



1859

UNIVERSIDAD NACIONAL DE LOJA

**ÁREA DE LA ENERGÍA, LAS INDUSTRIAS Y LOS
RECURSOS NATURALES NO RENOVABLES**

**CARRERA DE INGENIERÍA EN ELECTRÓNICA Y
TELECOMUNICACIONES**

**“DESARROLLO DE UN PROTOTIPO CONTROLADOR PARA
UN SISTEMA DOMÓTICO POR MEDIO DE UNA PANTALLA TFT
TOUCH SHIELD V2.0 Y ARDUINO”**

Tesis de Grado previo a optar por el
Título de Ingeniero en Electrónica y
Telecomunicaciones

AUTORA:

KAREN ELIZABETH LOAIZA JIMÉNEZ

DIRECTOR:

ING. LUIS ENRIQUE CHUQUIMARCA JIMÉNEZ M.I.

**LOJA – ECUADOR
2016**

CERTIFICACIÓN

Ingeniero.

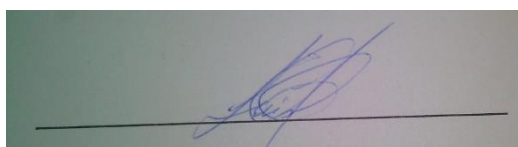
Luis Enrique Chuquimarca Jiménez M.I.

DIRECTOR DE TESIS

CERTIFICA:

Haber dirigido, asesorado, revisado y corregido el presente trabajo de tesis de grado, en su proceso de investigación, cuyo tema versa en **“DESARROLLO DE UN PROTOTIPO CONTROLADOR PARA UN SISTEMA DOMÓTICO POR MEDIO DE UNA PANTALLA TFT TOUCH SHIELD V2.0 Y ARDUINO”**, previo a la obtención del título de **Ingeniero en Electrónica y Telecomunicaciones**, realizado por la señorita egresada: **Karen Elizabeth Loaiza Jiménez**, la misma que cumple con la reglamentación y políticas de investigación, por lo que autorizo su presentación y posterior sustentación y defensa.

Loja, 24 de noviembre del 2015.

A rectangular box containing a handwritten signature in blue ink. The signature is stylized and appears to be 'Luis Enrique Chuquimarca Jiménez'. Below the signature is a horizontal line.

Ing. Luis Enrique Chuquimarca Jiménez, M.I.

DIRECTOR DE TESIS

AUTORÍA

Yo **KAREN ELIZABETH LOAIZA JIMÉNEZ**, declaro ser autora del presente trabajo de tesis y eximo expresamente a la Universidad Nacional de Loja y a sus representantes jurídicos de posibles reclamos o acciones legales por el contenido de la misma.

Adicionalmente acepto y autorizo a la Universidad Nacional de Loja, la publicación de mi trabajo de tesis en el Repositorio Institucional – Biblioteca Virtual.

Firma:

A handwritten signature in blue ink, appearing to be 'Karen Elizabeth Loaiza Jiménez', written over a light purple rectangular background.

Cédula: 1105086191

Fecha: 18/03/2016

CARTA DE AUTORIZACIÓN DE TESIS POR PARTE DE LA AUTORA, PARA LA CONSULTA, REPRODUCCIÓN PARCIAL O TOTAL Y PUBLICACIÓN ELECTRONICA DEL TEXTO COMPLETO.

Yo, **KAREN ELIZABETH LOAIZA JIMÉNEZ** declaro ser autora de la tesis titulada “**DESARROLLO DE UN PROTOTIPO CONTROLADOR PARA UN SISTEMA DOMÓTICO POR MEDIO DE UNA PANTALLA TFT TOUCH SHIELD V2.0 Y ARDUINO**”, como requisito para obtener el grado de: **INGENIERO EN ELECTRÓNICA Y TELECOMUNICACIONES**; autorizo al Sistema Bibliotecario de la Universidad Nacional de Loja para que con fines académicos, muestre al mundo la producción intelectual de la Universidad, a través de la visibilidad de su contenido de la siguiente manera en el Repositorio Digital Institucional:

Los usuarios pueden consultar el contenido de este trabajo en RDI, en las redes de información del país y del exterior, con las cuales tenga convenio la Universidad.

La Universidad Nacional de Loja, no se responsabiliza por el plagio o copia de la tesis que realice un tercero.

Para constancia de esta autorización, en la ciudad de Loja a los dieciocho días de marzo del dos mil dieciséis.

Firma:



Autor: Karen Elizabeth Loaiza Jiménez

Cédula: 1105086191

Dirección: Ciudadela Héroes del Cenepa. Calles Milton Patiño y Twintza.

Correo electrónico: karencit.lj@gmail.com

Teléfono: 072545855 **Celular:** 0996364586

DATOS COMPLEMENTARIOS

Director de Tesis: Ing. Luis Enrique Chuquimarca Jiménez, M.I

Tribunal de Grado: Ing. Paulo Alberto Samaniego Rojas, Mg. Sc.

Ing. Diego Fernando Carrera Moreno, Mg. Sc.

Ing. Benjamín Andrés Pusay Villaroel, Mg. Sc.

DEDICATORIA

Este proyecto de titulación está dedicado a Dios, a mis padres y hermanos, a mis abuelitos que estuvieron junto a mí, a mi familia, a mis amigos y a aquellas personas que forman parte esencial de mi vida.

Este éxito alcanzado está destinado a todos ellos por su apoyo, por el aliento y ayuda brindada día a día, por el respaldo y por el amor incondicional brindado a lo largo de mi vida y sobre todo durante mi formación académica.

AGRADECIMIENTO

Antes que nada quiero agradecer a Dios por su infinita bondad, a mis padres y a toda mi familia por el gran apoyo moral y económico que he recibido durante todo este tiempo, y a mis abuelitos que a pesar de ya no seguir junto a mí, me siguen demostrando que con perseverancia se pueden superar todos los problemas; consejos que me han servido para culminar con éxito mi carrera y llegar a ser la profesional que siempre deseé ser.

También quiero agradecer a la persona que estuvo junto a mí en las buenas y en las malas, e igualmente a mis amigos y grandes compañeros de estudio por la amistad brindada y por sobrellevar todas las adversidades que vivimos durante la estancia en la universidad y en la carrera. Es nostálgico y también emocionante culminar con éxito esta etapa de mi vida y a la vez pensar que cada uno de nosotros tomará rumbos diferentes, pero en este pequeño espacio quiero expresar mi sincera gratitud para todos ellos quienes compartieron conmigo las mayores alegrías y tristezas y poder manifestar que las verdaderas amistades se encuentran en esta maravillosa vida universitaria.

TABLA DE CONTENIDOS

CERTIFICACIÓN	II
AUTORÍA	III
CARTA DE AUTORIZACIÓN DE TESIS POR PARTE DE LA AUTORA	IV
DEDICATORIA	V
AGRADECIMIENTO.....	VI
ÍNDICE DE FIGURAS	X
ÍNDICE DE TABLAS	XII
1. TÍTULO.....	1
2. RESUMEN	2
2.1 ABSTRACT.....	3
3. INTRODUCCIÓN	4
4. REVISIÓN DE LITERATURA	5
4.1 CAPÍTULO I: DOMÓTICA.....	5
4.1.1 El hogar digital	5
4.1.2 La red domótica.....	6
4.1.2.1 Topología de la red domótica	6
4.1.2.2 Elementos constituyentes de la red domótica.....	9
4.1.2.3 Parámetros a controlar en la red domótica	10
4.2 CAPÍTULO II: COMUNICACIONES INALÁMBRICAS ORIENTADAS A LA DOMÓTICA.....	12
4.2.1 Comunicación por Radiofrecuencia	12
4.2.1.1 IEEE 802.11 – WLAN.....	13
4.2.1.2 <i>HomeRF</i>	14
4.2.1.3 IEEE 802.15 – WPAN.....	14
4.2.2 Transceptores inalámbricos	16
4.2.2.1 Transceiver inalámbrico Nordic NRF24L01	16
4.2.2.2 Modos de operación del transceptor nRF24L01	19
5. MATERIALES Y MÉTODOS.....	21
5.1 MATERIALES.....	21

5.2 MÉTODOS	22
5.2.1 PRIMERA PARTE: DESARROLLO DEL HARDWARE	22
5.2.1.1 COMUNICACIÓN INALÁMBRICA	25
5.2.1.2 INTERFAZ DE USUARIO.....	29
5.2.1.3 SEGUNDA PARTE: DESARROLLO DE SOFTWARE	36
5.2.2 FUNCIONALIDAD DEL PROTOTIPO	42
5.2.2.1 Pruebas y mediciones	42
6. RESULTADOS	44
6.1 SISTEMA IMPLEMENTADO	44
6.2 INTERFAZ DE USUARIO	46
6.3 VALIDACIÓN DEL PROTOTIPO	48
6.3.1 Validación Técnica.....	48
6.3.2 Validación Económica	50
6.3.3 Validación de características físicas y eléctricas	51
7. DISCUSIÓN	53
7.1 VENTAJAS DEL PROTOTIPO	54
8. CONCLUSIONES	56
9. RECOMENDACIONES	57
10. BIBLIOGRAFÍA	59
11. ANEXOS	62
ANEXO 1: Algoritmos de programación	62
ANEXO 1.1: Algoritmo para la pantalla TFT Touch	62
ANEXO 1.2: Algoritmo para los módulos nRF24L01	67
ANEXO 1.3: Algoritmo para el Arduino Mega 2560 Central	72
ANEXO 2: Especificaciones técnicas de los dispositivos	76
ANEXO 2.1: Pantalla TFT LCD 3.2' HY-TFT320 V2.0	76
ANEXO 2.2: Módulo nRF24L01	79
ANEXO 3: Fotos del prototipo controlador del sistema domótico	83
ANEXO 4: Dispositivos finales	84

ANEXO 4.1: Especificaciones.....	84
ANEXO 5: Circuitos impresos.....	85
ANEXO 5.1: PCB para el circuito de conmutación de iluminación y ventilación	85
ANEXO 6: Proforma de sistema comercial.....	86

ÍNDICE DE FIGURAS

Figura 1. Topología en anillo [4].....	7
Figura 2. Topología en bus [4]	7
Figura 3. Topología en estrella [4].	8
Figura 4. Diagrama de bloques del nRF24L01 [17].....	17
Figura 5. Diagrama de la red domótica [El Autor].....	22
Figura 6. Diagrama de conexión para la conmutación del ventilador [El Autor]	23
Figura 7. Conexión del motor para apertura y cierre de persianas [El Autor]	24
Figura 8. Circuito de conmutación para la iluminación [El Autor].....	25
Figura 9. Módulo nRF24L01 [17].....	26
Figura 10. Pinout del módulo nRF24L01 [17].	27
Figura 11. Conexión del Arduino Mega con el módulo nRF24L01 [El Autor]	28
Figura 12. Arduino Mega 2560 R3 [El Autor].....	30
Figura 13. Pantalla TFT Touch LCD [El Autor].....	34
Figura 14. Shield TFT LCD [El Autor].....	35
Figura 15. Conexión del prototipo controlador [El Autor].....	35
Figura 16. IDE o entorno de desarrollo integrado [El Autor]	36
Figura 17. Selección de tarjeta Arduino Mega [El Autor]	37
Figura 18. Selección de puerto serial [El Autor]	37
Figura 19. Diagrama de flujo para la pantalla táctil [El Autor]	39
Figura 20. Diagrama de flujo del módulo nRF24L01 [El Autor].	40
Figura 21. Diagrama de flujo para Arduino Mega central.	41
Figura 22. Circuito conmutador para ventilador [El Autor].....	45
Figura 23. Unidad controladora del sistema domótico [El Autor]	45
Figura 24. Opciones presentadas en la pantalla TFT [El Autor].....	46
Figura 25. Opción “Persiana” abierta [El Autor]	47
Figura 26. Información del estado actual del sistema [El Autor].....	48
Figura 27. Datos dimensionales del prototipo [El Autor]	52
Figura 28. Montaje, instalación de nodo periférico [El Autor]	52
Figura 29. Diagrama de pines de la pantalla TFT [26].	77

Figura 30. Datos técnicos de la pantalla TFT	78
Figura 31. Diagrama de pines del módulo nRF24L01 [27]	80
Figura 32. Vista frontal del prototipo controlador del sistema domótico [El Autor]	83
Figura 33. Vista interior del prototipo controlador del sistema domótico [El Autor]	83
Figura 34. Ventilador utilizado en el desarrollo del proyecto [El Autor]	84
Figura 35. Calefactor utilizado en el desarrollo del proyecto [El Autor]	84
Figura 36. Circuito de conmutación para los focos [El Autor]	85
Figura 37. Circuito de conmutación para ventilador [El Autor]	85
Figura 38. Proforma Kit Smart Home (sistema domótico comercial) [El Autor]	86

ÍNDICE DE TABLAS

Tabla 1. Formato del paquete Enhanced ShockBurst [El Autor].	18
Tabla 2. Modos de operación del nRF24L01 [8].....	20
Tabla 3. Materiales utilizados para el desarrollo del proyecto [El Autor].....	21
Tabla 4. Equipos utilizados para el desarrollo del proyecto [El Autor].	21
Tabla 5. Conexión de pines para el nRF24L01 con el Arduino Mega [El Autor].....	42
Tabla 6. Conexión de pines para la pantalla TFT con el Arduino Mega [El Autor].	43
Tabla 7. Mediciones obtenidas durante la validación del prototipo [El Autor].	49
Tabla 8. Datos obtenidos según los comandos enviados [El Autor]	49
Tabla 9. Presupuesto del sistema domótico y control remoto [El Autor].....	50
Tabla 10. Conexión de los pines de la pantalla TFT [El Autor].....	77
Tabla 11. Condiciones de operación del módulo nRF24L01 [27].....	80
Tabla 12. Especificaciones del nRF24L01 [27]	81
Tabla 13. Funciones de los pines del nRF24L01 [27].....	81
Tabla 14. Condiciones en radiofrecuencia del módulo nRF24L01 [27]	82
Tabla 15. Operación del nRF24L01 transmisor	82
Tabla 16. Parámetros del nRF24L01 en recepción.....	82

1. TÍTULO

“DESARROLLO DE UN PROTOTIPO CONTROLADOR PARA UN SISTEMA DOMÓTICO POR MEDIO DE UNA PANTALLA TFT TOUCH SHIELD V2.0 Y ARDUINO”

2. RESUMEN

El presente proyecto de titulación tiene por objetivo desarrollar un prototipo controlador para un sistema domótico implementado con Arduino, mediante la utilización de una pantalla TFT Touch Shield V2.0 y Arduino Mega 2560 R3; el mismo que está orientado al control de variables como la iluminación, ventilación y acondicionamiento de una habitación por medio de un enlace inalámbrico.

La particularidad de este proyecto es la utilización de los módulos transceptores inalámbricos nRF24L01, ejecutados por microcontroladores Atmega 8, que funcionan en la banda libre ISM a 2.4 GHz; dos placas Arduino Mega 2560 R3 empleadas para el procesamiento digital de las señales obtenidas y una pantalla TFT LCD Touch para la interfaz de usuario, utilizada para la visualización y selección de las diferentes acciones a ejecutar dentro de la red domótica.

El trabajo está dividido en dos etapas, es así que en la primera parte se presenta aspectos del hardware, en donde se detallan las características de los elementos del prototipo, tales como el módulo nRF24L01, Arduino Mega, y la pantalla resistiva TFT Touch. A su vez en el desarrollo del software, se incluye la programación de las placas Arduino junto con los módulos transceptores que son quienes permiten la comunicación inalámbrica por radiofrecuencia y la programación de la pantalla TFT Touch con la respectiva interfaz de usuario. Para la segunda etapa del proyecto se describe la validación del prototipo, la misma que está dividida en aspectos técnicos, económicos, autonomía y características físicas del mismo. La validación técnica detalla las mediciones de latencia realizadas a unas ciertas distancias con la finalidad de obtener información de la operabilidad y alcance de los módulos nRF24L01 y finalmente, la validación económica presenta el costo total del sistema implementado.

2.1 ABSTRACT

The present degree project aims to develop a domotic system prototype driver through a TFT Touch Screen Shield V2.0 and Arduino Mega 2560 R3; that is oriented to control variables such as lighting, ventilation and air conditioning of a room via a wireless link.

The particularity of this project is the use of wireless transceiver modules nRF24L01 executed by microcontrollers Atmega 8 operating in the free ISM band at 2.4 GHz; two plates Arduino Mega 2560 R3 are used for digital processing of the signals obtained TFT LCD Touch and screen for user interface, used for viewing and selection of the various actions to be executed within the automation network.

The work is divided into two stages, in the first part it presents aspects of hardware where the characteristics of the prototype elements, such as the nRF24L01 module, Arduino Mega, and the resistive TFT Touch are detailed presented. In turn of the software development, it includes the programming Arduino boards with the transceiver modules those that enable the wireless radio frequency communication and programming TFT Touch screen with the respective user interface. Likewise, for the second phase of the project validation prototype is described, the same is divided into technical, economic, autonomy and physical characteristics of the same aspects. Detailed technical validation latency measurements made at a certain distance in order to obtain information from the operability and scope of the nRF24L01 modules and finally the economic validation shows the total cost of the implemented system.

3. INTRODUCCIÓN

En la actualidad la tendencia a automatizar los hogares cada vez es mayor, la demanda por utilizar sistemas domotizados está en crecimiento, en donde el grado de automatización presente en dichas viviendas está en función del presupuesto disponible para ello y del grado de información que tengan los usuarios potenciales.

Poco a poco las viviendas han empezado a incorporar elementos automatizados, que si bien es cierto no se está hablando de un sistema domótico, ya empiezan a adentrarse en el mundo de las casas inteligentes. Además, para considerar un hogar completamente digital, la vivienda tendrá que incorporar en su totalidad elementos controlados tales como actuadores y sensores, por ejemplo termostatos, automatización de persianas, módulos por radiofrecuencia, etc. permitiendo al usuario gestionarla remotamente, haciendo posible un aumento del confort con la mínima intervención del usuario y mejorando su calidad de vida.

El presente proyecto de tesis se enfoca en la construcción de un prototipo capaz de controlar un sistema domótico, utilizando elementos de bajo costo tales como Arduino, pantallas resistivas y módulos RF transceptores; buscando conseguir un sistema de control domótico accesible, seguro y que sea intuitivo y amigable para el usuario. Así mismo, se intenta desarrollar una solución domótica cuya particularidad es la de ofrecer un dispositivo elaborado con herramientas libres (Open Source), haciendo posible su implementación e instalación a bajo costo sin necesidad de recurrir a soluciones domóticas comerciales que aún son inaccesibles en nuestro medio.

Si bien es cierto, empresas dedicadas a la automatización de hogares como por ejemplo las empresas españolas Domomarket, Casadomo y Dilartec, la americana Wintel y la alemana Gewiss, han empezado a desarrollar productos completos que satisfagan los requerimientos actuales que demanda la domótica, tales como seguridad con simulación de presencia, sistema de alarmas, control de elementos, dispositivos, iluminación y persianas, etc.; haciendo cada vez posible el hecho de tener una casa completamente inteligente. Pero como se sabe, dichas soluciones domóticas aún cuentan con costos elevados, lo que hace imposible adquirir los módulos comerciales residenciales que se ofrecen.

4. REVISIÓN DE LITERATURA

4.1 CAPÍTULO I: DOMÓTICA

El término domótica proviene del latín “DOMUS” que significa casa y “TICA” que significa automático, es decir que funciona por sí sola. Por tanto se entiende por domótica al conjunto de tecnologías aplicadas al control y la automatización inteligente de la vivienda, aportando servicios de gestión energética, seguridad y bienestar, los mismos que pueden estar integrados mediante el uso de redes interiores y exteriores de comunicación, cableadas o inalámbricas; cuyo control sea desde dentro o fuera del hogar [1].

Tomando en cuenta el concepto de Smart Home, la vivienda domótica integra una serie de automatismos referente al campo eléctrico, electrónico y de telecomunicaciones, con la finalidad de garantizar al usuario un aumento del confort, bienestar y ahorro energético dentro del hogar [1].

4.1.1 El hogar digital

La domótica actualmente se enfoca específicamente en ámbitos de telecomunicaciones, denominado hogar digital. El gran progreso de los sistemas de telecomunicación y el desarrollo de Internet, han aumentado la capacidad para crear información, almacenarla, transmitirla y procesarla. [2].

Algunas ventajas de vivir en un hogar digital son [2], [3]:

- **Climatización y consumo energético:** Esto se refiere a que los usuarios pueden realizar la programación del encendido y apagado de aparatos como ventiladores, aire acondicionado, iluminarias o luces de las habitaciones, según las condiciones ambientales que se presenten.

El ahorro energético es lo que actualmente preocupa a muchos de los usuarios, por tanto, este aspecto se refiere a que no es necesario sustituir aparatos o sistemas del hogar por otros que consuman menos, sino se refiere a una gestión eficiente de los mismos, ayudándose obviamente de la domótica.

- **Bienestar y confort:** Esta parte manifiesta que mediante la domótica se puede tener control de los dispositivos y equipos instalados en el hogar, desde un dispositivo inalámbrico, móvil, PC o por Internet.
- **Seguridad:** Hay una variedad de posibilidades que pueden ser ejecutadas en un sistema domótico, entre ellas tenemos: control de acceso a la vivienda, instalación de cámaras que permitan realizar grabaciones y monitorizar en tiempo real, manejo de alarmas de seguridad que permitan avisar al usuario en caso de que exista alguna intrusión o avería.
- **Comunicaciones, telegestión y accesibilidad:** Con esto nos estamos refiriendo a la infraestructura de comunicación que posee el hogar y el diseño que debe ser accesible para diversidad de usuarios. Por ejemplo, con la teleasistencia es posible ayudar a las personas ancianas o con discapacidad, realizando instalaciones propias para este tipo de circunstancias.

4.1.2 La red domótica

Como ya se manifestó anteriormente, la domótica es la implementación e integración de varias redes, elementos y dispositivos electrónicos en el hogar, los cuales permiten la automatización de las actividades cotidianas y el control local o remoto de la vivienda. Para que todos estos diferentes sistemas puedan trabajar en conjunto, se necesita que estén conectados a través de una red interna que generalmente se la conoce como red del hogar o doméstica (HAN, por sus siglas en inglés) [2]. Por lo tanto la red domótica estará caracterizada por la topología del sistema de control y los medios de transmisión a utilizarse [4].

4.1.2.1 Topología de la red domótica

El término topología está referido a la forma de interconexión de los equipos y los sistemas conectados a la red. Para la red domótica, la topología depende del dispositivo de control que se emplee y el medio de transmisión que se utilice [4].

Existen los siguientes tipos de topología usadas más comúnmente en las redes domóticas, las cuales se detallan a continuación [2], [4]:

- **Topología en anillo:** En esta topología cada equipo de control (EC) está conectado a otro EC y así sucesivamente, dando la apariencia de un anillo, (véase Figura 1). Aquí la comunicación entre equipos de control se suele utilizar de una manera unidireccional, lo que se facilita la comunicación, aunque también es usada de manera bidireccional. Posee la desventaja de que si existe un fallo en un dispositivo de control, falla todo el sistema.

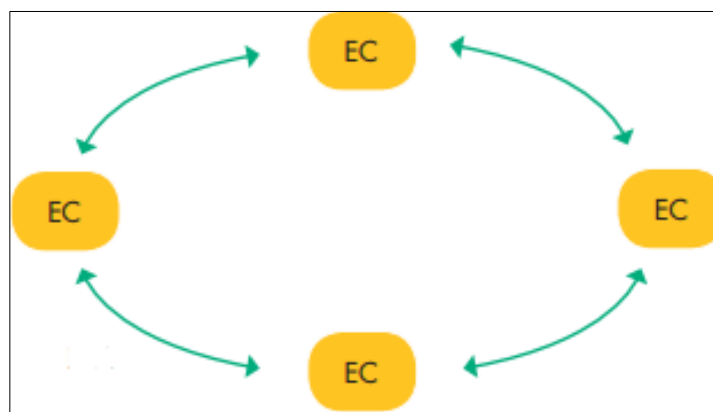


Figura 1. Topología en anillo [4]

- **Topología en bus:** Esta configuración requiere que todos los EC de la red se encuentren conectados a un bus que está compuesto de un par de conductores, (véase Figura 2).

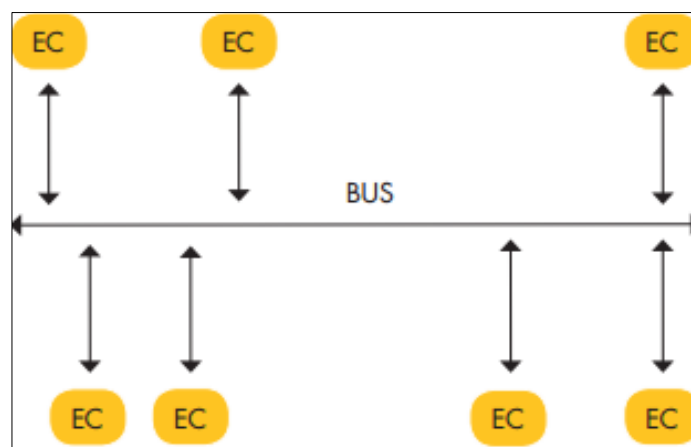


Figura 2. Topología en bus [4]

- Topología en estrella o centralizada:** Este tipo de topología es usada en los sistemas que requieran un control centralizado, ya que aquí existe un solo EC por el que va a pasar toda la información, tal como se observa en la Figura 3. Entonces todos los componentes de la red deben conectarse a este dispositivo de control central, el cual será quien active las salidas (S) correspondientes al recibir una orden específica de entrada (E). Estas acciones las realizará según la programación desarrollada para el sistema domótico.

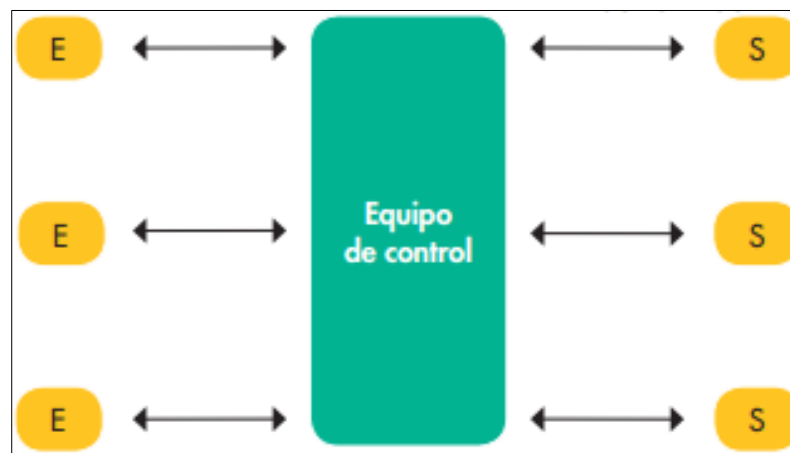


Figura 3. Topología en estrella [4].

- Topología punto a multipunto:** En una topología punto a multipunto existe un punto central que se comunica con varios otros puntos remotos. Generalmente esto implica que la comunicación es solamente entre el nodo central y los nodos remotos, y de éstos hacia el central; no existe comunicación entre los nodos remotos o periféricos. Generalmente esta topología suele ser confundida con la topología en estrella, la cual tiene el mismo concepto de funcionamiento diferenciándose en que una red punto multipunto no requiere que las comunicaciones siempre pasen por el nodo central.

Luego de elegir el tipo de topología de la red domótica que se desea implementar, es necesario conocer qué equipos, dispositivos y elementos actuadores formarán parte de la misma, ya que estos serán los principales agentes utilizados para lograr controlar las

diferentes variables o parámetros programados en la implementación de una red domótica, lo cual se verá en la siguiente sección.

4.1.2.2 Elementos constituyentes de la red domótica

La domotización de una vivienda supone la incorporación de elementos puntuales que permiten que la red domótica sea instalada y funcional, entre ellos tenemos [2]:

- **Sensor:** Es un dispositivo que recoge la información de los diferentes parámetros que controla, como puede ser la temperatura ambiente, la presencia de luz, humedad, etc. y la envía al sistema de control centralizado. Se presenta los siguientes tipos [1], [5]:
 - **Sensor de temperatura interior o exterior:** cuya función es exclusivamente la de medir la temperatura de la estancia en la que se encuentre situada.
 - **Sondas de humedad:** cuya misión es descubrir fugas de agua.
 - **Detectores de presencia:** que se acciona al detectar la presencia de intrusismo de personas ajenas a nuestra vivienda.

- **Procesador/controlador:** Este será el dispositivo capaz de recibir, procesar y tratar la información que llega de los distintos elementos de entrada como sensores, o en este caso la pantalla táctil, según el programa o algoritmo desarrollado. Una vez recibida esta información, se procesa y se comunica a los distintos elementos actuadores de la red domótica.

- **Actuadores/accionadores:** Son aquellos que reciben la señal del dispositivo controlador y actúan ejecutando la acción preestablecida, entre los más comúnmente utilizados están los relés, motores de persiana, e interruptores.

- **Elementos finales:** Son aquellos elementos que actúan luego de intervenir los accionadores o actuadores. Estos realizan la última función de salida, por ejemplo en un sistema de seguridad el elemento final es la cámara de video vigilancia.

- **Interface:** Estos son aquellos dispositivos como pantallas, teléfonos móviles, Internet en que se muestra la información del sistema para los usuarios y en donde los mismos pueden interactuar con el sistema.

4.1.2.3 Parámetros a controlar en la red domótica

En esta sección se especifican de manera sencilla como trabajan los dispositivos anteriormente mencionados, según los parámetros que se deseen controlar en el sistema domótico.

- **Control de iluminación:** En este parámetro se puede resaltar una ventaja ya que, con solo usar un sensor de luminosidad e interruptores, es posible controlar automáticamente las luces de las salas que formen parte de un área o zona específica, con el consiguiente ahorro de energía y de luminarias [6].

Además, cada cuarto que no está siempre habitado debe tener un actuador e interruptor, mientras que aquellas instancias que son comúnmente ocupadas, deberán ir gobernadas por actuadores, interruptores y posiblemente reguladores, los mismos que permitirán al usuario regular la intensidad de la luz a su gusto y adecuar la luz según sus necesidades. También a manera de seguridad encender o apagar las luces remotamente [6].

- **Control de persianas y toldos:** Automatizando el control de persianas el usuario ya no tiene la necesidad de tener que subir o bajarlas diariamente, pues con este parámetro se está ayudando al confort de los usuarios y al ahorro energético, ya que se aprovecha la energía solar y no sólo se limitarían al uso de calefacción. Así mismo, se tiene una mayor seguridad anti robos ya que se tendría una simulación de presencia en la vivienda [6], [7].

Para hacer posible la ejecución de este parámetro, es necesario utilizar elementos actuadores como relés, motores, sensores que se activen dependiendo de la intensidad de luz solar o época climática [6], [7].

- **Control de temperatura o climatización:** Mediante este parámetro es posible controlar la temperatura del hogar, ya que es posible regularla y ajustarla según la necesidad del usuario. Ya no es necesario dejar encendido el calefactor por horas y así llegar al hogar y tener un ambiente cálido, puesto que con el uso de sensores y electroválvulas es posible encender o apagar el dispositivo calefactor según la necesidad del usuario, y lo que es mejor el control es realizado remotamente, garantizando el confort, bienestar y ahorro de energía [6].

- **Seguridad:** Este aspecto es el más importante dentro de la instalación de un sistema domótico, es posible que si no existe este parámetro implementado, el sistema domótico pierda validez. Existen varios tipos de seguridad que deben ser implementados en un sistema domótico, los cuales se pueden citar a continuación [7]:
 - **Control de fugas de gas:** En este aspecto los sistemas al recibir la señal de cualquier detector de gas, inmediatamente se accionan y ejecutan las medidas oportunas para contrarrestar dicha fuga.

 - **Control de intrusos:** En este caso, se ejecutarán las medidas de prevención por invasión de propiedad. Así, en caso de accionarse los sensores de presencia inmediatamente se activan las alarmas y cámaras respectivas del sistema domótico. Si se quiere tener aún mejores prestaciones, es posible implementar sistemas de envío de mensajes que informen o detallen todo el acontecimiento de dicha intromisión.

4.2 CAPÍTULO II: COMUNICACIONES INALÁMBRICAS ORIENTADAS A LA DOMÓTICA

Una residencia automatizada debe poseer un sistema de comunicación que permita la conexión hacia el exterior, y a su vez, debe poseer otro sistema de comunicación dentro de ella. Para esto es necesario saber que existen dos tipos de comunicación, por medios guiados y no guiados, las cuales presentan varias ventajas que deben ser consideradas para poder utilizarse en una red domótica [8].

Las redes alámbricas o cableadas son aquellas que requieren un medio físico para poder realizar la comunicación, estos medios cableados pueden ser cables utp de par trenzado, coaxial, fibra óptica e incluso la red eléctrica. La ventaja que presenta este tipo de comunicación es que posee mayor velocidad de transmisión frente a una red inalámbrica, pero es más costosa por el tendido de cable a utilizar, e igualmente, en la actualidad las redes inalámbricas son las que han empezado a tener mayor auge [9].

Las redes inalámbricas no requieren de cables físicos para lograr la comunicación, son más versátiles puesto que permiten la movilidad a los usuarios, se pueden acoplar perfectamente al sistema tradicional cableado y así garantizar la continuidad a los servicios informáticos. A pesar de tener la desventaja de que el área de cobertura no está garantizada al 100% debido a que se encuentran limitadas por las características de la residencia como son el material de construcción usado, obstáculos metálicos, electrodomésticos, etc.; son las más utilizadas ya que como sabemos el uso de dispositivos inalámbricos son más utilizados actualmente [8], [9].

Las tecnologías inalámbricas están divididas en dos tipos, la infrarroja (IR) y la radiofrecuencia (RF), de las cuales las más utilizadas en el campo de la domótica son las comunicaciones por radiofrecuencia, misma que se detalla en la siguiente sección [8]–[10].

4.2.1 Comunicación por Radiofrecuencia

Este tipo de comunicación emplea el espacio como medio de transmisión, ya que la información está en forma de ondas electromagnéticas no guiadas. Por tanto, el transmisor como el receptor requiere de un transductor o antena para poder recibir la información que esté destinada para ellos [8], [10].

La radiofrecuencia permite comunicaciones de corto y medio alcance, además no está limitada por las características de la residencia y puede atravesar obstáculos y paredes sin dificultad. A su vez, hay una serie de parámetros que influyen en estos sistemas, así tenemos que el alcance depende de la frecuencia en que se está trabajando, de la potencia de salida, de la sensibilidad de recepción y de la ganancia de la antena. Igualmente el entorno o medio en que se está transmitiendo influye en la calidad de transmisión, es así que se tiene una gran diferencia al trabajar en campo abierto que en un edificio o vivienda [8], [10].

En estos momentos se puede distinguir tecnologías inalámbricas con protocolo estándar tales como IEEE 802.11 (WLAN - *WiFi*), *HomeRF*, IEEE 802.15 (WPAN), y dispositivos tales como transceptores inalámbricos [9], [10]. En la siguiente sección se describe lo que implica cada una de estas tecnologías.

4.2.1.1 IEEE 802.11 – WLAN

El estándar 802.11 es un estándar IEEE para las redes de área local inalámbricas (WLAN, por sus siglas en inglés), el mismo que define como se utiliza la radiofrecuencia (RF) en las bandas sin licencia o banda ISM, admite una potencia de transmisión máxima de 100mW y trabaja en las dos capas inferiores del modelo OSI que son la capa física y la capa de enlace (LLC y MAC) [10], [11].

La capa física define sobre la transmisión de datos, es decir se encarga de la modulación de ondas de radio y de la señalización para la transmisión de los mismos; mientras que la capa de enlace de datos define la interfaz entre el bus del equipo y la capa física y las reglas para la comunicación, de forma similar a una red cableada o Ethernet [12].

Cuando el estándar 802.11 se emitió por primera vez, prescribía tasas de datos de 1 a 2 Mbps en la banda de 2,4 GHz. En ese momento, las redes conectadas por cable operaban a 10 Mbps, de modo que la nueva tecnología inalámbrica no se adoptó con entusiasmo. A partir de entonces, los estándares de LAN inalámbricas mejoraron continuamente con la edición de IEEE 802.11a, IEEE 802.11b, IEEE 802.11g, y el actual 802.11n [12].

4.2.1.2 HomeRF

Es una tecnología de comunicación por radiofrecuencia basada en el protocolo de acceso compartido inalámbrico (SWAP, por sus siglas en inglés), el cual define una interfaz inalámbrica para lograr la integración de todos los elementos de una vivienda y construir una red doméstica. De manera similar a IEEE 802.11 opera en la banda de los 2.4Ghz [8], [10].

HomeRF trabaja a nivel de capa MAC para la transmisión y manejo de datos combinándola con la tecnología DECT que proporciona el soporte para las funciones de voz. Igualmente la capa física PHY, es la encargada de transmitir los datos por el aire utilizando el protocolo FH o Saltos de Frecuencia [8], [9], [11].

La velocidad máxima de transmisión en HomeRF es de 10Mbps, lo que hace que sea idóneo para aplicaciones caseras, consume menos potencia que IEEE 802.11 y tiene la particularidad de ser un estándar interoperable [10].

4.2.1.3 IEEE 802.15 – WPAN

Las redes inalámbricas de área personal (WPAN, por sus siglas en inglés), son aquellas que comúnmente cubren distancias del orden de los 10 metros y son más utilizadas para la comunicación de dispositivos portátiles personales. El tipo de ámbito y los bajos índices de datos de transmisión que poseen las hacen idóneas para su uso en redes domésticas, en donde se usan especialmente en dispositivos móviles pequeños que funcionen con baterías [13].

La particularidad del estándar 802.15 es que define las características de las capas físicas y de control de acceso al medio de las redes inalámbricas de área personal WPAN con tasas bajas de transmisión de datos, permitiendo utilizar dispositivos de fácil instalación para transmisiones a corta distancia y a precios económicos [14]. Además, el estándar 802.15 posee dos modos de operación, es decir se pueden usar en modo isócrono como también en modo asíncrono, dependiendo de los módulos inalámbricos comerciales que se adquiera. En el primer modo de operación isócrono, se necesita de un mínimo ancho de banda garantizado, por lo que requiere necesariamente del punto de acceso para poder establecer la comunicación con otro nodo. En cambio en el modo asíncronos no se requiere de un punto de acceso para establecer una comunicación con otros dispositivos, simplemente se enlazan

entre ellos, permitiendo a su vez conectar dispositivos tanto en topología estrella como topología punto a punto [8], [11].

El estándar 802.15 está regido bajo las normas del grupo de trabajo IEEE 802.15, el mismo que ha considerado pertinente definir tres tipos de redes inalámbricas de corto alcance, clasificadas según el rango de datos, consumo de energía y calidad de servicio (QoS, por sus siglas en inglés) [12], [13].

- **802.15.3:** Este estándar es utilizado para la transmisión de datos que requieren altos niveles de QoS, rápidos y eficientes. La característica principal de este estándar es su uso para la transmisión de video, pero su complejidad es una limitante por lo que es poco utilizado, a pesar de contar con la encriptación compartida de información basada en el estándar de encriptación avanzada (AES, por sus siglas en inglés).
- **802.15.1:** Este estándar está referido a las WPANs de alcance medio cuya utilización está destinada en la comunicación de teléfonos celulares, PDAs, Tablets, etc. y que tienen un QoS apropiado para aplicaciones de voz y envío de imágenes. Una tecnología de este estándar que ha tenido gran acogida es el bluetooth, el cual es un enlace radio de corto alcance, que posee un área de trabajo de hasta 10 metros. Opera en el rango de frecuencias de 2,402 GHz a 2,480 GHz comúnmente usadas en la banda de operación libre ISM (Industrial, Científica y Médica) por lo que no requiere licencia para su operación. Están previstas dos potencias de emisión en función de la distancia que se desea cubrir, 10 metros con 1 miliwatio y 100 metros con 100 miliwatios alcanzando velocidades de 432kbps simétricamente y 721kbps asimétricamente [9], [10].
- **802.15.4:** El estándar 802.15.4 conocido como LR – WPAN (Low rate – wireless personal area network) es un estándar utilizado para comunicaciones a cortas distancias que requieren bajas tasas de transmisión de datos, cuyos dispositivos son de fácil instalación, económicos y de bajo consumo de energía.
Las características que presenta este estándar son la de poseer velocidades de datos alcanzadas de 851 kbps, 250 kbps, 100 kbps, 40 y 20 kbps; tiene direccionamiento de

16-bit para direcciones cortas y de 64 para direcciones extendidas. Su modo de operación es tanto en estrella o como una red punto a punto y tienen un bajo consumo de energía [9].

4.2.2 Transceptores inalámbricos

Son dispositivos que tienen tanto un transmisor y un receptor que comparten la circuitería dentro de un mismo chipset. Funcionan como dispositivos semidúplex, es decir, pueden transmitir señales en ambos sentidos pero no simultáneamente [15].

Un claro ejemplo de este tipo de dispositivos es el módulo inalámbrico NRF24L01 de la empresa Nordic S.A, utilizado en el desarrollo del presente proyecto, el cual se detalla a continuación.

4.2.2.1 Transceiver inalámbrico Nordic NRF24L01

El módulo nRF24L01 es un dispositivo transceptor diseñado para la implementación de redes de corto alcance WPAN con aplicaciones inalámbricas de bajo consumo de potencia, trabaja en la banda ISM a frecuencias desde los 2,4 GHz a 2,4835 GHz; y a su vez opera con un protocolo motor embebido denominado *Enhanced ShockBurst* [8], [16].

Este módulo tiene la particularidad de tener integrado tanto la antena de transmisión y de recepción conjuntamente con los demás componentes pasivos, en una placa de poco más de 33.108 mm, lo que hace que sea un módulo de fácil adaptación y acople. Además opera a través de la comunicación serial periférica (SPI, por sus siglas en inglés) para poder manipular todos los registros de configuración del mismo [8], [15].

El rango de cobertura depende de aspectos tales como obstáculos o paredes, determinando que es posible tener un mayor alcance cuando la línea de vista no está muy obstruida. La distancia que Nordic como proveedor especifica es de 50 metros y puede transmitir desde 250kbps hasta 2Mbps [8], [15]. Además utiliza la modulación en frecuencia GFSK (Gaussian Frequency Shift Keying), este tipo de modulación codifica los datos como una serie de cambios de una forma de onda de modulación de frecuencia, donde un 1 lógico es representado mediante un incremento de la frecuencia de la portadora, y un 0 mediante un decremento de la misma. Adicional a esto, el Nordic nRF24L01 debe ser alimentado con una

tensión entre 2.7 a 3.3 voltios, por lo que en su mayoría se debe utilizar una fuente de energía externa como por ejemplo una batería AA [15], [16].

Como ya se explicó al inicio, los Nordic nRF24L01 pueden enviar como recibir datos en paquetes de varios bytes a la vez, tienen incluida una función de corrección de errores CRC automático y pueden conectarse hasta con 6 unidades similares al mismo tiempo. Permite utilizar direcciones de capa 2 de hasta cinco bytes y mensajes o payloads de hasta 32 bytes. Adicional a esto, opera con un protocolo motor embebido denominado Enhanced ShockBurst, del cual se detalla a continuación [8], [16].

- Protocolo *Enhanced ShockBurst*:** El bloque *Enhanced ShockBurst Baseband Engine* (véase Figura 4) es el núcleo del transceptor, el mismo que cumple la función automática de temporización, reconocimiento y retransmisión de paquetes dados entre dos dispositivos transceptores. El Nordic nRF24L01 tiene encapsulado el hardware que implementa este protocolo, siendo por tanto, el principal responsable de las funciones de enlace del sistema [15], [16].

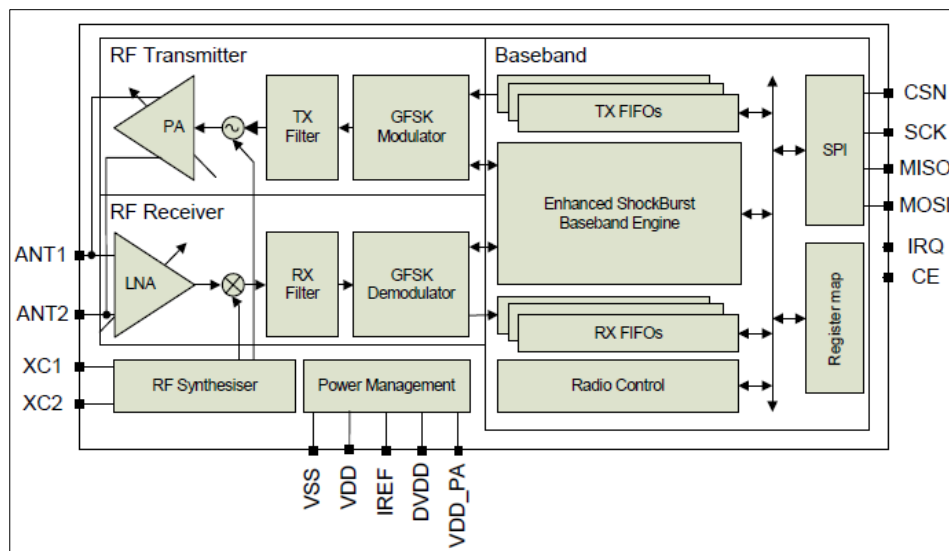


Figura 4. Diagrama de bloques del nRF24L01 [17].

Durante la transmisión, *ShockBurst* ensambla los paquetes y realiza la retransmisión de los bits del paquete a ser transmitido, mientras que el receptor busca una dirección válida. Cuando encuentra una señal válida, éste procesa el resto del paquete y lo

valida mediante una comprobación cíclica de errores (CRC, por sus siglas en inglés), si el paquete es correcto, entonces es colocado en el buffer de recepción [8], [16].

El manejo automático del Enhanced ShockBurst es de la siguiente forma [8], [16]:

- El usuario transmite un paquete desde el nRF24L01 transmisor (PTX) hacia el receptor (PRX), y el protocolo Enhanced ShockBurst automáticamente configura a PTX en modo recepción para que espere por el acuse de recibido ACK.
- Si el paquete es recibido por el PRX, entonces el protocolo automáticamente ensambla y transmite el ACK hacia el PTX, diciéndole que fue configurado en modo de recepción.
- En caso de que PTX no recibe el ACK dentro del tiempo establecido, el protocolo retransmitirá de forma automática el paquete de datos original y configurara a PTX en modo de recepción para que espere nuevamente el acuse de recibido desde el PRX.
- **Formato de paquete *Enhanced ShockBurst*:** En esta sección se presenta una descripción general de lo que representa el formato de un paquete *Enhanced ShockBurst* (véase Tabla 1), el mismo que contiene los siguientes campos: preámbulo, dirección, *payload* o carga útil y el campo CRC [8].

Tabla 1. Formato del paquete Enhanced ShockBurst [El Autor].

Preámbulo	Address	Packet Control	Payload	CRC
(1 byte)	(3 – 5 byte)	Field (9 bit)	(0 – 32 byte)	(1 – 2 byte)

- **Campo Preámbulo (1 byte):** Es una secuencia de 1 byte usada para detectar los niveles 0 y 1 en el lado del receptor. Si el primer bit es 1, el preámbulo es automáticamente establecido a 10101010 y si el primer bit es 0, el preámbulo será 01010101. Esto se realiza con el fin de asegurar que las transiciones son suficientes para estabilizar el receptor.

- **Campo Dirección:** Este campo es utilizado por el receptor para asegurarse que los paquetes que está detectando sean los correctos. La longitud puede ser configurada a 3, 4 o 5 bytes, empleando el registro AW.
- **Campo para el control del paquete:** En este campo se encuentra el PID o Identificación de paquete, mismo que es utilizado para detectar si el paquete recibido es nuevo o retransmitido. Así mismo, en este campo se encuentra la bandera NO_ACK que significa Sin acuse de recibo, la cual es utilizada cuando se emplea el modo automático de envío de acuses de recibo. Cuando esta bandera se configura en alto, el receptor asume que el paquete no se auto reconoce.
- **Carga útil o Payload:** Este es el contenido en sí del paquete, el cual puede tener un tamaño de 0 hasta 32 bytes de longitud y es retransmitida de forma inalámbrica.
- **CRC o Comprobación de redundancia cíclica:** Este es el mecanismo de comprobación o detección de errores en el paquete, es decir, si el CRC en un paquete falla, el Enhanced ShockBurst no lo aceptará. Este campo puede ser de 1 o 2 bytes de longitud y es calculado en función de la dirección, el control del paquete y de la carga útil.

4.2.2.2 Modos de operación del transceptor nRF24L01

El módulo nRF24L01 tiene cuatro modos de operación, que definen la función a realizar, de estas tenemos [8], [15], [16]:

- **Modo de bajo consumo de energía (POWER DOWN MODE):** en este modo, el transreceptor es deshabilitado para conseguir un consumo de corriente mínimo.
- **Modo de espera (STANBY MODES):** Existen dos modos, el I y el II, en el primer modo, se minimiza el consumo promedio de corriente manteniendo al mismo tiempo cortos periodos de tiempo de puesta en marcha; mientras que para el segundo modo,

el dispositivo PTX tiene en alto el pin CE y por ende se consume más potencia que el modo I.

- **Modo de recepción (RX MODE):** En este modo, las señales recibidas desde el medio, son demoduladas y presentadas al protocolo Enhanced ShockBurst, el mismo que analizará el paquete y lo colocará en el buffer de recepción. Si el buffer está lleno, descartará el paquete recibido.
- **Modo de transmisión (TX MODE):** Es el modo para permitir el envío de paquetes, para habilitar este modo se debe configurar el dispositivo según la plataforma de programación que se use, y habilitar el PWR_UP en alto. En la Tabla 2 se describe como configurar los modos de operación para el nRF24L01.

Tabla 2. Modos de operación del nRF24L01 [8]

MODO	REGISTRO PWR_UP	REGISTRO PRIM_RX	CE
Modo RX	1	1	1
Modo TX	1	0	1
Modo TX	1	0	10 μ s en alto
Standby-II	1	0	1
Standby-I	1	-	0
Power Down	0	-	-

Para el caso del presente proyecto de titulación se utilizó el módulo nRF24L01 en el modo de Operación TX y RX. Para el caso del modo de transmisión primario TX, se debe habilitar el PWR_UP en alto, por lo que se debe configurar el pin CE en 1. Mientras que para activar el modo de recepción RX, se debe configurar el pin CE en alto y así el nRF24L01 empezará a buscar los paquetes que se hayan recibido [8].

5. MATERIALES Y MÉTODOS

5.1 MATERIALES

Para la elaboración del presente proyecto de titulación se utilizaron los siguientes materiales y equipos:

Tabla 3. Materiales utilizados para el desarrollo del proyecto [El Autor].

MATERIALES	UNIDAD	CANTIDAD
Resistencias	Unidad	20
Diodos 2N2222	Unidad	6
Transistores	Unidad	12
Capacitores	Unidad	8
Relés	Unidad	6
Reguladores 7805	Unidad	2
Reguladores 7812	Unidad	2
Borneras	Unidad	6
Interruptor de tope	Unidad	2
Cables de conexión	Paquete	1
Caja para adecuación del prototipo	Unidad	3
Batería de Litio 11.1V	Unidad	1
Cautín	Unidad	1
Polea	Unidad	1
Cordel	Metros	3
Estaño	Metros	1
Pasta para soldar	Unidad	1

Tabla 4. Equipos utilizados para el desarrollo del proyecto [El Autor].

EQUIPOS	UNIDAD	CANTIDAD
Transformador 24V – 3A	Unidad	1
Arduino MEGA 2560 R3	Unidad	5
Pantalla TFT Touch HY-TFT320	Unidad	1
TFT Shield LCD	Unidad	1
NORDIC nRF24L01	Unidad	5

5.2 MÉTODOS

El desarrollo del presente proyecto está dividido en etapas, la primera parte define el hardware utilizado para cada parte del sistema, la comunicación inalámbrica utilizando los transceptores nRF24L01 y el desarrollo de la interfaz de usuario utilizando una pantalla TFT Touch y Arduino. En la segunda parte se detalla el desarrollo del software, las pruebas y mediciones realizadas al prototipo, y se anexa en resultados el presupuesto utilizado para el desarrollo total del proyecto.

5.2.1 PRIMERA PARTE: DESARROLLO DEL HARDWARE

La red domótica está constituida por los nodos periféricos que poseen un Arduino Mega 2560 como dispositivo inteligente, el cual es el encargado de procesar la información enviada por el prototipo controlador (control remoto) y enviarla hacia los elementos actuadores, según las acciones de control que solicite el usuario del sistema (véase Figura 5).

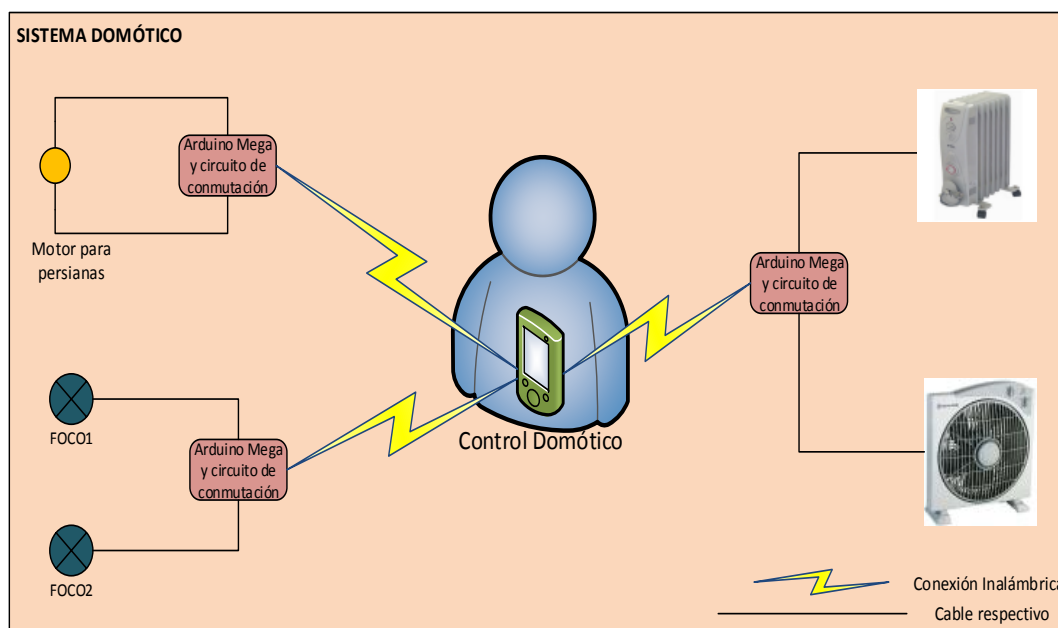


Figura 5. Diagrama de la red domótica [El Autor]

Los nodos periféricos constan de circuitos conmutadores, mediante el uso de relés como dispositivos actuadores. El diagrama utilizado para la conmutación ON/OFF del ventilador y calefactor está dado por la Figura 6.

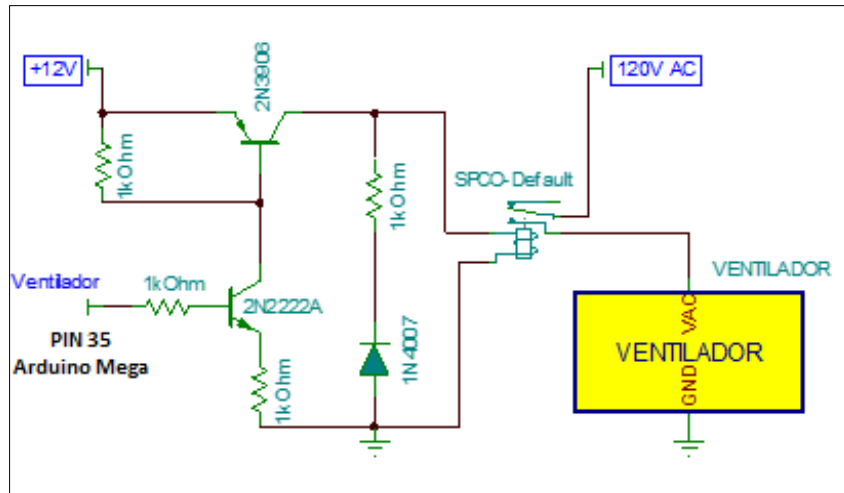


Figura 6. Diagrama de conexión para la conmutación del ventilador [El Autor]

El diagrama de la Figura 6 maneja el principio saturación de transistores bipolares BJT, en el cual lo que se desea es provocar la operación de dicho transistor en la región de saturación y obtener los 12V en corriente directa (DC, por sus siglas en inglés) de entrada entregados en este caso por la fuente de alimentación y conmutados gracias a la acción del PIN 35 del Arduino Mega; dicho voltaje accionará el relé de conmutación, permitiendo el paso de los 120V en corriente alterna AC, necesarios para funcionar el dispositivo final, es decir, el ventilador o calefactor (véase Anexo 4.1).

El circuito conmutador utilizado para la ventilación está orientado para operar con ventiladores de baja potencia comúnmente utilizados en un hogar. En el caso de este trabajo de titulación se utiliza un ventilador de pared, modelo MTX-BF1210 cuadrado de 12 pulgadas y que posee un consumo de potencia de 40W. La funcionalidad de dicho circuito de conmutación no ha sido probada con ventiladores industriales, por lo que no se garantiza la operabilidad del mismo. Igualmente el circuito conmutador utilizado para la calefacción está aplicado para trabajar con calefactores o termostatos eléctricos de bajo consumo dentro del hogar. Para el presente trabajo de titulación se utiliza un calefactor oleo eléctrico UT- 7T, sólido de acero esmaltado color blanco, peso de 116kg y consumo de potencia de 600W (véase Anexo 4.1, Figura 35).

Para la apertura y cierre de cortinas, se muestra la parte del manejo del motor con la finalidad de controlar el sentido de giro del mismo y debido a que el motor síncrono consume un voltaje de 34V, se realiza la alimentación desde la fuente de poder (véase Figura 7).

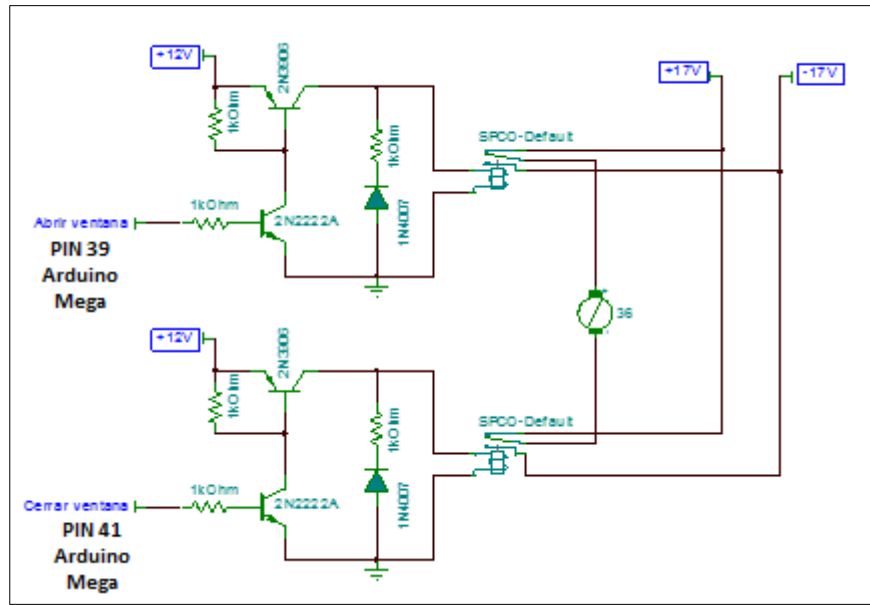


Figura 7. Conexión del motor para apertura y cierre de persianas [El Autor]

El diagrama de conexión para la apertura y cierre de persianas está basado en el mismo principio y configuración del diagrama de conexión para la conmutación del ventilador, es así que desde el PIN 39 del Arduino Mega se tiene destinado abrir las persianas, para esto se utiliza los +30V entregados por la fuente de poder y se alimenta dicho motor. Para el caso de cerrar las persianas, desde el PIN 41 del Arduino Mega se activa el circuito conmutador y se entregan 12V, mismos que accionan el relé y permite el paso de -30V, lo que provoca que el motor gire en sentido contrario y pueda ejecutar la acción solicitada.

El circuito conmutador utilizado para la apertura y cierre de persianas está orientado para operar con motores síncronos que tengan un consumo de aproximadamente 34V DC, para valores de tensión superiores no se recomienda su uso, ya que en este caso se debe realizar una nueva fuente de alimentación dependiendo del consumo del motor a utilizarse.

La apertura y cierre de las cortinas se realiza mediante el acarreo de las mismas, las cuales están sujetas al motor a través de un cordel, por lo que el peso y ancho de las cortinas no será un factor determinante en la búsqueda del voltaje necesario para realizar dicha acción. Además se utiliza un relé de tope para evitar que las cortinas se abran o cierren más de lo necesario.

Para la conmutación ON/OFF de la iluminación, se utiliza el circuito de conmutación de la Figura 8, el cual a más de ser accionado por el control remoto puede ser manipulado manualmente a través de su respectivo interruptor. Debido que el circuito conmutador para la iluminación trabaja como un interruptor, el dispositivo final a usarse puede ser cualquier tipo de luminarias, en este caso se utilizan los bombillos de luz usados comúnmente en un hogar cuyo consumo de potencia oscila entre los 8W y 12W tanto en colores blanco cálido y de luz del día.

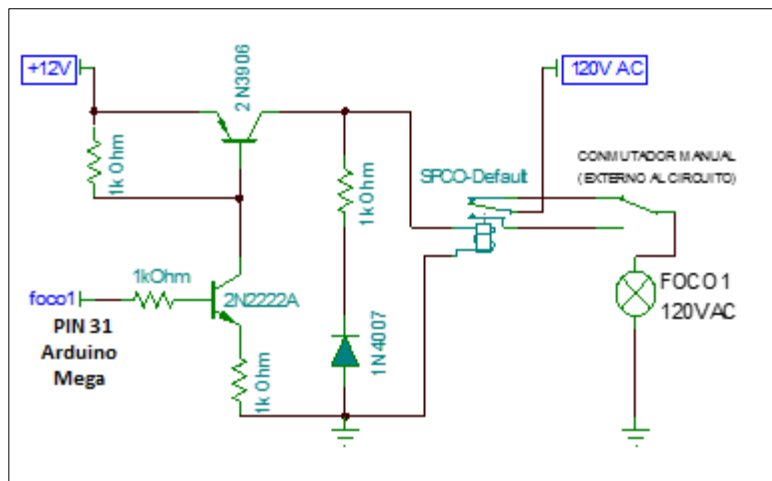


Figura 8. Circuito de conmutación para la iluminación [El Autor]

5.2.1.1 COMUNICACIÓN INALÁMBRICA

La comunicación inalámbrica está basada en una red WPAN con una estructura punto multipunto, mediante el uso de los módulos Nordic nRF24L01 y los Arduino Mega 2560 R3, distribuidos tanto en el control remoto como en los nodos periféricos. A continuación se realiza la definición del hardware del sistema, propia para este segmento.

- **Módulo nRF24L01:** Como se describe en la sección d.2.2.4, el transceptor inalámbrico nRF24L01 (véase Figura 9) es un módulo integrado de bajo consumo de energía, opera en la banda sin licencia ISM, es decir, en los 2.4 GHz a velocidades de hasta 2Mbps. Con corrientes menores a 14 mA y alimentación de 3.3 voltios, ofrece una solución que permite una larga duración de batería soportando una interfaz SPI de alta velocidad para el controlador de la aplicación, en este caso el microcontrolador Atmega 2560 perteneciente a la placa Arduino Mega, respectivamente [8], [16].

Además, incluye el protocolo de hardware *Enhanced ShockBurst*, encargado del manejo de tramas enviadas y recibidas, así como el uso de códigos de redundancia cíclicos, por lo que evita la retransmisión manual de tramas a causa de errores o interferencias [8], [16].

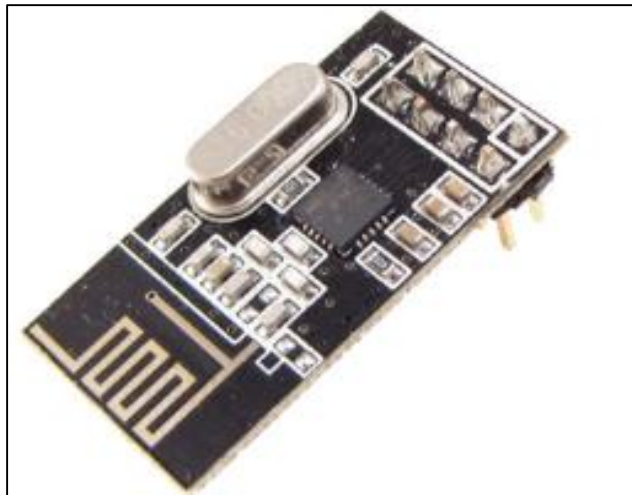


Figura 9. Módulo nRF24L01 [17]

Entre las características más interesantes que presenta el módulo nRF24L01 tenemos (véase Anexo 2.2) [17]:

- Módulo transceiver, es decir, opera como transmisor y receptor dependiendo de la declaración del modo de operación.
- Trabaja en 2.4GHz con 125 canales de operación.
- Alcance de 50 metros en interiores y de 70 a 100 metros en exteriores.
- Velocidades de 250 kbps a 2 Mbps.
- Bajo consumo de energía y fácil instalación.
- Fuente de alimentación de 3.3 V.
- Recepción de sensibilidad de -82dBm a 2Mbps y -85dBm a 1 Mbps.
- Potencia de transmisión configurable a 0, -6, -12 y -18dBm.
- Ganancia de antena de 2dBi.
- Modulación GFSK

El dispositivo se comunica por SPI, permitiendo mayores velocidades de comunicación entre el módulo nRF24L01 y los nodos, en este caso con el Arduino Mega 2560 R3 [19].

Es importante tener en cuenta los pines de conexión del nRF24L01 (véase Figura 10 y Figura 11), para poder establecer la comunicación del módulo con otros dispositivos, en este caso para establecer una conexión con el Arduino Mega [17].

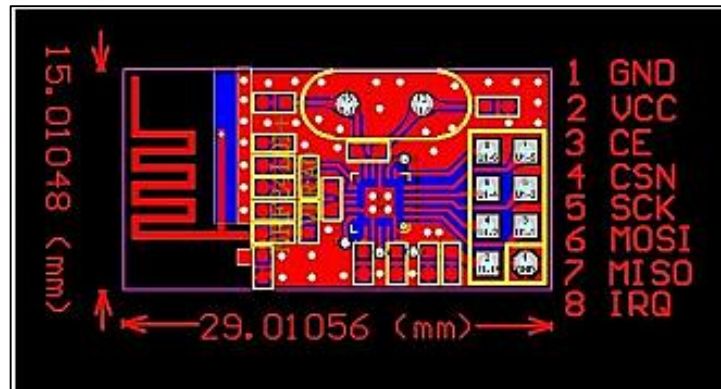


Figura 10. Pinout del módulo nRF24L01 [17].

A continuación se describe los respectivos pines de conexión del módulo transceptor nrF24L01 (véase Anexo 2.2) [17]:

- **Pines 1 y 2:** Pines de alimentación, GND (tierra) y VCC, respectivamente.
- **Pin 3:** Pin de Chip Enable (CE), el mismo que sirve para activar y desactivar la comunicación SPI.
- **Pin 4:** Pin CSN utilizado para indicar cuando se transmiten o reciben datos, igualmente utilizado para la comunicación SPI.
- **Pin 5:** Pin CSK o de reloj para la sincronización. Permite que durante una conmutación de nivel bajo a nivel alto. Se transmita un bit de datos.
- **Pines 6 y 7:** Pines MOSI y MISO. Estos son los conectan los cables por donde viajan los datos de entrada y salida del chipset NRF.
- **Pin 8:** Pin IRQ, No es muy usado, pero se lo puede configurar para que se active y avise la recepción de un dato por radio frecuencia.

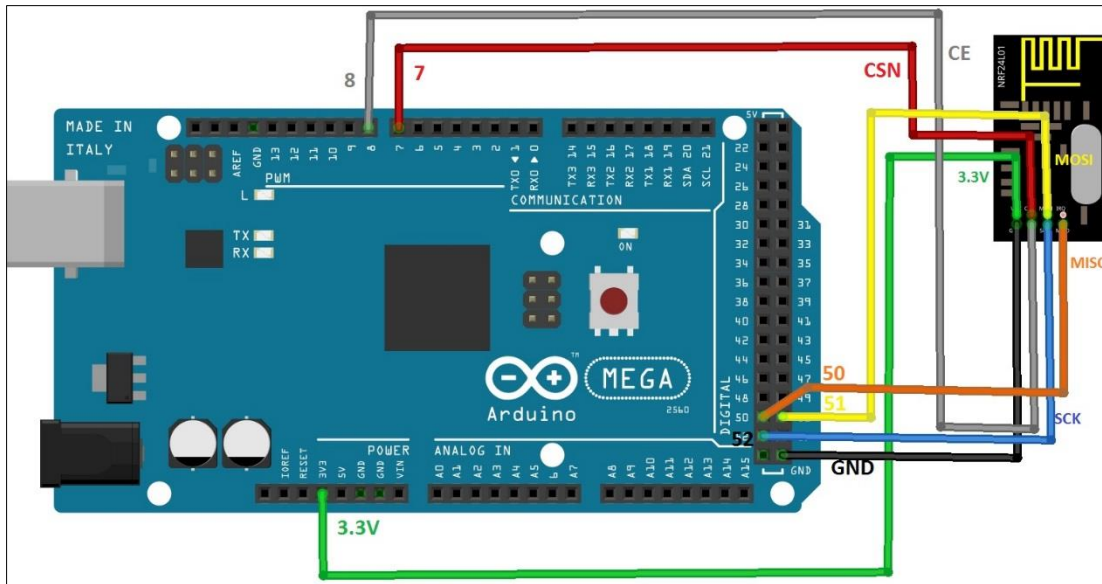


Figura 11. Conexión del Arduino Mega con el módulo nRF24L01 [El Autor]

De acuerdo a los modos de operación del nRF24L01, citados en la sección d.2.2.1.3, se utilizó el módulo nRF24L01 en el modo de Operación TX y RX. Para el caso del modo de transmisión primario TX, se debe habilitar el PWR_UP en alto, por lo que se debe configurar el pin CE en 1. Así, si existe un paquete en el buffer de transmisión, el nRF24L01 ingresa al modo PTX y transmite el paquete. En el caso de la bandera NO_ACK, estará habilitada para que el TX reconozca que por lo que el nRF24L01 ingresará al modo de recepción RX para recibir el correspondiente acuse de recibo ACK. Igualmente, el ACK que se reciba tendrá un valor 0, lo que permitirá al módulo nRF24L01 habilitarse en el modo Standby I y así consumirá menos energía [8].

Para activar el modo de recepción RX, el pin CE se configura en alto y así el nRF24L01 empezará a buscar los paquetes que se hayan recibido. Del mismo modo, como la bandera NO_ACK está habilitada entonces el módulo nRF24L01 ingresará al modo TX y emitirá el ACK con el valor de 0 y por ende ingresará al modo Standby I [8].

- **Comunicación SPI:** Cuando se quiere transmitir datos desde un componente electrónico a otro, se puede hacer de múltiples formas. Una de ellas es mediante la comunicación serial, en donde la información es transmitida bit a bit por un único

canal, enviando por tanto un bit tras otro en cada momento. La otra manera de transferir datos es utilizando la comunicación paralela, en la que se envían los bits de forma simultánea, cada uno por canales separados y sincronizados [20].

El protocolo de interfaz periférica serial (SPI, por sus siglas en inglés) es un tipo de comunicación serial utilizada por muchos componentes electrónicos, tal es el caso del Nordic nRF24L01 como del microcontrolador Atmega 2560 utilizados en este proyecto de titulación [20].

Este sistema de comunicación permite controlar dispositivos electrónicos digitales que acepten flujos de bits sincronizados. Utiliza cuatro líneas de comunicación, una línea llamada CSK que envía a todos los dispositivos la señal de reloj generada por el dispositivo maestro actual; otra línea conocida como CSN utilizada por el maestro para elegir con cual dispositivo esclavo se quiere comunicar; otra línea llamada MOSI que es utilizada para enviar los datos digitales (1s y 0s) desde el maestro hacia el esclavo elegido, y finalmente la línea MISO que es utilizada para enviar los datos en sentido contrario, es decir, desde el esclavo hacia el maestro [20].

5.2.1.2 INTERFAZ DE USUARIO

La interfaz de usuario está conformada por la plataforma Arduino Mega 2560 R3, una pantalla táctil resistiva TFT LCD, una shield TFT LCD Mega para la adaptación de la pantalla táctil con la placa Arduino y el correspondiente módulo nRF24L01 terminal para la comunicación inalámbrica con el Arduino Mega central. El diseño de la interfaz está desarrollada en el entorno de programación IDE de Arduino, para la correspondiente programación de la placa y compilación del mismo.

En el siguiente apartado se describe el hardware del sistema utilizado para el desarrollo tanto del software como del hardware de la interfaz de usuario.

- **Arduino Mega 2560 R3:** Actualmente, con el avance de la tecnología, existen empresas que desarrollan software y hardware libre, por lo que es posible optar por dispositivos y elementos que poseen la característica de ser Open Source, es decir con

licencia libre y de bajo coste, cuyo fin primordial es tener la acogida y aceptación por parte de los programadores potenciales que existen en el mundo [20].

La plataforma Arduino es un claro ejemplo de tecnología Open Source, está basado en el microcontrolador de marca Atmel y cuenta con puertos de entrada/salida tanto analógicas como digitales. Es una plataforma que simplifica el proceso de trabajo con microcontroladores y ofrece ventajas tanto para profesores como estudiantes y aficionados, entre estas ventajas tenemos [14], [21]:

- Económico,
- Soporte multiplataforma,
- Entorno de programación simple,
- Código abierto,
- Software y hardware extensible.

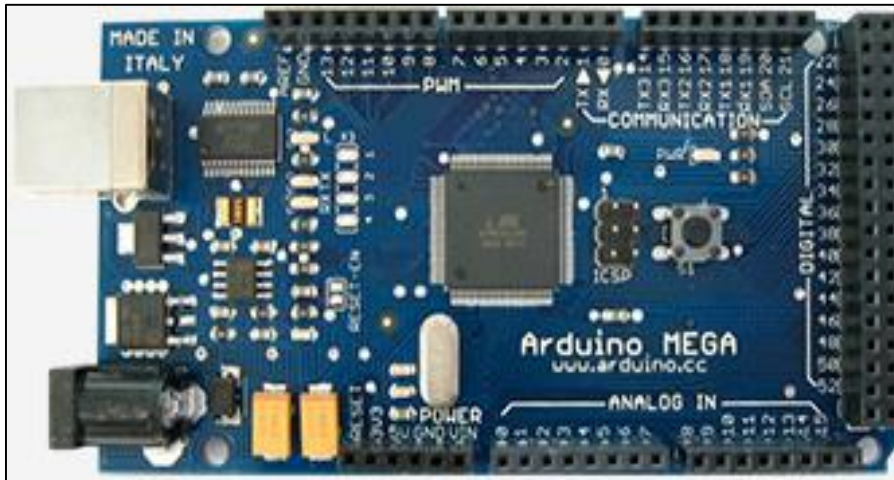


Figura 12. Arduino Mega 2560 R3 [El Autor]

El Arduino Mega 2560 R3 (véase Figura 12) está basado en el microcontrolador ATmega 2560, tiene 54 pines digitales distribuidos tanto para entrada como para salida, posee 16 entradas analógicas, 4 puertos seriales o UARTs, conector ICSP, jack para la conexión a la electricidad y un botón de reset [20], [21].

A continuación se resumen las características del módulo Arduino Mega 2560 R3, con la finalidad de conocer la capacidad de almacenamiento y las prestaciones técnicas que brinda esta plataforma [20], [21]:

- Trabaja a 16 MHz con un voltaje de 5 V.
- SRAM de 8 KB.
- Memoria EEPROM de 4 KB.
- Memoria FLASH de 256 KB.
- 54 pines digitales (15 pines para PWM)
- 16 pines analógicos.

El Arduino puede ser alimentado a través de la conexión USB o con un suministro de energía externo, esta alimentación puede venir desde un adaptador AC a DC o desde una batería. La placa puede operar con un suministro externo de 6 a 20 voltios, sin embargo, si se suministra menos de 5 voltios la placa puede ser inestable, y si se usa más de 12 voltios el regulador de tensión puede sobrecalentarse y dañar la placa. Arduino tiene los siguientes pines de alimentación [14], [21], [22]:

- **Vin:** es la entrada de tensión a la placa Arduino. Se puede suministrar tensión a través de este pin o, si se suministra tensión a través del conector de alimentación, se puede acceder a él a través de este pin.
- **5V:** esta es la fuente de tensión regulada y usada para alimentar al microcontrolador Atmega 2560.
- **3.3V:** Fuente de voltaje a 3.3 voltios generada por el chip FTDI de la placa, con una corriente máxima de 50mA.
- **GND:** Pines de tierra.

Para la comunicación con el computador u otro microcontrolador, el Arduino Mega posee cuatro puertos de comunicación UART vía serie. El chip integrado FTDI

FT232RL canaliza la comunicación serie a través del USB y los *drivers* que ya vienen incluidos en el software de Arduino proporcionan un puerto serie virtual en el ordenador. Además, el IDE de Arduino posee un monitor de puerto serial que permite visualizar textualmente el envío y recepción de los datos [14], [21].

- **Pantalla LCD TFT Touch:** Actualmente es común observar equipos electrónicos portátiles y fijos con interfaces de usuario, los mismos que tienen en común el uso de pantallas táctiles de muy buena resolución y colores. Aunque las pantallas táctiles tienen su aparición desde los años 60, no es sino desde inicios de los años 80 que empiezan a ser consideradas para formar parte de los primeros dispositivos táctiles, y de forma masiva tuvieron mayor acogida las pantallas resistivas [23].

Como definición se puede manifestar que una pantalla táctil es aquella que al realizarle un toque directo sobre su superficie permite la entrada de datos y órdenes al dispositivo sobre el que esté conectada, y a su vez muestra los resultados de las acciones que se introdujeron previamente, por tanto se puede decir que una pantalla táctil actúa como periférico de entrada y salida de datos [23].

Se sabe que existen dos tipos de pantallas táctiles, resistivas y capacitivas las mismas que son más comunes en la actualidad [24]. Las pantallas táctiles, tanto resistivas como capacitivas, han empezado a tener mayores aplicaciones debido a que los programadores prefieren usarlas ya que permiten que sus diseños sean más intuitivos para el usuario e igualmente son más interactivas [23]. De estos dos tipos de pantallas, daremos una referencia sobre las pantallas resistivas por ser la tecnología usada por la pantalla TFT Touch V2.0.

La pantalla táctil resistiva está formada por dos capas de material conductor transparente, con una cierta resistencia a la corriente eléctrica y con una separación entre las dos capas. Este tipo de pantallas poseen un sistema electrónico que detecta el contacto realizado en la superficie, mide la resistencia y luego calcula el punto de contacto [18], [24].

La principal característica de estas pantallas es que son económicas frente a las pantallas capacitivas; son más fiables y versátiles, además las pantallas resistivas tienen una desventaja con uso de material transparente sobre la pantalla, ya que este hace que con el tiempo la pantalla pierda luminosidad [18], [25].

Las pantallas táctiles de transistor de película delgada (TFT, por sus siglas en inglés), se tratan de pantallas táctiles resistivas con una tecnología basada en transistores de efecto de campo [14].

Para el desarrollo del presente trabajo, la pantalla TFT Touch (véase Figura 13), tiene una resolución de 320 por 240, su tamaño es de 3.2 pulgadas, tiene una resolución de 65.536 colores, tarjeta SD integrada y basa su funcionamiento en el controlador SSD1289 para gestionar la pantalla TFT y el controlador ADS7843 para gestionar el panel táctil. Como ya sabemos utiliza transistores, los cuales permiten mejorar la calidad de imagen en comparación con las tradicionales pantallas LCD, y, consumen menos energía [14].

Para iniciar con la programación es necesario conocer el modelo de controlador de la pantalla TFT ya que debe ser declarado en el código fuente. Cabe recalcar que actualmente existen muchas librerías para el manejo de estas pantallas, lo que facilita el trabajo al momento de programar, en este caso se utilizó la librería TFT creada por el desarrollador Henning Karlsen, a la cual se le debe modificar la línea **myGLCD** (**ITDB32S**, **38**, **39**, **40** **41**) por el controlador de la pantalla TFT que se tenga adquirida, en este caso se reemplaza el controlador **ITDB32S** por el controlador **SSD1289**, en caso de no hacerlo se tendrá inconvenientes puesto que la pantalla no funcionará correctamente y sólo nos presentará una pantalla en blanco [14].



Figura 13. Pantalla TFT Touch LCD [El Autor]

Los valores declarados en la línea **myGLCD (ITDB32S, 38, 39, 40 41)** corresponden a los pines de la placa Arduino donde equivalen a los pines RS, WR, CS, RSET de la pantalla respectivamente. Los valores vienen por defecto, pero pueden ser modificados de acuerdo al criterio del programador según los pines que se desee utilizar. Adicional a esto, para el uso de la tarjeta SD, podemos colocar una tarjeta en la parte posterior de la pantalla TFT, pero para la visualización de las imágenes deben ser convertidas en formato RAW y están limitadas al uso de memorias SD con formato FAT16 o de 2 GB de tamaño.

- **Shield TFT LCD Mega:** Las placas Shield para Arduino son aquellas que tienen la función de extender las capacidades del Arduino Mega, simplemente se las debe colocar sobre la placa Arduino y empezar a trabajar. Estas *shield* están elaboradas para facilitar las conexiones de dispositivos al Arduino, tal es el caso de la Shield TFT LCD Mega (véase Figura 14), usada en el desarrollo de este proyecto, la misma que permite una mejor conexión y adhesión con el Arduino Mega.

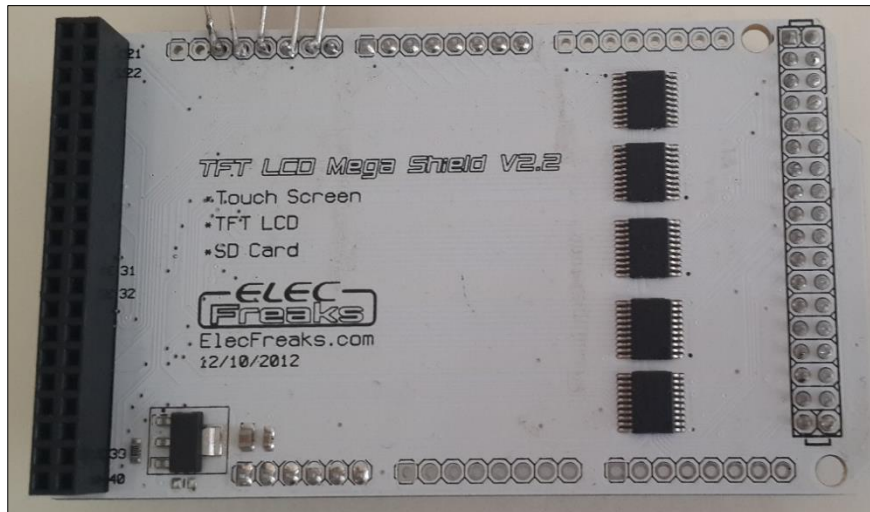


Figura 14. Shield TFT LCD [El Autor]

La conexión final del prototipo controlador será realizada como la Figura 15, para luego proceder a la programación del dispositivo y controlar el sistema domótico basado en Arduino Mega 2560 R3.

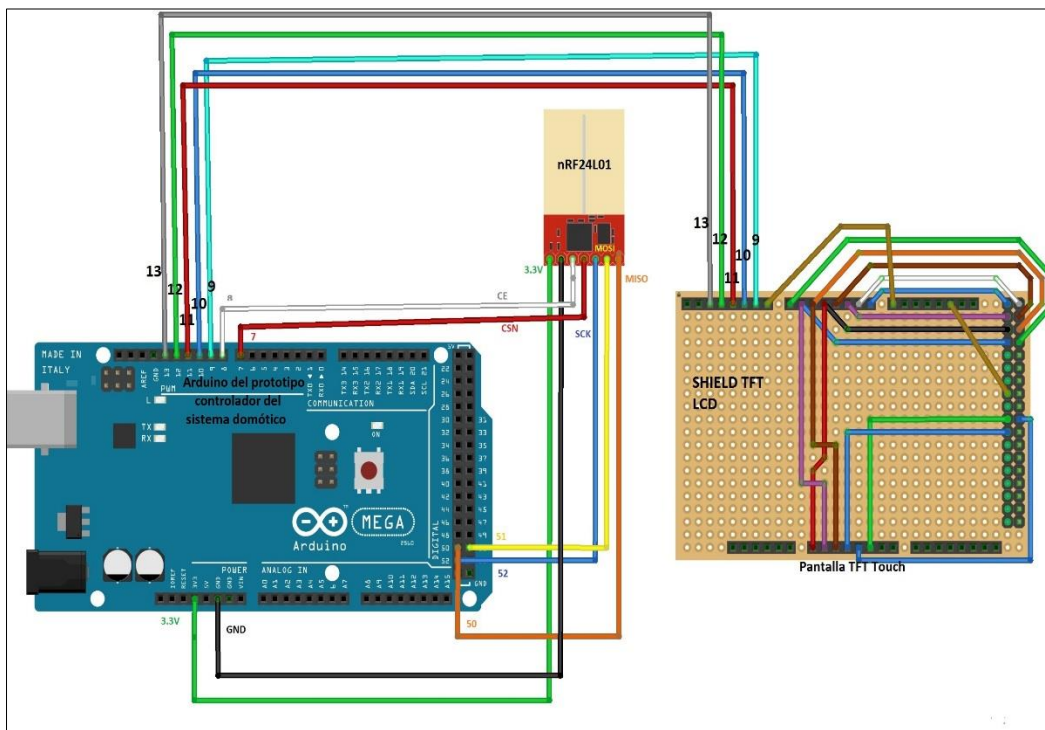


Figura 15. Conexión del prototipo controlador [El Autor]

5.2.1.3 SEGUNDA PARTE: DESARROLLO DE SOFTWARE

El desarrollo de software, tanto para los módulos nRF24L01, Arduino Mega y pantalla táctil resistiva TFT, está realizado con el entorno de desarrollo para la programación en Arduino, del cual se detalla a continuación:

- **Entorno de desarrollo para programación en Arduino:** Para la programación de microcontrolador Atmega de la placa Arduino es necesario tener instalado el IDE o entorno de desarrollo integrado cuya sintaxis de programación son los lenguajes C y C++. Además, esta plataforma de programación corre sobre cualquier Sistema Operativo conocido, como Windows, Mac OS y Linux. [20].

El IDE (véase Figura 16), está constituido por un editor de texto en donde se realizan los programas o sketch, un área de mensajes, una consola de texto y una barra de herramientas en donde se despliegan una serie de menús [14]. En el editor de texto se realiza la programación necesaria para luego ser compilada, en caso de que existan errores, en el área de mensajes se muestra cuáles son las líneas erróneas con el fin de que sean modificadas.

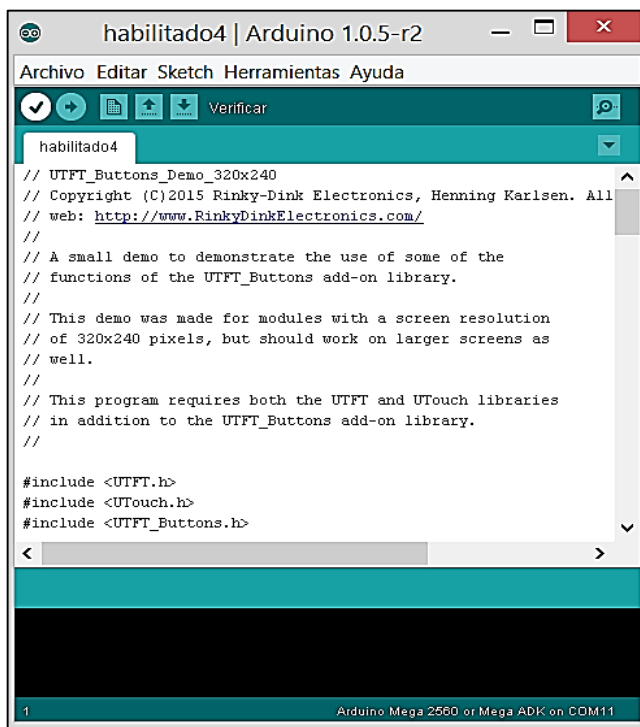


Figura 16. IDE o entorno de desarrollo integrado [El Autor]

Se inicia seleccionando el modelo de la placa Arduino con la que se va a trabajar (véase Figura 17), y en el menú Herramientas se selecciona la tarjeta Arduino Mega 2560.

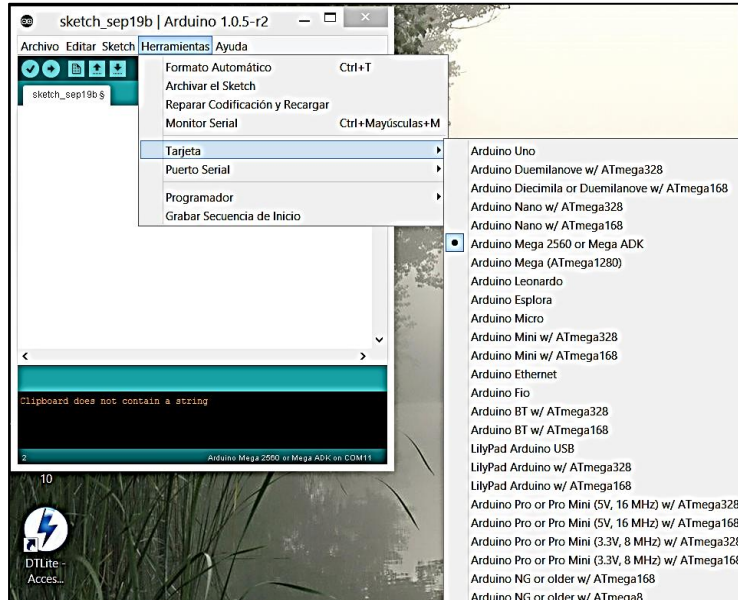


Figura 17. Selección de tarjeta Arduino Mega [El Autor]

Luego de elegir la tarjeta procedemos a seleccionar el puerto serial del Arduino (véase Figura 18).

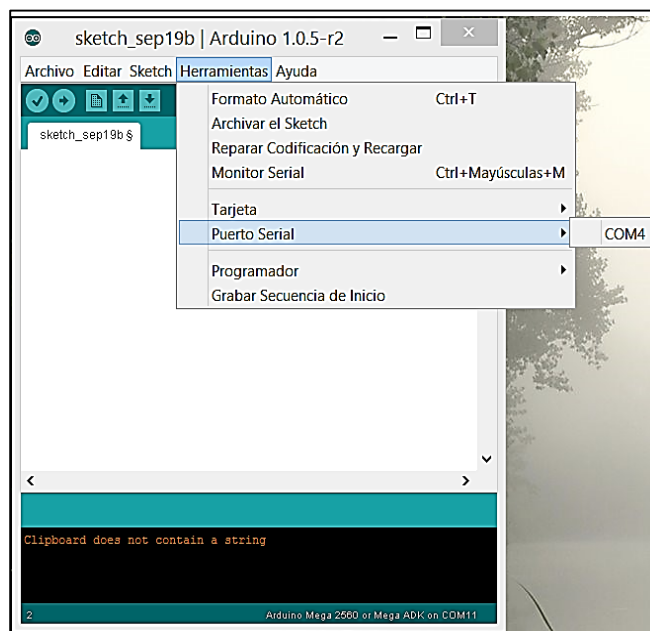


Figura 18. Selección de puerto serial [El Autor]

Una vez hecho los pasos anteriores, se empieza a escribir el sketch o programa deseado, tomando en cuenta el lenguaje de programación del IDE, además para la utilización de librerías propias de Arduino o de terceros se las debe importar descomprimiéndolas en la carpeta creada al momento de instalar el software.

- **Codificación de programa**

La programación de la interfaz de usuario, del Arduino Mega central y de los módulos nRF24L01 para la comunicación inalámbrica está desarrollada en el IDE de Arduino 1.0.5 descrito previamente. Para el caso del objeto de estudio, es decir de la pantalla TFT Touch, la programación de la interfaz de usuario está basada en la visualización de una pantalla de menú principal en donde se tiene las siguientes opciones:

- Persianas,
- Ventilador,
- Calefactor,
- Luz 1 y luz 2.

El diagrama de flujo desarrollado para la pantalla TFT y utilizado para la interfaz de usuario se presenta en la Figura 19. El algoritmo inicia declarando las librerías TinyFat y UTFT, las mismas que servirán para poder trabajar con la parte táctil de la pantalla TFT. Luego se declaran las diferentes constantes y variables del sistema con los respectivos pines del Arduino, es decir declaramos variables para los pines 9, 10, 11, 12 y 13 respectivamente. Una vez realizado esto, se procede a inicializar o activar la pantalla TFT LCD para poder realizar la configuración de la interfaz, en donde se fijará el tipo, tamaño, color de letra y se establecerán como SALIDAS en estado BAJO a los pines 9, 10, 11, 12 y 13. Seguido de esto se inicia la comunicación serial con el Arduino Mega 2560 R3 para poder realizar las acciones de control. Para lo cual, se debe crear en la pantalla los respectivos botones que serán quienes ejecuten la acción de “abrir”, “cerrar”, “encender” o “apagar” mediante los circuitos de conmutación instalados en el sistema domótico, aplicados a las persianas, el ventilador, las luces o el calefactor. Se imprimirá en la pantalla el estado en que se encuentre cada artefacto, esto con el fin de evitar confusión en el usuario. Por tanto

se tiene que al inicio de la ejecución de la pantalla, el usuario solamente puede seleccionar las opciones “abrir” y “encender”, el resto estarán inhabilitadas y se habilitarán en cuanto se hayan seleccionado cualquiera de las dos opciones anteriores.

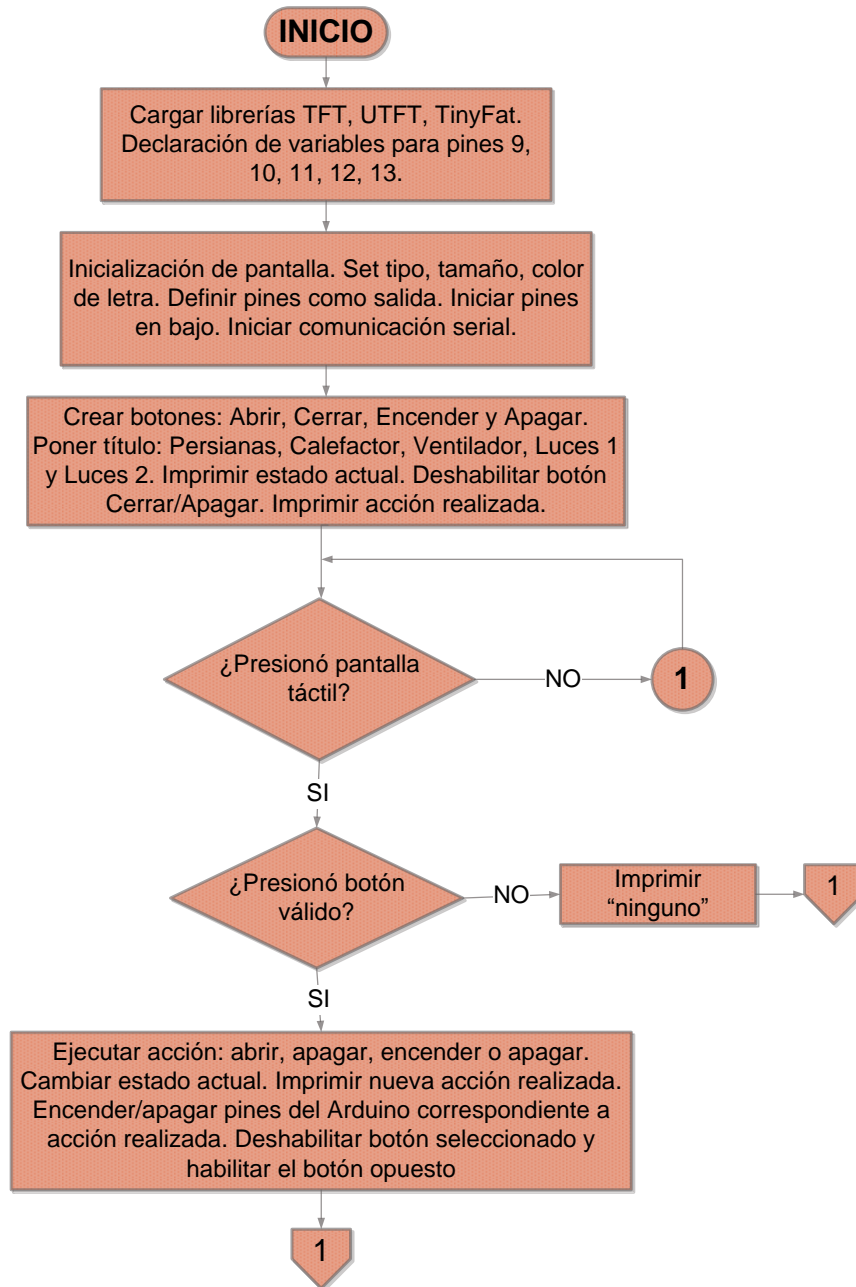


Figura 19. Diagrama de flujo para la pantalla táctil [El Autor]

Para la comunicación inalámbrica realizada por el módulo nRF24L01, se tiene el diagrama de la Figura 20 en donde se puede observar que el programa se inicializa con la declaración de variables utilizadas por los pines del Arduino Mega 2560 R3. Cabe recalcar que para mejor utilización de estos módulos es conveniente utilizar librerías que permiten el mejor manejo del mismo, en este caso se decidió utilizar la librería Mirf de Arduino ya que es la más completa y tiene muy buenas prestaciones para la realización de este proyecto de tesis.

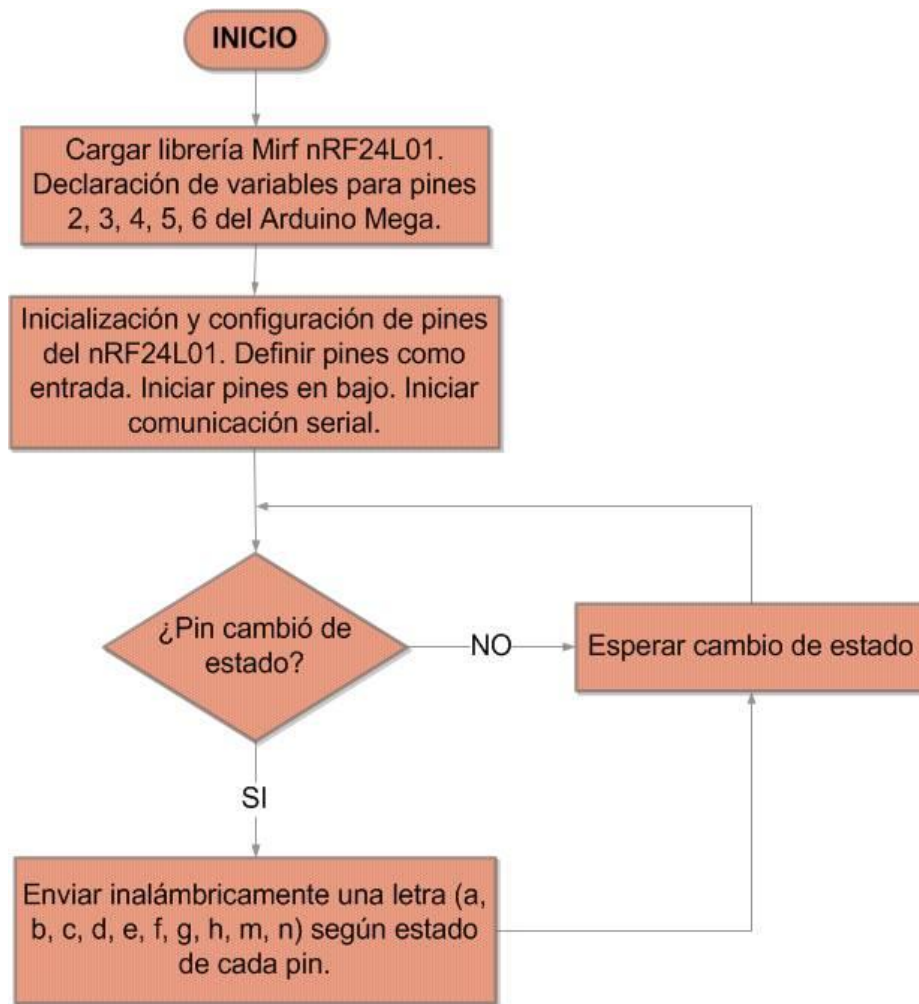


Figura 20. Diagrama de flujo del módulo nRF24L01 [El Autor].

La programación utilizada por la unidad central está basada en el siguiente diagrama de flujo (véase Figura 21), en esta parte central se configura el módulo nRF24L01 en modo RX, es decir en modo de recepción. Se leen los datos y finalmente se realiza la acción de control programada para cada pin del Arduino Mega 2560 R3.

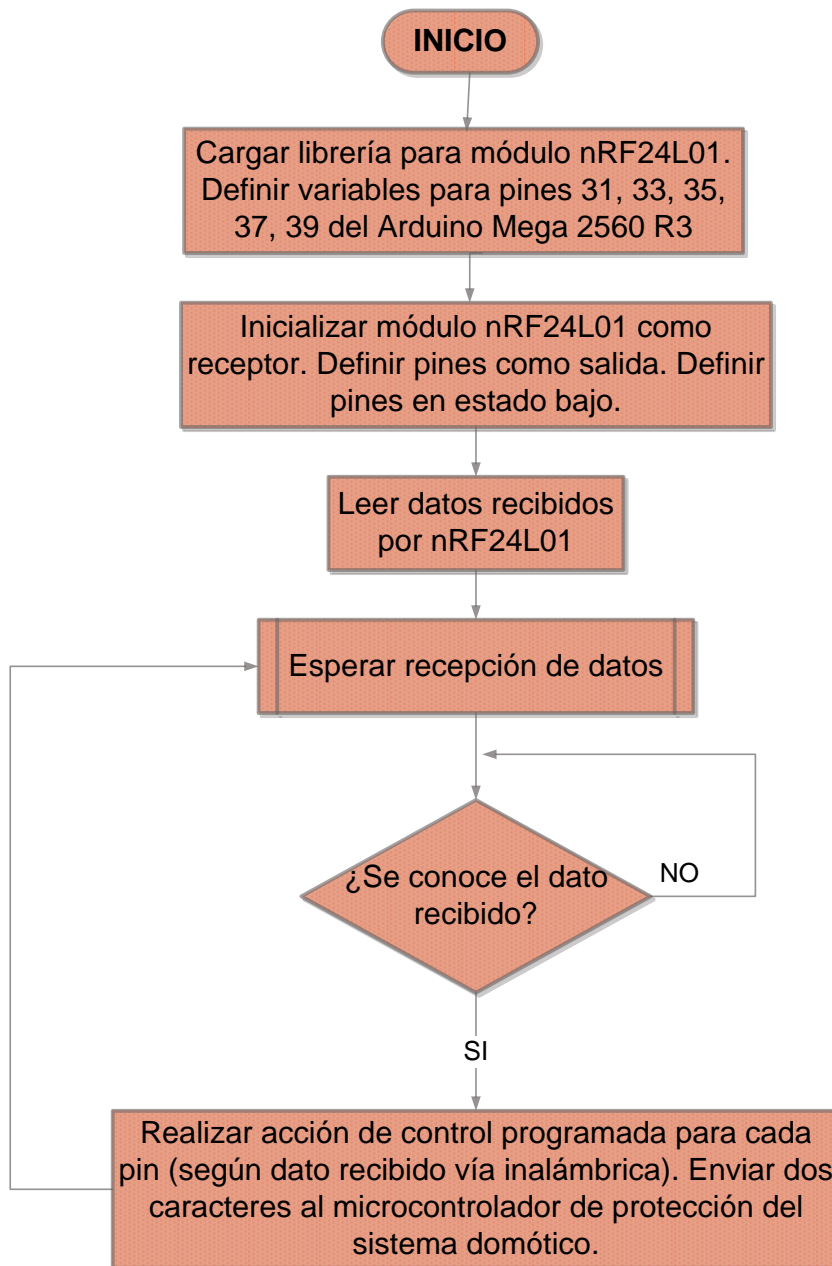


Figura 21. Diagrama de flujo para Arduino Mega central.

En cuanto a los códigos de programación propios para el prototipo, éstos se encuentran ubicados en el anexo A, los mismos que pueden ser utilizados para futuros proyectos o implementaciones domóticas, pues como se manifestó anteriormente la idea de utilizar herramientas Open Source es con el fin de contribuir a la distribución libre tanto de software como de hardware.

5.2.2 FUNCIONALIDAD DEL PROTOTIPO

5.2.2.1 Pruebas y mediciones

Una vez implementado el prototipo, se procede a realizar la validación del mismo, con el fin de comprobar que satisface el uso para el que ha sido creado. Es así, que se realizaron pruebas durante la implementación de la red domótica a controlar para verificar que el software diseñado se comportará correctamente y acorde a lo diseñado.

Para lograr obtener una buena funcionalidad, se realizaron pruebas de defecto, en donde se logró encontrar inconsistencias en el código del programa, gracias a estas pruebas se localizaron las fallas del código, se reemplazaron y se volvió a compilar el microcontrolador. Una vez realizada la corrección, se procedió a realizar nuevamente las pruebas de defecto para garantizar que el sistema estaba corregido.

Es importante considerar la conexión de los pines de los dispositivos usados, es decir, tanto la conexión de los pines de la pantalla TFT con el Arduino Mega, como la conexión del módulo transceiver nRF24L01 con el Arduino Mega 2560. En las siguientes tablas se resume la conexión de los pines de los dispositivos utilizados en este proyecto.

Tabla 5. Conexión de pines para el nRF24L01 con el Arduino Mega [El Autor].

PIN	MODULO nRF24L01	Arduino Mega
GND	1	GND
VCC	2	3.3 V
CE	3	9
CSN	4	53
SCK	5	52
MOSI	6	51
MISO	7	50
IRQ	8	-----

Tabla 6. Conexión de pines para la pantalla TFT con el Arduino Mega [El Autor].

PIN	Pantalla TFT Touch	Arduino Mega
GND	GND	36
VCC	5 V	5 V
CS	A0	A0
RS	A1	A1
WR	A2	A2
RD	A3	A3
IRQ	9	9
BUS DE DATOS	0 – 7	0 – 7

Como ya se había manifestado anteriormente, para la conexión de la pantalla TFT táctil con el Arduino Mega, se utilizó una Shield TFT Mega propia para la pantalla TFT, la misma que ayuda al conexionado de los pines, y evita molestos contratiempos.

Para verificar el correcto desempeño del sistema domótico junto con el prototipo controlador, y validar que cumple con las expectativas de diseño se instaló el sistema completo en una habitación, para luego proceder a realizar un test de verificación, esto con el fin de garantizar que tanto el hardware como el software diseñado cumplan las funciones previamente establecidas y que las acciones a controlar tengan retrasos aceptables.

6. RESULTADOS

6.1 SISTEMA IMPLEMENTADO

La red domótica implementada es una red punto a multipunto, conectada inalámbricamente, en donde cada nodo periférico está constituido por un Arduino Mega 2560 R3 que realiza el control de los elementos tales como las luminarias, motores y los sensores; y de un módulo nRF24L01 encargado de las recepciones de información en radiofrecuencia. Así mismo, en dichos nodos se encuentran ubicados la respectiva fuente de alimentación y el circuito conmutador destinado para el control de las persianas, calefactor y ventilador, según sea el caso.

Para la apertura y cierre de las cortinas se utiliza un cordel (propio para persianas/cortinas) y un motor síncrono, lográndose abrir y cerrar las cortinas a una velocidad media, con el fin de evitar una apertura estrepitosa y violenta. También, la programación está fijada para que las cortinas se abran hasta cierto punto debido a que posee relés de tope, y en caso de que las cortinas hayan quedado abiertas, si se presiona el botón “abrir” en el control remoto, no se ejecutará la acción. Lo mismo sucede al momento de cerrar las cortinas.

Para el nodo periférico destinado al control de la iluminación, se elaboró un circuito de conmutación pequeño que, realizando las respectivas conexiones con los cables eléctricos de dichos focos, permite la conmutación ON/OFF de los mismos, es decir, se puede encender un foco mediante el uso del control remoto diseñado, y también se puede apagar manualmente a través de su respectivo interruptor, y viceversa.

Al momento de inicializar el sistema domótico, este arranca con todos los pines en BAJO (0 lógico ó 0V), esperando que el control remoto envíe una acción de mando para poder accionar los dispositivos finales como ventilador, luces o persianas. En caso de que el usuario seleccione una de las opciones que ofrece el control remoto, se enviará la información al Arduino Mega periférico/receptor, y éste activará un pin determinado según la programación realizada, es decir, dicho pin se pondrá en ALTO (1 lógico o 5V) y permitirá el paso de corriente al circuito conmutador para que ejecute la acción de control correspondiente.

El módulo nRF24L01 transmisor enviará la información de control, solicitada por el usuario por intermedio de la pantalla TFT Touch, utilizando la banda ISM de 2.4 GHz a una velocidad de transferencia de 1 Mbps, hacia el módulo nRF24L01 receptor ubicado en el Arduino Mega del respectivo nodo periférico.

Para que no exista conflicto de acciones en los diferentes nodos periféricos, estos tienen un ID específico por lo que cuando el usuario necesite tomar una acción sobre la iluminación, calefacción o persianas, el módulo nRF24L01 receptor recibirá esta información y posteriormente la enviará a su Arduino Mega central, para que este a su vez reenvíe los datos a los dispositivos actuadores tales como relés, sensores o motores y así se ejecute la acción solicitada por parte del usuario.

6.2 INTERFAZ DE USUARIO

Para la interfaz de usuario que es presentada en el control remoto, se decidió utilizar una pantalla táctil ya que es amigable y facilita el uso hacia las personas mediante la aplicación de gráficas y órdenes sencillas. La pantalla utilizada es la Touch Screen HY-TFT320, la cual es un módulo de pantalla LCD de 320x240 píxeles de resolución con una gama de 65K colores, incluye el controlador SSD1289 más el integrado XPT2046 encargado de la interfaz táctil (véase Figura 13).

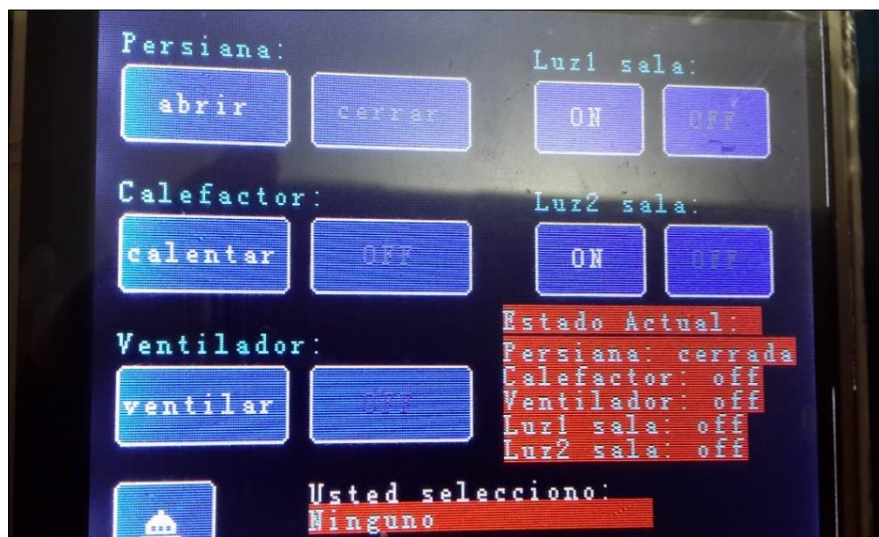


Figura 24. Opciones presentadas en la pantalla TFT [El Autor]

Además, el Arduino Mega 2560 R3 cuenta solo con pines hembra y para su conexión fue necesaria la utilización de la Shield TFT, la misma que permitió el correcto acoplamiento de la pantalla táctil hacia el Arduino Mega. La interfaz consta de cinco opciones, las cuales son: Persiana, Calefactor, Ventilador, Luz 1 y Luz 2 (véase Figura 24).

Cada opción tiene dos subpestañas con las opciones para abrir y cerrar los diferentes dispositivos a controlar. En el caso de “Persianas” se presentan las opciones “abrir” y “cerrar” las cuales pueden ser accionadas por el usuario mediante el panel táctil. Al momento de seleccionar la opción “abrir” se ejecutará la acción programada en el prototipo, es decir se abrirán las cortinas del sistema domótico, y a su vez solamente se habilitará la opción “cerrar” la cual será la contraparte a la acción antes ejecutada. Es decir, el usuario solamente puede hacer uso de una sola acción frente a las dos opciones presentadas “abrir” y “cerrar”. El mismo principio se utiliza en los casos del Calefactor, Ventilador y Luces (véase Figura 25).

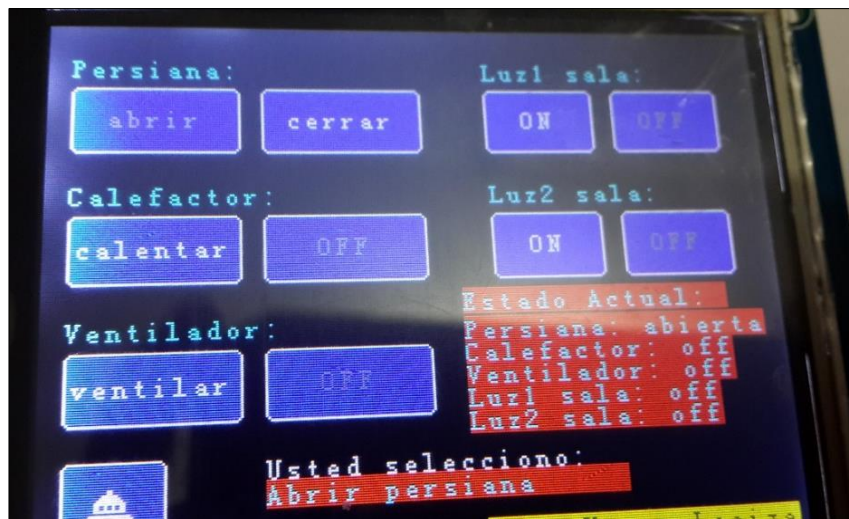


Figura 25. Opción “Persiana” abierta [El Autor]

Para mejor información se indica el estado actual de los parámetros del sistema domótico, es así que por ejemplo en el caso de “Persianas” al seleccionar la opción “abrir” en el recuadro Estado Actual de la interfaz aparecerá la información de: “Persiana: Abierta” y “Usted seleccionó: Abrir persiana” (véase Figura 26), así el usuario podrá recordar que parámetros tiene encendidos o apagados en el sistema domótico.

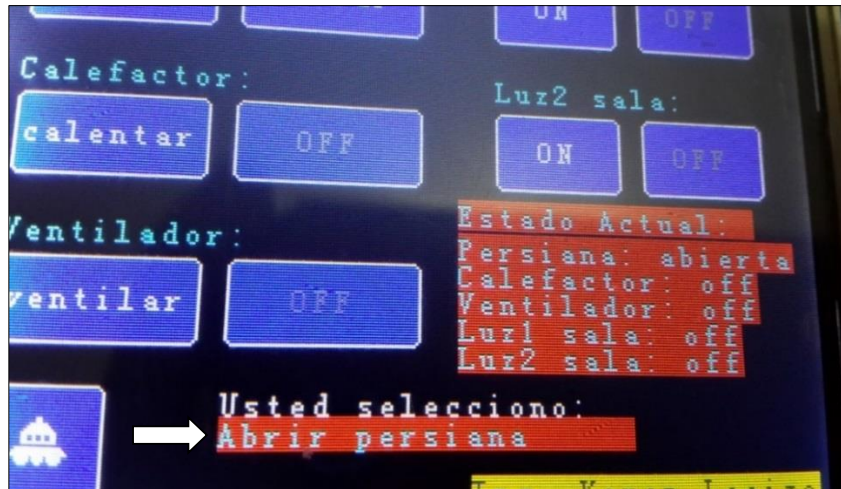


Figura 26. Información del estado actual del sistema [El Autor]

6.3 VALIDACIÓN DEL PROTOTIPO

La etapa de validación del prototipo está dividida en aspectos técnicos y económicos, así mismo se considera aspectos como la autonomía del prototipo controlador y características físicas del mismo.

6.3.1 Validación Técnica

Para esta validación se realizaron pruebas de medición de distancia con la finalidad de obtener información de la operabilidad y alcance de los módulos nRF24L01. Para esto se realizaron mediciones con el prototipo implementado, con la finalidad de obtener un tiempo promedio de retardo entre las acciones de control a ejecutarse por el dispositivo según la distancia en la que se encuentre. Como sabemos la banda ISM es una banda de frecuencias no licenciada muy utilizada para realizar comunicaciones en radiofrecuencia, lo que causa que muchos artefactos y dispositivos trabajen libremente en dicha banda y por ende la tienen saturada en su totalidad; esto hace que se produzcan interferencias en los módulos nRF24L01 que operan en las frecuencias del rango de los 2.4 GHz. Así mismo, tanto la distancia de ubicación como los obstáculos dentro de una vivienda o edificio tales como paredes y techos atenúan la señal de información, produciendo una recepción pobre e ineficiente.

Se ubicó al control remoto en diferentes sitios del domicilio, enviando 10 comandos de control hacia el Arduino o nodo central y se verificó su funcionamiento. En la Tabla 7 se

presentan los resultados obtenidos comprobándose así que el sistema de control doméstico trabaja correctamente enviando los comandos de control recibidos desde la pantalla TFT hacia los diferentes dispositivos actuadores y que se ejecuta normalmente la acción de cada uno de ellos.

Tabla 7. Mediciones obtenidas durante la validación del prototipo [El Autor].

Distancia	Número de comandos enviados	Tiempo de respuesta (segundos)		Tiempo promedio (segundos)	
		ON	OFF	ON	OFF
1 metro	4	0.63	0.65	0.64	0.63
		0.65	0.61		
5 metros	6	0.73	0.75	0.73	0.74
		0.70	0.60		
		0.75	0.87		

Se observó que bajo estas condiciones los tiempos de respuesta para la recepción y transmisión de datos aumentaron, pero aún bajo estas condiciones el prototipo realizaba las acciones de control del sistema doméstico. A continuación en la presente tabla se resumen los datos obtenidos para distancias mayores a 5 metros.

Tabla 8. Datos obtenidos según los comandos enviados [El Autor]

Distancia	Número de comandos enviados	Tiempo de respuesta (segundos)		Sensibilidad del Receptor
		ON	OFF	
10 metros	4	0.81	0.86	-85dBm @ 1Mbps
		0.87	0.97	
30 metros	2	1.02	1.17	

De estas pruebas realizadas se concluye que el prototipo es fiable y funciona correctamente a lo que se había esperado, alcanzando latencias aceptables para la ejecución de las acciones de control.

6.3.2 Validación Económica

A continuación se detallan los costos de los elementos utilizados en el proceso de implementación del prototipo de control domótico, cabe recalcar que dichos costos varían de acuerdo al número de dispositivos que se desee controlar. En la Tabla 9 se describen los elementos necesarios con su respectivo precio para el control de cuatro dispositivos como son persianas, ventilador, luces y calefactor, obteniéndose así el costo total del sistema.

Tabla 9. Presupuesto del sistema domótico y control remoto [El Autor]

DESCRIPCIÓN	UNIDAD	CANTIDAD	VALOR UNITARIO	TOTAL
Arduino Mega 2560 R3	Unidad	5	\$ 40,00	\$ 200,00
Pantalla TFT Touch 320x240 3,2"	Unidad	1	\$ 42,00	\$ 42,00
Shield Mega TFT LCD	Unidad	1	\$ 10,00	\$ 10,00
Módulos nRF24L01	Unidad	5	\$ 4,50	\$ 22,50
Instalación, diseño e ingeniería	Global	1	\$ 300,00	\$ 300,00
Batería de litio 11.1V	Unidad	1	\$ 30,00	\$ 30,00
Cargador 12V	Unidad	4	\$ 8,00	\$ 32,00
Transformador	Unidad	1	\$ 15,00	\$ 15,00
Circuito de conmutación 4 relés	Unidad	1	\$ 8,50	\$ 8,50
Circuito de conmutación 2 relés	Unidad	1	\$ 4,50	\$ 4,50
Motor síncrono	Unidad	1	\$ 10,00	\$ 10,00
Caja para control	Unidad	1	\$ 25,00	\$ 25,00
Caja para montaje modular	Unidad	3	\$ 12,00	\$ 36,00
TOTAL				\$ 735,50

De la Tabla 9 se tiene que el costo total del sistema es menor frente a diversas soluciones que se ofrecen actualmente en el mercado ecuatoriano.

6.3.3 Validación de características físicas y eléctricas

Este parámetro está referido a la autonomía del prototipo de control domótico y las características físicas del mismo.

La parte de autonomía concierne al tiempo de duración de batería, en donde el prototipo utiliza una batería de litio que comúnmente es usada en los modernos y medianos carros *RC* (Remote Control, por sus siglas en inglés) de juguete, la cual tiene la particularidad de proporcionar una fuente de alimentación muy eficiente, a la vez que son muy ligeras y soportan altos picos de corriente. Tiene una capacidad de corriente eléctrica de 2200mAh y el voltaje es de 11,1V distribuido en tres series de celdas con una tensión de 3,7V cada una. Se estima que la pantalla TFT Touch tiene un consumo de aproximadamente 70mAh, por lo que con esta batería se otorga la corriente necesaria para su funcionamiento, ya que el tiempo promedio de operación de este tipo de batería es de 5 años considerando el consumo total de los elementos y en las peores condiciones.

Así mismo, con el fin de obtener un control remoto para el sistema domótico implementado y a su vez garantizar la movilidad al usuario se ha considerado presentar el mismo en una caja de material PVC, con dimensiones similares a productos comerciales (véase Figura 27), la misma que ha sido elaborada considerando las dimensiones tanto de la pantalla TFT, de la batería y del Arduino Mega 2560. No se ha considerado colocar un adaptador para la batería, ya que este tipo de baterías difieren de las alcalinas y requieren de un cargador especial.

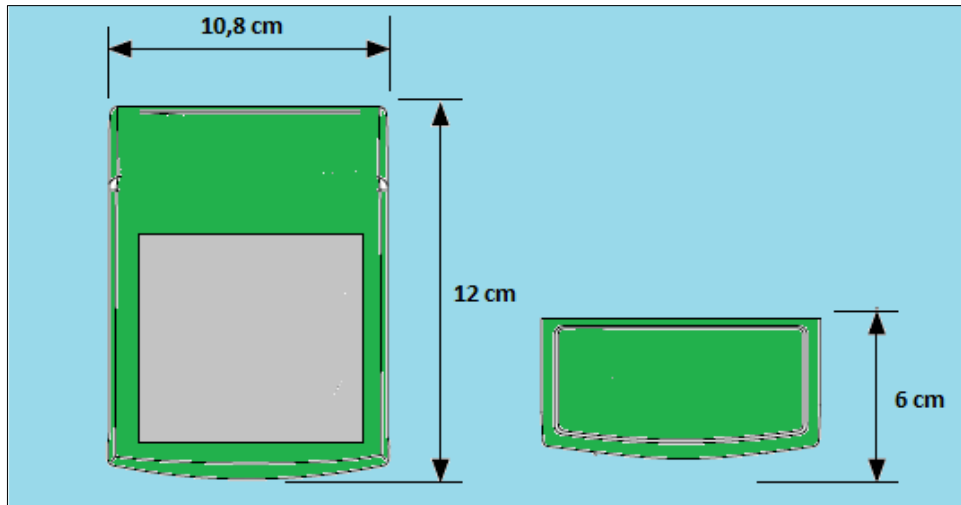


Figura 27. Datos dimensionales del prototipo [El Autor]

Una de las cualidades de este proyecto es el tiempo que se demora en la implementación total del sistema, el cual tentativamente se puede acotar a un día (8 horas laborables) máximo, debido a que el sistema está desarrollado en módulos de fácil instalación y acople para el escenario escogido (véase Figura 28). Cabe señalar que el sistema está sujeto a revisión del hogar previo al día de la instalación, debido que se tienen que conocer ciertos parámetros como el tipo de interruptores para las luminarias, y que dispositivos finales son los que se desearán controlar.

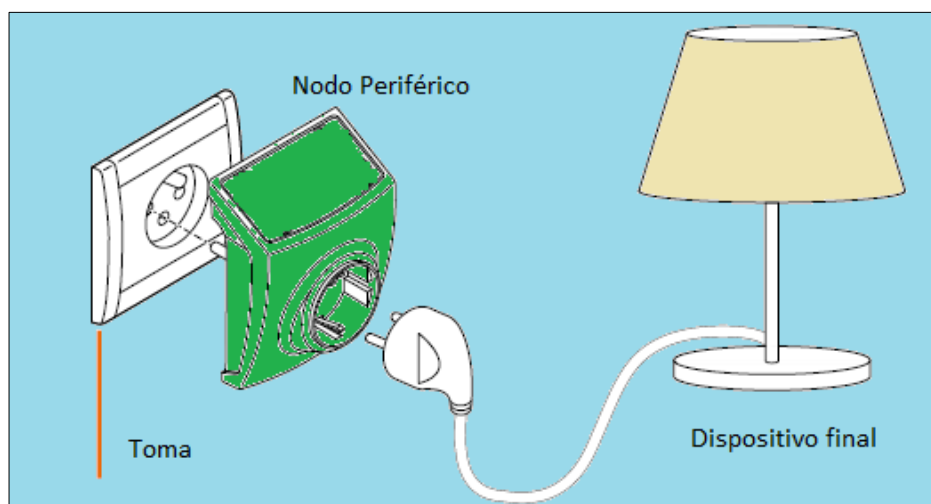


Figura 28. Montaje, instalación de nodo periférico [El Autor]

7. DISCUSIÓN

Para lograr la comunicación del prototipo controlador con la red domótica se estableció una comunicación inalámbrica punto a multipunto, constituida por los nodos receptores perteneciente a la red domótica implementada y un nodo transmisor constituido por el control remoto. Estos nodos tienen la particularidad de utilizar la plataforma de hardware libre Arduino Mega 2560 R3 y los módulos para comunicación inalámbrica nRF24L01. La idea de utilizar estos dispositivos vino a raíz de consultar en blogs y personas especializados en el tema. Así mismo, el costo de estos elementos es muy económico frente a otras soluciones que se ofrecen en el mercado actualmente, tal es el caso de los módulos Xbee que duplican el precio frente a los módulos nRF24L01. Además como se tenía planeado realizar un control inalámbrico, se decidió utilizar un transmisor de radiofrecuencia en la banda de 2.4 GHz, ya que esta es una banda no licenciada por lo que la transmisión es de uso libre. Para esto se eligió el transceptor nRF24L01 de la empresa Nordic Semiconductor, pues es un dispositivo que permite implementar redes inalámbricas WPAN orientadas al bajo consumo de energía y bajas tasas de transmisión de datos. Este dispositivo da la posibilidad de transmitir datos a 250 Kbps, 1 Mbps y 2 Mbps; y trabaja con la modulación GFSK utilizada en el estándar IEEE 802.15.

Respecto al sistema domótico se planteó como un objetivo realizar el control de cuatro variables específicas, esto es controlar:

- La apertura y cierre de una cortina motorizada,
- El encendido y apagado de las luminarias y,
- El encendido y apagado de dispositivos tales como el ventilador y calefactor ubicados en la habitación de un hogar.

Para el control remoto se decidió utilizar una interfaz gráfica que sea amigable para el usuario, por lo que se optó utilizar una pantalla LCD táctil que se adapte a la plataforma de trabajo Arduino y sea de fácil configuración, en este caso se utilizó la pantalla TFT HY-TFT320 de Elecfracks, la cual es un módulo de pantalla LCD de 320x240 píxeles de resolución con una gama de 65K colores. Como se ha venido manifestando anteriormente,

se utilizó una Shield TFT como medio de acople para la pantalla TFT y el Arduino Mega, garantizando la comunicación entre estos dos dispositivos.

Respecto a la validación del prototipo, se realizaron pruebas de mediciones con el sistema implementado, con la finalidad de obtener un tiempo promedio de retardo entre las acciones de control que se ejecuten por el control remoto según la distancia en la que se encuentre. Las primeras pruebas se realizaron considerando una distancia de 1 y 5 metros, en donde se encontraron niveles aceptables de recepción y transmisión. De acuerdo a la Tabla 7, según las distancias en que se realizaron las mediciones, se obtuvieron latencias promedio de 0.63 y 0.64 segundos para ejecutar el encendido y apagado de las distintas variables de control del sistema, por lo que se consideran aceptables tomando en cuenta que se trata de un prototipo. También, de la Tabla 8 se comprobó que se puede llegar a alcanzar distancias superiores a 30 metros pero con retardos mayores en la comunicación y el alcance se lo comprobó al realizar pruebas fuera de la vivienda, donde se obtuvo una comunicación ininterrumpida teniendo como obstáculos una pared de ladrillo, puertas y ventanas.

7.1 Ventajas del prototipo

Como la finalidad de este proyecto es brindar comodidad y bienestar al usuario dentro de su hogar, se ofrece la posibilidad de controlar una red domótica desde un dispositivo inalámbrico construido en nuestro país específicamente en la ciudad de Loja, el cual será usado como mando a distancia para controlar la iluminación, persianas y climatización del hogar, contribuyendo a la comodidad en el quehacer diario de las familias y aumentando el confort dentro del mismo.

Además, el desarrollo de tecnologías empleadas en el control domótico en nuestro país están desarrollándose paulatinamente, pero la desventaja de los productos que hay en el mercado es que son de muy alto costo, ya que todo el sistema viene pre-ensamblado en el extranjero y lo que se realiza aquí es instalarlo y realizar la configuración final. Frente a este escenario, la ventaja que ofrece este prototipo de control domótico es que el costo total para la implementación es menor frente a las soluciones comerciales que se ofrecen en el mercado (véase Tabla 9), teniendo un ahorro de aproximadamente un 53.6% (véase Anexo 6, Figura 38). Además se puede brindar servicio de garantía competitiva y soporte inmediato, como

también el proyecto puede ser mejorado en un futuro puesto que se trabaja con herramientas de hardware y software libre fáciles de programar y operar.

Igualmente dependiendo de la red domótica que se desee controlar, se pueden anexar más opciones de control en el dispositivo inalámbrico, se pueden adicionar servicios como el control de acceso, automatización y control de riego o de electrodomésticos, control de dispositivos de reproducción multimedia etc., lográndose la integración con otros sistemas mediante el Internet de las cosas (IoT, por sus siglas en inglés). Para esto se anexaría a la red elementos puntuales como son los sensores de fácil acople con la placa Arduino (sensor de temperatura DHT11, sensor de movimiento DYP-ME) y el uso de módulos que incorporen el stack de protocolos TCP/IP como por ejemplo el uso de Shield Ethernet o Wifi, dependiendo de los requerimientos del usuario.

8. CONCLUSIONES

- Se implementó el prototipo para el control de un sistema domótico, utilizando una pantalla TFT Touch Shield V2.0 como interfaz gráfica a través de la plataforma Open Source Arduino Mega 2560 R3; el mismo que está orientado al monitoreo y control inalámbrico de variables como la iluminación, ventilación y acondicionamiento de una habitación por medio de un enlace de radiofrecuencia.
- Para establecer la comunicación inalámbrica se utilizó el concepto de las redes de área personal WPAN 802.15 mediante una topología punto a multipunto, cuya implementación fue posible gracias al uso de los módulos transceiver nRF24L01 que trabajan en la frecuencia de 2.4 GHz.
- Se diseñó un algoritmo utilizado por el software Arduino, el cual permitió procesar todos los datos entregados por la pantalla TFT Touch de una manera correcta, sin dejar espacios a confusiones por parte del operador del control remoto o del sistema en sí.
- De las pruebas realizadas durante la validación, se tiene que el prototipo alcanza latencias aceptables para la ejecución de las acciones de control. Además, el funcionamiento del prototipo fue normal ya que en todos los lugares donde se ubicó el control remoto se mantuvo la comunicación.
- Se logró obtener un producto que brinda un comportamiento similar a las soluciones producidas por empresas, cuyo costo total es menor frente a los precios de las soluciones domóticas ofrecidas en el mercado.
- Una desventaja del prototipo de control domótico es que está destinado solamente para controlar un sistema domótico que esté implementado con Arduino y si se desea controlar otras variables adicionales a las ya programadas, se debe recurrir a la reprogramación de las placas transmisora y receptora, así como la adición de nuevas circuiterías en la habitación.

9. RECOMENDACIONES

- Debido a que el módulo nRF24L01 posee 125 canales con 1MHz de ancho de banda, se recomienda utilizar aquellos que no produzcan interferencia con los canales utilizados por el comúnmente utilizado Wifi u 802.11. Como norma, es preferible configurar a los módulos transmisor y receptor con el mismo canal de radiofrecuencia, en este caso se recomienda seleccionar el canal 90.
- Se recomienda aplicar, tanto a la pantalla táctil como a los módulos Arduino, una tensión que oscile entre los 5 V a 12 V y unos 600 mA como mínimo, dependiendo del consumo del circuito, ya que los valores superiores e inferiores a estos no permitirán que las placas Arduino y la pantalla táctil trabajen o incluso puedan encenderse.
- Al momento de realizar la instalación del nodo periférico para la iluminación, se recomienda realizar el correcto etiquetado de los cables de conexión y así evitar la confusión al momento de conectar las respectivas fases en el interruptor, ya que se produciría un cortocircuito e incluso la quema de los circuitos utilizados.
- Debido a que las baterías LiPo no se pueden recargar con cargadores convencionales ya que poseen un funcionamiento distinto a las baterías alcalinas, se debe usar un cargador especial para éste tipo de baterías, por lo que se recomienda el uso del cargador USB/DC con MAX1555 disponible en cualquier tienda de suministros electrónicos o de Arduino.
- En lo posible se recomienda emplear una pantalla táctil capacitiva ya que es más sensible al tacto que la pantalla resistiva TFT LCD empleada en este proyecto de titulación. Además se recomienda optimizar el prototipo con una pantalla táctil de mayor dimensión que permita una mejor visualización de la interfaz gráfica.

- Se recomienda trabajar en la creación de un sistema modular propio que reemplace totalmente la utilización de Arduino, garantizando así la obtención de un prototipo menos tosco y que permita la mayor movilidad al usuario final.

10. BIBLIOGRAFÍA

- [1] Pilar Serra, “Domótica,” 2010. [Online]. Available: <http://www.uv.es/mpisea/54656d615f385f446f6dc3b374696361.pdf>. [Accessed: 23-Aug-2015].
- [2] José Huidrobo, *Manual de domótica*, vol. 1, 1 vols. Madrid, España: Creaciones Copyright, 2010.
- [3] Ingrid Constantino, “Domótica e Inmótica: viviendas y edificios inteligentes,” Jul-2011. [Online]. Available: https://www.academia.edu/7301074/1.5.2.-_DESVENTAJAS_DE_UN_SISTEMA_DOMOTICO_26. [Accessed: 23-Aug-2015].
- [4] Leopoldo Molina, *Instalaciones domóticas*. Madrid, España: McGraw-Hill, 2010.
- [5] Ramón Pallás Areny, “Sensores,” in *Sensores y acondicionadores de señal*, Cuarta Edición., México D.F: Alfaomega Grupo Editor, 2007.
- [6] Alvaro Maldonado, “Diseño e instalación de la tecnología domótica en un bloque de viviendas,” 2010. [Online]. Available: http://e-archivo.uc3m.es/bitstream/handle/10016/10991/PFC_Alvaro_Maldonado_Esteban.pdf?sequence=4. [Accessed: 23-Aug-2015].
- [7] José Alejandro, “Sistema de control domótico de una vivienda,” 2009. [Online]. Available: <http://deeea.urv.cat/public/PROPOSTES/pub/pdf/1343pub.pdf>. [Accessed: 23-Aug-2015].
- [8] Leandro Pazmiño, “Diseño y construcción de un prototipo inalámbrico RF para el monitoreo de la seguridad residencial de forma remota empleando la Internet,” Escuela Politécnica Nacional, Quito Ecuador, 2010.
- [9] Gustavo Valle, “Sistema domótico con tecnología Eibkonnex para la automatización de servicios, confort y seguridad en la empresa SISTELDATA S.A,” Universidad Técnica de Ambato, 2012.
- [10] José Raya, Laura Raya, and Miguel Martínez, “Redes locales inalámbricas,” in *Redes locales. Instalación y configuración básicas*, Primera Edición., México D.F: Alfaomega Grupo Editor, p. 412.
- [11] Alberto Escudero, “Estándares en tecnologías inalámbricas.” [Online]. Available: http://www.itrainonline.org/itrainonline/mmtk/wireless_es/files/02_es_estandares-inalambricos_guia_v02.pdf. [Accessed: 29-Aug-2015].

- [12] Andrew Tanenbaum, *Redes de computadoras*. México: Pearson Prentice Hall, 2003.
- [13] Lis Rivera, “Tecnologías inalámbricas.” Catarina. México, 2011.
- [14] Tatiana Reyes, “Diseño e implementación de un prototipo de sistema de gestión inalámbrico de atención al cliente,” Universidad Nacional de Loja, Loja, Ecuador, 2015.
- [15] Mateus Streit, “Projeto de uma rede de comunicação sem fio baseada no transceptor nRF24L01 voltada para sistemas de automação predial,” Universidade Federal do Rio Grande do Sul, Brasil, 2014.
- [16] Nordic Semiconductor, “Product Specification nRF24L01,” Jul-2007. [Online]. Available: http://www.seeedstudio.com/document/pdf/DS_nRF24L01.pdf. [Accessed: 19-Sep-2015].
- [17] Nordic Semiconductor, “Pinout de la placa nRF24L01 Nordic Semiconductor,” 2007. [Online]. Available: <http://bibing.us.es/proyectos/abreproy/12159/fichero/3.pdf>. [Accessed: 19-Sep-2015].
- [18] Manfredy Chugñay and Germán Llamuca, “Diseño y construcción de dos módulos con pantallas táctiles para el laboratorio de control industrial de la facultad de mecánica,” Escuela Superior Politécnica de Chimborazo, Riobamba, 2010.
- [19] Erick Cafferata and Jorge Tello, “Diseño e implementación de una red inalámbrica de sensado orientada a domótica,” Universidad Peruana de ciencias aplicadas, Lima, Perú, 2013.
- [20] Oscar Torrente, *Arduino. Curso práctico de formación*, Primera Edición. México D.F: Alfaomega Grupo Editor.
- [21] Sergio Sáez and Vicente Atienza, “Diseño de un sistema de control domótico basado en la plataforma Arduino,” Universidad Politécnica de Valencia, España, 2012.
- [22] Rafael Herrador, “Guía de usuario de Arduino.” 2009.
- [23] Alberto Fuambuena, “Pantallas resisitivas,” Jul-2011. [Online]. Available: http://personales.alumno.upv.es/alrafua/asignaturas/SES/Perifericos/Interfaces_humanas/pantallas_tactiles.html. [Accessed: 13-Sep-2015].
- [24] Laura Lucía, “Tecnología,” 2011. [Online]. Available: <http://lauralucia10.blogspot.es/tags/tecnologia/>. [Accessed: 14-Sep-2015].
- [25] Carlos Bastidas, “Diseño e implementación de un dispositivo para aplicar multilíneas de adhesivo a pañales desechables,” Escuela Politécnica Nacional, Quito Ecuador, 2009.

[26] Future Electronics, “Datasheet TFT HY-TFT320.” .

[27] Nordic Semiconductor, “Datasheet nRF24L01.” .

11. ANEXOS

ANEXO 1: Algoritmos de programación

ANEXO 1.1: Algoritmo para la pantalla TFT Touch

```

//:.....//
// PROGRAMA PANTALLA TFT TOUCH V2.0      //
//:.....//

//Declaración de librerías
#include <UTFT.h>
#include <UTouch.h>
#include <UTFT_Buttons.h>

//Declaración de pines
int PIN_LED0=9;
int PIN_LED1=10;
int PIN_LED2=11;
int PIN_LED3=12;
int PIN_LED4=13;

// Declaración de las fuentes a usarse
extern uint8_t SmallFont[];
extern uint8_t BigFont[];
extern uint8_t Dingbats1_XL[];

// Configuración de pines de la pantalla con libreria UTFT
UTFT      myGLCD(ITDB32S,38,39,40,41);

// Configuración de pines del táctil y la librería UTouch
UTouch    myTouch(6,5,4,3,2);

// Declaración de botones para crear
UTFT_Buttons myButtons(&myGLCD, &myTouch);
//-----

//Inicialización de pantalla
void setup()
{
  myGLCD.InitLCD();
  myGLCD.clrScr();
  myGLCD.setFont(SmallFont);

```

```

myTouch.InitTouch();
myTouch.setPrecision(PREC_MEDIUM);

myButtons.setTextFont(SmallFont); // Letra pequeña para botones
myButtons.setSymbolFont(Dingbats1_XL);

//Definimos como salidas los pines
pinMode(PIN_LED0,OUTPUT);
pinMode(PIN_LED1,OUTPUT);
pinMode(PIN_LED2,OUTPUT);
pinMode(PIN_LED3,OUTPUT);
pinMode(PIN_LED4,OUTPUT);

//Todos los pines arrancan en bajo
digitalWrite(PIN_LED0,LOW);
digitalWrite(PIN_LED1,LOW);
digitalWrite(PIN_LED2,LOW);
digitalWrite(PIN_LED3,LOW);
digitalWrite(PIN_LED4,LOW);

Serial.begin(57600); //Inicializa Serial a 57600 baudios
Serial.println("Comenzo control domotico");
}

void loop()
{
  int but1, but2, but3, but4, but5, but6, but7, but8, but9, but10,
  boolean default_colors = true;

  but1 = myButtons.addButton( 10, 30, 70, 30, "abrir");
  but2 = myButtons.addButton( 90, 30, 70, 30, "cerrar");
  but3 = myButtons.addButton( 10, 90, 70, 30, "calentar");
  but4 = myButtons.addButton( 90, 90, 70, 30, "OFF");
  but5 = myButtons.addButton( 190, 30, 50, 30, "ON");
  but6 = myButtons.addButton( 250, 30, 50, 30, "OFF");
  but7 = myButtons.addButton( 190, 90, 50, 30, "ON");
  but8 = myButtons.addButton( 250, 90, 50, 30, "OFF");
  but9 = myButtons.addButton( 10, 150, 70, 30, "ventilar");
  but10 = myButtons.addButton( 90, 150, 70, 30, "OFF");

  butY = myButtons.addButton( 10, 195, 40, 40, "I", BUTTON_SYMBOL);
  myButtons.drawButtons();

  //HECHO POR KAREN LOAIZA
  myGLCD.setBackgroundColor(VGA_YELLOW); //color de fondo
  myGLCD.setColor(VGA_GRAY); //color de letra
  //myGLCD.print("Ing. Karen Loaiza", 180, 225);
  //myGLCD.print("Karen Loaiza", 220, 225);

```

```

myGLCD.setBackgroundColor(VGA_BLACK);
myGLCD.setColor(VGA_WHITE);
myGLCD.print("Usted selecciono:", 90, 195);
//myGLCD.print("Usted esta:", 90, 205); //ejemplo, como no se
//le puso antes setcolor, entonces
//toma el color x defecto (blanco)

//Titulo de cada objeto a realizar la acción
myGLCD.setColor(VGA_AQUA); // Color d letra para "persianas"
myGLCD.print("Persiana:", 10, 15);
myGLCD.print("Luz1 sala:", 190, 15);
myGLCD.print("Calefactor:", 10, 75);
myGLCD.print("Luz2 sala:", 190, 75);
myGLCD.print("Ventilador:", 10, 135);

//Nombramos acciones q están por defecto
myGLCD.setBackgroundColor(VGA_RED); //Color de fondo para c/letra
//por defecto
myGLCD.print("Estado Actual: ", 175, 125);
myGLCD.print("Persiana: cerrada", 175, 138);
myGLCD.print("Calefactor: off", 175, 148);
myGLCD.print("Ventilador: off", 175, 158);
myGLCD.print("Luz1 sala: off", 175, 168);
myGLCD.print("Luz2 sala: off", 175, 178);

myGLCD.print("Ninguno ", 90, 205); //En caso de
//seleccionar mal en la pantalla

//Desabilitamos los botones cuya acción está por defecto
myButtons.disableButton(but2, true);
myButtons.disableButton(but4, true);
myButtons.disableButton(but6, true);
myButtons.disableButton(but8, true);
myButtons.disableButton(but10, true);

while(1)
{
  if (myTouch.dataAvailable() == true)
  {
    pressed_button = myButtons.checkButtons();
    if (pressed_button==butY)
    {
      if (default_colors)
      {
        myButtons.setButtonColors(VGA_YELLOW, VGA_RED, VGA_OLIVE,
//1color=letra; 2=cambiar letra; 3=contorno cuadro; 4=?; 5=fondo
        myButtons.relabelButton(butY, "_");
        myButtons.drawButtons();
        default_colors=false;
      }
    }
  }
}

```

```

else
{
  myButtons.setButtonColors(VGA_WHITE, VGA_GRAY, VGA_WHITE, '
  myButtons.relabelButton(butY, "-");
  myButtons.drawButtons();
  default_colors=true;
}
}
else if (pressed_button==but1)
{
  myGLCD.print("Abrir persiana    ", 90, 205); //Todos los
  //tamaños d palabra acoplarlos a 19 (tamaño mas largo)
  myGLCD.print("Persiana: abierta", 175, 138);
  myButtons.disableButton(but1, true);
  myButtons.enableButton(but2, true);
  digitalWrite(PIN_LED4,HIGH);
  Serial.println("Abrir persiana");
}
else if (pressed_button==but2)
{
  myGLCD.print("Cerrar persiana   ", 90, 205);
  myGLCD.print("Persiana: cerrada", 175, 138);
  myButtons.disableButton(but2, true);
  myButtons.enableButton(but1, true);
  digitalWrite(PIN_LED4,LOW);
  Serial.println("Cerrar persiana");
}

else if (pressed_button==but3)
{
  myGLCD.print("Encender calefactor", 90, 205);
  myGLCD.print("Calefactor: on ", 175, 148);
  myButtons.disableButton(but3, true);
  myButtons.enableButton(but4, true);
  digitalWrite(PIN_LED3,HIGH);
  Serial.println("Encender calefactor");
}
else if (pressed_button==but4)
{
  myGLCD.print("Apagar calefactor  ", 90, 205);
  myGLCD.print("Calefactor: off", 175, 148);
  myButtons.disableButton(but4, true);
  myButtons.enableButton(but3, true);
  digitalWrite(PIN_LED3,LOW);
  Serial.println("Apagar calefactor");
}
}

```

```

else if (pressed_button==but5)
{
myGLCD.print("Encender luz1 sala ", 90, 205);
myGLCD.print("Luz1 sala: on ", 175, 168);
myButtons.disableButton(but5, true);
myButtons.enableButton(but6, true);
digitalWrite(PIN_LED2,HIGH);
Serial.println("Encender luz1");
}
else if (pressed_button==but6)
{
myGLCD.print("Apagar luz1 sala  ", 90, 205);
myGLCD.print("Luz1 sala: off", 175, 168);
myButtons.disableButton(but6, true);
myButtons.enableButton(but5, true);
digitalWrite(PIN_LED2,LOW);
Serial.println("Apagar luz1");
}
else if (pressed_button==but7)
{
myGLCD.print("Encender luz2 sala ", 90, 205);
myGLCD.print("Luz2 sala: on ", 175, 178);
myButtons.disableButton(but7, true);
myButtons.enableButton(but8, true);
digitalWrite(PIN_LED1,HIGH);
Serial.println("Encender luz2");
}
else if (pressed_button==but8)
{
myGLCD.print("Apagar luz2 sala  ", 90, 205);
myGLCD.print("Luz2 sala: off", 175, 178);
myButtons.disableButton(but8, true);
myButtons.enableButton(but7, true);
digitalWrite(PIN_LED1,LOW);
Serial.println("Apagar luz2");
}

else if (pressed_button==but9)
{
myGLCD.print("Encender ventilador", 90, 205);
myGLCD.print("Ventilador: on ", 175, 158);
myButtons.disableButton(but9, true);
myButtons.enableButton(but10, true);
digitalWrite(PIN_LED0,HIGH);
Serial.println("Encender ventilador");
}

```

```

else if (pressed_button==but10)
{
myGLCD.print("Apagar ventilador ", 90, 205);
myGLCD.print("Ventilador: off", 175, 158);
myButtons.disableButton(but10, true);
myButtons.enableButton(but9, true);
digitalWrite(PIN_LED0,LOW);
Serial.println("Apagar ventilador");
}
if (pressed_button==1)
myGLCD.print("Ninguno", 90, 205);
}
}
}

```

ANEXO 1.2: Algoritmo para los módulos nRF24L01

```

//:::::::::::::::::::::::::::::::::::::::::::::::::::::::::://
//      PROGRAMA PARA EL EMISOR nRF24L01      //
//:::::::::::::::::::::::::::::::::::::::::::::::::::::::::://

//Definición de librerías
#include <SPI.h>
#include <Mirf.h>
#include <nRF24L01.h>
#include <MirfHardwareSpiDriver.h>

char cadena[10]; //Cadena de caracteres para enviar
String inputstring = ""; //Cadena recibida desde el PC
boolean input_stringcomplete = false; //Cadena recibida
//completamente desde el PC

//Definición de pines del Arduino Mega
int PIN_LED0=6;
int PIN_LED1=5;
int PIN_LED2=4;
int PIN_LED3=3;
int PIN_LED4=2;

//Configuración del módulo
void setup()
{
Mirf.spi = <MirfHardwareSpi; //Inicializa controlador de SPI
Mirf.init(); //Inicializa PINES SPI del nRF24L01
Mirf.payload = 1; //Define el payload,
//cantidad de bytes que transmitirá, o sea 1Byte

```

```

Mirf.channel = 90; //Seleccion de canal, 90 de los 125
Mirf.config();
Mirf.configRegister(RF_SETUP,0x06); // 1MHz, Ancho de banda 1Mbps

Mirf.setTADDR((byte *)"TX_01"); // Seleccion de Direccion
Serial.begin(57600); //Inicializa Serial a 57600 baudios
Serial.println("Iniciando ...");

//Definimos como salidas los pines
pinMode(31,OUTPUT);
pinMode(33,OUTPUT);
pinMode(35,OUTPUT);
pinMode(37,OUTPUT);
pinMode(39,OUTPUT);

//Todos los pines arrancan en bajo
digitalWrite(PIN_LED0,LOW);
digitalWrite(PIN_LED1,LOW);
digitalWrite(PIN_LED2,LOW);
digitalWrite(PIN_LED3,LOW);
digitalWrite(PIN_LED4,LOW);
}

// ENVIO DE CADENA
void transmit(const char *string)
{
  byte c;
  for( int i=0 ; cadena[i]!=0x00 ; i++){
    c = cadena[i];
    Mirf.send(&c);
    while( Mirf.isSending() ) ;
  }
}

// ENVIO DE CR/LF
void transmitlf(void)
{
  byte c;
  c = '\r';
  Mirf.send(&c);
  while( Mirf.isSending() ) ;
  c = '\n';
  Mirf.send(&c);
  while( Mirf.isSending() ) ;
}

//Recepcion de caracteres desde serial
void serialEvent() {
  char inchar = (char)Serial.read();

```



```

// Concatena caracter ingresado
inputstring += inchar;

// Fin de la cadena, recibido <CR>
if(inchar == '\r') {input_stringcomplete = true;}
}

void loop()
{
// Condicion para enviar datos
if (input_stringcomplete) {
inputstring.toCharArray(cadena, 10); //Conversion de String a
//arreglo de caracteres
transmit(cadena); //Transmite cadena ingresada
transmitlf(); //Transmite CR/LF

Serial.println("Tx: ");
Serial.println(inputstring);
input_stringcomplete = false;
inputstring = "";
}

byte c;
int ascii;

//Rx señales por cable y tx por modulo RF
if (digitalRead(PIN_LED3)==HIGH){
// delay(500);
c = 'c';
Serial.println("Tx: ");
Serial.println("c");
Mirf.send(&c);
while( Mirf.isSending() );
c = '\n';
}
else if (digitalRead(PIN_LED3)==LOW){
// delay(500);
c = 'd';
Serial.println("Tx: ");
Serial.println("d");
Mirf.send(&c);
while( Mirf.isSending() );
c = '\n';
}

if (digitalRead(PIN_LED4)==HIGH){
// delay(500);

```

```

    c = 'a';
    Serial.println("Tx: ");
    Serial.println("a");
    Mirf.send(&c);
    while( Mirf.isSending() );
    c = '\n';
}
else if (digitalRead(PIN_LED4)==LOW){
//  delay(500);
    c = 'b';
    Serial.println("Tx: ");
    Serial.println("b");
    Mirf.send(&c);
    while( Mirf.isSending() );
    c = '\n';
}

if (digitalRead(PIN_LED2)==HIGH){
//  delay(500);
    c = 'e';
    Serial.println("Tx: ");
    Serial.println("e");
    Mirf.send(&c);
    while( Mirf.isSending() );
    c = '\n';
}
else if (digitalRead(PIN_LED2)==LOW){
//  delay(500);
    c = 'f';
    Serial.println("Tx: ");
    Serial.println("f");
    Mirf.send(&c);
    while( Mirf.isSending() );
    c = '\n';
}

if (digitalRead(PIN_LED1)==HIGH){
//  delay(500);
    c = 'g';
    Serial.println("Tx: ");
    Serial.println("g");
    Mirf.send(&c);
    while( Mirf.isSending() );
    c = '\n';
}
}

```

```

else if (digitalRead(PIN_LED1)==LOW){
  // delay(500);
  c = 'h';
  Serial.println("Tx: ");
  Serial.println("h");
  Mirf.send(&c);
  while( Mirf.isSending() );
  c = '\n';
}
if (digitalRead(PIN_LED0)==HIGH){
  // delay(500);
  c = 'm';
  Serial.println("Tx: ");
  Serial.println("m");
  Mirf.send(&c);
  while( Mirf.isSending() );
  c = '\n';
}

else if (digitalRead(PIN_LED0)==LOW){
  // delay(500);
  c = 'n';
  Serial.println("Tx: ");
  Serial.println("n");
  Mirf.send(&c);
  while( Mirf.isSending() );
  c = '\n';
}

//Condicion de Recepcion de datos
if( Mirf.dataReady() )
{
  Mirf.getData(&c); //Funcion de libreria MRF que recibe Byte
  ascii=c; //
  char caracter= (char)ascii; //Convertir de int a char

  Serial.print(caracter);

}
}

```

ANEXO 1.3: Algoritmo para el Arduino Mega 2560 Central

```
//:.....//
// PROGRAMA PARA EL RECEPTOR nRF24L01 DEL //
// ARDUINO CONTROLADOR DEL SISTEMA //
//:.....//

//Definimos las librerías
#include <SPI.h>
#include <Mirf.h>
#include <nRF24L01.h>
#include <MirfHardwareSpiDriver.h>

char cadena[10]; //Cadena de caracteres para enviar
String inputstring = ""; //Cadena recibida desde el PC
boolean input_stringcomplete = false; //Cadena recibida
//completamente desde el PC

//Definición de pines del Arduino Mega
int PIN_LED4=31;
int PIN_LED3=33;
int PIN_LED2=35;
int PIN_LED1=37;
int PIN_LED0=39;
int PIN_LED5=41;

//Configuración del módulo
void setup()
{
  Mirf.spi = &MirfHardwareSpi; //Inicializa controlador de SPI
  Mirf.init(); //Inicializa PINES SPI del nRF24L01
  Mirf.payload = 1; //Define el payload,
//cantidad de bytes que recibirá, o sea 1Byte
  Mirf.channel = 90; //Selección de canal, 90 de los 125
  Mirf.config();
  Mirf.configRegister(RF_SETUP,0x06); // 1MHz, Ancho de banda 1Mbps

  Mirf.setTADDR((byte *)"TX_01"); // Selección de Dirección
  Serial.begin(57600); //Inicializa Serial a 57600 baudios
  Serial.println("Iniciando ...");

// Definimos como salidas los pines
pinMode(PIN_LED0,OUTPUT);
pinMode(PIN_LED1,OUTPUT);
pinMode(PIN_LED2,OUTPUT);
pinMode(PIN_LED3,OUTPUT);
pinMode(PIN_LED4,OUTPUT);
pinMode(PIN_LED5,OUTPUT);
}
```

```

// Todos los pines arrancan en bajo
digitalWrite(PIN_LED0,LOW);
digitalWrite(PIN_LED1,LOW);
digitalWrite(PIN_LED2,LOW);
digitalWrite(PIN_LED3,LOW);
digitalWrite(PIN_LED4,LOW);
digitalWrite(PIN_LED5,LOW);
}

// ENVIO DE CADENA
void transmit(const char *string)
{
  byte c;
  for( int i=0 ; cadena[i]!=0x00 ; i++ ){
    c = cadena[i];
    Mirf.send(&c);
    while( Mirf.isSending() ) ;
  }
}

// ENVIO DE CR/LF
void transmitlf(void)
{
  byte c;
  c = '\r';
  Mirf.send(&c);
  while( Mirf.isSending() ) ;
  c = '\n';
  Mirf.send(&c);
  while( Mirf.isSending() ) ;
}

//Recepcion de caracteres desde serial
void serialEvent() {
  char inchar = (char)Serial.read();

  // Concatena caracter ingresado
  inputstring += inchar;

  // Fin de la cadena, recibido <CR>
  if(inchar == '\r') {input_stringcomplete = true;}
}

void loop()
{
  // Condicion para enviar datos
  if (input_stringcomplete) {
    inputstring.toCharArray(cadena, 10); //Conversion de String a
                                         //arreglo de caracteres
    transmit(cadena); //Recibe cadena ingresada
    transmitlf(); //Recibe CR/LF
  }
}

```

```

Serial.println("Central: ");
Serial.println(inputstring);
input_stringcomplete = false;
inputstring = "";
}

byte c;
int ascii;

//Condicion de Recepcion de datos
if( Mirf.dataReady() )
{
Mirf.getData(&c); //Funcion de libreria MIRF que recibe Byte
ascii=c; //
char caracter= (char)ascii; //Casting de char para convertir de
//int a char

//Recepción de datos a través de módulo RF y utilizar pines
if (ascii == 'a') {
digitalWrite(PIN_LED4,HIGH);
Serial.println("pa");
}
else if (ascii == 'b'){
digitalWrite(PIN_LED4,LOW);
Serial.println("pb");
}
else if (ascii == 'c') {
digitalWrite(PIN_LED3,HIGH);
Serial.println("qa");
}
else if (ascii == 'd'){
digitalWrite(PIN_LED3,LOW);
Serial.println("qb");
}
else if (ascii == 'e') {
digitalWrite(PIN_LED2,HIGH);
Serial.println("ra");
}
else if (ascii == 'f'){
digitalWrite(PIN_LED2,LOW);
Serial.println("rb");
}
else if (ascii == 'g') {
digitalWrite(PIN_LED1,HIGH);
Serial.println("sa");
}
else if (ascii == 'h'){
digitalWrite(PIN_LED1,LOW);
Serial.println("sb");
}
}

```

```
else if (ascii == 'm') {
    digitalWrite(PIN_LED0,HIGH);
    Serial.println("ta");
    delay(2000);
    digitalWrite(PIN_LED0,LOW);
}
else if (ascii == 'n'){
    digitalWrite(PIN_LED5,HIGH);
    Serial.println("tb");
    delay(2000);
    digitalWrite(PIN_LED5,LOW);
}
}
}
```

ANEXO 2: Especificaciones técnicas de los dispositivos

ANEXO 2.1: Pantalla TFT LCD 3.2' HY-TFT320 V2.0

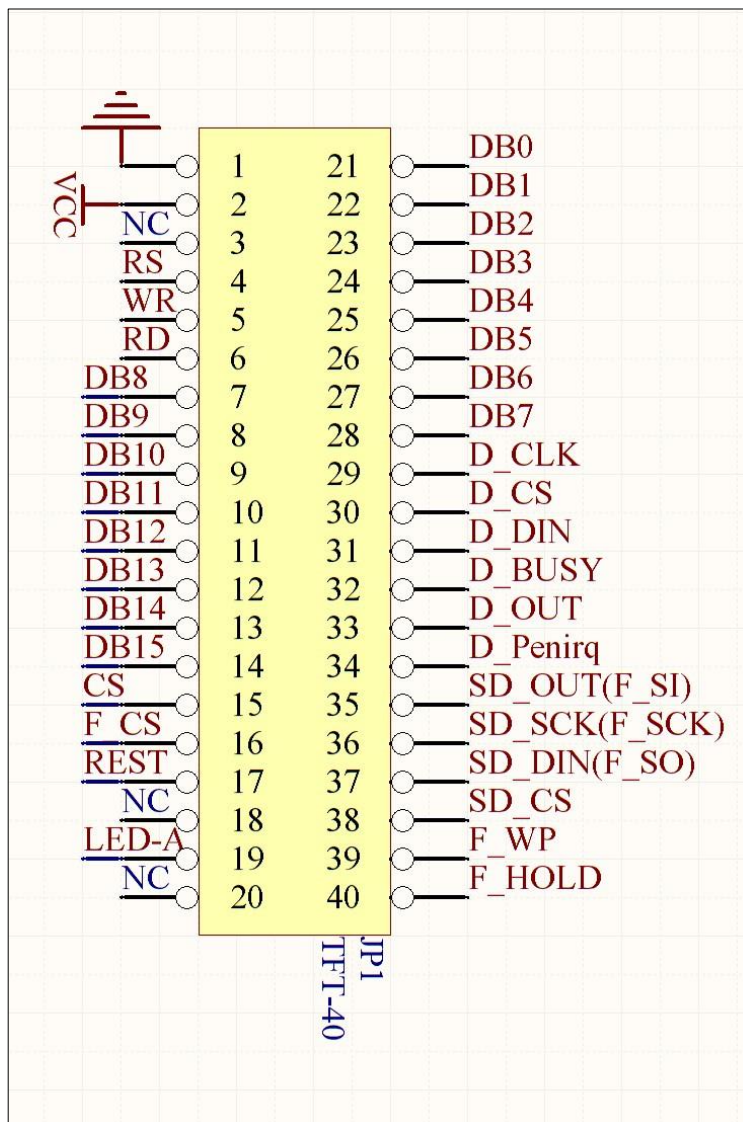


Figura 29. Diagrama de pines de la pantalla TFT [26].

Tabla 10. Conexión de los pines de la pantalla TFT [El Autor]

Pines de Pantalla	Pines Arduino Mega
T_IRQ	D2
T_DOUT	D3
T_DIN	D4
T_CS	D5
T_CLK	D6
DB8	D22
DB9	D23
DB10	D24
DB11	D25
DB12	D26
DB13	D27
DB14	D28
DB15	D29
DB7	D30
DB6	D31
DB5	D32
DB4	D33
DB3	D34
DB2	D35
DB1	D36
DB0	D37
RS	D38
WR	D39
CS	D40
RST	D41
SD_MISO	D50
SD_MOSI	D51

SD_SCK	D52
SD_NSS	D53

- LCD Features
 - Monitor Type: TFT LCD
 - Brightness: 250 cd/m2
 - Back-light tube: LED
 - Back light life-span (on the average): 50,000 hours
 - Contrast: 500:1
 - Visual angle (L/R/T/B): 60/60/50/55
- Touch screen features
 - Touch Screen type : 4wires resistive
 - Touch Mode : Continuous
 - Transmittance : 85%
 - Touch Frequency : > 1000000 times
- Power Supply
 - Input Voltage: 5V
- Enviromental Details
 - Operating Temperature: -20°C~70°C
 - Relative Humidity: 10%-90%@40°C, No condensation
 - Vibration Test: 10 to 25Hz (X,Y,Z direction 2G, 30mins)
 - ESD Test: ±8KV

Figura 30. Datos técnicos de la pantalla TFT

1.1 Features

Features of the nRF24L01+ include:

- Radio
 - ▶ Worldwide 2.4GHz ISM band operation
 - ▶ 126 RF channels
 - ▶ Common RX and TX interface
 - ▶ GFSK modulation
 - ▶ 250kbps, 1 and 2Mbps air data rate
 - ▶ 1MHz non-overlapping channel spacing at 1Mbps
 - ▶ 2MHz non-overlapping channel spacing at 2Mbps
- Transmitter
 - ▶ Programmable output power: 0, -6, -12 or -18dBm
 - ▶ 11.3mA at 0dBm output power
- Receiver
 - ▶ Fast AGC for improved dynamic range
 - ▶ Integrated channel filters
 - ▶ 13.5mA at 2Mbps
 - ▶ -82dBm sensitivity at 2Mbps
 - ▶ -85dBm sensitivity at 1Mbps
 - ▶ -94dBm sensitivity at 250kbps
- RF Synthesizer
 - ▶ Fully integrated synthesizer
 - ▶ No external loop filter, VCO varactor diode or resonator
 - ▶ Accepts low cost ± 60 ppm 16MHz crystal
- Enhanced ShockBurst™
 - ▶ 1 to 32 bytes dynamic payload length
 - ▶ Automatic packet handling
 - ▶ Auto packet transaction handling
 - ▶ 6 data pipe MultiCeiver™ for 1:6 star networks
- Power Management
 - ▶ Integrated voltage regulator
 - ▶ 1.9 to 3.6V supply range
 - ▶ Idle modes with fast start-up times for advanced power management
 - ▶ 26 μ A Standby-I mode, 900nA power down mode
 - ▶ Max 1.5ms start-up from power down mode
 - ▶ Max 130 μ s start-up from standby-I mode
- Host Interface
 - ▶ 4-pin hardware SPI
 - ▶ Max 10Mbps
 - ▶ 3 separate 32 bytes TX and RX FIFOs
 - ▶ 5V tolerant inputs
- Compact 20-pin 4x4mm QFN package

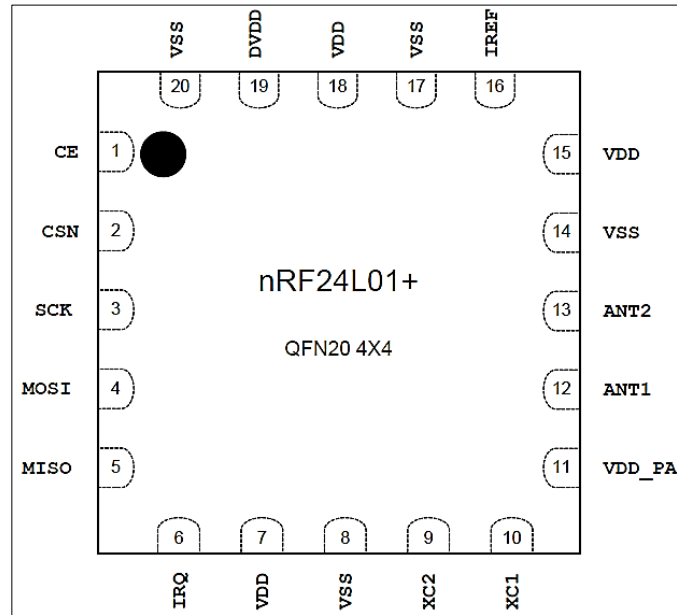


Figura 31. Diagrama de pines del módulo nRF24L01 [27]

Tabla 11. Condiciones de operación del módulo nRF24L01 [27]

Operating conditions	Minimum	Maximum	Units
Supply voltages			
VDD	-0.3	3.6	V
VSS		0	V
Input voltage			
V_I	-0.3	5.25	V
Output voltage			
V_O	VSS to VDD	VSS to VDD	
Total Power Dissipation			
P_D ($T_A=85^\circ\text{C}$)		60	mW
Temperatures			
Operating Temperature	-40	+85	$^\circ\text{C}$
Storage Temperature	-40	+125	$^\circ\text{C}$

Tabla 12. Especificaciones del nRF24L01 [27]

Supply voltage	3.0~3.6	V
Maximum output power	0	dBm
TX current consumption(Peak value)	95	mA
RX current consumption(Peak value)	15	mA
Operation temperatures	-20~70	°C
Sensitivity at 1Mbps	-85	dBm
Power down current	900	nA
Antenna peak gain	2	dBi
Coverage range(open space)	10~100	m

Tabla 13. Funciones de los pines del nRF24L01 [27]

Pin	Name	Pin function	Description
1	CE	Digital Input	Chip Enable Activates RX or TX mode
2	CSN	Digital Input	SPI Chip Select
3	SCK	Digital Input	SPI Clock
4	MOSI	Digital Input	SPI Slave Data Input
5	MISO	Digital Output	SPI Slave Data Output, with tri-state option
6	IRQ	Digital Output	Maskable interrupt pin. Active low
7	VDD	Power	Power Supply (+1.9V - +3.6V DC)
8	VSS	Power	Ground (0V)
9	XC2	Analog Output	Crystal Pin 2
10	XC1	Analog Input	Crystal Pin 1
11	VDD_PA	Power Output	Power Supply Output (+1.8V) for the internal nRF24L01+ Power Amplifier. Must be connected to ANT1 and ANT2
12	ANT1	RF	Antenna interface 1
13	ANT2	RF	Antenna interface 2
14	VSS	Power	Ground (0V)
15	VDD	Power	Power Supply (+1.9V - +3.6V DC)
16	IREF	Analog Input	Reference current. Connect a 22kΩ resistor to ground.
17	VSS	Power	Ground (0V)
18	VDD	Power	Power Supply (+1.9V - +3.6V DC)
19	DVDD	Power Output	Internal digital supply output for de-coupling purposes.
20	VSS	Power	Ground (0V)

Tabla 14. Condiciones en radiofrecuencia del módulo nRF24L01 [27]

Symbol	Parameter (condition)	Notes	Min.	Typ.	Max.	Units
f_{OP}	Operating frequency	a	2400		2525	MHz
PLL_{res}	PLL Programming resolution			1		MHz
f_{XTAL}	Crystal frequency			16		MHz
Δf_{250}	Frequency deviation @ 250kbps			± 160		kHz
Δf_{1M}	Frequency deviation @ 1Mbps			± 160		kHz
Δf_{2M}	Frequency deviation @ 2Mbps			± 320		kHz
R_{GFSK}	Air Data rate	b	250		2000	kbps
$F_{CHANNEL\ 1M}$	Non-overlapping channel spacing @ 250kbps/1Mbps	c		1		MHz
$F_{CHANNEL\ 2M}$	Non-overlapping channel spacing @ 2Mbps	c		2		MHz

Tabla 15. Operación del nRF24L01 transmisor

Symbol	Parameter (condition)	Notes	Min.	Typ.	Max.	Units
P_{RF}	Maximum Output Power	a		0	+4	dBm
P_{RFC}	RF Power Control Range		16	18	20	dB
P_{RFCR}	RF Power Accuracy				± 4	dB

Tabla 16. Parámetros del nRF24L01 en recepción

Datarate	Symbol	Parameter (condition)	Notes	Min.	Typ.	Max.	Units
	RX_{max}	Maximum received signal at <0.1% BER			0		dBm
2Mbps	RX_{SENS}	Sensitivity (0.1%BER) @2Mbps			-82		dBm
1Mbps	RX_{SENS}	Sensitivity (0.1%BER) @1Mbps			-85		dBm
250kbps	RX_{SENS}	Sensitivity (0.1%BER) @250kbps			-94		dBm

ANEXO 3: Fotos del prototipo controlador del sistema domótico



Figura 32. Vista frontal del prototipo controlador del sistema domótico [El Autor]

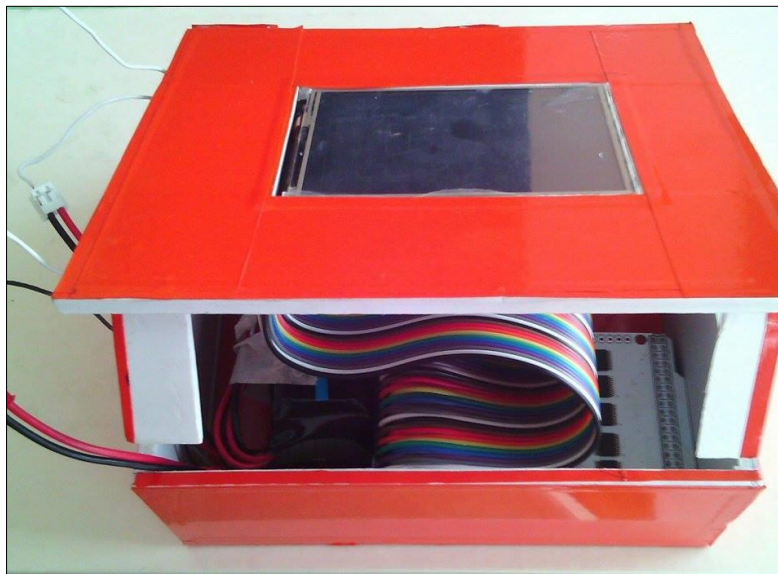


Figura 33. Vista interior del prototipo controlador del sistema domótico [El Autor]

ANEXO 4: Dispositivos finales

ANEXO 4.1: Especificaciones



Figura 34. Ventilador utilizado en el desarrollo del proyecto [El Autor]



Modelo	UT- 7T
Descripción	Calefactor Oleo Eléctrico UT- 7T
Producto	Calefactor
Material	Cuerpo íntegro de sólido de acero esmaltado.
Potencia	1500 Watts
Color	Blanco
Peso	116 kg
Marca	Ursus Trotter
Tipo	Eléctrico
Dimensiones (Largo x Ancho x Alto)	65 x 37 x 25 cm

Figura 35. Calefactor utilizado en el desarrollo del proyecto [El Autor]

ANEXO 5: Circuitos impresos

ANEXO 5.1: PCB para el circuito de conmutación de iluminación y ventilación

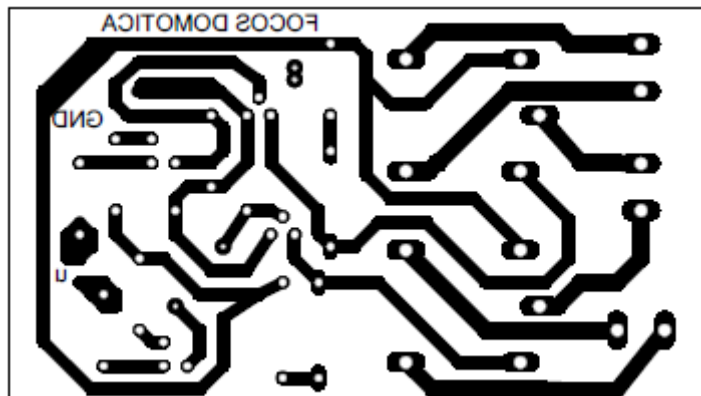


Figura 36. Circuito de conmutación para los focos [El Autor]

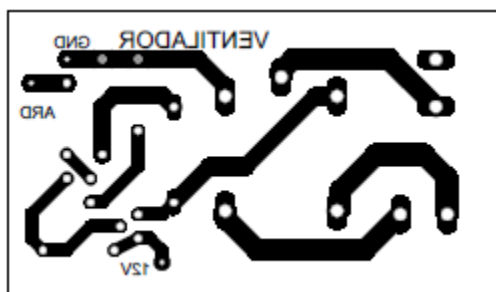


Figura 37. Circuito de conmutación para ventilador [El Autor]

Lic. Erika Pezo Badilla

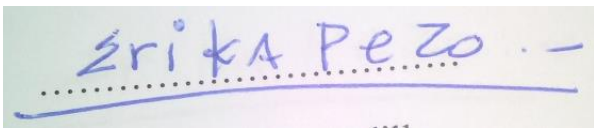
Profesora de inglés.

Certifico:

Por el presente tengo a bien certificar que, la traducción al idioma inglés del resumen de la tesis, ejecutada por la señorita **Karen Elizabeth Loaiza Jiménez**, egresada de la carrera de Ingeniería en Electrónica y Telecomunicaciones, la misma que versa como: **“DESARROLLO DE UN PROTOTIPO CONTROLADOR PARA UN SISTEMA DOMÓTICO POR MEDIO DE UNA PANTALLA TFT TOUCH SHIELD V2.0 Y ARDUINO”**, fue realizada bajo mi supervisión y se encuentra correctamente traducida bajo la estructura del lenguaje inglés.

Es todo cuanto puedo decir en honor de la verdad. La interesada puede hacer uso de este certificado como mejor convenga a sus intereses.

Atentamente,

A photograph of a handwritten signature in blue ink on a light-colored surface. The signature reads "Erika Pezo" followed by a horizontal line. There are dotted lines below the signature, suggesting it was written on a lined document.

Lic. Erika Pezo Badilla

CI: 1102884390