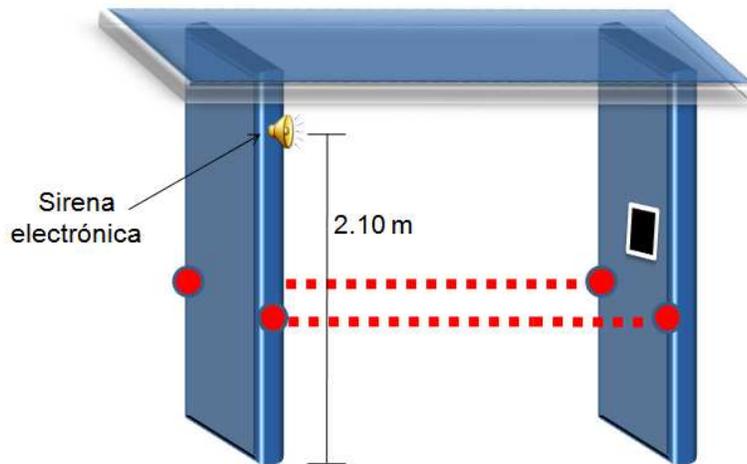


Figura 4.5 Colocación de la sirena de alarma

4.2.3.- Terminal de identificación

Se entiende por terminal de identificación a la PC, ésta tendrá siempre en pantalla el programa con el cual se llevará a cabo el registro de entradas y salidas al área de almacén, así como la comunicación serie al microcontrolador y será colocada en medio de los receptores infrarrojos a 1.20 m del nivel del piso, como se ilustra en la Figura 4.6.

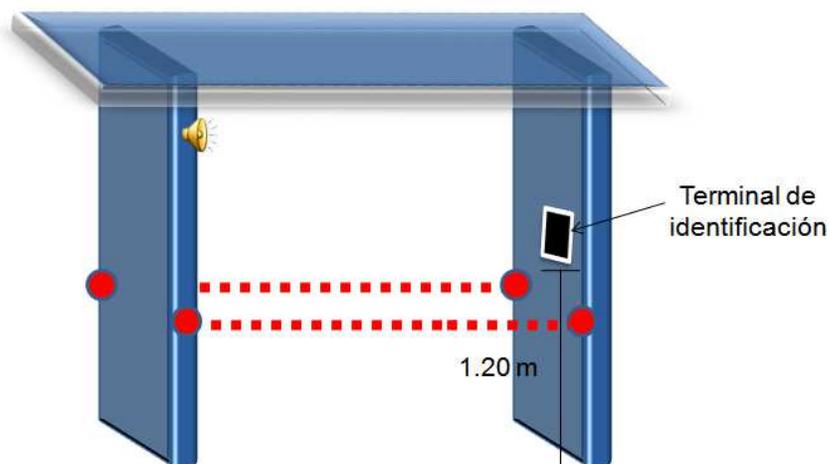


Figura 4.6 Colocación de la terminal de identificación

4.2.4.- Diagramas de conexión

- a) **De las barreras infrarrojas al microcontrolador.** Las barreras están conectadas al puerto A mediante los pines: PA0 para la barrera 1 y PA1 para la barrera 2 del microcontrolador, éstas se encuentran enviando un cero lógico hasta el momento en que son interrumpidas y esto hace que el microcontrolador reciba un uno lógico. En la Figura 4.7 se muestra la conexión física.

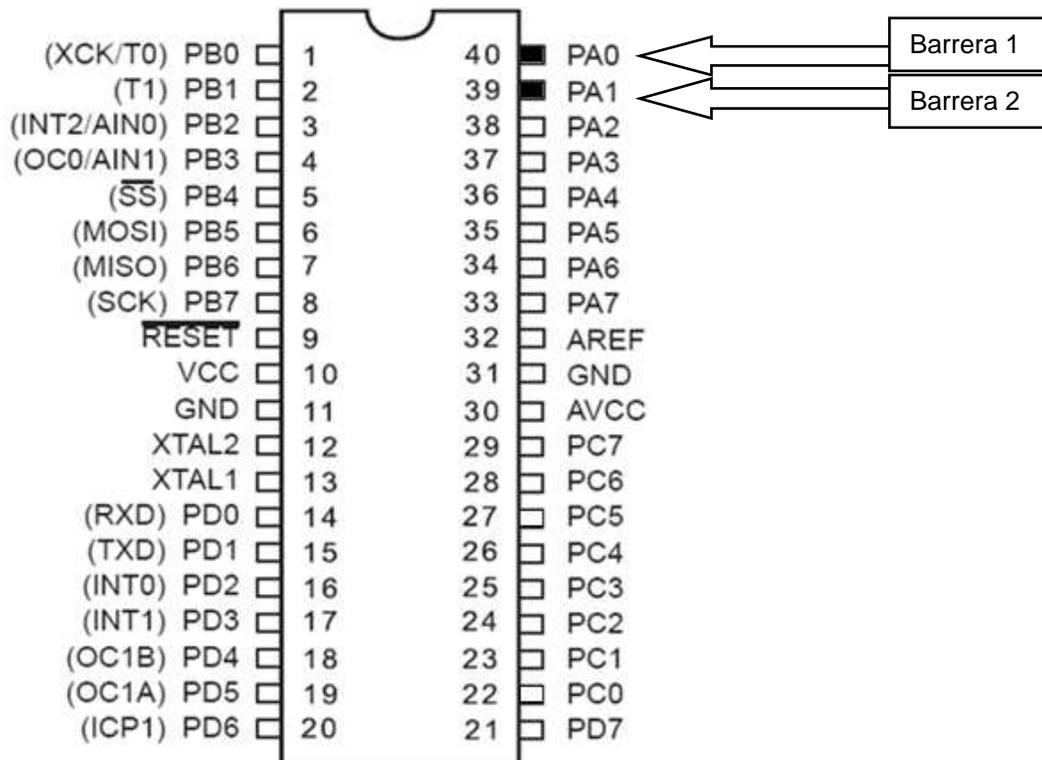


Figura 4.7 Diagrama de conexión de las barreras

- b) **Del microcontrolador al módulo de alarma.** La manera en que el microcontrolador activa y desactiva la central de alarma es mediante dos terminales de la alarma conectadas a los pines: PC0 para activar y PC5 para desactivar la alarma como lo muestra la Figura 4.8, ambos normalmente en uno lógico y a cero lógico para cumplir su función.

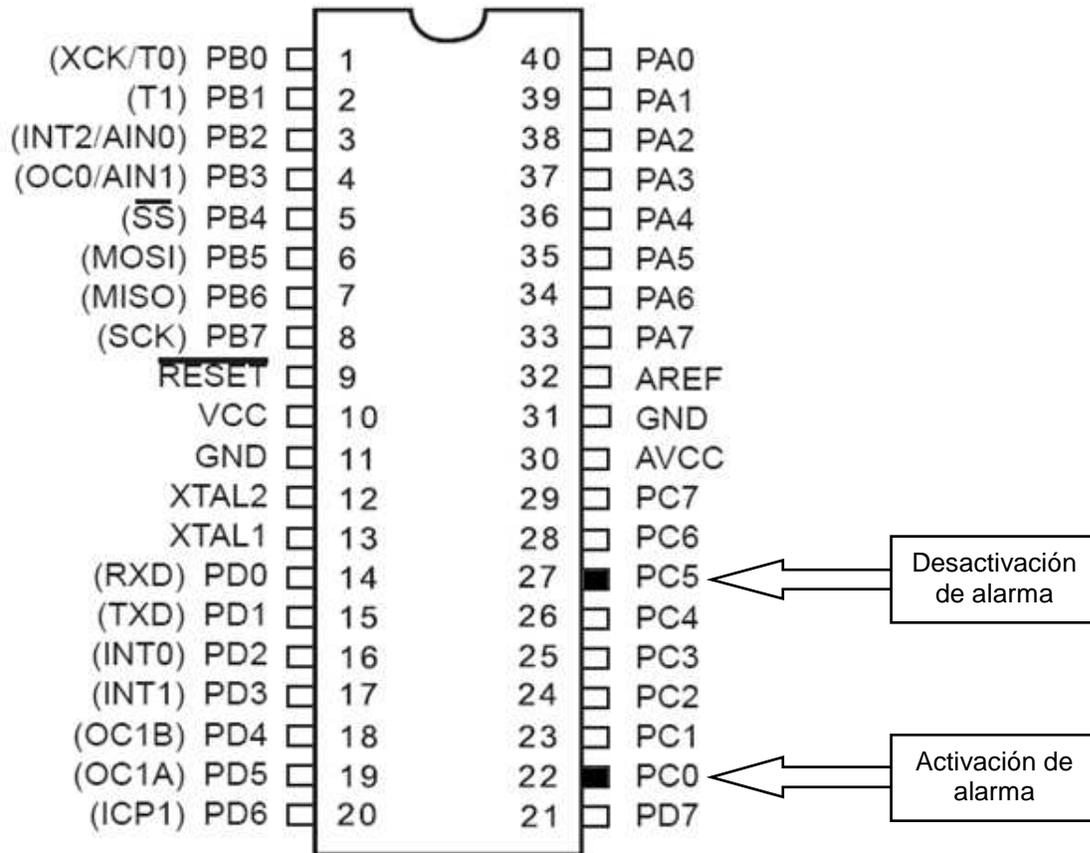


Figura 4.8 Diagrama de conexión de la alarma

- c) **Del Microcontrolador a la PC.** La comunicación del microcontrolador a la PC se lleva a cabo por medio del puerto serie. Para convertir los niveles de voltaje del puerto serie a los que emplea el microcontrolador y viceversa se utiliza el circuito MAX232 como se muestra en la Figura 4.9. El microcontrolador recibirá las instrucciones del puerto serie mediante los pines: RXD y TXD.

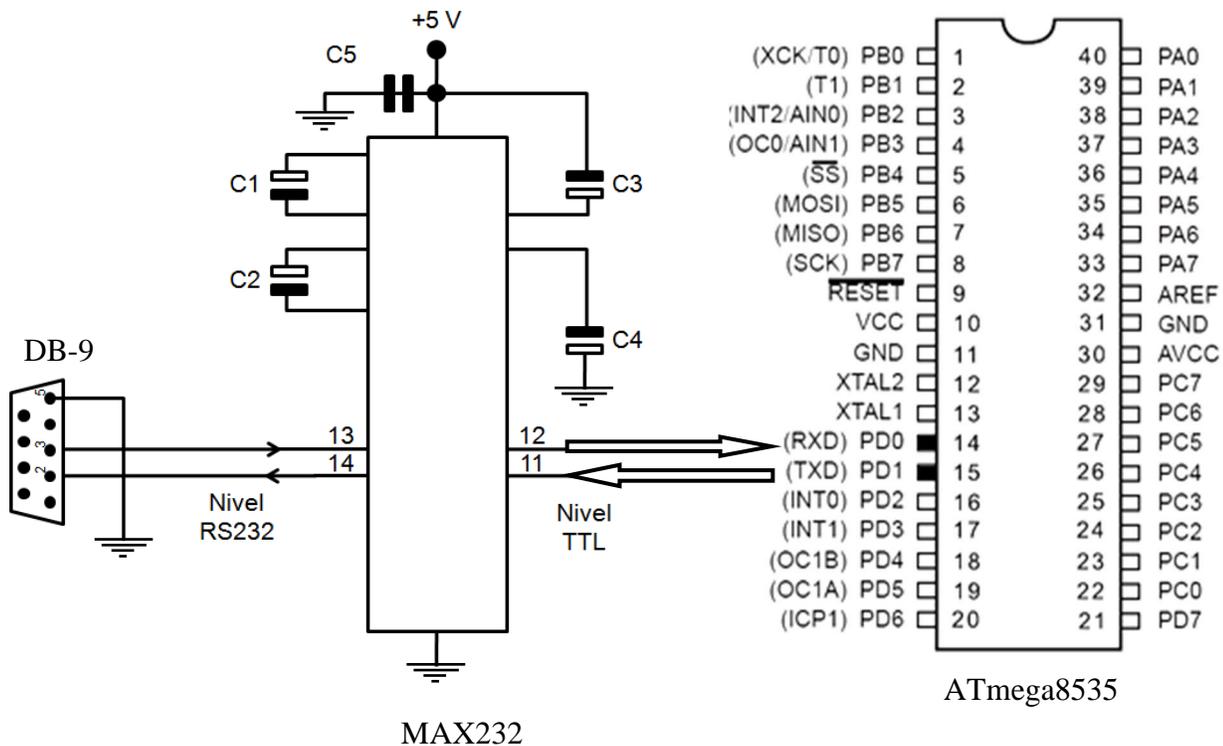


Figura 4.9 Diagrama de conexión MAX232

4.2.5.- Placa de circuito impreso (PCB) de interfaz

Para el presente trabajo se realizó el diseño de una placa PCB, en la cual se podrán interconectar todos los componentes y módulos antes mencionados. A continuación, se muestra el diseño de la placa antes mencionada elaborada mediante el *software Protel DXP*.

Como primera vista se ilustra en la Figura 4.10 la cara superior (*top layer*), en la cual se indican los principales componentes que hacen posible la interfaz, e indican los estados de las salidas.

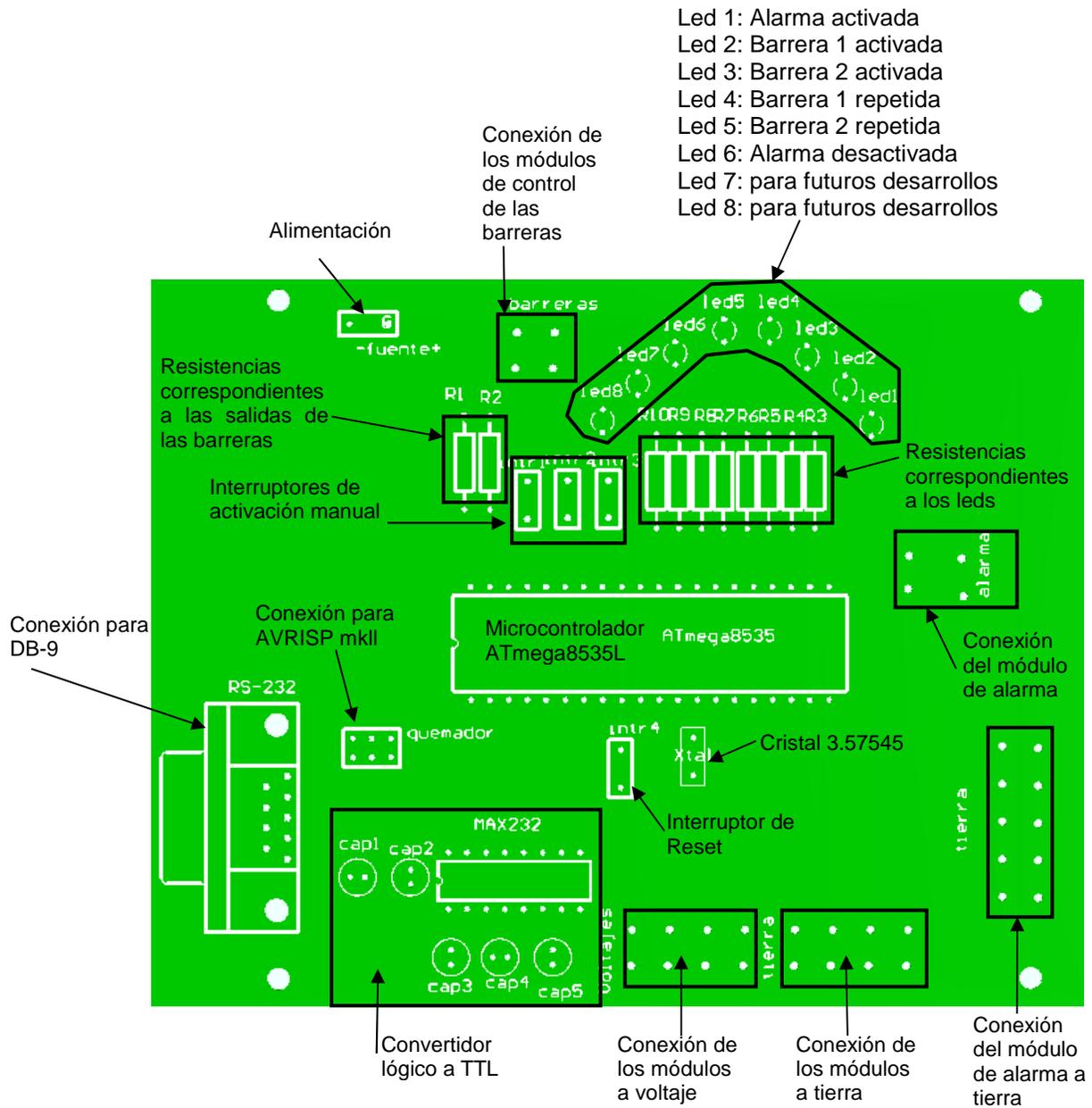


Figura 4.10 Placa PCB cara superior (*top layer*)

En la Figura 4.11 se visualiza la cara inferior (*bottom layer*) de la placa PCB, en la cual se observan las pistas que interconectan los componentes electrónicos ilustrados en la figura anterior.

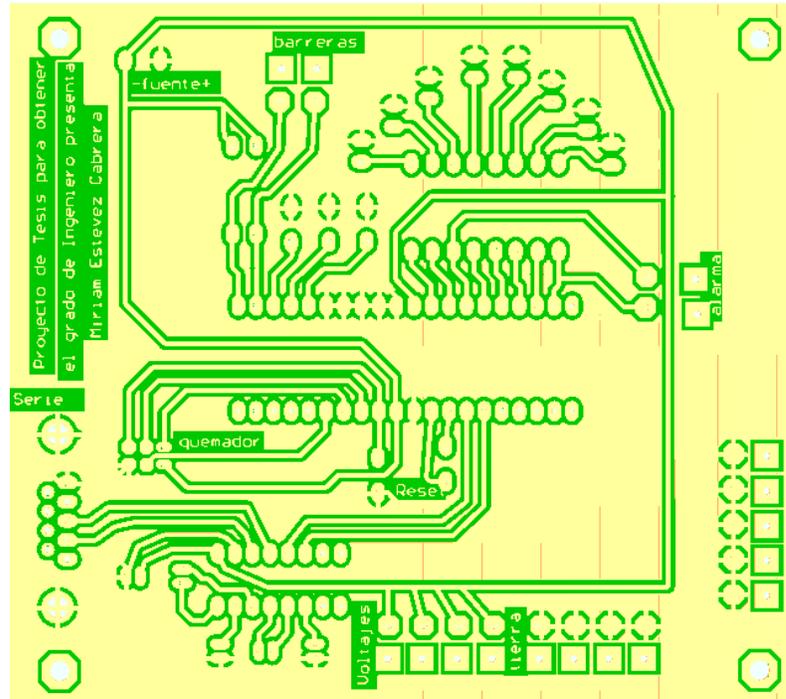


Figura 4.11 Placa PCB cara inferior (*bottom layer*)

Para garantizar una distribución uniforme de los componentes se presenta en la Figura 4.12 una visualización de la placa antes mencionada en tercera dimensión, donde se constata que los componentes electrónicos en cuanto a dimensiones físicas no se interfieren.

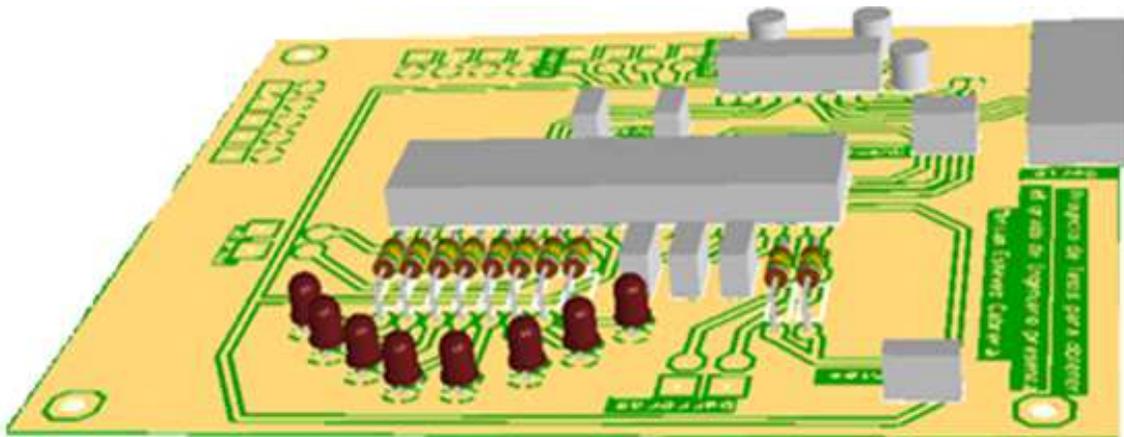


Figura 4.12 Placa PCB en tercera dimensión

Una vez que se comprobó que el diseño es el indicado, se realizó la implantación de los componentes físicos que se muestran en la siguiente Figura 4.13, indicando cada uno de los elementos que conforman la interfaz.

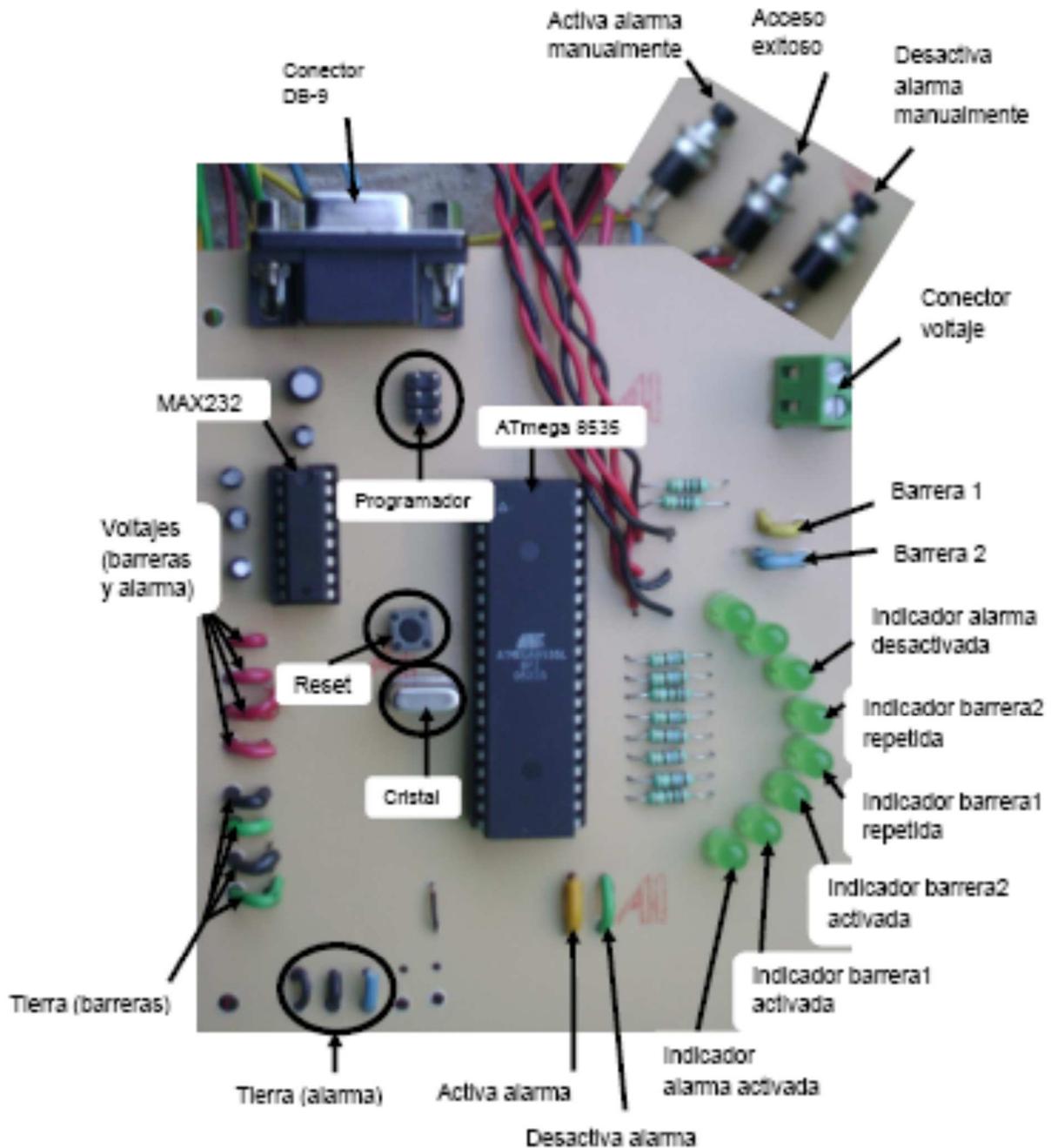


Figura 4.13 Placa PCB con componentes físicos

4.3.- Descripción del software

La arquitectura del *software* se encuentra dividida en dos partes:

- a) El *software* que se ejecuta desde la PC, el cual se encarga de llevar a cabo el programa de control de registro de acceso.
- b) El *firmware* localizado en el microcontrolador, que sirve como interfaz entre los dispositivos actuadores finales y la PC.

4.3.1.- Software de la PC

El *software ODBMS* de la terminal de identificación (PC) se realizó mediante la plataforma de *Visual Basic 6.0*, debido a su entorno gráfico y la fácil comunicación de datos seriales.

Las principales funciones de las cuales se encarga este programa son:

- a) Mantener el control de entradas y salidas de los usuarios mediante registros almacenados en una base de datos.
- b) Llevar a cabo la comunicación serial con el microcontrolador.
- c) Brindar instrucciones a los usuarios por medio del audio de la PC.

4.3.1.1- Declaración de funciones

Dentro del programa principal se utilizan funciones definidas por *Visual Basic 6.0* para realizar operaciones tales como: temporizadores, multimedia, puertos, etc.

La Figura 4.14 muestra el comando “timer”, con el cual se determina el tiempo que tiene el usuario para registrarse.

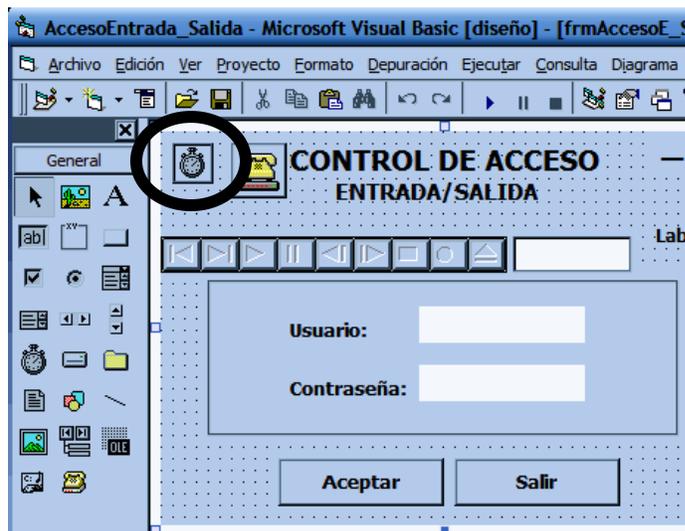


Figura 4.14 Comando timer

Dentro del comando *Private Sub Timer1_Timer()* se declaran las siguientes instrucciones:

Se establece la variable “I” como un contador incremental

```
I = I + 1
```

Por medio del siguiente comando se determina la velocidad con la que se incrementa el contador, con este arreglo se tienen minutos, segundos y milésimas de segundos. Éste es visualizado a través de la ventana de texto “*Text1.Text*”

```
Tiempo = Format(Int(I / 6000) Mod 60, "00") & ":" & _
```

```
Format(Int(I / 100) Mod 60, "00") & ":" & _
```

```
Format(I Mod 100, "00")
```

```
Text1.Text = Tiempo
```

Si el valor del tiempo mostrado en el “*Text1.Text*” es 25 segundos, que es el tiempo límite que tiene el usuario para registrarse entonces se envía a través del puerto serie el caracter “a” que el microcontrolador reconocerá como la activación de la alarma.

```
If Text1.Text = "00:25:00" Then
```

```
MSComm1.Output = a
```

Para poder hacer uso de los comandos de audio que se verán más adelante, es necesario agregar al formulario el comando multimedia “MMcontrol” como lo muestra la Figura 4.15.

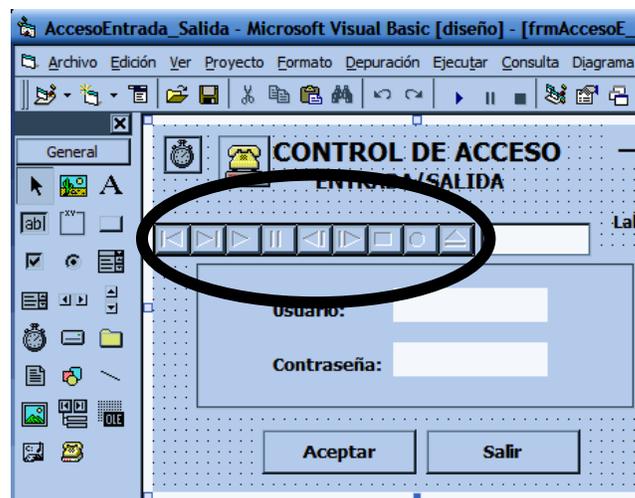


Figura 4.15 Comando multimedia

4.3.1.2.- Base de datos

Para manipular la base de datos es necesario hacer uso de la librería ADO que permite crear aplicaciones capaces de manipular bases de datos a través de un proveedor OLE DB (*Object Linking and Embedding for DataBase*). El objetivo de OLE DB es poner a disposición del programador una herramienta de nivel inferior que le de acceso universal a los datos con independencia del origen de datos. Debido a la complejidad de los elementos de OLE DB, no se puede acceder a ellos directamente desde Visual Basic; para ello se utilizan los objetos ADO que permiten acceder a la práctica totalidad de las funciones de OLE DB.

Una vez que se han establecido las librerías a utilizar, es necesario declarar las variables de tipo recordset "rs" y "rsHayMov", para trabajar con ADO.

Dim rs, rsHayMov As Recordset

Al entrar al área de registro y en la pantalla de la terminal de identificación se presione el botón "aceptar", lo primero que realizará el programa será verificar que los campos del usuario y contraseña no se encuentren vacíos, de ser así le indicará a la persona que complete los datos, una vez que los datos estén completos se accederá a la base de datos para comprobar que el usuario existe y su contraseña es correcta.

If txtUsuario.Text = "" And txtPasswd.Text = "" Then

MsgBox "Los campos: usuario y contraseña son obligatorios." & vbCrLf & vbCrLf & "Introduzca los datos correctos.", vbInformation, "Acceso"

End If

If txtUsuario.Text = "" Then

MsgBox "Escriba un nombre de usuario para iniciar sesión.", vbInformation, "Acceso"
txtUsuario.SetFocus

End If

If txtPasswd.Text = "" Then

MsgBox "El campo de contraseña está vacío, ingrese su contraseña e intente de nuevo.", vbInformation, "Acceso"

txtPasswd.SetFocus

End If

*SqlUsers = "SELECT * FROM TbUsuarios WHERE IdPersonal = " & txtUsuario.Text & "
and Contrase = " & txtPasswd.Text & ""*

Call OnRs(rs, SqlUsers)

*sqlAcceso = "SELECT * FROM TbAcceso WHERE IdPersonal = " & txtUsuario.Text & ""*

Call OnRs(rsAcceso, sqlAcceso)

If rsAcceso.BOF = True And rsAcceso.EOF = True Then

*MsgBox "El usuario y/o contraseña son incorrectos, por favor verifique e intente de
nuevo.", vbInformation, "Acceso"*

txtUsuario.Text = ""

txtPasswd.Text = ""

txtUsuario.SetFocus

End If

Una vez que sea comprobado que el usuario y contraseña son correctos y que se le permite el acceso a esta área, se realizará una consulta dentro de la base de datos en la tabla de movimiento (Tbmovimiento), en la cual se verificará si el usuario está entrando al área de trabajo o saliendo de esta, a continuación se muestra parte del código correspondiente a la entrada al área.

*sqlHayMov = "SELECT MovEntrada FROM TbMovimiento WHERE IdPersonal=" &
txtUsuario.Text & " And Fecha=" & Fecha & " And MovSalida = " & SALIDA & " " 'And
MovEntrada = " & ENTRADA & "*

Call OnRs(rsHayMov, sqlHayMov)

If rsHayMov.EOF = True And rsHayMov.BOF = True Then

'REGISTRA LA ENTRADA

*CONEXION.Execute "INSERT INTO TbMovimiento VALUES(" & txtUsuario.Text &
"',ENTRADA', " & HoraEntrada & ",",", " & Fecha & ",") "*

Nota: las dos líneas anteriores están encargadas de insertar en la tabla de movimiento los valores correspondientes a los campos de usuario, hora de entrada y fecha como se muestra en la Figura 4.16.

TbMovimiento						
IdPersonal	MovEntrada	HoraEntrada	MovSalida	HoraSalida	Fecha	TiempoSesion
001	ENTRADA	02:45:52 p.m.	SALIDA	02:45:59 p.m.	19/08/2008	0:0
001	ENTRADA	12:08:23 a.m.	SALIDA	12:08:50 a.m.	21/08/2008	0:0
001	ENTRADA	12:11:30 a.m.			21/08/2008	
002	ENTRADA	02:46:09 p.m.	SALIDA	02:49:11 p.m.	19/08/2008	0:3
*						

Figura 4.16 Tabla movimiento_entrada

MsgBox rsNombreUser.Fields("Nombre") & vbCrLf & "HORA ENTRADA: " & HoraEntrada & "", vbInformation, "Registro de ENTRADA"

Nota: el comando “MsgBox” indica al usuario que su acceso ha sido registrado exitosamente, como lo ilustra la Figura 4.17.



Figura 4.17 Registro entrada

```
MSComm1.Output = c 'envía registro exitoso al microcontrolador  
Timer1.Interval = 0 'detiene el contador encargado de activar la alarma  
txtUsuario.Text = ""  
txtPasswd.Text = ""  
txtUsuario.SetFocus  
HoraEntrada = Time
```

Cuando se presenta el caso en que el usuario está registrando su salida, *rsHayMov.EOF = False And rsHayMov.BOF = False* hará una consulta dentro de la base de datos en la tabla *TbMovimiento* para ingresar los valores de los campos faltantes para el usuario y fecha correspondientes.

```
Elseif rsHayMov.EOF = False And rsHayMov.BOF = False Then
    sqlHayMovSale = "SELECT * FROM TbMovimiento WHERE IdPersonal=" &
    txtUsuario.Text & " AND Fecha=" & Fecha & " AND MovEntrada='ENTRADA'"
    Call OnRs(rsHayMovSale, sqlHayMovSale)
    'REGISTRA SALIDA
```

HoraSalida = Time

MSComm1.Output = c 'registro exitoso

Timer1.Interval = 0

```
CONEXION.Execute "UPDATE TbMovimiento SET MovSalida='SALIDA',HoraSalida=" &
    HoraSalida & ",TiempoSesion=" & TotalTiempo & " where IdPersonal=" & txtUsuario.Text
    & " AND Fecha=" & Fecha & " "
```

Nota: las tres líneas anteriores están encargadas de insertar en la tabla de movimiento los valores correspondientes a los campos de hora de salida y tiempo total de sesión correspondientes al usuario y fecha correspondientes, como se muestra en la Figura 4.18.

TbMovimiento							
IdPersonal	MovEntrada	HoraEntrada	MovSalida	HoraSalida	Fecha	TiempoSesion	
001	ENTRADA	02:45:52 p.m.	SALIDA	02:45:59 p.m.	19/08/2008	0:0	
001	ENTRADA	12:08:23 a.m.	SALIDA	12:08:50 a.m.	21/08/2008	0:0	
001	ENTRADA	12:11:30 a.m.			21/08/2008		
001	ENTRADA	12:25:36 p.m.	SALIDA	12:45:26 p.m.	28/08/2008	0:20	
002	ENTRADA	02:46:09 p.m.	SALIDA	02:49:11 p.m.	19/08/2008	0:3	
*							

Figura 4.18 Tabla movimiento_salida

```
MsgBox rsNombreUser.Fields("Nombre") & vbCrLf & "HORA SALIDA: " & HoraSalida & "",
    vbInformation, "Registro de SALIDA"
```

Nota: el comando “MsgBox” indica al usuario que su salida ha sido registrada exitosamente, como lo ilustra la Figura 4.19.



Figura 4.19 Registro salida

```
txtUsuario.Text = ""  
txtPasswd.Text = ""  
txtUsuario.SetFocus
```

End If

4.3.1.3.- Comunicación serial

La comunicación serial dentro del entorno de *Visual Basic 6.0* se establece por medio del comando *MSComm* que debe ser agregado al formulario principal, Figura 4.20.



Figura 4.20 Comando MScomm

Una vez agregado este comando al formulario deben realizarse las siguientes configuraciones:

a) Dentro del Form_Load()

Habilitar el puerto serie

If MSComm1.PortOpen = False Then MSComm1.PortOpen = True

Determina la velocidad de transmisión en bits por segundos “9600”, el tipo de paridad ninguno “N”, los bits de datos a transmitir “8” y el bit de parada “1”.

MSComm1.Settings = "9600,N,8,1"

Se agregó el siguiente código que indica al usuario que el puerto serie no se pudo habilitar cuando así ocurra.

errores:

If (Err.Number = 8002) Then MsgBox "No se encontro el puerto", 48, "No se pudo conectar al puerto serie"

Se lee todo el contenido del buffer de recepción que posee el control MSComm.

MSComm1.InputLen = 0

Se determina que al recibir uno o más caracteres se generará el evento OnComm y la propiedad CommEvent contendrá el valor comEvReceive.

MSComm1.RThreshold = 1

Para enviar uno o más caracteres se generará el evento OnComm y la propiedad ComEvent contendrá el valor comEvSend

MSComm1.SThreshold = 1

b) Dentro de *Private Sub MSComm1_OnComm()*

Se establece que si recibe un caracter entonces éste se almacenará en la variable “rx”.

```
If MSComm1.CommEvent = comEvReceive Then  
rx = MSComm1.Input
```

Si el caracter almacenado en “rx” corresponde a la letra “a”, esto indica que la barrera 1 ha sido atravesada por lo que se inicia un temporizador que otorga 25 segundos al usuario para registrarse, y al mismo tiempo se escuchará por el altavoz de la PC “Favor de registrar su entrada”.

```
If rx = "a" Then 'barrera 1 atravesada  
    I = 0 'Inicializa el contador.  
    Timer1.Interval = 0 'Detiene el cronometro  
    Text1.Text = "" 'Limpia la etiqueta  
    Timer1.Interval = 1 'Inicia el cronometro  
    'vuelve a checar el puerto  
    MMControl1.FileName = "C:\entrada.wav"  
    MMControl1.Command = "close"  
    MMControl1.Command = "Open"  
    MMControl1.Command = "Play"
```

Si el caracter almacenado en “rx” corresponde a la letra “b”, indica que la barrera 2 ha sido atravesada por lo que se inicia un temporizador que otorga 25 segundos al usuario para registrarse, y al mismo tiempo se escuchará por el altavoz de la PC “Favor de registrar su salida”.

```
If rx = "b" Then 'barrera 2 atravesada  
    I = 0 'Inicializa el contador  
    Timer1.Interval = 0 'Detiene el cronometro  
    Text1.Text = "" 'Limpia la etiqueta
```

```
Timer1.Interval = 1 'Inicia el cronometro  
'vuelve a checar el puerto  
MMControl1.FileName = "C: \salida.wav"  
MMControl1.Command = "close"  
MMControl1.Command = "Open"  
MMControl1.Command = "Play"
```

Si el caracter almacenado en “rx” corresponde a la letra “c”, significa que alguna de las barreras es atravesada mientras que otra persona se encuentra en el área de registro, por lo que escuchara la orden “Por favor espere su turno”.

```
If rx = "c" Then  
MMControl1.FileName = "C: \espere.wav"  
MMControl1.Command = "close"  
MMControl1.Command = "Open"  
MMControl1.Command = "Play"
```

Si el caracter almacenado en “rx” corresponde a la letra “e”, significa que se llevó a cabo un registro exitoso, y la persona está saliendo del área de registro, se escuchará un sonido que indicará que la terminal de identificación esta lista para poder ser utilizada nuevamente.

```
If rx = "e" Then  
MMControl1.FileName = "C: \exito.wav"  
MMControl1.Command = "close"  
MMControl1.Command = "Open"  
MMControl1.Command = "Play"
```

Finalmente se agrego un *CommandButton* para que el administrador pueda hacer consultas de los registros de entrada y salida con respecto a fechas específicas. Para esto se hizo uso del control “*ActiveX actBar.ocx*” y “*arviewer.ocx*” agregándolos a la biblioteca

de referencia de *Visual Basic 6.0*. Para desplegar una hoja de reporte se emplea un *Data Dynamics ActiveReports 2.0*. Este diseño *ActiveX*, permite un enlace sencillo a la BDD de *Access* mediante el *DAO DataControl* como se ve en la siguiente Figura 4.21.

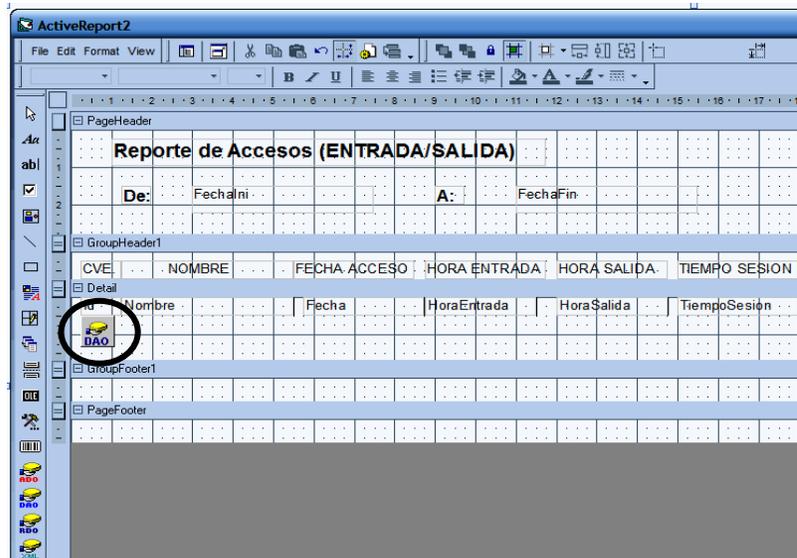


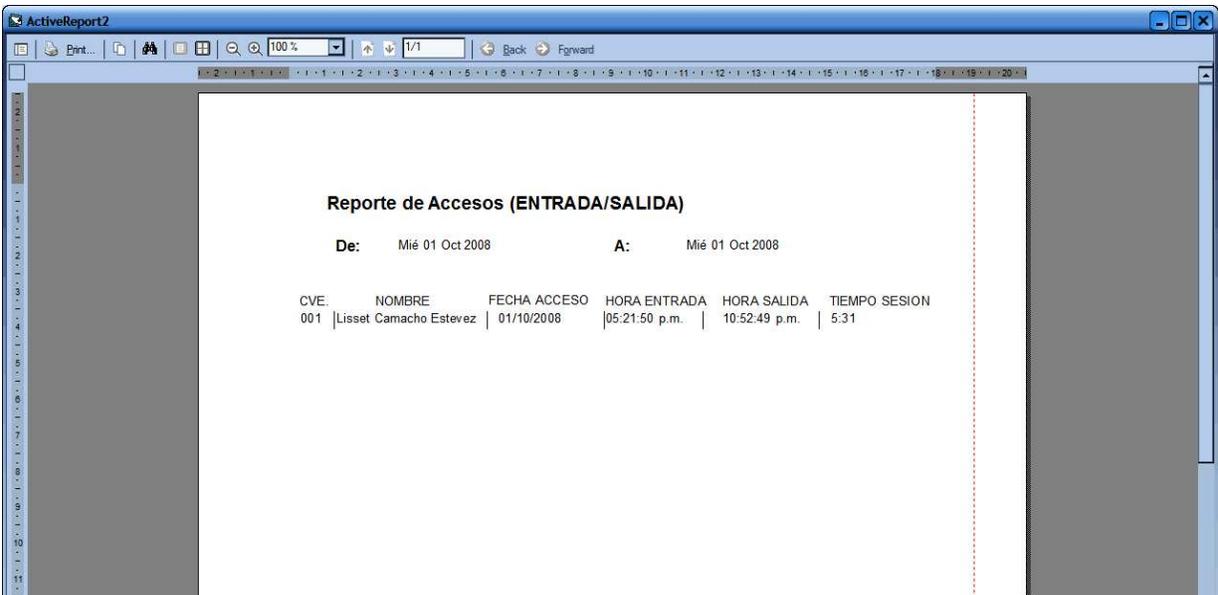
Figura 4.21 Comando DAO

Para evitar que cualquier usuario tenga acceso a estos reportes, se agrego un *InputBox* que permite solo al administrador que posee la contraseña, disponer de los reportes como lo muestra la Figura 4.22.



Figura 4.22 InputBox

Una vez que el administrador ha establecido el rango de fechas se despliega en pantalla una hoja de reporte cuyas características se pueden observar en la Figura 4.23.



The screenshot shows a web browser window titled "ActiveReport2". The main content is a report titled "Reporte de Accesos (ENTRADA/SALIDA)". The report details an access event for "Lisset Camacho Estevez" on "01/10/2008". The access occurred between "05:21:50 p.m." and "10:52:49 p.m.", with a session duration of "5:31".

CVE.	NOMBRE	FECHA ACCESO	HORA ENTRADA	HORA SALIDA	TIEMPO SESION
001	Lisset Camacho Estevez	01/10/2008	05:21:50 p.m.	10:52:49 p.m.	5:31

Figura 4.23 Active Report

4.3.2.- Firmware del microcontrolador

El programa contenido dentro del microcontrolador lleva a cabo dos objetivos principales: leer las señales binarias de entradas (barreras e interruptores) y salidas (alarma y LEDs indicadores), y comunicarle a la PC el estado de estas señales, así como recibir de la misma instrucción para la activación y desactivación de estos periféricos. A continuación se explicará brevemente las funciones necesarias para realizar los objetivos antes mencionados.

Como primer punto se hace uso de las librerías: "avr/io.h" y "avr/delay.h", para definir el uso de los registros de entradas - salidas y el uso de retardos de tiempo respectivamente.

El siguiente paso es la configuración de la USART (*Universal Synchronous and Asynchronous serial Receiver and Transmitter*) mediante la función "void usart_init(void)" que se describirá en seguida.

El registro UBRR es en realidad dos registros, UBRRH y UBRL, ambos de 8 bits. UBRRH guarda los 4 bits más significativos de UBRR, y UBRL guarda los 8 bits menos significativos de UBRR. Los 4 bits más significativos de UBRRH no se utilizan y en el *datasheet* se recomienda escribirlos como 0 para compatibilidad con dispositivos futuros.

$UBRRH = 0;$

La ecuación para el cálculo del valor de UBRR, tomando en cuenta que el modo de operación es asíncrono normal, es la siguiente:

$$UBRR = \left(\frac{3.57545MHz}{16 * 9600} \right) - 1 = 22.28$$

El resultado obtenido se redondea a 22, por lo tanto se utilizará el cristal de 3.57545 MHz propuesto en la ecuación anterior y el registro UBRRL adopta el siguiente valor:

$$UBRRL = 22;$$

Mediante el siguiente registro se activa la transmisión y recepción de datos.

$$UCSRB = (1 \ll RXEN) | (1 \ll TXEN);$$

El tamaño del carácter que se utiliza para la transmisión de información es de “8 bits” y se define de la siguiente forma:

$$UCSRC = (1 \ll URSEL) | (0 \ll USBS) | (3 \ll UCSZ0);$$

Una vez que se configura la comunicación serial, y dentro del programa principal se declarará al puerto “A” como entradas:

$$DDRA=0x00; //entradas$$

La Tabla 4.1 muestra la disposición de los pines del puerto A.

PIN	Periférico	Descripción
PA0	Barrera 1	Señal binaria de la barrera
PA1	Barrera 2	Señal binaria de la barrera
PA2	Alarma	Desactivación manual de la alarma
PA3	Señal de prueba	Acceso exitoso
PA4	Alarma	Activación manual de la alarma

Tabla 4.1 Entradas puerto A

Después de configurar el puerto A como entradas, se establece el puerto C como salidas.

DDRC=0xff; //salidas

En la Tabla 4.2 se describen las salidas a utilizar y su ubicación en el microcontrolador.

PIN	Periférico	Descripción
PC0	Alarma	Activación de la alarma
PC1	LED	LED indicador del estado lógico de la barrera 1
PC2	LED	LED indicador del estado lógico de la barrera 2
PC3	LED	LED indicador del estado lógico de la barrera 1 (cuando el sistema está ocupado)
PC4	LED	LED indicador del estado lógico de la barrera 2 (cuando el sistema está ocupado)
PC5	Alarma	Desactivación de la alarma

Tabla 4.2 Salidas puerto C

Se declaran las variables de tipo caracter sin signo cuyo tamaño es de 8 bits (un registro) que se utilizan en el programa.

```
unsigned char tempo=0;  
unsigned char i=0, j=0, ent, sal;  
unsigned char data;
```

Donde “j” es la variable asignada a un contador, que permite saber si alguien ya atravesó cualquier barrera y se reestablece cuando el usuario se registra exitosamente y sale del área.

Los datos que llegan del puerto A del microcontrolador son guardados en la variable “ent” y las salidas representadas en el puerto C se almacenan en la variable “sal”.

Una vez que se han declarado las variables, funciones y los puertos a utilizar, se comenzará un ciclo iterativo con las siguientes instrucciones:

- a) Se analiza si se ha recibido un caracter por el puerto serie, si es así éste caracter será guardado en la variable “data” y se comparará con los valores esperados para realizar una acción específica.

```
tempo=UCSRA;  
if((tempo & (1 << RXC))==(1 << RXC))  
data = UDR;
```

Si el caracter es igual al valor en decimal “97” indica que ha recibido el caracter “a” del código ASCII lo que corresponde a la acción de activar la alarma.

```
if(data==97)  
{  
PORTC=0xff; //limpia puerto de salidas  
sal=0x01; //señal que activa la alarma  
PORTC=~sal; //muestra la señal a través del puerto C  
}
```

Si el valor de data es igual a “98”, es decir “b” se desactiva la alarma con una estructura similar a la activación de la misma, esto es, solo cambia el bit con que es activada o desactivada.

```
else if(data==98)
{
PORTC=0xff;
sal=0x20;
PORTC=~sal;
PORTC=0xff;
}
```

Si el caracter recibido es una “c” (data=99), indica que el usuario se registro exitosamente, por lo que mediante el contador interno “j” no realizará ninguna operación la próxima vez que se active una barrera, puesto que esto significa que el usuario identificado está saliendo del área de registro.

```
else if(data==99)
{
sal=0x00;
PORTC=~sal;
j=2;
}
```

Nota: las tres funciones anteriores las cuales se ejecutan a partir de una instrucción del puerto serie, se repiten dentro de un “for” como se muestra a continuación, para evitar la malinterpretación de caracteres, esto no influye en los procesos que cumple cada una de estas funciones.

```
for(i=0;i<10;i++)
```

- b) Una vez que se ha analizado si se ha recibido un caracter por el puerto serie se leen las entradas del puerto A y se almacenan en la variable “ent” para ser procesadas y realizar las operaciones correspondientes.

ent=PIN A;

Las operaciones que realizará el microcontrolador corresponden a un bit de entrada, es necesario limpiar el registro y sólo analizar el bit requerido.

Si el bit de entrada PA0 es igual a '1' significa que la barrera 1 ha sido atravesada, para evitar los arcos voltaicos de activación, provenientes de las entradas analógicas, se mantendrá enclavada la lectura de este bit, hasta que sea diferente de '1', y para evitar los arcos voltaicos de desactivación se hace una pequeña pausa.

```
if((ent&0x01)==0x01) //barrera1 activa  
{  
while((ent&0x01)==0x01)ent=PIN A;  
_delay_ms(100);
```

Cuando la barrera 1 es atravesada pueden existir varias circunstancias que están definidas por el contador "j", si "j" es igual a cero, significa que el usuario está entrando al área de registro, por lo que se activa el LED indicador conectado a PC1 y al mismo tiempo envía a la PC un carácter "a" para que la terminal de identificación le pida registrarse.

```
if(j==0) // introducir clave en PC  
{  
j=1;  
sal=0x02;  
PORTC=~sal;  
if(!(UCSRA & (1 << UDRE))) ; //envío de "a" a la PC  
UDR = 97;  
}
```

Si "j" es igual a uno, valor que es asignado a "j" cuando existe un usuario registrándose, se activa el LED indicador en PC3 y se envía a la PC la instrucción (carácter "c") correspondiente al caso en que otro usuario intenta ingresar.

```
else if (j==1) //por favor espere su turno
{
sal=0x08;
PORTC=~sal;
if(!(UCSRA & (1 << UDRE))) ; //envío de c a la PC
UDR = 99;
}
```

Finalmente, si “j” es igual a dos, caso que se presenta cuando alguien se registró exitosamente, el contador se reestablece y envía la señal que representa la salida del área de registro (caracter “e”).

```
else if (j==2) //salida del área de registro
{
j=0;
if(!(UCSRA & (1 << UDRE))) ; //envío de “e” a la PC
UDR = 101;
}
}
```

Cuando la barrera 2 es atravesada, de igual manera que en el caso de la barrera 1, pueden existir varias circunstancias definidas por el contador “j”, si “j” es igual a cero, significa que el usuario está entrando al área de registro para salir del área de trabajo, por lo que se activa el LED indicador conectado a PC2 y al mismo tiempo envía a la PC un caracter “b” para que la terminal de identificación le pida registrar su salida.

```
else if((ent&0x02)==0x02) //barrera2 atravesada
{
while((ent&0x02)==0x02)ent=PINA;
_delay_ms(100);
if(j==0) // introducir clave en PC
```

```
{  
j=1;  
sal=0x04;  
PORTC=~sal;  
if(!(UCSRA & (1 << UDRE))) ; //envío de “b” a la PC  
UDR = 98;  
}
```

Si “j” es igual a uno, valor que es asignado a “j” cuando existe un usuario registrándose, se activa el LED indicador en PC4 y se envía a la PC la instrucción (carácter “c”) correspondiente al caso en que otro usuario intenta ingresar.

```
else if (j==1) //por favor espere su turno  
{  
sal=0x10;  
PORTC=~sal;  
if(!(UCSRA & (1 << UDRE))) ; //envío de “c” a la PC  
UDR = 99;  
}
```

De igual manera que para la barrera 1, si “j” es igual a dos, caso que se presenta cuando alguien se registró exitosamente, el contador se restablece y envía la señal que representa la salida del área de registro (carácter “e”).

```
else if (j==2) //salida del área de registro  
{  
j=0;  
if(!(UCSRA & (1 << UDRE))) ; //envío de “e” a la PC  
UDR = 101;  
}  
}
```

Adicionalmente a que la alarma es desactivada desde la PC, puede ser desactivada manualmente por medio de un interruptor localizado en el PA3, esta función es realizada mediante el siguiente código y bajo la estructura que se ha estado utilizando.

```
else if((ent&0x04)==0x04) //alarma desactivada manualmente
{
while((ent&0x04)==0x04)ent=PINA;
_delay_ms(100);
sal=0x20;
PORTC=~sal;
}
```

De igual manera se puede simular un acceso exitoso desde un interruptor ubicado en el PA4, cuyo código se muestra a continuación.

```
else if((ent&0x08)==0x08) //acceso exitoso desde interruptor
{
while((ent&0x08)==0x08)ent=PINA;
_delay_ms(100);
sal=0x00;
PORTC=~sal;
j=2;
}
```

La alarma puede ser activada manualmente por el interruptor ubicado en el PA5, como lo muestra la siguiente estructura.

```
else if((ent&0x10)==0x10) //alarma activada manualmente
{
while((ent&0x10)==0x10)ent=PINA;
_delay_ms(100);
sal=0x01;
PORTC=~sal;    }
```

CAPÍTULO V: CONCLUSIONES

5.1.- Pruebas y Resultados

El primer resultado que se puede considerar en el presente proyecto, es la obtención del propio dispositivo físico. Este constaba de varias partes, como son: la placa de interfaz, los módulos de barreras infrarrojas y módulo de alarma.

Se describe, por tanto, sobre fotografías, las distintas partes del dispositivo, y los componentes que las forman. No se pretende con esto volver a explicar en qué consisten los distintos elementos de circuitería, sino que se busca ofrecer al lector una imagen del dispositivo final.

La placa PCB de interfaz empleada se ilustra en la Figura 5.1 y Figura 5.2.

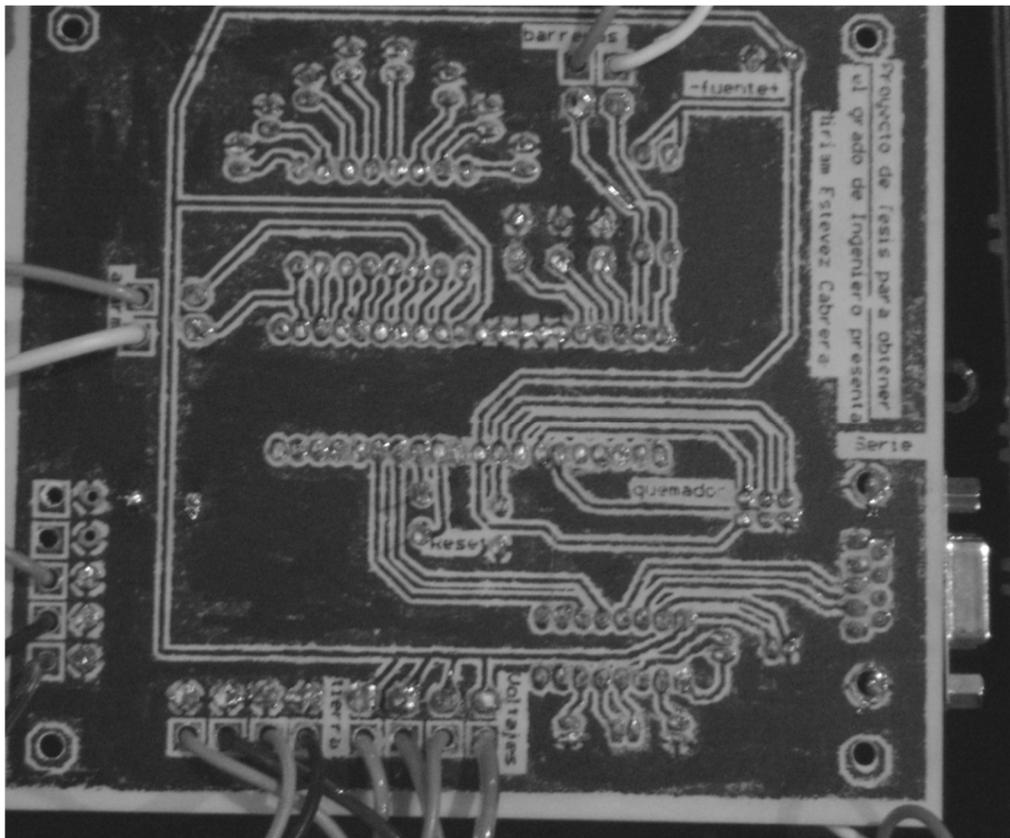


Figura 5.1 Placa PCB final (bottom layer)

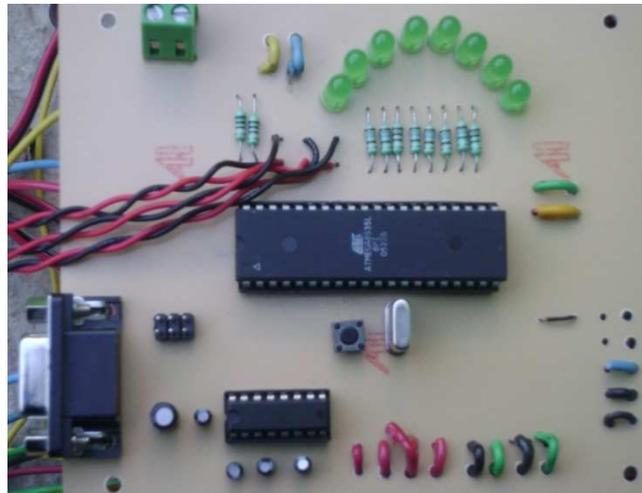


Figura 5.2 Placa PCB final (top layer)

Finalmente se integran todos los módulos como lo muestra la Figura 5.3.

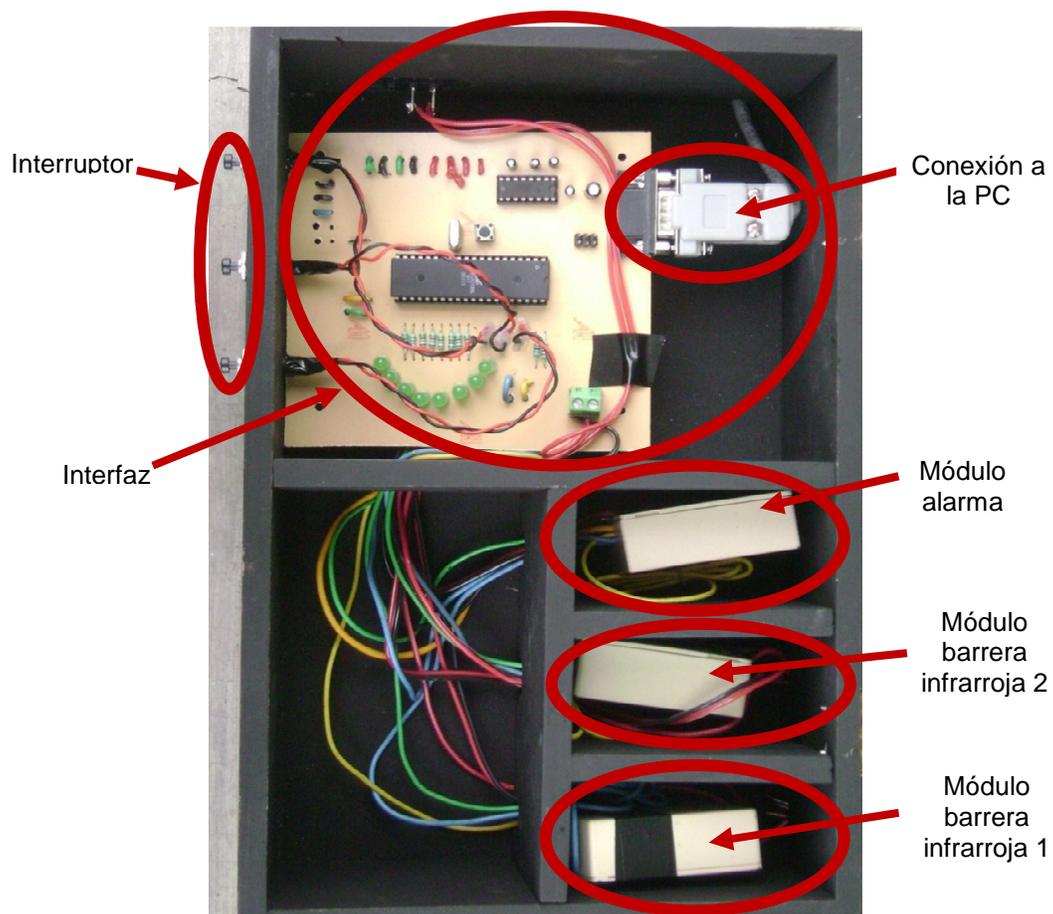


Figura 5.3 Interconexión física de los módulos

Las primeras medidas a obtener como resultado en cualquier dispositivo electrónico, son aquellas que ofrecen al usuario una respuesta sobre el correcto funcionamiento del objeto diseñado. En este caso, las comprobaciones que se han llevado a cabo son principalmente de cuatro tipos:

- Comprobación de que las soldaduras de los componentes están bien realizadas.
- Comprobación del correcto funcionamiento de la conexión entre la PC y el microcontrolador. Esta comprobación se efectúa mediante un programa de envío y recepción de texto elaborado en *Visual Basic 6.0*.
- Comprobación del diseño del código de programación, verificándose mediante el encendido de los LEDs indicadores de las distintas instrucciones de salidas digitales.
- Comprobación del funcionamiento final del dispositivo, mediante pruebas de ensayo y error.

Respecto a los resultados del correcto funcionamiento del dispositivo, se establece la siguiente conclusión:

- En el diseño de la placa se facilita la posibilidad de reprogramar en su totalidad el microcontrolador, lo que permite corregir errores de programación o agregar nuevos códigos, lo que proporciona una optimización del firmware.

Una vez que se comprobó el correcto funcionamiento de la interconexión de los módulos, se llevó a cabo la implementación del sistema en el área de almacén.

El sistema fue colocado conforme a la propuesta de diseño de la figura 2.2. Para instalar las barreras infrarrojas, se hicieron barrenos en los muros, para que los sensores infrarrojos estuvieran lo más discreto posible y su cableado oculto de la vista del usuario; para lograr su correcta alineación, al hacer el primer barreno del transmisor, en éste se colocó un indicador láser para señalar la mejor posición para colocar el receptor, ya que

en las pruebas realizadas a escala la falla más usual era la alineación incorrecta de los sensores, lo que provocaba falsos registros.

La PC se colocó del lado derecho en la parte central del área de registro, sólo el monitor y el teclado se encuentran a la vista, ya que se aprovechó y acondiciono una ventana que se encuentra en esta pared, que funcionaba anteriormente como caseta de entrega de material y que actualmente estaba en desuso, las bocinas de la PC se colocaron a la entrada y salida del área de registro, de tal forma que el usuario al atravesar una barrera escuchará las instrucciones de registro.

Cuando el sistema fue instalado correctamente, el siguiente paso fue comprobar su funcionamiento, punto en el cual se determinó que el tiempo propuesto inicialmente de 15 segundos, no era el más conveniente para los usuarios, debido a la falta de práctica en el uso de la PC; errores al ingresar sus datos; indisponibilidad al instante del uso de las manos por transportar materiales industriales, etc. Se llegó a la conclusión de otorgar a los usuarios mayor tiempo para registrarse, estableciendo un máximo de 25 segundos.

Como se mencionó anteriormente el sistema funciona con 6 Volts, en caso de una falla en el suministro de energía eléctrica, tanto la fuente que alimenta el sistema como la PC, pueden ser conectadas a un *no-break* para garantizar el suministro de energía.

La Figura 5.4 muestra el plano con el cableado correspondiente a la instalación del sistema de registro en el área de almacén.

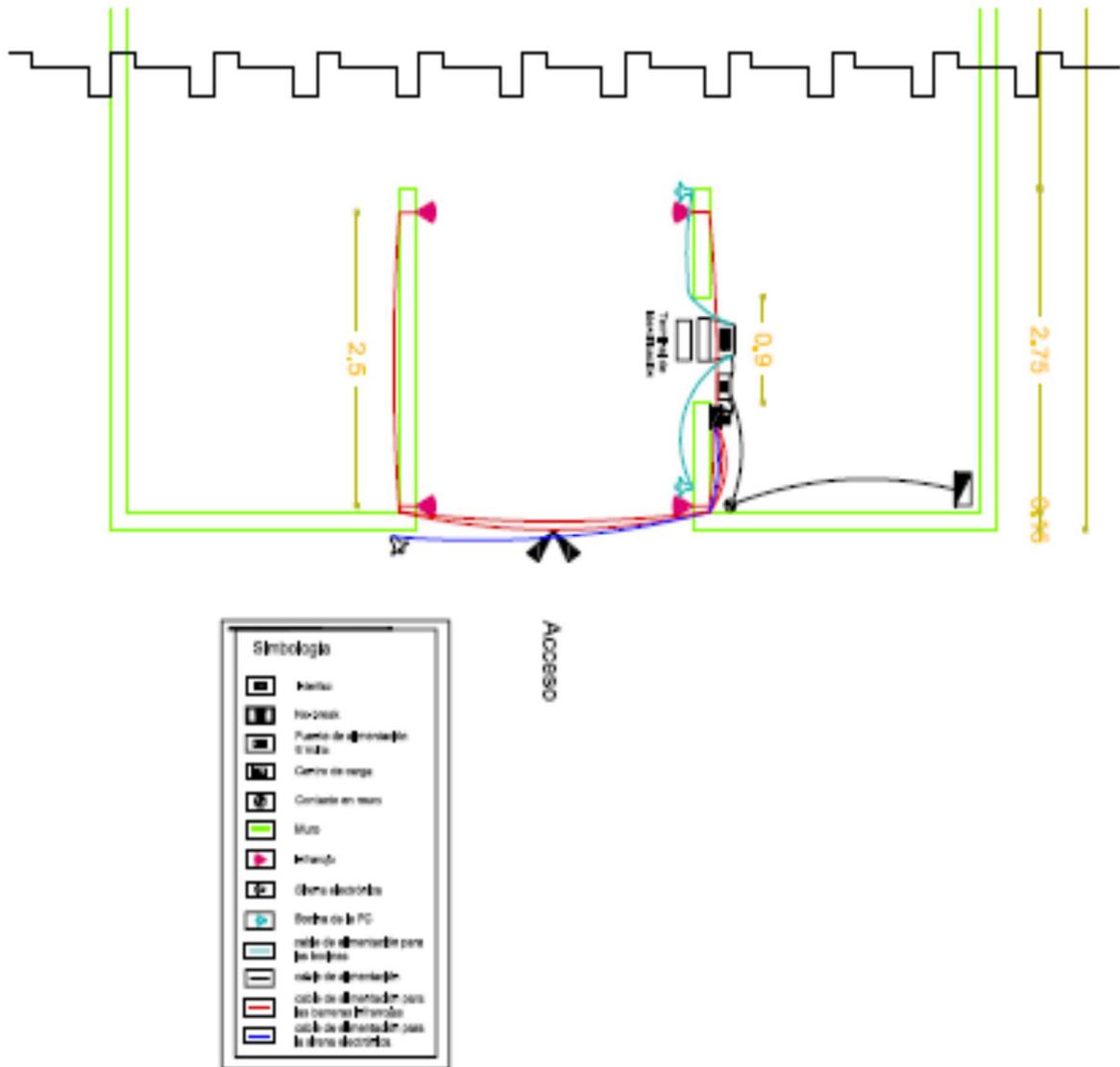


Figura 5.4 Plano del área de registro

5.2.- Estudio económico

El sistema desarrollado en este proyecto es un prototipo para aplicación concreta en el área de almacén de la empresa “Maquinados e Inyecciones Automotrices”. Debido a esto, los objetivos primordiales incluyeron hacer uso de una PC obsoleta para la empresa; partiendo de esto se realizó una terminal de identificación mediante un software (*ODBMS*) en *Visual Basic 6.0*, lo cual no generó ningún costo.

En la siguiente Tabla se desglosan los costos de los diversos módulos y materiales para la elaboración de este sistema.

Cantidad	Costo Unitario (\$ M.N.)	Descripción
2	270.00	Barrera infrarroja
1	285.00	Módulo de alarma
1	30.00	Zumbador
1	85.00	Microcontrolador ATmega8535L
1	10.00	Cto. MAX232
1	18.00	Placa fenólica
1	24.00	Conector DB-9
1	120.00	Cable RS-232
Varios	50.00	Resistencias, capacitores, LEDs, interruptores, etc.
Costo Total \$892.00		

Tabla 5.1 Costos

El costo neto es aproximado dado que el gasto real no ha sido elevado por los siguientes motivos:

- El sueldo de un ingeniero no se menciona porque es nulo, por ser este un proyecto de titulación llevado a cabo por quien escribe.
- La reutilización de la PC.
- El uso de una fuente de alimentación de 6V existente.

- La utilización de *software* libre como el *AVR studio versión 4.14*

El sistema además de llevar a cabo un control de registros de entradas y salidas, proporciona seguridad al contener un módulo de alarma. Esta es una razón más para que este proyecto sea factible de implementar en la empresa, por lo tanto se considera justificable la inversión.

5.3.- Futuros desarrollos

Este proyecto presenta un sistema con un vasto horizonte de mejoras. Esto es debido a que, dada su naturaleza experimental y añadiendo el hecho de que no tiene un objetivo comercial que limite sus posibilidades, dicho sistema podrá ser partícipe de modificaciones, mejoras y nuevos aditamentos.

Por otra parte, y analizando los objetivos planteados en el capítulo I y el entorno hacia el que va dirigido el sistema, habría modificaciones o mejoras a un primer nivel, o quizás más inmediatas, cuya complejidad no reporta excesivos recursos de tiempo ni dinero, y que podrían abarcarse en un futuro próximo. Tales podrían ser divididas principalmente en dos áreas:

- a) Dentro de la parte del sistema de software se podrán implementar las siguientes aplicaciones:
 - Hacer uso de una red de área local (LAN), para informar al supervisor general los movimientos que se llevan a cabo a través del área de registro en tiempo real.
 - Actualizar el *software* (*ODBMS*) para que el supervisor general pueda dar de alta, baja y modificar el estado de los usuarios, así como acceder a reportes generales, sin la necesidad de hacer uso de otro *RDBMS* (*Relational Data Base Management System*).
 - Captura de imágenes al ingresar al área de registro mediante cámaras web o de seguridad.

b) En este trabajo se presenta el diseño de una placa, la cual cumple con los objetivos del sistema, sin embargo y pensando en desarrollos futuros cuenta con la ventaja de poder reprogramar el microcontrolador en la misma placa de aplicación.

Por otra parte, se podrán sustituir las entradas tales como los interruptores por otros dispositivos de seguridad, y tomando en cuenta los LEDs indicadores de estado, se tendrán hasta ocho señales que pueden ser usadas como salidas.

Finalmente y a largo plazo se podrá hacer uso de terminales de identificación biométricas, aunque el uso de la terminal biométrica de reconocimiento dactilar no es la mejor opción por lo mencionado en el capítulo I; se podrá indagar en tecnologías de identificación facial, reconocimiento de voz, iris, por mencionar algunas.

5.4.- Conclusión

En este proyecto, se ha llevado a cabo el diseño, montaje, implementación, programación, desarrollo, ensayo y comprobación de un sistema capaz de controlar el registro de acceso a un área restringida de almacén, por medio de sensores infrarrojos y una interfaz con la PC, permitiendo realizar un registro permanente de las entradas y salidas a dicha área.

Se realizó tanto el software y hardware del sistema, ambas partes han sido desarrolladas satisfactoriamente, cumpliéndose los objetivos propuestos inicialmente para el proyecto. El uso de módulos para la elaboración de este proyecto permitió organizar y simplificar las tareas necesarias dentro del sistema y facilitar expansiones y mejoras en un futuro.

Al realizar pruebas de campo quedo comprobado que el sistema se desempeña según lo planeado, vale la pena resaltar que al implantar el sistema, fue importante alinear correctamente el receptor con el transmisor infrarrojo de las barreras, ya que esto permite evitar fallas en el sistema al momento de ingresar y/o salir del área de registro y de igual manera, se determino un tiempo adecuado que tendrán los usuarios para registrarse.

Dada la naturaleza del sistema y los constantes avances en la electrónica se podrá profundizar aun mas en las posibilidades que se plantean, en un futuro.

APÉNDICES

APÉNDICE A: Características específicas del microcontrolador ATmega8535L y MAX232

Features

- High-performance, Low-power AVR[®] 8-bit Microcontroller
- Advanced RISC Architecture
 - 130 Powerful Instructions – Most Single Clock Cycle Execution
 - 32 x 8 General Purpose Working Registers
 - Fully Static Operation
 - Up to 16 MIPS Throughput at 16 MHz
 - On-chip 2-cycle Multiplier
- Nonvolatile Program and Data Memories
 - 8K Bytes of In-System Self-Programmable Flash
 - Endurance: 10,000 Write/Erase Cycles
 - Optional Boot Code Section with Independent Lock Bits
 - In-System Programming by On-chip Boot Program
 - True Read-While-Write Operation
 - 512 Bytes EEPROM
 - Endurance: 100,000 Write/Erase Cycles
 - 512 Bytes Internal SRAM
 - Programming Lock for Software Security
- Peripheral Features
 - Two 8-bit Timer/Counters with Separate Prescalers and Compare Modes
 - One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode
 - Real Time Counter with Separate Oscillator
 - Four PWM Channels
 - 8-channel, 10-bit ADC
 - 8 Single-ended Channels
 - 7 Differential Channels for TQFP Package Only
 - 2 Differential Channels with Programmable Gain at 1x, 10x, or 200x for TQFP Package Only
 - Byte-oriented Two-wire Serial Interface
 - Programmable Serial USART
 - Master/Slave SPI Serial Interface
 - Programmable Watchdog Timer with Separate On-chip Oscillator
 - On-chip Analog Comparator
- Special Microcontroller Features
 - Power-on Reset and Programmable Brown-out Detection
 - Internal Calibrated RC Oscillator
 - External and Internal Interrupt Sources
 - Six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby and Extended Standby
- I/O and Packages
 - 32 Programmable I/O Lines
 - 40-pin PDIP, 44-lead TQFP, 44-lead PLCC, and 44-pad QFN/MLF
- Operating Voltages
 - 2.7 - 5.5V for ATmega8535L
 - 4.5 - 5.5V for ATmega8535
- Speed Grades
 - 0 - 8 MHz for ATmega8535L
 - 0 - 16 MHz for ATmega8535



8-bit AVR[®]
Microcontroller
with 8K Bytes
In-System
Programmable
Flash

ATmega8535
ATmega8535L

2502K-AVR-10/06

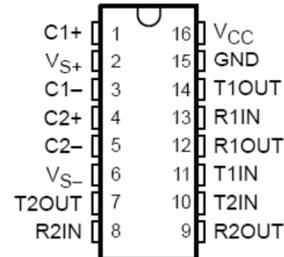


MAX232, MAX232I DUAL EIA-232 DRIVERS/RECEIVERS

SLLS0471 – FEBRUARY 1989 – REVISED OCTOBER 2002

- Meet or Exceed TIA/EIA-232-F and ITU Recommendation V.28
- Operate With Single 5-V Power Supply
- Operate Up to 120 kbit/s
- Two Drivers and Two Receivers
- ± 30 -V Input Levels
- Low Supply Current . . . 8 mA Typical
- Designed to be Interchangeable With Maxim MAX232
- ESD Protection Exceeds JESD 22 – 2000-V Human-Body Model (A114-A)
- Applications
 - TIA/EIA-232-F
 - Battery-Powered Systems
 - Terminals
 - Modems
 - Computers

MAX232 . . . D, DW, N, OR NS PACKAGE
 MAX232I . . . D, DW, OR N PACKAGE
 (TOP VIEW)



description/ordering information

The MAX232 is a dual driver/receiver that includes a capacitive voltage generator to supply EIA-232 voltage levels from a single 5-V supply. Each receiver converts EIA-232 inputs to 5-V TTL/CMOS levels. These receivers have a typical threshold of 1.3 V and a typical hysteresis of 0.5 V, and can accept ± 30 -V inputs. Each driver converts TTL/CMOS input levels into EIA-232 levels. The driver, receiver, and voltage-generator functions are available as cells in the Texas Instruments LinASIC™ library.

ORDERING INFORMATION

T _A	PACKAGE†		ORDERABLE PART NUMBER	TOP-SIDE MARKING
0°C to 70°C	PDIP (N)	Tube	MAX232N	MAX232N
	SOIC (D)	Tube	MAX232D	MAX232
		Tape and reel	MAX232DR	
	SOIC (DW)	Tube	MAX232DW	MAX232
		Tape and reel	MAX232DWR	
	SOP (NS)	Tape and reel	MAX232NSR	MAX232
-40°C to 85°C	PDIP (N)	Tube	MAX232IN	MAX232IN
	SOIC (D)	Tube	MAX232ID	MAX232I
		Tape and reel	MAX232IDR	
	SOIC (DW)	Tube	MAX232IDW	MAX232I
		Tape and reel	MAX232IDWR	

† Package drawings, standard packing quantities, thermal data, symbolization, and PCB design guidelines are available at www.ti.com/sc/package.



LinASIC is a trademark of Texas Instruments.

Please be aware that an important notice concerning availability, standard warranty, and use in critical applications of Texas Instruments semiconductor products and disclaimers thereto appears at the end of this data sheet.

PRODUCTION DATA Information is current as of publication date. Products conform to specifications per the terms of Texas Instruments standard warranty. Production processing does not necessarily include testing of all parameters.



POST OFFICE BOX 655303 • DALLAS, TEXAS 75265

Copyright © 2002, Texas Instruments Incorporated

APÉNDICE B: Fuente de alimentación

Diagrama electrónico de la fuente que suministra 6 volts de corriente continua, para alimentar el sistema de control de acceso.

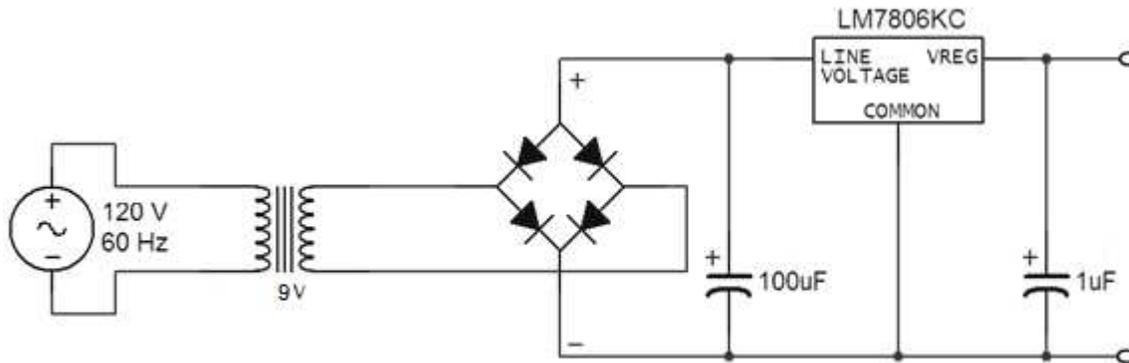


Figura. Fuente de alimentación

APÉNDICE C: Herramienta de Programación AVRISP mkII

La herramienta empleada para programar el microcontrolador ATmega8535L es la interfaz ISP de ATMEL Corporation “AVRISP mkII”.



Características:

- Compatible con *AVR studio* versión 4.12.
- Soporta microcontroladores AVR con interfaz ISP.
- Programa en memoria flash y EEPROM.
- Soporta programación de *fuses* y *lock bit*.
- Soporta voltajes de 1.8 a 5.5 V.
- Velocidad de programación ajustable (50Hz a 8MHz frecuencia SCK).
- Conector USB 2.0 (alta velocidad, 12 Mbps)
- No requiere alimentación externa
- Protección contra corto circuito

APÉNDICE D: Código fuente del firmware

```
#include <avr/io.h>
#include <util/delay.h>
////////////////////////////////////
void usart_init(void); //Función encargada de iniciar la USART (Universal Synchronous and Asynchronous Receiver and Transmitter)

unsigned char tempo=0; // Variable encargada de almacenar el valor del registro UCSRA
unsigned char i=0;      //Variable utilizada para realizar ciclo
unsigned char j=0;      //Variable empleada para identificar el estado de las barreras
unsigned char ent;      //Variable que almacena las entradas del Puerto A
unsigned char sal;      //Variable que almacena las salidas del Puerto C
unsigned char data;     //Variable que almacena dato recibido por UDR (USART Data Register)
////////////////////////////////////

int main () //Inicio programa principal
{
  DDRA=0x00; //Declaración del puerto A como entrada
  //////////////////////////////////////
  //barrera1=0x01
  //barrera2=0x02
  //desactiva alarma (interruptor físico)=0x04
  //acceso exitoso (interruptor físico)=0x08
  //activa alarma (interruptor físico)=0x10
  DDRC=0xff; //Declaración del puerto C como salida
  //0x01=activa alarma
  //0x20= desactiva alarma
  //0x02= a= barrera1
  //0x04= b= barrera2
  //0x08= c= barrera1 repetida
  //0x10= d= barrera2 repetida
  //////////////////////////////////////
  PORTC=0xff; //Inicia salidas todas a '1' lógico
  usart_init(); //Inicializa la USART
  for(;;) //Ciclo infinito
  {
    tempo=UCSRA; //Se captura el estado de la USART
    if((tempo & (1 << RXC))==(1 << RXC)){ // ¿Se ha recibido un dato?
      data = UDR; //Dato recibido se respalda en ésta variable
      for(i=0;i<10;i++) //Repite las acciones que se ejecutan a partir de la instrucción del puerto serie, para evitar malinterpretación de caracteres, esto no influye con los procesos que cumple cada una de las siguientes funciones.
      {
```

```
if(data==97) // Si el caracter recibido corresponde a "a" (en código ascii 97 o 0x61)
{
    PORTC=0xff;
    sal=0x01;           //activa la alarma
    PORTC=~sal;
}
else if(data==98) // Si el caracter recibido corresponde a "b" (en código ascii 98 o 0x62)
{
    PORTC=0xff;
    sal=0x20;           //desactiva la alarma
    PORTC=~sal;
}
else if(data==99) // Si el caracter recibido corresponde a "c" (en código ascii 99 o 0x63)
{
    sal=0x00;           //registro exitoso
    PORTC=~sal;
    j=2;               //Indica que alguien está a punto de salir del área de registro
}
} //fin del for i
} //fin del if lectura de puerto
ent=PINA;             //Almacena las entradas del puerto A en ésta variable
if((ent&0x01)==0x01) //Si la barrera1 es atravesada
{
    while((ent&0x01)==0x01)ent=PINA; //Pausa mientras se termina de atravesar la barrera1
    _delay_ms(100); //Pausa para evitar arcos voltaicos producidos por los relevadores de las barreras
    if(j==0) //Introducir no. de usuario y contraseña en ODBMS (PC)
    {
        j=1; //Indica que alguien se está registrando
        sal=0x02;
        PORTC=~sal; //Indica mediante el LED (PIN PC1) que la barrera1 ha sido activada
        if(!(UCSRA & (1 << UDRE))) ; //envía una "a" a la PC
        UDR = 97;
    }
    else if (j==1) //Indica que alguien se está registrando y dará la indicación de "por favor espere su turno"
    {
        sal=0x08;
        PORTC=~sal; //Indica mediante el LED (PIN PC3) que la barrera1 ha sido activada nuevamente
        if(!(UCSRA & (1 << UDRE))) ; //envía una "c" a la PC
        UDR = 99;
    }
    else if (j==2) //Indica que hubo un registro exitoso y la persona está saliendo del área de registro
    {
```

```
        j=0;//Se restablece esta variable a '0', para que el sistema esté listo para otro registro
        if(!(UCSRA & (1 << UDRE))) ; //envía una "e" a la PC
        UDR = 101;
    }
}
else if((ent&0x02)==0x02) // Si la barrera2 es atravesada
{
    while((ent&0x02)==0x02)ent=PINA; //Pausa mientras se termina de atravesar la barrera2
    _delay_ms(100); //Pausa para evitar arcos voltaicos producidos por los relevadores de las barreras
    if(j==0) //Introducir no. de usuario y contraseña en ODBMS (PC)
    {
        j=1;//Indica que alguien se está registrando
        sal=0x04;
        PORTC=~sal;//Indica mediante el LED (PIN PC2) que la barrera2 ha sido activada
        if(!(UCSRA & (1 << UDRE))) ; //envía una "b" a la PC
        UDR = 98;
    }
    else if (j==1) //Indica que alguien se está registrando y dará la indicación de "por favor espere su turno"
    {
        sal=0x10;
        PORTC=~sal;//Indica mediante el LED (PIN PC4) que la barrera1 ha sido activada nuevamente
        if(!(UCSRA & (1 << UDRE))) ; //envía una "c" a la PC
        UDR = 99;
    }
    else if (j==2) //Indica que hubo un registro exitoso y la persona está saliendo del área de registro
    {
        j=0;//Se restablece esta variable a '0', para que el sistema esté listo para otro registro
        if(!(UCSRA & (1 << UDRE))) ; //envía una "e" a la PC
        UDR = 101;
    }
}
else if((ent&0x04)==0x04) //Si la alarma es desactivada desde interruptor3 (PIN PA2)
{
    while((ent&0x04)==0x04)ent=PINA;//Pausa mientras se termina de oprimir el interruptor
    _delay_ms(100); //Pausa para evitar arcos voltaicos
    sal=0x20;
    PORTC=~sal;//Desactiva la alarma y mediante el LED (PIN PC5) lo indica
}
else if((ent&0x08)==0x08) //Si es activado el interruptor2 (acceso exitoso)
{
    while((ent&0x08)==0x08)ent=PINA;//Pausa mientras se termina de oprimir el interruptor
    _delay_ms(100);//Pausa para evitar arcos voltaicos
```

```
        sal=0x00;
        PORTC=~sal;//Indica que el acceso ha sido exitoso
        j=2;//Indica que hubo un registro exitoso y la persona está saliendo del área de registro
    }
    else if((ent&0x10)==0x10) //Si la alarma es activada desde interruptor1 (PIN PC0)
    {
        while((ent&0x10)==0x10)ent=PINA;//Pausa mientras se termina de oprimir el interruptor
        _delay_ms(100);//Pausa para evitar arcos voltaicos
        sal=0x01;
        PORTC=~sal;//Activa la alarma y mediante el LED (PIN PC0) lo indica
    }
}
//for(;;)
return 0;
}
////////////////////////////////////
void usart_init(void)
{
    //Configuración del registro de selección de baud rate de la USART
    UBRRH = 0;
    //9600 baudios
    UBRL = 22; //Debido a que se utiliza un oscilador cristal de 3.579545MHz
    //Activa la transmisión y recepción de datos
    UCSRB = (1<<RXEN)|(1<<TXEN);
    //Tamaño del caracter 8 bits
    UCSRC = (1<<URSEL)|(0<<USBS)|(3<<UCSZ0);
}
```

Bibliografía y fuentes consultadas

1. **PhD. Aldo Pardo García, Ing. Dimitri Pulido Álvarez.** Red Internacional de Control Automático. *Artículo*. [En línea] [Citado el: 28 de Marzo de 2007.] http://www.control-automatico.net/htm/PDF/ART_02_DIMITRI.pdf.
2. **Kirschning, Ingrid.** ICT - Interactive and Cooperative Technologies Lab. *Graduate Thesis*. [En línea] Departamento de Ingeniería en Sistemas Computacionales Universidad de las Américas Puebla, México. , junio de 1992. [Citado el: 6 de Abril de 2007.] http://ict.pue.udlap.mx./people/ingrid/ingrid/Tesis_EI/EI.html.
3. **IMEI- Instituto Mexicano del Edificio Inteligente, A.C.** Edificio Inteligente: IMEI - Instituto Mexicano del Edificio Inteligente, A.C. *sitio web de IMEI - Instituto Mexicano del Edificio Inteligente, A.C.* [En línea] 28 de Noviembre de 2007. [Citado el: 20 de Diciembre de 2007.] <http://www.imei.org.mx>.
4. **José Manuel Huidobro Moya, Ramón J. Millán Tejedor, David Roldán Martínez.** "Tecnologías de Telecomunicaciones". México : Alfaomega, 2006, págs. 343,345.
5. **López, D. Valentino Francisco Cornejo.** Huella Digital y Control de Acceso: Artículos: Virusprot.com. *Detrás de la Huella*. [En línea] julio de 2002. [Citado el: 13 de Mayo de 2007.] <http://www.virusprot.com./Art29.html>.
6. **ALAS - Asociación Latinoamericana de Seguridad.** Artículos: ALAS - Asociación Latinoamericana de Seguridad. *pdf6*. [En línea] 2007. [Citado el: 19 de Septiembre de 2007.] www.alas-la.org/pdf/msn/6.pdf.
7. **Frenzel, Louis E.** "Sistemas Electrónicos de Comunicaciones". *Cap. I Introducción a las Comunicaciones electronicas: Espectro electromagnético*. México : Alfaomega, 2003.
8. **López, Ing. Miguel Briseño.** Barrera Infrarroja: Infrarrojos.net. [En línea] [Citado el: 10 de Agosto de 2007.] www.infrarrojos.net.
9. **Anónimo.** *Microcontroladores*. Tizayuca, Hgo : UAEH, 2006.
10. **Atmel Corporation.** Atmel: resources. [En línea] 2004. [Citado el: 20 de Septiembre de 2006.] http://www.atmel.com/dyn/resources/prod_documents/doc0856.pdf .
11. **López Chau, Asdrúbal.** "Microcontroladores AVR configuración total de periféricos". Toluca, Méx : editorial UAEM, 2006.
12. **Atmel Corporation.** *ATmega8535 DATA SHEET. ATMEL* . [PDF]
13. **Russell, Travis.** "Telecomunicaciones". México, D.F. : McGraw-HILL Interamericana Editores, S.A de C.V., 2002.

Fuentes Consultadas:

Ceballos, Fco. Javier (2002), "Curso de Programación Visual Basic 6", Alfaomega Grupo Editor México.

J. Hernandez, Michael, "Database design for mere mortals", segunda edición Addison Wesley.

Atmel Corporation, "ATmega/LS8535 DATA SHEET", consultado el (18-02-08) de www.atmel.com

"MAX232-DATASHEET" consultado el (13-06-08) de http://datasheetcatalog.net/es/datasheets_pdf/M/A/X/2/MAX232.shtml