

ESCUELA POLITECNICA DEL EJERCITO

SEDE LATACUNGA

**FACULTAD DE INGENIERIA DE SISTEMAS E
INFORMATICA**

**DISEÑO E IMPLEMENTACIÓN DE UN PROTOTIPO DE DOMÓTICA PARA EL
CONTROL DE ILUMINACIÓN Y ACTIVACIÓN DE UN SISTEMA DE
SEGURIDAD DE UNA VIVIENDA CONTROLADO MEDIANTE INTERNET**

**PROYECTO PREVIO A LA OBTENCION DEL TITULO DE INGENIERO DE
SISTEMAS E INFORMATICA**

**ESTRELLA PRUNA ANA LUCIA
LOPEZ SANTACRUZ SUSANA DEL ROCIO**

Latacunga, Diciembre 2003

CERTIFICACIÓN

Se certifica que el presente trabajo fue desarrollado por Estrella Pruna Ana Lucía y López Santacruz Susana del Rocío, bajo nuestra supervisión.

Latacunga, Diciembre del 2003

Ing. Santiago Jácome
DIRECTOR DE PROYECTO

Ing. José M. Rodríguez
CODIRECTOR DE PROYECTO

AGRADECIMIENTO

Gracias Padre Dios, por el privilegio de la vida, por brindarme la sabiduría para alcanzar todas mis metas; y porque me alientas a que me esfuerce y sea valiente, a que no tema los retos, ni desmaye por las adversidades; porque tú estarás conmigo donde quiera que vaya. (Josué 1:9)

Gracias Queridos Padres, por su apoyo y sacrificio incondicional, por sus sabios consejos y el legado sin igual que he recibido de ustedes el Amor y la Verdad; por ello y por muchas cosas más, siempre los llevaré presente en mi mente, corazón y en cada uno de mis triunfos.

Gracias Esposo Amado, por tu comprensión, ejemplo y cariño; porque cuando más te he necesitado has estado conmigo sosteniéndome para no desmayar y seguir adelante en el camino que me he trazado; tú eres y serás el peldaño que me ayude a escalar, cada etapa de mi vida.

Susana López S.

DEDICATORIA

El presente trabajo fruto del esfuerzo y constancia, te lo dedico a tí Luis, mi Querido Esposo; por compartir nuestro tiempo con el esfuerzo para alcanzar mis sueños; y porque sé que mis alegrías y mis logros también son los tuyos.

Susana López S.

AGRADECIMIENTO

Doy Gracias a Dios, por todas las bendiciones recibidas a lo largo de mi vida; por la fortaleza y ayuda que me ha brindado para alcanzar todas mis metas.

Muy especialmente doy gracias a mis Padres, por la bondad y dedicación que han puesto en mi superación personal e intelectual; ya que sin su apoyo moral y económico no hubiera podido consolidar mis aspiraciones.

Agradezco a mi Esposo Xavier, por todo el apoyo brindado en la culminación de mi carrera.

Ana Lucía Estrella P.

DEDICATORIA

El presente Proyecto de Tesis, la dedico con todo mi amor a mi hija Maybrith Akane; porque con su amor e inocencia ha logrado despertar en mí el deseo de superación, para de esta manera poder brindarle un porvenir mejor.

Ana Lucía Estrella P.

10 de Diciembre del 2003

Ana L. Estrella P.

AUTORA

Susana R. López S.

AUTORA

Ing. Eddy Galarza

**DECANO DE LA FACULTAD DE INGENIERIA
DE SISTEMAS E INFORMATICA**

Dr. Mario Lozada P.

SECRETARIO ACADÉMICO

INDICE GENERAL

	PAGS.
I.- AUTOMATIZACIÓN DOMÉSTICA	1
1.1.- DEFINICIÓN	1
1.1.1.- ASPECTOS TÉCNICOS	3
1.2.- ESCENARIO DE INFORMATIZACION DOMESTICA	6
1.2.1.-LA INTERFAZ HOMBRE MAQUINA	6
1.2.2.-TIPOS DE PRODUCTOS	7
1.3.- TENDENCIAS ACTUALES DE ESTANDARIZACION	8
1.3.1.- NECESIDAD DE LA ESTANDARIZACION	8
1.3.2.- EST. ACT. ESTANDARIZACIÓN DOMÉSTICA	9
II.- ESTANDARES APLICADOS EN LA DOMÓTICA	12
2.1.- RESEÑA	12
2.2.- <i>CLASIFICACIÓN</i>	<i>14</i>
2.2.1.- ESTÁNDAR CONSUMER ELECTRONIC BUS	14
2.2.2.- ESTÁNDAR X10.	16
2.2.3.- ESTÁNDAR LONWORKS	18
2.2.4.- ESTÁNDAR SMART HOUSE.	20
2.3.- ARQUITECTURA DEL ESTANDAR CEBus	21
2.3.1.- ESTRUCTURA DE NIVELES	21
2.3.2.- EL MODELO OSI VRS. CEBus	26
2.3.3.- MODELO DEL NIVEL PAR A PAR	27
2.3.4.- ESTRUC. DEL PAQUETE DE DATOS CEBus	28

2.3.5.- FALLO EN LAS TRANSMISIONES	29
2.3.6.- EL NIVEL DE APLICACIÓN (NA)	30
2.3.7.- NIVEL DE RED (NR)	35
2.3.8.- NIVEL DE ENLACE DE DATOS	37
2.3.9.- NIVEL FÍSICO	39
2.4.- LENGUAJE DE APLICACIONES COMUNES (CAL)	41
2.4.1.- ESPECIFICACIONES DEL CAL	41
2.4.2.- OBJETIVOS DEL CAL	42
2.4.3.- MODELACIÓN DE PRODUCTOS CAL	43
2.4.4.- ESTRUCTURA DE DATOS DEL CONTEXTO	43
2.4.5.- INTERPRETE CAL	46
III.- ELEMENTOS X10 PARA IMPLEMENTACIÓN DE SISTEMAS DOMESTICOS	47
3.1.- SISTEMAS DE SEGURIDAD	47
3.2.- SISTEMA DE ILUMINACIÓN DE ARTEFACTOS	49
3.3.- SISTEMA DE AUDIO / VIDEO	51
3.4.- SISTEMA TELEFÓNICO Y DE COMUNICACIONES.	53
3.5.- SISTEMA DE AUTOMATIZACIÓN AMBIENTAL	55
IV.- OPCIONES DEL NIVEL FISICO	58
4.1.- GENERALIDADES	58
4.2.- EL MEDIO DE LÍNEA DE ALIMENTACIÓN ELÉCTRICA (PL)	59
4.3.- EL MEDIO DE CABLE DE PAR TRENZADO (TP)	61
4.4.- EL MEDIO DE RAYOS INFRARROJOS (IR)	63
4.5.- EL MEDIO DE RADIO FRECUENCIA (RF)	65

4.6.- EL MEDIO DE CABLE COAXIAL (CX)	67
4.7.- EL MEDIO DE FIBRA ÓPTICA (FO)	68
V.- CONSTRUCCIÓN DEL PROTOTIPO ELECTRÓNICO	72
5.1.- DESCRIPCIÓN	72
5.2.- DISEÑO DE LAS PLACAS DE CIRCUITO IMPRESO	75
5.2.1.- PLC. POTENCIA Y ACONDICIONAMIENTO DE SEÑAL DEL SENSOR DE TEMPERATURA	76
5.2.2.- PLC. CONTROL Y ADQUISICIÓN DE DATOS	77
5.2.3.- ASIGNACIÓN DE PINES DEL PUERTO PARALELO PARA EL PROYECTO	78
5.2.4.- PLC. CONTROL Y ADQUISICIÓN DE DATOS	82
VI.- CONSTRUCCIÓN DEL SOTWARE	84
6.1.- METODOLOGÍA UML	84
6.2.- NORMAS ISO EEE830	95
6.3.- ANÁLISIS DE LA APLICACIÓN	99
6.4.- DISEÑO DE LA APLICACIÓN	100
6.4.1.- INTRODUCCIÓN	100
6.4.2.- PROPÓSITO	101
6.4.3.- DEFINICIONES, ACRÓNIMOS Y ABREVIATURAS	101
6.4.4.- REFERENCIAS	102
6.4.5.- DESCRIPCIÓN GENERAL	102
6.4.6.- CARACTERÍSTICAS DEL USUARIO	107
6.4.7.- RESTRICCIONES	107
6.4.8.- REQUISITOS ESPECÍFICOS	108

6.4.9.- CASOS DE USO DE ALTO NIVEL	112
6.4.10.- CASOS DE USO EXPANDIDOS	119
6.4.11.- DISEÑO DE ALTO NIVEL	131
6.4.12.- CONTRATOS DE OPERACIÓN	135
6.4.13.- DIAGRAMAS DE COLABORACIÓN	145
6.5.- IMPLEMENTACIÓN Y PRUEBAS	162
6.5.1.- CONFIGURACIÓN DE LA RED	162
6.5.2.- CONFIGURACIÓN DEL SERVIDOR WEB	163
6.5.3.- PRUEBAS	164
6.5.4.- MANUAL DE OPERACIONES.	167
VII.- CONCLUSIONES Y RECOMENDACIONES	168
7.1.- CONCLUSIONES	168
7.2.- RECOMENDACIONES	171

INDICE DE TABLAS

PAGS

Tabla 2.1 Responsabilidad del Nivel de Red	36
Tabla 2.2 Responsabilidad del Nivel de Enlace	38
Tabla 2.3 Responsabilidades del Nivel de Red	40
Tabla 2.4 Sincronización de los símbolos	40
Tabla 5.1 Registros del Puerto Paralelo	73
Tabla 5.2 Pines del Registro 378H	74
Tabla 5.3 Pines del Registro 379H	74
Tabla 5.4 Pines del Registro 37A H	75
Tabla 5.5 Plc. Potencia y Acondicionamiento del S. Temperatura.	78
Tabla 5.5 Lista de Materiales de Cirt. de Potencia Activa Luces y Sirena	80
Tabla 5.6 Lista de Materiales del Cirt de Acondicionamiento de Señal del Sensor de Temperatura.	81
Tabla 5.7 Lista de Materiales del Circuito de Control	82
Tabla 5.8 Lista de Materiales del Circuito de Adquisición de Datos	83
Tabla 6.1 Definiciones de Smart House Control	101
Tabla 6.2 Acrónimos de Smart House Control	101
Tabla 6.3 Abreviaturas de Smart House Control	102
Tabla 6.15 Requisitos, Operaciones y Pruebas	166

INDICE DE FIGURAS

	PAGS.
Fig. 2.1 Diagrama del modelo OSI	23
Fig. 2.2 Modelo de Protocolo CEBus vrs. Modelo OSI	26
Fig. 2.3 Cada nivel del protocolo se comunica directamente con su homólogo en el nodo destino	28
Fig. 2.4 Estructura del Nivel de Aplicación	31
Fig. 4.1 Medio de Línea de Alimentación Eléctrica.	59
Fig. 4.2 Medio de Cable Par Trenzado	61
Fig. 4.3 Medio de Rayos Infrarrojos.	63
Fig. 4.4 Medio de Radio Frecuencia.	65
Fig. 4.5 Medio de Cable Coaxial.	67
Fig. 4.6 Medio de Fibra Óptica.	68
Fig. 5.1 Ficha DB25, funciones de los distintos pines	75
Fig. 5.2 Diagrama del Circuito de Potencia que Activa Luces y Sirena	79
Fig. 5.3 Diagrama del Circuito de Acondicionamiento de Señal del Sensor de Temperatura	80
Fig. 5.4 Diagrama del Circuito de Control y Adquisición de Datos	82
Fig. 6.1 Diagramas del Modelo UML	86
Fig. 6.2 Diagramas de Casos de Uso Ejemplo Maquina de café	87
Fig. 6.3 Actor	87

Fig. 6.4 Diagramas de Secuencia Ejemplo Maquina de café	88
Fig. 6.5 Diagramas de Colaboración Ejemplo Maquina de café	89
Fig. 6.6 Diagramas de Estados Ejemplo Maquina de café	90
Fig. 6.7 Diagramas de Actividades Ejemplo Maquina de café	92
Fig. 6.8 Diagrama de Componentes Ejemplo Maquina de café	93
Fig. 6.9 Diagrama de Secuencia Ejemplo Maquina de café	94
Fig. 6.10 Componentes Ejemplo Maquina de café	95
Fig. 6.11 Casos de Uso del Actor Administrador	112
Fig. 6.12 Casos de Uso Ingresar Permiso de Acceso	131
Fig. 6.13 Casos de Uso Encender Luz Habitación1	131
Fig. 6.14 Casos de Uso Apagar Luz Habitación1	132
Fig. 6.15 Casos de Uso Visualizar Est. Act. Sensor Puerta de Ingreso	133
Fig. 6.16 Casos de Uso Visualizar Est. Act. Sensor Ventana1	133
Fig. 6.17 Casos de Uso Visualizar Est. Act. Sensor de Movimiento	133
Fig. 6.18 Casos de Uso Visualizar Est. Temp. Interna de la Resistencia	134
Fig. 6.19 Casos de Uso Inicializar Estados	134
Fig. 6.20 Diagrama de Colaboración Solicitar Operación Ingresar Permiso de Acceso	146
Fig. 6.21 Diagrama de Secuencia Solicitar Operación Ingresar Permiso de Acceso	146
Fig. 6.22 Diagrama de Colaboración Ingresar Nombre del Permiso y Confirmar Operación	147
Fig. 6.23 Diagrama de Secuencia Ingresar Nombre del Permiso y Configurar Operación	147

Fig. 6.24 Diagrama de Colaboración Solicitar Operación	
Encender Luz Habitación1	148
Fig. 6.25 Diagrama de Secuencia Solicitar Operación Encender Luz Habitación1	
	148
Fig. 6.26 Diagrama de Colaboración Pulsa Botón LH1	149
Fig. 6.27 Diagrama de Secuencia Pulsa el Botón LH1	149
Fig. 6.28 Diagrama de Colaboración Solt. Operc. Apagar Luz Habit1	150
Fig. 6.29 Diagrama de Secuencia Solt. Operación Apagar Luz Habit1	150
Fig. 6.30 Diagrama de Colaboración Pulsa Botón LH1OFF	151
Fig. 6.31 Diagrama de Secuencia Pulsa el Botón LH1 OFF	151
Fig. 6.32 Diagrama de Colaboración Visualizar Sensor Puerta Ingreso1	152
Fig. 6.33 Diagrama de Secuencia Visualizar Sensor Puerta de Ingreso1	152
Fig. 6.34 Diagrama de Colaboración Visualizar Est. Act. Sensor Puerta de Ingreso1	153
Fig. 6.35 Diagrama de Secuencia Visualizar Est. Act. Puerta de Ingreso1	153
Fig. 6.36 Diagrama de Colaboración Visualizar Est. Sensor Ventana1	154
Fig. 6.37 Diagrama de Secuencia Visualizar Est. Sensor Ventana1	154
Fig. 6.38 Diagrama de Colaboración Visualizar Est. Act. Sensor Ventana1	155
Fig. 6.39 Diagrama de Secuencia Visualizar Est. Act. Sensor Ventana1	155
Fig. 6.40 Diagrama de Colaboración Visualizar Est. Sensor Movimiento	156
Fig. 6.41 Diagrama de Secuencia Visualizar Est. Sensor Movimiento	156
Fig. 6.42 Diagrama de Colaboración Visualizar Est. Act. Sensor Movimiento	157
Fig. 6.43 Diagrama de Secuencia Visualizar Est. Act. Sensor Movimiento	157

Fig. 6.44 Diagrama de Colaboración Visualizar Temperatura Interna de la Residencia en grados centígrados	158
Fig. 6.45 Diagrama de Secuencia Visualizar Temperatura Interna de la Residencia en Grados Centígrados	158
Fig. 6.46 Diagrama de Colaboración Visualiza Temperatura Interna de la Residencia en grados centígrados	159
Fig. 6.47 Diagrama de Colaboración Visualiza Temperatura Interna de la Residencia en grados centígrados	159
Fig. 6.48 Diagrama de Colaboración Solicita Operación Inic. Estados	160
Fig. 6.49 Diagrama de Secuencia Solicita Operación Inic. Estados	160
Fig. 6.50 Diagrama de Colaboración Pulsa Botón Reset	161
Fig. 6.51 Diagrama de Secuencia Pulsa Botón Reset	161

INDICE DE ANEXOS

DESCRIPCIÓN GRÁFICA DE ELEMENTOS ELECTRÓNICOS

A1

RESUMEN DE NOTACIÓN UML

INTRODUCCIÓN

Los países tecnológicamente desarrollados han puesto en el mercado una serie de propuestas para quienes desean disfrutar de las ventajas de un edificio o residencia automatizada. Actualmente se intenta saturar el mercado con los denominados Sistemas de Seguridad, los mismos que son monitoreados por las respectivas agencias que los distribuyen, haciendo que el “usuario” entre comillas, se mantenga al margen de lo que sucede en su negocio, oficina o vivienda .

Teniendo en cuenta este análisis se ha propuesto presentar el actual proyecto de tesis con la tentativa, de que se amplíen más los horizontes en lo que a Automatización Doméstica se refiere.

Con ello se ha planteado desarrollar un prototipo de Domótica para el Control de Iluminación y Activación de un Sistema de Seguridad de una Vivienda Controlado Mediante Internet, esta red mundial fusionada con el Sistema denominado “Smart House Control” (SHC), permitirá al usuario controlar y monitorear su residencia desde cualquier lugar en el que se encuentre, en el momento que desee, durante las 24 horas del día, los 365 días del año.

I.- AUTOMATIZACIÓN DOMÉSTICA

1.1.- DEFINICIÓN¹

Por definición, Automatización se refiere a un sistema o método por el cual ciertos eventos son ejecutados y controlados sin una actividad conciente directa. Las industrias comerciales han utilizado estas técnicas por algún tiempo, desarrollando productos superiores en forma rápida y eficiente.

Se denomina Automatización Doméstica a la interconexión de productos destinados a las tareas del hogar, los cuales se beneficiarán mutuamente al estar interconectados como pueden ser: televisor, video grabadora, sistema telefónico, sistema de seguridad, de iluminación, etc. La tendencia a la interconexión de tales productos se debe al echo de que estos ofrecen un mejor servicio con mejores características al estar interconectados que al trabajar independientemente.

Algunos beneficios que ofrece la Automatización Doméstica a los consumidores incluyen conveniencia (comodidad), seguridad (protección) y una mejor administración del consumo de la energía eléctrica, como también beneficios para personas inválidas y de avanzada edad. El beneficio más significativo para los consumidores es la comodidad.

¹ [http:// www.domotica/introduccion.html](http://www.domotica/introduccion.html)

La Automatización Doméstica permite que toda la residencia se encuentre automatizada, suministra facilidad y comodidad en las tareas diarias, con características de: encendido / apagado, comunicaciones y dispositivos inteligentes; todos estos contribuyen a la comodidad y facilidad que una residencia automatizada puede proporcionar. Por otro lado el acceso remoto a los distintos dispositivos lo hacen más conveniente, las características de seguridad pueden fácilmente incluirse en una residencia automatizada. Un simple ejemplo sería el encender la luz randómicamente y a varios momentos con el objeto de simular presencia en la residencia cuando realmente no hay nadie en ella.

Un sistema más completo incluye sensores los cuales podrían detectar movimiento en las afueras de la residencia, prender automáticamente las luces y proyectar una imagen desde una cámara en el exterior hasta un televisor en el interior, para que su ocupante pueda observar lo que sucede.

Características de protección podrá incluir el apagar dispositivos potencialmente peligrosos cuando los ocupantes no se encuentren. Una alarma podría poner en acción un sin número de eventos como un ejemplo tendríamos: una alarma de incendios que no solamente produzca ruidos, sino también ilumine el mejor camino de salida y seguramente llame hasta al Cuerpo de Bomberos. Todo esto es posible debido a la interconectividad de los elementos de una residencia.

La administración de energía en una residencia automatizada puede ayudar a reducir las planillas de consumo. Una forma en que la

Automatización Doméstica genera ahorros es suministrando calefacción y aire acondicionado más eficientemente.

Otra forma incluye la habilidad de monitoreo, el uso de energía de cada una de los electrodomésticos y apagar aquellos dispositivos que consumen energía cuando no es necesario.

La Automatización Doméstica se ha esforzado por lograr una amplia aceptación por parte de la comunidad; pero la naturaleza y la tecnología la hace especialmente útil para las personas discapacitadas y de avanzada edad.

Al automatizar una residencia se puede lograr de cierta forma ayudar a los discapacitados a mantener una vida activa, algo tan simple como encender una luz o el televisor puede ser una tarea difícil para una persona con incapacidad pero al automatizar esta tarea se logra eliminar o aliviar el grado de incapacidad de muchas personas. Más y más sistemas de Automatización Doméstica se encuentran incorporando el campo de las telecomunicaciones en el hogar, lo que permitiría tanto a los ancianos, como a los discapacitados notificar fácilmente a familiares o amigos en caso de una emergencia.

1.1.1.- ASPECTOS TÉCNICOS²

Ciertas técnicas son muy importantes para el desarrollo de cualquier producto, en el campo de la Automatización Doméstica hay varias técnicas de las cuales se puede elegir. Es muy importante no solo entender las técnicas utilizadas sino también como estas interactúan para producir el resultado final. Los aspectos técnicos más importantes en una residencia automatizada son los medios por los cuales las señales necesitan viajar, los dispositivos a ser controlados y el lenguaje que utilizan para comunicarse.

La estructura básica de la técnica empleada por la Automatización Doméstica empieza con una señal, esta señal necesita ser producida, transmitida, e interpretada.

- Se necesita un dispositivo de control el cual produce la señal, estos incluyen paneles de control, televisores, dispositivos de control remoto entre otros.

- Las señales necesitan un medio por el cual movilizarse, estos medios incluyen:
 1. Línea de alimentación eléctrica (PL)
 2. Cable de par trenzado (PT)
 3. Cable coaxial (CX)
 4. Radios infrarrojos (IR)

² <http://www.domotica/tecnologia.html>

5. Radio frecuencia (RF)
6. Fibra óptica (OF)

Los medios de transmisión son importantes para el desempeño de la red doméstica, estos suministran el sendero por el cual se transmiten las operaciones para entrar por entre ellas.

En el campo de la Automatización Doméstica existen 4 estándares que transmiten y reciben señales, estas son:

1. X10
2. CEBus
3. LonWorks
4. Smart House

Cada estándar se asemeja a un lenguaje, en este caso si en una residencia se utiliza determinado estándar entonces, los dispositivos no podrán comunicarse con otros regidos bajo un estándar diferente.

- **X10.-** Fue el primer estándar en ser establecido y a pesar de sus características limitadas su utilización ha ido aumentando debido al bajo costo de sus productos y fácil instalación.
- **CEBus y LonWorks.-** Poseen características más avanzadas.
- **Smart House.-** Es un estándar destinado a residencias totalmente nuevas.

La Automatización Doméstica no es solo poderosa sino también diversa. Actualmente la tecnología se encuentra avanzando de forma impresionante. Durante la década pasada, muchas compañías se dedicaron a desarrollar lo que ahora tenemos en nuestras manos; podemos imaginar que facilidades nos ofrecerán en el futuro.

1.2.- ESCENARIO DE INFORMATIZACION DOMESTICA

1.2.1.- LA INTERFAZ HOMBRE MAQUINA³

Se han establecido parámetros de evaluación de la interfaz Hombre-Maquina tomando en cuenta los procesos de operación y el comportamiento de hardware así:

- a. Posibilidad de operación y coordinación con el accionar humano y sus procesos de raciocinio.
- b. Respuestas rápidas y correctas por parte del equipamiento.
- c. Los equipos deberán proporcionar las instrucciones e indicaciones exactas.
- d. El operar los equipos no debe afectar la salud mental y física de los operarios.
- e. Los usuarios deben ser capaces de operar fácilmente los equipos y familiarizarse con las operaciones de control.
- f. Variación de los métodos de operación con el fin de ofrecer adaptabilidad para ancianos y discapacitados.

³ Zambrano Carlos, "Edificios Inteligentes". Págs. 22-26.

- g. Facilidad de aprendizaje.
- h. Ofrecer compatibilidad, los dispositivos deberán comunicarse fácilmente sin presentar ningún tipo de inconveniente.
- i. El equipamiento debe ir acorde con la decoración del hogar.
- j. Estándares internacionales para la operación e información.

1.2.2.- TIPOS DE PRODUCTOS

De acuerdo a sus características los productos se dividen en 3 categorías.:

- a. Productos interactivos inteligentes (PII)
- b. Subsistemas inteligentes (SI)
- c. Sistemas centralizados (SC)

1.2.2.1.- Productos Interactivos Inteligentes (PII)

Estos productos se caracterizan por ser dispositivos auto contenidos (stand alone), los cuales requieren de control manual, a estos se incluyen controles remotos programables, puertas de garaje controlados remotamente, sistemas de riego automático, etc.

1.2.2.2.- Subsistemas Inteligentes (SI)

Están constituidos por dos o más sistemas interactivos inteligentes los cuales son capaces de intercambiar información con el fin de llevar a cabo tareas más complicadas, por ejemplo: la interfaz de una contestadora telefónica, se podría relacionar con los sistemas de iluminación, aire acondicionado y calefacción.

1.2.2.3.- Sistemas Centralizados (SC)

Este es el más completo de los sistemas; consiste en un sistema centralizado el cual permite agrupar a todos los productos inteligentes interactivos, en una red integrada de la cual el usuario tiene completo control. Los sistemas centralizados pueden ganar una amplia aceptación.

1.3.- TENDENCIAS ACTUALES DE ESTANDARIZACION

1.3.1.- NECESIDAD DE LA ESTANDARIZACION

Existe un gran interés en el campo de la Automatización Doméstica, los involucrados provienen de varios campos: fabricantes, comerciantes, constructores, remodeladores y

en especial de los propietarios de las residencias. Cada cual tiene sus propias razones para estar involucrado.

Muchos están interesados principalmente en desarrollar sistemas y estándares. El término estándar determina el modo en que un dispositivo debe operar.

El estándar de comunicación se refiere al modo en que dos dispositivos deberán interactuar, en otras palabras si los productos en una residencia van a ser capaces de comunicarse, lo van a hacer mejor si hablan el mismo lenguaje.

Hay 2 maneras en que los estándares están desarrollados:

- **Estándares Oficiales.-** Son aquellos promulgados por entidades especializadas como la ISO (Open Systems Interconnection).
- **Estándares De Facto.-** Son aquellos desarrollados por fabricantes líderes en el mercado, los cuales los incorporan a sus equipos y luego son adoptados por otros proveedores.

Algunos sistemas y estándares pueden proporcionar compatibilidad con otros, mientras que otros no.

1.3.2.- Estado Actual de la Estandarización Doméstica

El estándar X10 ha sido considerado como el estándar de facto durante varios años en los EEUU. Sin embargo su limitación básica radica en que es un sistema concedido únicamente para la transmisión de eventos, no de datos.

Por otro lado los dispositivos basados en CEBus permiten llamar a un dispositivo y encontrar su estado actual, sus variables de control, identificar su dirección lógica, número de modelo, fabricante, etc. Además realiza un intercambio de información en forma de paquetes, permite a los dispositivos confirmar la recepción de información y / o retransmitir sus paquetes con un número de secuencia incluidos.

X10 es un sistema abierto con potencial limitado para control inteligente del hogar puesto que no considera una transferencia de información en forma de video, voz y / o datos, por lo cual no permite diálogo sofisticado entre dispositivos domésticos y lo que es más grave no permite una interconexión a medios de información y a sistemas multimedia públicos. En consecuencia a los alcances en cuanto al grado de sofisticación que puedan lograr sus aplicaciones están encontrando cada día más limitaciones.

Smart House se ha propuesto para construcciones nuevas; debido a que exige un esquema de cableado total, por lo que su implementación en residencias preexistentes resulta complicado y costoso.

LonWorks por su parte tiene gran afinidad con el estándar CEBus, la mayoría de sus características de diseño, requerimiento y condiciones son similares a aquellas desarrolladas por la EIA (Electronic Industries Association) y CEBus (Consumer Electronic Bus).

De lo indicado anteriormente en cuanto a estandarizaciones se advierte una tendencia influyente hacia el CEBus, considerando además que este es uno de los estándares que toma en cuenta mayor cantidad de medios físicos, línea de alimentación, cable de par trenzado, cable coaxial, cable de fibra óptica, rayos infrarrojos y de radio frecuencia; y que tienen una arquitectura basada en el esquema de niveles OSI y se podría considerar como uno de los estándares más completos. Por esta razón se lo ha elegido como objeto de estudio más detallado en capítulos posteriores.

II.- ESTANDARES APLICADOS EN LA DOMÓTICA

2.1.- RESEÑA

La década de los ochenta vivió un nuevo concepto denominado Automatización Doméstica (Domótica). La enciclopedia "Larousse" definía en 1988 el término Domótica como el siguiente: "el concepto de vivienda que integra todos los automatismos en materia de seguridad, gestión de la energía, comunicaciones, etc". Es decir, el objetivo es asegurar al usuario de la vivienda un aumento del confort, seguridad, ahorro energético y facilidades de comunicación.

Una definición más técnica del concepto es: "conjunto de servicios de la vivienda garantizado por sistemas que realizan varias funciones, los cuales pueden estar conectados entre sí y a redes interiores y exteriores de comunicación. Gracias a ello se obtiene un notable ahorro de energía, una eficaz gestión técnica de la vivienda, una buena comunicación con el exterior y un alto nivel de seguridad".

Para que un sistema pueda ser considerado “inteligente” debe incorporar elementos o sistemas basados en las Nuevas Tecnologías de la Información (NTI)

El uso de las NTI en la vivienda genera nuevas aplicaciones y tendencias basadas en la capacidad de procesos de información y en la integración y comunicación entre los equipos e instalaciones. Así concebida, una vivienda inteligente puede ofrecer una amplia gama de aplicaciones en áreas tales como: seguridad, gestión de la energía, automatización de tareas domésticas, formación, cultura, entretenimiento, teletrabajo, monitorización de salud, operación y mantenimiento de las instalaciones, etc.

Los elementos de campo (detectores, sensores, etc.), transmitirán las señales a una unidad central inteligente que tratará y elaborará la información recibida. En función de dicha información y de una determinada programación, la unidad central actuará sobre determinados circuitos de potencia relacionados con las señales recogidas por los elementos de campo correspondientes.

En este sentido, una vivienda Domótica se puede definir como: “aquella vivienda en la que existen agrupaciones automatizadas de equipos, normalmente asociados por funciones, que disponen de la capacidad de comunicarse interactivamente entre sí”.

Un ejemplo diario:

Son las siete de la mañana, suena el despertador, se levantan las persianas y se enciende la luz. Puntualmente como cada mañana el procesador le despierta, con la tranquilidad de saber que ha estado toda la noche cuidando su vivienda. Si hubiera habido algún escape de agua lo habría cortado y tendría un aviso. Cuando se va de casa, toca suavemente la pantalla táctil de la entrada, ésta le comunica que no hay ninguna ventana ni puerta abierta. Cuando llegue a la oficina, ingresará a su página en Internet, introducirá su código personal y durante toda la mañana sabrá lo que pasa en su vivienda. Si ha olvidado conectar algo, podrá activarlo desde su computador y le pedirá que lo haga por usted. Lo mismo hará cuando vaya de viaje.

2.2.- CLASIFICACIÓN

Existen varias tendencias de estandarización de los sistemas de Automatización Doméstica y se destacan:

- Estándar X10
- Consumer Electronic Bus (CEBus) y EIA-600(Electronic Industries Association)
- LonWorks
- Smart House.

2.2.1.- ESTÁNDAR CONSUMER ELECTRONIC BUS (CEBUS)

CEBus EIA-600 es un estándar de comunicaciones destinado a respaldar la interconexión e interoperabilidad entre dispositivos eléctricos de una residencia, desarrollados por Electronic Industries Association (EIA) y la Consumer Electronic Manufacturec Association.

El estándar CEBus estuvo en desarrollo por más de una década y finalmente fue aprobado en 1992. Describe un conjunto de protocolos de red, optimizadas con una línea de productos y un lenguaje de comunicación denominado Common Applications Languajes (CAL). Uno de sus principales objetivos es ofrecer compatibilidad; es decir que la implementación de EIA-600 sea capaz de coexistir con cualquier otra implementación, sin tomar en cuenta su fabricante. Se caracteriza por ser un estándar abierto, es decir que cualquier dispositivo conectado bajo este estándar será capaz de comunicarse con un dispositivo EIA-600 y el lenguaje utilizado para las funciones de control será encendido por todos los dispositivos, lo que permite que cualquier individuo pueda desarrollar productos utilizando este estándar de comunicaciones.

2.2.1.1.- Breves Fundamentos de Diseño de CEBus.

Se ha diseñado en base al modelo OSI, que es un conjunto de protocolos en siete niveles, los cuales se han tomado como modelos de referencia para interconexión de sistemas abiertos.

CEBus emplea cuatro de los siete niveles de dicho modelo:

- Nivel de aplicación
- Nivel de red
- Nivel de enlace de datos
- Nivel físico.

CEBus aborda varios medios de producción como son las líneas de alimentación eléctrica (PL), cable de par trenzado (TP), cable coaxial (CX), rayos infrarrojos (IR), radio frecuencia (FR), fibra óptica (FO).

CEBus permite que un nodo se comuniquen con otro sin tomar en cuenta el medio por el cual se establece la conexión; esta propiedad permite que los productos CEBus empleen cualquier medio que sea considerado el más apropiado, los cuales podrán conectarse entre ellos a través de routers, brouters y bridges.

2.2.2.- ESTÁNDAR X10.

El estándar X10 permite que dispositivos compatibles puedan comunicarse entre sí a través del cableado de alimentación eléctrica existente en una residencia, este estándar se ha expandido considerablemente desde su introducción en 1978 y un alto número de fabricantes producen o venden productos compatibles con X10.

Estos dispositivos son diseñados para proveer funciones tales como encender / apagar y atenuar luces, entre otras; un sistema completo puede realizar funciones como: cambios de canales y control de volúmenes en equipos de audio y video, etc.

2.2.2.1.- Fundamentos Básicos de X10.

X10 se caracteriza por ser el estándar más común y su formato de codificación X-10 está patentado. X10 es un protocolo de comunicaciones diseñado para el control remoto de equipos eléctricos, está proyectado para las comunicaciones entre transmisores X10 y receptores X10 los cuales se comunican a través del cableado estándar de cualquier residencia.

▪ TRANSMISORES X10.

El transmisor X10 más simple es un dispositivo con botones de comando. Los botones seleccionan que unidad va a ser controlada y que función de control va a ser enviada a la unidad seleccionada. Por ejemplo: prender, apagar, atenuar, apagar todas las luces.

También hay transmisores con temporizadores de reloj los cuales pueden ser programados a determinadas horas; y además transmisores que actúan en base a eventos como: amanecer o atardecer, tonos, o al detectar movimiento.

▪ RECEPTORES X10.

El receptor X10 más simple es un pequeño módulo que se interpone a una toma de pared estándar y el equipo sobre el que se desea controlar el suministro de energía eléctrica.

Los comandos (botones) sobre el módulo permiten configurar su código de unidad, el módulo tiene en su interior un conmutador automático (relé) el cual activa o desactiva su toma eléctrica de salida en respuesta a instrucciones X10 dirigidas a él.

2.2.3.- ESTÁNDAR LONWORKS

LonWorks en el campo de la Domótica facilita la comunicación telemática entre nodos sin perder recursos de cálculo, o sea que recibe un mensaje de red y ejecuta la orden inmediatamente como respuesta al mensaje. Cada nodo está constituido por un microcontrolador, que recoge la información de red y la comunica a los actuadores. La tecnología LonWorks fue definida por Echelon.

Los objetivos que persigue son flexibilidad y estandarización, interoperabilidad entre empresas fabricantes, compatibilidad total entre sistemas e

independencia y flexibilidad a bajo costo. En cuanto a técnica, presenta inteligencia en el nodo, seguridad de comunicación de los datos, independencia del medio físico utilizado y un lenguaje optimizado. Cada nodo de la red está constituido por un NEURON CHIP, fabricado por Motorola y Toshiba. Esta doble patente, hace que los precios sean más razonables. El NEURON CHIP tiene como características: 3 procesadores (2 para comunicación y 1 para aplicación) Memoria EEPROM, RAM y ROM, 11 pins I/O bidireccionales, 2 contadores, timers de 16 bits. El lenguaje de programación es el Neuron C, una variante especializada del "C", cosa que simplifica mucho la configuración de nodos y red.

Los elementos que caracterizan este lenguaje son las variables de red, la cláusula "when" que provoca la activación por eventos de diversas acciones que son ejecutadas de forma cooperativa, y los objetos de entrada / salida. Cada nodo de la red interactúa con elementos como sensores o actuadores a través de los pins de I/O. El nodo va conectado a la red a través de un transductor, que varía según el medio de transmisión.

El medio más utilizado es el par trenzado, las ventajas que incorpora son la gran estandarización de la tecnología y la facilidad de programar en un lenguaje de alto nivel. Puede llegar a velocidades de 125 ó 1,25 Mbps y el medio de transmisión es totalmente transparente al usuario del nodo. Permite comunicarse alternativamente por medio de: línea de alimentación eléctrica, cable telefónico, cable coaxial, rayo infrarrojo o fibra óptica.

LonWorks utiliza la codificación Manchester que genera 0's y 1's los cuales son transmitidos a la misma velocidad del estándar de comunicaciones que se aplica a la línea de alimentación eléctrica.

2.2.4.- ESTÁNDAR SMART HOUSE.

Es un estándar que fue desarrollado por la International Association of Home Builder. Soporta audio / video, telefonía y distribución de energía a través de toda la residencia. Además de las aplicaciones de Automatización Doméstica, utiliza un cable especial propio; cada cable consta de 6 hilos de bajo voltaje (par trenzado), utilizados para enviar y recibir señales de control a través del sistema para productos tales como: luces, switch, televisores, calentadores de agua, artefactos eléctricos, etc.

Se requiere también una conexión al cable de par trenzado con el fin de obtener todas las características y beneficios que este estándar ofrece. Ésta es la mayor diferencia con los estándares anteriormente estudiados los cuales no requieren de este cable en cada una de sus ubicaciones.

Smart House, tiene proyectado ser compatible con CEBus u otros estándares en el futuro; pero actualmente ofrecen diferencias muy significativas. Su uso requiere licencia, cableado y equipos esenciales necesarios para establecer la red doméstica. Se requiere además de personas experimentadas para realizar las instalaciones y servicios de soporte.

2.2.4.1.- Características

- Ofrece un excelente servicio de soporte
- La investigación y el desarrollo que se ha involucrado con el estándar Smart House es asombroso, y los productos que actualmente se presentan son de primera clase.
- Por otro lado Smart House no pretende crear aplicaciones finales de lujo, sino aplicaciones cuya finalidad es de prestar servicios con control inteligente. Smart House es un estándar muy costoso ya que está destinado a nuevas residencias; el adaptarlo a residencias ya existentes puede resultar bastante caro. Una alternativa razonable costo / beneficio está desarrollándose y pronto estará disponible en el mercado.

2.3.- ARQUITECTURA DEL ESTANDAR CEBus

2.3.1.- ESTRUCTURA DE NIVELES

El protocolo CEBus proporciona un sistema de entrega de mensajes fiables, flexibles y de doble vía (Full Dúplex) para grandes volúmenes de tráfico, en cada uno de los medios de

transmisión. Las principales ventajas que se persiguen con el desarrollo de este protocolo son:

- Bajo costo
- Alta funcionalidad
- Facilidad de operación

El CEBus define el formato de los paquetes a ser transmitidos, los servicios de entrega de paquetes y la técnica para acceder y utilizar el canal de control en cada medio de transmisión.

A principios de los años 80 la Organización Nacional de Estándares ISO adoptó un modelo para los sistemas de comunicación abiertos, el cual es conocido como Modelo de Referencia para Interconexiones de Sistemas Abiertos (OSI). La intención de este modelo fue utilizarlo en la definición de los actuales protocolos. El modelo OSI define las funciones y servicios presentes en cualquier protocolo de comunicaciones y es utilizado como fundamento para el diseño de protocolos.

El comité CEBus también escogió este modelo para escribir el protocolo CEBus en las secciones EIA-600. El protocolo CEBus está basado en la terminología y formato del modelo OSI.

El modelo tiene algunas ventajas:

- Divide las comunicaciones en siete niveles predefinidos por esta razón es llamado el modelo OSI de siete niveles.

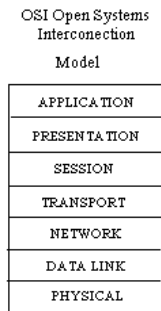


Fig. 2.1 Diagrama del modelo OSI

- El modelo define las siete funciones del protocolo de una manera jerárquica, cada nivel del modelo depende de los servicios de los niveles que se encuentran debajo de él. Cada nivel posee un conjunto específico de responsabilidades, e idealmente solo se comunica con el nivel que se encuentra sobre y debajo.
- Empezando por el NIVEL DE APLICACIÓN cada nivel inferior proporciona un servicio más primitivo que el nivel superior, el nivel de aplicación corresponde a la aplicación que utiliza el protocolo, el nivel inferior o Físico corresponde a las tareas de nivel más bajo, usualmente vinculado con el hardware.

- **NIVEL DE APLICACIÓN**

Es la aplicación que utiliza los servicios de comunicación de los medios más bajos; ejemplo: un programa de procesamiento de palabras, un programa de comunicaciones, etc.

- **NIVEL DE PRESENTACIÓN**

Realiza la traducción de los datos del nivel de aplicación en una forma tal que puede ser utilizado para la transmisión de paquetes, además es responsable de convertir los datos transmitidos y recibidos a una forma que pueda ser utilizada por el nivel de aplicación.

- **NIVEL DE SESION**

Maneja sesiones de comunicación o conexión hasta que la transmisión sea completa, este servicio es específicamente utilizado para protocolos de servicios conmutados como son: sistema telefónico, servicio de acceso compartido en tiempo, etc.

- **NIVEL DE TRANSPORTE**

Proporciona un servicio extremo a extremo, entre el nodo origen y el destino de una red, esto se realiza al requerir que el nodo destino certifique una recepción exitosa del mensaje y la retransmisión de los paquetes que no han sido reconocidos. Este nivel también maneja la fragmentación de mensajes que poseen un gran tamaño así como también la reconstrucción del mismo y

la clase de mantenimiento de conexiones de red que son requeridos por el nivel de sesión.

- **NIVEL DE RED**

Provee una comunicación confiable entre múltiples medios de transmisión, incluyendo funciones de comunicación infrarrojo como el manejo de las direcciones de red. Este nivel proporciona servicios de ruteo con el fin de asegurar que un paquete llegue a su nodo destino correcto a través del medio de transmisión adecuado.

- **NIVEL DE ENLACE DE RED**

Proporciona una comunicación confiable entre los nodos del mismo medio de transmisión. Este nivel define la unidad fundamental de transferencia de datos, la cual es generalmente conocida como paquete. La principal responsabilidad de este nivel es el acceso apropiado al medio de transmisión y la detección e impedimento de colisiones, también realiza detección de errores en los paquetes, tipo de transmisión de paquetes, reconocimiento de direcciones, rechazo a paquetes duplicados y retransmisión y detección de fallos de los paquetes transmitidos.

- **NIVEL FÍSICO**

Es responsable de transmitir y recibir de una manera confiable cada bit de símbolo a través del medio de transmisión especificado, todas las utilidades relacionadas con implementaciones mecánicas y eléctricas son determinadas en este nivel, enviando niveles de señal con cada transmisión de bits.

2.3.2.- EL MODELO OSI VRS. CEBus

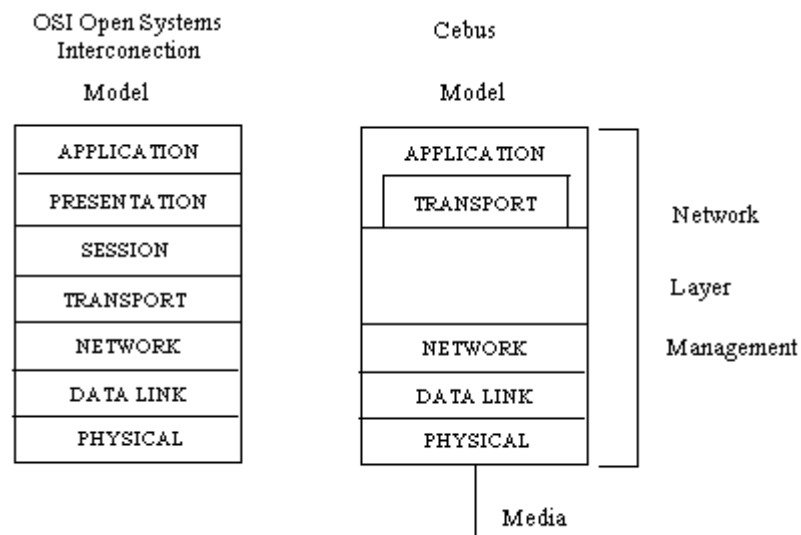


Fig. 2.2 Modelo de Protocolo CEBus vrs. Modelo OSI

El modelo es similar a la manera en que los lenguajes de programación utilizan subrutinas, cada nivel llama al nivel inferior para solicitar una función más primitiva

La figura 2.2 también muestra el modelo del protocolo CEBus con relación al modelo OSI. CEBus implementa un nivel físico, el nivel de enlace

de datos, el nivel de red y el nivel de aplicación; en adición a las funciones del protocolo definidas en el modelo OSI. El estándar CEBus también define las características físicas de cada uno de los modelos de transmisión permitidos. Este también implementa algunas transmisiones del nivel de transporte pero en vez de establecer un nivel de transporte separando, las funciones requeridas son incluidas en el nivel de aplicación.

El estándar CEBus también define funciones de administración del nivel de red en cada nivel y la interacción entre niveles. La administración del nivel de red, incluye aquellas funciones relacionadas con el mantenimiento internivel con funciones tales como: estado de espera, corrección de errores y administración de recursos.

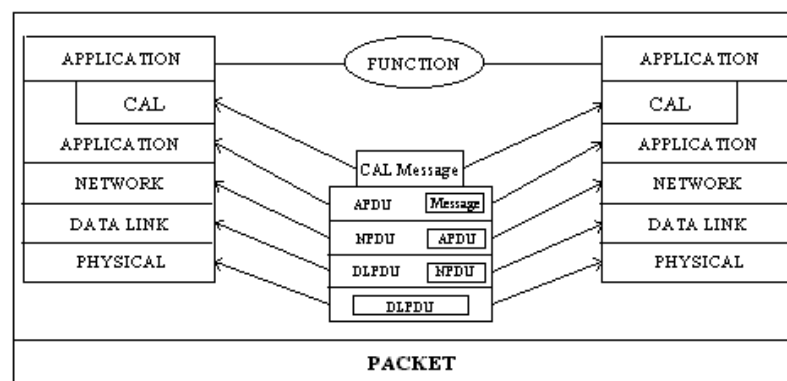
NOTA: CEBus es un servicio de protocolo no orientado a conexión, el nivel de sesión no es utilizado, el nivel de presentación tampoco es necesario ya que no se requiere ningún tipo de conversión de datos hacia o desde un formato específico. El CEBus se denomina aplicación intérprete del Lenguaje de Aplicaciones Comunes (Common Application Language CAL).

2.3.3.- MODELO DEL NIVEL PAR A PAR

El concepto de comunicaciones Par a Par es aplicable a cada nivel del protocolo, cada nivel se comunica sobre la base par a par, con el nivel

correspondiente del nodo al cual envía o del cual recibe información; cada nivel provee servicios a los niveles superiores, estos a su vez se suscriben a los servicios de los niveles inferiores, tentativos a ser enviados desde un proceso de aplicación origen a un dispositivo remoto, entran a un nodo fuente por medio del nivel de aplicación del nodo origen y se propagan hacia sus niveles inferiores, cada nivel añade su información específica al paquete de datos que recibe del nivel superior; finalmente los datos originales junto con toda la información adicional de cada nivel son enviados en una trama CEBus y transmitidos por el medio de comunicación respectivo. Una vez recibida esta trama en el nodo destino los datos siguen un proceso inverso hacia los niveles superiores, aquí cada nivel elimina la información añadida por su nivel homólogo en el nodo fuente y pasa el paquete a su nivel superior hasta llegar al nivel de aplicación del nodo destino.

La figura 2.3 indica el proceso de construcción de la trama de datos CEBus, desde que se genera la instrucción en el nivel de aplicación, hasta que el paquete completo es entregado por el nivel de enlace de datos al nivel físico para su transmisión.



La figura 2.3 Cada nivel del protocolo se comunica directamente con su homólogo en el nodo destino

2.3.4.- ESTRUCTURA DEL PAQUETE DE DATOS CEBus

La estructura de un paquete CEBus como se ilustra en la figura 2.3 refleja la contribución de cada nivel de protocolo al mensaje, el DLPDU (Data Link Protocol Data Unit); NPDU(Network Protocol Data Unit) y APDU (Application Protocol Data Unit) son originados por el nivel de enlace de datos, el nivel de red y el de aplicación respectivamente.

2.3.5.- FALLO EN LAS TRANSMISIONES

La transmisión de mensajes CAL desde el nodo destino hacia el nodo origen puede fallar en cualquier lugar del proyecto entre el intérprete CAL origen y el intérprete CAL destino, si ocurre fallo en el nodo origen los nodos involucrados son informados del error por el nivel inferior donde este ocurrió. Por ejemplo: si el nivel de enlace de datos no fue capaz de transmitir el mensaje, este informará al nivel que pidió la transmisión (el nivel de red), este nivel informará a su vez al nivel superior hasta llegar a la aplicación, cada nivel debe intentar de nuevo; si esto falla entonces renunciar. Eventualmente la aplicación debe decidir como manejar el error (tratar de informar al usuario o renunciar).

Si ocurre un fallo en el nodo destino, cada nivel es responsable de comunicar al nivel que originó el mensaje que no está en capacidad de manejar el mensaje. Por ejemplo: si es que el nodo origen y el nivel de aplicación requieren servicios de seguridad por parte del nodo destino, y si el nodo destino no es capaz de manejar mensajes seguros, la codificación es un servicio opcional. Luego el nodo destino del nivel de aplicación debe enviar un encabezado APDE (Application Protocol Data Error) de nuevo al nivel de aplicación de origen; el mensaje retornado (el mensaje APDE de rechazo), deberá indicar que el mensaje fue rechazado debido a que el nodo destino no esta en capacidad de manejar mensajes seguros. El paquete devuelto contiene solamente encabezado APDE y no incluye un mensaje CAL. El nivel de aplicación origen debe entonces informar a la aplicación del dispositivo que la aplicación a realizarse falla en el nodo destino.

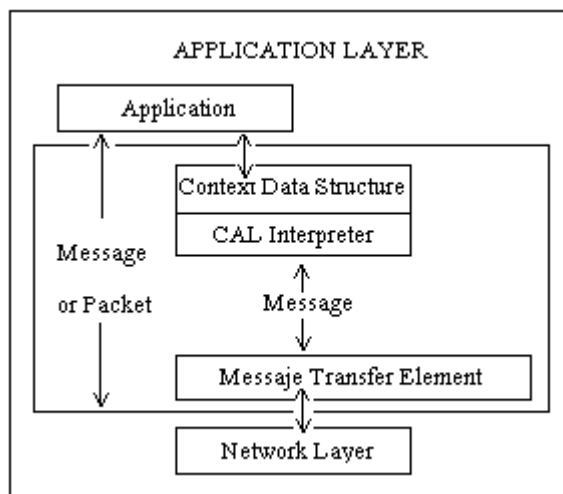
2.3.6.- EL NIVEL DE APLICACIÓN (NA)

El nivel superior del protocolo CEBus es el nivel de aplicación, este nivel ofrece sus servicios al proceso de aplicación por medio de un lenguaje denominado Lenguaje de Aplicaciones Comunes (Common Application Languages CAL). El CAL es un lenguaje de control completo que los dispositivos CEBus usan para comunicarse mutuamente, ubicar recursos disponibles en la red y realizar tareas de control.

La figura 2.4 Muestra el nivel de aplicación en forma detallada

Está formada por dos subpartes:

1. Elementos de Intérpretes CAL (CAL Interpreter Elements)
2. Elemento de Transferencia de Mensajes – Elemento MT (Message Transfer Element – MT Element).



La figura 2.4 Estructura del Nivel de Aplicación

El intérprete CAL recibe mensajes los analiza en la estructura de datos CONTEX DATA STRUCTURE, también genera un mensaje cuando la estructura de datos necesita reportar una condición determinada, por su parte en el elemento MT genera la llegada de datos del protocolo de aplicación APDU (Application Protocol Data Unit), además recibe los requerimientos de transferencia entre mensajes ya sea del intérprete CAL o directamente del proceso de aplicación, una vez que genera el APDU lo pasa al nivel de red junto a una

solicitud de los servicios que se requieren por parte del nivel de red y los niveles de protocolo. El elemento MT también recibe APDU's por parte del nivel de red, analiza el encabezado APDU del mensaje, ejecuta la tarea requerida en el encabezado y pasa el mensaje ya sea al intérprete CAL o al proceso de aplicación, dependiendo del tipo de mensaje recibido y de que el elemento originó el mensaje en el nodo origen. El elemento MT es el responsable de los servicios primarios característicos del CEBus.

- Servicio de reconocimiento extremo a extremo
- Servicio de autenticación.

2.3.6.1.- El Encabezado APDU

El elemento MT origen forma dicho encabezado para que sea utilizado por el elemento MT destino. Existen 2 tipos de encabezado:

- **Encabezado de único byte**, ésta es la forma más común de APDU y es utilizado para aquellos mensajes que no requieren mantener cierta seguridad.
- **Encabezado APDU múltiple, multi-byte**, el cual es utilizado únicamente para transmisión de mensajes seguros. Esto lo logra empleado un proceso de autenticación de mensajes con interpretación opcional.

2.3.6.2.- APDU's de Rechazo

Un APDU de rechazo es utilizado por el elemento MT de un nodo destino, cualquier momento que el elemento MT no sea capaz de procesar un APDU recibido. El APDU de rechazo resultante contiene un código de un byte en la posición del mensaje CAL, para indicar la razón por la cual ha sido rechazado. Los códigos son como se encuentran a continuación:

- El número de APDU no se soporta 30 HEX
- Servicios extendidos APDU no se soporta 31 HEX
- El nivel de aplicación se encuentra ocupado 32 HEX
- La autenticación de fallo 33 HEX

El tercer código, nivel de aplicación ocupado, generalmente es resultado de que el elemento MT intérprete CAL se encuentra procesando un APDU previo.

2.3.6.3.- Servicios Básicos de Entrega del Elemento MT

El elemento MT ofrece diversos servicios de entrega al intérprete CAL o al proceso de aplicación, los servicios aseguran una entrega de mensajes confiable al requerir que el nodo destino o nodos generen un mensaje de respuesta al nodo origen. Hay 4 servicios MT básicos:

- | | |
|---------------------------|---------------------|
| 1. Invocación Implícita | Implicit Invoque |
| 2. Invocación Explícita | Explicit Invoque |
| 3. Invocación Condicional | Conditional Invoque |

- **Invocación Implícita.-** Un mensaje es transmitido pero no existe petición de respuesta.
- **Invocación Explícita.-** Este servicio requiere de un reconocimiento extremo a extremo por parte del elemento destino. Es utilizado para asegurar que el nodo destino reciba el mensaje y que el resultado del mensaje sea devuelto.
- **Invocación Condicional.-** Este tipo requiere una respuesta sí y solo sí el mensaje transmitido produce un resultado positivo en el intérprete CAL de destino.
- **Retroceso Explícito.-** Este servicio es el mismo que el de Invocación Explícita con la adición de un servicio automático de reintento ejecutado en el MT. Cuando una respuesta APDU es recibida en un espacio de tiempo generalmente de 1.5 seg. el MT retransmite el APDU por segunda vez, los APDU recientes son devueltos como resultado de utilizar los servicios que abajo se listan.

Resultado.- Es el tipo de APDU de un mensaje devuelto o retornado, el cual contiene un resultado generado por un reconocimiento explícito o condicional.

Rechazo.- Es el tipo de APDU devuelto o retornado cuando el MT en el nodo destino se encuentra en una condición de error, y no puede procesar el mensaje.

Receipt – ACK.- Es el tipo de APDU devuelto o retornado cuando esta utilizando el servicio Invocación Explícita y el nodo destino se demora en responder.

2.3.7.- NIVEL DE RED (NR)

En la arquitectura del modelo de referencia OSI, el nivel de red que posteriormente se denominará NR es responsable de la gestión, encaminamiento y transmisión en la red además de la segmentación, secuenciamiento, detección y recuperación de errores y de las conexiones de red.

El NR del CEBus realiza todas estas funciones excepto la de segmentación y conexión de red; ya que el CEBus utiliza servicios no orientados a conexión. En este nivel el NR en cada nodo se comunica con el NR residente en otros nodos.

Routers y Brouters.- El NR intercambia información tanto con su nivel superior (nivel de aplicación), como con su inferior (nivel de enlace de

datos); las funciones del NR para la transmisión y recepción de mensajes se lo representa en la tabla 2.1. El elemento NT realiza peticiones al NR para la transmisión de APDU's utilizando determinados servicios de este nivel. El NR luego origina un encabezado Network Protocol Data Unit (NPDU) apropiado y de esta manera forma un NPDU para ser transmitido; posteriormente pasa el NPDU al nivel de enlace de datos, con una petición de aquellos servicios del nivel de enlace de datos que requiere.

ELEMENTO	TAREAS DE ENVIO	TAREAS DE RECIBO
RED	<ul style="list-style-type: none"> • Genera NPDU • Genera paquete ID • Servicio opcional de segmentación • Flujo de control opcional 	<ul style="list-style-type: none"> • Analiza NPDU • Detecta requerimientos de paquetes ID • Servicio opcional de segmentación • Flujo de control opcional • Rechazo de paquetes IR / RF

La Tabla 2.1 Responsabilidad del Nivel de Red

Cuando el nivel de enlace de datos recibe un NPDU el encabezado NPDU es analizado y los servicios requeridos del NR son ejecutados si es posible, el APDU es luego transmitido al elemento MT.

2.3.7.1.- Encabezado NPDU

El NR origina o genera el encabezado NPDU, el cual posee una longitud de 1 a 8 o más bytes; para luego comunicarse con el NR destino. El NPDU contiene los servicios del NR requeridos por el NR origen.

2.3.8.- NIVEL DE ENLACE DE DATOS

Es el responsable de la transmisión y recepción confiable de los paquetes a través del medio de transmisión especificado. El nivel de enlace de datos en cada nodo se comunica con el nivel de enlace de datos en todos los otros nodos del medio de transmisión. Este nivel realiza las siguientes funciones:

- Constituye y analiza el paquete completo incluyendo del DLPDU (Data Link Protocol Data Unit)
- Ejecuta el protocolo de acceso a canales
- Provee un servicio de reconocimiento inmediato de aquellos paquetes entregados en el medio de transmisión
- El nivel de red realiza peticiones de transmisión de NPDU's al nivel de enlace de datos, utilizando los servicios requeridos de este nivel

Cuando un NPDU es pasado desde el nivel de red, el nivel de enlace de datos realiza lo siguiente:

- Construye el DLPDU completo
- Logra el acceso al canal utilizando el protocolo de acceso al canal
- Transmite el paquete en orden consecutivo, un símbolo a la vez
- Espera por cualquier contestación requerida de recepción de paquetes
- Realiza una retransmisión del paquete si es necesario

ELEMENTO	TRAREAS DE ENVIO	TAREAS DE RECIBO
ENLACE DE DATOS	<ul style="list-style-type: none"> • Genera DLPDU • Genera preámbulo • Supresión de ceros de relleno • Protocolo de acceso al canal • Transmisión de paquetes de reconocimiento. 	<ul style="list-style-type: none"> • Recepción de paquetes • Filtro fragmentación de paquetes • Detección de errores • Analizar DLPDU • Reconocimiento de direcciones • Rechazo de paquetes duplicados

La Tabla 2.2 Responsabilidad del Nivel de Enlace

Cuando un paquete completo es recibido el nivel de enlace de datos realiza lo siguiente:

- Se analiza el paquete para ver si está completo y si carece de errores de recepción
- Se compara la dirección de destino del paquete con el nodo del nivel de enlace de datos y direcciones de grupo
- Si existe una coincidencia el NPDU es analizado y transmitido al nivel de red
- Si se requiere de una contestación se genera y transmite un paquete de reconocimiento

El nivel de enlace de datos es el más crítico en lo que a tiempo se refiere. Este opera en tiempo real, para mantenerse al corriente de la recepción y transmisión de paquetes.

2.3.9.- NIVEL FÍSICO

El nivel físico es el responsable de la transmisión y recepción de símbolos (1,0, EOF, EOP) el paquete construido por el nivel de enlace de datos debe ser transmitido serialmente un símbolo a la vez a través del medio de comunicación. Por otro lado los paquetes que llegan son recibidos un símbolo a la vez y transmitidos al nivel de enlace de datos para ser reconstruidos, además deben notificar al nivel de enlace cuando ha recibido todos los símbolos. El nivel físico del CEBus realiza las siguientes funciones:

1. Proveer cronometraje de los símbolos (symbol timing)

2. Detectar los símbolos válidos en la recepción
3. Para el medio de transmisión PL (Power Line) y RF (Radiate Frequency) añade un CRC (Critic Redondace Cycle) de 16 bits al final del paquete en la transmisión y verifica el CRC en la recepción. Este mantiene informado al nivel de enlace de datos, del estado actual del medio de transmisión para futuros accesos al mismo

Las responsabilidades individuales para las dos subcapas se resumen en la tabla 2.3, y la sincronización de símbolos se encuentra en la tabla 2.4

ELEMENTO	TRAREAS DE ENVIO	TAREAS DE RECIBO
SUBNIVEL DE CODIFICACIÓN DE SÍMBOLOS	<ul style="list-style-type: none"> • Codificación de símbolos • Cronometrage de símbolos • Generación de CRC (PL, RF) 	<ul style="list-style-type: none"> • Codificación de símbolos • Detección de errores • Detección del estado del canal
SUBNIVEL DEPENDIENTE DEL MEDIO	<ul style="list-style-type: none"> • Generación del estado • Superior / Inferios • Interfaz de medio 	<ul style="list-style-type: none"> • Detección del estado • Superior / Inferior • Interfaz de medio

La Tabla 2.3 Responsabilidades del Nivel de Red

SÍMBOLO	TIEMPO DE TRANSMISIÓN	TIEMPO DE RECEPCIÓN
1	100 microsegundos	90 a < 150 microsegundos

0	200 microsegundos	150 a < 250 microsegundos
EOF	300 microsegundos	250 a < 350 microsegundos
EOP	400 microsegundos	350 a < 450 microsegundos

La Tabla 2.4 Sincronización de los símbolos

2.4.- LENGUAJE DE APLICACIONES COMUNES (CAL)

2.4.1.- ESPECIFICACIONES DEL CAL

CAL es el lenguaje que utilizan los dispositivos CEBus para comunicarse. Es un lenguaje orientado a comandos que permite controlar dispositivos CEBus y asignar recursos. El lenguaje es un elemento de la capa de aplicación.

Las Funciones de asignación de recursos permiten pedir, usar y liberar recursos CEBus. Las funciones de control proporcionan la capacidad de enviar comandos CAL a dispositivos remotos, y responder a comandos CAL.

CAL utiliza el paradigma de programación orientada a objetos. Cuando un objeto recibe un mensaje se ejecuta alguno de los métodos disponibles. Un mensaje consiste en un identificador de método seguido de cero o

más parámetros. Cuando se recibe el mensaje, se busca en la lista de métodos cuál es el que tiene el identificador y si se encuentra se ejecuta. Ejemplo: si se quiere subir el volumen de la radio en tres unidades, habrá que mandar un mensaje al objeto que controla la radio en cuestión en el que se invoque el método de subir volumen.

2.4.2.- OBJETIVOS DEL CAL

Los requisitos de diseño del CAL fueron determinados por CEBus.

- Debe ser común entre todos los dispositivos de la residencia.- La fortaleza del CAL es su habilidad para modelar y controlar cualquier dispositivo que se encuentre en la residencia a través de un conjunto común de estructuras de datos. Estas estructuras de datos predefinidas aseguran la interoperabilidad del nivel de aplicación.
- Debe permitir la adquisición de las funciones de datos y control.- Los dispositivos deben ser capaces de ejecutar funciones de control y recibir datos de otros dispositivos, ya sea activamente (preguntando por los datos) o pasivamente (al proveer el destino para la difusión de los datos de la red)
- Debe permitir la administración de la red y de las funciones.- Todas las funciones que se relacionen con la administración de las residencias

requeridas por un nodo deberán ser ejecutadas por el lenguaje tales como: adquirir una dirección, asignar recursos, determinar que dispositivos se encuentran en la red, configurar otros nodos, etc. Pueden ser ejecutados por cualquier nodo que se encuentre utilizando CAL.

- Debe ser utilizable y extendible.- La independencia de la operación de los contactos permite a CAL adaptarse fácilmente a productos futuros en la residencia, sin tener que proveer cada nuevo producto imaginable. Si es que nuevos productos requieren de un nuevo contexto por ejemplo un lava vajillas, un nuevo contexto puede ser creado.
- Debe ser adaptable.- Un fabricante debe ser capaz de manejar funciones únicas de producto que no se encuentran disponibles en las especificaciones publicadas de CAL.

2.4.3.- MODELACIÓN DE LOS PRODUCTOS DEL CONSUMIDOR CON CAL

CAL trata cada producto como una colección de una o más partes comunes denominadas contextos; estos son diseñados para permitir accesos a las funciones de productos de una forma uniforme. Están diseñados para trabajar juntos en la red CEBus y para implementar interoperabilidad en el nivel de aplicación.

2.4.4.- ESTRUCTURA DE DATOS DEL CONTEXTO

La estructura de datos del contexto del CAL es un modelo de software en cada dispositivo CEBus, el cual modela la operación de todas las funciones disponibles en el dispositivo.

La estructura de datos del contexto y objeto en un producto es designada como una estructura real. La porción de mensajes de los paquetes recibidos es dirigida a un objeto particular en un contexto particular, cada contexto se encuentra especificado por su número de clase. Los objetos son involucrados con el contexto por su número y por su clase, porque existen típicamente muchas copias de la misma clase con cada contexto. Cada producto CEBus contiene el contexto universal 00, este contiene la información general del producto, su dirección y otro tipo de información general, todos los demás contextos son opciones y dependen de las funciones que el producto pueda soportar.

2.4.4.1.- Contextos

Un contexto define una subunidad subfuncional de un determinado producto y aplicación puede ser definida y permanece siempre igual sin importar donde va a ser utilizada. Un ejemplo típico del modelo de contexto puede ser encontrado en un TV; la mayoría de televisores poseen un reloj, un amplificador de sonido, un sincronizador y también sonido ambiental. Los televisores de todos los nodos están hechos de una o más funciones de audio / video como lo es un sincronizador, despliegue de pantalla, amplificador de sonido, etc. Cada función individual puede ser bien definida por lo cual cualquier TV

CEBus puede ser modelado como una colección de contextos en una dirección de nodo.

Dependiendo de las características del modelo, un televisor CEBus puede contener un contexto de despliegue de pantalla, un contexto de amplificación de sonido, un contexto de sincronización, un contexto de reloj, etc.

CAL define más de 60 contextos diferentes desde iluminación, seguridad, calefacción, aire acondicionado, hasta lavado y secado. Cada contexto independientemente de qué producto se trate opera de la misma manera. El amplificador de sonido de un televisor estéreo, micrófono de teléfono e intercomunicador operan de la misma manera.

2.4.4.2.- Objetos

Cada contexto consiste de uno o más objetos, un objeto representa un modelo de software de una función de control de un contexto. En los objetos modelan tareas ejecutadas por usuarios para controlar un producto, para encender o apagar un foco se utiliza un switch de dos posiciones: prendido / apagado, definidos en CEBus como un objeto switch binario (binary object switch) para ajustar el volumen de un estéreo o para aumentar o disminuir la temperatura de un termostato; generalmente se utiliza un control, botón definido en CEBus como un objeto análogo de control (analog control object). Todos los controles análogos son similares, estos poseen valores mínimos y máximos y

pueden asumir cualquier valor intermedio. Otros controles que se pueden encontrar en los productos de consumidor incluyen switch con posiciones múltiples como puede ser el caso de un switch que seleccione el origen del audio en el receptor del estéreo, paneles de control, etc.

En adición a los objetos los sensores proporcionan información sobre la función de un producto como son: temperatura, posición, nivel o tiempo.

CAL tiene un total de 25 modelos de objetos predefinidos, disponibles para ser utilizados en la generación de contexto.

2.4.5.- INTERPRETE CAL

El intérprete CAL es un elemento del nivel de aplicación. El intérprete ejecuta las instrucciones contenidas en los mensajes entrantes y genera cualquier mensaje requerido por la condición de reportes del objeto.

El intérprete CAL debe manejar las siguientes tareas mínimas:

- Analizar los mensajes entrantes
- Analizar la conversión numérica en los números de los mensajes
- Manejar los requerimientos de asignación de recursos
- Manejar la configuración de direcciones

- Interpretar las condiciones de reporte y generar mensajes de reporte
- Interfase con el objeto de aplicación
- Detectar condiciones de error de mensajes y enviar mensajes de error.

III.- ELEMENTOS X10 PARA IMPLEMENTACIÓN DE SISTEMAS DOMESTICOS⁴

3.1.- SISTEMAS DE SEGURIDAD

3.1.1.- SISTEMA BÁSICO: Costo estimado de material \$100 a \$200

Emplea productos basados en la línea de alimentación eléctrica, algunos de estos componentes son expandidos a otros sistemas de automatización. Básicamente se trata de subsistemas de bajo nivel. El controlador es un dispositivo de inteligencia distribuida. Controla hasta 16 zonas de seguridad; los sensores incluyen detectores metálicos que se comunican con otros módulos a través de la línea de alimentación eléctrica y tiene la posibilidad de interactuar con un marcador telefónico. Se puede agregar otros dispositivos tales como sensores de movimiento.

Lista de material estándar

- Un kit de seguridad de 4 piezas

⁴ <http://www.escproducts.com>. Marzo 2002

- Un receptor base con alarma incluido que trabaja como controlador
- Un sensor inalámbrico ventana / puerta con controles magnéticos
- Un control remoto para activar / desactivar
- Un módulo de lámpara

3.1.2.- SISTEMA BÁSICO EXPANDIDO: Costo estimado de material \$200 a \$500

- Inicialmente este sistema se lo puede desarrollar, partiendo del sistema básico anteriormente estudiado.
- Consiste de los componentes básicos previamente definidos y un controlador computarizado capaz de modificar el comportamiento del sistema básico. El controlador puede controlar otras luces y dispositivos.

Lista de material estándar

- Un kit de seguridad de 4 piezas
- Un receptor base con alarma incluido que trabaja como controlador
- Un sensor inalámbrico ventana / puerta con controles magnéticos
- Un control remoto para activar / desactivar
- Un módulo de lámpara
- El controlador computarizado, debe ser compatible con productos basados en la línea de alimentación eléctrica y estar en la capacidad de enviar y recibir señales de los módulos eléctricos y además y además soportar programación de instrucciones SI / ENTONCES.

- Sensor de movimiento y estación base, incluye un sensor de movimiento infrarrojo y un transmisor base que envía señales eléctricas a otros módulos, los sensores de movimiento que únicamente interactúan con el sistema de seguridad básico no son adecuados.
- Sirena remota, se la debe activar a través de la línea de alimentación eléctrica.

3.2.- SISTEMA DE ILUMINACIÓN DE ARTEFACTOS

3.2.1.- SISTEMA BÁSICO: Costo estimado de material \$28 a \$256

Emplea productos basados en la línea de alimentación eléctrica con el fin de proporcionar un sistema de automatización básico, incluye un controlador simple de 8 a 16 módulos y uno o más módulos de pared o módulos de lámpara artefacto; provee control de luces o artefactos eléctricos que se conectan a dichos módulos. Se pueden instalar interruptores de luz, controladores de módulos, sensores de interfases y controles remotos adicionales que permiten expandir el sistema según las necesidades.

Lista de material estándar

- Un minicontrolador, soporta hasta 8 módulos en el mismo código de casa.
- Un maxicontrolador, soporta hasta 16 módulos en el mismo código de casa.

- Uno o más módulos de pared, se los instala en el lugar de los interruptores de pared estándar corresponden a instrucciones / señales que reciben a través de la línea de alimentación eléctrica por parte de los dispositivos antes mencionados
- Uno o más módulos de lámpara / artefacto. Estos dispositivos se conectan a una toma de pared estándar que recibe señales de instrucciones de los dispositivos antes mencionados.

3.2.2.- SISTEMA BÁSICO EXPANDIDO: Costo estimado de material \$95 a \$400

Combina el controlador remoto inalámbrico y un controlador temporizador con los interruptores eléctricos estándar y módulos de lámpara / artefacto. El controlador remoto opera hasta 8 nodos con funciones encender / apagar y atenuar; y trabajan desde cualquier ubicación en la residencia y en áreas externas cercanas, lo que permite al usuario caminar con el control en sus manos. El controlador temporizador le permite trabajar con operaciones programadas a determinadas horas.

Lista de material estándar

- Un Sistema de control remoto inalámbrico incluye un controlador remoto y receptor, debe estar diseñado para realizar operaciones de control basado en la línea de alimentación eléctrica.

- Un Minitemporizador, es un dispositivo de control basado en una línea de alimentación eléctrica controla hasta 4 módulos de luces / artefactos y permite un control directo para ocho módulos en total.
- Uno o más módulos de pared, se los instala en el lugar de los interruptores de pared estándar corresponden a instrucciones / señales que reciben a través de la línea de alimentación eléctrica por parte de los dispositivos antes mencionados.

3.3.- SISTEMA DE AUDIO / VIDEO

3.3.1.- SISTEMA BÁSICO: Costo estimado de material \$100 a \$300

Se constituye como un sistema básico que emplea productos basados en la línea de alimentación eléctrica, incluye un controlador simple de módulos (8 a 16), y módulos de interfase. Para controlar los sistemas de audio provee conmutación de audio / video entre dos habitaciones; el sistema se puede expandir al repetir la instalación en otras habitaciones como también adicionando controles remotos y equipos adicionales. Se asume que ya se cuenta con televisores, VCR's, parlantes, etc. De lo contrario elevaría el costo.

Lista de material estándar

- Un Controlador MIDI (Musical Instrument Digital Interface), soporta hasta 6 módulos en el mismo código de casa.

- Dos Módulos receptores, se los debe de instalar en el lugar de los módulos.
- Cuatro Módulos de interfase, este módulo permite conmutar la salida de audio entre dos conjuntos de parlantes.
- Dos tomas de parlantes y cable coaxial, hardware , divisores y cables. No es necesario si los parlantes del equipo se conectan directamente, sin la necesidad de instalar tomas de pared adicionales.

3.3.2.- SISTEMA BÁSICO EXPANDIDO: Costo estimado de material \$450 a \$660

Se basa en el sistema de línea de alimentación eléctrica, previamente definida para un sistema de control remoto inalámbrico y un controlador temporizador. Controla también sistemas de automatización adicionales, el control remoto inalámbrico se debe adquirir con una estación base de control; esta pieza puede entonces definirse para controlar las funciones de comunicación y conmutación del equipo de audio y video, desde cualquier habitación en la residencia, a través del sistema eléctrico, el controlador temporizador permite controlar el equipo a horas determinadas.

Lista de material estándar

- Un Controlador temporizador, controla los módulos basados en la línea de alimentación eléctrica y horas determinadas.

- Un Remoto inalámbrico con receptor, provee control remoto para ejecutar funciones encender / apagar en cualquier habitación.
- Dos Módulos receptores, se los debe instalar en lugar de los módulos receptores estándar; son controlados por los dispositivos antes mencionados. Si se emplea para encender o pagar el equipo de sonido
- Cuatro Módulos de interfase, estos módulos permiten conmutar la salida de audio entre dos conjuntos de parlantes.
- Dos Tomas de parlantes, cable coaxial , hardware y divisores. No es necesario si los parlantes y equipos se conectan directamente sin la necesidad de instalar tomas de pared adicionales.

3.4.- SISTEMA TELEFÓNICO Y DE COMUNICACIONES.

3.4.1.- SISTEMA BÁSICO: Costo estimado de material \$90 a \$ 150

Incluye un contestador telefónico que responde llamadas entrantes y controla módulos basados en la línea de alimentación eléctrica se asume que tiene instalado una línea telefónica. El contestador trabaja únicamente con teléfonos de tono y se lo debe incorporar al mismo circuito de la línea entrante se asume además que ya se cuenta con un teléfono para las comunicaciones de voz, de lo contrario se elevará el costo estimado. Este sistema no está diseñado para soportar equipo adicional de bajo voltaje como pueden ser computadores termostatos, sensores etc. Se los puede incorporar

como una extensión a este sistema añadiendo cableado adicional y el hardware necesario.

Lista de material estándar

- Un contestador telefónico, se lo instala en un conector telefónico y eléctrica, debe contestar automáticamente las llamadas entrantes y enviar señales / instrucciones a través de la línea de alimentación eléctrica.
- Un módulo de lámpara artefacto, recibe señales / instrucciones del contestador telefónico a una toma o ubicación remota, las tomas se colocan en el mismo circuito que soporta la línea telefónica.

3.4.2.- SISTEMA BÁSICO EXPANDIDO: Costo estimado de material \$160 a \$ 450

Emplea los componentes básicos anteriormente definidos e incorpora conexión a una segunda habitación incluye teléfonos que incorporan funciones / intercomunicador, como también auto marcado. Al instalar este sistema en dos habitaciones, permite realizar llamadas marcadas automáticamente, proporciona respuestas automáticas a llamadas entrantes, control de equipos en cualquier habitación y comunicación habitación a habitación.

Lista de material estándar

- Dos teléfonos, los que deben ser teléfonos de parlante, que realicen las funciones propias de un intercomunicador, cuenta con características de marcador automático
- Un contestador telefónico, se lo instala en un conector telefónico y eléctrico, contesta automáticamente las llamadas entrantes y envía señales / instrucciones de un contestador telefónico, enciende o apaga los dispositivos conectados a este. Se puede utilizar módulos receptores o de pared.
- Dos divisores o toma telefónica, permite conectar el contestador telefónico en una forma o ubicación remota, las formas se conectan en el mismo circuito que soportan la línea telefónica.

3.5.- SISTEMA DE AUTOMATIZACIÓN AMBIENTAL

3.5.1.- SISTEMA BÁSICO: Costo estimado de material \$28 a \$ 256

Emplea productos basados en la línea de alimentación eléctrica con el fin de automatizar un sistema ambiental con un controlador de módulo simple, 8 0 16 módulos de artefacto o interfase. Controla los dispositivos conectado de los módulos de una sola fase. Si se desea expandir el sistema es posible instalar controladores adicionales, módulos, sensores de interfase, controladores remotos, etc.

Lista de material estándar

- Un controlador MIDI (Musical Instrument Digital Interface), soporta hasta 8 módulos en el mismo código de casa, o un controlador maxi soporta hasta 16 módulos en el mismo código / casa.
- Uno o más módulos de interfase o artefactos, se instala en una eléctrica estándar y se los conecta al equipo que se desea controlar. Son controlados por los dispositivos anteriormente mencionados, es posible utilizar módulos receptores en lugar de los módulos de artefacto.

3.5.2.- SISTEMA BÁSICO EXPANDIDO: Costo de material 495 a \$ 400

Emplea un controlador remoto inalámbrico y un controlador temporizador con los módulos de artefacto / interfase, el controlador remoto controla hasta 8 módulos con funciones encender / apagar y atenuar . Trabaja desde cualquier ubicación en la residencia. Un controlador temporizador, permite realizar ciertas operaciones de horas determinadas.

Lista de material estándar

- Sistema de control remoto inalámbrico, incluye un control remoto y receptor, está diseñado para enviar señales instrucciones a través de la línea de alimentación eléctrica.

- Un mini temporizador trabaja a través de la línea de alimentación eléctrica se lo programa para controlar hasta 4 módulos de luces / artefactos a horas determinadas y permite control directo (botone de control), de hasta 8 módulos en total.
- Uno o más módulos interfase o artefactos se instala en una toma eléctrica estándar y es posible utilizar módulos receptores, en lugar de los módulos artefactos.

3.5.3.- SISTEMA DE AUTOMATIZACIÓN MÁXIMA: Costo estimado de material \$1600 a \$4000

Incluye un controlador de avanzado nivel capaz de integrarse con los demás sistemas de la residencia, como puede ser: sistema de seguridad, sistema basados en la línea de alimentación eléctrica, etc. El controlador provee funciones encender / apagar o atenuar, además acepta la programación de instrucciones SI / ENTONCES y soporta un sistema de reconocimiento de voz.

Lista de material estándar

- Un controlador de automatización máxima, incluye un teclado a través del cual se realiza la programación, controles remotos y acepta múltiples operaciones de entrada y salida.
- Uno o más módulos de interfase a artefactos, se instala en una toma eléctrica estándar y se los conecta al equipo que se desea operar. Se los

controla por el dispositivo antes mencionado. Es posible utilizar módulos receptores en lugares de módulos / artefacto.

IV.- OPCIONES DEL NIVEL FISICO

4.1.- GENERALIDADES

En todo Sistema Domótico con arquitectura distribuida, los diferentes elementos de control deben intercambiar información unos con otros a través de un soporte físico (par trenzado, línea de potencia o red eléctrica, radio, infrarrojos, etc.), a esto se lo llama medio de transmisión. El uso de seis tipos de medios físicos en el CEBus y en general en cualquier sistema de Automatización Doméstica, tiene por objeto dar flexibilidad al sistema en cuanto a los reposos de frecuencia que cada medio podría soportar; las aplicaciones que se conecten a ellos, la magnitud que se quiera dar a la red y en último caso el precio que el usuario esté dispuesto a asumir.

Las especificaciones del medio, permiten a los fabricantes realizar dispositivos que podrán ser instalados en cualquier residencia; si el medio requerido ha sido instalado correctamente. Por ejemplo para utilizar la línea de alimentación eléctrica, un fabricante debe saber por lo menos sus características eléctricas y así sucesivamente.

Cada medio físico como son: línea de alimentación eléctrica (PL), cable coaxial (CX), cable par trenzado (TP), radio frecuencia (RF), rayos infrarrojos (IR) y fibra óptica (OF), en función de sus características particulares tienen ventajas y desventajas que a continuación se describen.

4.2.- EL MEDIO DE LÍNEA DE ALIMENTACIÓN ELÉCTRICA (PL)



Fig. 4.1 Medio de Línea de Alimentación Eléctrica.

Si bien no es el medio más adecuado para la transmisión de datos, si es una alternativa a tener en cuenta para las comunicaciones domésticas, dado que se trata de una instalación existente. Su implementación no exige requerimientos adicionales de cableado lo que implica ahorro de tiempo y de costo; esto ha sido una de las claves de éxito de los sistemas X10 durante la década pasada. En aquellos casos en los que las necesidades del sistema no

impongan requerimientos muy exigentes en cuanto a la velocidad de transmisión; la línea de distribución de energía eléctrica puede ser suficiente como soporte de dicha transmisión.

Dada las especiales características de este medio y sobretodo su idoneidad para las instalaciones domesticas, a continuación se detallan sus principales ventajas:

- Nulo coste de la instalación.
- Facilidad de conexionado.
- La ventaja principal del método de alimentación eléctrica (PL) radica en que es un medio al alcance de todos.

Por otro lado, el ahorro de recursos tiene como contra parte la inhospitabilidad de la red de alimentación, para el transporte de las señales de comunicación; ya que existe mucho ruido eléctrico, y variaciones de energía. Además existen dificultades que provienen de las características topológicas de la red de distribución de energía eléctrica así:

- Un transformador de distribución sirve generalmente a un conjunto de residencias, en consecuencia el cableado de alimentación es un medio compartido entre varias residencias; por tanto requiere de un esquema de direccionamiento que garantice la privacidad de la información y el libre tráfico en la red.

- La red de distribución está formada por varias fases, las características de este medio no garantizan la transferencia de información entre dispositivos conectados en las diferentes fases, por eso son imprescindibles acopladores de señal entre las fases de la red.
- Baja velocidad de transmisión.

4.3.- EL MEDIO DE CABLE DE PAR TRENZADO (TP)



Fig. 4.2 Medio de Cable Par Trenzado

El medio de par trenzado (TP), supera muchos de los problemas del medio de línea de alimentación y permite el intercambio de: datos, voz y audio; que ocupen en un ancho de banda de hasta 512 KHz.

4.3.1.- CARACTERÍSTICAS:

- Ofrece facilidad y bajo costo de instalación, su flexibilidad permite instalar por los bordes de las paredes, por techos y pisos falsos.
- El cable que se utiliza es el mismo que se emplea para la red telefónica.
- Permite suministrar alimentación DC limitada a equipos que requieren bajo voltaje.
- Es factible su coexistencia con la red telefónica.
- Si se configura a la red en forma de estrella, existe la ventaja de que fácilmente se pueda formar subsistemas que agrupen equipos con funcionalidades afines.
- Para el caso particular del CEBUS la especificación de bus de par trenzado TP-Bus permite con ciertas restricciones la reutilización de una red telefónica existente, de manera inversa. También el cableado del TP-Bus puede ser utilizado para la red telefónica, siempre y cuando se usen los equipos de distribución adecuados.

Los denominados cables de pares están formados por cualquier combinación de los tipos de conductores que a continuación se detallan:

- Cables, formados por un solo conductor con un aislamiento exterior plástico. (Por ejemplo los utilizados para la transmisión de las señales telefónicas).
- Par de cables, cada uno de los cables esta formado por un arrollamiento helicoidal de varios hilos de cobre. (Por ejemplo los utilizados para la distribución de señales de audio).

- Par apantallado, formado por dos hilos recubiertos por un trenzado conductor en forma de malla cuya misión consiste en aislar las señales que circulan por los cables de las interferencias electromagnéticas exteriores. (Por ejemplo los utilizados para la distribución de sonido alta fidelidad o datos).
- Par trenzado, esta formado por dos hilos de cobre recubiertos cada uno por un trenzado en forma de malla. El trenzado es un medio para hacer frente a las interferencias electromagnéticas. (Por ejemplo los utilizados para interconexión de ordenadores).

4.4.- EL MEDIO DE RAYOS INFRARROJOS (IR)

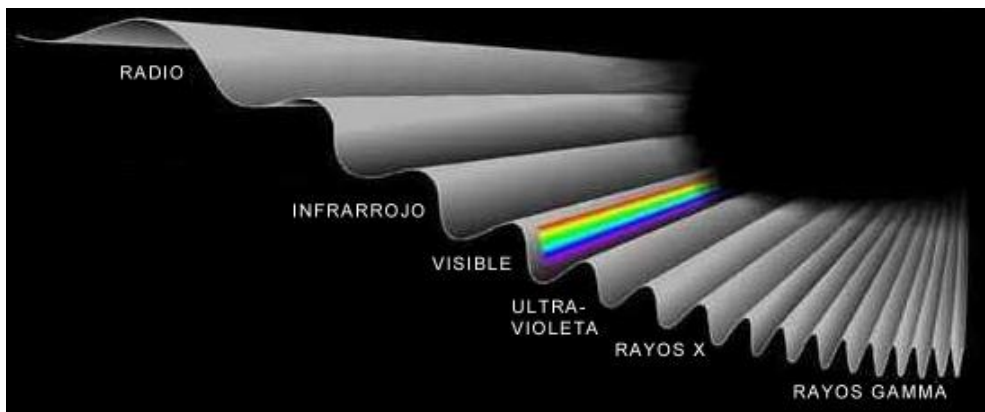


Fig. 4.3 Medio de Rayos Infrarrojos.

El uso de mandos a distancia basados en transmisión por infrarrojos (control remoto) esta ampliamente extendida en el mercado residencial para tele comandar equipos de Audio y Video.

La comunicación se realiza entre un diodo emisor que emite una luz en la banda de IR, sobre la que se superpone una señal, convenientemente modulada con la información de control, y un fotodiodo receptor cuya misión consiste en extraer de la señal recibida la información de control.

Los controladores de equipos domésticos basados en la transmisión de ondas en la banda de los infrarrojos tienen las siguientes ventajas:

- Comodidad y flexibilidad.
- Admiten gran número de aplicaciones.

Su principal limitación consiste en que es imprescindible que exista línea de vista entre el transmisor y el receptor, por lo que salvo por reflexiones accidentales no, se puede traspasar obstáculos; en consecuencia únicamente es posible usarlo en una habitación individual. Su limitación hace que no sea recomendable para la implementación de bus principal de datos en una residencia, sino como medio auxiliar para soporte de operaciones de control remoto a los dispositivos.

En caso de ser necesario extender su alcance a varias habitaciones, se requiere bridges y / o ruteadores, entre la habitación que se encuentra en control y la habitación del equipo que se controla, lo cual encarecería el sistema. En este caso existe una mejor opción que consiste en el uso del medio de radio frecuencia como se indica en la sección siguiente.

Al tratarse de un medio de transmisión óptico es inmune a las radiaciones electromagnéticas producidas por los equipos domésticos o por los demás medios de transmisión (coaxial, cables pares, red de distribución de energía eléctrica, etc.). Sin embargo, habrá que tomar precauciones en los siguientes casos:

- Las interferencias electromagnéticas sólo afectaran a los extremos del medio IR, es decir, a partir de los dispositivos opto electrónicos (diodo emisor y fotodiodo receptor).
- Es necesario tener en cuenta otras posibles fuentes de IR. Hoy en día, existen diferentes dispositivos de iluminación que emiten cierta radiación IR.

4.5.- EL MEDIO DE RADIO FRECUENCIA (RF)



Fig. 4.4 Medio de Radio Frecuencia.

El medio de radio frecuencia (RF) supera la dificultad fundamental del medio IR, ya que tiene la capacidad para traspasar obstáculos y

paredes, lo que permitiría implementar verdaderos controles remotos universales que podrían utilizarse en toda la residencia.

La introducción de las radio frecuencias, como soporte de transmisión en la residencia, ha venido precedida por la proliferación de los teléfonos inalámbricos y sencillos telemandos. Este medio de transmisión, puede parecer en principio idóneo para el control a distancia de los Sistemas Domóticos, dada la gran flexibilidad que supone su uso. Sin embargo resulta particularmente sensible a las perturbaciones electromagnéticas producidas, tanto por los medios de transmisión, como por los equipos domésticos.

Con respecto a este medio cabe hacer las siguientes observaciones:

- El medio de radio frecuencia puede emplearse como complemento al de rayos infrarrojos, puesto que este dispositivo RF podría controlar varios equipos IR distribuidos en diferentes habitaciones.
- Existen problemas de direccionamiento de instrucciones ya que en el medio RF es común para un entorno de casas vecinas, esta dificultad se puede resolver mediante el uso de direcciones de sistema que mantendrían la privacidad de cada sistema RF.
- Dificultad para la integración de las funciones de control y comunicación, en su modalidad de transmisión analógica.

4.6.- EL MEDIO DE CABLE COAXIAL (CX)

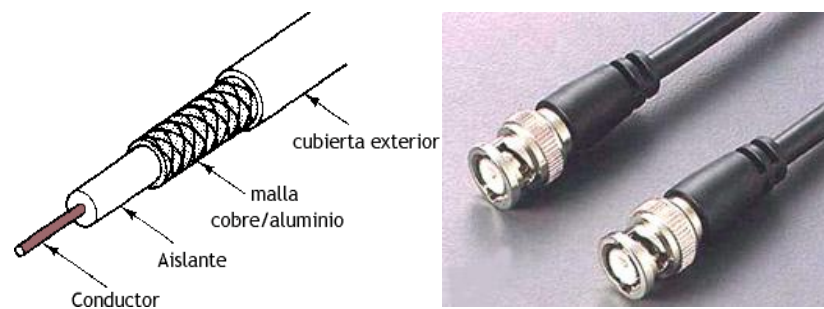


Fig. 4.5 Medio de Cable Coaxial.

Un cable coaxial es un circuito físico asimétrico, constituido por un conductor filiforme que ocupa el eje longitudinal del otro conductor en forma de tubo, manteniéndose la coaxialidad de ambos mediante un dieléctrico apropiado. El medio de cable coaxial (CX) tiene mayor capacidad de recursos de frecuencia (hasta 1GHz), que los otros medios indicados lo que permite soportar servicios de distribución de gran cantidad de datos inclusive de video, adecuadamente instalados presenta buenas características de aislamiento frente a interferencias externas.

Dentro del ámbito de la vivienda, el cable coaxial puede ser utilizado como soporte de transmisión para:

- Señales de teledifusión que provienen de las antenas (red de distribución de las señales de TV y FM).
- Señales procedentes de las redes de TV por cable.
- Señales de control y datos a media y baja velocidad.

Las desventajas de este medio provienen de las limitaciones intrínsecas del cable coaxial que en resumen son:

- Los sistemas de cable coaxial requieren de repetidores poco espaciados, cada 3000 a 7000 metros y de derivadores especiales para el caso de la conexión en red de varios dispositivos, lo que implica aumento en costos.
- El cable coaxial no es tan flexible comparado con otros medios, como el cable de par trenzado y el cable de fibra óptica; por tanto su instalación en un medio doméstico resulta más complicado.
- Si la instalación no se realiza adecuadamente existe susceptibilidad a interferencias electromagnéticas provenientes de fuentes de radio frecuencia, ordenadores personales, hornos microondas y aún de los mismos equipos terminales de la red de cable coaxial.

4.7.- EL MEDIO DE FIBRA ÓPTICA (FO)

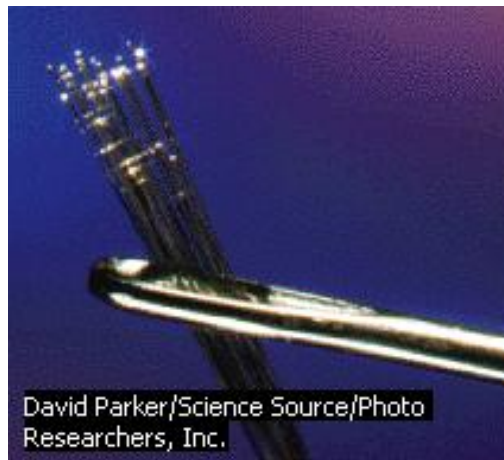


Fig. 4.6 Medio de Fibra Óptica.

La fibra óptica está constituida por un material dieléctrico transparente, conductor de luz, compuesto por un núcleo con un índice de refracción menor que el del revestimiento que envuelve a dicho núcleo. Estos dos elementos forman una guía para que la luz se desplace por la fibra. La luz transportada es generalmente infrarroja, y por lo tanto no es visible por el ojo humano.

El medio de fibra óptica (FO), supera muchas de las desventajas del método del cable coaxial, su incursión en los estándares de la Automatización Doméstica es plenamente justificado; considerando las perspectivas de este medio que llevarían a la integración total dentro de una resistencia y con el medio externo en donde los servicios basados en fibra, también evolucionan de una manera prometedora.

La fibra óptica, es el resultado de combinar dos disciplinas no relacionadas como son: la tecnología de semiconductores (que proporciona los materiales necesarios para las fuentes y los detectores de luz), y la tecnología de guiado de ondas ópticas (que proporciona el medio de transmisión (el cable de fibra óptica)).

A continuación se detallan sus ventajas:

- Las fibras tienen muchas ventajas ampliamente conocidas con respecto al cable coaxial así se tiene sus alcances mucho mayores en el orden de decenas de kilómetros.
- Gran ancho de banda.
- Capacidad para transmitir simultáneamente varias señales.
- Fiabilidad en la transferencia de datos.
- Inmunidad frente a interferencias electromagnéticas y de radiofrecuencias.
- Distancia entre los puntos de la instalación limitada, en el entorno doméstico estos problemas no existen.
- Transferencia de gran cantidad de datos

En cuanto a inconvenientes:

- Una desventaja práctica de la fibra se encuentra en la dificultad para realizar uniones (empalmes); sin embargo los progresos en el desarrollo de conectores

de plástico, cerámica y pegamento de resina tienden a minimizar este problema.

- Otra desventaja es el factor de costo, tanto de la fibra como de los componentes opto electrónicos, sin embargo, el advenimiento al mercado de los reproductores de discos óptimos han demostrado que el precio en las tecnologías de láser opto electrónicos pueden descender a niveles accesibles en el mercado; se estima que en un futuro cercano el factor de precios no será un factor determinante, sino que prevalecerán las ventajas técnico / prácticas de la fibra como: su alcance, capacidad, expansión, flexibilidad, etc.

Las especificaciones de los estándares de los servicios domésticos en fibra se encuentran en las primeras fases de definición de requerimientos y de consideraciones del diseño.

V.- CONSTRUCCIÓN DEL PROTOTIPO ELECTRÓNICO

5.1.- DESCRIPCIÓN

El sistema de control automático para el prototipo de una residencia está diseñado usando tecnología electrónica de tipo digital, y basado en un software especialmente escrito para dicha aplicación.

En el presente capítulo se detallan las fases de diseño y construcción del Hardware "Smart House Control", el mismo que se conectará al PC a través del puerto paralelo. Para fines de

mejor análisis y compresión, dividimos el diseño y construcción del hardware en dos tarjetas:

1. Placa de Potencia y Acondicionamiento de Señal para el Sensor de Temperatura
2. Placa de Control y Adquisición de Datos.

5.1.1.- PUERTO PARALELO⁵

El puerto paralelo es un dispositivo para el intercambio de datos digitales del PC con el exterior, siendo su uso generalizado en el manejo de la impresora, pero puede permitir otras aplicaciones conectando otros periféricos: comunicación entre PC's, lectura de sensores de alarmas, control sobre dispositivos de arranque / parada de motores, encendido / apagado de luces o sirenas, etc.

La interfaz paralelo está constituida por tres registros particulares de 8 bits cada uno, a los cuales se puede acceder en forma independiente:

- REGISTRO DE DATOS 378 (solo salida)
- REGISTRO DE ESTADO 379 (solo entrada)

⁵ Zelenousky Ricardo. "IBM-PC para Ingenieros".

➤ REGISTRO DE CONTROL 37A(bidireccional)

Dirección	Nº de bits	Entrada	Salida	Lectura	Escritura
378H	8		x	x	x
379H	5	x		x	
37A	4	x	x	x	x

Tabla 5.1 Registros del Puerto Paralelo

5.1.1.1.- Puerto de Salida de 8 Bits (378H)

La figura siguiente muestra cómo los registros están conectados a los pines del conector. El Registro de Datos tiene disponibles 8 bit de salida (OUT) .

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PINES	9	8	7	6	5	4	3	2

Tabla 5.2 Pines del Registro 378H

5.1.1.2.- Puerto de Entrada de 5 Bits (379H)

PINES	-	-	-	Hab IRQ7	17(L)	16(L)	14(L)	1(L)
RESET	-	-	-	0	1	0	1	1

Tabla 5.4 Pines del Registro 37A H

Desde el punto de vista físico se accede a los 3 registros del puerto paralelo a través de un conector tipo DB-25 hembra cuya identificación de pines es la siguiente:

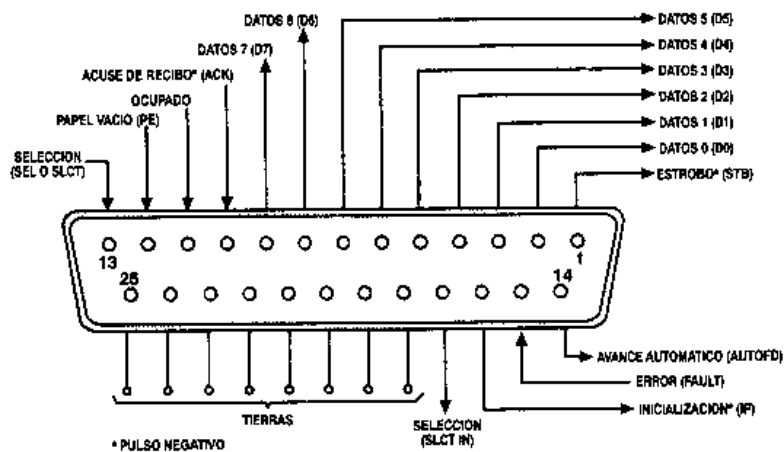
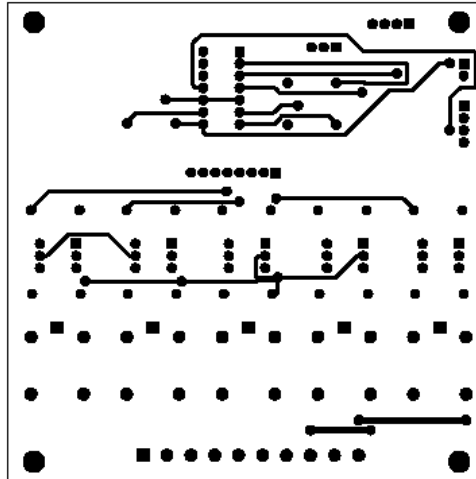


Fig. 5.1 Ficha DB25, funciones de los distintos pines

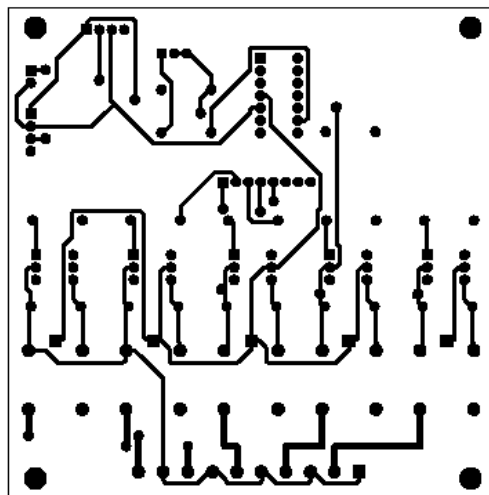
5.2.- DISEÑO DE LAS PLACAS DE CIRCUITO IMPRESO

5.2.1.- PLACA DE POTENCIA Y ACONDICIONAMIENTO DE SEÑAL DEL SENSOR DE TEMPERATURA

Lado superior

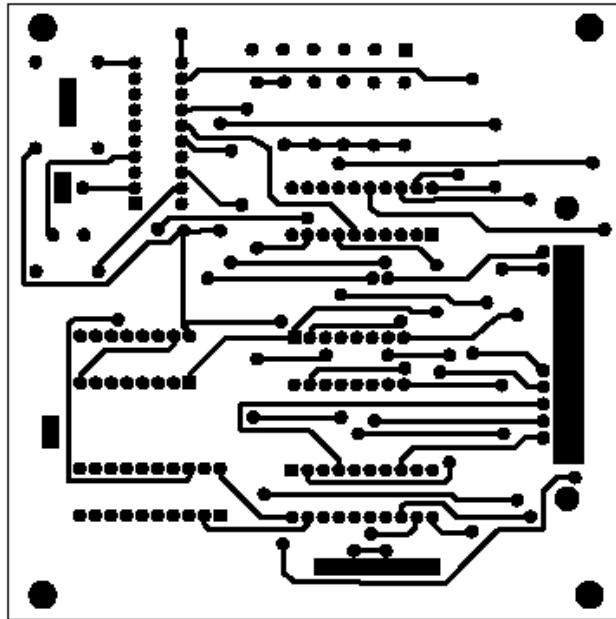


Lado inferior

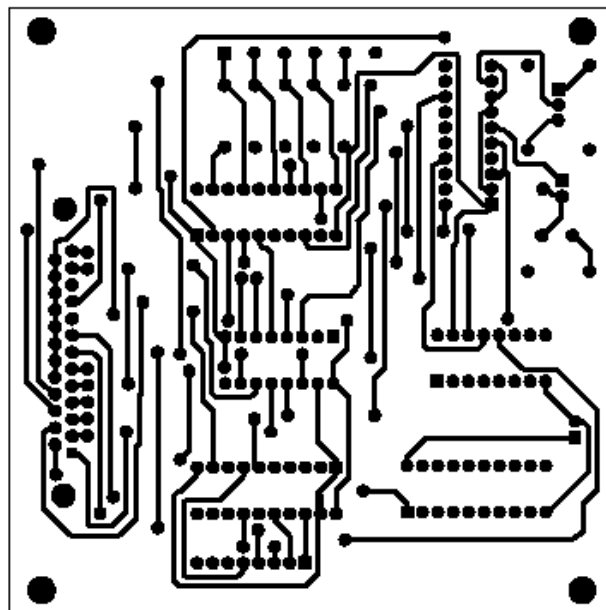


5.2.2.- PLACA DE CONTROL Y ADQUISICIÓN DE DATOS

Lado superior



Lado inferior



5.2.3.- ASIGNACIÓN DE PINES DEL PUERTO PARALELO PARA EL PROYECTO

A continuación se describen los pines del puerto paralelo, respectivamente con los dispositivos que cada uno controla en el prototipo.

No.	DESCRIPCIÓN	REGISTRO	PIN
1	Luz Habitación1 (LH1)		2
2	Luz Habitación2 (LH2)	3	3
3	Luz Habitación3 (LH3)	7	4
4	Luz Externa (LE)	8	5
5	Habilitador (=0 sensores)(=1 S Temperat)	H	6
6	Sirena		7
7	Pulso inicial de ADC(temporizador)		8
8	Puerta de Ingreso1(PI1) y Sensor Temperatura (b4)	3	15
9	Puerta de Ingreso2(PI2) y Sensor Temperatura (b5)	7	13
10	Sensor de Movimiento1 y Sensor Temperatura (b6)	9	12
11	Conversor del Sensor Temperatura y Sensor Temperatura (b7)	H	10
12	Sensor de Ventana1(SV1) y Sensor Temperatura (b0)	3	1L
13	Sensor de Ventana2(SV2) y Sensor Temperatura (b1)	7	14
14	Sensor Temperatura (b2)	A	16L
15	Sensor Temperatura (b3)	H	17L

Tabla 5.5 Placa de Potencia y Acondicionamiento de Señal del Sensor de Temperatura.

5.2.3.1.- Diseño del Circuito de Potencia.

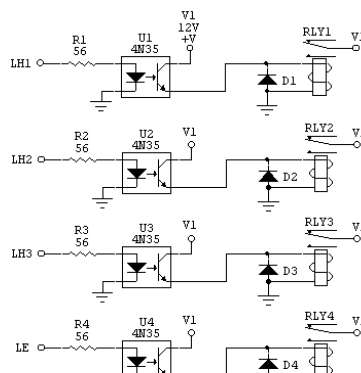


Fig. 5.2 Diagrama del Circuito de Potencia que Activa Luces y Sirena

Cálculos:

- Resistencia

$$R = \frac{V}{I}$$

$$R = \frac{5V - V_{ak}}{I}$$

$$R = \frac{5V - 1V}{50mA}$$

$R = 80\Omega$ Por tanto asumimos utilizar una resistencia de $R = 56\Omega$

- Potencia

$$P = I^2 * R$$

$$P = (50mA)^2 * 56\Omega$$

$P = 140mW$ Por tanto asumimos utilizar una potencia de $P = 500mW$

5.2.3.1.1.- Lista de Materiales.

Cant.	Nombre

5	Resistencia de 56 Ω . ¼ w.
5	Opto acopladores 4N35
5	Diodos 1N4004
5	Relés 12 v (contacto 10 ^a / 28 VDC)(contacto 10A / 220 VAC)
1	Potenciómetros de precisión 10K

Tabla 5.5 Lista de Materiales de Circuito de Potencia que Activa Luces y Sirena

Nota: Las luces y alarma funcionan con una Vcc de $\pm 12v$.

5.2.3.2.- Diseño del Circuito de Acondicionamiento de Señal del Sensor de Temperatura.

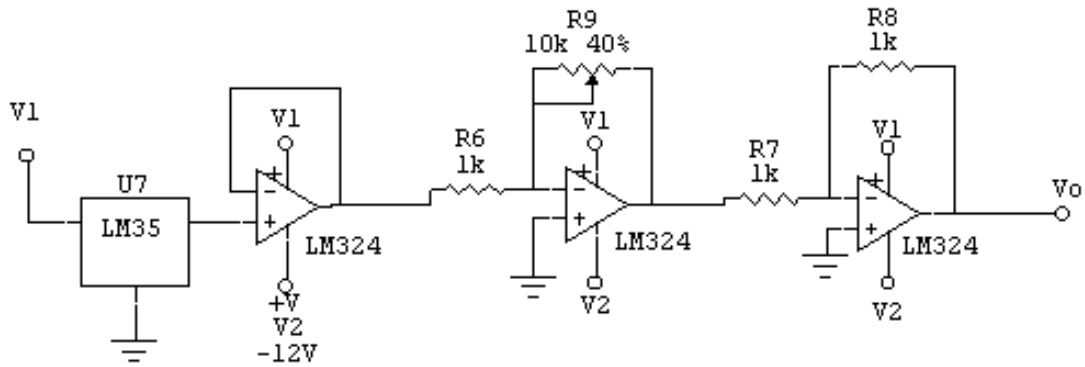


Fig. 5.3 Diagrama del Circuito de Acondicionamiento de Señal del Sensor de Temperatura

5.2.3.2.1.- Diseño para el Detector de Temperatura

- Se asume que la temperatura máxima a alcanzar el sensor LM35 será menor que 100 grados centígrados.

- $LM35 = \frac{10mV}{^{\circ}C}$
- El sensor normalmente alcanza los $150^{\circ}C$; pero por seguridad de los elementos asumimos como temperatura máxima $100^{\circ}C$

Cálculos:

$$G = \frac{R9}{R6}$$

$$V_o = \frac{-R9}{R6} * V_m$$

$$5 = \frac{-R9}{1K} * V_m$$

$$R9 = \frac{1K * 5V}{1V}$$

$R9 = 5K$ por tanto la ganancia establecida es de 5K

5.2.3.2.2.- Lista De Materiales.

Cant.	Nombre
1	Sensor de Temperatura LM35
1	Amplificador Operacional LM324
3	Resistencia de $1K\Omega$. $\frac{1}{4}$ w.
1	Potenciómetro de precisión 10K

Tabla 5.6 Lista de Materiales del Circuito de Acondicionamiento de Señal del Sensor de Temperatura.

5.2.4.- PLACA DE CONTROL Y ADQUISICIÓN DE DATOS

5.2.4.1.- Diseño del Circuito de Control y Adquisición de Datos (Ver Anexo 1)

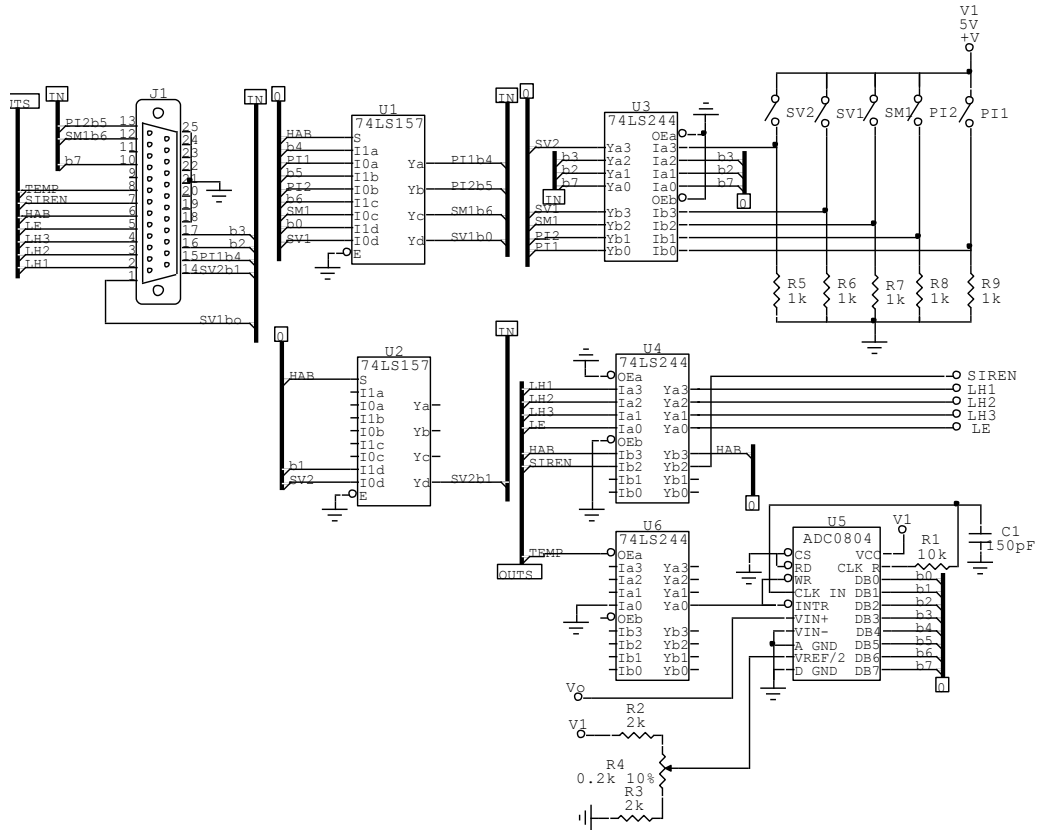


Fig. 5.4 Diagrama del Circuito de Control y Adquisición de Datos

5.2.4.1.1.- Lista de Materiales para Control

Cant.	Nombre
3	74LS244
2	74LS157
5	Resistencias de 1K. ¼ w
1	Conector DB25 para placa hembra

Tabla 5.7 Lista de Materiales del Circuito de Control

NOTA: Esta placa funciona con una Vcc de + 5v y tierra

5.2.4.1.2.-

Lista De Materiales Para Adquisición De Datos

Cant.	Nombre
1	Convertor Análogo/Digital ADC0804
2	Resistencias de $2K\Omega$. $\frac{1}{4}$ w.
1	Resistencia de $10K\Omega$. $\frac{1}{4}$ w.
1	Potenciómetro de 200Ω
1	Capacitor de 150Pf

Tabla 5.8 Lista de Materiales del Circuito de Adquisición de Datos

VI.- CONSTRUCCIÓN DEL SOTWARE

6.1.- METODOLOGÍA UML

6.1.1.- INTRODUCCIÓN

El Lenguaje de Modelamiento Unificado (UML - Unified Modeling Language) es un lenguaje gráfico para visualizar, especificar y documentar cada una de las partes que comprende el desarrollo de software. Su objetivo es entregar un material de apoyo, que le permita al lector poder definir diagramas propios, como también poder entender el modelamiento de diagramas ya existentes.

6.1.2.- ARQUITECTURA DEL UML⁶

Es una arquitectura de cuatro capas, definida a fin de cumplir con la especificación Meta Object Facility del OMG:

- Meta-metamodelo: Define el lenguaje para especificar metamodelos.
- Metamodelo: Define el lenguaje para especificar modelos.
- Modelo: Define el lenguaje para describir un dominio de información.
- Objetos de usuario: Define un dominio de información específico.

Según se indica en la Metodología OMT (Rumbaugh), los modelos permiten una mejor comunicación con el cliente por distintas razones:

⁶ <http://www.monografias.com/trabajos5/insof/insof.shtml>

- Es posible enseñar al cliente una posible aproximación de lo que será el producto final.
- Proporcionan una primera aproximación al problema que permite visualizar cómo quedará el resultado.
- Reducen la complejidad del original en subconjuntos que son fácilmente tratables por separado.

6.1.3.- MODELADO CON UML

El UML es una técnica de modelado de objetos y como tal supone una abstracción de un sistema para llegar a construirlo en términos concretos. El modelado no es más que la construcción de un modelo a partir de una especificación.

Un modelo es una abstracción de algo, que se elabora para comprender ese algo antes de construirlo. El modelo omite detalles que no resultan esenciales para la comprensión del original y por lo tanto facilita dicha comprensión. Utiliza métodos formales en el modelado, pero sus enfoques son diferentes, así:

- No formales: Usa un lenguaje natural
- Semi formales: Utiliza notaciones en parte gráficas, pero con ciertas reglas.
- Formales: Usa notación gráfica o textual basada en un sistema formal (matemático), que permite determinar con mayor rigor las propiedades del software.

El modelado con UML, está orientado a los diferentes diagramas con los que trabaja el lenguaje UML; a continuación detallamos cada uno de ellos.

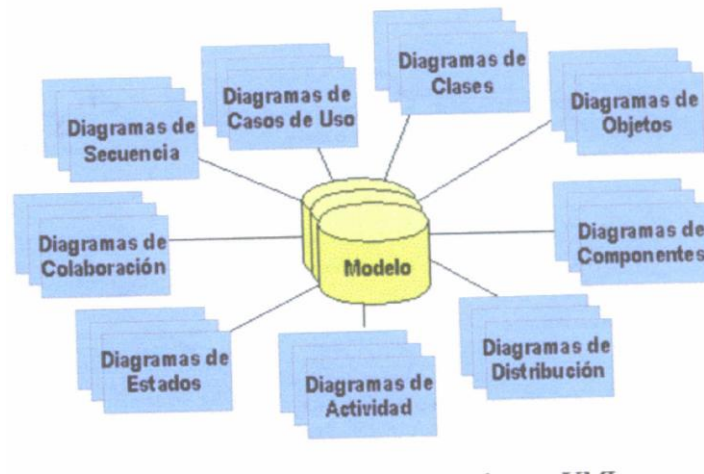


Fig. 6.1 Diagramas del Modelo UML

6.1.3.1.- Diagrama de Casos de Uso⁷

Un Diagrama de Casos de Uso muestra las distintas operaciones que se esperan de una aplicación o sistema y cómo se relaciona con su entorno (usuarios u otras aplicaciones). Se muestra como ilustración los casos de uso de la máquina de café.

⁷ <http://www.cs.ualberta.ca/~pfiguero/soo/uml/>

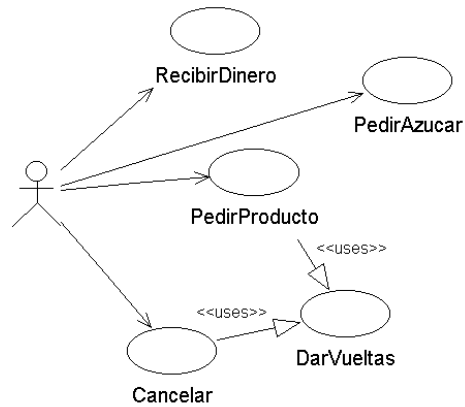



Fig. 6.2 Diagramas de Casos de Uso Ejemplo Maquina de café

Caso De Uso: Se representa en el diagrama por una elipse, denota un requerimiento solucionado por el sistema. Cada caso de uso es una operación completa desarrollada por los actores y por el sistema en un diálogo. El conjunto de casos de uso representa la totalidad de operaciones desarrolladas por el sistema. Va acompañado de un nombre significativo. En el caso del ejemplo se tienen como casos de uso de la cafetera RecibirDinero, PedirAzucar, PedirProducto, DarVueltas y Cancelar.

Actor: Es un usuario del sistema, que necesita o usa algunos de los casos de uso. Se representa mediante un , acompañado de un nombre significativo, si es necesario.

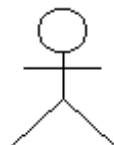


Fig. 6.3 Actor

6.1.3.2.- Diagrama de Secuencia

Un diagrama de secuencia muestra la interacción de un conjunto de objetos en una aplicación a través del tiempo. Esta descripción es importante porque puede dar detalle a los casos de uso, aclarándolos al nivel de mensajes de los objetos existentes, como también muestra el uso de los mensajes de las clases diseñadas en el contexto de una operación.

A continuación se muestra un ejemplo de diagrama de secuencia, que da detalle al caso de uso PedirProducto del ejemplo de la [cafetera](#).

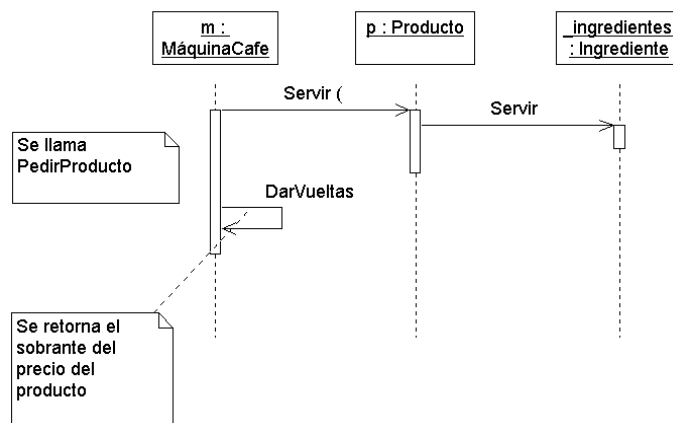


Fig. 6.4 Diagramas de Secuencia Ejemplo Maquina de café

6.1.3.3.- Diagrama De Colaboración

Un diagrama de colaboración es una forma de representar interacción entre objetos, alterna al diagrama de secuencia. A diferencia de los diagramas de secuencia, pueden mostrar el contexto de la operación (cuáles objetos son atributos, cuáles temporales) y ciclos en la ejecución. Se toma como ejemplo el caso de uso PedirProducto ya descrito como diagrama de secuencia.

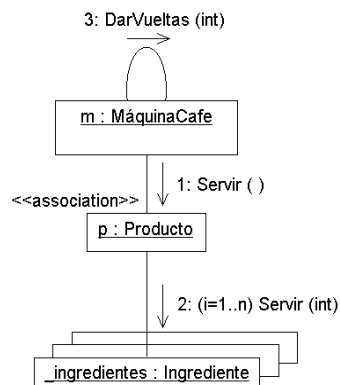


Fig. 6.5 Diagramas de Colaboración Ejemplo Maquina de café

- **Objeto:** Un objeto se representa con un rectángulo, que contiene el nombre y la clase del objeto en un formato **nombreObjeto: nombreClase.**
- **Enlaces:** Un enlace es una instancia de una asociación en un diagrama de clases. Se representa como una línea continua que une a dos objetos. Está acompañada por un número que indica el orden dentro de la interacción y por un estereotipo que indica que tipo de objeto recibe el mensaje.
- **Flujo de mensajes:** Expresa el envío de un mensaje. Se representa mediante una flecha dirigida cercana a un enlace.

6.1.3.4.- Diagrama de Estados

Muestra el conjunto de estados, por los cuales pasa un objeto durante su vida en una aplicación, junto con los cambios que permiten pasar de un estado a Un ejemplo en el caso de la cafetera son los estados posibles para la clase MaquinaCafe:

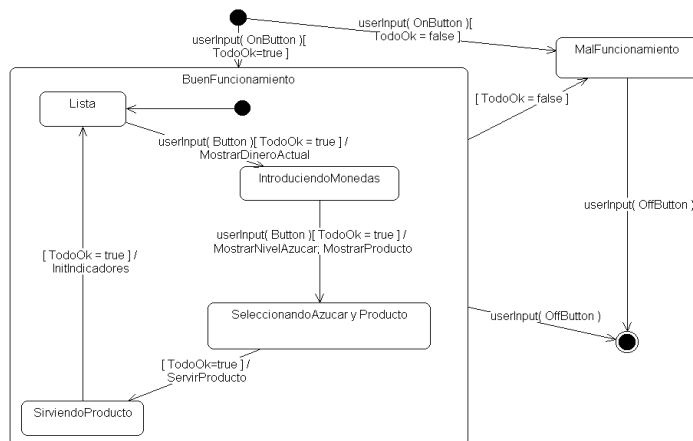



Fig. 6.6 Diagramas de Estados Ejemplo Maquina de café

- **Estado:** Identifica un periodo de tiempo del objeto (no instantáneo) en el cual el objeto esta esperando alguna operación, tiene cierto estado característico o puede recibir cierto tipo de estímulos. Se representa mediante un rectángulo con los bordes redondeados. En el caso del ejemplo anterior, se tienen cuatro estados (EnFuncionamiento, SinCambio, SinIngredientes, MalFuncionamiento) , en los cuales se desarrollan ciertas acciones al entrar; por ejemplo, al entrar al estado SinIngredientes se debe realizar la accion "Indicador SinIngredientes en On". Se marcan también los estados iniciales y finales mediante los símbolos ● y  , respectivamente.
- **Evento:** Es un suceso que puede causar la transición de un estado a otro de un objeto. Este suceso puede ser una de varias cosas:
 - Condición que toma el valor de verdadero o falso
 - Recepción de una señal de otro objeto en el modelo
 - Recepción de un mensaje
 - Paso de cierto período de tiempo, después de entrar al estado o de cierta hora y fecha particular

En el caso del ejemplo anterior se encuentra nombrado en varias transiciones el evento **userInput**, que recibe como parámetro un **Button** , para indicar el botón que ha sido presionado por el usuario de la máquina de café.

6.1.3.5.- Diagrama de Actividades

Un diagrama de actividades es un caso especial de un diagrama de estados, en el cual casi todos los estados son estados de acción (identifican que acción se ejecuta al estar en él) y casi todas las transiciones son enviadas al terminar la acción ejecutada en el estado anterior.

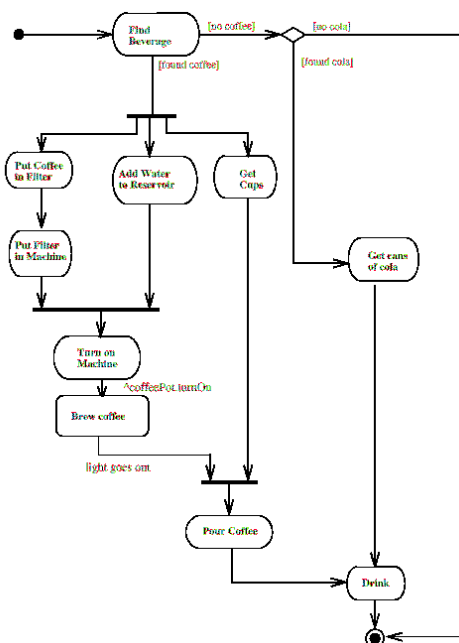


Fig. 6.7 Diagramas de Actividades Ejemplo Máquina de café

- **Estado de acción:** Representa un estado con acción interna, con por lo menos una transición que identifica la culminación de la acción (por medio de un evento implícito).
- **Transiciones:** Las flechas entre estados representan transiciones con evento implícito. Pueden tener una condición en el caso de decisiones.
- **Decisiones:** Se representa mediante una transición múltiple que sale de un estado, donde cada camino tiene un label distinto. Se representa mediante un diamante al cual llega la transición del estado inicial y del cual salen las múltiples transiciones de los estados finales. Un ejemplo se ve en la figura cuando *no hay café* y se toma una decisión entre *hay cola o no hay cola*.

6.1.3.6.- Diagrama de Implementación

Un diagrama de implementación muestra la estructura del código (Diagrama de componentes) y la estructura del sistema en ejecución (Diagrama de ejecución).

6.1.3.6.1.- Diagrama de Componentes

Un diagrama de componentes muestra las dependencias lógicas entre componentes software, sean éstos componentes fuentes, binarios o ejecutables. Un ejemplo es el siguiente:

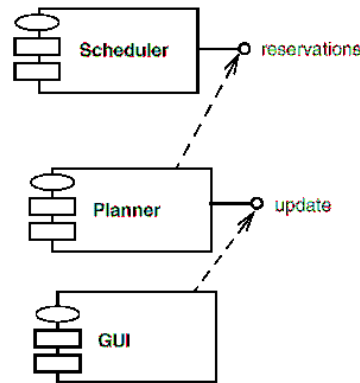


Fig. 6.8 Diagrama de Componentes Ejemplo Máquina de café

En este caso tenemos tres componentes, GUI dependiendo de la interfaz *update* provista por *Planner*, *Planner* dependiendo de la interfaz *reservations* provista por *Scheduler*.

6.1.3.6.2.- Diagrama de Ejecución

Un diagrama de ejecución muestra la configuración de los elementos de procesamiento en tiempo de ejecución y los componentes software, procesos y objetos que se ejecutan en ellos. Instancias de los componentes software representan manifestaciones en tiempo de ejecución del código. Un ejemplo de diagrama de ejecución es el siguiente:

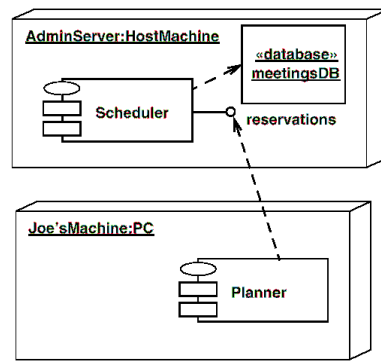


Fig. 6.9 Diagrama de Secuencia Ejemplo Maquina de café

- **Nodos:** Un nodo es un objeto físico en tiempo de ejecución que representa un recurso computacional, generalmente con memoria y capacidad de procesamiento. Pueden representarse instancias o tipos de nodos. Se representa como un cubo 3D en los diagramas de implementación.
- **Componentes:** Un componente representa una unidad de código (fuente, binario o ejecutable) que permite mostrar las dependencias en tiempo de compilación y ejecución. Las instancias de componentes de software muestran unidades de software en tiempo de ejecución y generalmente ayudan a identificar sus dependencias y su localización en nodos.

Un ejemplo de componente que implementa dos interfaces es:

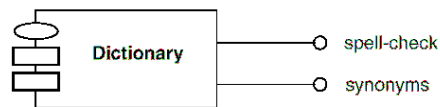


Fig. 6.10 Componentes Ejemplo Maquina de café

NOTA: Para un resumen de notación UML (Ver Anexo 2)

6.2.- NORMAS ISO EEE830⁸

Ámbito.

1. El IEEE está orientado a especificar requisitos de Software a ser desarrollados. Si se aplica este material a Software ya desarrollado resultará contraproducente.
2. El IEEE no cubre la especificación de requisitos de software que está siendo desarrollado usando herramientas prototipadas.

6.2.1.- ESPECIFICACIÓN DE REQUISITOS DE SOFTWARE (SRS)

El SRS es una especificación para un producto software, para un programa, o para un conjunto de programas que realizan diversas cosas.

Existen dos requisitos básicos en el SRS :

⁸ <http://www.informacionnormas/iso.html>

1. El SRS debe decir cosas verdaderas.
2. El SRS debe de decir cosas en ciertos caminos.

6.2.1.1.- El Entorno del SRS.

1. El SRS sólo debe definir correctamente todos los requisitos software.
2. El SRS no debe describir ningún diseño, verificación o detalles de un proyecto de dirección, excepto por necesidades de restricciones de diseño.

6.2.1.2.- Preparación Conjunta del SRS.

El Proceso de desarrollo del Software comienza con determinar un acuerdo entre proveedor y cliente para determinar lo que el Software debe hacer. Este acuerdo debe prepararse conjuntamente. El cliente y el Proveedor necesitan trabajo juntos para producir un SRS bien escrito y que sea completamente entendible. El SRS necesita evolucionar así como a la vez que progresa el producto software.

6.2.1.3.- Herramientas para Desarrollar un SRS.

Herramientas de producción.

Un procesador de textos es el apoyo mas usado para la producción.

Herramientas de representación

Algunas palabras referencian entidades y acciones. Estas palabras deben cumplir:

1. Verificar que cada entidad o acción siempre tiene el mismo nombre en todos los lugares del SRS
2. Es posible identificar cada lugar en la especificación donde una entidad o acción particular es descrita.

6.2.2.- DIVISIÓN DE LOS REQUISITOS DEL SOFTWARE

Los requisitos pueden dividirse en requisitos funcionales y requisitos no funcionales:

- Los requisitos funcionales definen las funciones que el sistema será capaz de realizar. Describen las transformaciones que el sistema realiza sobre las entradas para producir salidas.
- Los requisitos no funcionales tienen que ver con características que de una u otra forma puedan limitar el sistema, como por ejemplo, el rendimiento (en tiempo y espacio), interfaces de usuario, fiabilidad (robustez del sistema, disponibilidad de equipo), mantenimiento, seguridad, portabilidad, estándares, etc.

6.2.3.- CARACTERÍSTICAS DE LOS REQUISITOS DEL SOFTWARE

Las características de un requerimiento son sus propiedades principales. Un conjunto de requerimientos en estado de madurez, deben presentar una serie de características tanto individualmente como en grupo. A continuación se presentan las más importantes:

- Necesario: Un requerimiento es necesario si su omisión provoca una deficiencia en el sistema a construir, y además su capacidad, características físicas o factor de calidad no pueden ser reemplazados por otras capacidades del producto o del proceso.
- Conciso: Un requerimiento es conciso si es fácil de leer y entender. Su redacción debe ser simple y clara para aquellos que vayan a consultarlo en un futuro.

- Completo: Un requerimiento está completo si no necesita ampliar detalles en su redacción, es decir, si se proporciona la información suficiente para su comprensión.
- Consistente: Un requerimiento es consistente si no es contradictorio con otro requerimiento.
- No ambiguo: Un requerimiento no es ambiguo cuando tiene una sola interpretación. El lenguaje usado en su definición, no debe causar confusiones al lector.
- Verificable: Un requerimiento es verificable cuando puede ser cuantificado de manera que permita hacer uso de los siguientes métodos de verificación: inspección, análisis, demostración o pruebas.

6.3.- ANÁLISIS DE LA APLICACIÓN

Los países tecnológicamente desarrollados han puesto en el mercado una serie de propuestas para quienes desean disfrutar de las ventajas de un edificio o residencia automatizada. Actualmente se intenta saturar el mercado con los denominados Sistemas de Seguridad, los mismos que son monitoreados por las respectivas agencias que los distribuyen, haciendo que el “usuario” entre comillas, se mantenga al margen de lo que sucede en su negocio, oficina o vivienda .

Teniendo en cuenta este análisis se ha propuesto presentar el actual proyecto de tesis con la tentativa, de que se amplíen más los horizontes en lo que a Automatización Doméstica se refiere.

Con ello se ha planteado desarrollar un prototipo de Domótica para el Control de Iluminación y Activación de un Sistema de Seguridad de una Vivienda Controlado Mediante Internet, esta red mundial fusionada con el Sistema denominado “Smart House Control” (SHC), permitirá al usuario controlar y monitorear su residencia desde cualquier lugar en el que se encuentre, en el momento que desee, durante las 24 horas del día, los 365 días del año.

UML proporciona una notación orientada a objetos, la misma que integra con gran facilidad todo el ciclo de vida del sistema, desde que éste inicia hasta cuando finaliza; proporcionando una visión completa y coherente de la producción del software.

6.4.- DISEÑO DE LA APLICACIÓN

6.4.1.- INTRODUCCIÓN

El presente documento permite detallar la Especificación de Requisitos de Software (SHC) para el sistema “Smart House Control”, destinado al público en general que desee desarrollar o ampliar sus conocimientos en el campo de Edificios o Residencias Automatizadas.

6.4.2.- PROPÓSITO

El objetivo del presente documento es definir de manera clara y precisa todas las funcionalidades y restricciones del sistema a ser desarrollado, el mismo que deberá permitir: el control de luces, monitoreo de puertas, monitoreo de ventanas, detección de intrusos y temperatura interna de la residencia.

6.4.3.- DEFINICIONES, ACRÓNIMOS Y ABREVIATURAS

6.4.3.1.- Definiciones

Administrador	Persona encargada de la manipulación de la información registrada en el sistema.
Domótica	Concepto de vivienda que integra todos los automatismos en materia de seguridad, gestión de la energía, comunicaciones, etc.
Monitorear	Que sirve para avisar
Controlar	Comprobación, inspección, intervención; dirección, mando, regulación

Tabla 6.1 Definiciones de Smart House Control

6.4.3.2.- Acrónimos

SRS	Software Requirements Specification (Especificación de Requisitos de Software)
E/S	Entrada / Salida

Tabla 6.2 Acrónimos de Smart House Control

6.4.3.3.- Abreviaturas

LH1 ON	Encender Luz Habitación1
LH2 ON	Encender Luz Habitación2
LH3 ON	Encender Luz Habitación3
LE ON	Encender Luz Externa
LH1 OFF	Apagar Luz Habitación1
LH2 OFF	Apagar Luz Habitación2
LH3 OFF	Apagar Luz Habitación3
LE OFF	Apagar Luz Externa
PI1	Puerta de Ingreso1
PI2	Puerta de Ingreso2
V1	Ventana1
V2	Ventana2
SM	Sensor de Movimiento
ST	Sensor de Temperatura
SHC	Smart House Control

Tabla 6.3 Abreviaturas de Smart House Control

6.4.4.- REFERENCIAS

- IEEE Recommended Practice for Software Requirements Specifications
ANSI/IEEE830 1998

6.4.5.- DESCRIPCIÓN GENERAL

Esta sección nos presenta una descripción general del sistema, con el fin conocer las principales funciones que debe realizar, restricciones o cualquier factor que afecte el desarrollo mismo del sistema.

6.4.5.1.- Perspectivas del Producto

El sistema permitirá interactuar con la red Internet; la misma que será simulada a través de una Intranet y se podrá acceder desde un computador conectado a la red.

6.4.5.2.- Funciones del Sistema

El sistema deberá realizar tarea de Gestión en las siguientes áreas:

- Gestión Permiso de Acceso
- Gestión Encendido de Luces
- Gestión Apagado de Luces

- Gestión Visualizar Estado de Sensor de Puertas
- Gestión Visualizar Estado de Sensor de Ventanas
- Gestión Visualizar Estado de Sensor de Movimiento
- Gestión Visualizar Temperatura Interna de la Residencia
- Gestión Activar Alarma
- Gestión Cerrar Conexión
- Gestión Inicialización de Estados

6.4.5.2.1.- Gestión Permiso de Acceso.

Para ingresar el permiso de acceso el Administrador introducirá “controlhouse” o “CONTROLHOUSE”, en el formulario de Ingresar Permiso de Acceso y seleccionará la opción aceptar.

6.4.5.2.2.- Gestión Encendido de Luces

Para encender las luces de la residencia el Administrador pulsará el botón respectivo de la habitación en la que desea que se encienda la luz:

- LH1 ON para encender la luz de la habitación1
- LH2 ON para encender la luz de la habitación2

- LH3 ON para encender la luz de la habitación3
- LE ON para encender la luz externa

6.4.5.2.3.- Gestión Apagado de Luces

Para apagar las luces de la residencia el Administrador pulsará el botón respectivo de la habitación en la que desea que se apague la luz:

- LH1 OFF para apagar la luz de la habitación1
- LH2 OFF para apagar la luz de la habitación2
- LH3 OFF para apagar la luz de la habitación3
- LE OFF para apagar la luz externa

6.4.5.2.4.- Gestión Visualizar Estado de Sensor de Puertas.

Para visualizar el Estado de los Sensores de las Puerta el Administrador solicitará las opciones:

- Puerta de Ingreso1, para observar el estado del Sensor Puerta de Ingreso1
- Puerta de Ingreso2, para observar el estado del Sensor Puerta de Ingreso2

6.4.5.2.5.- Gestión Visualizar Estado de Sensor de Ventanas.

Para visualizar el Estado de los Sensores de las Ventanas el Administrador solicitará las opciones:

- Ventana1, para observar el estado del Sensor Ventana1
- Ventana2, para observar el estado del Sensor Ventana2

6.4.5.2.6.- Gestión Visualizar Estado de Sensor de Movimiento

Para visualizar el Estado del Sensor de Movimiento el Administrador solicitará la opción:

- Estado de Movimiento, para observar el estado del Sensor de Movimiento.

6.4.5.2.7.- Gestión Visualizar Temperatura Interna de la Residencia

Para visualizar la Temperatura Interna de la Residencia el Administrador solicitará la opción:

- Temperatura Interna, para observar la Temperatura Interna de la Residencia.

6.4.5.2.8.- Gestión Activar Alarma

Esta gestión la realiza el sistema cuando se ha producido una violación de la seguridad ya sea en:

- Puerta de Ingreso1
- Puerta de Ingreso2
- Ventana1
- Ventana2
- Si existe movimiento (intruso)
- Peligro de incendio

6.4.5.2.9.- Gestión Cerrar Conexión

Esta gestión se realiza cuando el Administrador desea desconectarse del equipo remoto con el que se encuentra conectado.

6.4.5.2.10.- Gestión Inicialización de Estados

Para Inicializar Estados de luces y demás sensores de la residencia el Administrador deberá pulsar el botón REINICIAR.

6.4.6.- CARACTERÍSTICAS DEL USUARIO

El sistema proporcionará una interfaz fácil de aprender y sencilla de manejar.

6.4.7.- RESTRICCIONES

- Se requerirá instalar una red con plataforma cliente / servidor, la restricción de hardware y software es de que el sistema se desarrolle para esta plataforma.
- El puerto paralelo debe ser configurable a modo SPP (Standar Parallel Port).

6.4.8.- REQUISITOS ESPECÍFICOS

A continuación se presentan una serie de requisitos funcionales que deberá satisfacer el sistema

6.4.8.1.- Requisitos Funcionales

6.4.8.1.1.- Gestión Permiso de Acceso.

- Req(01) Ingresar Permiso de Acceso

6.4.8.1.2.- Gestión Encendido de Luces

- Req(02) Encender Luz Habitación1
- Req(03) Encender Luz Habitación2
- Req(04) Encender Luz Habitación3
- Req(05) Encender Luz Externa

6.4.8.1.3.- Gestión Apagado de Luces

- Req(06) Apagar Luz Habitación1
- Req(07) Apagar Luz Habitación2
- Req(08) Apagar Luz Habitación3
- Req(09) Apagar Luz Externa

6.4.8.1.4.- Gestión Visualizar Estado de Sensor de Puertas y Ventanas

- Req(10) Visualizar Estado de Sensor Puerta de Ingreso1
- Req(11) Visualizar Estado de Sensor Puerta de Ingreso2
- Req(12) Visualizar Estado de Sensor Ventana1
- Req(13) Visualizar Estado de Sensor Ventana2

6.4.8.1.5.- Gestión Visualizar Estado de Sensor de Movimiento

- Req(14) Visualizar Estado de Sensor de Movimiento

6.4.8.1.6.- Gestión Visualizar Temperatura Interna de la Residencia

- Req(15) Visualizar Temperatura Interna de la Residencia

6.4.8.1.7.- Gestión Activar Alarma

- Req(16) Activar Alarma

6.4.8.1.8.- Gestión Cerrar Conexión

- Req(17) Cerrar Conexión

6.4.8.1.9.- Gestión Inicialización de Estados

- Inicializar estados de luces y demás sensores

6.4.8.2.- Requisitos de Interfaz

6.4.8.2.1.- Interfaz de Usuario

Esta interfaz estará orientada al manejo de páginas web y formularios y su manipulación podrá ser a través del mouse.

6.4.8.2.2.- Interfaz de Hardware

El sistema se basará en una plataforma cliente / servidor.

6.4.8.2.3.- Interfaz de Software

No existirá interfaz con sistemas externos.

6.4.8.2.4.- Interfaz de Comunicación

- La conexión se establecerá mediante una Intranet.
- Se utilizara el control Winsock para envío y recepción de datos

6.4.8.3.- Requisitos de Software

- La aplicación “Smart House Control Client” y “Smart House Control Server” así como las páginas web; residirán en el servidor web, en la dirección C:\Archivos de Programa\Apache Group\ Apache\htdocs, si la aplicación va a ser ejecutar en una intranet.
- Un PC conectado a la Intranet (en modo SPP- Standar Parallel Port), podrá acceder al servidor (también en modo SPP).
- Para una sola PC únicamente se ejecutará la aplicación “Smart House Control PC”.

- El sistema operativo que utilizaremos es Windows.
- Es necesario instalar la librería inpout32.dll en C:\WINDOWS\SYSTEM\ ya que es indispensable para trabajar con el puerto paralelo.

6.4.9.- CASOS DE USO DE ALTO NIVEL

6.4.9.1.- Modelo de Caso de Uso del Actor Administrador

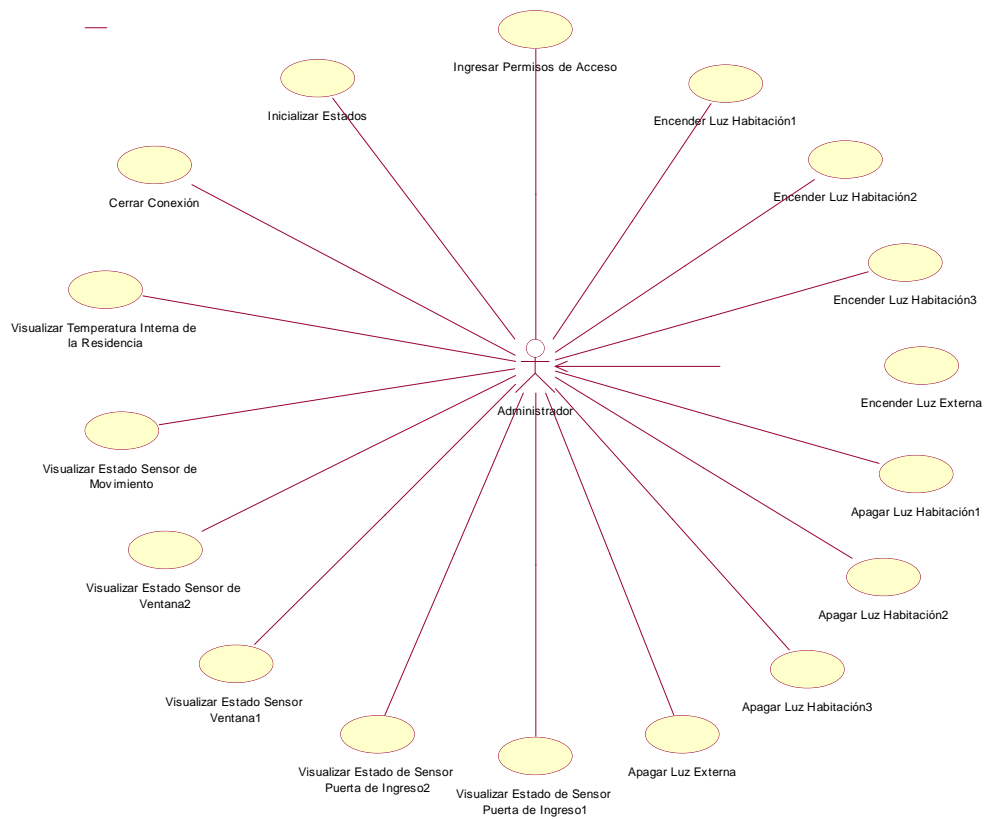


Fig. 6.11 Casos de Uso del Actor Administrador

6.4.9.1.1.- Descripción de los Casos de Uso Alto Nivel del Actor Administrador.

GESTION PERMISO DE ACCESO

CASO DE USO: Ingresar Permiso de Acceso

ACTOR: Administrador

DESCRIPCIÓN: Administrador solicita la operación de Ingresar Permiso de Acceso, el sistema despliega formulario para Ingreso Permiso de Acceso, el Administrador ingresa el nombre del permiso y confirma la operación, el sistema verifica la solicitud.

GESTION ENCENDIDO DE LUCES

CASO DE USO: Encender Luz Habitación1

ACTOR: Administrador

DESCRIPCIÓN: Administrador solicita la operación de Encender Luz Habitación1, el sistema presenta botón LH1 ON, el Administrador pulsa el botón LH1 ON, el Sistema enciende la Luz de la Habitación1.

CASO DE USO: Encender Luz Habitación2

ACTOR: Administrador

DESCRIPCIÓN: Administrador solicita la operación de Encender Luz Habitación2,
el sistema presenta botón LH2 ON, el Administrador pulsa el botón LH2 ON, el Sistema enciende la Luz de la Habitación2.

CASO DE USO: Encender Luz Habitación3

ACTOR: Administrador

DESCRIPCIÓN: Administrador solicita la operación de Encender Luz Habitación3,

el sistema presenta botón LH3 ON, el Administrador pulsa el botón LH3 ON, el Sistema enciende la Luz de la Habitación3.

CASO DE USO: Encender Luz Externa

ACTOR: Administrador

DESCRIPCIÓN: Administrador solicita la operación de Encender Luz Habitación Externa, el sistema presenta botón LE ON, el Administrador pulsa el botón LE ON, el Sistema enciende la Luz de la Habitación Externa .

GESTION APAGADO DE LUCES

CASO DE USO: Apagar Luz Habitación1

ACTOR: Administrador

DESCRIPCIÓN: Administrador solicita la operación de Apagar Luz Habitación1, el sistema presenta botón LH1 OFF, el Administrador pulsa el botón LH1 OFF, el Sistema apaga la Luz de la Habitación1.

CASO DE USO: Apagar Luz Habitación2

ACTOR: Administrador

DESCRIPCIÓN: Administrador solicita la operación de Apagar Luz Habitación2, el sistema presenta botón LH2 OFF, el Administrador pulsa el botón LH2 OFF, el Sistema apaga la Luz de la Habitación2.

CASO DE USO: Apagar Luz Habitación3

ACTOR: Administrador

DESCRIPCIÓN: Administrador solicita la operación de Apagar Luz Habitación3, el sistema presenta botón LH3 OFF, el Administrador pulsa el botón LH3 OFF, el Sistema apaga la Luz de la Habitación3.

CASO DE USO: Apagar Luz Externa

ACTOR: Administrador

DESCRIPCIÓN: Administrador solicita la operación de Apagar Luz Habitación Externa, el sistema presenta botón LE OFF, el Administrador pulsa el botón LE OFF, el Sistema apaga la Luz de la Habitación Externa.

GESTION VISUALIZAR ESTADO DE SENSOR DE PUERTAS

CASO DE USO: Visualizar Estado Actual del Sensor Puerta de Ingreso1

ACTOR: Administrador

DESCRIPCIÓN: El administrador solicita Visualizar Estado Actual del Sensor Puerta de Ingreso1, el sistema monitorea y presenta el estado solicitado, el administrador visualiza el estado actual de Puerta de Ingreso1.

CASO DE USO: Visualizar Estado Actual del Sensor Puerta de Ingreso2

ACTOR: Administrador

DESCRIPCIÓN: El administrador solicita Visualizar Estado Actual del Sensor Puerta de Ingreso2, el sistema monitorea y presenta el estado solicitado, el administrador visualiza el estado actualde Puerta de Ingreso2.

GESTION VISUALIZAR ESTADO DE SENSOR DE VENTANAS

CASO DE USO: Visualizar Estado Actual del Sensor de Ventana1

ACTOR: Administrador

DESCRIPCIÓN: El administrador solicita Visualizar Estado Actual del Sensor de Ventana1, el sistema monitorea y presenta el estado solicitado, el administrador visualiza el estado actual del Sensor de Ventana1.

CASO DE USO: Visualizar Estado Actual del Sensor de Ventana2

ACTOR: Administrador

DESCRIPCIÓN: El administrador solicita Visualizar Estado Actual del Sensor de Ventana2, el sistema monitorea y presenta el estado solicitado, el administrador visualiza el estado actual del Sensor de Ventana2.

GESTION VISUALIZAR ESTADO DE SENSOR DE MOVIMIENTO

CASO DE USO: Visualizar Estado Actual del Sensor de Movimiento

ACTOR: Administrador

DESCRIPCIÓN: El administrador solicita Visualizar Estado Actual del Sensor de Movimiento, el sistema monitorea y

presenta el estado solicitado, el administrador visualiza el estado actual del Sensor de Movimiento.

GESTION VISUALIZAR TEMPERATURA INTERNA DE LA RESIDENCIA

CASO DE USO: Visualizar Temperatura Interna de la Residencia

ACTOR: Administrador

DESCRIPCIÓN: El administrador solicita Visualizar Temperatura Interna de la Residencia, el sistema monitorea y presenta el estado solicitado, el administrador visualiza la Temperatura Interna de la residencia.

GESTION CERRAR CONEXIÓN

CASO DE USO: Cerrar Conexión

ACTOR: Administrador

DESCRIPCIÓN: Administrador solicita la operación Cerrar Conexión, el sistema presenta botón Disconnect, el Administrador pulsa el botón Disconnect, el Sistema cierra la conexión con el Host Remoto.

GESTION INICIALIZACION DE ESTADOS

CASO DE USO: Inicializar Estados de Monitoreo y Control Smart House

ACTOR: Administrador

DESCRIPCIÓN: Administrador solicita la operación Inicializar Estados, el sistema presenta botón RESET, el administrador pulsa botón RESET, el sistema inicializa estados.

6.4.9.2.- Descripción de los Casos de Uso Alto Nivel del Actor Sistema

GESTION ACTIVAR ALARMA

CASO DE USO: Activar Alarma

ACTOR: Sistema

DESCRIPCIÓN: El sistema detecta violación de seguridad en PI1, PI2, V1, V2, o SM, el sistema activa alarma y luego de 15 segundos presenta Formulario de Clave de Desactivación de alarma.

GESTION CONEXIÓN

CASO DE USO: Conectar con el Host Remoto

ACTOR: Sistema

DESCRIPCIÓN: El sistema realiza la petición de conexión con el Host Remoto, el sistema verifica la aceptación y realiza la conexión.

6.4.10.- CASOS DE USO EXPANDIDOS

Se van a desarrollar los casos de uso que son básicos y los más importantes para el desarrollo del presente proyecto.

6.4.10.1.- Descripción de los Casos de Uso Expandidos del Actor Administrador

GESTION PERMISO DE ACCESO

CASO DE USO: Ingresar Permiso de Acceso

ACTOR: Administrador (Iniciador)

PROPÓSITO: Ingresar un permiso de Acceso en el Sistema

VISIÓN GENERAL: Administrador solicita la operación de Ingresar Permiso de Acceso, el sistema despliega formulario para Ingreso Permiso de Acceso, el Administrador ingresa el nombre del permiso y

confirma la operación, el sistema verifica la solicitud y despliega formulario Monitoreo y Control Smart House.

TIPO: Primario Real

REFERENCIAS:

CURSO TÍPICO DE EVENTOS:

Acción del Actor	Respuesta del Sistema
1. Este caso de uso comienza cuando el Administrador solicita la operación de Ingresar Permiso de Acceso.	2. Presenta formulario de ingreso de permiso de acceso.
3. Ingresar el nombre del permiso y confirma la operación.	4. Verifica la operación y despliega formulario Monitoreo y Control Smart House.

Tabla 6.4 Caso de Uso Expandido Ingresar Permiso de Acceso

CURSOS ALTERNATIVOS:

Línea 2: No existe formulario de ingreso, termina el caso de uso.

Línea 4: No existe permiso de acceso, termina el caso de uso

GESTION ENCENDIDO DE LUCES

CASO DE USO: Encender Luz Habitación1

ACTOR: Administrador (Iniciador)

PROPÓSITO: Encender Luz Habitación1 en el Sistema y Prototipo

VISIÓN GENERAL: Administrador solicita la operación de Encender Luz Habitación1, el sistema presenta botón LH1 ON, el Administrador pulsa el botón LH1 ON, el Sistema enciende la Luz de la Habitación1 y visualiza en pantalla.

TIPO: Primario, Real

REFERENCIA:

CURSO TÍPICO DE EVENTOS:

Acción del Actor	Respuesta del Sistema
1. Este caso de uso comienza cuando el Administrador solicita la operación Encender Luz Habitación1.	2. Presenta botón LH1 ON.
3. Pulsa el botón LH1 ON	4. Enciende la luz de la habitación1 y visualiza en pantalla.

Tabla 6.5 Caso de Uso Expandido Encender Luz Habitación1

CURSOS ALTERNATIVOS:

Línea 1: Si el Administrador no solicita la operación Encender Luz habitación1

Línea 3: Si el Administrador no pulsa el botón LH1 ON, termina el caso de uso.

GESTION APAGADO DE LUCES

CASO DE USO: **Apagar Luz Habitación1**

ACTOR: Administrador (Iniciador)

PROPÓSITO: Apagar Luz Habitación1 en el Sistema y Prototipo

VISIÓN GENERAL: Administrador solicita la operación de Apagar Luz Habitación1, el sistema presenta botón LH1 OFF, el Administrador pulsa el botón LH1 OFF, el Sistema apaga la Luz de la Habitación1 y visualiza en pantalla.

TIPO: Primario, Real

REFERENCIA:

CURSO TÍPICO DE EVENTOS:

Acción del Actor	Respuesta del Sistema
1. Este caso de uso comienza cuando el Administrador solicita la operación Apagar Luz Habitación1.	2. Presenta botón LH1 OFF.
3. Pulsa el botón LH1 OFF	4. Apaga la luz de la habitación1 y visualiza en pantalla.

Tabla 6.6 Caso de Uso Expandido Apagar Luz Habitación1

CURSOS ALTERNATIVOS:

Línea 1: Si el Administrador no solicita la operación apagar Luz habitación1

Línea 3: Si el Administrador no pulsa el botón LH1 OFF, termina el caso de uso.

GESTION VISUALIZAR ESTADO DE SENSOR DE PUERTAS

CASO DE USO: Visualizar Estado Actual del Sensor Puerta de Ingreso1

ACTOR: Administrador

PROPÓSITO: Visualizar Estado Actual del Sensor Puerta de Ingreso1 en el Sistema.

VISIÓN GENERAL: El administrador Solicita Visualizar Estado Actual del Sensor Puerta de Ingreso1, el sistema monitorea y presenta el estado solicitado, el administrador visualiza el estado actual de Puerta de Ingreso1.

TIPO: Primario, Real

REFERENCIA:

CURSO TÍPICO DE EVENTOS:

Acción del Actor	Respuesta del Sistema
1. Solicita Visualizar Estado Actual del Sensor Puerta de Ingreso1.	2. Monitorea y presenta el estado solicitado.
3. El administrador visualiza el estado actual del Sensor Puerta de Ingreso1.	

Tabla 6.7Caso de Uso Expandido Visualizar Estado Actual del
Sensor Puerta de Ingreso1

CURSOS ALTERNATIVOS:

Línea1: Si el Administrador no solicita la operación Visualizar Estado Actual del Sensor Puerta de Ingreso1.

Línea3: No presenta estado solicitado, termina el caso de uso.

GESTION VISUALIZAR ESTADO DE SENSOR DE VENTANAS

CASO DE USO: Visualizar Estado Actual del Sensor Ventana1

ACTOR: Administrador

PROPÓSITO: Visualizar Estado Actual del Sensor Ventana1 en el Sistema.

VISIÓN GENERAL: El administrador Solicita Visualizar Estado Actual del Sensor Ventana1, el sistema monitorea y presenta el estado solicitado, el administrador visualiza el estado actual del Sensor Ventana1.

TIPO: Primario, Real

REFERENCIA:

CURSO TÍPICO DE EVENTOS:

Acción del Actor	Respuesta del Sistema
1. Solicita Visualizar Estado Actual del Sensor Ventana1.	2. Monitorea y presenta el estado solicitado.
3. El administrador visualiza el estado actual del Sensor Ventana1.	

Tabla 6.8 Caso de Uso Expandido Visualizar Estado Actual del Sensor Ventana1

CURSOS ALTERNATIVOS:

Línea1: Si el Administrador no solicita la operación Visualizar Estado Actual del Sensor Ventana1.

Línea3: No presenta estado solicitado, termina el caso de uso.

GESTION VISUALIZAR ESTADO DE SENSOR DE MOVIMIENTO

CASO DE USO: Visualizar Estado Actual del Sensor de Movimiento

ACTOR: Administrador

PROPÓSITO: Visualizar Estado Actual del Sensor de Movimiento en el Sistema.

VISIÓN GENERAL: El administrador Solicita Visualizar Estado Actual del Sensor de Movimiento, el sistema monitorea y presenta el estado solicitado, el administrador visualiza el estado actual del Sensor de Movimiento.

TIPO: Primario, Real

CURSO TÍPICO DE EVENTOS:

Acción del Actor	Respuesta del Sistema
1. Solicita Visualizar Estado Actual del Sensor de Movimiento.	2. Monitorea y presenta el estado solicitado.
3. El administrador visualiza el estado actual del Sensor de Movimiento.	

Tabla 6.9 Caso de Uso Expandido Visualizar Estado Actual del
Sensor de Movimiento

CURSOS ALTERNATIVOS:

Línea1: Si el Administrador no solicita la operación Visualizar Estado Actual del
Sensor de Movimiento.

Línea3: No presenta estado solicitado, termina el caso de uso.

GESTION VISUALIZAR TEMPERATURA INTERNA DE LA RESIDENCIA

CASO DE USO: Visualizar Temperatura Interna de la Residencia

ACTOR: Administrador

PROPÓSITO: Visualizar Temperatura Interna de la Residencia.

VISIÓN GENERAL: El administrador Solicita Visualizar Temperatura Interna de la Residencia, el sistema monitorea y presenta el estado solicitado, el administrador visualiza la Temperatura Interna de la residencia en grados centígrados.

TIPO: Primario, Real

REFERENCIA:

CURSO TÍPICO DE EVENTOS:

Acción del Actor	Respuesta del Sistema
1. Solicita Visualizar Temperatura Interna de la Residencia.	2. Monitorea y presenta el estado solicitado.
3. Visualiza la Temperatura Interna de la residencia en grados centígrados.	

Tabla 6.10 Caso de Uso Expandido Visualizar Temperatura Interna
de la Residencia

CURSOS ALTERNATIVOS:

Línea1: Si el Administrador no solicita la operación Visualizar Temperatura Interna de la Residencia.

Línea3: No presenta estado solicitado, termina el caso de uso.

GESTION CERRAR CONEXIÓN

CASO DE USO: Cerrar Conexión

ACTOR: Administrador (Primario)

PROPÓSITO: Cerrar conexión con el Host Remoto.

VISION GENERAL: Administrador solicita la operación Cerrar Conexión, el sistema presenta botón Disconnect, el Administrador pulsa el botón Disconnect, el Sistema cierra la conexión con el Host Remoto.

TIPO: Primario, Real

REFERENCIA:

CURSO TÍPICO DE EVENTOS:

Acción del Actor	Respuesta del Sistema
1. Solicitar la operación Cerrar Conexión	2. Presenta botón Disconnect

3. Pulsar el botón Disconnect	4. Cerrar la conexión con el Host Remoto.
-------------------------------	---

Tabla 6.11 Caso de Uso Expandido Cerrar Conexión

CURSOS ALTERNATIVOS:

Línea 1: Si el Administrador no solicita la operación Cerrar Conexión termina el caso de uso.

Línea 3: Si el Administrador no pulsa el botón Disconnect, termina el caso de uso.

GESTION INICIALIZACIÓN DE ESTADOS

CASO DE USO: Inicializar Estados de Monitoreo y Control Smart House

ACTOR: Administrador (Iniciador)

PROPÓSITO: Inicializar Estados de Monitoreo y Control Smart House en el Sistema y Prototipo

VISION GENERAL: Administrador solicita la operación Inicializar Estados de Monitoreo y Control Smart House, el sistema presenta botón RESET, el administrador pulsa botón RESET, el sistema inicializa estados.

TIPO: Primario, Real

CURSO TÍPICO DE EVENTOS:

Acción del Actor	Respuesta del Sistema
1. Este caso de uso comienza cuando el	2. Presenta botón RESET.

administrador solicita la operación Inicializar Estados.	
3. Pulsa botón RESET.	4. Inicializa estados.

Tabla 6.12 Caso de Uso Expandido Inicializar Estados de Monitoreo
y Control Smart House

CURSOS ALTERNATIVOS:

Línea1: El administrador no solicita operación Inicializar Estados

Línea 3: El administrador no pulsa el botón RESET, termina el caso de uso.

**6.4.10.2 .- Descripción de los Casos de Uso Expandido del Actor
Sistema**

GESTION ACTIVAR ALARMA

CASO DE USO: Activar Alarma

ACTOR: Sistema (Iniciador)

PROPÓSITO: Activar Alarma en el Sistema y Prototipo

VISIÓN GENERAL: El sistema detecta violación de seguridad en PI1, PI2, V1, V2, o SM, el sistema activa alarma y luego de 15 segundos presenta Formulario de Clave de Desactivación de alarma.

TIPO: Primario, Real

REFERENCIA:

CURSO TÍPICO DE EVENTOS:

Acción del Sistema
1. Detecta violación de seguridad en PI1, PI2, V1, V2, SM.
2. Activa alarma y luego de 15 segundos presenta Formulario de Clave de Desactivación de alarma.

Tabla 6.13 Caso de Uso Expandido Activar Alarma

CURSOS ALTERNATIVOS:

Línea 1: No se detecta intruso, termina el caso de uso.

GESTION CONEXIÓN

CASO DE USO: Conectar con el Host Remoto

ACTOR: Sistema (Iniciador)

PROPÓSITO: Establecer conexión con e el Host Remoto.

VISIÓN GENERAL: El sistema realiza la petición de conexión con el Host Remoto, el sistema verifica la aceptación y realiza la conexión.

TIPO: Primario, Real

REFERENCIA:

CURSO TÍPICO DE EVENTOS:

Acción del Sistema

1. Realiza la petición de conexión con el Host Remoto.

2. Verifica la aceptación y realiza la conexión.

Tabla 6.14 Caso de Uso Expandido Conectar con el Host Remoto

CURSOS ALTERNATIVOS:

Línea 1: No se realice la petición de conexión, termina el caso de uso.

6.4.11.- FASE DE CONSTRUCCIÓN: DISEÑO DE ALTO NIVEL

6.4.11.1.- Diagramas de Secuencia del Sistema

CASOS DE USO DEL ACTOR ADMINISTRADOR

Ingresar permiso de acceso

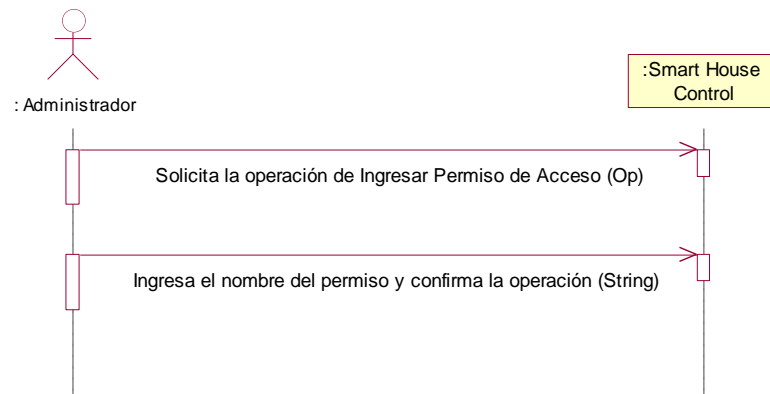


Fig. 6.12 Casos de Uso Ingresar Permiso de Acceso

Encender Luz Habitación1

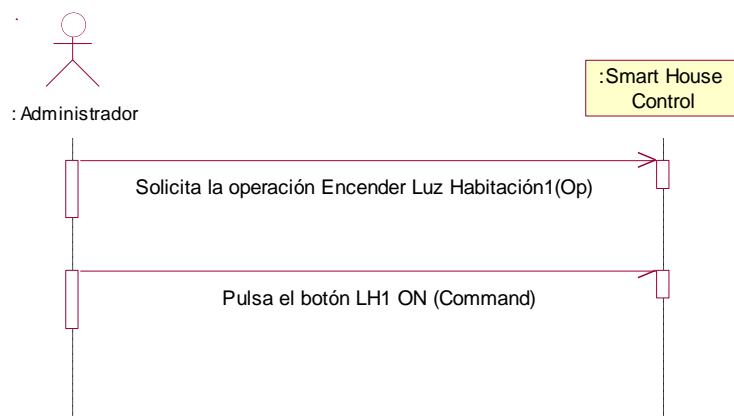


Fig. 6.13 Casos de Uso Encender Luz Habitación1

Apagar Luz Habitación1

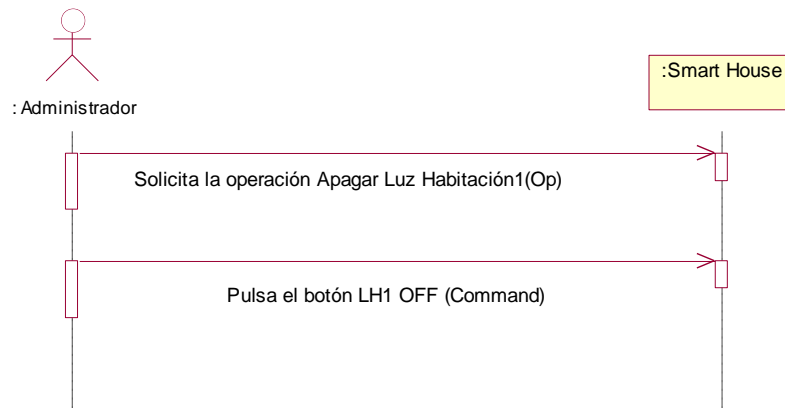


Fig. 6.14 Casos de Uso Apagar Luz Habitación1

Visualizar Estado de Sensor Puerta de Ingreso1

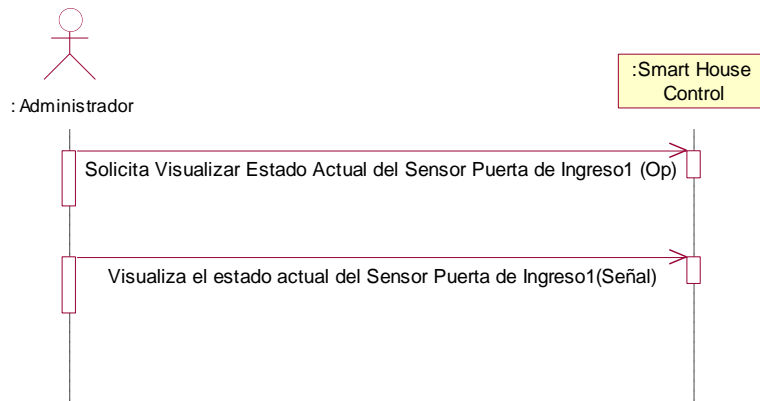


Fig. 6.15 Casos de Uso Visualizar Estado Actual del Sensor Puerta de Ingreso1

Visualizar Estado de Sensor de Ventana1

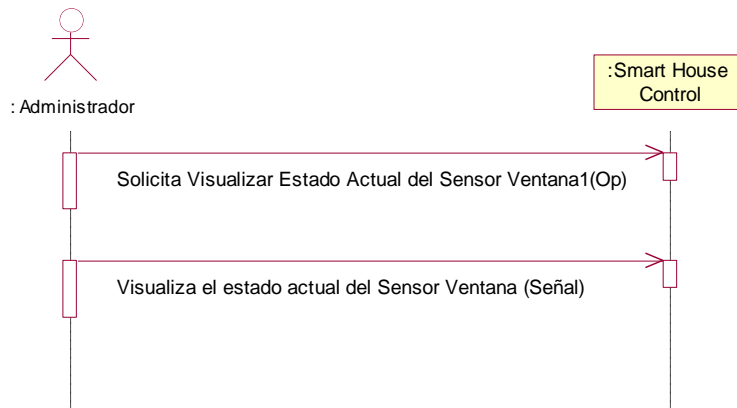


Fig. 6.16 Casos de Uso Visualizar Estado Actual del Sensor Ventana1

Visualizar Estado de Sensor de Movimiento

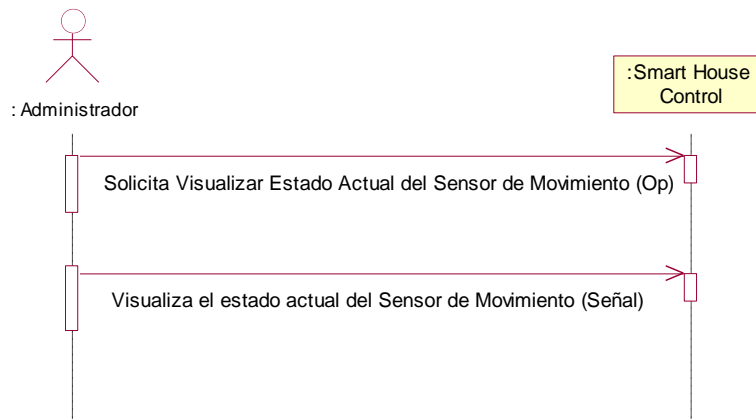


Fig. 6.17 Casos de Uso Visualizar Estado Actual del Sensor de Movimiento

Visualizar Estado Temperatura Interna de la Residencia

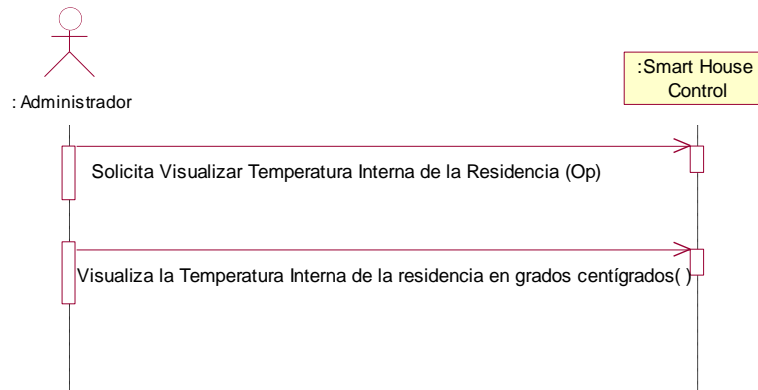


Fig. 6.18 Casos de Uso Visualizar Estado Temperatura Interna de la Residencia

Inicializar Estados

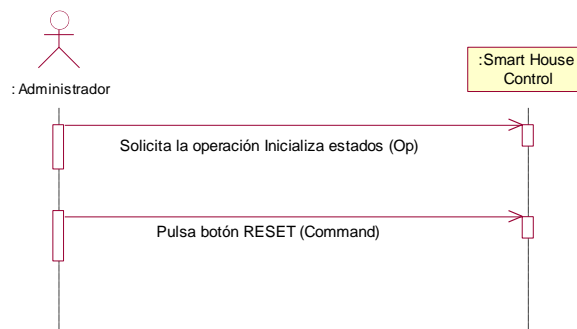


Fig. 6.19 Casos de Uso Inicializar Estados

6.4.12.- CONTRATOS DE OPERACIÓN

Casos de uso del Actor Administrador

INGRESAR PERMISO DE ACCESO

CONTRATO DE OPERACIÓN SOLICITA OPERACIÓN INGRESAR PERMISO DE ACCESO.

NOMBRE :	Solicita la Operación de Ingresar Permiso de Acceso
PROPÓSITO:	Presentar formulario para ingresar permiso de acceso.
TIPO:	Sistema
REFERENCIAS:	Caso de uso Ingresar Permiso de Acceso, diagrama de secuencia Ingresar Permiso de Acceso.
EXCEPCIONES:	No existe formulario.
SALIDA:	-
PRECONDICIONES:	Exista formulario para el ingreso del permiso.
POSTCONDICIONES:	Ninguna.

CONTRATO DE OPERACIÓN INGRESAR NOMBRE DEL PERMISO Y CONFIRMAR OPERACIÓN.

NOMBRE :	Ingresar nombre del permiso y confirmar Operación.
PROPÓSITO:	Permitir al Administrador el ingreso de datos del permiso, en el campo respectivo y confirmar operación.
TIPO:	Sistema

REFERENCIAS: Caso de uso Ingresar Permiso de Acceso, diagrama de secuencia Ingresar Permiso de Acceso.

EXCEPCIONES: No exista permiso de acceso.

SALIDA: -

PRECONDICIONES: Exista permiso de acceso.

POSTCONDICIONES: Se crea una instancia tipo permiso de acceso.

CONTRATO DE OPERACIÓN SOLICITA LA OPERACIÓN ENCENDER LUZ HABITACIÓN1

NOMBRE : Solicita la operación Encender Luz Habitación1.

PROPÓSITO: Permitir al Administrador solicitar operación encender la luz de la habitación1

TIPO: Sistema

REFERENCIAS: Caso de uso Encender Luz Habitación1, diagrama de secuencia Encender Luz Habitación1.

EXCEPCIONES: Si el Administrador no solicita la operación Encender Luz Habitación1

SALIDA: -

PRECONDICIONES: El Administrador solicite la operación Encender Luz habitación1

POSTCONDICIONES: Ninguna.

CONTRATO DE OPERACIÓN PULSA EL BOTÓN LH1 ON

NOMBRE :	Pulsa el botón LH1 ON
PROPÓSITO:	Permitir al Administrador encender la luz de la habitación1 en el sistema y prototipo.
TIPO:	Sistema
REFERENCIAS:	Caso de uso Encender Luz Habitación1, diagrama de secuencia Encender Luz Habitación1.
EXCEPCIONES:	El Administrador no pulse el botón LH1 ON
SALIDA:	-
PRECONDICIONES:	El Administrador pulse el botón LH1 ON
POSTCONDICIONES:	Ninguna.

NOTA: Los mismos contratos son establecidos para Encender de Luz de la Habitación2, Encender de Luz de la Habitación3, Encender Luz Externa.

CONTRATO DE OPERACIÓN SOLICITA LA OPERACIÓN APAGAR LUZ HABITACIÓN1

NOMBRE :	Solicita la operación Apagar Luz Habitación1.
----------	---

PROPÓSITO:	Permitir al Administrador solicitar operación apagar la luz de la habitación1
TIPO:	Sistema
REFERENCIAS:	Caso de uso Apagar Luz Habitación1, diagrama de secuencia Apagar Luz Habitación1.
EXCEPCIONES:	Si el Administrador no solicita la operación Apagar Luz habitación1
SALIDA:	-
PRECONDICIONES:	El Administrador solicite la operación Apagar Luz Habitación1
POSTCONDICIONES:	Ninguna.

CONTRATO DE OPERACIÓN PULSA EL BOTÓN LH1 OFF

NOMBRE : Pulsa el botón LH1 OFF

PROPÓSITO:	Permitir al Administrador apagar la luz de la habitación1 en el sistema y prototipo.
TIPO:	Sistema
REFERENCIAS:	Caso de uso Apagar Luz Habitación1, diagrama de secuencia Apagar Luz Habitación1.
EXCEPCIONES:	El Administrador no pulse el botón LH1 OFF
SALIDA:	-

PRECONDICIONES: El Administrador pulse el botón LH1 OFF

POSTCONDICIONES: Ninguna.

NOTA: Los mismos contratos son establecidos para Apagar de Luz de la Habitación2, Apagar de Luz de la Habitación3, Apagar Luz Externa.

CONTRATO DE OPERACIÓN VISUALIZAR ESTADO DE SENSOR DE PUERTA DE INGRESO1

NOMBRE : Solicita Visualizar Estado Actual del Sensor Puerta de Ingreso1.

PROPÓSITO: Permitir al Administrador solicitar visualizar estado actual del sensor puerta de ingreso1.

TIPO: Sistema

REFERENCIAS: Caso de uso Visualizar Estado Actual del Sensor Puerta de Ingreso1, diagrama de secuencia Visualizar Estado Actual del Sensor Puerta de Ingreso1.

EXCEPCIONES: Si el Administrador no solicita la operación Visualizar Estado Actual del Sensor Puerta de Ingreso1.

SALIDA: -

PRECONDICIONES: El Administrador solicite la operación Visualizar Estado Actual del Sensor Puerta de Ingreso1

POSTCONDICIONES: Ninguna.

CONTRATO DE OPERACIÓN VISUALIZA ESTADO ACTUAL DE SENSOR
DE PUERTA DE INGRESO1

NOMBRE : Visualiza el estado actual del
Sensor Puerta de Ingreso1.

PROPÓSITO: Permitir al Administrador visualizar estado actual del
sensor puerta de ingreso1.

TIPO: Sistema

REFERENCIAS: Caso de uso Visualizar Estado Actual del Sensor
Puerta de Ingreso1, diagrama de secuencia Visualizar
Estado Actual del Sensor Puerta de Ingreso1.

EXCEPCIONES: No presenta estado solicitado

SALIDA: -

PRECONDICIONES: El sistema presente estado solicitado

POSTCONDICIONES: Ninguna.

CONTRATO DE OPERACIÓN SOLICITA VISUALIZAR ESTADO DE
SENSOR DE VENTANA1

NOMBRE : Solicita Visualizar Estado Actual
del Sensor Ventana1.

PROPÓSITO: Permitir al Administrador solicitar visualizar estado actual del sensor Ventana1.

TIPO: Sistema

REFERENCIAS: Caso de uso Visualizar Estado Actual del Sensor Ventana1, diagrama de secuencia Visualizar Estado Actual del Sensor Ventana1.

EXCEPCIONES: Si el Administrador no solicita la operación Visualizar Estado Actual del Sensor Ventana1.

SALIDA: -

PRECONDICIONES: El Administrador solicite la operación Visualizar Estado Actual del Sensor Ventana1.

POSTCONDICIONES: Ninguna.

CONTRATO DE OPERACIÓN VISUALIZA ESTADO ACTUAL DE SENSOR
VENTANA1

NOMBRE : Visualiza el estado actual del
Sensor Ventana1.

PROPÓSITO: Permitir al Administrador visualizar estado actual del
sensor Ventana1.

TIPO: Sistema

REFERENCIAS: Caso de uso Visualizar Estado Actual del Sensor
Ventana1, diagrama de secuencia Visualizar Estado
Actual del Sensor Ventana1.

EXCEPCIONES: No presenta estado solicitado

SALIDA: -

PRECONDICIONES: El sistema presente estado solicitado

POSTCONDICIONES: Ninguna.

CONTRATO DE OPERACIÓN SOLICITA VISUALIZAR ESTADO ACTUAL
DEL SENSOR DE MOVIMIENTO

NOMBRE : Solicita Visualizar Estado Actual
del Sensor de movimiento.

PROPÓSITO: Permitir al Administrador solicitar visualizar estado actual del sensor de movimiento.

TIPO: Sistema

REFERENCIAS: Caso de uso Visualizar Estado Actual del Sensor de Movimiento, diagrama de secuencia Visualizar Estado Actual del Sensor de Movimiento.

EXCEPCIONES: Si el Administrador no solicita la operación Visualizar Estado Actual del Sensor de Movimiento.

SALIDA: -

PRECONDICIONES: El Administrador solicite la operación Visualizar Estado Actual del Sensor de Movimiento.

POSTCONDICIONES: Ninguna.

CONTRATO DE OPERACIÓN VISUALIZA EL ESTADO ACTUAL DEL SENSOR DE MOVIMIENTO

NOMBRE : Visualiza el estado actual del Sensor de Movimiento.

PROPÓSITO: Permitir al Administrador visualizar estado actual del sensor de movimiento.

TIPO: Sistema

REFERENCIAS: Caso de uso Visualizar Estado Actual del Sensor de Movimiento, diagrama de secuencia Visualizar Estado Actual del Sensor de Movimiento.

EXCEPCIONES: No presenta estado solicitado

SALIDA: -

PRECONDICIONES: El sistema presente estado solicitado

POSTCONDICIONES: Ninguna.

**CONTRATO DE OPERACIÓN SOLICITA
VISUALIZAR TEMPERATURA INTERNA DE LA
RESIDENCIA.**

NOMBRE : Solicita Visualizar Temperatura
Interna de la Residencia.

PROPÓSITO: Permitir al Administrador solicitar
visualizar temperatura interna de
la residencia.

TIPO: Sistema

REFERENCIAS: Caso de uso Visualizar
Temperatura Interna de la
Residencia, diagrama de
secuencia Visualizar
Temperatura Interna de la
Residencia.

EXCEPCIONES: Si el Administrador no solicita la operación Visualizar
Temperatura Interna de la Residencia.

SALIDA: -

PRECONDICIONES: El Administrador solicite la operación Visualizar
Temperatura Interna de la Residencia.

POSTCONDICIONES: Ninguna.

**CONTRATO DE OPERACIÓN VISUALIZA LA TEMPERATURA INTERNA DE
LA RESIDENCIA EN GRADOS CENTÍGRADOS.**

NOMBRE : Visualiza Temperatura Interna de la Residencia en grados centígrados.

PROPÓSITO: Permitir al Administrador visualizar temperatura interna de la residencia en grados centígrados.

TIPO: Sistema

REFERENCIAS: Caso de uso Visualizar Temperatura Interna de la Residencia, diagrama de secuencia Visualizar Temperatura Interna de la Residencia.

EXCEPCIONES: No presenta estado solicitado

SALIDA: -

PRECONDICIONES: El sistema presente estado solicitado

POSTCONDICIONES: Ninguna.

CONTRATO DE OPERACIÓN SOLICITA LA OPERACIÓN INICIALIZAR ESTADOS

NOMBRE : Solicita la Operación Inicializar Estados

PROPÓSITO: Permitir al Administrador solicitar la operación inicializar estados de control y monitoreo de Smart House Control.

TIPO: Sistema

REFERENCIAS: Caso de uso Inicializar Estados de Monitoreo y Control Smart House, diagrama de secuencia Inicializar Estados de Monitoreo y Control Smart House.

EXCEPCIONES: Si el Administrador no solicita la operación Inicializar Estados.

SALIDA: -

PRECONDICIONES: El Administrador solicite la operación Inicializar Estados.

POSTCONDICIONES: Ninguna.

CONTRATO DE OPERACIÓN PULSA BOTÓN RESET

NOMBRE : Pulsa botón RESET.

PROPÓSITO: Permitir al Administrador pulsar el botón RESET para inicializar estados de control y monitoreo de Smart House Control.

TIPO: Sistema

REFERENCIAS: Caso de uso Inicializar Estados de Monitoreo y Control Smart House, diagrama de secuencia Inicializar Estados de Monitoreo y Control Smart House.

EXCEPCIONES: El Administrador no pulsa el botón RESET.

SALIDA: -

PRECONDICIONES: El Administrador pulse el botón RESET.

POSTCONDICIONES: Ninguna.

6.4.13.- DIAGRAMAS DE COLABORACIÓN

6.4.13.1.- Casos de Uso del Actor Administrador

INGRESAR PERMISOS DE ACCESO

DIAGRAMA DE COLABORACIÓN SOLICITA OPERACIÓN INGRESAR PERMISO DE ACCESO.

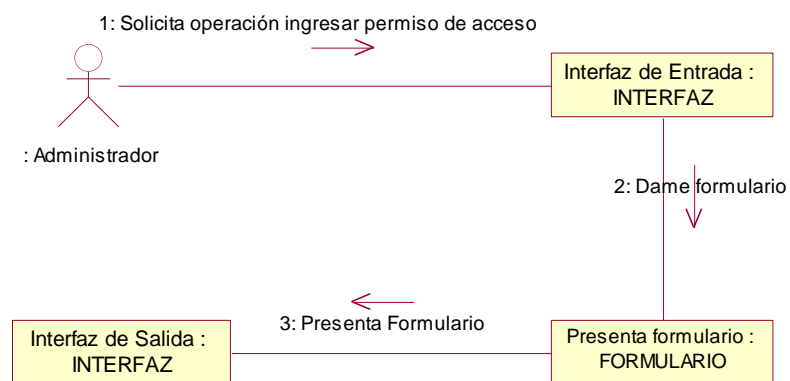


Fig. 6.20 Diagrama de Colaboración Solicitar Operación
Ingresar Permiso de Acceso

DIAGRAMA DE SECUENCIA SOLICITA OPERACIÓN INGRESAR PERMISO DE ACCESO.

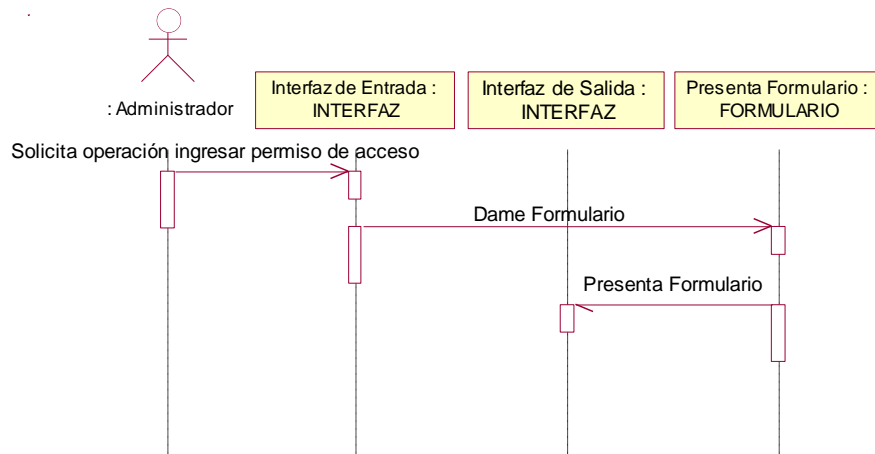


Fig. 6.21 Diagrama de Secuencia Solicitar Operación Ingresar Permiso de Acceso

DIAGRAMA DE COLABORACIÓN INGRESAR NOMBRE DEL PERMISO Y CONFIRMAR OPERACIÓN.

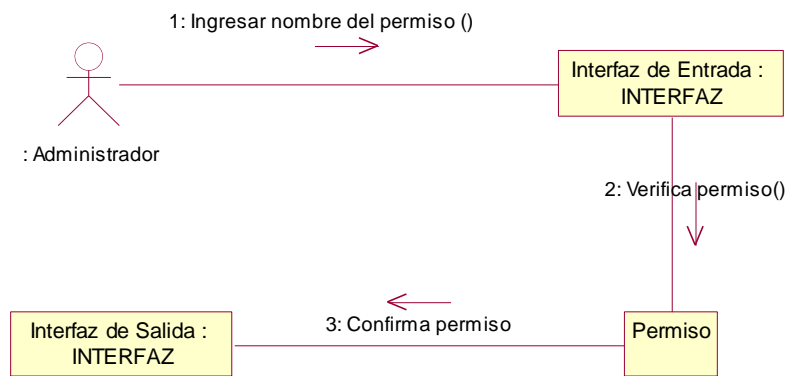


Fig. 6.22 Diagrama de Colaboración Ingresar Nombre del Permiso y Confirmar Operación

DIAGRAMA DE SECUENCIA INGRESAR NOMBRE DEL PERMISO Y CONFIRMAR OPERACIÓN.

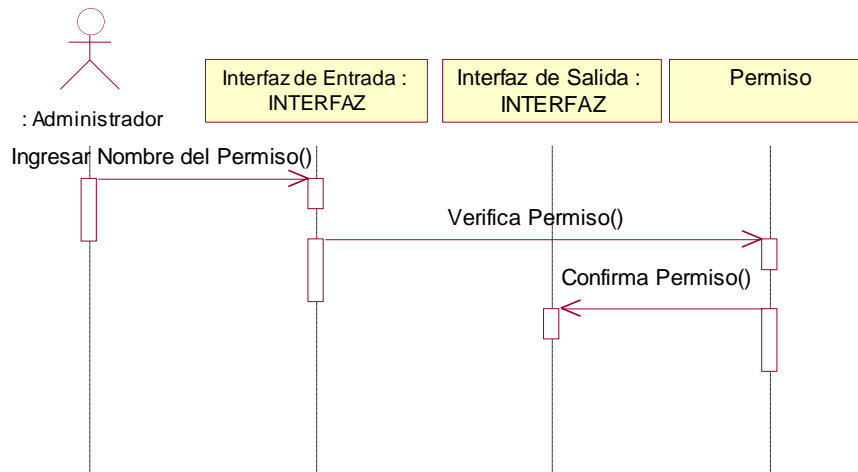


Fig. 6.23 Diagrama de Secuencia Ingresar Nombre del Permiso y Configurar Operación

ENCENDER LUZ HABITACIÓN1

DIAGRAMA DE COLABORACIÓN SOLICITA LA OPERACIÓN ENCENDER LUZ HABITACIÓN1.

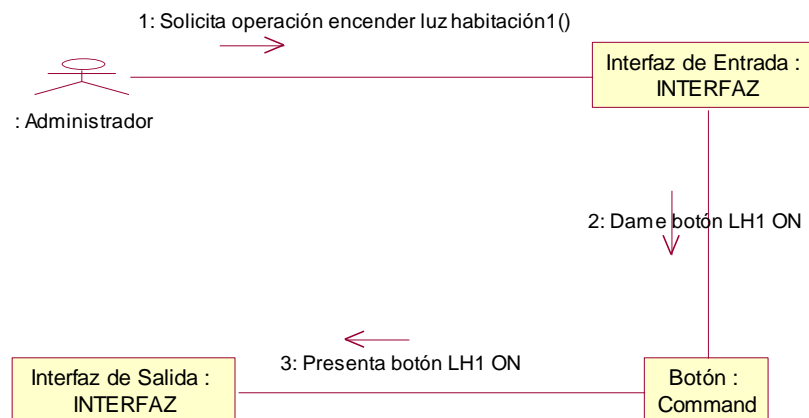


Fig. 6.24 Diagrama de Colaboración Solicitar Operación
Encender Luz Habitación1

DIAGRAMA DE SECUENCIA SOLICITA LA OPERACIÓN ENCENDER LUZ HABITACIÓN1.

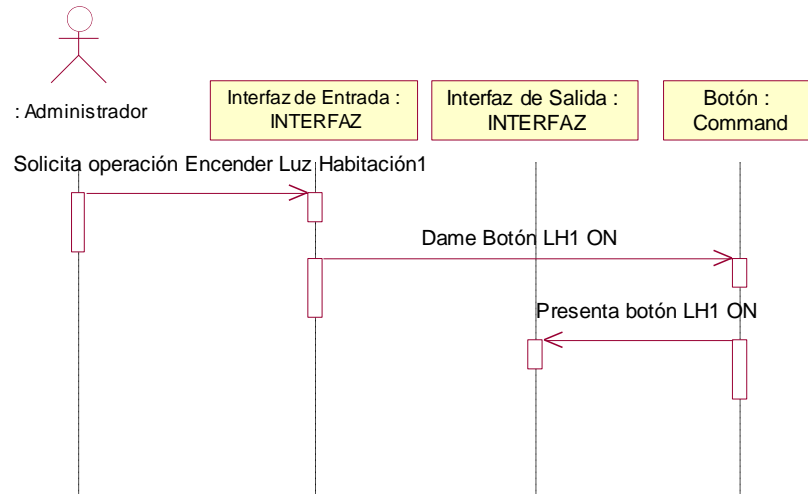


Fig. 6.25 Diagrama de Secuencia Solicitar Operación Encender Luz Habitación1

DIAGRAMA DE COLABORACIÓN PULSA EL BOTÓN LH1 ON

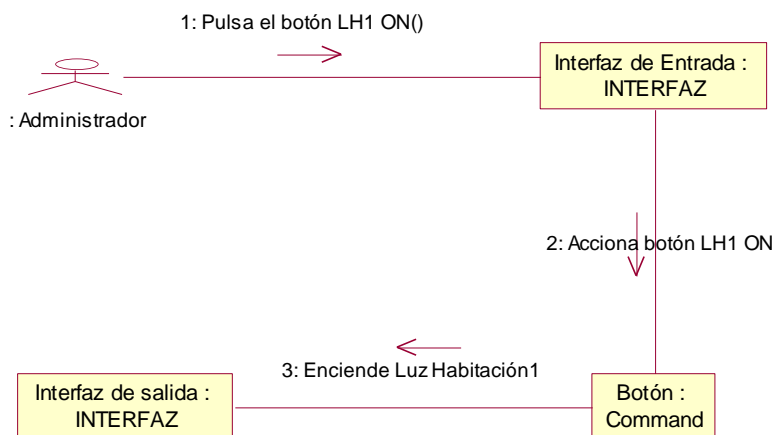


Fig. 6.26 Diagrama de Colaboración Pulsa Botón LH1

DIAGRAMA DE SECUENCIA PULSA EL BOTÓN LH1 ON

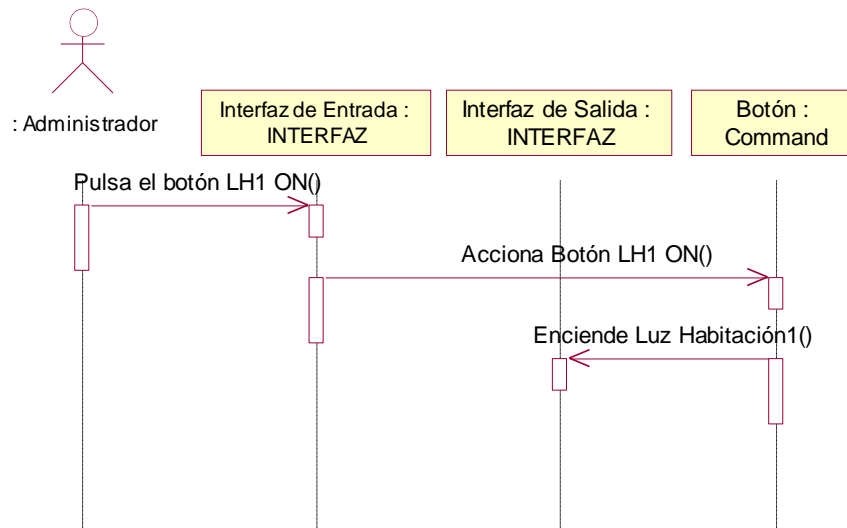


Fig. 6.27 Diagrama de Secuencia Pulsa el Botón LH1

APAGAR LUZ HABITACIÓN1

DIAGRAMA DE COLABORACIÓN SOLICITA LA OPERACIÓN APAGAR LUZ HABITACIÓN1.

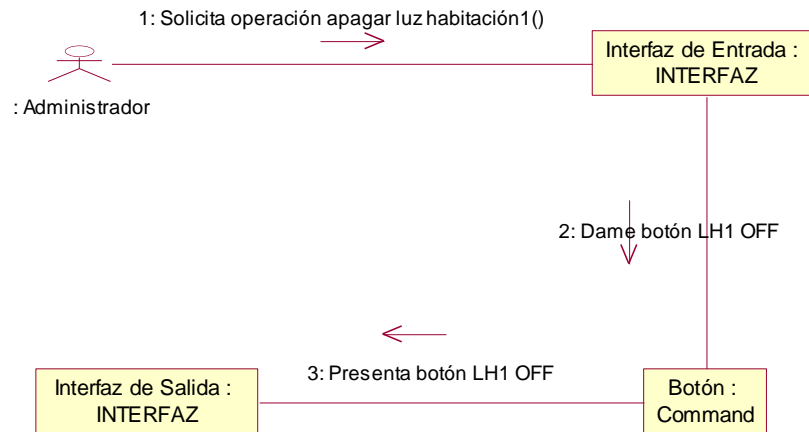


Fig. 6.28 Diagrama de Colaboración Solicita Operación Apagar Luz Habitación1

DIAGRAMA DE SECUENCIA SOLICITA LA OPERACIÓN APAGAR LUZ HABITACIÓN1.

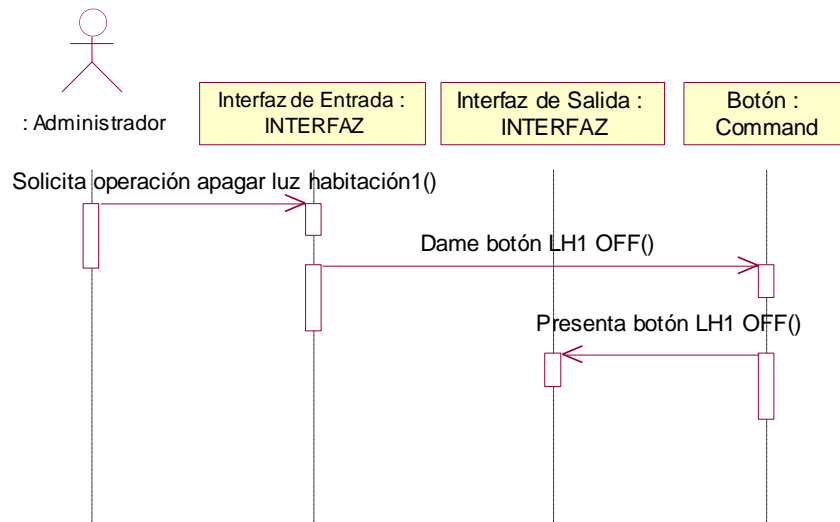


Fig. 6.29 Diagrama de Secuencia Solicita Operación Apagar Luz Habitación1

DIAGRAMA DE COLABORACIÓN PULSA EL BOTÓN LH1 OFF

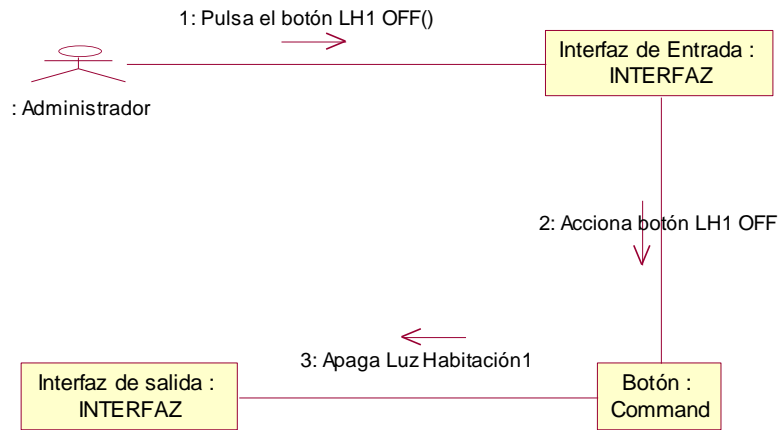


Fig. 6.30 Diagrama de Colaboración Pulsa Botón LH1OFF

DIAGRAMA DE SECUENCIA PULSA EL BOTÓN LH1 OFF

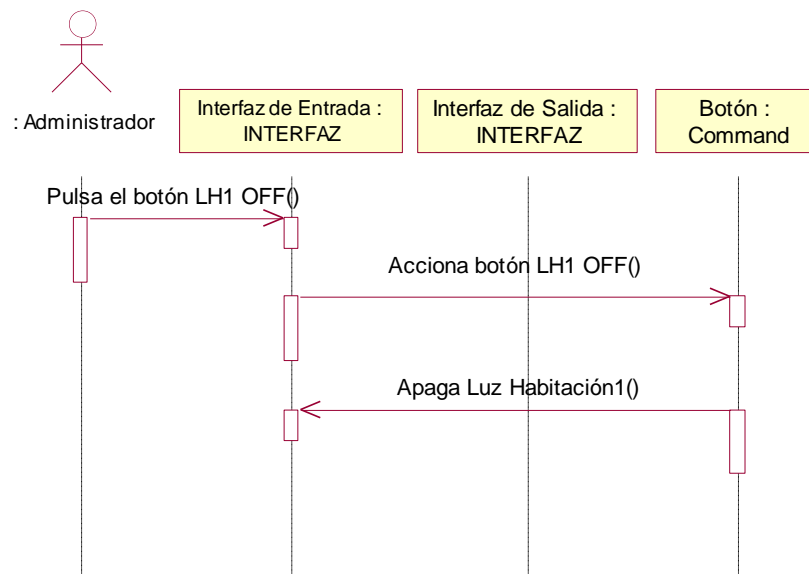


Fig. 6.31 Diagrama de Secuencia Pulsa el Botón LH1 OFF

VISUALIZAR ESTADO SENSOR PUERTA DE INGRESO1

DIAGRAMA DE COLABORACIÓN VISUALIZAR ESTADO DE SENSOR DE PUERTA DE INGRESO1

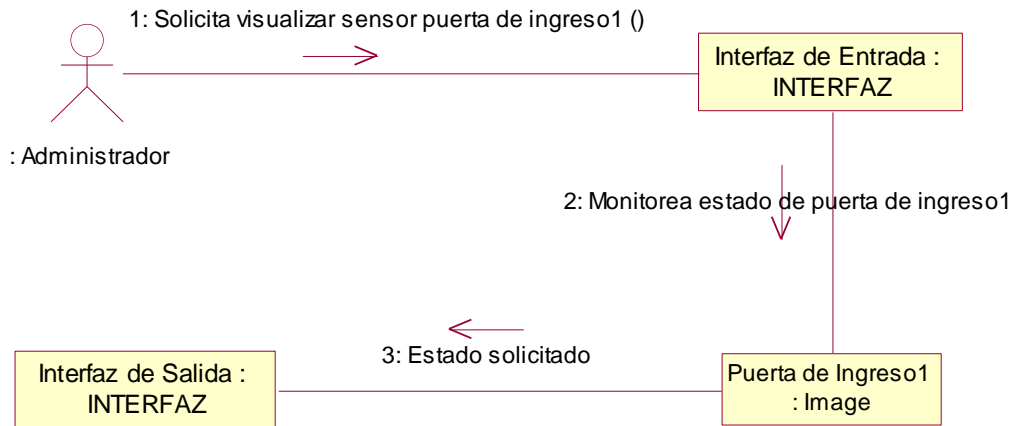


Fig. 6.32 Diagrama de Colaboración Visualizar Sensor Puerta de Ingreso1

DIAGRAMA DE SECUENCIA VISUALIZAR ESTADO DE SENSOR DE PUERTA DE INGRESO1.

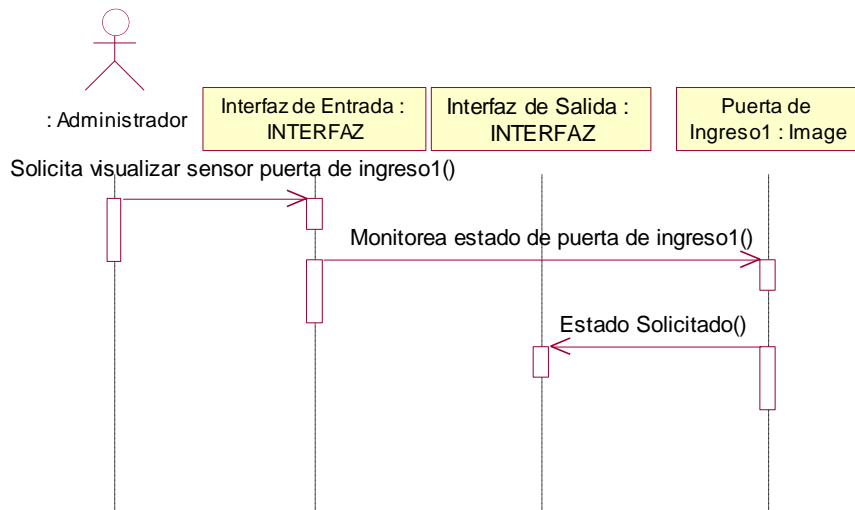


Fig. 6.33 Diagrama de Secuencia Visualizar Sensor Puerta de Ingreso1

DIAGRAMA DE COLABORACIÓN VISUALIZA ESTADO ACTUAL DE SENSOR DE PUERTA DE INGRESO1.

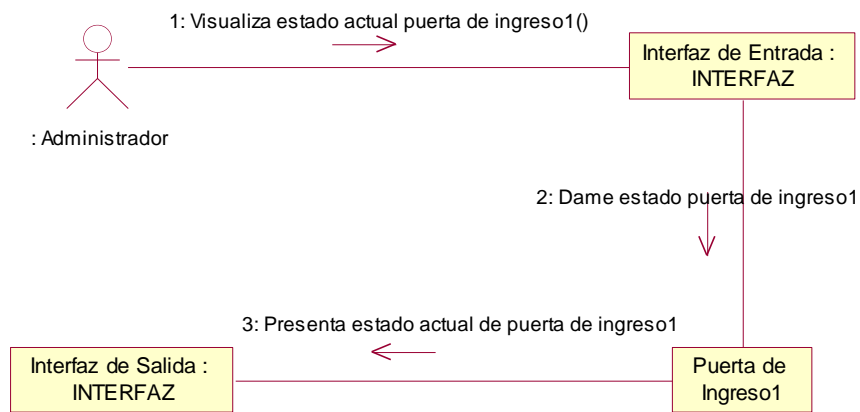


Fig. 6.34 Diagrama de Colaboración Visualizar Estado Actual Sensor Puerta de Ingreso1

DIAGRAMA DE SECUENCIA VISUALIZA ESTADO ACTUAL DE SENSOR DE PUERTA DE INGRESO1.

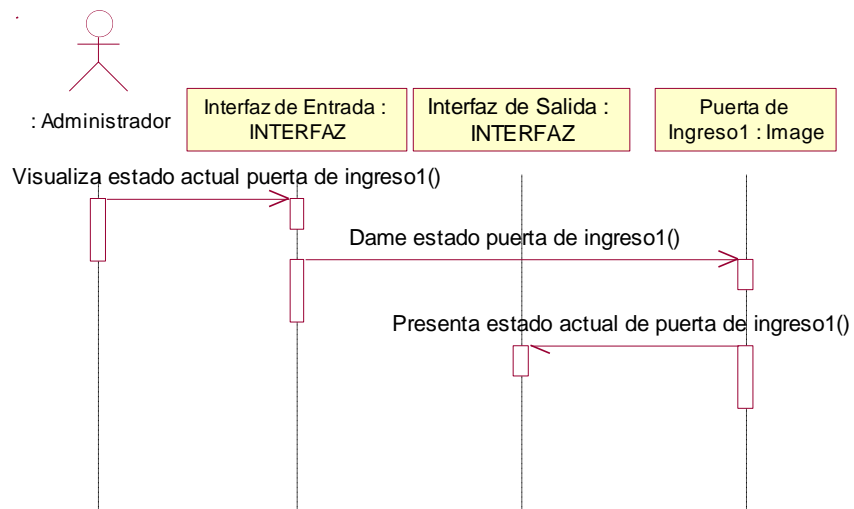


Fig. 6.35 Diagrama de Secuencia Visualizar Estado Actual Puerta de Ingreso1

VISUALIZAR ESTADO SENSOR VENTANA1

DIAGRAMA DE COLABORACIÓN VISUALIZAR ESTADO DE SENSOR DE VENTANA1.

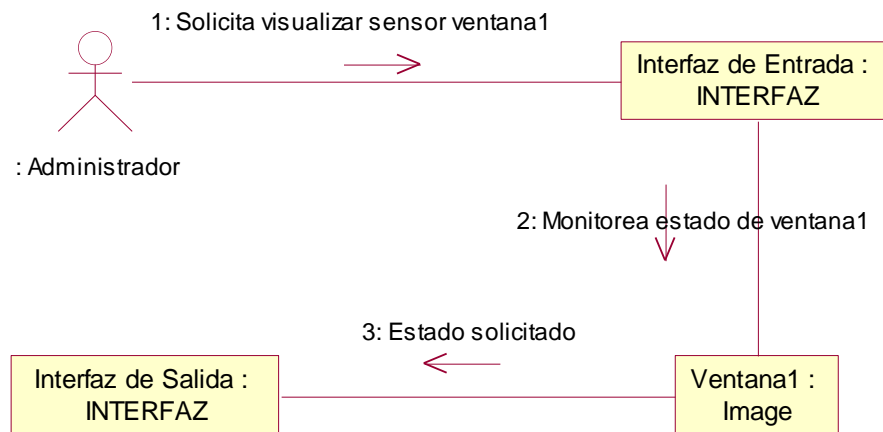


Fig. 6.36 Diagrama de Colaboración Visualizar Estado de Sensor Ventana1

DIAGRAMA DE SECUENCIA VISUALIZAR ESTADO DE SENSOR DE VENTANA1.

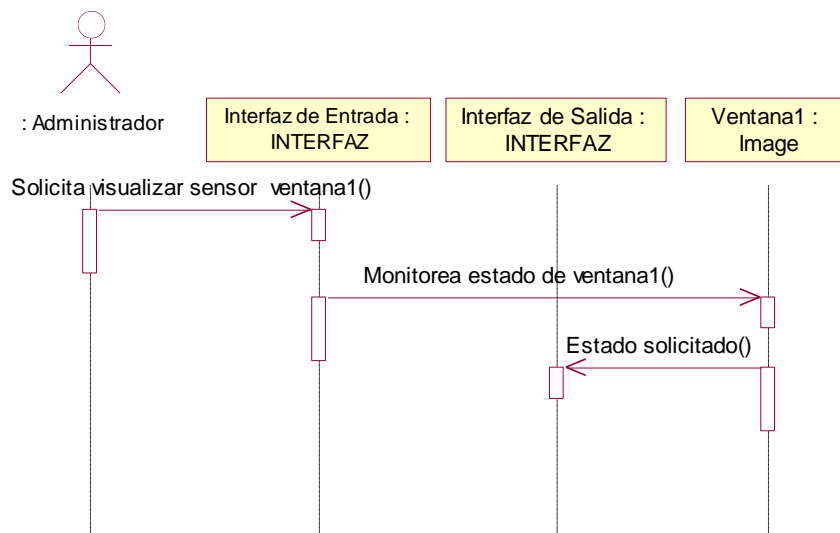


Fig. 6.37 Diagrama de Secuencia Visualizar Estado de Sensor Ventana1

DIAGRAMA DE COLABORACIÓN VISUALIZA ESTADO ACTUAL DE SENSOR DE VENTANA1.

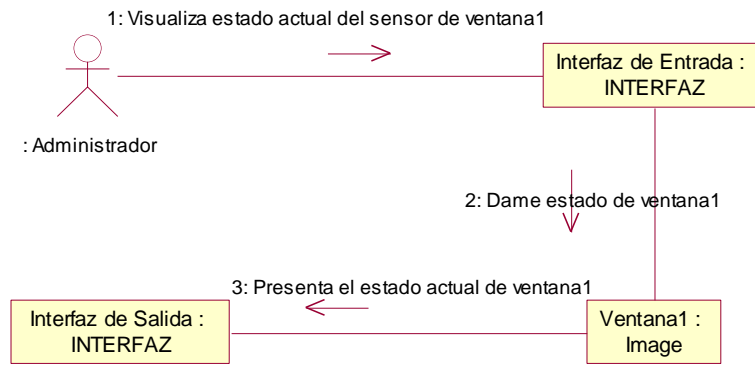


Fig. 6.38 Diagrama de Colaboración Visualizar Estado Actual Sensor de Ventana1

DIAGRAMA DE SECUENCIA VISUALIZA ESTADO ACTUAL DE SENSOR DE VENTANA1.

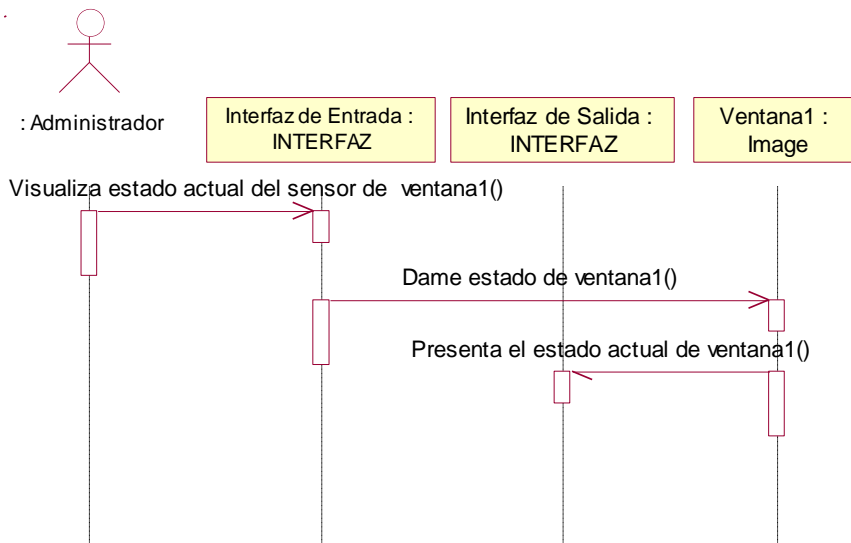


Fig. 6.39 Diagrama de Secuencia Visualizar Estado Actual Sensor de Ventana1

VISUALIZAR ESTADO SENSOR DE MOVIMIENTO

DIAGRAMA DE COLABORACIÓN VISUALIZAR ESTADO DE SENSOR DE MOVIMIENTO.

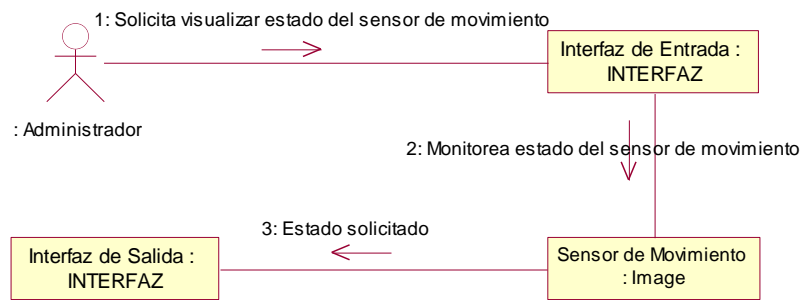


Fig. 6.40 Diagrama de Colaboración Visualizar Estado Sensor de Movimiento

DIAGRAMA DE SECUENCIA VISUALIZAR ESTADO DE SENSOR DE MOVIMIENTO.

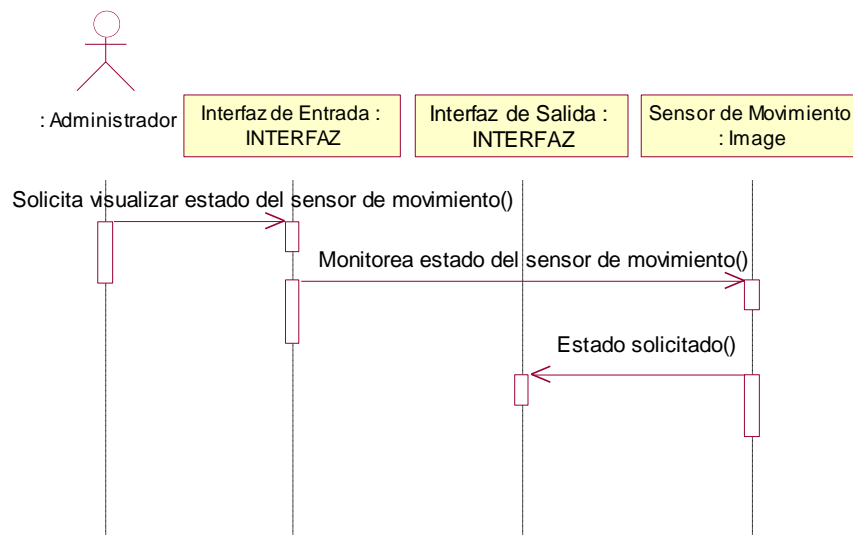


Fig. 6.41 Diagrama de Secuencia Visualizar Estado Sensor de Movimiento

DIAGRAMA DE COLABORACIÓN VISUALIZA ESTADO ACTUAL DE SENSOR DE MOVIMIENTO.

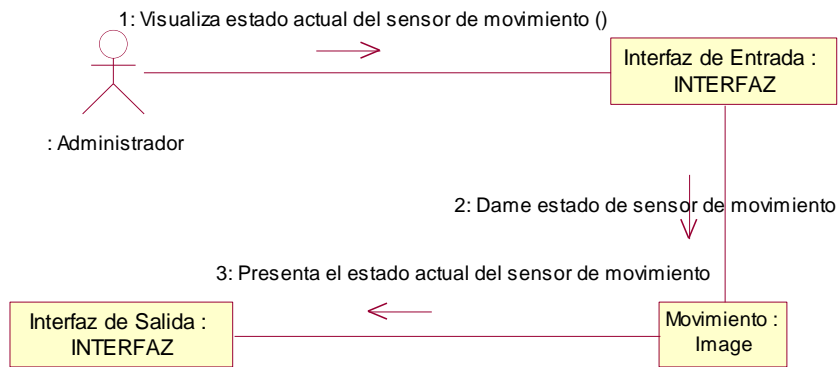


Fig. 6.42 Diagrama de Colaboración Visualizar Estado Actual Sensor de Movimiento

DIAGRAMA DE SECUENCIA VISUALIZA ESTADO ACTUAL DE SENSOR DE MOVIMIENTO.

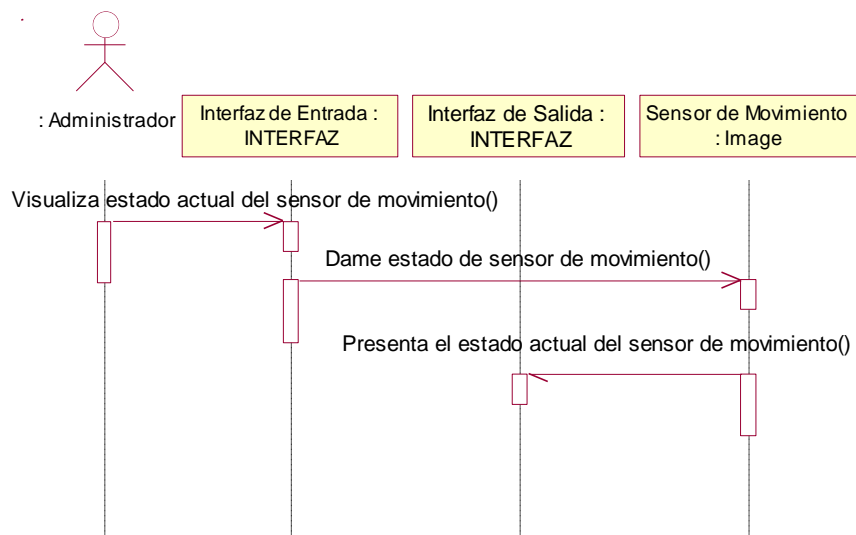


Fig. 6.43 Diagrama de Secuencia Visualizar Estado Actual Sensor de Movimiento

VISUALIZAR TEMPERATURA INTERNA DE LA RESIDENCIA

DIAGRAMA DE COLABORACIÓN VISUALIZAR TEMPERATURA INTERNA DE LA RESIDENCIA EN GRADOS CENTIGRADOS.

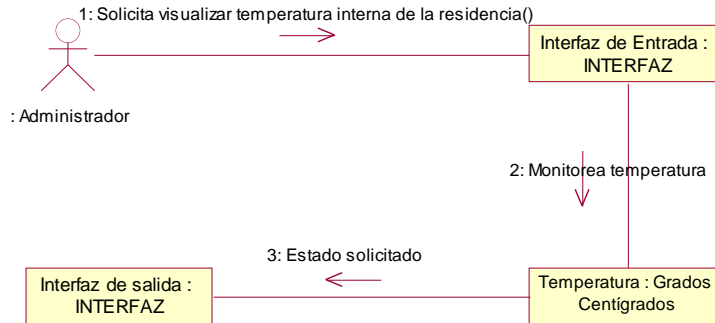


Fig. 6.44 Diagrama de Colaboración Visualizar Temperatura Interna de la Residencia en grados centígrados

DIAGRAMA DE SECUENCIA VISUALIZAR TEMPERATURA INTERNA DE LA RESIDENCIA EN GRADOS CENTIGRADOS.

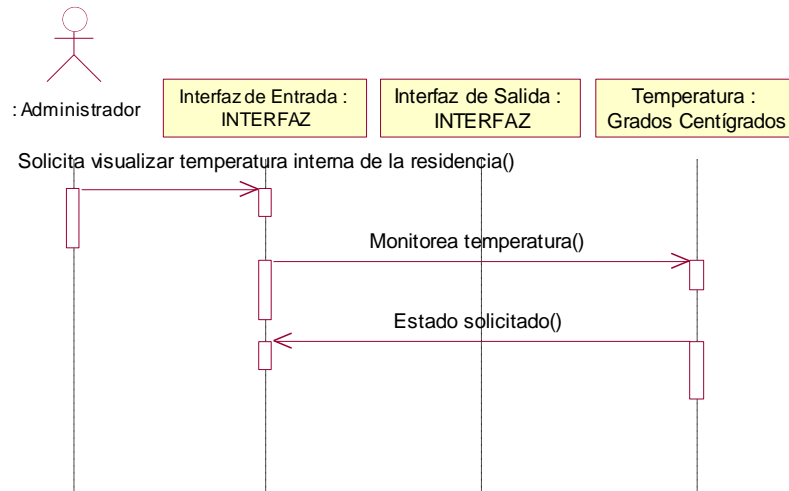


Fig. 6.45 Diagrama de Secuencia Visualizar Temperatura Interna de la Residencia en Grados Centígrados

DIAGRAMA DE COLABORACIÓN VISUALIZA TEMPERATURA INTERNA DE LA RESIDENCIA EN GRADOS CENTIGRADOS.

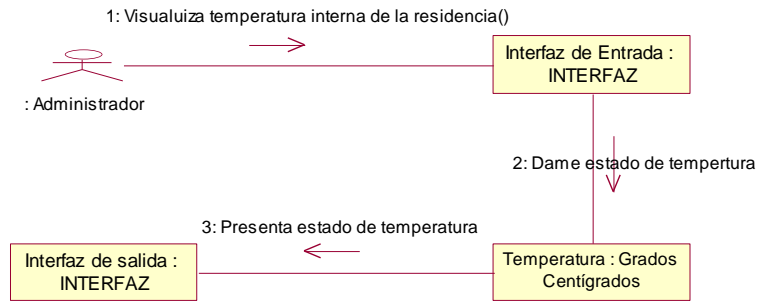


Fig. 6.46 Diagrama de Colaboración Visualiza Temperatura Interna de la Residencia en grados centígrados

DIAGRAMA DE SECUENCIA VISUALIZA TEMPERATURA INTERNA DE LA RESIDENCIA EN GRADOS CENTIGRADOS.

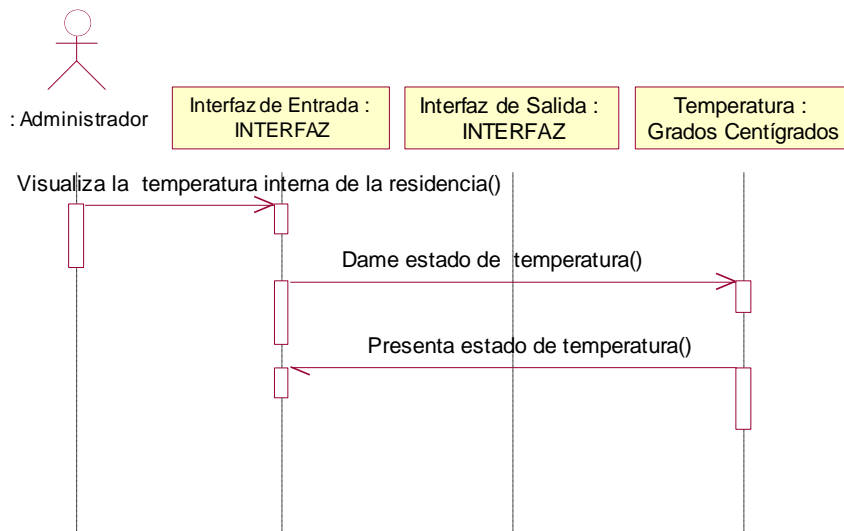


Fig. 6.47 Diagrama de Colaboración Visualiza Temperatura Interna de la Residencia en grados centígrados

INICIALIZAR ESTADOS

DIAGRAMA DE COLABORACIÓN SOLICITA LA OPERACIÓN INICIALIZAR ESTADOS.

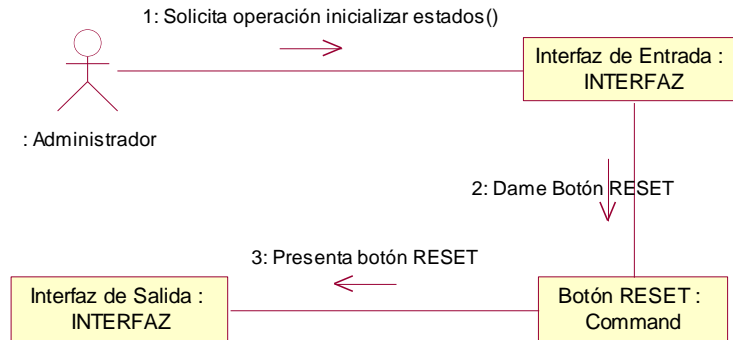


Fig. 6.48 Diagrama de Colaboración Solicita la Operación Inicializar Estados

DIAGRAMA DE SECUENCIA SOLICITA LA OPERACIÓN INICIALIZAR ESTADOS.

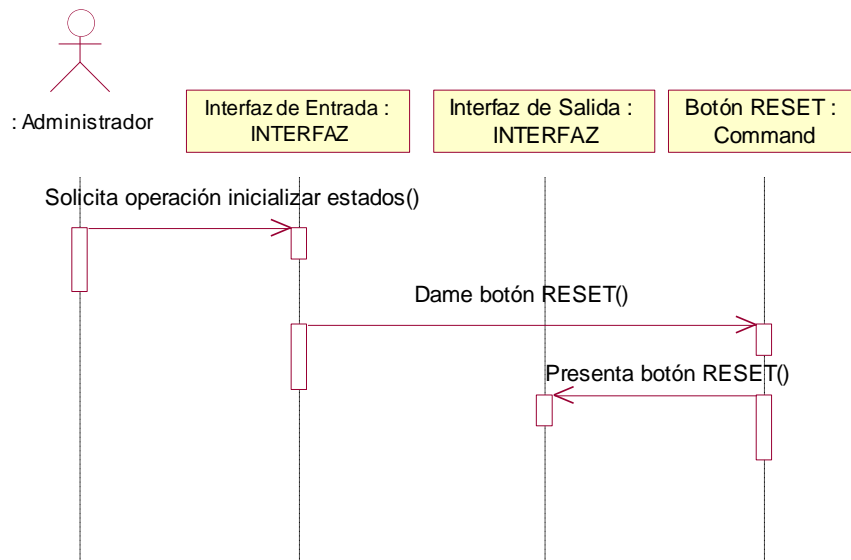


Fig. 6.49 Diagrama de Secuencia Solicita la Operación Inicializar Estados

DIAGRAMA DE COLABORACIÓN PULSA BOTÓN RESET.

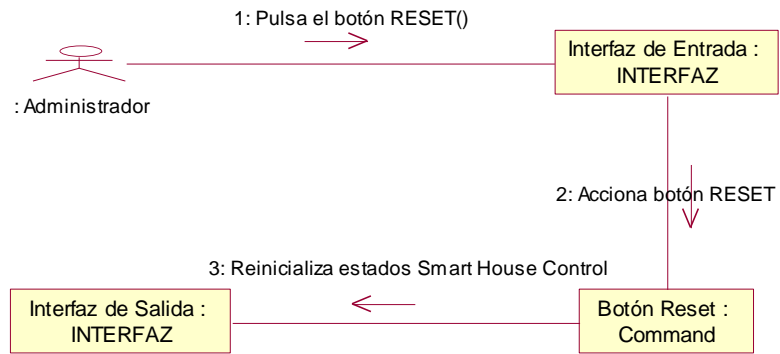


Fig. 6.50 Diagrama de Colaboración Pulsa Botón Reset

DIAGRAMA DE SECUENCIA PULSA BOTÓN RESET.

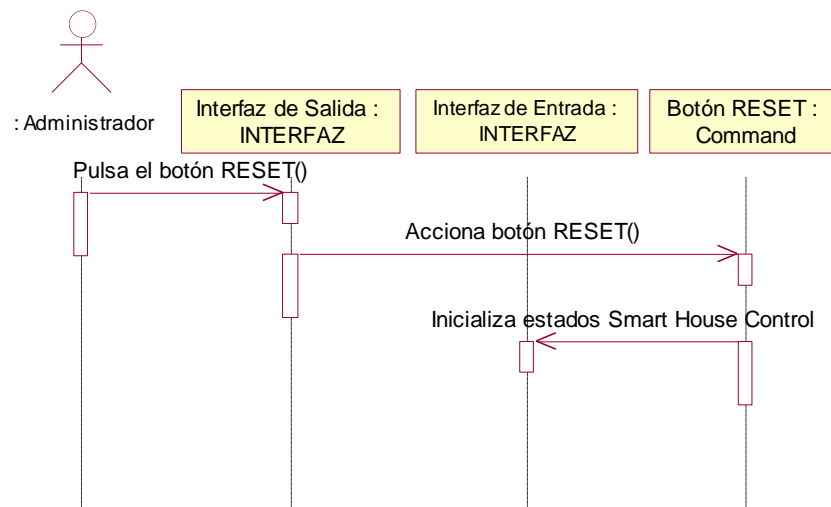


Fig. 6.51 Diagrama de Secuencia Pulsa Botón Reset

6.5.- IMPLEMENTACIÓN Y PRUEBAS

6.5.1.- CONFIGURACIÓN DE LA RED

6.5.1.1.- Configuración del Servidor

Procedemos ingresando a Panel de Control y posteriormente a Red, en Identificación establecemos el en Nombre del Pc: Redes8 y en Grupo de Trabajo: WORKGROUP para nuestro caso. Posteriormente en Configuración escogemos TCP/IP ->SiS 900 PCI Fast Ethernet Adapter, damos clic en propiedades; seleccionamos especificar una dirección IP. Establecemos en Dirección IP: 192.168.100.8 y en Máscara de subred: 255.255.255.0

Pulsamos en Configuración DNS, y seleccionamos activar DNS. Establecemos en Host: domotica, en Dominio: espe.edu.ec y en Orden de búsqueda del Servidor DNS: 192.188.58.8 y pulsamos agregar, salimos de la configuración aceptando todos los cambios realizados. Luego de lo cual se reiniciará el sistema. (Ver Anexo 3.1)

6.5.1.2.- Configuración del Cliente

Procedemos ingresando a Panel de Control y posteriormente a Red, en Identificación establecemos el en Nombre del Pc: Redes7 y en Grupo de Trabajo: WORKGROUP para nuestro caso. Posteriormente en Configuración escogemos TCP/IP ->SiS 900 PCI Fast Ethernet Adapter, damos clic en

propiedades; seleccionamos especificar una dirección IP. Establecemos en Dirección IP: 192.168.100.7 y en Máscara de subred: 255.255.255.0

Pulsamos en Configuración DNS, y seleccionamos activar DNS. Establecemos en Host: domotica, en Dominio: espe.edu.ec y en Orden de búsqueda del Servidor DNS: 192.188.58.8 y pulsamos agregar, salimos de la configuración aceptando todos los cambios realizados. Luego de lo cual se reiniciará el sistema. (Ver Anexo 3.2)

6.5.2.- Configuración Del Servidor Web

Para configurar el servidor y todos sus ficheros de configuración, es necesario editar el archivo httpd.conf que se encuentra en c:\Archivos de Programa\ApacheGroup\Apache\conf\, o a su vez en Inicio -> Programas -> Apache HTTP SERVER -> Configure Apache Server-> Edit the Apache httpd.conf Configuration File. (Ver Anexo 4.1)

- **ServerType:** Especifica el tipo de servidor que puede ser standalone o inetd (es un soporte para Unix)
 - Asigne el tipo de servidor
 - ServerType standalone
- **ServerRoot:** Es el path donde está instalado Apache y de él cuelgan los distintos subdirectorios como el conf y otros.
 - ServerRoot "C:/Archivos de programa/Apache Group/Apache"

- **ServerName:** Es el nombre del servidor, puede ser localhost, pero se puede asignar otro nombre.
 - ServerName 192.188.58.8
- **Document Root:** Es el path donde se ubican las páginas de la web.
 - DocumentRoot "C:/Archivos de programa/Apache Group/Apache/htdocs"
- **DirectoryIndex:** Es el directorio donde se almacenará la página con la que iniciará al momento de ejecutar la aplicación.
 - DirectoryIndex index.html
- Se debe reiniciar la máquina para que todos los cambios realizados tengan efecto.
- Para subir el servidor Apache se debe ejecutar Start Apache que se encuentra en la ubicación asignada en la instalación, en nuestro caso Inicio -> Programas -> Apache HTTP SERVER -> Start Apache in Console.
- No olvide ubicar sus páginas en c:\Archivos de programa\ApacheGroup\Apache\htdocs.

6.5.3.- PRUEBAS

6.5.3.1.- Especificaciones de Seguridad.

Se mencionó la utilización de claves para el acceso y desactivación de alarma, las cuales permitirán mantener seguridad, permitiendo

que únicamente el usuario que haya sido autorizado pueda tener acceso al monitoreo y control del prototipo de Automatización Doméstica.

6.5.3.2.- Plan de Pruebas

Las pruebas a emplearse a continuación permitirán verificar y comprobar el cumplimiento de los requisitos establecidos con anterioridad. En la siguiente tabla presentamos los requisitos, con la operación a ser probada y los tipos de pruebas a practicarse. (Ver Anexo 6)

Requisitos	Operación	Prueba
Gestión Permiso de Acceso	Verificar permiso de acceso	-Valores de Error -Valores Imposibles
Gestión Encendido de Luces	<ul style="list-style-type: none"> - Encender Luz Habitación1 - Encender Luz Habitación2 - Encender Luz Habitación3 - Encender Luz Externa 	-Valores de Error -Valores Imposibles
Gestión Apagado de Luces	<ul style="list-style-type: none"> - Apagar Luz Habitación1 - Apagar Luz Habitación2 - Apagar Luz Habitación3 - Apagar Luz Externa 	-Valores de Error -Valores Imposibles
Gestión Visualizar Estado de Sensor de Puertas	<ul style="list-style-type: none"> - Monitorea Estado de Sensor Puerta de Ingreso1 - Monitorea Estado de Sensor 	-Valores de Error -Valores Imposibles

	Puerta de Ingreso2	
Gestión Visualizar Estado de Ventanas	<ul style="list-style-type: none"> - Monitorea Estado de Sensor de Ventana1 - Monitorea estado de Sensor de Ventana2 	<ul style="list-style-type: none"> -Valores de Error -Valores Imposibles
Gestión Visualizar Estado de Sensor de Movimiento	<ul style="list-style-type: none"> - Monitorea Estado de Sensor de Movimiento 	<ul style="list-style-type: none"> -Valores de Error -Valores Imposibles
Gestión Visualizar Temperatura Interna de la Residencia	<ul style="list-style-type: none"> - Monitorea la Temperatura y devuelve el resultado en grados centígrados 	<ul style="list-style-type: none"> -Valores de Error -Valores Imposibles
Gestión Activar Alarma	<ul style="list-style-type: none"> - Enciende alarma en caso de haber intruso y peligro de incendio. 	<ul style="list-style-type: none"> -Valores de Error -Valores Imposibles
Gestión Desactivar Alarma	<ul style="list-style-type: none"> - Desactiva alarma 	<ul style="list-style-type: none"> -Valores de Error -Valores Imposibles
Gestión Inicializar Estados	<ul style="list-style-type: none"> - Inicializa estados de Luces y demás Sensores . 	<ul style="list-style-type: none"> -Valores de Error -Valores Imposibles

Tabla 6.15 Requisitos, Operaciones y Pruebas

6.5.4.- Manual De Operaciones.

En el manual de operaciones consta una breve introducción de el uso del software SHC, el manejo de cada uno de los botones, como la explicación del funcionamiento del mismo. (Ver Anexo 7)

VII.- CONCLUSIONES Y RECOMENDACIONES

7.1.- CONCLUSIONES

- Gracias a los beneficios de UML, se ha podido desarrollar el Software con el ciclo interactivo incremental, lo cual ha permitido realizar los cambios necesarios que se han ido dando a lo largo de la construcción de SHC, cumpliendo a cabalidad con nuestros objetivos.
- Se consiguió diseñar e implementar un Software de Control de Iluminación y Activación de un Sistema de Seguridad capaz de ser controlado en un ambiente interno y externo. Además de la construcción de tarjetas de circuito impreso, que permiten la interacción del computador con los demás elementos del diseño del prototipo.
- Debido a que la aplicación no requiere de tiempos extremadamente exactos de respuesta, Visual Basic 6.0 se constituyó en una de nuestras mejores opciones de programación.
- SHC permite operar en un ambiente interno(dentro de la residencia), a través del Software Smart House Control PC, creado para un solo computador.

- SHC también permite operar en un ambiente externo, es decir cuando se requiera verificar el estado de la residencia a través de la Web (para nuestro caso una Intranet). En este caso utilizaremos Smart House Control Server que correrá en el Servidor y Smart House Control Cliente que correrá en el Cliente.

- El diseño de las placas de circuito impreso bien puede ser utilizado para la puesta en ejecución de un Sistema de Control de Iluminación y Seguridad real, con cierto análisis de condiciones y requerimientos de los componentes electrónicos a utilizarse, como pueden ser: la variación de la resistencia y voltaje.

- Se ha logrado aprovechar el puerto paralelo de la impresora para la E / S de datos, el mismo que en el Servidor se encuentra conectado al prototipo de la residencia y en el Cliente debe permanecer sin conexión a impresora alguna, puesto que una desventaja es el no poder utilizar dicho puerto aunque este aparentemente se encuentre vacío. Se podrá usar el puerto para la impresora una vez finalizado el programa.

- En el puerto paralelo trabajamos con tres registros: 378H para salida de datos, 389H para entrada, y 37A bidireccional. Existen diferentes modos en los cuales opera el puerto paralelo así tenemos: ECP, EPP, SPP; para nuestro caso el modo en el que operaremos dicho puerto es el SPP

(Standar Parallel Port); debido a que este modo permite operar el registro 37A en forma bidireccional E / S .

- Con los componentes que tenemos en nuestro medio, y con el ingenio y la capacidad que se puede desarrollar, se ha demostrado que podemos ser los creadores e impulsores de nueva tecnología, a menor precio y alta competitividad.

7.2.- RECOMENDACIONES

- Se recomienda a las autoridades de la facultad incluir en la currícula académica materias que amplíen el conocimiento en el área electrónica, para complementar el perfil del Ingeniero de Sistemas de la Escuela Politécnica del Ejército, lo cual nos haría más competentes a nivel de otras instituciones.
- Recomendamos crear un Grupo de Desarrollo de Proyectos Mixtos, en donde se involucre a maestros y estudiantes de cualquier especialidad, a realizar temas de investigación conjunta, lo que permitirá adquirir y aportar conocimientos.
- Se recomienda establecer convenios con Empresas de renombre del Centro del País, para que se autorice el poder elaborar tesis en sus dependencias, ya que es difícil conseguir auspicio personalmente.
- Recomendamos que la presente tesis, pueda constituirse en un punto de partida para una nueva Tesis o que la misma pueda ser implementada realmente.
- Se recomienda tener cuidado con la conservación del prototipo, para que en lo posterior se pueda dar a conocer los beneficios de una residencia automatizada .

- Para realizar las conexiones de las tarjetas con el prototipo, recomendamos seguir las instrucciones del manual, así como para la ejecución de las aplicaciones.
- Se recomienda manipular los circuitos integrados de las placas con cuidado debido a que si no se tiene el debido conocimiento puede incidir en la inoperabilidad del mismo.
- El cable de la impresora no puede ser reemplazado por un cable normal debido a que está específicamente diseñado para interactuar con el prototipo, por tanto recomendamos tener cuidado de extraviar todos los componentes del prototipo.
- Para aplicar este proyecto en un sistema de seguridad real, es indispensable contar con un UPS; en caso de cortes de energía eléctrica.

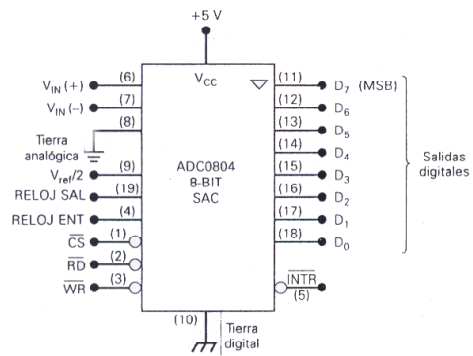
BIBLIOGRAFÍA

- Zelenousky Ricardo, "PC - IBM para Ingenieros"
- Zambrano Carlos, "Edificios Inteligentes"

DIRECCIONES DE INTERNET

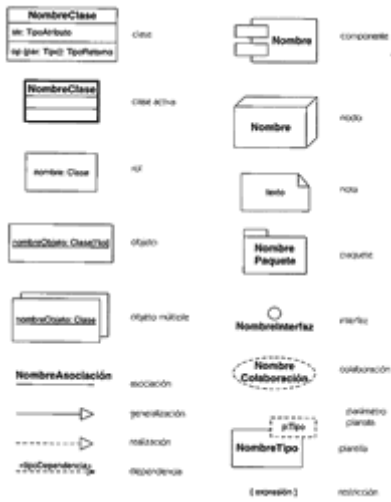
- <http://www.domotica.net>
- <http://www.monografias.com/trabajos5/insof/insof.shtml>
- <http://www.cs.ualberta.ca/~pfiguero/soo/uml/>
- <http://www.creangel.com/uml/resumen.html>
- <http://www.domosys.com>
- <http://www.domos.com>
- <http://www.monografias.com>
- <http://www.soloprogramadores.com>
- <http://www.informacionnormas/iso.html>
- <http://www.escproducts.com>

ADC0804.- Conversor análogo – digital (CMOS)

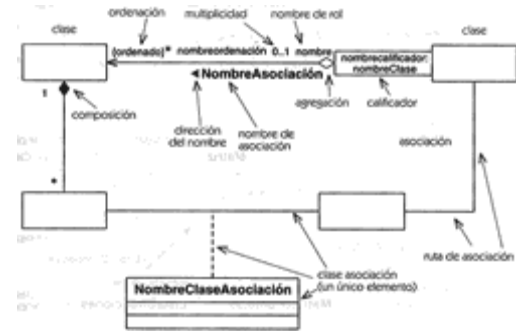


Resumen de Notación

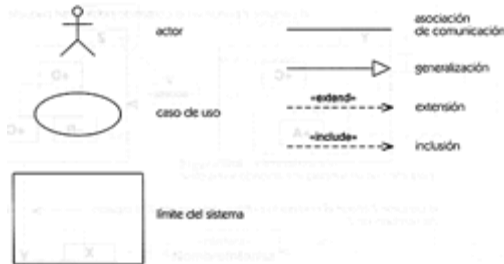
Iconos utilizados en UML



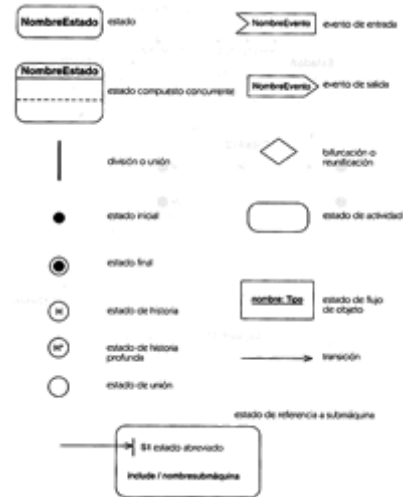
Asociaciones en Clases y Objetos



Iconos del Caso de Uso



Iconos del Diagrama de Estados

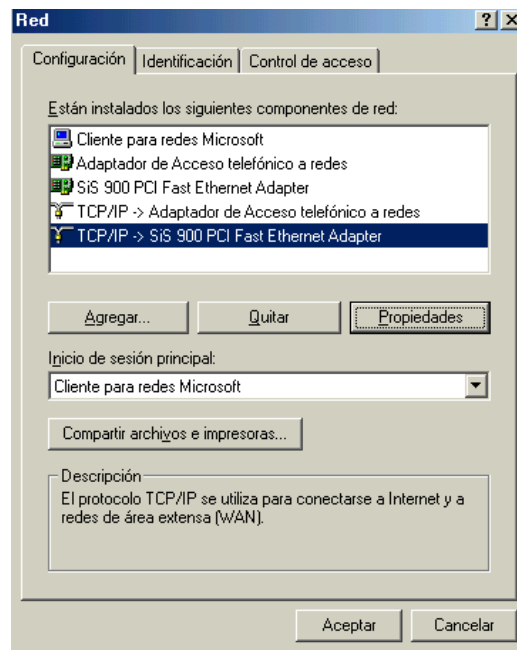


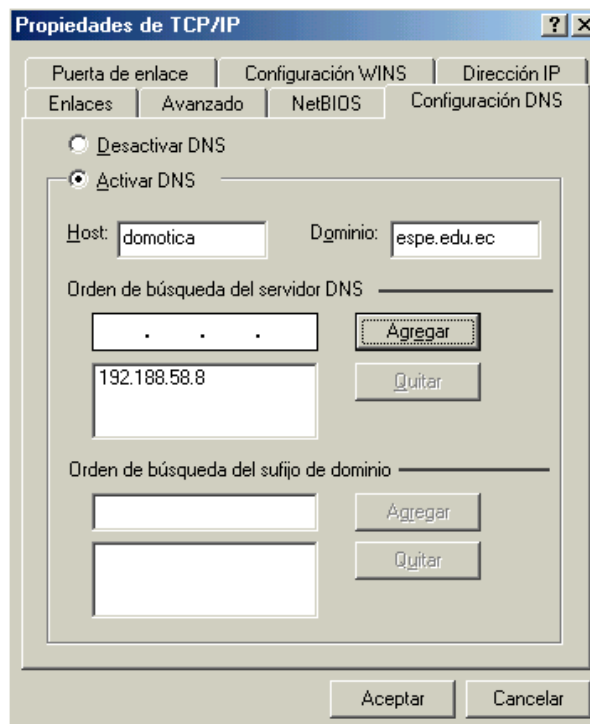
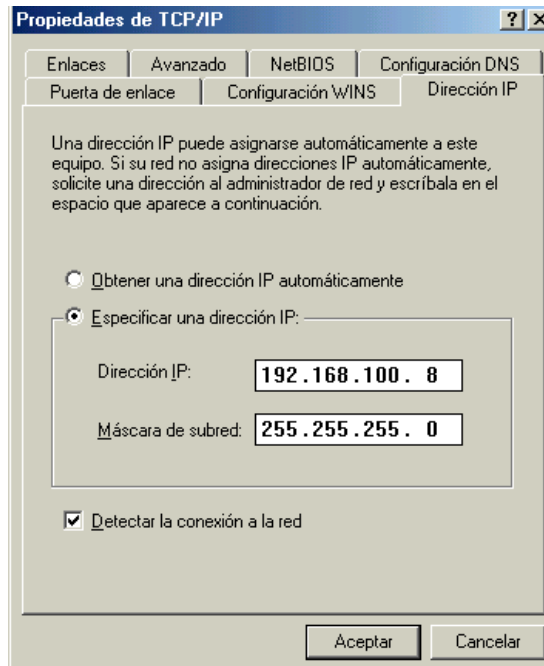
Iconos en Mensajes

-  mensaje secuencial
-  mensaje asíncrono
-  llamada

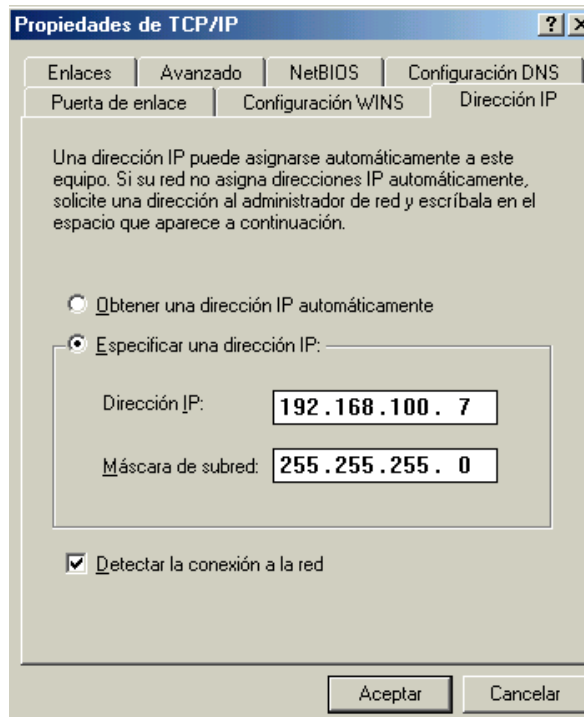
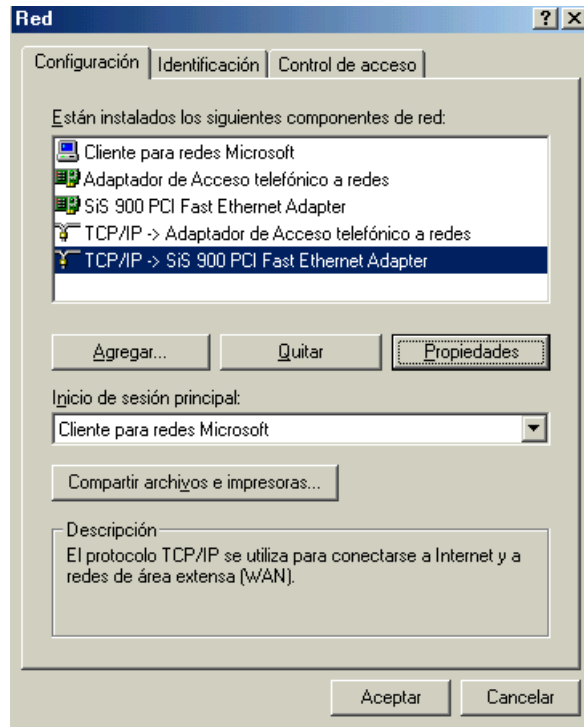
CONFIGURACIONES CLIENTE - SERVIDOR

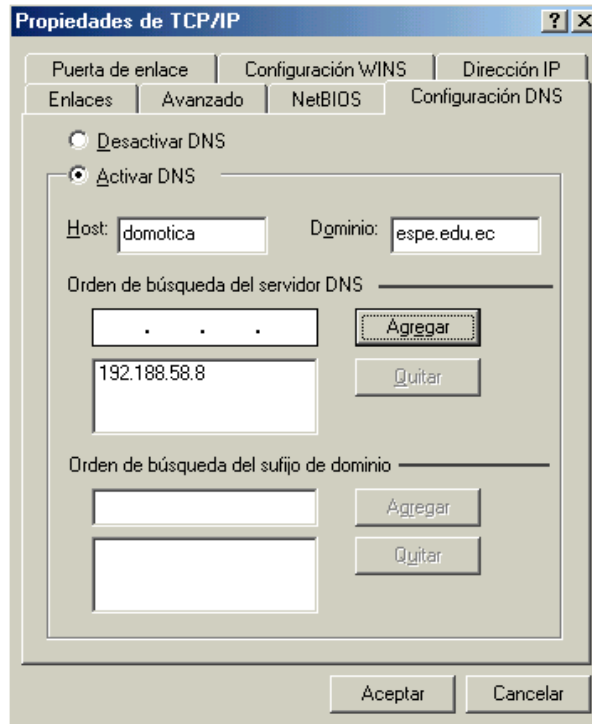
3.1.- CONFIGURACION DEL SERVIDOR





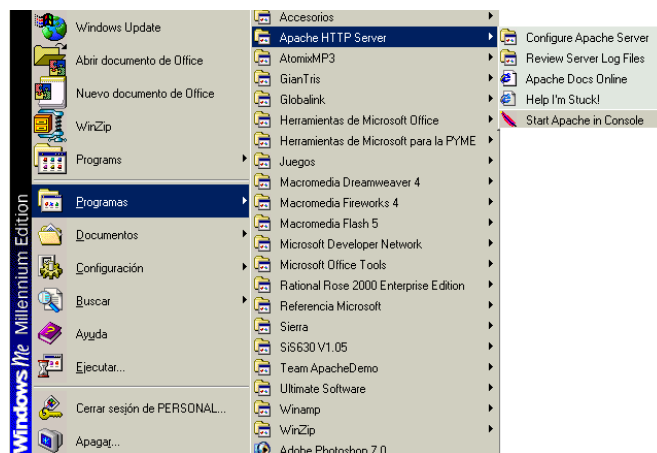
3.2.- CONFIGURACION DEL CLIENTE





CONFIGURACIÓN DEL SERVIDOR DE INTERNET

4.1.- CONFIGURACION DEL SERVIDOR WEB



Private Habilitador As Boolean
Private StrobeBeep As Boolean
Private DatoTemp As Integer
Private Alarma As Integer
Private Boton_Status As Integer

```
/*  
* NOMBRE DEL PROYECTO: Smart House Control Client *  
/*  
* FUNCIÓN: Private Sub btnLE_Click() FECHA: 10-12-2003 *  
/*  
* PROPÓSITO: Activar Luz Externa en el prototipo, visualizar en el cliente y enviar *  
* dato del puerto al servidor *  
/*  
* ESTRUCTURA DE LA FUNCIÓN *  
/*
```

```
Private Sub btnLE_Click()  
Boton_Status = 7  
Dato378 = Inp(&H378)  
Dato378 = Dato378 Or 8  
enviar = Dato378  
Cliente.SendData Boton_Status  
Cliente.SendData enviar  
Out &H378  
imgonLE.Visible = True  
imgoffLE.Visible = False  
btnLE.Enabled = False  
btnLEoff.Enabled = True  
End Sub
```

```
/*  
* NOMBRE DEL PROYECTO: Smart House Control Client *  
/*
```

/*

*/

* FUNCIÓN: Private Sub btnLEoff_Click() FECHA: 10-12-2003 *

/*

*/

* PROPÓSITO: Desactivar Luz Externa en el prototipo, visualizar en el cliente y *
* enviar dato del puerto al servidor *

/*

*/

* ESTRUCTURA DE LA FUNCIÓN *

/*

*/

Private Sub btnLEoff_Click()

Boton_Status = 8

Dato378 = Inp(&H378)

Dato378 = Dato378 And 7

enviar = Dato378

Cliente.SendData Boton_Status

Cliente.SendData enviar

Out &H378

imgonLE.Visible = False

imgoffLE.Visible = True

btnLE.Enabled = True

btnLEoff.Enabled = False

End Sub

/*

*/

* NOMBRE DEL PROYECTO: Smart House Control Client *

/*

*/

* FUNCIÓN: Private Sub btnLH1_Click() FECHA: 10-12-2003 *

/*

*/

* PROPÓSITO: Activar Luz Habitación1 en el prototipo, visualizar en el cliente y *
* enviar dato del puerto al servidor *

/*

*/

* ESTRUCTURA DE LA FUNCIÓN *

/*

*/

```

Private Sub btnLH1_Click()
If Cliente.State = sckConnected Then
Boton_Status = 1
Dato378 = Inp(&H378)
Dato378 = Dato378 Or 1
enviar = Dato378
Cliente.SendData Boton_Status
Cliente.SendData enviar
Out &H378
imgonLH1.Visible = True
imgoffLH1.Visible = False
btnLH1.Enabled = False
btnLH1off.Enabled = True
End If
End Sub

```

```

/*****
* NOMBRE DEL PROYECTO: Smart House Control Client *
/*****
* FUNCIÓN: Private Sub btnLH1off_Click() FECHA: 10-12-2003 *
/*****
* PROPÓSITO: Desactivar Luz Habitación1 en el prototipo, visualizar en el cliente y *
* enviar dato del puerto al servidor *
/*****
* ESTRUCTURA DE LA FUNCIÓN *
/*****

```

```

Private Sub btnLH1off_Click()
If Cliente.State = sckConnected Then
Boton_Status = 2
Dato378 = Inp(&H378)

```

```
Dato378 = Dato378 And 14
enviar = Dato378
Cliente.SendData Boton_Status
Cliente.SendData enviar
Out &H378, Dato378
imgonLH1.Visible = False
imgoffLH1.Visible = True
btnLH1.Enabled = True
btnLH1off.Enabled = False
End If
End Sub
```

```
/*
* NOMBRE DEL PROYECTO: Smart House Control Client
*/
/*
* FUNCIÓN: Private Sub btnLH2_Click() FECHA: 10-12-2003
*/
* PROPÓSITO: Activar Luz Habitación2 en el prototipo, visualizar en el cliente y
* enviar dato del puerto al servidor
*/
* ESTRUCTURA DE LA FUNCIÓN
*/
Private Sub btnLH2_Click()
Boton_Status = 3
Dato378 = Inp(&H378)
Dato378 = Dato378 Or 2
enviar = Dato378
Cliente.SendData Boton_Status
Cliente.SendData enviar
Out &H378
imgonLH2.Visible = True
imgoffLH2.Visible = False
```

```
btnLH2.Enabled = False
btnLH2off.Enabled = True
End Sub
```

```
/*
*****
*/
```

```
* NOMBRE DEL PROYECTO: Smart House Control Client *
```

```
/*
*****
*/
```

```
* FUNCIÓN: Private Sub btnLH2off_Click() FECHA: 10-12-2003 *
```

```
/*
*****
*/
```

```
* PROPÓSITO: Desactivar Luz Habitación2 en el prototipo, visualizar en el cliente y *
```

```
* enviar dato del puerto al servidor *
```

```
/*
*****
*/
```

```
* ESTRUCTURA DE LA FUNCIÓN *
```

```
/*
*****
*/
```

```
Private Sub btnLH2off_Click()
```

```
Boton_Status = 4
```

```
Dato378 = Inp(&H378)
```

```
Dato378 = Dato378 And 13
```

```
enviar = Dato378
```

```
Cliente.SendData Boton_Status
```

```
Cliente.SendData enviar
```

```
Out &H378
```

```
imgonLH2.Visible = False
```

```
imgoffLH2.Visible = True
```

```
btnLH2.Enabled = True
```

```
btnLH2off.Enabled = False
```

```
End Sub
```

```
/*
*****
*/
```

```
* NOMBRE DEL PROYECTO: Smart House Control Client *
```

```
/*
*****
*/
```

```
* FUNCIÓN: Private Sub btnLH3_Click() FECHA: 10-12-2003 *
```

```

/*****/
* PROPÓSITO: Activar Luz Habitación3 en el prototipo, visualizar en el cliente y *
* enviar dato del puerto al servidor *

```

```

/*****/
*
* ESTRUCTURA DE LA FUNCIÓN *
/*****/

```

```

Private Sub btnLH3_Click()
Boton_Status = 5
Dato378 = Inp(&H378)
Dato378 = Dato378 Or 4

enviar = Dato378
Cliente.SendData Boton_Status
Cliente.SendData enviar
Out &H378
imgonLH3.Visible = True
imgoffLH3.Visible = False
btnLH3.Enabled = False
btnLH3off.Enabled = True
End Sub

```

```

/*****/
* NOMBRE DEL PROYECTO: Smart House Control Client *
/*****/

```

```

* FUNCIÓN: Private Sub btnLH3off_Click() FECHA: 10-12-2003 *
/*****/

```

```

* PROPÓSITO: Desactivar Luz Habitación3 en el prototipo, visualizar en el cliente y *
* enviar dato del puerto al servidor *

```

```

/*****/
*
* ESTRUCTURA DE LA FUNCIÓN *
/*****/

```

```
Private Sub btnLH3off_Click()
Boton_Status = 6
Dato378 = Inp(&H378)
Dato378 = Dato378 And 11
enviar = Dato378
Cliente.SendData Boton_Status
Cliente.SendData enviar
Out &H378
imgonLH3.Visible = False
imgoffLH3.Visible = True
btnLH3.Enabled = True
btnLH3off.Enabled = False
End Sub
```

```
/*
* NOMBRE DEL PROYECTO: Smart House Control Client *
/*
* FUNCIÓN: Private Sub btnReset_Click() FECHA: 10-12-2003 *
/*
* PROPÓSITO: Inicializar variables, estados de luces, estado de sensores de Puertas *
* Ventanas, Censor de Temperatura en el prototipo y visualizar estados en el cliente *
/*
* ESTRUCTURA DE LA FUNCIÓN *
/*
```

```
Private Sub btnReset_Click()
FlagAlarmas = False
StrobeBeep = False

Out &H37A, &H4
```


Out &H378, &H0

Dato378 = 0

Boton_Status = 9

Cliente.SendData Boton_Status

Cliente.SendData Dato378

imgoffLH1.Visible = True

imgoffLH2.Visible = True

imgoffLH3.Visible = True

imgoffLE.Visible = True

imgonLH1.Visible = False

imgonLH2.Visible = False

imgonLH3.Visible = False

imgonLE.Visible = False

btnLH1.Enabled = True

btnLH2.Enabled = True

btnLH3.Enabled = True

btnLE.Enabled = True

btnLH1off.Enabled = False

btnLH2off.Enabled = False

btnLH3off.Enabled = False

btnLEoff.Enabled = False

shPI1.BackColor = &H0&

shPI2.BackColor = &H0&

shSV1.BackColor = &H0&

shSV2.BackColor = &H0&

imgvioPI1.Visible = False

imgsegPI1.Visible = True

imgvioPI2.Visible = False

imgsegPI2.Visible = True

imgvioV1.Visible = False

```
imgsegV1.Visible = True
imgvioV2.Visible = False
imgsegV2.Visible = True
imgintruso.Visible = False
imgnomov.Visible = True
```

```
Txttmin.BackColor = &HFFFC0
Txttmin1.BackColor = &H0&
Txttmed.BackColor = &H0&
Txttmed1.BackColor = &H0&
Txttmed2.BackColor = &H0&
Txttmax.BackColor = &H0&
End sub
```

```

/*****
* NOMBRE DEL PROYECTO: Smart House Control Client *
/*****
* FUNCIÓN: Private Sub btnSalir_Click() FECHA: 10-12-2003 *
/*****
* PROPÓSITO: Salir de la aplicación *
/*****
* ESTRUCTURA DE LA FUNCIÓN *
/*****
Private Sub btnSalir_Click()
End
End Sub
```

```

/*****
* NOMBRE DEL PROYECTO: Smart House Control Client *
/*****
* FUNCIÓN: Private Sub Cliente_Close() FECHA: 10-12-2003 *
/*****
```

```
* PROPÓSITO: Cierra la conexión con el equipo remoto *
/*****/
```

```
*                ESTRUCTURA DE LA FUNCIÓN                *
/*****/
```

```
Private Sub Cliente_Close()
Beep
MsgBox "Se ha cerrado la conexión"
Cliente.Close
Unload Me
End Sub
```

```
/*****/
* NOMBRE DEL PROYECTO: Smart House Control Client *
/*****/
```

```
* FUNCIÓN: Private Sub Cliente_Error()                FECHA: 10-12-2003 *
/*****/
```

```
* PROPÓSITO: Manipula el control de errores del socket de nombre Cliente *
/*****/
```

```
*                ESTRUCTURA DE LA FUNCIÓN                *
/*****/
```

```
Private Sub Cliente_Error(ByVal Number As Integer, Description As String, ByVal Scode
As Long, ByVal Source As String, ByVal HelpFile As String, ByVal HelpContext As
Long, CancelDisplay As Boolean)
CancelDisplay = False
Cliente.Close
End Sub
```

```
/*****/
* NOMBRE DEL PROYECTO: Smart House Control Client *
/*****/
```

```
* FUNCIÓN: Private Sub Cliente_Sens_Close()                FECHA: 10-12-2003 *
/*****/
```

```
* PROPÓSITO: Cierra la conexión del socket de nombre Cliente_Sens *
/*****/
```

```
* ESTRUCTURA DE LA FUNCIÓN *
/*****/
```

```
Private Sub Cliente_Sens_Close()
Cliente_Sens.Close
End Sub
```

```
/*****/
```

```
* NOMBRE DEL PROYECTO: Smart House Control Client *
/*****/
```

```
* FUNCIÓN: Private Sub Cliente_Sens_Error() FECHA: 10-12-2003 *
/*****/
```

```
* PROPÓSITO: Manipula el control de errores del socket de nombre Cliente_Sens *
/*****/
```

```
* ESTRUCTURA DE LA FUNCIÓN *
/*****/
```

```
Private Sub Cliente_Sens_Error(ByVal Number As Integer, Description As String, ByVal
Scode As Long, ByVal Source As String, ByVal HelpFile As String, ByVal HelpContext
As Long, CancelDisplay As Boolean)
CancelDisplay = False
Cliente_Sens.Close
End Sub
```

```
/*****/
```

```
* NOMBRE DEL PROYECTO: Smart House Control Client *
/*****/
```

```
* FUNCIÓN: Private Sub Cliente_Temp_Close() FECHA: 10-12-2003 *
/*****/
```

```
* PROPÓSITO: Cierra la conexión del socket de nombre Cliente_Temp *
/*****/
```

```
* ESTRUCTURA DE LA FUNCIÓN *
```

/*
*/

Private Sub Cliente_Temp_Close()

Beep

Cliente_Temp.Close

End Sub

/*
*/

* NOMBRE DEL PROYECTO: Smart House Control Client *

/*
*/

* FUNCIÓN: Private Sub Cliente_Temp_Error() FECHA: 10-12-2003 *

/*
*/

* PROPÓSITO: Manipula el control de errores del socket de nombre Cliente_Temp *

/*
*/

* ESTRUCTURA DE LA FUNCIÓN *

/*
*/

Private Sub Cliente_Temp_Error(ByVal Number As Integer, Description As String, ByVal
Scode As Long, ByVal Source As String, ByVal HelpFile As String, ByVal HelpContext
As Long, CancelDisplay As Boolean)

CancelDisplay = False

Cliente_Temp.Close

End Sub

/*
*/

* NOMBRE DEL PROYECTO: Smart House Control Client *

/*
*/

* FUNCIÓN: Private Sub cmdDesconectar_Click() FECHA: 10-12-2003 *

/*
*/

* PROPÓSITO: Cierra la conexión de los sockets y finaliza la aplicación del Cliente *

/*
*/

* ESTRUCTURA DE LA FUNCIÓN *

/*
*/

```
Private Sub cmdDesconectar_Click()  
    Cliente.Close  
    Cliente_Sens.Close  
    Cliente_Temp.Close  
    cmdDesconectar.Enabled = False  
    Unload Me  
End Sub
```

/*
*/

* NOMBRE DEL PROYECTO: Smart House Control Client *

/*
*/

* FUNCIÓN: Private Sub Form_Load() FECHA: 10-12-2003 *

/*
*/

* PROPÓSITO: Establece los valores iniciales de configuración de pantalla , inicializar *
* variables, estados de luces, estado de sensores de Puerta de Ingreso1, Puerta de *
* Ingreso2, Censor de Movimiento, Censor de Ventana1, Censor de Ventana2, Censor de *
* Temperatura *
* Pone los puertos en escucha y establece por qué puertos se va a tener acceso al host *
* remoto *

/*
*/

* ESTRUCTURA DE LA FUNCIÓN *

/*
*/

```
Private Sub Form_Load()  
    AnchoPantalla = Screen.Width  
    AltoPantalla = Screen.Height  
    Me.Top = (AltoPantalla) / 2 - Me.Height / 2  
    Me.Left = (AnchoPantalla) / 2 - Me.Width / 2  
  
    FlagAlarmas = False  
    StrobeBeep = False  
    Boton_Status = 0
```

Timer_Status = 0

Out &H37A, &H4

Cliente_Sens.RemotePort = 2000

Cliente_Sens.RemoteHost = "192.168.100.8"

Cliente_Sens.Connect

Cliente.RemotePort = 2001

Cliente.RemoteHost = "192.168.100.8"

Cliente.Connect

Cliente_Temp.RemotePort = 2002

Cliente_Temp.RemoteHost = "192.168.100.8"

Cliente_Temp.Connect

Out &H378, &H0

imgoffLH1.Visible = True

imgoffLH2.Visible = True

imgoffLH3.Visible = True

imgoffLE.Visible = True

imgonLH1.Visible = False

imgonLH2.Visible = False

imgonLH3.Visible = False

imgonLE.Visible = False

btnLH1.Enabled = True

btnLH2.Enabled = True

btnLH3.Enabled = True

btnLE.Enabled = True

btnLH1off.Enabled = False

btnLH2off.Enabled = False

btnLH3off.Enabled = False

btnLEoff.Enabled = False

shPI1.BackColor = &H0&

shPI2.BackColor = &H0&

shSV1.BackColor = &H0&

shSV2.BackColor = &H0&

imgvioPI1.Visible = False

imgsegPI1.Visible = True

imgvioPI2.Visible = False

imgsegPI2.Visible = True

imgvioV1.Visible = False

imgsegV1.Visible = True

imgvioV2.Visible = False

imgsegV2.Visible = True

imgintruso.Visible = False

imgnomov.Visible = True

Txttmin.BackColor = &HFFFFFF0

Txttmin1.BackColor = &H0&

Txttmed.BackColor = &H0&

Txttmed1.BackColor = &H0&

Txttmed2.BackColor = &H0&

Txttmax.BackColor = &H0&

End Sub

/*****/

* NOMBRE DEL PROYECTO: Smart House Control Client *

/*****/

* FUNCIÓN: Private Sub Cliente_Temp_DataArrival(ByVal bytesTotal As Long) *

* FECHA: 10-12-2003 *

/*****/

* PROPÓSITO: Recibe los datos procedentes del sensor de temperatura y visualiza en *

* el termómetro *


```
/******  
*/
```

```
*                               ESTRUCTURA DE LA FUNCIÓN                               *
```

```
/******  
*/
```

```
Private Sub Cliente_Temp_DataArrival(ByVal bytesTotal As Long)
```

```
    Cliente_Temp.GetData Celsius
```

```
    txtTemperatura.Text = Celsius
```

```
    If Celsius <= 15 Then
```

```
        Txttmin.BackColor = &HFFFFC0
```

```
        Txttmin1.BackColor = &H0&
```

```
        Txttmed.BackColor = &H0&
```

```
        Txttmed1.BackColor = &H0&
```

```
        Txttmed2.BackColor = &H0&
```

```
        Txttmax.BackColor = &H0&
```

```
        lblComentario.Caption = ""
```

```
        lblComentario.Caption = "AMBIENTE FRIO"
```

```
    End If
```

```
    If (Celsius > 15 And Celsius <= 25) Then
```

```
        Txttmin.BackColor = &HFFFFC0
```

```
        Txttmin1.BackColor = &H80FF80
```

```
        Txttmed.BackColor = &H0&
```

```
        Txttmed1.BackColor = &H0&
```

```
        Txttmed2.BackColor = &H0&
```

```
        Txttmax.BackColor = &H0&
```

```
        lblComentario.Caption = ""
```

```
        lblComentario.Caption = "TEMPERATURA NORMAL"
```

```
    End If
```

```
    If (Celsius > 25 And Celsius <= 35) Then
```

```
        Txttmin.BackColor = &HFFFFC0
```

```
        Txttmin1.BackColor = &H80FF80
```

```
        Txttmed.BackColor = &HFFFF&
```

```
        Txttmed1.BackColor = &H0&
```

```
Txttmed2.BackColor = &H0&
Txttmax.BackColor = &H0&
lblComentario.Caption = ""
lblComentario.Caption = "AMBIENTE CALIDO"
End If
If (Celsius > 35 And Celsius <= 40) Then
Txttmin.BackColor = &HFFFFC0
Txttmin1.BackColor = &H80FF80
Txttmed.BackColor = &HFFFF&
Txttmed1.BackColor = &H80FF&
Txttmed2.BackColor = &H0&
Txttmax.BackColor = &H0&
lblComentario.Caption = ""
lblComentario.Caption = "AMBIENTE MUY CALIDO"
End If
If (Celsius > 40 And Celsius <= 70) Then
Txttmin.BackColor = &HFFFFC0
Txttmin1.BackColor = &H80FF80
Txttmed.BackColor = &HFFFF&
Txttmed1.BackColor = &H80FF&
Txttmed2.BackColor = &H8080FF
Txttmax.BackColor = &H0&
lblComentario.Caption = ""
lblComentario.Caption = "ADVERTENCIA"
End If
If (Celsius > 70 And Celsius <= 100) Then
Txttmin.BackColor = &HFFFFC0
Txttmin1.BackColor = &H80FF80
Txttmed.BackColor = &HFFFF&
Txttmed1.BackColor = &H80FF&
Txttmed2.BackColor = &H8080FF
Txttmax.BackColor = &HFF&
```

```
lblComentario.Caption = ""
lblComentario.Caption = "PELIGRO DE INCENDIO"
ElseEnd If
End Sub
```

```
/*
* NOMBRE DEL PROYECTO: Smart House Control Client
*/
/*
* FUNCIÓN: Private Sub Cliente_DataArrival(ByVal bytesTotal As Long)
*/
/*
* PROPÓSITO: Recibe los datos de luces y visualiza su estado
*/
*
* ESTRUCTURA DE LA FUNCIÓN
*
*/
```

```
Private Sub Cliente_DataArrival(ByVal bytesTotal As Long)
Cliente.GetData Boton_Status
Cliente.GetData Dato378
```

```
Select Case Boton_Status
```

```
Case 1:
```

```
    If Dato378 = Dato378 Or 1 Then
        imgonLH1.Visible = True
        imgoffLH1.Visible = False
        btnLH1.Enabled = False
        btnLH1off.Enabled = True
    End If
```

```
Case 2:
```

```
    If Dato378 = Dato378 And 14 Then
        imgonLH1.Visible = False
        imgoffLH1.Visible = True
        btnLH1.Enabled = True
        btnLH1off.Enabled = False
```

End If

Case 3:

If Dato378 = Dato378 Or 2 Then

imgonLH2.Visible = True

imgoffLH2.Visible = False

btnLH2.Enabled = False

btnLH2off.Enabled = True

End If

Case 4:

If Dato378 = Dato378 And 13 Then

imgonLH2.Visible = False

imgoffLH2.Visible = True

btnLH2.Enabled = True

btnLH2off.Enabled = False

End If

Case 5:

If Dato378 = Dato378 Or 4 Then

imgonLH3.Visible = True

imgoffLH3.Visible = False

btnLH3.Enabled = False

btnLH3off.Enabled = True

End If

Case 6:

If Dato378 = Dato378 And 11 Then

imgonLH3.Visible = False

imgoffLH3.Visible = True

btnLH3.Enabled = True

btnLH3off.Enabled = False

End If

Case 7:

If Dato378 = Dato378 Or 8 Then

imgonLE.Visible = True

```
imgoffLE.Visible = False
btnLE.Enabled = False
btnLEoff.Enabled = True
End If
```

Case 8:

```
If Dato378 = Dato378 And 7 Then
imgonLE.Visible = False
imgoffLE.Visible = True
btnLE.Enabled = True
btnLEoff.Enabled = False
End If
```

Case 9:

```
If Dato378 = 0 Then
imgoffLH1.Visible = True
imgoffLH2.Visible = True
imgoffLH3.Visible = True
imgoffLE.Visible = True
imgonLH1.Visible = False
imgonLH2.Visible = False
imgonLH3.Visible = False
imgonLE.Visible = False
btnLH1.Enabled = True
btnLH2.Enabled = True
btnLH3.Enabled = True
btnLE.Enabled = True
btnLH1off.Enabled = False
btnLH2off.Enabled = False
btnLH3off.Enabled = False
btnLEoff.Enabled = False
End If
```

End Select

End Sub

```

/*****/
* NOMBRE DEL PROYECTO: Smart House Control Client *
/*****/
* FUNCIÓN: Private Sub Cliente_Sens_DataArrival(ByVal bytesTotal As Long) *
* FECHA: 10-12-2003 *
/*****/
* PROPÓSITO: Recibe los datos de sensores de puertas, ventanas y sensor de *
* movimiento y visualiza su estado *
/*****/
*
*          ESTRUCTURA DE LA FUNCIÓN *
/*****/

Private Sub Cliente_Sens_DataArrival(ByVal bytesTotal As Long)
  Cliente_Sens.GetData DatoPI1
  Cliente_Sens.GetData DatoPI2
  Cliente_Sens.GetData DatoSM1
  Cliente_Sens.GetData DatoSV1
  Cliente_Sens.GetData DatoSV2

  If DatoPI1 = &H8 Then
    shPI1.BackColor = &H0&
    imgvioPI1.Visible = False
    imgsegPI1.Visible = True
  Else
    shPI1.BackColor = &HFF&
    imgvioPI1.Visible = True
    imgsegPI1.Visible = False
  End If

  If DatoPI2 = &H10 Then
    shPI2.BackColor = &H0&
    imgvioPI2.Visible = False
    imgsegPI2.Visible = True

```

```
Else
shPI2.BackColor = &HFF&
imgvioPI2.Visible = True
imgsegPI2.Visible = False
End If
If DatoSM1 = &H20 Then
shSM1.BackColor = &H0&
imgintruso.Visible = False
imgnomov.Visible = True
Else
shSM1.BackColor = &HFF&
imgintruso.Visible = True
imgnomov.Visible = False
End If
If DatoSV1 = &H1 Then
shSV1.BackColor = &HFF&
imgvioV1.Visible = True
imgsegV1.Visible = False
Else
shSV1.BackColor = &H0&
imgvioV1.Visible = False
imgsegV1.Visible = True
End If
If DatoSV2 = &H2 Then
shSV2.BackColor = &HFF&
imgvioV2.Visible = True
imgsegV2.Visible = False
Else
shSV2.BackColor = &H0&
imgvioV2.Visible = False
imgsegV2.Visible = True
End If
```

End Sub

5.2.- CODIFICACIÓN SMART HOUSE CONTROL SERVER

```
/*  
* NOMBRE DEL PROYECTO: Smart House Control Server    FECHA: 10-12-2003 *  
/*  
* PROPÓSITO: Declarar variables *  
/*  
*          DECLARACIÓN DE VARIABLES *  
/*
```

```
Private Habilitador As Boolean  
Private StrobeBeep As Boolean  
Private DatoTemp As Integer  
Private Alarma As Integer
```

```
/*  
* NOMBRE DEL PROYECTO: Smart House Control Server *  
/*  
* FUNCIÓN: Private Sub btnLE_Click()                FECHA: 10-12-2003 *  
/*  
* PROPÓSITO: Activar Luz Externa en el prototipo, visualizar en el servidor y enviar *  
* dato del puerto al cliente *  
/*  
*          ESTRUCTURA DE LA FUNCIÓN *  
/*
```

```
Private Sub btnLE_Click()  
If Servidor.State = sckConnected Then  
Boton_Status = 7  
Dato378 = Inp(&H378)
```



```

Dato378 = Dato378 Or 8
enviar = Dato378
Servidor.SendData Boton_Status
Servidor.SendData enviar
Out &H378
imgonLE.Visible = True
imgoffLE.Visible = False
btnLE.Enabled = False
btnLEoff.Enabled = True
Else
MsgBox "Restablezca la conexión con el equipo remoto"
End If
End Sub

```

```

/*****
* NOMBRE DEL PROYECTO: Smart House Control Server *
/*****
* FUNCIÓN: Private Sub btnLEoff_Click() FECHA: 10-12-2003 *
/*****
* PROPÓSITO: Desactivar Luz Externa en el prototipo, visualizar en el servidor y *
* enviar dato del puerto al cliente *
/*****
* ESTRUCTURA DE LA FUNCIÓN *
/*****

Private Sub btnLEoff_Click()
If Servidor.State = sckConnected Then
Boton_Status = 8
Dato378 = Inp(&H378)
Dato378 = Dato378 And 7
enviar = Dato378

```

```

Servidor.SendData Boton_Status
Servidor.SendData enviar
Out &H378
imgonLE.Visible = False
imgoffLE.Visible = True
btnLE.Enabled = True
btnLEoff.Enabled = False
Else
MsgBox "Restablezca la conexión con el equipo remoto"
End If
End Sub

```

```

/*****/
* NOMBRE DEL PROYECTO: Smart House Control Server *
/*****/
* FUNCIÓN: Private Sub btnLH1_Click() FECHA: 10-12-2003 *
/*****/
* PROPÓSITO: Activar Luz Habitación1 en el prototipo, visualizar en el servidor y *
* enviar dato del puerto al cliente *
/*****/
* ESTRUCTURA DE LA FUNCIÓN *
/*****/

Private Sub btnLH1_Click()
If Servidor.State = sckConnected Then
Boton_Status = 1
Dato378 = Inp(&H378)
Dato378 = Dato378 Or 1
enviar = Dato378
Servidor.SendData Boton_Status
Servidor.SendData enviar
Out &H378
imgonLH1.Visible = True

```

```
imgoffLH1.Visible = False
btnLH1.Enabled = False
btnLH1off.Enabled = True
Else
MsgBox "Restablezca la conexión con el equipo remoto"
End If
End Sub
```

```
/******
```

```
* NOMBRE DEL PROYECTO: Smart House Control Server *
```

```
/******
```

```
* FUNCIÓN: Private Sub btnLH1off_Click() FECHA: 10-12-2003 *
```

```
/******
```

```
* PROPÓSITO: Desactivar Luz Habitación1 en el prototipo, visualizar en el servidor y *
```

```
* enviar dato del puerto al cliente *
```

```
/******
```

```
* ESTRUCTURA DE LA FUNCIÓN *
```

```
/******
```

```
Private Sub btnLH1off_Click()
```

```
If Servidor.State = sckConnected Then
```

```
Boton_Status = 2
```

```
Dato378 = Inp(&H378)
```

```
Dato378 = Dato378 And 14
```

```
enviar = Dato378
```

```
Servidor.SendData Boton_Status
```

```
Servidor.SendData enviar
```

```
Out &H378
```

```
imgonLH1.Visible = False
```

```
imgoffLH1.Visible = True
```

```
btnLH1.Enabled = True
```

```
btnLH1off.Enabled = False
```

```
Else
```

```
MsgBox "Restablezca la conexión con el equipo remoto"  
End If  
End Sub
```

```
/*  
* NOMBRE DEL PROYECTO: Smart House Control Server *  
/*  
* FUNCIÓN: Private Sub btnLH2_Click() FECHA: 10-12-2003 *  
/*  
* PROPÓSITO: Activar Luz Habitación2 en el prototipo, visualizar en el servidor y *  
* enviar dato del puerto al cliente *  
/*  
* ESTRUCTURA DE LA FUNCIÓN *  
/*
```

```
Private Sub btnLH2_Click()  
If Servidor.State = sckConnected Then  
Boton_Status = 3  
Dato378 = Inp(&H378)  
Dato378 = Dato378 Or 2  
enviar = Dato378  
Servidor.SendData Boton_Status  
Servidor.SendData enviar  
Out &H378  
imgonLH2.Visible = True  
imgoffLH2.Visible = False  
btnLH2.Enabled = False  
btnLH2off.Enabled = True  
Else  
MsgBox "Restablezca la conexión con el equipo remoto"  
End If  
End Sub
```

```
/*  
* NOMBRE DEL PROYECTO: Smart House Control Server *
```

```
/*  
* FUNCIÓN: Private Sub btnLH2off_Click() FECHA: 10-12-2003 *
```

```
/*  
* PROPÓSITO: Desactivar Luz Habitación2 en el prototipo, visualizar en el servidor y  
* enviar dato del puerto al cliente *
```

```
/*  
* ESTRUCTURA DE LA FUNCIÓN *
```

```
/*
```

```
Private Sub btnLH2off_Click()
```

```
If Servidor.State = sckConnected Then
```

```
Boton_Status = 4
```

```
Dato378 = Inp(&H378)
```

```
Dato378 = Dato378 And 13
```

```
enviar = Dato378
```

```
Servidor.SendData Boton_Status
```

```
Servidor.SendData enviar
```

```
Out &H378
```

```
imgonLH2.Visible = False
```

```
imgoffLH2.Visible = True
```

```
btnLH2.Enabled = True
```

```
btnLH2off.Enabled = False
```

```
Else
```

```
MsgBox "Restablezca la conexión con el equipo remoto"
```

```
End If
```

```
End Sub
```

```
/*  
* NOMBRE DEL PROYECTO: Smart House Control Server *
```

```
/*  
* FUNCIÓN: Private Sub btnLH3_Click() FECHA: 10-12-2003 *
```

/*
*/

* PROPÓSITO: Activar Luz Habitación3 en el prototipo, visualizar en el servidor y *

* enviar dato del puerto al cliente *

/*
*/

* ESTRUCTURA DE LA FUNCIÓN *

/*
*/

Private Sub btnLH3_Click()

If Servidor.State = sckConnected Then

Boton_Status = 5

Dato378 = Inp(&H378)

Dato378 = Dato378 Or 4

enviar = Dato378

Servidor.SendData Boton_Status

Servidor.SendData enviar

Out &H378

imgonLH3.Visible = True

imgoffLH3.Visible = False

btnLH3.Enabled = False

btnLH3off.Enabled = True

Else

MsgBox "Restablezca la conexión con el equipo remoto"

End If

End Sub

/*
*/

* NOMBRE DEL PROYECTO: Smart House Control Server *

/*
*/

* FUNCIÓN: Private Sub btnLH3off_Click() FECHA: 10-12-2003 *

/*
*/

* PROPÓSITO: Desactivar Luz Habitación3 en el prototipo, visualizar en el servidor y *

* enviar dato del puerto al cliente *

/*
*/

```

*                               ESTRUCTURA DE LA FUNCIÓN                               *
/*****/

Private Sub btnLH3off_Click()
If Servidor.State = sckConnected Then
Boton_Status = 6
Dato378 = Inp(&H378)
Dato378 = Dato378 And 11
enviar = Dato378
Servidor.SendData Boton_Status
Servidor.SendData enviar
Out &H378
imgonLH3.Visible = False
imgoffLH3.Visible = True
btnLH3.Enabled = True
btnLH3off.Enabled = False
Else
MsgBox "Restablezca la conexión con el equipo remoto"
End If
End Sub

/*****/
* NOMBRE DEL PROYECTO: Smart House Control Server *
/*****/
* FUNCIÓN: Private Sub btnReset_Click()                               FECHA: 10-12-2003 *
/*****/
* PROPÓSITO: Inicializar variables, estados de luces, estado de sensores de Puertas *
* Ventanas, Censor de Temperatura en el prototipo y visualizar estados en el servidor *
/*****/
*                               ESTRUCTURA DE LA FUNCIÓN                               *
/*****/

Private Sub btnReset_Click()
FlagAlarmas = False

```

StrobeBeep = False

Out &H37A, &H4

Out &H378, &H0

Dato378 = 0

Boton_Status = 9

Servidor.SendData Boton_Status

Servidor.SendData Dato378

imgoffLH1.Visible = True

imgoffLH2.Visible = True

imgoffLH3.Visible = True

imgoffLE.Visible = True

imgonLH1.Visible = False

imgonLH2.Visible = False

imgonLH3.Visible = False

imgonLE.Visible = False

btnLH1.Enabled = True

btnLH2.Enabled = True

btnLH3.Enabled = True

btnLE.Enabled = True

btnLH1off.Enabled = False

btnLH2off.Enabled = False

btnLH3off.Enabled = False

btnLEoff.Enabled = False

shPI1.BackColor = &H0&

shPI2.BackColor = &H0&

shSV1.BackColor = &H0&

shSV2.BackColor = &H0&

imgvioPI1.Visible = False

imgsegPI1.Visible = True


```
imgvioPI2.Visible = False
imgsegPI2.Visible = True
imgvioV1.Visible = False
imgsegV1.Visible = True
imgvioV2.Visible = False
imgsegV2.Visible = True
imgintruso.Visible = False
imgnomov.Visible = True
```

```
Txttmin.BackColor = &HFFFC0
Txttmin1.BackColor = &H0&
Txttmed.BackColor = &H0&
Txttmed1.BackColor = &H0&
Txttmed2.BackColor = &H0&
Txttmax.BackColor = &H0&
```

```
timMux.Enabled = True
Timseguridad.Enabled = False
Timtemperatura.Enabled = False
timBeep.Enabled = False
End Sub
```

```
/*
* NOMBRE DEL PROYECTO: Smart House Control Server *
/*
* FUNCIÓN: Private Sub btnSalir_Click() FECHA: 10-12-2003 *
/*
* PROPÓSITO: Salir de la aplicación *
/*
* ESTRUCTURA DE LA FUNCIÓN *
/*
Private Sub btnSalir_Click()
```

End

End Sub

/*

*/

* NOMBRE DEL PROYECTO: Smart House Control Server *

/*

*/

* FUNCIÓN: Private Sub cmdClose_Click() FECHA: 10-12-2003 *

/*

*/

* PROPÓSITO: Envía mensaje de cierre reconexión del servidor al cliente y cierra la *

* conexión de los sockets de luces, sensores (puertas, ventanas, movimiento) y sensor de *

* temperatura *

/*

*/

* ESTRUCTURA DE LA FUNCIÓN *

/*

*/

Private Sub cmdClose_Click()

If Servidor.State = sckConnected Then

Servidor.SendData "El servidor se ha cerrado"

Servidor.Close

Servidor_Sens.Close

Servidor_Temp.Close

Unload Me

Else

Servidor.Close

Servidor_Sens.Close

Servidor_Temp.Close

End If

End Sub

/*

*/

* NOMBRE DEL PROYECTO: Smart House Control Server *

/*

*/

* FUNCIÓN: Private Sub Form_Load() FECHA: 10-12-2003 *

```

/*****/
* PROPÓSITO: Establece los valores iniciales de configuración de pantalla , inicializar *
* variables, estados de luces, estado de sensores de Puerta de Ingreso1, Puerta de *
* Ingreso2, Censor de Movimiento, Censor de Ventana1, Censor de Ventana2, Censor de*
* Temperatura en el servidor *
* Establece los números de puertos por los cuales se establecerá la conexión *
/*****/
*
*                      ESTRUCTURA DE LA FUNCIÓN
*
/*****/

Private Sub Form_Load()
AnchoPantalla = Screen.Width
AltoPantalla = Screen.Height
Me.Top = (AltoPantalla) / 2 - Me.Height / 2
Me.Left = (AnchoPantalla) / 2 - Me.Width / 2

FlagAlarmas = False
StrobeBeep = False
Boton_Status = 0
Timer_Status = 0 ' variable del estado de los timers

Out &H37A, &H4

Servidor_Sens.LocalPort = 2000
Servidor_Sens.Listen
Servidor.LocalPort = 2001
Servidor.Listen
Servidor_Temp.LocalPort = 2002
Servidor_Temp.Listen

Out &H378, &H0
imgoffLH1.Visible = True
imgoffLH2.Visible = True

```

imgoffLH3.Visible = True
imgoffLE.Visible = True
imgonLH1.Visible = False
imgonLH2.Visible = False
imgonLH3.Visible = False
imgonLE.Visible = False
btnLH1.Enabled = True
btnLH2.Enabled = True
btnLH3.Enabled = True
btnLE.Enabled = True
btnLH1off.Enabled = False
btnLH2off.Enabled = False
btnLH3off.Enabled = False
btnLEoff.Enabled = False

shPI1.BackColor = &H0&
shPI2.BackColor = &H0&
shSV1.BackColor = &H0&
shSV2.BackColor = &H0&
imgvioPI1.Visible = False
imgsegPI1.Visible = True
imgvioPI2.Visible = False
imgsegPI2.Visible = True
imgvioV1.Visible = False
imgsegV1.Visible = True
imgvioV2.Visible = False
imgsegV2.Visible = True
imgintruso.Visible = False
imgnomov.Visible = True

Txttmin.BackColor = &HFFFC0
Txttmin1.BackColor = &H0&

```
Txttmed.BackColor = &H0&
Txttmed1.BackColor = &H0&
Txttmed2.BackColor = &H0&
Txttmax.BackColor = &H0&
```

```
timMux.Enabled = True
Timseguridad.Enabled = False
Timtemperatura.Enabled = False
timBeep.Enabled = False
End Sub
```

```
/*
*****
*/
```

```
* NOMBRE DEL PROYECTO: Smart House Control Server *
```

```
/*
*****
*/
```

```
* FUNCIÓN: Private Sub Servidor_Close() FECHA: 10-12-2003 *
```

```
/*
*****
*/
```

```
* PROPÓSITO: Cierra la conexión del socket de luces y luego lo pone en escucha *
```

```
/*
*****
*/
```

```
* ESTRUCTURA DE LA FUNCIÓN *
```

```
/*
*****
*/
```

```
Private Sub Servidor_Close()
```

```
Servidor.Close
```

```
Servidor.LocalPort = 2001
```

```
Servidor.Listen
```

```
End Sub
```

```
/*
*****
*/
```

```
* NOMBRE DEL PROYECTO: Smart House Control Server *
```

```
/*
*****
*/
```

```
* FUNCIÓN: Private Sub Servidor_Sens_Close() FECHA: 10-12-2003 *
```

```
/*
*****
*/
```

```
* PROPÓSITO: Cierra la conexión del socket de los sensores de puertas, ventanas y mo- *
```

```
* vimiento y posteriormente lo pone en escucha *
/*****/
```

```
*          ESTRUCTURA DE LA FUNCIÓN          *
/*****/
```

```
Private Sub Servidor_Sens_Close()
Servidor_Sens.Close
Servidor_Sens.LocalPort = 2000
Servidor_Sens.Listen
End Sub
```

```
/*****/
```

```
* NOMBRE DEL PROYECTO: Smart House Control Server *
/*****/
```

```
* FUNCIÓN: Private Sub Servidor_Sens_Error()          FECHA: 10-12-2003 *
/*****/
```

```
* PROPÓSITO: Manipula el control de errores del socket de nombre Servidor_Sens *
/*****/
```

```
*          ESTRUCTURA DE LA FUNCIÓN          *
/*****/
```

```
Private Sub Servidor_Sens_Error(ByVal Number As Integer, Description As String, ByVal
Scode As Long, ByVal Source As String, ByVal HelpFile As String, ByVal HelpContext
As Long, CancelDisplay As Boolean)
CancelDisplay = False
Servidor_Sens.Close
End Sub
```

```
/*****/
```

```
* NOMBRE DEL PROYECTO: Smart House Control Server *
/*****/
```

```
* FUNCIÓN: Private Sub Servidor_Error ()          FECHA: 10-12-2003 *
/*****/
```

```
* PROPÓSITO: Manipula el control de errores del socket de nombre Servidor *
/*****/
```

/*

*/

* ESTRUCTURA DE LA FUNCIÓN *

/*

*/

Private Sub Servidor_Error(ByVal Number As Integer, Description As String, ByVal
Scode As Long, ByVal Source As String, ByVal HelpFile As String, ByVal HelpContext
As Long, CancelDisplay As Boolean)

CancelDisplay = False

Servidor.Close

End Sub

/*

*/

* NOMBRE DEL PROYECTO: Smart House Control Server *

/*

*/

* FUNCIÓN: Private Sub Servidor_Temp_Close() FECHA: 10-12-2003 *

/*

*/

* PROPÓSITO: Cierra la conexión del socket del sensor de temperatura y posteriormente*

* lo pone en escucha *

/*

*/

* ESTRUCTURA DE LA FUNCIÓN *

/*

*/

Private Sub Servidor_Temp_Close()

Servidor_Temp.Close

Servidor_Temp.LocalPort = 2002

Servidor_Temp.Listen

End Sub

/*

*/

* NOMBRE DEL PROYECTO: Smart House Control Server *

/*

*/

* FUNCIÓN: Private Sub Servidor_Temp_Error () FECHA: 10-12-2003 *

/*

*/

* PROPÓSITO: Manipula el control de errores del socket de nombre Servidor_Temp *


```
/*
* FUNCIÓN: Private Sub Timseguridad_Timer()          FECHA: 10-12-2003 *
*/
```

```
/* PROPÓSITO: Detecta violación de seguridad en puertas, ventanas, y censor de movi- *
* miento del prototipo y visualiza estados en el servidor, además envía estados de censo- *
* al Cliente *
*/
```

```
/*
```

```
          ESTRUCTURA DE LA FUNCIÓN          *
```

```
/*
```

```
Private Sub Timseguridad_Timer()
```

```
Dato378 = Inp(&H378)
```

```
Dato378 = Dato378 And &HEF
```

```
Out &H378, Dato378
```

```
Dato379 = Inp(&H379)
```

```
Dato37A = Inp(&H37A)
```

```
DatoPI1 = Dato379 And &H8
```

```
DatoPI2 = Dato379 And &H10
```

```
DatoSM1 = Dato379 And &H20
```

```
DatoSV1 = Dato37A And &H1
```

```
DatoSV2 = Dato37A And &H2
```

```
If Servidor_Sens.State = sckConnected Then
```

```
Servidor_Sens.SendData DatoPI1
```

```
Servidor_Sens.SendData DatoPI2
```

```
Servidor_Sens.SendData DatoSM1
```

```
Servidor_Sens.SendData DatoSV1
```

```
Servidor_Sens.SendData DatoSV2
```

```
End If
```

```
If DatoPI1 = &H8 Then
```

```
    FlagPI1 = False
```

```
shPI1.BackColor = &H0&
imgvioPI1.Visible = False
imgsegPI1.Visible = True
Else
FlagPI1 = True
shPI1.BackColor = &HFF&
imgvioPI1.Visible = True
imgsegPI1.Visible = False
End If
If DatoPI2 = &H10 Then
FlagPI2 = False
shPI2.BackColor = &H0&
imgvioPI2.Visible = False
imgsegPI2.Visible = True
Else
FlagPI2 = True
shPI2.BackColor = &HFF&
imgvioPI2.Visible = True
imgsegPI2.Visible = False
End If
If DatoSM1 = &H20 Then
FlagSM1 = False
shSM1.BackColor = &H0&
imgintruso.Visible = False
imgnomov.Visible = True
Else
FlagSM1 = True
shSM1.BackColor = &HFF&
imgintruso.Visible = True
imgnomov.Visible = False
End If
If DatoSV1 = &H1 Then
```

```

FlagSV1 = True
shSV1.BackColor = &HFF&
imgvioV1.Visible = True
imgsegV1.Visible = False
Else
FlagSV1 = False
shSV1.BackColor = &H0&
imgvioV1.Visible = False
imgsegV1.Visible = True
End If
If DatoSV2 = &H2 Then
FlagSV2 = True
shSV2.BackColor = &HFF&
imgvioV2.Visible = True
imgsegV2.Visible = False
Else
FlagSV2 = False
shSV2.BackColor = &H0&
imgvioV2.Visible = False
imgsegV2.Visible = True
End If
If (FlagPI1 Or FlagPI2 Or FlagSM1 Or FlagSV1 Or FlagSV2) = True Then
Dato378 = Inp(&H378)
Dato378 = Dato378 Or &H20
Else
Dato378 = Inp(&H378)
Dato378 = Dato378 And &HDF
End If
End Sub

```

```

/*****

```

```

* NOMBRE DEL PROYECTO: Smart House Control Server

```

```

*

```

```

/*****
* FUNCIÓN: Private Sub Timtemperatura_Timer()          FECHA: 10-12-2003 *
/*****
* PROPÓSITO: Detecta la temperatura interna de la residencia y visualiza en el servidor *
* la temperatura en grados centígrados, además envía el dato de temperatura al Cliente *
/*****
*
*                      ESTRUCTURA DE LA FUNCIÓN
*
/*****

Private Sub Timtemperatura_Timer()
Dim aux As Integer
Dato378 = Inp(&H378)
Dato378 = Dato378 Or &H10
Out &H378
DatoTemp = DatoTemp And &HF
DatoTemp = DatoTemp Xor &HB
aux = Inp(&H379)
aux = aux * &H2
aux = aux And &HF0
DatoTemp = DatoTemp + aux
Celsius = DatoTemp * (100 / 255)
If Servidor_Temp.State = sckConnected Then
Servidor_Temp.SendData Celsius
End If
txtTemperatura.Text = Celsius
If Celsius <= 15 Then
Ttxtmin.BackColor = &HFFFC0
Ttxtmin1.BackColor = &H0&
Ttxtmed.BackColor = &H0&
Ttxtmed1.BackColor = &H0&
Ttxtmed2.BackColor = &H0&
Ttxtmax.BackColor = &H0&
lblComentario.Caption = ""

```

```
lblComentario.Caption = "AMBIENTE FRIO"
Dato378 = Inp(&H378)
Dato378 = Dato378 And &HDF
timBeep.Enabled = False
End If
If (Celsius > 15 And Celsius <= 25) Then
Ttxtmin.BackColor = &HFFFC0
Ttxtmin1.BackColor = &H80FF80
Ttxtmed.BackColor = &H0&
Ttxtmed1.BackColor = &H0&
Ttxtmed2.BackColor = &H0&
Ttxtmax.BackColor = &H0&
lblComentario.Caption = ""
lblComentario.Caption = "TEMPERATURA NORMAL"
Dato378 = Inp(&H378)
Dato378 = Dato378 And &HDF
timBeep.Enabled = False
End If
If (Celsius > 25 And Celsius <= 35) Then
Ttxtmin.BackColor = &HFFFC0
Ttxtmin1.BackColor = &H80FF80
Ttxtmed.BackColor = &HFFFF&
Ttxtmed1.BackColor = &H0&
Ttxtmed2.BackColor = &H0&
Ttxtmax.BackColor = &H0&
lblComentario.Caption = ""
lblComentario.Caption = "AMBIENTE CALIDO"
Dato378 = Inp(&H378)
Dato378 = Dato378 And &HDF
timBeep.Enabled = False
End If
If (Celsius > 35 And Celsius <= 40) Then
```

```
Txttmin.BackColor = &HFFFC0
Txttmin1.BackColor = &H80FF80
Txttmed.BackColor = &HFFF&
Txttmed1.BackColor = &H80FF&
Txttmed2.BackColor = &H0&
Txttmax.BackColor = &H0&
lblComentario.Caption = ""
lblComentario.Caption = "AMBIENTE MUY CALIDO"
Dato378 = Inp(&H378)
Dato378 = Dato378 And &HDF
timBeep.Enabled = False
End If
If (Celsius > 40 And Celsius <= 70) Then
Txttmin.BackColor = &HFFFC0
Txttmin1.BackColor = &H80FF80
Txttmed.BackColor = &HFFF&
Txttmed1.BackColor = &H80FF&
Txttmed2.BackColor = &H8080FF
Txttmax.BackColor = &H0&
lblComentario.Caption = ""
lblComentario.Caption = "ADVERTENCIA "
Dato378 = Inp(&H378)
Dato378 = Dato378 Or &H20
Out &H378
End If
If (Celsius > 70 And Celsius <= 100) Then
Txttmin.BackColor = &HFFFC0
Txttmin1.BackColor = &H80FF80
Txttmed.BackColor = &HFFF&
Txttmed1.BackColor = &H80FF&
Txttmed2.BackColor = &H8080FF
Txttmax.BackColor = &HFF&
```

```
lblComentario.Caption = ""
lblComentario.Caption = "PELIGRO DE INCENDIO"
Dato378 = Inp(&H378)
Dato378 = Dato378 Or &H20
Out &H378
Else
End If
End Sub
```

```
/*
* NOMBRE DEL PROYECTO: Smart House Control Server *
*/
* FUNCIÓN: Private Sub Timtemperatura_Timer()          FECHA: 10-12-2003 *
*/
* PROPÓSITO: Habilita la ejecución del timer que controla seguridad, y del timer que *
* controla temperatura *
*/
*                               ESTRUCTURA DE LA FUNCIÓN                               *
*/
Private Sub timMux_Timer()
If Habilitador = True Then
    Timtemperatura.Enabled = False
    Timseguridad.Enabled = True
Else
    Timseguridad.Enabled = False
    Timtemperatura.Enabled = True
End If
Habilitador = Not Habilitador
End Sub
```

```
/*
* NOMBRE DEL PROYECTO: Smart House Control Server *
*/
```

```

/*****
* FUNCIÓN: Private Sub Servidor_ConnectionRequest(ByVal requestID As Long) *
* FECHA: 10-12-2003 *
/*****
* PROPÓSITO: Acepta la petición de conexión del socket de nombre Servidor *
/*****
*
*          ESTRUCTURA DE LA FUNCIÓN *
/*****

Private Sub Servidor_ConnectionRequest(ByVal requestID As Long)
If Servidor.State <> sckClosed Then
Servidor.Close
Servidor.Accept requestID
End If
End Sub

```

```

/*****
* NOMBRE DEL PROYECTO: Smart House Control Server *
/*****
* FUNCIÓN: Private Sub Servidor_Sens_ConnectionRequest(ByVal requestID As Long)*
* FECHA: 10-12-2003 *
/*****
* PROPÓSITO: Acepta la petición de conexión del socket de nombre Servidor_Sens *
/*****
*
*          ESTRUCTURA DE LA FUNCIÓN *
/*****

```

```

Private Sub Servidor_Sens_ConnectionRequest(ByVal requestID As Long)
If Servidor_Sens.State <> sckClosed Then
Servidor_Sens.Close
Servidor_Sens.Accept requestID
End If

```


End Sub

```

/*****
* NOMBRE DEL PROYECTO: Smart House Control Server *
/*****
* FUNCIÓN: Private Sub Servidor_Temp_ConnectionRequest(ByVal requestID As Long)
* FECHA: 10-12-2003 *
/*****
* PROPÓSITO: Acepta la petición de conexión del socket de nombre Servidor_Temp *
/*****
*
*          ESTRUCTURA DE LA FUNCIÓN
*
/*****
Private Sub Servidor_Temp_ConnectionRequest(ByVal requestID As Long)
If Servidor_Temp.State <> sckClosed Then
Servidor_Temp.Close
Servidor_Temp.Accept requestID
End If
End Sub

```

```

/*****
* NOMBRE DEL PROYECTO: Smart House Control Server *
/*****
* FUNCIÓN: Private Sub Servidor_DataArrival(ByVal bytesTotal As Long) *
* FECHA: 10-12-2003 *
/*****
* PROPÓSITO: Recibe los datos de luces y visualiza su estado en el servidor *
/*****
*
*          ESTRUCTURA DE LA FUNCIÓN
*
/*****
Private Sub Servidor_DataArrival(ByVal bytesTotal As Long)
Servidor.GetData Boton_Status

```

Servidor.GetData Dato378

Select Case Boton_Status

Case 1:

If Dato378 = Dato378 Or 1 Then

Out &H378

imgonLH1.Visible = True

imgoffLH1.Visible = False

btnLH1.Enabled = False

btnLH1off.Enabled = True

End If

Case 2:

If Dato378 = Dato378 And 14 Then

Out &H378

imgonLH1.Visible = False

imgoffLH1.Visible = True

btnLH1.Enabled = True

btnLH1off.Enabled = False

End If

Case 3:

If Dato378 = Dato378 Or 2 Then

Out &H378

imgonLH2.Visible = True

imgoffLH2.Visible = False

btnLH2.Enabled = False

btnLH2off.Enabled = True

End If

Case 4:

If Dato378 = Dato378 And 13 Then

Out &H378

imgonLH2.Visible = False

imgoffLH2.Visible = True

```
btnLH2.Enabled = True
btnLH2off.Enabled = False
End If
```

Case 5:

```
If Dato378 = Dato378 Or 4 Then
Out &H378
imgonLH3.Visible = True
imgoffLH3.Visible = False
btnLH3.Enabled = False
btnLH3off.Enabled = True
End If
```

Case 6:

```
If Dato378 = Dato378 And 11 Then
Out &H378
imgonLH3.Visible = False
imgoffLH3.Visible = True
btnLH3.Enabled = True
btnLH3off.Enabled = False
End If
```

Case 7:

```
If Dato378 = Dato378 Or 8 Then
Out &H378
imgonLE.Visible = True
imgoffLE.Visible = False
btnLE.Enabled = False
btnLEoff.Enabled = True
End If
```

Case 8:

```
If Dato378 = Dato378 And 7 Then
Out &H378
imgonLE.Visible = False
imgoffLE.Visible = True
```

```
btnLE.Enabled = True
btnLEoff.Enabled = False
End If
```

Case 9:

```
If Dato378 = 0 Then
imgoffLH1.Visible = True
imgoffLH2.Visible = True
imgoffLH3.Visible = True
imgoffLE.Visible = True
imgonLH1.Visible = False
imgonLH2.Visible = False
imgonLH3.Visible = False
imgonLE.Visible = False
```

```
btnLH1.Enabled = True
btnLH2.Enabled = True
btnLH3.Enabled = True
btnLE.Enabled = True
btnLH1off.Enabled = False
btnLH2off.Enabled = False
btnLH3off.Enabled = False
btnLEoff.Enabled = False
End If
```

End Select

End Sub

```
/*
* NOMBRE DEL PROYECTO: Smart House Control Server *
*
* MÓDULO: Variables FECHA: 10-12-2003 *
*
* PROPÓSITO: Declarar variables públicas del sistema *
```

/******
/

* ESTRUCTURA DE LA FUNCIÓN *

/******
/

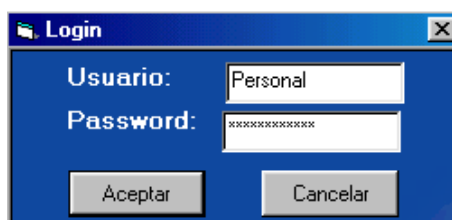
- Public Dato378 As Integer
- Public Dato379 As Integer
- Public Dato37A As Integer
- Public DatoPI1 As Integer
- Public DatoPI2 As Integer
- Public DatoSM1 As Integer
- Public DatoSV1 As Integer
- Public DatoSV2 As Integer
- Public FlagPI1 As Boolean
- Public FlagPI2 As Boolean
- Public FlagSM1 As Boolean
- Public FlagSV1 As Boolean
- Public FlagSV2 As Boolean
- Public Celsius as Double
- Public Boton_Status as Integer
- Public enviar as Variant

PRUEBAS SMART HOUSE CONTROL

6.1.- PRUEBA PERMISO DE ACCESO SHC-PC

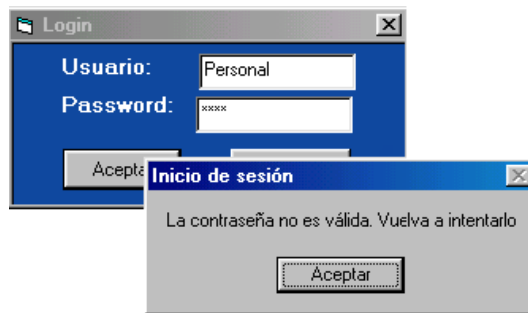
Caso 1

Resultado Esperado

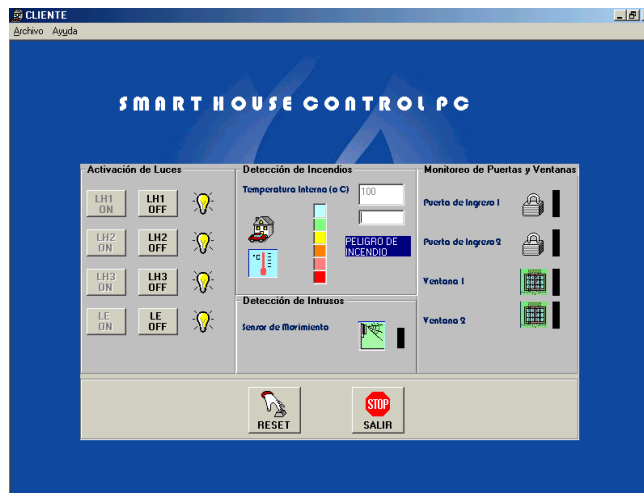


Caso 2

Resultado Esperado

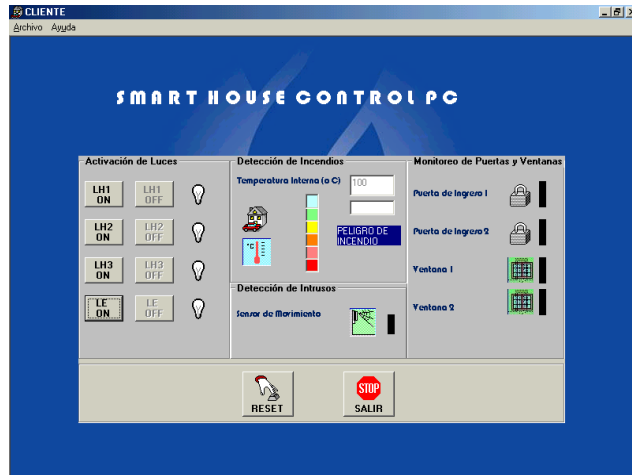


6.2.- PRUEBA ENCENDIDO DE LUCES SHC-PC



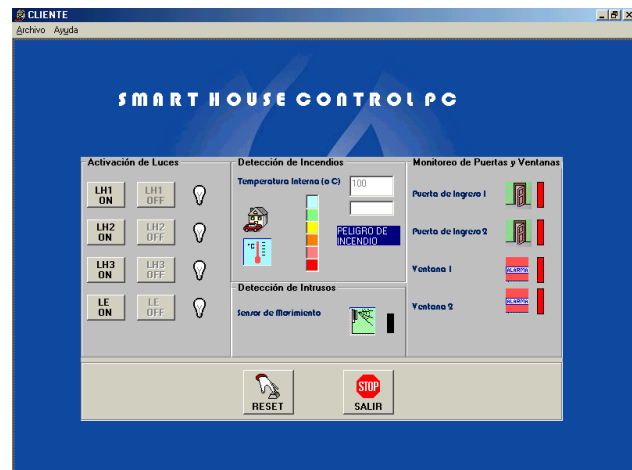
6.3.- PRUEBA APAGADO DE LUCES SHC-PC

Resultado Esperado



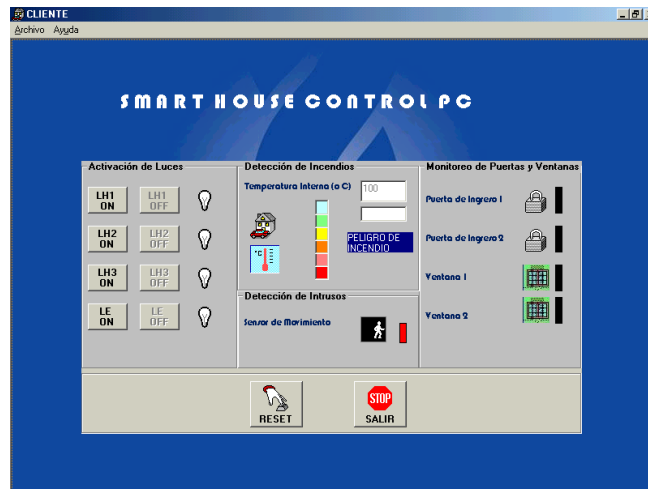
6.4.- VISUALIZAR ESTADOS DE SENSORES DE PUERTAS Y VENTANAS SHC-PC

Resultado Esperado



6.5.- VISUALIZAR ESTADO DEL SENSOR DE MOVIMIENTO SHC-PC

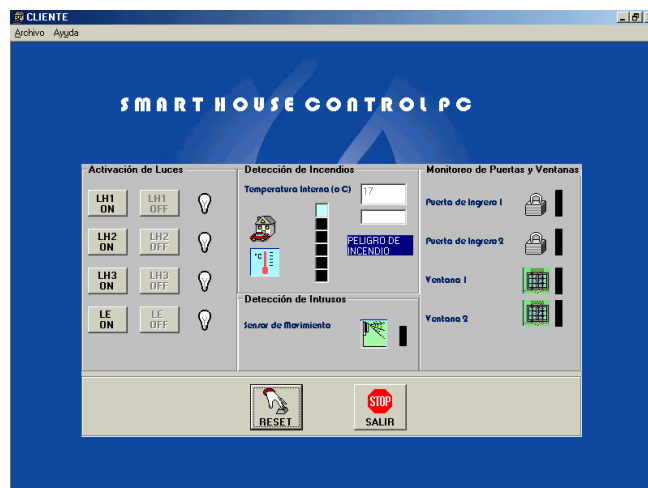
Resultado Esperado



6.6.- VISUALIZAR TEMPERATURA INTERNA DE LA RESIDENCIA SHC-PC

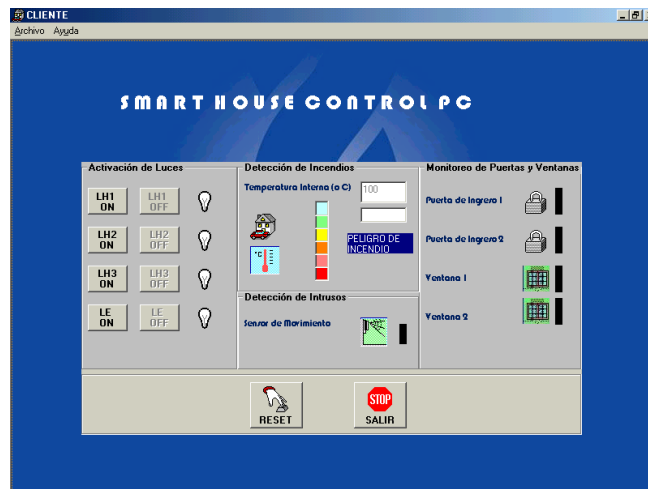
Caso 1

Resultado Esperado



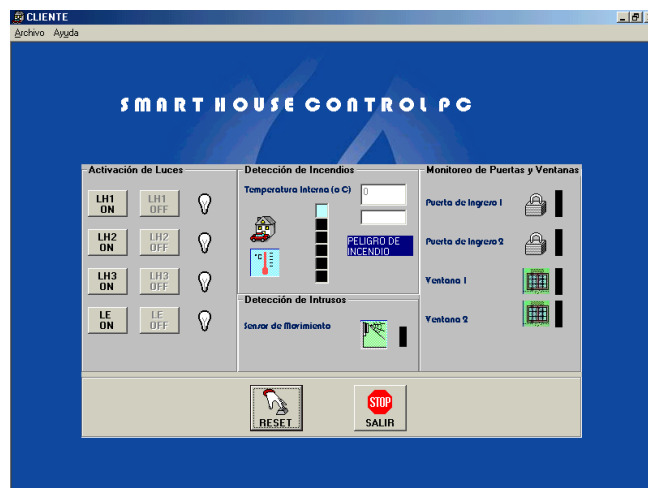
Caso 2

Resultado Esperado



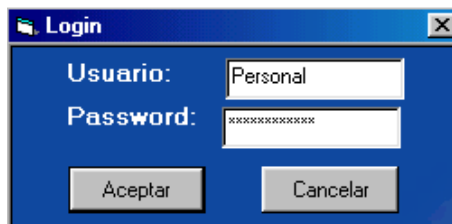
6.7.- PRUEBA INICIALIZAR ESTADOS SHC-PC

Resultado Esperado



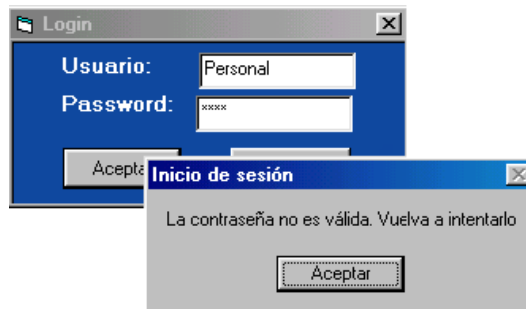
6.8.- PRUEBA PERMISO DE ACCESO SHC-CLIENT

Caso 1



Caso 2

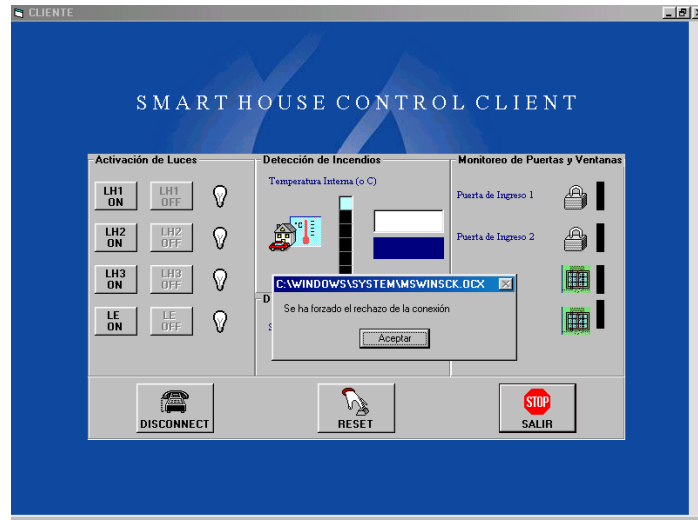
Resultado Esperado



6.9.- PRUEBA DE CONEXIÓN SHC-CLIENT

Caso 1

Resultado Esperado



Caso 2

Resultado Esperado



6.10.- PRUEBA ENCENDIDO DE LUCES SHC-CLIENT

Resultado Esperado



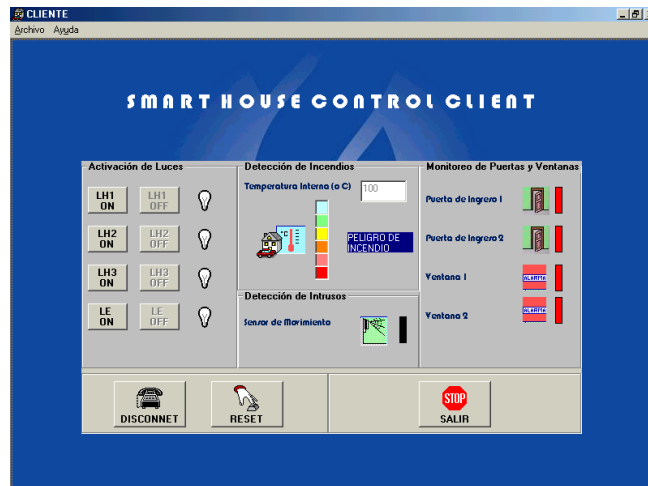
6.11.- PRUEBA APAGADO DE LUCES SHC-CLIENT

Resultado Esperado



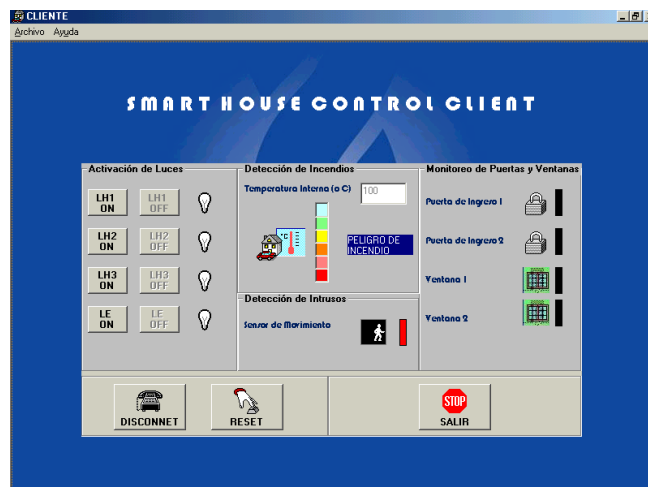
6.12.- VISUALIZAR ESTADOS DE SENSORES DE PUERTAS Y VENTANAS SHC-CLIENT

Resultado Esperado



6.13.- VISUALIZAR ESTADO DEL SENSOR DE MOVIMIENTO SHC-CLIENT

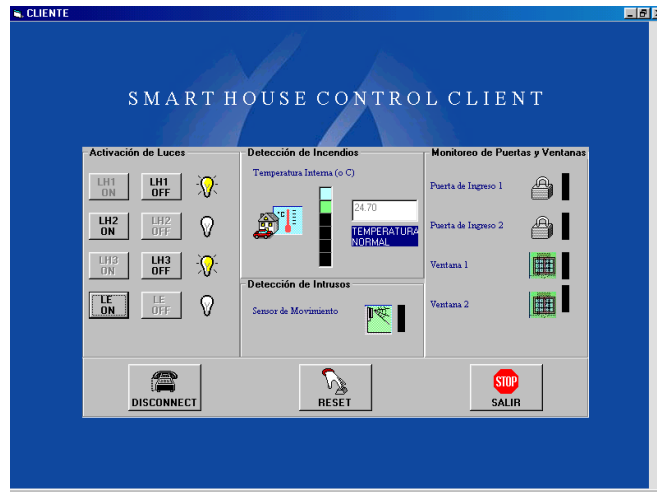
Resultado Esperado



6.14.- VISUALIZAR TEMPERATURA INTERNA DE LA RESIDENCIA SHC-CLIENT

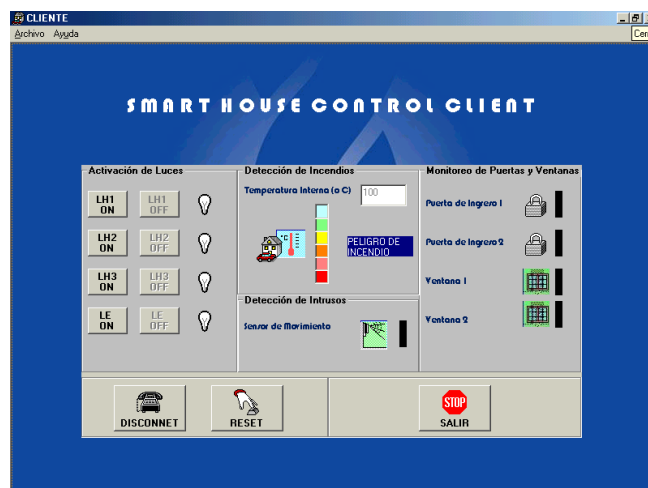
Caso 1

Resultado Esperado



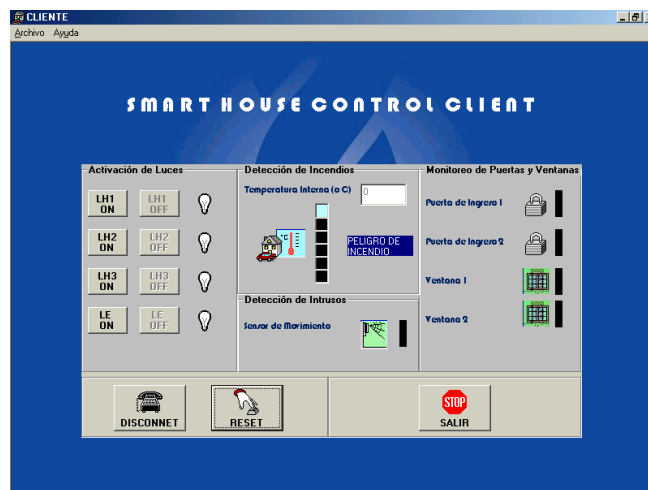
Caso 2

Resultado Esperado



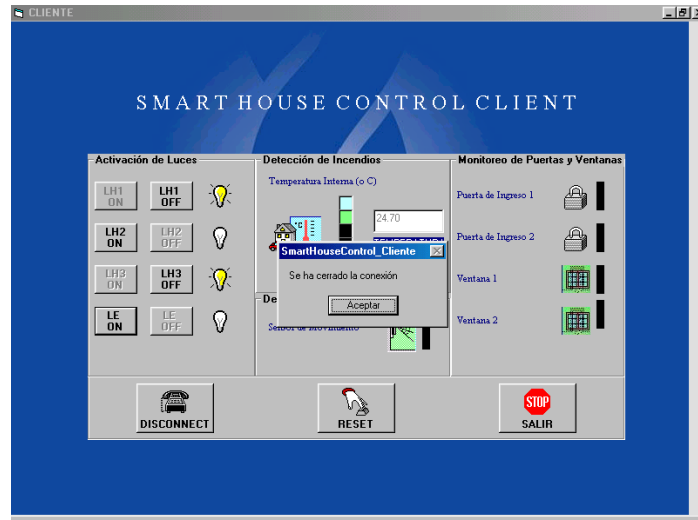
6.15.- PRUEBA INICIALIZAR ESTADOS SHC-CLIENT

Resultado Esperado



6.16.- PRUEBA CERRAR CONEXIÓN

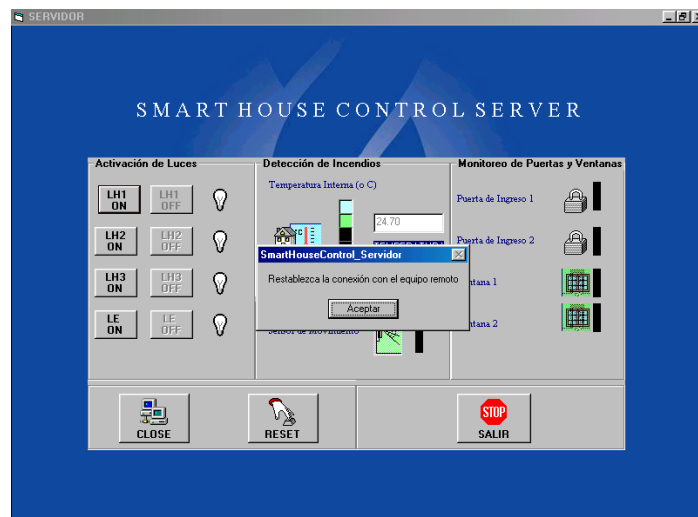
Resultado Esperado



6.17.- PRUEBA DE CONEXIÓN SHC-SERVER

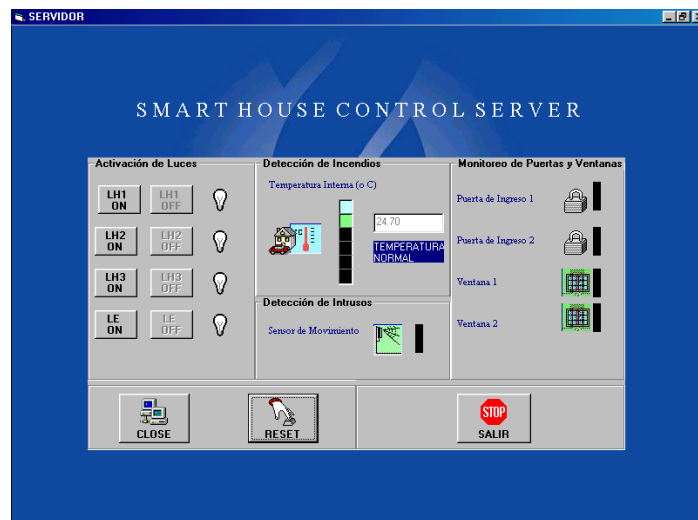
Caso 1

Resultado Esperado



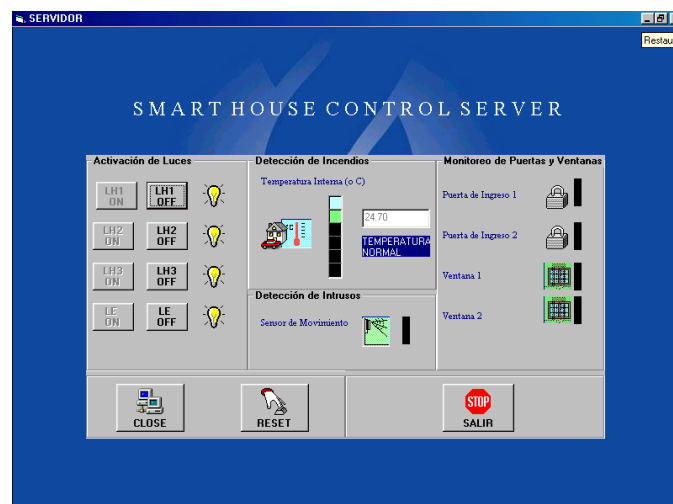
Caso 2

Resultado Esperado



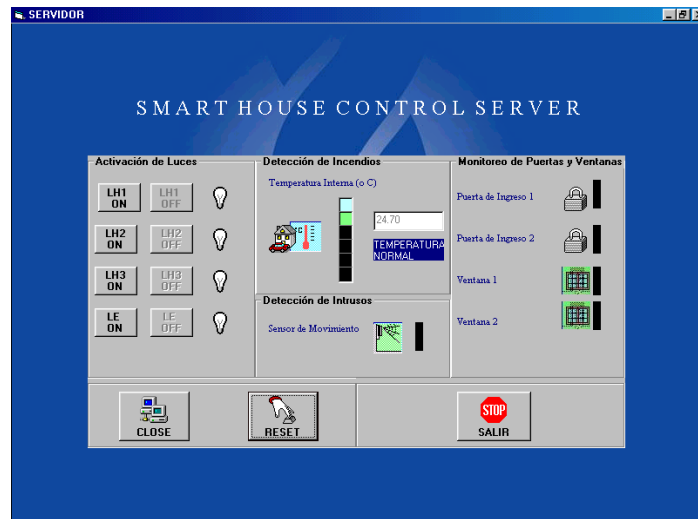
6.18.- PRUEBA ENCENDIDO DE LUCES SHC-SERVER

Resultado Esperado



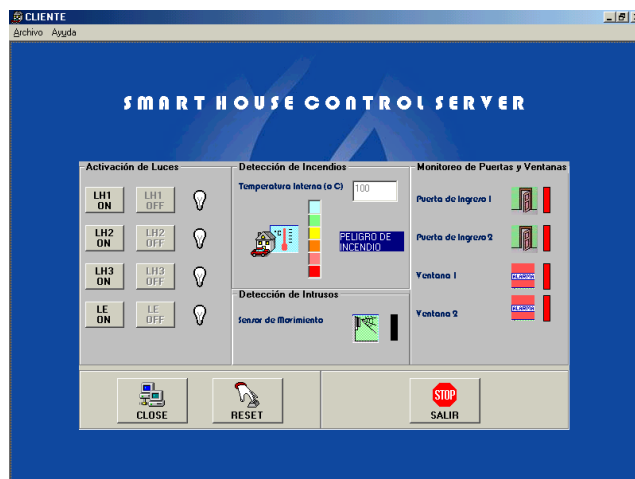
6.19.- PRUEBA APAGADO DE LUCES SHC-SERVER

Resultado Esperado



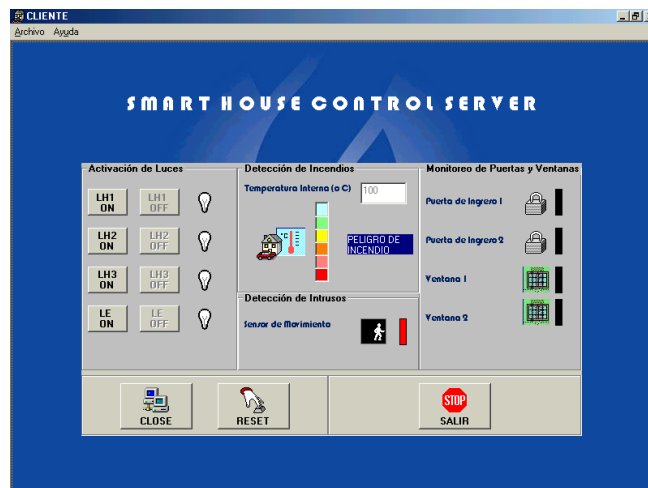
6.20.- VISUALIZAR ESTADOS DE SENSORES DE PUERTAS Y VENTANAS SHC-SERVER

Resultado Esperado



6.21.- VISUALIZAR ESTADO DEL SENSOR DE MOVIMIENTO SHC-SERVER

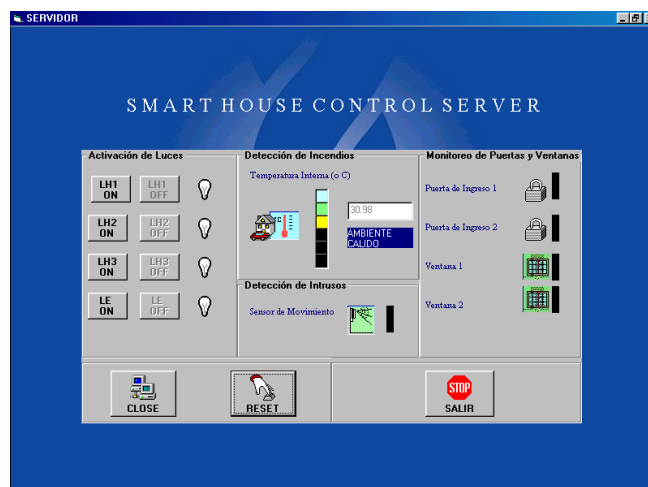
Resultado Esperado



6.22.- VISUALIZAR TEMPERATURA INTERNA DE LA RESIDENCIA SHC-SERVER

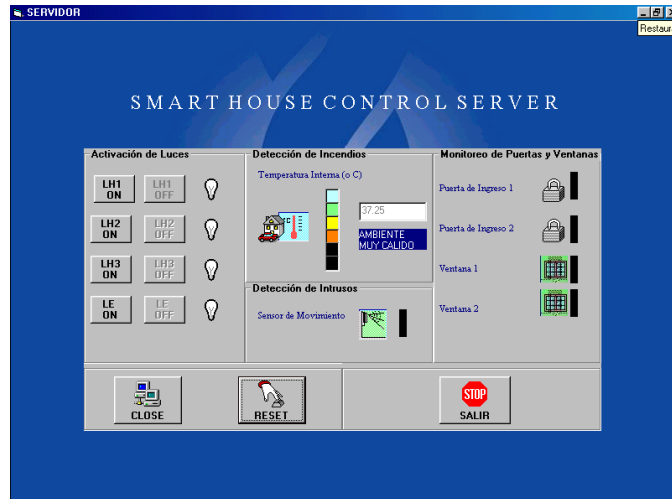
Caso 1

Resultado Esperado



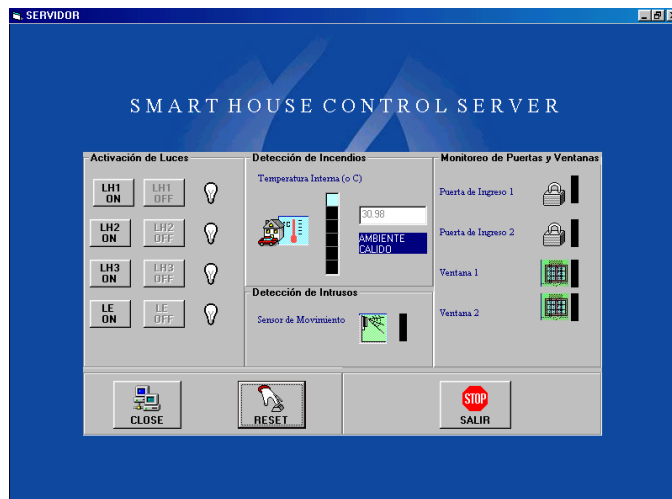
Caso 2

Resultado Esperado



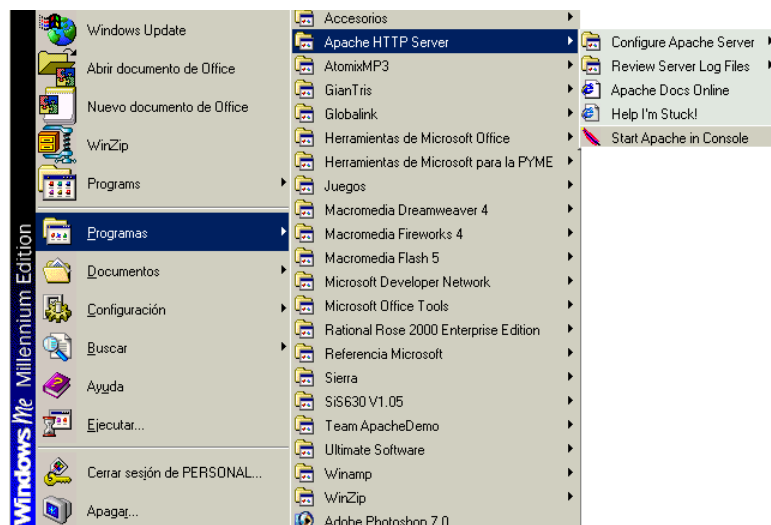
6.23.- PRUEBA INICIALIZAR ESTADOS SHC-SERVER

Resultado Esperado



7.1.- MANUAL DE OPERACIÓN DEL SOFTWARE

- En el Servidor ejecute la operación Start Apache en Console, para subir el servidor de páginas web.

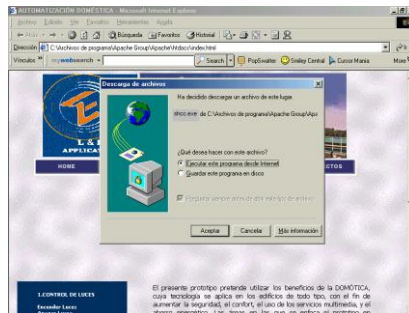


- En el cliente digite `http:\\www.index.html` con lo cual se abrirá la Página Web inicial, desde la cual usted podrá desplazarse hacia las demás páginas. Para levantar estas páginas en el Servidor, basta con ejecutar Internet Explorer.

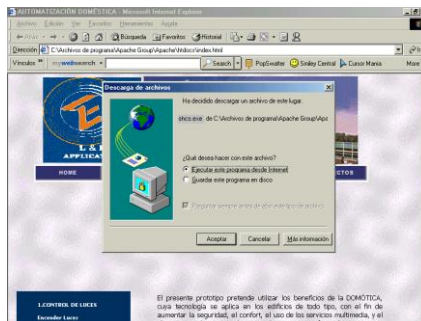


- Para ejecutar la Aplicación Domótica:

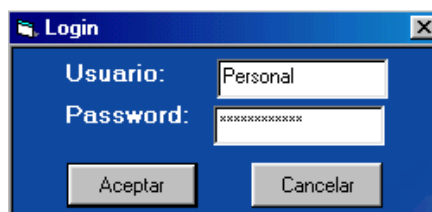
- En el Cliente: Pulse en el botón Proyectos, en esta página seleccione Ejecutar SHCC.exe (Smart House Control Client)
En la ventana que se despliega elija Ejecutar este Programa desde Internet y de clic en aceptar.



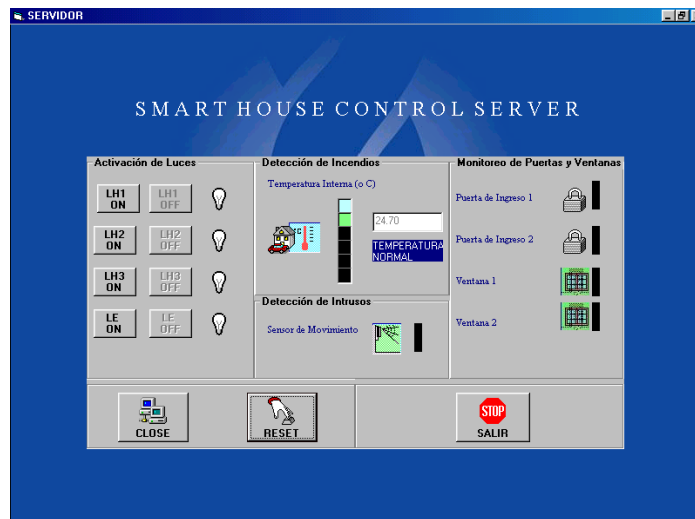
- En el Servidor: Pulse en el botón Proyectos, en esta página seleccione Ejecutar SHCS.exe (Smart House Control Server)
En la ventana que se despliega elija Ejecutar este Programa desde Internet y de clic en aceptar.



- En el Cliente observará una ventana de Login en la que usted escribirá en **Nombre de Usuario:** Su nombre (campo opcional) y en **Password:** controlhouse (campo obligatorio) y de clic en aceptar.



- Tanto en el Cliente, como en el Servidor observará una ventana como la siguiente, desde la cual podrá controlar luces en las diferentes habitaciones del prototipo y visualizar estados de seguridad en puertas y ventanas, detección de intrusos, y visualizar la temperatura interna de la residencia.



7.2.- MANUAL DE OPERACIÓN DEL HARDWARE

- Las placas que controlan al prototipo presentan conectores de tipo hembra, mientras que los conectores procedentes de la maqueta son del tipo macho, para lograr una conexión rápida y sencilla.
- En la placa inferior se observan seis zócalos con la respectiva especificación del dispositivo y acompañado del número de zócalo al que deberá conectarse (numeración izquierda a derecha).

Nota.- Los dispositivos a conectarse son: Sensor Puerta de Ingreso1 (PI1), Sensor Puerta de Ingreso2 (PI2), Sensor de Movimiento (SM), Sensor de Ventana1 (V1), Sensor de Ventana2 (V2).

- De la misma forma, en la placa superior utilizamos la misma especificación anterior para los diez zócalos existentes; con la diferencia de que a estos se conectarán luces y una sirena.

NOTA.- Las luces a conectarse son: Luz Habitación1 (LH1), Luz Habitación2 (LH2), Luz Habitación3 (LH3), Luz Externa (LE); además de una sirena. LH3 y la sirena poseen una conexión a tierra; mientras que LH1, LH2 y LE poseen una tierra común en LH2 Gnd.

- Para conexión a la fuente y entre placas utilizamos números con sus respectivas correspondencias (Ej: Conector1 de la placa inferior, con Conector1 de la placa superior).

NOTA.- La fuente de alimentación para sensores es de 5v y para luces es de 12v.

GLOSARIO DE TERMINOS Y ABREVIATURAS

8.1.- GLOSARIO DE TÉRMINOS

Analógico: Señal continua con muchos posibles valores aleatorios como la distancia y la temperatura.

ADC0804: Es un CI de 20 terminales fabricado con tecnología CMOS y que lleva a cabo la conversión de señales analógicas en digitales.

Amplificadores: Son circuitos que se utilizan para aumentar (amplificar) el valor de la señal de entrada y así obtener una señal a la salida con una forma mucho mayor a la señal de entrada.

Buffer: Amplificador y desacoplador de puerto.

Digital: Sistema que usa muestras digitales (valores discretos codificados en binario) para representar señales analógicas

Diodo: Es un dispositivo semiconductor que permite el paso de la corriente eléctrica en una dirección y la bloque en la opuesta.

Disipador o Refrigerador: Pieza que va asociada a un componente para liberar el calor producido por este.

Domótica: Concepto de vivienda que integra todos los automatismos en materia de seguridad, gestión de la energía, comunicaciones, etc.

Estándar: Reglas que determina el modo en que un dispositivo debe operar.

Evento: Es un suceso que puede causar la transición de un estado a otro de un objeto. Este suceso puede ser una de varias cosas:

LM35: Sensor de temperatura o transductor, convierte la temperatura ambiental (objeto de medición) en una señal eléctrica, cuya función de transferencia es 10 milivoltios(mV) por cada grado centígrado.

LM244 : Es un buffer que permite desacoplar el puerto paralelo de E/S por seguridad, ante posibles sobrecargas.

LM324: Es un amplificador operacional, el cual proporciona ganancia a los valores de entrada.

Memoria de acceso aleatorio o RAM: Memoria basada en semiconductores que puede ser leída y escrita por el microprocesador u otros dispositivos de *hardware*. Es un acrónimo del inglés *Random Access Memory*.

Memoria de sólo lectura o ROM: Memoria basada en semiconductores que contiene instrucciones o datos que se pueden leer pero no modificar. En las computadoras IBM PC y compatibles, las memorias ROM suelen contener el *software* necesario para el funcionamiento del sistema.

Memoria programable y borrable de sólo lectura o EPROM: Son chips de memoria que se programan después de su fabricación; pueden borrarse por lo general, retirando una cubierta protectora de la parte superior del chip y exponiendo el material semiconductor a radiación ultravioleta, después de lo cual pueden reprogramarse. Es un acrónimo inglés *Erasable Programmable Read Only Memory*.

Opto acoplador 4N35: Es un circuito integrado óptico, se utiliza como desacoplador para activar los elementos de salida digitales.

Opto Acopladores 4N35: Desacopla las señales de control de cargas, mediante una señal de luz.

Potenciómetro: Resistencia variable que se usa para graduar intensidad de corriente.

Puentes (Bridges): Son dispositivos que pueden interconectar segmentos de red a través de medios físicos diferentes. Además, pueden adaptar diferentes protocolos de bajo nivel (capa de enlace de datos y física de modelo OSI).

Puentes Ruteadores (Brouters): Son una especie de híbrido entre los puentes y ruteadores. En realidad los puentes ruteadores toman la decisión de si un paquete utiliza un protocolo que pueda ser enrutable. Así, enrutan aquellos que puede y puentean el resto.

Puerto Paralelo: También llamado puerto de la impresora, está diseñado de forma que permita la conexión de impresoras paralelas, más también puede ser usado como un puerto de entrada / salida genérico.

Requisito: Condición o capacidad que debe estar presente en un sistema o componentes de sistema para satisfacer un contrato, estándar, especificación u otro documento formal.

Resistencia: Es un componente electrónico cuya misión consiste en controlar, dosificar y limitar el valor de la corriente, para lo cual aprovecha la resistividad de una barra de carbón.

Voltaje DC: Es el voltaje de corriente continua (+12, -12, +5, -5).

74LS157: Selector Multiplexores de datos de 2 entradas y 1 salida. Selecciona entre 2

señales; así tenemos que si la entrada es A automáticamente la entrada B se bloquea y cuya única salida es Y, lo mismo sucede con la entrada B.

74LS244: Amplifica la señal entre 1 entrada y 1 salida.

8.2.- ABREVIATURAS

Aplication Protocol Data Error	APDE
Aplication Protocol Data Unit	(APDU)
Audio/Video	(A/V)
Common Applications Languajes	(CAL)
Consumer Electronic Bus	(CEBUS)
Critic Redondace Cycle	(CRC)
Data Link Protocol Data Unit	(DLPDU)
Electronic Industries Association	(EIA)
Especificación de Requisitos de Software	(SRS)
Musical Instrument Digital Interface	(MIDI)
Network Protocol Data Unit	(NPDU)
Nivel de Red	(NR)
Nuevas Tecnologías de la Información	(NTI)
Power Line	(PL)
Productos Interactivos Inteligentes	(PII)
Radiate Frequency	(RF)
Red Digital Integrada	(ISDM)
Red Integrada de banda ancha	(DISDM)
Reproductor y/o grabador de video	(VCR)
Subsistemas inteligentes	(SI)
Sistemas centralizados	(SC)

Televisión por antena comunitaria	(CATV)
-----------------------------------	--------