



RESIDENCIA PROFESIONAL

# Sistema de supervisión y ahorro de energía en aulas



INGENIERÍA EN ELECTRÓNICA

Asesor

M.C. Roberto Ibáñez Córdova

Revisor

Ing. Roger E. Castellanos Galdámez

Presenta

Ricardo Pérez Aguilar

INSTITUTO TECNOLÓGICO DE  
TUXTLA GUTIÉRREZ

31/07/2009

## Índice

Introducción.....	1
Justificación.....	1
Objetivo.....	2
General.....	2
Específicos.....	2
Caracterización del área en que participó.....	2
Problemas a resolver, priorizándolos.....	6
Alcances y limitaciones.....	6
Fundamento teórico.....	7
Procedimiento y descripción de las actividades realizadas.....	23
Resultados, planos, gráficas, prototipos y programas.....	25
Conclusiones y recomendaciones.....	40
Referencias bibliográficas y virtuales.....	40

## INTRODUCCION

Desde la revolución industrial, el hombre ha creado dispositivos electrónicos, que hoy en día es la tecnología más exquisita, ya que por medio de esta el ser humano ha solventado la gran mayoría de sus necesidades, por ejemplo, la automatización de un sinnúmero de procesos.

Cuando automatizamos procesos en una casa, nos referimos a Domótica del latín domus (casa), a su vez del griego doma (cúpula); y robótica, del checo robota (esclavo), sin embargo, la Domótica no se refiere solo a viviendas sino que se amplía para englobar todos los posibles campos de aplicación.

La Domótica, entonces, agrupa todo aquel conjunto de aplicaciones de tecnología en las edificaciones destinadas a dar una mayor calidad de vida, un aumento de la seguridad, el bienestar de los habitantes o usuarios y una racionalización de los consumos energéticos.

## JUSTIFICACION

Las escuelas de diferentes niveles educativos sufren deterioro por parte de personas que operan inadecuadamente las cosas, ocasionando la destrucción del mobiliario, robo de equipos, etc.

Las aulas de clases cuentan con diversos equipos que funcionan con electricidad, la mayor parte de estos se deterioran por el mal uso, debido principalmente a que se mantienen en funcionamiento durante largos periodos de tiempo sin que se aprovechen los beneficios del equipo porque simplemente se les olvido apagarlos, ocasionando con esto desperdicio de energía y de dinero.

El proyecto sistema de seguridad y ahorro de energía es una aplicación de la electrónica orientada a las aulas de clases que trae grandes beneficios:

- Control de acceso.
- Seguridad.
- Ahorro de energía.
- Ahorro de dinero.
- Conservación del mobiliario.

La implementación de la domótica en los salones de clases es viable debido a que contribuye al monitoreo y automatización de equipo eléctrico (aire acondicionado, encendido y apagado de lámparas) apertura y cierre de puertas, mayor vida útil de los equipos eléctricos, seguridad y ahorro de dinero.

El sistema de seguridad y ahorro de energía nos permite tener el control y monitoreo automático de diversas variables en un aula entre ellas: seguridad, control de acceso, control del clima, control de luces, mal uso del salón de clases, desperdicio de energía y la inasistencia por parte del profesor.

## OBJETIVO

### *General*

- Diseñar y crear un sistema de seguridad y ahorro de energía que sea capaz de operar sobre una maqueta o aula modelo.

### *Específicos*

- Controlar el acceso a las diferentes aulas de clase.
- Monitorear remotamente las diferentes aulas de clase. II FASE
- Checar la entrada y salida de los catedráticos a su clase.
- Optimizar lo más que se pueda el uso de la energía eléctrica.
- Restringir el acceso a las aulas exclusivamente a los maestros de acuerdo a su horario.
- Conservación y mayor duración del inmueble, debido a la presencia del docente en el aula.

## CARACTERIZACIÓN DEL ÁREA EN QUE PARTICIPÓ

En la década de los 70's, se incorpora el estado de Chiapas al movimiento educativo nacional extensión educativa, por intervención del Gobierno del Estado de Chiapas ante la federación.

Esta gestión dio origen a la creación del Instituto Tecnológico Regional de Tuxtla Gutiérrez (ITRTG) hoy Instituto Tecnológico de Tuxtla Gutiérrez (ITTG).

El día 23 de agosto de 1971 el Gobernador del Estado, Dr. Manuel Velasco Suárez, colocó la primera piedra de lo que muy pronto sería el Centro Educativo de nivel medio superior más importante de la entidad.

El día 22 de octubre de 1972, con una infraestructura de 2 edificios con 8 aulas, 2 laboratorios y un edificio para talleres abre sus puertas el Instituto Tecnológico de Tuxtla Gutiérrez con las carreras de Técnico en Máquinas de Combustión Interna, Electricidad, Laboratorista Químico y Máquinas y Herramientas.

En el año 1974 dio inicio la modalidad en el nivel superior, ofreciendo las carrera de Ingeniería Industrial en Producción y Bioquímica en Productos Naturales. En 1980 se amplió la oferta educativa al incorporarse las carreras de Ingeniería Industrial Eléctrica e Ingeniería Industrial Química.

En 1987 se abre la carrera de Ingeniería en Electrónica y se liquidan en 1989 las carreras del sistema abierto del nivel medio superior y en el nivel superior se reorientó la oferta en la carrera de Ingeniería Industrial Eléctrica y se inicia también Ingeniería Mecánica.

En 1991 surge la licenciatura en Ingeniería en Sistemas Computacionales.

Desde 1997 el Instituto Tecnológico de Tuxtla Gutiérrez ofrece la Especialización en Ingeniería Ambiental como primer programa de postgrado.

En 1998 se estableció el programa interinstitucional de postgrado con la Universidad Autónoma de Chiapas para impartir en el Instituto Tecnológico la Maestría en Biotecnología.

En el año 1999 se inició el programa de Maestría en Administración como respuesta a la demanda del sector industrial y de servicios de la región.

A partir de 2000 se abrió también la Especialización en Biotecnología Vegetal y un año después dio inicio el programa de Maestría en Ciencias en Ingeniería Bioquímica y la Licenciatura en Informática.

La Secretaría de Educación Pública, en el afán de garantizar la calidad de la educación superior pública, impulsa programas extraordinarios a través de los cuales se pretende atender la demanda de educación superior, ofreciendo servicio educativo con Calidad. Entre los objetivos estratégicos que se alcanzarán hacia el 2006 a través de estos programas especiales, se encuentran la ampliación de cobertura con equidad, calidad de planes y programas de estudio, así como los servicios que ofrecen las IES públicas y la integración, coordinación y gestión del sistema de educación superior.

El Instituto Tecnológico de Tuxtla Gutiérrez ha hecho patente siempre su compromiso con la sociedad y con los jóvenes de Chiapas y México, en respuesta

a estos retos, a principios del 2004 implementó el Programa de Innovación y Calidad y en Diciembre de ese mismo año, inició la implementación de su Sistema de Gestión de la Calidad basado en la norma ISO 9001:2000, incorporando así las herramientas necesarias para responder con éxito las exigencias del mundo actual.

El personal directivo, docentes, de apoyo administrativo y alumnos, están unidos en este esfuerzo por consolidar al Instituto Tecnológico de Tuxtla Gutiérrez como una Institución reconocida por la calidad de sus servicios educativos a través de la Certificación de su Proceso Educativo, conforme a la Norma ISO 9001:2000.

### **Política de Calidad**

El SNEST establece el compromiso de implementar todos sus procesos, orientándolos hacia la satisfacción de sus clientes sustentada en la Calidad del Proceso Educativo, para cumplir con sus requerimientos mediante la eficacia de un Sistema de Gestión de la Calidad y de mejora continua, conforme a la norma ISO 9001:2000/NMX-CC-9001-IMNC-2000.

### **Los valores de la Organización son:**

- El ser humano
- El espíritu de servicio
- El liderazgo
- El Trabajo en equipo
- La calidad
- Alto desempeño

### ***Objetivos de la Calidad***

### **Objetivo General:**

“Proporcionar el Servicio Educativo de Calidad, orientado hacia el Aprendizaje significativo en el alumno”.

## ***Objetivos de Procesos Estratégicos***

### **Académico:**

Gestionar los Planes y Programas de Estudio, así como los programas de formación y actualización docente y profesional en el servicio educativo.

### **Planeación:**

Definir el rumbo estratégico mediante la planeación y realizar la programación, presupuestación, seguimiento y evaluación de las acciones para cumplir con los requisitos del servicio.

### **Vinculación:**

Contribuir a la formación integral del Alumno, a través de su vinculación con el sector productivo y la sociedad, así como del deporte y la cultura.

### **Administración de los recursos:**

Determinar y proporcionar los recursos necesarios para implementar, mantener y mejorar el SGC y lograr la conformidad con los requisitos del Servicio Educativo.

### **Calidad:**

Promover una cultura de calidad al interior de la ORGANIZACIÓN y asegurar la satisfacción del alumno.

## **Misión**

Formar de manera integral profesionistas de excelencia en el campo de la ciencia y la tecnología con actitud emprendedora, respeto al medio ambiente y apego a los valores éticos.

## **Visión**

Ser una Institución de excelencia en la educación superior tecnológica del Sureste, comprometida con el desarrollo socioeconómico sustentable de la región.

## **Valores**

- El ser humano
- El espíritu de servicio
- El liderazgo
- El trabajo en equipo
- La calidad
- El alto desempeño

Las actividades se realizaron en los laboratorios de ingeniería electrónica, ubicados en el edificio I del Instituto Tecnológico de Tuxtla Gutiérrez.

## **PROBLEMAS A RESOLVER, PRIORIZÁNDOLOS**

- 1.- Automatizar el acceso a las diferentes aulas de clase.
- 2.- reducir el gasto excesivo de la energía eléctrica en las aulas de clases.
- 3.- Restringir el acceso a las aulas exclusivamente a los maestros de acuerdo a su horario.

## **ALCANCES Y LIMITACIONES**

El sistema de seguridad y ahorro de energía, en su primera fase, será capaz de controlar automáticamente el acceso a los salones, encendido y apagado de luces, climas, registrando principalmente el acceso al aula en una base de datos que se maneja desde una PC. El acceso se dará de la siguiente manera; el maestro ingresa su tarjeta, el sistema evalúa si es el salón correcto, la hora correcta, y evidentemente que sea el maestro correcto, si estas tres condiciones



se cumplen el catedrático podrá acceder al salón, y al momento deberán encenderse las luces y el clima si es necesario.

Cuando la clase del maestro haya terminado, habrá también un buzzer o una campanilla que indique el término de esta, el maestro saldrá del aula y todo el salón quedara sin energía automáticamente, a excepción de la placa de control, hasta que un nuevo catedrático ingrese a dicha aula.

Un edificio modelo que podrá manejar hasta 7 aulas, las cuales pueden contener para su control lo siguiente:

- a) 2 climas por aula.
- b) 3 bloques de cuatro lámparas cada una.
- c) Contactos eléctricos.
- d) 1 sensor de movimiento.
- e) 1 contrachapa eléctrica.
- f) 1 cámara de supervisión remota.

## FUNDAMENTO TEÓRICO

### Visual FoxPro 9.0

Visual FoxPro es uno de los gestores de base de datos relacionales mas rápidos y flexibles del mercado disponiendo de un complejo entorno de desarrollo totalmente orientado al objeto y altamente integrado en el sistema operativo Windows 95. Como cualquier otra aplicación Windows estándar, visual FoxPro soporta interfaces MDI (interfaz de documento múltiple), como la de Word, barras de herramientas, ayuda sensible al contexto, múltiples fuentes, acceso a bibliotecas de enlace dinámico (DLL), etc.

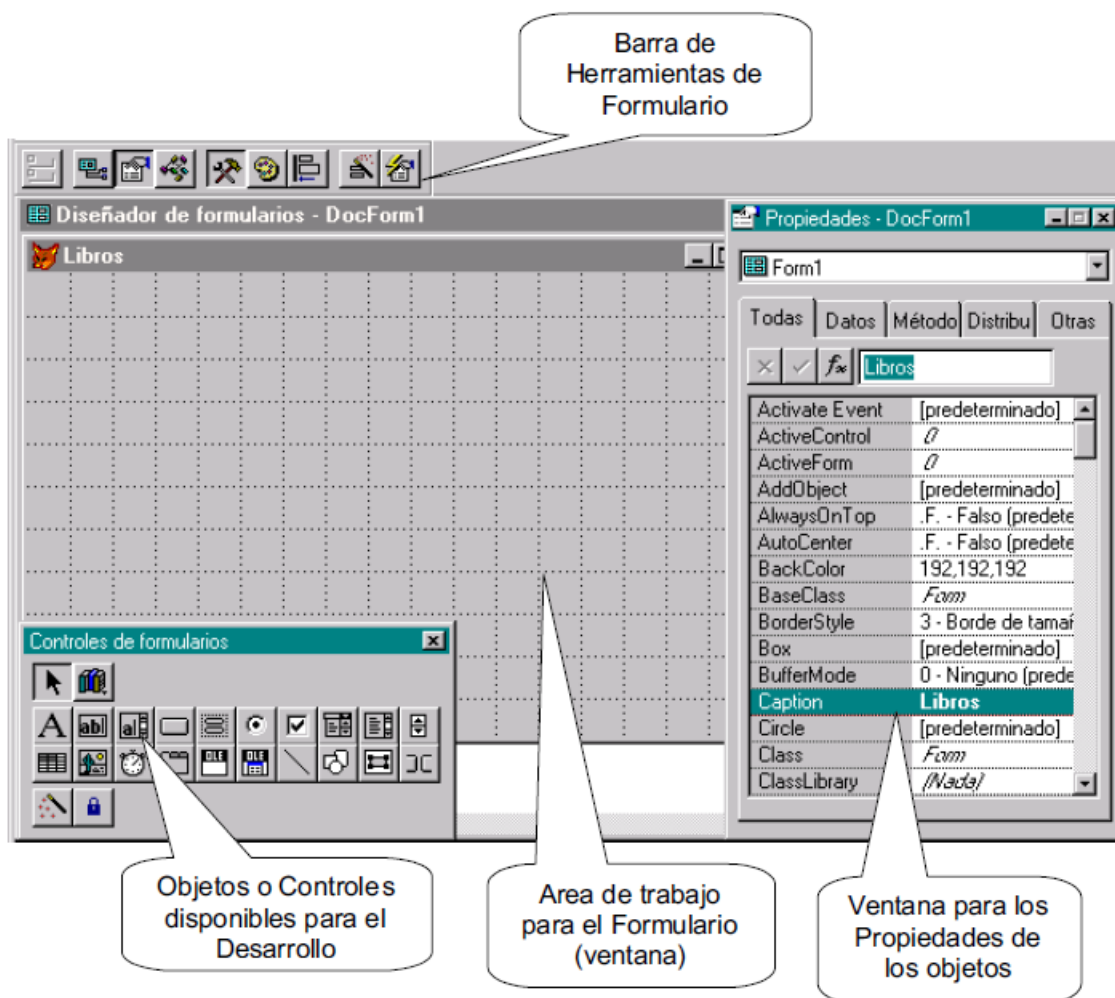
Aparte del acceso a tablas en formato nativo de visual FoxPro, se puede acceder virtualmente a cualquier fuente de datos cuyo acceso se pueda hacer a través de ODBC (open database connectivity). Este estándar, definido como conectividad abierta de bases de datos, permite abrir y consultar diversas bases de datos a través de un conjunto de controladores, utilizando SQL como lenguaje de consulta.

Con el advenimiento de Visual Fox, los desarrolladores de aplicaciones tienen acceso a multitud de eventos y opciones de los programadores de Visual Basic y visual c tuvieron durante años. Se elimina la necesidad del uso del read y su confusa nomenclatura (fundamental read).

## El diseño de formularios

El diseñador de formularios es el lugar donde realmente se lleva a cabo la programación, como ya se ha dicho aquí es donde se utilizan los objetos visuales (controles), a los cuales cada uno tiene sus propiedades (características) y eventos. Hay eventos que son comunes para la mayoría de los objetos (controles), como por ejemplo, el CLICK o el MOUSE MOVE, por mencionar algunos.

Para crear un formulario sin los asistentes, primero se debe ir a la página **Documentos**, elegir la opción formularios y hacer click sobre el botón **nuevo**, posteriormente a esto Visual FoxPro preguntará si desea utilizar el asistente o un formulario en blanco. En este caso será un formulario en blanco; al hacer esto debe aparecer el Diseñador de Formularios como se muestra en la siguiente figura.



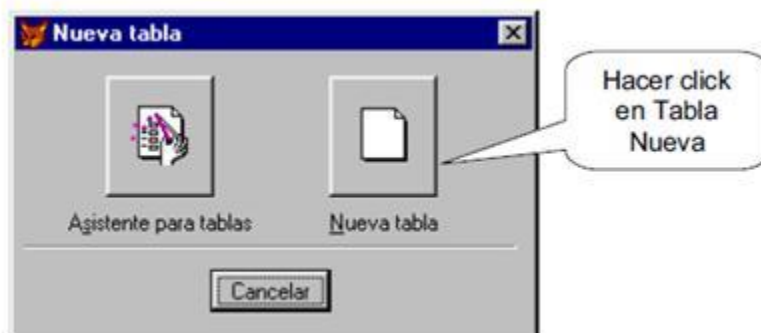
## Crear una tabla

Para crear una tabla, debes ir primero a la pagina Datos y seleccionar la base de datos a la cual le quieres agregar la tabla, dándole doble click sobre ella, y posteriormente seleccionar la opción Tablas, para después hacer click sobre el botón Nuevo, como se muestra en la siguiente figura.



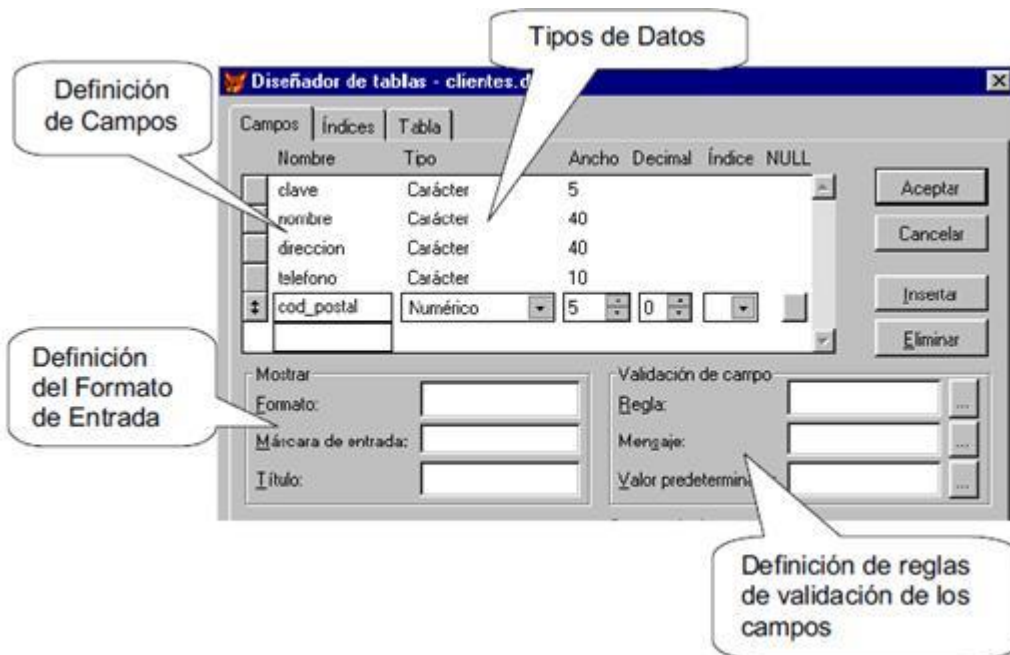
Al hacer click sobre el botón Nuevo aparecerá la ventana en la cual nos pregunta si deseamos una tabla en blanco, o queremos usar el asistente para tablas.

Seleccionaremos tabla en blanco, como se muestra en la siguiente figura.



Al hacer click sobre el botón tabla nueva entonces deberá aparecer de nuevo la ventana donde nos pide un nombre y una ubicación para la tabla (la tabla se guarda con extensión DBF), como ya se ha visto este procedimiento ya no se mostrara esta pantalla.

Una vez habiendo elegido un nombre y una ubicación donde quedara almacenada la tabla que vamos a crear, nos aparecerá el diseño de tablas, como se muestra en la siguiente figura.



#### Definición de campos:

Igual que en las versiones anteriores, la definición de campos es igual. Se introduce el nombre del campo (nombre) el tipo de datos que soportara (tipo) y el tamaño.

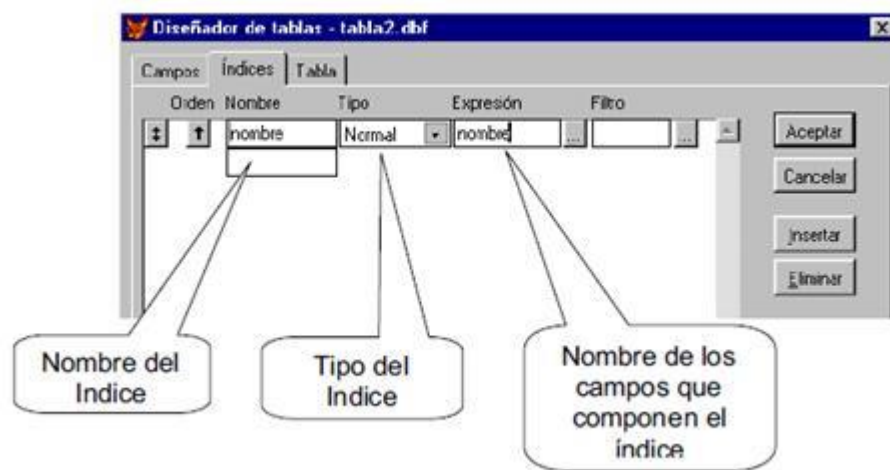
#### Definición de mostrar

Formato: es el formato de salida del campo, ejemplo si fuera numérico 9,999.99 o si fuera carácter !!!!!!!!!!!!!, para que solo acepte mayúsculas.

Máscara de entrada: igual que el formato, lo único que es para la entrada de datos.

Título: descripción más precisa del campo, es de utilidad al diseñar los formularios.

Luego de hacer la definición de los campos, pasamos a la definición de índices de la tabla, que será muy importante a la hora de crear las relaciones entre las tablas.



Luego de hacer la definición de los campos, pasamos a la definición de índices de la tabla, que será muy importante a la hora de crear las relaciones entre las tablas.

Nombre del índice (nombre)

Introducir el nombre del archivo índice. Es conocido también como TAG.

Tipo (tipo del índice)

Primary : Llave primaria.

Candidate : Llave candidata.

Unique : Llave única.

Regular : Llave foránea.

Interface de comunicación



El **Universal Serial Bus** (bus universal en serie) o **Conductor Universal en Serie (CUS)**, abreviado comúnmente **USB**, es un puerto que sirve para conectar periféricos a una computadora. Fue creado en 1996 por siete empresas: IBM, Intel, Northern Telecom, Compaq, Microsoft, Digital Equipment Corporation y NEC.

El estándar incluye la transmisión de energía eléctrica al dispositivo conectado. Algunos dispositivos requieren una potencia mínima, así que se pueden conectar

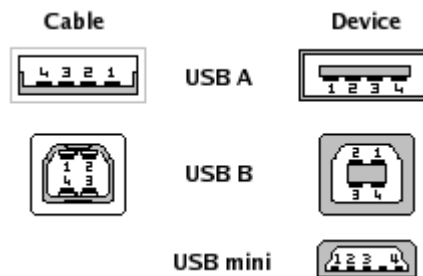
varios sin necesitar fuentes de alimentación extra. La gran mayoría de los concentradores incluyen fuentes de alimentación que brindan energía a los dispositivos conectados a ellos, pero algunos dispositivos consumen tanta energía que necesitan su propia fuente de alimentación. Los concentradores con fuente de alimentación pueden proporcionarle corriente eléctrica a otros dispositivos sin quitarle corriente al resto de la conexión (dentro de ciertos límites).

El diseño del USB tenía en mente eliminar la necesidad de adquirir tarjetas separadas para poner en los puertos bus ISA o PCI, y mejorar las capacidades plug-and-play permitiendo a esos dispositivos ser conectados o desconectados al sistema sin necesidad de reiniciar. Cuando se conecta un nuevo dispositivo, el servidor lo enumera y agrega el software necesario para que pueda funcionar.

El USB puede conectar los periféricos como ratones, teclados, escáneres, cámaras digitales, teléfonos móviles, reproductores multimedia, impresoras, discos duros externos, tarjetas de sonido, sistemas de adquisición de datos y componentes de red. Para dispositivos multimedia como escáneres y cámaras digitales, el USB se ha convertido en el método estándar de conexión. Para impresoras, el USB ha crecido tanto en popularidad que ha desplazado a un segundo plano a los puertos paralelos porque el USB hace mucho más sencillo el poder agregar más de una impresora a una computadora personal.

En el caso de los discos duros, es poco probable que el USB reemplace completamente a los buses (el ATA (IDE) y el SCSI), pues el USB tiene un rendimiento más lento que esos otros estándares. Sin embargo, el USB tiene una importante ventaja en su habilidad de poder instalar y desinstalar dispositivos sin tener que abrir el sistema, lo cual es útil para dispositivos de almacenamiento externo. Hoy en día, una gran parte de los fabricantes ofrece dispositivos USB portátiles que ofrecen un rendimiento casi indistinguible en comparación con los ATA (IDE). Por el contrario, el nuevo estándar Serial ATA permite tasas de transferencia de hasta aproximadamente 150/300 MB por segundo, y existe también la posibilidad de extracción en caliente e incluso una especificación para discos externos llamada eSATA.

#### Características de transmisión



Pin	Signal	Color	Description
1	VCC	■	+5V
2	D-	□	Data -
3	D+	■	Data +
4	GND	■	Ground

Los dispositivos USB se clasifican en cuatro tipos según su velocidad de transferencia de datos:

- Baja velocidad (1.0): Tasa de transferencia de hasta 1'5 Mbps (192 KB/s). Utilizado en su mayor parte por dispositivos de interfaz humana (*Human interface device*, en inglés) como los teclados, los ratones y los joysticks.
- Velocidad completa (1.1): Tasa de transferencia de hasta 12 Mbps (1'5 MB/s). Ésta fue la más rápida antes de la especificación USB 2.0, y muchos dispositivos fabricados en la actualidad trabajan a esta velocidad. Estos dispositivos dividen el ancho de banda de la conexión USB entre ellos, basados en un algoritmo de búferes FIFO.
- Alta velocidad (2.0): Tasa de transferencia de hasta 480 Mbps (60 MB/s). Lo usa por ejemplo la consola portátil de Sony PSP
- Super alta velocidad (3.0): Actualmente en fase experimental y con tasa de transferencia de hasta 4.8 Gbps (600 MB/s). Esta especificación será lanzada a mediados de 2009 por Intel, de acuerdo con información recabada de Internet; Aunque actualmente cualquier distribución GNU/Linux es capaz en teoría de soportar el nuevo estandar ya que aun no hay hardware disponible <sup>1</sup>. La velocidad del bus será diez veces más rápida que la del USB 2.0, debido a la que han incluido 5 conectores extra, desechando el conector de fibra óptica propuesto inicialmente, y será compatible con los estándares anteriores. Se espera que los productos fabricados con esta tecnología lleguen al consumidor en 2009 o 2010.<sup>2 3</sup>

Las señales del USB se transmiten en un cable de par trenzado con impedancia de  $90 \Omega \pm 15\%$ , cuyos hilos se denominan D+ y D-.<sup>4</sup> Estos, colectivamente, utilizan señalización diferencial en half dúplex para combatir los efectos del ruido electromagnético en enlaces largos. D+ y D- suelen operar en conjunto y no son conexiones simples. Los niveles de transmisión de la señal varían de 0 a 0'3 V para bajos (ceros) y de 2'8 a 3'6 V para altos (unos) en las versiones 1.0 y 1.1, y en  $\pm 400$  mV en alta velocidad (2.0). En las primeras versiones, los alambres de los cables no están conectados a masa, pero en el modo de alta velocidad se tiene una terminación de  $45 \Omega$  a tierra o un diferencial de  $90 \Omega$  para acoplar la impedancia del cable. Este puerto sólo admite la conexión de dispositivos de bajo consumo, es decir, que tengan un consumo máximo de 100 mA por cada puerto;

sin embargo, en caso de que estuviese conectado un dispositivo que permite 4 puertos por cada salida USB (extensiones de máximo 4 puertos), entonces la energía del USB se asignará en unidades de 100 mA hasta un máximo de 500 mA por puerto.

Tipos diferentes de conectores USB (de izquierda a derecha): micro USB macho, mini USB tipo B macho, Tipo B macho, Tipo A hembra, Tipo A macho



El estándar USB especifica tolerancias para impedancia y de especificaciones mecánicas relativamente bajas para sus conectores, intentando minimizar la incompatibilidad entre los conectores fabricados por distintas compañías —una meta a la que se ha logrado llegar. El estándar USB, a diferencia de otros estándares también define tamaños para el área alrededor del conector de un dispositivo, para evitar el bloqueo de un puerto adyacente por el dispositivo en cuestión.

Las especificaciones USB 1.0, 1.1 y 2.0 definen dos tipos de conectores para conectar dispositivos al servidor: A y B. Sin embargo, la capa mecánica ha cambiado en algunos conectores. Por ejemplo, el IBM UltraPort es un conector USB privado localizado en la parte superior del LCD de los computadoras portátiles de IBM. Utiliza un conector mecánico diferente mientras mantiene las señales y protocolos característicos del USB. Otros fabricantes de artículos pequeños han desarrollado también sus medios de conexión pequeños, y ha aparecido una gran variedad de ellos, algunos de baja calidad.

Una extensión del USB llamada "USB-On-The-Go" (*sobre la marcha*) permite a un puerto actuar como servidor o como dispositivo - esto se determina por qué lado del cable está conectado al aparato. Incluso después de que el cable está conectado y las unidades se están comunicando, las 2 unidades pueden "cambiar de papel" bajo el control de un programa. Esta facilidad está específicamente diseñada para dispositivos como PDA, donde el enlace USB podría conectarse a un PC como un dispositivo, y conectarse como servidor a un teclado o ratón. El "USB-On-The-Go" también ha diseñado 2 conectores pequeños, el mini-A y el mini-B, así que esto debería detener la proliferación de conectores miniaturizados de entrada.



## Como Funciona

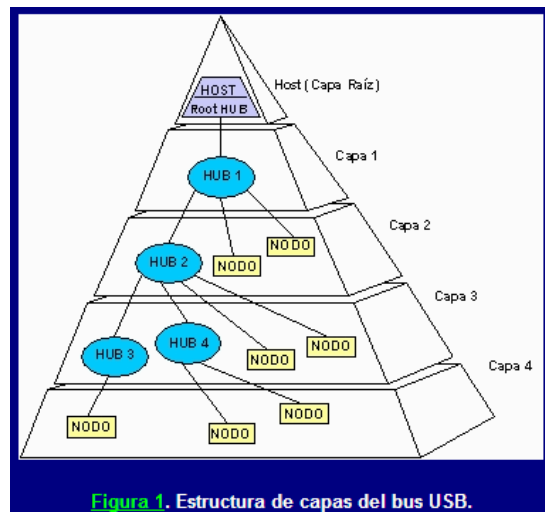
Trabaja como interfaz para transmisión de datos y distribución de energía, que ha sido introducida en el mercado de PC's y periféricos para mejorar las lentas interfaces serie (RS-232) y paralelo. Esta interfaz de 4 hilos, 12 Mbps y "plug and play", distribuye 5V para alimentación, transmite datos y está siendo adoptada rápidamente por la industria informática.

Es un bus basado en el paso de un testigo, semejante a otros buses como los de las redes locales en anillo con paso de testigo y las redes FDDI. El controlador USB distribuye testigos por el bus. El dispositivo cuya dirección coincide con la que porta el testigo responde aceptando o enviando datos al controlador. Este también gestiona la distribución de energía a los periféricos que lo requieran.

Emplea una topología de estrellas apiladas que permite el funcionamiento simultáneo de 127 dispositivos a la vez. En la raíz o vértice de las capas, está el controlador anfitrión o host que controla todo el tráfico que circula por el bus. Esta topología permite a muchos dispositivos conectarse a un único bus lógico sin que los dispositivos que se encuentran más abajo en la pirámide sufran retardo. A diferencia de otras arquitecturas, USB no es un bus de almacenamiento y envío, de forma que no se produce retardo en el envío de un paquete de datos hacia capas inferiores.

El sistema de bus serie universal USB consta de tres componentes:

- Controlador
- Hubs o Concentradores
- Periféricos



## Controlador

Reside dentro del PC y es responsable de las comunicaciones entre los periféricos USB y la CPU del PC. Es también responsable de la admisión de los periféricos dentro del bus, tanto si se detecta una conexión como una desconexión. Para

cada periférico añadido, el controlador determina su tipo y le asigna una dirección lógica para utilizarla siempre en las comunicaciones con el mismo. Si se producen errores durante la conexión, el controlador lo comunica a la CPU, que, a su vez, lo transmite al usuario. Una vez se ha producido la conexión correctamente, el controlador asigna al periférico los recursos del sistema que éste precise para su funcionamiento.

El controlador también es responsable del control de flujo de datos entre el periférico y la CPU.

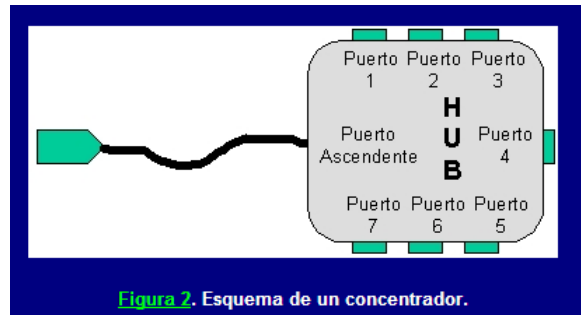
### **Concentradores o hubs**

Son distribuidores inteligentes de datos y alimentación, y hacen posible la conexión a un único puerto USB de 127 dispositivos. De una forma selectiva reparten datos y alimentación hacia sus puertas descendentes y permiten la comunicación hacia su puerta de retorno o ascendente. Un hub de 4 puertos, por ejemplo, acepta datos del PC para un periférico por su puerta de retorno o ascendente y los distribuye a las 4 puertas descendentes si fuera necesario.

Los concentradores también permiten las comunicaciones desde el periférico hacia el PC, aceptando datos en las 4 puertas descendentes y enviándolos hacia el PC por la puerta de retorno.

Además del controlador, el PC también contiene el concentrador raíz. Este es el primer concentrador de toda la cadena que permite a los datos y a la energía pasar a uno o dos conectores USB del PC, y de allí a los 127 periféricos que, como máximo, puede soportar el sistema. Esto es posible añadiendo concentradores adicionales. Por ejemplo, si el PC tiene una única puerta USB y a ella le conectamos un hub o concentrador de 4 puertas, el PC se queda sin más puertas disponibles. Sin embargo, el hub de 4 puertas permite realizar 4 conexiones descendentes. Conectando otro hub de 4 puertas a una de las 4 puertas del primero, habremos creado un total de 7 puertas a partir de una puerta del PC. De esta forma, es decir, añadiendo concentradores, el PC puede soportar hasta 127 periféricos USB.

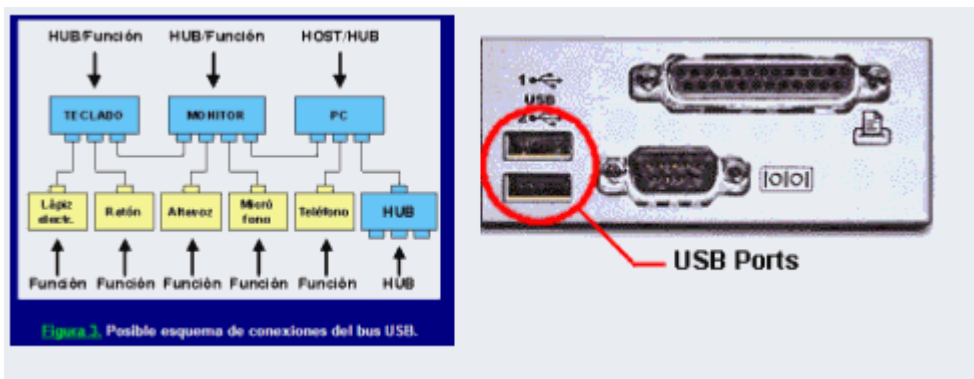
La mayoría de los concentradores se encontrarán incorporados en los periféricos. Por ejemplo, un monitor USB puede contener un concentrador de 7 puertas incluido dentro de su chasis. El monitor utilizará una de ellas para sus datos y control y le quedarán 6 para conectar allí otros periféricos.



## Periféricos

USB soporta periféricos de baja y media velocidad. Empleando dos velocidades para la transmisión de datos de 1.5 y 12 Mbps se consigue una utilización más eficiente de sus recursos. Los periféricos de baja velocidad tales como teclados, ratones, joysticks, y otros periféricos para juegos, no requieren 12 Mbps. Empleando para ellos 1,5 Mbps, se puede dedicar más recursos del sistema a periféricos tales como monitores, impresoras, módems, scanner, equipos de audio . . . , que precisan de velocidades más altas para transmitir mayor volumen de datos o datos cuya dependencia temporal es más estricta.

En las figuras 3 y 4 se puede ver cómo los hubs proporcionan conectividad a toda una serie de dispositivos periféricos



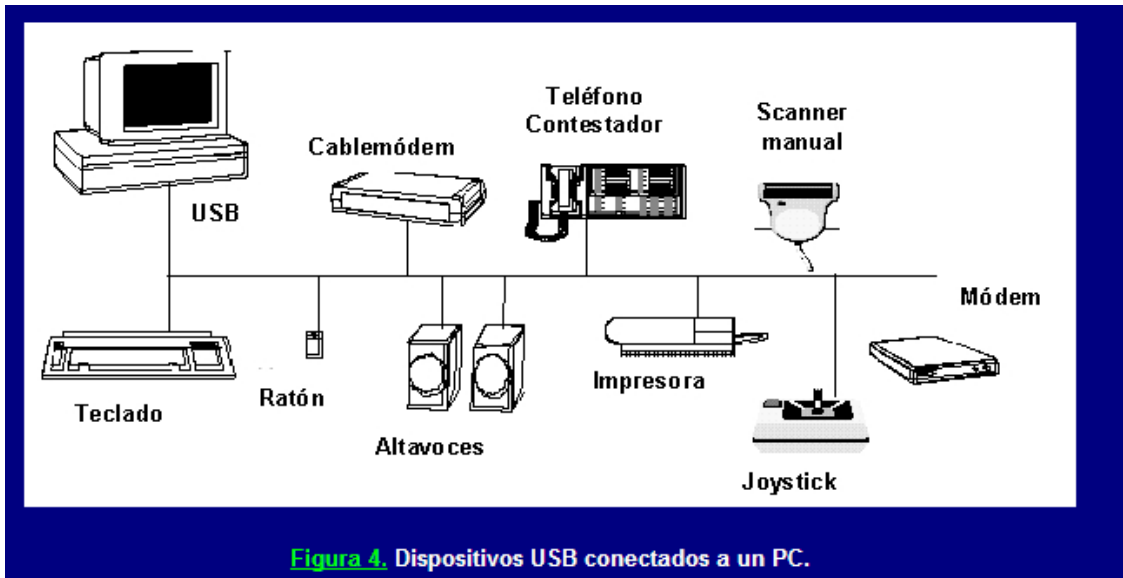


Figura 4. Dispositivos USB conectados a un PC.

#### 40-Pin PDIP

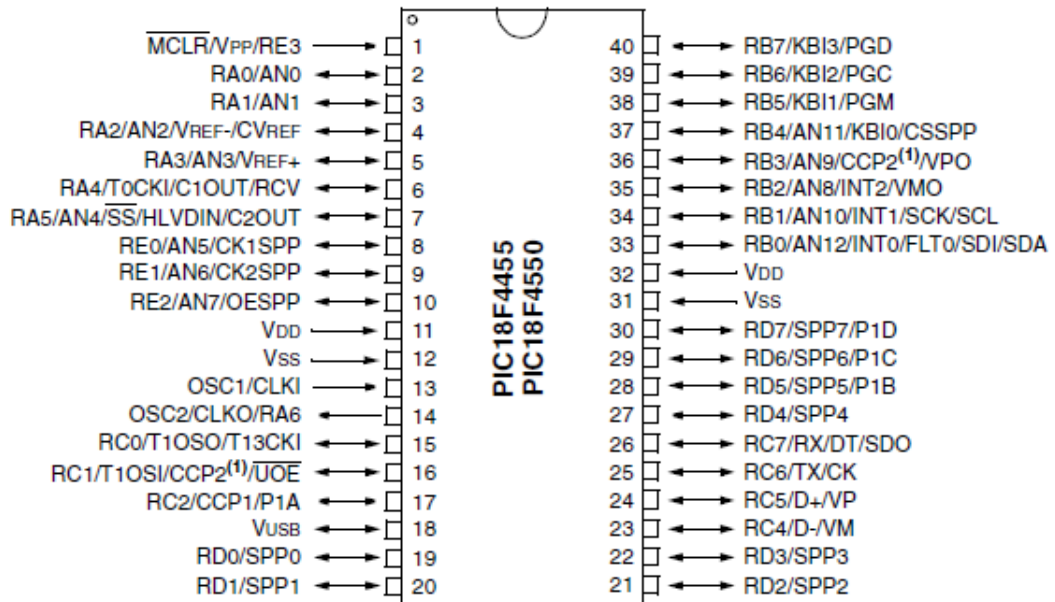


Figura 1. Imagen física del microcontrolador

Device	Program Memory		Data Memory		I/O	10-Bit A/D (ch)	CCP/ECCP (PWM)	SPP	MSSP		EAUSART	Comparators	Timers 8/16-Bit
	Flash (bytes)	# Single-Word Instructions	SRAM (bytes)	EEPROM (bytes)					SPI	Master I <sup>2</sup> C™			
PIC18F2455	24K	12288	2048	256	24	10	2/0	No	Y	Y	1	2	1/3
PIC18F2550	32K	16384	2048	256	24	10	2/0	No	Y	Y	1	2	1/3
PIC18F4455	24K	12288	2048	256	35	13	1/1	Yes	Y	Y	1	2	1/3
PIC18F4550	32K	16384	2048	256	35	13	1/1	Yes	Y	Y	1	2	1/3

Features	PIC18F2455	PIC18F2550	PIC18F4455	PIC18F4550
Operating Frequency	DC – 48 MHz	DC – 48 MHz	DC – 48 MHz	DC – 48 MHz
Program Memory (Bytes)	24576	32768	24576	32768
Program Memory (Instructions)	12288	16384	12288	16384
Data Memory (Bytes)	2048	2048	2048	2048
Data EEPROM Memory (Bytes)	256	256	256	256
Interrupt Sources	19	19	20	20
I/O Ports	Ports A, B, C, (E)	Ports A, B, C, (E)	Ports A, B, C, D, E	Ports A, B, C, D, E
Timers	4	4	4	4
Capture/Compare/PWM Modules	2	2	1	1
Enhanced Capture/Compare/PWM Modules	0	0	1	1
Serial Communications	MSSP, Enhanced USART	MSSP, Enhanced USART	MSSP, Enhanced USART	MSSP, Enhanced USART
Universal Serial Bus (USB) Module	1	1	1	1
Streaming Parallel Port (SPP)	No	No	Yes	Yes
10-Bit Analog-to-Digital Module	10 Input Channels	10 Input Channels	13 Input Channels	13 Input Channels
Comparators	2	2	2	2
Resets (and Delays)	POR, BOR, RESET Instruction, Stack Full, Stack Underflow (PWRT, OST), MCLR (optional), WDT	POR, BOR, RESET Instruction, Stack Full, Stack Underflow (PWRT, OST), MCLR (optional), WDT	POR, BOR, RESET Instruction, Stack Full, Stack Underflow (PWRT, OST), MCLR (optional), WDT	POR, BOR, RESET Instruction, Stack Full, Stack Underflow (PWRT, OST), MCLR (optional), WDT
Programmable Low-Voltage Detect	Yes	Yes	Yes	Yes
Programmable Brown-out Reset	Yes	Yes	Yes	Yes
Instruction Set	75 Instructions; 83 with Extended Instruction Set enabled	75 Instructions; 83 with Extended Instruction Set enabled	75 Instructions; 83 with Extended Instruction Set enabled	75 Instructions; 83 with Extended Instruction Set enabled
Packages	28-pin PDIP 28-pin SOIC	28-pin PDIP 28-pin SOIC	40-pin PDIP 44-pin QFN 44-pin TQFP	40-pin PDIP 44-pin QFN 44-pin TQFP

Figura 2. Tabla de datos del microcontrolador PIC 18F4550

Etapa de entrada de sensores

**LOGIC AND CONNECTION DIAGRAMS DIP (TOP VIEW)**

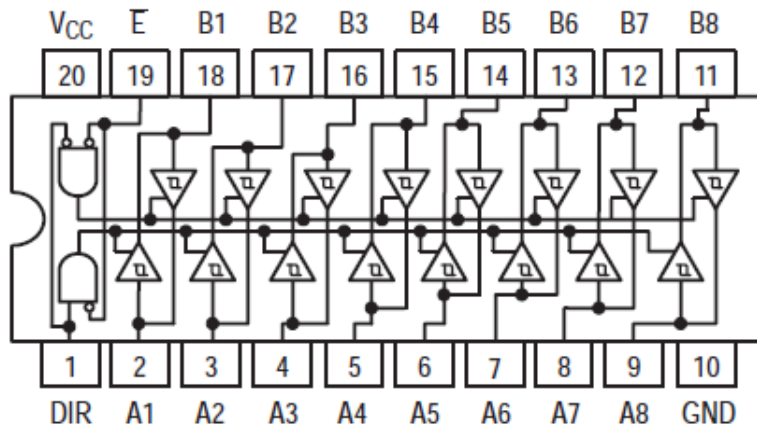


Figura 3. Imagen física del 74LS245

INPUTS		OUTPUT
E	DIR	
L	L	Bus B Data to Bus A
L	H	Bus A Data to Bus B
H	X	Isolation

H = HIGH Voltage Level  
 L = LOW Voltage Level  
 X = Immaterial

Figura 4. Tabla de verdad del 74LS245

Etapa de salida

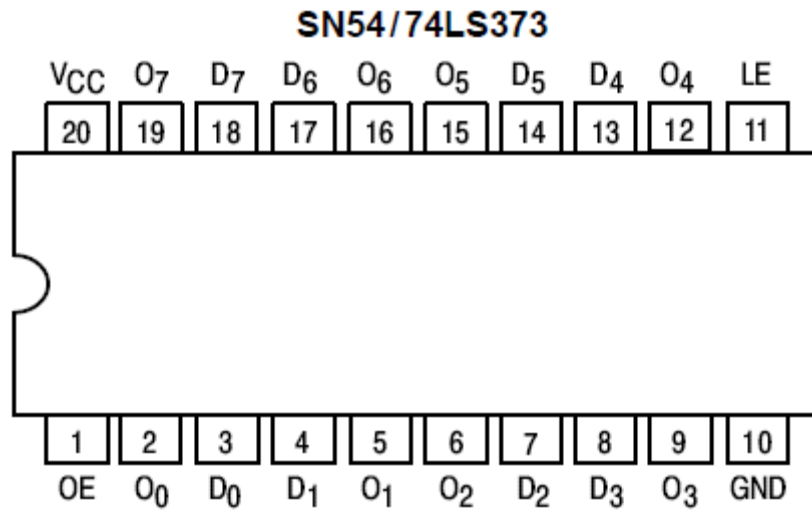


Figura 5. Imagen física del 74LS373

**LS373**

$D_n$	LE	OE	$O_n$
H	H	L	H
L	H	L	L
X	L	L	$Q_0$
X	X	H	$Z^*$

H = HIGH Voltage Level

L = LOW Voltage Level

X = Immaterial

Z = High Impedance

Figura 6. Tabla de verdad del 74LS373

Etapa de salida (potencia)

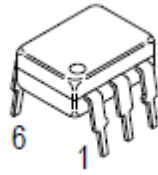


Figura 7. Imagen física del MOC3011

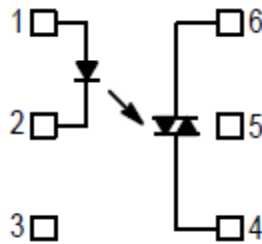


Figura 8. Esquema del MOC3011

Características del MAC8D:

Bloqueo de tensión a 800 voltios

El Estado actual de Calificación de 8,0 Amperios RMS a 100 ° C

Uniforme de la Puerta de activación Corrientes en tres modos

Alta inmunidad a  $dv / dt$  - 250 V / ms como mínimo a 125 ° C

Minimiza Snubber Redes para Protección

normas de la industria del paquete A-220AB

Alta Commutating  $di / dt$  - 6.5 A / ms como mínimo a 125 ° C

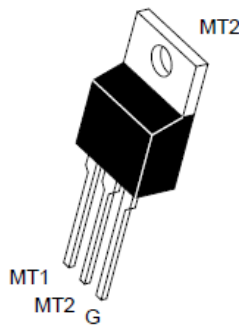


Figura 9. Imagen física del MAC8D



## PROCEDIMIENTO Y DESCRIPCIÓN DE LAS ACTIVIDADES REALIZADAS

Diseño del plano del aula a automatizar.

Se considero todos los posibles elementos a controlar, por lo que se construyó una maqueta o aula modelo para implementar todas las etapas de control en tiempo y forma real.

- Diseño y construcción de la etapa de control.

La etapa de control consta principalmente de una CPU comunicada con cada una de las aulas del edificio escolar, monitoreando así constantemente las variables que serán controladas.

- Diseño y construcción del programa para la etapa de control.

Se utilizo el programa Microsoft Visual FoxPro 9.0 para la elaboración de un software con el cual se tiene el control desde la PC, el cual se encarga de monitorear continuamente las variables que serán controladas y tomando decisiones en función de las condiciones actuales de cada aula.

- Diseño y construcción de la interfaz de comunicación entre cada aula del edificio y la CPU

La interfaz de comunicación es lo suficientemente rápida para enviar y recibir la información a la CPU en tiempo real con retardos de unos cuantos nanosegundos, estando en comunicación constante y sin interrupciones.

Esta cuenta con un microprocesador (PIC18F4550), el cual es el encargado del control de la comunicación entre la CPU y las aulas. Utilizamos este microcontrolador ya cuenta con comunicación USB y esto nos ayuda con la comunicación entre PC y aula.

- Diseño y construcción de la etapa de sensores.

La etapa de sensores entra en función cuando se de acceso al salón de clases, estos es que, cuando el acceso se dé el sistema hará una evaluación de las condiciones de iluminación y temperatura y de acuerdo a esto, se encenderán luces y climas, o se quedara tal y como esta. En esta etapa utilizamos los 74LS245 para poder mantener un control a la hora de estar checando los sensores de las diferentes aulas. Cada aula tendrá un 74LS245 entre sus sensores y la placa de control. Así por ejemplo podemos ver los sensores del aula 1 simplemente activando su respectivo 74LS245.

- Diseño y construcción de la etapa lógica-potencia para el encendido de lámpara y climas.

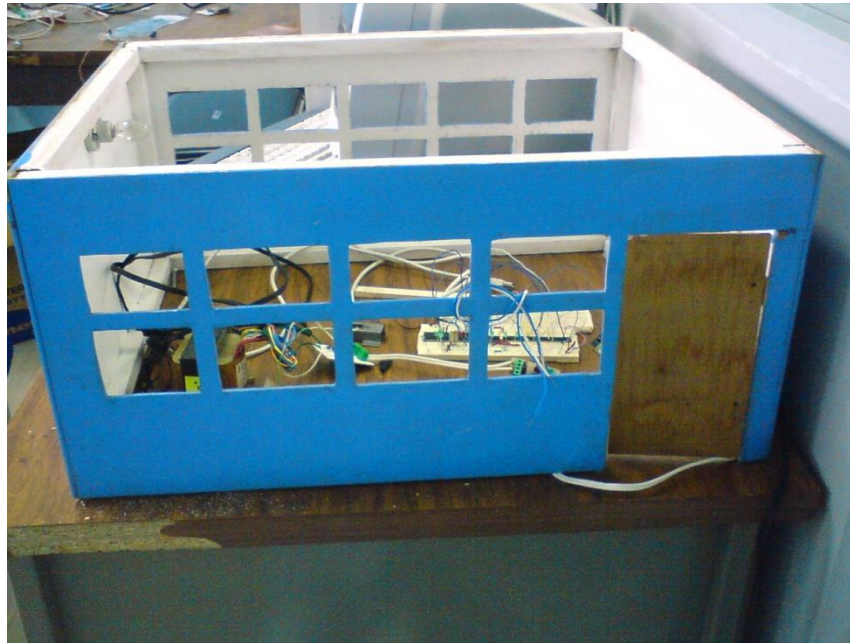
Esta etapa se encarga de establecer una comunicación entre la placa de control y los diferentes eventos a controlar. En esta etapa contamos con unos 74LS373 los cuales son registros y ayudan mantener lo q se tiene en la salida del microcontrolador. Al igual que en la etapa de sensores también cada salón contara con su respectivo registro (74LS373) para así activar o desactivar los diferentes eventos a controlar.

Para activar/desactivar cada evento a controlar se utiliza un optoacoplador (MOC3011) el cual se encarga de aislar, para así evitar daños en la etapa de control.

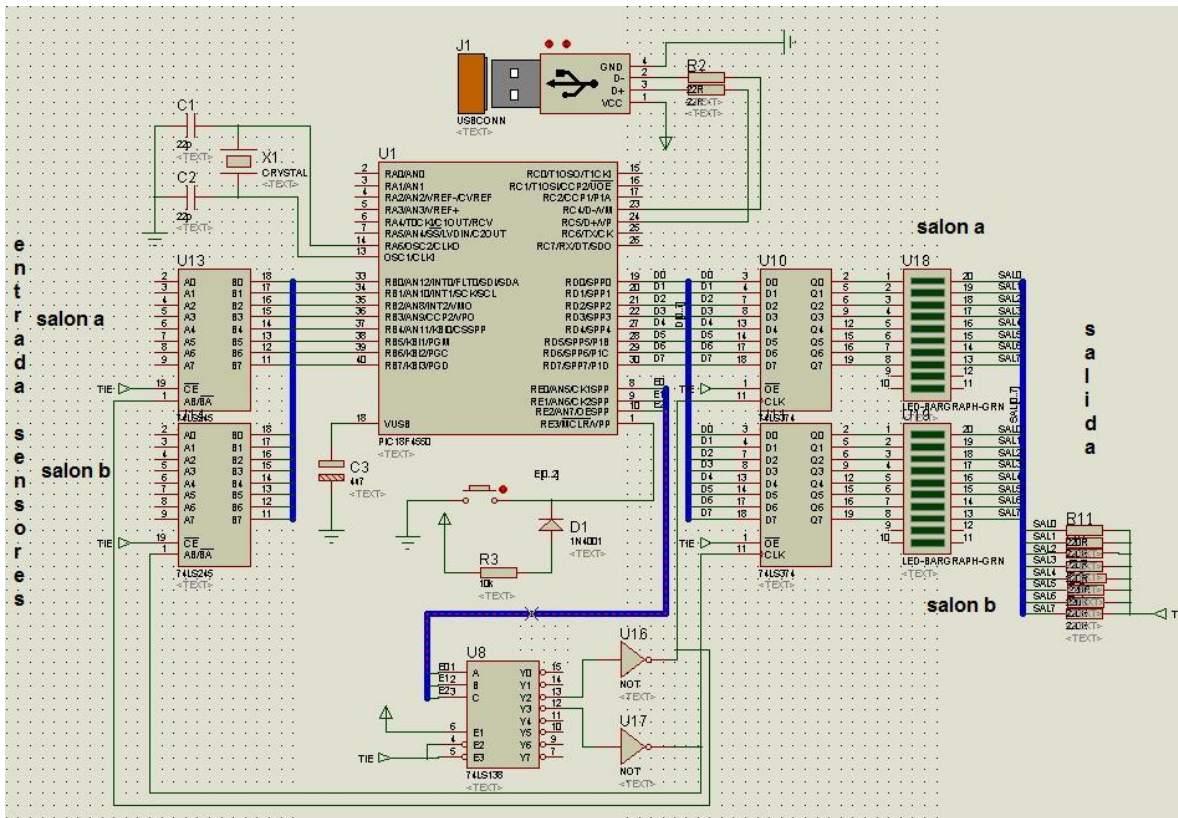
La etapa de potencia también cuenta con un MAC8D que actúa como un switch para los eventos a controlar.

# RESULTADOS, PLANOS, GRÁFICAS, PROTOTIPOS Y PROGRAMAS

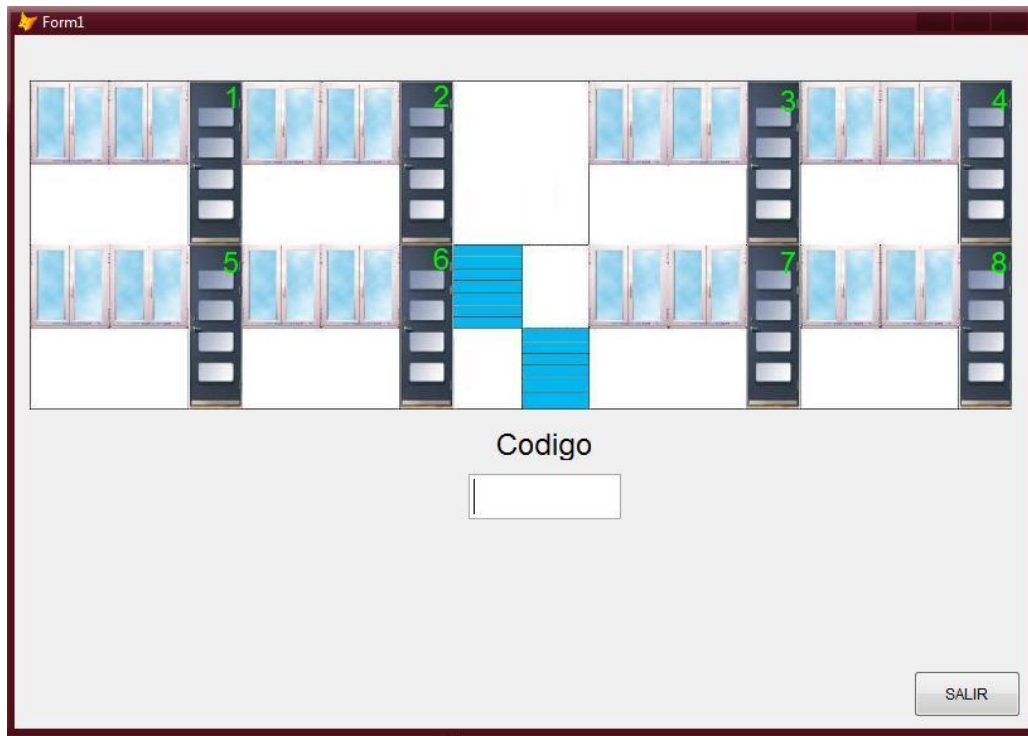
## a) Maqueta



## b) Prototipo



c) Visualización del programa



Esta es la pantalla de inicio del programa, en la cual el catedratico ingresa su codigo y dependiendo si es el dia y hora de alguna de sus clases el programa activara el aula correspondiente.

Ejemplo: si al catedratico le corresponde dar clases en el aula 1, al ingresar su codigo el aula 1 se activara, y en la pantalla se visualizara lo siguiente:

Form1

Activo

Codigo  
0000001

SALIR

Al termino de la hora de su clase el aula se desactivara automaticamente.

#### d) Base de datos

Horario2 - Microsoft Visual FoxPro

Archivo Edición Ver Herramientas Programa Tabla Ventana Ayuda

Clave	Maestro	Materia	Aula1	Lunes_e	Lunes_s	Aula2	Martes_e	Martes_s	Aula3	Miercole_e	Miercole_s	Aula4	Jueves_e	Jueves_s	Aula5	Viernes_e	Viernes_s
0000001	Maria Candelaria Gutierrez Gomez	Soft. de sistem	1	07:00	08:59	1	07:00	08:59	1	07:00	07:59						
0000002	Daniel Samayoa Penagos	Calid. del software							1	08:00	08:59	1	07:00	08:59	1	07:00	08:59
0000003	Galdino Belzario Nangos Solis	Fund. de redes	1	09:00	10:59	1	09:00	10:59	1	09:00	09:59						
0000004	Roberto Cruz Gordillo	Programacion II							1	10:00	10:59	1	09:00	10:59	1	09:00	10:59
0000004	Roberto Cruz Gordillo	Sist. de inf. I	1	11:00	12:59	1	11:00	12:59	1	11:00	11:59						
0000005									1	12:00	12:59	1	11:00	12:59	1	11:00	12:59
0000006			1	13:00	14:59	1	13:00	14:59	1	13:00	13:59						
0000007									1	14:00	14:59	1	13:00	14:59	1	13:00	14:59
0000008			1	15:00	16:59	1	15:00	16:59	1	15:00	15:59						
0000009									1	16:00	16:59	1	15:00	16:59	1	15:00	16:59
0000010			1	17:00	18:59	1	17:00	18:59	1	17:00	17:59						
0000011									1	18:00	18:59	1	17:00	18:59	1	17:00	18:59
0000012			1	19:00	20:59	1	19:00	20:59	1	19:00	19:59						
0000013									1	20:00	20:59	1	19:00	20:59	1	19:00	20:59
0000014			2	07:00	08:59	2	07:00	08:59	2	07:00	07:59						
0000015									2	08:00	08:59	2	07:00	08:59	2	07:00	08:59
0000016			2	09:00	10:59	2	09:00	10:59	2	09:00	09:59						
0000017									2	10:00	10:59	2	09:00	10:59	2	09:00	10:59
0000018			2	11:00	12:59	2	11:00	12:59	2	11:00	11:59						
0000019									2	12:00	12:59	2	11:00	12:59	2	11:00	12:59
0000020			2	13:00	14:59	2	13:00	14:59	2	13:00	13:59						
0000021									2	14:00	14:59	2	13:00	14:59	2	13:00	14:59
0000022			2	15:00	16:59	2	15:00	16:59	2	15:00	15:59						
0000023									2	16:00	16:59	2	15:00	16:59	2	15:00	16:59
0000024			2	17:00	18:59	2	17:00	18:59	2	17:00	17:59						
0000025									2	18:00	18:59	2	17:00	18:59	2	17:00	18:59
0000026			2	19:00	20:59	2	19:00	20:59	2	19:00	19:59						
0000027									2	20:00	20:59	2	19:00	20:59	2	19:00	20:59

## e) Código de Microsoft Visual FoxPro 9.0

```
cNomDia = Cday(Date())
HORA = LEFT(TIME(), 5)

IF cNomDia = "Lunes" THEN
    cNomDia = "1"
ENDIF
IF cNomDia = "Martes" THEN
    cNomDia = "2"
ENDIF
IF cNomDia = "Miércoles" THEN
    cNomDia = "3"
ENDIF
IF cNomDia = "Jueves" THEN
    cNomDia = "4"
ENDIF
IF cNomDia = "Viernes" THEN
    cNomDia = "5"
ENDIF
IF cNomDia = "Sabado" THEN
    cNomDia = "6"
ENDIF
IF cNomDia = "Domingo" THEN
    cNomDia = "7"
ENDIF

SET EXACT ON

Do Case
    Case cNomDia = "1"
        SELECT horario2
        SEEK nNumMat
        IF FOUND()
            DO WHILE ((HORA > lunes_e AND HORA >= lunes_s) OR
(HORA < lunes_e AND HORA < lunes_s))
                SKIP
                IF nNumMat != clave
                    SKIP-1
                    EXIT
                ENDIF
            ENDDO
            IF (HORA >= lunes_e AND HORA < lunes_s) THEN
                IF aula1 = "1" THEN
                    thisform.com.output = "a"
                    thisform.Activo1.Visible = .T.
                ENDIF
                IF aula1 = "2" THEN
                    thisform.com.output = "b"
                    thisform.Activo2.Visible = .T.
                ENDIF
            ENDIF
        ENDIF
    ENDIF

GO TOP
```

```

a = 1
DO WHILE a = 1
    IF HORA = lunes_s THEN
        IF aula1 = "1" THEN
            thisform.com.output = "A0"
            thisform.Activo1.Visible = .F.
        ENDIF
        IF aula1 = "2" THEN
            thisform.com.output = "B0"
            thisform.Activo2.Visible = .F.
        ENDIF
    ENDIF
    IF EOF() THEN
        a = 0
    ELSE
        SKIP
    ENDIF
ENDDO

Case cNomDia = "2"
    SELECT horario2
    SEEK nNumMat
    IF FOUND()
        DO WHILE ((HORA > martes_e AND HORA >= martes_s)
OR (HORA < martes_e AND HORA < martes_s))
            SKIP
            IF nNumMat != clave
                SKIP-1
                EXIT
            ENDIF
        ENDDO
        IF (HORA >= martes_e AND HORA < martes_s) THEN
            IF aula2 = "1" THEN
                thisform.com.output = "a"
                thisform.Activo1.Visible = .T.
            ENDIF
            IF aula2 = "2" THEN
                thisform.com.output = "b"
                thisform.Activo2.Visible = .T.
            ENDIF
        ENDIF
    ENDIF
ENDIF

GO TOP
a = 1
DO WHILE a = 1
    IF HORA = martes_s THEN
        IF aula2 = "1" THEN
            thisform.com.output = "A0"
            thisform.Activo1.Visible = .F.
        ENDIF
        IF aula2 = "2" THEN
            thisform.com.output = "B0"
            thisform.Activo2.Visible = .F.
        ENDIF
    ENDIF
    IF EOF() THEN

```

```

        a = 0
    ELSE
        SKIP
    ENDIF
ENDDO

Case cNomDia = "3"
    SELECT horario2
    SEEK nNumMat
    IF FOUND()
        DO WHILE ((HORA > miercole_e AND HORA >=
miercole_s) OR (HORA < miercole_e AND HORA < miercole_s))
            SKIP
            IF nNumMat != clave
                SKIP-1
                EXIT
            ENDIF
        ENDDO
        IF (HORA >= miercole_e AND HORA < miercole_s) THEN
            IF aula3 = "1" THEN
                thisform.com.output = "a"
                thisform.Activo1.Visible = .T.
            ENDIF
            IF aula3 = "2" THEN
                thisform.com.output = "b"
                thisform.Activo2.Visible = .T.
            ENDIF
        ENDIF
    ENDIF
ENDIF

GO TOP
a = 1
DO WHILE a = 1
    IF HORA = miercole_s THEN
        IF aula3 = "1" THEN
            thisform.com.output = "A0"
            thisform.Activo1.Visible = .F.
        ENDIF
        IF aula3 = "2" THEN
            thisform.com.output = "B0"
            thisform.Activo2.Visible = .F.
        ENDIF
    ENDIF
ENDIF

IF EOF() THEN
    a = 0
ELSE
    SKIP
ENDIF
ENDDO

Case cNomDia = "4"
    SELECT horario2
    SEEK nNumMat
    IF FOUND()
        DO WHILE ((HORA > jueves_e AND HORA >= jueves_s)
OR (HORA < jueves_e AND HORA < jueves_s))

```



```

                SKIP
                IF nNumMat != clave
                    SKIP-1
                    EXIT
                ENDIF
            ENDDO
        IF (HORA >= jueves_e AND HORA < jueves_s) THEN
            IF aula4 = "1" THEN
                thisform.com.output = "a"
                thisform.Activo1.Visible = .T.
            ENDIF
            IF aula4 = "2" THEN
                thisform.com.output = "b"
                thisform.Activo2.Visible = .T.
            ENDIF
        ENDIF
    ENDIF
ENDIF

GO TOP
a = 1
DO WHILE a = 1
    IF HORA = jueves_s THEN
        IF aula4 = "1" THEN
            thisform.com.output = "A0"
            thisform.Activo1.Visible = .F.
        ENDIF
        IF aula4 = "2" THEN
            thisform.com.output = B0"
            thisform.Activo2.Visible = .F.
        ENDIF
    ENDIF
ENDIF
IF EOF() THEN
    a = 0
ELSE
    SKIP
ENDIF
ENDDO

Case cNomDia = "5"
    SELECT horario2
        SEEK nNumMat
        IF FOUND()
            DO WHILE ((HORA > viernes_e AND HORA >=
viernes_s) OR (HORA < viernes_e AND HORA < viernes_s))
                SKIP
                IF nNumMat != clave
                    SKIP-1
                    EXIT
                ENDIF
            ENDDO
        IF (HORA >= viernes_e AND HORA < viernes_s) THEN
            IF aula5 = "1" THEN
                thisform.com.output = "a"
                thisform.Activo1.Visible = .T.
            ENDIF
            IF aula5 = "2" THEN
                thisform.com.output = "b"

```

```

                                thisform.Activo2.Visible = .T.
                                ENDIF
                                ENDIF
                                ENDIF

GO TOP
a = 1
DO WHILE a = 1
    IF HORA = viernes_s THEN
        IF aula5 = "1" THEN
            thisform.com.output = "A0"
            thisform.Activo1.Visible = .F.
        ENDIF
        IF aula5 = "2" THEN
            thisform.com.output = "B0"
            thisform.Activo2.Visible = .F.
        ENDIF
    ENDIF
    IF EOF() THEN
        a = 0
    ELSE
        SKIP
    ENDIF
ENDDO

Case cNomDia = "6"
    SELECT horario2
    SEEK nNumMat
    IF FOUND()
        DO WHILE ((HORA > sabado_e AND HORA >= sabado_s)
OR (HORA < sabado_e AND HORA < sabado_s))
            SKIP
            IF nNumMat != clave
                SKIP-1
                EXIT
            ENDIF
        ENDDO
        IF (HORA >= sabado_e AND HORA < sabado_s) THEN
            IF aula6 = "1" THEN
                thisform.com.output = "a"
                thisform.Activo1.Visible = .T.
            ENDIF
            IF aula6 = "2" THEN
                thisform.com.output = "b"
                thisform.Activo2.Visible = .T.
            ENDIF
        ENDIF
    ENDIF

GO TOP
a = 1
DO WHILE a = 1
    IF HORA = sabado_s THEN
        IF aula6 = "1" THEN
            thisform.com.output = "A0"
            thisform.Activo1.Visible = .F.
        ENDIF
    ENDIF

```

```

                IF aula6 = "2" THEN
                    thisform.com.output = "B0"
                    thisform.Activo2.Visible = .F.
                ENDIF
            ENDIF
        IF EOF() THEN
            a = 0
        ELSE
            SKIP
        ENDIF
    ENDDO

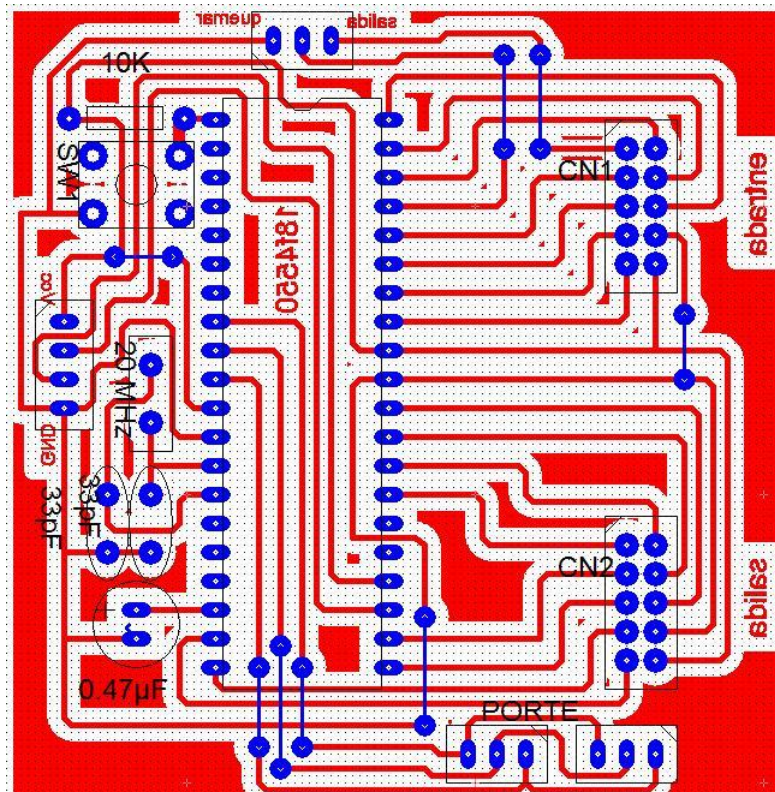
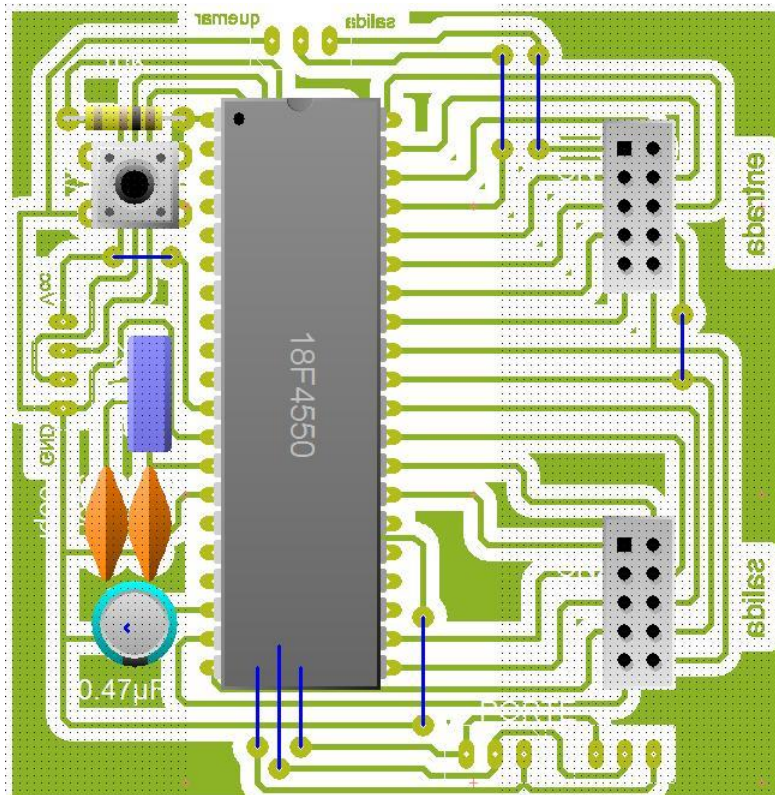
Case cNomDia = "7"
    SELECT horario2
        SEEK nNumMat
        IF FOUND()
            DO WHILE ((HORA > domingo_e AND HORA >=
domingo_s) OR (HORA < domingo_e AND HORA < domingo_s))
                SKIP
                IF nNumMat != clave
                    SKIP-1
                    EXIT
                ENDIF
            ENDDO
            IF (HORA >= domingo_e AND HORA < domingo_s) THEN
                IF aula7 = "1" THEN
                    thisform.com.output = "a"
                    thisform.Activo1.Visible = .T.
                ENDIF
                IF aula7 = "2" THEN
                    thisform.com.output = "b"
                    thisform.Activo2.Visible = .T.
                ENDIF
            ENDIF
        ENDIF
    ENDIF

    GO TOP
    a = 1
    DO WHILE a = 1
        IF HORA = domingo_s THEN
            IF aula7 = "1" THEN
                thisform.com.output = "A0"
                thisform.Activo1.Visible = .F.
            ENDIF
            IF aula7 = "2" THEN
                thisform.com.output = "B0"
                thisform.Activo2.Visible = .F.
            ENDIF
        ENDIF
    ENDIF
    IF EOF() THEN
        a = 0
    ELSE
        SKIP
    ENDIF
ENDDO

```

EndCase

f) placa de control



g) Código de PIC C Compiler para el microcontrolador PIC18F4550

```
#include <18F4550.h>
#fuses
HSPLL,NOWDT,NOPROTECT,NOLVP,NODEBUG,USBDIV,PLL5,CPUDIV1,VREGEN
#use delay(clock=4800000)
#use fast_io(B)          /// con esta instruccion evitamos que
                        /// se este configurando cada vez que
usamos
#use fast_io(a)         /// alguna instruccion de entrada o
salida
#use fast_io(c)
#include <usb_cdc.h>
#include <usb_desc_cdc.h>
#define LOADER_END 0x7FF
#define LOADER_SIZE 0x6FF
#build(reset=LOADER_END+1, interrupt=LOADER_END+9)
#org 0, LOADER_END {} // nothing will replace the bootloader memory
space
                        // END OF bootloader definition
#define USB_HID_DEVICE FALSE //deshabilitamos el uso de las
directivas HID
#define USB_EP1_TX_ENABLE USB_ENABLE_BULK //turn on EP1(EndPoint1)
for IN bulk/interrupt transfers
#define USB_EP1_RX_ENABLE USB_ENABLE_BULK //turn on EP1(EndPoint1)
for OUT bulk/interrupt transfers
#define USB_EP1_TX_SIZE 1 //size to allocate for the tx endpoint 1
buffer
#define USB_EP1_RX_SIZE 3 //size to allocate for the rx endpoint 1
buffer

byte entrada;
int8 sensor1,sensor2;
int a=0,b=0,antes=0,P1=1;

void cambio(void)
{
antes=~input_b();
output_d(antes);
}

void main()
{
set_tris_b(0x00);
set_tris_d(0xFF);
set_tris_e(0xFF);

usb_cdc_init(); // iniciamos comunicacion usb
usb_init(); //inicializamos el USB

        output_b(0);
        output_e(0);
        output_d(0);

output_e(0b011);
output_d(0b00000000);
output_e(0b000);
```

```

while(1)
{
    usb_task();
    {
        if(usb_enumerated())
        {
            entrada=usb_cdc_getc();
//-----
//entrada de codigo

            if(entrada==97)        //variable de activacion salon #1
            {
                a=97;
            }
            if(entrada==98)        //variable de activacion salon #2
            {
                b=98;
            }

            if(entrada==65)        //variable de desactivacion salon #1
            {
                A=65;
            }
            if(entrada==66)        //variable de desactivacion salon #2
            {
                B=66;
            }
//-----
//seleccion de salon

            if(entrada!=48)        //activar salón, no termino de hora
            {
                {
                    if(a==97)            //activar salon #1
                    {
                        output_e(0b001);
                        cambio();
                    }
                }
                {
                    if(b==98)            //activar salon #2
                    {
                        output_e(0b010);
                        cambio();
                    }
                }
            }
            if(P1==1)
            {
                P1=2;
            }
            else
            {
                P1=1;
            }
        }
    }
}

```

```

//-----
//apagado

    if(entrada==48)                //apagar salon termino de hora
    {

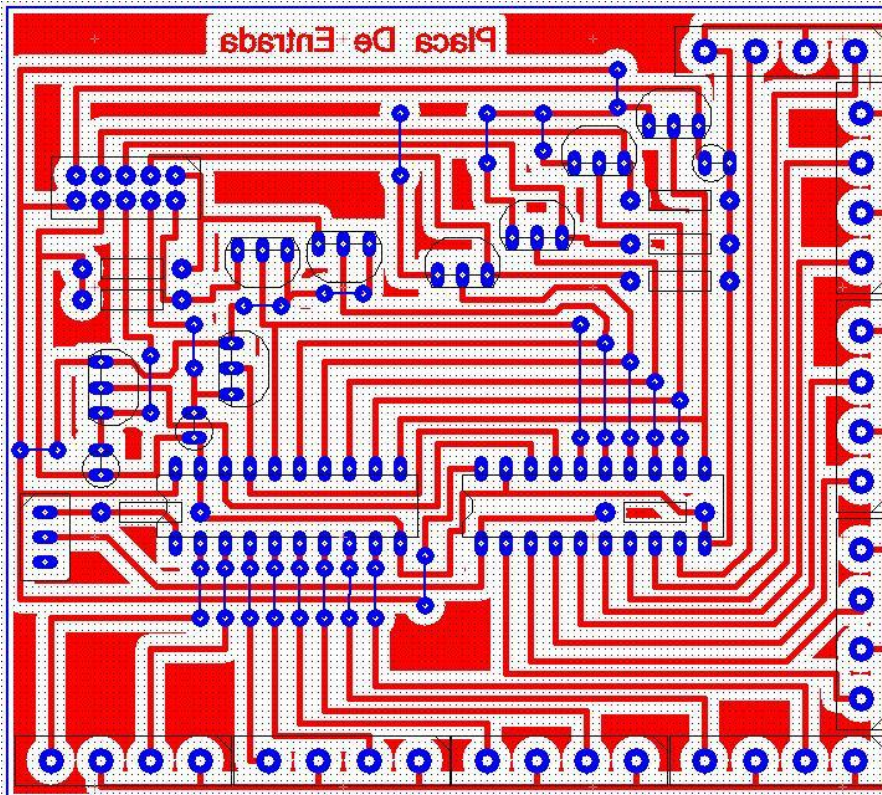
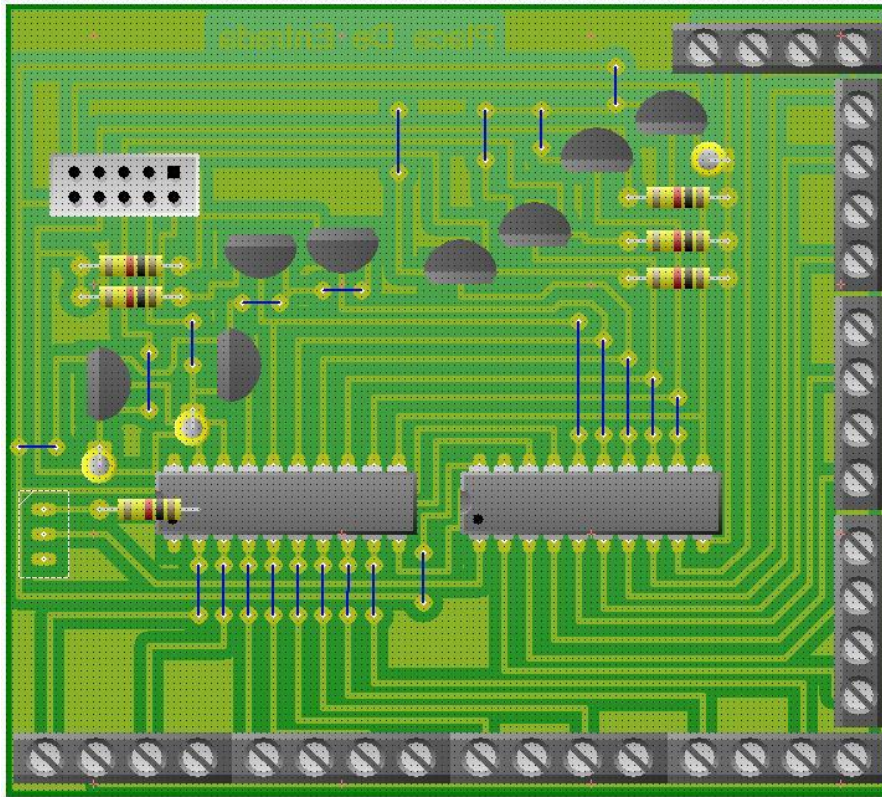
        if(A==65)                  //apagar salon #1
        {
            output_e(0b001);
            output_d(0b00000000);
            a=0;
        }
        if(B==66)                  //apagar salon #2
        {
            output_e(0b010);
            output_d(0b00000000);
            b=0;
        }
        A=0;
        B=0;
        output_e(0b000);
    }

//-----

    }
}
}
}
}

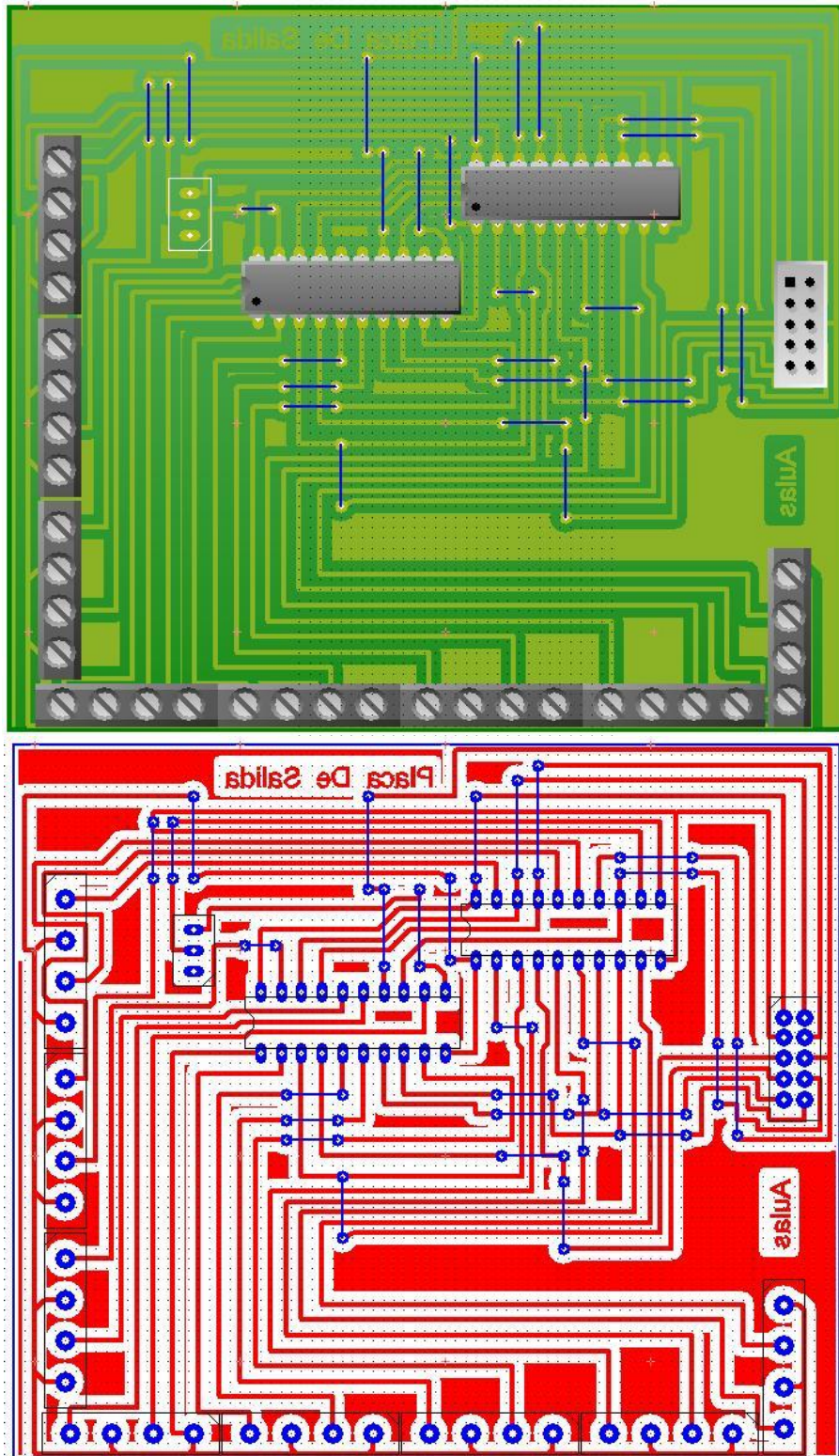
```

h) placa de entrada (sensores)





i) placa salida



## CONCLUSIONES Y RECOMENDACIONES

1. Se obtiene aproximadamente el 40 % de ahorro de energía eléctrica.
2. Abra un mayor aumento en la puntualidad, asistencia y confort de los alumnos y catedráticos en las aulas.
3. Abra modernización en los edificios y el ITTG estará entre los primeros centros educativos de nivel superior del país.
4. Se ratifica el compromiso de la Dirección del ITTG con el Sistema de Gestión de la Calidad. (ISO9000-2001)

## REFERENCIAS BIBLIOGRÁFICAS Y VIRTUALES

- [1] <http://www.edutec-america.com/home.htm>
- [2] <http://www.nexo-digital.com/vivienda-domotica-31-re.asp>
- [3] [www.InstitutoTecnológicodeChihuahua.com](http://www.InstitutoTecnológicodeChihuahua.com)
- [4] [www.microchip.com](http://www.microchip.com)
- [5] Manual del programador: Programación en Visual FoxPro
- [6] Manual de referencia: C Compiler

### Programas utilizados

1. - Microsoft Visual FoxPro 9.0
2. - PIC C Compiler
3. - Proteus 7 Professional
4. - PCB Wizard