

**UNIVERSIDAD MAYOR DE SAN ANDRÉS
FACULTAD DE CIENCIAS PURAS Y NATURALES
CARRERA DE INFORMATICA**



TESIS DE GRADO

**“DISEÑO E IMPLEMENTACION DE UN SISTEMA DOMOTICO PARA
EL CONTROL DE ENERGIA ELECTRICA”**

PARA OPTAR AL TÍTULO DE LICENCIATURA EN INFORMATICA
MENCION: INGENIERIA DE SISTEMAS INFORMATICOS

POSTULANTE: REYNALDO CORONEL LAZO
TUTOR METODOLOGICO: LIC. GROVER ALEX RODRIGUEZ RAMIREZ
ASESOR: M. Sc. ROSA FLORES MORALES

LA PAZ – BOLIVIA

2014



**UNIVERSIDAD MAYOR DE SAN ANDRÉS
FACULTAD DE CIENCIAS PURAS Y NATURALES
CARRERA DE INFORMÁTICA**



LA CARRERA DE INFORMÁTICA DE LA FACULTAD DE CIENCIAS PURAS Y NATURALES PERTENECIENTE A LA UNIVERSIDAD MAYOR DE SAN ANDRÉS AUTORIZA EL USO DE LA INFORMACIÓN CONTENIDA EN ESTE DOCUMENTO SI LOS PROPÓSITOS SON ESTRICTAMENTE ACADÉMICOS.

LICENCIA DE USO

El usuario está autorizado a:

- a) visualizar el documento mediante el uso de un ordenador o dispositivo móvil.
- b) copiar, almacenar o imprimir si ha de ser de uso exclusivamente personal y privado.
- c) copiar textualmente parte(s) de su contenido mencionando la fuente y/o haciendo la referencia correspondiente respetando normas de redacción e investigación.

El usuario no puede publicar, distribuir o realizar emisión o exhibición alguna de este material, sin la autorización correspondiente.

TODOS LOS DERECHOS RESERVADOS. EL USO NO AUTORIZADO DE LOS CONTENIDOS PUBLICADOS EN ESTE SITIO DERIVARA EN EL INICIO DE ACCIONES LEGALES CONTEMPLADOS EN LA LEY DE DERECHOS DE AUTOR.

DEDICATORIA

El presente trabajo de investigación está dedicado con todo cariño y amor para mi Madre Elena Lazo Mamani por toda la confianza depositada en mi persona y mi padre Máximo Ramón Coronel Lazo por el apoyo moral que me brindó y gracias a Dios por bendecirme con unos padres en esta tierra ¡gracias!

AGRADECIMIENTOS

Quiero agradecer a mi asesora: Mg. Sc. Rosa Flores Morales por la comprensión y tiempo que le dedico a mi proyecto de investigación por los consejos que me dio en el momento preciso sobre todo por los comentarios que enriquecieron mi trabajo de investigación.

Quiero agradecer a mi tutor metodológico Lic. Grover Alex Rodriguez Ramirez por la orientación el apoyo la motivación y las recomendaciones.

Un agradecimiento muy especial a todos mis familiares por el apoyo la confianza que depositaron en mi persona permitiéndome lograr la culminación de este proyecto de investigación.

Finalmente quiero agradecer a mis hermanos Betty por brindándome su atención, Adan por la paciencia que supo darme. Marcos, Guadalupe y Olga Magaly por la comprensión en momentos de dedicación a mi familia y sobre todo por la importancia que tiene una formación académica, gracias a mis compañeros de la carrera de informática.

RESUMEN

El presente trabajo de tesis demuestra la posibilidad de reducir el consumo de la energía eléctrica mediante la implementación de un prototipo de un sistema domótico, para el control de energía eléctrica se realiza el control de dispositivos como la iluminación y la temperatura de una vivienda.

El acceso a los sistemas domóticos no se han desarrollado en nuestro medio pese a las ventajas que conlleva su implementación en las edificaciones, se tiene poco conocimiento respecto al hardware y software libre que permita la construcción de estos sistemas.

En este trabajo se construyó un sistema domótico para el control específicamente de la iluminación y la temperatura ambiental mediante una tarjeta Arduino que permite extender módulos como Shield de Ethernet para el control mediante la web. La lógica del controlador se implementó con lógica difusa esto para la automatización de procesos que permite reducir el consumo de energía eléctrica también el control directo de los dispositivos se realiza mediante la web obteniendo el estado de los dispositivos conectados al controlador en tiempo real tanto en el hardware como en las especificaciones del software se logra resultados importantes

Este sistema además es posible controlarlo remotamente desde una página web, la misma que se implementó gracias a la placa electrónica Arduino y Shield Ethernet este último permite la navegación por la red de internet como parte del control y monitoreo de estado de los ambientes de un hogar.

En función a los resultados obtenidos a partir de la implementación del sistema de control domótico se realiza un análisis de consumo de energía eléctrica con domótica y sin domótica contrastando con métodos estadísticos, generamos la validez de los resultados realizando comparaciones determinando que la implementación del sistema de control domótico reduce el consumo de energía eléctrica así como el costo de la implementación.

INDICE DE CONTENIDO

1. MARCO REFERENCIAL	1
1.1. Introducción	1
1.2. Antecedentes	2
1.3. Planteamiento del Problema.....	3
1.4. Justificación	4
1.5. Hipótesis	5
1.6. Objetivos	6
1.6.1. Objetivo General.....	6
1.6.2. Objetivos Específicos	6
1.7. Alcances	6
1.8. Metodología	7
1.9. Aportes	9
2. MARCO TEORICO	10
2.1. Introducción	10
2.1.1. Tipos de Edificios Inteligentes	11
2.1.1.1. Inmótica	11
2.1.1.2. Hogar Digital	12
2.2. Gestión Domótica	14
2.2.1. Gestión Energética.....	14
2.2.2. Confort.....	15
2.2.3. Seguridad	15
2.2.4. Comunicación.....	15
2.2. Arquitectura	16
2.3.1. Arquitectura Centralizada.....	16
2.3.2. Arquitectura Descentralizada.....	17
2.3.3. Arquitectura Distribuida	18
2.4. Topología de Redes.....	19
2.4.1. Topología en Estrella.....	19
2.4.2. Topología en Bus.....	20
2.4.3. Topología en Anillo.....	21
2.5. Medios de Transmisión y comunicación	21

2.5.1.	Corrientes Portadoras.....	22
2.5.2.	Soporte Metálico.....	22
2.5.2.1.	Par Metálico	22
2.5.2.2.	Cable Coaxial.....	23
2.5.3.	Fibra Óptica	24
2.5.4.	Conexión sin Hilos	24
2.5.4.1.	Infrarrojos	24
2.5.4.2.	Radiofrecuencia	24
2.6.	Protocolos de Comunicación	25
2.6.1.	Tipo de Protocolos.....	25
2.6.1.1.	Propietario o cerrado.....	25
2.6.1.2.	Estándar o abierto	25
2.7.	Componentes de un sistema domótico.....	28
2.7.1.	Características.....	28
2.7.2.	Tipos de Sensores	30
2.7.2.1.	Según el tipo Alimentación.....	30
2.7.2.2.	Según el tipo de tipo de señal implicada.....	30
2.7.2.3.	Según el ámbito de aplicación	31
2.7.3.	Sensor de Luminosidad.....	33
2.7.3.1.	Sensor de luz LDR (Light Dependent Resistor)	33
2.7.4.	Sensor de Temperatura	34
2.7.4.1.	Sensor de temperatura LM35.....	34
2.8.	Placas Electrónicas.....	36
2.8.1.	Arduino	36
2.8.2.	Características Arduino	36
2.8.3.	Shield Ethernet.....	37
2.8.4.	Características Shield Ethernet.....	38
2.8.6.	Entorno de Desarrollo.....	40
2.8.6.1.	Estructura Básica de un Programa	40
2.8.6.2.	Funciones	41
2.8.6.3.	Variables	41
2.8.6.4.	Tipos de Datos	41
2.8.6.5.	Operadores Aritméticos	42
2.8.6.6.	Sentencias Condicionales.....	43

2.8.6.7.	Funciones de Tiempo y Matemáticas Delay(ms).....	44
2.8.6.8.	Puerto Serie Serial.begin(rate)	44
2.9.	Metodología	45
2.9.1.	Planificación	46
2.9.1.1.	Historias de usuarios	46
2.9.1.2.	Plan de entregas (“Release Plan”).....	46
2.9.1.3.	Plan de iteraciones (“Iteration Plan”).....	47
2.9.2.	Diseño	47
2.9.1.1.	Simplicidad	47
2.9.1.2.	Recodificación	48
2.9.3.	Desarrollo	48
2.9.3.1.	Disponibilidad del cliente	48
2.9.3.2.	Uso de estándares.....	48
2.9.3.3.	Integraciones permanentes.....	48
2.9.3.4.	Propiedad colectiva del código	49
2.9.3.5.	Ritmo sostenido	49
2.9.4.	Pruebas.....	49
2.9.4.1.	Pruebas Unitarias	49
2.9.4.2.	Detección y corrección de errores.....	49
2.9.4.3.	Pruebas de Aceptación	49
2.10.	Lógica Difusa	50
2.10.1.	Concepto de Conjunto Difuso.....	50
2.10.1.1.	Conjuntos Clásicos	50
2.10.1.2.	Conjuntos Difusos	51
3.	DISEÑO METODOLOGICO	52
3.1.	Introducción	52
3.2.	Arquitectura del Controlador	52
3.3.	Materiales del controlador	53
3.4.	Construcción del Controlador	54
3.5.	Funcionamiento del controlador	58
3.5.1.	Entrada.....	58
3.5.2.	Proceso.....	58
3.5.3.	Salida	58
3.6.	Control Mediante Lógica Difusa	59

3.6.1.	Fusificación de Entradas.....	59
3.6.2.	Base de Reglas.....	62
3.6.3.	Defusificación.....	63
3.7.	Desarrollo de Software	67
3.7.1.	Planificación.....	67
3.7.1.1.	Propósito del sistema	67
3.7.1.2.	Historias de Usuario.....	68
3.7.2.	Diseño.....	71
3.7.2.1.	Especificación de requerimientos	71
3.7.2.2.	Modelo Entidad Relación del Prototipo.....	72
3.7.2.3.	Desarrollo.....	72
3.7.3.	Codificación.....	80
3.7.4.	Pruebas.....	82
4.	EVALUACION DE RESULTADOS.....	86
4.1.	Introducción	86
4.2.	Datos obtenidos para la demostración	86
4.3.	Demostración de la Hipótesis	102
4.4.	Costo del Controlador Domótico	103
5.	CONCLUSIONES Y RECOMENDACIONES	104
5.1.	Introducción	104
5.2.	Cumplimiento de objetivos	104
5.3.	Solución de Problemas.....	105
5.4.	Estado de la Hipótesis	105
5.5.	Recomendaciones	106
5.6.	Trabajos Futuros	106
	REFERENCIAS BIBLIOGRAFICAS	107
	Anexo.....	109

INDICE DE FIGURAS

Figura	Comentario	Página
Capitulo II		
FIGURA 2. 1	HOGAR DIGITAL	13
FIGURA 2. 2	ARQUITECTURA CENTRALIZADA	16
FIGURA 2. 3	ARQUITECTURA DESCENTRALIZADA.....	17
FIGURA 2. 4	ARQUITECTURA DISTRIBUIDA.....	18
FIGURA 2. 5	TOPOLOGÍA EN ESTRELLA	20
FIGURA 2. 6	TOPOLOGÍA EN BUS	20
FIGURA 2. 7	TOPOLOGÍA EN ANILLO	21
FIGURA 2. 8	TIPOS DE MEDIOS DE TRANSMISIÓN.....	22
FIGURA 2. 9	ARDUINO UNO REV 3	36
FIGURA 2. 10	SHIELD ETHERNET	39
FIGURA 2. 11	ENTORNO DE DESARROLLO	40
FIGURA 2. 12	PROGRAMACIÓN EXTREMA.....	45
FIGURA 2. 13	CONJUNTOS DIFUSOS	51
Capitulo III		
FIGURA 3. 1	ARQUITECTURA DEL CONTROLADOR	53
FIGURA 3. 2	CONEXIÓN COMPUTADORA PLACA ELECTRÓNICA.....	54
FIGURA 3. 3	MONTAJE DE LA PLACA ARDUINO Y SHIELD ETHERNET.....	55
FIGURA 3. 4	CONFIGURACIÓN DEL CIRCUITO DE TEMPERATURA.....	55
FIGURA 3. 5	CONFIGURACIÓN DEL CIRCUITO DE ILUMINACIÓN	56
FIGURA 3. 6	INTEGRACIÓN DE LOS MÓDULOS	56
FIGURA 3. 7	INTEGRACIÓN DEL SISTEMA DE CONTROL.....	57
FIGURA 3. 8	INTEGRANDO LOS SENSORES A LA VIVIENDA.....	57
FIGURA 3. 9	CONJUNTO DIFUSO PARA LA ILUMINACIÓN.....	60

FIGURA 3. 10 CONJUNTO DIFUSO PARA LA TEMPERATURA	61
FIGURA 3. 11 VELOCIDAD DEL VENTILADOR	64
FIGURA 3. 12 INTENSIDAD LUMÍNICA	65
FIGURA 3. 13 FUNCIÓN DE PERTENENCIA GLOBAL DE LA TEMPERATURA	66
FIGURA 3. 14 FUNCIÓN DE PERTENENCIA GLOBAL DE LA ILUMINACIÓN	66
FIGURA 3. 15 DIAGRAMA DE BLOQUES DEL CONTROLADOR DOMÓTICO	68
FIGURA 3. 16 MODO SERVIDOR	75
FIGURA 3. 17 CLIENTE.....	75
FIGURA 3. 18 SOLICITUD CLIENTE.....	76
FIGURA 3. 19 RESPUESTA SERVIDOR.....	77
FIGURA 3. 20 PETICIÓN TARJETA SD.....	77
FIGURA 3. 21 DIAGRAMA DE FLUJO ARDUINO CONEXIÓN WEB.....	78
FIGURA 3. 22 DIAGRAMA DE FLUJO SISTEMA GENERAL	79
FIGURA 3. 23 INTERFAZ DEL PROGRAMA ARDUINO	80
FIGURA 3. 24 FRAGMENTO CÓDIGO CONTROL AUTOMÁTICO	81
FIGURA 3. 25 FRAGMENTO CÓDIGO CONTROL MANUAL	81
FIGURA 3. 26 INICIO DE SESIÓN PARA USUARIOS	82
FIGURA 3.27 OPCIONES DE CONTROL.....	83
FIGURA 3. 28 OPCIONES DE MODO	83
FIGURA 3. 29 MODO MANUAL.....	84
FIGURA 3. 30 MODO AUTOMÁTICO	85
FIGURA 3. 31 ADMINISTRACIÓN DE USUARIOS	85

INDICE DE TABLAS

Tabla	Página
TABLA 2. 1 DESCRIPCIÓN DE TIPO DE SENSORES	32
TABLA 3. 1 COMPONENTES PARA LA CONSTRUCCIÓN DEL SISTEMA	53
TABLA 3. 2 PARÁMETROS DE LA TEMPERATURA.....	59
TABLA 3. 3 PARÁMETROS DE LA ILUMINACIÓN	59
TABLA 3. 4 HISTORIA DE USUARIO ACCESO AL SISTEMA DE CONTROL.....	69
TABLA 3. 5 HISTORIA DE USUARIO CONTROL AUTOMÁTICO	69
TABLA 3. 6 HISTORIA DE USUARIO CONTROL MANUAL.....	70
TABLA 3. 7 HISTORIA DE USUARIO ADMINISTRAR	71
TABLA 3. 8 PROCESOS DEL PROTOTIPO	72
TABLA 4. 1 SALÓN DE UN DEPARTAMENTO.....	87
TABLA 4. 2 DORMITORIO DE UN DEPARTAMENTO.....	87
TABLA 4. 3 COCINA DE UN DEPARTAMENTO.....	87
TABLA 4. 4 PASILLO DE UN DEPARTAMENTO.....	88
TABLA 4. 5 SALÓN DE UN DEPARTAMENTO.....	88
TABLA 4. 6 DORMITORIO DE UN DEPARTAMENTO.....	89
TABLA 4. 7 COCINA DE UN DEPARTAMENTO.....	89
TABLA 4. 8 PASILLO DE UN DEPARTAMENTO.....	89
TABLA 4. 9 SALÓN DE UN DEPARTAMENTO.....	90
TABLA 4. 10 DORMITORIO DE UN DEPARTAMENTO.....	90
TABLA 4. 11 COCINA DE UN DEPARTAMENTO.....	91
TABLA 4. 12 PASILLO DE UN DEPARTAMENTO.....	91
TABLA 4. 13 SALÓN DE UN DEPARTAMENTO.....	92
TABLA 4. 14 DORMITORIO DE UN DEPARTAMENTO.....	92
TABLA 4. 15 COCINA DE UN DEPARTAMENTO.....	92
TABLA 4. 16 PASILLO DE UN DEPARTAMENTO.....	93
TABLA 4. 17 SALÓN DE UN DEPARTAMENTO.....	93

TABLA 4. 18 DORMITORIO DE UN DEPARTAMENTO.....	94
TABLA 4. 19 COCINA DE UN DEPARTAMENTO.....	94
TABLA 4. 20 PASILLO DE UN DEPARTAMENTO.....	94
TABLA 4. 21 SALÓN DE UN DEPARTAMENTO.....	95
TABLA 4. 22 DORMITORIO DE UN DEPARTAMENTO.....	95
TABLA 4. 23 COCINA DE UN DEPARTAMENTO.....	96
TABLA 4. 24 PASILLO DE UN DEPARTAMENTO.....	96
TABLA 4. 25 SALÓN DE UN DEPARTAMENTO.....	97
TABLA 4. 26 DORMITORIO DE UN DEPARTAMENTO.....	97
TABLA 4. 27 COCINA DE UN DEPARTAMENTO.....	97
TABLA 4. 28 PASILLO DE UN DEPARTAMENTO.....	98
TABLA 4. 29 SALÓN DE UN DEPARTAMENTO.....	98
TABLA 4. 30 DORMITORIO DE UN DEPARTAMENTO.....	99
TABLA 4. 31 COCINA DE UN DEPARTAMENTO.....	99
TABLA 4. 32 PASILLO DE UN DEPARTAMENTO.....	99
TABLA 4. 33 RESULTADOS SIN DOMÓTICA.....	100
TABLA 4. 34 RESULTADOS CON DOMÓTICA.....	101
TABLA 4. 35 COMPARACIÓN CON DOMÓTICA Y SIN DOMÓTICA	101

1. MARCO REFERENCIAL

1.1. Introducción

El constante crecimiento de la tecnología en las telecomunicaciones y la electrónica, nos conlleva a desarrollar sistemas informáticos cada vez más complejos para satisfacer las necesidades que existe en la sociedad, el poder brindar control, automatización, y optimización son procesos que mejoran la calidad de vida.

Gracias a la evolución de la electrónica, áreas como las telecomunicaciones e informática han tenido un gran desarrollo. Con la incorporación de nuevas tecnologías al hogar surge la idea de automatizar la vivienda de forma que brinde mayores comodidades y necesite menor dedicación a estas. Es así como surge el concepto de domótica.

Servicios que ofrece la domótica a nuestro ámbito cotidiano son un aumento de comodidad y seguridad, aporte de información sobre el estado del hogar y ahorro energético que se puede conseguir. Todas estas ventajas se pueden resumir en un aumento de la calidad de vida del propietario de la vivienda.

Hay empresas que ofrecen servicios de domótica cuyo importe es elevado e inaccesible en nuestro medio por el costo de importación porque este sistema domótico implica la incorporación de sensores actuadores y el software que permita controlar todos estos dispositivos además, el protocolo que utiliza para su instalación es propia de cada empresa.

El uso de microcontroladores aplicado a las viviendas nos permite recibir información (nivel de temperatura, luz) para poder determinar qué acción tomar (aumentar la temperatura o bajar, de igual manera la iluminación) esto de forma automática con facilidades de comunicación mediante teléfonos móviles y computadoras. Los sistemas inteligentes para casas, integran y controlan áreas como la informática, las comunicaciones,

la seguridad, ambiente climático, la iluminación, brindando beneficios en cuanto a confort, seguridad y ahorro de energía eléctrica.

Sin embargo, a pesar de las ventajas que ofrece instalar una solución domótica en un hogar, la domótica ha tenido poco éxito en el mercado, limitándose en gran medida a su instalación en viviendas de nueva construcción.

El objetivo de este proyecto, basándose en proyectos ya desarrollados, es el de crear un sistema de control domótico que mantenga las ventajas anteriormente mencionadas y que solucione algunas de las ventajas de los sistemas actuales que provocan que la domótica no se extienda a cualquier tipo de usuario. Las características que debe reunir el sistema en cuestión son: fácil instalación, autónomo, sencillo de utilizar y que sea accesible económicamente. Además de estas características, el sistema implementara una solución web que permite ver el estado del sistema domótico a través de internet en cualquier momento. La domótica y la lógica difusa generan una aplicación que determina comportamientos de nivel respecto temperatura y luminosidad, la cual permitirá controlar y automatizar de forma adecuada y eficiente.

1.2. Antecedentes

El origen de los edificios inteligentes se remonta a los años 70, donde se consideró el sistema con procesadores de computadora, pero había una carencia de integración con otros sistemas. En los años 80 aparece la automatización en iluminación, seguridad y protección del ocupante pero continúan siendo sistemas independientes. A los finales de los 80 y principios de los 90 aparece la segunda generación de edificios inteligentes donde ya se consideraba la integración de los sistemas independientes basados en computadora pero se incluye el ahorro de energía como tema principal. Desde los años 2000 a la fecha la tendencia es concentrar su atención en economía y sensibilidad de ambientación, pero que sea capaz de adaptarse al rápido crecimiento tecnológico y los cambios constantes del mundo moderno.

Mucho antes de que el término domótica surgiera, se hablaba del protocolo X-10, éste era y aún sigue siendo el lenguaje que usan algunos dispositivos para comunicarse entre ellos, todos los dispositivos X-10 son compatibles entre sí, combinándose para formar sistemas, permitiéndoles controlar luces y electrodomésticos, aprovechando las instalaciones eléctricas sin instalar cables adicionales; el sistema X-10 desde sus inicios ha tenido una ventaja con relación a otros sistemas domóticos, éste solo utiliza la red eléctrica interna de la vivienda, conectando solo un módulo que maneja las funcionalidades y cualquier otro dispositivo que se conectara a ese modulo podría ser controlado en forma remota. X-10 la tecnología de corrientes portadoras fue desarrollado entre 1976-1978 por ingenieros de Pico Electronics Ltd, proveniente de la familia de chips de la serie X, ideándolo como un circuito para implementarlo en un dispositivo y que pudiera ser controlado remotamente.

Los proveedores de servicio controlan de forma remota la energía y tendrán el control sobre la demanda. De allí que un papel fundamental del futuro serán los científicos sociales, quienes tendrán partida en el desarrollo del entendimiento de todo el mercado domótico, haciendo que las personas entiendan la tecnología y la usen de la forma más adecuada.

1.3. Planteamiento del Problema

La implementación de sistemas domóticos en nuestro medio es inexistente debido al costo elevado que ésta presenta en cuanto a dispositivos tecnológicos y por otro lado al desconocimiento de nuevas tecnologías y soluciones tanto en software como en hardware libre que permitan automatizar viviendas.

Los principales problemas identificados son detallados a continuación.

- Los sistemas domóticos no permiten la ampliación de módulos según la necesidad que tengan estas viviendas debido a que tienen sus protocolos patentados por la empresa que ofrece los servicios.
- La implementación de estos sistemas requieren de un diseño arquitectónico por lo que su implementación se basa en nuevas construcciones.

- El conocimiento que se tiene de tecnologías nuevas tanto en software como en hardware libre para procesos de automatización en nuestro medio es insuficiente.
- El sistema de iluminación de una vivienda y el aire acondicionado permanecen en un estado activo cuando no se requiere, como efecto tenemos consumo innecesario.

1.4. Justificación

Analizando el estado actual de la domótica, y de los diferentes tipos de sistemas de control domótico que se han desarrollado hasta el momento, se evidencia claramente el avance tecnológico que ha tenido este tipo de sistemas. Aunque son una infinidad los proyectos de domótica que han creado, en nuestro medio son proyectos que no tuvieron mucho impacto pero que es importante promover el concepto de la domótica, Se ha visto muy poco la implementación de sistemas de control domótico controlados por medio de la web. Este tipo de sistemas de control, podrían generar un auge, ya que permiten una mayor interacción entre el usuario y el sistema, y a la vez mayor comodidad para la persona, debido a que puede controlar su vivienda por medio de la web desde cualquier punto donde se encuentre de forma que pueda ver el estado que tiene los diferentes dispositivos conectados al sistema de control.

El mercado actual se encuentran dispositivos electrónicos portables en su gran mayoría con conexión a internet por lo tanto hay un crecimiento acelerado, donde las tendencias y las preferencias de los usuarios cada vez es más notoria, donde los dispositivos con tecnología web están marcando la diferencia esto permite poder integrar de forma directa con nuestro sistema de control. También el presente trabajo profundiza la investigación más sobre la automatización y el control de módulos sensoriales sobre la energía eléctrica controlar el consumo innecesario de la energía eléctrica de forma que a futuro no sea un problema el consumo excesivo del energético.

Este sistema nos permitirá el ahorro al momento de pagar facturas elevadas por consumo de energía eléctrica, un sistema de este tipo en el mercado es inaccesible debido a

su elevado costo por importación entonces la implementación de este sistema nos permite contar con los recursos necesarios y tecnológicos para desarrollar este controlador.

La automatización de dispositivos en ambientes se hace cada vez más imprescindible debido a la comodidad que ofrece estos sistemas de manera que contribuye en particular este trabajo.

1.5. Hipótesis

La implementación de un prototipo de control domótico construido con una placa Arduino y software libre reducirá el consumo de energía eléctrica y el costo del controlador respecto a aplicaciones comerciales.

Variables de Análisis

X₁: Consumo de Energía Eléctrica en viviendas.

X₂: Costo del controlador

Indicador: Porcentaje

Instrumento de Medida: Comparación del consumo generado por mes en el prototipo, esto se realizara cuando se lleve a cabo la implementación del sistema de control domótico que permitirá la posterior adaptación del sistema en viviendas.

Indicador: Costo del controlador

Instrumento de Medida: Comparación de costo del controlador domótico frente al costo del controlador domótico comercial

1.6. Objetivos

1.6.1. Objetivo General

Diseñar un prototipo que permita automatizar tareas rutinarias como la iluminación y la temperatura de un ambiente, el control directo sobre los sensores y actuadores mediante páginas web como parte del control y la automatización para lograr la reducción del consumo de la energía eléctrica.

1.6.2. Objetivos Específicos

- Diseñar el software prototipo que permita el control de dispositivos como los sensores y actuadores.
- Analizar circuitos electrónicos Arduino para la comunicación con la computadora y dispositivos domóticos por puerto USB v2.0.
- Analizar circuitos módulo shield Ethernet para el control mediante la web.
- Diseñar e implementar hardware para la comunicación sobre los dispositivos sensores, actuadores.
- Desarrollar una página web para el control de sensores de iluminación y temperatura en modos manual y automático.
- Integrar los elementos software y hardware
- Probar y evaluar el funcionamiento del prototipo de sistema domótico.

1.7. Alcances

En este trabajo de investigación se realizará un controlador de dispositivos domótico y con un software propio para la automatización de dispositivos de una vivienda con mayor énfasis en la luz eléctrica y el control de la temperatura, esto gracias a los sensores de temperatura, sensores de iluminación para la prueba de este trabajo se realizara una maqueta de la vivienda.

Es importante tomar en cuenta que el presente proyecto de tesis no abarca como parte del ahorro de energía eléctrica el control sobre las cámaras web monitorización de ambientes en seguridad específicamente el trabajo de investigación está enfocado al control de la iluminación y temperatura de un ambiente .

El sistema domótico desarrollado en el presente trabajo muestra las siguientes características:

- El sistema se experimenta en una maqueta alimentada con corriente continua de 5V.
- Todos los componentes que conforman las pruebas responden a tensiones de 5V.
- Los sensores y actuadores también responden a estos niveles de tención.
- Los sensores son: sensores de temperatura sensores de iluminación como parte fundamental del sistema.
- El control se aplicara a la iluminación y la temperatura ambiente de forma automatizada.
- Las pruebas se realizan a nivel local por parte de la navegación web.

1.8. Metodología

Inicialmente se analiza los medios hardware y software para el sistemas domótico de control eléctrico, se estudia los elementos a controlarse y automatizarse, se integran los módulos actuadores y sensores al módulo Arduino pero también la comunicación entre un cliente y el servidor Arduino que proporciona páginas web para el control domótico a partir de la tarjeta SD. Se define la lógica de control en términos de programas en el Arduino, se experimentara en un maqueta el funcionamiento del nuevo sistema, finalmente se evalúa los resultados y analiza el estado de la hipótesis.

Para el desarrollo de este proyecto domótico basado en controladores lógicos programables. Es importante considerar la metodología clara y detallada en este proyecto de grado se utilizara el método científico que permitirá la realización de los siguientes puntos.

➤ **Planteamiento del Problema**

La implementación de estos sistemas implica la integración de varios componentes tales como control, monitoreo, comunicación, ahorro y automatización de tareas rutinarias para tal efecto acudimos a técnicas de modelado que permitan y nos faciliten el desarrollo realizara la recopilación de la información necesaria acerca del problema mediante consultas bibliográficas y sitios de internet que impliquen los temas de diseño e implementación de sistemas domóticos.

➤ **Especificación de la Hipótesis**

Mediante el análisis y estudio de la información recopilada y tomando en cuenta las características del problema se realizara la formulación de la hipótesis.

➤ **Análisis y Verificación mediante prototipo**

Entonces una vez analizada esta información se procederá a la implementación del prototipo de control domótico.

➤ **Verificación de Resultados**

Mediante casos de prueba con el prototipo se evaluarán los resultados tanto obtenidos como esperados de esta manera se pueda comprobar el estado de la hipótesis planteada contrastando su cumplimiento.

➤ **Conclusiones**

Finalmente se detalla un informe acerca de la investigación y desempeño del sistema domótico proporcionando además las conclusiones y recomendaciones para posteriores investigaciones.

1.9. Aportes

El presente trabajo de investigación hace un profundo estudio sobre la domótica de forma que contribuya al consumo adecuado de la energía eléctrica esto básicamente gracias a los sensores y actuadores.

La domótica en términos generales ayuda y contribuye a mejorar la calidad de vida del usuario, en los siguientes acápite se muestra los aportes de estos sistemas:

- El estudio de nuevos dispositivos electrónicos sobre todo en placas electrónicas introducidos en nuestro medio y de forma accesible sacar beneficios que contribuyan tanto en las áreas de la informática y la electrónica desarrollando nuevos prototipos que sean útiles a nuestra sociedad.
- Gestionar de manera automática la iluminación, climatización de ambientes de un hogar para comodidad del usuario.
- Facilita el manejo de los dispositivos del hogar a las personas con discapacidades de la forma que más se ajuste a sus necesidades.
- Garantiza las comunicaciones con recepción de avisos de anomalías e información del funcionamiento de equipos e instalaciones, gestión remota del hogar, etc.

2. MARCO TEORICO

2.1. Introducción

El origen de la domótica se remonta a la década de los setenta, cuando tras muchas investigaciones aparecieron los primeros dispositivos de automatización de edificios basados en la aun exitosa tecnología X-10. Durante los años siguientes la comunidad internacional mostro un creciente interés por la búsqueda de la casa ideal, comenzando diversos ensayos con avanzados electrodomésticos y dispositivos automáticos para el hogar. Los primeros sistemas comerciales fueron instalados, sobre todo, en Estados Unidos y se limitaban a la regulación de la temperatura ambiente de los edificios de oficinas y poco más. Más tarde, tras el auge de los PC (Personal Computer), a finales de la década de los 80 y principios de la de los noventa, se empezaron a incorporar en estos edificios los SCE (Sistema de Cableado Estructurado) para facilitar la conexión de todo tipo de terminales y periféricos entre sí, utilizando un cableado estándar y tomas repartidas por todo el edificio. Además de los datos, estos sistemas de cableado permitían el transporte de la voz y la conexión de algunos dispositivos de control y de seguridad, por lo que a estos edificios que disponían de un SCE se les empezó a llamar **edificios inteligentes**.

Posteriormente, los automatismos destinados a edificios de oficinas, se han ido aplicando también a las viviendas particulares u otro tipo de edificios, donde el número de necesidades a cubrir es mucho más amplio, dando origen a la **vivienda domótica**. Los diccionarios franceses incorporaron el termino domotique a partir de 1998. Esta palabra, traducida al castellano por domótica, es originaria de la palabra latina domus (de la que ha derivado la raíz domo que quiere decir casa) y de la palabra francesa informatique (de la que ha derivado la palabra informática) o, según otros autores, robotique (robotica). Este termino de uso común en España, no ha conseguido, por el momento, imponerse en diversos países de la Iberoamérica; donde aún se han quedado con el termino inteligente, sin avanzar hacia la diferenciación entre domótica e inmótica.

Los orígenes de la domótica en España se dieron en los años noventa, en los que tienen lugar las primeras iniciativas de promociones y el mayor conocimiento de sus beneficios; pero no será hasta estos dos últimos años cuando pasa a ser un concepto notable conocido por la sociedad. En la actualidad, el número de viviendas domotizadas es todavía relativamente bajo respecto al total de viviendas, pero el interés en su adopción está creciendo progresivamente. Del mismo modo en que en nuestros días no es aceptable que una vivienda no tenga corriente eléctrica o agua corriente, dentro de muy poco no se concebirán viviendas que no estén mínimamente domotizadas. El principal problema que se encontraba para la introducción de la domótica era que muy pocas personas estaban dispuestas a pagar los costes adicionales que implica construir una vivienda inteligente, pero el actual descenso de los precios, ha hecho de la vivienda domótica un sueño asequible [Huidobro, 2010].

Considerando esta información se puede ver que la domótica tiene un origen anglosajón y además que la importancia de la implementación de estos sistemas en esta parte del continente, también se tendría que pensar como una necesidad más que un lujo debido a que se puede disminuir costos de elevadas facturas sobre consumo de energía eléctrica de ahí la importancia de implementar estos sistemas con la ayuda de la informática y la tecnología en cuestión de dispositivos electrónicos se desarrollaría estos sistemas de automatización.

2.1.1. Tipos de Edificios Inteligentes

La automatización de edificios inteligentes conlleva a un estudio amplio sobre los tipos que existen mencionar cada uno de estos no se podría terminar porque simplemente aún siguen implementando sistemas de automatización en edificios con diferentes características pero es importante conocer las que tienen mayor relación a este proyecto por las cuales renombramos a continuación.

2.1.1.1. Inmótica

El termino inmótica, también llamado como "building management system", que hace referencia a la coordinación y gestión de las instalaciones con que se equipan las

edificaciones, así como a su capacidad de comunicación, regulación y control. La inmótica motiva la productividad en el trabajo al gestionar las instalaciones del edificio como una herramienta para favorecer la producción de los empleados que se encuentran en su interior. El origen de este término es francés, y aunque es de uso común en España. Todavía no ha sido recogido por el diccionario de la Real Academia. Mientras la palabra domótica se aplica al hogar, por inmótica se entiende la incorporación de sistemas de gestión técnica automatizada a las instalaciones del sector terciario como son plantas industriales, hoteles, hospitales, aeropuertos, edificios de oficinas, parques tecnológicos, grandes superficies, universidades, instalaciones comunitarias en edificios de viviendas. En realidad, los sistemas y aplicaciones inmóticas son muy similares a la domótica y por ello, a menudo se emplea el concepto de sistemas domóticos referidos también a este sector [Huidobro, 2010].

2.1.1.2. Hogar Digital

Como se ha expuesto, la domótica también se suele asociar actualmente, sobre todo en ámbitos de telecomunicaciones, al denominado **Hogar Digital u hogar conectado**. El gran progreso tecnológico sufrido por los sistemas de telecomunicación y el desarrollo y proliferación de internet, han incrementado exponencialmente nuestra capacidad para crear información, almacenarla, transmitirla, recibirla, y procesarla. El mayor acceso a la información, ha venido además asociado a una mayor facilidad para comunicarnos, para establecer nuevas vías de dialogo con el resto del mundo, en cualquier momento y desde cualquier lugar. Tras una etapa de introducción lenta de la tecnología digital, ahora estamos en los comienzos de una revolución de servicios para el hogar en la figura 2.1 se muestra los componentes básicos de un hogar digital, donde las pasarelas residenciales, apoyadas con conexiones de banda ancha, conectarán inteligentemente todos los dispositivos del hogar, soportando servicios interactivos y de valor añadido de diversa índole [Huidobro, 2010].

Algunas de las ventajas de un hogar digital en cuanto a climatización y consumo energético:

- Programación del encendido y apagado de todo tipo de aparatos (calderas, aire acondicionado, toldos, luces, etc.), según las condiciones ambientales.

- Acomodación a los planes de tarifas reducidas (tarifa nocturna).
- Contadores electrónicos que informan del consumo energético.

Entretenimiento y confort.

- Conexión a internet desde cualquier punto. Juegos en red.
- Visión de canales de TV (televisión) en cualquier habitación.
- Control de los dispositivos eléctricos/electrónicos del hogar, desde un PC, por internet, o desde un teléfono móvil.

Seguridad

- Configuración de procedimientos de avisos en caso de intrusión o avería (alarma técnica).
- Instalación de cámaras y micrófonos para ver lo que ocurre.
- Control del acceso a la vivienda.

Servicios Comunitarios.

- Control de la iluminación de las zonas comunes.
- Manejo de alarmas de seguridad y alarmas técnicas.
- Servicios Web para la comunidad de propietarios.



Figura 2. 1 Hogar Digital

Fuente: [López, 2007]

2.2. Gestión Domótica

Hay una gran variedad de dispositivos electrónicos que se integran a un sistema domótico que simplemente se lograría desarrollar un gran sistema con todas las comodidades y necesidades, pero teóricamente la domótica se puede categorizar de la siguiente forma:

2.2.1. Gestión Energética

Es la acción de administrar la energía que se utiliza en un inmueble, dicha administración se apoya en tres pilares fundamentales que son: el ahorro energético, la eficiencia energética y la generación de energía. La domótica juega un papel muy importante en este punto ya que cuenta con la inteligencia suficiente para realizar dichas acciones. La gestión energética es considerada una de las aplicaciones más trascendentales de la domótica.

Es importante aclarar que cuando hablamos de energía hacemos referencia a todos los tipos de energía. No solamente a la energía eléctrica. De acuerdo a lo explicado, la gestión energética se apoya en los siguientes pilares:

➤ **Ahorro Energético**

En esta sub-categoría entran todas las aplicaciones que persiguen reducir el consumo evitando el derroche de la energía, por ejemplo, aviso de puertas o ventanas abiertas cuando está encendida la calefacción.

➤ **Eficiencia Energética**

En esta sub-categoría entran las aplicaciones que no reducen el consumo de energía, pero logran que el mismo sea aprovechado al máximo. Por ejemplo con un sistema corrector de factor de potencia, evitamos la aparición de potencias reactivas que sobrecargan las líneas no siendo utilizadas para ningún fin.

➤ **Generación Energética**

En esta sub-categoría entran las aplicaciones responsables de controlar todo tipo de sistemas responsables de generación de cualquier tipo de energía.

2.2.2. Confort

Cuando su vivienda se adecua por si misma a sus necesidades mejora su calidad de vida. La domótica juega un papel muy importante en este punto ya que posee el control de todo el inmueble, dicho control es usado fundamentalmente para, en primer lugar, programar escenas para que el inmueble se adapte a las necesidades de cada persona. El confort desde el punto de vista de la domótica es básicamente el control de dispositivos el cual se divide, por citar algunos ejemplos, en: control de iluminación, control de clima, control de aberturas, control de riego, control multimedia, generación de escenas, etc.

2.2.3. Seguridad

Consiste en una red de seguridad encargada de proteger las personas y los bienes, esta aplicación se apoya en dos pilares que son la prevención y la detección para la acción. Como la domótica tiene pleno conocimiento del estado de las puertas, ventanas y sensores dentro y fuera de la vivienda, puede de una manera muy sencilla y eficiente, tomar control de esa información y poder, mediante la programación instalada, proteger todo el hogar. Esta es también una de las aplicaciones más importantes de la domótica, ya que la figura de la persona que lleva adelante la responsabilidad global del hogar durante todo el día es cada vez menos frecuente en las familias actuales; en su remplazo están los elementos que permiten saber lo que está pasando, sea de manera local o a distancia.

2.2.4. Comunicación

Esta aplicación puede que aparezca poco importante, pero en realidad es la encargada o va de la mano con el resto de las aplicaciones ya que sin ella sería imposible conocer el estado y controlar los sistemas a distancia.

Lo que se consigue aquí es la posibilidad de conectarse con el hogar y dentro del mismo con la mayor cantidad de medios de comunicación disponibles, pudiendo de esta manera controlar la vivienda a distancia (tele gestión) y aumentar la interactividad entre las personas y el hogar [CIEC, 2012].

2.2. Arquitectura

La arquitectura en un sistema domótico, se caracteriza por la ubicación de los diferentes elementos de control. Básicamente existen tres arquitecturas: La Arquitectura Centralizada, La Arquitectura Descentralizada y La Arquitectura Distribuida.

2.3.1. Arquitectura Centralizada

Es aquella en la que los elementos a controlar y supervisar (sensores, luces, válvulas, etc.) se centraliza todo el cableado en un sistema de control (autómata, PC, etc.). Toda la información recolectada por los sensores se los envía a un controlador para que este pueda determinar una acción o una decisión y luego se las comunica a los diferentes actuadores en la figura 2.2 se muestra el comportamiento. El sistema de control es el núcleo o el corazón del edificio, si ocurre una falla todo dejaría de funcionar.

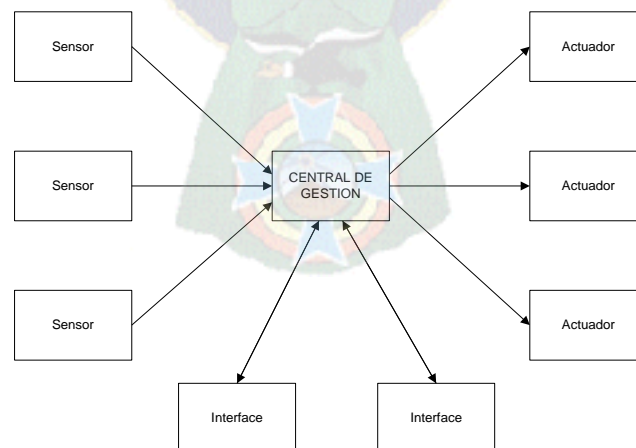


Figura 2. 2 Arquitectura Centralizada

Fuente: [Elaboración Propia]

Al tratarse de una arquitectura centralizada también es importante explicar cuáles son sus ventajas y desventajas.

Las ventajas de los sistemas centralizados son:

- Los elementos sensores y actuadores son del tipo universal
- Coste reducido o moderado
- Fácil uso y formación
- Instalación sencilla

Las desventajas o inconvenientes son:

- Cableado significativo
- Sistema dependiente del funcionamiento óptimo de la central
- Modularidad difícil
- Reducida y ampliable
- Capacidad del sistema (canales o puntos).

2.3.2. Arquitectura Descentralizada

Esta es una arquitectura opuesta al anterior. En la arquitectura descentralizada, todos los elementos del sistema trabajan independientemente de forma que son autónomas en sus decisiones. Este sistema debe disponer de un bus compartido, que permita la comunicación de todos los elementos de control como se ve en la figura 2.3 arquitectura descentralizada.

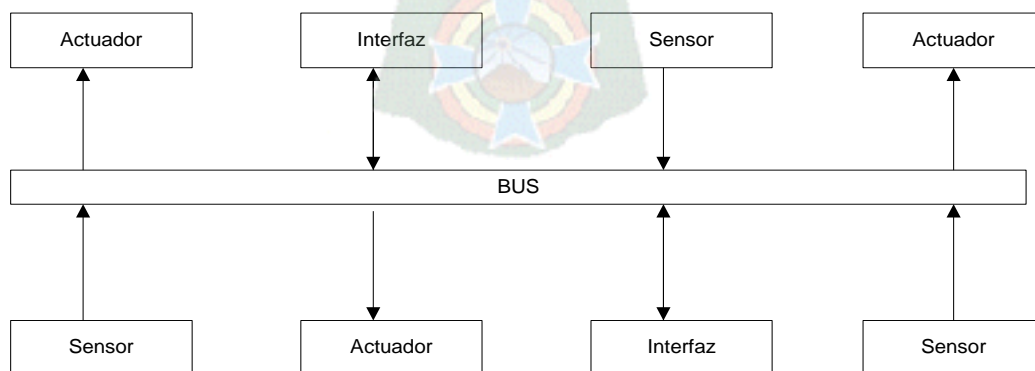


Figura 2. 3 Arquitectura Descentralizada

Fuente: [Elaboración Propia]

Las ventajas de los sistemas descentralizados son las siguientes:

- Seguridad de funcionamiento.
- Posibilidad de rediseño de la red.
- Reducido del cableado.
- Fiabilidad de productos.
- Fácil y ampliable.

Los inconvenientes o desventajas son:

- Elementos de sensores no universales y limitados a la oferta.
- Coste elevado de la solución
- Más próximos a “Edificios Inteligentes” que a “Viviendas Inteligentes”.
- Complejidad de programación.

2.3.3. Arquitectura Distribuida

Esta arquitectura combina las dos anteriores con el objetivo de mejorar las dos arquitecturas en la figura 2.4 se muestra el comportamiento, para ello el elemento de control se sitúa próximo a los elementos de control. Ahora no existe un único elemento de control que gobierna todo, sino que existe varios elementos entre lo que se reparte la tarea de control, estos nuevos elementos de control se denominan nodos.

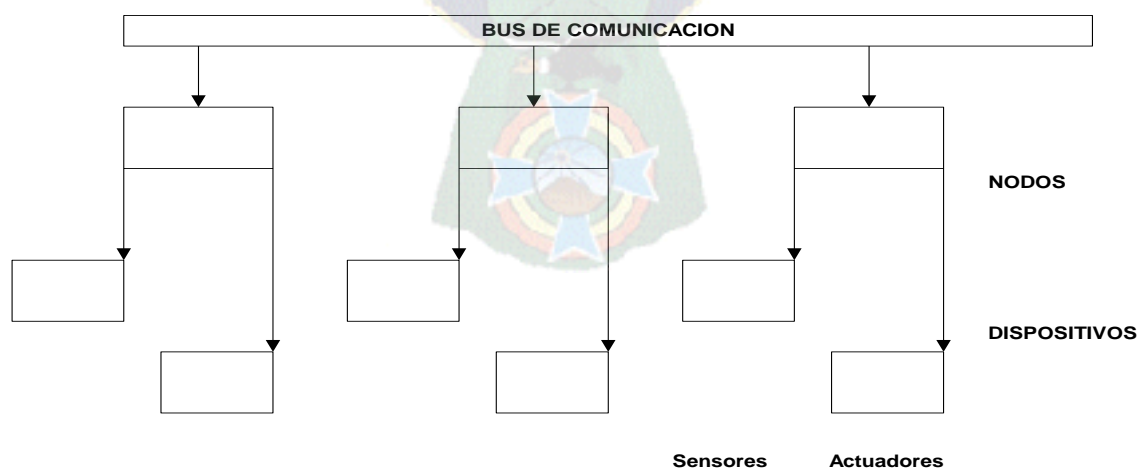


Figura 2. 4 Arquitectura Distribuida

Fuente: [Elaboración Propia]

Es necesario, al igual que en el caso de los sistemas descentralizados, un protocolo de comunicaciones para que todos los módulos produzcan una acción coordinada.

Las ventajas de los sistemas distribuidos son:

- Seguridad de funcionamiento.
- Posibilidad de rediseño de la red.
- Fácil y ampliable.
- Sensores y actuadores de tipo universal (económicos y de gran oferta).
- Coste moderado
- Cableado moderado.

Como el principal inconveniente se destaca el hecho de que requieren programación o configuración [CIEC, 2012].

2.4. Topología de Redes

Una característica importante es la topología que se refiere a la forma en que una red se constituye de forma física y lógica. La conexión de los componentes de una instalación domótica (sensores, actuadores, unidades de control etc.), de acuerdo a esta característica se diseñan físicamente las necesidades de una vivienda.

La Topología de una red es la representación de la relación entre los dispositivos (habitualmente denominados nodos), y todos los enlaces que los unen entre sí [Flores, 2012].

2.4.1. Topología en Estrella

En la figura 2.5 muestra la configuración donde cada dispositivo de entrada (sensores) tiene solamente un enlace “punto a multipunto” conectado con el controlador central o central de gestión, también los de salida (actuadores) van conectados a la central de gestión desde donde se efectúa el tratamiento de los datos del conjunto.

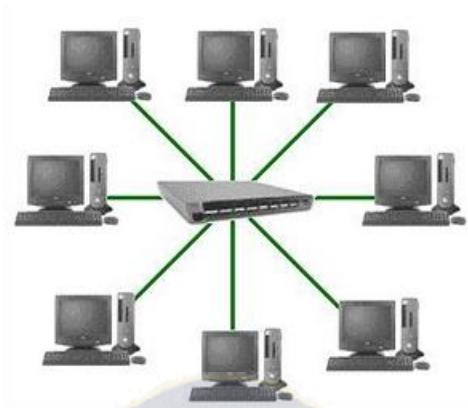


Figura 2. 5 Topología en Estrella

Fuente: [Elaboración Propia]

2.4.2. Topología en Bus

Esta topología se basa en tener un bus (cable) como se ve en la figura 2.6 la topología en bus, que funciona como una red troncal que conecta todos los dispositivos en la red, esta topología es multipunto. Los nodos se comunican al bus mediante cables de conexión y derivadores. Un cable de conexión permite la conectividad al dispositivo con el cable principal. Una de sus principales ventajas es la de poder suprimir un nodo de tal forma que no afecta a todo el sistema.



Figura 2. 6 Topología en Bus

Fuente: [Elaboración Propia]

2.4.3. Topología en Anillo

En la figura 2.7 muestra esta topología donde cada dispositivo tiene una línea de conexión “punto a punto” con los dispositivos que están a sus lados. La señal que genera pasa a lo largo del anillo en una dirección, o de dispositivo a dispositivo, hasta que alcanza su destino, al tener este tipo de configuración cerrada es compleja la inserción de un nuevo elemento ya que tendría que detener el funcionamiento de todo el sistema pero en relación de cableado es de menor consumo al resto de las topologías.



Figura 2. 7 Topología en Anillo

Fuente: [Elaboración Propia]

2.5. Medios de Transmisión y comunicación

El medio de transmisión es el soporte físico que utilizan los diferentes elementos para intercambiar información unos con otros, en la figura 2.8 se muestra los medios de transmisión (par trenzado, línea de potencia o red eléctrica, radio, infrarrojos, etc.). A continuación un detalle de los medios de transmisión más adecuados para realizar comunicación [Martínez, 2012].

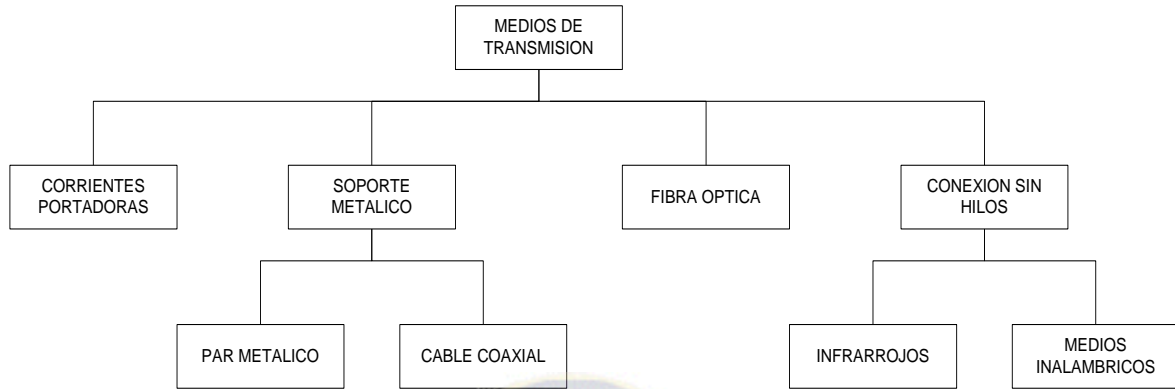


Figura 2. 8 Tipos de medios de Transmisión

Fuente: [Elaboración Propia]

2.5.1. Corrientes Portadoras

Utilizan líneas de distribución ya existentes en la vivienda para la transmisión de datos. Las más utilizadas son las líneas de distribución de energía eléctrica, aunque también se está comenzando a utilizar la línea telefónica tradicional. Si bien no ese medio más adecuado para la transmisión de datos, si es una alternativa a tener en cuenta para las comunicaciones domesticas dado el bajo coste que implica su uso, ya que se trata de una instalación existente. Las especiales características de este medio lo hacen idóneo para su uso en las instalaciones domesticas ya existentes. Sus principales ventajas son el bajo costo de la instalación. Y sus inconvenientes son la poca fiabilidad en la transmisión de los datos y la baja velocidad de transmisión.

2.5.2. Soporte Metálico

Son cables metálicos de cobre como soporte de transmisión de las señales eléctricas que procesa. En general se pueden distinguir dos tipos de cables metálicos:

2.5.2.1. Par Metálico

Los cables formados por varios conductores de cobre pueden dar soporte a un amplio rango de aplicaciones.

Este tipo de cables pueden transportar: datos, voz y alimentación. Los denominados cables de pares están formados por cualquier combinación de los tipos de conductores que a continuación se detallan: Tipos de conductores de par trenzado.

Cables formado 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 está 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 de alta fidelidad o datos).

Par trenzado, está formado por dos hilos recubiertos por un trenzado en forma de malla (por ejemplo los utilizados para interconexión de ordenadores).

2.5.2.2. Cable Coaxial

Un par coaxial es un circuito físico asimétrico, constituido por un conductor en forma de tubo, se mantiene el carácter coaxial de ambos mediante un dieléctrico apropiado. Este tipo de cables permite el transporte de las señales de video y señales de datos de alta velocidad. Dentro del ámbito de la vivienda, el cable coaxial puede ser utilizado como soporte de transmisión para: Utilidades del cable coaxial. 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, fidelidad o datos.

2.5.3. 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. Sus ventajas son: Fiabilidad en la transferencia de datos, inmunidad frente a interferencias electromagnéticas, alta seguridad en la transmisión de datos, distancia entre los puntos de la instalación ilimitada, y transferencia de gran cantidad de datos. Su principal inconveniente es el elevado coste de los cables y las conexiones.

2.5.4. Conexión sin Hilos

2.5.4.1. Infrarrojos

La comunicación se realiza entre un diodo emisor que emite una luz en la banda de IR, sobre la que se supone 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. 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.).

2.5.4.2. Radiofrecuencia

La introducción de las radiofrecuencias como soporte de transmisión en la vivienda, ha venido precedida por la proliferación de los teléfonos inalámbricos y sencillos teletandos. 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 electro-magnéticas en el medio de transmisión.

2.6. Protocolos de Comunicación

Los protocolos son estándares que permiten la comunicación, sobre todo en domótica es utilizada frecuentemente para la comunicación de nodos para ello detallaremos los dos tipos de protocolos que existen.

2.6.1. Tipo de Protocolos

2.6.1.1. Propietario o cerrado

Son protocolos específicos de una marca en particular y que solo son usados por dicha marca. Pueden ser variantes de Protocolos Estándar, son protocolos cerrados de manera que solo el fabricante puede realizar mejoras y fabricar dispositivos que “hablen” el mismo idioma. Esto protege los derechos del fabricante, pero limita la aparición de continuas evoluciones en los sistemas domóticos, con lo que, a medida que los sistemas con protocolo estándar se han desarrollado, van ganando cuota de mercado a los sistemas de protocolo propietario. Otro problema que tienen: es la vida útil del sistema domótico, en un sistema propietario que depende en gran medida de la vida de la empresa y de la política que siga, si la empresa desaparece, el sistema desaparece y las instalaciones se quedan sin soporte ni recambios.

2.6.1.2. Estándar o abierto

Son protocolos definidos entre varias compañías con el fin de unificar criterios, son abiertos (open systems) es decir que no existen patentes sobre el protocolo de manera que cualquier fabricante puede desarrollar aplicaciones y productos que lleven implícito el protocolo de comunicación.

Los protocolos conocidos en estas tecnologías se detallan con sus diferentes características.

Tecnologías y estándares

- **X-10:** EE.UU finales de los 70.
- **EHS (European Home System):** 1992 Unión Europea.
- **EIB (European Installation Bus).**
- **BatiBUS**
- **KONNEX:** EHS+EIB+BatiBUS.
- **BIODOM:** Desarrollado por españoles.

X-10: EE.UU finales de los 70: Para el control remoto de dispositivos eléctricos. Utiliza la línea eléctrica (220V o 110) para transmitir señales de control entre equipos de automatización del hogar en formato digital.

Domótica X-10 es el sistema más extendido y utilizado en los hogares del mundo debido a su sencillez de instalación y su fácil manejo que se adapta a todas las necesidades actuales de control domótica en los hogares. X-10 es el "lenguaje" de comunicación que utilizan los productos compatibles X-10 para hablarse entre ellos y que le permiten controlar las luces y los electrodomésticos de su hogar, aprovechando para ello la instalación eléctrica existente de 220V de su casa, y evitando tener que instalar cables. Este es el principal motivo por el que X-10 se considera un sistema de domótica sin instalación. A nivel físico hace uso de las líneas de baja tensión (ondas portadoras), la transmisión completa de una orden X-10 necesita once ciclos de corriente, que se divide en tres campos:

- Código de Inicio
- Código de Casa (letras A-P)
- Código Numérico (1-16) o bien el Código de Función (encender, apagar, aumentó de intensidad).

En el protocolo X-10 existen tres tipos de dispositivos X-10:

- Los que solo pueden transmitir órdenes.
- Los que solo pueden recibirlas.
- Los que pueden enviar/recibir.

A medida que fueron desarrollándose nuevos estándares se unieron en un solo protocolo llamado Konnex.

KONNEX: EHS+EIB+BatiBUS: Esta tecnología contempla varios sistemas de transmisión (cable, radio, ondas portadoras, fibra óptica).

EHS (European Home System): Está basada en una tecnología de niveles OSI (Open Standard Interconnection), y se especifican los niveles: físico, de enlace de datos, de red y de aplicación.

EIB (European Installation Bus): Es un sistema de domótica basado en bus de datos. Utiliza su propio cableado.

BatiBUS: Se basa en la tecnología de par trenzado pudiendo transmitir hasta 4800 bps (comprobar). El sistema es centralizado, pudiendo controlar cada central hasta 500 puntos de control.

BIODOM: Desarrollado por españoles. Basado en la standard EHS. Se trata de un sistema propio, desarrollado completamente por Bioingeniería Aragonesa, lo que lo convierte en un sistema fácilmente adaptable a nuevas necesidades.

LONWORKS: LonWorks es una tecnología de control domótico propietaria de la compañía americana Echelon Corp.

Al igual que KNX, LonWorks puede utilizar una gran variedad de medios de transmisión: aire, par trenzado, coaxial, fibra o red eléctrica. Requiere la instalación de “nodos” a lo largo de la red que gestionan los distintos sensores y actuadores.

LonWorks es una tecnología muy robusta y fiable por lo que está especialmente indicada para la automatización industrial, ámbito del que procede. Esta más implementada en Estados Unidos que en Europa [Botero, 2003].

2.7. Componentes de un sistema domótico

Una instalación domótica está compuesta por una serie de elementos, los cuales detectan un cambio de estado en una variable física; estos dispositivos llamados sensores transmiten la información al sistema de control mediante interfaces y acondicionadores de señal para adaptar las señales entre los distintos componentes del hardware, utilizando una estructura de comunicaciones para que interactúen con otros dispositivos llamados actuadores, encargados de ejecutar, en consecuencia, las acciones de control en función de unas normas establecidas por el usuario[Lledó, 2012].

2.7.1. Características

Para realizar las mediciones de magnitudes mecánicas, térmicas, eléctricas, físicas, químicas, entre otras, se emplean dispositivos comúnmente llamados sensores y transductores.

El sensor percibe los cambios de la magnitud en cuestión, como temperatura, posición o concentración química, mientras que el transductor convierte estas mediciones en señales generalmente eléctricas para suministrar la información a instrumentos de lectura y registro o para el control de las magnitudes medidas. Estos dispositivos pueden estar ubicados en posiciones alejadas del observador, así como en entornos inadecuados, imperceptibles o impracticables para los seres humanos.

Los sensores son considerados elementos transductores de entrada en un sistema domótico porque permiten obtener información de los parámetros que se desea monitorear y/o controlar en un recinto, llevando a cabo la conversión de magnitudes para transmitirla a la unidad encargada del procesamiento y control del estado de las variables a gestionar.

En la gran mayoría de los casos se encuentran protegidos por un encapsulado el cual logra reducir o evitar las interferencias externas distintas de la magnitud en medición, permitiendo un correcto y confiable funcionamiento.

Dentro de las características que posee un sensor se pueden destacar:

- Conversión de una variable física en otra diferente más fácil de evaluar y procesar.
- Aunque generalmente proporcionan señales eléctricas, en algunos casos pueden generar otros tipos de señales.
- Pueden funcionar con contacto físico (sensores de toque) o sin contacto físico (sensores ópticos).

Para evaluar y valorar la calidad de un sensor se debe tener en cuenta una serie de conceptos y definiciones que los caracterizan, dentro de los cuales se encuentran:

- **Amplitud:** Es la diferencia entre los límites de la medición.
- **Calibración:** Es un patrón de la variable medida que se aplica mientras se observa la señal de salida.
- **Error:** Es la diferencia entre el valor medido y el valor real.
- **Error de linealidad:** Es la máxima desviación de la función de transferencia del sensor respecto a su recta de ajuste.
- **Exactitud:** Es la concordancia entre el valor medido y el valor real.
- **Factor de escala:** Es la relación entre la salida y la variable medida.
- **Fiabilidad:** Es la probabilidad de no generar error.
- **Histéresis:** Es una trayectoria o recorrido diferente de la medida cuando aumenta o disminuye.
- **Offset:** Es el valor de la salida del sensor cuando la magnitud medida es cero.
- **Precisión:** Es la dispersión de los valores de salida. Se determina como el cociente entre el máximo error de la señal de salida respecto del máximo valor de salida y se expresa en porcentaje.
- **Rango dinámico:** Es la diferencia entre los valores máximo y mínimo que pueden ser medidos por el sensor.
- **Rango de error:** Es una banda de desviaciones permisibles de la salida.

- **Rango de temperatura de servicio:** Es el rango de temperaturas de trabajo o funcionamiento en el cual la señal de salida permanece dentro del error especificado.
- **Resolución:** Es el menor cambio detectable en la magnitud medida que puede causar un cambio en la magnitud de salida.
- **Ruido:** Es una perturbación aleatoria no deseada que modifica el valor medido.
- **Sensibilidad:** Es la relación entre la variación de la salida y el cambio en la variable medida.

2.7.2. Tipos de Sensores

En la actualidad existe un gran número de sensores de distintos tipos y con diversas funcionalidades, los cuales se pueden clasificar de acuerdo con determinados criterios.

2.7.2.1. Según el tipo Alimentación

Activos y Pasivos.

Los sensores activos necesitan ser alimentados eléctricamente y ajustados a los niveles apropiados de voltaje, corriente, etc. Son los más comunes en las instalaciones domóticas.

Las sondas de temperatura son ejemplos de sensores activos; su resistencia cambia con la temperatura, haciendo variar la corriente que circula por ella y que es suministrada por un generador correspondiente.

Los sensores pasivos no necesitan ser alimentados eléctricamente, por lo tanto no suelen ser aplicados comúnmente en la industria o en la domótica. Algunos ejemplos de sensores pasivos son los termómetros de mercurio y los indicadores de presión.

2.7.2.2. Según el tipo de tipo de señal implicada

Continuos y discretos

Haciendo referencia a su nombre, los sensores continuos son aquellos que proporcionan señales continuas y los discretos cuando las señales que suministran son discretos.

Un sensor continuo tiene como salida una magnitud cuyo valor medido varía de forma continua en el tiempo, pudiendo presentar infinitos valores dentro de su rango. Estas magnitudes en la salida del sensor son llamadas señales analógicas. Algunos ejemplos de este tipo de sensores son los de iluminación, humedad, presión, temperatura, magnitud y dirección del viento, entre otros.

Los sensores discretos solo disponen de un número finito de posibles salidas que corresponden a estados posibles limitados de la variable a medir. La magnitud en la salida es llamada señal discreta, caracterizada por poseer un número finito de valores dentro de su rango, pero poseen mayor interés las que se aplican comúnmente en el campo de la domótica, resultando ser aquellas que presentan únicamente dos estados: encendido y apagado.

A estas se les denomina señales binarias, en la mayoría de los casos los sensores discretos son denominados detectores, pues su funcionalidad es la detección de dos estados, por ejemplo circuito abierto o cerrado y detección de la presencia o ausencia de alguna condición física como iluminación, humo, agua, gas, incendio, apertura de puertas o ventanas, rotura de cristales, proximidad (mediante barreras ópticas), etc.

2.7.2.3. Según el ámbito de aplicación

Gestión climática, gestión contra incendio, gestión contra intrusión y/o robo, control de presencia e iluminación, entre otros. La tabla 2.1 realiza una descripción de los criterios más comunes de clasificación correspondiente al ámbito de utilización, permitiendo así una gestión y un control directo de diferentes factores que influyen en las instalaciones domóticas e inmoticas.

Tabla 2. 1 Descripción de tipo de sensores

Fuente: [Romero, 2005]

Ámbito de Aplicación	Tipo de Sensor
Gestión climática	Sensores de temperatura (resistivos, semiconductores, termopares, etc.), termostatos, sondas de temperatura para inmersión, para conductos, para tuberías, sensores de humedad, sensores de presión, etc.
Gestión contra Incendio	Sensores iónicos, termo velocimétricos, sensores ópticos, infrarrojos, de barrera óptica, sensores ópticos de humo, de dilatación etc.
Gestión contra intrusión y/o robo	Sensores de presencia por infrarrojos, por microondas o por ultrasonidos, sensores de aperturas de puertas o ventanas, sensores de rotura de cristales, sensores microfónicos, sensores de alfombra pisada, etc.
Control de Presencia	Lector de teclado, lector de tarjetas, identificadores corporales (biométricos).
Control de iluminación	Sensor de luminosidad.
Otros sistemas	Sensores de lluvia, de viento, de CO2, de gas, de inundación, de consumo eléctrico, de consumo de agua, de nivel de depósitos, etc

2.7.3. Sensor de Luminosidad

En la automatización de edificios se necesitan sensores de luminosidad, aptos para uso continuo y protegidos contra sobretensiones y señales transitorias. Otros aspectos son la elección entre la alimentación a CC o CA, así como la salida con señal estandarizada. Los valores de medición de la luminosidad se pueden utilizar en una variedad de aplicaciones, como control dinámico de dispositivos en la automatización de edificios, o por ejemplo la utilización como crepuscular.

Para la medición de la luminosidad se utiliza un fotodiodo de precisión, con especificaciones industriales. Para el tratamiento de la señal de medición se utilizan modernas tecnologías de sensores. La alta sensibilidad en un amplio rango de luminancia también en situaciones con mucha luz. Mediante la calibración se obtiene una buena precisión, en un amplio rango de luminancia, apto para las típicas tareas de control.

2.7.3.1. Sensor de luz LDR (Light Dependent Resistor)

Un LDR es un resistor que varía su valor en función a la resistencia eléctrica dependiendo de la cantidad de luz que incide sobre él. Es conocido también como foto resistor o fotorresistencia. El valor que emite esta resistencia eléctrica conocida como LDR es bajo cuando hay luz incidiendo en el (en algunos casos puede descender a tan bajo como 50 ohms) y muy alto cuando está a oscuras (puede ser varios megaohms).

Los LDR básicamente fueron fabricados en base a un cristal semiconductor fotosensible como el sulfuro de cadmio (Cds). Estas celdas son sensibles a un rango amplio de frecuencias lumínicas, desde la luz infrarroja, pasando por la luz visible, y hasta la ultravioleta, la variación de valor resistivo de un LDR tiene cierto retardo, que es diferente si se pasa de oscuro a iluminado o de iluminado a oscuro, por esta razón un LDR no se puede utilizar en algunas aplicaciones, en especial en aquellas en la que la señal luminosa varía con rapidez, el tiempo de respuesta típico de un LDR está en el orden de una décima de segundo.

La lentitud relativa del cambio es una ventaja en algunos casos, porque así se filtran variaciones rápidas de iluminación que podrían hacer inestable un sensor (por ejemplo cuando está iluminado por un tubo fluorescente alimentado por corriente alterna), En otras aplicaciones (como la detección de luminosidad para saber si es de día o es de noche) la lentitud de la detección no es importante.

2.7.4. Sensor de Temperatura

Es un sensor destinado a regular la temperatura interna de una vivienda por medio de calefacción o refrigeración. Podemos mencionar que un termostato es un interruptor que provoca la conexión o desconexión cuando la temperatura alcanza un cierto valor, los sensores de temperatura están compuestos de un termistor. El termistor es un sensor resistivo de temperatura y su funcionamiento se basa en variar resistencia que presenta un semiconductor respecto a la temperatura. En el mercado podemos encontrar dos clases:

- Termistor PTC (Positive Temperature Coefficient): El valor de la resistencia va aumentando a medida que se incrementa la temperatura.
- Termistor NTC (Negative Temperature Coefficient): El valor de la resistencia va disminuyendo a medida que se incrementa la temperatura [isa, 2010].

2.7.4.1. Sensor de temperatura LM35

Es un integrado de precisión, cuya tensión de salida es linealmente proporcional a temperatura en °C (grados centígrados). El Sensor LM35 por lo tanto tiene una ventaja sobre los sensores de temperatura lineal calibrada en grados kelvin: que el usuario no está obligado a restar una gran tensión constante para obtener grados centígrados. Este dispositivo no requiere de ningún tipo de calibración externa o ajuste para poder proporcionar una precisión típica de ± 1.4 °C a temperatura ambiente y ± 3.4 °C a lo largo de su rango de temperatura (de -55 a 150 °C). El dispositivo se ajusta y calibra durante el proceso de producción.

La baja impedancia de salida, la salida lineal y la precisa calibración inherente, permiten la creación de circuitos de lectura o control especialmente sencillos. El LM35 puede funcionar con alimentación simple o alimentación doble (+ y -) Requiere solo 60 μA para alimentarse, y bajo factor de auto-calentamiento, menos de 0,1 $^{\circ}\text{C}$ en aire estático. También este dispositivo está preparado para trabajar en una gama de temperaturas que abarca desde los - 55 $^{\circ}\text{C}$ bajo cero a 150 $^{\circ}\text{C}$ mientras que el LM35C está preparado para trabajar entre -40 $^{\circ}\text{C}$ y 110 $^{\circ}\text{C}$ (con mayor precisión).

Características

- Calibrado directamente en grados Celsius (Centígrados)
- Factor de escala lineal de +10 mV / $^{\circ}\text{C}$
- 0,5 $^{\circ}\text{C}$ de precisión a +25 $^{\circ}\text{C}$
- Rango de trabajo: -55 $^{\circ}\text{C}$ a +150 $^{\circ}\text{C}$
- Apropiado para aplicaciones remotas
- Funciona con alimentaciones entre 4V y 30V
- Menos de 60 μA de consumo
- Bajo auto-calentamiento (0,08 $^{\circ}\text{C}$ en aire estático)
- Baja impedancia de salida, 0,1W para cargas de 1mA

Básicamente este dispositivo sensor funciona con una precisión calibrada de 1 $^{\circ}\text{C}$ y un rango que abarca desde -55 $^{\circ}$ a +150 $^{\circ}\text{C}$. La salida es lineal y equivalente a 10 V/ $^{\circ}\text{C}$ por lo tanto.

- +1500mV = 150 $^{\circ}\text{C}$
- +250mV = 25 $^{\circ}\text{C}$
- -550mV = -55 $^{\circ}\text{C}$

Disponible en distintos encapsulados (TO-92, TO-46, SO-8, TO-220. El más común es el TO-92 que es igual a un típico transistor de tres patas dos de ellas que son de alimentación de energía y la del centro que el la pata común esta nos entrega un valor proporcional a la temperatura [isa, 2010].

2.8. Placas Electrónicas

2.8.1. Arduino

Arduino es una plataforma amplia en electrónica de código abierto para la creación de prototipos basada en software y hardware flexibles y fáciles de usar. Se creó para artistas, diseñadores, aficionados y cualquiera interesado en crear entornos u objetos interactivos. Arduino puede tomar información del entorno a través de sus pines de entrada de toda una gama de sensores y puede afectar aquello que le rodea controlando luces, motores y otros actuadores.

El microcontrolador en la placa Arduino se programa mediante el lenguaje de programación Arduino(basado en Wiring) y el entorno de desarrollo Arduino (basado en Processing). Los proyectos hechos con Arduino pueden ejecutarse sin necesidad de conectar a un ordenador, si bien tienen la posibilidad de hacerlo y comunicar con diferentes tipos de software (p.ej. Flash, Processing, MaxMSP).

2.8.2. Características Arduino

Se trata de unas placas open hardware por lo que su diseño es de libre distribución y utilización, (En la Figura 2.9 se observa el aspecto de la placa) y la descripción correspondiente.



Figura 2. 9 Arduino Uno REV 3

Fuente: [Arduino, 2010]

El programa se implementa haciendo uso del entorno de programación propio de Arduino y se transferirá empleando un cable USB. Si bien en el caso de la placa USB no es preciso utilizar una fuente de alimentación externa, ya que el propio cable USB la proporciona, para la realización de algunos de los experimentos prácticos sí que será necesario disponer de una fuente de alimentación externa ya que la alimentación proporcionada por el USB puede no ser suficiente. El voltaje de la fuente puede estar entre 6 y 25 Voltios. También es importante conocer las características físicas de hardware que tiene este dispositivo Arduino UNO [Arduino, 2010].

- 14 pines de entrada/salida digitales, de los cuales 6 pueden ser usados como salidas PWM
- Posee 6 entradas analógicas
- Los pines 0 y 1 pueden funcionar como RX y TX serial.
- Un oscilador de crystal de 16 MHz
- Conector USB
- Un Jack de poder
- Un conector ICSP
- Botón de reset

2.8.3. Shield Ethernet

También es una placa electrónica cuya funcionalidad es la de brindar un servicio en la web para poder gestionar los diferentes dispositivos tales como sensores actuadores. Con el Shield Ethernet para Arduino y sus aplicaciones orientadas al gobierno y la monitorización de señales en modo remoto, haciendo uso de la comunicación de protocolo TCP/IP característica de la red Internet, bajo este concepto se incrementa potencialmente la plataforma Open Hardware Arduino y se crean nuevas expectativas a la integración de Arduino en Internet bajo el nuevo paradigma denominado "El internet de las cosas". La conexión de este Shell es a través de un cable de red de conector RJ45, como siempre con Arduino todos los elementos de la plataforma-hardware, software y documentación son de libre acceso y de código abierto.

2.8.4. Características Shield Ethernet

El Shield Ethernet tiene una conexión estándar RJ-45, con un transformador de línea integrado y Alimentación a través de Ethernet activado. Hay una ranura para tarjetas micro-SD, que se puede utilizar para almacenar archivos para servir a través de la red. Es compatible con el Arduino Uno y Mega (utilizando la librería Ethernet). El lector de tarjetas microSD es accesible a través de la Biblioteca SD. El escudo actual tiene un Power over Ethernet (módulo shield) diseñado para extraer energía de un cable trenzado convencional Ethernet par Categoría 5 (En la Figura 2.10 se observa el aspecto de la placa).

- Ondulación baja de la salida y el ruido (100mVpp).
- Rango de tensión de entrada de 36V a 57V
- Protección contra sobrecargas y cortocircuitos
- 9V de salida
- Alta eficiencia del convertidor DC / DC: 75% a 50% de carga

Arduino se comunica tanto con la W5100 y la tarjeta SD con el bus SPI (a través de la cabecera ICSP). Esto es en los pines digitales 10, 11, 12 y 13 en el Uno y los pines 50, 51, y 52 en el Mega. En ambas tablas, el pin 10 se utiliza para seleccionar el W5100 y el pin 4 para la tarjeta SD. Estos pines no se pueden utilizar para general I / O.

Es importante tener en cuenta que debido a que el W5100 y la tarjeta SD comparten el bus SPI, sólo uno puede estar activo a la vez. Si está utilizando ambos periféricos en su programa, esto debe ser atendido por las bibliotecas correspondientes. Si no se está utilizando uno de los periféricos en su programa, sin embargo, tendrá que anular la selección de forma explícita.

El protector provee un estándar RJ45 conector Ethernet.

El botón de reinicio en que se restablezca el escudo W5100 y la placa Arduino.

El escudo contiene un número de informativos LED:

- **PWR:** indica que la junta y el escudo son powered.
- **ENLACE:** indica la presencia de un enlace de red y parpadea cuando el escudo transmite o recibe datos.
- **FULLD:** indica que la conexión de red es full duplex
- **100 M:** indica la presencia de una conexión de red de 100 Mb / s (en contraposición a 10 Mb / s)
- **RX:** parpadea cuando el escudo recibe datos
- **TX:** parpadea cuando el escudo envía datos
- **COLL:** parpadea cuando se detectan colisiones de red

El puente de soldadura marcada "INT" puede conectarse a permitir que la placa Arduino para recibir una notificación por interrupciones de los acontecimientos desde el W5100, pero esto no es apoyado por la biblioteca Ethernet. [Arduino, 2010].

- Tensión de alimentación de 5V (se alimenta directamente desde la tarjeta Arduino)
- Controlador Ethernet: W5100 con una memoria interne de 16K
- Velocidad de conexión: 10/100Mb [Arduino, 2010].



Figura 2. 10 Shield Ethernet

Fuente: [Arduino, 2010]

2.8.6. Entorno de Desarrollo

Para la programación de la placa Arduino es necesario descargar de la página web de Arduino el entorno de desarrollo (IDE). Aquí se dispone de diferentes versiones tanto para Windows y para MAC, así como las fuentes para compilarlas en LINUX. En la figura 2.11 se muestra el aspecto del entorno de programación. En el caso de disponer de una placa USB es necesario instalar los drivers FTDI. Estos drivers vienen incluidos en el paquete de Arduino. Existen en la web para diferentes sistemas operativos.

2.8.6.1. Estructura Básica de un Programa

La estructura básica de programación de Arduino es bastante simple y divide la ejecución en dos partes: setup y loop. Setup() constituye la preparación del programa y loop() es la ejecución.

The image shows a screenshot of the Arduino IDE window titled "Blink Arduino 1.5.5". The window has a menu bar with "Archivo", "Editar", "Programa", "Herramientas", and "Ayuda". Below the menu bar is a toolbar with icons for file operations and execution. The main text area contains the following code:

```
Blink$

int led = 13;

// the setup routine runs once when you press reset:
void setup() {
  // initialize the digital pin as an output.
  pinMode(led, OUTPUT);
}

// the loop routine runs over and over again forever:
void loop() {
  digitalWrite(led, HIGH); // turn the LED on (HIGH is the volta
  delay(1000); // wait for a second
  digitalWrite(led, LOW); // turn the LED off by making the vol
  delay(1000); // wait for a second
}
```

At the bottom of the window, there is a status bar that says "1" and "Arduino Uno on COM3".

Figura 2. 11 Entorno de Desarrollo

Fuente: [Arduino, 2010]

En la función Setup() se incluye la declaración de variables y se trata de la primera función que se ejecuta en el programa.

Esta función se ejecuta una única vez y es empleada para configurar el pinMode (p. ej. Si un determinado pin digital es de entrada o salida) e inicializar la comunicación serie. La función loop() incluye el código a ser ejecutado continuamente (leyendo las entradas de la placa, salidas, etc.).

2.8.6.2. Funciones

Simplemente es un bloque de código identificado por un nombre que se ejecuta cuando dicho nombre es solicitado por el programa. La declaración de una función incluye el tipo de datos que devuelve la función (e.j. int si lo que devuelve es un valor entero). Después del tipo de datos se especifica el nombre de la función y los parámetros de la misma.

2.8.6.3. Variables

Una variable debe ser declarada y opcionalmente asignada a un determinado valor. En la declaración de la variable se indica el tipo de datos que almacenará (int, float, long). Una variable puede ser declarada en el inicio del programa antes de setup(), localmente a una determinada función e incluso dentro de un bloque como pueda ser un bucle. El sitio en el que la variable es declarada determina el ámbito de la misma.

2.8.6.4. Tipos de Datos

- **Byte.** Almacena un valor numérico de 8 bits. Tienen un rango de 0-255.
- **Int.** Almacena un valor entero de 16 bits con un rango de 32,767 a -32,768.
- **Long.** Valor entero almacenado en 32 bits con un rango de 2,147,483,647 a -2,147,483,648.
- **Float.** Tipo coma flotante almacenado en 32 bits con un rango de 3.4028235E+38 a -3.4028235E+38.
- **Arrays.** Se trata de una colección de valores que pueden ser accedidos con un número de índice (el primer valor del índice es 0). Ejemplos de utilización:

- ✓ Definición y asignación. `int myArray[] = {value0, value1, value2...}`
- ✓ Definición. `int myArray[5]; // declara un array de 6 enteros`
- ✓ Asignación del cuarto componente. `myArray[3] = 10;`
- ✓ Recuperar el cuarto componente y asignarlo a x. `x = myArray[3];`

2.8.6.5. Operadores Aritméticos

Empleando variables, valores constantes o componentes de un array pueden realizarse operaciones aritméticas y se puede utilizar el operador cast para conversión de tipos. Ej. `int a = (int)3.5;` Además pueden hacerse las siguientes asignaciones:

`x ++`. Lo mismo que `x = x + 1`.

`x --`. Lo mismo que `x = x - 1`, or decrements x by -1.

`x += y`. Lo mismo que `x = x + y`, or increments x by +y. `x -= y`. Lo mismo que `x = x - y`.

`x *= y`. Lo mismo que `x = x * y`.

`x /= y`. Lo mismo que `x = x / y`.

Para su utilización en sentencias condicionales u otras funciones Arduino permite utilizar los siguientes operadores de comparación:

`x == y`. x es igual a y.

`x != y`. x no es igual a y.

`x < y`, `x > y`, `x <= y`, `x >= y`.

Y los siguientes operadores lógicos:

Y lógico: `if (x > 0 && x < 5)`. Cierto si las dos expresiones lo son.

O lógico: `if (x > 0 || y > 0)`. Cierto si alguna expresión lo es.

NO lógico: `if (!x > 0)`. Cierto si la expresión es falsa.

HIGH/LOW. Estas constantes definen los niveles de los pines como HIGH o LOW y son empleados cuando se leen o escriben en las entradas o salidas digitales.

HIGH se define como el nivel lógico 1 (ON) o 5 V. **LOW** es el nivel lógico 0, OFF, o 0 V.

INPUT/OUTPUT. Constantes empleadas con la función pinMode() para definir el tipo de un pin digital usado como entrada INPUT o salida OUTPUT.

2.8.6.6. Sentencias Condicionales

El lenguaje de Arduino permite realizar sentencias condicionales if, if... else, for, while. Su utilización es similar a las funciones correspondientes en C. Las entradas y salidas digitales y analógicas.

➤ **Función pinMode(pin, mode)**

Función usada en la función setup() para configurar un pin dado para comportarse como INPUT o OUTPUT. Ej. pinMode(pin, OUTPUT); configura el pin número 'pin' como de salida. Los pines de Arduino funcionan por defecto como entradas, de forma que no necesitan declararse explícitamente como entradas empleando pinMode().

➤ **Función digitalRead(pin)**

Lee el valor desde un pin digital específico. Devuelve un valor HIGH o LOW. El pin puede ser especificado con una variable o una constante (0-13). Ej. v = digitalRead(Pin);

➤ **Función digitalWrite(pin, value)**

Introduce un nivel alto (HIGH) o bajo (LOW) en el pin digital especificado. De nuevo, el pin puede ser especificado con una variable o una constante 0-13. Ej. digitalWrite(pin, HIGH);

➤ **Función analogRead(pin)**

Lee el valor desde el pin analógico especificado con una resolución de 10 bits. Esta función solo funciona en los pines analógicos (0-5). El valor resultante es un entero de 0 a 1023. Los pines analógicos, a diferencia de los digitales no necesitan declararse previamente como INPUT o OUTPUT.

➤ **Función analogWrite(pin, value)**

Escribe un valor pseudo-analógico usando modulación por ancho de pulso (PWM) en un pin de salida marcado como PWM. Esta función está activa para los pines 3, 5, 6, 9, 10, 11. Puede especificarse un valor de 0 – 255. Un valor 0 genera 0 V en el pin especificado y 255 genera 5 V. Para valores de 0 a 255, el pin alterna rápidamente entre 0 V y 5 V, cuanto mayor sea el valor, más a menudo el pin se encuentra en HIGH (5 V).

2.8.6.7. Funciones de Tiempo y Matemáticas Delay(ms)

Realiza una pausa en el programa la cantidad de tiempo en milisegundos especificada en el parámetro (máximo 1000, mínimo 1).

2.8.6.8. Puerto Serie Serial.begin(rate)

Abre un Puerto serie y especifica la velocidad de transmisión. La velocidad típica para comunicación con el ordenador es de 9600 aunque se pueden soportar otras velocidades.

➤ **Serial.println(data).**

Imprime datos al puerto serie seguido por un retorno de línea automático. Este comando tiene la misma forma que Serial.print() pero este último sin el salto de línea al final. Este comando puede emplearse para realizar la depuración de programas. Desde el entorno de programación de Arduino, activando el “Serial Monitor” se puede observar el contenido del puerto serie, y, por lo tanto, los mensajes de depuración. Para observar correctamente el contenido del puerto serie se debe tener en cuenta que el “Serial Monitor” y

el puerto serie han de estar configurados a la misma velocidad (Para configurar la velocidad del puerto serie se hará con el comando `Serial.begin(rate)`).

➤ **Serial.read().**

Lee o captura un byte (un carácter) desde el puerto serie. Devuelve -1 si no hay ningún carácter en el puerto serie.

➤ **Serial.available().**

Devuelve el número de caracteres disponibles para leer desde el puerto serie [Arduino, 2010].

2.9. Metodología

La metodología para el desarrollo de software es importante. La programación Extrema, como proceso de creación de software diferente al convencional, nace de la mano de Kent Beck (guru de la XP y autor de los libros más influyentes sobre el tema). La metodología XP tiene un conjunto importante de reglas y prácticas. En forma genérica, en la figura 2.12 se observa cómo se pueden agrupar en:

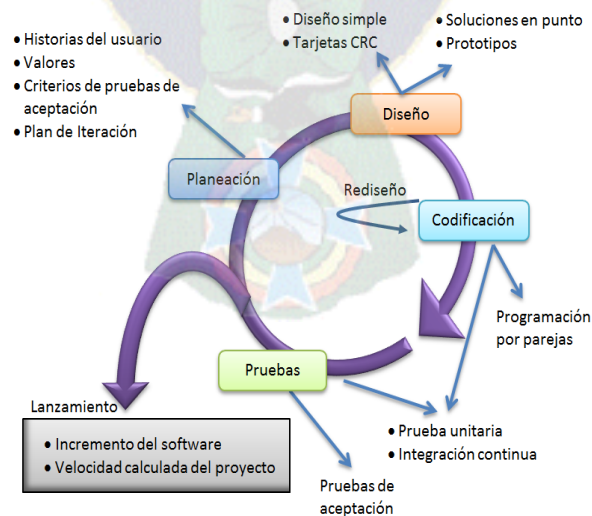


Figura 2. 12 Programación Extrema

Fuente: [Joskowicz, 2008]

2.9.1. Planificación

La metodología XP plantea la planificación como un dialogo continuo entre las partes involucradas en el proyecto, incluyendo al cliente, a los programadores y a los coordinadores o gerentes. El proyecto comienza recopilando “Historias de usuarios”, las que sustituyen a los tradicionales “casos de uso”. Si alguna de ellas tiene “riesgos” que no permiten establecer con certeza la complejidad del desarrollo, se realizan pequeños programas de prueba (“spikes”), para reducir estos riesgos, se organiza una reunión de planificación, con los diversos actores del proyecto (cliente, desarrolladores, gerentes), a los efectos de establecer un plan o cronograma de entregas (“Release Plan”) en los que todos estén de acuerdo. Una vez acordado este cronograma, comienza una fase de iteraciones, en dónde en cada una de ellas se desarrolla, prueba e instala unas pocas “historias de usuarios”.

2.9.1.1. Historias de usuarios

Las “Historias de usuarios” (“User stories”) sustituyen a los documentos de especificación funcional, y a los “casos de uso”. Estas “historias” son escritas por el cliente, en su propio lenguaje, como descripciones cortas de lo que el sistema debe realizar. La diferencia más importante entre estas historias y los tradicionales documentos de especificación funcional se encuentra en el nivel de detalle requerido.

Las historias de usuario deben tener el detalle mínimo como para que los programadores puedan realizar una estimación poco riesgosa del tiempo que llevará su desarrollo. Cuando llegue el momento de la implementación, los desarrolladores dialogarán directamente con el cliente para obtener todos los detalles necesarios. Si la estimación es superior a tres semanas, debe ser dividida en dos o más historias. Si es menos de una semana, se debe combinar con otra historia.

2.9.1.2. Plan de entregas (“Release Plan”)

El cronograma de entregas establece qué historias de usuario serán agrupadas para conformar una entrega, y el orden de las mismas. Este cronograma será el resultado de una reunión entre todos los actores del proyecto (cliente, desarrolladores, gerentes, etc.).

El cronograma de entregas se realiza en base a las estimaciones de tiempos de desarrollo realizadas por los desarrolladores. Luego de algunas iteraciones es recomendable realizar nuevamente una reunión con los actores del proyecto, para evaluar nuevamente el plan de entregas y ajustarlo si es necesario.

2.9.1.3. Plan de iteraciones (“Iteration Plan”)

Las historias de usuarios seleccionadas para cada entrega son desarrolladas y probadas en un ciclo de iteración, de acuerdo al orden preestablecido. Al comienzo de cada ciclo, se realiza una reunión de planificación de la iteración, cada historia de usuario se traduce en tareas específicas de programación, para cada historia de usuario se establecen las pruebas de aceptación. Las pruebas de aceptación que hayan fallado en el ciclo anterior son analizadas para evaluar su corrección, así como para prever que no vuelvan a ocurrir. Reuniones diarias de seguimiento (“Stand-up meeting”).

2.9.2. Diseño

La metodología XP hace especial énfasis en los diseños simples y claros. Los conceptos más importantes de diseño en esta metodología son los siguientes:

2.9.1.1. Simplicidad

Un diseño simple se implementa más rápidamente que uno complejo. Por ello XP propone implementar el diseño más simple posible que funcione. Se sugiere nunca adelantar la implementación de funcionalidades que no correspondan a la iteración en la que se esté trabajando. **Soluciones “skipe”** Cuando aparecen problemas técnicos, o cuando es difícil de estimar el tiempo para implementar una historia de usuario, pueden utilizarse pequeños programas de prueba (llamados “spike”), para explorar diferentes soluciones. Estos programas son únicamente para probar o evaluar una solución, y suelen ser desechados luego de su evaluación.

2.9.1.2. Recodificación

La recodificación (“refactoring”) consiste en escribir nuevamente parte del código de un programa, sin cambiar su funcionalidad, a los efectos de hacerlo más simple, conciso y/o entendible. Muchas veces, al terminar de escribir un código de programa, pensamos que, si lo comenzáramos de nuevo, lo hubiéramos hecho en forma diferente, más clara y eficientemente. Los resultados de ésta práctica tienen sus frutos en las siguientes iteraciones, cuando sea necesario ampliar o cambiar la funcionalidad.

2.9.3. Desarrollo

2.9.3.1. Disponibilidad del cliente

Uno de los requerimientos de XP es tener al cliente disponible durante todo el proyecto. No solamente como apoyo a los desarrolladores, sino formando parte del grupo. El involucramiento del cliente es fundamental para que pueda desarrollarse un proyecto con la metodología XP.

2.9.3.2. Uso de estándares

Si bien esto no es una idea nueva, XP promueve la programación basada en estándares, de manera que sea fácilmente entendible por todo el equipo, y que facilite la recodificación. Programación dirigida por las pruebas (“Test-driven programming”). La metodología XP propone un modelo inverso, en el que, lo primero que se escribe son los test que el sistema debe pasar. Luego, el desarrollo debe ser el mínimo necesario para pasar las pruebas previamente definidas.

2.9.3.3. Integraciones permanentes

Todos los desarrolladores necesitan trabajar siempre con la “última versión”. Realizar cambios o mejoras sobre versiones antiguas causan graves problemas, y retrasan al proyecto. Es por eso que XP promueve publicar lo antes posible las nuevas versiones, aunque no sean las últimas, siempre que estén libres de errores.

2.9.3.4. Propiedad colectiva del código

En un proyecto XP, todo el equipo puede contribuir con nuevas ideas que apliquen a cualquier parte del proyecto. Asimismo, cualquier pareja de programadores puede cambiar el código que sea necesario para corregir problemas, agregar funciones o recodificar.

2.9.3.5. Ritmo sostenido

La metodología XP indica que debe llevarse un ritmo sostenido de trabajo. Anteriormente, ésta práctica se denominaba “Semana de 40 horas”. Sin embargo, lo importante no es si se trabajan, 35, 40 o 42 horas por semana. El concepto que se desea establecer con esta práctica es el de planificar el trabajo de manera de mantener un ritmo constante y razonable, sin sobrecargar al equipo.

2.9.4. Pruebas

2.9.4.1. Pruebas Unitarias

Las pruebas unitarias son una de las piedras angulares de XP. Todos los módulos deben de pasar las pruebas unitarias antes de ser liberados o publicados. Por otra parte, como se mencionó anteriormente, las pruebas deben ser definidas antes de realizar el código (“Test-driven programming”). En este sentido, el sistema y el conjunto de pruebas debe ser guardado junto con el código, para que pueda ser utilizado por otros desarrolladores, en caso de tener que corregir, cambiar o recodificar parte del mismo.

2.9.4.2. Detección y corrección de errores

Cuando se encuentra un error (“bug”), éste debe ser corregido inmediatamente, y se deben tener precauciones para que errores similares no vuelvan a ocurrir. Asimismo, se generan nuevas pruebas para verificar que el error haya sido resuelto.

2.9.4.3. Pruebas de Aceptación

Las pruebas de aceptación son creadas en base a las historias de usuarios, en cada ciclo de la iteración del desarrollo.

El cliente debe especificar uno o diversos escenarios para comprobar que una historia de usuario ha sido correctamente implementada. Las pruebas de aceptación son consideradas como “pruebas de caja negra” (“Black box system tests”). Una historia de usuario no se puede considerar terminada hasta tanto pase correctamente todas las pruebas de aceptación.

2.10. Lógica Difusa

La lógica difusa es una lógica alternativa a la lógica clásica que pretende introducir un grado de vaguedad en las cosas que evalúa. En el mundo en que vivimos existe mucho conocimiento ambiguo e impreciso por naturaleza. El razonamiento humano con frecuencia actúa con este tipo de información. La lógica difusa se inició en 1965 por Lotfi A. Zadeh, profesor de la Universidad de California en Berkeley. Surgió como una herramienta importante para el control de sistemas y procesos industriales complejos, así como también para la electrónica de entretenimiento y hogar, sistemas de diagnóstico y otros sistemas expertos.

2.10.1. Concepto de Conjunto Difuso

La lógica difusa permite tratar con información que no es exacta o con un alto grado de imprecisión a diferencia de la lógica convencional la cual trabaja con información precisa. El problema principal surge de la poca capacidad de expresión de la lógica clásica.

2.10.1.1. Conjuntos Clásicos

Los conjuntos clásicos surgen por la necesidad del ser humano de clasificar objetos y conceptos. Estos conjuntos pueden definirse como un conjunto bien definido de o mediante una función de pertenencia μ que toma valores de 0 ó 1 de un universo en discurso para todos los elementos que pueden o no pertenecer al conjunto.

Un conjunto clásico se puede definir con la función de pertenencia mostrada en la ecuación.

$$\mu_A(x) = \begin{cases} 0 & \text{si } x \notin A \\ 1 & \text{si } x \in A \end{cases}$$

2.10.1.2. Conjuntos Difusos

La necesidad de trabajar con conjuntos difusos surge del hecho que existen conceptos que no tienen límites claros. Un conjunto difuso se encuentra asociado por un valor lingüístico que está definido por una palabra, etiqueta lingüística o adjetivo. En los conjuntos difusos la función de pertenencia puede tomar valores del intervalo entre 0 y 1, y la transición del valor entre cero y uno es gradual y no cambia de manera instantánea como pasa con los conjuntos clásicos. Un conjunto difuso en un universo en discurso puede definirse como lo muestra la ecuación.

$$A = \{(x, \mu_A(x)) | x \in U\}$$

Donde $\mu_A(x)$ es la función de pertenencia de la variable x , y U es el universo en discurso. Cuando más cerca este la pertenencia del conjunto A al valor de 1, mayor será la pertenencia de la variable x al conjunto A , esto se puede ver en la figura 2.13 [Andia, 2001].

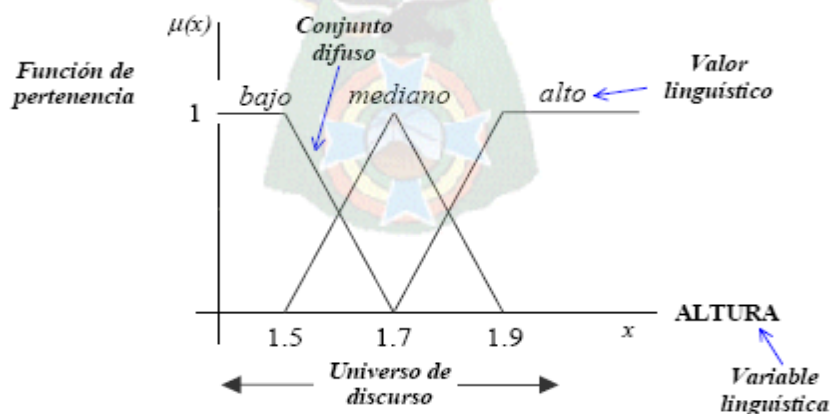


Figura 2. 13 Conjuntos Difusos

Fuente: [Ramírez, 2007]

3. DISEÑO METODOLOGICO

3.1. Introducción

En el presente capítulo presentamos el diseño del sistema de control domótico se realiza la descripción básica de la arquitectura del controlador, presentado en la figura 3.1 donde la interfaz de usuario la desarrollamos en la etapa de software la parte de hardware la detallamos a continuación.

Una vez identificada la arquitectura del controlador se procede a la descripción de los componentes básicos para la construcción del controlador domótico para la implementación es importante tomar en cuenta que los componentes son esenciales para la construcción.

La etapa de construcción del controlador domótico permite tener un panorama amplio del sistema general mostrando la secuencia de pasos que se realiza para la construcción la configuración de los sensores de iluminación y la temperatura. En función a las entradas se determina una acción, es importante realizar la lógica que permitirá la automatización con la cual recurrimos a la lógica difusa para lograr realizar tareas de automatización.

Una vez realizada el control mediante la lógica difusa se desarrolla el software como prototipo, la interfaz de comunicación con el usuario la lógica programada al controlador y su posterior conexión con una página web para el control de dispositivos domóticos.

3.2. Arquitectura del Controlador

El tipo de arquitectura para el sistema de control domótico es la centralizada esto debido a que nuestros dispositivos tanto como sensores y actuadores estarán centralizados a una placa electrónica que al recibir información de los sensores permitirá realizar acciones sobre los dispositivos actuadores como se puede ver en la figura 3.1 los componentes mencionados son detallados en el capítulo anterior.

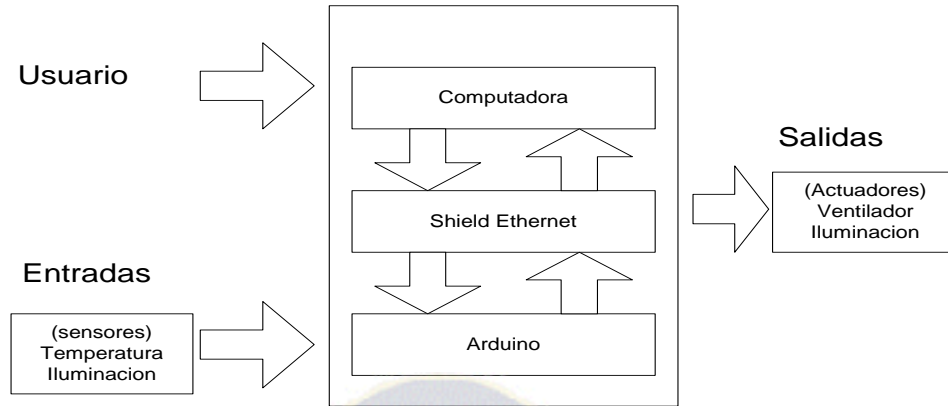


Figura 3. 1 Arquitectura del Controlador

Fuente: [Elaboración Propia]

3.3. Materiales del controlador

Los materiales necesarios para la implementación del prototipo del controlador domótico son los siguientes componentes que se detallan en la tabla 3.1. Donde cada elemento es importante al momento de implementar el sistema sobre todo en la descripción.

Tabla 3. 1 Componentes para la construcción del sistema

Fuente: [Elaboración Propia]

Nro.	Componente	Cantidad	Descripción
1	Computadora	1	Personal
2	Placa Arduino	1	UNO REV 3
3	Arduino Ethernet	1	W5100
4	Sensor de Iluminación	1	LDR
5	Sensor de Temperatura	1	LM35
6	Led	5	Emisor de luz
7	Ventilador	1	Emisor de Aire
8	Protoboard	1	
9	Resistencia	5	1 K Ω
10	Tarjeta SD	1	2 GB
11	Cable de conexión red	1	Categoría 5e

3.4. Construcción del Controlador

Las figuras desplegadas a continuación corresponden al diseño del circuito de sistema, posteriormente se realiza el armado del circuito utilizando la lista de materiales citada en la tabla 3.1

La conexión física de la tarjeta Arduino juntamente con el Shield de Ethernet como se muestra en la figura 3.2 esto permitirá la configuración de puertos de comunicación por Ethernet programación de la placa Arduino que gestiona todo el código para su posterior acción dispositivos actuadores como sensores.



Figura 3. 2 Conexión computadora Placa Electrónica

Fuente: [Elaboración Propia]

Montaje de la placa Arduino con la Shield de Ethernet para la comunicación mediante SPI una librería implementada para Arduino con la finalidad de la comunicación mediante puertos de Ethernet en la figura 3.3 muestra el montaje de la placa Arduino y Shield Ethernet.



Figura 3. 3 Montaje de la Placa Arduino y Shield Ethernet

Fuente: [Elaboración Propia]

Una vez realizada la conexión física se realiza la programación correspondiente con el entorno de desarrollo Arduino. La configuración de los sensores permitirán la captura de datos para su posterior acción a realizar para ello tenemos los dos módulos que son de iluminación y temperatura.

La configuración del circuito de la temperatura la presentamos en la siguiente figura 3.4 ahora realizaremos una descripción breve del circuito, el sensor de temperatura está conectado en ambos extremos a una fuente de alimentación de 5 Voltios y el pin central es encargada de capturar datos analógicos a la placa Arduino.

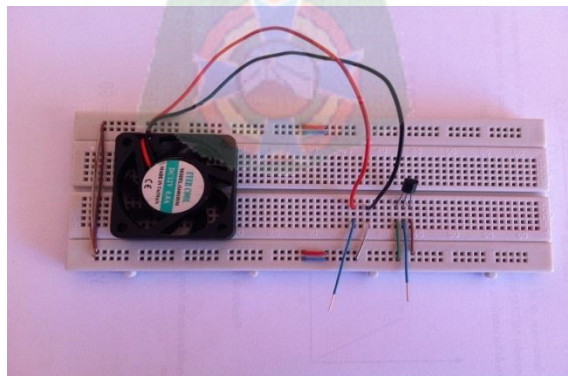


Figura 3. 4 Configuración del Circuito de Temperatura

Fuente: [Elaboración Propia]

La configuración del circuito de iluminación de igual forma la presentamos en la siguiente figura 3.5 a continuación realizamos su descripción el sensor de iluminación consta de dos pines uno de los extremos están contactados a una fuente de alimentación de 5 v uno de los pines tiene una conexión a una resistencia toda esta configuración nos permitirá enviar datos analógicos a nuestro microcontrolador para su posterior tratamiento y su ejecución.

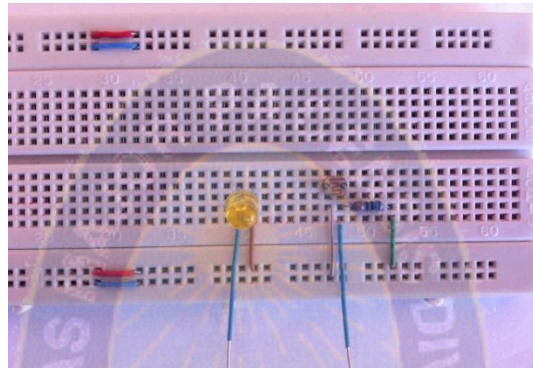


Figura 3. 5 Configuración del Circuito de Iluminación

Fuente: [Elaboración Propia]

La siguiente etapa es la integración de estos dos módulos y la configuración en la placa Arduino de forma que la planteamos como una arquitectura centralizada figura 3.6 esta es la configuración básica de los sensores centralizados mediante un controlador para posteriormente se realiza pruebas de captura de datos de los sensores analógicos.

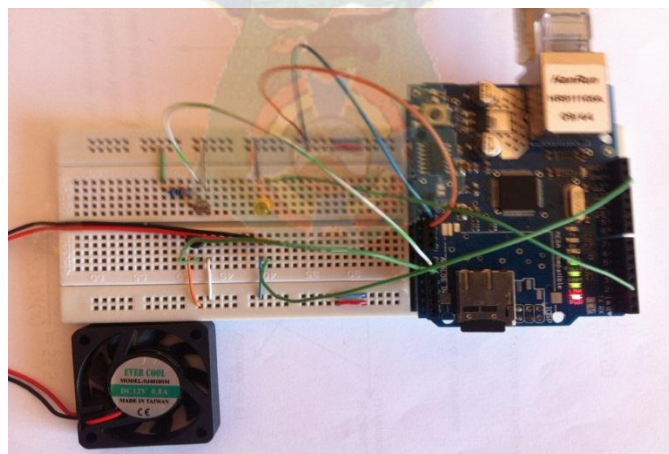


Figura 3. 6 Integración de los Módulos

Fuente: [Elaboración Propia]

Ahora se tiene integrada los módulos de control de iluminación y de temperatura con la correspondiente placa Arduino con conexión Ethernet como se muestra en la figura 3.7 para luego implementar en un prototipo de una vivienda.

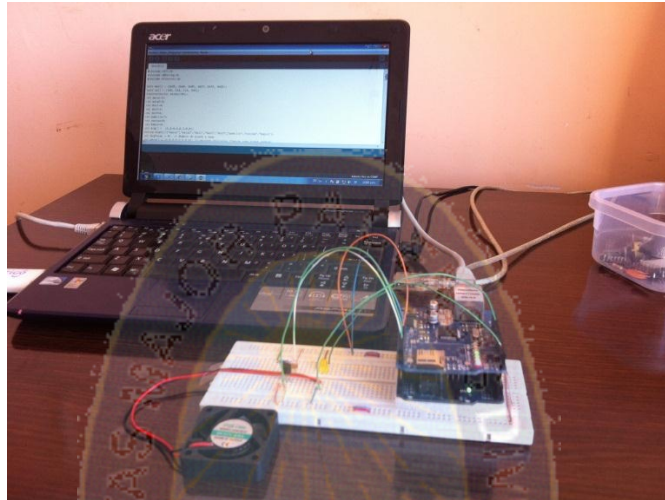


Figura 3. 7 Integración del Sistema de Control

Fuente: [Elaboración Propia]

Ahora se integra al prototipo de una vivienda ubicando los sensores en posiciones adecuadas para la obtención de datos para realizar acciones de control de forma adecuada como se muestra en la figura 3.8 los sensores están ubicados en la parte posterior de la vivienda.

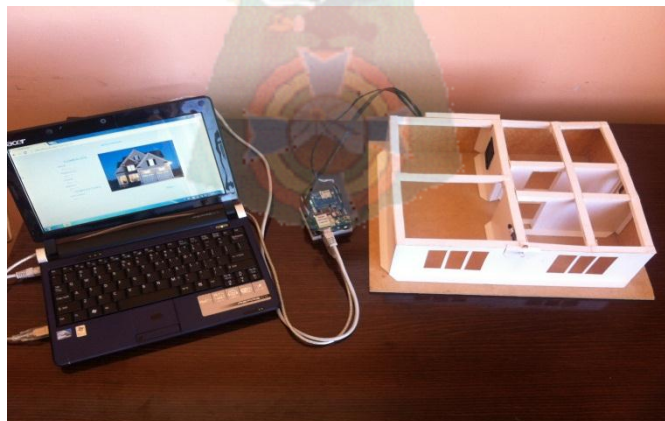


Figura 3. 8 Integrando los sensores a la vivienda

Fuente: [Elaboración Propia]

3.5. Funcionamiento del controlador

El funcionamiento del sistema se desarrolla bajo los componentes como entrada, proceso y salida.

3.5.1. Entrada

Son eventos físicos que se realizan en un ambiente y acciones realizadas por los habitantes de un hogar.

Eventos físicos son: la iluminación natural y la temperatura presente en un ambiente en cualquier momento del día.

Estas entradas son capturadas por los siguientes dispositivos: sensor de temperatura y sensor de iluminación.

3.5.2. Proceso

Está conformado por la captura de datos desde los sensores y actuadores, estos envían tensiones eléctricas ya sea digital o analógica constantemente, después de haber sido instalado en un ambiente. El proceso que se sigue a continuación es obtener estos datos externos verificar el tipo de sensor o actuador se determina una acción para posteriormente enviar a los actuadores o efectores una tensión eléctrica que se transforma en una acción.

3.5.3. Salida

De acuerdo a lo descrito anteriormente los resultados son:

- Apagado y encendido automático de la iluminación.
- Apagado y encendido automático del ventilador según el sensor de temperatura.
- Apagado y encendido manual de la iluminación según el usuario.
- Apagado y encendido manual del ventilador según el usuario.

3.6. Control Mediante Lógica Difusa

El comportamiento de la vivienda automatizada se podrá lograr a partir de la lógica difusa por la medida que tomara de la lectura de los datos enviados por los sensores que se tendrá que determinar cuándo esta oscuro o simplemente cuando el ambiente presente una temperatura no aceptable estos cuadros que presentará la vivienda será de vital importancia.

3.6.1. Fusificación de Entradas

Para el sistema de control domótico se considera las siguientes entradas; temperatura y luminosidad de un ambiente, el rango (dominio) para la temperatura se puede dividir en tres conjuntos difusos que se muestra en la tabla 3.2 parámetros que se deben considerar.

Tabla 3. 2 Parámetros de la Temperatura

Fuente: [Elaboración Propia]

FRIO	0 °C - 10°C
TEMPLADO	10 °C - 25 °C
CALOR	25 ° C - 35 °C

De la misma manera para la intensidad lumínica que existe en un ambiente se considera tres conjuntos difusos para el sistema de control domótico la intensidad de la iluminación tiene como unidad de medida el lux en la tabla 3.3 se considera estos parámetros.

Tabla 3. 3 Parámetros de la Iluminación

Fuente: [Elaboración Propia]

OSCURO	0 – 90 lux
AGRADABLE	60 – 120 lux
BRILLANTE	90 – 150 lux

El dominio de los valores de entrada se denota de la siguiente manera N conjunto difuso denominados $A_1, A_2, A_3, \dots, A_N$, los valores que se obtienen después de la etapa de Fusificación son los grados de pertenencia del valor de entrada x a cada uno de estos conjuntos difusos: $\mu_{A_1}(x), \mu_{A_2}(x), \dots, \mu_{A_N}(x)$.

Para el sistema de la iluminación se denota el conjunto difuso en la figura 3.9.

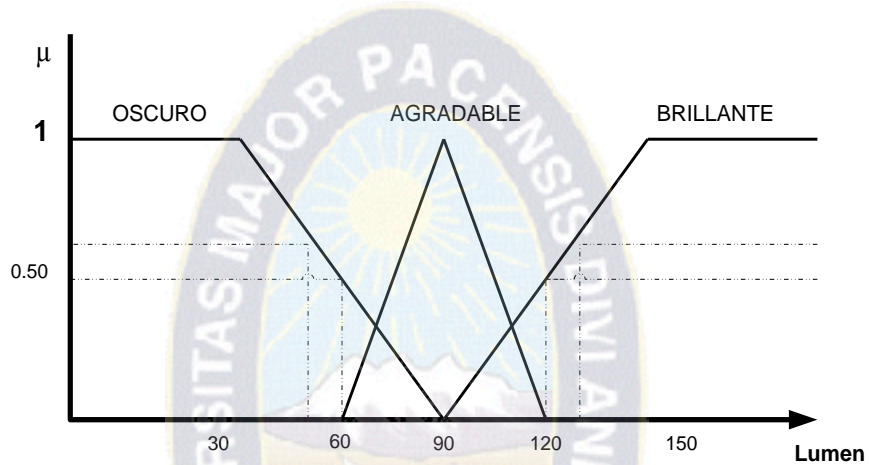


Figura 3. 9 Conjunto difuso para la Iluminación

Fuente: [Elaboración Propia]

Y su correspondiente función de pertenencia para el control de iluminación.

$$\mu_{Oscuro}(x) = \begin{cases} 1, & x \leq a \\ \frac{c-x}{c-a}, & a \leq x \leq c \end{cases}$$

$$\mu_{Agradable}(x) = \begin{cases} 0, & x \leq b \\ \frac{x-b}{c-b}, & b \leq x \leq c \\ \frac{d-x}{d-c}, & c \leq x \leq d \\ 0, & d \leq x \end{cases}$$

$$\mu_{Brillante}(x) = \begin{cases} 0, & x \leq c \\ \frac{x - c}{e - c}, & c \leq x \leq e \\ 1, & e \leq x \leq f \end{cases}$$

Para el cálculo de la iluminación recurrimos a las ecuaciones de función de pertenencia con un valor de $x=70$ lux por lo tanto tenemos los siguientes resultados.

$$\mu_{OSCURO}(70) = 0.33$$

$$\mu_{AGRADABLE}(70) = 0.33$$

$$\mu_{OSCURO}(70) = 0$$

Este valor en nuestro sistema significaría, que el valor ingresado por el sensor de luminosidad fue de 70 lux por lo tanto la acción que se toma es no activar el actuador de luminosidad.

Ahora para el sistema de la climatización se denota el conjunto difuso en la figura 3.10

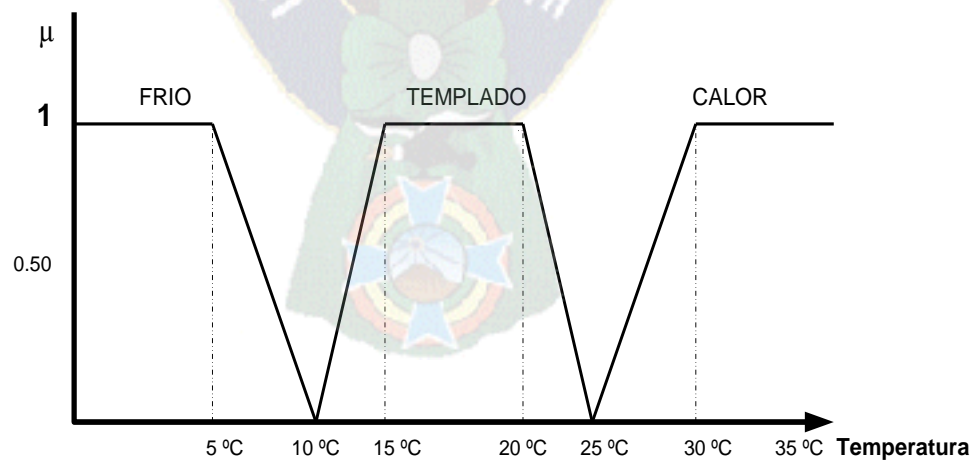


Figura 3. 10 Conjunto difuso para la Temperatura

Fuente: [Elaboración Propia]

Y su correspondiente función de pertenencia para el control de Temperatura.

$$\mu_{Frio}(x) = \begin{cases} 1, & x \leq a \\ \frac{b-x}{b-a}, & a \leq x \leq b \end{cases}$$

$$\mu_{Templado}(x) = \begin{cases} 0, & x \leq b \text{ o } x \geq e \\ \frac{x-b}{c-b}, & b \leq x \leq c \\ 1, & c \leq x \leq d \\ \frac{e-x}{e-d}, & d \leq x \leq e \end{cases}$$

$$\mu_{Calor}(x) = \begin{cases} 0, & x \leq e \\ \frac{x-e}{f-e}, & e \leq x \leq f \\ 1, & f \leq x \leq g \end{cases}$$

Para el cálculo de la temperatura de un ambiente recurrimos a las ecuaciones de función de pertenecía con un valor de $x=12^{\circ}\text{C}$ por lo tanto tenemos los siguientes resultados.

$$\mu_{\text{TEMPLADO}}(12) = 0.4$$

$$\mu_{\text{CALOR}}(12) = 0$$

Este valor en nuestro sistema significaría, que el valor ingresado por el sensor de temperatura fue de 12°C por lo tanto la acción que se toma no activar el ventilador porque está en una temperatura ambiente.

3.6.2. Base de Reglas

Las reglas de comportamiento aplicadas a nuestro controlador, se determinan de la siguiente forma SI – ENTONCES para indicar la acción a realizar en función del conjunto al que pertenece la entrada al controlador de dispositivos. La regla general de una regla es: R: SI (x es A) ENTONCES (y será C).

Entonces se genera las reglas básicas que permitían el tipo de velocidad en función a la temperatura y determinar el tipo de intensidad lumínica en función a la luminosidad.

Control de Temperatura

R1: Si (Calor) entonces (activar ventilador)

R2: Si (Frio) entonces (desactivar ventilador)

R3: Si (Templado) entonces (no realiza ninguna acción)

Control de Iluminación

R1: Si (Oscuro) entonces (activar)

R2: Si (Agradable) entonces (No realiza ninguna acción)

R3: Si (Brillante) entonces (desactivar la iluminación)

3.6.3. Defusificación

El trabajo realizado por la iluminación y el ventilador está en función de la capacidad tanto potencia lumínica como velocidad del ventilador, la siguiente formula permitirá el cálculo de la velocidad como de la potencia que emplea la iluminación.

$$Y_{centroide} = \frac{\sum_{x \in X} x \mu_A(x)}{\sum_{x \in X} \mu_A(x)}$$

Fórmula para el cálculo del centroide discreto

Los parámetros de medida de velocidad del ventilador $V_1 = [0,100] \%$

Los parámetros de medida lumínica en cuanto a potencia $V_2 = [0,100] \%$

El comportamiento de los dispositivos actuadores se definen en variables lingüísticas función de las salidas del controlador difuso mediante las siguientes variables.

$VV \approx V_1$ velocidad de ventilación

$PI \approx V_2$ potencia lumínica

Estas variables lingüísticas se describen a continuación con sus respectivos conjuntos difusos.

Velocidad de la ventilación

VV= {Lenta, Media, Rápida}

Estos parámetros lingüísticos se asocian a funciones de pertenencia figura 3.11 la interpretación de estos conjuntos difusos se detalla a continuación:

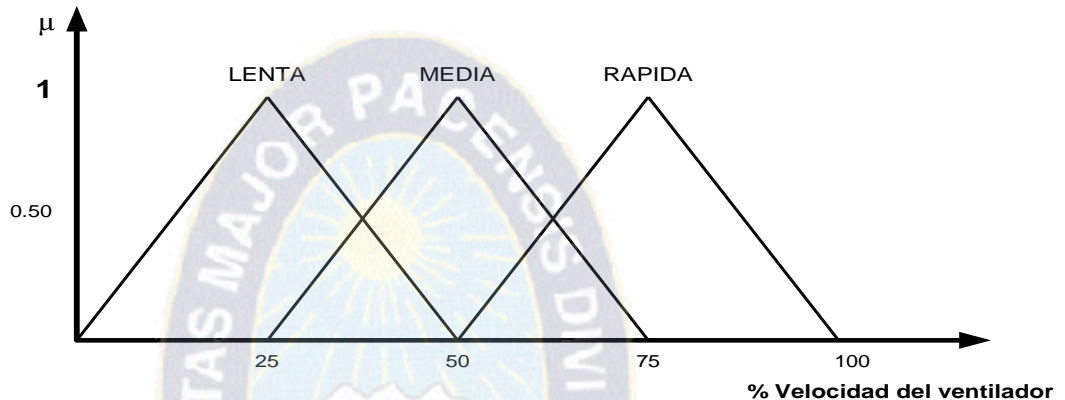


Figura 3. 11 Velocidad del ventilador

Fuente: [Elaboración Propia]

Lenta: La velocidad es lenta en función del valor del intervalo [0, 50] y $x=25$ este es el valor máximo para el grado de pertenencia al conjunto que representa y el funcionamiento será del 25% de su capacidad.

Media: La velocidad es media en función del valor del intervalo [25, 75] y $x=50$ este es el valor máximo para el grado de pertenencia al conjunto que representa cuyo funcionamiento es de 50% de su capacidad.

Rápida: La capacidad es rápida en función del valor del intervalo [50, 100] y $x=75$ este es el valor máximo para el grado de pertenencia al conjunto que representa cuyo funcionamiento es de 75% de su capacidad.

Potencia lumínica

PL = {Mínima, Media, Máxima}

Estos parámetros lingüísticos se asocian a funciones de pertenencia figura 3.12 la interpretación de estos conjuntos difusos se detalla a continuación:

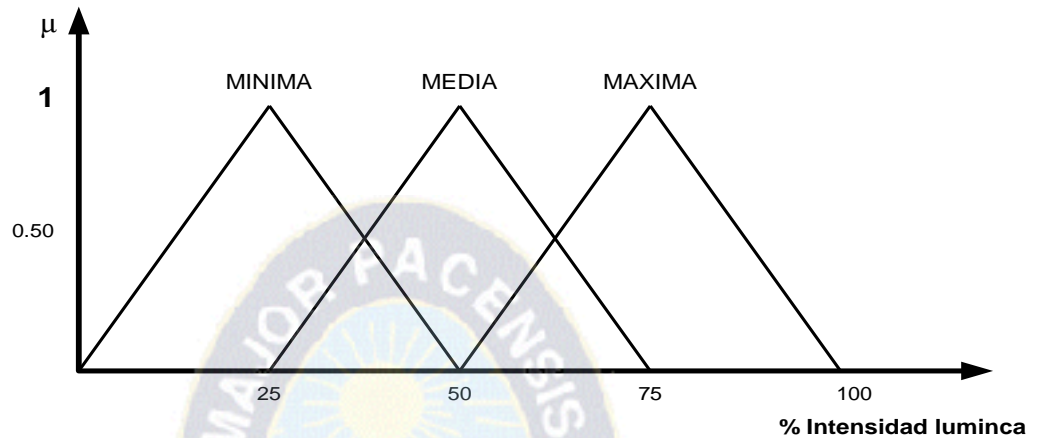


Figura 3. 12 Intensidad Lumínica

Fuente: [Elaboración Propia]

Mínima: La intensidad lumínica es mínima en fusión del valor mínimo que emite el controlador con un intervalo de $[0, 50]$ y $x=25$ este es el valor máximo para el grado de pertenencia al conjunto que representa y el funcionamiento será del 25% de su capacidad.

Media: La intensidad lumínica es media en fusión del valor que emite el controlador con un intervalo de $[25, 75]$ y $x=50$ este es el valor máximo para el grado de pertenencia al conjunto que representa y el funcionamiento será del 50% de su capacidad.

Máxima: La intensidad lumínica es máxima en fusión del valor que emite el controlador con un intervalo de $[50, 100]$ y $x=75$ este es el valor máximo para el grado de pertenencia al conjunto que representa y el funcionamiento será del 75% de su capacidad.

En esta etapa aplicando la fórmula del centroide se obtiene la velocidad que debe emprender el ventilador en función de las reglas, se necesita de un valor exacto como se muestra en la figura 3.13 para realizar la acción del ventilador por lo tanto el nuevo valor de salida es la siguiente la suma de las funciones sombreadas genera el valor esperado.

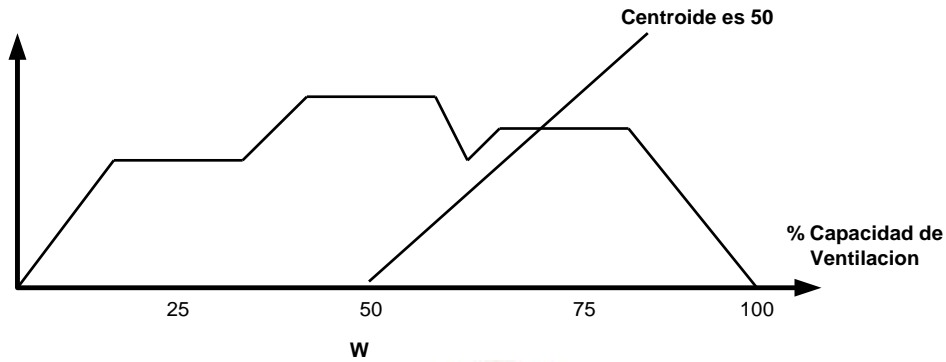


Figura 3. 13 Función de pertenencia global de la temperatura

Fuente: [Elaboración Propia]

El cálculo considerando la función de pertenencia con un rango de 0 a 100.

$$Y_{centroide} = \frac{\sum_{x \in X} x \mu_A(x)}{\sum_{x \in X} \mu_A(x)} = 50$$

La iluminación es de 90% generando la regla dos correspondiente a la velocidad media calculando las funciones sombreadas.

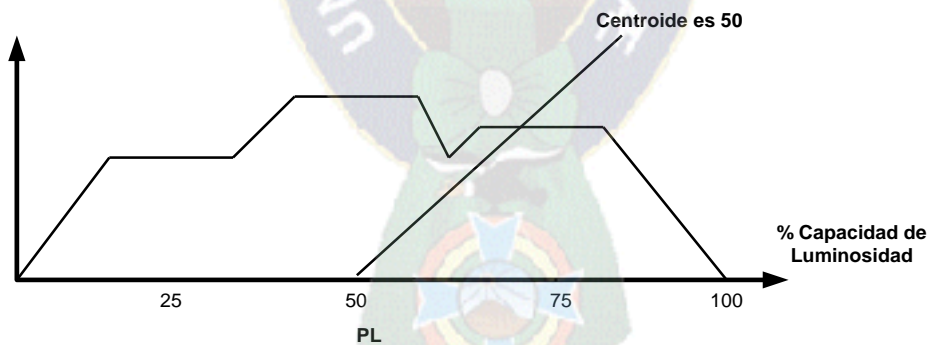


Figura 3. 14 Función de pertenencia global de la Iluminación

Fuente: [Elaboración Propia]

El cálculo considerando la función de pertenencia con un rango de 0 a 100.

$$Y_{centroide} = \frac{\sum_{x \in X} x \mu_A(x)}{\sum_{x \in X} \mu_A(x)} = 50$$

3.7. Desarrollo de Software

El diseño y desarrollo del software para el prototipo sigue la metodología de programación XP (programación extrema) que se describe en el capítulo dos del presente trabajo de investigación. Para lo cual se sigue las fases para el desarrollo del software rápido y funcional para su implementación.

3.7.1. Planificación

Se realiza un plan de desarrollo de proyecto en función de las especificaciones, propósito del sistema, historias de usuario.

3.7.1.1. Propósito del sistema

El diagrama de bloque (figura 3.15) representa el sistema general y especifica las utilidades de cada uno de los componentes y su conjunto muestra las utilidades del sistema, el controlador domótico se puede implementar en diferentes plataformas como Windows, Linux, Mac OS pero para nuestras pruebas se realizó en Windows7.

Los modos en los que opera el sistema de control son dos: modo manual modo automático.

Modo Manual. Significa activar o desactivar los actuadores iluminación ventiladores conectados al sistema de control mediante la computadora.

Modo Automático. Significa que el sistema tomará una determinada acción sobre los dispositivos actuadores como ser iluminación y ventiladores.

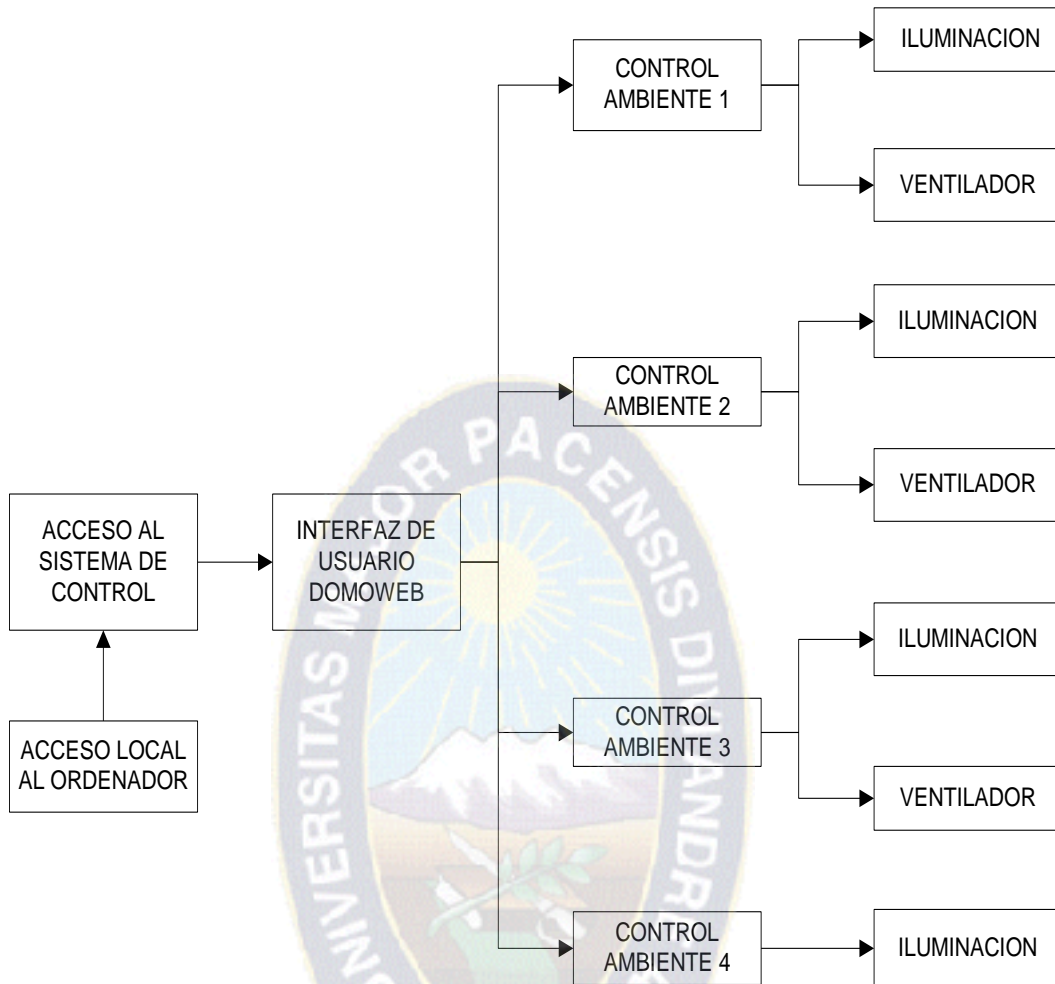


Figura 3. 15 Diagrama de bloques del controlador doméstico

Fuente: [Elaboración Propia]

Una vez especificada el propósito del sistema de control doméstico a partir del diagrama de bloques de la figura 3.15 generamos las historias de usuario.

3.7.1.2. Historias de Usuario

El acceso al sistema de control es exclusivamente para usuarios en la tabla 3.4 se especifica el proceso, el ingreso al sistema se realiza a partir de un administrador de sistema.

Tabla 3. 4 Historia de usuario acceso al sistema de control

Fuente: [Elaboración Propia]

Historia de Usuario	
Número: 1	Usuario: Habitante del hogar
Nombre historia: Usuario Propietario	
Prioridad en negocio: Alta	Riesgo en desarrollo: Baja
Programador responsable: Reynaldo Coronel Lazo	
Descripción: <ul style="list-style-type: none"> • Acceso al sistema de control domótico mediante su correspondiente usuario, contraseña. • Despliegue del menú de selección de control automático o manual pero también la administración de usuarios. 	
Observaciones: El ingreso se habilita a usuarios registrados.	

En las siguientes tablas se muestra los principales componentes del sistema de control domótico el: en modo automático tabla 3.5 monitorea el estado actual de los ambientes en función a los sensores conectados en el hogar, realizando acciones de encendido a apagado de los dispositivos de forma automática, modo manual tabla 3.6 verifica constantemente el estado de los dispositivos actuadores de un hogar permitiendo accionar de forma directa sobre estos dispositivos conectados al sistema de control domótico.

Tabla 3. 5 Historia de usuario control automático

Fuente: [Elaboración Propia]

Historia de Usuario	
Número: 2	Usuario: Habitante del hogar
Nombre historia: Control automático	
Prioridad en negocio: Alta	Riesgo en desarrollo: Baja
Programador responsable: Reynaldo Coronel Lazo	

<p>Descripción:</p> <ul style="list-style-type: none"> • Una vez ingresado al sistema domótico se realiza una selección del modo de control de los dispositivos para su posterior acción. • Al seleccionar la configuración de modo automático nos desplegara los mensajes de estado. • Esta configuración se activa o desactiva de acuerdo a los valores programados los valores que emiten los sensores serán visualizados en la interfaz.
<p>Observaciones:</p> <p>El sistema de control automático recibe información de los sensores permitiendo ver el estados de los ambientes en cuanto a clima.</p>

Tabla 3. 6 Historia de usuario control manual

Fuente: [Elaboración Propia]

Historia de Usuario	
Número: 3	Usuario: Habitante del hogar
Nombre historia: Control manual	
Prioridad en negocio: Alta	Riesgo en desarrollo: Baja
Programador responsable: Reynaldo Coronel Lazo	
<p>Descripción:</p> <ul style="list-style-type: none"> • Una vez ingresado al sistema domótico se realiza una selección del modo de control de los dispositivos para su posterior acción. • Al seleccionar habitaciones se activa el modo manual la cual permite ver los estados de los actuadores permite la acción de encender o apagar la iluminación, ventilador de un ambiente. 	
<p>Observaciones: En el modo manual el propietario de la vivienda puede realizar la verificación de los estados de los actuadores conectados a la red domótica.</p>	

Los requerimientos que debe cumplir administrar usuarios para el control domótico se detalla en la tabla 3.7, se genera a partir de un usuario administrador que permite el acceso

a otros usuarios por el cual se realiza un registro de nuevo usuario en el sistema para posteriores intervenciones con el sistema.

Tabla 3. 7 Historia de usuario Administrar

Fuente: [Elaboración Propia]

Historia de Usuario	
Número: 4	Usuario: Habitante del hogar
Nombre historia: Administración	
Prioridad en negocio: Alta	Riesgo en desarrollo: Baja
Programador responsable: Reynaldo Coronel Lazo	
Descripción: <ul style="list-style-type: none"> • Una vez ingresado al sistema domótico se realiza una selección del modo de control de los dispositivos para su posterior acción. • La administración de usuarios solo registrados permite al acceso quedando la posibilidad de registrar un nuevo usuario para que acceda al sistema. 	
Observaciones: El acceso solo se permite a usuarios registrados como administradores, se realiza la verificación de usuarios.	

3.7.2. Diseño

El diseño del sistema de control domótico en cuanto a software está en función a una especificación de requerimientos el modelo entidad relación para la administración de dispositivos y cuentas de usuario posteriormente se realiza la instalación para un cliente específico posteriormente realizar una comprobación y la verificación de resultados.

3.7.2.1. Especificación de requerimientos

El desarrollo del prototipo del sistema domótico para el control de la iluminación y la temperatura tiene como objetivo crear un controlador en donde los usuarios habitantes de los ambientes de un departamento tendrán el control de la iluminación y la temperatura de forma manual y automática. La especificación de los procesos se detalla en la siguiente tabla 3.8

Tabla 3. 8 Procesos del Prototipo

Fuente: [Elaboración Propia]

Nro.	Proceso
1	Desplegar una pantalla de autenticación para el ingreso al sistema por medio de un navegador.
2	Desplegar una pantalla de portada del hogar visualizando el contexto de los ambientes
3	Desplegar una pantalla del estado de los actuadores encendido o apagado de forma manual
4	Desplegar una pantalla del contexto del modo automático.
5	Desplegar una pantalla el apagado o encendido de los actuadores de forma automática en función de los sensores
6	Desplegar una pantalla para la administración de nuevos usuarios

3.7.2.2. Modelo Entidad Relación del Prototipo

El modelo entidad relación permite el ingreso al sistema de control domótico y administrar los dispositivos el diagrama se muestra (en ANEXO A) también permite gestionar nuevos usuarios.

Para el desarrollo del prototipo de control domótico la secuencia de pasos que realiza la placa electrónica presentada en la figura 3.16 que indica la secuencia de pasos que sigue la placa electrónica.

3.7.2.3. Desarrollo

El desarrollo del software controlador domótico consta de un diseño de datos básicamente se define las librerías de comunicación entre Arduino, shield Ethernet el diseño de software.

➤ **Diseño de datos**

El diseño de datos es una de las fases más importantes en el desarrollo de software, traduce los objetos de datos en estructuras globales a nivel componentes de software el desarrollo del prototipo se compone de un conjunto específicamente de tres librerías propia de la plataforma Arduino.

Librería SPI

Esta librería le permite comunicarse con los dispositivos SPI, con el Arduino como dispositivo maestro.

De serie Peripheral Interface (SPI) es un protocolo de datos en serie síncrono utilizado por los microcontroladores para comunicarse con uno o más dispositivos periféricos rápidamente en distancias cortas. También se puede utilizar para la comunicación entre dos microcontroladores.

Con una conexión de SPI siempre hay un dispositivo maestro (por lo general un microcontrolador) que controla los dispositivos periféricos. Normalmente hay tres líneas comunes a todos los dispositivos:

MISO (Master In Slave Out) La línea de esclavos para el envío de datos al maestro,

MOSI (Master Out Slave In) La línea principal para el envío de datos a los periféricos,

SCK (Serial Clock) Los impulsos de reloj que sincronizan la transmisión de datos generada por el maestro y una línea específica para cada dispositivo:

SS (Slave Select) - el pasador en cada dispositivo que el maestro puede utilizar para activar y desactivar dispositivos específicos.

Librería Ethernet

Junto con el shield Ethernet de Arduino Ethernet Shield, esta librería permite a la placa Arduino de conectarse a Internet. Puede funcionar tanto como servidor capaz de aceptar conexiones entrantes, como cliente permitiendo realizar conexiones de salida. La librería permite hasta cuatro conexiones simultáneas (entrantes, salientes, o una combinación de ambas).

Librería SD

La librería SD permite la lectura y escritura en las tarjetas SD, en el Arduino Ethernet Shield. Está construido sobre sdfatlib, la biblioteca es compatible con FAT16 y FAT32 sistemas de archivos en tarjetas SD estándar y tarjetas SDHC.

La comunicación entre el microcontrolador y la tarjeta SD utiliza SPI , que tiene lugar en los pines digitales 11, 12 y 13 (en la mayoría de las placas Arduino) o 50, 51, y 52 (Arduino Mega). Además, otro pin debe ser utilizado para seleccionar la tarjeta SD. Este puede ser el pin hardware SS - pin 10 (en la mayoría de las placas Arduino) o pin 53 (en la Mega).

➤ Diseño de software

El diseño para el desarrollo de software básicamente se constituye como se muestra en la (figura 3.16), el chip w5100 que es la base del módulo shield Ethernet, la descripción se encuentra en la ficha técnica vea en (ANEXO B) este chip tiene el comportamiento de cliente servidor o servidor cliente para el prototipo se configura de la siguiente forma nuestro servidor está en base a shield Ethernet y la función de cliente la página web como aplicación del controlador domótico. Arduino almacena en la tarjeta SD la configuración para el control y la monitorización desde un navegador web en la red Ethernet.

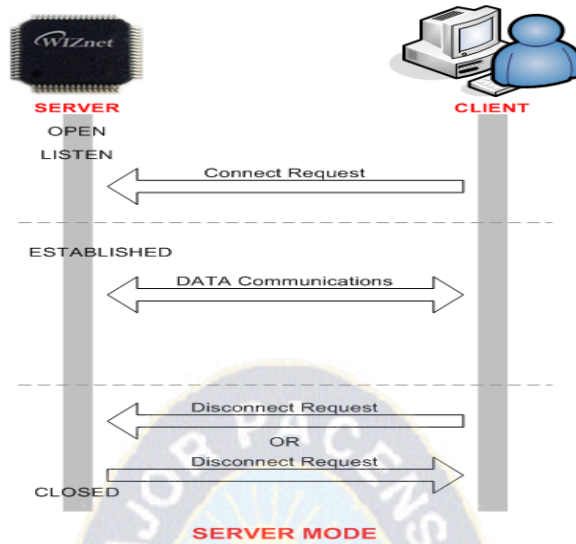


Figura 3. 16 Modo servidor

Fuente: [wiznet]

Ciente

Por el lado cliente el proceso que sigue se muestra en la figura 3.17 en su descripción general el navegador realiza peticiones mediante JavaScript, el modulo Ajax permite actualizar constantemente los campos de datos de los sensores y actuadores que están en el Arduino en formato xml para luego recibirlas en funciones JavaScript las cuales devuelven en formato HTML esto se gestiona en la tarjeta SD de la placa Arduino mediante las peticiones realizadas por el cliente.

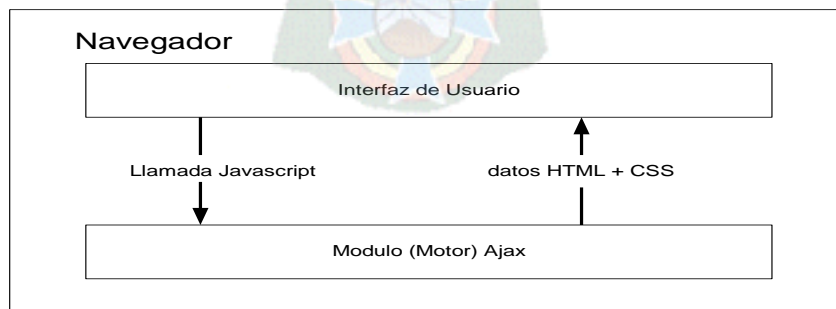


Figura 3. 17 Cliente

Fuente: [Elaboración Propia]

Solicitud del cliente

La navegación que realiza un cliente en la IP del servidor de Arduino web (cliente) enviará una solicitud, como el que se muestra a continuación en el fragmento de código de peticiones de la (figura 3.18). La información en la solicitud puede diferir, dependiendo del navegador y sistema operativo que se envía la solicitud.

Los `\r\n` caracteres que ve al final de cada línea de texto en la solicitud son caracteres no visibles (caracteres no imprimibles). `\r` es el carácter de retorno de carro y `\n` es el carácter de salto de línea (o carácter de nueva línea).

```
GET / HTTP/1.1 \r\n
Anfitrión: 10.0.0.20 \r\n
User-Agent: Mozilla/5.0 (X11; Ubuntu, Linux i686; rv: 17.0) Gecko/20100101
Firefox/17.0 \r\n
Accept: text/html, application/xhtml+xml, application/xml; q=0.9, */*; q=
0.8 \r\n
Accept-Language: en-ZA, en-GB; q=0.8, es-ES; q=0.5, es; q=0.3 \r\n
Accept-Encoding: gzip, deflate \r\n
Conexión: keep-alive \r\n
\r\n
```

Figura 3. 18 Solicitud cliente

Fuente: [Elaboración Propia]

La última línea de la solicitud es simplemente `\r\n` sin precedentes y texto. Arduino antes de enviar una respuesta al navegador web cliente. Se realiza la lectura de cada carácter de la solicitud arriba y sabe cuándo se ha alcanzado el final de la petición, ya que se encuentra la línea en blanco.

Respuesta del Servidor

Después de recibir la petición de una página web por parte del cliente, el servidor envía primero una respuesta HTTP estándar y luego la propia página web. La respuesta enviada desde el Arduino es como sigue en la (figura 3.19)

```
HTTP/1.1 200 OK \r\n
Content-Type: text/html \r\n
Connection: close \r\n
\r\n
```

Figura 3. 19 Respuesta servidor

Fuente: [Elaboración Propia]

La petición y la respuesta mediante HTTP (Hypertext Transfer Protocol)la prueba de comunicación con la tarjeta Arduino muestra en la figura 3.20 en la cual al ejecutar en el navegador por el cliente se recibe esta petición de inicio de la tarjeta SD donde se gestiona la página web esta verificación se realiza por el lado de Arduino.

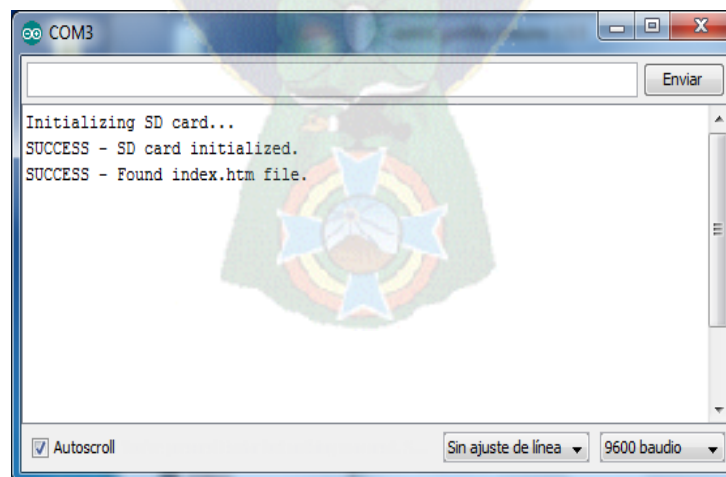


Figura 3. 20 Petición Tarjeta SD

Fuente: [Elaboración Propia]

➤ **Desarrollo**

La lógica de servidor cliente se muestra en el diagrama de flujo presentado en la figura 3.21 que es la configuración básica de un servidor Arduino.

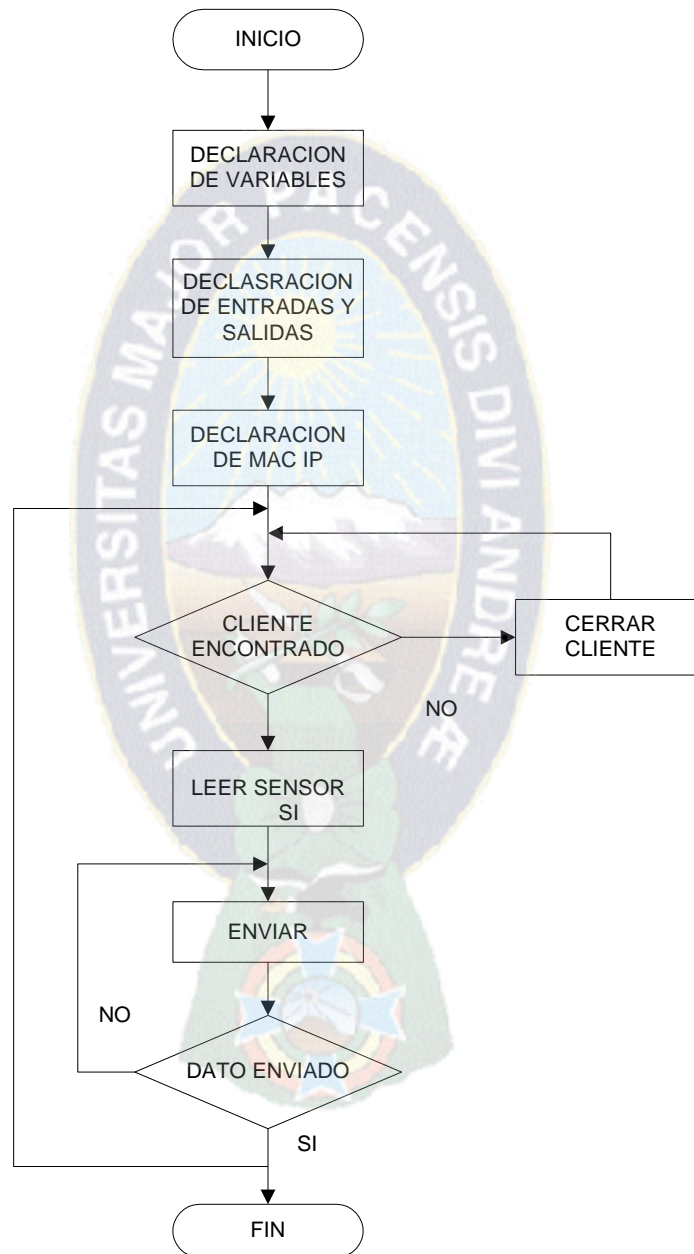


Figura 3. 21 Diagrama de Flujo Arduino Conexión web

Fuente: [Elaboración Propia]

El sistema propuesto de forma general se muestra en la figura 3.22 es un programa estructurado básicamente donde contiene la lógica de control automático y manual.

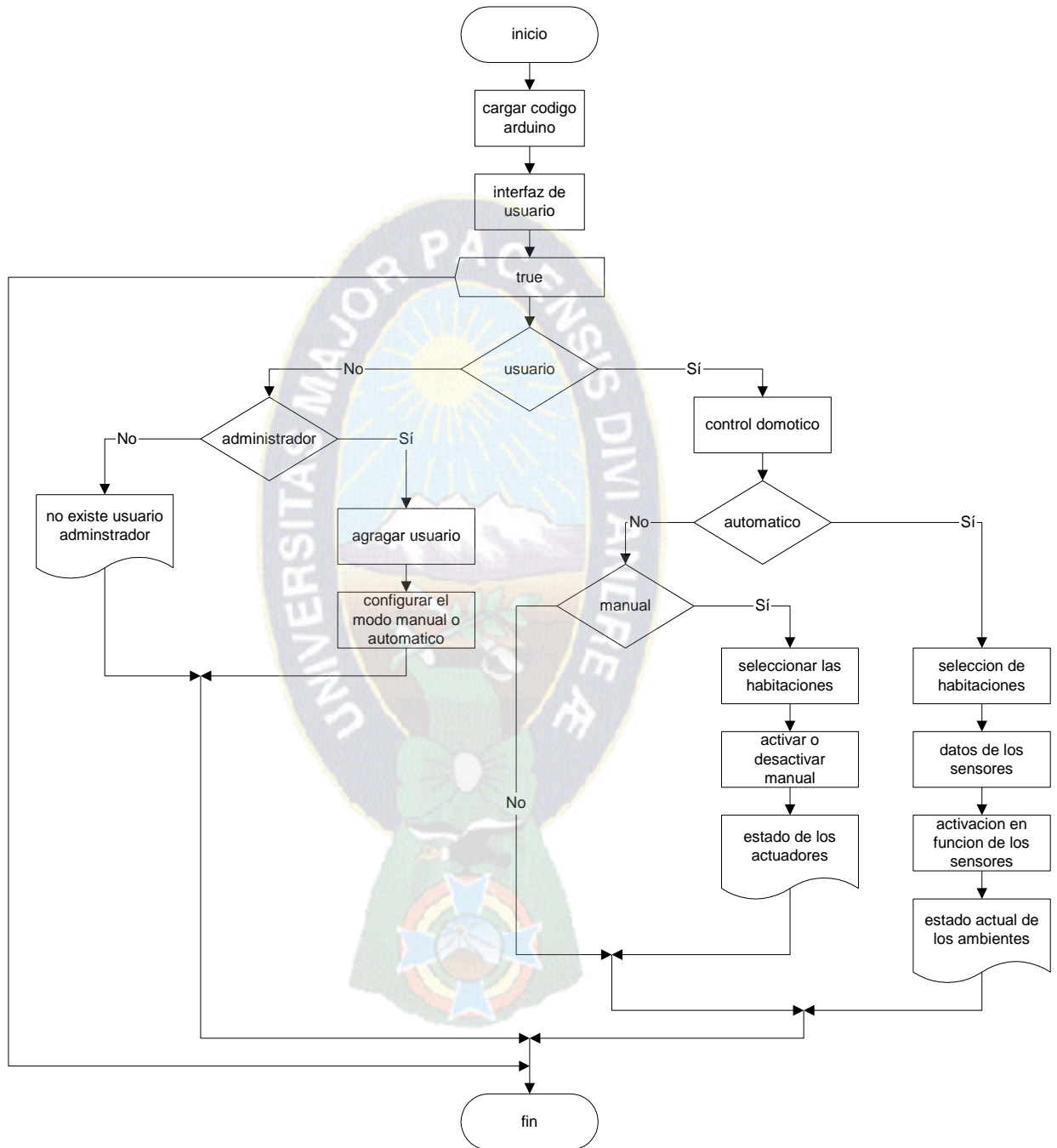
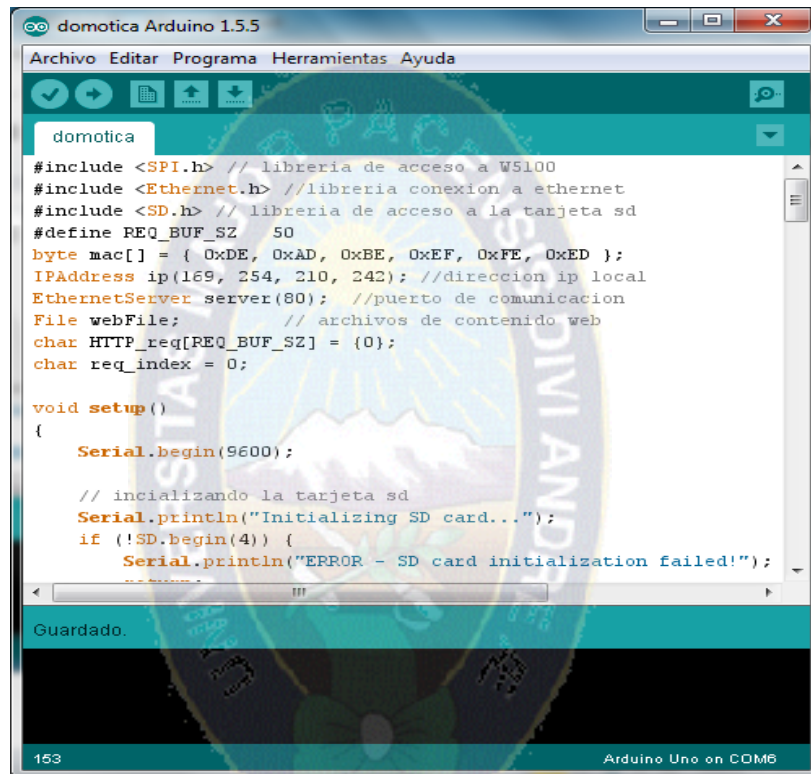


Figura 3. 22 Diagrama de flujo sistema general

Fuente: [Elaboración Propia]

3.7.3. Codificación

El entorno de desarrollo para el control domótico se muestra en la figura 3.23 donde se define la conexión por puerto Ethernet que es una etapa importante en esta parte para el sistema donde se determina una dirección ip(protocolo de internet) de prueba.



```
domotica Arduino 1.5.5
Archivo  Editor  Programa  Herramientas  Ayuda
domotica
#include <SPI.h> // libreria de acceso a W5100
#include <Ethernet.h> //libreria conexion a ethernet
#include <SD.h> // libreria de acceso a la tarjeta sd
#define REQ_BUF_SZ  50
byte mac[] = { 0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED };
IPAddress ip(169, 254, 210, 242); //direccion ip local
EthernetServer server(80); //puerto de comunicacion
File webFile; // archivos de contenido web
char HTTP_req[REQ_BUF_SZ] = {0};
char req_index = 0;

void setup()
{
  Serial.begin(9600);

  // incializando la tarjeta sd
  Serial.println("Initializing SD card...");
  if (!SD.begin(4)) {
    Serial.println("ERROR - SD card initialization failed!");
  }
}

Guardado.
153 Arduino Uno on COM6
```

Figura 3. 23 Interfaz del programa Arduino

Fuente: [Elaboración Propia]

El código del sistema de control de dispositivos domótico se muestra en (ANEXO C) y la lógica de control de forma automática y manual en forma detallada la estructura del controlador configuración de servidor cliente en Arduino.

La codificación para el control automático se muestra en la figura 3.24 donde se observa, un fragmento sobre la lógica de control automático de la iluminación y temperatura.

```
automatico
sensor = 5*analog_val*100/1024;
if(sensor > 35)
  digitalWrite(temp, HIGH);
else
  digitalWrite(temp, LOW);
cl.print("<analog>");
cl.print(temp);
cl.println("</analog>");
}
else
{
  if(sensor < 180)
    digitalWrite(luz, HIGH);
  else
    digitalWrite(luz, LOW);
cl.print("<analog>");
cl.print(sensor);
cl.println("</analog>");
}
nin ++;
```

Guardado.

23 Arduino Uno en CQM3

Figura 3. 24 Fragmento código control automático

Fuente: [Elaboración Propia]

La lógica de control manual se muestra en la figura 3.25, en el método XML_response se realiza la asignación en función de un evento realizado por una página web y al mismo tiempo se verifica el estado del actuador y se envía en formato xml.

```
manual
void SetLEDs(void)
{
  if (StrContains(HTTP_req, "LED1=1")) {
    LED_state[0] = 1;
    digitalWrite(SALON, HIGH);}
  else if (StrContains(HTTP_req, "LED1=0")) {
    LED_state[0] = 0;
    digitalWrite(SALON, LOW);}
}
void XML_response(EthernetClient cl)
{
  cl.print("<?xml version = '1.0' ?>");
  cl.print("<inputs>");
  cl.print("<LED>");
  if (LED_state[0]) {
    cl.print("on");}
  else {
    cl.print("off");
  }
  cl.println("</LED>");
  cl.print("</inputs>");
}
}
```

Guardado.

Figura 3. 25 Fragmento código control Manual

Fuente: [Elaboración Propia]

3.7.4. Pruebas

La implementación del sistema cuenta con dos componentes principales: modo manual y modo automático así como la conexión como parte importante de la funcionalidad, el servidor Ethernet del shield de Arduino permite gestionar la interfaz de usuario mediante una página web.

➤ Interfaz de inicio de sesión

El despliegue de pantalla que apreciamos en la figura 3.26 permite el ingreso al sistema de control domótico esto es solo para usuarios correctamente identificados.

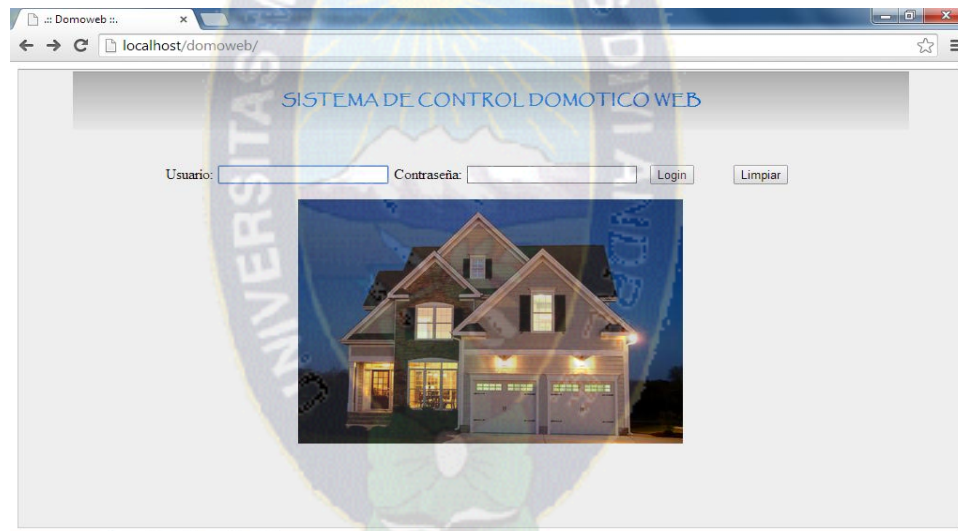


Figura 3. 26 Inicio de sesión para usuarios

Fuente: [Elaboración Propia]

➤ La interfaz opciones de control

En la figura 3.27 se observa desde un menú de accesos con las cuales se administra el tipo de acción que quiere tomar esto se realiza solo para usuarios propietarios de la vivienda pero también a partir de un usuario identificado se autoriza el acceso a otros usuarios bajo el permiso de un administrador para posteriormente registrarse como usuario nuevo considerar que solo puede gestionar los dispositivos usuarios que están registrados en el sistema.

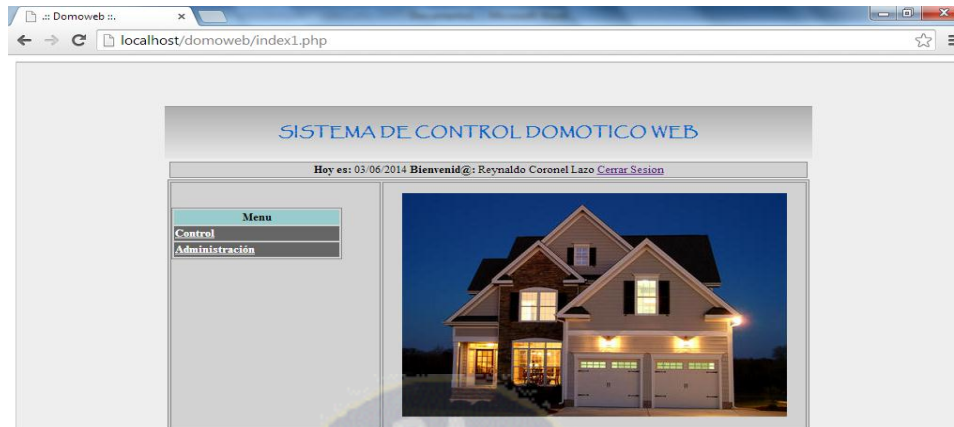


Figura 3.27 Opciones de Control

Fuente: [Elaboración Propia]

➤ **Acceso control**

Selección manual y automática de un hogar en el modo manual se tiene todos los ambientes conectados el modo automático genera el estado sobre climatización en la figura 3.28 el control de accesos.

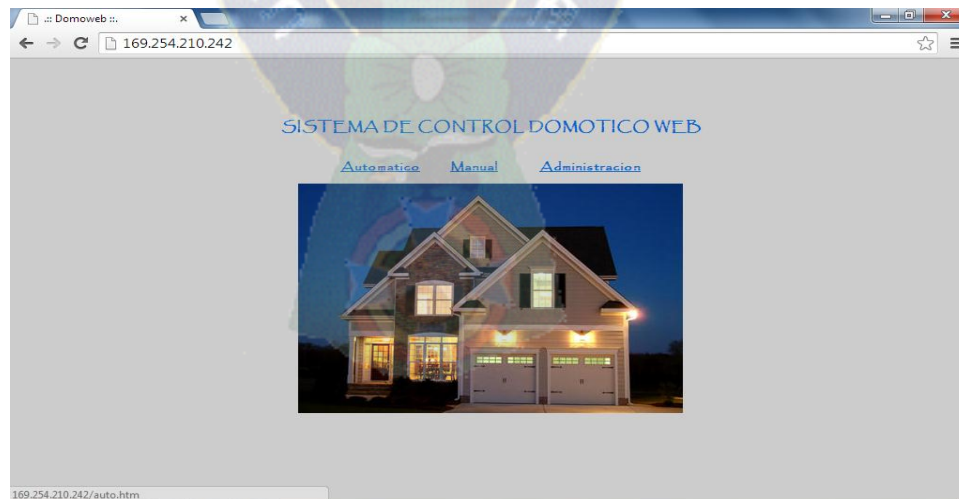


Figura 3. 28 Opciones de Modo

Fuente: [Elaboración Propia]

➤ **Modo manual**

Este modo se encarga de ejecutar acciones de forma manual pero teniendo un panorama de las habitaciones del departamento de la iluminación y la temperatura, aquí se ve en la figura 3.29 el accionar de forma directa con los dispositivos conectados al controlador.

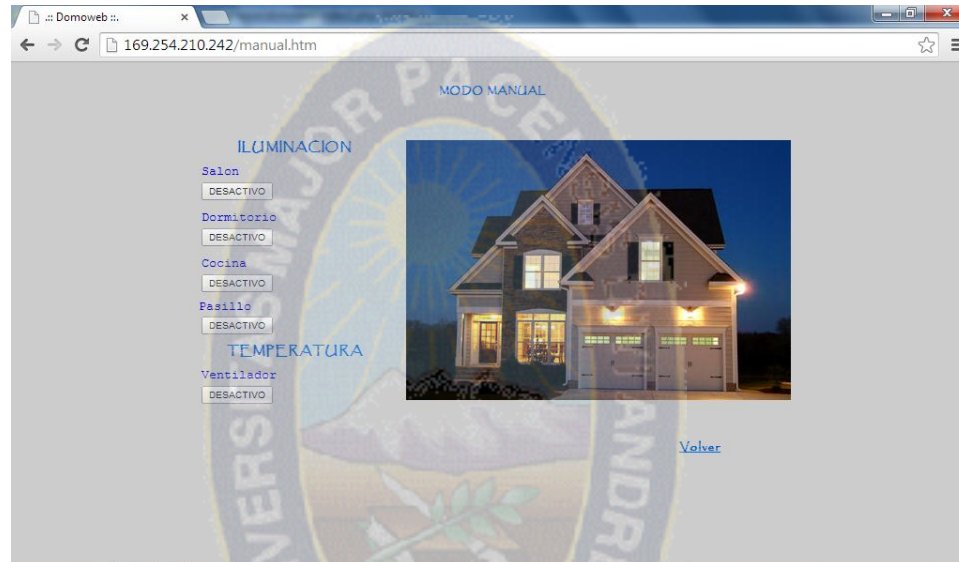


Figura 3. 29 Modo Manual

Fuente: [Elaboración Propia]

➤ **Modo automático**

Esta encargada de realizar acciones de forma automática, en la figura 3.30 se observa el estado en el cual se encuentra un ambiente de acuerdo a la lógica programada del controlador, se ejecuta la acción de encendido o apagado de los actuadores conectados al controlador domótico en función a los datos proporcionados en tiempo real. La temperatura esta calibrada en función a las reacciones físicas del ambiente, de igual manera el sensor de iluminación y su posterior actuador lo que permite el control automático.

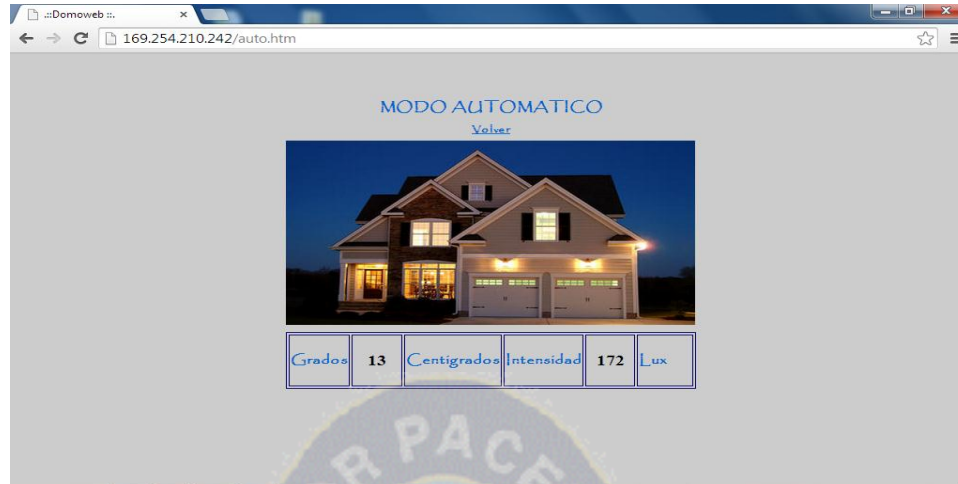


Figura 3. 30 Modo Automático

Fuente: [Elaboración Propia]

➤ **Administración**

La Administración permite el ingreso a usuarios nuevos, para el registro de nuevos usuarios se llena los siguientes campos que se muestra en la figura 3.31, un administrador tiene el privilegio de gestionar nuevos usuarios para que realice acciones de control.



Figura 3. 31 Administración de usuarios

Fuente: [Elaboración Propia]

4. EVALUACION DE RESULTADOS

4.1. Introducción

En el presente capítulo se realiza un análisis de los resultados obtenidos y recolectados de los experimentos realizados con la implementación del prototipo de control domótico y datos obtenidos de una vivienda sin domótica para posteriormente generar una tabla final donde se muestra resultados experimentados con domótica y sin domótica esto permitirá demostrar la hipótesis planteada pero también se realiza el análisis de costos de estos sistemas de control domótico.

El análisis de consumo de la energía eléctrica se realiza en función a la fórmula presentada a continuación.

$$\text{Consumo Mensual} = (\text{Horas de utilización mensual} * \text{potencia})/1000$$

$$\text{Costo de Kwh} = \text{Monto total a pagar} / \text{Consumo}$$

Considerando que 1 kwh tiene un costo de 0,58 en área urbana y de 0,60 en el área rural esta información se obtiene de la empresa de electricidad de la ciudad de la paz DELPAZ con esta información se realiza los cálculos sobre el consumo de energía eléctrica por mes.

4.2. Datos obtenidos para la demostración

Recolección de Datos Sin Domótica

Escenario 1

La recolección de datos se realizó en ambientes que se describen a continuación: Departamento con un Salón tabla 4.1 Dormitorio tabla 4.2 Cocina tabla 4.3 Pasillo tabla 4.4 con características de consumo elevado en Kwh.

Tabla 4. 1 Salón de un departamento

Fuente: [Elaboración Propia]

Nro.	Cantidad	Dispositivo	Descripción	Potencia(Wats)	Tiempo (horas)	Periodo(días)	Consumo Mes (Kwh)
1	2	Luminaria	Incandescente	100	5	30	30
2	1	Ventilador	Grande	200	8	30	48
Consumo total aproximado(Kwh)							78
Consumo total por mes en bolivianos							45,55
Costo total año							546,6

Tabla 4. 2 Dormitorio de un departamento

Fuente: [Elaboración Propia]

Nro.	Cantidad	Dispositivo	Descripción	Potencia(Wats)	Tiempo (horas)	Periodo(días)	Consumo Mes (Kwh)
1	1	Luminaria	Incandescente	100	7	30	21
2	1	Ventilador	Grande	200	4	30	24
Consumo total aproximado(Kwh)							45
Consumo total por mes en bolivianos							26,28
Costo total año							315,36

Tabla 4. 3 Cocina de un departamento

Fuente: [Elaboración Propia]

Nro.	Cantidad.	Dispositivo	Descripción	Potencia(Wats)	Tiempo (horas)	Periodo(días)	Consumo Mes (Kwh)
1	1	Luminaria	Incandescente	100	4	30	12
2	1	Ventilador	Grande	200	2	30	12
Consumo total aproximado(Kwh)							24
Consumo total por mes en bolivianos							14,02
Costo total año							168,24

Tabla 4. 4 Pasillo de un departamento

Fuente: [Elaboración Propia]

Nro.	Cantidad.	Dispositivo	Descripción	Potencia(Wats)	Tiempo (horas)	Periodo(días)	Consumo Mes (Kwh)
1	1	Luminaria	Incandescente	100	8	30	24
2	0	Ventilador		0	0	30	0
Consumo total aproximado(Kwh)							24
Consumo total por mes en bolivianos							14,02
Costo total año							168,24

Escenario 2

En el siguiente escenario se recolectaron datos con características detalladas a continuación: Departamento con un Salón tabla 4.5 Dormitorio tabla 4.6 Cocina tabla 4.7 Pasillo tabla 4.8 con características de consumo reducido en Kwh.

Tabla 4. 5 Salón de un departamento

Fuente: [Elaboración Propia]

Nro.	Cantidad.	Dispositivo	Descripción	Potencia(Wats)	Tiempo (horas)	Periodo(días)	Consumo Mes (Kwh)
1	2	Luminaria	Ahorrador	20	5	30	6
2	1	Ventilador	Normal	100	8	30	24
Consumo total aproximado(Kwh)							30
Consumo total por mes en bolivianos							17,52
Costo total año							210,24

Tabla 4. 6 Dormitorio de un departamento

Fuente: [Elaboración Propia]

Nro.	Cantidad.	Dispositivo	Descripción	Potencia(Wats)	Tiempo (horas)	Periodo(días)	Consumo Mes (Kwh)
1	1	Luminaria	ahorrador	20	7	30	4,2
2	1	Ventilador	Normal	100	4	30	12
Consumo total aproximado(Kwh)							16,2
Consumo total por mes en bolivianos							9,46
Costo total año							113,52

Tabla 4. 7 Cocina de un departamento

Fuente: [Elaboración Propia]

Nro.	Cantidad.	Dispositivo	Descripción	Potencia(Wats)	Tiempo (horas)	Periodo(días)	Consumo Mes (Kwh)
1	1	Luminaria	Ahorrador	20	4	30	2,4
2	1	Ventilador	Normal	100	2	30	6
Consumo total aproximado(Kwh)							8,4
Consumo total por mes en bolivianos							4,91
Costo total año							58,92

Tabla 4. 8 Pasillo de un departamento

Fuente: [Elaboración Propia]

Nro.	Cantidad	Dispositivo	Descripción	Potencia(Wats)	Tiempo (horas)	Periodo(días)	Consumo Mes (Kwh)
1	1	Luminaria	Ahorrador	20	8	30	4,8
2	0	Ventilador		0	0	30	0
Consumo total aproximado(Kwh)							4,8
Consumo total por mes en bolivianos							2,8
Costo total año							33,6

Escenario 3

En el siguiente escenario se recolectaron datos con características detalladas a continuación: Departamento con un Salón tabla 4.9 Dormitorio tabla 4.10 Cocina tabla 4.11 Pasillo tabla 4.12 con características de consumo elevado en Kwh.

Tabla 4. 9 Salón de un departamento

Fuente: [Elaboración Propia]

Nro.	Cantidad	Dispositivo	Descripción	Potencia(Wats)	Tiempo (horas)	Periodo(días)	Consumo Mes (Kwh)
1	2	Luminaria	Incandescente	100	6	30	36
2	1	Ventilador	Grande	200	3	30	18
Consumo total aproximado(Kwh)							54
Consumo total por mes en bolivianos							31,54
Costo total año							378,48

Tabla 4. 10 Dormitorio de un departamento

Fuente: [Elaboración Propia]

Nro.	Cantidad	Dispositivo	Descripción	Potencia(Wats)	Tiempo (horas)	Periodo(días)	Consumo Mes (Kwh)
1	1	Luminaria	Incandescente	100	8	30	24
2	1	Ventilador	Grande	200	1	30	6
Consumo total aproximado(Kwh)							30
Consumo total por mes en bolivianos							17,52
Costo total año							210,24

Tabla 4. 11 Cocina de un departamento

Fuente: [Elaboración Propia]

Nro.	Cantidad	Dispositivo	Descripción	Potencia(Wats)	Tiempo (horas)	Periodo(días)	Consumo Mes (Kwh)
1	1	Luminaria	Incandescente	100	5	30	15
2	1	Ventilador	Grande	200	1	30	6
Consumo total aproximado(Kwh)							21
Consumo total por mes en bolivianos							12,26
Costo total año							147,12

Tabla 4. 12 Pasillo de un departamento

Fuente: [Elaboración Propia]

Nro.	Cantidad	Dispositivo	Descripción	Potencia(Wats)	Tiempo (horas)	Periodo(días)	Consumo Mes (Kwh)
1	1	Luminaria	Incandescente	100	9	30	27
2	0	Ventilador		0	0	30	0
Consumo total aproximado(Kwh)							27
Consumo total por mes en bolivianos							15,77
Costo total año							189,24

Escenario 4

En el siguiente escenario se recolectaron datos con características detalladas a continuación: Departamento con un Salón tabla 4.13 Dormitorio tabla 4.14 Cocina tabla 4.15 Pasillo tabla 4.16 con características de consumo reducido en Kwh.

Tabla 4. 13 Sal3n de un departamento

Fuente: [Elaboraci3n Propia]

Nro.	Cantidad	Dispositivo	Descripci3n	Potencia(Wats)	Tiempo (horas)	Periodo(días)	Consumo Mes (Kwh)
1	2	Luminaria	Ahorrador	20	6	30	7,2
2	1	Ventilador	Normal	100	3	30	9
Consumo total aproximado(Kw/h)							16,2
Consumo total por mes en bolivianos							9,46
Costo total a3o							113,52

Tabla 4. 14 Dormitorio de un departamento

Fuente: [Elaboraci3n Propia]

Nro.	Cantidad	Dispositivo	Descripci3n	Potencia(Wats)	Tiempo (horas)	Periodo(días)	Consumo Mes (Kwh)
1	1	Luminaria	ahorrador	20	8	30	4,8
2	1	Ventilador	Normal	100	1	30	3
Consumo total aproximado(Kw/h)							7,8
Consumo total por mes en bolivianos							4,55
Costo total a3o							54,6

Tabla 4. 15 Cocina de un departamento

Fuente: [Elaboraci3n Propia]

Nro.	Cantidad	Dispositivo	Descripci3n	Potencia(Wats)	Tiempo (horas)	Periodo(días)	Consumo Mes (Kwh)
1	1	Luminaria	Ahorrador	20	5	30	3
2	1	Ventilador	Normal	100	1	30	3
Consumo total aproximado(Kwh)							6
Consumo total por mes en bolivianos							3,5
Costo total a3o							42

Tabla 4. 16 Pasillo de un departamento

Fuente: [Elaboración Propia]

Nro	Cantidad	Dispositivo	Descripción	Potencia(Wats)	Tiempo (horas)	Periodo(días)	Consumo Mes (Kwh)
1	1	Luminaria	Incandescente	20	9	30	5,4
2	0	Ventilador		0	0	30	0
Consumo total aproximado(Kwh)							5,4
Consumo total por mes en bolivianos							3,15
Costo total año							37,8

Recolección de Datos Con Domótica

Escenario 1

La recolección de datos se realizó en ambientes que se describen a continuación: Departamento con un Salón tabla 4.17 Dormitorio tabla 4.18 Cocina tabla 4.19 Pasillo tabla 4.20 con características de consumo elevado en Kwh.

Tabla 4. 17 Salón de un departamento

Fuente: [Elaboración Propia]

Nro.	Cantidad	Dispositivo	Descripción	Potencia(Wats)	Tiempo (horas)	Periodo(días)	Consumo Mes (Kwh)
1	2	Luminaria	Incandescente	100	3	30	18
2	1	Ventilador	Grande	200	5	30	30
Consumo total aproximado(Kwh)							48
Consumo total por mes en bolivianos							28,03
Costo total año							336,36

Tabla 4. 18 Dormitorio de un departamento

Fuente: [Elaboración Propia]

Nro.	Cantidad	Dispositivo	Descripción	Potencia(Wats)	Tiempo (horas)	Periodo(días)	Consumo Mes (Kwh)
1	1	Luminaria	Incandescente	100	4	30	12
2	1	Ventilador	Grande	200	5	30	30
Consumo total aproximado(Kwh)							42
Consumo total por mes en bolivianos							24,52
Costo total año							294,24

Tabla 4. 19 Cocina de un departamento

Fuente: [Elaboración Propia]

Nro.	Cantidad	Dispositivo	Descripción	Potencia(Wats)	Tiempo (horas)	Periodo(días)	Consumo Mes (Kwh)
1	1	Luminaria	Incandescente	100	3	30	9
2	1	Ventilador	Grande	200	3	30	18
Consumo total aproximado(Kwh)							27
Consumo total por mes en bolivianos							15,77
Costo total año							189,24

Tabla 4. 20 Pasillo de un departamento

Fuente: [Elaboración Propia]

Nro.	Cantidad	Dispositivo	Descripción	Potencia(Wats)	Tiempo (horas)	Periodo(días)	Consumo Mes (Kwh)
1	1	Luminaria	Incandescente	100	3	30	9
2	0	Ventilador		0	0	30	0
Consumo total aproximado(Kwh)							9
Consumo total por mes en bolivianos							5,26
Costo total año							63,12

Escenario 2

En el siguiente escenario se recolectaron datos con características detalladas a continuación: Departamento con un Salón tabla 4.21 Dormitorio tabla 4.22 Cocina tabla 4.23 Pasillo tabla 4.24 con características de consumo reducido en Kwh.

Tabla 4. 21 Salón de un departamento

Fuente: [Elaboración Propia]

Nro.	Cantidad	Dispositivo	Descripción	Potencia(Wats)	Tiempo (horas)	Periodo(días)	Consumo Mes (Kwh)
1	2	Luminaria	Ahorrador	20	3	30	3,6
2	1	Ventilador	Normal	100	5	30	15
Consumo total aproximado(Kwh)							18,6
Consumo total por mes en bolivianos							10,86
Costo total año							130,32

Tabla 4. 22 Dormitorio de un departamento

Fuente: [Elaboración Propia]

Nro.	Cantidad	Dispositivo	Descripción	Potencia(Wats)	Tiempo (horas)	Periodo(días)	Consumo Mes (Kwh)
1	1	Luminaria	Ahorrador	20	4	30	2,4
2	1	Ventilador	Normal	100	5	30	15
Consumo total aproximado(Kwh)							17,4
Consumo total por mes en bolivianos							10,16
Costo total año							121,92

Tabla 4. 23 Cocina de un departamento

Fuente: [Elaboración Propia]

Nro.	Cantidad	Dispositivo	Descripción	Potencia(Wats)	Tiempo (horas)	Periodo(días)	Consumo Mes (Kwh)
1	1	Luminaria	Ahorrador	20	3	30	1,8
2	1	Ventilador	Normal	100	3	30	9
Consumo total aproximado(Kwh)							10,8
Consumo total por mes en bolivianos							6,31
Costo total año							75,72

Tabla 4. 24 Pasillo de un departamento

Fuente: [Elaboración Propia]

Nro.	Cantidad	Dispositivo	Descripción	Potencia(Wats)	Tiempo (horas)	Periodo(días)	Consumo Mes (Kwh)
1	1	Luminaria	Ahorrador	20	3	30	1,8
2	0	Ventilador		0	0	30	0
Consumo total aproximado(Kwh)							1,8
Consumo total por mes en bolivianos							1,05
Costo total año							12,6

Escenario 3

En el siguiente escenario se recolectaron datos con características detalladas a continuación: Departamento con un Salón tabla 4.25 Dormitorio tabla 4.26 Cocina tabla 4.27 Pasillo tabla 4.28 con características de consumo elevado en Kwh.

Tabla 4. 25 Salón de un departamento

Fuente: [Elaboración Propia]

Nro.	Cantidad	Dispositivo	Descripción	Potencia(Wats)	Tiempo (horas)	Periodo(días)	Consumo Mes (Kwh)
1	2	Luminaria	Incandescente	100	3	30	18
2	1	Ventilador	Grande	200	2	30	12
Consumo total aproximado(Kwh)							30
Consumo total por mes en bolivianos							17,52
Costo total año							210,24

Tabla 4. 26 Dormitorio de un departamento

Fuente: [Elaboración Propia]

Nro.	Cantidad	Dispositivo	Descripción	Potencia(Wats)	Tiempo (horas)	Periodo(días)	Consumo Mes (Kwh)
1	1	Luminaria	Incandescente	100	6	30	18
2	1	Ventilador	Grande	200	1	30	6
Consumo total aproximado(Kwh)							24
Consumo total por mes en bolivianos							14,02
Costo total año							168,24

Tabla 4. 27 Cocina de un departamento

Fuente: [Elaboración Propia]

Nro.	Cantidad	Dispositivo	Descripción	Potencia(Wats)	Tiempo (horas)	Periodo(días)	Consumo Mes (Kwh)
1	1	Luminaria	Incandescente	100	4	30	12
2	1	Ventilador	Grande	200	1	30	6
Consumo total aproximado(Kwh)							18
Consumo total por mes en bolivianos							10,51
Costo total año							126,12

Tabla 4. 28 Pasillo de un departamento

Fuente: [Elaboración Propia]

Nro.	Cantidad	Dispositivo	Descripción	Potencia(Wats)	Tiempo (horas)	Periodo(días)	Consumo Mes (Kwh)
1	1	Luminaria	Incandescente	100	6	30	18
2	0	Ventilador		0	0	30	0
Consumo total aproximado(Kwh)							18
Consumo total por mes en bolivianos							10,51
Costo total año							126,12

Escenario 4

En el siguiente escenario se recolectaron datos con características detalladas a continuación: Departamento con un Salón tabla 4.29 Dormitorio tabla 4.30 Cocina tabla 4.31 Pasillo tabla 4.32 con características de consumo ahorrador en Kwh.

Tabla 4. 29 Salón de un departamento

Fuente: [Elaboración Propia]

Nro.	Cantidad	Dispositivo	Descripción	Potencia(Wats)	Tiempo (horas)	Periodo(días)	Consumo Mes (Kwh)
1	2	Luminaria	Ahorrador	20	3	30	3,6
2	1	Ventilador	Normal	100	2	30	6
Consumo total aproximado(Kwh)							9,6
Consumo total por mes en bolivianos							5,61
Costo total año							67,32

Tabla 4. 30 Dormitorio de un departamento

Fuente: [Elaboración Propia]

Nro.	Cantidad	Dispositivo	Descripción	Potencia(Wats)	Tiempo (horas)	Periodo(días)	Consumo Mes (Kw/h)
1	1	Luminaria	Ahorrador	20	6	30	3,6
2	1	Ventilador	Normal	100	1	30	3
Consumo total aproximado(Kw/h)							6,6
Consumo total por mes en bolivianos							3,85
Costo total año							46,2

Tabla 4. 31 Cocina de un departamento

Fuente: [Elaboración Propia]

Nro.	Cantidad	Dispositivo	Descripción	Potencia(Wats)	Tiempo (horas)	Periodo(días)	Consumo Mes (Kwh)
1	1	Luminaria	Ahorrador	20	4	30	2,4
2	1	Ventilador	Normal	100	1	30	3
Consumo total aproximado(Kw/h)							5,4
Consumo total por mes en bolivianos							3,15
Costo total año							37,8

Tabla 4. 32 Pasillo de un departamento

Fuente: [Elaboración Propia]

Nro.	Cantidad	Dispositivo	Descripción	Potencia(Wats)	Tiempo (horas)	Periodo(días)	Consumo Mes (Kwh)
1	1	Luminaria	Ahorrador	20	6	30	3,6
2	0	Ventilador		0	0	30	0
Consumo total aproximado(Kwh)							3,6
Consumo total por mes en bolivianos							2,1
Costo total año							25,2

Análisis de los datos obtenidos de los escenarios. A continuación se presenta la siguiente Tabla 4.33. Donde se experimentaron con 16 ambientes sin domótica con un consumo promedio de 397,8 Kwh por mes que representa un gasto en bolivianos de 232,31 y al año con 2787,72 bolivianos aproximadamente.

Tabla 4. 33 Resultados sin domótica

Fuente: [Elaboración Propia]

Nro.	Consumo Mes Kwh Aproximado	Consumo Total por mes Bolivianos	Consumo total por Año
1	78	45,55	546,6
2	45	26,28	315,36
3	24	14,02	168,24
4	24	14,02	168,24
5	30	17,52	210,24
6	16,2	9,46	113,52
7	8,4	4,91	58,92
8	4,8	2,8	33,6
9	54	31,54	378,48
10	30	17,52	210,24
11	21	12,26	147,12
12	27	15,77	189,24
13	16,2	9,46	113,52
14	7,8	4,55	54,6
15	6	3,5	42
16	5,4	3,15	37,8
Total	397,8	232,31	2787,72

Análisis de los datos obtenidos de los escenarios con la instalación del sistema domotico. A continuación se presenta la siguiente Tabla 4.34 Donde se experimentaron con 16 ambientes con domótica con un consumo promedio de 289,2 Kwh por mes que representa un gasto en bolivianos de 169,23 y al año con 2030,76 bolivianos aproximadamente.

Tabla 4. 34 Resultados con Domótica

Fuente: [Elaboración Propia]

Nro.	Consumo Mes Kwh Aproximado	Consumo Total por mes Bolivianos	Consumo total por Año
1	48	28,03	336,36
2	42	24,52	294,24
3	27	15,77	189,24
4	9	5,26	63,12
5	18	10,86	130,32
6	17,4	10,16	121,92
7	10,8	6,31	75,72
8	1,8	1,05	12,6
9	30	17,52	210,24
10	24	14,02	168,24
11	18	10,51	126,12
12	18	10,51	126,12
13	9,6	5,61	67,32
14	6,6	3,85	46,2
15	5,4	3,15	37,8
16	3,6	2,1	25,2
Total	289,2	169,23	2030,76

De las tablas 4.33 y 4.34 Se obtienen los siguientes resultados donde se muestra que el ahorro aproximado por año es hasta en un 27%. El cálculo para determinar el ahorro en porcentaje por año es Consumo aproximado sin Domótica por cien todo esto en proporcional al consumo sin domótica tabla 4.35

Tabla 4. 35 Comparación con Domótica y sin Domótica

Fuente: [Elaboración Propia]

	Sin Domótica	Con Domótica	Ahorro por Año	Ahorro en porcentaje
Costo Total por Año Aproximado	2787,72	2030,76	756,96	27,15%

4.3. Demostración de la Hipótesis

La hipótesis planteada tiene la siguiente formulación:

La implementación de un prototipo de control domótico construido con una placa Arduino y software libre reducirá el consumo de energía eléctrica y el costo del controlador respecto a aplicaciones comerciales.

Para la demostración de la hipótesis se realiza un estadístico de prueba con un margen de error del 5% para demostrar la validez del enunciado.

Prueba de hipótesis siendo \hat{P} y \hat{Q} la proporción de ambientes con domótica y sin domótica.

$$\hat{P} = 0,5 \text{ con domotica}$$

$$\hat{Q} = 0,5 \text{ sin domotica}$$

La hipótesis planteada

$H_0: P = 0,27$ se rechaza la afirmación

$H_1: P > 0,27$ se acepta la afirmación

La siguiente fórmula nos permite realizar nuestro estadístico de prueba con la cual se comprueba la hipótesis planteada.

$$Z_c = \frac{\hat{p} - P_0}{\sqrt{\frac{P_0 * Q_0}{n}}}$$

Fórmula para determinar el estadístico de prueba

$$Z_c = \frac{0,5 - 0,27}{\sqrt{\frac{0,5 * 0,5}{32}}} = 2,60$$

Margen de Error de 5% nos permite realizar un contraste con la tabla de la distribución normal.

$$\alpha = 0,05$$

$$1 - \alpha = 0,95$$

$$Z_{(0,95)} = 1,64$$

$$2,60 > 1,64$$

Bajo esta afirmación la hipótesis que se plantea como nula se rechaza permitiendo aceptar la hipótesis alterna que se enuncia de la siguiente manera reducción en consumo hasta en un 27% en una gestión.

4.4. Costo del Controlador Domótico

El costo del sistema de control domótico frente al costo de sistemas domóticos ofertados en el mercado.

El costo del sistema de control domótico implementado con software libre tiene un costo aproximado de 100 dólares esto a nivel prototipo y experimental, el costo real de un sistema domótico implementado con software libre tiene un costo aproximado de 500 dólares como mínimo en función a las habitaciones, que es básicamente, todo el cableado de un departamento, su base está en un controlador Arduino y su respectivo módulo Shield Ethernet que permite el control mediante la web.

El costo de un sistema de control domótico de este tipo en nuestro medio está en un promedio de 3000 dólares con todas las funciones pero bajo los siguientes parámetros éstos sistemas responden a una cantidad establecida de ambientes los sensores son de importación posibles daños son remitidos a la fábrica, si bien el costo del software está incluido en el precio el mantenimiento se realiza periódicamente bajo un costo establecido por licencia de funcionamiento del sistema de control domótico.

5. CONCLUSIONES Y RECOMENDACIONES

5.1. Introducción

En presente capítulo detalla las conclusiones del proyecto de investigación cumplimiento de objetivos tanto específicos como también generales. En función al objetivo general se realiza la pruebas experimentales al prototipo generando datos que nos permite tomar decisiones en cuanto a la implementación del sistema de control domótico, obtenida esta información experimental al sistema domótico generando la solución de los problemas que se tiene en el campo de la domótica en nuestro medio, posteriormente se realiza un contraste de hipótesis generando la validez o no del sistema de control domótico para luego recomendar trabajos posteriores.

5.2. Cumplimiento de objetivos

El objetivo planteado es el siguiente:

“Diseñar un prototipo que permita automatizar tareas rutinarias como la iluminación y la temperatura de un ambiente, el control directo sobre los sensores y actuadores mediante páginas web como parte del control y la automatización para lograr la reducción del consumo de la energía eléctrica”. Planteado en el capítulo uno se cumplió con la construcción del prototipo y la respectiva funcionalidad de la placa electrónica en función a la lógica difusa para la automatización y también el análisis de resultados permite afirmar la reducción de consumo de energía eléctrica.

Los objetivos específicos planteados en el trabajo de investigación, se describe a continuación conjuntamente el grado de cumplimiento de cada uno de ellos.

- Se desarrolló el software controlador del prototipo permitiendo el control de dispositivos como los sensores y actuadores.

- Se realizó un análisis de circuitos permitiendo la comunicación directa en el momento de programación esto se logra a través del puerto USB.
- Se realizó el correspondiente análisis del módulo shield Ethernet.
- Se desarrolló las páginas web necesarias para su navegación correspondiente.
- Se realizaron pruebas de instalación del prototipo, presentados sobre ello una maqueta que implementa los dispositivos domóticos.

5.3. Solución de Problemas

- El presente trabajo de investigación muestra que la plataforma Arduino permite automatizar tareas de rutina control de dispositivos conectados a una vivienda.
- El costo de los dispositivos es accesible para este tipo de proyectos.
- La plataforma Arduino es amplia, en cuanto a la programación brinda mayores facilidades al proveer varias librerías de aplicación, ejemplos de código permitiendo realizar desde proyectos sencillos hasta proyectos complejos sobre todo en la automatización de procesos de control domótico.

5.4. Estado de la Hipótesis

Para demostrar la hipótesis se obtuvo varios resultados generados mediante la implementación del sistema de control domótico, donde se pudo observar que el promedio de consumo de energía eléctrica es de menos 27% anual frente al consumo total de un hogar sin un sistema domótico.

La demostración se realizó con satisfacción por la reducción de consumo de energía eléctrica en diferentes escenarios descritos en el capítulo anterior, evidenciando que la implementación de un sistema de control domótico permite el ahorro de consumo de energía eléctrica y un ahorro de costo, siendo este sistema accesible frente a sistemas ofertados en el mercado.

5.5. Recomendaciones

Trabajar con placas electrónicas que existen en el mercado con mayor cantidad de salidas digitales como también analógicas permitiendo automatizar más tareas recomendable usar el Arduino mega que es el más aceptable para este tipo de proyectos y también para la navegación web contar con dispositivos como un router para la conexión remota esto en cuanto a hardware.

Es importante en este tipo de proyectos considerar que se trabaja con tensiones de 5 voltios esto a nivel prototipo de maqueta para la implementación del sistema en una vivienda por lo tanto no es recomendable para sistemas domóticos con tensiones de 220 voltios se recomienda el uso de reles y otros dispositivos de amplitud en función al voltaje para ese tipo de conexiones importante realizar una lectura previa a la ficha técnica de cada dispositivo.

5.6. Trabajos Futuros

En cuanto a trabajos posteriores la aplicación en teléfonos móviles para el control manual y automático bajo plataformas android integrara la domótica con Arduino y Ethernet pero controlada con este dispositivo móvil.

La automatización de más dispositivos conectados en una vivienda como multimedia que ya se realizó esta investigación pero no en el campo de Arduino ni mucho menos con el shield Ethernet control de persianas, puertas, cámaras web para el acceso en cuanto a seguridad, monitorizar el consumo de energía eléctrica realizando una aplicación móvil.

REFERENCIAS BIBLIOGRAFICAS

[José Manuel Huidobro Moya, Ramón J Millar] 2010, Edificios Inteligentes, **Recuperado:** <http://www.fenercom.com/pdf/publicaciones/la-domotica-como-solucion-de-futuro-fenercom.pdf>.

[Carlos López Jimeno] 2007, La domótica como una solución al futuro **Recuperado:**

[Colegio de Ingenieros Especialistas de Córdoba] 2012, Elaboración de un proyecto domótico **Recuperado:** <http://www.ciec.com.ar/images/archivos/Domotica-CIEC.pdf>

[Flores Arriaga Motserrat, Sánchez Alvares Carlos Alberto] 2012, Instalación de un prototipo de arquitectura domótica mixta usando comunicación ZibBee. **Recuperado:** http://postgrado.info.unlp.edu.ar/Carreras/Especializaciones/Redes_y_Seguridad

[Javier Martínez Panta] 2012, Control domótico por voz **Recuperado:** Tesis Universidad Politécnica de Valencia. Escuela Técnica Superior de Ingeniería Informática.

[Valentina Botero, Diana Marcela Londoño] 2003, **Recuperado:** Centro de Investigaciones y desarrollo, Facultad de Ingeniería Universidad de Manizales.

[Emilio Lledó Sánchez] 2012, Diseño de un control domótico basado en Arduino **Recuperado:** Universidad Técnica de Valencia.

[Felipe Monasterios Martin, Reyes Poo Arguelles] 2010, Automatización Integral de Edificios **Recuperado:** <http://isa.uniovi.es/docencia/AutomEdificios>.

[Arduino] 2010, Arduino Página Oficial **Recuperado:** <http://arduino.cc>

[Joskowicz] 2008, Reglas y Practicas en eXtreme Programing. **Recuperado:** <http://iie.fing.edu.uy/~josej/docs/XP%20-%20Jose%20Joskowicz.pdf>

[Eduardo Andia Gutierrez] 2001, Control Difuso para un sistema de aire acondicionado
Recuperado: “Control Difuso para un sistema de aire acondicionado”

Tesis Carrera de Informática U.M.S.A.

[Cristóbal Romero Morales, Francisco Vásquez Serrano, Carlos de Castro Lozano] 2005,
Domótica e Inmótica **Recuperado:** <http://www.nebrija.es/~jmaestro/ATA018/Domotica.pdf>.

[Electricidad] 2012, Autoridad de Fiscalización y control Social de Electricidad
Recuperado: <http://www.ae.gob.bo/node/112>.

[Chip w5100] 2010, wiznet **Recuperado:** <http://www.wiznet.co.kr/>

[Hugo Martín Domínguez, Fernando Sáez Vacas] 2006, Domótica: Un enfoque sociotécnico
Recuperado: http://www.gsi.dit.upm.es/~fsaez/intl/libro_domotica.pdf

[Viessmann] 2014, Tecnologías de calefacción de alta eficiencia energética **Recuperado:**
www.viessmann.es.

[Departamento Técnico de ANFALUM] 2010, Eficiencia energética y garantías de seguridad
de los servicios eléctricos generales en comunidades de propietarios **Recuperado:**
www.anfalum.com

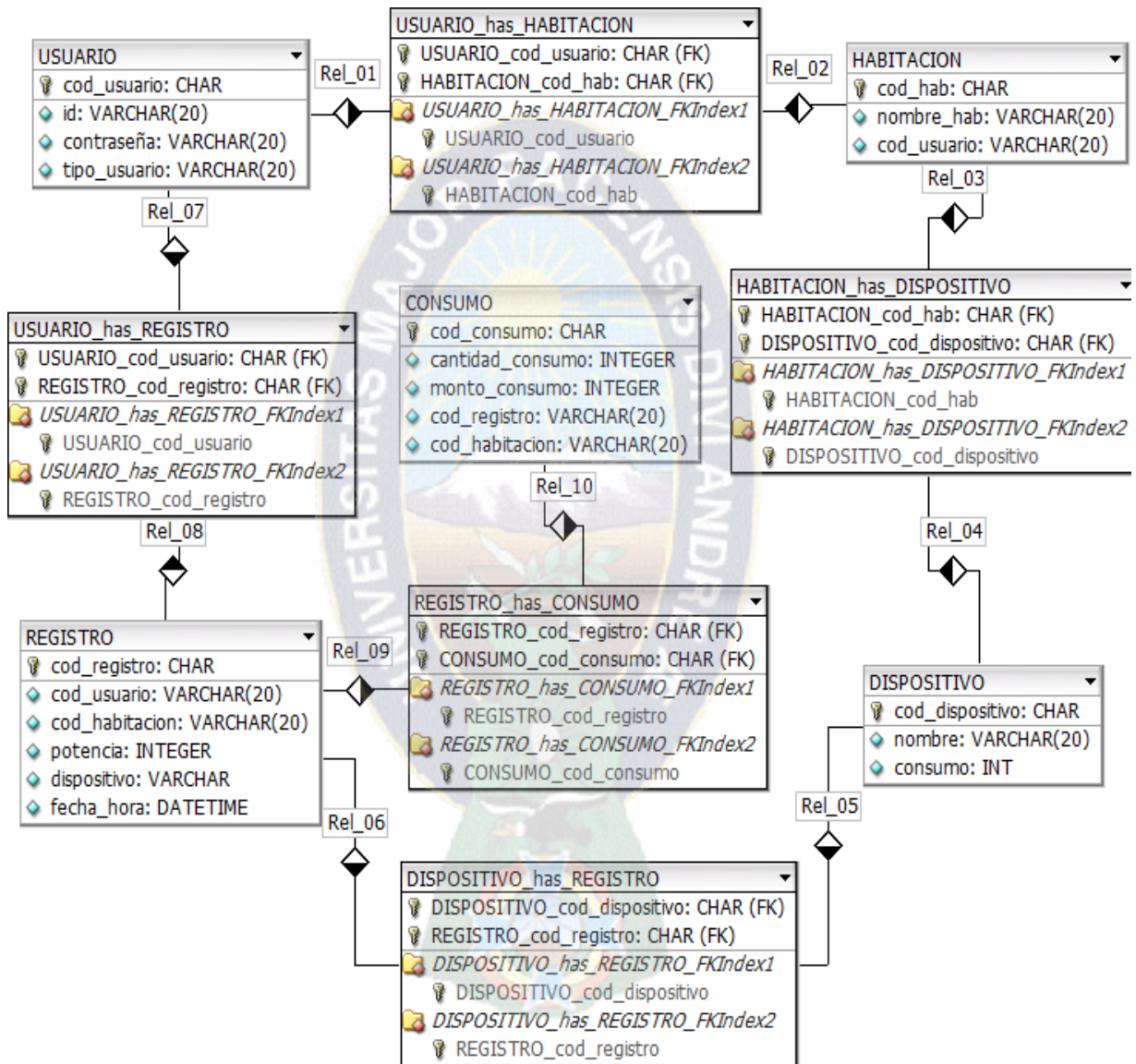
[Endesa] 2012, Medidas para la eficiencia energética **Recuperado:** www.endesa.es

[Fenercom] 2012, Guía sobre Ahorro y Eficiencia energética en Comunidades **Recuperado:**
www.fenercom.com.

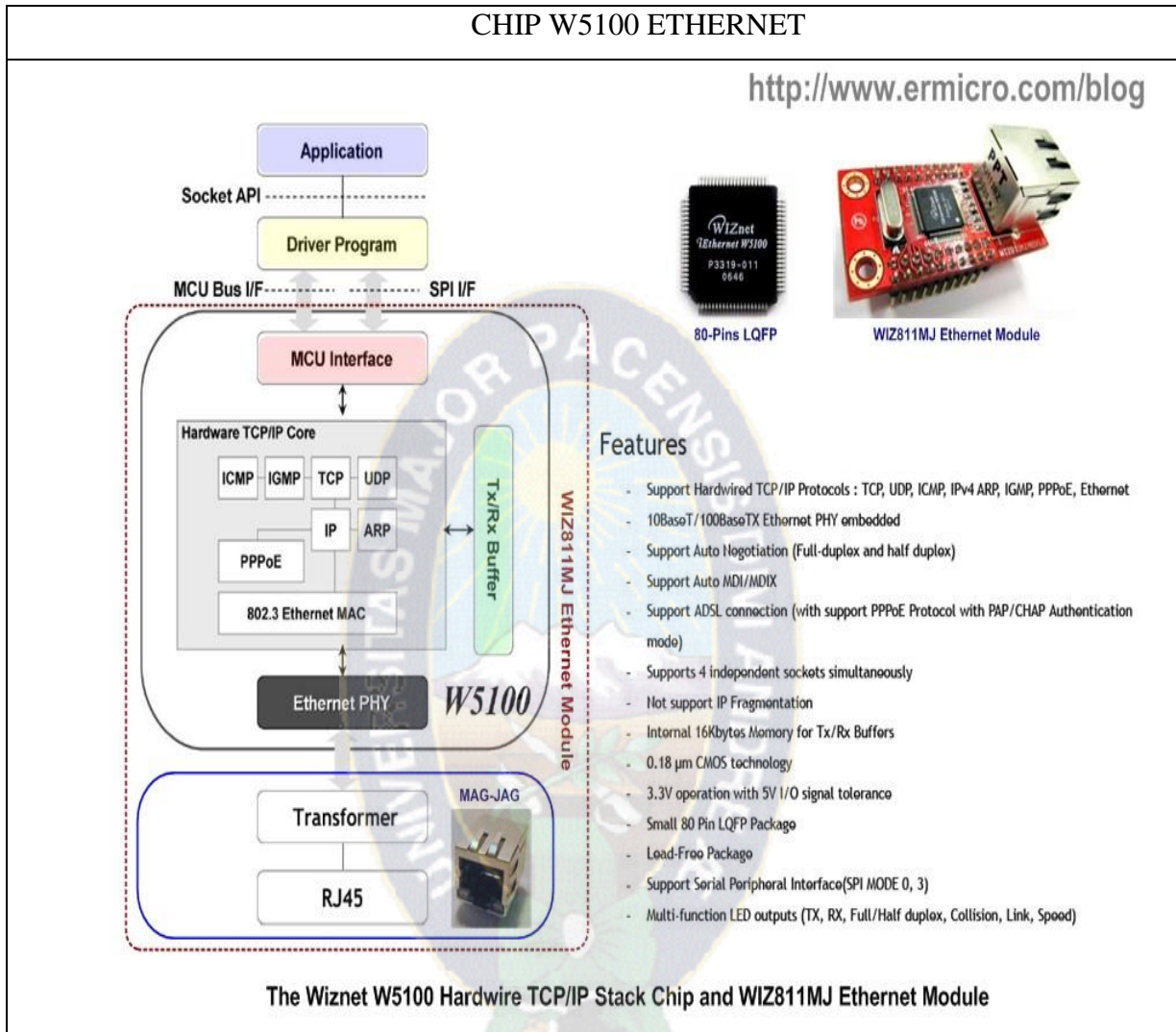
[Biblioteca de Ingeniería de la Universidad de Sevilla] 2010, Diseño de un Sistema
Domótico Para la adecuación de una vivienda unifamiliar para personas discapacitadas
Recuperado: <http://bibing.us.es/proyectos>.

Anexo
Anexo A

MODELO ENTIDAD RELACION



Anexo B



Anexo C

CODIGO ARDUINO

```

/*
#####
#####
# UNIVERSIDAD MAYOR DE SAN ANDRES #
# FACULTAD DE CIENCIAS PURAS Y NATURALES #
# CARRERA DE INFORMATICA #
# DISEÑO E IMPLEMENTACION DE UN SISTEMA DOMOTICO PARA EL
CONTROL DE ENERGIA#
# AUTOR: UNIV. REYNALDO CORONEL LAZO#
#####

```

```

#####
*/
/* CONFIGURACION BASICA ETHERNET */
#include <SPI.h>
#include <Ethernet.h>
#include <SD.h>
#define REQ_BUF_SZ 60
byte mac[] = { 0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED };
IPAddress ip(169, 254, 210, 242);
EthernetServer server(80);
File webFile;
char HTTP_req[REQ_BUF_SZ] = {0};
char req_index = 0;
/*INICIALIZACION DE VARIABLES */
boolean LED_state[5] = {0};
//boolean LED_stateA[2] = {0};
/* CONFIGURACION DE ENTRADAS Y SALIDAS */
void setup()
{
    pinMode(10, OUTPUT);
    digitalWrite(10, HIGH);

    Serial.begin(9600);
    if (!SD.begin(4)) {
        return;
    }
    if (!SD.exists("menu.htm")) {
        return;
    }
    // SALIDAS
    pinMode(2, OUTPUT);
    pinMode(3, OUTPUT);
    pinMode(5, OUTPUT);
    pinMode(6, OUTPUT);
    pinMode(7, OUTPUT);
    Ethernet.begin(mac, ip);
    server.begin();
}
/* CUERPO PRINCIPAL DEL SISTEMA */
void loop()
{
    EthernetClient client = server.available();

    if (client) {
        boolean currentLineIsBlank = true;
        while (client.connected()) {
            if (client.available()) {
                char c = client.read();
                if (req_index < (REQ_BUF_SZ - 1)) {
                    HTTP_req[req_index] = c;
                    req_index++;
                }
            }
        }
    }
}

```

```

    }
    if (c == '\n' && currentLineIsBlank) {
        client.println("HTTP/1.1 200 OK");
        if (StrContains(HTTP_req, "ajax_inputs"))
        {
            client.println("Content-Type:
text/xml");
            client.println("Connection:      keep-
alive");

            client.println();
            SetLEDs();
            XML_response(client);
            StrClear(HTTP_req, REQ_BUF_SZ);
        }
        else if (StrContains(HTTP_req, "GET / "
|| StrContains(HTTP_req, "GET
/menu.htm")) {
            client.println("HTTP/1.1 200 OK");
            client.println("Content-Type:
text/html");
            client.println("Connection: close");
            client.println();
            webFile = SD.open("menu.htm");
            StrClear(HTTP_req, REQ_BUF_SZ);
        }
        else if (StrContains(HTTP_req, "GET
/manual.htm")) {
            client.println("HTTP/1.1 200 OK");
            client.println("Content-Type:
text/html");
            client.println("Connection: close");
            client.println();
            webFile = SD.open("manual.htm");
            StrClear(HTTP_req, REQ_BUF_SZ);
        }
        else if (StrContains(HTTP_req, "GET
/auto.htm")) {
            client.println("HTTP/1.1 200 OK");
            client.println("Content-Type:
text/html");
            client.println("Connection: close");
            client.println();
            webFile = SD.open("auto.htm");
            StrClear(HTTP_req, REQ_BUF_SZ);
        }
        else if (StrContains(HTTP_req, "GET
/domotica.jpg")) {
            webFile = SD.open("domotica.jpg");
            if (webFile) {

```

```

        client.println("HTTP/1.1 200 OK");
        client.println();
        StrClear(HTTP_req, REQ_BUF_SZ);
    }
}
if (webFile) {
    while(webFile.available()) {
        client.write(webFile.read());
    }
    webFile.close();
}
req_index = 0;
StrClear(HTTP_req, REQ_BUF_SZ);
break;
}
if (c == '\n') {
    currentLineIsBlank = true;
}
else if (c != '\r') {
    currentLineIsBlank = false;
}
}
}
delay(1);
client.stop();
}
}
void SetLEDs(void)
{
    // SALON (pin 2)
    if (StrContains(HTTP_req, "LED1=1")) {
        LED_state[0] = 1;
        digitalWrite(2, HIGH);
    }
    else if (StrContains(HTTP_req, "LED1=0")) {
        LED_state[0] = 0;
        digitalWrite(2, LOW);
    }
    // DORMITORIO (pin 3)
    if (StrContains(HTTP_req, "LED2=1")) {
        LED_state[1] = 1;
        digitalWrite(3, HIGH);
    }
    else if (StrContains(HTTP_req, "LED2=0")) {
        LED_state[1] = 0;
        digitalWrite(3, LOW);
    }
    // COCINA (pin 5)
    if (StrContains(HTTP_req, "LED3=1")) {
        LED_state[2] = 1;
        digitalWrite(5, HIGH);
    }
}

```

```

    }
    else if (StrContains(HTTP_req, "LED3=0")) {
        LED_state[2] = 0;
        digitalWrite(5, LOW);
    }
    // PASILLO (pin 6)
    if (StrContains(HTTP_req, "LED4=1")) {
        LED_state[3] = 1;
        digitalWrite(6, HIGH);
    }
    else if (StrContains(HTTP_req, "LED4=0")) {
        LED_state[3] = 0;
        digitalWrite(6, LOW);
    }
    // VENTILADOR (pin 7)
    if (StrContains(HTTP_req, "LED5=1")) {
        LED_state[4] = 1;
        digitalWrite(7, HIGH);
    }
    else if (StrContains(HTTP_req, "LED5=0")) {
        LED_state[4] = 0; // save LED state
        digitalWrite(7, LOW);
    }
}
void XML_response(EthernetClient cl)
{
    int analog_val;
    int count = 2;
    cl.print("<?xml version = \"1.0\" ?>");
    cl.print("<inputs>");
    while (count < 4)
    {
        analog_val = analogRead(count);
        if(count % 2 == 0){
            analog_val = 5*analog_val*100/1024;
            if(analog_val > 25)
                digitalWrite(7, HIGH);
            else
                digitalWrite(7, LOW);
            cl.print("<analog>");
            cl.print(analog_val);
            cl.println("</analog>");
        }else{
            if(analog_val < 150)
                digitalWrite(9, HIGH);
            else
                digitalWrite(9, LOW);
            cl.print("<analog>");
            cl.print(analog_val);
            cl.println("</analog>");
        }
    }
}

```

```

        count ++;
    }
    // SALON
    cl.print("<LED>");
    if (LED_state[0]) {
        cl.print("on");
    }
    else {
        cl.print("off");
    }
    cl.println("</LED>");
    // DORMITORIO
    cl.print("<LED>");
    if (LED_state[1]) {
        cl.print("on");
    }
    else {
        cl.print("off");
    }
    cl.println("</LED>");
    // COCINA
    cl.print("<LED>");
    if (LED_state[2]) {
        cl.print("on");
    }
    else {
        cl.print("off");
    }
    cl.println("</LED>");
    // PASILLO
    cl.print("<LED>");
    if (LED_state[3]) {
        cl.print("on");
    }
    else {
        cl.print("off");
    }
    cl.println("</LED>");
    // VENTILADOR
    cl.print("<LED>");
    if (LED_state[4]) {
        cl.print("on");
    }
    else {
        cl.print("off");
    }
    cl.println("</LED>");
    cl.print("</inputs>");
}
void StrClear(char *str, char length)
{

```



```
        for (int i = 0; i < length; i++) {
            str[i] = 0;
        }
    }
    char StrContains(char *str, char *sfind)
    {
        char found = 0;
        char index = 0;
        char len;
        len = strlen(str);
        if (strlen(sfind) > len) {
            return 0;
        }
        while (index < len) {
            if (str[index] == sfind[found]) {
                found++;
                if (strlen(sfind) == found) {
                    return 1;
                }
            }
            else {
                found = 0;
            }
            index++;
        }
        return 0;
    }
}
```