

FACULTAD DE INFORMÁTICA Y CIENCIAS APLICADAS.

TÉCNICO EN INGENIERÍA DE HARDWARE.



TEMA:

DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE TIMBRE
AUTOMATIZADO, COMO APOYO AL CORRECTO CONTROL DE LOS
PERIODOS DE HORAS CLASE DURANTE LA JORNADA LABORAL DEL
CENTRO ESCOLAR PROFESOR JESÚS LEOCADIO PALENCIA, DE LA CUIDAD
DE SAN PABLO TACACHICO.

TRABAJO DE GRADUACIÓN PRESENTADO POR:

DANIEL OSWALDO ESTRADA ARAGÓN

FREDY ANTONIO BARAHONA

MIGUEL ALEXANDER RODRÍGUEZ LÓPEZ

PARA OPTAR AL GRADO DE:

TÉCNICO EN INGENIERÍA DE HARDWARE.

MARZO DE DE 2012.

SAN SALVADOR, EL SALVADOR, CENTROAMÉRICA.

INDICE

Contenido	Páginas
INTRODUCCIÓN.	i
1.1.- SITUACIÓN PROBLEMÁTICA.	1
1.2 ENUNCIADO DEL PROBLEMA.	2
1.3 JUSTIFICACIÓN.	2
1.4 OBJETIVOS.	4
1.4.1 Objetivo General.	4
1.4.2 Objetivos Específicos.....	4
1.5 ALCANCES.	4
1.6.- ESTUDIO DE FACTIBILIDAD.	6
1.6.1 Estudio económico.	7
1.6.2.- Estudio técnico.....	9
2.1 MARCO TEÓRICO DE REFERENCIA	16
2.1.1 ¿Que es un sistema electrónico embebido?.....	16
2.1.2 Características principales de un sistema electrónico embebido.	18
2.1.3 Bloques funcionales de un Sistema Embebido.	20
2.1.4 Áreas de aplicación de los sistemas electrónicos embebidos.	24
2.1.5 Preguntas frecuentes sobre sistemas embebidos y prototipos.	27
2.2 MARCO TEÓRICO DE SOLUCIÓN.	29
2.2.1 Definición de la solución.	29
2.3 MARCO TEÓRICO CONCEPTUAL.	32
2.4 DOCUMENTACIÓN TÉCNICA.	36
2.4.1 Características principales de Microcontrolador ATMEGA 168.....	37

2.4.2 Características generales de la pantalla LCD.....	38
2.4.3 Características principales del RTC (reloj en tiempo real) DS1307.	39
2.4.4 Características principales del teclado de membrana de 16 teclas.....	41
2.4.5 Características principales del Relé.	42
2.4.6 Características principales del BUZZER	43
2.4.7 Características generales del prototipo.....	45
3.1 PROPUESTA DE LA SOLUCIÓN.	46
3.1.1 ALGORITMO.....	48
3.1.2 FLUJOGRAMA.....	51
3.1.1 FIRMWARE.	53
3.1.4 DIAGRAMA.....	74
3.1.5 PCB O CIRCUITO IMPRESO DEL PROYECTO.	75
3.1.6 DESCRIPCIÓN DEL ENSAMBLE.	76
3.2 CONCLUSIONES.....	79
3.3 RECOMENDACIONES.....	80
3.4 BIBLIOGRAFÍA.....	81
ANEXOS.....	82
MANUAL DEL TIMBRE AUTOMATIZADO.....	88

INTRODUCCIÓN.

En el presente documento el lector tendrá a su disposición información sobre el desarrollo, diseño y construcción del proyecto llamado: Diseño e implementación de un *sistema de timbrado automático*, en el cual se pretende optimizar el recurso humano.

El primer capítulo, contiene características detalladas acerca de la formulación del proyecto, se especifican los objetivos perseguidos, los beneficios a obtener con la implementación, así como los estudios de factibilidad tanto económica como técnica, de los componentes principales del proyecto. De antemano se destacan las principales características: funcionalidad, facilidad de uso, precisión y exactitud la cual consideramos como principal atractivo ya que en ellas se encierran todos los beneficios y soluciones que se pretende dar a la institución.

Este documento también incluye la carta de aceptación donde se estipula que la institución permite que se implemente en su campus el presente proyecto de graduación, además se anexa a este la matriz de congruencia donde se muestran los objetivos, alcances y el producto final.

Para el diseño e implementación de un *sistema de timbre automatizado*, como apoyo al correcto control de los periodos de horas clase durante la jornada laboral del centro

escolar Profesor Jesús Leocadio Palencia de la ciudad de San Pablo Tacachico, el capítulo dos se divide en cuatro sub contenidos que son:

1. **Marco teórico de referencia:** donde se describen las áreas de aplicación en que se fundamenta el proyecto.
2. **Marco teórico de solución:** es donde se presenta la teoría referente a la solución provista por el proyecto, dando a conocer los componentes del circuito, así como su implementación en las instalaciones del centro escolar.
3. **Marco teórico conceptual:** donde se presentan una serie de conceptos teóricos que se van utilizando en el desarrollo del proyecto.
4. **Documentación técnica:** contiene las características técnicas generales de los componentes eléctricos y electrónicos de todo el proyecto.

En el capítulo tres se muestra el desarrollo e implementación del *sistema de timbre automatizado* como solución a la problemática planteada por el centro escolar, así como de la elaboración del algoritmo para mayor comprensión del usuario, ya que en este se muestra el comportamiento que tendrá el circuito. El flujograma como la representación gráfica del algoritmo, el desarrollo del firmware o programa de control del circuito que será cargado al microcontrolador a través de un programador y con el apoyo del software BASCOM-AVR.

Se presenta el diagrama (Figura 3.2 Pág. 73) del circuito como una guía en la fabricación de las pistas y a su vez para el ensamble de este, así se podrá verificar la forma de

conectar cada uno de los componentes que serán utilizados en la fabricación de este proyecto.

Se dan a conocer las recomendaciones y conclusiones que se tienen al finalizar este proyecto, además de los anexos y el manual de usuario que se agrega para una mayor comprensión del sistema automatizado de timbre que será de gran apoyo para las personas que manipulen este sistema.

CAPITULO I: SITUACIÓN PROBLEMÁTICA

1.1.- SITUACIÓN PROBLEMÁTICA.

El Centro Escolar Profesor Jesús Leocadio Palencia, se encuentra ubicado en la 5ª Avenida de la Ciudad de San Pablo Tacachico, en el departamento de La Libertad, su área cubre una manzana, posee tres pabellones donde están ubicadas las aulas, en el primer pabellón están de sexto a noveno grado, en el segundo pabellón de segundo al quinto grado, y en el tercer pabellón de parvularia a primer grado.

El centro escolar cuenta actualmente con una población estudiantil de cuatrocientos ochenta alumnos, divididos en diecinueve secciones, en los turnos matutino y vespertino. Se cuenta con espacios de apoyo académico para los alumnos como: Aula de Informática y salón de usos múltiples; además cuenta con los servicios de: servicios sanitarios, bodega, cocina, cafetines, cancha de basquetbol y cancha de futbol.

Actualmente se cuenta con un sistema de timbre manual para dar aviso a los estudiantes sobre el inicio y finalización de las diversas actividades que se desarrollan a través de la jornada académica tales como: inicio y finalización de clases etc., Esta situación genera problemas de diversas índoles, desde la alteración en las actividades hasta el uso de recurso humano que se podría dedicar a otras actividades más importantes que estar pendiente de hacer sonar un timbre.

Las autoridades del Centro Escolar Profesor Jesús Leocadio Palencia se acercaron a la Universidad Tecnológica de El Salvador (UTEC), específicamente a los alumnos de la carrera de Técnico en Ingeniería de Hardware, de la Escuela de Informática y Ciencias Aplicadas a plantear dicho problema para obtener una solución inmediata.

1.2 ENUNCIADO DEL PROBLEMA.

¿Cómo lograr que el Centro Escolar Profesor Jesús Leocadio Palencia tenga una solución de hardware con tecnología actualizada, para su problema con el sistema manual de timbrado?

1.3 JUSTIFICACIÓN.

Es necesario e indispensable que el Centro Escolar Profesor Jesús Leocadio Palencia cuente con posibles soluciones para su problema que le afecta en gran medida. Por tal motivo el proyecto propuesto ofrece una solución factible a la problemática que presenta el centro escolar con respecto al sistema de timbrado manual con el que cuentan actualmente, de tal manera que permitirá que los estudiantes de la Universidad Tecnológica de El Salvador puedan desarrollar un prototipo de sistema embebido que pueda dar solución a la problemática.

La situación actual del sistema de timbrado de la institución acarrea una serie de problemas como: los tiempos estipulados para cada una de las actividades de la jornada académica carecen de precisión y esto genera trastornos en el horario ya establecido, el recurso humano asignado para esta tarea se puede utilizar en otro tipo de actividades de mayor importancia, además el centro escolar debe disponer de sistemas actualizados para el funcionamiento de sus instalaciones.

El proyecto propuesto como una posible solución consiste en diseñar, fabricar e instalar dentro de las instalaciones del Centro Escolar Profesor Jesús Leocadio Palencia, un sistema electrónico con la función de automatizar el encendido y apagado del timbre, con una interfaz de usuario para la programación de las diversas alarmas u horarios de timbrado. Dicho sistema está basado en tecnología embebida mediante la utilización de un microcontrolador, constará de una Pantalla De Cristal Líquido (LCD) para visualizar la hora y tiempos de timbrado, junto con un teclado el cual servirá para configurar o modificar los datos, además de un pulsador que se utilizara para hacer sonar el timbre de forma manual.

Algunas de los beneficios que dicha institución obtendrá son:

- Por ser un sistema automatizado tendrá un reloj interno que accionará el timbre en las horas programadas.
- Ser un apoyo para el profesor o encargado de hacer cumplir el horario de clases dentro de la institución.
- Optimizar el recurso humano de centro educativo.

- Ser un apoyo en el correcto control de las diversas jornadas de trabajo.

1.4 OBJETIVOS.

1.4.1 Objetivo General.

Desarrollar e implementar un sistema electrónico para automatizar el sistema de timbrado que ayude al control del horario establecido de la jornada académica, para el centro escolar profesor Jesús Leocadio Palencia de la ciudad de San Pablo Tacachico.

1.4.2 Objetivos Específicos.

- Construir un prototipo electrónico para el control automático de un timbre, utilizando tecnología innovadora y de bajo costo.
- Diseñar un código en lenguaje propietario para micro-controlador Atmega 168, que controle el funcionamiento del sistema automatizado de timbre.
- Implementar en el centro escolar el sistema automatizado de timbre, y así brindar una solución de hardware eficiente y adecuado, como apoyo en el control de los horarios de clases en el transcurso de la jornada laboral.

1.5 ALCANCES.

Para la realización de este proyecto se han propuesto determinados alcances, los cuales se convierten en promesas que se buscan alcanzar el desarrollo de este proyecto,

además cada alcance conlleva la realización de un producto al final del trabajo. A continuación se hace un desglose de estos alcances y productos para el proyecto.

PROMESA.	PRODUCTO.
1.- Construir un prototipo electrónico para el control automático de un timbre, utilizando tecnología innovadora y de bajo costo.	1.- Prototipo funcional: que cumpla con la función de accionamiento automatizado del timbre.
2.- Diseñar un código en lenguaje propietario para micro-controlador Atmega 168, que controle el funcionamiento del sistema automatizado de timbre.	2.- Código fuente funcional: en lenguaje de programación BASIC, para ser descargado en el Micro-controlador y controlar el sistema de timbrado.
3.- Brindar una solución de hardware eficiente para el centro escolar, como apoyo en el control de los horarios de clases en el transcurso de la jornada laboral.	3.- Sistema de timbre automatizado: que se programará para controlar los diferentes horarios de la jornada de trabajo del Centro Escolar Profesor Jesús Leocadio Palencia.

1.6.- ESTUDIO DE FACTIBILIDAD.

Para analizar la factibilidad técnica y económica de este proyecto se debe definir qué clase de elementos se utilizarán, donde se compraran y se debe conocer cada uno de los componentes electrónicos que forma nuestro sistema para poder iniciar la investigación del funcionamiento y del costo de cada uno de los componentes electrónicos.

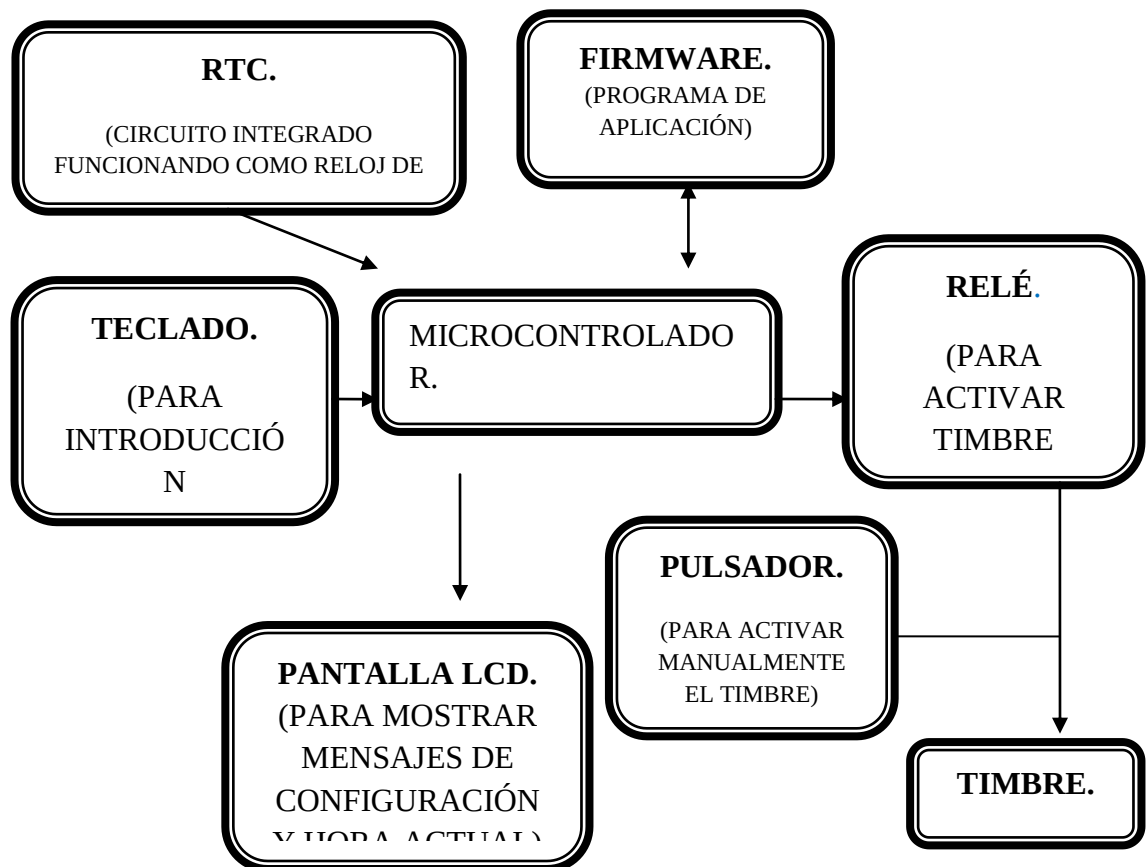


Figura. 1.

FIGURA 1. Esquema general de bloques del proyecto: Esto es la forma lógica de cómo funciona el prototipo, en él se muestran los principales bloques funcionales que lo componen.

1.6.1 Estudio económico.

En el siguiente apartado se muestra una comparación de precios de los diferentes componentes a utilizar en la elaboración del proyecto, con el objetivo de apreciar la factibilidad económica que ofrece cada componente.

MICROCONTROLADOR.			
Características.	Pic16f876	Atmega 168	Picaxe28
Precio de cotización (USD).	\$15.00	\$8.00	\$17.00
Lugar de cotización.	www.mouser.com	www.mouser.com	www.rev-ed-co.uk

Elección de microcontrolador: para el desarrollo de este prototipo electrónico, el microcontrolador a utilizar es uno de los principales componentes del proyecto, ya que este es el encargado de enviarle las ordenes a los demás componentes dentro del circuito, motivo por lo cual se optó por el microcontrolador Atmega 168, de la familia AVR fabricado por la compañía de ATMEL, la elección se tomó ya que es el de menor costo entre las ofertas presentadas.

PANTALLA DE DESPLIEGUE (LCD).			
Características.	Hd44780 16x4	Hd44780 8x4	Hd44780 16x 6
Precio de cotización (USD)	\$20	\$17	\$25
Lugar de cotización	www.mouser.com	www.mouser.com	www.mouser.com

Elección de pantalla LCD: esta es la encargada de mostrar los datos, en esta ocasión se elige una pantalla de 16 caracteres y 4 líneas, ya que se piensa que por tener 4 filas se tienen más opciones para mostrar y el costo económico es bastante accesible como para tomarlo en cuenta dentro del proyecto

CIRCUITO INTEGRADO RELOJ DE TIEMPO REAL (RTC).			
Características.	DS 1307	DS 1308	DS 1309
Precio de cotización (USD).	\$13.00	\$15.00	\$17.00
Lugar de cotización.	JOSNAB	JOSNAB	JOSNAB

Elección de circuito integrado (RTC): para el desarrollo de este prototipo, el circuito integrado a utilizar será el DS 1307, que tendrá la función de guardar la hora, para cuando ocurra un corte de energía no se resetee la hora del sistema, además de tener menor costo entre los ofertados.

TECLADO MATRICIAL DE MEMBRANA.			
Características.	16 teclas	12 teclas	09 teclas
Precio de cotización (USD).	\$10.00	\$8.00	\$6.00
Lugar de cotización.	JOSNAB	CASA RIVAS	JOSNAB

Elección del teclado de membrana de 16 teclas T-105: para el desarrollo de este prototipo electrónico el teclado a utilizar es uno de los principales componentes dentro del circuito, ya que ofrece mayor número de variables a la hora de modificar el software de programación del sistema.

1.6.2.- Estudio técnico.

Aquí se define la elección de los componentes desde el punto de vista técnico, tomando en cuenta las características electrónicas, capacidad de almacenamiento y velocidad lógica y física del componente o dispositivo, para que sea posible que el circuito funcione correctamente y puedan cumplirse los objetivos planteados dentro del proyecto.

MICROCONTROLADOR. (PORCENTAJE MÍNIMO DE ACEPTACIÓN = 75%)							
Características.		Pic16f876		Atmega 168		Picaxe28	
	Peso	Valor	Porcentaje	Valor	Porcentaje	Valor	Porcentaje
Velocidad	20 %	2	40 %	2	40 %	2	40 %
Memoria	30 %	2	60%	2	60 %	2	60 %
Lenguaje de programación	20 %	2	40 %	2	40 %	2	40 %
Vía de programación	15 %	1	15%	2	30 %	2	30 %
Precio	15 %	1	15 %	2	30 %	1	15%
Total		85 %		100 %		92.5 %	

Escala de ponderación: 0 = no cumple; 1 = bueno; 2 = excelente.

Elección del micro-controlador: Atmega 168 de la familia AVR del fabricante ATMEL, esto debido a que cumple con todos los requisitos técnicos necesarios para la implementación de este proyecto, una de las características más importante es el lenguaje y compilador para su programación, como lo es BASIC, usando el BASCOM-AVR, el cual es un IDE y compilador de lenguaje Basic para micro-controlador de la familia AVR.

PANTALLA DE CRISTAL LÍQUIDO (LCD). (PORCENTAJE MÍNIMO DE ACEPTACIÓN = 75%)							
Características.		LCD 16x4		LCD 8x4		LCD 16x6	
	Peso	Valor	Porcentaje	Valor	Porcentaje	Valor	Porcentaje
Nº de pines	10 %	2	20 %	2	20 %	2	20 %
Color	20 %	2	40%	1	20 %	2	40 %
Tamaño	20 %	2	40 %	2	40 %	2	40 %
Nº de caracteres	30 %	2	60%	0	00 %	2	60 %
Nº de filas	20 %	2	40 %	2	40 %	1	20 %
Total		100 %		60 %		90 %	

Escala de ponderación: 0 = no cumple; 1 = bueno; 2 = excelente.

Elección de pantalla de cristal líquido (LCD): 16x4, se opta por este modelo debido a que tiene mayor despliegue de caracteres, mejor resolución, tamaño adecuado, un consumo reducido de energía.

CIRCUITO INTEGRADO (RTC).							
(PORCENTAJE MÍNIMO DE ACEPTACIÓN = 75%)							
Características.		DS 1307		DS 1308		DS 1309	
	Peso	Valor	Porcentaje	Valor	Porcentaje	Valor	Porcentaje
N° de pines	20 %	2	40 %	1	20 %	2	40 %
Protocolo de comunicación	20 %	2	40%	1	20 %	1	20 %
Memoria interna	20 %	2	40 %	1	20 %	0	00 %
Velocidad	20 %	2	40%	2	40 %	1	20 %
Voltaje	20 %	2	40 %	1	20 %	1	20 %
Total		100 %		60 %		50 %	

Escala de ponderación: 0 = no cumple; 1 = bueno; 2 = excelente.

Elección del circuito integrado: el DS 1307, esto debido a que cumple con todos los requisitos técnicos necesarios para la implementación de este proyecto.

TECLADOS DE MEMBRANA.

(PORCENTAJE MÍNIMO DE ACEPTACIÓN = 75%)							
Características.		DS 1307		DS 1308		DS 1309	
	Peso	Valor	Porcentaje	Valor	Porcentaje	Valor	Porcentaje
N° de teclas	30 %	2	60 %	1	30 %	2	60 %
Tamaño	20 %	2	40 %	2	40 %	1	20 %
Tipo de conector	10 %	2	20 %	0	00 %	2	20 %
Diseño	20 %	2	40 %	2	40 %	1	20 %
Voltaje	20 %	2	40 %	1	20 %	1	20 %
Total		100 %		65 %		70 %	

Escala de ponderación: 0 = no cumple; 1 = bueno; 2 = excelente.

Elección del teclado de membrana T-105: para este proyecto elegimos uno de 16 teclas, debido a que posee mayor número de líneas y esto nos proporciona mayores opciones de configuración, cumpliendo con los requisitos técnicos necesarios para este proyecto.

MATRIZ DE CONGRUENCIA.

MATRIZ DE CONGRUENCIA.		
TEMA: <i>Diseño e implementación de un sistema de timbre automatizado como apoyo al control de los periodos de clase/receso para el Centro Escolar Prof. Jesús Leocadio Palencia de La Ciudad de San Pablo Tacachico, Departamento de La Libertad.</i>		
ENUNCIADO DEL PROBLEMA: ¿Cómo lograr que el Centro Escolar Profesor Jesús Leocadio Palencia tenga una solución de hardware con tecnología actualizada, para su problema con el sistema manual de timbrado?		
OBJETIVO GENERAL: Desarrollar e implementar un sistema electrónico para automatizar el sistema de timbrado que ayude al control del horario establecido de la jornada académica, para el centro escolar profesor Jesús Leocadio Palencia de la ciudad de San Pablo Tacachico.		
OBJ. ESPECIFICO 1: Construir un prototipo electrónico para el control automático de un timbre, utilizando tecnología innovadora y de bajo costo.	OBJ. ESPECIFICO 2: Diseñar un código en lenguaje propietario para micro-controlador atmegax8, que controle el funcionamiento del sistema automatizado de timbre.	OBJ. ESPECIFICO 3: Implementar en el centro escolar el sistema automatizado de timbre, y así brindar una solución de hardware eficiente y adecuado, como apoyo en el control de los horarios de clases en el transcurso de la jornada laboral.
ALCANCE 1: Construir un prototipo electrónico para el control automático de un timbre, utilizando tecnología innovadora y de bajo costo.	ALCANCE 2: Diseñar un código en lenguaje propietario para micro-controlador atmegax8, que controle el funcionamiento del sistema automatizado de timbre.	ALCANCE 3: Brindar una solución de hardware eficiente para el centro escolar, como apoyo en el control de los horarios de clases en el transcurso de la jornada laboral.
PRODUCTO 1: Prototipo funcional que cumpla con la función de accionamiento automatizado del timbre.	PRODUCTO 2: Código fuente funcional en lenguaje de programación BASIC, para ser descargado en el Micro-controlador y controlar el sistema de timbrado.	PRODUCTO 3: Sistema de timbre automatizado que se programará para controlar los diferentes horarios de la jornada de trabajo del Centro Escolar Profesor Jesús Leocadio Palencia.
DOCUMENTACION TECNICA:		
PROYECTO TEMATICO: Sistemas domoticos.		
DETALLE DEL PRESUPUESTO PROYECTADO:		
OFERTA ECONOMICA:		

CHARTA DE ACEPTACION

CAPITULO II: MARCO TEÓRICO DE REFERENCIA

2.1 MARCO TEÓRICO DE REFERENCIA

Este apartado presenta información importante de cada área en las que se fundamentará el proyecto denominado *“Diseño e implementación de un sistema de timbre automatizado como apoyo al control de los periodos de clase/receso para el Centro Escolar Prof. Jesús Leocadio Palencia de La Cuidad de San Pablo Tacachico, Departamento de La Libertad”*.

2.1.1 ¿Que es un sistema electrónico embebido?

Un sistema embebido (SE), es un conjunto de electrónica, informática y hasta mecánica, diseñado para la realización de una función, aplicación o tarea específica. En algunos casos estos sistemas embebidos dentro de un sistema de mayor escala. Los sistemas embebidos se utilizan para usos muy específicos, y en la mayoría de estos sistemas los componentes se encuentran incluidos en la placa base del dispositivo.

Algo muy notable en estos sistemas es el precio y el consumo. Esto debido a que los sistemas embebidos se fabrican por unidad, decenas o por millones de unidades, logrando así reducir los costos. Los sistemas embebidos suelen usar un procesador y una memoria pequeña para reducir los costos y reducir el consumo eléctrico.

Usualmente estos sistemas están basados en microprocesadores o microcontroladores y la tarea o función que realizan esta especificada por el firmware o programa de aplicación específico, alojado en la memoria del sistema. En cuanto a la velocidad,

dependerá de los elementos que se utilicen, ya que lo que se pretende con estos sistemas es simplificar toda la arquitectura haciendo posible que un solo dispositivo realice una función específica de forma práctica y rápida.

Para ello un sistema embebido debe estar diseñado para presentar en tiempo real las entradas, activando salidas, es decir su funcionamiento es en tiempo real y según lo que se guarde en sus entradas así se activarán las salidas:

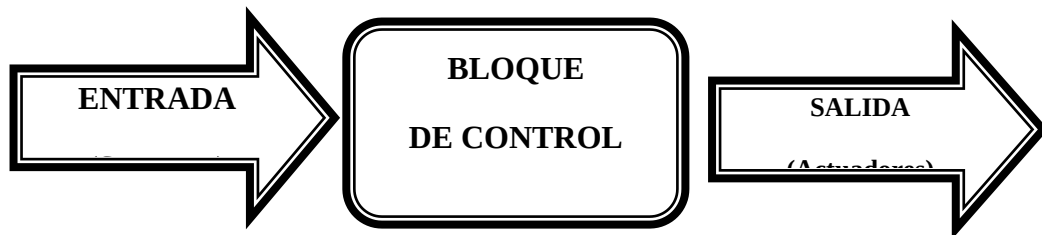


Figura 2.1: Esquema general de un sistema electrónico embebido.

La figura 2.1 muestra los tres bloques generales por lo que está compuesto un Sistema Embebido, como lo son las entradas por la que se introducen los datos de un usuario externo, el bloque de control de todo el dispositivo en el que se guarda el firmware, y como salida el contador interno que enviara los pulsos al timbre.

Como se mencionó anteriormente, parte fundamental del sistema embebido es el software de aplicación específica o firmware. El firmware funciona para SE de igual diseño y iguales componentes, es decir es un programa informático para un hardware específico, no es compatible con otro hardware. Este firmware debe ser diseñado en un

lenguaje de programación específico para el microcontrolador usado en él SE, normalmente se utilizan lenguajes de alto nivel como C o Basic.

2.1.2 Características principales de un sistema electrónico embebido.

Un sistema embebido en general debe poseer ciertas características indispensables con las funciones adecuadas, haciendo posible que el sistema embebido cumpla con la función que se ha diseñado. Estas características se listan y describen a continuación.

Diseñado para una tarea específica: se dice que un sistema embebido debe ser diseñado para realizar una única tarea, por lo que son sistemas muy eficientes y dedicados, logrando así ser sistemas que se integren dentro de sistemas de mayor escala.

Es por eso que los sistemas embebidos son dispositivos electrónicos usados para controlar y operar equipos, dispositivos, maquinas, aparatos domésticos, equipos móviles, automóviles, elementos electrónicos, etc.

Integran hardware y software: además de ser dispositivos formados por elementos electrónicos, estos sistemas embebidos integran un software o firmware, el cual es un código de programa que se descarga a la memoria del dispositivo, dicho programa debe estar bien diseñado específicamente para responder al hardware electrónico diseñado, ya que de él depende el correcto funcionamiento de la parte lógica del dispositivo. Este

firmware debe ser escrito en cualquier lenguaje de programa compatible con la CPU o procesador central del sistema embebido.

Trabaja en tiempo real: existen dispositivos diseñados para realizar funciones específicas como los controladores que se utilizan en centrales telefónicas, ascensores, sistemas de adquisición de datos. Sistemas de diagnóstico y control: todas estas funciones son realizadas en tiempo real. Es decir, el sistema debe ser capaz de responder instantáneamente a cualquier cambio de sus entradas (sensores, pulsadores, interruptores), realizando algún proceso o modificando su entorno.

Manejan entradas: todo sistema necesitará entradas que procesar, para decidir cómo actuar en dependencia del estado de estas, en el caso de los sistemas embebidos es común el uso de sensores, los cuales son capaces de captar las señales físicas y enviarlas al bloque de control de dispositivos, para que este reaccione de acuerdo a su funcionamiento. Estas entradas pueden provenir del entorno del sistema, por ejemplo, un sensor de temperatura, un sensor de presencia, etc., estas entradas pueden ser internas al sistema, por ejemplo, la hora actual, el valor de una variable, etc.

Manejan salidas: como se menciono, los sistemas reaccionan a sus entradas modificando salidas, las cuales pueden ser internas y externas, una salida interna puede ser: la escritura de una posición en memoria, el envío de información a un chip interno, etc., por el contrario una salida externa se define como aquella que puede modificar el

entorno externo del sistema, para esto se dota de actuadores, los actuadores son los encargados de interactuar y/o modificar el entorno físico del sistema embebido, algunos ejemplos de estos son los diodos emisores de luz, parlantes, motores, etc.

Bajo costo económico: debido a su especialización y a que los sistemas embebidos combinan tanto software como hardware electrónico, su costo de elaboración es relativamente bajo ya que son sistemas para realizar tareas específicas, por lo que el costo de los materiales no es muy elevado, lo que garantiza un costo económico accesible.

Eficiencia (bajo consumo de potencia): Los sistemas embebidos deben ser diseñados teniendo en mente el consumo de potencia y la eficiencia de esta, debido a que son sistemas que estarán dedicados a tareas específicas y serán sistemas que no dependerán de una constante intervención de un usuario. Debido a su poca complejidad en su diseño son dispositivos eficientes y su consumo potencial es mínimo, ya que los materiales que se utilizan son de calidad y por su tamaño que suele ser pequeño su consumo eléctrico es mínimo.

2.1.3 Bloques funcionales de un Sistema Embebido.

Como todo sistema, los sistemas embebidos pueden estructurarse como una unión de bloques funcionales en los cuales se agrupan las sub-funciones internas del sistema. En

la figura 2.2 se puede ver un detalle de la estructura general de un SE. A continuación se describen cada uno de estos bloques.

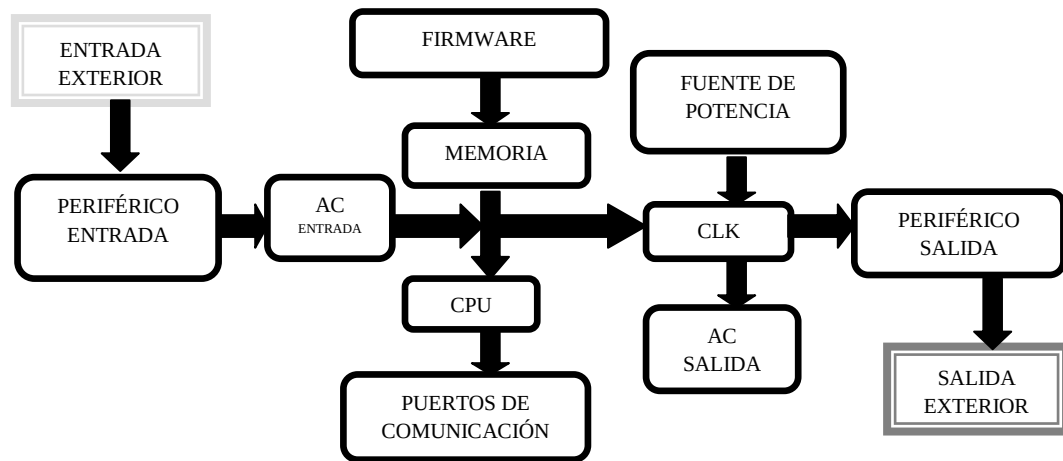


Figura 2.2: Esquema de bloques internos de un sistema embebido en general, donde se puede observar que este sistema internamente debe estar compuesto por bloques específicos que cumplen funciones que son parte de un todo. Cada bloque será constituido por diversos componentes de electrónica.

Periféricos de entrada: es un componente electrónico que hace posible la conexión con su entorno físico, es decir con otro dispositivo, en el caso de los sistemas embebidos los periféricos de entrada le permiten al sistema recibir información desde el equipo externo, por medio de estos dispositivos se hace posible transferir la información necesaria desde un ambiente físico externo hacia el interior del sistema. Por ejemplo, están los sensores de luz, teclados, micrófonos, etc.

Acondicionamiento de señal de entrada: Este bloque es el encargado de tomar las señales provenientes de los periféricos de entrada y enviársela al bloque correspondiente dentro del sistema embebido, por ejemplo pueden ser los sensores ya que estos se encargan de captar la señal física exterior y en algunos casos la convierten en energía eléctrica para enviársela al bloque de control.

Firmware: todo sistema embebido debe estar controlado por órdenes específicas que dentro de un programa o software, es un sistema operativo de bajo nivel diseñado específicamente para una marca o modelo. Este programa de aplicación puede estar escrito en cualquier lenguaje de programación compatible con el procesador a utilizar. Este firmware puede ser actualizado, pero debe tomarse muy en cuenta el sistema embebido que se está manejando, ya que según la implementación de hardware que se tenga así debe ser la estructura lógica del firmware que se le debe descargar para que el dispositivo cumpla con la función que se pretende.

Memoria: ¿Qué es la memoria? En este caso, la memoria que se utiliza ya viene incorporada en el microcontrolador y en cuanto al tamaño de esta debe considerarse cuál es la finalidad del dispositivo ya que así será el tamaño del firmware que se le debe descargar. Este bloque puede estar formado por memoria de tipo RAM.

Unidad central de proceso CPU: Este es el bloque encargado de ejecutar las acciones, se encarga de dar las ordenes a los demás bloques, con respecto a que acción realizara y se encuentra en la parte central del esquema en donde se ubica el micro procesador o

micro controlador, es decir la CPU o unidad que aporta capacidad al sistema teniendo la opción de incluir memoria interna o externa.

Puertos de comunicación: Estos dispositivos son parte fundamental en el circuito, ya que proveen al sistema embebido (SE) comunicación bi-direccional con otros sistemas por ejemplo: RS-232, I 2C, estos son protocolos que permiten la programación del sistema embebido.

Fuente De Potencia: para implementar este bloque puede utilizarse una fuente de corriente o baterías alcalinas, esto es opcional pero debe tomarse en cuenta que el bloque de potencia es el encargado de proveer la energía necesaria a todo el circuito electrónico, en este caso deben tomarse en cuenta las exigencias de los dispositivos electrónicos y la finalidad del sistema embebido para determinar que magnitud de corriente se le debe adaptar, esto depende del consumo del sistema embebido.

Reloj del sistema CLK: el klok, es un reloj que marca el tiempo real, desde MHz hasta GHz, y trabaja con un cristal externo, el cual le permite que su rendimiento se adapte al tiempo real externo y para implementarlo basta con adaptar el cristal externo y adaptarlo al tiempo real.

Acondicionamiento de señal de Salida: como salidas están los actuadores, que pueden

ser motores. Estos son los encargados de interactuar con el entorno exterior, los que se encargan de hacer posible el desplazamiento del dispositivo, si fuere el caso, este bloque es el que proporciona la energía necesaria a los periféricos de salida, para que sea posible mostrar el funcionamiento del sistema embebido.

Periféricos de Salida: son los bloques o puertos de conexión encargados de permitir las salidas del sistema, a dispositivos externos o actuadores, los periféricos o componentes que permiten obtener el resultado final del funcionamiento completo del sistema embebido, en estos se pueden encontrar, leds , pantallas LCD, display, etc.

2.1.4 Áreas de aplicación de los sistemas electrónicos embebidos.

En el mundo actual los sistemas embebidos están inmersos dentro de nuestra vida diaria. Sus aplicaciones abarcan muchos sistemas y aparatos, los cuales son utilizados por los seres humanos en el diario vivir, algunos de estos se mencionan a continuación:

Hogar: En los hogares es muy común encontrar sistemas embebidos, aunque muchas veces no están muy visibles, pero se encuentran dentro de las lavadoras de ropa, dentro de los televisores digitales, sistemas de audio, reproductores MP3, controles remotos, alarmas electrónicas, en los sistemas de climatización y temperatura, calefacción, sistemas de video como grabadores y reproductores, teléfonos celulares, en módems, routers, entre otros.

Consumo: Los equipos que se clasifican en esta área son de uso muy común, en especial en los hogares, ya que son sistemas diseñados para realizar tareas específicas en tiempo real, entre ellos están: televisores, pantallas, microondas, refrigeradoras, lavadoras, MP3player, celulares entre otros.

Oficina: En las oficinas también es posible encontrarlos como en cámaras fotográficas digitales, alarmas digitales, sistemas de control de acceso, aire acondicionado y muchos más.

Transporte: Uno de los ejemplos más comunes e indispensables que se usan en la actualidad son, los automóviles modernos, estas máquinas tienen más de 200 sistemas embebidos en su infraestructura. Ya que para controlar muchas partes del vehículo, se utilizan sistemas muy precisos como lo son los frenos ABS, el sistema de inyección de combustible, aceleración, carburación, sistema de luces y señalización, luces de cabina, climatización y aire acondicionado, limpia vidrios entre otros.

Comunicaciones: En esta área es fácil identificar muchos aparatos que están basados en sistemas embebidos, como los ya mencionados routers, access point, comunicación WIFI, switches de comunicación administrables remotamente, sistemas de seguridad informática como los cortafuegos, sistemas de filtrado de paquetes TCP/IP, módems, sistemas utilizando fibra óptica, estos entre una gran variedad de sistemas embebidos.

clasificados en esta área.

Medicina: En la actualidad en el área de la salud también se encuentra equipos basados en sistemas embebidos, los cuales van desde simples termómetros digitales hasta sistemas de monitoreo para pacientes con problemas cardiacos, también ecógrafos digitales, monitores de apnea del sueño, y en otros aparatos electrónicos que ya existen y sin duda se seguirá creando sistemas embebido en el área de la medicina.

Robótica: La mayoría de robots diseñados en el área de entretenimiento y muchos industriales, se basan en sistemas embebidos, capaces de manejar los sistemas articulares, de video, transporte por el control de velocidad y dirección de motores , en el área de comunicaciones, en oficinas y muchas otras aéreas en las que se utilizan sistemas electrónicos embebidos.

Domótica: En esta área se encuentran los sistemas encargados de automatizar las viviendas, se basan completamente en sistemas embebidos, los cuales están desde la climatización y calefacción, sistemas de seguridad para el hogar, sistemas de iluminación, control de consumo de energía, medios de comunicación del hogar, servicios de TV digital, sistemas de riesgos, entre otros mas, ubicados en el área de la domótica.

2.1.5 Preguntas frecuentes sobre sistemas embebidos y prototipos.

¿Qué es un prototipo electrónico?

Es un primer modelo o molde original que se fabrica y de este modelo se parte para la elaboración de lo que se pretende, un objeto, herramienta, dispositivo electrónico, etc.

¿Qué tipos de prototipos electrónicos existen?

Existen una gran variedad de prototipos electrónicos que luego se convierten en ejemplares para el diseño y desarrollo de muchos equipos de los mismos, entre ellos se encuentren: prototipos mecánicos, eléctricos, análogos, digitales, industriales, médicos, hogareños, prototipos diseñados para el área de comunicación, etc.

¿Qué es un Microcontrolador y cómo funciona?

Un microcontrolador es un chip o circuito integrado en cuyo interior incorpora los componentes o bloque básicos para conformar un sistema embebido. Una consecuencia de su pequeño tamaño es que los recursos (memoria velocidad) están limitados en comparación a una PC o a un sistema embebido basado en microprocesadores.

En términos funcionales un microcontrolador es un chip reprogramable que controla un sistema

embebido y físicamente el microcontrolador es un chip con muchos pines (desde 8 hasta más de 40) estos pines son usados para alimentación del reloj, puertos digitales, puertos

análogos, comunicación y más.

¿Qué método de programación se debe utilizar?

Existen una amplia gama de software para desarrollar sistemas embebidos hay programas diseñados específicamente para la descarga de códigos hacia los sistemas embebidos así como también existen muchas herramientas diferentes. Lenguajes de programación sencillos o de alto nivel, siendo la herramienta disponible para que el usuario elija la que más le parezca y se adecúe a la finalidad del sistema embebido.

¿Que es una pantalla LCD?

Una pantalla de cristal liquido o LCD (acrónimo en inglés de liquid crystal display) es una pantalla delgada y plana formada por un número de pixeles en color o monocromos colocados delante de una fuente de luz o reflectora. A menudo se utiliza en dispositivos electrónicos de pila, ya que utiliza cantidades muy pequeñas de energía eléctrica.

¿Cómo funciona el buffer de corriente?

Funciona como circuito amplificador de corriente para el correcto funcionamiento de los dispositivos que se utilizan en el círculo electrónico, encargándose de que la corriente que llegue a ellos sea la adecuada.

2.2 MARCO TEÓRICO DE SOLUCIÓN.

Se ha propuesto del diseño y construcción de un sistema automatizado de timbre para el correcto control de los periodos de horas clase dentro de la jornada de trabajo de los docentes del Centro Escolar Profesor Jesús Leocadio Palencia, con su implementación permitirá tener un control preciso de los cambio de horas clase y con lo que se estaría optimizando el recurso humano.

2.2.1 Definición de la solución.

La solución propuesta es la construcción de un SE basado en microcontrolador que funciones como un sistema electrónico automatizado para el control de horas clase, como apoyo a la institución para optimizar recursos y tiempo, para el Centro Escolar Profesor Jesús Leocadio Palencia.

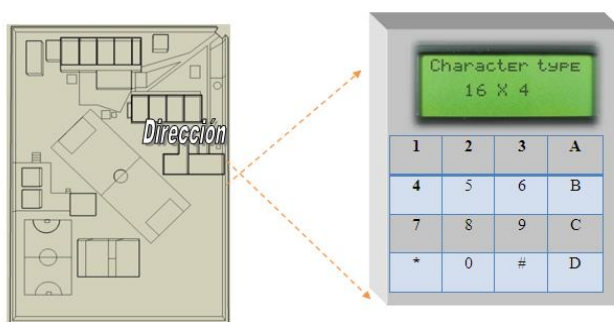


Figura 2.3: Ubicación del circuito electrónico dentro de las instalaciones del centro escolar.

En la Figura 2.3 se muestra la ubicación física del circuito electrónico que será el encargado de controlar los tiempos de timbrado. El SE contará con una pantalla de cristal líquido (LCD) donde se podrá observar la hora y fecha actual, además servirá para mostrar y configurar las alarmas programadas en las que sonara el timbre, como entradas de usuario se dispondrá un teclado de membrana de 16 teclas mediante el cual

se podrá modificar las alarmas y tiempos de timbrado, para el control del tiempo real se tendrá un circuito integrado que funciona como un reloj de tiempo real el cual contara con una batería de 3 voltios para cuando haya un corte de energía eléctrica este no se resetee, como salida el Sistema Embebido (SE) maneja un relé que funciona como un interruptor controlado por un circuito eléctrico en el que, por medio de una bobina y un electroimán se acciona un juego de uno o varios contactos que permiten abrir o cerrar otros circuitos eléctricos independientes, en este caso activa o desactiva el timbre; y como controlador central del SE se tendrá un microcontrolador funcionando con un firmware diseñado para que el SE funcione correctamente.

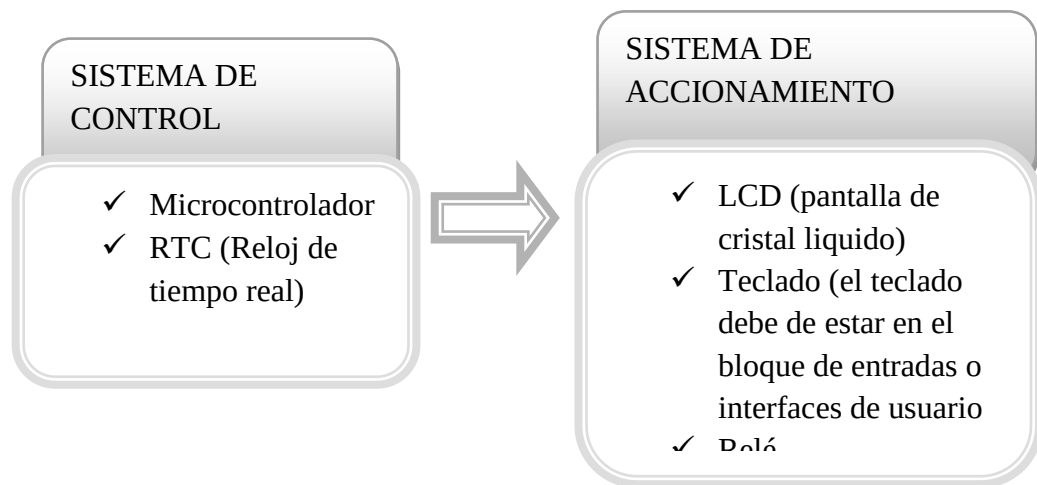


Figura 2.4: Esquema general de las etapas de control y accionamiento.

Sistema de control del proyecto.

El bloque de control del proyecto es el encargado de procesar la información programada para realizar las funciones que desarrollara el prototipo, está compuesto normalmente por un dispositivo digital programable, el componente principal en el proyecto que comandara este proceso es el Microcontrolador ATMEGA 168, que es un chip integrado que en su interior está constituido por un sistema en bloques básico necesario para formar una computadora o un sistema embebido y este lo hace leer, decodificar y actuar por medio de los comandos u órdenes que se le introducen a este chip electrónico, es esta parte si no funciona la parte del cerebro no funcionara el accionamiento y por esto este circuito integrado es muy importante porque es el encargado de que funcione el sistema embebido.

En el bloque de control también se tendrá un circuito integrado DS1307 que se encargada de calcular el tiempo real, y se comunicara con el microcontrolador para que este decida qué hacer, mostrar la hora y/o activar el timbre.

Sistema de entrada.

El proyecto tendrá como entrada, una interface de usuario, un teclado de membrana con 16 teclas, el cual servirá para introducir comandos de configuración de alarmas y de manipular los tiempos de encendido y apagado del timbre.

Sistema de accionamiento.

Es el subsistema que convierte los impulsos recibidos por el sistema de control y lo convierte en sonido. Esta etapa es la encargada de realizar acciones, dichas acciones son realizadas por el relé que a su vez hace funcionar el timbre. Donde el relé juega un papel muy importante ya que es accionado por pequeños impulsos eléctricos y puede trabajar con voltajes mayores, lo que lo hace ideal para combinar su salida con la de un timbre que funciona con un voltaje de 110Voltios, cuando el microcontrolador solo envía 5Voltios.

2.3 MARCO TEÓRICO CONCEPTUAL.

A continuación se presenta un listado de conceptos que se han utilizado en el diseño del prototipo:

- **Sistema:** conjunto de partes o cosas coordinadas por reglamentos diseñados para un fin, o una aplicación.
- **Electrónica:** área de la ingeniería y de la física aplicada al diseño de dispositivos, por lo general circuitos electrónicos, cuyo funcionamiento depende del flujo de electrones haciendo posible la transmisión, recepción almacenamiento de información. esta información consistir en voz (señales de voz) o en música en un receptor de radio, en una imagen en una pantalla de televisión, o en números u otros datos en una computadora.

- **Embebido:** es cuando un conjunto de componentes de electrónica o informática están embebidos o empotrados dentro de un sistema de mayor escala, es decir el pequeño sistema esta embebido o integrado en otro de mayor nivel.
- **Prototipo:** es un primer modelo o molde original que se fabrica y de este modelo se parte para la elaboración de lo que se pretende, un objeto, herramienta, dispositivo electrónico, etc.
- **Hardware:** es la parte tangible, lo que se puede ver y tocar, de un determinado sistema o equipo electrónico o informático.
- **Software:** es la parte interna que en algunos casos no se puede ver y no se puede tocar; es el bloque lógico encargado de indicar que es lo que se debe hacer, esto según lo que se le haya programado al dispositivo.
- **Periférico:** son los dispositivos de entrada y salida que permiten la conexión con el entorno físico, como entradas de información y salidas, es decir donde se muestra la función final del equipo o dispositivo.
- **Firmware:** es un sistema operativo de bajo nivel similar al BIOS en el que todas las aplicaciones más importantes utilizan su funcionalidad. Aunque a diferencia del BIOS en el firmware no hay una versión común disponible, ya que cada firmware es diseñado específicamente para un modelo, y no hay compatibilidad con otras marcas.
- **Memoria:** es un dispositivo diseñado y fabricado específicamente para guardar en ella información. La capacidad dependerá del tipo de equipo que se esté implementando, y puede ser que la memoria sea externa o que ya venga

incorporada en uno de los dispositivos que se están usando, estas pueden ser ROM, PROM, EPROM, EEPROM, RAM, entre otras.

- **CPU:** unidad central de procesamiento, este es un bloque principal dentro de un dispositivo, es el cerebro, se encarga de dar las ordenes a los demás dispositivos, en algunos casos pueden usarse microcontroladores o microprocesadores como parte fundamental del bloque de control.
- **Puertos electrónicos:** son dispositivos físicos que sirven como puertos de conexión que permite la entrada y salida de información al dispositivo, es el medio donde se conecta la entrada y la salida.
- **Fuente de potencia:** es el bloque encargado de proporcionar la potencia eléctrica necesaria para que el equipo funcione correctamente, la fuente está alimentando al equipo durante el tiempo que esta encendido, y además de una fuente también puede utilizarse baterías alcalinas, solo debe tomarse en cuenta cual es la magnitud que se necesita.
- **CLK:** el reloj puede ser interno o externo, este se encarga de mostrar el tiempo en el que trabaja el dispositivo puede ser en tiempo real o no, esto depende de la finalidad del equipo.
- **Teclado:** es a prueba de polvo y suciedad, construido con dos láminas (membranas) plásticas delgadas que contienen circuitos impresos flexibles hechos con tinta conductora de electricidad. La membrana superior es el teclado impreso, y en medio hay una lámina espaciadora con orificios. Cuando el usuario presiona una tecla simulada, la membrana superior es empujada a través del

orificio del espaciador, haciendo contacto con la membrana inferior y completando el circuito.

- **Robótica:** conjunto de conocimientos prácticos que permite obtener, realizar y automatizar sistemas basándose en estructuras metálicas empleando un cierto grado de inteligencia, para que sea posible lograr que un equipo destinado a desempeñarse en un determinado ambiente.
- **Pantalla LCD:** es una lamina que se sujeta delante o alrededor de la luz artificial, en el cual pueden proyectarse imágenes o textos, lo que ofrece una vista bien definida del objeto o imagen que se proyecta en ella. En este caso la pantalla estará formada con matrices de leds.
- **IDE Bascom-AVR:** el ambiente de desarrollo integrado BASCOM-AVR, es un software bastante complejo ya que integra un compilador que convierte el lenguaje Basic a lenguaje maquina haciendo posible que el microcontrolador lo pueda interpretar y ejecutarlo. Además es un programa que proporciona una interfaz que permite descargar el firmware a la memoria del Microcontrolador a través del puertos paralelo de la PC, y también el programa incluye herramientas de ayuda para su utilización, lo que facilita la programación del sistema embebido.
- **Domotica:** la palabra domotica etimológicamente viene de la unión de dos palabras, del latín “DOMUS” que significa casa y AUTOMÁTICA, por lo tanto se denomina como el área de aplicación que tiene por objetivo la automatización

de procesos dentro del hogar, además de ser un área multidisciplinaria que requiere conocimientos de electrónica e informática para su implementación.

- **Circuito lógico digital:** es la interconexión de componentes electrónicos con la característica de que estos componentes trabajan con entradas y salidas de datos digitales, es decir niveles de voltaje discreto de cero o cinco voltios, con su equivalente digital 0 y 1; estos circuitos también se denominan circuitos combinatorios cuyos componentes principales son las distintas compuertas lógicas digitales.
- **Sistema digital programable:** son aquellos sistemas que funcionan igual que un circuito lógico con la diferencia de que la función lógica puede ser cambiada vía programación, el componente principal de estos sistemas son dispositivos programables, por ejemplo un Microcontrolador.

2.4 DOCUMENTACIÓN TÉCNICA.

En la documentación técnica se describen todas las características que poseen cada uno de los componentes que se van a utilizar en este proyecto.

2.4.1 Características principales de Microcontrolador ATMEGA 168.

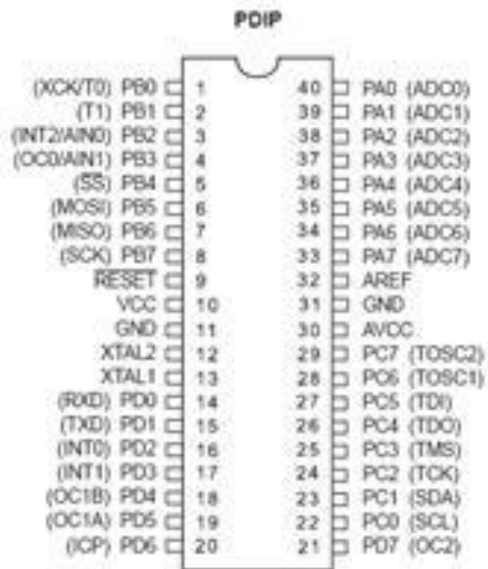


Figura 2.5 pin out del ATMEGA 168

- ✓ Microcontrolador AVR de 8 bits de alto rendimiento y de bajo consumo.
- ✓ Arquitectura RISC avanzada.
- ✓ Características especiales del Microcontrolador.
 - Reset de power - on y detección de Brown -out programable.
 - Oscilador RC interno calibrado.
- ✓ Memoria de programa y de datos no volátil.
 - 8kbite de memoria flash auto programable de sistema.
 - Sección de código añadida opcional con bits de bloqueo.

- ✓ Tensiones de funcionamiento.
 - 2.7 – 5.5 voltios (Atmega 168).
 - 4.5 – 5.5 voltios (Atmega 168).
- ✓ Nivel de velocidad.
 - 0 – 8 MHz (Atmega 168).
 - 0 – 16 MHz (Atmega 168).
- ✓ Consumo de energía a 4 MHz, 3v. 25 °c.
 - activo: 3.6 mA.
 - Modo Idle: 1.0 mA.
 - Modo power – down: 0.5 μ A.

2.4.2 Características generales de la pantalla LCD.

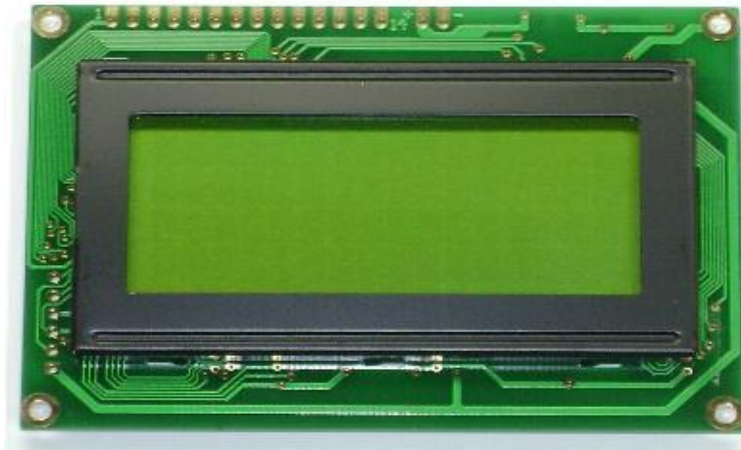


Figura 2.6 Pantalla LCD (Pantalla de cristal liquido) de 16x4

Las pantallas de cristal líquido (LCD), utilizan una interfaz estándar de 14 pines y los que tienen luces de fondo tienen 16 pines, las cuales cuentan con las siguientes características.

- ✓ Pin a tierra.
- ✓ voltaje (3.3 V a 5 V).
- ✓ ajuste de contraste (VO).
- ✓ Seleccione el registro (RS). RS = 0; Comando, RS = 1; Datos.
- ✓ Lectura / escritura (R / W). R / W = 0: escribir, R / W = 1: leer.
- ✓ Reloj (Habilitar). La caída de borde accionado.
- ✓ Bit 0 (no se utiliza en la operación de 4 bits).
- ✓ Bit 1 (no se utiliza en la operación de 4 bits).
- ✓ Bit 2 (no se utiliza en la operación de 4 bits).
- ✓ Bit 3 (no se utiliza en la operación de 4 bits).
- ✓ Bit 4, Bit 5, Bit 6, bit 7.
- ✓ Ánodo de luz de fondo (+).
- ✓ Cátodo de luz de fondo (-).

2.4.3 Características principales del RTC (reloj en tiempo real) DS1307.

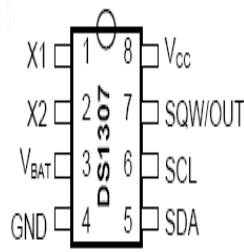


Figura 2.7 Reloj RTC DS1307

- ✓ Cuenta con 56 bytes, con respaldo de batería, no volátil (NV) de RAM para el almacenamiento de datos, dos cables de interfaz serie Señal cuadrada de salida programable.
- ✓ Encendido automático no detectar y cambiar los circuitos.
- ✓ Consume menos de 500nA en el modo de copia de seguridad de la batería con un oscilador de funcionamiento.
- ✓ Rango de temperatura opcional industrial: -40 ° C a +85 ° C.
- ✓ DIP de 8pines Underwriters Laboratory(UL) ha reconocido el DS1307, serie reloj en tiempo real es un bajo consumo de energía, lleno de código binario decimal (BCD) reloj/calendario, más 56bytes de SRAM NV.
- ✓ Dirección y los datos se transfieren a través de una serie de 2 hilos, bidireccional del bus.
- ✓ El reloj / calendario provee segundos, minutos, horas, día, fecha, mes y año.
- ✓ El fin de mes a la fecha se ajusta automáticamente durante meses con menos de 31 días, incluidas las correcciones de los años bisiestos.
- ✓ El reloj funciona tanto en el formato de 24 horas o 12 horas con indicador AM / PM.
- ✓ El DS1307 tiene un circuito integrado de potencia sentido que detecta cortes de energía y cambia automáticamente a la fuente de la batería.

2.4.4 Características principales del teclado de membrana de 16 teclas.

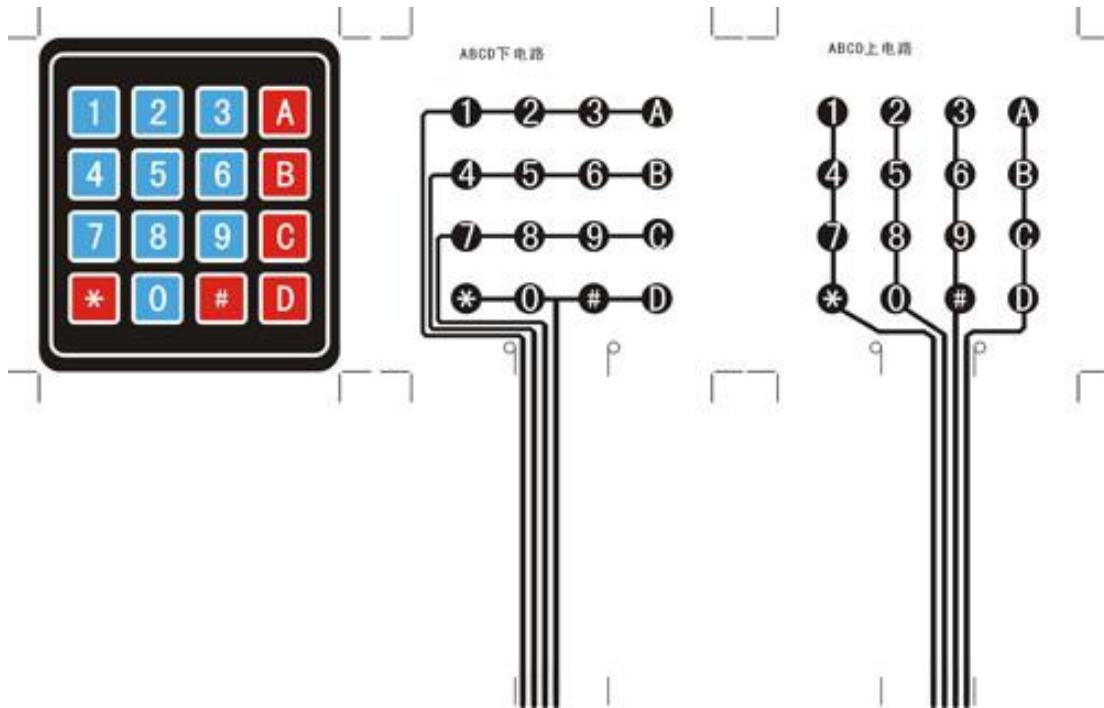


Figura 2.8 Teclado de membrana de 16 teclas adhesivo

- ✓ Cuenta con 16 teclas de membrana.
- ✓ Tamaño Pad: 77 x 70 x 0,8 mm.
- ✓ Transparencia longitud del cable: 86 mm.
- ✓ Peso: 10 gramos.
- ✓ Conector: 7 pines (Paso de 2,54 mm).
- ✓ Montaje: La adhesión.
- ✓ Máxima capacidad del circuito: 35VDC, 100mA.
- ✓ Resistencia de aislamiento: 100 Mh, 100V.
- ✓ Resistencia dieléctrica: 250 Vrms (50-60Hz, 1min).

- ✓ Póngase en contacto con rebote: ≤ 5 ms.
- ✓ Esperanza de vida: 1 millón de cierres.
- ✓ Temperatura de funcionamiento: 0 a 70 grados centígrados.
- ✓ Humedad: 40 grados centígrados, 90% -95%, 240 horas.

2.4.5 Características principales del Relé.

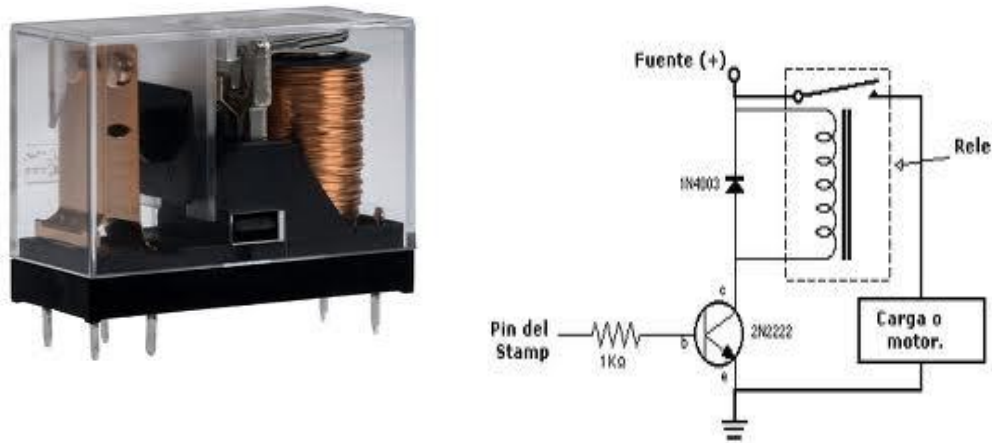


Figura 2.9 Relé de montaje PCB

- ✓ Formulario de contacto: 1a, 1b, 1c.
- ✓ La carga nominal: 10A250VAC/28VDC, 10A125VAC/28VDC, 10 a 125VAC/28VDC.
- ✓ Resistencia de los contactos: $\leq 100(\text{ohm})$.
- ✓ Vida útil eléctrica: 100.000 H.
- ✓ Vida mecánica: 10 millones.
- ✓ Bobina de tensión nominal: 3-48VDC.

- ✓ Potencia de la bobina: 0.36W, 0.45W.
- ✓ Bobina de toma de tensión: $\leq 75\%$.
- ✓ Bobina de abandono de tensión: $\geq 10\%$.
- ✓ Temperatura ambiente: -25°C a $+70$ grados Celsius.
- ✓ Bobina y contactos: 1500VAC/min.
- ✓ Contacto y contactos: 1000VAC/min.
- ✓ Resistencia de aislamiento: $\geq 100\text{ m}(\Omega)$.
- ✓ Forma de montaje: PCB.
- ✓ Peso (g): 10 gramos.
- ✓ Dimensiones externas (mm): 19.0x15.5x 15.0.

2.4.6 Características principales del BUZZER



Características

- Estos zumbadores altos de pieza de la confiabilidad son aplicables en general a equipos de electrónica.
- Condensar, fijar el tipo terminal zumbador de Pieza con 4 kilociclos de salida.
- El perno tipo construcción terminal permite el montaje directo sobre tableros de circuito impresos. C.C. Del Voltaje De Funcionamiento 1.5-27V

Current Consumption 15mA

Frequency 4 KHz +/- 500 Hz

Sound Pressure at 10cm 85 db at 12V DC

Capacitance 17,000 pf

Operating Temperature -20 to +50 deg C

Type Self Oscillating type tone

on DC power, Fixed

Continuous Tone

Mounting PCB

Termination 0.8mm Leads

2.4.7 Características generales del prototipo.

Dentro de las características más importantes de este prototipo encontramos las siguientes:

- Una de las características técnicas más importantes de este dispositivo es que trabaja en tiempo real, gracias al circuito integrado RTC (reloj de tiempo real).
- Estará dotado de una pantalla LCD (pantalla de cristal liquido) que será la encargada de mostrar la hora.
- El consumo de energía será mínimo ya que trabajara con 5V y además contara con una batería interna para no perder las configuraciones, cuando haya un corte de energía.
- Contara con un relé para poder conectar el microcontrolador con el timbre que trabaja a 110V.
- Se ha incorporado un teclado de membrana el cual servirá para realizar configuraciones al sistema.
- Además una de las grandes ventajas de nuestro sistema es que se colocara un interruptor para poder accionar el timbre en horas que no se hayan programado en el *sistema de timbre automatizado*.

CAPITULO III: DESARROLLO DE LA SOLUCIÓN.

3.1 PROPUESTA DE LA SOLUCIÓN.

El objetivo principal del sistema de timbrado automatizado es ser una herramienta de apoyo en la optimización de el tiempo del personal que labora en dicha institución y de la población estudiantil, además de ser preciso y exacto en cuanto al timbrado que delimita el cumplimiento del horario establecido, su implementación permitirá que cada toque de timbre sea automático a la hora establecida por el horario de clases.

¿Cuál es la motivación en implementar dicho proyecto en el centro Escolar Profesor Jesús Leocadio Palencia? Entre los tantos motivos que existen se han identificado los que se consideran principales y entre ellos se encuentra, el hecho de que en muchas ocasiones se ha dado el problema que el personal se encuentra realizando otra tareas quizás de mayor importancia o que requieran de mayor concentración y estas son interrumpidas por el hecho, que el actual sistema de timbre necesita: Que siempre debe haber una persona encargada de activar y desactivar cada timbrado que se hace a lo largo del día, es aquí donde nuestro proyecto vendrá a ser como un trabajador más de la institución con la gran diferencia que no representara costos económicos adicionales, más que la inversión inicial, será fácil de utilizar y de sencillo mantenimiento, además de preciso y exacto y sobre todo optimiza el tiempo de todos los trabajadores y alumnos de la institución.

En Conclusión se podría describir este proyecto como un dispositivo electrónico embebido con la función de asistente electrónico administrador de tiempos.

Este dispositivo estará compuesto de dos etapas: las cuales se describen a continuación:

La etapa de control es la encargada de realizar las comparaciones de las fechas y horas, de las alarmas programadas del sistema, también es la encargada de enviar los datos los impulsos a la etapa de accionamiento entre los dispositivos más importantes encontramos, el microcontrolador ATMEGA 168 que es el encargado de controlar los procesos y de realizar las comparaciones de tiempos y fechas realizando consultas al DS1307 que es el calendario y reloj del sistema, además tiene la función de enviar los impulsos a la etapa de accionamiento. También cuenta con un teclado de membrana de 16 teclas con el que se puede ingresar al menú principal y configurar las alarmas, cambiar la hora y fecha del sistema.

El firmware especialmente diseñado para este proyecto, se descargara en la memoria del microcontrolador y controlara el accionar de este, se ha escrito utilizando el lenguaje BASIC y desarrollado usando el IDE BASCOM-AVR.

La etapa de accionamiento, es la que realizara las acciones a partir de la etapa de control, para este proyecto se tiene una pantalla LCD, BUZZER, RELÉ dicha LCD tiene la función de mostrar la pantalla de bienvenida y de interactuar con el teclado para realizar las configuraciones del sistema, el BUZZER indicara por medio de un sonido cuando se

cambien las alarmas, el RELÉ es el encargado de comunicar el timbre con el microcontrolador por su conexión de baja a alta frecuencia.

Para el diseño del prototipo se tomaron en cuenta muchas variables, dentro de las que se mencionan, que la institución no tiene una persona encargada directamente a esa acción por lo que es difícil controlar dicha tarea y se pierde mucho tiempo.

3.1.1 ALGORITMO.

- Iniciar el programa.
- Configurar e inicializar los parámetros internos del microcontrolador.
- Declaración de variables a utilizar en el programa.

Programa Principal:

- Mostrar pantalla de bienvenida.
- Mostrar la hora y fecha actual.
- Leer teclado, esperar por la tecla #.
- Si la tecla es #, mostrar el menú de opciones.

Fin del Programa Principal.

Menú de Opciones:

A - ver alarmas.

B - cambiar alarmas.

C - cambiar hora.

D - cambiar fecha.

- Leer teclado, esperar por la tecla A, B, C o D.
- Si tecla = A ir a la subrutina mostrar alarmas.
- Si tecla = B ir a la subrutina cambiar alarmas.
- Si tecla = C ir a la subrutina cambiar hora.
- Si tecla = D ir a la subrutina cambiar fecha.

Termina menú de opciones.

Mostrar alarmas:

Si la tecla es A mostrar menú de ver alarmas.

A= Matutino.

B =Vespertino.

Leer teclado, esperar por la tecla A o B.

Si la tecla es A mostrar menú de alarma que desea ver de la 0 a la 7.

- alarma 1, mostrar la alarma 1 del turno matutino.
- alarma 2, mostrar la alarma 2 del turno matutino.
- alarma 3, mostrar la alarma 3 del turno matutino.
- alarma 4, mostrar la alarma 4 del turno matutino.
- alarma 5, mostrar la alarma 5 del turno matutino.
- alarma 6, mostrar la alarma 6 del turno matutino.
- alarma 7, mostrar la alarma 7 del turno matutino.
- alarma 8, mostrar la alarma 8 del turno matutino.

Si la tecla es B mostrar menú de alarma que desea ver de la 0 a la 7.

- alarma 1, mostrar la alarma 1 del turno vespertino.
- alarma 2, mostrar la alarma 2 del turno vespertino.
- alarma 3, mostrar la alarma 3 del turno vespertino.
- alarma 4, mostrar la alarma 4 del turno vespertino.
- alarma 5, mostrar la alarma 5 del turno vespertino.
- alarma 6, mostrar la alarma 6 del turno vespertino.
- alarma 7, mostrar la alarma 7 del turno vespertino.
- alarma 8, mostrar la alarma 8 del turno vespertino.

Cambiar alarmas:

1. Matutino.

Si la tecla es A mostrar menú de alarma que desea modificar de la 0 a la 7.

1. alarma 1, cambiar la alarma 1 del turno matutino.
2. alarma 2, cambiar la alarma 2 del turno matutino.
3. alarma 3, cambiar la alarma 3 del turno matutino.
4. alarma 4, cambiar la alarma 4 del turno matutino.
5. alarma 5, cambiar la alarma 5 del turno matutino.
6. alarma 6, cambiar la alarma 6 del turno matutino.
7. alarma 7, cambiar la alarma 7 del turno matutino.
8. alarma 8, cambiar la alarma 8 del turno matutino.

2. Vespertino.

Si la tecla es A mostrar menú de alarma que desea modificar de la 0 a la 7.

- alarma 1, cambiar la alarma 1 del turno vespertino.
- alarma 2, cambiar la alarma 2 del turno vespertino.
- alarma 3, cambiar la alarma 3 del turno vespertino.
- alarma 4, cambiar la alarma 4 del turno vespertino.
- alarma 5, cambiar la alarma 5 del turno vespertino.
- alarma 6, cambiar la alarma 6 del turno vespertino.
- alarma 7, cambiar la alarma 7 del turno vespertino.
- alarma 8, cambiar la alarma 8 del turno vespertino.

Finalizar sub rutina.

Cambiar hora:

-leer el teclado y modifican los registros de horas, minutos del DS1307.

Fin de la subrutina.

Cambiar fecha:

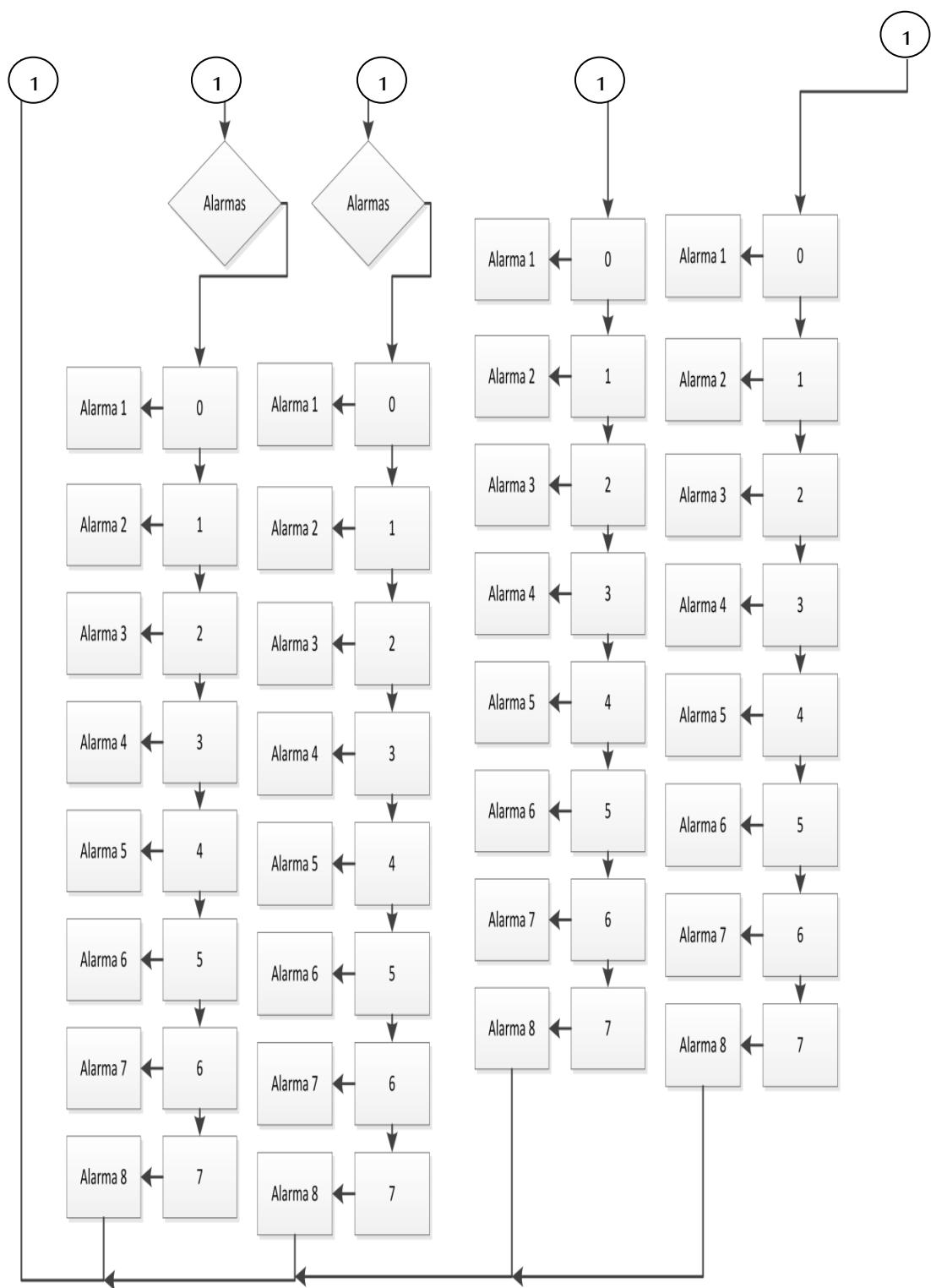
-leer el teclado y modifican los registros de día, mes, año del DS1307.

Fin de la subrutina.

FIGURA 3.1

```
graph TD; Inicio([Inicio]) --> ConfigMicro[Configuración de microcontrolador y cristal]; ConfigMicro --> ConfigPines[Configurar pines del microcontrolador]; ConfigPines --> DimVars[Dimensionar variables a usar]; DimVars --> ConfigLCD[Configurar pantalla LCD]; ConfigLCD --> ConfigDS1307[Configuración del chip DS1307]; ConfigDS1307 --> InicialMenus[Inicializar los menus]; InicialMenus --> DefinAlarms[Definir las alarmas por defecto]; DefinAlarms --> MostrarHora[Mostrar hora y fecha en pantalla]; MostrarHora --> LeerTeclado{Leer el teclado si se presiona #}; LeerTeclado -- No --> MostrarHora; LeerTeclado -- Si --> MenusOperaciones[Mostrar el menú de operaciones<br/>A: Ver alarmas<br/>B: cambiar alarmas<br/>C: cambiar hora<br/>D: cambiar fecha]; MenusOperaciones --> SI_A_1{SI A}; SI_A_1 -- Si A --> PrecionesAB[Precione «A» Matutinas<br/>Preciones «B» Vespertinas]; PrecionesAB --> A_1{A}; A_1 -- SI A --> SeleccionaAlarmaVesp[Selecciona la alarma que desea ver del 0 al 7, del turno Vespertino]; SeleccionaAlarmaVesp --> Alarmas_Vesp{Alarmas}; Alarmas_Vesp -- 1 --> Inicio; A_1 --> SeleccionaAlarmaMat[Selecciona la alarma que desea ver del 0 al 7, del turno Matutino]; SeleccionaAlarmaMat --> Alarmas_Mat{Alarmas}; Alarmas_Mat -- 1 --> Inicio; SI_A_1 --> B_1{B}; B_1 -- SI B --> PrecionesAB; B_1 --> C_1{C}; C_1 -- SI C --> CambiarHora[Cambiar hora]; CambiarHora --> C_2{C}; C_2 -- SI C --> CambiarFecha[Cambiar fecha]; CambiarFecha --> Inicio; C_2 --> SeleccionaAlarmaVesp; SeleccionaAlarmaVesp --> Alarmas_Vesp; Alarmas_Vesp -- 1 --> Inicio; SeleccionaAlarmaMat --> Alarmas_Mat; Alarmas_Mat -- 1 --> Inicio;
```

The flowchart illustrates the initialization and operation of an alarm system. It begins with a sequence of initialization steps: configuring the microcontroller and crystal, setting pins, dimensioning variables, configuring the LCD screen, setting up the DS1307 chip, initializing menus, and defining default alarms. The main loop starts by displaying the time and date. A decision point checks if a key is pressed. If no key is pressed, it loops back to displaying the time and date. If a key is pressed, it shows a menu of operations (A: View alarms, B: Change alarms, C: Change time, D: Change date). From the menu, pressing 'A' leads to another decision point. If 'A' is pressed again, it prompts the user to select an alarm to view for either the morning or evening shift. This selection process involves choosing from 0 to 7 and then viewing the selected alarm status. Pressing 'B' leads to a similar process for changing alarms. Pressing 'C' leads to changing the time, which then leads to changing the date. All paths eventually lead back to the start of the main loop.



3.1.1 FIRMWARE.

'CONTROL DE TIMBRADO AUTOMATICO
'CENTRO ESCOLAR PROFESOR JESUS
'LEOCADIO PALENCIA, DE LA CIUDAD
'DE SAN PABLO TACACHICO
'DEPARTAMENTO DE LA LIBERTAD
'USANDO ATMEGA 168.

***Directivas

\$regfile = "m168def.dat"

\$crystal = 8000000

\$lib "mcsbyte.lbx"

\$lib "ds1307clock.lib"

***Configuraciones

Config Portc = Output

Config Debounce = 30

Config Kbd = Portb

'---configurar el LCD

Config Lcd = 20 * 4

Config Lcdpin = Pin , Db4 = Portd.3 , Db5 = Portd.2 , Db6 = Portd.1 , Db7 = Portd.0 , E
= Portd.4 , Rs = Portd.5

Cls

Cursor Off Noblink

'---configurar los pines I2C

Config Sda = Portd.7

Config Scl = Portd.6

'---configurar direcciones del DS1307

Const Ds1307w = &HD0

Const Ds1307r = &HD1

***Dim variables

Dim Segundos As Byte , Minutos As Byte , Horas As Byte

Dim Dia As Byte , Mes As Byte , Anio As Byte

Dim Horas_12 As Byte

Dim Numero_de_dia As Byte

Dim Meridiano As String * 2

Dim Nombre_de_dia As String * 3

Dim Captura As Byte

Dim Tecla As Byte

```

Dim Num_alarma As Byte
Dim Matutino As String * 8
Dim Vesperti As String * 8
Dim Turno As String * 8
Dim Hrs_alarma As Byte
Dim Min_alarma As Byte
Dim Direccion As Byte
Dim Mer As Byte
Dim Puntero As Byte

```

```

'---arreglo con las alarmas, un byte para la hora , byte para minutos,
'---van seguidas las 20 , 10 por turno, desde 1 a 20.

```

```

'$eprom

```

```

'Alarmas:

```

```

Dim Alarmas(40) As Byte

```

```

'$data

```

```

'***Inicializaciones

```

```

'---las horas se deben especificar en formato de 24HRS

```

Alarmas(1) = 7	'hora, alarma 0
Alarmas(2) = 00	'minutos, alarma 0
Alarmas(3) = 7	'hora, alarma 1
Alarmas(4) = 45	'minutos, alarma 1
Alarmas(5) = 8	'hora, alarma 2
Alarmas(6) = 30	'minutos, alarma 2
Alarmas(7) = 8	'hora, alarma 3
Alarmas(8) = 50	'minutos, alarma 3
Alarmas(9) = 9	'hora, alarma 4
Alarmas(10) = 35	'minutos, alarma 4
Alarmas(11) = 10	'hora, alarma 5
Alarmas(12) = 20	'minutos, alarma 5
Alarmas(13) = 10	'hora, alarma 6
Alarmas(14) = 30	'minutos, alarma 6
Alarmas(15) = 11	'hora, alarma 7
Alarmas(16) = 15	'minutos, alarma 7

Alarmas(17) = 255	'hora, alarma 8
Alarmas(18) = 255	'minutos, alarma 8
Alarmas(19) = 255	'hora, alarma 9
Alarmas(20) = 255	'minutos, alarma 9
'---de la tarde	
Alarmas(21) = 12	'hora, alarma 0
Alarmas(22) = 00	'minutos, alarma 0
Alarmas(23) = 13	'hora, alarma 1

Alarmas(24) = 00	'minutos, alarma 1
Alarmas(25) = 13	'hora, alarma 2
Alarmas(26) = 45	'minutos, alarma 2
Alarmas(27) = 14	'hora, alarma 3
Alarmas(28) = 30	'minutos, alarma 3
Alarmas(29) = 14	'hora, alarma 4
Alarmas(30) = 50	'minutos, alarma 4
Alarmas(31) = 15	'hora, alarma 5
Alarmas(32) = 35	'minutos, alarma 5
Alarmas(33) = 16	'hora, alarma 6
Alarmas(34) = 20	'minutos, alarma 6
Alarmas(35) = 16	'hora, alarma 7
Alarmas(36) = 40	'minutos, alarma 7
Alarmas(37) = 17	'hora, alarma 8
Alarmas(38) = 25	'minutos, alarma 8
Alarmas(39) = 255	'hora, alarma 9
Alarmas(40) = 255	'minutos, alarma 9
Alarmas(41) = 255	'hora, alarma 10
Alarmas(42) = 255	'minutos, alarma 10

```

Matutino = "Matutino"
Vesperti = "Vesperti"
Turno = "xxxxxxx"
Direccion = 1
Gosub Leer_ds1307
Dia = 28 : Mes = 6 : Anio = 11 : Numero_de_dia = 4
Horas = 6 : Minutos = 59 : Segundos = 40
Gosub Ini_ds1307

```

```

Upperline
Lcd "*****"
Lowerline
Lcd "* TRABAJO DE GRAD *"
Thirdline
Lcd "* Tec Hardware *"
Fourthline
Lcd "*****"
Wait 3
Cls
Upperline
Lcd "*****"

```

```

Lowerline
Lcd "* Sistema Timbre *"
Thirdline
Lcd "* Automatico 2011 *"
Fourthline
Lcd "*****"
Wait 3

```

```

***Programa Principal
Principal:
  Gosub Leer_ds1307
  Gosub Mostrar
  Captura = Getkbd()
  If Captura <> 16 Then Gosub Cual_tecla
  If Tecla = 11 Then Gosub Menu_1
  If Numero_de_dia <> 7 And Numero_de_dia <> 1 Then Gosub Compa_alar
  Goto Principal
End
***Fin del Programa Principal

```

```

***Tablas de Datos
Teclas:
'--- *=10 #=11 A=12 B=13 C=14 D=15
  Data 1 , 4 , 7 , 10 , 2 , 5 , 8 , 0 , 3 , 6 , 9 , 11 , 12 , 13 , 14 , 15

```

```

***Subrutinas
'-----
Compa_alar:
'---verificar_alarm

  Hrs_alarma = Alarmas(direccion)
  Incr Direccion
  Min_alarma = Alarmas(direccion)

  If Horas = Hrs_alarma And Minutos = Min_alarma And Segundos < 7 Then

    Cls
    Upperline
    Lcd "-----"
    Lowerline

```

```

    Lcd "*****TIMBRE*****"
    Portc.0 = 1
    Thirdline
    Lcd "*****ACTIVADO*****"
    Fourthline
    Lcd "-----"
    Sound Portc.1 , 300 , 150
    Wait 4
    Portc.0 = 0
End If
Incr Direccion
If Direccion > 40 Then Direccion = 1
Return
'-----
'-----
Menu_1:
'---menu de opciones a configurar
    Tecla = 16
    Upperline
    Lcd "- Presione opcion -"
    Lowerline
    Lcd "A-Ver_Ala B-Cam_Ala"
    Thirdline
    Lcd "C-Cam_Hor D-Cam-Fec"
    Fourthline
    Lcd "    salir *    "
    Captura = Getkbd()
    If Captura <> 16 Then Gosub Cual_tecla
    If Tecla = 10 Then Goto Principal
    If Tecla = 12 Then Goto Ver_alarmas
    If Tecla = 13 Then Goto Cambiar_alarmas
    If Tecla = 14 Then Goto Cambiar_hora
    If Tecla = 15 Then Goto Cambiar_fecha
    Goto Menu_1
Return
'-----
'-----
Cambiar_alarmas:
'---ver alarmas seteadas, son 10 por turno
    Tecla = 16
    Upperline
    Lcd " Cambiar Alarmas "
    Lowerline
    Lcd " A - Matutino    "

```

```

Thirdline
Lcd " B - Vespertino  "
Fourthline
Lcd "   salir *   "
Captura = Getkbd()
If Captura <> 16 Then Gosub Cual_tecla
If Tecla = 10 Then Goto Menu_1
If Tecla = 12 Then Goto Cambiar_matutino
If Tecla = 13 Then Goto Cambiar_vespertino
Goto Cambiar_alarmas

```

```

'-----
'-----

```

```

Cambiar_matutino:
'---10 alarmas, 0 a 9, lunes a viernes
Tecla = 16
Turno = "matutino"
Upperline
Lcd "Matutino son 8 alarm"
Lowerline
Lcd " lunes a viernes  "
Thirdline
Lcd "alarma cambiar? 0-7 "
Fourthline
Lcd "   salir *   "
Captura = Getkbd()
If Captura <> 16 Then Gosub Cual_tecla
If Tecla = 10 Then Goto Cambiar_alarmas
'If Tecla = 0 Then
'  Num_alarma = 0
'  Goto Ver_una_alarma
'End If
Select Case Tecla
Case 0 : Num_alarma = 1
        Goto Cambiar_una_alarma
Case 1 : Num_alarma = 3
        Goto Cambiar_una_alarma
Case 2 : Num_alarma = 5
        Goto Cambiar_una_alarma
Case 3 : Num_alarma = 7
        Goto Cambiar_una_alarma
Case 4 : Num_alarma = 9
        Goto Cambiar_una_alarma
Case 5 : Num_alarma = 11

```

```

        Goto Cambiar_una_alarma
    Case 6 : Num_alarma = 13
        Goto Cambiar_una_alarma
    Case 7 : Num_alarma = 15
        Goto Cambiar_una_alarma
    Case 8 : Num_alarma = 17
        Goto Cambiar_una_alarma
    Case 9 : Num_alarma = 19
        Goto Cambiar_una_alarma
End Select
Goto Cambiar_matutino
'-----
'-----
Cambiar_vespertino:
'---10 alarmas, 0 a 9, lunes a viernes
Tecla = 16
Turno = "matutino"
Upperline
Lcd "Vesperti son 8 alarm"
Lowerline
Lcd " lunes a viernes  "
Thirdline
Lcd "alarma cambiar? 0-7 "
Fourthline
Lcd "   salir *   "
Captura = Getkbd()
If Captura <> 16 Then Gosub Cual_tecla
If Tecla = 10 Then Goto Cambiar_alarmas

Select Case Tecla
Case 1 : Num_alarma = 1
    Goto Cambiar_una_alarma
Case 2 : Num_alarma = 3
    Goto Cambiar_una_alarma
Case 3 : Num_alarma = 5
    Goto Cambiar_una_alarma
Case 4 : Num_alarma = 7
    Goto Cambiar_una_alarma
Case 5 : Num_alarma = 9
    Goto Cambiar_una_alarma
Case 6 : Num_alarma = 11
    Goto Cambiar_una_alarma
Case 7 : Num_alarma = 13
    Goto Cambiar_una_alarma

```

```

        Case 8 : Num_alarma = 15
            Goto Cambiar_una_alarma
        Case 9 : Num_alarma = 17
            Goto Cambiar_una_alarma
        Case 0 : Num_alarma = 19
            Goto Cambiar_una_alarma
    End Select
Goto Cambiar_vespertino
'-----
'-----
Cambiar_una_alarma:
'---10 alarmas, 0 a 9, lunes a viernes
    Tecla = 16
    Direccion = Num_alarma
    Puntero = Direccion
    Hrs_alarma = Alarmas(direccion)
    Incr Direccion
    Min_alarma = Alarmas(direccion)
    'If Num_alarma <= 20 Then
        Direccion = Num_alarma / 2
        Incr Direccion
        Num_alarma = Num_alarma - Direccion
        If Num_alarma > 10 Then Num_alarma = Num_alarma - 10
        'If Num_alarma > 20 Then Num_alarma = Num_alarma - 11

    '---¿am o pm?
    If Hrs_alarma = 0 Then
        Hrs_alarma = 12
        Meridiano = "AM"
    Else
        'Hrs_alarma = Hrs_alarma
        Meridiano = "AM"
    End If

    If Hrs_alarma = 12 Then Meridiano = "PM"

    If Hrs_alarma > 12 Then
        Hrs_alarma = Hrs_alarma - 12
        Meridiano = "PM"
    End If
    If Meridiano = "PM" Then Mer = 1
    If Meridiano = "AM" Then Mer = 0
Cambiar_alarma:
    If Meridiano = "PM" Then Mer = 1

```

```

If Meridiano = "AM" Then Mer = 0
Upperline
Lcd "Alarma " ; Num_alarma ; " L-V " ; Turno
Lowerline
Lcd " 2.HRS 5.MIN 8.MER"
Thirdline
Lcd "   " ; Hrs_alarma ; ":" ; Min_alarma ; " " ; Meridiano ; "   "
Fourthline
Lcd " Salir *      OK #   "
Captura = Getkbd()
If Captura <> 16 Then Gosub Cual_tecla
If Tecla = 10 Then Goto Cambiar_alarmas
If Tecla = 11 Then Goto Confirmar_cambio_alarma

If Tecla = 2 Then Incr Hrs_alarma
If Tecla = 5 Then Incr Min_alarma

If Tecla = 8 Then
    If Mer = 0 Then Meridiano = "PM"
    If Mer = 1 Then Meridiano = "AM"
End If
If Tecla = 8 And Meridiano = "AM" Then Meridiano = "PM"
Tecla = 16
Goto Cambiar_alarma
'-----

```

Confirmar_cambio_alarma:

```

If Mer = 1 Then Hrs_alarma = Hrs_alarma + 12

```

```

Alarmas(puntero) = Hrs_alarma
Incr Puntero
Alarmas(puntero) = Min_alarma

```

```

Upperline
Lcd "*****"
Lowerline
Lcd "*   Alarma No. " ; Num_alarma ; "   *"
Thirdline
Lcd "*   MODIFICADA   *"
Fourthline
Lcd "*****"
Wait 3

```

Goto Cambiar_alarmas

```
'-----
Cambiar_hora:
'---cambiar la hora actual
    Tecla = 16
    Upperline
    Lcd " Cambiar Hora  "
    Lowerline
    Lcd "2. +HRS    8. +MIN"
    '---¿am o pm?
    Gosub Leer_ds1307
    If Horas = 0 Then
        Horas_12 = 12
        Meridiano = "AM"
    Else
        Horas_12 = Horas
        Meridiano = "AM"
    End If
    If Horas = 12 Then Meridiano = "PM"
    If Horas > 12 Then
        Horas_12 = Horas - 12
        Meridiano = "PM"
    End If
'---mostrar hora
    If Horas_12 < 10 And Minutos < 10 And Segundos < 10 Then
        Thirdline : Lcd "  " ; "  " ; Horas_12 ; ":0" ; Minutos ; ":0" ; Segundos ; "  " ;
        Meridiano ; "  "
    End If
    If Horas_12 < 10 And Minutos < 10 And Segundos > 9 Then
        Thirdline : Lcd "  " ; "  " ; Horas_12 ; ":0" ; Minutos ; ":" ; Segundos ; "  " ;
        Meridiano ; "  "
    End If
    If Horas_12 < 10 And Minutos > 9 And Segundos < 10 Then
        Thirdline : Lcd "  " ; "  " ; Horas_12 ; ":" ; Minutos ; ":0" ; Segundos ; "  " ;
        Meridiano ; "  "
    End If
    If Horas_12 < 10 And Minutos > 9 And Segundos > 9 Then
        Thirdline : Lcd "  " ; "  " ; Horas_12 ; ":" ; Minutos ; ":" ; Segundos ; "  " ;
        Meridiano ; "  "
    End If

    If Horas_12 > 9 And Minutos < 10 And Segundos < 10 Then
```

```

Thirdline : Lcd "  "; Horas_12 ; ":0" ; Minutos ; ":0" ; Segundos ; "  " ; Meridiano
; "  "
End If
If Horas_12 > 9 And Minutos < 10 And Segundos > 9 Then
Thirdline : Lcd "  "; Horas_12 ; ":0" ; Minutos ; ":" ; Segundos ; "  " ; Meridiano ;
"  "
End If
If Horas_12 > 9 And Minutos > 9 And Segundos < 10 Then
Thirdline : Lcd "  "; Horas_12 ; ":" ; Minutos ; ":0" ; Segundos ; "  " ; Meridiano ;
"  "
End If
If Horas_12 > 9 And Minutos > 9 And Segundos > 9 Then
Thirdline : Lcd "  "; Horas_12 ; ":" ; Minutos ; ":" ; Segundos ; "  " ; Meridiano ;
"  "
End If

```

```

Fourthline
Lcd "  salir *  "
Captura = Getkbd()
If Captura <> 16 Then Gosub Cual_tecla
If Tecla = 10 Then Goto Menu_1
If Tecla = 2 Then Gosub Set_horas
If Tecla = 8 Then Gosub Set_minutos

```

Goto Cambiar_hora

```

'-----
'-----

```

Ver_alarmas:

```

'---ver alarmas seteadas, son 10 por turno
Tecla = 16
Upperline
Lcd "VER alar - 8/turno"
Lowerline
Lcd " A - Matutino  "
Thirdline
Lcd " B - Vespertino  "
Fourthline
Lcd "  salir *  "
Captura = Getkbd()
If Captura <> 16 Then Gosub Cual_tecla
If Tecla = 10 Then Goto Menu_1
If Tecla = 12 Then Goto Ver_matutino
If Tecla = 13 Then Goto Ver_vespertino
If Tecla = 12 Then Goto Cambiar_alarma

```

```

        Goto Ver_alarmas
Return
'-----
'-----
Cambiar_fecha:
'---cambiar la fecha actual
    Tecla = 16
    Upperline
    Lcd "  Cambiar Fecha  "
    Lowerline
    Lcd "2.DI 5.ME 8.AN 0.NO"
    Gosub Leer_ds1307
    '...¿nombre_de_dia?
    If Numero_de_dia = 1 Then Nombre_de_dia = "DO"
    If Numero_de_dia = 2 Then Nombre_de_dia = "LU"
    If Numero_de_dia = 3 Then Nombre_de_dia = "MA"
    If Numero_de_dia = 4 Then Nombre_de_dia = "MI"
    If Numero_de_dia = 5 Then Nombre_de_dia = "JU"
    If Numero_de_dia = 6 Then Nombre_de_dia = "VI"
    If Numero_de_dia = 7 Then Nombre_de_dia = "SA"
    '---mostrar fecha
    If Dia < 10 And Mes < 10 And Anio < 10 Then
        Thirdline : Lcd "  " ; Dia ; "/" ; Mes ; "/" ; Anio ; " " ; Nombre_de_dia ; "  "
    End If
    If Dia < 10 And Mes < 10 And Anio > 9 Then
        Thirdline : Lcd "  " ; Dia ; "/" ; Mes ; "/" ; Anio ; " " ; Nombre_de_dia ; "  "
    End If
    If Dia < 10 And Anio < 10 And Mes > 9 Then
        Thirdline : Lcd "  " ; Dia ; "/" ; Mes ; "/" ; Anio ; " " ; Nombre_de_dia ; "  "
    End If
    If Dia < 10 And Anio > 9 And Mes > 9 Then
        Thirdline : Lcd "  " ; Dia ; "/" ; Mes ; "/" ; Anio ; " " ; Nombre_de_dia ; "  "
    End If
    If Dia > 9 And Mes < 10 And Anio < 10 Then
        Thirdline : Lcd "  " ; Dia ; "/" ; Mes ; "/" ; Anio ; " " ; Nombre_de_dia ; "  "
    End If
    If Dia > 9 And Mes < 10 And Anio > 9 Then
        Thirdline : Lcd "  " ; Dia ; "/" ; Mes ; "/" ; Anio ; " " ; Nombre_de_dia ; "  "
    End If
    If Dia > 9 And Anio < 10 And Mes > 9 Then
        Thirdline : Lcd "  " ; Dia ; "/" ; Mes ; "/" ; Anio ; " " ; Nombre_de_dia ; "  "
    End If
    If Dia > 9 And Anio > 9 And Mes > 9 Then
        Thirdline : Lcd "  " ; Dia ; "/" ; Mes ; "/" ; Anio ; " " ; Nombre_de_dia ; "  "

```

```

End If
Waitms 250
Fourthline
Lcd "  salir *  "
Captura = Getkbd()
If Captura <> 16 Then Gosub Cual_tecla
If Tecla = 10 Then Goto Menu_1
If Tecla = 2 Then Gosub Set_dia
If Tecla = 5 Then Gosub Set_mes
If Tecla = 8 Then Gosub Set_anio
If Tecla = 0 Then Gosub Set_dia_sem

```

```

Goto Cambiar_fecha

```

```

'-----
'-----
'-----
'-----

```

```

Ver_matutino:

```

```

'---10 alarmas, 0 a 9, lunes a viernes

```

```

    Tecla = 16

```

```

    Turno = "matutino"

```

```

Upperline

```

```

Lcd "Matutino son 8 alar"

```

```

Lowerline

```

```

Lcd " lunes a viernes  "

```

```

Thirdline

```

```

Lcd " alarma a ver? 0 a 7"

```

```

Fourthline

```

```

Lcd "  salir *  "

```

```

Captura = Getkbd()

```

```

If Captura <> 16 Then Gosub Cual_tecla

```

```

If Tecla = 10 Then Goto Ver_alarmas

```

```

'If Tecla = 0 Then

```

```

'  Num_alarma = 0

```

```

'  Goto Ver_una_alarma

```

```

'End If

```

```

Select Case Tecla

```

```

Case 0 : Num_alarma = 1

```

```

    Goto Ver_una_alarma

```

```

Case 1 : Num_alarma = 3

```

```

    Goto Ver_una_alarma

```

```

Case 2 : Num_alarma = 5

```

```

    Goto Ver_una_alarma

```

```

Case 3 : Num_alarma = 7

```

```

        Goto Ver_una_alarma
Case 4 : Num_alarma = 9
        Goto Ver_una_alarma
Case 5 : Num_alarma = 11
        Goto Ver_una_alarma
Case 6 : Num_alarma = 13
        Goto Ver_una_alarma
Case 7 : Num_alarma = 15
        Goto Ver_una_alarma
Case 8 : Num_alarma = 17
        Goto Ver_una_alarma
Case 9 : Num_alarma = 19
        Goto Ver_una_alarma
End Select
Goto Ver_matutino
'-----
'-----
Ver_vespertino:
'---10 alarmas, 0 a 9, lunes a viernes
    Tecla = 16
    Turno = "vesperti"
Upperline
Lcd "Vespertino- 8 almrs "
Lowerline
Lcd " lunes a viernes  "
Thirdline
Lcd " alarma a ver? 0 a 7"
Fourthline
Lcd " salir *      "
Captura = Getkbd()
If Captura <> 16 Then Gosub Cual_tecla
If Tecla = 10 Then Goto Ver_alarmas
'If Tecla = 0 Then
' Num_alarma = 0
' Goto Ver_una_alarma
'End If
Select Case Tecla
Case 0 : Num_alarma = 21
        Goto Ver_una_alarma
Case 1 : Num_alarma = 23
        Goto Ver_una_alarma
Case 2 : Num_alarma = 25
        Goto Ver_una_alarma
Case 3 : Num_alarma = 27

```

```

        Goto Ver_una_alarma
    Case 4 : Num_alarma = 29
        Goto Ver_una_alarma
    Case 5 : Num_alarma = 31
        Goto Ver_una_alarma
    Case 6 : Num_alarma = 33
        Goto Ver_una_alarma
    Case 7 : Num_alarma = 35
        Goto Ver_una_alarma
    Case 8 : Num_alarma = 37
        Goto Ver_una_alarma
    Case 9 : Num_alarma = 39
        Goto Ver_una_alarma
End Select
Goto Ver_vespertino
'-----
'-----
Ver_una_alarma:
'---10 alarmas, 0 a 9, lunes a viernes
    Tecla = 16
    'If Num_alarma = 0 Then Direccion = Num_alarma
    'If Num_alarma > 0 Then Direccion = Num_alarma * 2
    Direccion = Num_alarma
    Hrs_alarma = Alarmas(direccion)
    Incr Direccion
    Min_alarma = Alarmas(direccion)
    'If Num_alarma <= 20 Then
    Direccion = Num_alarma / 2
    Incr Direccion
    Num_alarma = Num_alarma - Direccion
    If Num_alarma > 10 Then Num_alarma = Num_alarma - 10
    'If Num_alarma > 20 Then Num_alarma = Num_alarma - 11

    '---¿am o pm?
    If Hrs_alarma = 0 Then
        Hrs_alarma = 12
        Meridiano = "AM"
    Else
        'Hrs_alarma = Hrs_alarma
        Meridiano = "AM"
    End If
    If Hrs_alarma = 12 Then Meridiano = "PM"
    If Hrs_alarma > 12 Then
        Hrs_alarma = Hrs_alarma - 12

```

```

Meridiano = "PM"
End If

```

Veralarma:

```

Upperline
Lcd " Alarma No " ; Num_alarma ; " "
Lowerline
Lcd "lun a vie / " ; Turno
Thirdline
Lcd " " ; Hrs_alarma ; ":" ; Min_alarma ; " " ; Meridiano ; " "
Fourthline
Lcd " salir * "
Captura = Getkbd()
If Captura <> 16 Then Gosub Cual_tecla
If Tecla = 10 Then Goto Ver_alarmas
Goto Veralarma

```

'-----

'-----

```

Cual_tecla:
'---verificar cual tecla se presiono
Sound Portc.1 , 150 , 150
Tecla = Lookup(captura , Teclas)
'Lcd Tecla

```

Return

'-----

```

Ini_ds1307:
'---inicializar ds1307

```

```

Segundos = Makebcd(segundos) : Minutos = Makebcd(minutos) : Horas =
Makebcd(horas)
Dia = Makebcd(dia) : Mes = Makebcd(mes) : Anio = Makebcd(anio) :
Numero_de_dia = Makebcd(numero_de_dia)
I2cstart
I2cwbyte Ds1307w
I2cwbyte 0
I2cwbyte Segundos
I2cwbyte Minutos
I2cwbyte Horas
I2cwbyte Numero_de_dia
I2cwbyte Dia
I2cwbyte Mes

```

```

    I2cwbyte Anio
    I2cstop
Return

```

```

'-----
Leer_ds1307:
    I2cstart
    I2cwbyte Ds1307w
    I2cwbyte 0
    I2cstart
    I2cwbyte Ds1307r
    I2crbyte Segundos , Ack
    I2crbyte Minutos , Ack
    I2crbyte Horas , Ack
    I2crbyte Numero_de_dia , Ack
    I2crbyte Dia , Ack
    I2crbyte Mes , Ack
    I2crbyte Anio , Nack
    I2cstop
    Segundos = Makedec(segundos) : Minutos = Makedec(minutos) : Horas =
Makedec(horas)
    Dia = Makedec(dia) : Mes = Makedec(mes) : Anio = Makedec(anio) :
Numero_de_dia = Makedec(numero_de_dia)
Return

```

```

'-----
'-----
Mostrar:
'---¿am o pm?
    If Horas = 0 Then
        Horas_12 = 12
        Meridiano = "AM"
    Else
        Horas_12 = Horas
        Meridiano = "AM"
    End If
    If Horas = 12 Then Meridiano = "PM"
    If Horas > 12 Then
        Horas_12 = Horas - 12
        Meridiano = "PM"
    End If
'...¿nombre_de_dia?
    If Numero_de_dia = 1 Then Nombre_de_dia = "DO"
    If Numero_de_dia = 2 Then Nombre_de_dia = "LU"

```

```

    If Numero_de_dia = 3 Then Nombre_de_dia = "MA"
    If Numero_de_dia = 4 Then Nombre_de_dia = "MI"
    If Numero_de_dia = 5 Then Nombre_de_dia = "JU"
    If Numero_de_dia = 6 Then Nombre_de_dia = "VI"
    If Numero_de_dia = 7 Then Nombre_de_dia = "SA"
'---Mensajes
Upperline : Lcd "   C.E. P. J. L. PALENCIA  "
Fourthline : Lcd "       menu #       "
'---mostrar hora
    If Horas_12 < 10 And Minutos < 10 And Segundos < 10 Then
        Lowerline : Lcd "   " ; "   " ; Horas_12 ; ":0" ; Minutos ; ":0" ; Segundos ; "   " ;
Meridiano ; "   "
    End If
    If Horas_12 < 10 And Minutos < 10 And Segundos > 9 Then
        Lowerline : Lcd "   " ; "   " ; Horas_12 ; ":0" ; Minutos ; ":" ; Segundos ; "   " ;
Meridiano ; "   "
    End If
    If Horas_12 < 10 And Minutos > 9 And Segundos < 10 Then
        Lowerline : Lcd "   " ; "   " ; Horas_12 ; ":" ; Minutos ; ":0" ; Segundos ; "   " ;
Meridiano ; "   "
    End If
    If Horas_12 < 10 And Minutos > 9 And Segundos > 9 Then
        Lowerline : Lcd "   " ; "   " ; Horas_12 ; ":" ; Minutos ; ":" ; Segundos ; "   " ;
Meridiano ; "   "
    End If
    If Horas_12 > 9 And Minutos < 10 And Segundos < 10 Then
        Lowerline : Lcd "   " ; Horas_12 ; ":0" ; Minutos ; ":0" ; Segundos ; "   " ;
Meridiano ; "   "
    End If
    If Horas_12 > 9 And Minutos < 10 And Segundos > 9 Then
        Lowerline : Lcd "   " ; Horas_12 ; ":0" ; Minutos ; ":" ; Segundos ; "   " ; Meridiano
; "   "
    End If
    If Horas_12 > 9 And Minutos > 9 And Segundos < 10 Then
        Lowerline : Lcd "   " ; Horas_12 ; ":" ; Minutos ; ":0" ; Segundos ; "   " ; Meridiano
; "   "
    End If
    If Horas_12 > 9 And Minutos > 9 And Segundos > 9 Then
        Lowerline : Lcd "   " ; Horas_12 ; ":" ; Minutos ; ":" ; Segundos ; "   " ; Meridiano ;
"   "
    End If
'---mostrar fecha
    If Dia < 10 And Mes < 10 And Anio < 10 Then
        Thirdline : Lcd "   " ; Dia ; "/0" ; Mes ; "/0" ; Anio ; "   " ; Nombre_de_dia ; "   "

```

```

End If
If Dia < 10 And Mes < 10 And Anio > 9 Then
  Thirdline : Lcd "  "; Dia ; "/" ; Mes ; "/" ; Anio ; " "; Nombre_de_dia ; " "
End If
If Dia < 10 And Anio < 10 And Mes > 9 Then
  Thirdline : Lcd "  "; Dia ; "/" ; Mes ; "/" ; Anio ; " "; Nombre_de_dia ; " "
End If
If Dia < 10 And Anio > 9 And Mes > 9 Then
  Thirdline : Lcd "  "; Dia ; "/" ; Mes ; "/" ; Anio ; " "; Nombre_de_dia ; " "
End If
If Dia > 9 And Mes < 10 And Anio < 10 Then
  Thirdline : Lcd "  "; Dia ; "/" ; Mes ; "/" ; Anio ; " "; Nombre_de_dia ; " "
End If
If Dia > 9 And Mes < 10 And Anio > 9 Then
  Thirdline : Lcd "  "; Dia ; "/" ; Mes ; "/" ; Anio ; " "; Nombre_de_dia ; " "
End If
If Dia > 9 And Anio < 10 And Mes > 9 Then
  Thirdline : Lcd "  "; Dia ; "/" ; Mes ; "/" ; Anio ; " "; Nombre_de_dia ; " "
End If
If Dia > 9 And Anio > 9 And Mes > 9 Then
  Thirdline : Lcd "  "; Dia ; "/" ; Mes ; "/" ; Anio ; " "; Nombre_de_dia ; " "
End If
Waitms 250
Return

```

```

'-----
'-----

```

```

Set_minutos:
  Incr Minutos
  If Minutos > 59 Then Minutos = 0
  Minutos = Makebcd(minutos)
  I2cstart
  I2cwbyte Ds1307w
  I2cwbyte 1
  I2cwbyte Minutos
  I2cstop

```

```

Return
'-----

```

```

Set_horas:
  Incr Horas
  If Horas > 23 Then Horas = 0
  Horas = Makebcd(horas)
  I2cstart
  I2cwbyte Ds1307w

```

```

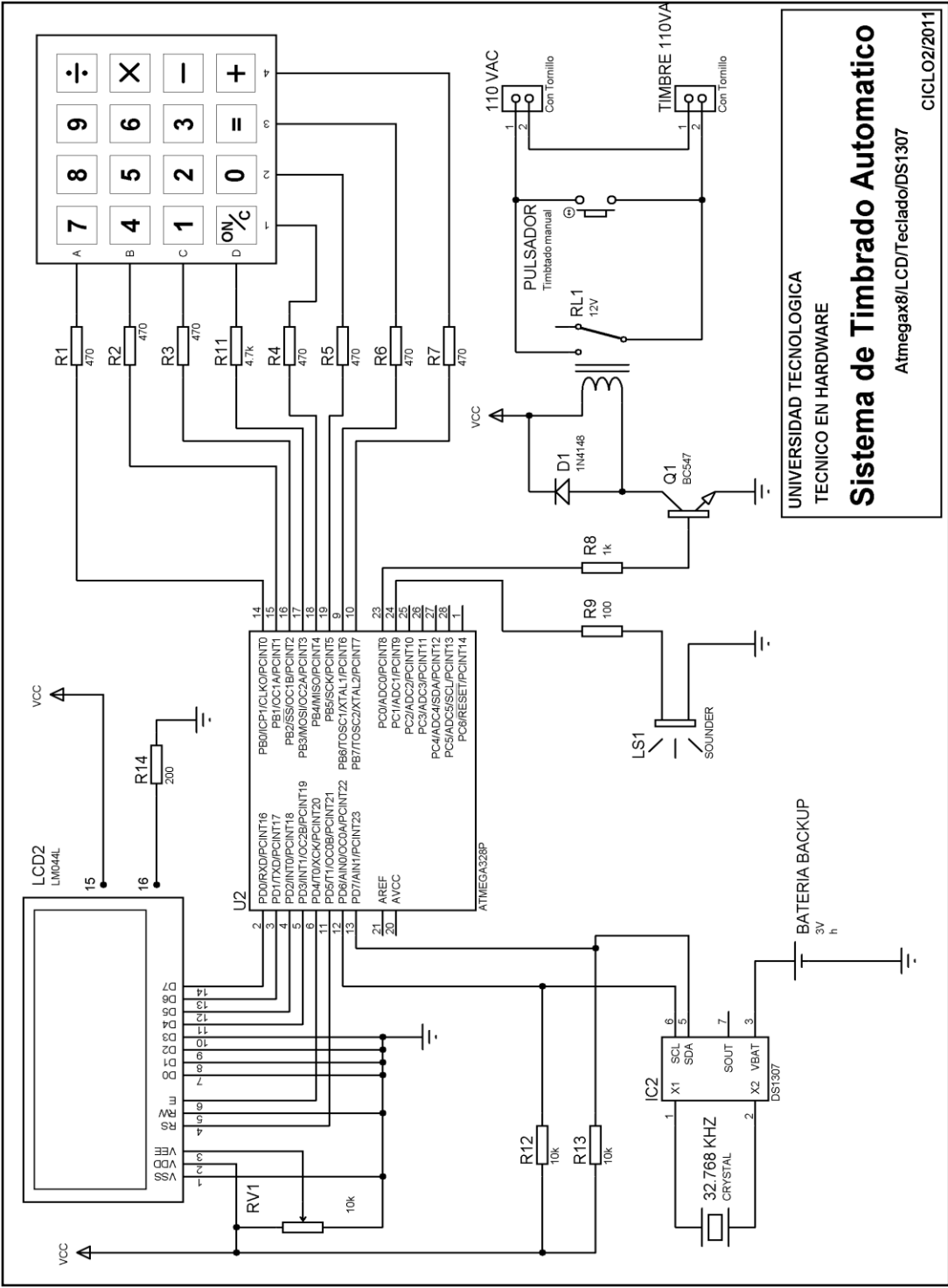
        I2cwbyte 2
        I2cwbyte Horas
        I2cstop
Return
'-----
Set_dia:
    Incr Dia
    If Mes = 1 Or Mes = 3 Or Mes = 5 Or Mes = 7 Or Mes = 8 Or Mes = 10 Or Mes =
12 And Dia > 31 Then Dia = 1
    If Mes = 4 Or Mes = 6 Or Mes = 9 Or Mes = 11 And Dia > 30 Then Dia = 1
    If Mes = 2 And Dia > 28 Then Dia = 1
    Dia = Makebcd(dia)
    I2cstart
    I2cwbyte Ds1307w
    I2cwbyte 4
    I2cwbyte Dia
    I2cstop
Return
'-----
Set_mes:
    Incr Mes
    If Mes > 12 Then Mes = 1
    Mes = Makebcd(mes)
    I2cstart
    I2cwbyte Ds1307w
    I2cwbyte 5
    I2cwbyte Mes
    I2cstop
Return
'-----
Set_anio:
    Incr Anio
    If Anio > 99 Then Anio = 0
    Anio = Makebcd(anio)
    I2cstart
    I2cwbyte Ds1307w
    I2cwbyte 6
    I2cwbyte Anio
    I2cstop
Return
'-----
Set_dia_sem:
    Incr Numero_de_dia
    If Numero_de_dia > 7 Then Numero_de_dia = 1

```

```
Numero_de_dia = Makebcd(numero_de_dia)
I2cstart
I2cwbyte Ds1307w
I2cwbyte 3
I2cwbyte Numero_de_dia
I2cstop
Return
'-----
'-----
```

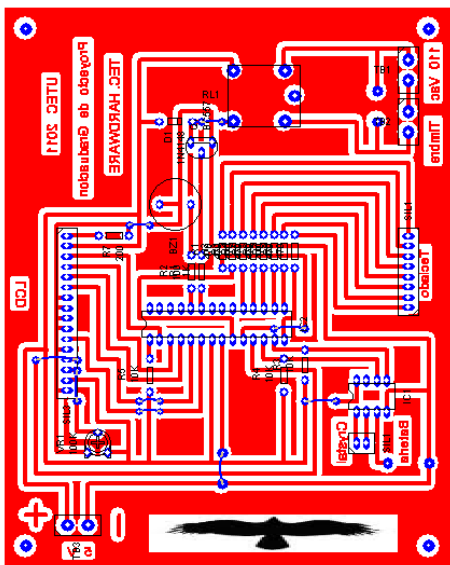
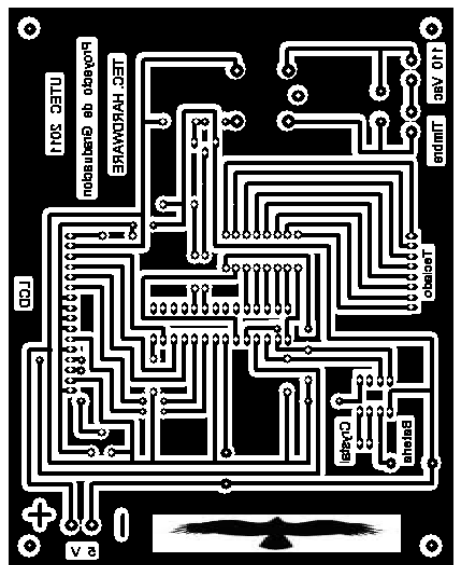
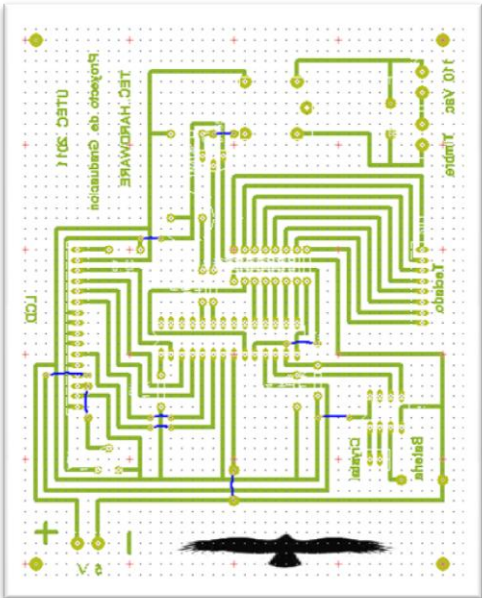
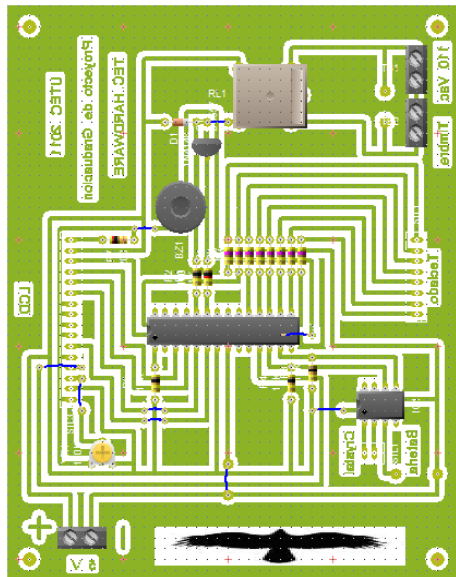
3.1.4 DIAGRAMA.

FIGURA 3.2



3.1.5 PCB O CIRCUITO IMPRESO DEL PROYECTO.

FIGURA 3.3



3.1.6 DESCRIPCIÓN DEL ENSAMBLE.

Teniendo todos los elementos físicos (microcontrolador, teclado, lcd, relé, buzzer, resistencias, RTC, diodos) y lógico (diagrama, algoritmo, firmware), se tiene que proceder a construir el circuito, lo primero que se tiene que hacer es identificar todos y cada uno de los componentes, identificar sus pines, positivos, negativos y guiarse con el diagrama.

En una breadboard, (tablero de ensayo) se coloca el microcontrolador, ATMEGA168 calculando las dimensiones para los demás componentes, una vez colocado el μC , se distribuye los componentes y se comienza a unir según el diagrama esquemático, tratando de hacerlo de la forma más ordenada posible, esto para hacer pruebas previas al montaje de los componentes electrónicos.

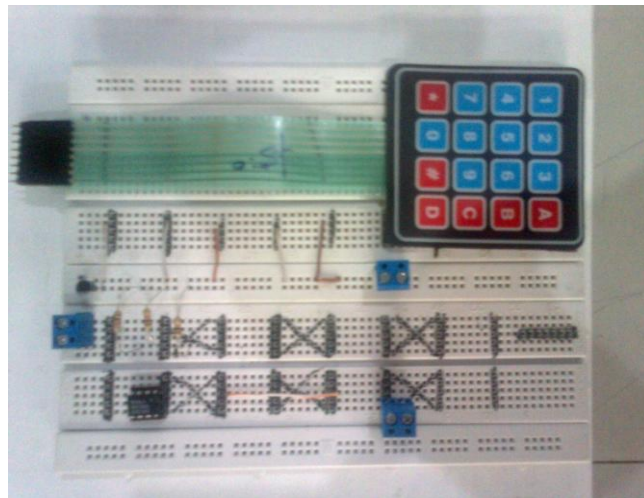


FIGURA 3.4 Montaje en breadboard.

El diagrama esquemático permitirá conocer que componentes se conectarán con el otro, formando así el circuito, no se debe olvidar la parte donde se conectará el programador;

ya teniendo previamente elaborado el FIRMWARE o programa de control del circuito se le descarga al microcontrolador mediante el programa BASCOM-AVR. Se verifica que todo esté conectado según el diagrama, se debe de estar seguro que todo está en orden para posteriormente realizar las pruebas preliminares sobre el funcionamiento de cada elemento y de todos ellos en conjunto. Ya hechas las pruebas y teniendo en cuenta que todo ha funcionado perfectamente se procede a la realización de la placa impresa, la cual se puede hacer mediante diferentes programas, pero en este caso se realizo con el **Circuit Wizard**.

Ya realizado el impreso cada uno de los elementos tendrá que soldarse con el debido cuidado para que no se queden pistas haciendo contacto ya que pueden ocasionar un cortocircuito. Ya soldados los elementos a la placa impresa se debe montar el circuito en el chasis que se va a utilizar.



FIGURA 3.5 Transferencia del circuito impreso a la tableta de cobre.

Al momento de colocar los componentes se debe tener cuidado con la polaridad, las conexiones positiva y negativa u otras conexiones que puedan dañar los componentes o afectar el funcionamiento del timbre automatizado.

Después de simulado, montado y terminado el prototipo se procede a instalarlo en un lugar visible y de fácil acceso. Además se proporciona como anexo un manual de usuario para mejor comprensión y uso del sistema.

3.2 CONCLUSIONES.

En el anterior proyecto se dio a conocer de forma escrita y paso a paso la implementación de un sistema de timbrado automatizado, partiendo de conceptos y teorías acerca de la implementación de dicho proyecto en el Centro Escolar Profesor Jesús Leocadio Palencia, de la ciudad de San Pablo Tacachico”.

En el cual se ha localizado como problema principal y necesidad que exista un control más preciso y exacto de las jornadas educativas. Ante dicha necesidad se crea esta innovación tecnológica con el fin de automatizar el control de cambios de horas clases.

A partir de ello el objetivo principal del sistema de timbre automatizado es ser una herramienta (asistente electrónico automatizado de alto rendimiento y muy bajo costo económico) al servicio del personal docente y alumnos de dicho centro escolar. Por medio de esta herramienta se pretende reducir notoriamente el tiempo invertido en la supervisión del timbre del Centro Escolar.

Con lo anterior se cumplió con el diseño, construcción e implementación que se tenía como promesa en la descripción escrita de dicho proyecto, dando a conocer su principal aplicación y funcionamiento, el cual está destinado para las instalaciones del Centro Escolar Profesor Jesús Leocadio Palencia, de la ciudad de San Pablo Tacachico”.

Logrando a la vez un óptimo desempeño en el control de los periodos de horas clase y para goce de los beneficios que este sistema trae, se da a conocer las diferentes etapas

en las que está constituido (etapa de control y de accionamiento), detallando su funcionamiento y la manera en que trabajaran los dispositivos.

Se cumplió el objetivo de brindarle una buena solución al problema que presentaba el Centro Escolar Profesor Jesús Leocadio Palencia.

Se creó un prototipo funcional que demuestra nuestro trabajo y nuestro profesionalismo cumpliendo con las expectativas de solución.

3.3 RECOMENDACIONES.

- ✓ Se recomienda capacitar previamente al Director y Docentes para que puedan usar y programar adecuadamente el sistema de timbrado automatizado.
- ✓ El dispositivo debe ser colocado en un lugar visible y de fácil acceso para la persona encargada de monitorear y supervisar que se cumplan los horarios de clases.
- ✓ Para mayor facilidad y conveniencia se recomienda comprar los componentes al por mayor.
- ✓ Poseer un sistema de emergencia de alimentación en caso falte la energía eléctrica.
- ✓ Producirlo en masa para ser aplicado en otros centros educativos

3.4 BIBLIOGRAFÍA.

Bergren, (2003) Antony of a robot. New York: McGaw-Hill.

Comité Español de Autonomía. (2006) *Libro blanco de la robótica.*

Madrid: CEA: GTRob.

Salido Tercero, J. (2009) *Cibernética aplicada: robots educativos.*

(1ª Edición). Madrid: Universidad Politécnica.

Uncontrol. (2010) uctrl. *Recuperado de. www.ucontrol.com.ar.*

ANEXOS.



Figura 3.6 Después de imprimir el circuito se pasa al planchado.



Figura 3.7 Quemado de la tableta.

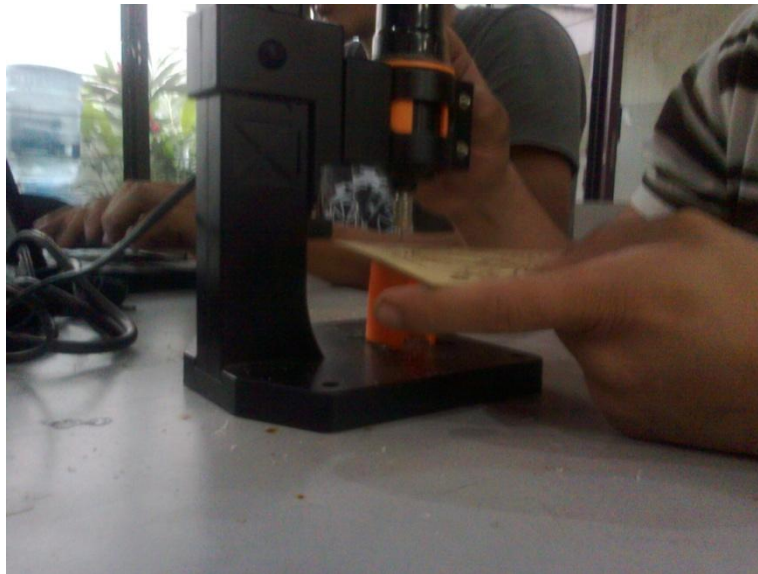


Figura 3.8 Taladrado de la tableta.

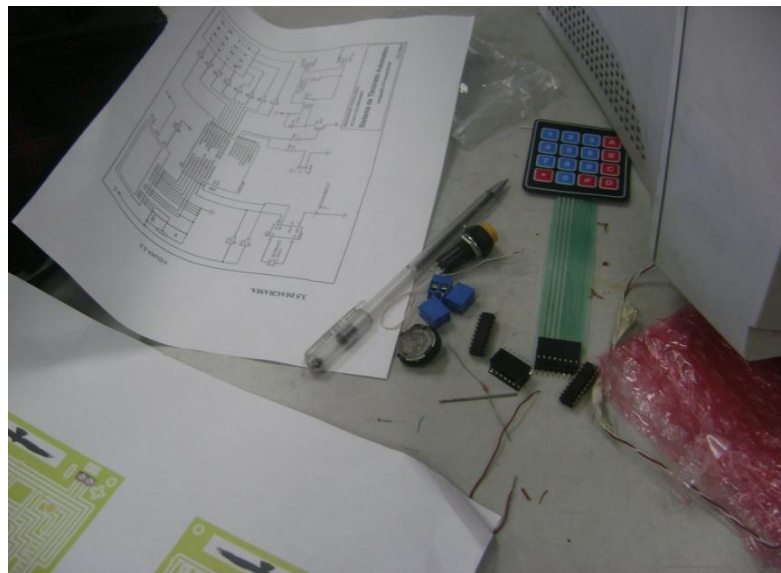


Figura 3.9 Componentes



Figura 4.0 Pruebas de conexión

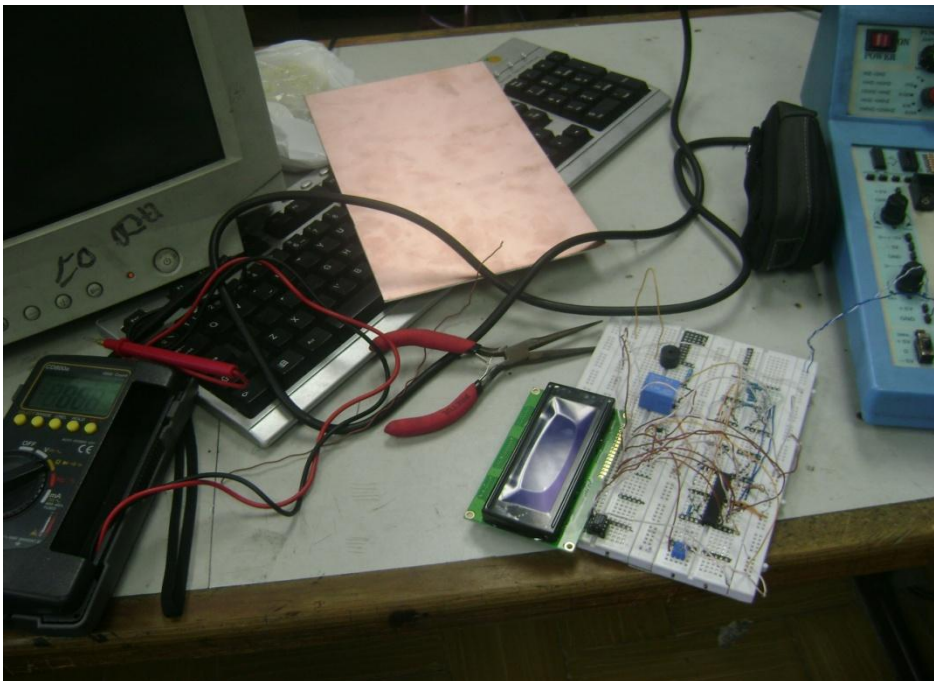


Figura 4.1 Simulación en Breadboard



Figura: A Pruebas finales



Figura: B Prototipo terminado



Figura 4.2 Timbre de la institución



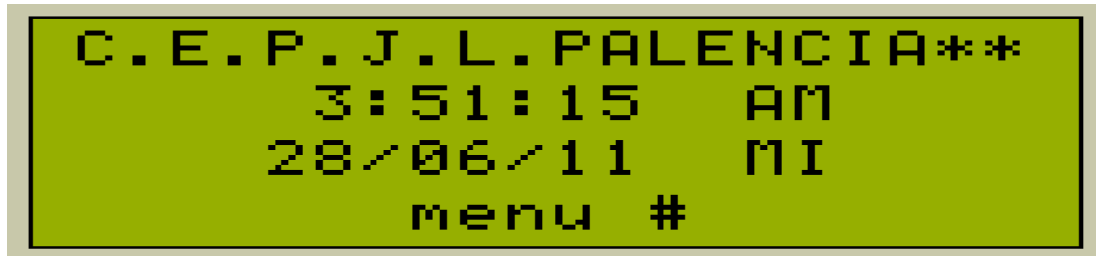
Figura 4.3 Sistema de timbre antiguo



Figura 4.4 Ubicación del sistema de timbre

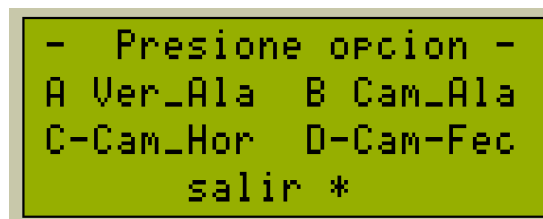
MANUAL DEL TIMBRE AUTOMATIZADO.

Pantalla de inicio del sistema.



1. Ingresar al menú principal.

Debe presionar la tecla numeral (#) aparecerá un menú con las operaciones que se pueden efectuar en este sistema. De la siguiente manera:



Tecla	Función
A	Ver alarmas programadas.
B	Cambiar todas las alarmas.
C	Cambiar la hora.
D	Cambiar la fecha.

Para seleccionar la opción deseada se debe presionar la tecla que aparece al inicio como por ejemplo la letra (A).

A. Ver alarmas.

```
VER alar - 10/turno
A - Matutino
B - Vespertino
salir *
```

Para poder ver las alarmas programadas para el turno matutino debe presionar la tecla “A” de su teclado. Y aparecerá una ventana como la siguiente.

```
Matutino son 8 alar
lunes a viernes
alarma a ver? 0 a 7
salir *
```

Para poder ver la alarma deseada debe presionar el numero de de alarma que desea ver.

Para ello se ha creado una tabla donde se muestran los equivalentes de las alarmas ya que se he utilizado la numeración de 0 a 7 para las 8 alarmas de cada jornada.

Y si no desea efectuar ninguna operación debe presionar * para salir.

B. Cambiar alarmas.

```
- Presione opcion -
A Ver_Ala B Cam_Ala
C-Cam_Hor D-Cam-Fec
salir *
```

Para cambiar las alarmas debe presionar la tecla “B” y se desplegara un menú como el siguiente.

```

Cambiar Alarmas
A - Matutino
B - Vespertino
salir *

```

Después presioné la tecla del turno que desea modificar por ejemplo del turno matutino que es la opción “A” y aparece una menú como el que se presenta a continuación.

```

Matutino son 8 alarm
lunes a viernes
alarma cambiar? 0-7
salir *

```

Y le preguntara que alarma desea modificar debe presionar el numero de la alarma deseada. Por ejemplo “0” que este caso sería la alarma numero uno o la correspondiente a las 7:00am.

```

Alarma 0 L-V matutino
2.HRS 5.MIN 8.MER
7:0 AM
Salir *      OK #

```

Para efectuar la modificación de dicha alarma debe hacer lo siguiente:

Tecla	Función
2	Cambia la hora del sistema.
5	Cambia los minutos.
8	Cambia el meridiano (am o pm).
#	Salir y guardar los cambios.
*	Salir sin guardar cambios.

C. Cambiar hora.

```
- Presione opcion -  
A Ver_Ala  B Cam_Ala  
C-Cam_Hor  D-Cam-Fec  
salir *
```

Para cambiar la hora de su sistema debe presionar la tecla “C” y se desplegara un menú como el siguiente:

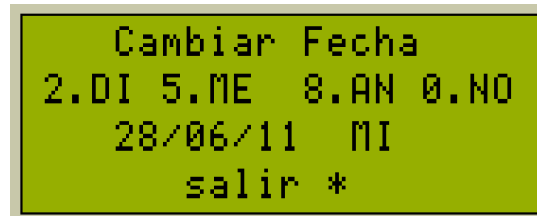
```
Cambiar Hora  
2. +HRS      8. +MIN  
3:56:58 AM  
salir *
```

Tecla	Función
2	Cambia las horas.
8	Cambia los minutos.
*	Salir y guardar cambios.

D. Cambiar fecha.

```
- Presione opcion -  
A Ver_Ala  B Cam_Ala  
C-Cam_Hor  D-Cam-Fec  
salir *
```

Para poder modificar la fecha del sistema presione la tecla “D” y se despliega un menú como el que se muestra a continuación.



2 cambia los días.

5 cambia los meses.

8 cambia los años.

0 este cambia los nombres de los días, ya que nuestro sistema sabe las fechas, pero no que día es.

* Salir y guardar los cambios.

Tecla	Función
2	Cambia los días.
5	Cambia los meses.
8	Cambia los años.
0	Este cambia los nombres de los días, ya que nuestro sistema sabe las fechas, pero no el nombre del día.
*	Salir y guardar los cambios

CENTRO ESCOLAR PROFESOR**JESÚS LEOCADIO PALENCIA****HORARIO DE CLASES**

<i>N</i>	<i>HORA</i>	
1	7:00 - 7:45	PRIMERA CLASE
2	7:45- 8:30	SEGUNDA CLASE
R	8:30 - 8:50	RECESO
3	8:50 - 9:35	TERCERA CLASE
4	9:35 - 10:20	CUARTA CLASE
R	10:20 - 10:30	RECESO
5	10:30 - 11:15	ULTIMA CLASE DE LA MAÑANA
6	11:15 - 12:00	V5
A	12:00 - 1:00	ALMUERZO
1	1:00 - 1:45	PRIMERA CLASE
2	1:45 - 2:30	SEGUNDA CLASE
R	2:30 - 2: 50	RECESO
3	2:50 - 3:35	TERCERA CLASE
4	3:35 - 4:20	CUARTA CLASE
R	4:20 - 4:40	RECESO
5	4:40 - 5:25	ULTIMA CLASE DE LA TARDE

V5= Clase de Informática para Docentes