

UNIVERSIDAD MAYOR DE SAN ANDRES
FACULTAD DE TECNOLOGIA
CARRERA: ELECTRONICA Y TELECOMUNICACIONES



PROYECTO DE GRADO

NIVEL LICENCIATURA

**“SISTEMA DOMOTICO IMPLEMENTADO EN MICROSERVIDOR
WEB EMBEBIDO CON CONEXIÓN INALAMBRICA PARA
SMARTPHONES”**

Postulante: DAVID QUISPE APAZA

Tutor: Ing. ARTURO MARIN THAMES

La Paz- Bolivia

2015

DEDICATORIA

Quiero dedicar este proyecto a mis padres a quienes les debo toda mi formación personal en mi vida, por el apoyo y consejos que siempre me dieron durante los estudios de mi carrera universitaria para poder llegar a esta instancia y así realizar este proyecto.

AGRADECIMIENTOS

Mi sincero agradecimiento a todos los Catedráticos de mi querida Facultad quienes son gestores de aprendizaje. En especial estoy muy agradecido al Ing. Arturo Marin Thames, por sus valiosos consejos técnicos y humanos, por su inestimable asesoramiento en los aspectos técnicos de este proyecto y por la confianza depositada en mí.

RESUMEN

En el presente proyecto de grado titulado “SISTEMA DOMOTICO IMPLEMENTADO EN MICROSERVIDOR WEB EMBEBIDO CON CONEXIÓN INALAMBRICA PARA SMARTPHONES” presenta un contenido donde se desarrollan todas las bases teóricas y prácticas para el diseño e implementación del mismo.

En el primer capítulo se plantea la problemática en la cual está basada el proyecto además se mencionan los objetivos y las respectivas justificaciones para cada caso.

En el segundo capítulo se hace una breve descripción de los conocimientos teóricos fundamentales involucrados en la domótica, las características propias de estos sistemas, las múltiples aplicaciones que ofrecen como también los pasos para el diseño de un sistema autónomo.

En el tercer capítulo se describe el concepto de Microservidor Web y todos los elementos de red que están involucrados en su implementación, revisando los protocolos que utilizan y las interfaces físicas que emplean.

En el cuarto capítulo se da a conocer una descripción de la tecnología Bluetooth y su interfaz empleada en el proyecto. También se describe al sistema operativo Android y los softwares empleados para el diseño de aplicaciones.

El quinto capítulo muestra el desarrollo de la ingeniería de proyecto en la cual se presentan todas las consideraciones de diseño, mostrando de forma detallada cada subsistema que presenta el prototipo propuesto, el tratamiento del tráfico de datos que realizan de forma física con los circuitos y lógica en los Firmwares, empleados en los accesos remoto y local. También se explicara el diseño del servidor web embebido en el microcontrolador, y la aplicación para los dispositivos Android.

El sexto capítulo está orientado al respectivo análisis de costos y presupuestos en donde se incluyen tablas de referencia de costos directos e indirectos.

El séptimo capítulo describe las conclusiones y recomendaciones del diseño y la implementación del proyecto así como algunos comentarios útiles a la hora de probar el prototipo implementado, además se incluyen la bibliografía y los datos anexos para una mejor comprensión.

INDICE

	Pagina
DEDICATORIA.....	i
AGRADECIMIENTO.....	ii
RESUMEN.....	iii
ÍNDICE.....	v
ÍNDICE DE FIGURAS.....	x
ÍNDICE DE TABLAS.....	xiii

CAPITULO I

1. INTRODUCCION.....	1
1.1. PLANTEAMIENTO DEL PROBLEMA.....	3
1.2. OBJETIVOS DEL PROYECTO.....	4
1.2.1. Objetivo general.....	4
1.2.2. Objetivos Especificos.....	4
1.3. JUSTIFICACION.....	5
1.3.1. Justificación Académica.....	5
1.3.2. Justificación Tecnológica.....	5
1.3.3. Justificación Social.....	5
1.4. DELIMITACIÓN.....	6
1.4.1. Delimitación temática.....	6
1.4.2. Delimitación temporal.....	6
1.4.3. Delimitación espacial.....	6
1.5. METODOS DE INVESTIGACION.....	7

CAPITULO II

2. MARCO TEORICO: DOMOTICA.....	8
2.1. DEFINICION DE DOMOTICA.....	8
2.2. LA DOMOTICA Y EL MODELOS DE TRES NIVELES DE COMPLEJIDAD.....	8
2.3. PRIMER NIVEL: DISPOSITIVOS DOMOTICOS.....	9
2.3.1. ACTUADORES.....	10
2.3.2. SENSORES.....	11

2.4. SEGUNDO NIVEL: INTERCONEXION DE EQUIPOS.....	12
2.4.1. RED DE CONTROL.....	13
2.4.2. PASARELA RESIDENCIAL.....	14
2.5. TERCER NIVEL: APLICACIONES Y SERVICIOS.....	15
2.5.1. SEGURIDAD.....	16
2.5.2. AHORRO ENERGETICO.....	16
2.5.3. CONFORT.....	17

CAPITULO III

3. MARCO TEORICO: MICROSERVIDOR WEB EMBEBIDO.....	19
3.1. ETHERNET.....	19
3.1.1. FORMATO DE LA TRAMA 802.3.....	19
3.1.2. TECNOLOGIAS ETHERNET.....	20
3.2. MODELO DE REFERENCIA OSI.....	21
3.3. MODELO TCP/IP.....	23
3.4. REDES LAN.....	24
3.5. REDES WAN.....	25
3.6. SERVIDOR WEB.....	26
3.6.1. PROTOCOLO HTTP.....	26
3.6.2. PETICION GET.....	27
3.6.3. PETICION POST.....	28
3.7. PAGINAS WEB DINAMICAS EN HTML.....	28
3.7.1. HTML.....	28
3.7.2. JAVASCRIPT.....	29
3.7.1. HTML.....	28
3.8. SISTEMAS EMBEBIDOS.....	29
3.9. MICROSERVIDOR WEB.....	30
3.9.1. ETHERNET EMBEBIDO.....	30
3.9.2. CARACTERISTICAS DEL MICROSERVIDOR WEB.....	31
3.9.3. SOLUCIONES EN HARDWARE.....	31
3.9.4. SOLUCIONES EN SOFTWARE.....	34
3.10. CONTROLADOR ENC28J60.....	35

3.11. PIC18F4620.....	38
-----------------------	----

CAPITULO IV

4. MARCO TEORICO: TECNOLOGIA BLUETOOTH Y APLICACIONES	
ANDROID.....	40
4.1. REDES DE AREA PERSONAL PAN	40
4.2. BLUETOOTH.....	41
4.2.1. CLASES DE BLUETOOTH.....	41
4.2.2. CARACTERISTICAS EN RADIOFRECUENCIA.....	42
4.2.3. PRINCIPIOS DE COMUNICACIÓN.....	44
4.3. MODULO HC-06.....	44
4.3.1. COMANDOS DE CONFIGURACION.....	46
4.4. SISTEMA OPERATIVO ANDROID.....	46
4.5. DESARROLLO DE APLICACIONES ANDROID.....	47
4.6. APP INVENTOR.....	48

CAPITULO V

5. INGENIERIA DEL PROYECTO.....	50
5.1. DIAGRAMA EN BLOQUES.....	50
5.2. ANALISIS PREVIO CON EL MODELO DE TRES NIVELES DE	
COMPLEJIDAD.....	51
5.2.1. TERCER NIVEL CON SERVICIOS Y APLICACIONES.....	51
5.2.2. SEGUNDO NIVEL DE INTERCONEXION DE EQUIPOS.....	52
5.2.3. PRIMER NIVEL DE SENSORES Y ACTUADORES.....	53
5.3. SUBSISTEMA MICROSERVIDOR WEB.....	55
5.3.1. DIAGRAMA DEL CIRCUITO.....	55
5.3.2. CIRCUITO DEL MICROSERVIDOR WEB.....	56
5.3.3. INTERFAZ RJ-45 CON FILTROS.....	57
5.3.4. CONTROLADOR ENC28J60.....	57
5.3.5. ADAPTADOR DE TENSION.....	59
5.3.6. EL PIC18F4620 EN EL MICROSERVIDOR.....	59
5.4. SUBSISTEMA CENTRAL DE CONTROL.....	60
5.4.1. DIAGRAMA DEL CIRCUITO.....	60

5.4.2. CIRCUITO DEL CENTRAL DE CONTROL.....	61
5.4.3. EL PIC16F877A EN EL CENTRAL DE CONTROL.....	62
5.4.4. INTERFAZ SPI EN EL PIC16F877A.....	62
5.4.5. EXPANSIÓN DE PUERTOS DE SALIDA.....	64
5.4.6. EXPANSIÓN DE PUERTOS DE ENTRADA.....	65
5.5. SUBSISTEMA BLUETOOTH LOCAL.....	66
5.5.1. DIAGRAMA DEL CIRCUITO.....	66
5.5.2. CIRCUITO DEL BLUETOOTH LOCAL.....	67
5.5.3. COMUNICACIÓN SERIAL CON EL MODULO HC-06.....	67
5.6. SENSORES Y ACTUADORES.....	69
5.6.1. SENSORES.....	71
5.6.2. ACTUADORES.....	78
5.7. REGISTROS DEL SISTEMA.....	83
5.8. PROGRAMA DEL SUBSISTEMA MICROSERVIDOR WEB.....	85
5.8.1. METODO GET EMPLEADO EN EL SUBSISTEMA.....	85
5.8.2. FIRMWARE DEL PIC18F4620.....	85
5.9. PROGRAMA DEL SUBSISTEMA CONTROL CENTRAL.....	88
5.10. PROGRAMA DE LA APLICACIÓN PARA EL BLUETOOTH LOCAL.....	91
5.11. TOPOLOGIA DE LA RED ETHERNET DE ACCESO REMOTO.....	92
5.12. VISUALIZACION DE LA INTERFAZ DE USUARIO REMOTO.....	93
5.13. VISUALIZACION DE LA INTERFAZ DE USUARIO LOCAL.....	95

CAPITULO VI

6. ANALISIS DE COSTOS.....	99
6.1. COSTOS DIRECTOS.....	99
6.1.1. COSTOS DE LA INFORMACION TECNICA.....	99
6.1.2. COSTOS DE SENSORES Y COMPONENTES ELECTRONICOS.....	99
6.2. COSTOS INDIRECTOS.....	100
6.3. COSTO DE DISEÑO DE HARDWARE Y SOFTWARE.....	100
6.4. COSTO TOTAL.....	101

CAPITULO VII

7.1. CONCLUSIONES.....	102
7.2. RECOMENDACIONES.....	103
7.3. BIBLIOGRAFÍA.....	103
7.4. ANEXOS.....	105
7.4.1. CODIGO DEL PIC18F4620 DEL MICROSERVIDOR WEB.....	105
7.4.2. CODIGO DEL PIC16F877A DEL SUBSISTEMA CONTROL CENTRAL.....	114
7.4.3. HOJA DE ESPECIFICACIONES DEL PIC16F877.....	117
7.4.4. HOJA DE ESPECIFICACIONES DEL PIC18F4620.....	118

INDICE DE FIGURAS

	Página
CAPITULO II	
Figura. 2.1. Los tres modelos de complejidad.....	9
Figura. 2.2. Modelo reticular del hogar domótico.....	12
Figura. 2.3. Arquitecturas de la red de control.....	14
CAPITULO III	
Figura. 3.1. Formato de la Trama 802.3.....	20
Figura. 3.2. Estructura de la dirección MAC.....	22
Figura. 3.3. Clases de dirección IP y sus rangos binarios.....	22
Figura. 3.4. Modelos TCP y OSI.....	24
Figura. 3.5. Esquema de una red local LAN.....	25
Figura. 3.6. Esquema de una red WAN.....	25
Figura. 3.7. Pasos de solicitud una web.....	26
Figura. 3.8. Petición HTTP.....	27
Figura. 3.9. Método GET cliente y servidor.....	27
Figura. 3.10. Estructura básica de una página web en HTML.....	29
Figura. 3.11. Características de un Microservidor web según el modelo TCP.....	31
Figura. 3.12. Encapsulados del ENC28J60.....	32
Figura. 3.13. PIC18F97J60.....	32
Figura. 3.14. Shield Ethernet de Arduino.....	33
Figura. 3.15. Módulo PXI de National Instrument.....	33
Figura. 3.16. Interfaz del compilador MikroC.....	34
Figura. 3.17. Interfaz del IDE Arduino.....	35
Figura. 3.18. Pines del ENC28J60.....	36
Figura. 3.19. Diagrama interno del ENC28J60.....	37
Figura. 3.20. Interfaces del ENC29J60.....	38
Figura. 3.21. Pines del PIC18F4620.....	39
CAPITULO IV	
Figura. 4.1. Ejemplo de Red Piconet.....	44
Figura. 4.2. Modulo Bluetooth HC-06.....	44
Figura. 4.3. Diagrama esquemático del módulo HC-06.....	45
Figura. 4.4. Interfaz IDE Eclipse.....	48

Figura. 4.5. Interfaz del App Inventor.....	49
---	----

CAPITULO V

Figura. 5.1. Diagrama de bloque del sistema de Domótico.....	50
Figura. 5.2. Redes interna y externa de la vivienda.....	52
Figura. 5.3. Diagrama del subsistema Microservidor web.....	55
Figura. 5.4. Detalle del circuito del subsistema Microservidor web.....	54
Figura. 5.5. Conector RJ-45 ERNI.....	57
Figura. 5.6. Circuito del ENC28J60.....	58
Figura. 5.7. Adaptadores de voltaje SPI.....	59
Figura. 5.8. Diagrama del subsistema Control Central.....	60
Figura. 5.9. Detalle del circuito del subsistema Control Central.....	61
Figura. 5.10. Comunicación SPI de un Maestro con tres Esclavos.....	63
Figura. 5.11. Pines del 74HC595.....	64
Figura. 5.12. Registros 74HC595 en serie ampliando salidas paralelas.....	64
Figura. 5.13. Pines del 74HC165.....	65
Figura. 5.14. Registros 74HC165 en serie ampliando entradas paralelas.....	65
Figura. 5.15. Diagrama del subsistema Bluetooth local.....	66
Figura. 5.16. Detalle del circuito del subsistema Control Central.....	67
Figura. 5.17. Ejemplo de un Byte transmitido en USART.....	68
Figura. 5.18. Módulo conmutador de efecto Hall 44e938.....	72
Figura. 5.19. Módulo sensor de barrera H21A1.....	72
Figura. 5.20. Módulo conmutador magnético Reed CD30.....	73
Figura. 5.21. Módulo conmutador Reed con salida digital regulable.....	74
Figura. 5.22. Módulo sensor de vibración SW18010P.....	74
Figura. 5.23. Detector de movimiento pasivo PIR.....	75
Figura. 5.24. Módulo micrófono Electret con sensibilidad regulable.....	76
Figura. 5.25. Módulo sensor de Flama YG1006.....	76
Figura. 5.26. Módulo sensor de temperatura con termistor NTC.....	77
Figura. 5.27. Módulo sensor de temperatura NTC con salida digital regulable.....	77
Figura. 5.28. Módulo sensor de temperatura LM35.....	78
Figura. 5.29. Módulo Buzzer pasivo.....	79
Figura. 5.30. Módulo Buzzer activo.....	79

Figura. 5.31. Módulo Bi-Color LED.....	80
Figura. 5.32. Módulo LED Glow.....	81
Figura. 5.33. Módulo Relé.....	81
Figura. 5.34. Módulo Pack Relé.....	82
Figura. 5.35. Módulo LED RGB.....	83
Figura. 5.36. Módulo LED RGB de alto brillo.....	83
Figura. 5.37. Vector Registros.....	84
Figura. 5.38. Estructura del mensaje de un Byte.....	84
Figura. 5.39. Estructura de los hipervínculos empleando el método GET.....	85
Figura. 5.40. Diagrama de flujo para el Firmware del programa principal del Subsistema Microservidor Web.....	86
Figura. 5.41. Diagrama de flujo para la función SPI_Ethernet_UserTCP del Microservidor Web.....	87
Figura. 5.42. Diagrama de flujo para el Firmware del programa principal del Subsistema Control Central (primera parte).....	89
Figura. 5.43. Diagrama de flujo para el Firmware del programa principal del Subsistema Control Central (segunda parte).....	90
Figura. 5.44. Diagrama de flujo para el Firmware de la aplicación Android.....	91
Figura. 5.45. Topología de la red Ethernet de acceso remoto.....	92
Figura. 5.46. Interfaz gráfica de la página menú Principal.....	93
Figura. 5.47. Interfaz gráfica de la página menú Sala.....	93
Figura. 5.48. Interfaz gráfica de la página menú Cocina.....	94
Figura. 5.49. Interfaz gráfica de la página menú Dormitorio.....	94
Figura. 5.50. Interfaz gráfica de la página menú Exterior.....	95
Figura. 5.51. Interfaz gráfica de la página menú Seguridad.....	95
Figura. 5.52. Interfaz gráfica de la aplicación en el menú Principal.....	96
Figura. 5.53. Interfaz gráfica de la aplicación en el menú Sala.....	96
Figura. 5.54. Interfaz gráfica de la aplicación en el menú Cocina.....	97
Figura. 5.55. Interfaz gráfica de la aplicación en el menú Dormitorio.....	97
Figura. 5.56. Interfaz gráfica de la aplicación en el menú Exterior.....	98
Figura. 5.57. Interfaz gráfica de la aplicación en el menú Seguridad.....	98

INDICE DE TABLAS

	Pagina
CAPITULO III	
Tabla. 3.1. Tecnologías Ethernet.....	21
CAPITULO IV	
Tabla. 4.1. Comparación de tecnologías inalámbricas.....	40
Tabla. 4.2. Clases de dispositivos Bluetooth según su nivel de potencia de transmisión y distancia.....	42
Tabla. 4.3. Versiones Bluetooth.....	42
Tabla. 4.4. Características de la tecnología Bluetooth en radiofrecuencia.....	43
Tabla. 4.5. Características técnicas del módulo HC-06.....	45
Tabla. 4.6. Comandos de configuración del módulo HC-06.....	46
CAPITULO V	
Tabla. 5.1. Requerimiento básico de sensores y actuadores para cada servicio...54	54
Tabla. 5.2. Descripción de los pines del puerto SPI del PIC16F877A.....	63
Tabla. 5.3. Parámetros de configuración del módulo HC-06.....	68
CAPITULO VI	
Tabla. 6.1. Costo para los componentes electrónicos.....	100
Tabla. 6.2. Costo de Diseño.....	101
Tabla. 6.3. Costo total para la implementación del proyecto.....	101

CAPITULO I

1. INTRODUCCION

El presente proyecto tiene como finalidad principal el diseño e implementación de un sistema con funcionalidad domótica, empleando redes de telecomunicaciones cuyo uso ya es parte de la cotidianidad de las personas, aprovechando la capacidad que presenta esta infraestructura para brindar este servicio de valor agregado.

La tecnología basada en direccionamiento IP forma parte ya de todo dispositivo móvil de telecomunicaciones, teniendo acceso a la red global desde cualquier área de cobertura. Se puede aprovechar este panorama para brindar un servicio de control y monitoreo a distancia del domicilio del usuario, mediante una aplicación web y un sistema instalado en casa que tenga compatibilidad con las redes de datos del proveedor de servicios.

Los altos costos en sistemas de telecontrol y telemetría recaen en el uso de dispositivos terminales prediseñados que brindan una interfaz entre el dispositivo controlador domótico y la red de Internet, es por ello que se requiere de un sistema único que integre todas las características de un equipo terminal IP y que a su vez controle los sensores y actuadores mientras se reciben las órdenes del usuario mediante su computadora o Smartphone conectada a la red global.

Es necesario emplear sistemas embebidos microcontrolados con la capacidad de realizar de forma autónoma funciones compatibles con el modelo TCP/IP de redes de computadoras, para resolver el problema de la compatibilidad de tecnologías, y poder conectar al dispositivo de control con la red del proveedor de servicios de Internet.

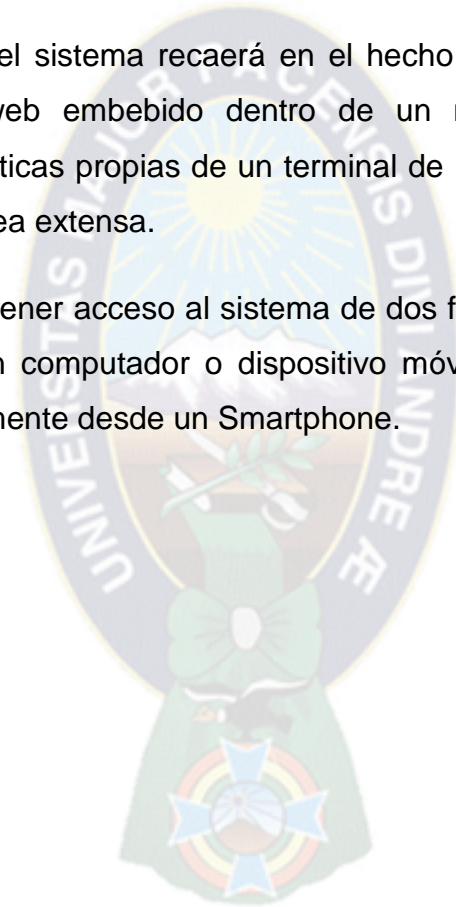
El sistema domótico tiene como principal función la de brindar comodidad y confort al usuario, es por ello que también se requiere monitorear y controlar la vivienda de forma local dentro de la misma, haciendo necesario la integración de

una interface de acceso inalámbrico que pueda ofrecer una conexión directa al sistema.

Es por ello que el propósito del presente proyecto es el de diseñar e implementar un sistema que nos permita controlar dispositivos actuadores instalados en la vivienda y monitorear parámetros relevantes dentro de la misma empleando sensores, solucionando de forma directa problemáticas recurrentes como ser seguridad, ahorro de energía y falta de accesibilidad del usuario.

La complejidad del sistema recaerá en el hecho de que el mismo deberá emplear un servidor web embebido dentro de un microcontrolador, el cual proporcionará características propias de un terminal de red, permitiendo al mismo integrarse a redes de área extensa.

El usuario podrá tener acceso al sistema de dos formas, de manera remota podrá hacerla desde un computador o dispositivo móvil mediante la web, y de forma local inalámbricamente desde un Smartphone.



1.1. PLANTEAMIENTO DEL PROBLEMA

- ✓ Los robos a domicilios son frecuentes a causa de que los mismos no están habitados gran parte del día, ya sea por trabajos absorbentes, constantes viajes y diferentes labores que los habitantes del domicilio realizan, haciendo necesario tener recursos de accesibilidad que permitan conocer el estado en que se encuentra la vivienda y tomar acciones preventivas de seguridad.
- ✓ El ahorro energético es de suma importancia para la economía de toda familia, siendo una problemática latente la falta de uso racional de servicios básicos dentro del hogar, haciendo necesario sistemas que aporten al usuario en la accesibilidad del control y uso de dispositivos eléctricos en casa.
- ✓ Muchas personas con discapacidad presentan problemas por su movilidad limitada dentro del hogar, teniendo dificultades en el control y supervisión de aspectos básicos de su residencia, por la carencia de un sistema de control de mando que le permita controlar distintos dispositivos dentro de la misma vivienda.
- ✓ Actualmente los sistemas domóticos tienen un precio muy alto de instalación con lo cual solo es posible verlo en residencias lujosas, donde suelen utilizar buses de transmisión de información que posibilitan una domótica robusta como son el EIB, X10, CEBus, LonWorks/LongTalk y ZigBee, existiendo la necesidad de implementar sistemas alternativos de menor costo.
- ✓ En toda la red troncal del país las empresas de telecomunicaciones que son proveedoras de servicio de Internet no aprovechan en su totalidad la red de la cual disponen, ya que cuentan con la infraestructura necesaria para poder ofrecer al usuario servicios de valor agregado, que incentive al mismo a adquirir una aplicación útil y práctica empleando su red extensa como una interfaz de conexión al servicio.

1.2. OBJETIVOS DEL PROYECTO

1.2.1. Objetivo general

- ✓ Diseñar e implementar un prototipo de sistema de control y monitoreo Domótico con dos modalidades de acceso para el usuario. La primera de forma remota mediante la web compatible con redes y dispositivos que empleen direccionamiento IP y la segunda de forma local con interfaz inalámbrica para Smartphones con sistema operativo Android.

1.2.2. Objetivos específicos

- ✓ Implementar un circuito que permita al sistema acceder a una conexión Ethernet IEEE para la comunicación remota entre el usuario y su vivienda.
- ✓ Diseñar un Firmware para el Microservidor web que permita alojar páginas HTML para el acceso del usuario remoto.
- ✓ Diseñar páginas web dinámicas que ofrezcan una interfaz directa al sistema instalado en la vivienda mediante el protocolo HTTP.
- ✓ Diseñar una aplicación Android como interfaz de usuario para el control y monitoreo de la vivienda desde un dispositivo móvil empleando la interfaz Bluetooth integrada en la misma.
- ✓ Implementar dispositivos actuadores en un modelo de vivienda estándar que permitan al usuario controlar la iluminación, realizar la apertura de persianas y controlar el garaje.
- ✓ Implementar sensores en un modelo de vivienda estándar para el monitoreo del estado actual de los actuadores y la recepción de alertas de seguridad.

1.3. JUSTIFICACIÓN

1.3.1. Justificación Académica:

- ✓ Empleando en la práctica importantes conocimientos adquiridos en nuestra facultad con referente a microcontroladores, creación de software y firmwares, manejo de protocolos de comunicación y uso de sensores y actuadores, se tiene fundamentos sólidos para diseñar e implementar un sistema con funcionalidad domótica orientada a la conexión remota y local que trabaje de forma eficiente y de bajo costo.

1.3.2. Justificación Tecnológica:

- ✓ Gracias a la disponibilidad de materiales comerciales, la utilización de tecnología costosa no es necesaria, y por lo tanto puede ser utilizada en el hogar de familias que requieren de un sistema que mejore su calidad de vida, impulsando de esta manera la utilización de tecnología hecha en el país, útil, económica, fiable y de fácil implementación.

1.3.3. Justificación Social:

- ✓ El diseño e implementación del proyecto es conveniente pues se busca hacer un aporte social en los hogares de cada uno de los usuarios, puesto que el rápido avance de la tecnología ha permitido a las personas disponer de equipos móviles que les permita acceder a un servicio en domótica accesible económicamente, brindando mayor seguridad y reduciendo el mal uso energético de la vivienda.

1.4. DELIMITACIÓN

El presente proyecto de grado es un prototipo que puede ser mejorado en varias formas y tiene las siguientes delimitaciones.

1.4.1. Delimitación temática

En el diseño del proyecto recurre a la teoría relacionada con el diseño de Firmwares para sistemas embebidos en microcontroladores orientadas a la conexión con interfaces Ethernet y Bluetooth, además del diseño de aplicaciones con formato APK y el empleo de protocolos HTTP en redes IP.

1.4.2. Delimitación temporal

El desarrollado del proyecto empleara un tiempo estimado de 6 meses como máximo desde su inicio hasta la conclusión del mismo, luego de su implementación tendrá un tiempo de vida promedio de cuatro a cinco años, realizando el soporte técnico de los circuitos electrónicos ya sea por la mala manipulación del equipo o por el desgaste de los componentes.

1.4.3. Delimitación espacial

El presente proyecto es desarrollado en los laboratorios de la carrera de Electrónica y Telecomunicaciones de la Facultad de Tecnología y está destinado a ser un servicio de valor agregado para un proveedor de servicios de internet, en el eje troncal de Bolivia, instalado en diferentes hogares como un equipo terminal de red y a su vez controle y monitoree los domicilios que dispongan de este sistema.

1.5. METODOS DE INVESTIGACION

Para la elaboración del presente proyecto de grado se analizarán circuitos digitales protocolos de red lo que nos lleva a hacer diseños analíticos así como pruebas experimentales, buscando que la parte teórica se relacione con la práctica.

La experimentación científica es el método que nos hará sentir más seguros de lo que estamos realizando, pudiendo modificar algunos circuitos para alcanzar el objetivo del proyecto.

El proyecto presenta diferentes etapas, por lo cual se utilizaran diferentes tipos de metodología las cuales se detallan a continuación.

- El método analítico será utilizado en el análisis teórico de los protocolos de comunicación que se emplearan en los circuitos digitales utilizados como interfaz para la etapa de comunicación.
- El método lógico deductivo se lo empleará en los circuitos de control digital y el software de visualización para la interfaz de usuario puesto que los procesos que se llevaran a cabo dentro de los mismos son deducidos mediante un razonamiento lógico.
- El método experimental es primordial en el presente proyecto porque el sistema propuesto deberá ser compatible con tecnologías existentes como ser equipos terminales móviles, cuya eficiencia y estabilidad será puesta a prueba de forma exhaustiva experimentando el correcto funcionamiento de los diversos subsistemas que la conforman.

CAPITULO II

2. MARCO TEORICO: DOMOTICA

2.1. DEFINICION DE DOMOTICA

Domótica es el conjunto de sistemas capaces de automatizar una vivienda, aportando servicios de gestión energética, seguridad, bienestar y control, que pueden estar integrados por medio de redes interiores y exteriores de comunicación, cableadas o inalámbricas. Se podría definir como la integración de la tecnología en el diseño inteligente de un recinto cerrado.

El término domótica viene de la unión de las palabras domus (que significa casa en latín) y tica (de automática, palabra en griego, 'que funciona por sí sola').

2.2. LA DOMOTICA Y EL MODELOS DE TRES NIVELES DE COMPLEJIDAD

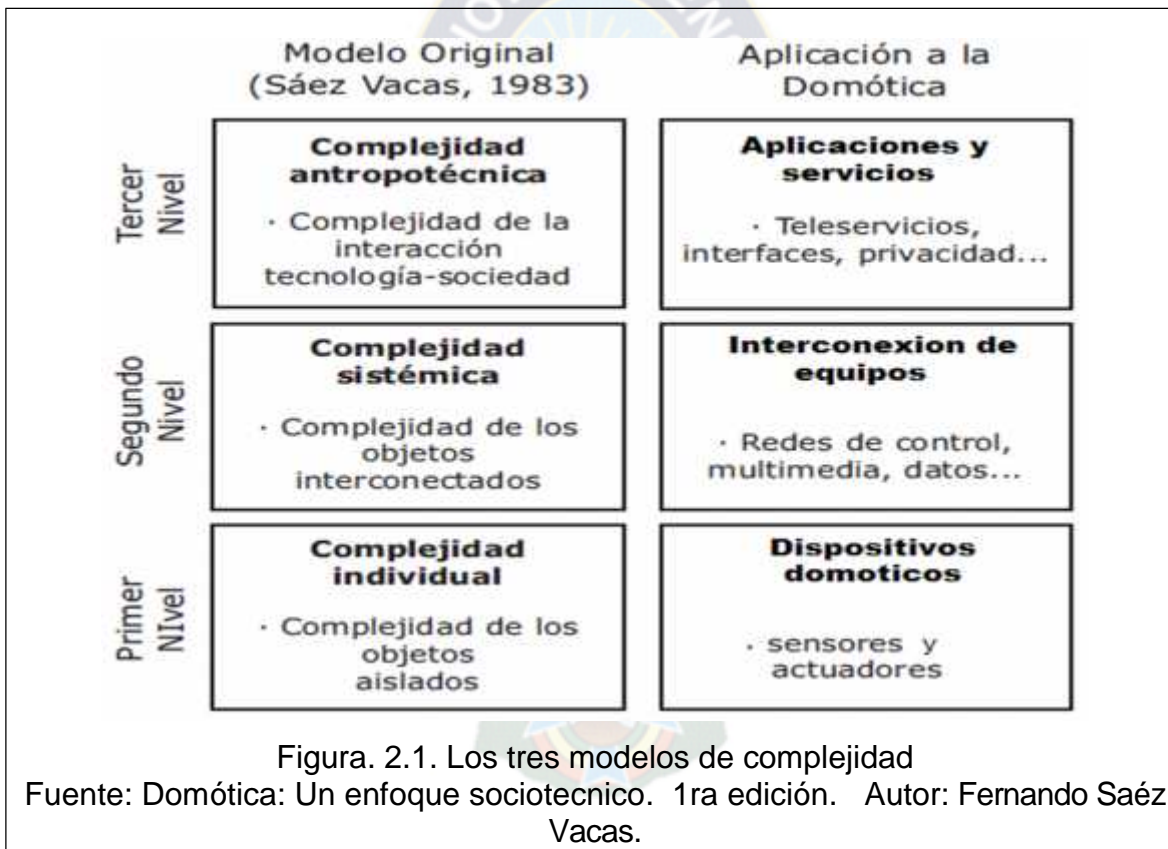
El modelo de tres niveles de complejidad fue propuesta por Fernando Sáez Vacas en 1983 para estudiar y estratificar la complejidad que caracteriza los entornos informáticos. Aunque los razonamientos se establecieron hace más de veinte años y concretamente se desarrollaron para el ámbito de la informática, aplicados posteriormente a la ofimática la esencia del modelo goza aún hoy de buena salud y sigue siendo bastante independiente del campo de aplicación, empleándose en este caso a la domótica.

Sáez Vacas jerarquiza la complejidad en tres niveles. El primero de ellos recoge la complejidad de los objetos aislados; en el caso de la informática comprende la complejidad de algoritmos, funciones y programas tratados como objetos separados de otros. Para la Domótica estos elementos son sensores, actuadores.

Luego parece el segundo nivel de complejidad donde los objetos ya no están aislados sino que se interrelacionan a fin de lograr unos determinados objetivos. Así aparecen los sistemas y la complejidad de orden superior denominándola la complejidad sistémica, donde se analiza el comportamiento

conjunto. Existen múltiples ejemplos tanto en el campo informático (sistemas operativos, redes de ordenadores) como en el de nuestro particular interés, la Domótica (sistemas de control automatizado, sistemas avanzados de comunicaciones, redes multimedia).

Por último, aparece el tercer nivel de complejidad, la interacción de los sistemas tecnológicos con los sistemas sociales. El estudio de las interfaces, la aceptación social de la tecnología, el impacto económico de una cierta innovación tecnológica, todos ellos caen dentro de este tercer nivel. En Domótica se hace referencia a las aplicaciones y servicios que el sistema ofrece al usuario.



2.3. PRIMER NIVEL: DISPOSITIVOS DOMOTICOS

Existe gran variedad de dispositivos domóticos, aun así la constante mejora tecnológica y las cada vez más frecuentes innovaciones introducen constantes cambios en los mismos. Aun así se los puede distinguir en dos categorías, como ser sensores y actuadores.

Los sensores o detectores son dispositivos capaces de recoger la información de los distintos parámetros que controlan y de transmitir esta información para su procesamiento.

Los actuadores son dispositivos capaces de recibir una orden procedente de un sistema de control y realizar una acción que modifique el estado de un determinado equipo o instalación: encendido o apagado, subida o bajada, apertura o cierre.

Los sensores y actuadores más comunes en domótica se pueden reflejar en el siguiente listado, con una descripción general.

2.3.1. ACTUADORES

Según el tipo de señal de entrada:

- Actuator todo o nada: Encendido o apagado directo de un grupo dispositivo.
- Actuator digital: Envío de instrucciones mediante tramas digitales al dispositivo actuator.
- Actuator analógico: Señales variables enviadas al actuator para que regule la variable controlada.

Según la magnitud controlada:

- Relés electromecánicos: En su mayoría se pueden distinguir los de núcleo móvil, que permiten manejar mayores corrientes, los relés reed de muy reducido tamaño y los relés polarizados con sentido de alimentación para apertura y cierre.
- El contactor: Está pensado para trabajar como interruptor automático, pero con corrientes y tensiones más elevadas. Los contactos principales son los destinados a las maniobras del circuito de potencia.
- El motor eléctrico: Es sin duda el actuator eléctrico más utilizado y que más aplicaciones tiene en todos los niveles de la utilización de la

electricidad. Existe una gama muy amplia de tipo de motores, que se elegirán en función de la aplicación y de la red eléctrica que se disponga en el lugar de utilización.

- La iluminación: Es una necesidad, pero supone también una de las principales fuentes de confort y consumo de energía en una vivienda o edificio.
- Calefacción: Dependerá del tipo de fluido térmico y del sistema empleado (agua, aire, electricidad, etc.) pero típicamente está asociado a la conexión/desconexión del sistema general, o de uno o varios calefactores instalados en una estancia.
- Electroválvulas: Empleado para corte de fuentes de suministros de agua o gas en caso de emergencia requieren de un cierre desde el controlador central.

2.3.2. SENSORES

Según el tipo de señal de salida:

- Sensor todo o nada: En la práctica son sensores que cierran contactos enviando al controlador un nivel alto o bajo de tensión.
- Sensor digital: El controlador recibe tramas digitales con información proveniente del sensor.
- Sensor analógico: La señal de salida del sensor varía de forma continua según lo hace la variable que se está monitoreando.

Según la magnitud medida:

- Sensor de temperatura: Los diferentes sistemas recurren a termostatos (detectores de tipo todo/nada que abren o cierran un contacto cuando el margen de temperatura ha sido superado) o bien a sensores o sondas de temperatura cuya señal eléctrica de salida (analógica o digital) es proporcional al valor de la temperatura real.

- Sensor de humo o fuego: Es un detector de incendios cuya eficiencia depende de distintos factores, entre ellos el desarrollo probable del incendio en sus fases iniciales, la altura y volumen de la estancia, la existencia de posibles generadores de falsas alarmas.
- Sensor de gas: Se utilizan para la detección de posibles fugas de gas, permitiendo evitar así la intoxicación de personas en un edificio y/o reducir los riesgos de explosión.
- Sensor de presencia: Cumple funciones de seguridad patrimonial y/o para automatizar otras funciones como la iluminación. Pueden ser volumétricos (interiores) o paramétricos (exteriores).
- Sensor magnético: Se colocan en ventanas y puertas para registrar su estado de apertura o cierre.

2.4. SEGUNDO NIVEL: INTERCONEXION DE EQUIPOS

Para evaluar las redes domésticas contemplaremos los conceptos de exterior, frontera e interior en un sistema domótica.

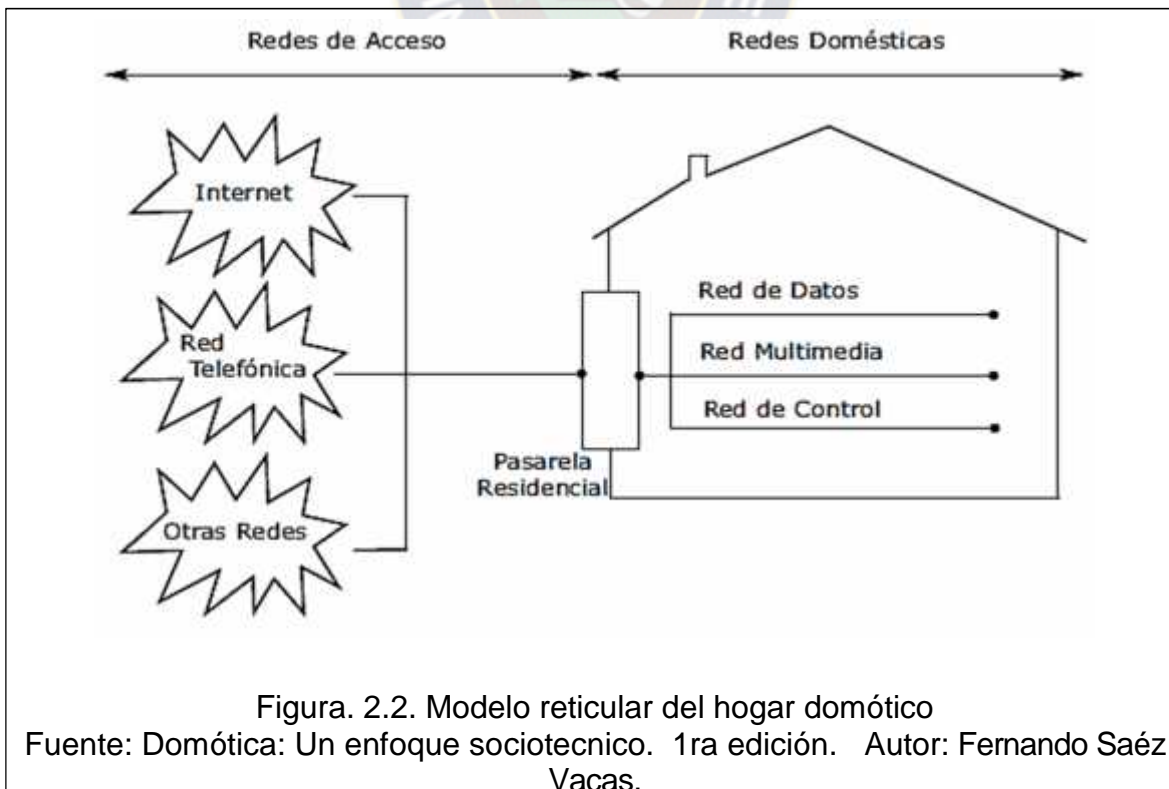


Figura. 2.2. Modelo reticular del hogar domótico

Fuente: Domótica: Un enfoque sociotecnico. 1ra edición. Autor: Fernando Saéz Vacas.

Considerando el exterior del hogar como un conjunto de redes que facilitan la comunicación a distancia con otros individuos u organizaciones, al tiempo que proporcionan un medio de acceso remoto a la propia vivienda. Nos referimos a las redes de acceso.

Por otra parte consideremos el interior del hogar como un conjunto de redes domésticas, integradas por los dispositivos de nuestro gabinete principal. Se puede distinguir tres redes o subsistemas domésticos desde el punto de vista funcional.

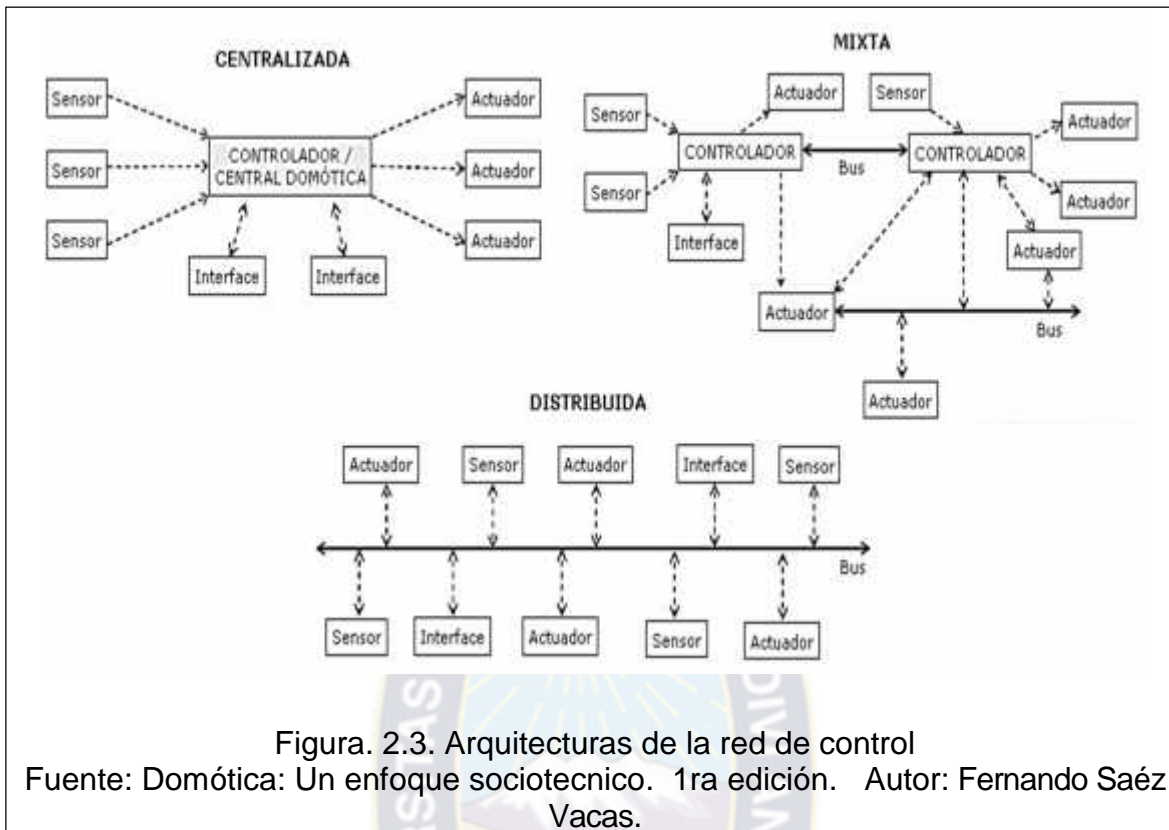
- Red de datos: para el envío y recepción de mensajes y ficheros entre ordenadores, periféricos y demás recursos informáticos.
- Red multimedia o de entretenimiento: para el soporte de reproductores de audio y vídeo, consolas de videojuegos y demás plataformas de ocio.
- Red de control: para el control y monitorización de sensores, actuadores y electrodomésticos de la vivienda.

Como frontera o nexo de unión entre interior y exterior situamos la pasarela residencial, vínculo entre las redes de acceso y las redes domésticas, y de éstas, entre sí. La pasarela facilita la comunicación entre aquellos dispositivos situados en el interior de la vivienda, y permite el diálogo de éstos con cualquier otro equipo exterior conectado a la red de comunicaciones adecuada.

2.4.1. RED DE CONTROL

La red de control de dispositivos domóticos es la responsable del control sobre la automatización de la vivienda. Básicamente es independiente de las redes de datos y multimedia, pero esta independencia tiende a desaparecer en la actualidad con la introducción en el mercado de los primeros sensores y actuadores basados en sistemas que trabajan bajo protocolos IP.

Existen tres arquitecturas domóticas básicas para el diseño de la red de control, se reflejan en la figura 2.3.



Mientras que por lo general la red de datos y multimedia son sistemas distribuidos, con la inteligencia desplazada de la red hacia los terminales, es habitual que la red de control esté centralizada. Esto se debe a la necesidad de reducir la complejidad, tamaño y coste de los sensores y actuadores.

El sistema de control centralizado es el elemento encargado de recoger toda la información que proporcionan los sensores distribuidos en los distintos puntos de control de la vivienda, procesarla y generar las órdenes que ejecutarán los actuadores e interruptores. Este sistema constituye el núcleo o cerebro de la red de control, el elemento central en el que reside el grueso de su inteligencia.

2.4.2. PASARELA RESIDENCIAL

La pasarela residencial es un equipo frontera entre exterior e interior del hogar digital, el nexo de unión entre las distintas redes de acceso externas y las redes domésticas internas.

El objetivo de la pasarela residencial es facilitar la intercomunicación y la eventual convergencia de los tres tipos de redes del entorno doméstico (datos, multimedia y control) y conectar estas redes con el exterior, facilitando así el acceso a redes de banda ancha. Esta pasarela es una interfaz de terminación de red inteligente, normalizada y flexible, que recibe y transfiere señales con origen o destino en alguna de las redes mencionadas, todo ello de forma transparente.

La pasarela residencial incorpora los componentes y las funciones siguientes:

- La terminación física de los accesos externos y de los medios de distribución internos: En tanto que la pasarela liga dispositivos internos de la vivienda a una o varias redes públicas de datos, debe contar con los puertos físicos precisos que permitan la conexión a los distintos tipos de redes.
- La adaptación de protocolos: Las redes de la vivienda pueden emplear diferentes medios físicos, tecnologías y protocolos. Una misión definitoria de la pasarela es adaptar los protocolos de todos los dispositivos y redes para que puedan entenderse entre sí.
- La gestión de las propias redes domésticas internas: La pasarela monitoriza y supervisa el funcionamiento de las tres redes internas de la vivienda, permitiendo detectar y resolver ciertas anomalías, así como cambiar parámetros de configuración.
- La gestión de los dispositivos internos: A través de la pasarela se puede controlar los dispositivos domóticos de la casa, si bien esta labor suele recaer en último término en el sistema de control centralizado.

2.5. TERCER NIVEL: APLICACIONES Y SERVICIOS

Entendemos por aplicación el uso de sistemas técnicos que implementen ciertas funciones con el fin de satisfacer una determinada necesidad del usuario final ofreciendo así un servicio requerido por el mismo. Cualquier aplicación es susceptible de ser ofrecida como un servicio por parte de un proveedor de

servicios especializado que permite el acceso, proporciona el mantenimiento o gestiona las funciones del sistema domótico.

Existe gran variedad de aplicaciones domóticas dirigidas a brindar un servicio al usuario, las cuales se describirán a continuación.

2.5.1. SEGURIDAD

La seguridad es una de las aplicaciones más adelantadas de la Domótica, y la que hoy en día contribuye en mayor medida a la introducción real de sistemas domóticos en los hogares. Mejorar la seguridad de los hogares es una importante necesidad social que repercute no sólo en la protección de los bienes privados, sino también en la salvaguardia de los propios individuos. Entre los tipos de seguridad que se pueden aplicar en domótica tenemos los siguientes:

- Seguridad perimetral de intrusión: Se emplea sensores perimetrales, volumétricos y detectores de presencia distribuidos por la vivienda, que en caso de alarma desencadenarán una serie de acciones previamente programadas, como puede ser la activación de una sirena, el envío de un aviso por SMS al propietario y la notificación de la incidencia a la empresa de seguridad correspondiente. La simulación de presencia constituye un interesante método de seguridad pasiva, pues permite recrear una situación similar a la que se daría en una vivienda habitada cuando sus propietarios se encuentren ausentes.
- Seguridad técnica: Se presenta cuando se presentan problemas como ser la existencia de un incendio. El sistema puede alertar al usuario, avisar a la compañía de seguridad correspondiente y cerrar las válvulas o llaves de paso pertinentes para evitar la propagación del problema.

2.5.2. AHORRO ENERGETICO

El ahorro energético no es algo tangible, sino un concepto al que se puede llegar de muchas maneras. En muchos casos no es necesario sustituir los

aparatos o sistemas del hogar por otros que consuman menos sino una gestión eficiente de los mismos. Entre algunos ejemplos que se pueden mencionar para esta aplicación tenemos los siguientes:

- **Accesibilidad:** Seremos capaces de apagar todas las luces al salir de casa así como otros dispositivos como aires acondicionados. Todo con sólo pulsar un botón, sin preocuparse por si se la luz del dormitorio encendida.
- **Climatización:** Mediante sensores de temperatura podremos saber desde nuestros terminales de interfaz de usuario cuando encender la calefacción o abrir una ventana de forma rápida y cómoda.
- **Programar iluminación:** Empleando células fotosensibles se puede encender de forma automática la iluminación interna y externa de la vivienda.

2.5.3. CONFORT

Estas aplicaciones y servicios permiten mejorar la calidad de vida de los usuarios al aportar soluciones que facilitan la realización de tareas domésticas rutinarias, que suponen una mayor comodidad para el usuario. De igual forma el acceso remoto a la vivienda a través de dispositivos móviles, teléfonos convencionales y computadores conectados a Internet para monitorizar su estado o realizar cambios en el mismo son de gran ayuda.

- **Portero automático integrado:** La apertura del garaje y la comunicación con el visitante desde un terminal son aplicaciones convencionales en una vivienda con sistema domótico.
- **Control de iluminación:** En una vivienda domótica el usuario puede decidir qué luz o grupo de luces encender generando distintos ambientes según sus deseos (ver la televisión, leer, dormir, etc.). También se puede hacer uso de detectores de presencia para apagar automáticamente las luces de aquellas estancias en las que no haya nadie. Las posibilidades son ciertamente amplias.

- Control y monitoreo mediante Internet: Todas las funciones que se pueden realizar localmente con el terminal del usuario, se pueden integrar a la red global para tener un acceso remoto permanente.



CAPITULO III

3. MARCO TEORICO: MICROSERVIDOR WEB EMBEBIDO

3.1. ETHERNET

Ethernet es un estándar de redes de área local para computadores con acceso al medio por detección de la onda portadora y con detección de colisiones (CSMA/CD). Ethernet define las características de cableado y señalización de nivel físico y los formatos de tramas de datos del nivel de enlace de datos del modelo OSI.

Ethernet se tomó como base para la redacción del estándar internacional IEEE 802.3. Posteriormente ha habido ampliaciones sucesivas al estándar que cubrieron las ampliaciones de velocidad (Fast Ethernet, Gigabit Ethernet y el de 10 Gigabits Ethernet), redes virtuales, hubs, conmutadores y distintos tipos de medios, tanto de fibra óptica como de cables de cobre (tanto par trenzado como coaxial).

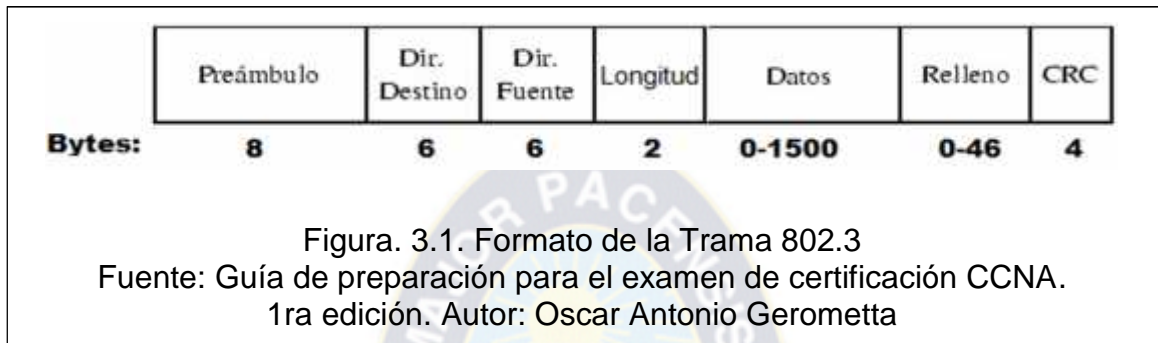
3.1.1 FORMATO DE LA TRAMA 802.3

La trama de bits del estándar 802.3 presenta varios campos destinados a funciones específicas en la comunicación, los cuales se describen a continuación.

- Preámbulo: El patrón de unos y ceros alternados que avisan al receptor la llegada de una trama y se sincronice.
- Direcciones destino y origen: Vienen determinadas por las direcciones MAC únicas de cada tarjeta de red (6 bytes en hexadecimal)
- Longitud: La longitud indica la cantidad de bytes de datos que sigue este campo.
- Datos: Una vez que se ha completado el procesamiento de la capa física y de la capa de enlace, los datos se envían a un protocolo de capa superior, que debe estar definido dentro de la porción de datos de la trama. Si los datos de la trama no son suficientes para llenar la trama hasta una cantidad

mínima de 64 bytes, se insertan bytes de relleno para asegurar que por lo menos haya una trama de 64 bytes (tamaño mínimo de trama).

- Secuencia de verificación: Esta secuencia contiene un valor de verificación CRC de 4 bytes, creado por el dispositivo emisor y recalculado por el dispositivo receptor para verificar la existencia de tramas dañadas.



3.1.2 TECNOLOGIAS ETHERNET

Con el pasar de los años Ethernet ha evolucionado, mejorando la velocidad de transmisión y empleando nuevos medios de conexión. En la tabla 2.1 se puede apreciar las características de cada una de las tecnologías existentes en la actualidad.

Tecnología	Velocidad de transmisión	Tipo de cable	Distancia máxima	Topología
10Base2	10 Mbit/s	Coaxial	185 m	Bus (Conector T)
10BaseT	10 Mbit/s	Par Trenzado	100 m	Estrella (Hub o Switch)
10BaseF	10 Mbit/s	Fibra óptica	2000 m	Estrella (Hub o Switch)
100BaseT4	100 Mbit/s	Par Trenzado (categoría 3UTP)	100 m	Estrella. Half Duplex (hub) y Full Duplex (switch)
100BaseTX	100 Mbit/s	Par Trenzado (categoría 5UTP)	100 m	Estrella. Half Duplex (hub) y Full Duplex (switch)
100BaseFX	100 Mbit/s	Fibra óptica	2000 m	No permite el uso de hubs

1000BaseT	1000 Mbit/s	4 pares trenzado (categoría 5e ó 6UTP)	100 m	Estrella. Full Duplex (switch)
1000BaseSX	1000 Mbit/s	Fibra óptica (multimodo)	550 m	Estrella. Full Duplex (switch)
1000BaseLX	1000 Mbit/s	Fibra óptica (monomodo)	5000 m	Estrella. Full Duplex (switch)

Tabla. 3.1. Tecnologías Ethernet

Fuente: Guía de preparación para el examen de certificación CCNA. 1ra edición.

Autor: Oscar Antonio Gerometta

3.2. MODELO DE REFERENCIA OSI

El modelo de interconexión de sistemas abiertos (ISO/IEC 7498-1), más conocido como modelo OSI, es el modelo de red descriptivo, que fue creado en el año 1980 por la Organización Internacional de Normalización.

Es un marco de referencia para la definición de arquitecturas en la interconexión de los sistemas de comunicaciones. Este modelo está dividido en siete capas o niveles, cada uno con una función específica.

1. Capa física: Define las especificaciones eléctricas, mecánicas, de procedimiento y funcionales para activar, mantener y desactivar el enlace físico entre sistemas finales. Las características tales como niveles de voltaje, temporización de cambios de voltaje, velocidad de datos físicos, distancias de transmisión máximas, conectores físicos y otros atributos similares son definidas por las especificaciones de la capa física.
2. Capa de enlace de datos: La capa de enlace de datos proporciona tránsito de datos confiable a través de un enlace físico. Al hacerlo, la capa de enlace de datos se ocupa del direccionamiento físico (MAC) cuyo número identifica de forma única a la interfaz. Esta capa también abarca áreas como ser la topología de red, el acceso a la red, la notificación de errores, entrega ordenada de tramas y control de flujo.

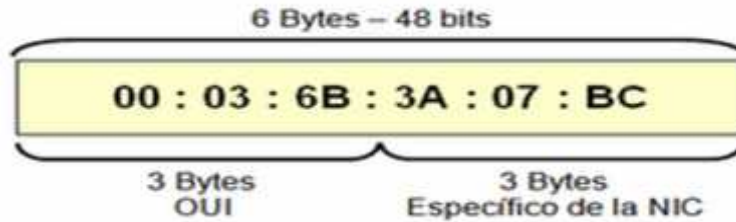


Figura. 3.2. Estructura de la dirección MAC
Fuente: Guía de preparación para el examen de certificación CCNA.
1ra edición. Autor: Oscar Antonio Gerometta

3. Capa de red: Se encarga de identificar el enrutamiento existente entre una o más redes, siendo el paquete la unidad de información de esta capa. Se emplea el direccionamiento lógico, cuya clasificación esta se da en dos formas:
- Clases de dirección IP: Las clases se distinguen a través del rango de bits de red y bits de host como se ve en la figura 2.3.
 - Direcciones IP públicas y privadas: Las IP públicas se asignan a puertos cuyo acceso se lo realiza mediante Internet, y las IP privadas se emplean en redes internas locales.



Figura. 3.3. Clases de dirección IP y sus rangos binarios
Fuente: Guía de preparación para el examen de certificación CCNA.
1ra edición. Autor: Oscar Antonio Gerometta

4. La capa de transporte: Capa encargada de efectuar el transporte de los datos que se encuentran dentro del paquete de la máquina origen a la de

destino, independizándolo del tipo de red física que esté utilizando. Su unidad de información es el segmento o datagrama. Sus protocolos son:

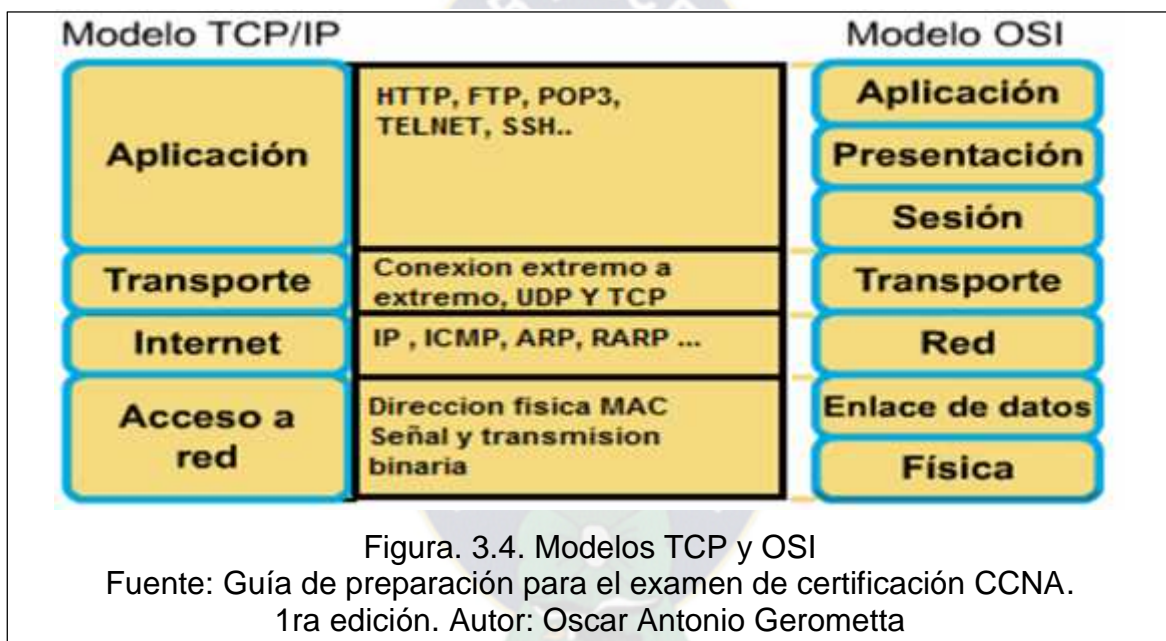
- TCP (Protocolo de Control de Transmisión): Está orientado a conexión. Garantiza que los datos serán entregados en su destino sin errores y en el mismo orden en que se transmitieron empleando control de flujo. Se lo emplea en la transmisión de datos.
 - UDP (Protocolo de datagrama de usuario): Permite el envío de datagramas a través de la red sin que se haya establecido previamente una conexión y sin control de flujo. Su uso principal es para la transmisión de audio y vídeo en real, donde no es posible realizar retransmisiones por los estrictos requisitos de retardo.
5. La capa de sesión: Establece, administra y finaliza las sesiones entre dos hosts que se están comunicando. Sincroniza el diálogo entre las capas de presentación de los dos hosts y administra su intercambio de datos.
6. La capa de presentación: El objetivo es encargarse de la representación de la información, de manera que aunque distintos equipos puedan tener diferentes representaciones internas de caracteres los datos lleguen de manera reconocible.
7. La capa de aplicación: Ofrece a las aplicaciones de usuario la posibilidad de acceder a los servicios de las demás capas y define los protocolos que utilizan las aplicaciones para intercambiar datos. En esta capa aparecen diferentes protocolos y servicios:
- Protocolos: Entre los más destacables tenemos FTP, DNS, DHCP, HTTP, SMTP, TELNET.
 - Servicios: Aplicaciones de red y World Wide Web.

3.3. MODELO TCP/IP

El modelo TCP/IP (Protocolo de Control de Transmisión/Protocolo de Internet) describe un conjunto de guías generales de diseño e implementación de protocolos de red específicos para permitir que un equipo pueda comunicarse en

una red. TCP/IP provee conectividad de extremo a extremo especificando como los datos deberían ser formateados, direccionados, transmitidos, enrutados y recibidos por el destinatario.

Presenta cuatro capas jerarquizadas, cada capa debe ocuparse exclusivamente de su nivel inmediatamente inferior, a quien solicita servicios, y del nivel inmediatamente superior, a quien devuelve resultados. Las funciones de cada capa se pueden comparar con las del modelo OSI, como se ve en la figura 2.4.



3.4. REDES LAN

La red de área local es un grupo de equipos que pertenecen a la misma organización y están conectados dentro de un área geográfica pequeña a través de una red, generalmente con la misma tecnología, siendo Ethernet la más utilizada.

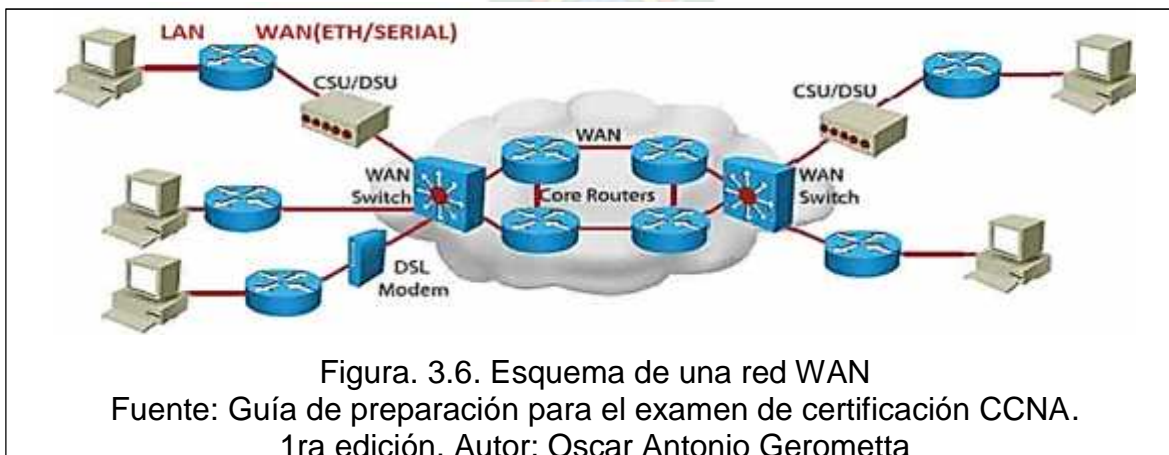
Es una red en su versión más simple. La velocidad de transferencia de datos en una red de área local puede alcanzar hasta 10 Mbps (en una red Ethernet) y 1 Gbps (en FDDI o Gigabit Ethernet). Una red de área local puede contener 100, o incluso 1000, usuarios.



3.5. REDES WAN

Una red de área amplia es una red de computadoras que abarca varias ubicaciones físicas, proveyendo servicio a una zona, un país, incluso varios continentes. Es cualquier red que une varias redes locales, llamadas LAN, por lo que sus miembros no están todos en una misma ubicación física.

Muchas WAN son construidas por organizaciones o empresas para su uso privado, otras son instaladas por los proveedores de internet (ISP) para proveer conexión a sus clientes.



3.6. SERVIDOR WEB

Son servidores web cuya tarea es alojar sitios y/o aplicaciones, las cuales son accedidas por los clientes utilizando un navegador que se comunica con el servidor utilizando el protocolo HTTP.

Básicamente un servidor web consta de un intérprete HTTP el cual se mantiene a la espera de peticiones de clientes y le responde con el contenido según sea solicitado. El cliente, una vez recibido el código, lo interpreta y lo exhibe en pantalla.

Además los servidores pueden disponer de un intérprete de otros lenguajes de programación dentro del código HTML de las páginas que contiene el sitio antes de enviar el resultado al cliente.



3.6.1. PROTOCOLO HTTP

Este protocolo se emplea en cada transacción de la World Wide Web. Es un protocolo orientado a transacciones y sigue el esquema petición-respuesta entre un cliente y un servidor.

Al cliente efectúa la petición mediante un navegador web, la información transmitida se la llama recurso y se la identifica mediante un localizador uniforme de recursos o URL.

La primera línea de una petición contiene los comandos HTTP conocidos como métodos. Existen varios, pero los más conocidos y utilizados son el GET y el POST.



Figura. 3.8. Petición HTTP

Fuente: Cómo funciona la web. 1ra edición. Autor: Claudio Gutiérrez Gallardo.

3.6.2. PETICION GET

Es el método empleado para la petición de información del servidor al cliente. El recurso se solicita a través de la URL al servidor Web.

En la barra de URL de un navegador cualquiera, una petición es análoga a una dirección web, como se puede observar en la figura 2.9. La respuesta de todo servidor está formado por encabezados seguido del recurso solicitado, en el caso de un servidor web es la página solicitada.

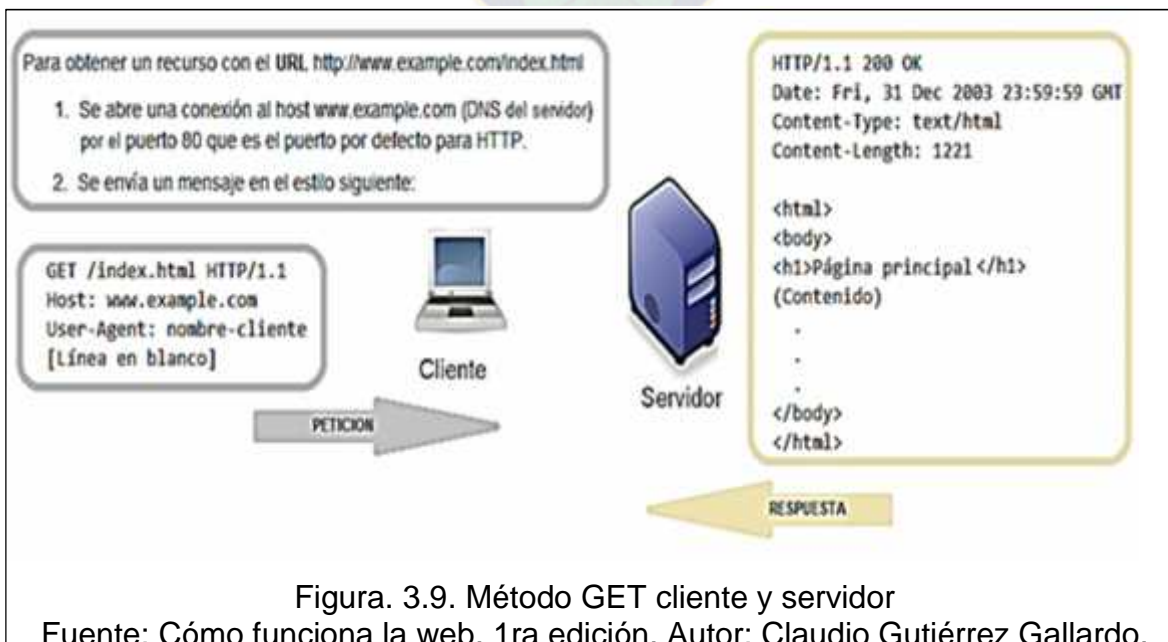


Figura. 3.9. Método GET cliente y servidor

Fuente: Cómo funciona la web. 1ra edición. Autor: Claudio Gutiérrez Gallardo.

3.6.3. PETICION POST

Es el método que se usa para enviar información desde el cliente para que sea procesada y actualice o agregue información en el servidor. La característica del método GET se da en el envío de datos, el cual es parte del cuerpo de la petición.

En el caso de un envío de datos usando el método POST, los datos no se visualizaran en la URL. Para poder recuperar los valores de los datos en el caso de un envío con el método se requiere de herramientas adicionales.

3.7. PAGINAS WEB DINAMICAS EN HTML

3.7.1. HTML

El lenguaje de marcas de hipertexto es un estándar que sirve de referencia para la elaboración de páginas web en sus diferentes versiones, define una estructura básica y un código para la definición de contenido de una página web, como texto, imágenes, videos, entre otros. Es un estándar a cargo de la W3C, organización dedicada a la estandarización de casi todas las tecnologías ligadas a la web.

El HTML se escribe en forma de etiquetas, rodeadas por corchetes angulares (<,>). Dichas etiquetas consisten en breves instrucciones de comienzo y final, mediante las cuales se determina la forma en la que debe aparecer en su navegador el texto, así como también las imágenes y los demás elementos, en la pantalla del ordenador. En general las reglas básicas que cumplen son las siguientes:

- Las etiquetas están encerradas entre los signos "<" y ">".
- Generalmente vienen en pares <p> y </p> para la apertura y cierre.
- El texto que se encuentra entre dos etiquetas es el contenido del elemento.
- Las etiquetas no son sensibles a las mayúsculas y minúsculas, o sea es lo mismo que .

<code><html></code>	(Definición de inicio)
<code><head></code>	(Cabecera)
<code><title>Una página Web</title></code>	(Titulo)
<code></head></code>	
<code><body></code>	(Cuerpo)
Hola a todos. <code>Este texto es en negrita</code>	(Contenido)
<code></body></code>	
<code></html></code>	

Figura. 3.10. Estructura básica de una página web en HTML
Fuente: Cómo funciona la web. 1ra edición. Autor: Claudio Gutiérrez Gallardo.

3.7.2. JAVASCRIPT

El JavaScript es un lenguaje de programación que surgió por la necesidad de ampliar las posibilidades de diseño de páginas web. HTML solamente provee de elementos que actúan exclusivamente sobre el texto y su estilo y mostrarlos en el navegador de forma estática, pero no permite que esta se modifique de forma dinámica, como por ejemplo abrir una nueva ventana o emitir un mensaje de aviso.

JavaScript se diseñó con una sintaxis similar al C, aunque adopta nombres y convenciones del lenguaje de programación Java. Sin embargo Java y JavaScript no están relacionados y tienen semánticas y propósitos diferentes. Todos los navegadores modernos interpretan el código JavaScript integrado en las páginas web de forma automática y su uso es común.

Para insertar códigos JavaScript en una página HTML, se utiliza la etiqueta `<SCRIPT>`. Dentro de la misma se aloja el programa que se ejecutara empleando variables cuyos resultados se traducirán en cambios en la página dependiendo de los requerimientos del diseñador.

3.8. SISTEMAS EMBEBIDOS

Son sistemas que emplean electrónica programable especialmente diseñada para realizar una o algunas pocas funciones específicas, donde el hardware y software que la conforman están destinadas a cumplir ese rol.

Frecuentemente trabajan en tiempo real y con requerimientos de alta confiabilidad. Al contrario de lo que ocurre con los ordenadores de propósito general, como por ejemplo una computadora personal o PC que están diseñados para cubrir un amplio rango de necesidades, los sistemas embebidos se diseñan para cubrir necesidades específicas.

Las principales características de un sistema embebido son el bajo costo y consumo de potencia. Generalmente, los sistemas embebidos emplean procesadores muy básicos y memorias pequeñas para minimiza los costos. Los programas en estos sistemas se ejecutan minimizando los tiempos muertos y enfrentando fuertes limitaciones de hardware.

3.9. MICROSERVIDOR WEB

El denominativo de Microservidor Web esta dado tanto por el uso de un microcontrolador para el uso del servidor como por la dimensión que este tiene a comparación de un servidor web convencional.

Para cumplir funciones de control y monitoreo de un host remoto conformado por el sistema microcontrolado se puede emplear la arquitectura cliente-servidor de forma básica. El sistema embebido debe cumplir funciones de red similares a la de un servidor web, el cliente solicitará ver la página HTML alojada en el servidor, siendo esta página la que contenga todos los datos que el usuario remoto quiere revisar.

3.9.1. ETHERNET EMBEBIDO

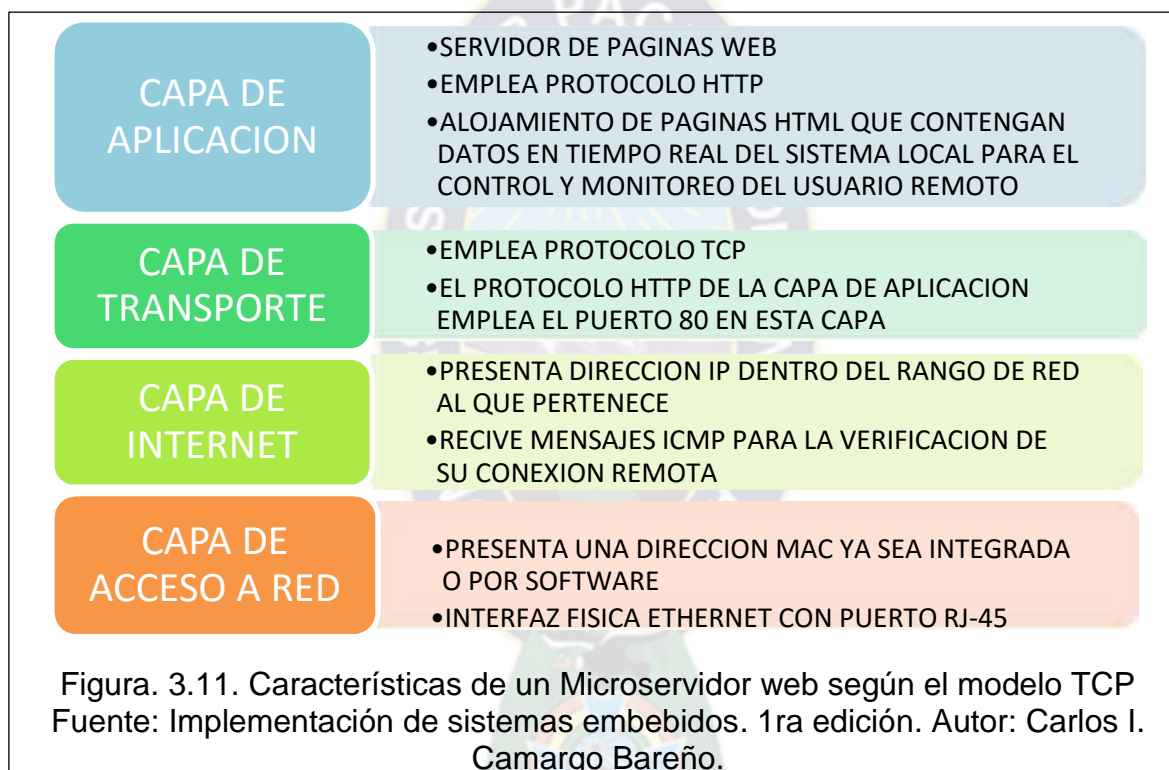
Los sistemas basados en Ethernet tienen la ventaja de poder comunicarse con diversos terminales a través de la red global, puesto que el medio físico está instalado en casi toda residencia.

Sistemas de Ethernet embebido trabajan específicamente en la comunicación de sistemas microcontrolados a través de la red Ethernet para el

monitoreo y control de sus sistemas locales. Para este fin debe trabajar al mismo nivel de un Host de red.

3.9.2. CARACTERÍSTICAS DEL MICROSERVIDOR WEB

Las características de un servidor de este tipo se pueden describir a nivel de red empleando el modelo TCP/IP para visualizar la forma de trabajo en cada capa y las funciones que cumple para un rendimiento similar al de un servidor web. En la figura 2.11 se hace la descripción en cada capa.



3.9.3. SOLUCIONES EN HARDWARE

El dispositivo a emplearse en la implementación de un Microservidor web debe resolver la problemática de brindar al sistema funciones de capa de acceso de red del modelo TCP/IP.

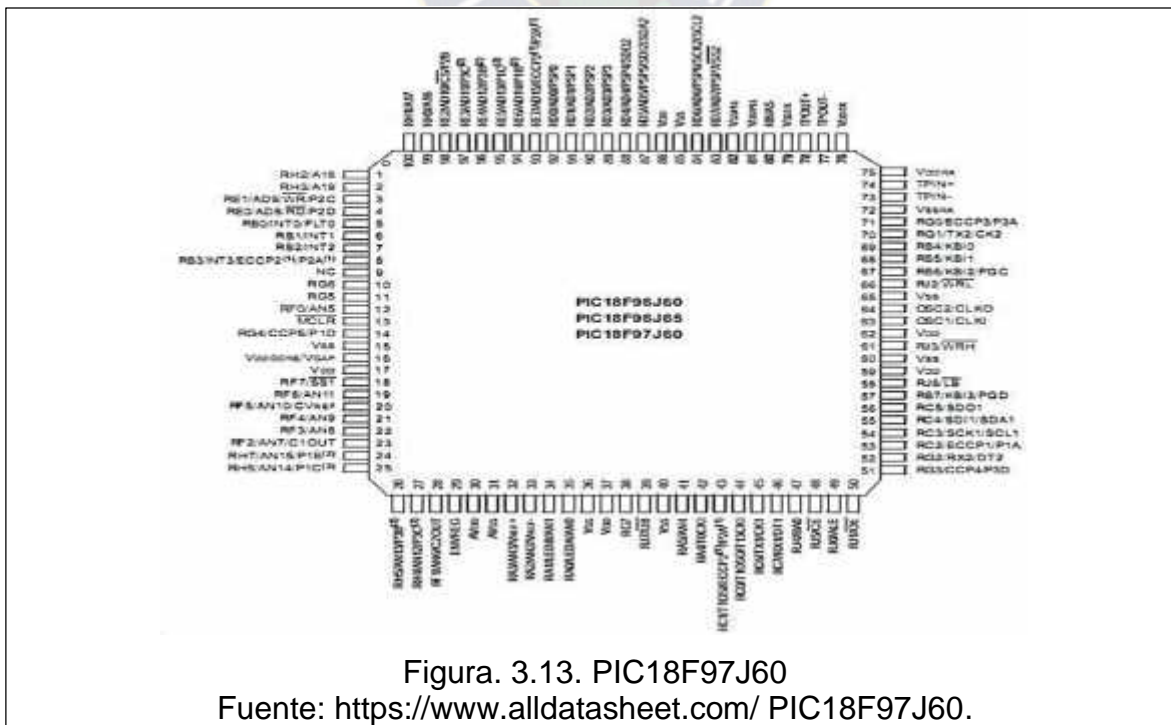
Para la conexión Ethernet física se requiere de soluciones en microelectrónica que permitan a un servidor embebido enviar la información requerida a nivel de bit por un puerto RJ-45 como lo hace cualquier host de red.

La corporación Microchip ofrece soluciones tanto en controladores externos como también puertos internos para conectar sus dispositivos con redes Ethernet.

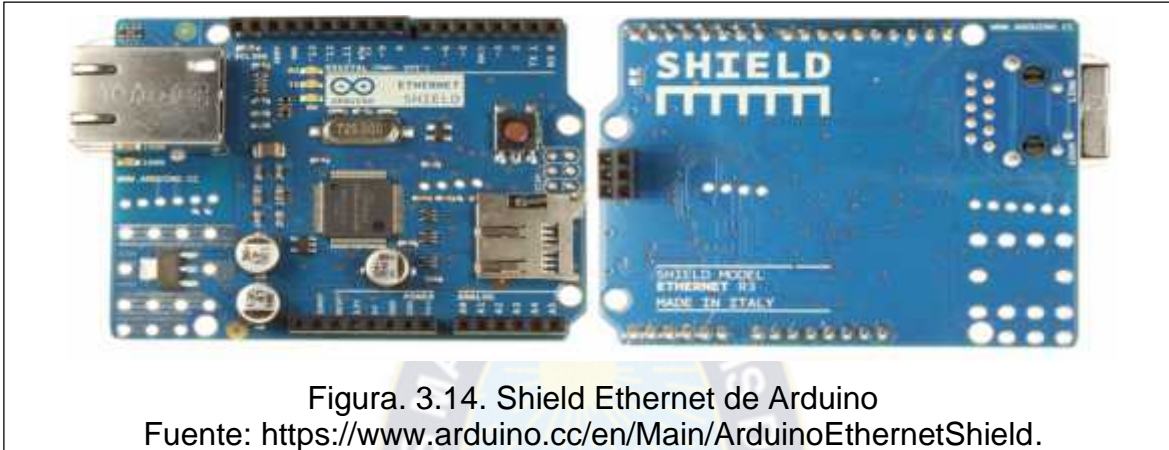
En cuanto a controladores ofrece el ENC28j60 que es un módulo que permite a los microcontroladores tener una interfaz de red de forma externa.



La solución de Microchip para microcontroladores con interfaz Ethernet incorporada de forma interna es el PIC18f97J60, el cual trabaja a 10BASE-T.



Otra opción que se usa para proyectos que tiene esta misma finalidad es la de Arduino y su Shield de Ethernet que opera mediante el chip de Wiznet. Su uso es común por su sencilla programación y por no requerir muchas nociones de Electrónica.



En sistemas más robustos y de alto rendimiento existen sistemas propietarios que emplean controladores embebido a un alto nivel, es el caso de National Instrument, que dispone del módulo de adquisición de datos PXI que presenta distintos periféricos de conexión y entre ellos Ethernet.



3.9.4. SOLUCIONES EN SOFTWARE

Para implementar un servidor web a base de microcontroladores, se debe utilizar un software de compilación que nos permita emplear códigos de programación para realizar los siguientes acciones, tales como:

- Albergar varias páginas HTML en la memoria ROM del microcontrolador.
- Disponer de librerías que permitan trabajar con conexión Ethernet en la arquitectura cliente-servidor.
- Insertar direcciones físicas (MAC) y lógicas (IP) en el firmware del microcontrolador.
- Permitir al dispositivo realizar acciones de control y monitoreo de sus interfaces digitales.

Existe gran variedad de compiladores que presentan facilidades para todo tipo de aplicación en base a microcontroladores.

La compañía MikroElektronika ofrece la plataforma de compilación MikroC basada en lenguaje C. Dispone de la librerías para la comunicación del microcontrolador a través de redes Ethernet, tanto para interfaces internas como externas, permitiendo cumplir con los objetivos de un sistema cliente-servidor.

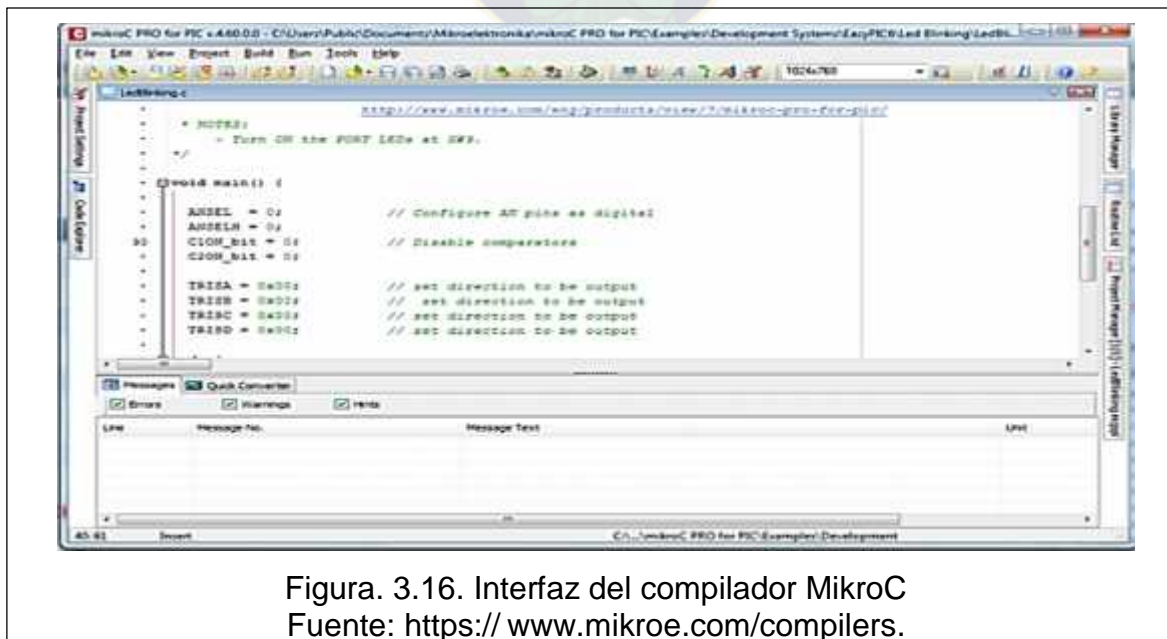


Figura. 3.16. Interfaz del compilador MikroC
Fuente: [https:// www.mikroe.com/compilers](https://www.mikroe.com/compilers).

Otra de las opciones en software de compilación es la solución que brinda la plataforma Arduino, la cual también dispone de librerías que permiten trabajar al dispositivo a nivel de red.

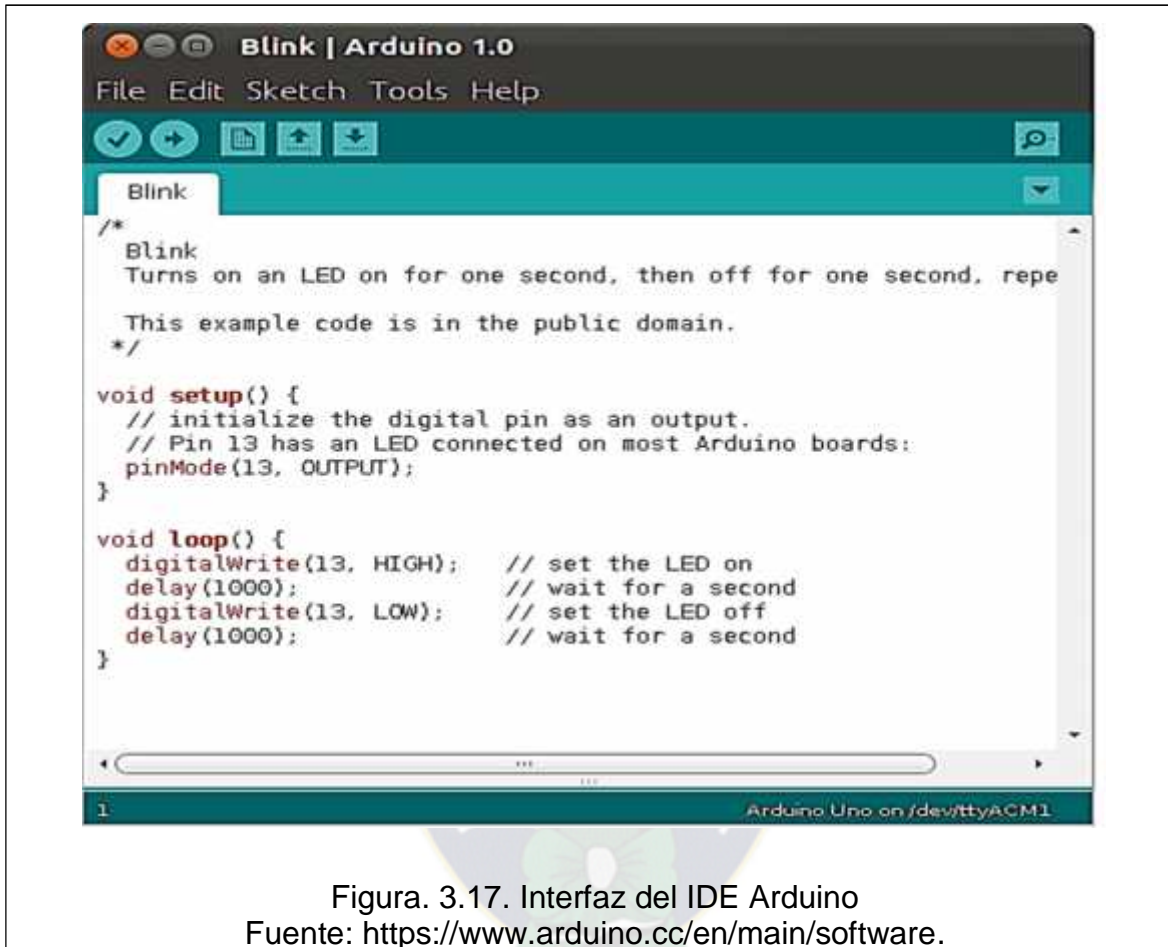


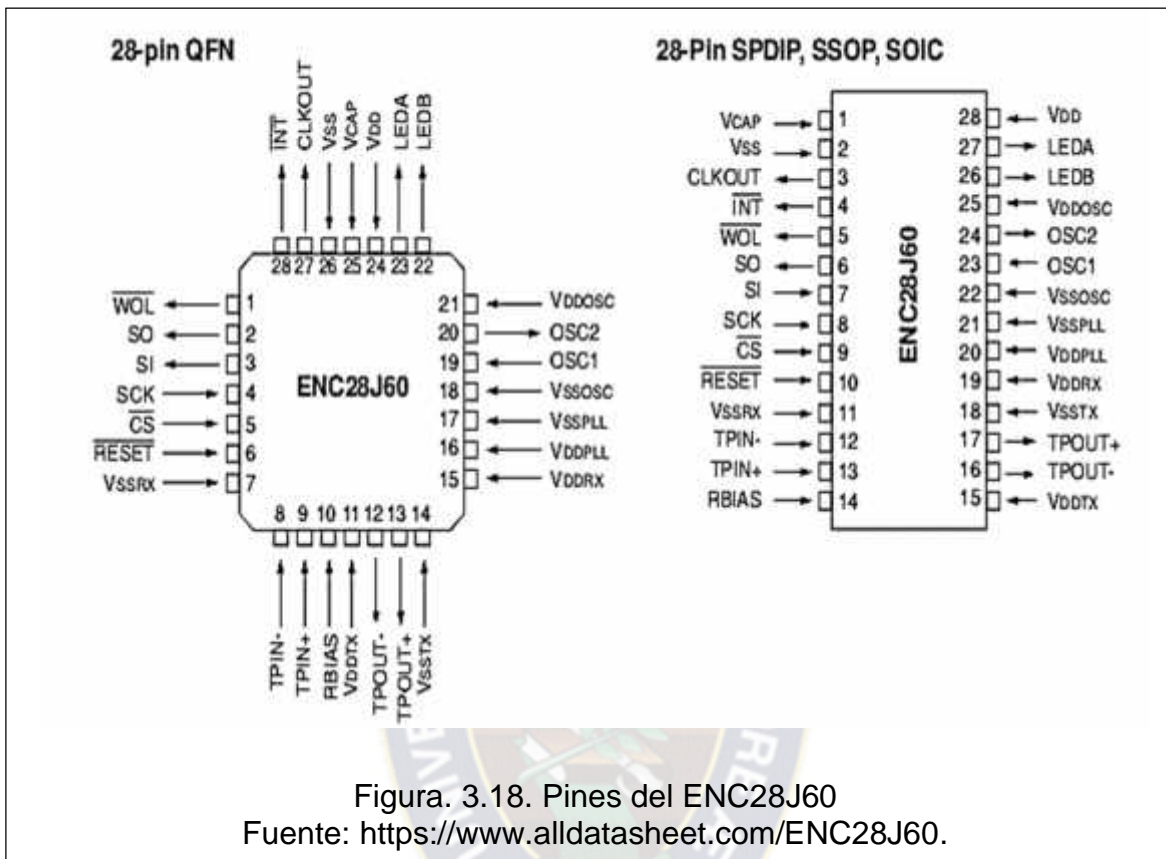
Figura. 3.17. Interfaz del IDE Arduino
Fuente: <https://www.arduino.cc/en/main/software>.

3.10. CONTROLADOR ENC28J60

El ENC28J60 es un controlador Ethernet dotado de un puerto SPI especialmente diseñado para actuar como un puente entre una red Ethernet y un microcontrolador equipado con SPI. Las características más relevantes de este dispositivo son las siguientes:

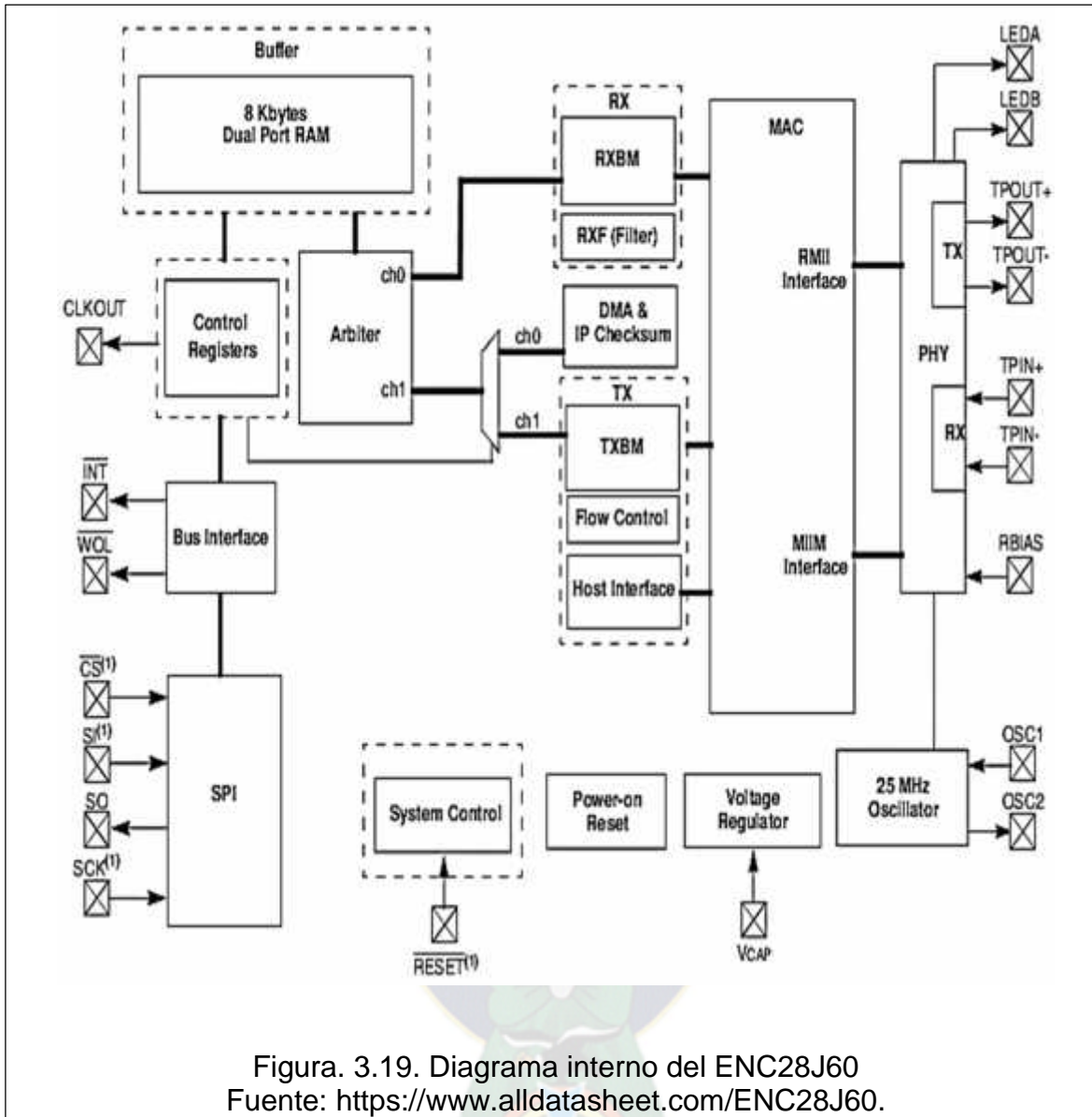
- Cumple con las especificaciones IEEE 802.3
- Tecnología 10BASE-T
- Trabaja en Full/Half Duplex

- Frecuencia de trabajo de 25 Mhz mediante cristal externo
- Voltaje de alimentación de 3.24 a 3.45 V.



El EN28J60 está conformado por siete bloques funcionales importantes los cuales son los siguientes:

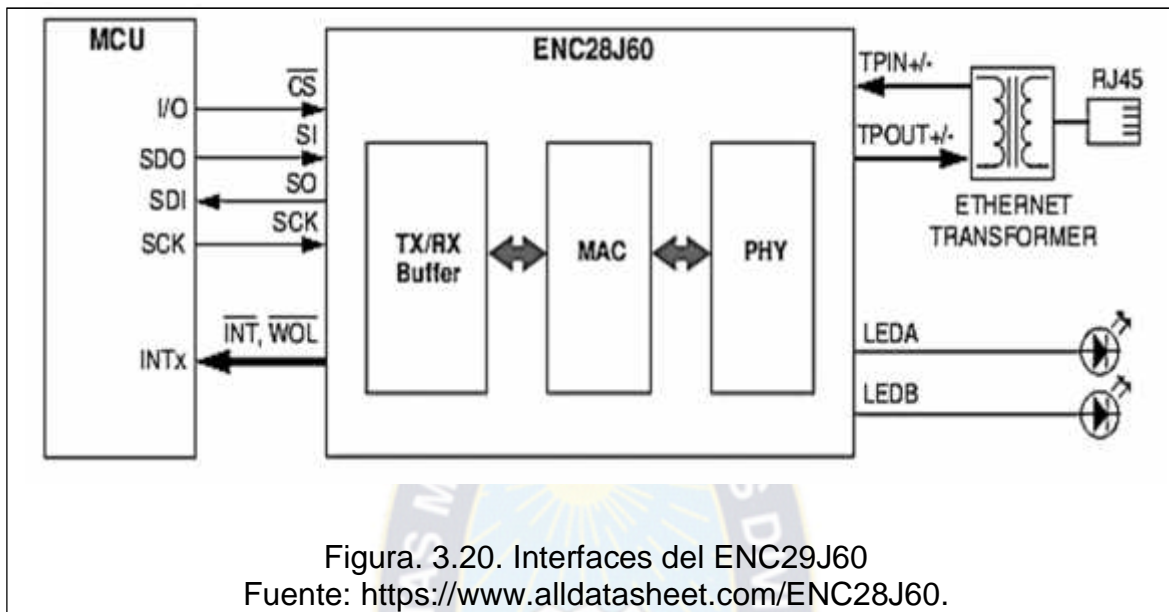
- La interfaz SPI que se encarga de comunicarlo con el microcontrolador host.
- Los registros que se emplean para monitorear y controlar al ENC28J60.
- Un buffer RAN dual para los paquetes recibidos y enviados.
- Un arbitro que controla los accesos al buffer RAM mediante DMA.
- Un bus que interpreta los datos y comandos recibidos vía SPI.
- Un módulo MAC (Maduim Access Control) que cumple con IEEE 802,3.
- Una módulo PHY (Physical Layer) que codifica y decodifica los datos análogos presentes en el par trenzado.



En la figura 2.19 se da a conocer un esquema interno básico. La interfaz SPI se conecta con el microcontrolador que cumplirá el rol de servidor web. Los Buffers de transmisión y recepción enviarán y recibirán la información del bloque SPI.

El módulo MAC proveerá de una dirección física al host empleando una dirección asignada mediante software, ya que no tiene una dirección interna fija. Para la conexión física a la red se emplea el módulo PHY que se conectará con el puerto RJ-45 con los arreglos circuitales necesarios. La señalización integrada al

dispositivo se presenta en dos leds A y B para indicar que existe conexión y el estado de la transmisión respectivamente.



3.11. PIC18F4620

Es un microcontrolador Microchip de gama alta. Las características más importantes de este dispositivo son las siguientes:

- Memoria de programa ROM FLASH de 64Kb
- Memoria RAM de 3.968 Bytes
- Presenta cuatro modos de oscilación por cristal hasta de 40 Mhz
- CPU que trabaja hasta 10 MIPS
- Bajos consumos de corriente en modos Idle y SLEEP de 5.8 y 0.1 uA respectivamente.
- Tiene 12 puertos analógicos, con 10 bits de resolución y hasta 100K muestras por segundo.
- Cuenta con 2 comparadores analógicos.
- Periféricos de comunicación: 1-UART, 1-A/E/USART, 1-SPI, 1-I2C1-MSSP.

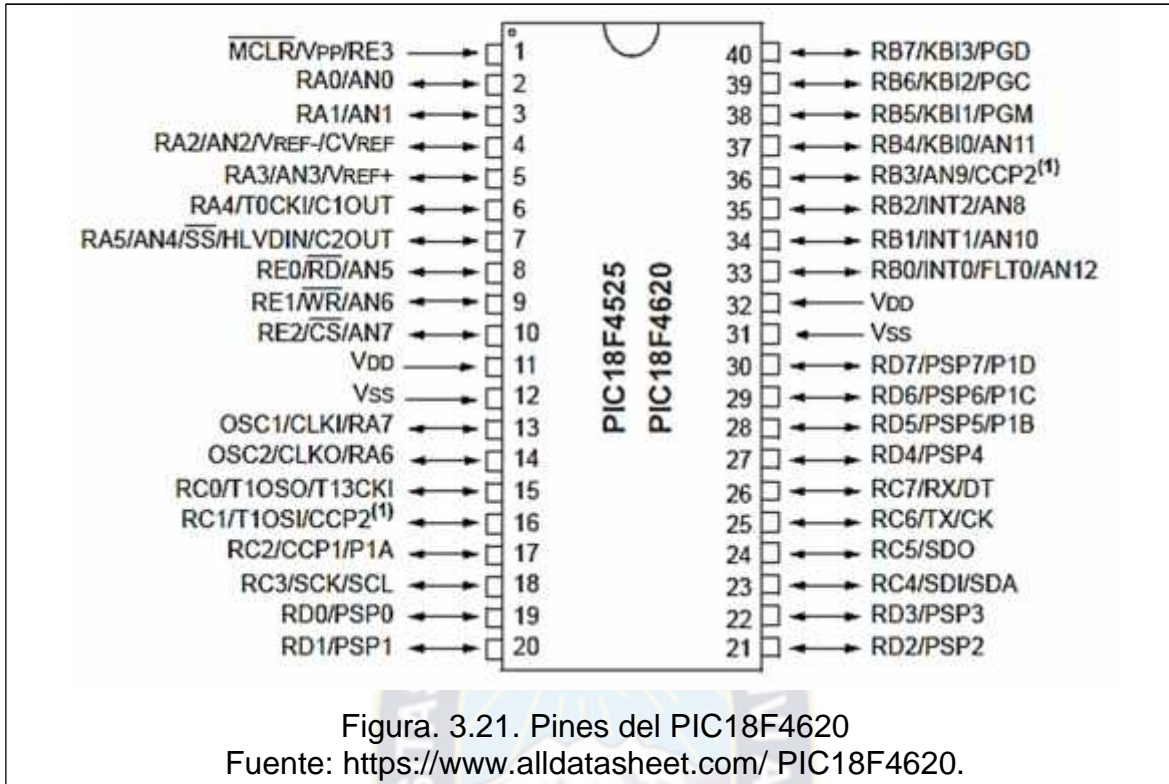


Figura. 3.21. Pines del PIC18F4620

Fuente: <https://www.alldatasheet.com/PIC18F4620>.

La característica principal de este microcontrolador para el desarrollo de aplicaciones orientadas a la red es su alta capacidad de almacenamiento en su memoria RAM, puesto que a comparación de los de gama media cuyo promedio es de 8Kb, este cuenta con 64Kb. El mismo es de gran ayuda para el almacenamiento de Firmwares con gran cantidad de código como el requerido en estos sistemas.

CAPITULO IV

4. MARCO TEORICO: TECNOLOGIA BLUETOOTH Y APLICACIONES ANDROID

4.1. REDES DE AREA PERSONAL PAN

Este tipo de redes nace de la necesidad de permitir que una persona pueda conectar la mayor cantidad de dispositivos electrónicos, conocidos como gadgets, a su celular de manera simple. La popularización de las PAN se dio gracias a Bluetooth y a la sencillez que se daba para conectar varios accesorios a los celulares.

Las PAN son también conocidas como WPAN (Wireless Personal Area Network) que hace más evidente la característica inalámbrica de la red pero que indican el mismo concepto de conectar dispositivos cercanos a la persona, establecido como el estándar IEEE 802.15. El alcance puede variar dependiendo de la tecnología que se utilice pero generalmente permiten alcanzar desde diez a treinta metros. También de acuerdo a la tecnología se tiene la frecuencia a utilizar, aunque 2.4 Ghz parece ser la más prometedora y la que más avances ha tenido gracias a la tecnología Bluetooth. En la tabla 4.1 se presenta una comparativa entre tres diferentes tipos de tecnología definidas en IEEE 802.15.

	Bluetooth 802.15.1	UWB 802.15.3	ZigBee 802.15.4
Frecuencia	2.4 - 2.48GHz	3.1-10.6GHz	868MHz 902-928MHz 2.4-2.48GHZ
Alcance	10 m	10 m	100 m
Tasa de Transferencia	3 Mbps	1 Gbps*	40 – 50 Kbps
Modulación	GFSK	BPSK	BPSK

Tabla. 4.1. Comparación de tecnologías inalámbricas
Fuente: <http://www.emb.cl/electroindustria/articulo.mvc>.

4.2. BLUETOOTH

Bluetooth es una especificación abierta de tecnología de radio de corto alcance de área personal (WPAN) que posibilita la transmisión de voz y datos entre diferentes dispositivos mediante un enlace por radiofrecuencia en la banda de los 2,4 GHz.

Los dispositivos que con mayor frecuencia utilizan esta tecnología pertenecen a sectores de las telecomunicaciones y la informática personal, como PDA's, teléfonos, ordenadores personales, impresoras o cámaras digitales, entre otros. Los objetivos que esta tecnología busca conseguir se pueden resumir en los siguientes puntos:

- Facilitar las comunicaciones entre equipos móviles y fijos.
- Eliminar cables y conectores entre éstos.
- Ofrecer la posibilidad de crear pequeñas redes inalámbricas y facilitar la sincronización de datos entre equipos personales.

4.2.1. CLASES DE BLUETOOTH

Las comunicaciones se realizan por radiofrecuencia de forma que los dispositivos no tienen que estar alineados y pueden incluso estar en habitaciones separadas si la potencia de transmisión es suficiente. Sin embargo se deben considerar los alcances máximos en distancia y los niveles de potencia que pueden alcanzar según sus requerimientos.

Es por ello que estos dispositivos se clasifican en tres clases en referencia a su potencia de transmisión, siendo totalmente compatible la comunicación de dispositivos de distinta clase. En la tabla 4.2 se describe la clasificación de dispositivos Bluetooth según el nivel de potencia de transmisión y distancia máxima.

Clase	Máxima potencia (W)	Distancia (m)
Clase 1	100 mW	100
Clase 2	2.5 mW	22
Clase 3	1 mW	6

Tabla. 4.2. Clases de dispositivos Bluetooth según su nivel de potencia de transmisión y distancia

Fuente: <http://www.bluetooth.com/Pages/Basics.aspx>.

También se puede clasificar al dispositivo Bluetooth según la versión que presenta, siendo el principal parámetro de referencia su ancho de banda.

Versión	Ancho de banda
Versión 1.2	1 Mbit/s
Versión 2.0 + EDR	3 Mbit/s
Versión 3.0 + HS	24 Mbit/s
Versión 4.0	24 Mbit/s

Tabla. 4.3. Versiones Bluetooth

Fuente: <http://www.bluetooth.com/Pages/Basics.aspx>.

4.2.2. CARACTERÍSTICAS EN RADIOFRECUENCIA

Bluetooth fue diseñado para operar en un entorno de radio frecuencia ruidosa y para ello utiliza un esquema de reconocimiento rápido y saltos de frecuencia para garantizar la robustez del enlace.

Opera en la banda de frecuencia de no regulada ISM de 2.4 GHz, el rango de frecuencias es desde 2.4 hasta 2.48 GHz, con 79 canales de frecuencias de radio de 1MHz.

La modulación que emplea Bluetooth es GFSK (Gaussian Frequency Shift Keying). El índice de modulación debe estar entre 0.28 y 0.35. Un uno binario se

representa por una desviación positiva de frecuencia y un cero binario como una desviación negativa. La desviación mínima no ha de ser menor de 115 KHz.

El aspecto más importante en el dispositivo receptor es el nivel de sensibilidad. Para poder medir una tasa de error de bit, el equipo receptor envía de vuelta la información decodificada. Para una tasa de error o BER del 0.1% se define el nivel de sensibilidad de un receptor Bluetooth mayor o igual a -70dBm .

Debido a que la banda ISM está abierta a cualquier sistema de radio el dispositivo Bluetooth deberá estar preparado para evitar múltiples interferencias. Por esta se emplea la técnica de acceso al medio FHSS (transmisión de ensanchamiento del espectro por saltos de frecuencia), para minimizar interferencias y mejorar el nivel de seguridad.

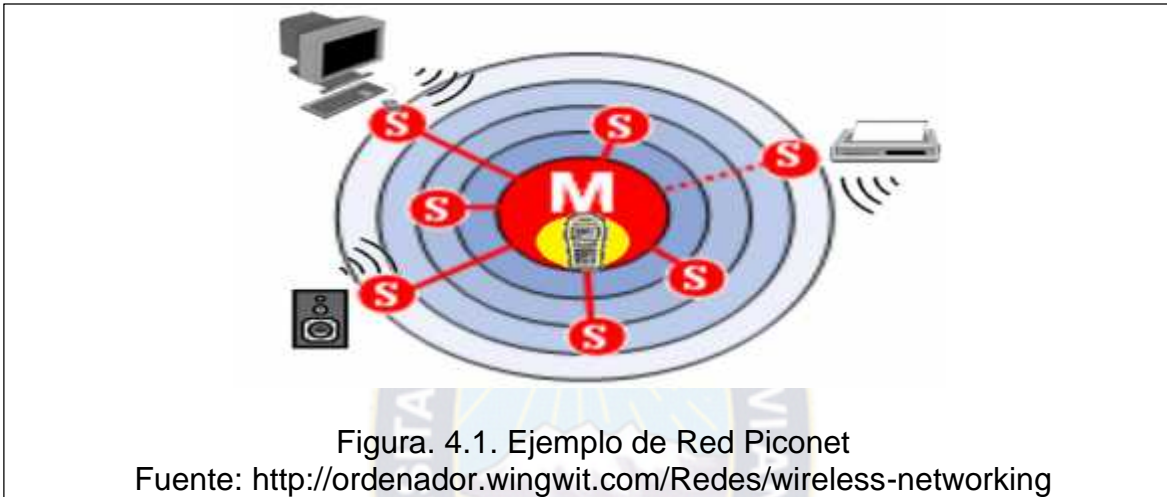
Dada la existencia de 79 canales en el espectro a utilizarse, se van realizando saltos de frecuencia cada 625 us de manera pseudo-aleatoria, reduciéndose el ancho de banda instantáneo con pocas posibilidades de interferir un canal ya ocupado.

Parámetro	Valor
Tipos de antena	Omnidireccional
Rango de Frecuencias	2,4 GHz a 2,483 GHz.
Modulación	GFSK
Acceso al medio	FHSS
Nº de Canales	79
Banda del Canal	1 MHz
Potencia de transmisión	Clase 1: 100 mW Clase 2: 2,5 mW (típica) Clase 3: 1 Mw

Tabla. 4.4. Características de la tecnología Bluetooth en radiofrecuencia
Fuente: <http://www.bluetooth.com/Pages/Basics.aspx>.

4.2.3. PRINCIPIOS DE COMUNICACIÓN

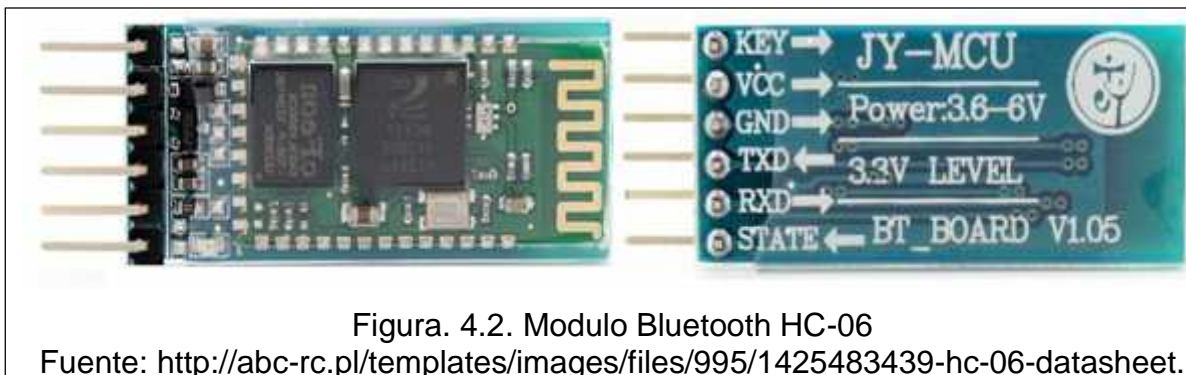
Se emplea una Piconet, la cual es una red Bluetooth que opera en modo maestro y esclavos, formada por hasta siete dispositivos esclavos y uno maestro. Todos los dispositivos están sincronizados al reloj y secuencia de salto de frecuencia del maestro como se ve en la figura 4.1.



En realidad, en un momento determinado, el dispositivo maestro sólo puede conectarse con un solo esclavo. Por lo tanto, rápidamente cambia de esclavo para aparentar una conexión simultáneamente con todos los dispositivos.

4.3. MODULO HC-06

El HC-06 es un módulo Bluetooth Maestros/Esclavo HC-06 utiliza el protocolo UART para puertos con RS-232. Tiene un coste reducido y bajo consumo tanto en funcionamiento como en modo reposo.



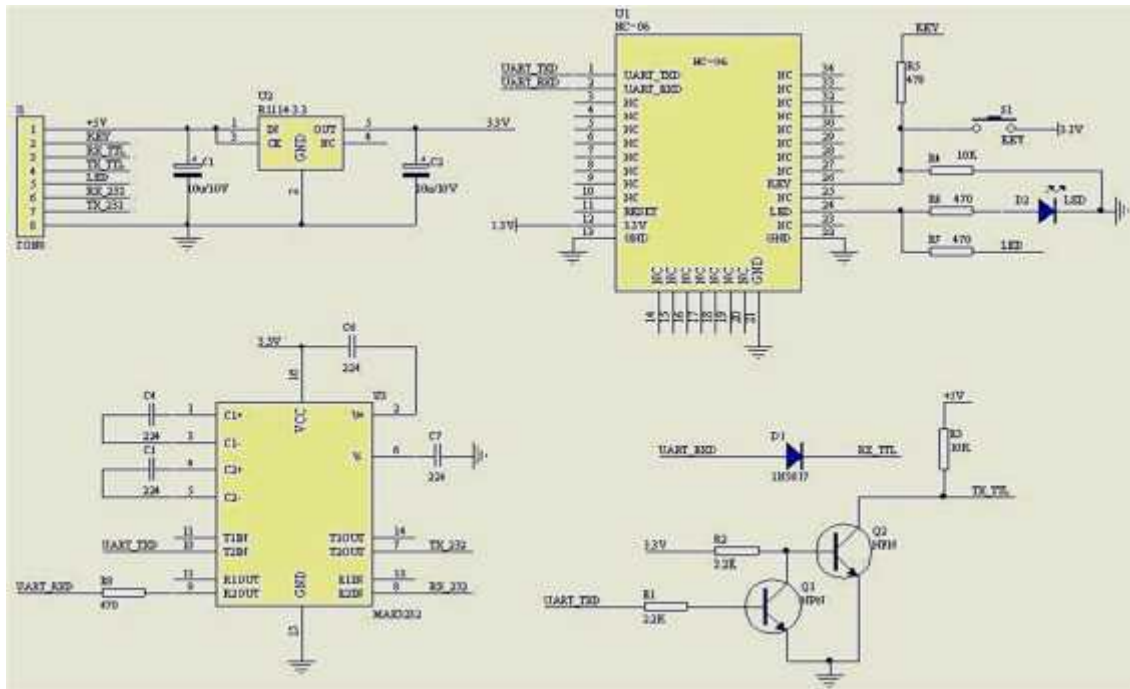


Figura. 4.3. Diagrama esquemático del módulo HC-06

Fuente: <http://abc-rc.pl/templates/images/files/995/1425483439-hc-06-datasheet>.

Característica	Valor
Voltaje de entrada	3.6V a 6V
Corriente de operación	< 40mA
Corriente modo suspendido	< 1mA
Potencia de salida	4dBm, clase 2
Sensibilidad	-84dBm
Versión	Versión 2.0 + EDR
Seguridad	Autenticación y cifrado
Frecuencia de operación	2.4GHz (banda ISM)
Modulación	GFSK
Velocidad Asíncrona	2.1Mbps(máxima)/900Kbps
Velocidad Síncrona	1.5Mbps/1.5Mbps
Temperatura de operación	-40°C a 86°C
Rango de alcance	Clase 2 (22 metros)

Tabla. 4.5. Características técnicas del módulo HC-06

Fuente: <http://abc-rc.pl/templates/images/files/995/1425483439-hc-06-datasheet>.

4.3.1. COMANDOS DE CONFIGURACION

Por fabrica inicialmente el modulo está configurado como esclavo, y requiere de comandos AT para configurar parámetros. La tabla 4.6 muestra los comandos de consulta y configuración para el HC-06.

Instrucción	Respuesta del HC-06	Descripción
AT	OK	Verificación de respuesta
AT+VERSION?	OKLinvorV1.8	Consulta de la versión del Firmware que está utilizando
AT+NAME?	HC-06	Consulta de nombre actual del dispositivo
AT+NAME=X	OK	Asignamos un nombre al dispositivo con el que será visible para los dispositivos externos.
AT+BAUD4	OK	Se configura el baud rate a 9600 típico en comunicación serial
AT+PIN2314	OK	Se asigna el código de autenticación para la vinculación con otro dispositivo, el cual debe ser de 4 dígitos.

Tabla. 4.6. Comandos de configuración del módulo HC-06

Fuente: <http://abc-rc.pl/templates/images/files/995/1425483439-hc-06-datasheet>

4.4. SISTEMA OPERATIVO ANDROID

Android es un sistema operativo basado en el núcleo Linux. Fue diseñado principalmente para dispositivos móviles con pantalla táctil, como teléfonos inteligentes, tabletas, relojes inteligentes, televisores y automóviles. Inicialmente fue desarrollado por Android Inc., empresa que Google respaldó económicamente y más tarde, en 2005, la compró.

El primer móvil con el sistema operativo Android fue el HTC Dream y se vendió en octubre de 2008. Los dispositivos de Android venden más que las ventas combinadas de Windows Phone y IOS.

Se puede resumir las principales características de este sistema operativo en el siguiente listado.

- Emplea código abierto.
- Núcleo basado en el Kernel de Linux.
- Adaptable a muchas pantallas y resoluciones.
- Utiliza SQLite para el almacenamiento de datos.
- Ofrece diferentes formas de mensajería.
- Navegador web basado en WebKit incluido.
- Soporte de Java y muchos formatos multimedia.
- Soporte de HTML, HTML5, Adobe Flash Player, etc.
- Catálogo de aplicaciones gratuitas o pagas en el que pueden ser descargadas e instaladas (Google Play).
- Multitarea real de aplicaciones.

Existen varias versiones del sistema operativo desde la 1.0 hasta la 5.0, las cuales mejoran en rendimiento, seguridad, estética y compatibilidad de aplicaciones. Cada versión viene de fábrica con el dispositivo inteligente, la cual no presenta opciones de actualización.

4.5. DESARROLLO DE APLICACIONES ANDROID

El Desarrollo de programas que se ejecuten en este sistema operativo se hacen habitualmente en código Java con el conjunto de herramientas de desarrollo SDK, pero hay otras opciones disponibles. La plataforma Android ha crecido hasta ser una de las preferidas por los desarrolladores para plataformas móviles.

Existen diversos entornos de trabajo en software para la programación de aplicaciones para Android, la más popular es la IDE llamada Eclipse. La misma

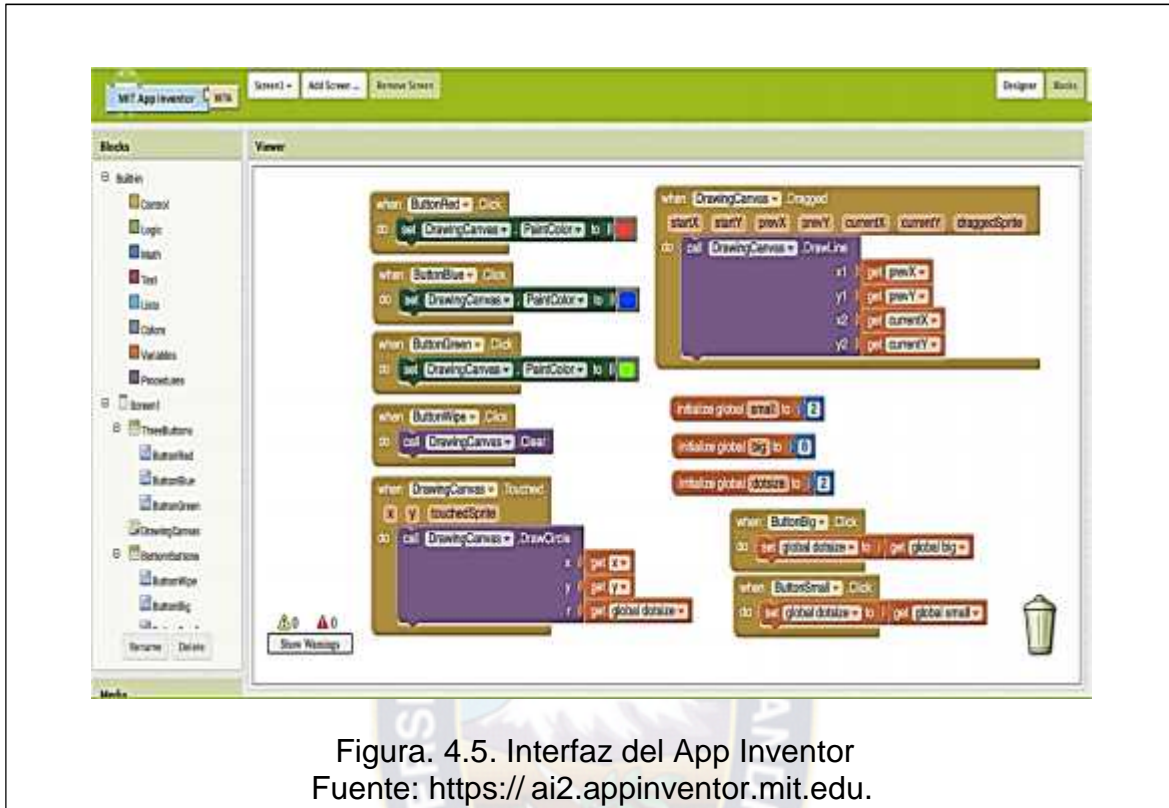
requiere de un Plugin para la programación compatible con las distintas versiones de este sistema operativo móvil.



Una vez terminada la programación de la aplicación en cualquier software de desarrollo se compila el mismo en un archivo ejecutable con extensión APK (Application Package File). Este formato es una variante del formato JAR de Java y se usa para distribuir e instalar componentes empaquetados para la plataforma Android para Smartphones y tabletas.

4.6. APP INVENTOR

Es una plataforma de Google para crear aplicaciones de software para el sistema operativo Android. De forma visual y a partir de un conjunto de herramientas básicas, el usuario puede ir enlazando una serie de bloques para crear la aplicación. El desarrollo se lo realiza mediante la web desde el navegador.



App Inventor se ejecuta como un servicio web administrado por personal del Centro del MIT para el aprendizaje móvil. Es compatible con una comunidad mundial de casi dos millones de usuarios que representan a 195 países en todo el mundo. Más de 85 mil usuarios semanales han construido más de 4,7 millones de aplicaciones de Android. Una herramienta de código abierto que pretende realizar la programación y la creación de aplicaciones accesibles a una amplia gama de audiencias.

Los requisitos que requiere en hardware es solo que el equipo cuente con 250 MB de RAM como mínimo. En software solo requiere de tener instalado el JDK de Java 1.6 y contar con un navegador web.

CAPITULO V

5. INGENIERIA DEL PROYECTO

5.1. DIAGRAMA EN BLOQUES

A lo largo de este capítulo veremos todos los aspectos relacionados con el diseño en hardware del sistema domóticos para el control y monitoreo de la vivienda de forma remota a través de Internet y de forma local mediante la interfaz Bluetooth.

También se analizará los Firmwares que presentan los microcontroladores mediante diagramas de flujo que describan el funcionamiento de los subsistemas.

A continuación se muestra el diagrama de bloque del Sistema Domótico implementado en Microservidor web embebido con conexión inalámbrica para Smartphones.

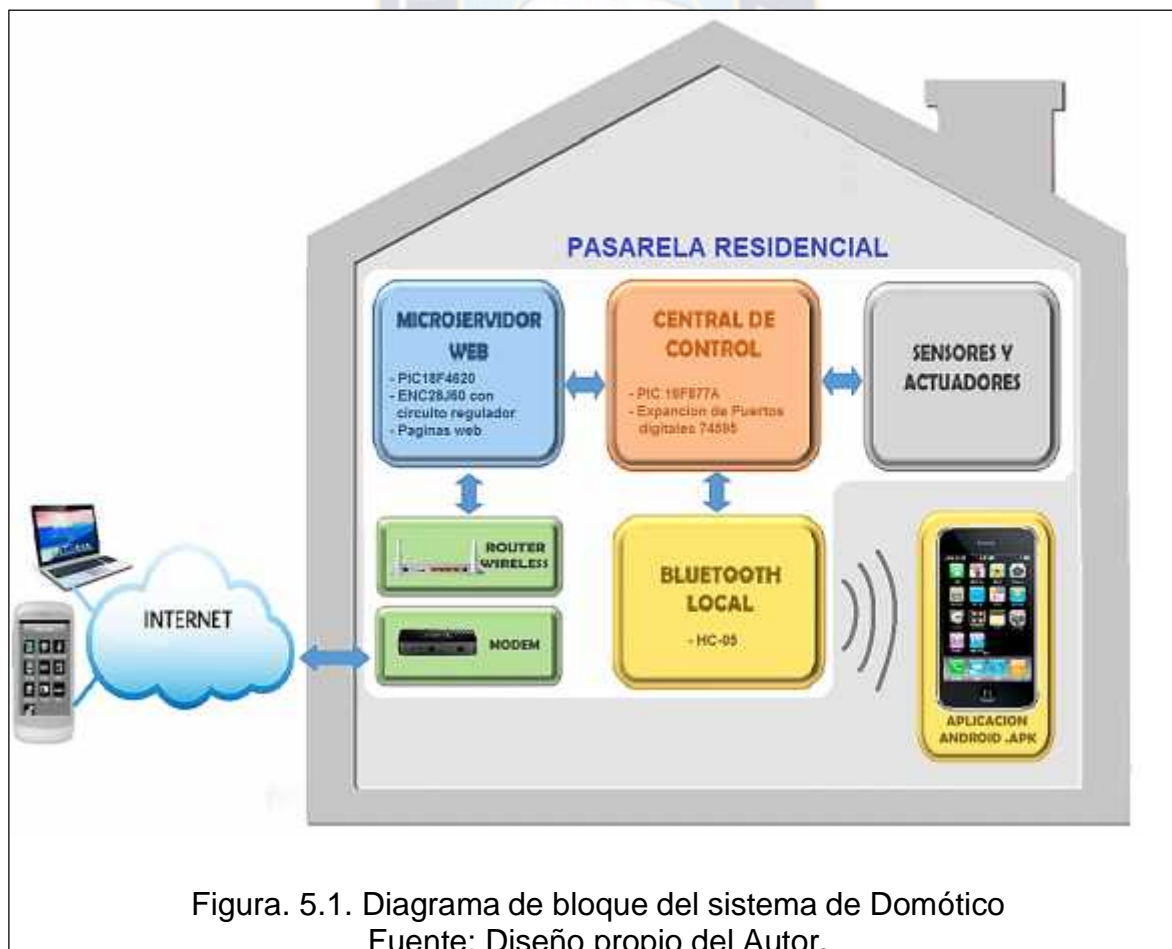


Figura. 5.1. Diagrama de bloque del sistema de Domótico
Fuente: Diseño propio del Autor.

5.2. ANALISIS PREVIO CON EL MODELO DE TRES NIVELES DE COMPLEJIDAD

Para definir la funcionalidad objetiva del sistema que se va diseñar e implementar en Domótica, es de gran utilidad emplear el modelo de tres niveles de complejidad, para separar aspectos como ser servicios, aplicaciones, dispositivos y la interconexión de los mismos en el sistema en su conjunto.

Es por ello que se definirá cada nivel desde el tercero al primero para delimitar aspectos técnicos específicos que se abordaran con el presente sistema propuesto.

5.2.1. TERCER NIVEL CON SERVICIOS Y APLICACIONES

En este nivel se define los servicios que ofrecerá el presente sistema domótico especificando los alcances funcionales que presenta para un rendimiento óptimo, los cuales son los siguientes:

- Seguridad perimetral: El acceso al perímetro delimitado por la casa y su interior está definido por la apertura de puertas y ventanas, las cuales tendrán sensores que detectaran cambios de estado de los mismos, alertando en ese instante al usuario.
- Seguridad de intrusión: En caso de que cualquier intruso este dentro de la vivienda sin autorización se empleará sensores de presencia que detecten cambios volumétricos en áreas estratégicas para alertar al usuario.
- Seguridad técnica: Se alertará al usuario en caso de incendio mediante alertas a la interfaz remota y tomando acciones de forma automática como ser la activación de extintores de agua.
- Seguridad pasiva: Se tendrá la posibilidad de emplear el servicio de simulación de presencia en la vivienda para que el usuario tenga la seguridad de que intrusos no se percaten de su ausencia.

- **Accesibilidad remota:** Se dispondrá de un acceso cómodo empleando la red de Internet para monitorear y controlar la vivienda desde su dispositivo terminal utilizando un navegador web convencional.
- **Accesibilidad local:** Los Smartphones del usuario serán capaces de conectarse localmente al sistema mediante una aplicación con extensión .apk con interfaz Bluetooth y tener la posibilidad de tomar acciones desde su terminal de forma cómoda.
- **Ahorro energético:** Gracias al fácil acceso se podrá desactivar la iluminación y controlar tomas de corriente sin tener que estar físicamente en la vivienda.
- **Portero automático integrado:** El control del acceso al garaje se lo realizara desde el interior de la vivienda sin la intervención directa del usuario, empleando únicamente las interfaces de usuario.

5.2.2. SEGUNDO NIVEL DE INTERCONEXION DE EQUIPOS

En este nivel se define la interconexión que empleará el sistema con el exterior e interior de la vivienda, que de forma básica puede resumirse en la figura 5.2.

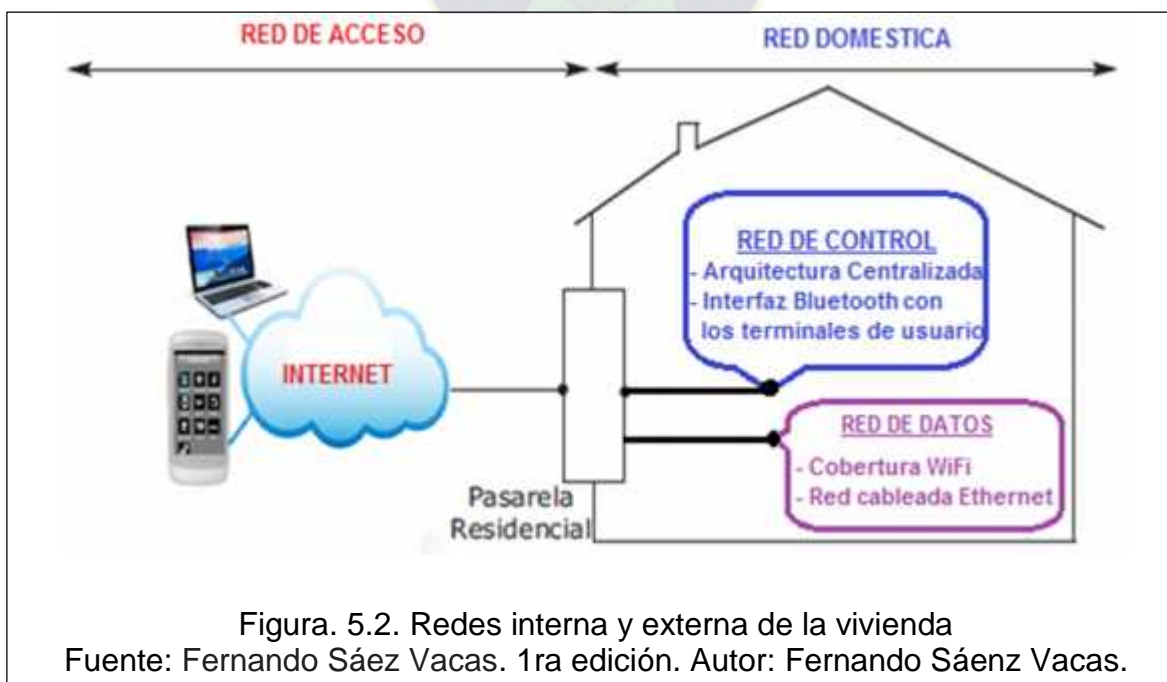


Figura. 5.2. Redes interna y externa de la vivienda

Fuente: Fernando Sáez Vacas. 1ra edición. Autor: Fernando Sáenz Vacas.

La interconexión de la pasarela residencial con la red de acceso se realizara a través de Internet empleando un proveedor de servicios ISP, el cual además de proveer el servicio de datos se lo empleará para el acceso del usuario remoto al sistema domótico.

La red doméstica está compuesta por la red de datos y la red de control, siendo esta ultima la que se implementará con el sistema domótico propuesto. La arquitectura de la red de control será centralizada, para tener todos los sensores y actuadores que están distribuidos en la vivienda se conecten a un controlador principal ubicado en la pasarela residencial.

Los datos actuales que recogen los sensores serán presentados al usuario por medio de una interfaz remota y local. El acceso local del usuario se realizara a través de la interfaz Bluetooth empleando una aplicación instalada en los Smartphones de los habitantes de la vivienda.

5.2.3. PRIMER NIVEL DE SENSORES Y ACTUADORES

Conociendo los servicios que ofrecerá el sistema domótico, la arquitectura de la red de control y las interfaces de usuario remoto y local se define los sensores y actuadores que se emplearán, los cuales deben cumplir roles específicos.

Sensores discretos como también analógicos son requeridos dependiendo de la variable de control. De igual forma los actuadores discretos requeridos dependerán de la acción que requiere realizarse.

La cantidad total de sensores y actuadores dependerá de la dimensión de la vivienda, pero el servicio tendrá de forma definida los elementos mínimos que requiere, los mismos se detallan en la tabla 5.1.

Servicio	Sensores	Actuadores
Seguridad perimetral	<ul style="list-style-type: none"> • Sensores magnéticos para puertas y ventanas • Sensor de vibración para detección de forzado de puerta 	<ul style="list-style-type: none"> • Bocina de alarma • Luces de alerta
Seguridad de intrusión	<ul style="list-style-type: none"> • Sensor volumétrico • Sensor acústico 	<ul style="list-style-type: none"> • Bocina de alarma • Luces de alerta
Seguridad técnica	<ul style="list-style-type: none"> • Detector de incendio 	<ul style="list-style-type: none"> • Bocina de alarma • Luces de alerta
Seguridad pasiva	<ul style="list-style-type: none"> • No requiere por ser activado por el usuario 	<ul style="list-style-type: none"> • Luces incandescentes internas y externas
Accesibilidad remota y local	<ul style="list-style-type: none"> • Sensor de temperatura ambiente • Todos los sensores de los demás servicios son visualizados por las interfaces remota y local 	<ul style="list-style-type: none"> • Luces incandescentes RGB • Todos los actuadores de los demás servicios son controlados por las interfaces remota y local
Ahorro energético	<ul style="list-style-type: none"> • No requiere por ser activado por el usuario 	<ul style="list-style-type: none"> • Interruptor de tomas de corriente • Luces incandescentes internas y externas
Portero automático	<ul style="list-style-type: none"> • Activado por el usuario 	<ul style="list-style-type: none"> • Motor de apertura

Tabla. 5.1. Requerimiento básico de sensores y actuadores para cada servicio
Fuente: Diseño propio del Autor.

5.3. SUBSISTEMA MICROSERVIDOR WEB

El acceso remoto del usuario al control y monitoreo de la vivienda lo realizara empleando un navegador web en su dispositivo terminal, siendo una ventaja para el usuario al no requerir de un software adicional instalado en su dispositivo. Esto lleva a requerir de una interfaz web que trabaje en una arquitectura cliente servidor, es por ello que se emplea un servidor local web en la vivienda, que muestre el estado actual de la vivienda y permita realizar acciones en tiempo real.

Este subsistema presenta elementos en hardware y en software que permiten que realice la función de servidor local de páginas web con interfaz física a redes de computadoras. En el mismo se alojara las páginas web que el usuario empleara para observar estados de los sensores y controlar los actuadores en la casa.

En hardware se tiene dispositivos que son de gran relevancia para cumplir estas funciones. El PIC18F4620 es en esencia el servidor web de la vivienda y el ENC28J60 es el dispositivo que permite al subsistema acceder de forma física a la red Ethernet.

En software se tiene al Firmware del PIC18F4620 que tiene embebidas a las páginas web que se emplearan como interfaz de usuario remoto y las direcciones IP lógica y MAC física que identifican al sistema.

5.3.1. DIAGRAMA DEL CIRCUITO

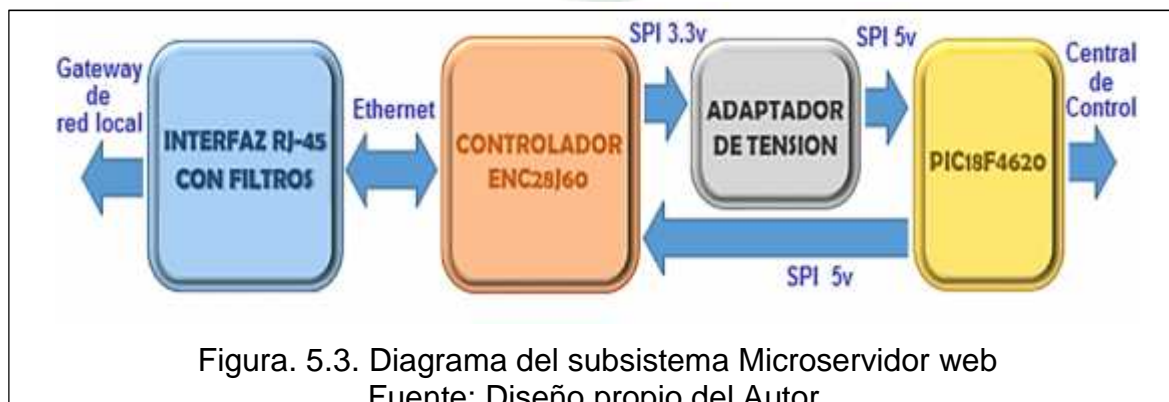


Figura. 5.3. Diagrama del subsistema Microservidor web
Fuente: Diseño propio del Autor.

5.3.2. CIRCUITO DEL MICROSERVIDOR WEB

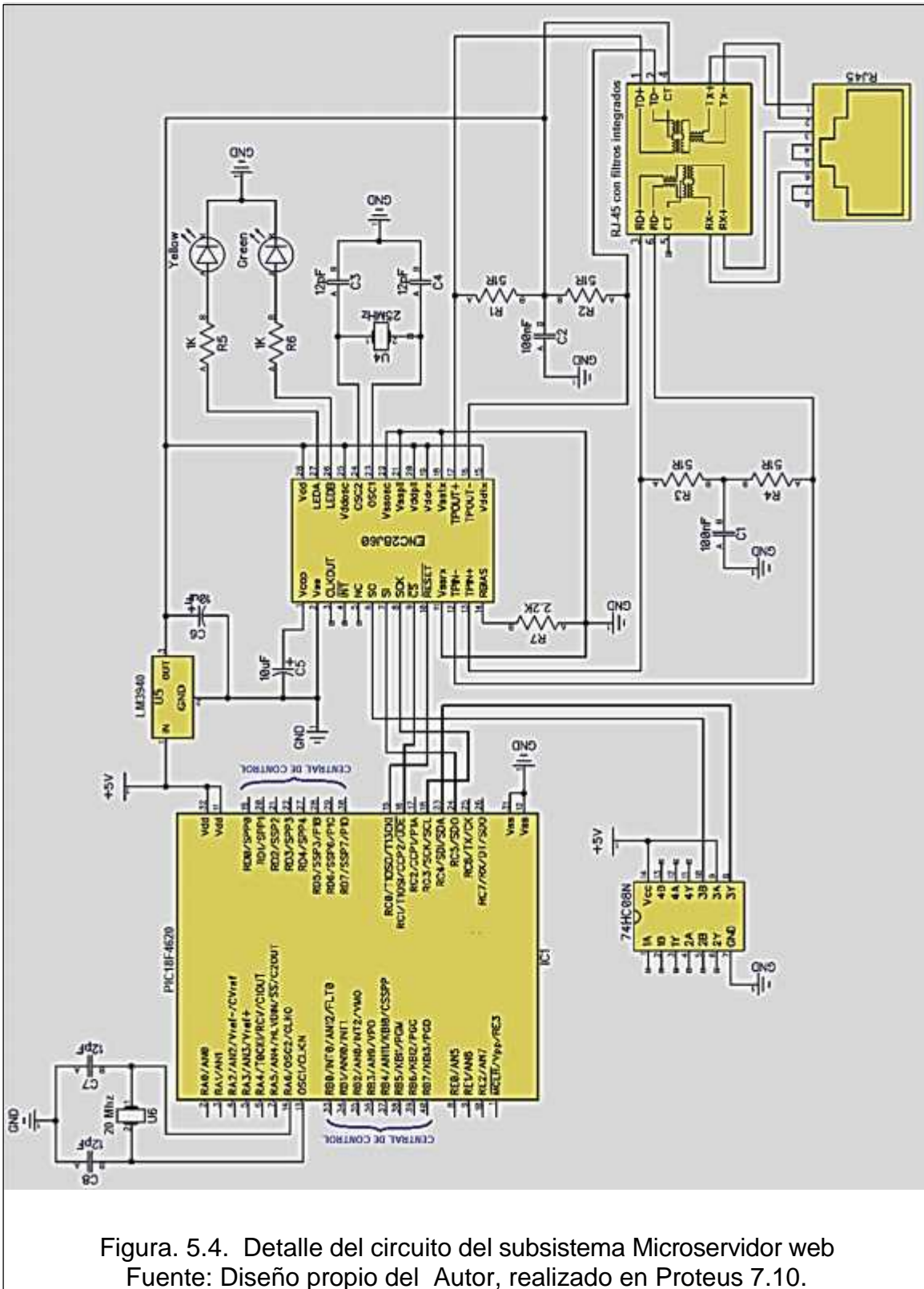
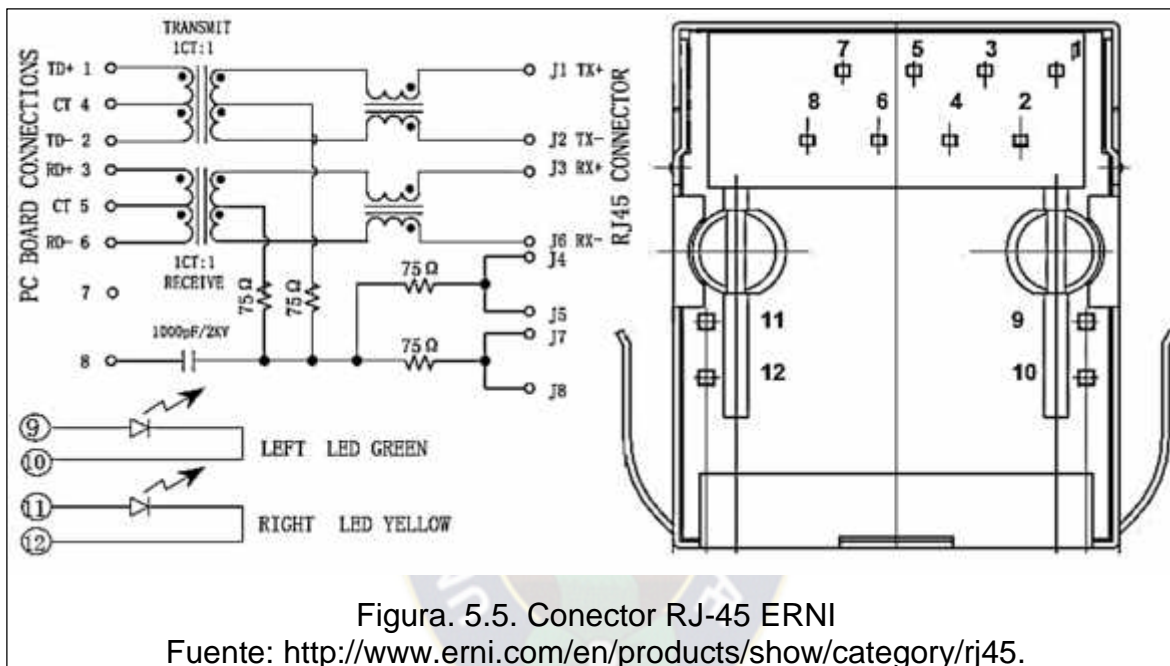


Figura. 5.4. Detalle del circuito del subsistema Microservidor web
Fuente: Diseño propio del Autor, realizado en Proteus 7.10.

5.3.3. INTERFAZ RJ-45 CON FILTROS

El conector RJ45 requiere de filtros magnéticos para discriminar armónicos de frecuencias indeseadas, ya sean estos integrados o sean añadidos al puerto de forma externa. En el presente sistema se usa el conector ERNI con filtros integrados el mismo que dispone de pines para su integración a la placa PCB y con leds indicadores de transmisión y recepción.



5.3.4. CONTROLADOR ENC28J60

El ENC28J60 requiere estar alimentado por una tensión de 3.3V con una corriente máxima de 180mA, es por ello que requiere de un regulador de voltaje LM3940.

El controlador ENC28J60 presenta 28 terminales, cada uno con una función específica. Para la comunicación SPI entre el microcontrolador y el controlador ENC28J60 se usaran cinco terminales, los cuales son los siguientes:

- RESET: Pin de reinicio del controlador con nivel bajo.
- SCK: Entrada de reloj.

- SI: Entrada de datos SPI del controlador.
- SO: Salida de datos SPI del controlador.
- CS: Selección del controlador con nivel bajo. En SPI es el selector del esclavo con quien se iniciara la comunicación, en este caso alerta al controlador que se iniciara la comunicación.

Para funcionar y poder conectarse a la red Ethernet, el ENC28J60 necesita de algunos componentes externos, tal como puede verse en la figura 5.6. El chip está diseñado para funcionar a 25MHz, el valor de C1 y C2 no está en la hoja de datos, pero 12pf parecen funcionar bien en la mayoría de las pruebas realizadas.

Los pines TPIN+ y TPIN- deben conectarse a un transformador 1:1 especial para redes 10BASE-T. Los pines TPOUT+ y TPOUT- necesitan de un transformador de pulso con relación 1:1 y punto medio. Este transformador debe ser capaz de proveer una aislación como mínimo de 2000V. Además, todos los pines mencionados necesitan de un resistor de 51 ohms con un 1% de tolerancia.

Es por los requerimientos mencionados anteriormente que el subsistema utilizara un conector RJ-45 ERNI que tenga integrada todos estos componentes para la conexión con la red Ethernet.

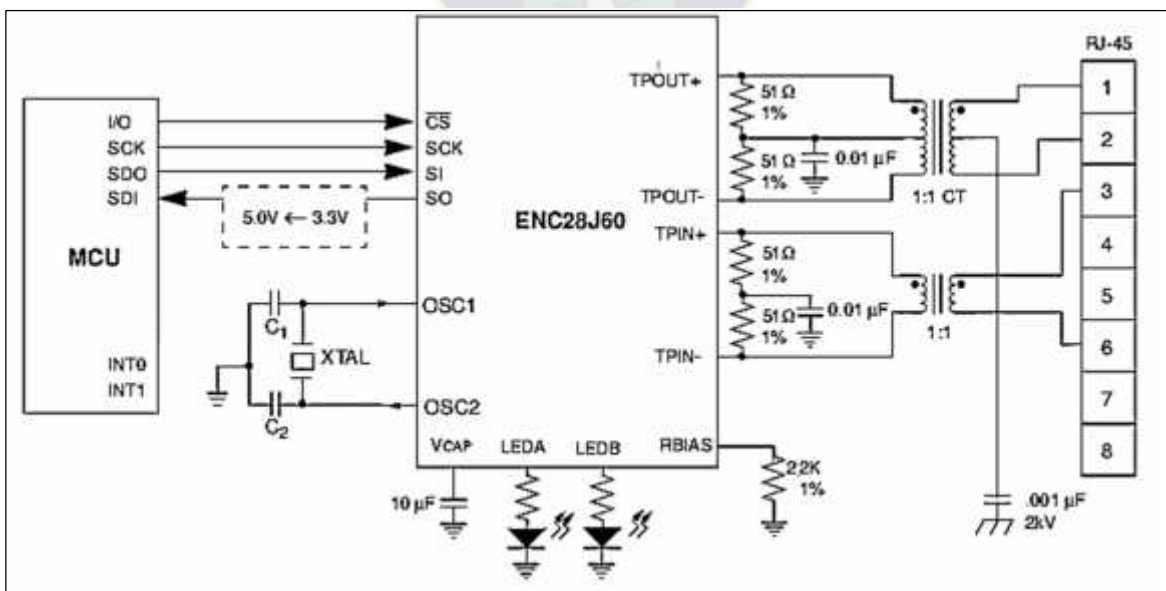


Figura. 5.6. Circuito del ENC28J60
Fuente: <https://www.alldatasheet.com/ENC28J60>.

La circuitería analógica interna del controlador de red también requiere que conectemos un resistor de 2.1 Kohms con un 1% de tolerancia entre el pin RBIAS y GND. El chip es capaz de operar también con una tensión de alimentación de solo 2.5V, en cuyo caso habrá que disponer un condensador de 10 uF entre el pin VCAP y GND.

5.3.5. ADAPTADOR DE TENSION

El ENC28J60 es un chip diseñado para operar a 3.3V, sin embargo puede ser integrado con facilidad en circuitos que se alimentan con 5V. Las tensiones de entrada al controlador de los pines RESET, SI, CS y SCK son tolerables a los 5V, la adaptación de voltaje se debe realizar en el pin SO cuyo voltaje es 3.3V.

Puede emplearse un 74HC08 o un 74ACT125, cuyos buffers de entrada adaptados para niveles TTL puede ser usado como adaptador. En el sistema se optó por la primera opción por el fácil acceso al componente.



5.3.6. EL PIC18F4620 EN EL MICROSERVIDOR

Este dispositivo es el corazón de este subsistema, puesto que el programa que tiene alojado permite que sistemas basados en microcontroladores realicen aplicaciones orientadas al uso de redes Ethernet. EL rol que cumple este dispositivo se puede resumir en los siguientes pasos:

- 1) Recibir las tramas provenientes del controlador ENC28J60 y en ellas las peticiones HTTP del cliente.

- 2) Procesar la información y solicita al subsistema Central de Control que informe el estado actual de la vivienda o que realice una acción en la misma.
- 3) Con la respuesta del Subsistema Central de Control se adecua la información para luego enviar la respuesta al requerimiento HTTP con la información solicitada por medio de una página web.
- 4) Las tramas se envían al controlador ENC29J60 para que las adecue y envíe a la red Ethernet.

La comunicación con el subsistema Central de Control se realiza a través de los puertos PORTD y PORTB para la adquisición y envío de datos respectivamente. El diagrama de flujo y las características del Firmware se describirán en el punto 5.8.

5.4. SUBSISTEMA CENTRAL DE CONTROL

5.4.1. DIAGRAMA DEL CIRCUITO

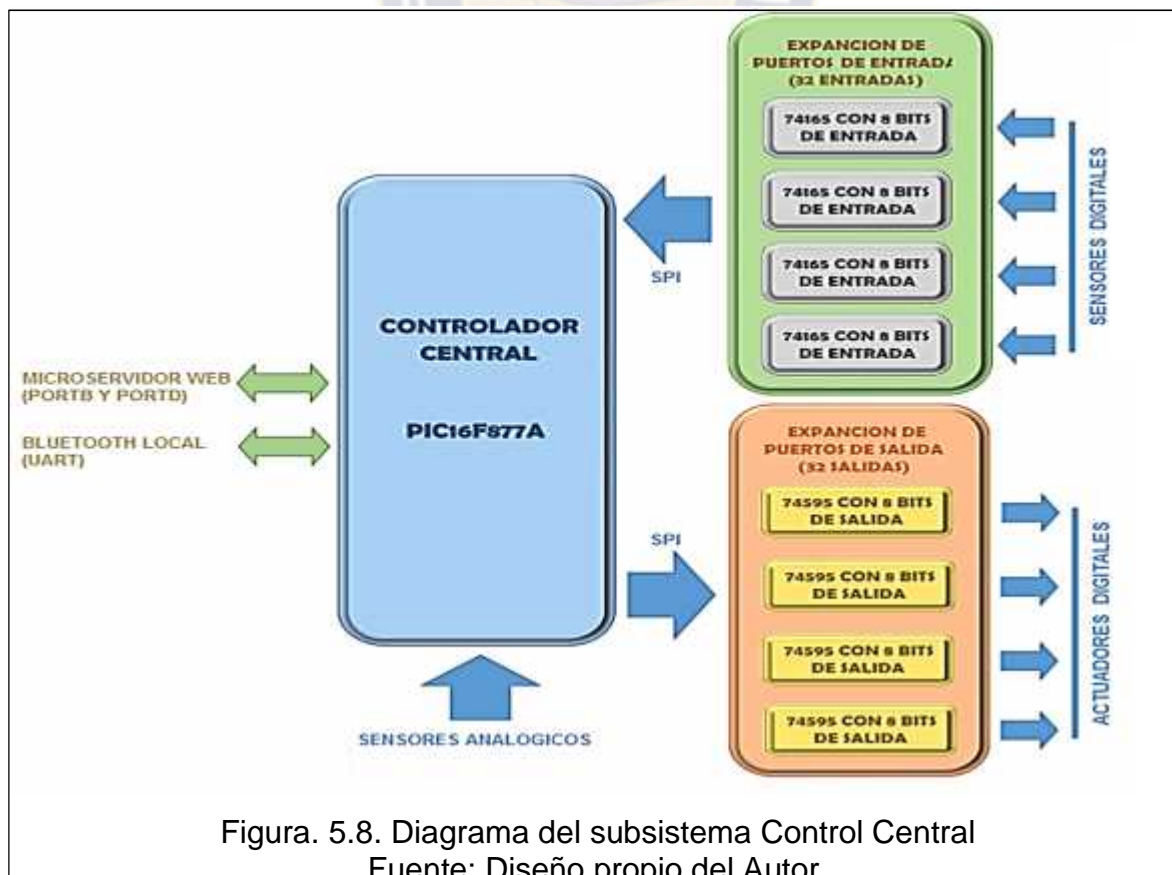


Figura. 5.8. Diagrama del subsistema Control Central
Fuente: Diseño propio del Autor.

5.4.2. CIRCUITO DEL CENTRAL DE CONTROL

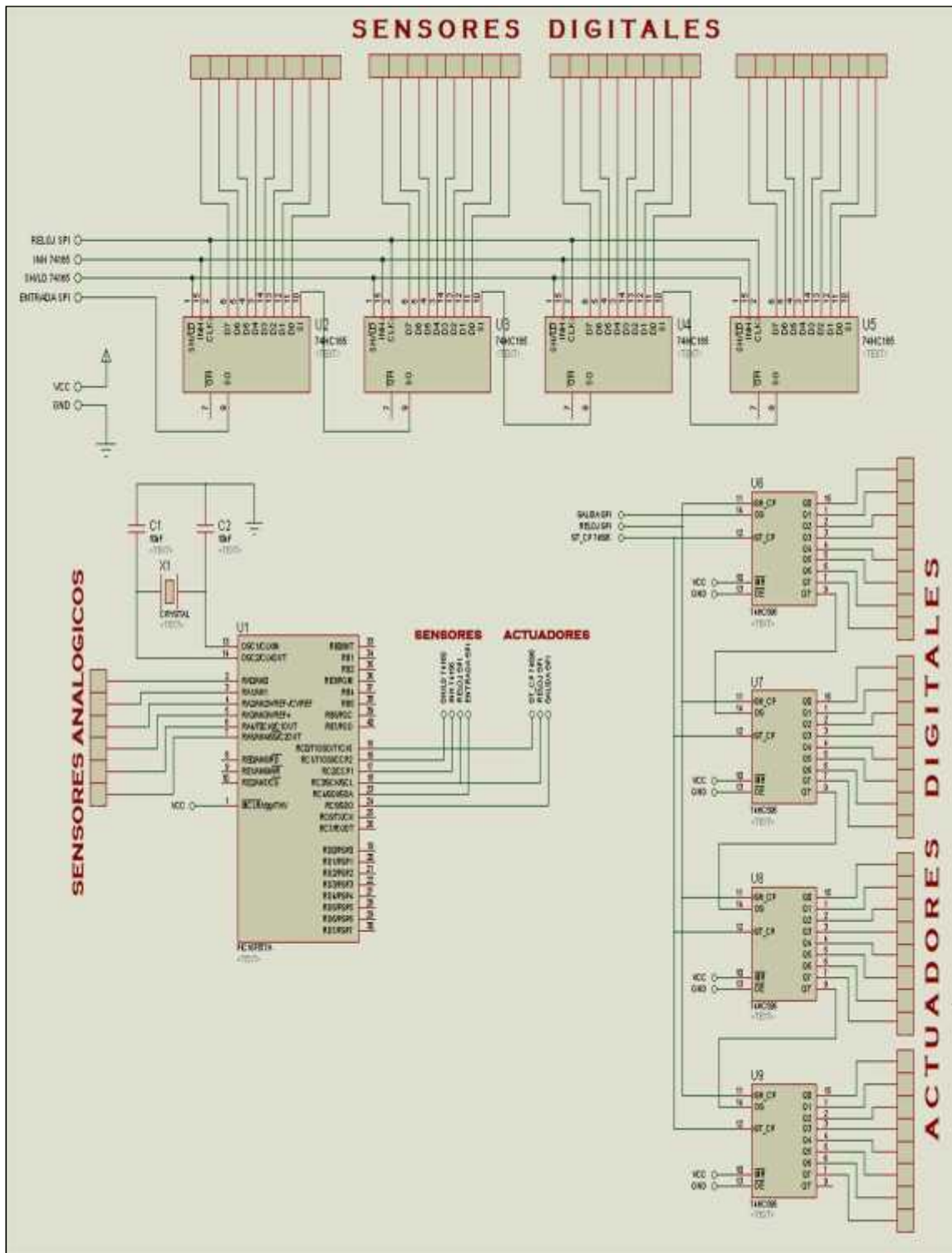


Figura. 5.9. Detalle del circuito del subsistema Control Central
Fuente: Diseño propio del Autor, realizado en Proteus 7.10.

5.4.3. EL PIC16F877A EN EL CENTRAL DE CONTROL

El microcontrolador PIC16F877A es el dispositivo central del sistema, cumpliendo tres funciones primordiales:

- Recolectar las órdenes de control proveniente de los subsistemas de comunicación Ethernet y Bluetooth que emplea el usuario como interfaz.
- Enviar datos provenientes de los sensores de la vivienda hacia los subsistemas de comunicación que el usuario empleará para el monitoreo.
- Realizar el control directo de los actuadores digitales
- Ejecutar acciones programadas de forma automática

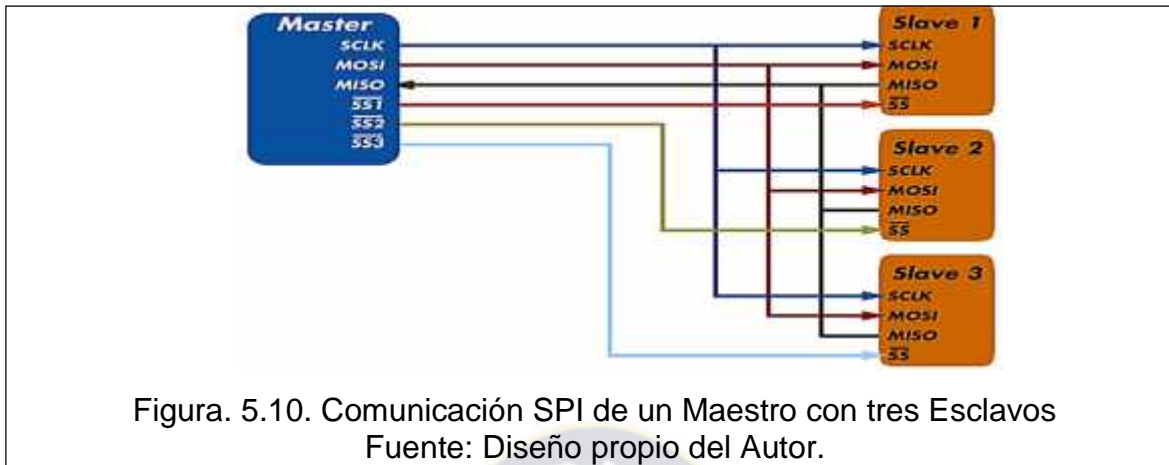
Se emplea de forma directa los puertos digitales PORTD y PORTB para la adquisición de datos provenientes de los sensores digitales, los cuales indicaran el estado actual de la vivienda.

La comunicación con el subsistema Bluetooth local se lo realiza empleando con el puerto UART, y para el subsistema Microservidor web se emplea los puertos B y D para la recepción y transmisión de información.

5.4.4. INTERFAZ SPI EN EL PIC16F877A

La interface periférica serial SPI es un estándar establecido por Motorola que utiliza un bus de 3 líneas, sobre el cual se transmiten paquetes de información de 8 bits. El microcontrolador 16F877A tiene integrada esta interfaz.

Este protocolo permite la comunicación serial síncrona entre dos o más dispositivos. Uno de ellos se denomina Maestro y los demás Esclavos. Los microcontroladores contienen ambas interfaces en el mismo chip, por lo tanto pueden funcionar en ambos modos.



SPI transfiere datos a alta velocidad, en la hoja de datos de la misma especifica un periodo de reloj de 20 nseg, trabajando de forma síncrona en modo full-dúplex. Requiere de tres líneas de comunicación llamadas MISO, MOSI y SCK además de habilitadores SS, cada una de estos de describe en la tabla 5.2.

Pin SPI	Pin en el PIC16F877A	Función
MOSI	SDO (Pin 24)	Master Output, Slave Input. Pin configurado como Salida para el Maestro y como Entrada para los Esclavos.
MISO	SDI (Pin 23)	Master Input, Slave Output. Pin configurado como Entrada para el Maestro y como Salida para los Esclavos.
SCK	SCK (Pin 18)	Serial Clock. Reloj serial controlado por el Maestro. Pin configurado como Salida para el Master y como Entrada para los Esclavos.
SS	Pin asignado por el Firmware	Slave Select. Para seleccionar el chip como Esclavo.

Tabla. 5.2. Descripción de los pines del puerto SPI del PIC16F877A
Fuente: <https://www.alldatasheet.com/PIC16F877A>.

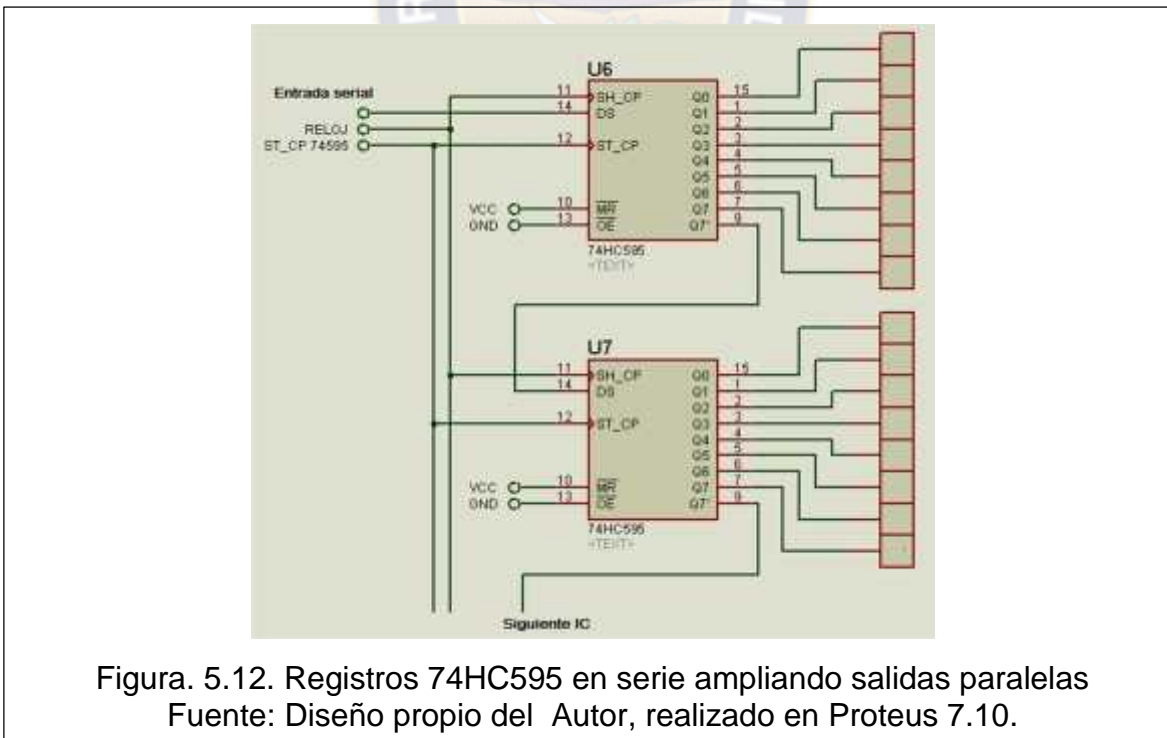
El subsistema de Control central utiliza esta interfaz para la expansión de puertos de salida y entrada del microcontrolador, el mismo cumple el rol de Maestro y los circuitos integrados 74595 y 74165 son dispositivos esclavos.

5.4.5. EXPANSIÓN DE PUERTOS DE SALIDA

Empleando registros de desplazamiento se puede ampliar las salidas digitales de un microcontrolador, para el control de múltiples salidas digitales.



El IC 74HC595 es un registro de desplazamiento de entrada serie y salida paralela. La ventaja de su implementación se da en la posibilidad de expandir las salidas digitales en cascada, y así agregar 8 puertos por cada integrado agregado.

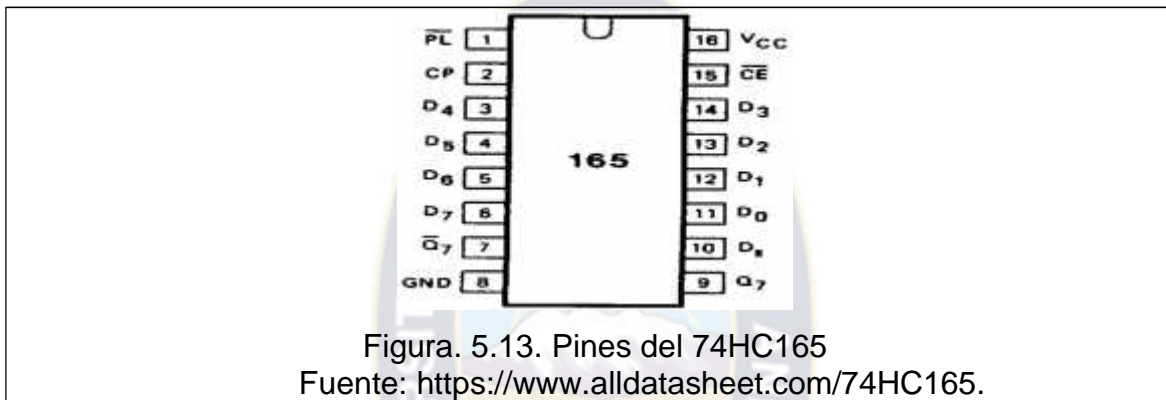


Los datos se reciben de forma serial al terminal DS acompañado de la señal de reloj en el pin SH_CP. Una vez guardados todos los datos en los registros, se

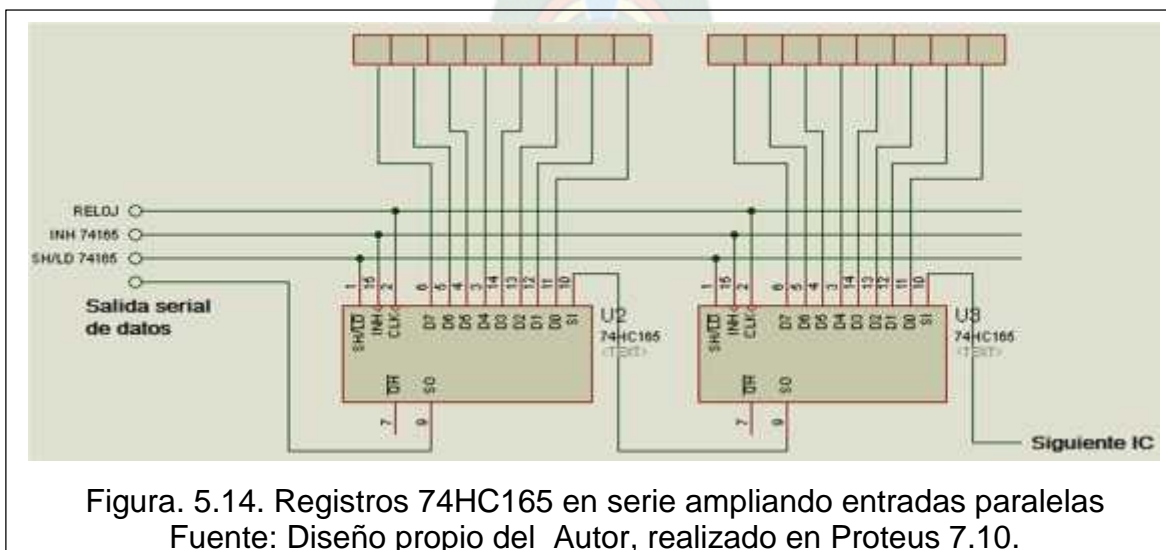
habilita la salida con el Latch ST_CP. Para agregar otro circuito integrado a la cascada se conecta la última salida Q7 con la entrada serial SI del nuevo chip.

5.4.6. EXPANSION DE PUERTOS DE ENTRADA

Para los puertos de entrada se emplea el integrado 74HC165, cuyo funcionamiento es básicamente similar al sistema de expansión de salidas digitales, con la diferencia que los registros de desplazamiento ahora son de entrada paralela y salida serial.



Este dispositivo tiene dos modos de trabajo, la carga paralela de datos y el desplazamiento serial de los mismos, para la selección del modo se emplea el pin SH/LD. Una vez cargado los datos, se procede a desplazar los mismos y enviarlos a la salida serial SO. Para la adición de más integrados a la cascada se conecta la entrada serial de la misma con la salida serial del nuevo chip.



5.5. SUBSISTEMA BLUETOOTH LOCAL

El usuario remoto tiene la posibilidad de acceder al control y monitoreo de la vivienda a través de Internet empleando la red del proveedor de servicios ISP, siendo necesario para la información realizar ese recorrido. Sin embargo, para el usuario local se considera innecesario que realice las acciones empleando la red remota.

El usuario local puede acceder al sistema de forma directa empleando la interfaz inalámbrica Bluetooth, evitando el mal uso de recursos y la demora de acción sobre la vivienda, siendo esa la función que cumplirá este subsistema.

5.5.1. DIAGRAMA DEL CIRCUITO

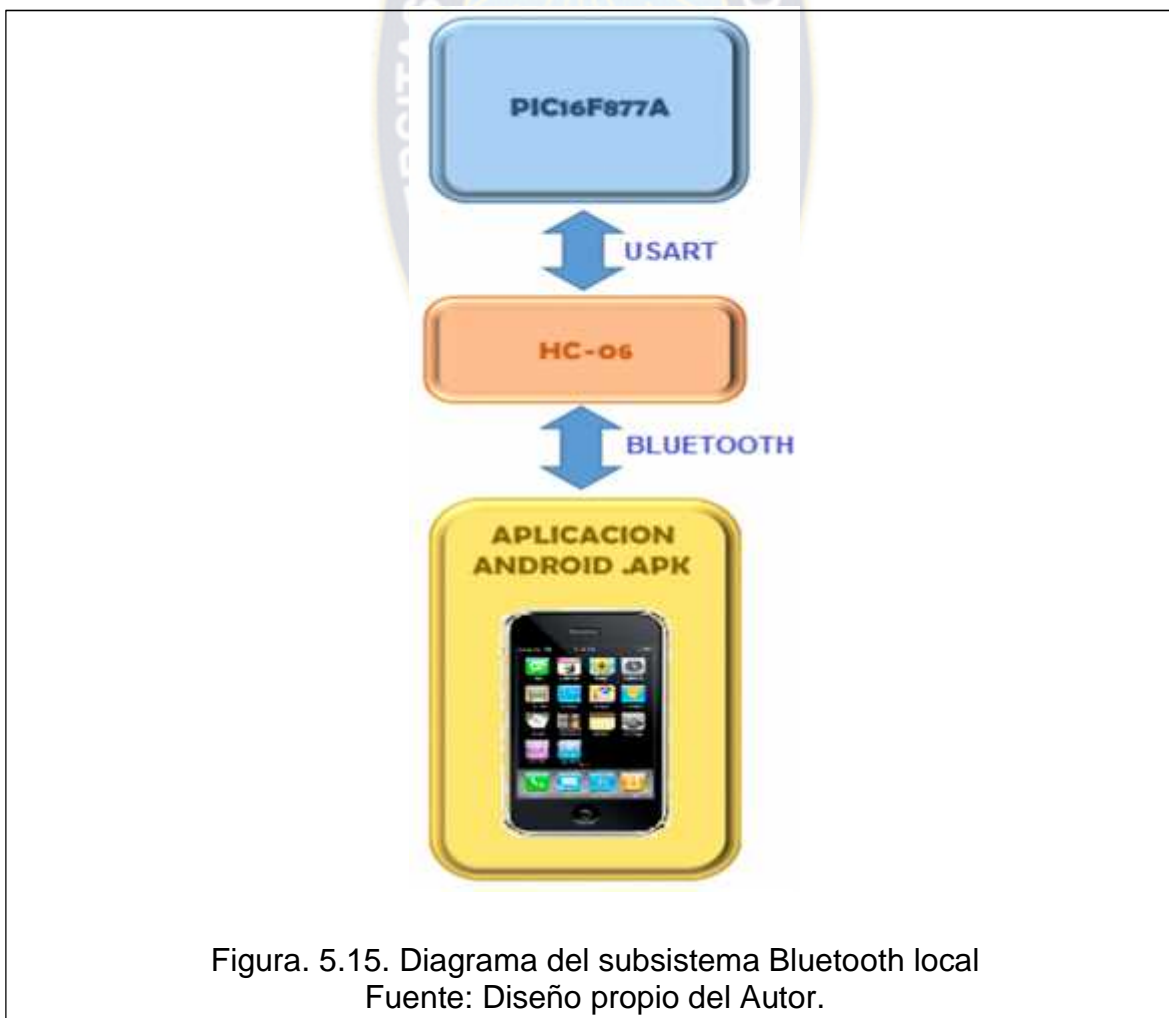


Figura. 5.15. Diagrama del subsistema Bluetooth local
Fuente: Diseño propio del Autor.

5.5.2. CIRCUITO DEL BLUETOOTH LOCAL

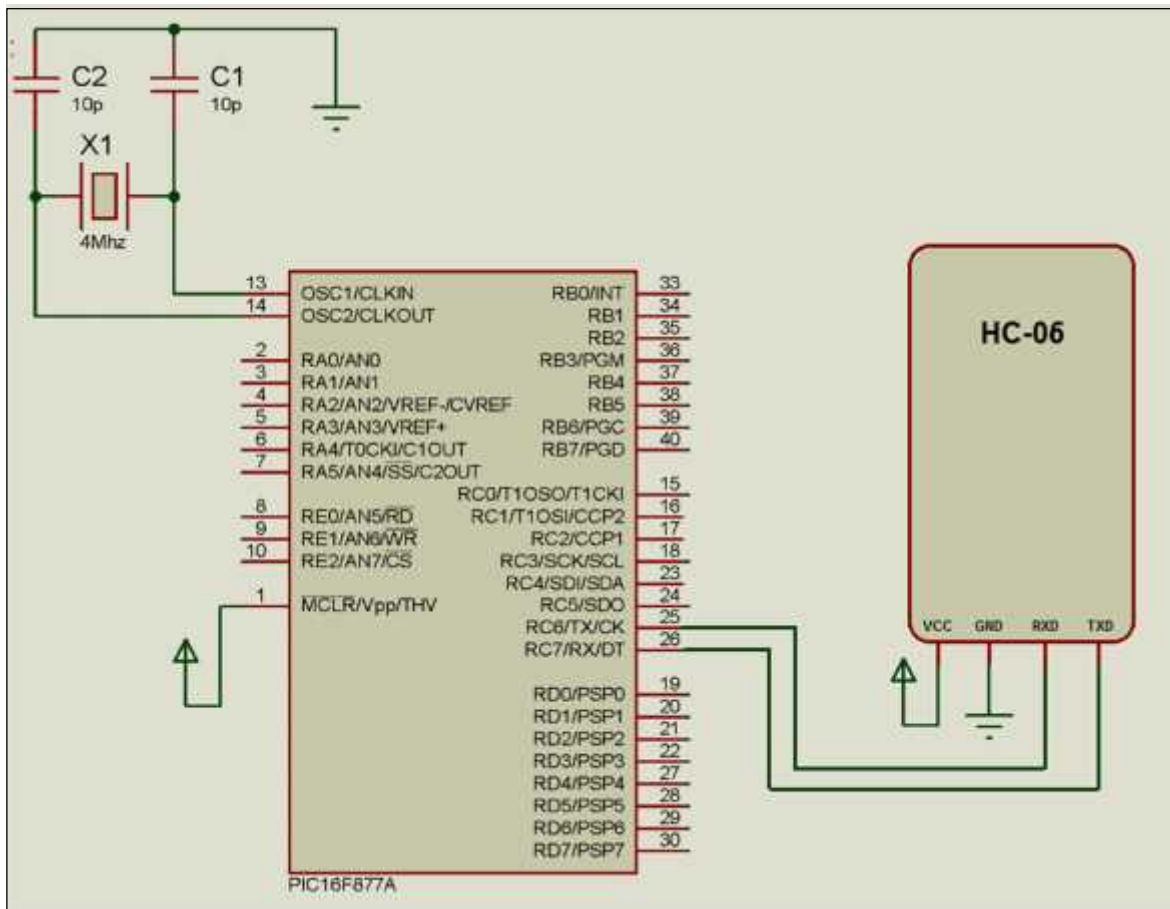


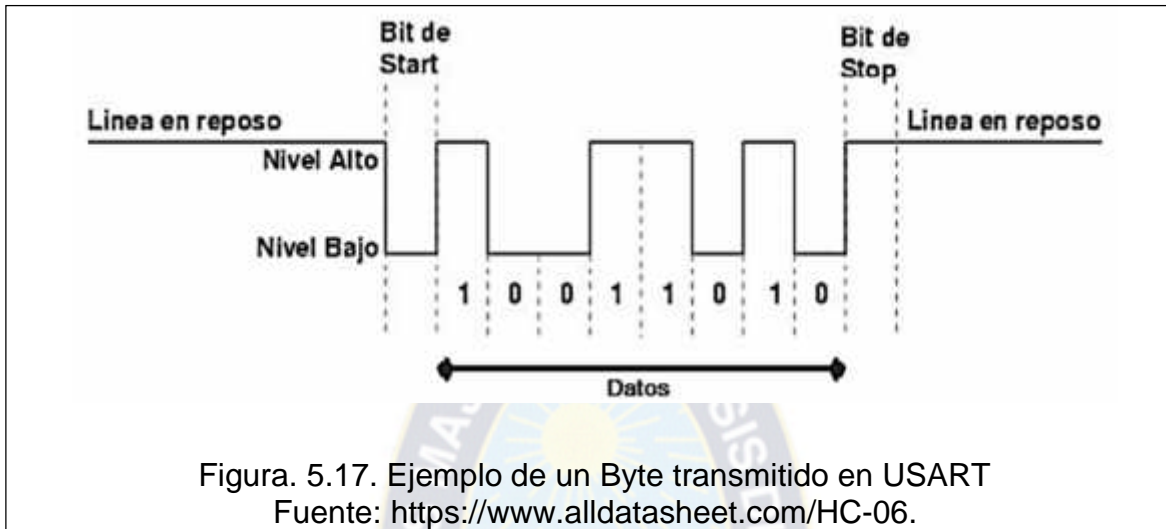
Figura. 5.16. Detalle del circuito del subsistema Control Central
Fuente: Diseño propio del Autor, realizado en Proteus 7.10.

5.5.3. COMUNICACIÓN SERIAL CON EL MODULO HC-06

El USART (Universal Synchronous Asynchronous Receiver Transmitter) es uno de los dos puertos series de los que dispone los PIC16F877A. Con la comunicación asíncrona se transmiten datos en serie y cada dispositivo debe sincronizarse independientemente pues no existe un reloj maestro.

Previamente se debe acordar que ambos dispositivos transmitirán datos a la misma velocidad (Baudios=bit/s) para transmitir. La velocidad es inversamente proporcional a la distancia de comunicación entre dispositivos. Un valor típico de

velocidad es el de 9600 Baud que permite distancias de hasta 15 metros. Los datos serie se encuentran encapsulados en tramas cuya forma se aprecia en la figura 5.17.



Primero se envía un bit de Start, a continuación los bits de datos (primero el bit de mayor peso) y finalmente los bits de Stop. El número de bits de datos varía entre 5 a 9 y los bits de STOP pueden ser 1, 1½ o 2. Estos parámetros pueden ser configurables entre los terminales de comunicación.

La comunicación entre el PIC16F877A con el módulo HC-06 se lo realiza por medio de la interfaz USART que disponen ambos dispositivos. Los parámetros de comunicación configuradas en el módulo son los siguientes:

Parámetro	Valor
Bits de parada	1
Bits de datos	8
Paridad	Ninguno

Tabla. 5.3. Parámetros de configuración del módulo HC-06
Fuente: Diseño propio del Autor.

Ambos dispositivos no requieren de una conversión a niveles de tensión manejados en el RS-232, pues la conexión es directa empleando niveles de voltaje TTL.

5.6. SENSORES Y ACTUADORES

El sistema realiza funciones de control y monitoreo por lo cual requiere de dispositivos actuadores y de censado digital y analógico. El requerimiento de los mismos se definió en el análisis previo empleando el Modelo de tres niveles de complejidad, especificándose cada uno en el primer nivel.

Los sensores y actuadores de cada servicio se identificaron de forma genérica, en base a ello se tiene la siguiente tabla 5.4, especificando de los dispositivos que forman parte del sistema definitivo.

Servicio	Sensores	Actuadores
Seguridad perimetral	Sensores magnéticos: <ul style="list-style-type: none"> • Modulo conmutador de efecto Hall 44e938. • Modulo sensor de barrera H21A1. • Modulo conmutador magnético Reed CD30. • Modulo conmutador Reed con salida digital regulable. Sensor de vibración: <ul style="list-style-type: none"> • Modulo sensor de vibración SW18010P. 	Bocina de alarma : <ul style="list-style-type: none"> • Módulo Buzzer pasivo. • Módulo Buzzer activo. Luces de alerta: <ul style="list-style-type: none"> • Modulo Bi-Color LED • Modulo LED de 7 Colores.
Seguridad de intrusión	Sensor volumétrico: <ul style="list-style-type: none"> ➤ Detector de movimiento pasivo PIR 	Bocina de alarma : <ul style="list-style-type: none"> • Módulo Buzzer pasivo. • Módulo Buzzer activo.

	<p>Sensor acústico:</p> <ul style="list-style-type: none"> ➤ Módulo micrófono Electret con sensibilidad regulable. 	<p>Luces de alerta:</p> <ul style="list-style-type: none"> • Modulo Bi-Color LED • Módulo LEDGlow con alto brillo de 7 Colores.
Seguridad técnica	<p>Sensor de llama:</p> <ul style="list-style-type: none"> ➤ Modulo sensor de Flama YG1006. 	<p>Riego contra incendios:</p> <ul style="list-style-type: none"> ➤ Módulo Rele de 5v 10A. <p>Bocina de alarma :</p> <ul style="list-style-type: none"> • Módulo Buzzer pasivo. • Módulo Buzzer activo. <p>Luces de alerta:</p> <ul style="list-style-type: none"> • Modulo Bi-Color LED • Modulo LED de 7 Colores.
Seguridad pasiva	<p>No requiere por ser activado por el usuario</p>	<p>Luces incandescentes internas y externas:</p> <ul style="list-style-type: none"> ➤ Módulo Rele de 5v 10A.
Accesibilidad remota y local	<p>Sensores de temperatura ambiente:</p> <ul style="list-style-type: none"> ➤ Modulo sensor de temperatura con termistor NTC ➤ Modulo sensor de temperatura NTC con sensibilidad regulable. ➤ Modulo sensor de temperatura LM35 <p>Todos los sensores de los demás servicios son</p>	<p>Luces incandescentes RGB:</p> <ul style="list-style-type: none"> • Modulo LED RGB superficial. • Modulo LED RGB de alto brillo <p>Todos los actuadores de los demás servicios son controlados por las interfaces remota y local</p>

	visualizados por las interfaces remota y local	
Ahorro energético	No requiere por ser activado por el usuario	<p>Interruptor de tomas de corriente:</p> <ul style="list-style-type: none"> ➤ Módulo Rele de 5v 10A. <p>Luces incandescentes internas y externas:</p> <ul style="list-style-type: none"> ➤ Módulo pack de 4 Reles de 5v 10A.
Portero automático	Activado por el usuario	<p>Motor de apertura:</p> <ul style="list-style-type: none"> ➤ Módulo Rele de 5v 10A.

Tabla. 5.4. Sensores y Actuadores utilizados en el sistema
Fuente: Diseño propio del Autor.

5.6.1. SENSORES

Los sensores utilizados en este sistema se implementaron en módulos PCB de dimensiones pequeñas y acondicionados previamente para facilitar el uso e instalación. La descripción de cada uno de los sensores y los detalles técnicos de los módulos se describen a continuación.

➤ **Módulo conmutador de efecto Hall 44e938.**

Este módulo es utilizado para detectar la presencia de un campo magnético. El 44e938 es un interruptor de efecto Hall de alta sensibilidad ideal para la detección de un imán de barra. Cada dispositivo incluye un regulador de voltaje que permite el funcionamiento con tensiones de alimentación de hasta 24V.

- Pines del módulo: GND, VCC, Señal de salida.
- Alimentación: 3.5 a 24 V.
- Corriente: 3 mA en pasivo, 8 mA con conmutador activado.
- Temperatura: 150° máximo.

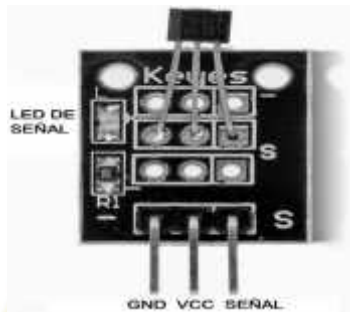


Figura. 5.18. Módulo conmutador de efecto Hall 44e938
Fuente: <https://tkkrlab.nl/sensors>.

➤ **Módulo sensor de barrera H21A1**

El sensor H21A1 tiene integrado un diodo emisor infrarrojo y un fototransistor, separados a 3mm una con la otra. Cuando un objeto rompe la barrera el fototransistor entra en corte, el mismo con una polarización adecuada envía una señal de nivel alto en el emisor.

- Pines del módulo: Señal de salida, VCC, GND.
- Alimentación: 5 V.
- Corriente: 30 mA máximo.
- Temperatura: -55 a 100 °C.

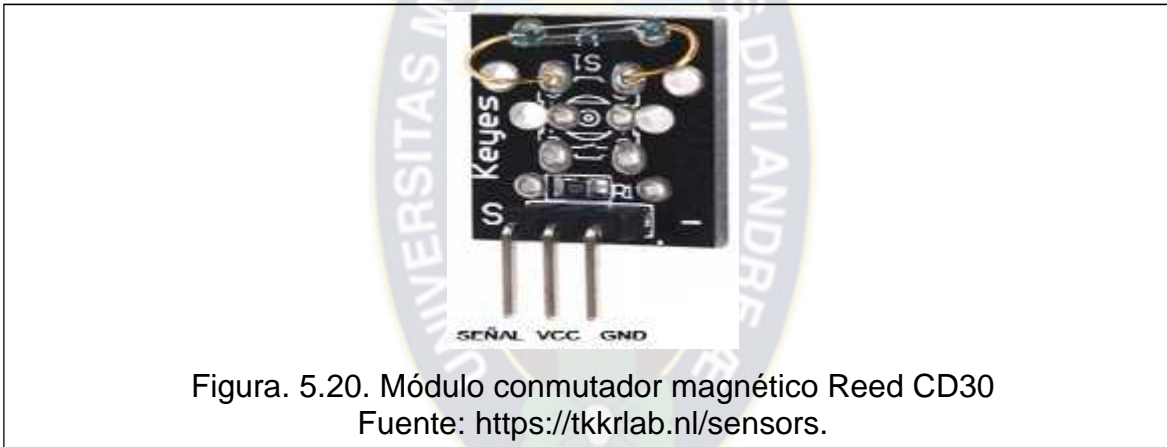


Figura. 5.19. Módulo sensor de barrera H21A1
Fuente: <https://tkkrlab.nl/sensors>.

➤ **Módulo conmutador magnético Reed CD30**

El conmutador Reed CD30 consiste en un par de contactos ferrosos encerrados al vacío dentro un tubo de vidrio. Cada contacto está sellado en los extremos opuestos del tubo de vidrio. El tubo tiene 10 mm de largo y 3 mm de diámetro. Cuando los contactos están normalmente abiertos se cierran en la presencia de un campo magnético.

- Pines del módulo: Señal de salida, VCC, GND.
- Alimentación: 5 V.
- Corriente: 1 mA máximo.
- Temperatura: -20 a 125 °C.



➤ **Módulo conmutador Reed con salida digital regulable**

El modulo presenta el mismo sensor magnético CD30, pero adicionalmente lleva un circuito que permite la regulación de la sensibilidad de detección. Basado en un arreglo de resistencias, un comparador de bajo voltaje LM393 y un mini potenciómetro para su regulación, permitiendo medir la intensidad de campo magnético y niveles alto y bajo de detección en su salida digital.

- Pines del módulo: Señal analógica, GND, VCC, Señal digital.
- Alimentación: 5 V.
- Corriente: 50 mA máximo.
- Temperatura: -0 a 70 °C.

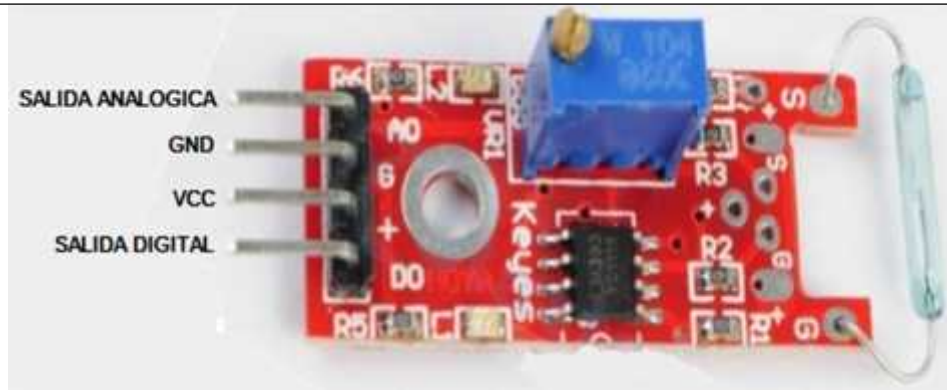


Figura. 5.21. Módulo conmutador Reed con salida digital regulable
Fuente: <https://tkkrlab.nl/sensors>.

➤ **Modulo sensor de vibración SW18010P**

El sensor SW18010P es un conmutador cuyos contactos se cierran al detectar de vibraciones en una superficie normalmente estática.

- Pines del módulo: Señal de salida, VCC, GND.
- Alimentación: 3.5 a 12 V.
- Corriente: 50 mA máximo.
- Temperatura: -25 a 85 °C.



Figura. 5.22. Módulo sensor de vibración SW18010P
Fuente: <https://tkkrlab.nl/sensors>.

➤ **Detector de movimiento pasivo PIR**

Un sensor PIR permite la medición de cambios en los niveles de radiación infrarroja emitida por los objetos a su alrededor. Los cambios en la radiación producen cambios de voltaje amplificados por el modulo, empleándose también filtros Frenel que enfocan las señales infrarrojas sobre el elemento sensor.

- Pines del módulo: Señal de salida, VCC, GND.
- Alimentación: 3 a 6 V.
- Corriente: 25 mA máximo.
- Rango de medición: 6 mts.



➤ **Módulo micrófono Electret con sensibilidad regulable**

El micrófono a condensador Electret ECM presenta una salida de alta impedancia y un amplificador integral FET. Envía a su salida de 1 a 2 mV con sonidos aproximadamente a un metro de distancia de la cápsula del micrófono. El modulo emplea un arreglo de resistencias, un comparador de bajo voltaje LM393 y un mini potenciómetro para la que la salida digital envíe un nivel alto cuando se exceda el valor asignado.

- Pines del módulo: Señal analógica, GND, VCC, Señal digital.
- Alimentación: 5 V.
- Impedancia: 2.2 Kohm máximo.

- Corriente: 50 mA máximo.
- Sensibilidad: -70 a 2dB.
- Temperatura: -0 a 70 °C.



➤ Módulo sensor de Flama YG1006

El modulo emplea el sensor YG1006 es empleado para detectar fuentes de fuego, es un fototransistor infrarrojo preparado para responder a radiaciones infrarrojas con longitudes de onda comprendidas entre 760 y 1100nm, la emisión infrarroja de una llama está dentro de este rango de longitudes de onda, por lo que este sensor es apropiado para detectarlas. Al igual que los anteriores módulos con sensibilidad regulada emplea un mini potenciómetro y un comparador de bajo voltaje LM393.

- Pines del módulo: Señal analógica, GND, VCC, Señal digital.
- Alimentación: 5 V.
- Corriente: 50 mA máximo.
- Temperatura: -25 a 85 °C.



➤ **Módulo sensor de temperatura con termistor NTC**

El sensor empleado en este módulo es un termistor NTC pues tiene un coeficiente de temperatura negativo.

- Pines del módulo: Señal de salida, VCC, GND.
- Alimentación: 5 V.
- Rango de medición: -40 a 125 °C.
- Rango resistivo: 3.3 ohms a 2.8 kohms

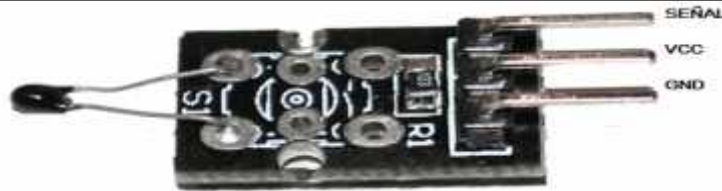


Figura. 5.26. Módulo sensor de temperatura con termistor NTC
Fuente: <https://tkkrlab.nl/sensors>.

➤ **Módulo sensor de temperatura NTC con salida digital regulable.**

Presenta una funcionalidad similar al Módulo sensor de temperatura con termistor NTC, mejorándose el mismo al ampliar una salida digital adicional, cuyo nivel será alto cuando exceda el valor de referencia asignado por el potenciómetro.

- Pines del módulo: Señal analógica, GND, VCC, Señal digital.
- Alimentación: 5 V.
- Rango de medición: -40 a 125 °C.
- Rango resistivo: 3.3 ohms a 2.8 kohms

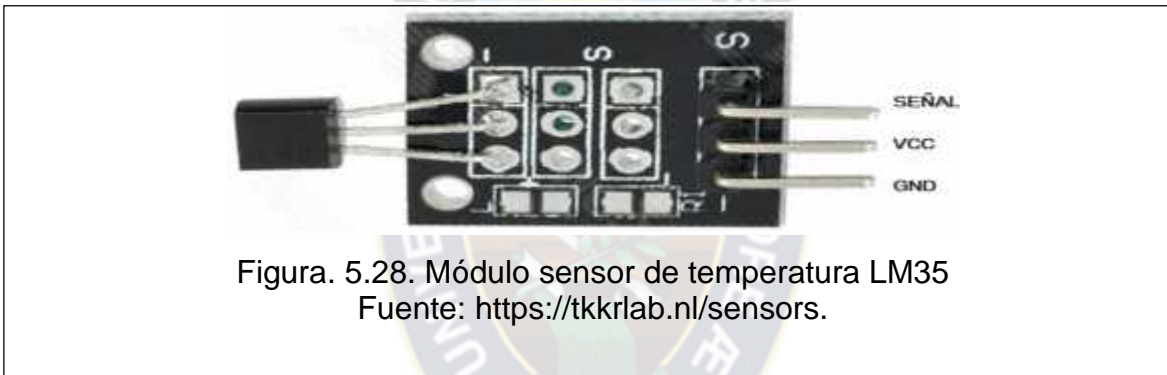


Figura. 5.27. Módulo sensor de temperatura NTC con salida digital regulable
Fuente: <https://tkkrlab.nl/sensors>.

➤ **Módulo sensor de temperatura LM35**

Este sensor LM35 es comúnmente empleado en la medición de temperatura ambiente, al no requerir de circuitería externa hace que este dispositivo sea instalado fácilmente en cualquier circuito de control. La salida es lineal y cada grado centígrado equivale a 10mV.

- Pines del módulo: Señal de salida, VCC, GND.
- Alimentación: 5 V.
- Rango de medición: -55 a 150 °C.
- Precisión: 1°C por cada 10 mV.
- Corriente: 60uA.



5.6.2. ACTUADORES

Los actuadores utilizados en este sistema, al igual que los sensores, se implementaron en módulos PCB de dimensiones pequeñas y acondicionados previamente para facilitar el uso e instalación.

La descripción de cada uno de los actuadores y los detalles técnicos de los módulos se describen a continuación.

➤ **Módulo Buzzer pasivo**

Los buzzers de tipo pasivo tienen la característica de emitir sonidos con distintos tipos de tono, en base a la ciclo de trabajo si se emplea señales PWM, así como también el nivel de potencia del sonido dependerá de la amplitud de la señal.

- Pines del módulo: Señal de entrada, VCC, GND.
- Alimentación: 1.5V-15V
- Señal de audio: 1.5 a 2.5 KHz



Figura. 5.29. Módulo Buzzer pasivo. Fuente: <https://tkkrlab.nl/sensors>.

➤ **Módulo Buzzer activo**

Los buzzers de tipo activo solo requieren de alimentación en tensión para que emitan un sonido único el cual no puede modificarse como los buzzers de tipo pasivo. En el caso del módulo, existirá un pin de señal de entrada que con un nivel alto se activara el Buzzer.

- Pines del módulo: Señal de entrada, VCC, GND.
- Alimentación: 1.5V-15V
- Señal de audio: 1.8 KHz



Figura. 5.30. Módulo Buzzer activo. Fuente: <https://tkkrlab.nl/sensors>.

➤ **Módulo Bi-Color LED**

La representación de dos colores y la combinación de ambos en un solo led es posible con un Bi-color led. La emisión de los colores rojo y verde depende de la tensión provista a uno de los terminales de activación. Para la combinación de ambos colores es necesario el empleo de señales PWM.

- Pines del módulo: GND, Rojo, Verde.
- Alimentación: 5V.
- Corriente: 30 mA máximo.
- Temperatura: -55 a 100 °C.

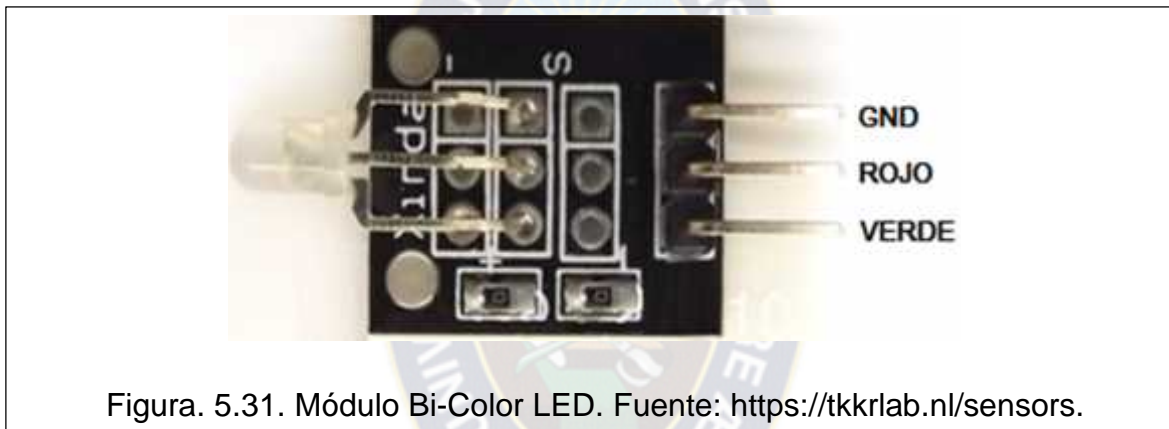


Figura. 5.31. Módulo Bi-Color LED. Fuente: <https://tkkrlab.nl/sensors>.

➤ **Módulo LED Glow con alto brillo de 7 Colores**

El módulo led de 7 colores emplea un solo terminal para el envío de pulsos de señal para el cambio de color. Por cada nivel alto y bajo el LED cambia de color automáticamente.

- Pines del módulo: Señal de entrada, VCC, GND.
- Alimentación: 3 a 4.5V.
- Corriente: 30 mA máximo.
- Temperatura: -55 a 100 °C.

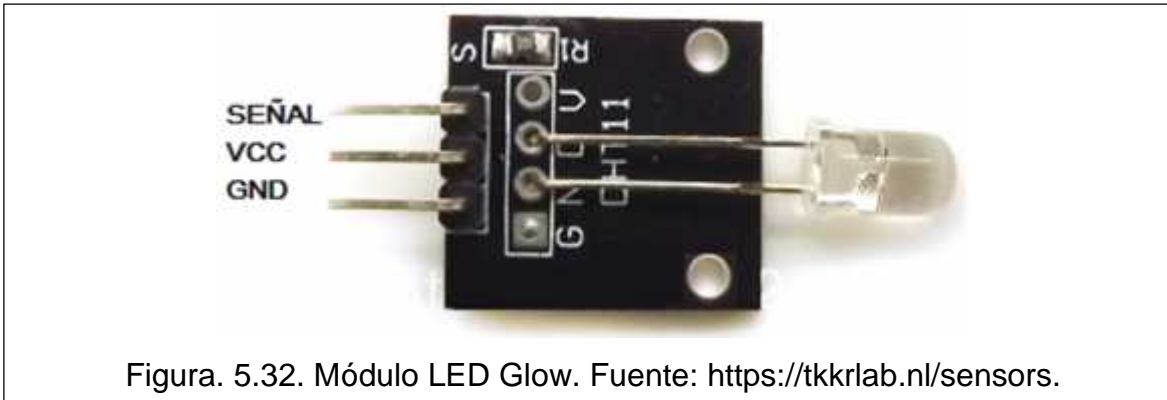


Figura. 5.32. Módulo LED Glow. Fuente: <https://tkkrlab.nl/sensors>.

➤ **Módulo Relé de 5v 10A**

Este módulo acondiciona los terminales del relé en 3 pines de operación con un terminal de activación mediante una señal digital de nivel alto. A la salida del módulo se dispone de tres terminales, un común y dos salidas NC y NO.

- Pines del módulo: Señal de entrada, VCC, GND.
- Alimentación: 5 V.
- Corriente: 10 A en la salida.
- Temperatura: -40 a 85 °C.



Figura. 5.33. Módulo Relé. Fuente: <https://tkkrlab.nl/sensors>.

➤ **Módulo pack de 4 Relés de 5v 10A.**

Al igual que el modulo anterior, se acondiciona los terminales de cuatro relés con una alimentación común y un terminal de activación para cada relé. A la salida del módulo se dispone de múltiples salidas NC y NO.

- Pines del módulo: VCC, Señal de entrada 1, Señal de entrada 2, Señal de entrada 3, Señal de entrada 4, GND.
- Alimentación: 5 V.
- Corriente: 10 A en la salida.
- Temperatura: -40 a 85 °C.

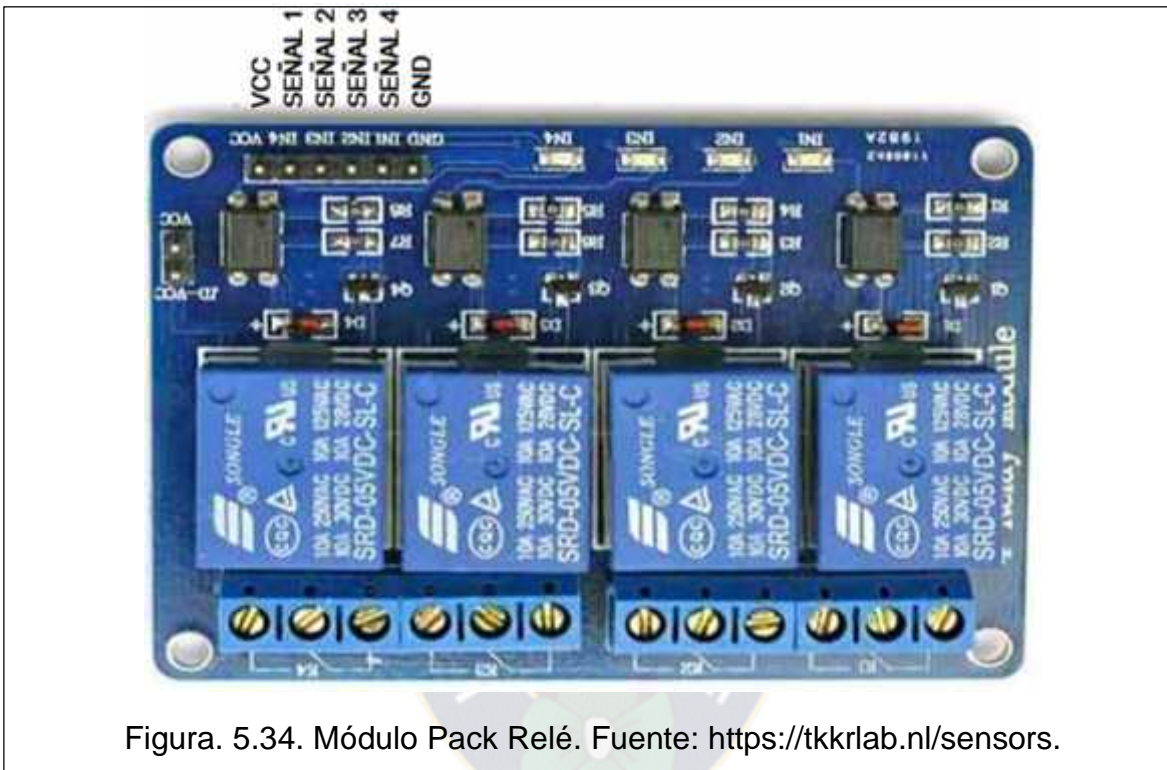


Figura. 5.34. Módulo Pack Relé. Fuente: <https://tkkrlab.nl/sensors>.

➤ **Modulo LED RGB superficial.**

El diodo LED RGB de superficie puede tener la capacidad de cambiar el color que emite cuando se aplican niveles altos en cualquiera de sus tres terminales. Al emplear señales PWM se puede combinar tonos de color específicos variando los ciclos de trabajo de cada color.

- Pines del módulo: GND, Señal rojo, señal verde, señal azul.
- Alimentación: 5 V.
- Corriente: 10 mA máximo.
- Temperatura: -45 a 80 °C.

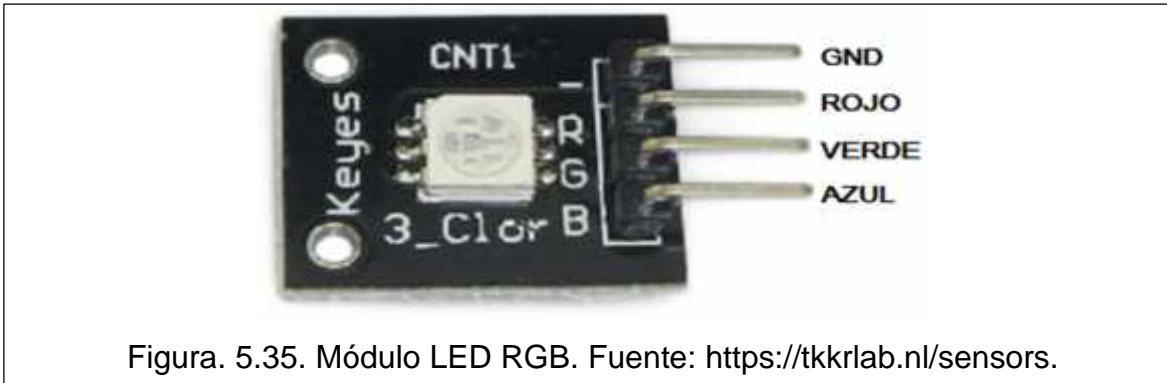


Figura. 5.35. Módulo LED RGB. Fuente: <https://tkkrlab.nl/sensors>.

➤ **Módulo LED RGB de alto brillo**

Al igual que el modulo anterior, este diodo LED RGB puede cambiar el color que emite cuando se aplica niveles altos en cualquiera de sus tres terminales. Con su led de alto brillo, se puede emplear señales PWM para combinar tonos de color.

- Pines del módulo: GND, Señal rojo, señal verde, señal azul.
- Alimentación: 5 V.
- Corriente: 10 mA máximo.
- Temperatura: -45 a 100 °C.



Figura. 5.36. Módulo LED RGB de alto brillo. Fuente: <https://tkkrlab.nl/sensors>.

5.7. REGISTROS DEL SISTEMA

El sistema domótico emplea un conjunto de registros internos que se actualizan cuando el usuario realiza acciones de control, como también cuando el sistema detecta cambios de estado en los sensores.

Cada cambio en los sensores y actuadores se los guarda en registros de un Byte, dentro de un vector llamado "Registros" que contendrá todos los registros para un manejo rápido de los mismos en el momento de actualizar sus estados. En la figura 5.37 se describe la composición de este vector.

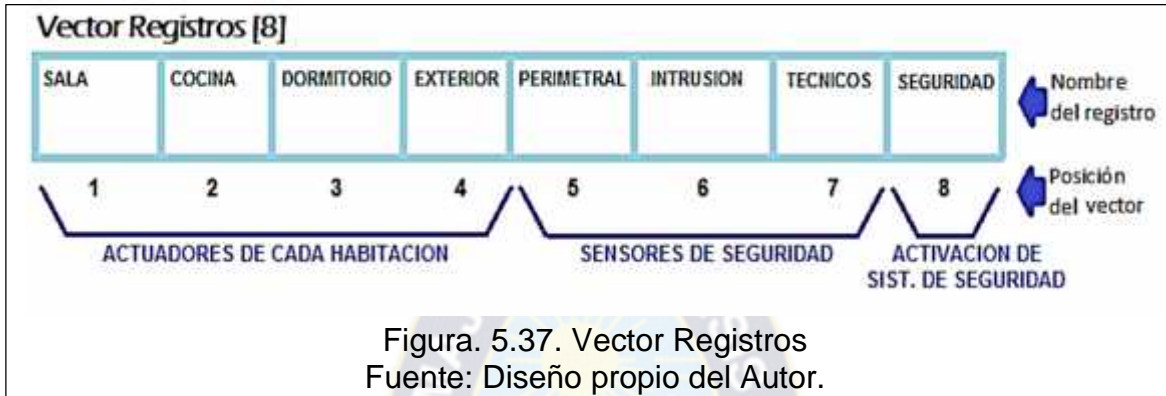


Figura. 5.37. Vector Registros
Fuente: Diseño propio del Autor.

Cada subsistema tendrá este conjunto de registros internamente. El cambio de estos registros tendrá como origen a dos actores: El usuario que realiza acciones de control (desde el subsistema Microservidor Web o Bluetooth local) y los sensores para el monitoreo (subsistema Control Central).

Una vez realizado el cambio del registro en uno de los subsistemas, el mismo debe compartir la información con el resto enviando un mensaje de un Byte con toda la información del cambio para que actualicen sus registros. La composición de dicho mensaje es la siguiente.



Figura. 5.38. Estructura del mensaje de un Byte
Fuente: Diseño propio del Autor.

El software del sistema, al igual que el hardware, tiene la característica de ser escalable, pues dependiendo de las dimensiones de la vivienda se puede incrementar los registros con más habitaciones y servicios que solicita el usuario.

5.8. PROGRAMA DEL SUBSISTEMA MICROSERVIDOR WEB

5.8.1. METODO GET EMPLEADO EN EL SUBSISTEMA

La interfaz del usuario remoto emplea páginas web con enlaces de hipervínculo, los cuales envían una petición mediante el método GET al servidor.

El Microservidor web implementado en el sistema recibirá dicha petición y la analizará para realizar la acción en la vivienda. Cada instrucción enviada empleando el método GET modificará los registros internos de todos los subsistemas, teniendo como origen la dirección URL con la solicitud que realizó el usuario. En la figura 5.39 se observa la estructura de cada mensaje enviado al Microservidor web.



5.8.2. FIRMWARE DEL PIC18F4620

El Firmware que se emplea en el microcontrolador PIC18F4620 del subsistema Microservidor Web es el que permite la conexión del sistema domótico con las redes Ethernet. El código fuente se lo realizó empleando el compilador MikroC, pues este compilador dispone de la librería Ethernet_SPI, la cual permite al microcontrolador emplear el puerto SPI para comunicación mediante el protocolo TCP.

El compilador MicroC emplea el lenguaje C, y la estructura del Firmware del Microservidor Web de forma general se pueden apreciar en las figuras 5.40 y 5.41 que se aprecia a continuación.

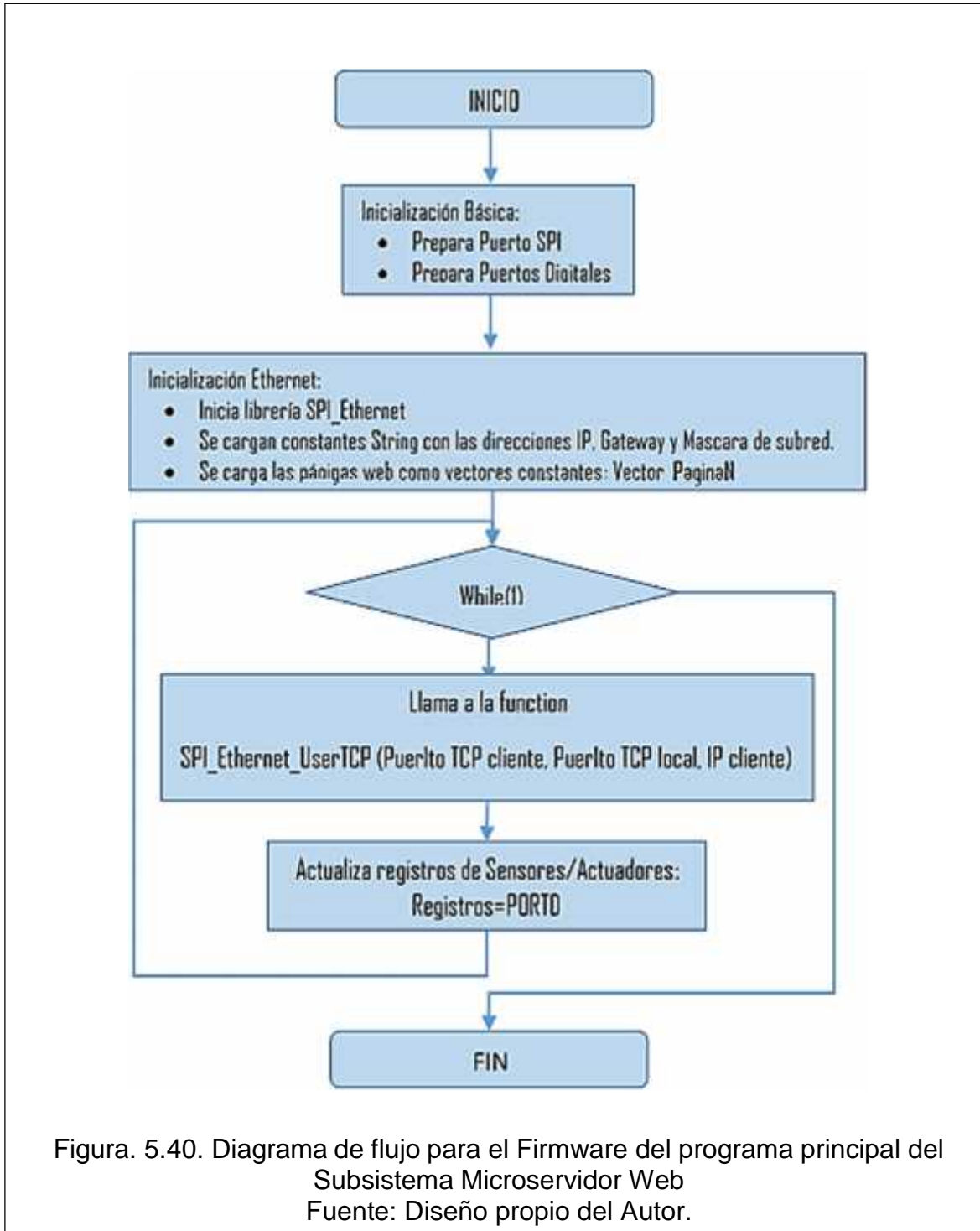
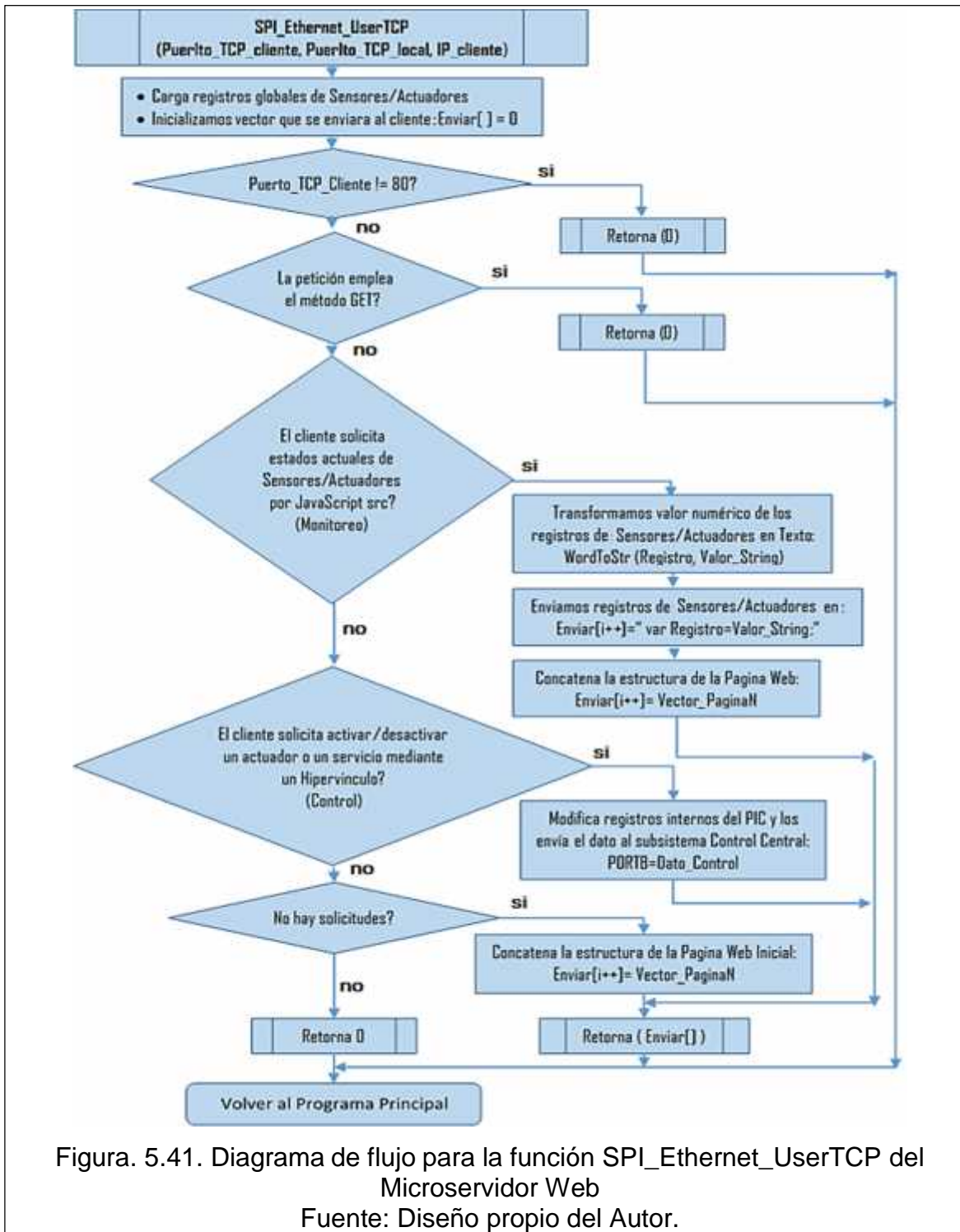


Figura. 5.40. Diagrama de flujo para el Firmware del programa principal del Subsistema Microservidor Web
Fuente: Diseño propio del Autor.



Las librerías de MikroC presentan funciones que facilitan procesos que presentan un grado de complejidad considerable. Las funciones más relevantes empleadas en el Firmware son las siguientes:

- **SPI_Ethernet_doPacket:** Esta rutina procesa el paquete recibido en caso de existir. Los paquetes se procesan bajo los siguientes pasos:
 1. Peticiones ARP y ICMP se contestan de forma automática.
 2. Si existe una petición TCP se llama a la función SPI_Ethernet_userTCP.
 3. Si existe una petición UDP se llama a la función SPI_Etherne_UserUDP.

- **SPI_Ethernet_UserTCP:** Esta función se llama internamente por la librería. La función debe devolver la longitud en bytes con la respuesta TCP / HTTP, o el valor de cero si no hay nada que transmitir (return 0). Los parámetros de esta función son los siguientes:
 - remoteHost: Dirección IP del cliente.
 - remotePort: Puerto TCP del cliente.
 - localPort: Puerto al que se envía la solicitud.

- **SPI_Ethernet_getByte:** Lectura de un Byte recibido por el ENC28J60. Cada uno de estos Bytes forman parte de la solicitud recepcionada y debe interpretarse independientemente al tipo de solicitud que sea.

Las páginas web que se enviarán como respuesta a las peticiones del usuario remoto se los realizan mediante la función SPI_Ethernet_UserTCP, enviando el contenido de los vectores que alojan los caracteres que componen las páginas web.

5.9. PROGRAMA DEL SUBSISTEMA CONTROL CENTRAL

El Firmware del subsistema Microservidor Web se comunica mediante los puertos B y D digitales, enviando y recibiendo los datos. El subsistema Control Central hace lo propio, ya que ambos subsistemas actualizan sus registros de Sensores y Actuadores internamente entre si empleando dichos puertos.

La modificación de estos registros Actuadores se realiza dependiendo de los cambios que el usuario desea realizar en los estados de actuadores

específicos o activación de servicios. El origen de estos cambios se da en los subsistemas de comunicación con el usuario como ser el Microservidor Web y Bluetooth Local.

Los registros de los Sensores se modifican dependiendo de los estados actuales de los mismos. El origen de estos cambios se da en el subsistema Control Central pues es el que administra estos dispositivos. El programa que ejecuta este subsistema se aprecia en las figuras 5.42 y 5.43.

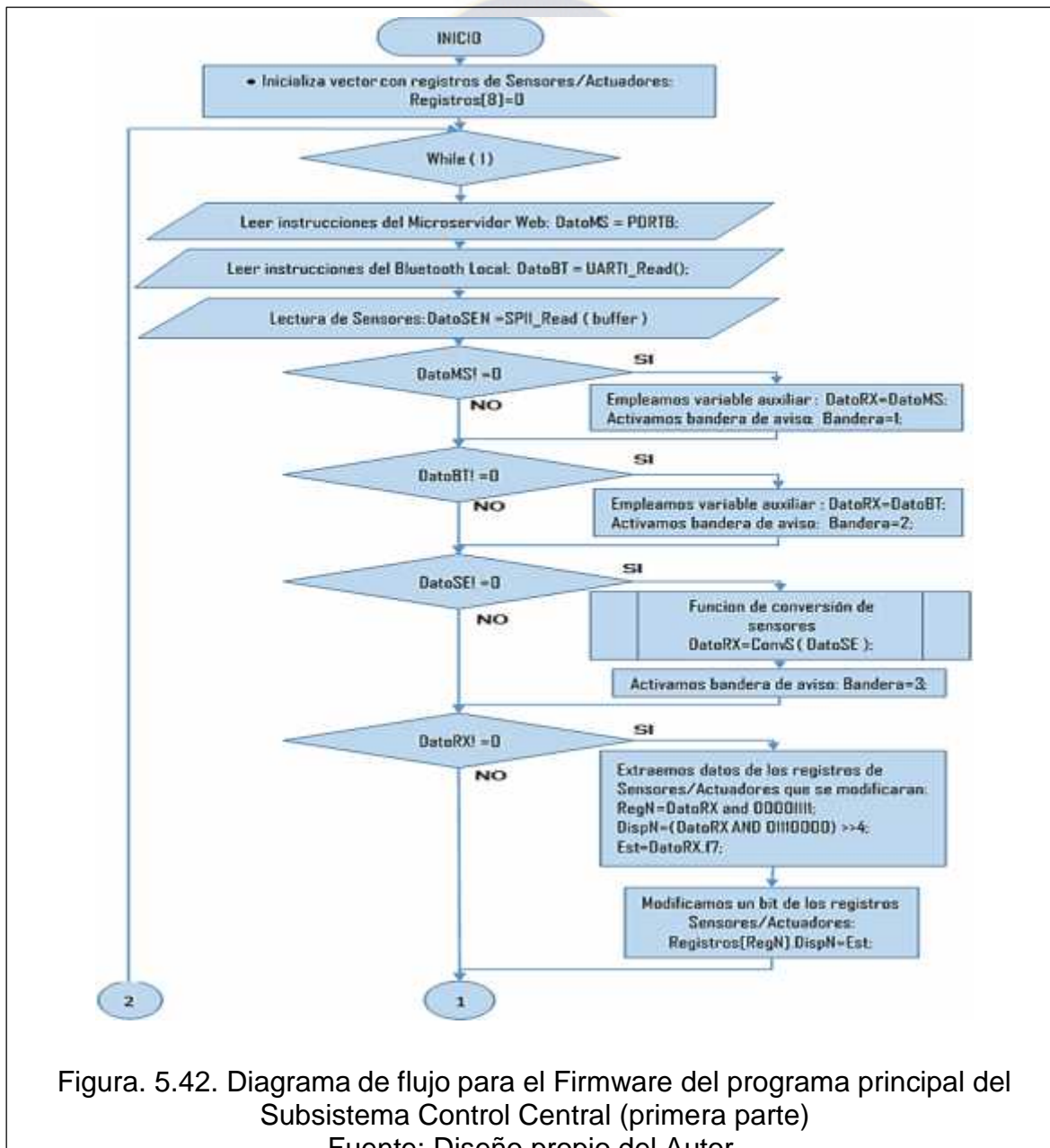


Figura. 5.42. Diagrama de flujo para el Firmware del programa principal del Subsistema Control Central (primera parte)

Fuente: Diseño propio del Autor.

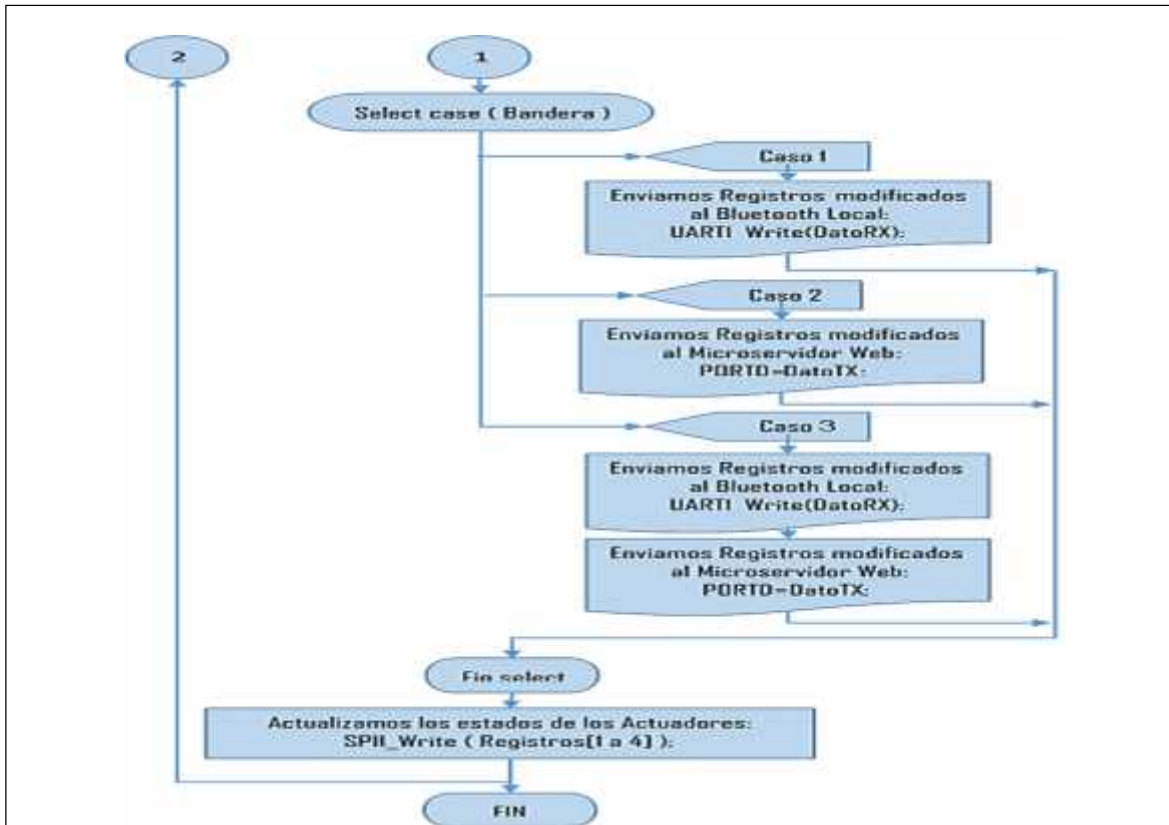


Figura. 5.43. Diagrama de flujo para el Firmware del programa principal del Subsistema Control Central (segunda parte)
Fuente: Diseño propio del Autor.

Las funciones más relevantes empleadas en el Firmware son las siguientes:

SPIx_Read= Lectura de un byte desde el bus SPI. El subsistema emplea esta función para la lectura de puertos de entrada expandidos para los sensores.

SPIx_Write= Escritura de un byte mediante el bus SPI. El subsistema emplea esta función para la escritura de puertos de salida expandidos para los actuadores.

UARTx_Read= Recibe un byte a través del puerto serial UART. El subsistema utiliza esta función para recibir datos enviados desde el módulo Bluetooth que tuvo como origen la aplicación empleada por el usuario.

UARTx_Write= Envía un byte a través del puerto serial UART. El subsistema utiliza esta función para enviar datos al módulo Bluetooth con destino final hacia el usuario.

5.10. PROGRAMA DE LA APLICACIÓN PARA EL BLUETOOTH LOCAL

La aplicación que dispondrá el Smartphone del usuario presenta un algoritmo de funcionamiento. Inicialmente se deberá conectar el dispositivo con el módulo HC-05 que es la interfaz Bluetooth del sistema domótico.

El código empleado se lo realizó mediante programación por bloques, empleando el programa APP Inventor. Sin embargo, el programa se puede describir mediante el siguiente diagrama de flujo.

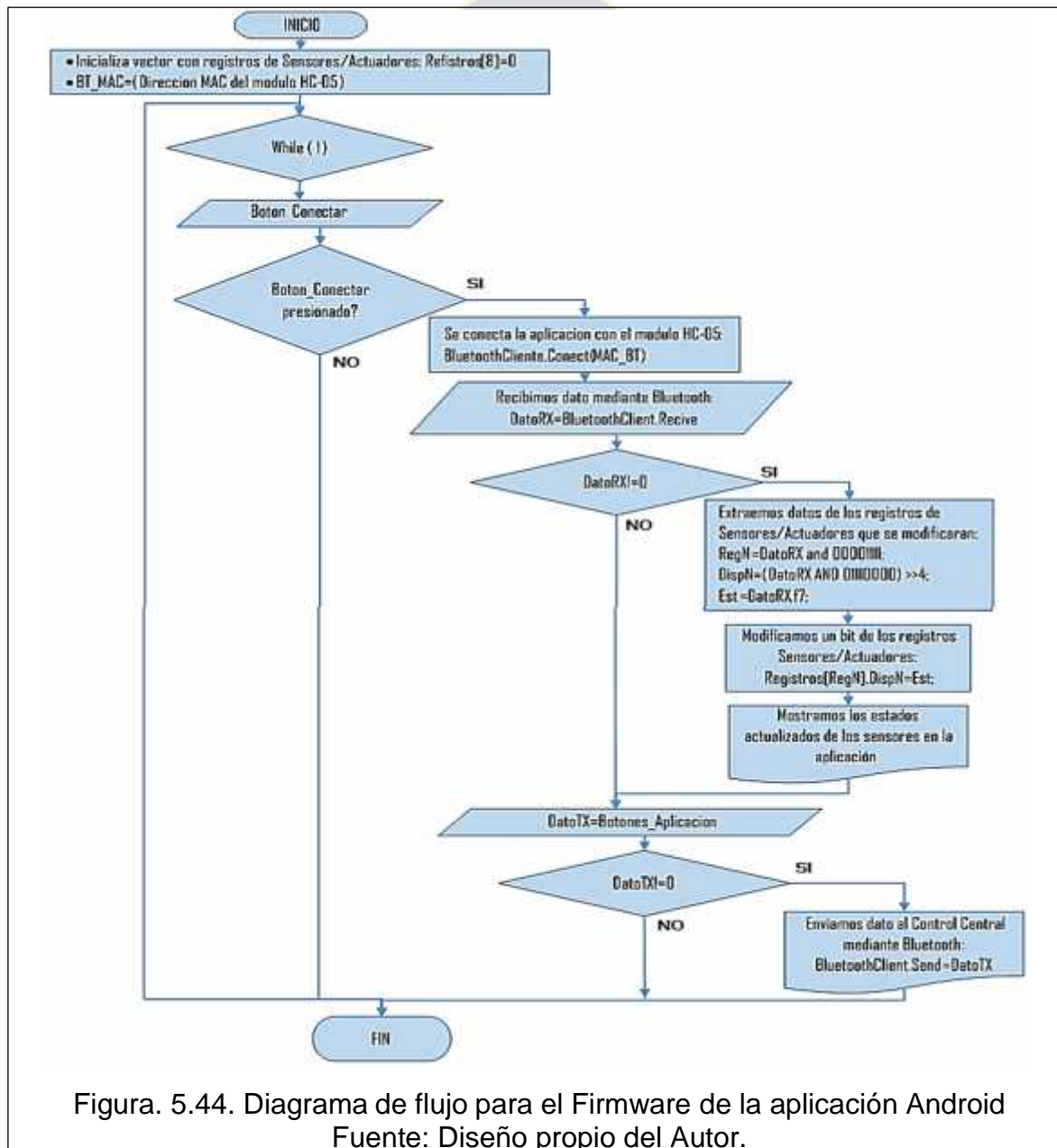


Figura. 5.44. Diagrama de flujo para el Firmware de la aplicación Android
Fuente: Diseño propio del Autor.

5.11. TOPOLOGIA DE LA RED ETHERNET DE ACCESO REMOTO

La red Ethernet que empleara el sistema tiene una estructura definida, la cual se visualiza en la figura 5.45.

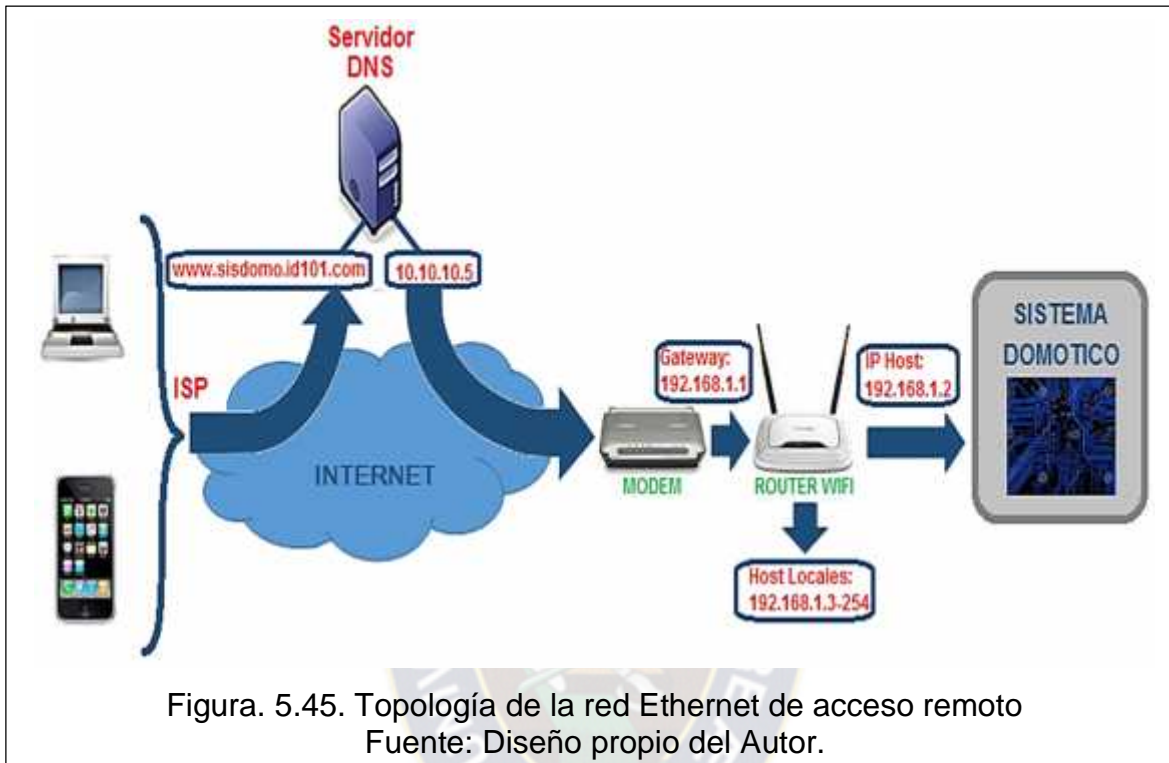


Figura. 5.45. Topología de la red Ethernet de acceso remoto
Fuente: Diseño propio del Autor.

El usuario empleara un proveedor de servicios de Internet, cuya infraestructura permite que el mismo acceda desde cualquier área de cobertura. Para que se realice una traducción de nombre de la dirección URL del sistema domótico a una dirección IP pública se emplea un servidor DNS para este fin.

El módem local de la vivienda recibirá todo paquete externo, teniendo como IP local a la puerta de enlace (Gateway) de toda la red de la casa. El router inalámbrico proveerá una cobertura WIFI para los dispositivos host de usuario, siendo el sistema domótico el primero y más importante dispositivo de la red.

El router WIFI será el encargado de enlazar al sistema domótico con el servidor DNS al emplear herramientas como ser Dynamic DNS y Virtual Server, los cuales dispone internamente y funcionan previa configuración del mismo.

5.12. VISUALIZACION DE LA INTERFAZ DE USUARIO REMOTO

La interfaz gráfica del usuario remoto es simple y fácil de usar, ya que accederá al sistema empleando un navegador web y en ella estarán todos los controles disponibles para su control. El menú principal y las páginas de cada área de control son los que se observan a continuación.



Figura. 5.46. Interfaz gráfica de la página menú Principal
Fuente: Diseño propio del Autor.



Figura. 5.47. Interfaz gráfica de la página menú Sala
Fuente: Diseño propio del Autor.



Figura. 5.48. Interfaz gráfica de la página menú Cocina
Fuente: Diseño propio del Autor.



Figura. 5.49. Interfaz gráfica de la página menú Dormitorio
Fuente: Diseño propio del Autor.



Figura. 5.50. Interfaz gráfica de la página menú Exterior
Fuente: Diseño propio del Autor.



Figura. 5.51. Interfaz gráfica de la página menú Seguridad
Fuente: Diseño propio del Autor.

5.13. VISUALIZACION DE LA INTERFAZ DE USUARIO LOCAL

El usuario local utiliza la interfaz que brinda la aplicación Android que previamente tendrá instalada en su Smartphone, estando en ella todos los controles disponibles para el control domótico. El menú principal y las páginas de cada área de control son los que se observan a continuación.



Figura. 5.52. Interfaz gráfica de la aplicación en el menú Principal
Fuente: Diseño propio del Autor.



Figura. 5.53. Interfaz gráfica de la aplicación en el menú Sala
Fuente: Diseño propio del Autor.



Figura. 5.54. Interfaz gráfica de la aplicación en el menú Cocina
Fuente: Diseño propio del Autor.



Figura. 5.55. Interfaz gráfica de la aplicación en el menú Dormitorio
Fuente: Diseño propio del Autor.



Figura. 5.56. Interfaz gráfica de la aplicación en el menú Exterior
Fuente: Diseño propio del Autor.



Figura. 5.57. Interfaz gráfica de la aplicación en el menú Seguridad
Fuente: Diseño propio del Autor.

CAPITULO VI

6. ANALISIS DE COSTOS.

Los costos en los cuales incurrimos para la elaboración del presente proyecto de grado se detallan a continuación

6.1. COSTOS DIRECTOS

6.1.1. COSTOS DE LA INFORMACION TECNICA.

La información técnica utilizada fue obtenida de varias fuentes entre ellas libros, impresiones de textos virtuales y la web.

6.1.2. COSTO DE SENSORES Y COMPONENTES ELECTRONICOS

Nro	DETALLE	CANTIDAD	COSTO UNITARIO (Bs)	COSTO TOTAL (Bs)
4	PIC16F877	2	50	100
5	PIC18F4550	1	65	65
	LM3940	1	8	8
	74HC08	1	3	3
	ENC28J60	1	70	70
	74HC595	4	7	28
	74HC165	4	7	28
	Modulo Bluetooth HC-05	1	110	110
	Kit de sensores y actuadores	1	350	350
15	Cristal 4Mhz	3	10	30
	Conector RJ-45 hembra	1	7	7
18	Pulsadores	2	2	4

19	Resistores	47	0.30	14
20	Capacitores Electrolíticos	2	2	4
22	Capacitores Cerámicos	9	2	18
23	Diodos led	3	1	3
29	Zócalos de 8 pines	10	1	10
32	Placa PCB	3	30	90
	Panel para pasarela residencial	1	60	60
37	Otros	1	40	40
TOTAL				1042

Tabla 6.1. Costo para los componentes electrónicos
Fuente: Proforma de cotización.

En la tabla 6.1 se muestra el costo de los componentes, sensores y actuadores electrónicos utilizados en el presente proyecto, cuyos datos fueron obtenidos de la proforma de cotización del distribuidor de componentes

6.2. COSTO INDIRECTO

El presente proyecto de grado fue desarrollado en el laboratorio de la carrera de Electrónica y Telecomunicaciones de la Facultad de Tecnología, por lo que no incurrimos en los costos de las licencias de los programas informáticos.

6.3. COSTO DE DISEÑO DE HARDWARE Y SOFTWARE

El diseño de todo el sistema genero una inversión de costo, los cuales son mostrados en la tabla 6.2.

Nro	DETALLE	COSTO TOTAL APROXIMADO (Bs)
1	Hardware	1700
2	Software	1800
TOTAL		3500

Tabla 6.2. Costo de Diseño

6.4. COSTO TOTAL

El costo total para la implementación del presente proyecto de grado es mostrado en la tabla 6.3, el costo total es la suma de todos los costos independientes de los cuales puede notarse que el más elevado es el costo de las licencias de legalidad de los programas informáticos.

Nro	DETALLE	COSTOS (Bs)
1	Información Técnica	350
2	Componentes Electrónicos	1042
3	Licencias de Legalidad	1743
4	Diseño de Hardware y Software	3500
TOTAL		6635 (Bs)

Tabla 6.3. Costo total para la implementación del proyecto

Aun tomando el costo total de 8580 Bs, la implementación del presente proyecto como prototipo para uso comercial es factible pues los costos de este tipo de sistemas electrónicos tranquilamente superan los 15.000 Bs aproximadamente, no se incluyen marcas por razones éticas.

CAPITULO VII

7.1. CONCLUSIONES

De acuerdo a los resultados obtenidos en las pruebas de funcionamiento, se diseñó un prototipo funcional de un sistema sistema de control y monitoreo Domótico con dos modalidades de acceso. De forma remota mediante la WEB compatible con redes y dispositivos que empleen direccionamiento IP y la segunda de forma local con interfaz inalámbrica para dispositivos Android, alcanzando los objetivos propuestos en el desarrollo del proyecto con las siguientes conclusiones.

- ✓ El diseño y la implementación de un prototipo de control y monitoreo Domótico con acceso remoto y local resultó ser un sistema funcional alcanzando las expectativas iniciales.
- ✓ La implementación de un circuito que permita a un sistema electrónico acceder a una conexión Ethernet provee una interfaz segura y rápida para la comunicación remota entre el usuario y su vivienda.
- ✓ El Firmware diseñado para la transmisión y recepción de tramas Ethernet es compatibles con el estándar IEEE 802.3 permitiendo una conexión segura y estable con redes IP.
- ✓ El diseño de varias páginas WEB dinámicas embebidas en el microcontrolador PIC18F4620 ofrecen una interfaz directa al sistema instalado en la vivienda empleando el protocolo HTTP.
- ✓ La implementación en hardware de un subsistema de conexión local inalámbrica vía Bluetooth brinda al usuario comodidad en el manejo de actuadores y monitoreo de sensores al funcionar correctamente.
- ✓ Se diseñó la aplicación Android compatible con el sistema propuesto.
- ✓ La implementación de sensores y actuadores en la vivienda cumplió de forma factible la función de control y monitoreo de la misma al estar directamente conectada al sistema domótico.

7.2. RECOMENDACIONES

De acuerdo a los resultados obtenidos en el diseño e implementación, así como en las pruebas de funcionamiento del proyecto se realizan las respectivas recomendaciones.

- ✓ Al emplear microcontroladores con puertos integrados y no externos se tiene un mejor manejo de la interfaz, reduciendo el riesgo de falla por conexión. EL PIC18F97J60 dispone de forma integrada la interfaz Ethernet y se recomienda su uso para una mejor versión del sistema.
- ✓ Se recomienda utilizar sensores con mayor potencia en el voltaje de salida.
- ✓ Se recomienda el empleo de sistemas RTOs en el Firmware de cada microcontrolador, para priorizar funciones y procedimientos en tiempo real.
- ✓ Para una mejor cobertura Bluetooth se recomienda el empleo de módulos de comunicación de Clase 1, mejorando únicamente la potencia de transmisión.

7.3. BIBLIOGRAFIA

[1].- Fernando Saéz Vacas. (2006) “Domótica: Un enfoque sociotecnico”. Editorial: E.T.S.I.

[2].- Oscar Antonio Gerometta. (2011) “Guía de preparación para el examen de certificación CCNA”. Editorial: Libronauta.

[3].- Claudio Gutiérrez Gallardo. (2008) “Cómo funciona la web”. Editorial: LOM editores.

[4].- Carlos I. Camargo Bareño. (2011) “Implementación de sistemas embebidos”. Editorial: Ra-Ma.

[5].- Romero Morales C. (2008) “Domotica e Inmotica”. Editorial: Alfaomega Ra-Ma.

[6].- Juan Ricardo Clavijo Mendoza. (2011) “Diseño y simulación de sistemas microcontrolados en lenguaje C”. Editor: Dr. Duilio Cruz Becerra.

Páginas Web

[7].- http://www.ie.itcr.ac.cr/acotoc/CISCO/R&S%20CCNA1/R&S_CCNA1_ITN_Chapter5_Ethernet

Página referencial de dispositivos CISCO. Última visita: marzo 2015.

[8].- www.alldatasheet.com.

Hojas de datos de componentes electrónicos. Última visita: mayo 2015.

[9].- <https://www.arduino.cc/en/Main>.

Página de referencia de dispositivos Arduino. Última visita: mayo 2015.

[10].- <https://www.mikroe.com/compilers>.

Detalles del compilador MikroC. Última visita: febrero 2015.

[11].- <http://www.emb.cl/electroindustria/articulo.mvc>.

Comparación de interfaces inalámbricas. Última visita: junio 2015.

[12].- <http://www.bluetooth.com/Pages/Basics.aspx>.

Información del estándar Bluetooth. Última visita: febrero 2015.

[13].- <http://ordenador.wingwit.com/Redes/wireless-networking>.

Redes de área personal. Última visita: mayo 2015.

[14].- <http://abc-rc.pl/templates/images/files/995/1425483439-hc-06-datasheet>.

Características del módulo HC-06. Última visita: mayo 2015.

7.4. ANEXOS

7.4.1. CODIGO DEL PIC18F4620 DEL MICROSERVIDOR WEB

```
////////////////////////////////////// Inicializacion ////////////////////////////////////////

#define Spi_Ethernet_FULLDUPLEX    0x01           //Definimos comunicacion FullDuplex

sfr sbit SPI_Ethernet_Rst at LATC0_bit;           //Asignamos pines para el ENC28J60
sfr sbit SPI_Ethernet_CS at LATC1_bit;
sfr sbit SPI_Ethernet_Rst_Direction at TRISC0_bit;
sfr sbit SPI_Ethernet_CS_Direction at TRISC1_bit;

const code unsigned char httpHeader[] = "HTTP/1.1 200 OK\nContent-type: "; //Encabezado de la respuesta
HTTP
const code unsigned char httpMimeTypeHTML[] = "text/html\n\n";
const code unsigned char httpMimeTypeScript[] = "text/plain\n\n";
unsigned char httpMethod[] = "GET /";           //Empleamos metodo GET
unsigned int banderita=1;

////////////////////////////////////// Pagina HTML del menú Principal ////////////////////////////////////////

const char *paginamenu =
"<HTML><HEAD><TITLE>PROYECTO DE GRADO UMSA</TITLE></HEAD>\
<BODY BGCOLOR=#4771A5 TEXT=\"BLACK\">\
<HR COLOR=#8CFACA SIZE=10>\
<FONT SIZE=6 COLOR=#B9FF4D FACE=AHARONI><l>SISTEMA DE CONTROL Y MONITOREO\
DOMOTICO</l></FONT>\
<HR COLOR=#FFFFFF SIZE=10><br><HR COLOR=#78C7EB SIZE=2><HR COLOR=#C3C8CD SIZE=2>\
<CENTER><FONT SIZE=8 COLOR=#FF9F00 FACE=AHARONI><U>MENU PRINCIPAL</U></FONT>\
<HR COLOR=#C3C8CD SIZE=2><HR COLOR=#78C7EB SIZE=2><BR>\
<FONT SIZE=3 COLOR=#FFD02A FACE=AHARONI>BIENVENIDO. ACCEDA AL CONTROL Y\
MONITOREO DE</FONT><BR>\
<FONT SIZE=3 COLOR=#FFD02A FACE=AHARONI>LA VIVIENDA INGRESANDO AL MENU DE CADA\
HABITACION</FONT>\
<BR><BR><table border=10 CELLSPACING=10>\
<tr><td bgcolor=#07B9FC><CENTER><FONT SIZE=5 FACE=AHARONI><a\
href=/S>SALA</a></FONT></CENTER></td></tr>\
<tr><td bgcolor=#FFD933><CENTER><FONT SIZE=5 FACE=AHARONI><a\
href=/C>COCINA</a></FONT></CENTER></td></tr>\
<tr><td bgcolor=#12F729><CENTER><FONT SIZE=5 FACE=AHARONI><a\
href=/D>DORMITORIO</a></FONT></CENTER></td></tr>\
<tr><td bgcolor=#AAD1B7><CENTER><FONT SIZE=5 FACE=AHARONI><a\
href=/E>EXTERIOR</a></FONT></CENTER></td></tr>\
<tr><td bgcolor=#FF3E33><CENTER><FONT SIZE=5 FACE=AHARONI><a\
href=/X>SEGURIDAD</a></FONT></CENTER></td></tr>\
</table></CENTER><HR COLOR=#78C7EB SIZE=2><HR COLOR=#C3C8CD SIZE=2></BODY></HTML>";

////////////////////////////////////// Pagina HTML del menú Sala ////////////////////////////////////////

const char *paginasala1 =
"<HTML><HEAD><TITLE>PROYECTO DE GRADO UMSA</TITLE></HEAD>\
<BODY BGCOLOR=#4771A5 TEXT=\"BLACK\">\
<HR COLOR=#78C7EB SIZE=10>\
<FONT SIZE=6 COLOR=#FFFFFF FACE=AHARONI><l>SISTEMA DE CONTROL Y MONITOREO\
DOMOTICO</l></FONT>\
<HR COLOR=#C3C8CD SIZE=10><br><HR COLOR=#78C7EB SIZE=2><HR COLOR=#C3C8CD SIZE=2>\
<CENTER><FONT SIZE=8 COLOR=#FFD02A FACE=AHARONI><U>SALA</U></FONT></CENTER>\
<HR COLOR=#C3C8CD SIZE=2><HR COLOR=#78C7EB SIZE=2>\
<FONT SIZE=5 COLOR=#FFD02A FACE=AHARONI><U>CONTROL DE ILUMINACION:</U></FONT>\
<script src=/s></script>
```

```

<script>
var str;
str="";
str+="<CENTER><table border=4><tr><td bgcolor=#96B1D2><a href=/o101>ENCENDER</a></td>";
str+="<td bgcolor=#2A6AFF><a href=/o100><BR>APAGAR<BR><BR></a></td></tr>";
if(SALA&(1<<0)){str+="<td bgcolor=#FFFF00><B>ENCENDIDO</B></td></tr>";}
else {str+="<td bgcolor=#FFCC99><B>APAGADO</B></td></tr>";}
str+="</table></CENTER>";
str+="<FONT SIZE=5 COLOR=#FFD02A FACE=AHARONI><U>CONTROL DE LUCES
RGB:</U></FONT>";
str+="<CENTER><table border=4><tr><td bgcolor=RED><a href=/t0><BR>ROJO<BR><BR></a></td>";
str+="<td bgcolor=#FFD02A><a href=/o111>VERDE</a></td>";
str+="<td bgcolor=#00FF00><a href=/o121>AZUL</a></td>";
str+="<td bgcolor=#808080><a href=/o131>APAGAR</a></td>";
str+="</table></CENTER>";

```

```

const char *paginasala2 =
"str+="<FONT SIZE=5 COLOR=#FFD02A FACE=AHARONI><U>APERTURA Y CIERRE DE
PERSIANA:</U></FONT>";
str+="<CENTER><table border=4><tr><td bgcolor=#96B1D2><a href=/o141>ABRIR</a></td>";
str+="<td bgcolor=#2A6AFF><a href=/o140><BR>CERRAR<BR><BR></a></td></tr>";
if(SALA&(1<<1)){str+="<td bgcolor=#FFFF00><B>ABIERTO</B></td></tr>";}
else {str+="<td bgcolor=#808080><B>CERRADO</B></td></tr>";}
str+="</table></CENTER>";
str+="<FONT SIZE=5 COLOR=#FFD02A FACE=AHARONI><U>CONTROL DE TOMAS:</U></FONT>";
str+="<CENTER><table border=4><tr><td bgcolor=#96B1D2><a href=/o151>ACTIVAR</a></td>";
str+="<td bgcolor=#2A6AFF><a href=/o150><BR>DESACTIVAR<BR><BR></a></td></tr>";
if(SALA&(1<<2)){str+="<td bgcolor=#FFFF00><B>ACTIVADO</B></td></tr>";}
else {str+="<td bgcolor=#808080><B>DESACTIVADO</B></td></tr>";}
str+="</table></CENTER>";
document.write(str);
</script>
<script src=/t></script>
<FONT SIZE=5 COLOR=#FFD02A FACE=AHARONI><U>TEMPERATURA AMBIENTE:</U></FONT>
<CENTER><table border=4><tr><th bgcolor=#00CCFF>TEMPERATURA</th>
<tr><td bgcolor=#00FF00><FONT SIZE=10 COLOR=#FFD02A
FACE=AHARONI><script>document.write(AN0)</script> °C</FONT></td>
</table></CENTER></BODY></HTML>";

```

//////////////////////////////////// Pagina HTML del menú Cocina //////////////////////////////////////

```

const char *paginacocina =
"<HTML><HEAD><TITLE>PROYECTO DE GRADO UMSA</TITLE></HEAD>
<BODY BGCOLOR=#FEDD27 TEXT="BLACK"><HR COLOR=#DF2823 SIZE=10>
<FONT SIZE=6 COLOR=#902147 FACE=AHARONI><I>SISTEMA DE CONTROL Y MONITOREO
DOMOTICO</I></FONT>
<HR COLOR=#F76A04 SIZE=10><br><HR COLOR=#DF2823 SIZE=2><HR COLOR=#F76A04
SIZE=2><CENTER>
<FONT SIZE=8 COLOR=#DF2823 FACE=AHARONI><U>COCINA</U></FONT></CENTER>
<HR COLOR=#F76A04 SIZE=2><HR COLOR=#DF2823 SIZE=2>
<FONT SIZE=5 COLOR=#DF2823 FACE=AHARONI><U>CONTROL DE ILUMINACION:</U></FONT>
<script src=/c></script>
<script>
var str;
str="";
str+="<CENTER><table border=4><tr><td bgcolor=#FBF621><a href=/o201>ENCENDER</a></td>";
str+="<td bgcolor=#FAC211><a href=/o200><BR>APAGAR<BR><BR></a></td></tr>";
if(COCINA&(1<<0)){str+="<td bgcolor=#FFFF00><B>ENCENDIDO</B></td></tr>";}
else {str+="<td bgcolor=#FFCC99><B>APAGADO</B></td></tr>";}
str+="</table></CENTER>";
str+="<FONT SIZE=5 COLOR=#DF2823 FACE=AHARONI><U>CONTROL DE TOMAS DE
CORRIENTE:</U></FONT>";

```



```

str+="<CENTER><table border=4><tr><td bgcolor=#FBF621><a href=/o211>ACTIVAR</a></td>\";\
str+="<td bgcolor=#FAC211><a href=/o210><BR>DESACTIVAR<BR><BR></a></td><td></td>\";\
if(COCINA&(1<<1)){str+="<td bgcolor=#FFFF00><B>ACTIVADO</B></td></tr>\";\
else {str+="<td bgcolor=#808080><B>DESACTIVADO</B></td></tr>\";\
str+="</table></CENTER>\";\
document.write(str);\
</script></CENTER></BODY></HTML>";

```

//////////////////////////////////// Pagina HTML del menú Dormitorio //////////////////////////////////////

```

const char *paginadorm1 =
"<HTML><HEAD><TITLE>PROYECTO DE GRADO UMSA</TITLE></HEAD>\
<BODY BGCOLOR=#A9E558 TEXT=\\"BLACK\\"><HR COLOR=#00519B SIZE=10>\
<FONT SIZE=6 COLOR=#006544 FACE=AHARONI><I>SISTEMA DE CONTROL Y MONITOREO\
DOMOTICO</I></FONT>\
<HR COLOR=#009543 SIZE=10><br><HR COLOR=#00519B SIZE=2><HR COLOR=#009543\
SIZE=2><CENTER>\
<FONT SIZE=6 COLOR=#002D62 FACE=AHARONI><U>DORMITORIO</U></FONT></CENTER>\
<HR COLOR=#009543 SIZE=2><HR COLOR=#00519B SIZE=2>\
<FONT SIZE=5 COLOR=#002D62 FACE=AHARONI><U>CONTROL DE ILUMINACION:</U></FONT>\
<script src=/d></script>\
<script>\
var str;\
str="";\
str+="<CENTER><table border=4><tr><td bgcolor=#96B1D2><a href=/o301>ENCENDER</a></td>\";\
str+="<td bgcolor=#2A6AFF><a href=/o300><BR>APAGAR<BR><BR></a></td><td></td>\";\
if(DOR&(1<<0)){str+="<td bgcolor=#FFFF00><B>ENCENDIDO</B></td></tr>\";\
else {str+="<td bgcolor=#FFCC99><B>APAGADO</B></td></tr>\";\
str+="</table></CENTER>\";";

```

```

const char *paginadorm2 =
"str+="<FONT SIZE=5 COLOR=#002D62 FACE=AHARONI><U>APERTURA Y CIERRE DE\
PERSIANA:</U></FONT>\";\
str+="<CENTER><table border=4><tr><td bgcolor=#96B1D2><a href=/o311>ABRIR</a></td>\";\
str+="<td bgcolor=#2A6AFF><a href=/o310><BR>CERRAR<BR><BR></a></td><td></td>\";\
if(DOR&(1<<1)){str+="<td bgcolor=#FFFF00><B>ABIERTO</B></td></tr>\";\
else {str+="<td bgcolor=#808080><B>CERRADO</B></td></tr>\";\
str+="</table></CENTER>\";\
str+="<FONT SIZE=5 COLOR=#002D62 FACE=AHARONI><U>CONTROL DE TOMAS:</U></FONT>\";\
str+="<CENTER><table border=4><tr><td bgcolor=#96B1D2><a href=/o321>ACTIVAR</a></td>\";\
str+="<td bgcolor=#2A6AFF><a href=/o320><BR>DESACTIVAR<BR><BR></a></td><td></td>\";\
if(DOR&(1<<2)){str+="<td bgcolor=#FFFF00><B>ACTIVADO</B></td></tr>\";\
else {str+="<td bgcolor=#808080><B>DESACTIVADO</B></td></tr>\";\
str+="</table></CENTER>\";\
document.write(str);\
</script><script src=/t></script>\
<FONT SIZE=5 COLOR=#002D62 FACE=AHARONI><U>TEMPERATURA AMBIENTE:</U></FONT>\
<CENTER><table border=4><tr><th bgcolor=#00CCFF>TEMPERATURA</th>\
<tr><td bgcolor=#00FF00><FONT SIZE=10 COLOR=#FFD02A\
FACE=AHARONI><script>document.write(AN1)</script> °C</FONT></td>\
</table></CENTER></BODY></HTML>";

```

//////////////////////////////////// Pagina HTML del menú Exterior //////////////////////////////////////

```

const char *paginaext1 =
"<HTML><HEAD><TITLE>PROYECTO DE GRADO UMSA</TITLE></HEAD>\
<BODY BGCOLOR=#6EFFFF TEXT="BLACK"><HR COLOR=#FFAD2F SIZE=10>\
<FONT SIZE=6 COLOR=#7D0996 FACE=AHARONI><I>SISTEMA DE CONTROL Y MONITOREO\
DOMOTICO</I></FONT>\
<HR COLOR=#A9FF2F SIZE=10><br>\
<HR COLOR=#FFAD2F SIZE=2><HR COLOR=#A9FF2F SIZE=2>\

```

```

<CENTER><FONT SIZE=6 COLOR=#902147 FACE=AHARONI><U>EXTERIOR</U></FONT></CENTER>\
<HR COLOR=#A9FF2F SIZE=2><HR COLOR=#FFAD2F SIZE=2>\
<FONT SIZE=5 COLOR=#902147 FACE=AHARONI><U>CONTROL DE LUMINARIAS:</U></FONT>\
<script src=/e></script>\
<script>\
var str;\
str=""\;\
str+="<CENTER><table border=4><tr><td bgcolor=#96B1D2><a href=/o401>ENCENDER</a></td>\";\
str+="<td bgcolor=#2A6AFF><a href=/o400><BR>APAGAR<BR><BR></a></td><td></td>\";\
if(EXT&(1<<0)){str+="<td bgcolor=#FFFF00><B>ENCENDIDO</B></td></tr>\";\
else {str+="<td bgcolor=#FFCC99><B>APAGADO</B></td></tr>\";\
str+="</table></CENTER>\";\
str+="<FONT SIZE=5 COLOR=#902147 FACE=AHARONI><U>APERTURA Y CIERRE DE GARAJE:</U></FONT>\";\
str+="<CENTER><table border=4><tr><td bgcolor=#96B1D2><a href=/o411>ABRIR</a></td>\";\
str+="<td bgcolor=#2A6AFF><a href=/o410><BR>CERRAR<BR><BR></a></td><td></td>\";\
if(EXT&(1<<1)){str+="<td bgcolor=#FFFF00><B>ABIERTO</B></td></tr>\";\
else {str+="<td bgcolor=#808080><B>CERRADO</B></td></tr>\";\
str+="</table></CENTER>\";\
str+="<FONT SIZE=5 COLOR=#902147 FACE=AHARONI><U>CONTROL DE RIEGO:</U></FONT>\";\
str+="<CENTER><table border=4><tr><td bgcolor=#96B1D2><a href=/o421>ACTIVAR</a></td>\";\
str+="<td bgcolor=#2A6AFF><a href=/o420><BR>DESACTIVAR<BR><BR></a></td><td></td>\";\
if(EXT&(1<<2)){str+="<td bgcolor=#FFFF00><B>ACTIVADO</B></td></tr>\";\
else {str+="<td bgcolor=#808080><B>DESACTIVADO</B></td></tr>\";\
str+="</table></CENTER>\";\
document.write(str);\
</script></BODY></HTML>";

```

//////////////////////////////////// Pagina HTML del menú Seguridad //////////////////////////////////////

```

const char *paginaseg =
"<HTML><HEAD><TITLE>PROYECTO DE GRADO UMSA</TITLE></HEAD>\
<BODY BGCOLOR=#FF2A2C TEXT="BLACK"><HR COLOR=#7C144D SIZE=10>\
<FONT SIZE=6 COLOR=#7F4233 FACE=AHARONI><I>SISTEMA DE CONTROL Y MONITOREO DOMOTICO</I></FONT>\
<HR COLOR=#6C5C32 SIZE=10><br>\
<HR COLOR=#7C144D SIZE=2><HR COLOR=#6C5C32 SIZE=2>\
<CENTER><FONT SIZE=8 COLOR=#000000 FACE=AHARONI><U>SEGURIDAD</U></FONT></CENTER>\
<HR COLOR=#6C5C32 SIZE=2><HR COLOR=#7C144D SIZE=2>\
<FONT SIZE=5 COLOR=#000000 FACE=AHARONI><U>ACTIVAR SISTEMA CONTRA INTRUSOS:</U></FONT>\
<script src=/x></script>\
<script>\
var str;\
str=""\;\
str+="<CENTER><table border=4><tr><td bgcolor=#FBF621><a href=/o801>ACTIVAR</a></td>\";\
str+="<td bgcolor=#FAC211><a href=/o800><BR>DESACTIVAR<BR><BR></a></td><td></td>\";\
if(SEG&(1<<0)){str+="<td bgcolor=#FFFF00><B>ACTIVADO</B></td></tr>\";\
else {str+="<td bgcolor=#FFCC99><B>DESACTIVADO</B></td></tr>\";\
str+="</table></CENTER>\";\
str+="<FONT SIZE=5 COLOR=#000000 FACE=AHARONI><U>ACTIVAR SIMULADOR DE PRESENCIA:</U></FONT>\";\
str+="<CENTER><table border=4><tr><td bgcolor=#FBF621><a href=/o811>ACTIVAR</a></td>\";\
str+="<td bgcolor=#FAC211><a href=/o810><BR>DESACTIVAR<BR><BR></a></td><td></td>\";\
if(SEG&(1<<1)){str+="<td bgcolor=#FFFF00><B>ACTIVADO</B></td></tr>\";\
else {str+="<td bgcolor=#808080><B>DESACTIVADO</B></td></tr>\";\
str+="</table></CENTER>\";\
document.write(str) ;\
</script></CENTER></BODY></HTML>\";\

```

```
////////////////////////////////// Direccionamiento IP //////////////////////////////////////
```

```
unsigned char myMacAddr[6] = {0x00, 0x14, 0xA5, 0x76, 0x19, 0x3f} ; // MAC  
unsigned char myIpAddr[4] = {192, 168, 20, 60}; // IP  
unsigned char gwIpAddr[4] = {192, 168, 20, 1}; // gateway  
unsigned char ipMask[4] = {255, 255, 255, 0}; // mascara  
unsigned char dnsIpAddr[4] = {8, 8, 8, 8}; // DNS
```

```
unsigned char getRequest[15]; //Variables auxiliares  
unsigned char get_Request, digit_getRequest, etat_interrupteur;  
unsigned char dyna[29] ;  
unsigned long httpCounter = 0;
```

```
////////////////////////////////// Registros de sensores y actuadores //////////////////////////////////////
```

```
//Registros:  
//[1] Sala [5] Sensores perimetrales  
//[2] Cocina [6] Sensores de instrumentos  
//[3] Dormitorio [7] Sensores de falla tecnica  
//[4] Exterior [8] Sistemas de seguridad  
unsigned short registros[8];
```

```
////////////////////////////////// Variables auxiliares //////////////////////////////////////
```

```
unsigned short nreg,nbit,estb;
```

```
#define putConstString SPI_Ethernet_putConstString  
#define putString SPI_Ethernet_putString
```

```
////////////////////////////////// Funcion de respuesta TCP //////////////////////////////////////
```

```
unsigned int SPI_Ethernet_UserTCP(unsigned char *remoteHost, unsigned int remotePort, unsigned int  
localPort, unsigned int reqLength, char *canClose//, TEthPktFlags *flags)  
{  
    unsigned int len;
```

```
////////////////////////////////// Prevencion del uso del puerto 80 //////////////////////////////////////
```

```
    if(localPort != 80)  
    {  
        return(0) ;  
    }
```

```
    for(len = 0 ; len < 10 ; len++)  
    {  
        getRequest[len] = SPI_Ethernet_getByte() ;  
    }  
    getRequest[len] = 0 ;  
    len = 0;
```

```
    if(memcmp(getRequest, httpMethod, 5))  
    {  
        return(0) ;  
    }
```

```
    get_Request = getRequest[5];
```

```
////////////////////////////////// Enviamos temperatura actual a las páginas web //////////////////////////////////////
```

```
    if(get_Request == 't')  
    {  
        len = putConstString(httpHeader) ;
```

```
len += putConstString(httpMimeTypeScript) ;
```

```
WordToStr(ADC_Read(0), dyna);  
len += putConstString("var AN0=");  
len += putString(dyna);  
len += putConstString(";");
```

```
WordToStr(ADC_Read(1), dyna);  
len += putConstString("var AN1=");  
len += putString(dyna);  
len += putConstString(";");  
}
```

//////////////////////////////////// Enviamos estados de monitoreo de la vivienda a las páginas web //////////////////////////////////////

```
if(get_Request == 's')  
{  
len = putConstString(httpHeader) ;  
len += putConstString(httpMimeTypeScript) ;
```

```
len += putConstString("var SALA=") ;  
WordToStr(Registros[1], dyna);  
len += putString(dyna);  
len += putConstString(";");  
}
```

```
if(get_Request == 'c')  
{  
len = putConstString(httpHeader) ;  
len += putConstString(httpMimeTypeScript) ;
```

```
len += putConstString("var COCINA=") ;  
WordToStr(Registros[2], dyna);  
len += putString(dyna);  
len += putConstString(";");  
}
```

```
if(get_Request == 'd')  
{  
len = putConstString(httpHeader) ;  
len += putConstString(httpMimeTypeScript) ;
```

```
len += putConstString("var DOR=") ;  
WordToStr(Registros[3], dyna);  
len += putString(dyna);  
len += putConstString(";");  
}
```

```
if(get_Request == 'e')  
{  
len = putConstString(httpHeader) ;  
len += putConstString(httpMimeTypeScript) ;
```

```
len += putConstString("var EXT=") ;  
WordToStr(Registros[4], dyna);  
len += putString(dyna);  
len += putConstString(";");  
}
```

```
if(get_Request == 'x')  
{  
len = putConstString(httpHeader) ;
```

```

len += putConstString(httpMimeTypeScript) ;

len += putConstString("var SEG=") ;
WordToStr(Registros[8], dyna);
len += putString(dyna);
len += putConstString(";");
}

```

//////////////////////////////////// Recibimos acción de control URL del usuario //////////////////////////////////////

```

if(getRequest[5] == 'o')
{
    nreg=0;
    nbit=0;
    estb=0;
    switch(getRequest[6])
    {
        case '1': nreg=1;
        break;
        case '2': nreg=2;
        break;
        case '3': nreg=3;
        break;
        case '4': nreg=4;
        break;
        case '5': nreg=5;
        break;
        case '6': nreg=6;
        break;
        case '7': nreg=7;
        break;
        case '8': nreg=8;
        break;
    }

    switch(getRequest[7])
    {
        case '0': nbit=0;
        break;
        case '1': nbit=1;
        break;
        case '2': nbit=2;
        break;
        case '3': nbit=3;
        break;
        case '4': nbit=4;
        break;
        case '5': nbit=5;
        break;
        case '6': nbit=6;
        break;
        case '7': nbit=7;
        break;
    }

    if(getRequest[8] == '1')
    {
        estb=1;
    }
    else
    {

```



```

        estb=0;
    }

    switch(nbit)
    {
        case 1:registros[nbit].f0=estb;
        break;
        case 2:registros[nbit].f1=estb;
        break;
        case 3:registros[nbit].f2=estb;
        break;
        case 4:registros[nbit].f3=estb;
        break;
        case 5:registros[nbit].f4=estb;
        break;
        case 6:registros[nbit].f5=estb;
        break;
        case 7:registros[nbit].f6=estb;
        break;
        case 8:registros[nbit].f7=estb;
        break;
    }

    nbit= (nbit)<<4;
    estb= (estb)<<7;

    PORTB= nreg & nbit & estb; //Enviamos la informacion del
                                //actuador que cambiara de estado
                                //al Sub. Central de Control

    delay_ms(10);
}

////////////////////////////////// Estructuramos la paginas web solicitada //////////////////////////////////

if(get_Request == 'M')
{
    len = putConstString(httpHeader);
    len += putConstString(httpMimeTypeHTML);
    len += putConstString(paginamenu1);
    len += putConstString(paginamenu2);
}

if(get_Request == 'S')
{
    len = putConstString(httpHeader);
    len += putConstString(httpMimeTypeHTML);
    len += putConstString(paginasala1);
    len += putConstString(paginasala2);
}

    if(get_Request == 'C')
    {
        len = putConstString(httpHeader);
        len += putConstString(httpMimeTypeHTML);
        len += putConstString(paginacocina);
    }

    if(get_Request == 'E')
    {
        len = putConstString(httpHeader);
        len += putConstString(httpMimeTypeHTML);
    }
}

```

```

len += putConstString(paginaexterior);
}

    if(get_Request == 'X')
    {
len = putConstString(httpHeader);
len += putConstString(httpMimeTypeHTML);
len += putConstString(paginaseguridad);
    }

if(len == 0)
{
len = putConstString(httpHeader);
len += putConstString(httpMimeTypeHTML);
len += putConstString(paginamenu1);
len += putConstString(paginamenu2);
}

return(len);
}

unsigned int SPI_Ethernet_UserUDP(unsigned char *remoteHost, unsigned int remotePort, unsigned int
destPort, unsigned int reqLength, TEthPktFlags *flags)
{
return(0);
}

//////////////////////////////////// Programa Principal //////////////////////////////////////

void main()
{
PORTA = 0;
TRISA = 0xff;

PORTB = 0;
TRISB = 0;

PORTD = 0;
TRISD = 0xff;

ADC_CS = 1;
ADC_CS_Direction = 0;

SPI1_Init();
SPI_Rd_Ptr = SPI1_Read;
SPI_Ethernet_Init(myMacAddr, myIpAddr, Spi_Ethernet_FULLDUPLEX);
SPI_Ethernet_confNetwork(ipMask, gwIpAddr, dnsIpAddr);
while(1)
{
SPI_Ethernet_doPacket(); //Procesar los paquetes Ethernet entrantes

    if(PORTD!=0) //Actualización de registros locales
    { //enviados por el Sub. Control Central
nreg=PORTD & 0b00001111;
nbit=(PORTD & 0b011110000) >> 4;
estb=PORTD.f7;

switch(nbit)
{
case 1:registros[nbit].f0=estb;
break;
}
}
}
}

```

```

        case 2:registros[nbit].f1=estb;
        break;
        case 3:registros[nbit].f2=estb;
        break;
        case 4:registros[nbit].f3=estb;
        break;
        case 5:registros[nbit].f4=estb;
        break;
        case 6:registros[nbit].f5=estb;
        break;
        case 7:registros[nbit].f6=estb;
        break;
        case 8:registros[nbit].f7=estb;
        break;
    }
}
}

```

7.4.2. CODIGO DEL PIC16F877A DEL SUBSISTEMA CONTROL CENTRAL

```

//////////////////////////////////// Definimos variables //////////////////////////////////////
unsigned short DatoMS,DatoBT,DatoRX,DatoSEN1,DatoSEN2,DatoSEN3,DatoSEN4;
unsigned short buffer,;
unsigned short nreg,nbit,estb;

//////////////////////////////////// Registros de sensores y actuadores //////////////////////////////////////

//Registros:
//[1] Sala           [5] Sensores perimetrales
//[2] Cocina         [6] Sensores de intrusos
//[3] Dormitorio    [7] Sensores de falla tecnica
//[4] Exterior       [8] Sistemas de seguridad
unsigned short registros[8];

//////////////////////////////////// Programa Principal //////////////////////////////////////

void main() {

TRISD=0xFF;
TRISB=0x00;
TRISC=0xFF;

//////////////////////////////////// Inicializamos interfaz SPI //////////////////////////////////////

SPI1_Init_Advanced(_SPI_MASTER_OSC_DIV4, _SPI_DATA_SAMPLE_MIDDLE, _SPI_CLK_IDLE_LOW,
_SPI_LOW_2_HIGH);

////////////////////////////////////Inicializamos interfaz Serial con Modulo Bluetooth////////////////////////////////////

UART1_Init(9600);           // Initialize UART module at 9600 bps
Delay_ms(100);             // Wait for UART module to stabilize

DatoMS=0;

```



```

DatoBT=0;
DatoSEN1=0;
DatoSEN2=0;
DatoSEN3=0;
DatoSEN4=0;

while(1)
{
    DatoMS=PORTD;                //Recibimos dato del Sub. Microservidor web

    if (UART1_Data_Ready())      //Recibimos dato del Modulo Bluetooth
    {
        DatoBT = UART1_Read();    //Lectura del puerto serie
    }

    DatoSEN1=SPI1_read(buffer);   //Lectura de los 32 puertos de entrada (Sensores)
    DatoSEN2=SPI1_read(buffer);
    DatoSEN3=SPI1_read(buffer);
    DatoSEN4=SPI1_read(buffer);

    if(DatoMS!=0)                //Existe dato proveniente del Microservidor web
    {
        DatoRX=DatoMS;
        bandera=1;
    }

    if(DatoBT!=0)                //Existe dato proveniente del Microservidor web
    {
        DatoRX=DatoMS;
        bandera=2;
    }

    if(DatoSEN1!=0 || DatoSEN2!=0 || DatoSEN3!=0 || DatoSEN4!=0) //Existe dato proveniente
                                                                    //del Microservidor web
    {
        DatoRX=ConvS(DatoSEN1,DatoSEN2,DatoSEN3,DatoSEN4);
        bandera=3;
    }

    if(DatoRX!=0)                //Recibimos datos de Control
    {                             //enviados por el Sub. Microservidor web
        nreg=DatoRX & 0b00001111;
        nbit=(DatoRX & 0b01110000) >> 4;
        estb=DatoRX.f7;

        switch(nbit)
        {
            case 1:registros[nbit].f0=estb; //Actualizamos registros locales
            break;
            case 2:registros[nbit].f1=estb;
            break;
            case 3:registros[nbit].f2=estb;
            break;
            case 4:registros[nbit].f3=estb;
            break;
            case 5:registros[nbit].f4=estb;
            break;
            case 6:registros[nbit].f5=estb;
            break;
            case 7:registros[nbit].f6=estb;
            break;
            case 8:registros[nbit].f7=estb;
        }
    }
}

```

```

        break;
    }
}

switch(bandera)
{
    case 1:UART1_Write(DatoRX); //Actualizamos registros del Dispositivo Bluetooth
    break;
    case 2:PORTD=DatoRX; //Actualizamos registros del Microservidor web
    break;
    case 3:UART1_Write(DatoRX); //Actualizamos registros del Dispositivo Bluetooth
    PORTD=DatoRX; //Actualizamos registros del Microservidor web
    break;
}

SPI1_Write(Registros[1]); //Actualizamos estados de los actuadores
SPI1_Write(Registros[2]);
SPI1_Write(Registros[3]);
SPI1_Write(Registros[4]);
}

```



7.4.3. HOJA DE ESPECIFICACIONES DEL PIC16F877



PIC16F87XA

28/40/44-Pin Enhanced Flash Microcontrollers

Devices Included in this Data Sheet:

- PIC16F873A
- PIC16F874A
- PIC16F876A
- PIC16F877A

High-Performance RISC CPU:

- Only 35 single-word instructions to learn
- All single-cycle instructions except for program branches, which are two-cycle
- Operating speed: DC – 20 MHz clock input
DC – 200 ns instruction cycle
- Up to 8K x 14 words of Flash Program Memory, Up to 368 x 8 bytes of Data Memory (RAM), Up to 256 x 8 bytes of EEPROM Data Memory
- Pinout compatible to other 28-pin or 40/44-pin PIC16CXXX and PIC16FXXX microcontrollers

Peripheral Features:

- Timer0: 8-bit timer/counter with 8-bit prescaler
- Timer1: 16-bit timer/counter with prescaler, can be incremented during Sleep via external crystal/clock
- Timer2: 8-bit timer/counter with 8-bit period register, prescaler and postscaler
- Two Capture, Compare, PWM modules
 - Capture is 16-bit, max. resolution is 12.5 ns
 - Compare is 16-bit, max. resolution is 200 ns
 - PWM max. resolution is 10-bit
- Synchronous Serial Port (SSP) with SPI™ (Master mode) and I²C™ (Master/Slave)
- Universal Synchronous Asynchronous Receiver Transmitter (USART/SCI) with 9-bit address detection
- Parallel Slave Port (PSP) – 8 bits wide with external RD, WR and CS controls (40/44-pin only)
- Brown-out detection circuitry for Brown-out Reset (BOR)

Analog Features:

- 10-bit, up to 8-channel Analog-to-Digital Converter (A/D)
- Brown-out Reset (BOR)
- Analog Comparator module with:
 - Two analog comparators
 - Programmable on-chip voltage reference (V_{REF}) module
 - Programmable input multiplexing from device inputs and internal voltage reference
 - Comparator outputs are externally accessible

Special Microcontroller Features:

- 100,000 erase/write cycle Enhanced Flash program memory typical
- 1,000,000 erase/write cycle Data EEPROM memory typical
- Data EEPROM Retention > 40 years
- Self-reprogrammable under software control
- In-Circuit Serial Programming™ (ICSP™) via two pins
- Single-supply 5V In-Circuit Serial Programming
- Watchdog Timer (WDT) with its own on-chip RC oscillator for reliable operation
- Programmable code protection
- Power saving Sleep mode
- Selectable oscillator options
- In-Circuit Debug (ICD) via two pins

CMOS Technology:

- Low-power, high-speed Flash/EEPROM technology
- Fully static design
- Wide operating voltage range (2.0V to 5.5V)
- Commercial and Industrial temperature ranges
- Low-power consumption

Device	Program Memory		Data SRAM (Bytes)	EEPROM (Bytes)	I/O	10-bit A/D (ch)	CCP (PWM)	MSSP		USART	Timers 8/16-bit	Comparators
	Bytes	# Single Word Instructions						SPI	Master I ² C			
PIC16F873A	7.2K	4096	192	128	22	5	2	Yes	Yes	Yes	2/1	2
PIC16F874A	7.2K	4096	192	128	33	8	2	Yes	Yes	Yes	2/1	2
PIC16F876A	14.3K	8192	368	256	22	5	2	Yes	Yes	Yes	2/1	2
PIC16F877A	14.3K	8192	368	256	33	8	2	Yes	Yes	Yes	2/1	2

7.4.4. HOJA DE ESPECIFICACIONES DEL PIC18F4620



28/40/44-Pin Enhanced Flash Microcontrollers with 10-Bit A/D and nanoWatt Technology

Power Management Features:

- Run: CPU on, Peripherals on
- Idle: CPU off, Peripherals on
- Sleep: CPU off, Peripherals off
- Ultra Low 50nA Input Leakage
- Run mode Currents Down to 11 μ A Typical
- Idle mode Currents Down to 2.5 μ A Typical
- Sleep mode Current Down to 100 nA Typical
- Timer1 Oscillator: 900 nA, 32 kHz, 2V
- Watchdog Timer: 1.4 μ A, 2V Typical
- Two-Speed Oscillator Start-up

Flexible Oscillator Structure:

- Four Crystal modes, up to 40 MHz
- 4x Phase Lock Loop (PLL) – Available for Crystal and Internal Oscillators
- Two External RC modes, up to 4 MHz
- Two External Clock modes, up to 40 MHz
- Internal Oscillator Block:
 - Fast wake from Sleep and Idle, 1 μ s typical
 - 8 use-selectable frequencies, from 31 kHz to 8 MHz
 - Provides a complete range of clock speeds from 31 kHz to 32 MHz when used with PLL
 - User-tunable to compensate for frequency drift
- Secondary Oscillator using Timer1 @ 32 kHz
- Fail-Safe Clock Monitor:
 - Allows for safe shutdown if peripheral clock stops

Peripheral Highlights:

- High-Current Sink/Source 25 mA/25 mA
- Three Programmable External Interrupts
- Four Input Change Interrupts
- Up to 2 Capture/Compare/PWM (CCP) modules, one with Auto-Shutdown (28-pin devices)
- Enhanced Capture/Compare/PWM (ECCP) module (40/44-pin devices only):
 - One, two or four PWM outputs
 - Selectable polarity
 - Programmable dead time
 - Auto-shutdown and auto-restart

Peripheral Highlights (Continued):

- Master Synchronous Serial Port (MSSP) module Supporting 3-Wire SPI (all 4 modes) and I²C™ Master and Slave modes
- Enhanced Addressable USART module:
 - Supports RS-485, RS-232 and LIN/J2602
 - RS-232 operation using internal oscillator block (no external crystal required)
 - Auto-wake-up on Start bit
 - Auto-Baud Detect
- 10-Bit, up to 13-Channel Analog-to-Digital (A/D) Converter module:
 - Auto-acquisition capability
 - Conversion available during Sleep
- Dual Analog Comparators with Input Multiplexing
- Programmable 16-Level High/Low-Voltage Detection (HLVD) module:
 - Supports interrupt on High/Low-Voltage Detection

Special Microcontroller Features:

- C Compiler Optimized Architecture:
 - Optional extended instruction set designed to optimize re-entrant code
- 100,000 Erase/Write Cycle Enhanced Flash Program Memory Typical
- 1,000,000 Erase/Write Cycle Data EEPROM Memory Typical
- Flash/Data EEPROM Retention: 100 Years Typical
- Self-Programmable under Software Control
- Priority Levels for Interrupts
- 8 x 8 Single-Cycle Hardware Multiplier
- Extended Watchdog Timer (WDT):
 - Programmable period from 4 ms to 131s
- Single-Supply 5V In-Circuit Serial Programming™ (ICSP™) via Two Pins
- In-Circuit Debug (ICD) via Two Pins
- Wide Operating Voltage Range: 2.0V to 5.5V
- Programmable Brown-out Reset (BOR) with Software Enable Option

Device	Program Memory		Data Memory		I/O	10-Bit A/D (ch)	CCP/ ECCP (PWM)	MSSP		USART	Comp.	Timers 8/16-Bit
	Flash (bytes)	# Single-Word Instructions	SRAM (bytes)	EEPROM (bytes)				SPI	Master I ² C™			
PIC18F2525	48K	24576	3968	1024	25	10	2/0	Y	Y	1	2	1/3
PIC18F2620	64K	32768	3968	1024	25	10	2/0	Y	Y	1	2	1/3
PIC18F4525	48K	24576	3968	1024	36	13	1/1	Y	Y	1	2	1/3
PIC18F4620	64K	32768	3968	1024	36	13	1/1	Y	Y	1	2	1/3