PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR SEDE AMBATO

SERÉIS MIS TESTIGOS

ESCUELA DE SISTEMAS

Tema:

APLICACIONES DOMÓTICAS PARA EL LABORATORIO DE ELECTROLOGIA DE LA ESCUELA DE SISTEMAS DE LA PUCESA

Disertación de Grado

Previo a la obtención del título de Ingeniero en Sistemas y Computación

Autor:

ISRAEL OSWALDO LÓPEZ AGUIRRE

PONTIFICIA UNIVERSIDAD CATOLICA DEL ECUADOR
SEDE AMBATO
BIBLIOTECA

15 ENE 2010

Director:

ING. DARIO ROBAYO

Ambato – Ecuador

NOVIEMBRE – 2009

PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR

SEDE AMBATO

ESCUELA DE INGENIERIA EN SISTEMAS

HOJA DE APROBACIÓN

Tema:

APLICACIONES DOMÓTICAS PARA EL LABORATORIO DE ELECTROLOGIA DE LA PUCESA

Autor:

ISRAEL OSWALDO LÓPEZ AGUIRRE

Darío Robayo, Ing. f. _____

DIRECTOR DE DISERTACION.

Marco Polo Silva, Ing. f. _____

CALIFICADOR.

Patricio medina, Ing. f. _____

CALIFICADOR.

Santiago Acurio, Ing. f. _____

DIRECTOR ESCUELA DE SISTEMAS

Pablo Poveda, Dr. f. _____

SECRETARIO GENERAL PUCESA

SECRETARIA GENERAL
PROCURADURIA

DECLARACIÓN DE AUTENTICIDAD

Y RESPONSABILIDAD

Yo, Israel Oswaldo López Aguirre portador de la cédula de ciudadanía No. 1803223922 declaro que los resultados obtenidos en la investigación que presento como informe final, previo la obtención del título de Ingeniero en Sistemas son absolutamente originales, auténticos y personales.

En tal virtud, declaro que el contenido, las conclusiones y los efectos legales y académicos que se desprenden del trabajo propuesto de investigación y luego de la redacción de este documento son y serán de mi sola y exclusiva responsabilidad legal y académica.

Israel Oswaldo López A.

CI. 1803223922

AGRADECIMIENTO

Esta tesis de ingeniería, si bien ha requerido de esfuerzo y mucha dedicación por parte del autor y su director de tesis Ing. Darío Robayo, a quien doy un especial agradecimiento por su apoyo, no hubiese sido posible su finalización sin la cooperación desinteresada de todas y cada una de las personas que a continuación citaré y muchas de las cuales han sido un soporte muy fuerte en momentos de angustia y desesperación.

Agradecer hoy y siempre a mis padres Mónika, Oswaldo y Mario, que siempre procuraron mi bienestar, y está claro que si no fuese por el esfuerzo realizado por ellos, mi estudio no hubiese sido posible.

A mis abuelitos Elsie y Cesar, tíos Enrique, Zandra e Ileana, y resto de mi familia, porque gracias al apoyo y alegría que me brindaron me dieron la fortaleza necesaria para seguir adelante.

Igualmente agradecer a Ma. Isabel y Javier, por brindarme su apoyo, ánimo y colaboración en todo momento y sobre todo cuando más necesitaban de ellos, a demás de sus conocimientos y tiempo, sin poner nunca peros o darme negativas, sino todo lo contrario.

A mis profesores y maestros, quienes supieron compartir sus conocimientos a lo largo de mi carrera universitaria.

En general quisiera agradecer a todas y cada una de las personas que han vivido conmigo la realización de esta tesis de ingeniería, con sus altos y bajos y que no necesito nombrar porque tanto ellos como yo sabemos que desde los más profundo de mi corazón les agradezco el haberme brindado todo el apoyo, colaboración, ánimo y sobre todo cariño y amistad.

Muchas gracias de todo corazón

Israel.

DEDICATORIA

El esfuerzo y dedicación que he puesto en esta tesis, va dedicado con mucho cariño a mis hermanos Ricardo y Zamira.

También a mis padres, familia y amigos, cuyo afecto y comprensión ha sido mi inspiración y sus consejos y apoyo han sido parte fundamental de este esfuerzo.

RESUMEN

Este trabajo de tesis se realizo con el objetivo de implementar algunas aplicaciones Domóticas en el laboratorio de electrología de la Escuela de Sistemas de la Pontificia Universidad Católica del Ecuador Sede Ambato.

Lo que se ha buscado con la elaboración de este trabajo es dejar plantados los primeros pasos para investigaciones futuras de los estudiantes de la escuela de sistemas en el área de la Domótica.

Las aplicaciones que serán implementadas están dadas en el encendido y apagado de luces y televisión, apertura y cierre de persianas, y apertura de una puerta eléctrica. Además para el control de estos dispositivos eléctricos se desarrollo su propio software (Aplicación de Computador), y hardware (Circuito Electrónico).

El software desarrollado para el control de los dispositivos fue elaborado en Visual Basic 6.0 el mismo que es un lenguaje de programación que nos permite desarrollar aplicaciones de 32 bits para el sistema operativo Windows.

Así mediante la ayuda del software y hardware desarrollado podremos controlar algunos dispositivos eléctricos para llegar a la automatización de procesos dentro del hogar y la oficina que es lo que se busca cuando se habla de Domótica.

Los resultados logrados con esta tesis están basados al funcionamiento y control de dispositivos eléctricos por medio del computador, ayudándonos de esta forma a simplificar, modernizar y mejorar los procesos cotidianos que ejecutamos en nuestros hogares y oficinas, permitiéndonos de esta forma tener centralizado el funcionamiento de todos estos dispositivos.

ABSTRACT

This thesis was done in order to deploy some Home Automation applications in the electronic laboratory of the School of Systems at the Pontificia Universidad Católica del Ecuador Sede Ambato.

The aim of this work is to give the first steps for future research of students at the School of Systems in the area of Home Automation.

The applications to be implemented deal with lights and TV, opening and closing blinds and opening an electric gate. In addition to control these electrical devices its own software (Computer Application) and hardware (electronic circuit) was developed.

The software for the control of the devices was developed in Visual Basic 6.0; it is a programming language that allows us to develop 32-bit applications for the Windows operating system.

Meanwhile, using the developed hardware and software, we can control some electrical devices to achieve automation of processes within the home and office that is what we want when we talk about Home Automation.

The results achieved in this thesis are based on operation and control of electrical devices through the computer, helping to simplify, modernize and improve the processes that run daily in our homes and offices, thus allowing us to have the operation of all these centralized devices.

Tabla de Contenidos

CAPITULO I 1

PROYECTO DE TESIS 1

1.1. Introducción 1

1.2. Planteamiento del Problema 2

1.2.1. Antecedentes 2

1.2.2. Significado del Problema 3

1.2.3. Definición del Problema 4

1.3. Delimitación 4

1.4. Justificación 5

1.5. Objetivos 6

1.5.1. Objetivo General 6

1.5.2. Objetivos Específicos 6

1.6. Métodos de Investigación 6

1.6.1. Investigación Explorativa 6

1.6.2. Investigación Descriptiva 6

1.6.3. Investigación Bibliográfica 7

1.6.4. Modelo de Programación en Prototipos 7

CAPITULO II 8

MARCO TEORICO 8

2.1. Electricidad 8

2.1.1. Conceptos básicos de electricidad 8

2.1.2. Conocimientos básicos de conexiones eléctricas 8

2.1.3. Aplicaciones de la electricidad 13

2.1.4. Funcionamiento Básico de Motores 14

2.2. Electrónica 17

2.2.1. Historia 17

2.2.2. Conceptos básicos sobre circuitos electrónicos 18

2.2.3. Simbología de un circuito electrónico 19

2.2.4. Funcionamiento de componentes electrónicos 21

2.2.5. Aplicaciones de la electrónica 25

2.3. Funcionamiento y uso del Puerto Paralelo de un Computador 26

2.3.1. Historia 26

2.3.2. Tipos de puerto paralelo 26

2.3.3. El hardware del Puerto Paralelo 27

2.3.4. Como Obtener la Dirección del Puerto Instalado 30

2.4. Visual Basic 6.0 31

2.4.1. Historia 31

2.4.2. Conceptos básicos sobre Visual Basic 6.0 31

PONTIFICIA UNIVERSIDAD CATOLICA DEL ECUADOR SEDE AMBATO BIBLIOTECA

2.4.3. Nociones de programación en Visual Basic 6.0 34

2.4.4. Manipulación y control del puerto paralelo en Visual Basic 6.0 . 41

CAPITULO III 47

PROYECTO DE SOFTWARE 47

3.1. Análisis de Software 47

3.1.1. Necesidades del software 47

3.1.2. Viabilidad del software 48

3.1.3. Funciones del software 48

3.1.4. Ámbito de la información 49

3.1.5. Flujo de la información 49

3.1.6. Partición del software 50

3.1.7. Análisis estructurado 50

3.2. Diseño de Software 56

3.2.1 Diseño Arquitectónico 56

3.2.2. Diseño Procedimental 58

3.2.3. Diseño De Interfaz 61

3.2.4. Diagrama de Menú 65

3.3. Codificación del Sistema 66

3.3.1. Código para Deshabilitar botón de cerrar en los formularios 66

3.3.2. Modulo para poder abrir documentos de extensión .chm 66

3.3.3. Modulo de Funciones 67

3.3.4. Modulo para el manejo del puerto paralelo 68

3.3.5. Instrucciones Para el control del Puerto paralelo 68

3.3.6. Instrucciones para el manejo de archivos randómicos 68

3.3.7. Tipos De datos definidos 69

3.3.8. Código del Formulario para Salvapantalla 70

3.3.9. Modulo se Clase para la elaboración del reloj digital 71

3.4. Diseño del circuito electrónico 76

3.4.1. Metodología de Elaboración 76

3.4.2. Diseño de las pistas del circuito 77

3.4.3. Modelo 3d del Circuito 77

3.4.4. Componentes y Materiales a usar 79

3.5. Elaboración del Circuito electrónico 82

3.6. Implementación 86

3.6.1. Instalación de canaletas, cables y rieles 87

3.6.2. Instalación de puerta eléctrica. 87

3.6.3. Instalación de luces de los cuartos 88

3.6.4. Instalación de control para televisión 89

3.6.5. Instalación de Persianas 89

3.7. Mantenimiento y Pruebas 90

3.7.1. Pruebas del Sistema 90

CAPITULO IV 94

4.1. Conclusiones 94

4.2. Recomendaciones 95

Bibliografía 97

Glosario 99

Anexo 1 Manual de usuario 101

Tabla de Gráficos

Fig1. Símbolos Eléctricos 9

Fig2. Motor de dos polos 15

Fig3. Motor de Corriente Continua 16

Fig4. Diagrama en bloque 18

Fig5. Diagrama esquemático 19

Fig6. Diagrama en pictórico 19

Fig7. Simbología de circuitos electrónicos 20

Fig8. Ejemplo de Circuito electrónico 21

Fig9. Resistencias 21

Fig10. Condensador no polarizado/Condensador variable 22

Fig11. Diodo rectificador / Diodo emisor de luz (LED) 22

Fig12. Bobinas 22

Fig13. Pila o acumulador de batería 23

Fig14. Fusible 23

Fig15. Relé rápido / Relé con doble bobinado 24

Fig16. Transistor NPN / Transistor PNP 24

Fig17. (IC)Circuito integrado símbolo genérico 24

Fig18. Paralelo Hembra / Paralelo Macho 27

Fig19. Pines del Puerto Paralelo 27

Fig19. Entradas y Salidas del Puerto Paralelo 29

Fig20. Ventana MS-DOS ejecutando Debug.exe 30

Fig21. Barra de Controles 32

Fig22. Caja de Proyectos 32

Fig23. Caja de Propiedades 33

Fig24. Barra de Menús y Herramientas 33

Fig25. Software Para el Control del Puerto Paralelo 45

Fig26. Ingreso de Usuarios 62

Fig27. Control de Dispositivos 62

Fig28. Acción del Dispositivo Ejecutada 63

Fig29. Acción del Dispositivo Ejecutada 63

Fig30. Ventana de Mensajes 63

Fig31.Ventana Principal 64

Fig32. Pistas del circuito electrónico 77

Fig33. Modelo 3d del circuito electrónico 1 77

Fig34. Modelo 3d del circuito electrónico 2 78

Fig35. Modelo 3d del circuito electrónico 3 78

Fig36. Vista de un canal del circuito integrado ULN2803AG 79

Fig37. Componentes Electrónicos para el circuito 82

Fig38. Impresión en papel químico 82

Fig39. Baquelita 83

Fig40. Termotransferencia 83

Fig41. Preparación del cloruro férrico 84

Fig42. Corrosión del cobre mediante cloruro férrico 84

Fig43. Elaboración de orificios en la placa 85

Fig44. Colocación de los componentes en la placa 85

Fig45. Soldar compontes a la placa 86

Fig46. Placa Terminada 86

Fig47. Fijación del taco Fischer a la pared para colocar canaleta 87

Fig48. Dispositivo de Control remoto para la chapa eléctrica 88

Fig49. Transformador de corriente de 110v a 12v 88

Tablas

Tab1. Puertos Paralelos 27

Tab2. Configuración del puerto paralelo estándar 28

Tab3: Direcciones base en el BIOS 29

Tab4: Operadores Lógicos 34

Tab5: Operadores Aritméticos 35

Diagramas

Diag1. Flujo de la Información 49

Diag2. Partición del Software 50

Diag3. DFD nivel 0 50

Diag4. DFD Nivel 1 52

Diag5. DFD Nivel 2 52

Diag6. Trancición de Estados 55

Diag7. Diseño Arquitectónico 56

Diag8. Diseño Arquitectónico – Selección de Petición 57

Diag9. Diseño Arquitectónico – Petición Seleccionada 57

Diag10. Diseño Arquitectónico – Ejecucion de la Petición 58

Diag11. Flujograma de Acceso al Sistema 59

Diag12. Flujograma de Ejecución del Softwate 60

Diag13. Menús 65

Diag 14. Pruebas de Caja Blanca 91

CAPITULO I

PROYECTO DE TESIS

1.1. Introducción

Dadas las recientes tendencias en los hogares, los mismos que siempre están buscando la forma de automatizar los procesos repetitivos y cotidianos que los usuarios suelen realizar y viendo la ventaja de tener todos estos procesos centralizados, de un modo flexible y de fácil uso nace la idea y la esencia de Domótica.

El cambio a las nuevas tecnologías de e – Home ha proporcionado un vinculo para facilitarnos la vida y las tareas diarias, generando de esta forma grandes cambios dentro de nuestra sociedad, para así buscar formas más eficientes de hacer las cosas sin importar el lugar donde nos encontremos: trabajo, viaje, o en nuestro hogar.

Los servicios y usos que pueden encontrarse dentro de la Domótica son muy extensos, por ejemplo: se puede tener desde cordeles de ropa automáticos hasta simulaciones humanas para crear una presencia ficticia dentro del hogar y de esta manera tener alejado a cualquier intruso que quisiese irrumpir en el domicilio.

Estas características nos llevan a analizar la propuesta de mejorar nuestra vivienda común, a un hogar inteligente, donde exista una interfaz capaz de interactuar con el usuario para así centralizar todas las funciones cotidianas posibles, y facilitar el uso, a demás de mejorar su eficiencia.

Buscando una actualización dentro de nuestros hogares, para hacer de estos un lugar más confortable; El siguiente proyecto propone un acercamiento a esta rama de la ingeniería en sistemas, dando a conocer dentro de su contenido los principios básicos de la Domótica de una forma fácil para el aprendizaje y entendimiento de todos aquellos que busquen una guía básica en esta rama.

También se busca dejar plasmado dentro del proyecto, los primeros pasos en el campo de la Domótica, adjuntando también el software y hardware requerido, el mismo que servirá de material didáctico para todos aquellos estudiantes que gusten del estudio de esta extensa ciencia.

1.2. Planteamiento del Problema

1.2.1. Antecedentes

La Domótica es una serie de dispositivos y artefactos, tanto software como hardware, que ayudan y aportan una serie de servicios para el hogar, como pueden ser seguridad, bienestar, comunicaciones, gestión de energía eléctrica, climatización, etc.; pero no solo para el hogar sino también es muy utilizada dentro del campo empresarial para mejorar y facilitar el uso de estos recursos, naciendo así el término de edificios inteligentes.

Esta rama se ha venido implantando desde hace décadas usando las redes de comunicación como su canal de trasmisión, y hoy en la actualidad con la aparición del Internet y el desarrollo inmenso de las redes inalámbricas ha tomado un nuevo giro, permitiendo la evolución de mejores y nuevas aplicaciones demóticas.

Tomando en cuenta las características de esta rama y viendo la necesidad de implementarla en el laboratorio de electrología de la Escuela de Sistemas de la

Universidad Católica del Ecuador Sede Ambato, ya que no cuenta con aplicaciones demóticas para facilitar a los estudiantes su uso y dada la necesidad de tener un mayor control de ciertos dispositivos eléctricos y electrónicos, dentro de un laboratorio de electrología, se ha creado la posibilidad de realizar algunas adecuaciones demóticas que puedan mantener un control de dispositivos como persianas, luces y puerta.

Además en vista de la carencia de material didáctico de este tipo, este proyecto busca dejar una base en el estudio de la Domótica, ya que será el primer paso para la introducción a esta rama de la ingeniería de sistemas.

El proyecto servirá de material didáctico para todos los estudiantes de la escuela, dándoles las nociones básicas de la Domótica, y de esta forma también motivar a sus futuras investigaciones.

1.2.2. Significado del Problema

Con el desarrollo de un sistema de control para dispositivos domóticos, se logrará tener un mejor control dentro de un laboratorio teniendo centralizado el uso y control de varias funciones, para así facilitar el manejo de estas, y a su vez lograr controlar todos estos dispositivos desde un solo lugar.

Además se busca con la implementación de estas funciones mantener actualizado el laboratorio con tecnologías nuevas y acordes a los avances, para así prestar un mejor servicio de los usuarios del mismo.

También se busca, con el desarrollo de este proyecto, introducir material didáctico para el laboratorio de la escuela de sistemas, y de esta forma dejar implantado un punto de partida para el desarrollo o investigaciones futuras.

1.2.3. Definición del Problema

Cómo controlo dispositivos eléctricos y electrónicos desde un computador?

Qué dispositivos pueden ser controlados desde un computador?

Qué conocimientos de electrónica se necesitan para controlar dispositivos eléctricos y electrónicos desde el computador?

Qué tipo de software se puede usar para crear una aplicación para control y uso del puerto paralelo?

Hacia donde está orientado el desarrollo y la implementación de aplicaciones Domóticas?

1.3. Delimitación

La implementación de una aplicación Domótica está dirigida para el laboratorio de electrología de la Escuela de Sistemas de la PUCESA, tendrá un tiempo estimado de 6 meses para su desarrollo y 1 mes para su implementación.

Para el desarrollo del software de este proyecto se ha pensado hacerlo con una metodología de prototipos por la flexibilidad que este ofrece al momento de hacer evaluaciones y cambios.

Para la parte electrónica del proyecto se requerirá conocimientos básicos de diseño de circuitos y conocimientos básicos del funcionamiento de los componentes electrónicos que se va a usar.

El software del proyecto estará dedicado únicamente a controlar las funciones del sistema dentro del laboratorio las mismas que serán: control de una puerta eléctrica,

control de apertura y cierre de persianas, encendido / apagado de una bombilla de luz y encendido / apagado de un televisor.

Además la aplicación estará instalada en un computador de características básicas, las mismas que sean suficientes para poder ejecutar el software del proyecto en perfecto rendimiento. Esto se ha estipulado ya que la aplicación no requerirá de grandes recursos, y por esta razón únicamente requiere un hardware que se adapte únicamente a la misma, para así evitar la pérdida y desperdicio de recursos.

1.4. Justificación

Con el desarrollo del presente proyecto. se logrará tener un punto base o punto de partida para el estudio de esta tecnología y manejo de este tipo de hardware, permitiendo así al laboratorio de electrología de la escuela de sistemas. contar con material didáctico para el estudio de esta rama.

Existe la necesidad de controlar ciertos dispositivos eléctricos del laboratorio de electrología de la Escuela de Sistemas para dar facilidad y comodidad en su uso. Para así prestar un servicio de actualidad dentro del laboratorio y que los usuarios sientan un adelanto en tecnología dentro de su propia escuela.

A demás la implementación de este sistema brindará una mejor presencia para el laboratorio y siendo su función principal la de centralizar el control de algunos de los dispositivos eléctricos como son luces, puerta, persianas y televisión, podrá darse un tenerse un mayor control de los mismos y hacer más fácil su manipulación.

1.5. Objetivos

1.5.1. Objetivo General

Desarrollar e implementar una aplicación Domótica para el laboratorio de electrología dc la Escuela dc Sistemas de la PUCESA.

1.5.2. Objetivos Específicos

- Investigar el uso del puerto paralelo para aplicaciones de Visual Basic 6.0
- Investigar sobre el control de dispositivos eléctricos mediante un circuito electrónico.
- Centralizar el control de diversos aparatos eléctricos dentro del laboratorio de electrología.
- Migrar del uso común de los dispositivos eléctricos a una forma cómoda y de actualidad para su control.

1.6. Métodos de Investigación

1.6.1. Investigación Explorativa

La investigación para este proyecto será explorativa pues será necesario conocer la infraestructura del laboratorio de electrología de la Escuela de Sistemas para así escoger adecuadamente los dispositivos eléctricos a controlar desde la aplicación a desarrollarse. Además será necesario conocer la estructura con la que se armara el sistema para aprovechar de mejor manera el medio de trabajo.

1.6.2. Investigación Descriptiva

Para complementar la investigación explorativa se identificarán los elementos que intervienen en el sistema y las características del mismo. Se describirán las

necesidades del laboratorio de electrología de la Escuela de Sistemas de la PUCESA. Se identificarán a los usuarios del sistema, los dispositivos a controlarse desde el mismo y como se los controlaran y cuál será su funcionamiento.

1.6.3. Investigación Bibliográfica

Se usara una investigación bibliográfica, para poder obtener información acertada, específica y confiable sobre los diferentes tópicos a tratarse en este proyecto tanto en software como en hardware. Además es necesario tener a disponible el conocimiento de expertos en la rama, quienes generalmente comparten sus estudios e investigaciones en medios escritos.

1.6.4. Modelo de Programación en Prototipos

Al Usar un modelo de programación en prototipos lo que se lograra es que el Programador pueda elaborar un modelo previo de la aplicación a desarrollarse para así analizar este prototipo, ya que este contara con ciertas funciones que el sistema en su totalidad deberá cumplir. A demás a partir de estas funciones básicas del prototipo se lograra analizar, diseñar y dar una solución más efectiva al sistema completo.

CAPITULO II

MARCO TEORICO

2.1. Electricidad

2.1.1. Conceptos básicos de electricidad

2.1.2. Conocimientos básicos de conexiones eléctricas

Las conexiones eléctricas no tienen mayor dificultad lo único que se busca es trazar líneas, horizontales y/o verticales, las cuales nos permitan crear una interconexión entre los artefactos eléctricos que deseamos integrar a la conexión de nuestra estructura. Por motivos de seguridad y conducción de energía, hay ciertos parámetros y normas que debemos tener en cuenta cuando nos preparamos a realizar conexiones eléctricas, los mismos que trataremos, a continuación, de irlos explicando de una forma fácil, para poder realizar una conexión eléctrica satisfactoria sin poner en riesgo el proyecto.

2.1.2.1. Simbología para conexiones eléctricas

A continuación se mostrara una grafica con los símbolos más frecuentes dentro de las instalaciones eléctricas.

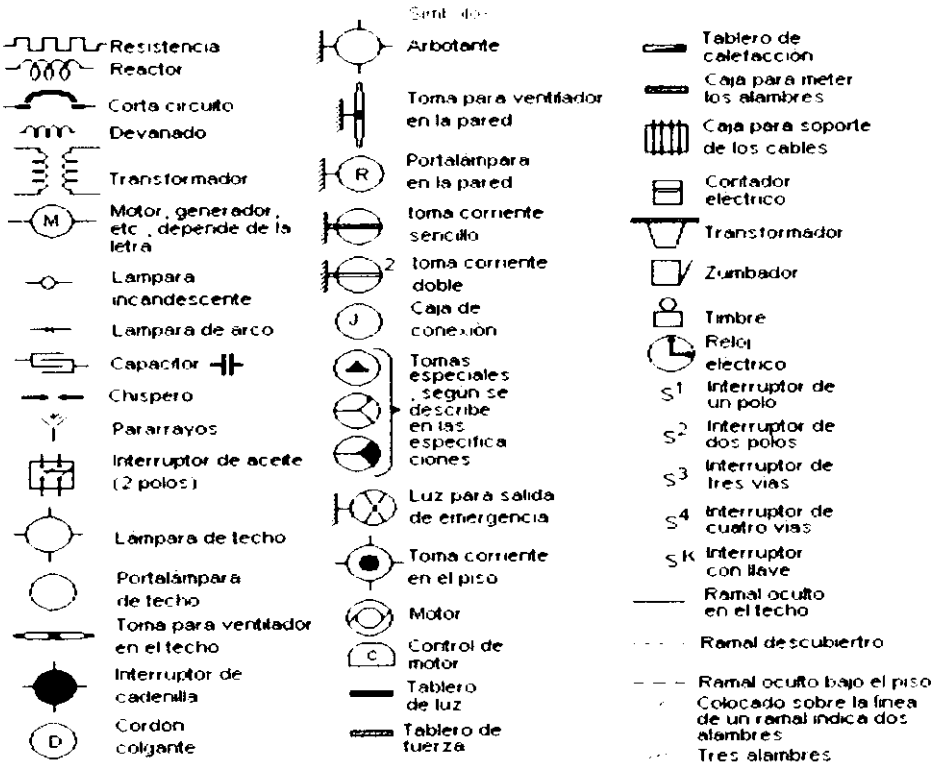


Fig1. Símbolos Eléctricos. (Fuente: www.electricidadbasica.net (Hugo Méndez))

2.1.2.2. Conceptos Principales

Corriente Alterna: La corriente alterna o CA se refiere a la manera en la cual la energía o electricidad llega a las viviendas o empresas. No tiene polaridad y es a la cual se conecta normalmente los aparatos eléctricos.

Acometida: La acometida de una instalación eléctrica se forma de la unión de una línea desde la red eléctrica general hasta la edificación o vivienda.

Medidor: Es el aparato destinado a registrar la energía eléctrica consumida por el usuario, marcara la cantidad de electricidad consumida por el usuario.

Conductores: Estos son los elementos que nos permitirán conducir el flujo eléctrico. Se emplea en las instalaciones o circuitos eléctricos para unir el generador con el receptor.

Cable: Es un conductor formado por uno o varios hilos o cordones aislado eléctricamente entre sí.

Según el número de conductores aislados que lleva un cable se denomina unipolar, si lleva uno solo; bipolar, si lleva dos hilos.

Los cables son canalizados en las instalaciones mediante tubos para protegerlos de agentes externos como los golpes, la humedad, la corrosión, etc.

Normalmente en las viviendas se usan cables de 8, 10, 12 y 14 mm de diámetro.

Interruptores, apagadores o swiches: Los interruptores son aparatos diseñados para poder conectar o interrumpir una corriente que circula por un circuito. Se accionan manualmente.

Conmutadores: Los conmutadores son aparatos que interrumpen un circuito para establecer contactos con otra parte de éste a través de un mecanismo interior que dispone de dos posiciones: conexión y desconexión.

Cajas de empalmes y derivación: Las cajas de empalme (cajetines) se utilizan para alojar las diferentes conexiones entre los conductores de la instalación. Son cajas de forma rectangular o redonda, dotadas de guías laterales para unirlas entre sí.

2.1.2.3. Normas de Seguridad en las conexiones eléctricas

Con todos los aparatos eléctricos, las herramientas, los cordones y las conexiones que utilizamos en nuestra vida diaria, es importante saber cómo utilizarlos de forma

segura, para así evitar cualquier tipo de percance cuando manipulemos nuestras conexiones eléctricas.

A continuación se darán algunos consejos para tener en cuenta cuando nos dispongamos a manejar algún tipo de conexión eléctrica.

2.1.2.3.1. Para los Electrodomésticos y/o aparatos eléctricos

Mantenga los aparatos, lejos de las tinas, los charcos, las albercas y de las manos mojadas.

Desconecte un electrodoméstico antes de limpiarlo – aún si éste se encuentra apagado, puede descargarse, y la piel mojada disminuye significativamente su resistencia a la electricidad.

Nunca ponga objetos de metal en partes "conductoras de electricidad" de los aparatos o dentro de los enchufes.

2.1.2.3.2. Para el Aislamiento de los Cordones

El aislamiento de los cordones no resistirá el calor directo, los jalones y dobleces repetidos o la humedad constante. Nunca cargue un electrodoméstico por su cordón.

No coloque un cordón debajo de un tapete o debajo de los muebles. Puede sobrecalentarse o dañarse.

2.1.2.3.3. Para evitar las Fallas en el Traslado de Energía a Tierra y Su Seguridad

Una falla en el traslado de energía a tierra sucede cuando la electricidad viaja fuera del camino que está diseñado para su transcurso, debido a un alambre debilitado o a un aparato con fallas, y ésta trata de llegar al suelo por el camino más corto. A menos que usted tenga un enchufe con un Interruptor de Circuito de Tierra, usted puede recibir una descarga seria o quemarse seriamente debido a que usted puede ser la ruta más corta al suelo.

2.1.2.3.4. Para Realizar una Conexión de Tres Puntos

Cuando se utiliza una conexión de tres puntos, el tercer punto conecta dentro de la conexión con un "alambre de tierra" que generalmente se conecta a una tubería de agua o a una viga de tierra en el panel de servicio. Como resultado de lo anterior, en un corto circuito, la electricidad debería correr hacia el suelo en lugar de a través de usted. Nunca remueva una conexión de tres puntos.

2.1.2.3.5. Prevenga los Incendios Eléctricos

Si usted ha tocado alguna vez una bombilla de luz caliente, usted sabe lo caliente que puede ponerse – hasta 300 grados Fahrenheit para una bombilla de luz de 100 vatios. Por lo que mantenga cualquier cosa que pueda quemarse alejada de las bombillas de luz, los calentadores portátiles, o los tostadores.

Apague los electrodomésticos para calentar o cocinar antes de salir de su hogar.

No sobrecargue los enchufes.

Si usted debe utilizar un cordón de extensión de forma temporal, iguale los límites de amperaje o voltaje que están marcados en el cordón y los electrodomésticos para evitar un peligro de incendio.

2.1.2.3.6. Fuera de su Hogar

Cargue y utilice las escaleras de mano y las herramientas de forma que se mantengan lejos de todos los cables de electricidad. Esté consciente de cables de electricidad cercanos la próxima vez que usted utilice una escalera de mano, que pode árboles o limpie una piscina. También tenga cuidado cuando usted instale o remueva una antena, cuando trabaje sobre la azotea, cuando cargue herramientas o tubos largos, o cuando cave en el patio.

Los enchufes eléctricos exteriores deben tener cubiertas para protegerlos del clima. Cuando se utilizan sierras, cortadores, o taladros portátiles, mantenga el cordón detrás de usted, donde no pueda cortarse. Siempre utilice enchufes con Interruptores de Circuito de Tierra para protegerlo de una descarga eléctrica seria.

2.1.3. Aplicaciones de la electricidad

Una serie de descubrimientos científicos se produjeron después de haberse descubierto la forma de generar energía eléctrica. El fenómeno electromagnético fue lo que culmino el final de la electricidad.

Una de las ventajas de la electricidad es su fácil trasporte por medio de cableados eléctricos, llevándola así hacia un lugar y luego ser utilizada. Lo primero que se logro por medio de la electricidad fue el telégrafo, el cual se convirtió, en la segunda guerra mundial, en una de las aplicaciones eléctricas económicas. La electricidad juega un papel importante en el campo de la informática ya que los dispositivos

como los transistores, la televisión, la radio, el Internet y la computación, usan esta para lograr su funcionamiento.

El uso principal de la electricidad fue puesta en práctica en los alumbrados públicos y luego los procesos industriales como los motores eléctricos, en la comunicación el teléfono y la radio. La utilización de la energía eléctrica formo parte en la utilización de todos los aparatos domésticos que necesitaban de esta para su total funcionamiento.

El principal uso de la electricidad es la que se le da en las industrias y las empresas en diversas tareas. Estas aplicaciones industriales se llevan a cabo mediante el funcionamiento de motores eléctricos de diversos tipos y potencias. En las empresas también están los aires acondicionados y calefactores los que condicionan el lugar para los trabajadores. Las señales luminosas en las calles, los semáforos, funcionan con electricidad y son conocidas como señalizaciones de seguridad, son utilizadas también en zonas industriales.

2.1.4. Funcionamiento Básico de Motores

¿Cómo funciona un motor eléctrico?: Es una bobina de cobre enrollada de tal forma que toma una forma cilíndrica la cual se electrifica y hace girar un núcleo que se compone de imanes, Esto pasa por la combinación de magnetismo con electricidad, estos son los que permiten el giro del núcleo.

2.1.4.1. Motor Eléctrico

Este es un dispositivo que permite la conversión de la energía eléctrica en energía mecánica. Estos se pueden construir de cualquier tamaño y pueden alcanzar una gran fuerza de torque

También los motores se pueden usar de una forma inversa, para así convertir la energía mecánica en energía eléctrica, debido a su potencia de giro.

2.1.4.2. Partes de un Motor

Usaremos para motivos explicativos un motor DC de dos polos. Un motor simple tiene 6 pares como se muestra a continuación:

- Una armadura o rotor. [1]
- Un conmutador.
- Cepillos.
- Un eje.
- Un Imán de campo.
- Una fuente de poder DC de algún tipo.

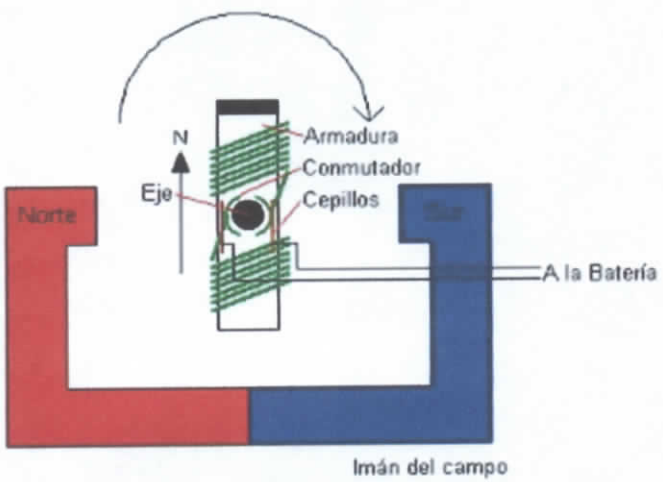


Fig2. Motor de dos polos (Fuente: http://www.info-ab.uclm.es/labelec/solar/electronica/elementos/Electricos/Motor-labels.gif)

[1] http://www.info-ab.uclm.es/labelec/solar/electronica/elementos/Electricos/Motor-labels.gif

Según la corriente eléctrica transformada los motores se pueden clasificar en:

2.1.4.3. Motores de corriente continúa

La acción entre una corriente eléctrica y un campo magnético es la que permite la conversión a energía mecánica de un motor. El campo magnético que se forma entre los polos opuestos de un imán es la que genera la fuerza que ejercida sobre algunos metales formara el movimiento. En los motores se aprovecha este tipo de fuerzas para así poder hacer girar a sus ejes, trasformando así la corriente eléctrica en un movimiento propiamente mecánico.

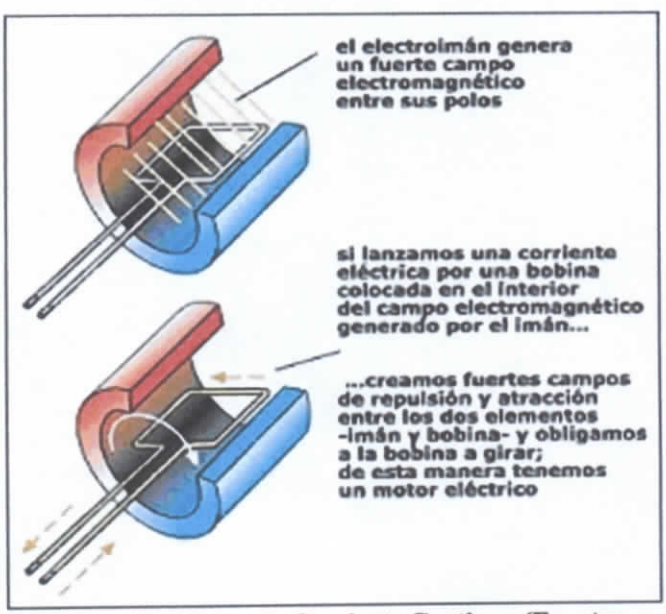


Fig3. Motor de Corriente Continua (Fuente: http://www.unesa.net/unesa/elementos/datosbasicos/largoviaje/12motorelectrico.jpg)

2.1.4.4. Motores de corriente alterna

Son similares a los motores de corriente continua, con ligeras variaciones en la fabricación de las bobinas. Según su forma de funcionar se pueden clasificar en: motores de inducción, motores sincrónicos y motores de colector.

2.1.4.4.1. Motores de inducción

Su armadura es de placas de metal magnetizable. Se genera un campo magnético que arrastra las placas de metal magnetizable, y hace que se produzca el movimiento. Este motor es el más utilizado debido a su versatilidad en construcción su bajo costo y su rendimiento, a demás de adaptare a velocidades constantes.

2.1.4.4.2. Motores sincrónicos

Estos funcionan a una velocidad directamente ligada a la corriente alterna proporcionada. Se construyen de una forma parecida a los alternadores. Pese a ser un motor de inducción más simple se ha mantenido su uso dentro del campo industrial, por su cómodo arranque y bajo costo.

2.1.4.4.3. Motores de colector

Los motores de colector resuelven el problema de potencia y velocidad del resto de motores. Según las fases se clasifican en: monofásicos y Polifásicos, los monofásicos son los más utilizados en su clase de motores serie y de repulsión.

2.2. Electrónica

2.2.1. Historia

La electrónica como ciencia tiene sus inicios en el año de 1895 con la teoría de partículas cargadas de energía llamadas electrones, que fueron parte del proceso experimental desarrollado dos años después por el científico Thomson para así comprobar su existencia.[2]

---

[2] http://www.comunidadelectronicos.com/articulos/historia.htm

Luego en 1897 se hace pública la invención del primer tubo electrónico, el cual sería el antecesor de los tubos de rayos catódicos que fueran usados por los televisores.[3]

2.2.2. Conceptos básicos sobre circuitos electrónicos

¿Qué es un circuito electrónico?: Es un conjunto de componentes eléctricos y electrónicos, los cuales están unidos en serie o en paralelo para así tener una continuidad y que la corriente eléctrica se pueda trasmitir.

Un circuito electrónico está formado por tres partes importantes, las cuales son la fuente, los conductores y la carga (esta aprovecha la energía emitida por la fuente).

Hay varias formas de representar en papel un circuito electrónico las mismas que son detalladas a continuación:[4]

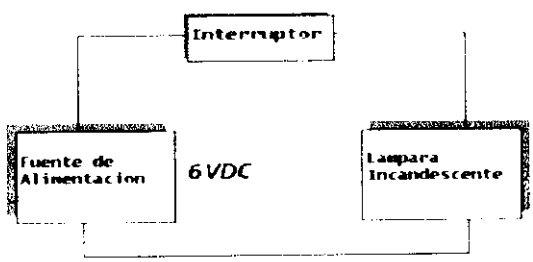


Diagrama de bloques

Fig4. Diagrama en bloque (Fuente: http://www.monografias.com/trabajos39/circuito-basico/circuito-basico.shtml)

[3] http://www.forosdeelectronica.com/tutoriales/historia.htm
[4] http://www.monografias.com/trabajos39/circuito-basico/circuito-basico.shtml

Diagrama esquemático

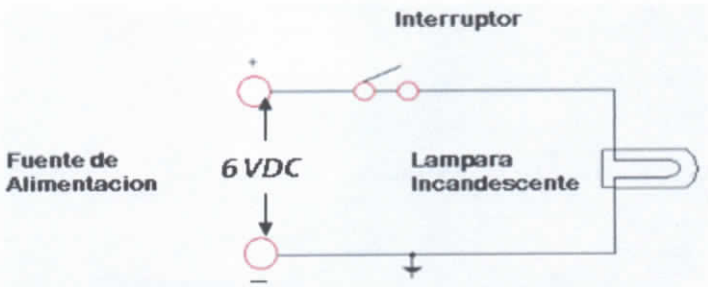


Fig5. Diagrama esquemático (Fuente: http://www.monografias.com/trabajos39/circuito-basico/circuito-basico.shtml)

Diagrama pictórico

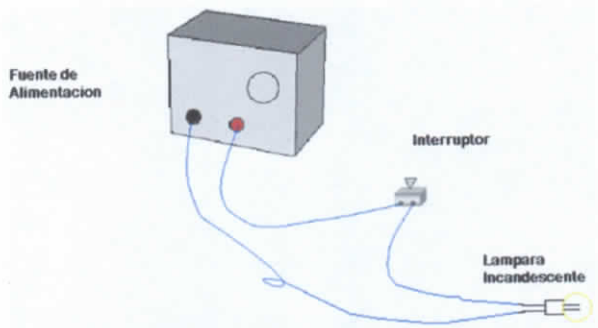


Fig6. Diagrama en pictórico (Fuente: http://www.monografias.com/trabajos39/circuito-basico/circuito-basico.shtml)

2.2.3. Simbología de un circuito electrónico

Hay formas de representar un circuito electrónico en papel, para de esta forma lograr un mejor entendimiento al diseñador que guste de la construcción o elaboración del mismo. A continuación tendremos algunos ejemplos sobre la simbología para dibujar un circuito electrónico.

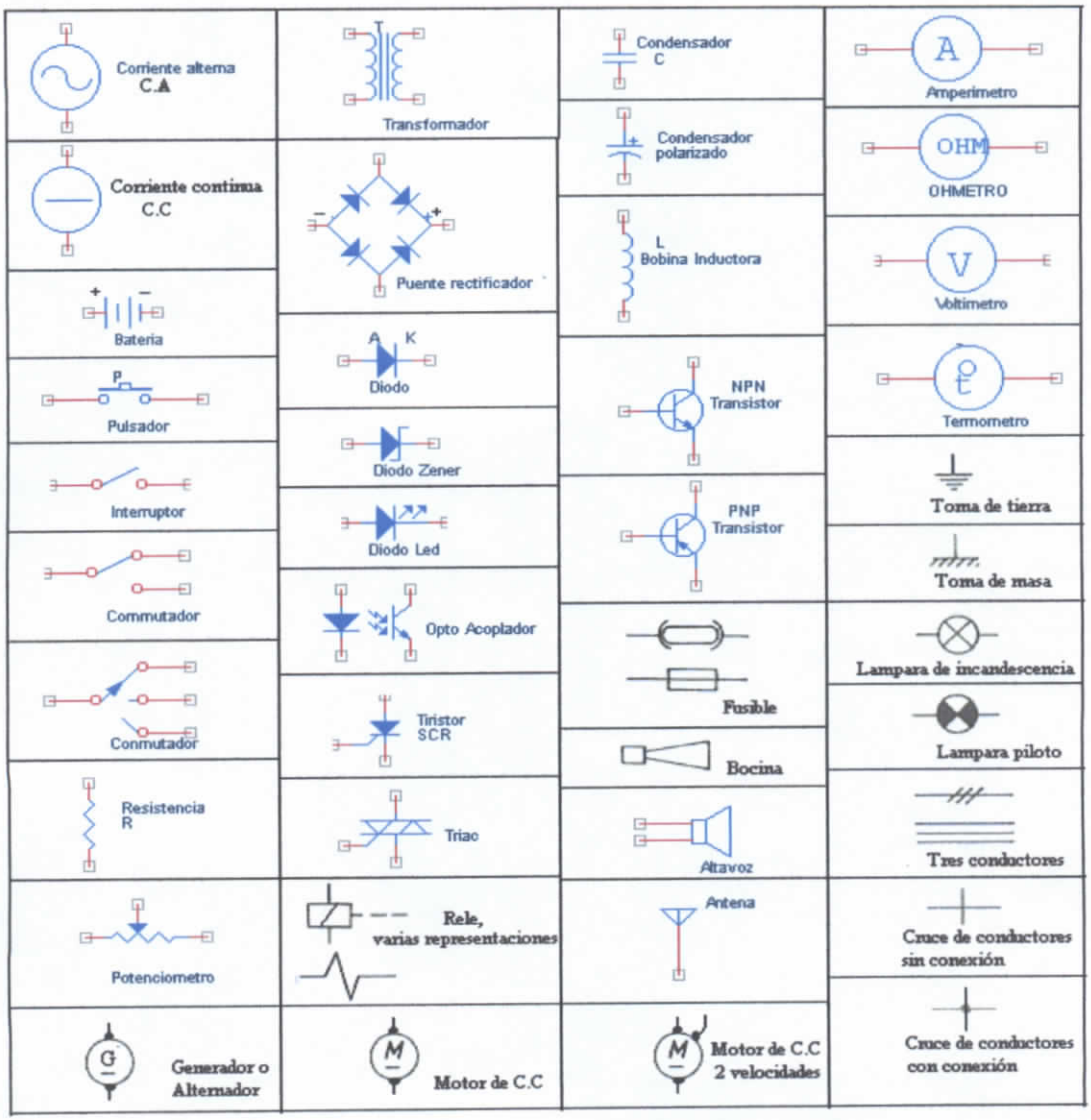


Fig7. Simbología de circuitos electrónicos (Fuente: http://mecanicavirtual.iespana.es/curso_simbologia.htm)

Usando esta simbología se pueden realizar un sinfín de circuitos electrónicos. Y usarlos para diferentes fines y funciones, a continuación dejare como ejemplo un diseño de un circuito, el mismo que sirve para probar componentes electrónicos.

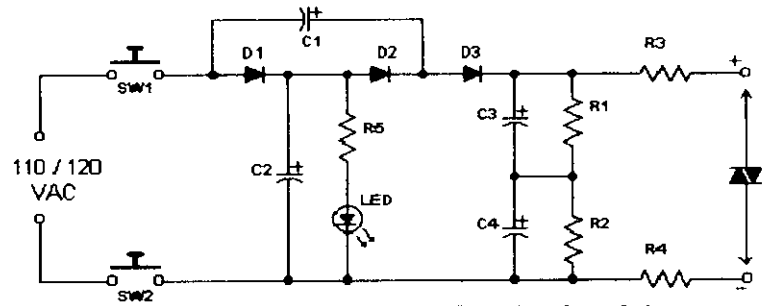


Fig8. Ejemplo de Circuito electrónico

2.2.4. Funcionamiento de componentes electrónicos[5]

Hay una diversidad de componentes electrónicos, los mismos que tienen su particular funcionamiento, aquí se detallara el funcionamiento de los componentes básicos con los cuales se trabaja en electrónica.

2.2.4.1. Resistencias

Estas son las que se oponen o degradan el paso de energía por un circuito, es decir se oponen al paso de corriente eléctrica. La resistencia de un circuito esta directamente influida o determinada por la ley de OHM, esta es la que determina cuanta corriente eléctrica pasa por un circuito y su unidad es el ohmio (Ω).



Fig9. Resistencias (Fuente: http://www.lu1dma.com.ar/ciberprofe/resistencias.jpg)

[5] http://www.american.edu.co/vs/webelectric/componentes1.html

2.2.4.2. Condensador

Este sirve para el almacenamiento de energía, es uno de los elementos pasivos de los circuitos. Conformado por laminas metálicas que en entre sí tienen un material dieléctrico (material aislante). Los condensadores simples tienen dos terminales los cuales se conectan a las láminas conductoras.



Fig10. Condensador no polarizado/Condensador variable (Fuente: http://members.fortunecity.es/electronico/tutoriales/simbologia/Condensadores.gif)

2.2.4.3. Diodo

Componente que permite el paso de corriente hacia un solo sentido. Los diodos más empleados en los circuitos electrónicos actuales son los diodos fabricados con material semiconductor.



Fig11. Diodo rectificador / Diodo emisor de luz (LED) (Fuente: http://members.fortunecity.es/electronico/tutoriales/simbologia/Diodos.gif)

2.2.4.4. Bobina

Los también llamados inductores consisten en un hilo conductor enrolladlo. Permitiendo así la creación de in campo electromagnético para oponerse a la intensidad de la corriente.



Fig12. Bobinas (Fuente: http://www.salesianos.edu/alcoy.juanxxiii/dpts/web_a_navegar_8_2007/bobinas/bobina.gif)

2.2.4.5. Pila (Acumulador, Batería)

Convierte la energía química en energía eléctrica. Las pilas consisten en un electrolito (que puede ser líquido, sólido o en pasta), un electrodo positivo y un electrodo negativo. El electrolito es un conductor; uno de los electrodos produce electrones y el otro electrodo los recibe. Al conectar los electrodos al circuito que hay que alimentar, se produce una corriente eléctrica.



Fig13. Pila o acumulador de batería (Fuente: http://www.american.edu.co/vs/webelectric/componentes1.html)

2.2.4.6. Fusible

Este componente sirve para tener protección de un exceso de corriente en el circuito. En su interior tiene un hilo de metal que se derrite a una temperatura determinada.

Fig14. Fusible (Fuente:



http://www.american.edu.co/vs/webelectric/componentes1.html)

2.2.4.7. Relé

Este componente tiene como función principal la de controlar a un dispositivo de potencia mayor desde uno que tenga una potencia inferior. Formado por un electroimán y unos contactos mecánicos que son controlados directamente por el electroimán.

El relé solo requiere cocos voltios pero sus contactos pueden estar sometidos a grandes voltios y muchos amperios. Entonces, permite que una corriente y tensión pequeñas controlen una corriente y tensión mayores.



Fig15. Relé rápido / Relé con doble bobinado (Fuente: http://www.american.edu.co/vs/webelectric/componentes1.html)

2.2.4.8. Transistores

Formados por semiconductores. Cuando se conecta a una batería directamente la corriente pasa de una forma fácil sin ningún obstáculo. Mientras que al conectarlo a una batería de forma inversa la corriente es prácticamente cero o nula.

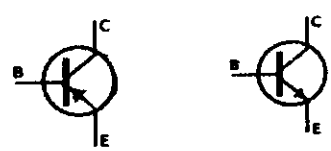


Fig16. Transistor NPN / Transistor PNP (Fuente: http://www.american.edu.co/vs/webelectric/componentes1.html)

2.2.4.9. Circuitos integrados

Son chips sobre los cuales se fabrican transistores. Con las técnicas actuales es posible fabricar millones de transistores sobre un chip.

La mayoría de circuitos integrados se llaman monolíticos por que están fabricados sobre una lámina de cristal de silicio. Los chips permiten ahorro de potencia y costo al estar fabricados de una manera compacta y no como se lo haría en un circuito de escala normal que tenga transistores individuales, y que logre la misma función del chip.

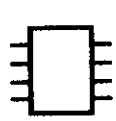


Fig17. (IC)Circuito integrado símbolo genérico (Fuente: http://www.american.edu.co/vs/webelectric/componentes1.html)

2.2.5. Aplicaciones de la electrónica

La electrónica se puede aplicar en diversas áreas contribuyendo así a facilitar muchos procesos, entre las áreas que se puede aplicar tenemos:

Computadores o electrónica digital: La automatización creciente de sistemas y procesos que conlleva necesariamente a la utilización eficiente de los computadores digitales. Las aéreas más comunes son redes de computadores, sistemas operativos y diseño de sistemas utilizando microcomputadores o microprocesadores, ya que esto conlleva a diseñar sistemas que utilicen dispositivos electrónicos.

Control de Procesos Industriales: la actividad principal se centra en la planificación, diseño, supervisión y explotación de sistemas de control automático en diversas líneas. Entre las empresas que pueden recurrir al uso de estos sistemas de control podemos encontrar a las mineras, las de pulpa y papel, las pesqueras, las textiles, las de manufacturas, etc.

Electrónica Industrial: el uso de esta energía de forma eficiente requiere de la planificación, diseño y administración de los sistemas de instrumentación, automatización y control de la energía eléctrica en una gran diversidad de procesos entre los cuales destacan los que se encuentran en empresas papeleras, pesqueras, minería, industrias manufactureras y empresas de servicios.

Telecomunicaciones: el manejo de información en gran escala requiere de la planificación, diseño y administración de los sistemas de radiodifusión, televisión, telefonía, redes de computadores, redes de fibra óptica, las redes satelitales y los más usados en nuestros días los sistemas de comunicación inalámbricos, como la telefonía celular y personal.

2.3. Funcionamiento y uso del Puerto Paralelo de un Computador

El puerto paralelo se usa en la mayoría de casos para el control de impresoras. Pero dado que este cuenta con un número de entradas y salidas de datos se puede emplear también para la experimentación de lectura y escritura de datos de algunos dispositivos o circuitos.

2.3.1. Historia

En 1981, la IBM (International Business Machines) SE incluye en las computadoras personales o PCs el Puerto Paralelo (Standart Parallel Port SPP). Para reemplazar el bajo rendimiento del puerto serial, el puerto paralelo a su vez sería utilizado como controlador de las impresoras de matriz de punto de alto desempeño. El puerto paralelo tiene la capacidad de transmitir 8 bits de datos a la vez.

En un principio se tuvo problemas con la velocidad de transferencia, dependiendo esta del software que se usase, a demás no existían estándares que rigieran la interface que fuera a usarse. Como otro punto en contra fue que su diseño solo permitía extensiones de cable cortas hasta de más o menos 1.8 m.

2.3.2. Tipos de puerto paralelo

En la actualidad se conoce cuatro tipos de puerto paralelo:

- Puerto paralelo estándar (Standart Parallel Port SPP)
- Puerto Paralelo PS/2 (bidireccional)
- Enhanced Parallel Port (EPP)
- Extended Capability Port (ECP)

En la tabla a continuación se detallara un poco más el funcionamiento de este tipo de puertos paralelos.[6]

	SPP	PS/2	EPP	ECP
Fecha de Introducción	1981	1987	1994	1994
Fabricante	IBM	IBM	Intel, Xircom y Zenith Data Systems	Hewlett Packard y Microsoft
Bidireccional	No	Si	Si	Si
DMA	No	No	No	Si
Velocidad	150Kbyte/seg.	150Kbytes/seg.	2 Mbytes/seg.	2 Mbytes/seg.

Tab1. Puertos Paralelos (Fuente: http://www.globu.net/pp/PP/historia.htm)

2.3.3. El hardware del Puerto Paralelo

El Puerto común de las computadoras personales usa un conector hembra de tipo D de 25 patitas (DB-25 S).



Fig18. Paralelo Hembra / Paralelo Macho (Fuente: http://josebitabakio.wikispaces.com/file/view/12-c25-01.jpg)

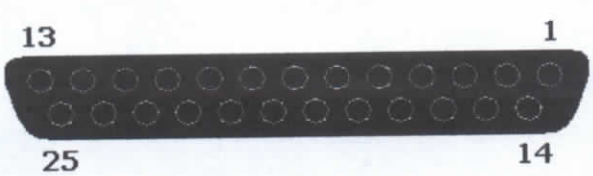


Fig19. Pines del Puerto Paralelo (Fuente: http://iie.fing.edu.uy/ense/asign/tap/gente04/obrar04/alvaroGalante_elec/paginas%20bajadas/paralelo.htm)

---

[6] http://www.globu.net/pp/PP/historia.htm

La tabla que se muestra a continuación muestra el funcionamiento de cada pin del puerto:[7]

Pin	E/S	Polaridad activa	Descripción
1	Salida	0	Strobe
2 ~ 9	Salida	-	Líneas de datos (bit 0/patita 2, bit 7/patita 9)
10	Entrada	0	Línea acknowledge (activa cuando el sistema remoto toma datos)
11	Entrada	0	Línea busy (si está activa, el sistema remoto no acepta datos)
12	Entrada	1	Línea Falta de papel (si está activa, falta papel en la impresora)
13	Entrada	1	Línea Select (si está activa, la impresora se ha seleccionado)
14	Salida	0	Línea Autofeed (si está activa, la impresora inserta una nueva línea por cada retorno de carro)
15	Entrada	0	Línea Error (si está activa, hay un error en la impresora)
16	Salida	0	Línea Init (Si se mantiene activa por al menos 50 micro-segundos, ésta señal auto inicializa la impresora)
17	Salida	0	Línea Select input (Cuando está inactiva, obliga a la impresora a salir de línea)
18 ~ 25	-	-	Tierra eléctrica

Tab2. Configuración del puerto paralelo estándar (Fuente: http://www.globu.net/pp/PP/descrip.htm)

[7] http://www.globu.net/pp/PP/descrip.htm

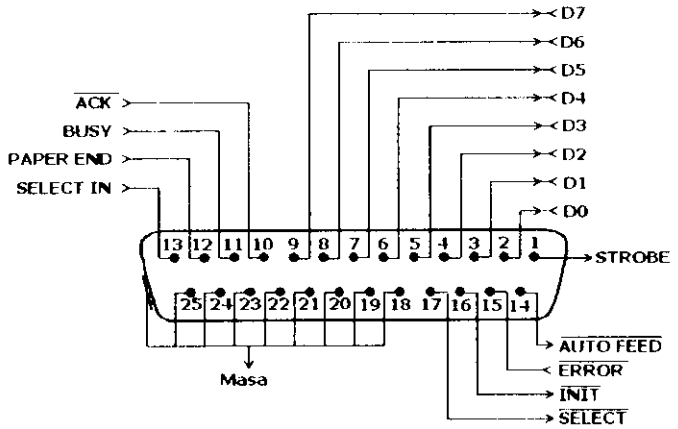


Fig19. Entradas y Salidas del Puerto Paralelo (Fuente: http://www.ucontrol.com.ar/Articulos/monitorpp/Fig05.gif)

Se logra ver que el puerto tiene 12 salidas (8 pines de datos, strobe, autofeed, init, y select input) y 5 de entrada (acknowledge, busy, falta de papel, select y error).

Se debe tomar en cuenta que un PC soportas hasta tres puertos paralelos por lo que se puede tener hasta tres juegos de registros dentro del sistema. Las direcciones conocidas para el puerto paralelo son: 0x3BCh, 0x378h y 0x278h, refiriéndonos a estas como dirección del puerto LPT1, LPT2 y LPT3, respectivamente. Cuando hay más de un puerto paralelo el PC y su BIOS asignara directamente la dirección del puerto de acuerdo a la siguiente tabla:

Dirección inicial	Función
0000:0408	Dirección base para LPT1
0000:040A	Dirección base para LPT2
0000:040C	Dirección base para LPT3
0000:040E	Dirección base para LPT4

Tab3: Direcciones base en el BIOS (Fuente: Investigador)

2.3.4. Como Obtener la Dirección del Puerto Instalado[8]

Para trabajar con el puerto primero debemos conocer la dirección base asignada a este por el BIOS del PC para ello podemos escribir "Debug" o "Debug.exe" dentro de una ventana de MS-DOS y si usamos plataformas superiores a Windows 98 en una ventana de comandos CMD. El programa ejecutado responderá con un "-" donde escribiremos sin dejar espacios en blanco "d040:08L8" y luego la tecla Enter, Seguido a esto el programa nos mostrara el o las direcciones asignadas a o los puertos paralelos del sistema, en la imagen a continuación podemos ver como lo ejecuta la maquina:

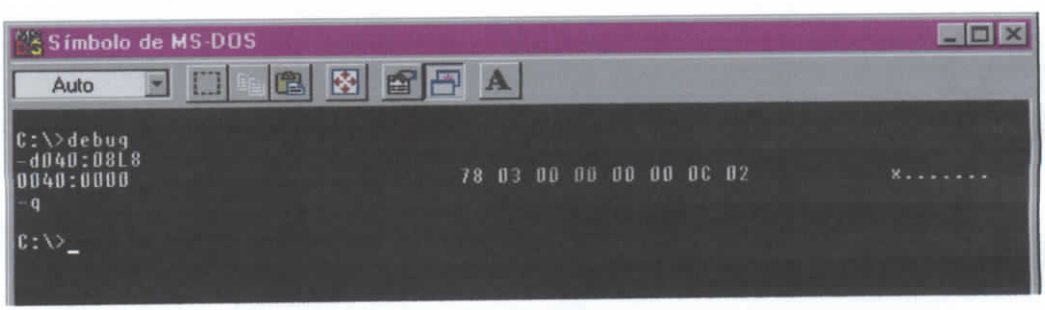


Fig20. Ventana MS-DOS ejecutando Debug.exe (Fuente: http://iie.fing.edu.uy/ense/asign/tap/gente04/obrar04/alvaroGalante_elec/paginas%20bajadas/paralelo.htm)

Como resultado de la ejecución de este programa se puede apreciar pares de números de dos dígitos (ocho en total), Estos responden al volcado de memoria que empezó en la dirección 40:0008h.

Los primeros seis pares de números representan las direcciones de puertos paralelos instalados, en la imagen anterior se puede observar el único puerto instalado que se

---

[8] http://iie.fing.edu.uy/ense/asign/tap/gente04/obrar04/alvaroGalante_elec/paginas%20bajadas/paralelo.htm

encuentra en la dirección 0x378h (78 03). Estos números se encuentran de una forma invertida ya que Intel almacena tal información en un formato de "byte de bajo orden - byte de alto orden".

2.4. Visual Basic 6.0

2.4.1. Historia[9]

Es un lenguaje desarrollado por Alan Cooper para Microsoft, Basic significa (**beginner's all-purpose simbolic instruction Codic**). Este lenguaje de programación fue creado en 1964 y tubo modificaciones hasta 1978 llegando así a un estándar. Tuvo varios nombres como "**gwbasic**" o "**QuickBasic**" hasta el que conocemos hasta la actualidad que es "**Visual Basic**", teniendo diferentes versiones hasta llegar a la .Net.

Las Aplicaciones desarrolladas dentro de visual basic versión 5 o versión 6, serán estrictamente para plataformas de 32 bits, es decir aplicaciones para Windows.

2.4.2. Conceptos básicos sobre Visual Basic 6.0

2.4.2.1. Entorno de Trabajo de Visual Basic

2.4.2.1.1. IDE – Interface de Entorno de Desarrollo

Son las distintas partes que forman nuestro entorno de trabajo, o con de las cuales el software está compuesto para brindarnos un espacio de trabajo a los usuarios;

A continuación demostraremos desde como abrir el programa visual Basic 6.0 hasta sus componentes más importantes:

---

[9] http://es.wikipedia.org/wiki/BASIC

Abrir El Programa después de haberlo Instalado, desde el menú inicio > programas > Microsoft visual Studio> Microsoft visual Basic 6.0.

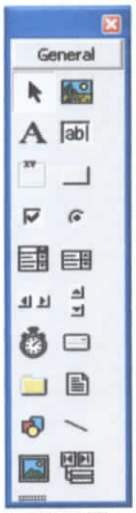


Fig21. Barra de Controles (Fuente: Investigador)

La imagen superior muestra la barra de controles. Contiene los controles estándar dentro de visual basic. Estos son los controles estándar puesto que son los que visual basic añade automáticamente al abrir una aplicación nueva, si tenemos controles de sistema, creados por el programador u otros los podemos agregar desde el *Menú Componentes* o con la combinación de teclas *Ctrl+T*.

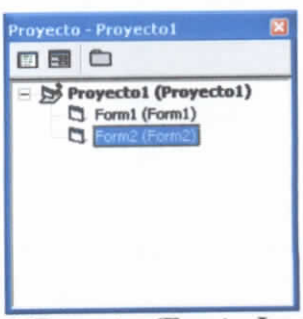


Fig22. Caja de Proyectos (Fuente: Investigador)

La imagen anterior muestra la caja de proyectos aquí se abrirá todos los proyectos con los que estemos trabajando, con su respectivo nombre y formando arboles con

sus elementos, los cuales pueden ser módulos, formularios, OCX, reportes y oros que sean parte de un mismo proyecto. A demás esta ventana nos permite navegar a través de los elementos de un proyecto creado, su código o viéndolo en fona forma de diseño con todos sus elementos

También podemos desde esta ventana, Guardar el formulario, eliminarlo y varias otras acciones.

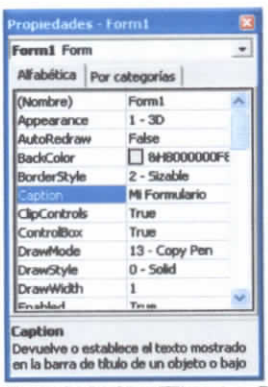


Fig23. Caja de Propiedades (Fuente: Investigador)

Desde esta ventana de propiedades podemos modificas la mayoría de propiedades o características de un objeto.

La ventana cambiara cada vez que seleccionemos un elemento distinto mostrando sus características y facultades propias de cada objeto.

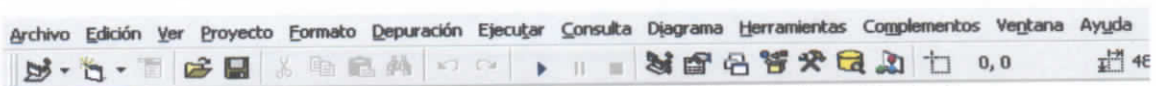


Fig24. Barra de Menús y Herramientas (Fuente: Investigador)

Desde esta barra de menús y herramientas se puede configurar todo el entorno IDE de la herramienta Visual Basic, se puede elegir el tipo de proyecto que se va a usar, etc.

2.4.3. Nociones de programación en Visual Basic 6.0

Para poderse comunicar entre el programador y el equipo de cómputo es necesario contar con un lenguaje que los dos puedan entender. Las computadoras pueden entender lenguajes muy complejos como códigos de unos y ceros, a estos se lo llama lenguajes de programación. Existen dos tipos de lenguajes de programación: los lenguajes de bojo nivel como el ensamblador y los de alto nivel como el visual basic, estos son de un fácil entendimiento para el usuario, para que luego por medio de una herramienta sean convertidos en lenguaje de máquina. Para hacer esta conversión existen los compiladores y traductores.

A continuación dejaremos los ejemplos de las instrucciones y operadores más usadas por el programador dentro de la herramienta visual basic 6.0, para el desarrollo de aplicaciones bajo plataformas Windows.

2.4.3.1. Operadores Lógicos Aritméticos y con Cadenas

Operadores lógicos

Estos operadores devuelven valores boléanos es decir cero (0) o uno (1), verdadero o falso.

Como operadores lógicos en Visual Basic tenemos:

AND	Y lógico
OR	O lógico
Xor	Xor
Not	NO lógico

Tab4: Operadores Lógicos (Fuente: Investigador)

Operadores de cadenas

Estos sirven para unir cadenas o concadenarlas se una dos métodos el "&" y el "+", con estos podremos unir frases o palabras o variables de tipo Stirling:

```
cadena1 = "Hola "

cadena2 = "Pepe"

concatenar = cadena1 & cadena2

concatenar = cadena1 + cadena2
```

Operadores Aritméticos

Son aquellos que nos permiten realizar las operaciones matemáticas y estos en Visual Basic son:

+	Suma
-	Resta
*	Multiplicación
/	División en coma flotante. Es la división normal. Devuelve un numero real si es el resultado
\	División de enteros Devuelve un numero entero, resultado de la división.
^	Potencia
Mod	Resto de la división

Tab5: Operadores Aritméticos (Fuente: Investigador)

2.4.3.2. ¿Qué es una variable?

Son lugares dentro de la memoria a los cuales se les asigna un tipo de información, estas dentro del código fuente se defines de la siguiente manera:

Dentro de un procedimiento:

```
Private Sub Form_Load()

   Variable1    = 5

   Variable2 = Variable1 * 7

   Print Variable2

End Sub
```

Fíjese que en este ejemplo primero asignamos un valor (5) a la Variable1 y después de realizar una multiplicación alojaremos el resultado en la Variable2.

Hay que tener en cuenta que si las variables son usadas dentro de un procedimiento serán variables locales que solo servirán o guardaran la información para ese procedimiento, no podrán ser usadas dentro del código fuente. Ahora bien si necesitamos de variables que se las pueda usar a lo largo del código fuente sin que estas pierdan su información, debemos declararlas en la parte superior del código de la siguiente manera:

```
Dim Variable1, Variable2 as integer

Private Sub Form_Load()

   Variable1 = 5

End Sub

Private Sub Form_Click()

   Variable2 = Variable1 + 7

End Sub
```

Los bucles en la programación son una de las partes más importantes, estos son los que nos permiten ejecutar un grupo de instrucciones y volverlas a ejecutar cuantas veces sea necesario.

2.4.3.3. El Bucle For en Visual Basic

El bucle For, su ejemplo en lenguaje Visual Basic podría ser:

```
For i=0 To 10 Step 1

Print                                                    i

Next
```

Este ejemplo imprimirá los números del 0 al 10, con su estructura de definir el valor inicial de la variable y su valor final, a demás teniendo en cuenta su incremento con "Step" (de ser 1 puede omitirse quedando For i=0 To 10), y para cerrar este bucle usaremos la palabra next.

2.4.3.4. El Bucle While en Visual Basic

```
A = 1

While A <= 10

Print A

A = A + 1

Wend
```

La instrucción anterior (WHILE/WEND) es ya de poco uso, sin embargo su uso seria: MIENTRAS la expresión sea cierta, repite todo lo que haya hasta el WEND

```
WHILE <expresión>

<instrucciones si se cumple>

WEND
```

2.4.3.5. El Bucle DO/LOOP en Visual Basic

```
Do
Print "Hola Mundo"
Loop
```

Podemos a demás agregarle algunas instrucciones o una expresión para saber hasta cuando se repetirá el bucle como se muestra a continuación.

```
A = 1
Do While A <= 10
Print A
A = A + 1
Loop
```

Si lo usamos como DO WHILE <expresión>... la forma de actuar es igual que WHILE/WEND, es decir, se evalúa la expresión y sólo en caso de que sea cierta, se ejecuta lo que está dentro del bucle, es decir entre DO y LOOP. Pero si evaluamos la expresión en LOOP, se ejecutará todo lo que hay tras el DO, como mínimo una vez y se seguirá repitiendo si se cumple la condición.

Pero con DO/LOOP también puede usarse UNTIL, es este caso se ejecutaran el grupo de instrucciones hasta que cumpla la expresión.

```
A = 1
Do Until A > 10
Print A
A = A + 1
Loop
```

2.4.3.6. Instrucción de Condición SELECT/CASE Visual Basic

Ejecuta un grupo de instrucciones, controladas por una expresión o condición. Como mostraremos en el siguiente ejemplo, definiremos una variable asignándole el valor, y buscara dentro de la instrucción select el valor de la variable hasta dar con la condición verdadera y ejecutara el grupo de instrucciones dentro de esta.

```
Dim number As Integer = 8

Select Case number

Case 1 To 5

Debug.WriteLine("Entre 1 y 5")

Case 6, 7, 8

Debug.WriteLine("Entre 6 y 8")

Case 9 To 10

Debug.WriteLine("Igual a 9 o 10")

Case Else

Debug.WriteLine("No está ente 1 ni 10")

End Select
```

2.4.3.7. Instrucción de Condición IF/(ELSE)/THEN Visual Basic

Esta instrucción también nos permite hacer condicionales, para de esta forma ejecutar un grupo de instrucciones dentro de una expresión cuando esta se cumple. Como veremos él en ejemplo siguiente después de definir dos variables una como cadena de caracteres y la otra como entero realizaremos un if comprobando una condición y cuando esta sea verdadera ejecutara el primer bloque de instrucciones de caso contrario se ejecutaran las instrucciones que están posterior al ELSE

```
Dim numero As Integer

Dim cadena As Stirling

numero = 53

If numero < 10 Then

cadena = "Cumple con la Expresión"

Else

cadena = "No Cumple con la Expresión"

End If
```

2.4.4. Manipulación y control del puerto paralelo en Visual Basic 6.0

Para poder manejar el puerto paralelo desde Visual Basic necesitaremos incluir un DLL dentro de nuestro sistema operativo este es el Inpout32.dll, este DLL se lo puede descargar gratuitamente desde internet, a continuación veremos el funcionamiento básico de este DLL .

2.4.4.1. Cómo funciona el Inpout32.dll?

Este DLL está diseñado para funcionar tanto como DLL sismo o dentro de código de usuario, en sistemas operativos como Windows 98 o en Windows Xp, siendo que se lo ocupara en el primero, Windows 98, este ocupara funciones directas para hablar con el puerto paralelo, tales como son el inp() o outp(), para escribir y leer datos del puerto; a su vez si es ocupado bajo Windows Xp este instalara un driver en el kernel para poder comunicarse con el puerto paralelo.

Este DLL está construido básicamente por dos archivos de código, el primero que es donde están insertadas las funciones para el manejo y control del puerto como tal, a demás de instalar el driver para que pueda funcionar bajo sistemas NT como lo dijimos anteriormente, este archivo es el Inpout32drv.cpp, y un segundo archivo el osversion.cpp, el cual es encargado de verificar y determinar qué sistema operativo se está usando para que el DLL sepa cómo reaccionar de acuerdo al SO que se ocupe.

Hay dos funciones importantes dentro del inpout32.dll y estas son:

- Inp32() que es la que sirve para leer datos desde el puerto paralelo
- Out32() esta escribirá datos en el puerto paralelo.

A demás existen otras funciones adicionales dentro de esta librería, a continuación se explica algunas.

- DllMain(), esta es llamada cuando el dll es levantado o dejado de usar. Cuando el dll es cargado esta revisa el sistema operativo y carga hwinterface.sys si es necesario.
- Closedriver(), esta cierra el controlador abierto, y es llamada antes de descargar el controlador.
- Opendriver(), abre el driver para que hwinteface lo pueda ocupar.
- inst() , extrae 'hwinterface.sys' desde un recurso binario y lo instala en el directorio 'systemroot\drivers' y crea un servicio. Esta función es llamada cuando la función 'Opendriver' falla en su apertura para el driver del servicio de 'hwinterface'.

- start() , comienza el servicio de hwinterface usando el Service Control Manager APIs.

- SystemVersion(), chequea el sistema operativo y regresa el código apropiado para trabajar con este.

Luego de tener instalado el dll dentro de nuestro sistema operativo lo siguiente que nos queda por hacer es abrir una nueva aplicación EXE desde Visual Basic en la cual agregaremos un modulo para poder trabajar con el dll explicado anteriormente. El código del modulo que deberá ser el siguiente.

2.4.4.1.1. Modulo para control del Inpout32.dll

```
Public Declare Function Inp Lib "inpout32.dll" _
Alias "Inp32" (ByVal PortAddress As Integer) As Integer
Public Declare Sub Out Lib "inpout32.dll" _
Alias "Out32" (ByVal PortAddress As Integer, ByVal Value As Integer)
```

A partir de este punto todo lo demás será programado dentro de los formularios de Visual Basic teniendo en cuenta las dos funciones definidas en el modulo anterior las cuales son

- Out, la cual nos servirá para enviarles datos al puerto en el siguiente formato:

  Out <Dirección del puerto>, <valor del dato del puerto>

  Out &H378, 2

- Inp, esta servirá para leer los datos del puerto, de la siguiente manera:

  <Variable de tipo entero> = Inp <Dirección del puerto>

  Dato_recojido = Inp &H378

2.4.4.1.1. Ejemplo de programa de visual Basic 6.0 para leer un dato del puerto

Lo primero que hacemos es abrir una aplicación de tipo EXE, a la cual agregamos un modulo con el código anterior, luego creamos dos botones (command Button) en el formulario (form), a los cuales les agregamos respectivamente el siguiente código:

Botón 1

```
Private Sub Command1_Click()
Out &H378, 0
End Sub
```

Botón 2

```
Private Sub Command2_Click()
Out &H378, 1
End Sub
```

Luego ejecutamos el código, y tendremos un programa que nos permite apagar y prender un punto del Puerto paralelo, ha este le podremos agregar un circuito que nos permita controlar un led, y así apagar y prender un led.

2.4.4.1.2. Ejemplo de programa de visual Basic 6.0 para control de todos los bits del puerto[10]

Con el siguiente ejemplo aprenderemos como se puede manejar el puerto paralelo por medio de vb, el DLL input32 y un modulo al que agregaremos sentencias para control del DLL.

Lo que se requiere hacer es controlar todas las salidas del puerto paralelo del computador. Disponemos de 8 bits de salida o lo que es igual a 1byte.

---

[10] http://www.taringa.net/posts/ebooks-tutoriales/1359294/Puerto-paralelo-y-vb-6.html

Hará encender todas las salidas debemos pensar en el puerto como un numero binario, siendo los 1 la representación un estado "alto" y los 0 un estado "bajo".

Si buscamos encender todos los leds enviaremos un "11111111", podemos mandarlo en decimal y el mismo programa lo pasa a binario, enviaremos entonces un 255.

000=00000000
001=00000001
002=00000010
...
255=11111111

Software para este envío de datos:

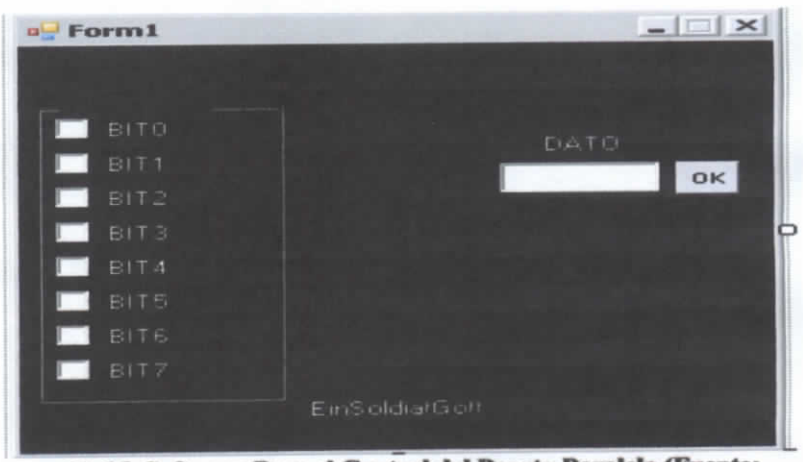


Fig25. Software Para el Control del Puerto Paralelo (Fuente: http://www.forosdeelectronica.com/upload/EinSoldiatFiles/formlptout.gif)

Los Bits pueden encenderse o apagarse desde los 8 checkbox o puede ingresarse el dato en el texbox.

Agregar 1 checkbox y copiarlo y pegarlo 7 veces para hacer una matriz, entonces se simplifica el trabajo pues solo damos la instrucción una vez y solo debemos indexarlos para hacer referencia a cada uno.

Crear el modulo para el manejo del inpout32.dll.

```
Public Declare Function Inp Lib "inpout32.dll" _
Alias "Inp32" (ByVal PortAddress As Integer) As Integer
Public Declare Sub Out Lib "inpout32.dll" _
Alias "Out32" (ByVal PortAddress As Integer, ByVal Value As Integer)
```

- Realizar un código para que revise los checkbox

```
Private Sub revisa()
contador = 0
dato = 0
While contador < 8
        If Check1(contador).Value = 1 Then
        arreglo(contador + 1) = 1
        Else
                arreglo(contador + 1) = 0
                dato = dato + (arreglo(contador + 1) * (2 ^ contador))
                contador = contador + 1
        End if
Wend
Text1 = dato
Out &H378, dato
End Sub

Private Sub Check1_Click(Index As Integer)
revisa
End Sub
```

CAPITULO III

PROYECTO DE SOFTWARE

3.1. Proyecto de Software

3.1.1. Necesidades del software

El usuario requiere tener un sistema en donde pueda controlar ciertos dispositivos eléctricos los cuales son: puerta, persianas, luces y televisión, para esto requiere mantener un contacto con estos desde el computador, centralizando así su manejo. A demás busca un software donde pueda tener una interfaz amigable, para el uso de estos dispositivos, para con esto facilitar su uso, y dar un aspecto moderno y electrónico a su vivienda.

Siendo así el software a diseñar y programar contara con las siguientes necesidad

Será necesario conocer el lenguaje de programación en el cual se codificara para tener un software de calidad con una codificación bien realizada evitando por ejemplo la redundancia.

Será a demás apropiado que conozcamos la información que se va a manejar en el sistema para poder realizar una mejor función del mismo. Es decir debemos estar al tanto de cómo se maneja manualmente el sistema que vamos a automatizar.

Contar con el ambiente de trabajo necesario donde se implantara el sistema para así poder hacer las pruebas reglamentarias, e instalar todos los dispositivos de hardware necesarios.

3.1.2. Viabilidad del software

3.1.2.1. Legal

Contener las debidas licencias de los programas o lenguajes en los que vayamos a trabajar.

Las funciones que realizará el software estarán enmarcadas dentro de procesos legales.

3.1.2.2. Financiera

Para lograr la viabilidad financiera, debemos mantenernos dentro del presupuesto propuesto.

3.1.2.3. De recursos

El sistema es viable en cuestión recursos puesto que ya contamos con lo necesario en hardware y software para la elaboración del mismo, como son los lenguajes de programación, el computador, y los componentes electrónicos.

Además utilizaremos algunos recursos adicionales como son el Internet y fuentes de conocimiento como bibliotecas.

3.1.3. Funciones del software

Maneja desde el computador, dispositivos eléctricos, tales como, una luz, persianas, puerta eléctrica y un televisor.

- El sistema será capaz de apagar y encender un televisor

- Realizará el apagado y encendido de una bombilla eléctrica
- Permitirá la apertura y cierre de persianas.
- Servirá para abrir y cerrar una puerta eléctrica

Enviara pulsos por el puerto paralelo hacia un circuito electrónico el cual nos permitirá controlar los elementos electros.

Nos presentará una interfaz amigable con la que el usuario tendrá un fácil desenvolvimiento.

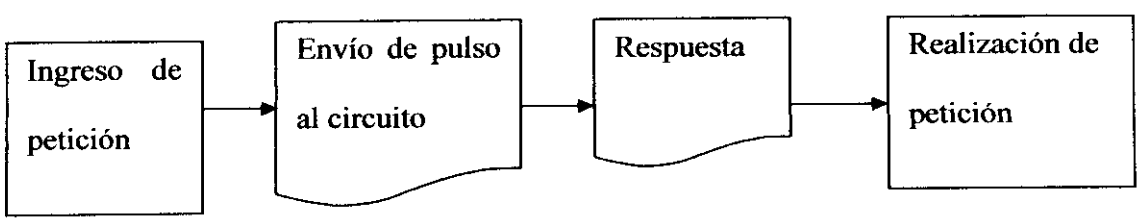
3.1.4. Ámbito de la información

3.1.4.1. Tipo de la información

Esta sistema utilizara manejara solo la información emitida por el puerto paralelo para con estos pulsos poder controlar mediante un software, todo el manejo eléctrico correspondiente a los dispositivos a ocuparse.

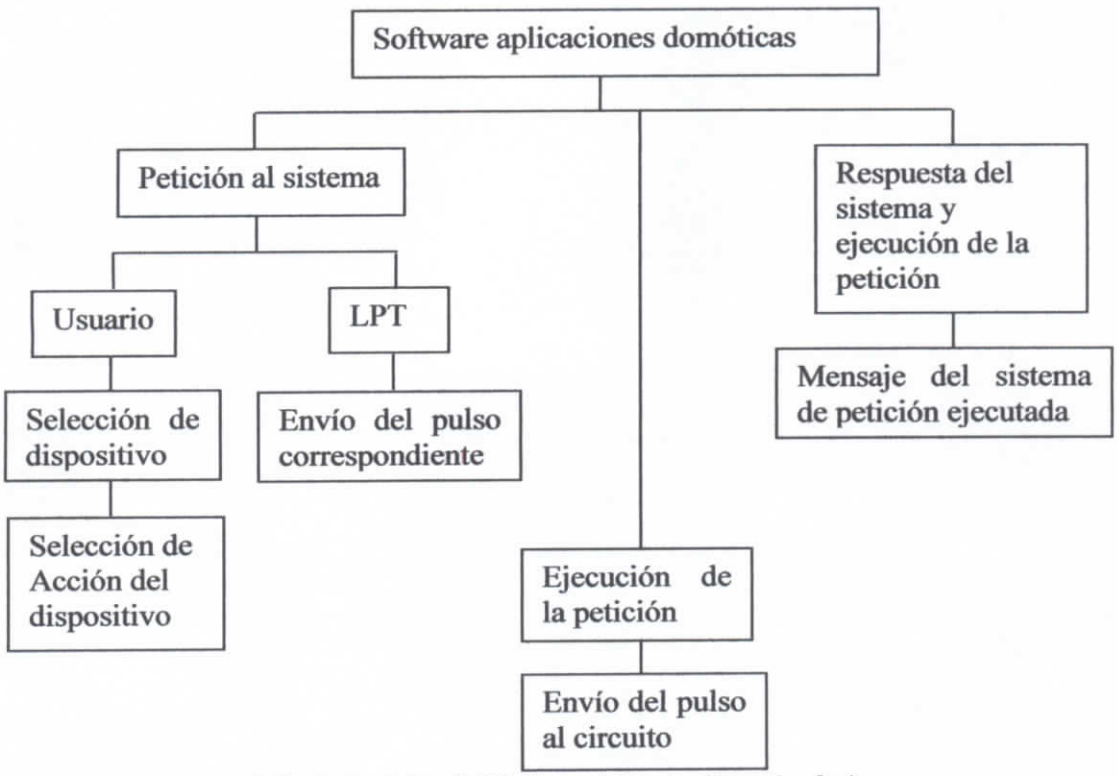
3.1.5. Flujo de la información

El flujo de información puede estar definido por el siguiente diagrama siendo el principal para poder desglosarlo posteriormente.



Diag1. Flujo de la Información (Fuente: Investigador)

3.1.6. Partición del software

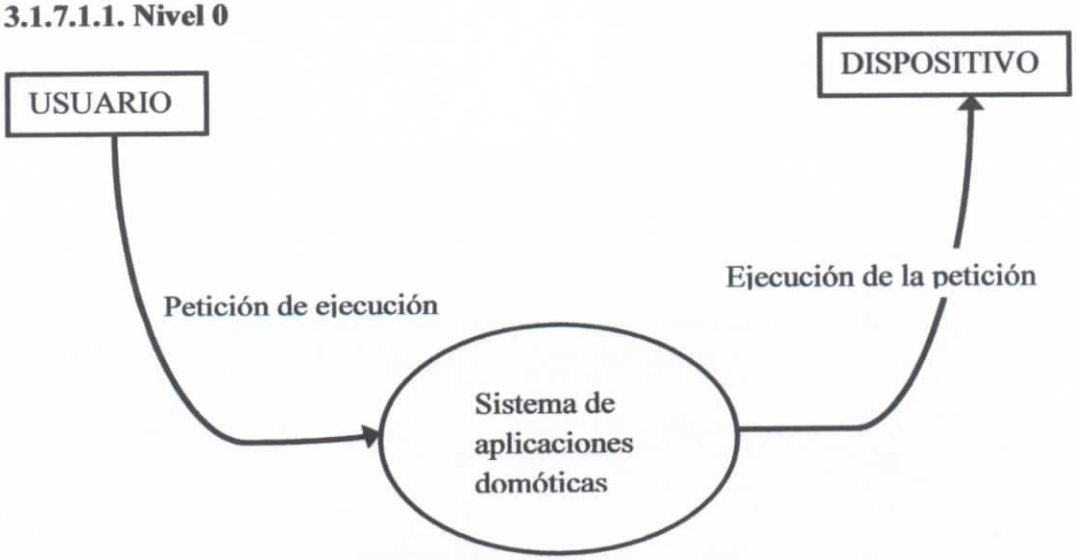


Diag2. Partición del Software (Fuente: Investigador)

3.1.7. Análisis estructurado

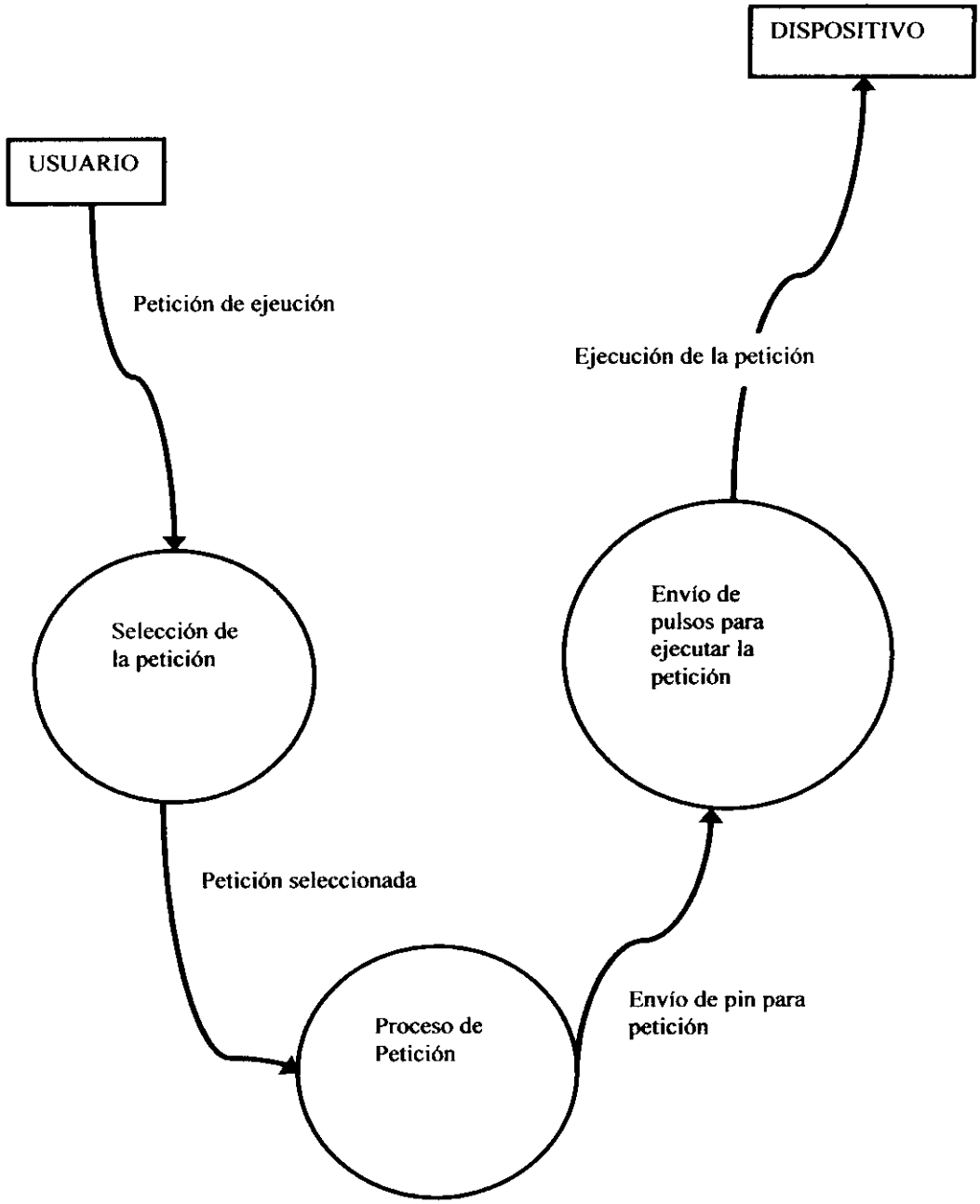
3.1.7.1. Diagramas de flujo de datos

3.1.7.1.1. Nivel 0



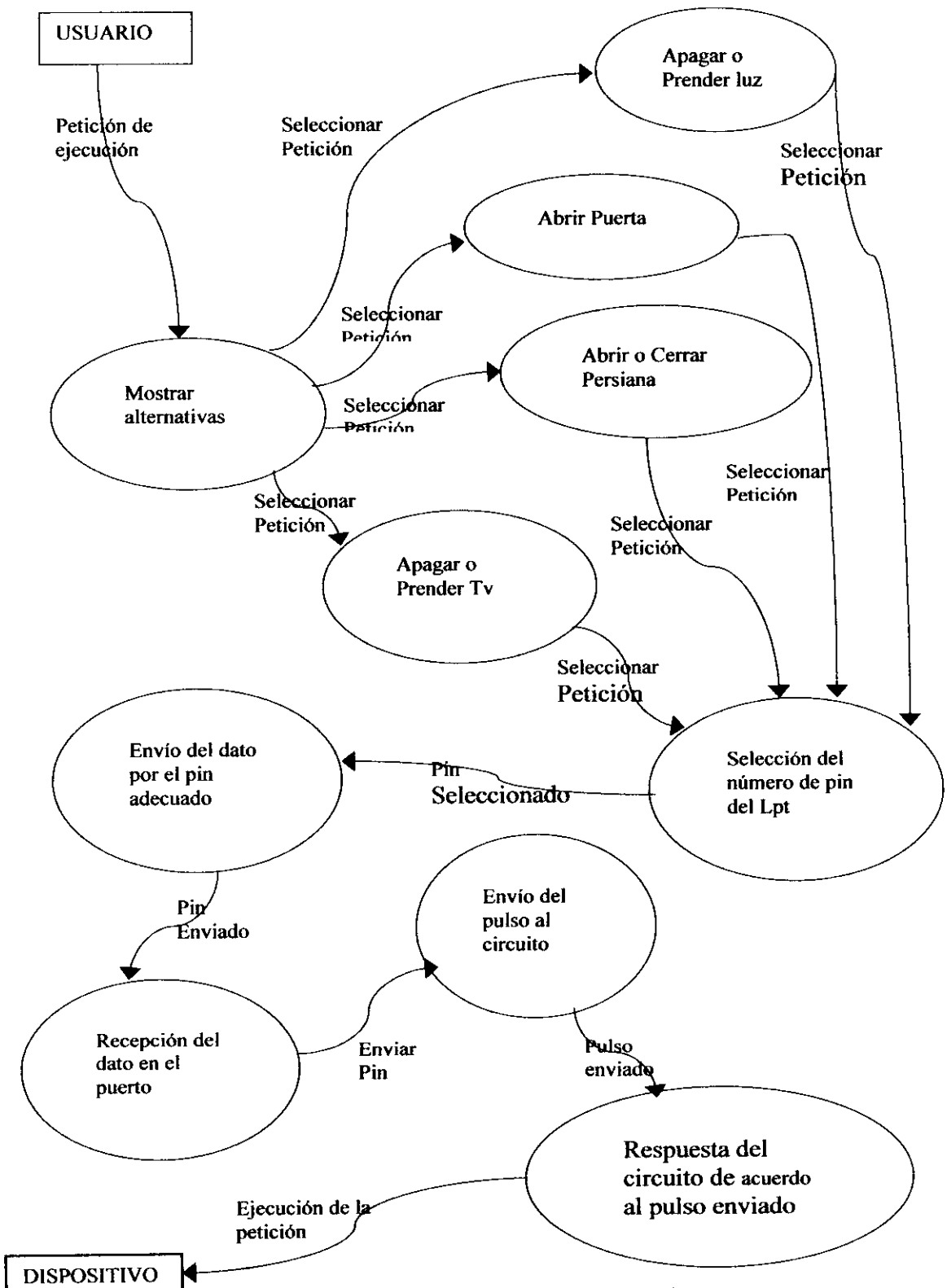
Diag3. DFD nivel 0 (Fuente: Investigador)

3.1.7.1.2. Nivel 1



Diag4. DFD Nivel 1 (Fuente: Investigador)

3.1.7.1.3. Nivel 2



Diag5. DFD Nivel 2 (Fuente: Investigador)

3.1.7.2. Diccionario de requisitos

3.1.7.2.1. Datos de entrada

Los datos que se van a ingresar en el sistema de aplicaciones demóticas serán:

Escoger el dispositivo eléctrico a controlar

Determinar si apagar o encender (Caso de Tv, Luz)

Determinar si abrir o cerrar (Caso de persianas)

Seleccionar la opción abrir (Caso Puerta)

3.1.7.2.2. Datos de salida

Los datos de salida que tendremos del sistema serán:

Envío según la petición de control del dispositivo eléctrico al puerto paralelo para su ejecución

3.1.7.2.3. Procesos

Entre los procesos podemos contemplar:

Envío del pulso desde el puerto paralelo hacia el circuito para ejecución de la petición.

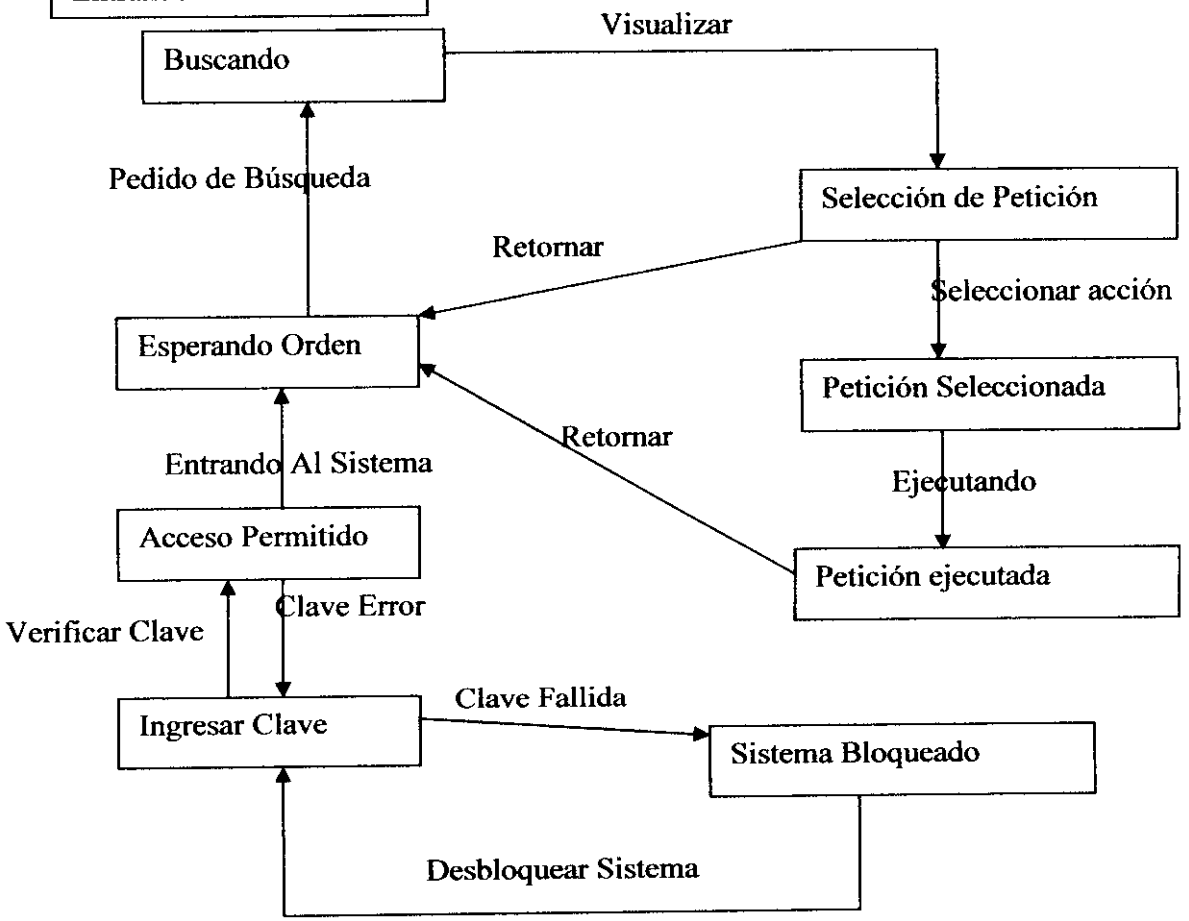
3.1.7.2.4. Automatización

Se requiere automatizar el uso de una habitación, siendo los procesos manuales a automatizar: el prendido y apagado de luces y tv, la apertura y cierre de persianas y abrir una puerta eléctrica, teniendo en cuenta el proceso manual que se realiza para hacer dichas acciones, se generara el sistema de computador para manejar y

centralizar estos procesos, a demás se construirá el circuito que sirva de interfaz entre el computador y los dispositivos eléctricos, permitiéndonos así tener control sobre los mismos.

3.1.7.3. Diagrama de transición de estados

Acciones	Estados
Pedido de búsqueda	Esperando Orden
Visualizar	Buscando
Retornar	Selección de Petición
Seleccionar acción	Petición Seleccionada
Ejecutando	Ejecutar Petición
Clave fallida	Sistema Bloqueado
Desbloquear sistema	Ingresar Clave
Verificar clave	Acceso Permitido
Clave error	
Entrando al sistema	

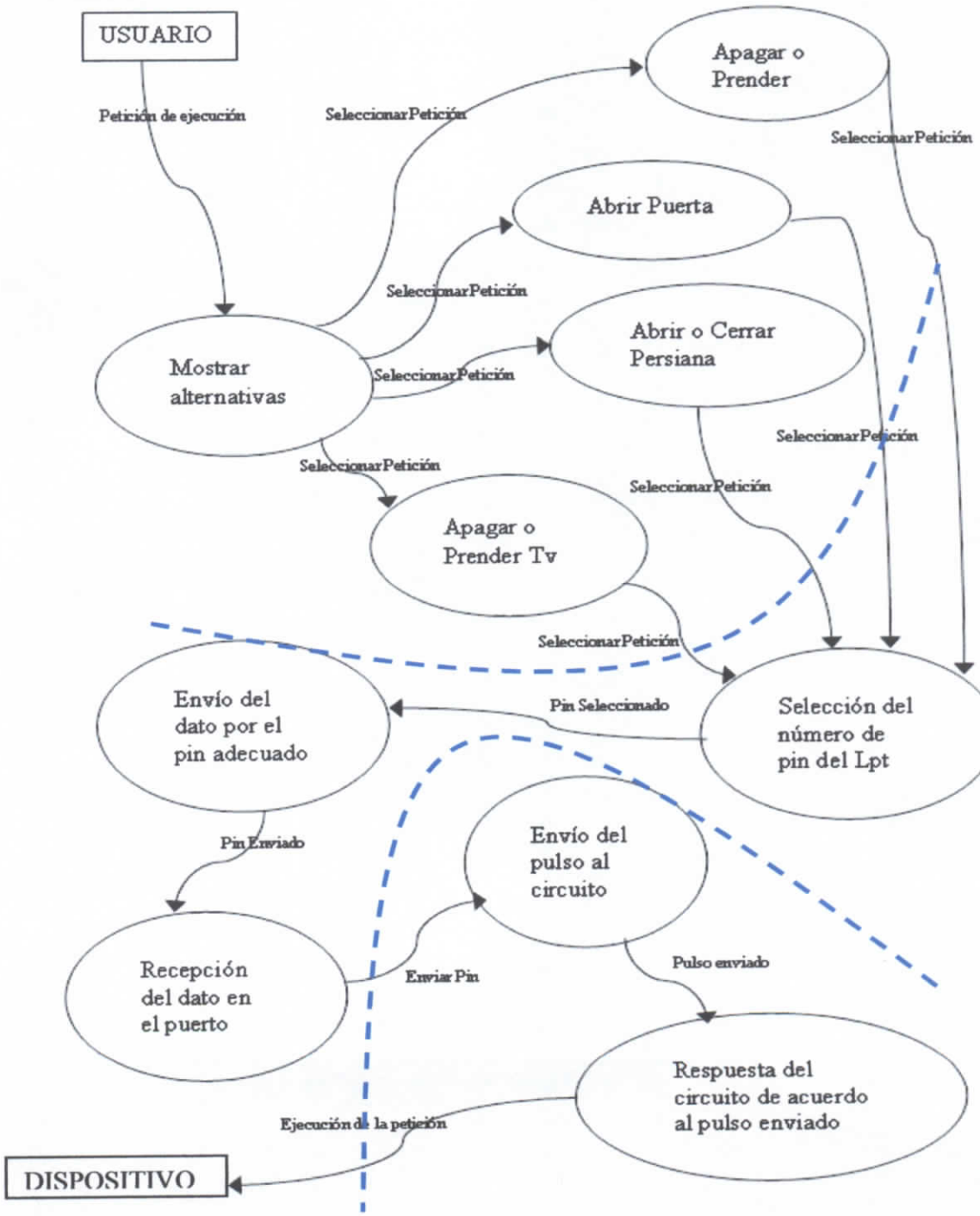


Diag6. Trancición de Estados (Fuente: Investigador)

3.2. Diseño de Software

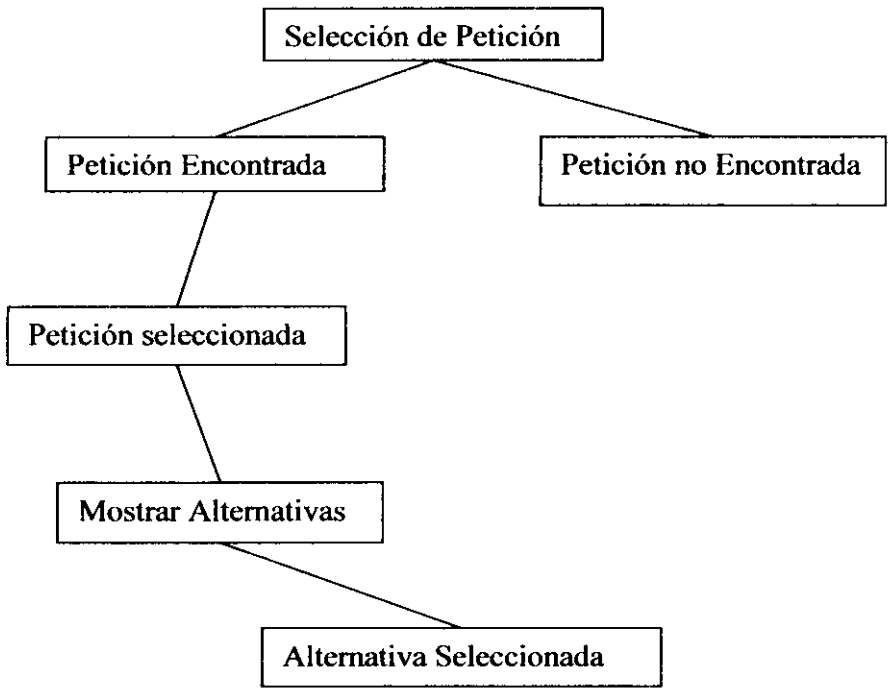
3.2.1 Diseño Arquitectónico

3.2.1.1. DFD Nivel 2



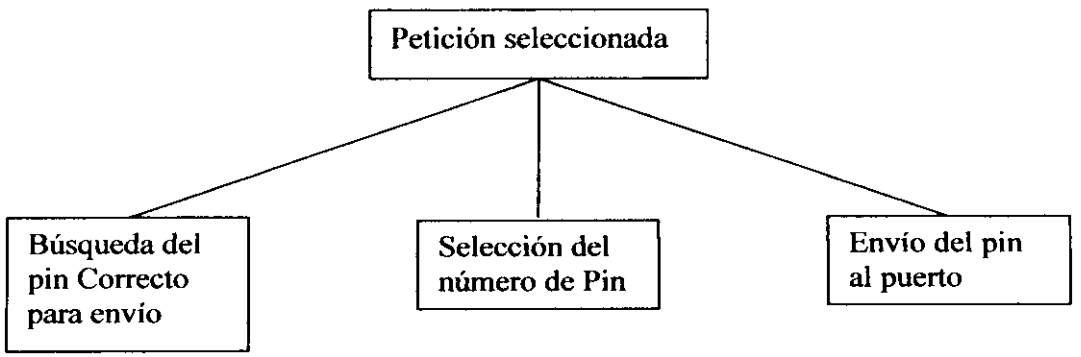
Diag7. Diseño Arquitectónico (Fuente: Investigador)

3.2.1.1.1. Diseño Arquitectónico (Selección de Petición)



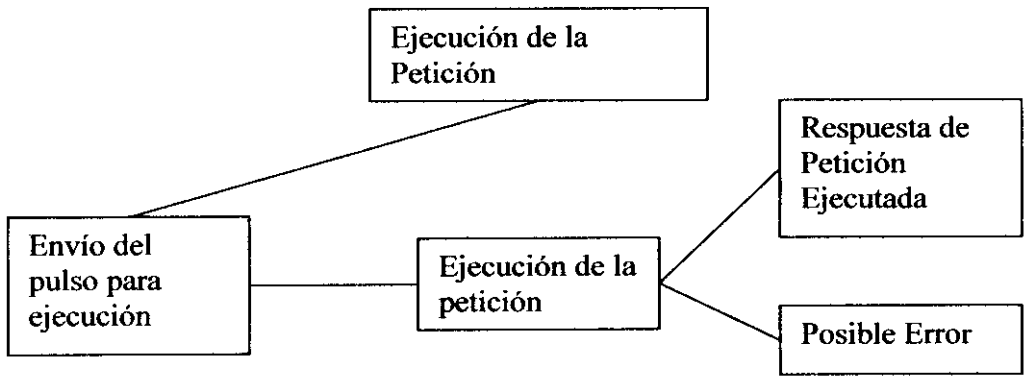
Diag8. Diseño Arquitectónico – Selección de Petición (Fuente: Investigador)

3.2.1.1.2. Diseño Arquitectónico (Petición seleccionada)



Diag9. Diseño Arquitectónico – Petición Seleccionada (Fuente: Investigador)

3.2.1.1.3. Diseño Arquitectónico (Ejecución de la petición)

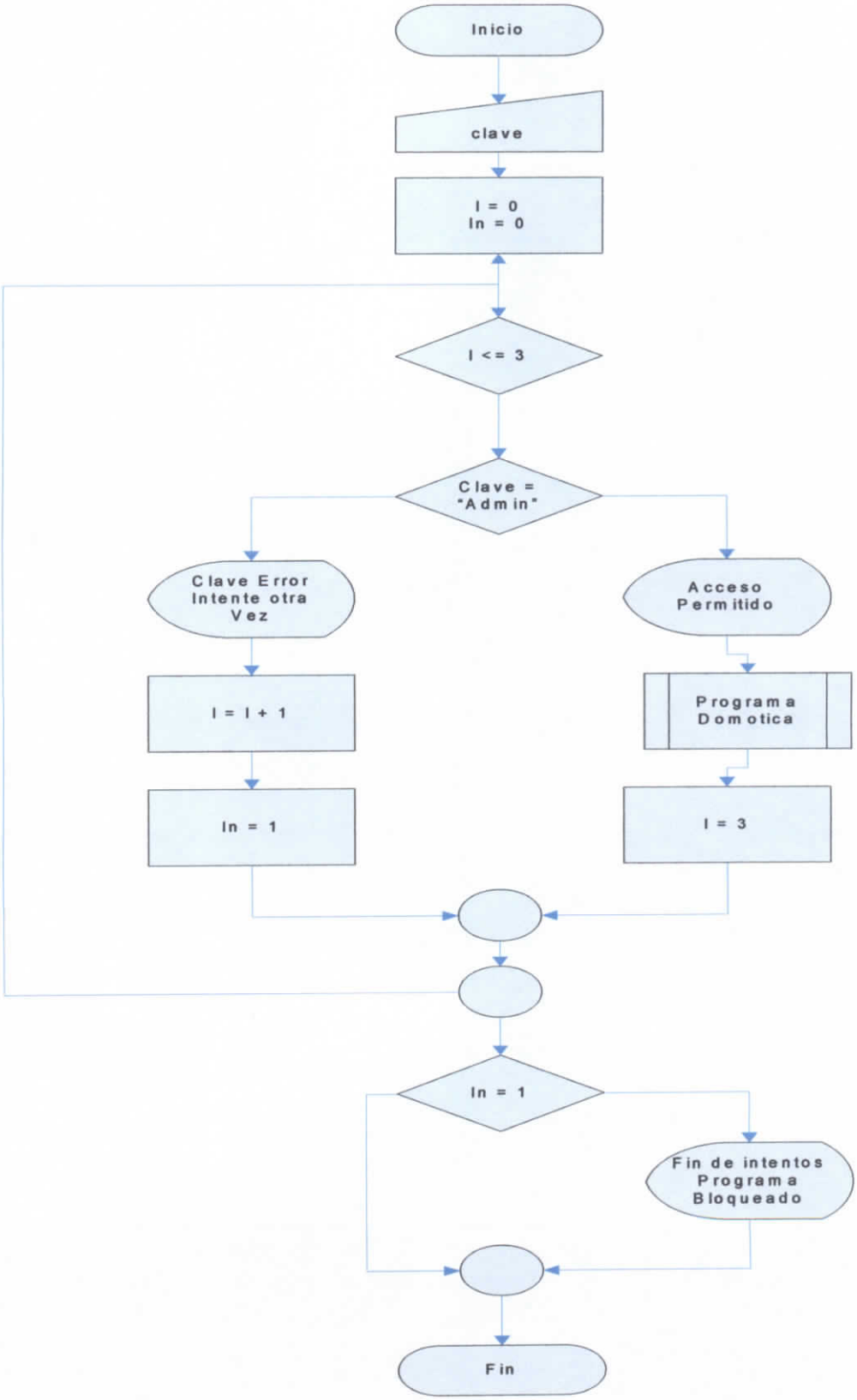


Diag10. Diseño Arquitectónico – Ejecucion de la Petición (Fuente: Investigador)

3.2.2. Diseño Procedimental

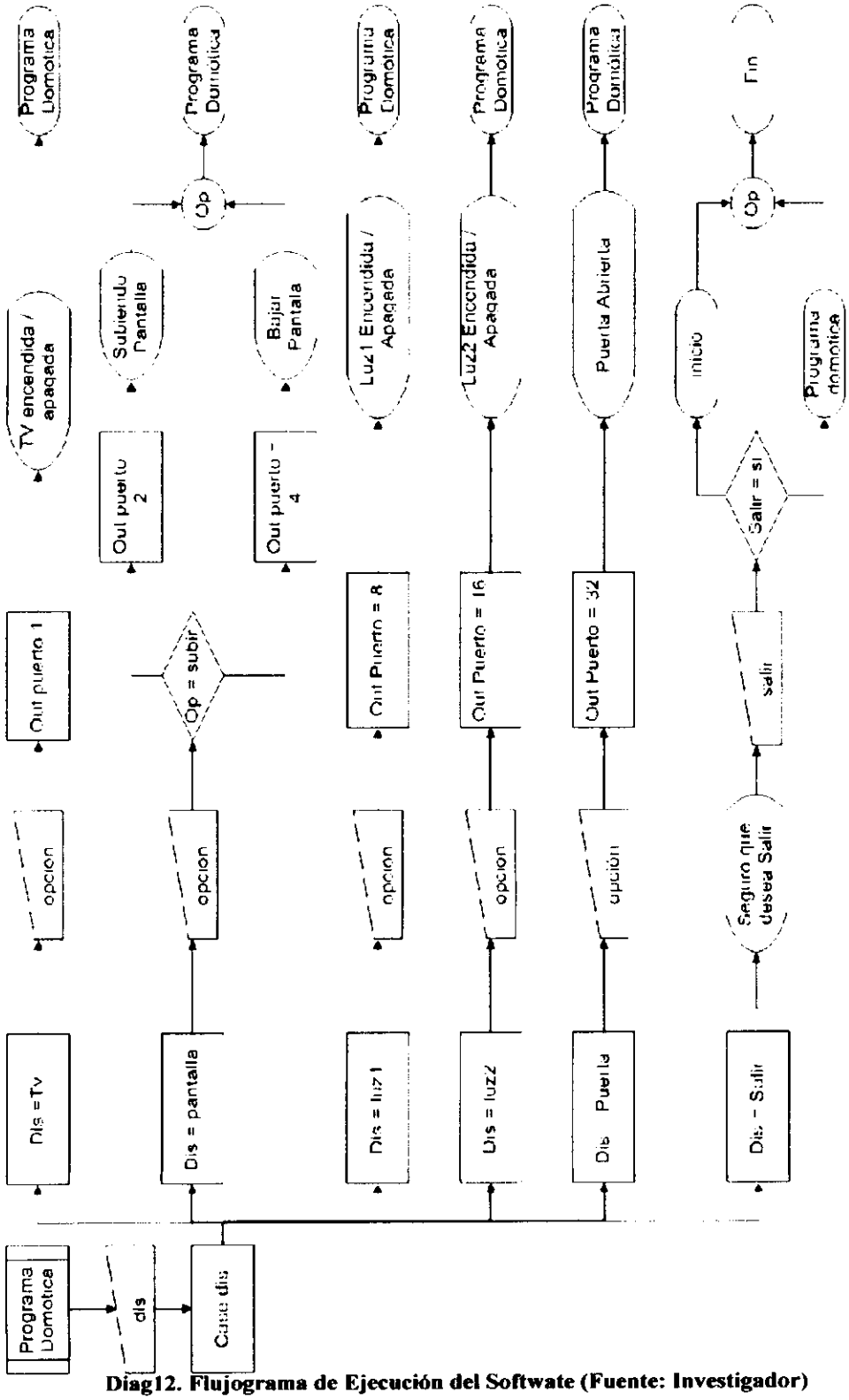
3.2.2.1. Diagramas de flujo

3.2.2.1.1. Flujograma de Acceso al Sistema



Diag11. Flujograma de Acceso al Sistema (Fuente: Investigador)

3.2.2.1.2. Flujograma de Ejecución del Software



Diag12. Flujograma de Ejecución del Softwate (Fuente: Investigador)

3.2.3. Diseño De Interfaz

El sistema contará con una interfaz grafica con la que el usuario se podrá familiarizar fácilmente, pues esta será diseñada de una forma que las tareas que realiza se puedan intuir por la persona que maneja el sistema.

También todas las peticiones para control de los dispositivos electrónicos serán colocadas en botones los mismos que desplegaran el submenú de este dispositivo preguntando la acción a realizarse.

El software contara también con menús en la parte superior para de esta manera tener una forma rápida y organizada para acceso a las tareas del sistema, siendo estas administrativas o de usuario.

El sistema a demás nos ofrece toda la ayuda necesaria para que el usuario este informado de la tarea a realizarse, la que está en ejecución y la que se ha ejecutado, esto permitirá informarnos por medio de mensajes desplegables, barras de progreso y alertas del sistema.

Contara también con información de errores, si el caso amerita, para de esta forma permitir al usuario buscar una solución adecuada a dicho error.

El software también tendrá un archivo de ayuda el cual incluirá toda la información necesaria para el aprendizaje del manejo del sistema, además explicado las funciones de este sistema para su mejor manejo.

3.2.3.2. Descripción de interfaz de ingreso de los usuarios

En esta pantalla los usuarios pueden ingresar su nombre y contraseña para identificarse y poder ingresar a las opciones que de acuerdo a sus privilegios, es decir si el administrador, o el usuario. El usuario administrador tiene total control de la aplicación. El usuario solo podrá Tener manejo de los dispositivos.



Fig26. Ingreso de Usuarios (Fuente: Investigador)

3.2.3.3. Descripción de interfaz de manejo de dispositivos

Esta es la pantalla básica Aparecen todos los dispositivos que se puede controlar a través del sistema.

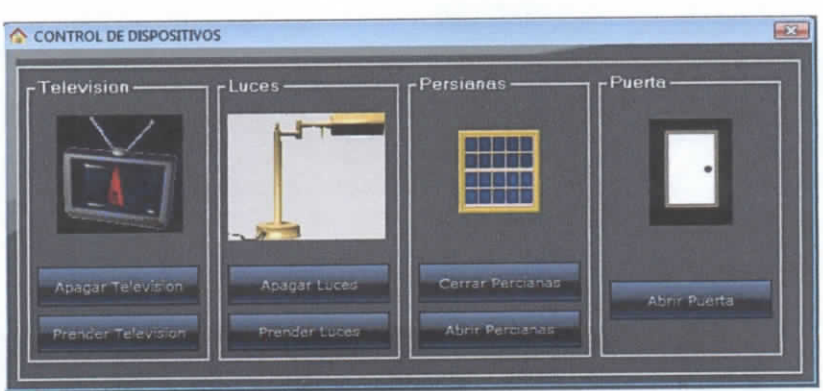


Fig27. Control de Dispositivos (Fuente: Investigador)

3.2.3.4. Descripción de interfaz de manejo de cada uno de los dispositivos

Cada vez que el usuario seleccione uno de los dispositivos en la pantalla anterior este mostrara ejecutara la acción de cada uno de los botones mostrando una pantalla para confirmación de acción del dispositivo (caso de Puerta, Luces, y Televisión) y una pantalla mostrando el progreso de apertura o cierre (caso de Persianas).

Fig28. Acción del Dispositivo Ejecutada



Fig29. Acción del Dispositivo Ejecutada (Fuente: Investigador)

3.2.3.5. Descripción de interfaz de mensajes

Las ventanas de mensajes serán mostradas en diferentes procesos a lo largo de la aplicación y servirán para verificar que información se envía o anunciaran algún cambio.

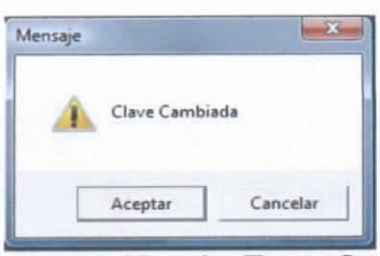


Fig30. Ventana de Mensajes (Fuente: Investigador)

3.2.3.6. Descripción de interfaz Principal del sistema

Esta será la ventana principal al abrir el sistema donde podemos encontrar la barra de menús y en su lado derecho nos mostrara las tareas programadas para cada dispositivo.

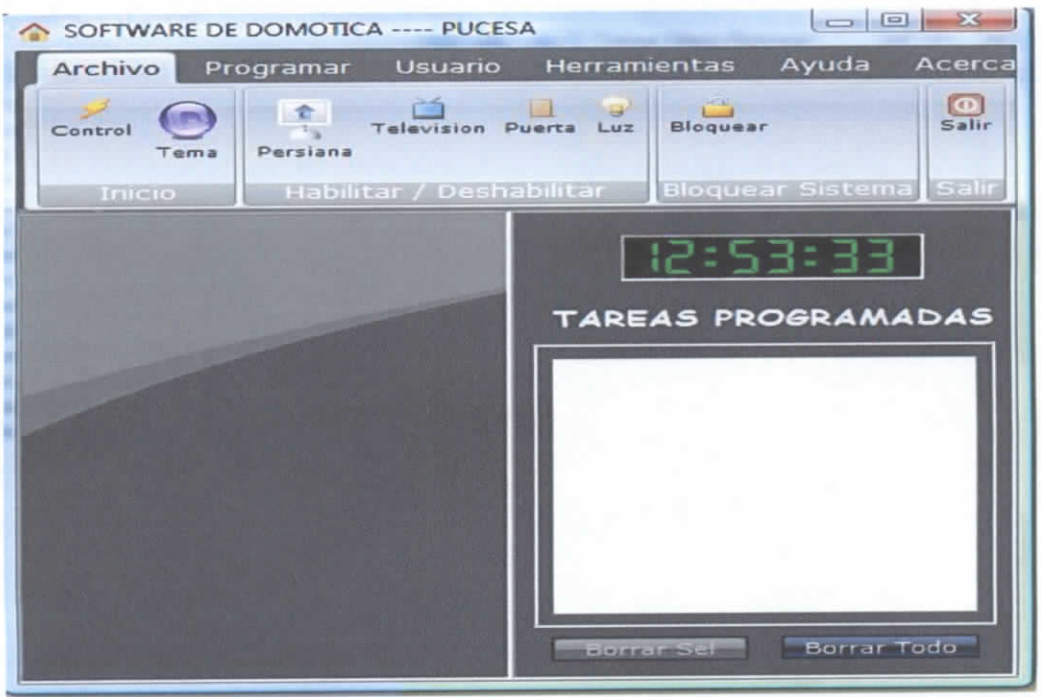
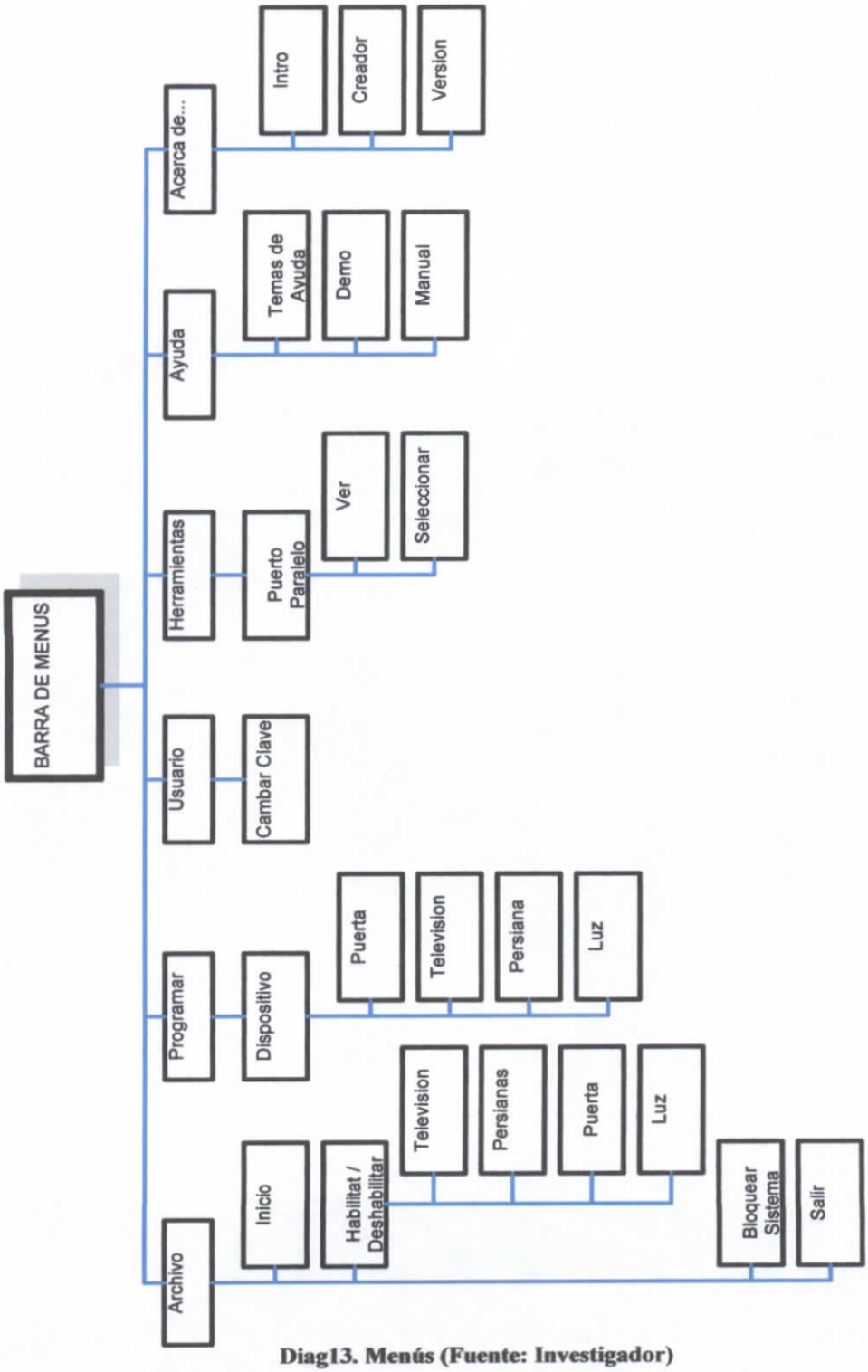


Fig31.Ventana Principal (Fuente: Investigador)

3.2.4. Diagrama de Menú



Diag13. Menús (Fuente: Investigador)

3.3. Codificación del Sistema

3.3.1. Código para Deshabilitar botón de cerrar en los formularios

```
Option Explicit

Private Declare Function GetSystemMenu Lib "user32" (ByVal hwnd As Long, ByVal bRevert As Long) As Long

Private Declare Function RemoveMenu Lib "user32" (ByVal hMenu As Long, ByVal nPosition As Long, ByVal wFlags As Long) As Long

Private Const MF_BYPOSITION = &H400&

Private Const MF_REMOVE = &H1000&


Public Sub RemoveCancelMenuItem(frm As Form)

Dim hSysMenu As Long

hSysMenu = GetSystemMenu(frm.hwnd, 0)

Call RemoveMenu(hSysMenu, 6, MF_BYPOSITION Or MF_REMOVE)

Call RemoveMenu(hSysMenu, 5, MF_BYPOSITION Or MF_REMOVE)

End Sub
```

3.3.2. Modulo para poder abrir documentos de extensión .chm

```
Public Enum HH_COMMAND

HH_DISPLAY_TOPIC = &H0

HH_HELP_FINDER = &H0

HH_DISPLAY_TOC = &H1

HH_DISPLAY_INDEX = &H2

HH_DISPLAY_SEARCH = &H3

HH_SET_WIN_TYPE = &H4

HH_GET_WIN_TYPE = &H5

HH_GET_WIN_HANDLE = &H6

HH_GET_INFO_TYPES = &H7

HH_SET_INFO_TYPES = &H8
```

```
HH_SYNC = &H9
HH_ADD_NAV_UI = &HA
HH_ADD_BUTTON = &HB
HH_GETBROWSER_APP = &HC
HH_KEYWORD_LOOKUP = &HD
HH_DISPLAY_TEXT_POPUP = &HE
HH_HELP_CONTEXT = &HF
HH_TP_HELP_CONTEXTMENU
HH_TP_HELP_WM_HELP = &H11
HH_CLOSE_ALL = &H12
HH_ALINK_LOOKUP = &H13
End Enum

Public Declare Function HtmlHelp Lib "hhctrl.ocx" Alias "HtmlHelpA" _
(ByVal hwndCaller As Long, ByVal pszFile As Stirling, _
ByVal uCommand As HH_COMMAND, ByVal dwData As Long) As Long
```

3.3.3. Modulo de Funciones

```
Public Declare Sub Sleep Lib "kernel32" (ByVal dwMilliseconds As Long)
Global indexboton, dato As Integer
Global disp As Stirling
Global usuarios As tclaves
Global tmpuser As tclaves
Global tdishora
Global Gdatos As f02
Public Sub skinboton(ByVal sboton As Integer)
For Each xBtn In MDIForm1
        If TypeOf xBtn Is ucBtnSkin Then
                xBtn.Skin = sboton
```

```
        End If
Next
For i = 1 To Forms.Count - 1
        For Each xBtn In Forms(i).Controls
                If TypeOf xBtn Is ucBtnSkin Then
                        xBtn.Skin = sboton
                End If
        Next
Next
End Sub
```

3.3.4. Modulo para el manejo del puerto paralelo

```
Global dpuerto As Stirling
Global valpuerto As Integer
Public Declare Function Inp Lib "inpout32.dll" _
Alias "Inp32" (ByVal PortAddress As Integer) As Integer
Public Declare Sub Out Lib "inpout32.dll" _
Alias "Out32" (ByVal PortAddress As Integer, ByVal Value As Integer)
```

3.3.5. Instrucciones Para el control del Puerto paralelo

```
        Out dpuerto, 1
        Out dpuerto, 0
```

3.3.6. Instrucciones para el manejo de archivos randómicos

```
Open App.Path & "\file01.dat" For Random As #1 Len = Len(usuarios)
Seek #1, reg
Get #1, , tmpuser
Put #1, i, usuarios
```

3.3.7. Tipos De datos definidos

```
'tipo de datos para variables de control
Public Type controlvar
   val_per As Boolean
End Type

'tipo de datos para guardar dirección del puerto
Public Type dirpuerto
dirreccion As Stirling * 5
End Type

'tipo de dato para manejar el tiempo de las persianas
Public Type timepersiana
   time As Long
End Type


'tipo de datos para manejo de usuarios
Public Type tclaves
user As Stirling * 10
password As Stirling * 10
ind As Integer
End Type

'tipo de datos para manejo de los temas
Public Type f02
Gtheme As Integer
Gboton As Integer
End Type
```

3.3.8. Código del Formulario para Salvapantalla

```
Dim X(100), Y(100), pace(100), size(100) As Integer
Private Sub Form_Activate()
Randomize
For i = 1 To 100
        x1 = Int(screensaver.Width * Rnd)
        y1 = Int(screensaver.Height * Rnd)
        pace1 = Int(500 - (Int(Rnd * 499)))
        size1 = 25 * Rnd
        X(i) = x1
        Y(i) = y1
        pace(i) = pace1
        size(i) = size1
Next
End Sub
Private Sub Timer1_Timer()
For i = 1 To 100
        Circle (X(i), Y(i)), size(i), BackColor
        Y(i) = Y(i) + pace(i)
If Y(i) >= screensaver.Height Then Y(i) = 0: X(i) = Int(screensaver.Width * Rnd)
                Circle (X(i), Y(i)), size(i)
Next
End Sub
```

3.3.9. Modulo se Clase para la elaboración del reloj digital

```
Option Explicit
Private Type Coordinate
  X As Integer
  Y As Integer
End Type

Dim BasePoint As Coordinate
Dim SegWidth As Integer
Dim SegHeight As Integer
Dim p As PictureBox

Property Let BackColor(Color As Long)
  p.BackColor = Color
End Property
Private Sub DrawNumber(Number As Integer)
  Select Case Number
  Case 0
    DrawSegment (1)
    DrawSegment (2)
    DrawSegment (3)
    DrawSegment (4)
    DrawSegment (5)
    DrawSegment (6)
  Case 1
    DrawSegment (2)
    DrawSegment (3)
  Case 2
    DrawSegment (1)
```

DrawSegment (2)

DrawSegment (7)

DrawSegment (5)

DrawSegment (4)

Case 3

DrawSegment (1)

DrawSegment (2)

DrawSegment (7)

DrawSegment (3)

DrawSegment (4)

Case 4

DrawSegment (2)

DrawSegment (3)

DrawSegment (7)

DrawSegment (6)

Case 5

DrawSegment (1)

DrawSegment (6)

DrawSegment (7)

DrawSegment (3)

DrawSegment (4)

Case 6

DrawSegment (1)

DrawSegment (6)

DrawSegment (7)

DrawSegment (3)

DrawSegment (4)

DrawSegment (5)

```
Case 7
    DrawSegment (1)
    DrawSegment (2)
    DrawSegment (3)
  Case 8
    DrawSegment (1)
    DrawSegment (2)
    DrawSegment (3)
    DrawSegment (4)
    DrawSegment (5)
    DrawSegment (6)
    DrawSegment (7)
  Case 9
    DrawSegment (1)
    DrawSegment (2)
    DrawSegment (3)
    DrawSegment (4)
    DrawSegment (6)
    DrawSegment (7)
  End Select
End Sub

Private Sub DrawSegment(SegNum As Integer)
  Select Case SegNum
  Case 1
    p.Line (BasePoint.X + 1, BasePoint.Y)-(BasePoint.X + SegWidth - 1,
BasePoint.Y)
    p.Line (BasePoint.X + 2, BasePoint.Y + 1)-(BasePoint.X + SegWidth - 2,
BasePoint.Y + 1)
```

```
    p.Line (BasePoint.X + 3, BasePoint.Y + 2)-(BasePoint.X + SegWidth - 3,
BasePoint.Y + 2)

  Case 2

    p.Line (BasePoint.X + SegWidth - 1, BasePoint.Y + 1)-(BasePoint.X +
SegWidth - 1, BasePoint.Y + (SegHeight \ 2) - 1)

    p.Line (BasePoint.X + SegWidth - 2, BasePoint.Y + 2)-(BasePoint.X +
SegWidth - 2, BasePoint.Y + (SegHeight \ 2))

    p.Line (BasePoint.X + SegWidth - 3, BasePoint.Y + 3)-(BasePoint.X +
SegWidth - 3, BasePoint.Y + (SegHeight \ 2) - 1)

  Case 3

    p.Line (BasePoint.X + SegWidth - 1, BasePoint.Y + (SegHeight \ 2) + 2)-
(BasePoint.X + SegWidth - 1, BasePoint.Y + SegHeight)

    p.Line (BasePoint.X + SegWidth - 2, BasePoint.Y + (SegHeight \ 2) + 1)-
(BasePoint.X + SegWidth - 2, BasePoint.Y + SegHeight - 1)

    p.Line (BasePoint.X + SegWidth - 3, BasePoint.Y + (SegHeight \ 2) + 2)-
(BasePoint.X + SegWidth - 3, BasePoint.Y + SegHeight - 2)

  Case 4

    p.Line (BasePoint.X + 3, BasePoint.Y + SegHeight - 2)-(BasePoint.X +
SegWidth - 3, BasePoint.Y + SegHeight - 2)

    p.Line (BasePoint.X + 2, BasePoint.Y + SegHeight - 1)-(BasePoint.X +
SegWidth - 2, BasePoint.Y + SegHeight - 1)

    p.Line (BasePoint.X + 1, BasePoint.Y + SegHeight)-(BasePoint.X + SegWidth -
1, BasePoint.Y + SegHeight)

  Case 5

    p.Line (BasePoint.X, BasePoint.Y + (SegHeight \ 2) + 2)-(BasePoint.X,
BasePoint.Y + SegHeight)

    p.Line (BasePoint.X + 1, BasePoint.Y + (SegHeight \ 2) + 1)-(BasePoint.X + 1,
BasePoint.Y + SegHeight - 1)

    p.Line (BasePoint.X + 2, BasePoint.Y + (SegHeight \ 2) + 2)-(BasePoint.X + 2,
BasePoint.Y + SegHeight - 2)

  Case 6

    p.Line (BasePoint.X, BasePoint.Y + 1)-(BasePoint.X, BasePoint.Y + (SegHeight
\ 2) - 1)

    p.Line (BasePoint.X + 1, BasePoint.Y + 2)-(BasePoint.X + 1, BasePoint.Y +
(SegHeight \ 2))
```

```
    p.Line (BasePoint.X + 2, BasePoint.Y + 3)-(BasePoint.X + 2, BasePoint.Y +
(SegHeight \ 2) - 1)

  Case 7

    p.Line (BasePoint.X + 3, BasePoint.Y + (SegHeight \ 2) - 1)-(BasePoint.X +
SegWidth - 3, BasePoint.Y + (SegHeight \ 2) - 1)

    p.Line (BasePoint.X + 2, BasePoint.Y + (SegHeight \ 2))-(BasePoint.X +
SegWidth - 2, BasePoint.Y + (SegHeight \ 2))

    p.Line (BasePoint.X + 3, BasePoint.Y + (SegHeight \ 2) + 1)-(BasePoint.X +
SegWidth - 3, BasePoint.Y + (SegHeight \ 2) + 1)

  End Select

End Sub

Public Property Let Caption(ByVal Value As Stirling)

Dim OrigX As Integer

  OrigX = BasePoint.X

  p.Cls

  While Value <> ""

    If Left$(Value, 1) <> ":" Then

      DrawNumber (Val(Left$(Value, 1)))

      BasePoint.X = BasePoint.X + SegWidth + 3

    Else

      p.Line (BasePoint.X + (SegWidth \ 2) - 4, BasePoint.Y + (SegHeight \ 2) - 6)-
(BasePoint.X + (SegWidth \ 2), BasePoint.Y + (SegHeight \ 2) - 3), , BF

      p.Line (BasePoint.X + (SegWidth \ 2) - 4, BasePoint.Y + (SegHeight \ 2) + 4)-
(BasePoint.X + (SegWidth \ 2), BasePoint.Y + (SegHeight \ 2) + 7), , BF

      BasePoint.X = BasePoint.X + SegWidth

    End If

    Value = Right$(Value, Len(Value) - 1)

  Wend

  BasePoint.X = OrigX

End Property

Property Let ForeColor(Color As Long)
```

```
  p.ForeColor = Color
End Property
Public Sub NewLCD(PBox As PictureBox)
  Set p = PBox
  p.ScaleMode = 3            ' pixel
  p.AutoRedraw = True
  BasePoint.X = 2
  BasePoint.Y = 2
  SegHeight = p.ScaleHeight - 6
  SegWidth = (SegHeight \ 2) + 2
End Sub
```

3.4. Diseño del circuito electrónico

3.4.1. Metodología de Elaboración

3.4.1.1. Termotransferencia

Termotransferencia quiere decir transferencia por calor, mediante un sistema de impresión, pues la transferencia se realiza usando como medio el calor a lo que se quiere imprimir, para esto se usa un papel térmico especial que lleva una película en una de sus caras de un producto químico que en presencia de calor se tiñe. No existe contacto entre las cabezas impresoras y el papel, por lo que la impresión es razonablemente rápida y silenciosa.

De esta forma al tener el circuito impreso en este tipo de papel lo vamos a aplicar a la placa por medio de calos para que lo que está impreso se quede marcado en la baquelita donde vamos a elaborar el circuito. Esto permite tener circuitos electrónicos elaborados de una forma más profesional y a demás se puede tener un mejor orden en las pistas del circuito.

3.4.2. Diseño de las pistas del circuito

El Siguiente grafico detalla cómo estarán impresas las pistas del circuito electrónico a realizar para el control de los dispositivos eléctricos.

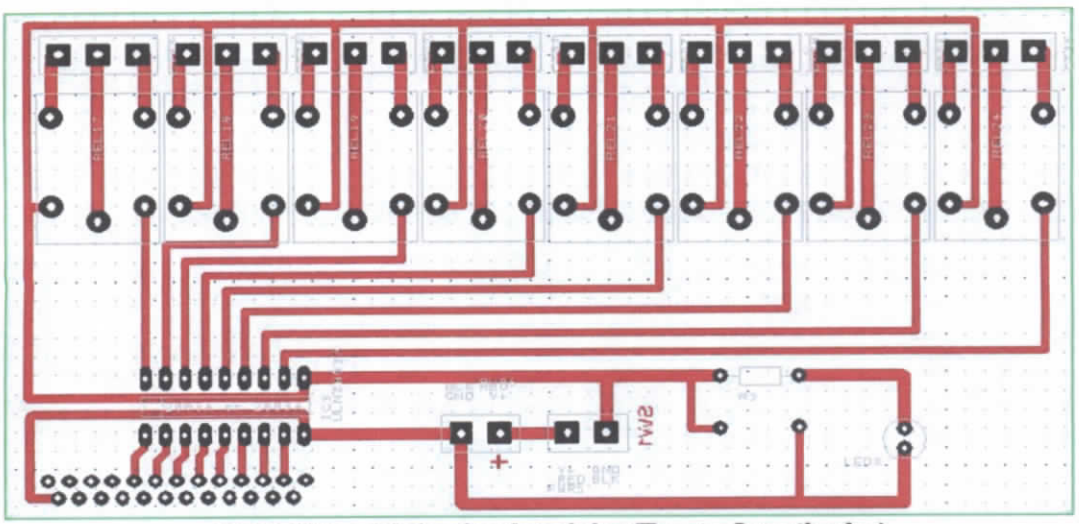


Fig32. Pistas del circuito electrónico (Fuente: Investigador)

3.4.3. Modelo 3d del Circuito

Las Siguientes imágenes expresan como irán repartidos los componentes electrónicos en el circuito.

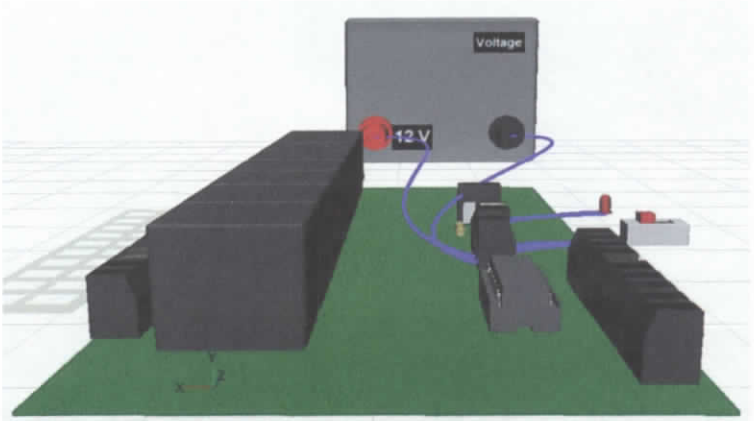


Fig33. Modelo 3d del circuito electrónico 1 (Fuente: Investigador)

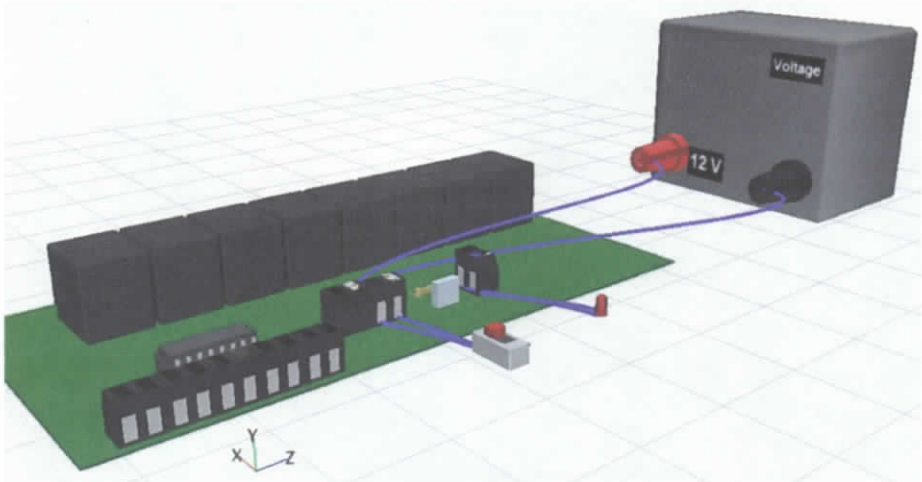


Fig34. Modelo 3d del circuito electrónico 2 (Fuente: Investigador)

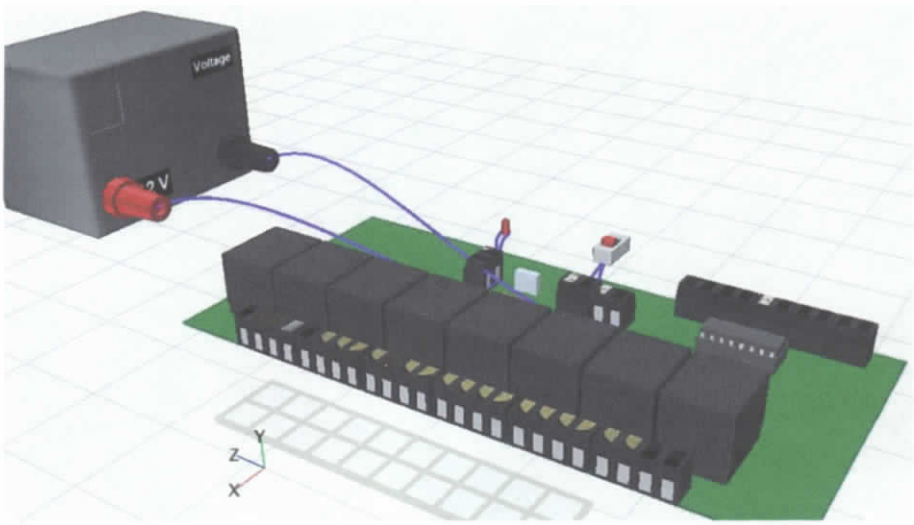


Fig35. Modelo 3d del circuito electrónico 3 (Fuente: Investigador)

3.4.4. Componentes y Materiales a usar

Relé genérico mecánico de 12v: El relé o relevador es un dispositivo electromecánico, que funciona como un interruptor controlado por un circuito eléctrico en el que, por medio de un electroimán, se acciona un juego de uno o varios contactos que permiten abrir o cerrar otros circuitos eléctricos independientes. Ya que el relé es capaz de controlar un circuito de salida de mayor potencia que el de entrada, puede considerarse en forma de amplificador eléctrico.

Circuito integrado ULN2803AG: En el interior de este circuito integrado podemos encontrar ocho esquemas compuestos de resistencias y transistores, uno para cada uno de los canales.

Las resistencias nos permiten una oposición en la corriente, mientras que los transistores nos ayudaran a la hora de amplificar, oscilar, conmutar o rectificar una señal.

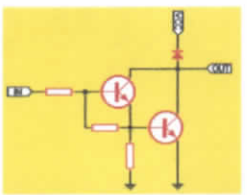


Fig36. Vista de un canal del circuito integrado ULN2803AG (Fuente: Investigador)

Condensador 130 pF: Condensador o capacitor es un componente que almacena energía eléctrica, es un componente pasivo. Está formado por un par de superficies conductoras en situación de influencia total, generalmente en forma de tablas, esferas o láminas, separados por un material dieléctrico o por el vacío, que, sometidos a una diferencia de potencial adquieren una determinada carga eléctrica, positiva en una de las placas y negativa en la otra (siendo nula la carga total almacenada).

Resistencia 220 Ohm: Resistencia eléctrica, *R*, presenta oposición a la corriente eléctrica encontrada dentro del circuito. Su valor viene dado en ohmios, se designa con la letra griega omega mayúscula (Ω), y se mide con el Ohmímetro.

Borneras: Estas nos permiten conectar las punteras a la placa del circuito.

Sócalo: El zócalo es una pieza de plástico que funciona como intermediario entre la placa del circuito y el circuito integrado. Posee en su superficie plana superior una matriz de pequeños agujeros donde encajan los pines del integrado. Mientras que por su parte inferior cuenta con los pines que irán soldados a la placa del circuito

Diodo Led: Conocido como LED que emite luz cuando se polariza de forma directa. El color, depende del material semiconductor empleado en la construcción del diodo y puede ser desde ultravioleta, siendo esta la visible, hasta infrarrojo.

Switch: es un dispositivo para cambiar o cortar el curso de la corriente de un circuito. El modelo prototípico es un dispositivo mecánico que puede ser desconectado de un curso y unido al otro.

Baquelita: Es una sustancia plástica totalmente sintética no conductora pro en su parte inferior está recubierta de cobre ya que este es un material conductor. Sobre esta se ensamblara el circuito electrónico. A demás permitirá la creación de las pistas del circuito por su parte inferior.

Adaptador Universal de 12v 1000mA: Este nos sirve para dar corriente al circuito. Va enchufado a la luz eléctrica y por el otro extremo al circuito, de esta forma alimentara de corriente a la placa, permitiendo de esta forma la activación o el funcionamiento de los componentes.

Punteras Eléctricas: Estas nos ayudan a conectar los cables a las borneras. Se enlaza el cable a este y de allí a la bornera, evitando de esta forma los posibles cortos circuitos.

Conector LPT Hembra: Es el conector del puerto paralelo. Este ira conectado al circuito para poder enlazar la placa con el computador

Conector LPT Macho: es el conector del cable del puerto paralelo, este nos ayudara en la creación del cable que trasmitirá los datos desde el puerto paralelo al circuito.

Cable UTP de 6 Pares: Con este elaboraremos el cable para la transmisión de los pines desde el puerto paralelo al circuito.

Cloruro Férrico: Con este acido corroeremos la baquelita en la elaboración de las pistas de la placa.

Estaño: Nos sirve para soldar los componentes electrónicos en la baquelita.

Cautín: Es la herramienta que permite desleír el estaño, calentarlo para que tome su forma líquida y luego el estaño al enfriarse volverá a su forma solida.

Taladro y brocas: Permitirán la creación de los agujeros en la baquelita para en ellos insertar los componentes electrónicos.

Barniz: Una vez creada la placa del circuito, le echaremos barniz para evitar su deterioro.

Lija de agua: La lija nos permite limpiar la baquelita en su parte de cobre para evitar cualquier impureza en esta y que las pistas queden correctamente elaboradas.

Cierra: Con esta cortaremos la baquelita hasta darle la forma adecuada para nuestro circuito.

3.5. Elaboración del Circuito electrónico

Para elaborar el circuito electrónico que nos ayudara a controlar los dispositivos eléctricos es necesario seguir algunos pasos, los cuales detallaremos a continuación:

Tener todos los componentes electrónicos y materiales listos para la elaboración del circuito.



Fig37. Componentes Electrónicos para el circuito (Fuente: Investigador)

Imprimir el diseño de las pistas para el circuito en el papel químico. Para posteriormente hacer la termotransferencia a la baquelita.

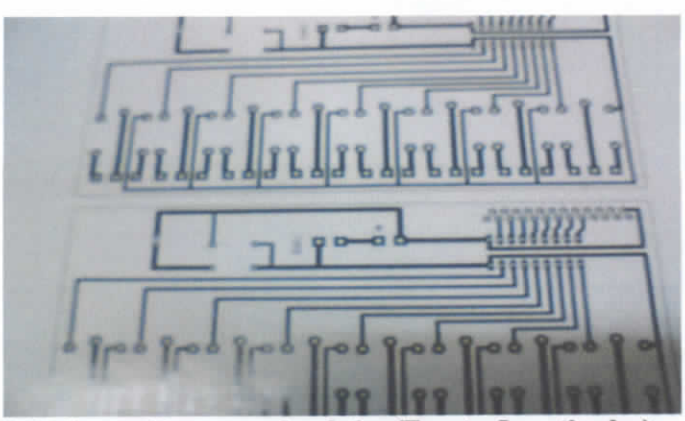


Fig38. Impresión en papel químico (Fuente: Investigador)

Limpiar bien la baquelita con una lija de agua delgada, hasta que esta quede sin ninguna impureza.



Fig39. Baquelita (Fuente: Investigador)

Adherir las pistas desde el papel químico hacia la baquelita, este proceso se llama termotransferencia, el cual consiste en por medio de calor transferir las impresiones del papel al cobre de la placa. Esto lo podemos hacer con una plancha domestica.



Fig40. Termotransferencia (Fuente: Investigador)

Preparar un recipiente plástico con el cloruro férrico y agua. Se recomienda cuando se trabaje con esta solución se mantenga fuera del alcanse de ojos y boca ya que es toxica.



Fig41. Preparación del cloruro férrico (Fuente: Investigador)

Se inserta la baquelita ya con las pistas impresas en el cloruro férrico, para de esta forma corroer el cobre que es innecesario en nuestro circuito, se debe mantener en movimiento para un mejor resultado.

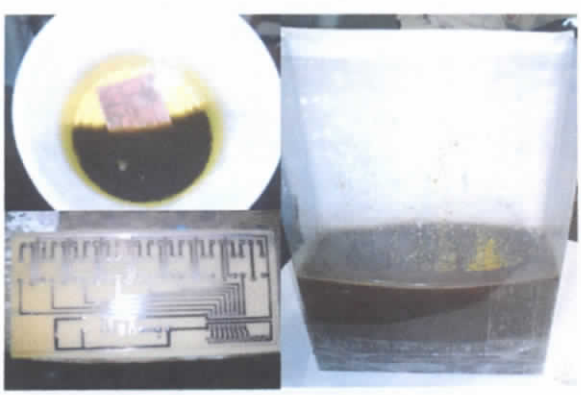


Fig42. Corrosión del cobre mediante cloruro férrico (Fuente: Investigador)

Una vez creadas las pistas procedemos a perforar para crear los agujeros donde se insertaran los componentes electrónicos.

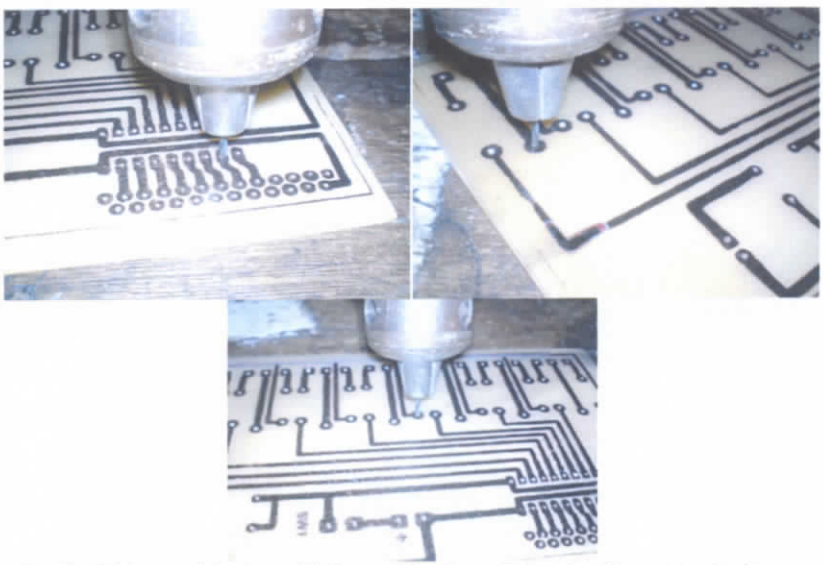


Fig43. Elaboración de orificios en la placa (Fuente: Investigador)

Insertaremos los componentes electrónicos tal y como se realizaron en el diagrama del circuito.



Fig44. Colocación de los componentes en la placa (Fuente: Investigador)

Se irán soldando cada uno de los componentes electrónicos a la baquelita teniendo cuidado en levantas o cortar alguna pista de nuestro circuito.

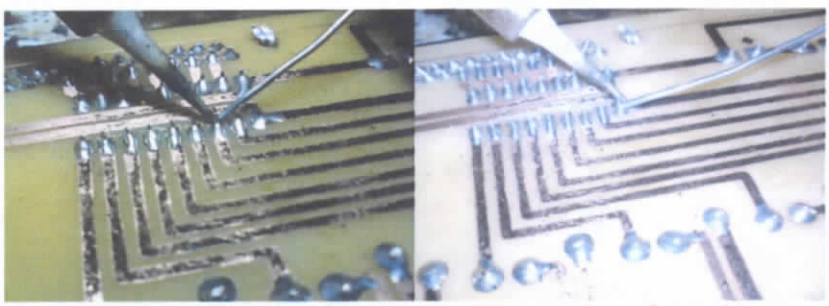


Fig45. Soldar compontes a la placa (Fuente: Investigador)

La placa lista quedara de la siguiente forma. Lo que nos quedaría es revestirla de barniz por la parte inferior para evitar que se desgasten las pistas o pueda romperse su continuidad.

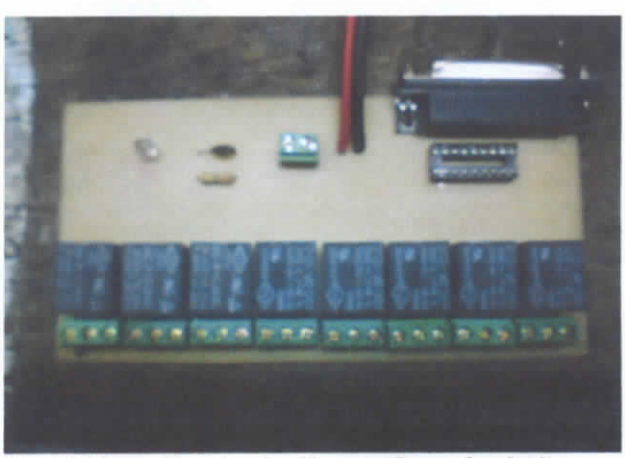


Fig46. Placa Terminada (Fuente: Investigador)

3.6. Implementación

Para la implementación de este sistema iremos revisando ciertos pasos básicos dentro de la instalación de los dispositivos eléctricos para tener clara la forma en cómo se deberá conectar cada uno de ellos explicando además ciertas cosas que debemos tener en cuenta a la hora de su instalación.

3.6.1. Instalación de canaletas, cables y rieles

Para la instalación de canaletas cables y rieles es necesario que tengamos claro por dónde pasaran nuestros cables manteniendo siempre en cuenta que debe ser una instalación que guarde seguridades además que se presente de una forma estética.

Las canaletas deberán ir figadas la las paredes o pisos donde estas serán colocadas, mediante tacos Fisher y tornillos, para de esta forma evitar que se despejen y deterioren los cables en su interior.



Fig47. Fijación del taco Fischer a la pared para colocar canaleta (Fuente: Investigador)

3.6.2. Instalación de puerta eléctrica.

Para instalar la puerta eléctrica debemos tener en cuenta dos pasos importantes estos son: la instalación del dispositivo electrónico que nos permitirá abrir y cerrar la puerta por medio de los controles remotos de esta y además también es importante la instalación de un transformador a 12v puesto que la chapa eléctrica funciona bajo este voltaje.



Fig48. Dispositivo de Control remoto para la chapa eléctrica (Fuente: Investigador)



Fig49. Transformador de corriente de 110v a 12v (Fuente: Investigador)

3.6.3. Instalación de luces de los cuartos

Para instalar nuestro sistema en las luces de los cuartos solo es necesario hacer una conexión dentro del cajetín de swicth de la luz en la cual queremos que sea controlada por el sistema, en esta instalación lo que hacemos es extender dos cables más como si quisiéramos añadirle otro swicth al foco el que queremos encender o apagar.

3.6.4. Instalación de control para televisión

Instalar el sistema para apagar y encender el televisor no presentara mucha dificultad ya que solo será necesaria al extender el botón de encendido y apagado del control remoto de nuestro televisor hacia el circuito de control del sistema. Para esto soldaremos los dos pines que sirven para perder y apagar del circuito electrónico del control remoto a unos cables que nos permitan luego ser conectados en alguno de los relés de nuestro circuito electrónico.

3.6.5. Instalación de Persianas

Las persianas irán instaladas en una de las ventanas designadas, debemos tener en cuenta que el espacio sea el adecuado ya que no se podrá usar el tamaño completo de pared a pared, siendo uno de los impedimentos el ancho del motor el cual dará movimiento a nuestra persiana.

Para la colocación de la persiana debemos tomar en cuenta los siguientes aspectos: el eje debe encajar correctamente al motor, el eje debe estar bien anclado a techo o pared donde será fijado para evitar así cualquier inconveniente, el motor necesitara conectarse a una fuente de energía de 12v y 3A para el correcto funcionamiento del mismo.

Para el mantenimiento de la misma se diseño un modelo donde el desprendimiento de la cortina sea de una manera fácil, para así poder labrarla o cambiarla si se diera el caso.

El mecanismo automático de la persiana funcionara conjuntamente con el sistema de control desde el computador para los dispositivos (DomoSoft).

3.7. Mantenimiento y Pruebas

3.7.1. Pruebas del Sistema

3.7.1.1. Pruebas de caja blanca

Las pruebas de caja blanca se realizaron al sistema tomando en cuenta los siguientes factores: Cobertura de sentencias y cobertura de condiciones. Teniendo los siguientes resultados:

Cobertura de sentencias

En esta parte se verifico que un 80% de las líneas de código sean ejecutadas por lo menos una vez dentro del software, teniendo resultados satisfactorios, y llegando así a comprobar que no tenemos líneas de código que no se usen o redundantes.

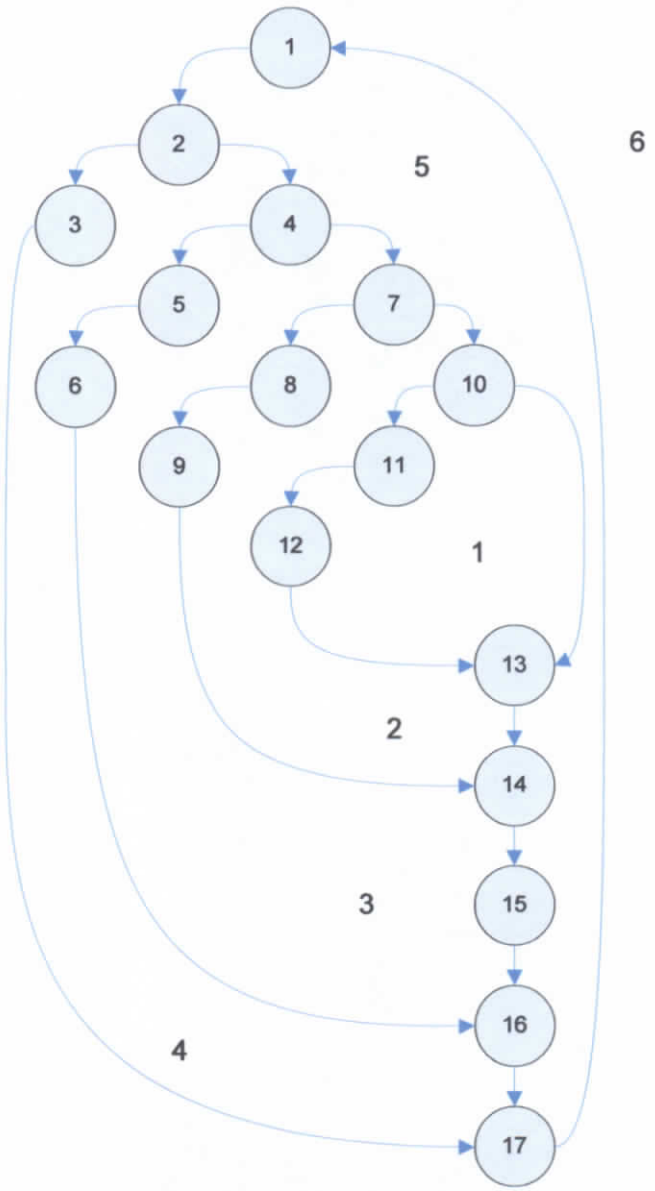
Para la realización de esta prueba se uso el compilador paso a paso de visual basic 6.0, el cual nos permitió ir de instrucción en instrucción verificando su ejecución.

Cobertura de condiciones

Esta prueba será realizada a la parte del código del software donde se maneja el puerto paralelo ya que esta es la más importante para el correcto funcionamiento del sistema para así poder demostrar su funcionamiento.

Para esto se toma como referencia la sección de código de uno de los botones de ejecución de acción, y se realiza el proceso de caja blanca a la sección del puerto paralelo. Quedando así comprobado que el software es eficiente a la hora de controlar los dispositivos eléctricos.

Se realizara esta prueba a uno solo de los botones de ejecución ya que el código que se ejecuta es similar para todos los botones que controlan los dispositivos eléctricos. A continuación se detallan los resultados.



Diag 14. Pruebas de Caja Blanca (Fuente: Investigador)

Nodos = 17

Aristas = 21

Regiones = 6

V(G) = Aristas – Nodos + 2

V(G) = 21 – 17 + 2 = 6

Regiones = V(G) => 6 = 6

Caminos

1: 1, 2, 3, 16, 17

2: 1, 2, 4, 5, 6, 15, 16, 17

3: 1, 2, 4, 7, 8, 9, 14, 15, 16, 17

4: 1, 2, 4, 7, 10, 11, 12, 13, 14, 15, 16, 17

5: 1, 2, 4, 7, 10, 13, 14, 15, 16, 17

6: 1, 17

Casos de Prueba

Variable = v1	Variable = v2
v1=0	v2=0
v1=1	v2=1
v1<>0	v2<>0
v1<>1	v2<>0

3.7.1.2. Pruebas de caja Negra

Las pruebas de caja negra fueron realizadas al software en diversos aspectos de los cuales podemos, decir que resultaron satisfactorias. Los casos en que el software fue evaluado se detallan a continuación.

En el ingreso del usuario y clave se evaluó probando con un usuario y clave correcta, un usuario existente y una clave errónea, un usuario erróneo y una clave existente, un usuario y una clave de otro usuario, y sin usuario ni clave. Los resultados fueron satisfactorios permitiendo el acceso solo en el caso de que el usuario y contraseña se pertenezcan y sean correctos.

Se evaluó el funcionamiento correcto de todos los botones del menú. Las pruebas dieron resultados correctos, los botones muestran la información correcta.

Por otro lado se probó el correcto funcionamiento de los botones que controlan los dispositivos, dando como resultado que todos los dispositivos son controlados por los mismos de una forma correcta.

A demás se evaluó la creación eliminación y modificación de usuarios y claves, teniendo casos de prueba individuales y con diferente acción de ingreso de datos, se concluyo que estos procesos funcionan correctamente.

Se puso a prueba además las extensiones permitidas para las cadenas de caracteres y que en los campos donde se llenan letras o números estén con el correcto formato para evitar errores en el momento de guardarlos en los archivos.

Fue evaluada la parte de programación de tareas para poder definir si las órdenes programadas son ejecutadas, en el tiempo estimado y programado.

CAPITULO IV

4.1. Conclusiones

- La implementación del sistema de aplicaciones Domóticas fue cambiada de destino. En un principio se lo iba a implantar en el laboratorio de electrología de la escuela de sistemas de la PUCESA, pero por motivos de seguridad de todos los implementos y usuarios del sistema, además de prestar una mejor función, su implementación se la hizo en la oficina de dirección de la escuela de sistemas de la PUCESA.

- La instalación de la persiana fue sustituida por la instalación de una pantalla para proyección, esto se dio con el fin de prestar un servicio más notorio para los profesores y estudiantes de la universidad.

- El software desarrollado para el control de dispositivos cumplió con los requerimientos y objetivos especificados en su planeación.

- La automatización de una casa u oficina presenta un gran cambio al automatizar los procesos produciendo una gran innovación a los mismos.

- Podemos concluir que el puerto paralelo del computador se puede usar para programar diferentes aplicaciones y controlar distintos dispositivos eléctricos o electrónicos.

- El termino Domótica en la actualidad todavía no es común en la mayoría de personas y este proyecto puede ayudarnos a acercarnos a esas nuevas técnicas y tecnologías.

- El proyecto ayudara a muchos estudiantes a incursionar en estas aplicaciones de control de dispositivos dando los primeros pasos para el desarrollo de otros sistemas de esta índole.

4.2. Recomendaciones

- Mantener un orden en la programación para que el código se pueda entender con mayor facilidad. Colocando comentarios a lo largo del código fuente para de esta forma poder recordar con facilidad para que se uso cierta sección de código. A demás tratar de usar módulos separados para cada uno de los procesos que se han de rehusar con frecuencia en la programación, añadiendo a estos las variables y funciones públicas o globales. También disminuir las líneas de código tanto como sea posible, creando funciones, procedimientos o métodos que se puedan rehusar.

- Mantener siempre claro las polaridades de cada dispositivo electrónico para evitar daños en el mismo. Además tener en cuenta el voltaje que entra al circuito y verificar si es el correcto ya que el exceso de este puede provocar daños en la placa y/o dispositivos electrónicos.

- Evitar colocar piezas o elementos conductores en la parte inferior de la placa, ya que estos pueden provocar un corto circuito.

- Alimentar el circuito con una fuente de corriente continua bien filtrada, para así evitar el mal funcionamiento y perdida de los elementos.

- Cuidar que las conexiones eléctricas estén elaboradas de una correcta forma para evitar corto circuitos.

- Exponer los dispositivos eléctricos al voltaje adecuado para evitar que se echen a perder, esto se da por que los dispositivos eléctricos pueden funcionar en diferentes rangos de voltaje.

Bibliografía

Consultada:

- Marco Antonio Tiznado Santana, *Enter Plus Visual Basic 6.0*, Bogotá-Colombia: McGraw-Hill Interamericana S.A., Marzo 2000

- Juan José Castañeda León, *Visual Basic 6.0 Como debe ser*, Perú: Megabyte, Abril 2002

- Ramón A Mata-Toledo y Pauline K. Cushman, *Introducción a la Programacion en Visual Basic, C, C++ y Java*, Mexico: McGrew-Hill, 2001

- Pablo Alcalde S. Miguel, *Principios Fundamentales de Electronica 4 Edicion*, España: Thomson-Paraninfo, 2003

- Prof. Ángel Zetina M. y Ing. Ángel Zetina C., *Electrónica Básica*, Mexico: Limusa-Noriega Editores, 2003

Citada:

http://www.info-ab.uclm.es/labelec/solar/electronica/elementos/Electricos/Motor-labels.gif *Septiembre 2008*

http://www.comunidadelectronicos.com/articulos/historia.htm *Septiembre 2008*

http://www.forosdeelectronica.com/tutoriales/historia.htm *Septiembre 2008*

http://www.monografias.com/trabajos39/circuito-basico/circuito-basico.shtml *Septiembre 2008*

http://www.american.edu.co/vs/webelectric/componentes1.html Octubre 2008

http://www.globu.net/pp/PP/historia.htm *Octubre 2008*

http://www.globu.net/pp/PP/descrip.htm *Noviembre 2008*

http://iie.fing.edu.uy/ense/asign/tap/gente04/obrar04/alvaroGalante_elec/paginas%20bajadas/paralelo.htm *Noviembre 2008*

http://es.wikipedia.org/wiki/BASIC *Noviembre 2008*

http://www.taringa.net/posts/ebooks-tutoriales/1359294/Puerto-paralelo-y-vb-6.html

Internet

http://es.wikipedia.org/wiki/Automatizaci%C3%B3n *Julio, 2008*

http://es.wikipedia.org/wiki/Dom%C3%B3tica *Julio, 2008*

http://es.wikipedia.org/wiki/Puerto_paralelo *Julio, 2008*

http://es.kioskea.net/contents/pc/serie.php3 *Julio, 2008*

http://cfievalladolid2.net/tecno/cyr_01/control/ *Julio, 2008*

http://www.todorobot.com.ar/proyectos/paralelo/paralelo.htm *Agosto, 2008*

http://www.recursosvisualbasic.com.ar/htm/menu-principal/mis_utilidades.htm *Agosto, 2008*

http://www.visual-basic.com.ar/links-es.htm *Agosto, 2008*

http://www.mundoprogramacion.com/ *Agosto, 2008*

http://www.abcdatos.com/programas/programacion/visualbasic.html *Agosto, 2008*

Glosario

API: Application Programming Interface. Es el conjunto de funciones y procedimientos

AUTOMATIZACION: Es el uso de sistemas o elementos computarizados para controlar maquinarias y/o procesos substituyendo a operadores humanos.

BIOS: El sistema Básico de entrada/salida Basic Input-Output System (BIOS) es un código de software que localiza y carga el sistema operativo

BUCLE: Sección de instrucciones de programación que se repetirán un cierto número de veces de acuerdo a una condición.

CMD: Ventana de Windows donde se puede escribir códigos de MS-DOS.

DFD: Diagrama de flujo de Datos.

DLL: Dynamic Linking Library (Bibliotecas de Enlace Dinámico), término con el que se refiere a los archivos con código ejecutable que se cargan bajo demanda del programa por parte del sistema operativo.

DOCUMENTOS DE EXTENCION .CHM: Documentos de ayuda dentro del sistema operativo Windows.

DOMOTICA: proviene de la unión de las palabras domus (que significa casa en latín) y tica (de automática, palabra en griego, 'que funciona por sí sola').

EXE: es una extensión que se refiere a un archivo ejecutable de código reubicable.

EXPRESION: Línea de instrucciones de programación. Puede ser una condición o una instrucción simple.

HARDWARE: Parte física dentro de un sistema.

IDE: interfaz de Entorno de Desarrollo.

INPUT32.dll: Librerías para sistema operativo Windows que sirven para el control de puertos.

LPT: Nombre que se le da al puerto paralelo del PC.

OCX: Es un acrónimo que significa "OLE Control Extension". OLE a su vez significa Object Linking and Embedding. Hace referencia a módulos que publican controles y funciones para ser utilizados en programas para Windows.

PISTAS DE UN CIRCUITO: Líneas de cobre de una placa de circuito por donde corre o se transmite electricidad.

PUCESA: Pontificia Universidad Católica del Ecuador Sede Ambato.

SOFTWARE: Parte no física, es decir la aplicación o los datos de un sistema.

SPP: Estándar Parallel Port

TERMOTRANSFERENCIA: Técnica para transferir o imprimir algo por medio de calor.

UTP: Unshielded Twisted Pair, par trenzado no apantallado. Es un tipo de cable utilizado principalmente para comunicaciones.

VERSATILIDAD: Facilidad grande para el cambio.

VISUAL BASIC: Lenguaje de Programación de fácil entendimiento.

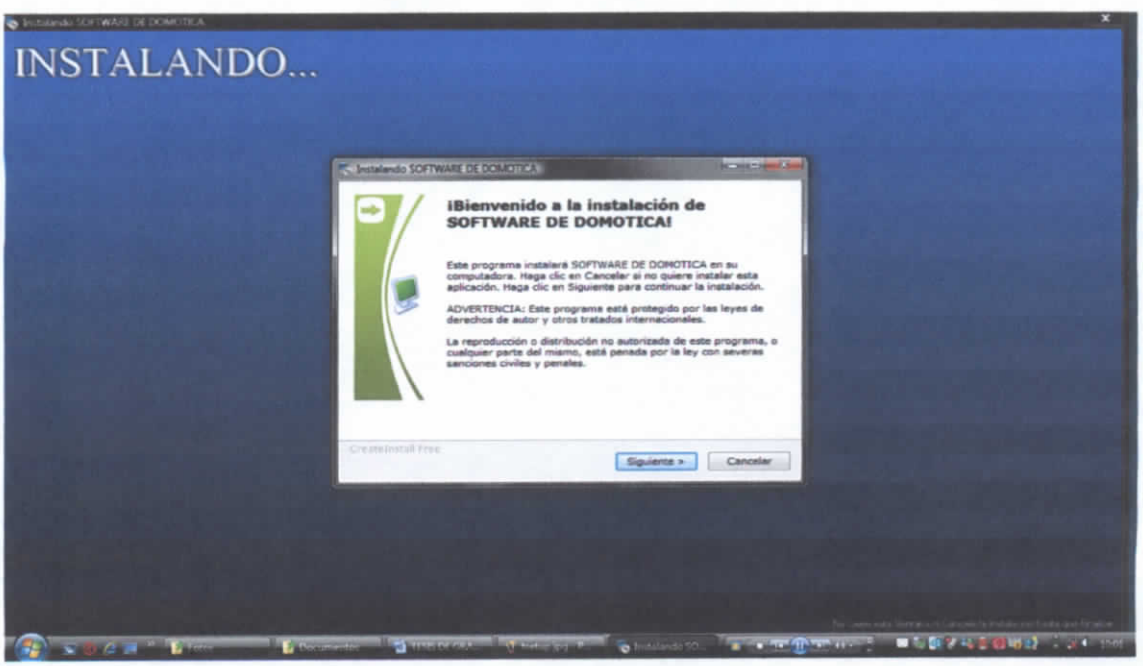
Anexo 1 Manual de usuario

Instalación del Software

La instalación comienza dando doble clic en el icono de setup del software. Para ejecutar la aplicación de instalación.

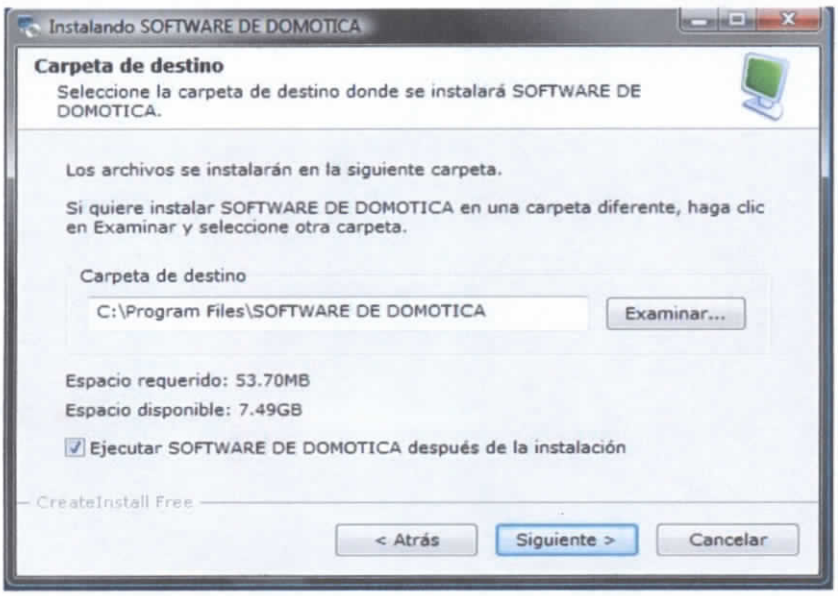


Luego se nos abrirá una ventana indicando que vamos a instalar el software.

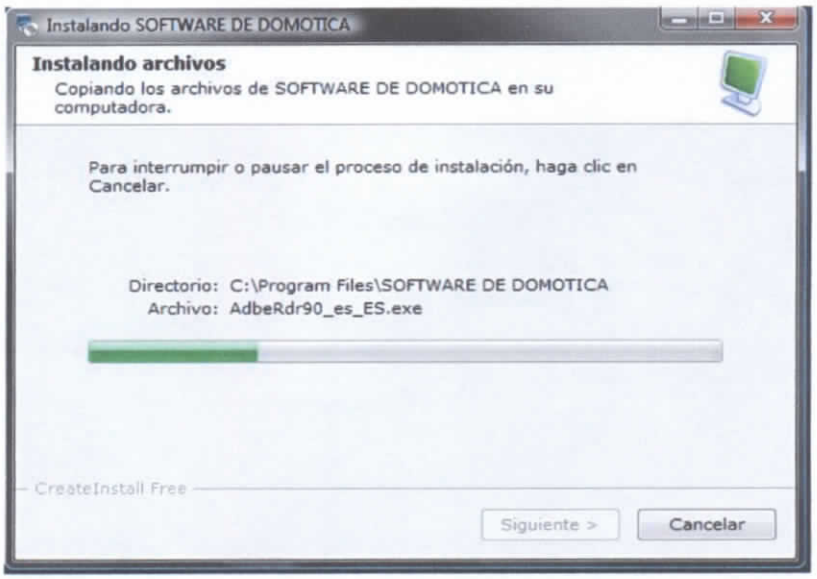


Al dar clic en siguiente nos aparece la ventana en la cual esta predeterminada la dirección o ruta donde el software se va a instalar, esta puede ser modificada en el caso de ser necesario.

A demás muestra una casilla donde pregunta si deseamos ejecutar el software al terminar la instalación, esto será elección del usuario.



Al dar clic en siguiente empezara la copia de archivos, incluyendo las aplicaciones necesarias para la ejecución correcta del software las cuales son: flash player, Shockwave y Adobe Reader. Estas aplicaciones son necesarias no cancele su instalación. Una vez terminada la instalación indicara una nueva pantalla con un botón finalizar.

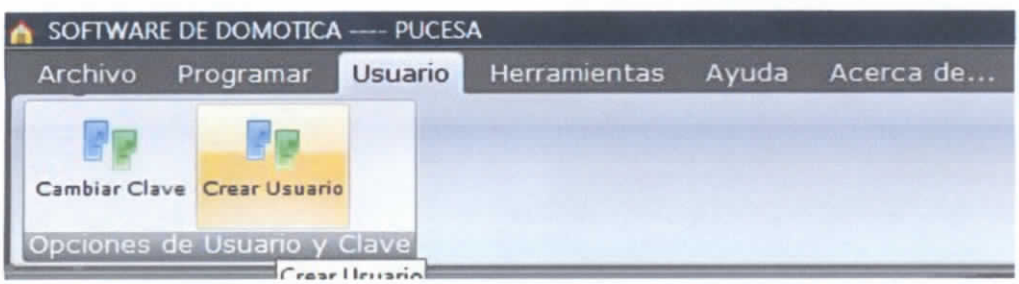


Para terminar la instalación de clic en el botón Finalizar.

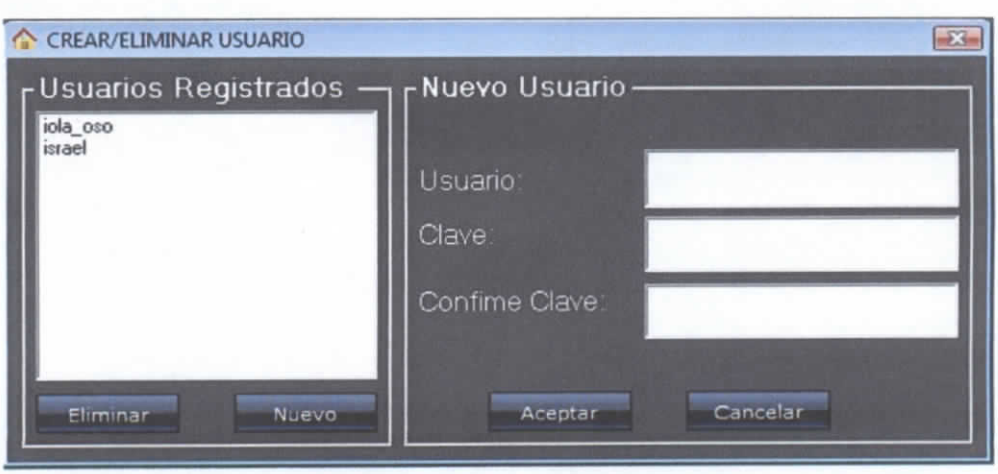


Ingreso de Nuevo Usuario

Para el ingreso de un nuevo usuario debemos posicionarnos en la pestana Usuarios del menú principal y damos un clic en el botón Crear Usuario.



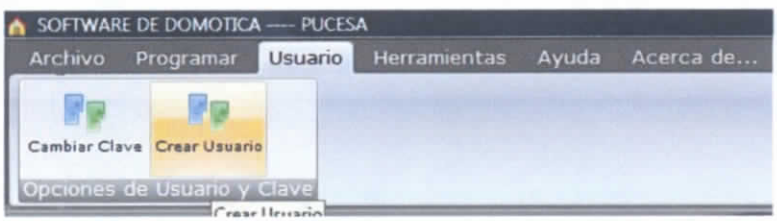
Al dar clic en este botón nos despliega una pantalla en la cual haremos clic en el botón nuevo, que contiene las siguientes opciones:



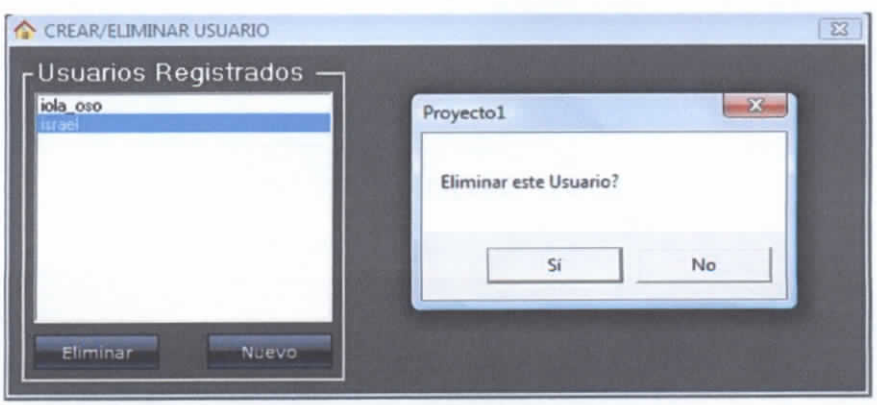
Tenemos en la parte izquierda de la pantalla los usuarios que están registrados en el sistema y en la parte derecha tenemos los campos necesarios para crear un usuario nuevo.

Eliminar Usuario

Para la eliminación usuario debemos posicionarnos en la pestaña Usuarios del menú principal y damos un clic en el botón Crear Usuario.

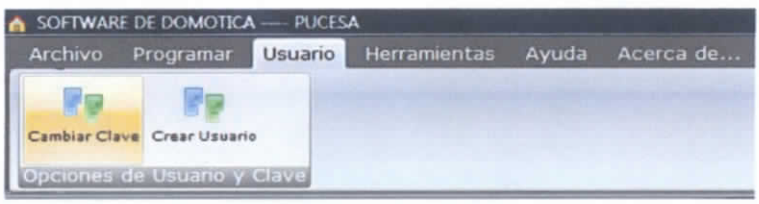


Al Dar clic en este botón nos despliega una pantalla en la cual seleccionaremos en el lado izquierdo el usuario a ser eliminado, haremos clic en el botón eliminar, nos pedirá la Confirmación para eliminar el usuario:

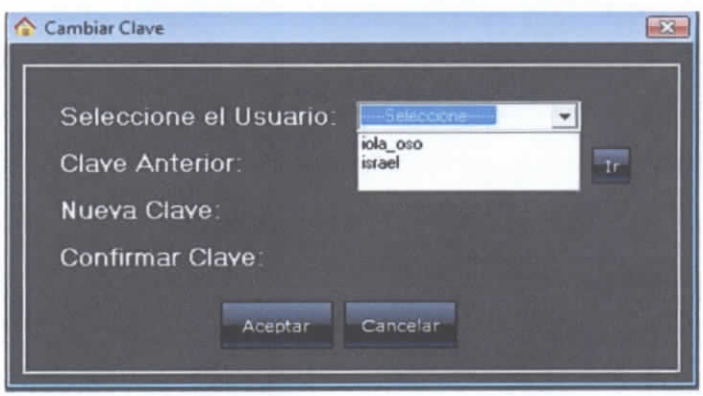


Cambiar Clave

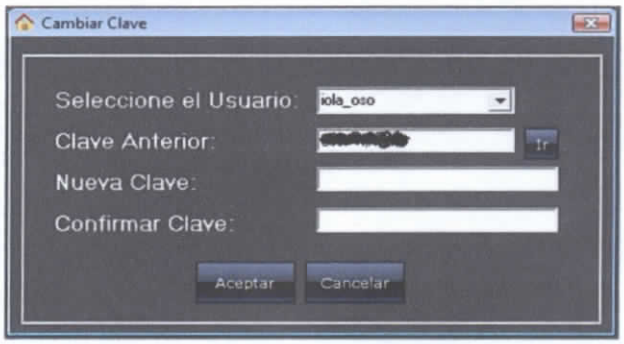
Para cambiar la clave de un usuario debemos posicionarnos en la pestaña Usuarios del menú principal y damos un clic en el botón Cambiar Clave.



Al Dar clic en este botón nos despliega una pantalla en la cual seleccionaremos el usuario el cual queremos asignarle una nueva clave.



Una vez seleccionado el usuario a modificar su clave escribimos la clave anterior y damos un clic en el botón ir para que nos muestre los campos donde podremos escribir la nueva clave con su respectiva confirmación, luego presionamos el botón aceptar.



Acceso al Sistema

Al dar doble clic en el icono del sistema este nos mostrara la pantalla de Intro la cual se muestra mientras el sistema es cargado.



Posteriormente se despliega la pantalla de acceso al sistema en la cual escribiremos nuestro nombre de usuario y la clave, luego damos clic en el botón aceptar.

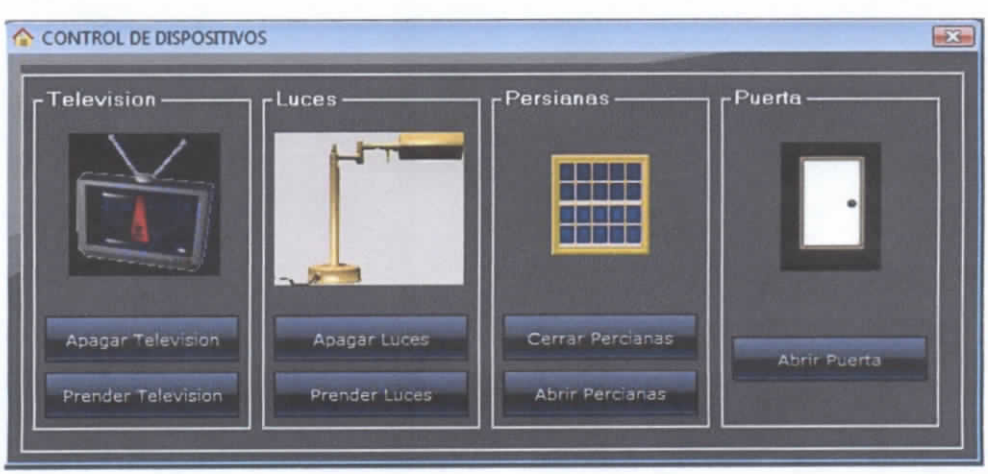


Control de Dispositivos

Para tener el control de los dispositivos debemos posicionarnos en la pestana Archivo del menú principal y damos un clic en el botón Inicio.

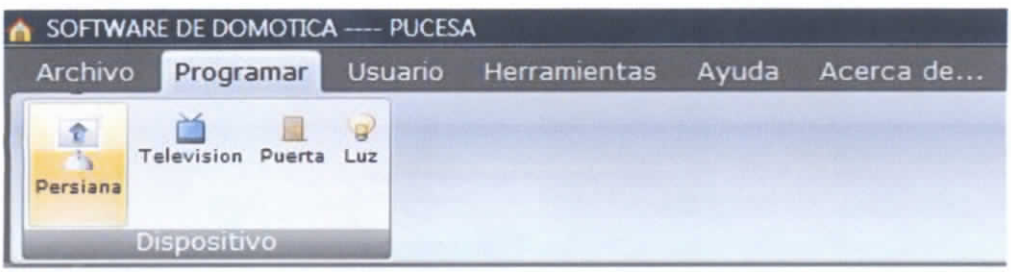


Se despliega una pantalla la que contiene las opciones de los dispositivos que podemos controlar con el sistema, allí podremos escoger una acción determinada para cada uno.



Programar un Dispositivo

Dando clic en la pestaña Programar del menú principal, escogemos el dispositivo a programar



Se despliega una pantalla la que contiene las opciones para programar dispositivos en la parte izquierda tenemos los dispositivos que pueden ser programados, mientras que en la parte superior derecha nos muestra la hora actual, y en la parte inferior derecha las opciones de programación para cada uno de los dispositivos.



Tenemos que tener en cuenta que las dos horas deben ser programas con una diferencia por lo menos de un minuto entre tarea y tarea para que el sistema pueda realizarlas.

Tareas Programadas

En esta sección se adjuntaran todas las tareas que han sido previamente programadas en la sección Programar Dispositivo, se mostrara la hora en que uno u otra tarea va ha ser ejecutada, se debe tomar en cuenta que el sistema debe esta activo para la realización de las tareas.

A demás contamos con dos opciones que nos permiten borrar todas las tareas a realizar y borrar una tarea en especial si queremos escoger solo una.



Habilitar/Deshabilitar Dispositivo

Desde el Menú principal podemos habilitar o deshabilitar los dispositivos, dando un clic en la pestaña Archivo en la sección Habilitar/Deshabilitar. Esto impedirá que el funcionamiento de cierto dispositivo seleccionado.



Cambiar Tema

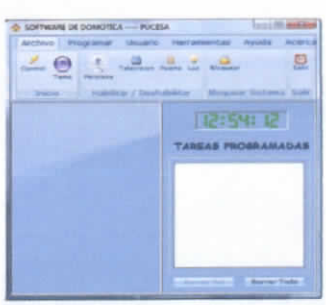
Desde el menú Archivo en la sección de Inicio podemos escoger con el botón Tema, los diferentes temas con que el Software de Domótica cuenta.

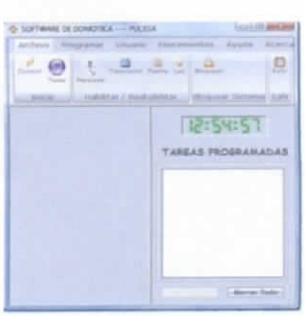


El Software cuenta con 3 diferentes temas que el usuario podrá escoger a su gusto para confinar con su tema de escritorio.

Black Theme **Blue Theme** **Silver Theme**







Bloquear Sistema

Desde el menú Archivo en la sección Bloquear Sistema, con el botón Bloquear Podemos dejar Nuestro sistema bloqueado para así impedir la manipulación del mismo, y evitar el uso de los dispositivos controlados por este.



Al bloquear el sistema nos presentara una pantalla donde nos solicita el ingreso del usuario y su contraseña para poder volver a activar el mismo.



Ver Puerto

En el menú Herramientas en la sección de Puerto Paralelo, en el botón Ver podemos saber que dirección esta asignada en el sistema para el control del puerto paralelo.

Pero cabe destacar que esta dirección puede cambiar de acuerdo al modelo, marca, tecnología, u otras características propias del computador.