



UNIVERSIDAD ESTATAL PENÍNSULA DE SANTA ELENA
FACULTAD DE SISTEMAS Y TELECOMUNICACIONES
CARRERA DE TELECOMUNICACIONES

TRABAJO DE INTEGRACIÓN CURRICULAR PREVIO A LA OBTENCIÓN DEL
TÍTULO DE
INGENIERO EN TELECOMUNICACIONES

“IMPLEMENTACIÓN DE UN MÓDULO EDUCATIVO PARA EL ANÁLISIS DE LA
VULNERABILIDAD DE DISPOSITIVOS IOT, EMPLEANDO SDR Y GNU RADIO.”

AUTORES:

BELTRAN MUÑIZ KERLY MALENA

RODRÍGUEZ TORRES BRYAN JOSÉ.

DOCENTE TUTOR:

ING. AMAYA FARIÑO LUIS MIGUEL, MGTR

LA LIBERTAD-ECUADOR

2023-1

DECLARACIÓN DE DOCENTE TUTOR

DECLARACIÓN DE DOCENTE TUTOR

En mi calidad de Docente Tutor del Trabajo de Integración Curricular, "**Implementación de un módulo educativo para el análisis de la vulnerabilidad de dispositivos IoT, empleando SDR y GNU Radio**", elaborado por los señores **Beltran Muñiz Kerly Malena y Rodríguez Torres Bryan José**, estudiantes de la Carrera de Telecomunicaciones, Facultad de Sistemas y Telecomunicaciones de la Universidad Península de Santa Elena, previo a la obtención del título de Ingenieros en Telecomunicaciones, me permito afirmar que, tras supervisar el desarrollo y estructura final de su trabajo, este cumple con los estándares académicos requeridos. En consecuencia, lo considero apto en todos sus aspectos y listo para ser evaluado por el docente especialista.

Atentamente



Ing. Luis Amaya Fariño, Mgtr.

DOCENTE TUTOR

DECLARACIÓN AUTORÍA DEL ESTUDIANTE

El presente trabajo de Integración Curricular, con el título “**Implementación de un módulo educativo para el análisis de la vulnerabilidad de dispositivos IoT, empleando SDR y GNU Radio**”, declaro que la concepción, análisis y resultados son originales a la actividad educativa en el área de Telecomunicaciones.

Atentamente,



Rodríguez Torres Bryan José

C.I. 0928311018



Beltran Muñoz Kerly Malena

C.I. 2450324807

DECLARACIÓN DE DOCENTE ESPECIALISTA

DECLARACIÓN DE DOCENTE ESPECIALISTA

En mi calidad de Docente Especialista del Trabajo de Integración Curricular, "**Implementación de un módulo educativo para el análisis de la vulnerabilidad de dispositivos IoT, empleando SDR y GNU Radio**", elaborado por los señores **Beltran Muñiz Kerly Malena y Rodríguez Torres Bryan José**, estudiantes de la Carrera de Telecomunicaciones, Facultad de Sistemas y Telecomunicaciones de la Universidad Península de Santa Elena, previo a la obtención del título de Ingenieros en Telecomunicaciones, me permito afirmar que, tras supervisar el desarrollo y estructura final de su trabajo, este cumple con los estándares académicos requeridos. En consecuencia, lo considero apto en todos sus aspectos y listo para la sustentación del trabajo.

Atentamente



Ing. Daniel Jaramillo Chamba, M.Sc.

DOCENTE TUTOR

TRIBUNAL DE GRADO


TRIBUNAL DE GRADO



Ing. Ronald Rovira Jurado, Ph. D.
DIRECTOR DE LA CARRERA



Ing. Daniel Jaramillo Chamba, M.Sc.
DOCENTE ESPECIALISTA



Ing. Luis Amaya Fariño, Mgtr.
DOCENTE TUTOR GUÍA



Ing. Corina Gonzabay De la A, Mgtr.
SECRETARIA

DECLARATORIA DE RESPONSABILIDAD

Quienes suscriben, **Rodríguez Torres Bryan José** con C.I. 0928311018 y **Beltran Muñiz Kerly Malena** con C.I. 2450324807, estudiantes de la carrera de Telecomunicaciones que presentaron el trabajo de titulación denominado “**Implementación de un módulo educativo para el análisis de la vulnerabilidad de dispositivos IoT, empleando SDR y GNU Radio**” pertenece y es exclusiva responsabilidad de los autores y pertenece al patrimonio intelectual de la Universidad Estatal Península de Santa Elena.

Atentamente,



Rodríguez Torres Bryan José

C.I. 0928311018



Beltran Muñiz Kerly Malena

C.I. 2450324807

AGRADECIMIENTO

Deseo expresar mi más sincero agradecimiento a todas las personas que contribuyeron de manera significativa en la realización de este trabajo. Mi gratitud se extiende a mi familia por su apoyo incondicional desde que inicié mis estudios universitarios. Agradezco profundamente a mi asesor de tesis, Luis Miguel Amaya Fariño, por sus valiosas sugerencias y dedicación en orientarme en cada etapa de este trabajo de titulación. También quiero agradecer a mis amigos y colegas que conocí, por sus consejos, conversaciones inspiradoras y momentos de distracción, especialmente a Kerly, quien con sus palabras de aliento iluminó mi camino y me ayudó a mantener mi confianza en mí mismo. Cada uno de ustedes ha dejado una huella imborrable en este logro y les estoy sinceramente agradecido. Además, agradezco a la Universidad Estatal Península de Santa Elena y a todo su personal docente que ha sido fundamental en mi desarrollo académico y profesional.

Bryan José Rodríguez Torres

En este momento de culminación me siento profundamente agradecida por el apoyo y la guía que he recibido a lo largo de mi trayectoria estudiantil. Este trabajo no lo habría logrado sin el respaldo de mi familia. Agradezco a mi tutor de tesis el Ingeniero Luis Miguel Amaya Fariño, cuya dedicación y orientación ha sido de gran ayuda en cada etapa de este proyecto. Su sabiduría y conocimiento han enriquecido enormemente mi comprensión del tema y me han motivado a alcanzar estándares de excelencia. A mi amigo Joseph Domínguez, quien me ha brindado su amistad sincera, tu apoyo inquebrantable y palabras de aliento han sido un regalo preciado. Y principalmente a ti Bryan, que eres una persona tan especial para mí, siendo ese

pilar fundamental que me ha dado el aliento para seguir y no decaer, quien siempre ha estado de manera incondicional y sobre todo me ha brindado su amor de la forma más pura que jamás pude imaginar. También mi gratitud se extiende a la Universidad Estatal Península de Santa Elena por darme la oportunidad de realizarme como profesional y a los docentes que con dedicación forjaron mi camino. Este trabajo no solo representa el resultado de una investigación, sino también el fruto del amor, el apoyo y la dedicación de quienes me rodean. A todos ustedes, ¡mil gracias!

Kerly Malena Beltran Muñiz

DEDICATORIA

Este trabajo se lo dedico a mis padres, José y Mirella, quienes han sido una guía, mi inspiración y mi mayor apoyo a lo largo de esta travesía. A mis hermanos, Silvia y José, quienes son un ejemplo de superación para mí. A ti, Kerly, dedico este trabajo de titulación con todo mi corazón y gratitud. Tu presencia constante en mi vida ha llenado cada paso con amor y confianza, y hoy, este logro lleva tu huella indeleble. Y a mi fiel amigo Cándido, quien con su lealtad y cariño incondicional ha llenado mis días de alegría.

Bryan José Rodríguez Torres

Le dedico este trabajo a Dios, por mantenerme con firmeza y estar siempre conmigo en el transcurso de mi vida estudiantil. A mi madre Jenny Muñiz, por ser un ejemplo de mujer luchadora que, ante cualquier adversidad, siempre ha salido adelante, y que con su esfuerzo me ha convertido en una mujer llena de valores y principios. A mis hermanos Stalin, Carlos, Valentina y Ezequiel, quienes con su cariño y apoyo me han acompañado en este trayecto de mi vida, y han inculcado en mí ese deseo de superación. A mis padres Armando Beltran y Leodan Baque por estar siempre presentes y apoyarme en cada paso que doy.

A ti mi fiel compañero Bryan Rodríguez te dedico este trabajo por darme tu apoyo que ha sido fundamental para mí y por tenerme la paciencia necesaria para finalizar este proyecto que construimos con mucho esfuerzo, gracias por enseñarme el verdadero significado del trabajo en equipo. Hoy le damos un cierre a este capítulo de nuestras vidas, y me llena de felicidad que sea junto a ti.

Kerly Malena Beltran Muñiz

RESUMEN

El desarrollo del módulo electrónico educativo para el análisis de vulnerabilidades de dispositivos IoT mediante SDR y GNU Radio se realizó con la intención de ampliar más áreas de estudio con el fin de realizar análisis de señales para la búsqueda y la detección de vulnerabilidades en dispositivos IoT, en el laboratorio de Telecomunicaciones de tal manera que esta implementación se realice en un entorno controlado y supervisado.

Así mismo, se establece una alternativa para el diseño de un dispositivo IoT con funcionalidades similares al equipo comercial, de tal forma que se pueda aprovechar las nuevas implementaciones IoT en el laboratorio para facilitar la conexión hacia la red eléctrica y proporcionar otro punto de conexión. El proyecto, también incluye un entorno de prueba enfocado a la aplicación de ataques inalámbricos con el objetivo de explotar las vulnerabilidades presentes a nivel de comunicación, mediante el uso conjunto de las herramientas GNU Radio, los SDR HackRF One y RTL-SDR.

Palabras Clave: IoT, Radiofrecuencia, ISM, SDR, GNU Radio, HackRF One, RTL-SDR, NodeMCU, Arduino.

ABSTRACT

The development of the educational electronic module for the analysis of vulnerabilities of IoT devices through SDR and GNU Radio was carried out with the intention of expanding more areas of study in order to carry out signal analysis for the search and detection of vulnerabilities in IoT devices, in the Telecommunications laboratory in such a way that this implementation is carried out in a controlled and supervised environment.

Likewise, an alternative is established for the design of an IoT device with similar functionalities to the commercial equipment, in such a way that the new IoT implementations in the laboratory can be used to facilitate the connection to the electrical network and provide another connection point. The project also includes a test environment focused on the application of wireless attacks with the aim of exploiting the vulnerabilities present at the communication level, through the joint use of the GNU Radio tools, the SDR HackRF One and RTL-SDR.

Keywords: IoT, Radio frequency, ISM, SDR, GNU Radio, HackRF One, RTL-SDR, NodeMCU, Arduino.

ÍNDICE GENERAL

DECLARACIÓN DE DOCENTE TUTOR	ii
DECLARACIÓN AUTORÍA DEL ESTUDIANTE	iii
DECLARACIÓN DE DOCENTE ESPECIALISTA	iv
TRIBUNAL DE GRADO	v
DECLARATORIA DE RESPONSABILIDAD	vi
AGRADECIMIENTO	vii
DEDICATORIA	ix
RESUMEN	x
ABSTRACT	xi
ÍNDICE GENERAL	xii
ÍNDICE DE FIGURAS.....	xvii
ÍNDICE DE TABLAS	xxi
ÍNDICE DE ANEXOS.....	xxii
ÍNDICE DE ABREVIATURAS.....	xxiii
INTRODUCCIÓN	1
CAPITULO I.....	3
1 Generalidades de la propuesta	3
1.1 Antecedentes	3
1.2 Planteamiento del problema	8
1.3 Objetivos.....	11
1.3.1 Objetivo General	11
1.3.2 Objetivos Específicos	11
1.4 Justificación	12
1.5 Alcance	13
CAPITULO II.....	15
2 Fundamentación Teórica.....	15
2.1 Antecedentes Investigativos	15
2.2 Espectro Electromagnético basados en rangos de vulnerabilidad IoT	16
2.3 Aplicaciones de las comunicaciones inalámbricas en el rango de frecuencia 433 MHz.....	18
2.4 Modulaciones de señales digitales.....	19
2.4.1 Análisis de la trama de datos en la Modulación ASK.....	20
2.4.2 Análisis de la trama de datos en la Modulación FSK	21
2.4.3 Análisis de la trama de datos de la Modulación PSK	21
2.4.4 Análisis de la trama de datos de la Modulación QPSK	22
2.4.5 Análisis de la Modulación OFDM.....	23

2.5	Tecnología SDR.....	25
2.5.1	Equipos tecnológicos de radio definido por software	26
2.5.1.1	HackRF One.....	26
2.5.1.2	RTL-SDR.....	27
2.5.1.3	NOOELEC NESDR SMART HF V4.....	28
2.5.1.4	Comparación de los equipos tecnológicos de Radio Definido por Software	29
	NOOELEC NESDR SMART HF V4.....	30
2.6	Internet de las Cosas	31
2.6.1	Aplicaciones IoT	32
2.6.2	Seguridad en sistemas de Internet de las Cosas	33
2.7	Vulnerabilidades	34
2.7.1	Vulnerabilidades en sistemas de Internet de las Cosas	34
2.7.1.1	Debilidad en contraseñas	35
2.7.1.2	Inseguridad en los servicios de red	35
2.7.1.3	Interfaces inseguras en el ecosistema IoT.....	35
2.7.1.4	Falta de un mecanismo de actualización seguro	36
2.7.1.5	Uso de componentes desactualizados e inseguros	36
2.7.1.6	Insuficiente protección de privacidad.....	36
2.7.1.7	Falta de seguridad en el almacenamiento y transferencia de datos	36
2.7.1.8	Gestión de dispositivos inadecuada.....	36
2.7.1.9	Configuraciones predeterminadas inseguras.....	37
2.7.1.10	Falta de seguridad física	37
2.8	Ataques a Internet de las Cosas	37
2.8.1	Ataques Pasivos	38
2.8.1.1	Replay	38
2.8.1.2	Sniffing	38
2.8.1.3	Wardriving	39
2.8.1.4	Snooping-Donwloading.....	39
2.8.1.5	Evil Twin	39
2.8.2	Ataques Activos	39
2.8.2.1	Denegación De Servicio	40
2.8.2.2	Inhibidor de Señal	40
2.8.2.3	Fuerza Bruta.....	40
2.8.2.4	Spoofing.....	41
2.8.2.5	Ramsonware.....	41
2.9	Módulo de control relé basados en Radiofrecuencia	41
2.9.1	Smart switch relé SONOFF 4CHPRO.....	42

2.9.2	Módulo relé Sra RF programable	43
2.9.3	Switch Relé AK-RK02E	45
2.9.4	Smart Power Strip	47
2.9.5	Comparación de los dispositivos comerciales	48
2.10	Componentes para ejemplificar el diseño del Prototipo	49
2.10.1	Tarjetas para procesos físicos de Código Abierto	49
2.10.1.1	Placas Arduino	49
2.10.1.2	Microcontroladores PIC	52
2.10.1.3	Raspberry Pi	55
2.10.1.4	Comparación de las tarjetas en base a la aplicación del proyecto	59
2.10.2	Módulos de comunicación por frecuencias de radio	60
2.10.2.1	Módulo TX/RX (FS1000A/XY-MK-5V)	60
2.10.2.2	Módulo TX/RX (STX882/ SRX882).....	61
2.10.2.3	Módulos NODEMCU.....	63
2.10.2.4	Comparación de los módulos de comunicación en base a la aplicación del proyecto	64
2.10.3	Botones	65
2.10.4	Líneas de transmisión.....	67
2.10.5	Bases para el soporte de componentes.....	69
2.11	Software para el diseño y desarrollo de circuitos	73
2.11.1	Proteus VSM.....	73
2.11.2	Fritzing.....	75
2.11.3	Autodesk Eagle	75
2.12	Software para la asignación de tareas y programación de las tarjetas.....	77
2.12.1	Arduino IDE.....	77
2.12.2	Arduino IoT Cloud.....	78
2.13	Software para el análisis de información en Sistemas Inalámbricos.....	80
2.13.1	GNU Radio	80
2.13.2	RTL_433	81
2.13.3	Equipo computacional para la instalación de los softwares necesarios para el proyecto	82
CAPITULO III.....		84
3	METODOLOGÍA.....	84
3.1	Investigación Exploratoria	84
3.2	Investigación Aplicada.....	84
3.3	Desarrollo de la propuesta:.....	85
3.3.1	Fase 1: Investigación bibliográfica y elección de componentes electrónicos y softwares necesarios para el desarrollo de la implementación.....	85

3.3.2	Fase 2: Diseño y configuración de equipos	85
3.3.3	Fase 3: Análisis de datos y búsqueda de vulnerabilidades del dispositivo IoT	86
3.3.4	Fase 4: Comparación de resultados y alternativas para mitigar posibles ataques.....	86
3.4	Arquitectura del proyecto.....	87
3.4.1	Diseño de ataque replay o reproducción.....	88
3.4.1.1	Diagrama de bloque para la captura de señal.....	89
3.4.1.2	Diagrama de bloque para retransmitir la señal.....	91
3.4.2	Diseño de inhibidor de señal.	93
3.5	Análisis del modelo y características del diseño final a nivel comercial	96
3.5.1	Hardware del dispositivo.....	97
3.5.2	Software del dispositivo a nivel comercial	98
3.6	Desarrollo e implementación del prototipo	99
3.6.1	Desarrollo del hardware para el dispositivo.....	100
3.6.1.1	Definición de requisitos.....	101
3.6.1.2	Diseño del esquema.....	102
3.6.1.3	Diseño de PCB	105
3.6.1.4	Fabricación de PCB.....	107
3.6.1.5	Ensamblaje y soldadura.....	108
3.6.1.6	Programación del microcontrolador	109
3.6.1.7	Pruebas y depuración.....	110
3.6.1.8	Iteración y mejoras	111
3.6.2	Desarrollo del software para el dispositivo.....	112
3.6.2.1	Definir los objetivos y requisitos.....	112
3.6.2.2	Diseño de la interfaz de usuario	113
3.6.2.3	Desarrollo de la aplicación	115
3.6.2.3.1	Configuración de la App Amazon Alexa.....	123
3.6.2.4	Pruebas y depuración.....	124
3.6.2.5	Mantenimiento y actualizaciones	124
3.6.3	Implementación de componentes externos para receptor	126
3.6.4	Costos de implementación.....	128
3.7	Área de laboratorio destinada para aplicación del prototipo de dispositivo IoT y el dispositivo Sonoff 4CH Pro	130
CAPITULO IV.....		133
4	Pruebas y resultados de la propuesta	133
4.1	Funcionamiento de los dispositivos IoT	133
4.1.1	Pruebas de la comunicación entre un Smartphone y el Sonoff 4CH PRO	133
4.1.2	Pruebas de la comunicación entre aplicación diseñada y el Prototipo de dispositivo IoT	134

4.2	Vulnerabilidades detectadas en la comunicación	135
4.2.1	Protocolos de transmisión RF débiles.....	135
4.3	Análisis de ataques ejecutados	137
4.3.1	Ataques realizados al dispositivo Sonoff 4CH PRO.....	137
4.3.2	Ataques realizados al Prototipo de dispositivo IoT.....	138
4.4	Comparación de los resultados.....	144
4.5	Alternativas para mitigar posibles ataques	148
4.5.1	Conexión a redes conocidas y confiables	148
4.5.2	Actualizaciones de sistema.....	149
4.5.3	Conexión a múltiples redes Wifi	149
4.5.4	Librería Virtualwire para la comunicación inalámbricas en Arduino	149
CONCLUSIONES		150
RECOMENDACIONES.....		152
BIBLIOGRAFÍA		153
ANEXOS		156

ÍNDICE DE FIGURAS

Figura 1 Modulación ASK.....	20
Figura 2 Modulación FSK	21
Figura 3 Modulación PSK	22
Figura 4 Modulación QPSK.....	23
Figura 5 Modelo básico de un sistema OFDM de transmisión y recepción	24
Figura 6 Diagrama de bloques de un equipo SDR	25
Figura 7 HackRF One.....	27
Figura 8 RTL-SDR	28
Figura 9 NOOELEC NESDR SMART HF V4	29
Figura 10 Campos de Internet de las Cosas	32
Figura 11 Smart switch Relé SONOFF 4CHPRO.....	42
Figura 12 Módulo Relé RF Programable	43
Figura 13 Switch relé AK-RK02E.....	45
Figura 14 Smart Power Strip.....	47
Figura 15 Arduino UNO	50
Figura 16 Arduino NANO	51
Figura 17 Arduino Pro Mini	51
Figura 18 Microcontrolador PIC16F84A.....	53
Figura 19 Microcontrolador PIC16F877A.....	54
Figura 20 Microcontrolador PIC18F452.....	54
Figura 21 Raspberry Pi 2 Modelo B	56
Figura 22 Raspberry Pi 3 model B+.....	57
Figura 23 Raspberry Pi 4 model B.....	57
Figura 24 Módulo de Radiofrecuencia 433MHz.....	61
Figura 25 Módulo STX882/SRX882	62
Figura 26 Módulo NodeMCU ESP8266	63

Figura 27 Botón pulsador 4 PIN 6X6X13.....	66
Figura 28 Botón pulsador 2 PIN 6X6X5.....	66
Figura 29 Botón pulsador con Micro Interruptor	67
Figura 30 Jumper Dupont	68
Figura 31 Protoboard de 400 puntos	70
Figura 32 Placa PCB.....	72
Figura 33 Proteus.....	74
Figura 34 Fritzing	75
Figura 35 Autodesk Eagle.....	76
Figura 36 Arduino IDE.....	78
Figura 37 Interfaz gráfica de la plataforma IoT Cloud.....	80
Figura 38 GNU Radio.....	81
Figura 39 RTL-433.....	82
Figura 40 Sistema operativo de ASUS VivoBook	83
Figura 41 Diagrama general para el desarrollo de la propuesta.....	87
Figura 42 Arquitectura del proyecto	88
Figura 43 Diagrama de conexiones para la capturar la señal con el RTL-SDR.....	89
Figura 44 Diagrama de bloques para la captura de la señal con el RTL-SDR en GNU Radio	91
Figura 45 Diagrama de conexiones para retransmitir la señal con el HackRF One.....	92
Figura 46 Diagrama de bloques para transmitir la señal con el HackRF One en GNU Radio.....	92
Figura 47 Diagrama de bloques del HackRF One.....	93
Figura 48 Entorno de configuración de dispositivos en el programa GQRX	94
Figura 49 Analizador de redes en la aplicación WiFi Data	95
Figura 50 Diagrama de bloques correspondiente al Inhibidor de Señal	96
Figura 51 Interfaz de control del dispositivo Sonoff 4CH PRO.....	99
Figura 52 Pasos para el desarrollo del prototipo similar al dispositivo comercial.....	101
Figura 53 Esquema de conexiones del receptor	104
Figura 54 Esquema de conexiones del transmisor RF.....	105

Figura 55 Diseño del modelo PCB del receptor	106
Figura 56 Diseño del modelo PCB del transmisor	107
Figura 57 Fabricación de las placas PCB por medio de control numérico computarizado (CNC) ..	108
Figura 58 Componentes ensamblados en las placas PCB del receptor y el transmisor	108
Figura 59 Entorno de programación del receptor en Arduino IDE.....	109
Figura 60 Entorno de programación del transmisor en Arduino IDE.....	110
Figura 61 Pasos para el desarrollo de la aplicación móvil del proyecto	112
Figura 62 Distribución de los componentes definida para el computador.....	114
Figura 63 Distribución de los componentes definida para teléfonos móviles.....	114
Figura 64 Primera instancia	116
Figura 65 Ventana de desarrollo	117
Figura 66 Ventana para agregar las variables.....	117
Figura 67 ID y código del módulo NodeMCU ESP8266	120
Figura 68 Configuración para la conexión a Internet	120
Figura 69 Ventana de programación de la aplicación	121
Figura 70 Componentes distribuidos para la vista desde el computador	122
Figura 71 Componentes distribuidos para la vista desde el Smartphone.....	122
Figura 72 Interfaz de control en Amazon Alexa	124
Figura 73 Esquema de conexiones del dispositivo final.....	127
Figura 74 Presentación final del Prototipo de dispositivo IoT	128
Figura 75 Ubicación de los dispositivos IoT en el Laboratorio	131
Figura 76 Escenarios posibles de aplicación para los dispositivos IoT	132
Figura 77 Pruebas de funcionamiento del equipo comercial	134
Figura 78 Pruebas de funcionamiento del prototipo de dispositivo IoT	135
Figura 79 Identificación de librerías vulnerables mediante el hardware RTL-SDR y el software RTL-433.....	136
Figura 80 Interrupción de la comunicación entre el Smartphone y el Sonoff 4CH PRO.....	137
Figura 81 Señal emitida por el transmisor y receptada por el RTL-SDR	139

Figura 82 Comportamiento de la señal en función del tiempo	140
Figura 83 Diagrama de Constelación de la Señal.....	140
Figura 84 Representación de la señal mediante el tren de pulsos.....	141
Figura 85 Activación del puerto de conexión mediante el control RF	142
Figura 86 Ataque de repetición en la señal RF	143
Figura 87 Desconexión de la red y la aplicación del control del Prototipo de dispositivo IoT	143

ÍNDICE DE TABLAS

Tabla 1 Tipos de módulos de transmisión y recepción.....	18
Tabla 2 Cadena de bits, cambios de fase y constelación de la modulación QPSK	22
Tabla 3 Funciones de los diagramas de bloques presentes en un equipo SDR.....	25
Tabla 4 Comparativa de los equipos de tecnología SDR.....	29
Tabla 5 Características técnicas de un Módulo Relé Sra Rf Programable	44
Tabla 6 Características del dispositivo Switch Relé AK-RK02E.....	46
Tabla 7 Dispositivo Smart Power Strip	48
Tabla 8 Datos y especificaciones del fabricante de las tarjetas Arduino	52
Tabla 9 Especificaciones de los microcontroladores PIC.....	55
Tabla 10 Especificaciones de los módulos Raspberry Pi.....	58
Tabla 11 Características de los módulos transmisores y receptores	62
Tabla 12 Características de las tarjetas NodeMCU	63
Tabla 13 Componentes del dispositivo comercial	97
Tabla 14 Componentes necesarios para el desarrollo del receptor	103
Tabla 15 Componentes necesarios para un control RF.....	104
Tabla 16 Eventos detectados durante las pruebas.....	111
Tabla 17 Elementos necesarios de Arduino IoT Cloud para el proyecto.....	115
Tabla 18 Definición de variables para la aplicación móvil.....	118
Tabla 19 Mejoras para el dispositivo en desarrollo	125
Tabla 20 Costos de implementación del transmisor RF	129
Tabla 21 Costos de implementación del receptor.....	129
Tabla 22 Costos totales para el tablero de distribución eléctrica	129
Tabla 23 Costos totales del proyecto.....	130
Tabla 24 Análisis de Efectividad en las primeras pruebas de comunicación del Prototipo	146
Tabla 25 Análisis de Efectividad en la aplicación del Ataque Replay al Prototipo	147
Tabla 26 Análisis de la Efectividad del Ataque de Inhibidor de Señal	148

ÍNDICE DE ANEXOS

Anexo A Diagrama de conexiones del Receptor.....	156
Anexo B Programación receptor RF a 433.92 MHz	159
Anexo C Vinculación del dispositivo ESP8266 con la plataforma Arduino IoT Cloud.....	164
Anexo D Programación del receptor a 2.4 GHz para la aplicación	165
Anexo E Ubicación de los Widgets	167
Anexo F Parte interna del prototipo receptor.....	168
Anexo G Comprobación del paso de la energía por el puerto de conexión eléctrico	170

ÍNDICE DE ABREVIATURAS

IoT Internet of Things (Internet de las Cosas).

ARPANET Advanced Research Projects Agency Network (Red de Agencias de Proyectos de Investigación Avanzada)

Wi-Fi Wireless Fidelity o fidelidad inalámbrica es una tecnología inalámbrica que facilita la transferencia de datos entre dispositivos y proporciona la capacidad de conectarse a internet.

ISM Banda de Radio reservadas para usos Industriales, Científicos y Médicos.

SDR Radio Definida por Software es una tecnología utilizada para el desarrollo de sistemas de comunicación.

RTL-SDR es un receptor de radio definido por software de bajo costo que permite una amplia gama de aplicaciones en la recepción y decodificación de señales de radio.

HackRF One Es un dispositivo SDR de código abierto utilizado para capturar, transmitir y manipular señales.

RTL-433 Software de código abierto para la decodificación y el análisis de las señales de radiofrecuencia de diferentes dispositivos inalámbricos.

IP Internet Protocol o Protocolo de internet

GNU Radio Herramienta de desarrollo de código libre que utiliza bloques de procesamiento de señal para implementar sistemas de radio definida por software.

RF Radiofrecuencia

OWASP Open Web Application Security Project's

KHz Kilohercio es una unidad que mide la radiación de la radiofrecuencia

GHz Gigahercio es una unidad que mide la radiación de la radiofrecuencia

MHz Megahercio es una unidad que mide la radiación de la radiofrecuencia

ITU Unión Internacional de las Telecomunicaciones

Tx Transmisor

Rx Receptor

ASK Modulación por desplazamiento de amplitud.

OOK On-Off Keying conocido como Modulación Binaria Sencilla

FSK Modulación por desplazamiento de frecuencia.

PSK Modulación por desplazamiento de fase.

OFDM Multiplexación por división ortogonal en frecuencia.

IFT Transformada Inversa de Fourier.

DAC Convertidor Analógico-Digital.

ADC Convertidor Digital-Analógico.

FFT Transformada Rápida de Fourier.

IF Frecuencia Intermedia.

DoS Denegación de Servicios.

PKI Infraestructura de clave Pública.

NodeMCU Plataforma de desarrollo basada en código abierto para Internet de las Cosas.

GRC Herramienta que facilita la construcción de flujogramas para el procesamiento de señales en GNU Radio mediante un entorno gráfico.

MQTT Message Queing Telemetry Transport es un protocolo de comunicación de código abierto diseñado para facilitar la transferencia de mensajes entre dispositivos conectados a IoT.

INTRODUCCIÓN

El presente trabajo de investigación tiene como propósito realizar el acondicionamiento de un módulo educativo para el estudio de las vulnerabilidades en el laboratorio de Telecomunicaciones de la Universidad Estatal Península De Santa Elena para el beneficio de los estudiantes de la facultad de Sistemas y Telecomunicaciones y estudiantes de Maestría.

El trabajo de investigación está dividido en cuatro capítulos cada uno de ellos estructurados de la siguiente manera:

En el capítulo I, se plantea un estudio introductorio partiendo de los antecedentes de la investigación, el planteamiento del problema de estudio, teniendo en claro los límites definidos por el tema de implementación de la propuesta, resaltando los objetivos que se desean alcanzar para el desarrollo del proyecto, la justificación donde se presentará el motivo de realización de la investigación y por último el alcance en el cual se definirá hasta donde se quiere llegar con la implementación de la propuesta.

En el capítulo II, se presenta la fundamentación teórica, donde se realizará la investigación de las definiciones principales de las partes de hardware y software que intervienen en el despliegue del laboratorio, para entender de mejor manera el funcionamiento y las características al momento de empezar con la construcción del proyecto.

En el capítulo III, se dará a conocer la metodología de investigación utilizada para desarrollar de mejor manera la propuesta tecnológica y facilitar el diseño de todos los equipos y diagramas o códigos de programación que serán empleados en el módulo educativo en un ambiente controlado dentro de las instalaciones del laboratorio de Telecomunicaciones de la Universidad Estatal Península de Santa Elena, siendo el diseño y construcción de los ataques la primera sección y en la segunda, el desarrollo y comparación de las funcionalidades del equipo comercial con el prototipo, los cuales serán vulnerados y atacados con respecto al

análisis de la información recabada y el diseño de los ataques que están descritos en la primera sección.

Y por último el capítulo IV, en este apartado se ejecutan las pruebas y análisis para determinar cómo afecta el tipo de ataque diseñado para vulnerar el dispositivo IoT (Internet de las cosas) tomando en cuenta un entorno de aplicación real para luego describir las conclusiones en base a los resultados obtenidos con respecto a los ataques y el prototipo diseñado para en el anterior capítulo y determinar las alternativas que permiten evadir los ataques que vulneran la seguridad de los equipos instalados en el área de pruebas del laboratorio o evitar que existan fugas de información en el sistema de dispositivos conectados a la red de Internet de las cosas.

CAPITULO I

1 Generalidades de la propuesta

En este capítulo, se abordarán temas importantes para el estudio introductorio de este trabajo tales como los antecedentes de la investigación, el planteamiento del problema de estudio, además se describen cada uno de los objetivos que se desea alcanzar para el desarrollo del proyecto, la justificación y por último el alcance de la propuesta.

1.1 Antecedentes

La red ARPANET era un sistema que comprendía una serie de computadoras conectadas mediante los primeros protocolos de comunicación desarrollados con fines investigativos y de estudio de las conectividades de los objetos, este proyecto lo llevó a cabo el Departamento de Defensa de los Estados Unidos (DOD) en el año 1970 el cual limitó su uso para estudios militares y académicos (Baró, Inicios de la construcción de la primera red, 2002). Durante los años 70 y 80 las innovaciones aplicadas a esta red fueron lentas por lo que en los próximos años se evidenció una deficiencia en la velocidad de las comunicaciones, debido a esto procedieron con la creación de nuevas redes denominadas heterogéneas las cuales no fueron compatibles entre ellas. Hasta que a mediados de los años 90 se empezó a interconectar las redes creadas años atrás mediante un protocolo de internet de esos tiempos, el protocolo de control de transmisión TCP/IP que permitió la expansión de la red ARPANET que hoy en día es conocida como la red de Internet (Baró, Motivaciones originales de ARPANET e Internet, 2002).

Durante los años 70 y 80 las innovaciones aplicadas a esta red fueron lentas por lo que en los próximos años se evidenció una deficiencia en la velocidad de las comunicaciones, debido a esto procedieron con la creación de nuevas redes denominadas heterogéneas las cuales no fueron compatibles entre ellas. Hasta que a mediados de los años 90 se empezó a

interconectar las redes creadas años atrás mediante un protocolo de internet de esos tiempos, el protocolo de control de transmisión TCP/IP que permitió la expansión de la red ARPANET que hoy en día es conocida como la red de Internet (Baró, Motivaciones originales de ARPANET e Internet, 2002).

A la par de la aparición de Internet surgieron los experimentos que relacionaban a dispositivos del hogar conectados a la red, idea que fue tomada por John Romkey y Simón Hackett quienes presentaron la primera tostadora de pan enlazada a Internet en la feria Interop en Estados Unidos en el año 1990 mientras que para el año 1991 realizaron modificaciones al primer diseño con la finalidad de simplificar la interacción humana, en este caso agregando un sistema de grúas que se encargaba de depositar la rebanada de pan dentro del tostador y así automatizar el proceso (LivingInternet, 2022). Con estos estudios y experimentos se dio un nuevo punto de vista con respecto al uso que se le puede dar a las redes, sin embargo, las comunicaciones se realizaban mediante medios guiados lo cual imposibilitaba que la idea de conectar dispositivos de manera inalámbrica tenga buena acogida.

En el siglo XXI se emplea el uso de las conectividades por medios inalámbricos, fue donde el Wi-fi se convirtió en uno de los más solicitados para este tipo de conexiones gracias al acceso a internet, esto abrió las puertas a las nuevas tecnologías orientadas a la automatización de procesos para diferentes fines. Internet de las cosas tuvo sus inicios en el año 1999 donde se usaba para fines investigativos hasta el año 2009 que el ingeniero especializado en la identificación por radiofrecuencia Kevin Ashton comenzó a emplear de manera formal el término IoT.

Internet de las cosas o por sus siglas en inglés IoT, es una tecnología que permite mantener una interconexión de dispositivos electrónicos, también se define como la conexión de objetos a Internet, con el fin de intercambiar o procesar datos sobre un entorno físico para

proveer servicios a los usuarios finales. La tecnología IoT es una infraestructura caracterizada por representar la gran evolución del Internet que supuso un avance que va más allá de la comunicación entre personas, debido a que esta permite proveer servicios mediante la interconectividad de objetos, sean estos físicos o virtuales, de esta manera IoT busca eliminar las barreras existentes entre un mundo físico y virtual (Barrio, 2022).

Ante el aumento de diversos dispositivos que se vinculan a la red, aumentan las probabilidades de ocupar información personal para la ejecución de ciertas actividades, desde el uso de contraseñas para abrir cerraduras electrónicas, cuentas de banco o archivos de gran importancia. Con esta exposición de estos datos e información personal es posible ser blanco fácil de ciberataques efectuados por personas maliciosas que pretenden extraer información de todo tipo para darle un uso inadecuado.

Con la aparición de esta nueva modalidad de robo de información, se solicitaron trabajadores capaces de hallar solución ante el robo de datos personales y fueron los denominados hackers quienes proponían diferentes formas para asegurar que la información no se filtre. Existe un mal punto de vista ante el termino hacker al cual en ocasiones se refieren como una persona que encargada de realizar ataques para causar algún tipo de alteración en un sistema de información.

Según la RAE, el termino Hacker hace referencia a una persona dedicada al manejo de computadoras que analiza y advierte los fallos existentes en un sistema informático y a su vez da a conocer diversas alternativas que permitan la solución de estos (Real Academia Española, 2014).

En la actualidad, la fabricación de estos dispositivos que operan en la banda ISM busca la automatización de los procesos en casas inteligentes, empresas o laboratorios, donde ofrecen servicios automatizados que llaman la atención de aquellos que buscan simplificar la

interacción humana de algún dispositivo que está conectado a la red, el cual es manipulado por medio de algún tipo de App o acceso al control del equipo. A medida que el tema de IoT se hizo popular, se convirtió en una puerta para que los fabricantes de nuevas tecnologías desarrollen todo tipo de dispositivos y sistemas automáticos para cubrir la demanda (Santos, 2020).

Sin embargo, la seguridad de los datos no es algo todos toman en cuenta a la hora de adquirir alguno de estos aparatos, para los desinteresados podría no ser de importancia confiar parte de su información personal a una máquina, esta carencia de seguridad generalmente se da en equipos donde su calidad y valores comerciales son bajos.

Si bien es cierto, con la aparición de estos dispositivos IoT surgieron ataques hacia su sistema, no requieren necesariamente un contacto físico para causar algún tipo de daño a la víctima, basta con interceptar la frecuencia en que se comunican estos equipos para extraer algún tipo de información, los sistemas que comúnmente son vulnerables a este tipo de ataques son las tecnologías empleadas para automatizar el hogar, como portones, garajes, switch inteligentes, sistema inteligente de luces, calefactores, aires acondicionados, entre otros. Cabe recalcar que estos dispositivos inteligentes operan en la banda ISM (Mora, 2001).

Aunque a nivel de aplicación los dispositivos IoT representan una de las tecnologías con mayor demanda en el mercado mundial por sus múltiples beneficios, estos también presentan falencias a nivel de seguridad fortificadas, y no solo por tener configuraciones de contraseñas débiles, pues debido a la ausencia de seguridad, los atacantes pueden controlar estas tecnologías mediante la explotación de vulnerabilidades. Por este motivo los dispositivos IoT se consideran el eslabón más débil en un sistema inalámbrico, ya que estos permiten que los atacantes informáticos ingresen a una red, controlen computadoras o incluso infecten de malware a los dispositivos.

Considerando estos aspectos, se ve la necesidad de crear un módulo educativo el cual es una herramienta que proporciona los elementos necesarios para aplicar todos los conocimientos teóricos en entornos prácticos, con la ayuda de los software y hardware necesarios, debido a que la enseñanza teórica en temas de seguridad inalámbrica al momento de analizar sus vulnerabilidades debe ser complementada con prácticas de laboratorio que aporten un mayor enfoque en lo aprendido. Por esta razón se realiza el acondicionamiento del laboratorio para practicar ciberataques con la intención de interceptar la información de algún equipo automatizado con un sistema IoT, y así analizar sus vulnerabilidades.

Por otra parte, para realizar una comparativa entre un equipos se pretende desarrollar un prototipo que cumplirá las funciones del equipo comercial IoT adquirido diseñado mediante tarjetas de la marca Arduino para ser previamente programados en una especie de modelo transmisor-receptor con la ayuda de módulos de radiofrecuencia que trabajan en la banda ISM para hacer que la comunicación sea inalámbrica entre el prototipo destinado para la distribución de energía eléctrica a través de sus tomas de corriente del tipo B.

Con respecto al medio que realizará el análisis de la información entre una comunicación inalámbrica se toma en cuenta la tecnología de radio definido por software (SDR) que proporciona una amplia gama de funcionalidades en dependencia del modelo de dispositivo debido a la existencia de varios hardware que se caracterizan por su diseño, componentes, capacidad de transmisión o recepción, anchos de banda y otras características que un solo dispositivo puede tener, cabe recalcar que gracias a los avances tecnológicos se puede integrar las funciones y tareas que realizaban otros equipos en un solo hardware, otro aspecto muy importante es su disponibilidad para aceptar una reprogramación del dispositivo con la finalidad de ejercer tareas diferentes (Lopez, 2019).

Con el equipo de radio definido por software modelo RTL-SDR y el HackRF One, se pretende realizar un análisis para empezar a identificar vulnerabilidades en este tipo de comunicaciones con ayuda de software libres como RTL-433, GNU Radio y Universal Radio Hacker que son compatibles para estos dispositivos, de tal forma que posibilite el análisis de gráficas o todo tipo de información que se logre extraer de la comunicación y determinar qué tipo de ataques podrían ejecutarse.

1.2 Planteamiento del problema

Durante los últimos años la humanidad ha vivido un progresivo crecimiento tecnológico, donde el internet paso de ser un medio que proporcionaba acceso a la información a convertirse en un espacio que posibilita la comunicación entre diferentes dispositivos, permitiendo su acceso y control a distancia. En la actualidad el número de dispositivos conectados a internet excede los 30 billones, por lo que se estima que para el 2025 esta cifra aumente a 75 billones (Pérez, 2021).

El Internet de las cosas comprende todos y cada uno de los productos que están conectados a internet, por lo que resulta común disponer en la oficina de dispositivos tecnológicos para ofrecer nuevas funciones, mejorar la eficacia y reducir costos, o en el hogar aplicado a optimar el estilo de vida y confort de las personas. Así como estas tecnologías benefician a los hogares, las fábricas y las ciudades, también pueden introducir puntos ciegos y riesgos de seguridad en forma de vulnerabilidades tanto a nivel lógico como físico en protocolos, infraestructuras, aplicaciones o servicios, lo cual aumenta exponencialmente la cantidad de ataques a los que están expuestos los usuarios y empresas, debido a que puede extraerse información importante. Esto se debe a que los fabricantes no incorporan la seguridad necesaria para contrarrestar amenazas de este tipo, solo por cubrir la demanda del mercado en disponer de nuevas alternativas de dispositivos IoT.

La seguridad de las IoT es el aspecto más importante a considerarse en un sistema de comunicación, sin embargo la ISM en el 2016 demostró que la mayoría de los dispositivos evidencian problemas de privacidad a nivel de seguridad informática, lo que prácticamente significa quedar vulnerable por: radiofrecuencia o a través de la red IP, donde fácilmente se podría duplicar dispositivos para controlarlos de forma remota dejando expuesta y a disposición información personal de suma importancia generando problemas no solo de manera personal sino también a un campo de alto interés como la industria causando pérdidas económicas si se filtra información, en el área de la salud alterando resultados provocando fallos en casos clínicos de alto interés, desventajas en el transporte privado; alteración en los requerimientos de usuario y cambios de ruta para fines antisociales (Ziegler, Arn, & Chambers, 2017).

Los dispositivos IoT tienen un alcance flexible ya sea a través de medios inalámbricos o cableados. El medio de conectividad inalámbrico se clasifica en corto, mediano y largo alcance. Siendo la conectividad de corto alcance la más común donde los canales de frecuencias utilizables van desde los 4.33 MHz en la banda ISM (Industrial, Científico y Médico), 5GHz y 2.4 GHz, aunque la mayoría de estos dispositivos operan en la banda de frecuencia ISM, el cual es un entorno donde la comunicación resulta ser vulnerable por que utiliza ondas de radio para emitir señales de corto alcance, sin embargo, este tipo de tecnología requiere de poco consumo de potencia, tales como sistemas de telemetría y domótica. La aplicación de las comunicaciones inalámbricas facilita el trabajo en diferentes entornos, pero a su vez es una puerta de ataques mediante el estudio del espectro de frecuencia. La aplicación de la tecnología IoT facilita el trabajo en diferentes entornos, pero a su vez abre una puerta a varios tipos de ataques cibernéticos. Motivo por el cual, el objetivo del trabajo es demostrar el uso del dispositivo SDR y el software GNU Radio como una herramienta de análisis, desde los

diferentes tipos de ataques, para detectar la vulnerabilidad que presentan las comunicaciones de los dispositivos IoT.

Además de carecer de seguridad incorporada, la vulnerabilidad de los dispositivos IoT se debe a que sus usuarios, contribuyen a que las amenazas afecten a estos equipos, a continuación, se detallan algunas de las razones por la tecnología IoT sigue siendo vulnerable:

- La tecnología de transmisión es variada, dificultando la aplicación de métodos y protocolos de protección.
- Al cumplir con funciones específicas poseen capacidades informáticas limitadas y restricciones de hardware.
- Poseen componentes vulnerables.
- Los usuarios no poseen el conocimiento necesario a nivel de seguridad, por lo que los dispositivos están en constante riesgo.

Existen varios casos que comprueban el impacto de las vulnerabilidades en equipos IoT, muchos de ellos implican entornos de investigación y del mundo real. OWASP, es una empresa que en el 2018 publicó un listado de las principales vulnerabilidades IoT, entre las cuales cita las siguientes.

- Contraseñas fáciles de descifrar, los nuevos malware utilizan esta vulnerabilidad.
- Plataformas automatizadas inseguras que encadenan funciones de varios dispositivos.
- Servicios inseguros a nivel de red que pueden revelar y dar acceso a la información del usuario a personas externas.

1.3 Objetivos

1.3.1 Objetivo General

- Diseñar los ataques y estudiar las vulnerabilidades más comunes en los dispositivos IoT operando en la banda ISM mediante el uso de los equipos SDR HackRF One, RTL-SDR y el software GNU Radio.

1.3.2 Objetivos Específicos

- Analizar los diferentes ataques a los que están expuestos los dispositivos IoT para identificar las posibles vulnerabilidades que se podrían presentar en un entorno de aplicación real.
- Construir un módulo electrónico educativo destinado para el desarrollo de prácticas que beneficien a los estudiantes, creando un entorno para el análisis de las comunicaciones inalámbricas e identificar las vulnerabilidades en sistemas IoT.
- Implementar un circuito de radiofrecuencia que opere a 433 MHz empleando un módulo RF transmisor-receptor con el uso de tarjetas Arduino Nano programadas para simular el funcionamiento del mando switch relé.
- Establecer una lógica de bloques que permita la captura y la transmisión de señales por medio del dispositivo RTL-SDR y HackRF One mediante el software GNU Radio.
- Examinar la información almacenada representando por medio de gráficos el comportamiento de la frecuencia, el diagrama de constelación y el tren de datos de la señal que indica el tipo de modulación mediante el software GNU Radio.
- Acondicionar un área en el laboratorio de telecomunicaciones para destinar una aplicación real del dispositivo de uso profesional switch relé.
- Diseñar y ejecutar los ataques al entorno práctico y real para comparar los resultados obtenidos en ambos escenarios implementados en el área de pruebas del laboratorio.

- Determinar las posibles soluciones que permitan evitar que se vulnere la seguridad en la comunicación de los dispositivos instalados en el área de prácticas del laboratorio.

1.4 Justificación

Esta investigación se encamina hacia los ataques que pueden llegar a sufrir algunos aparatos que forman parte del Internet de las cosas los cuales operan a cierto rango de frecuencia, para ello el uso de las herramientas y softwares libres para la simulación es una de las mejores opciones para mantener un punto de vista de lo que se puede lograr en este tipo de programas.

El trabajo conjunto de las herramientas GNU Radio y SDR, forma una herramienta extremadamente atractiva, en el campo de los sistemas de comunicación digital, ya que permite construir y modelar elementos como filtros, conversores analógico-digital o digital- analógicos, receptores y emisores, entre otras aplicaciones, facilitando de esta manera la comprensión de estos temas para los estudiantes.

Ante los ataques que se puede dar en un entorno real al estar expuesto, es posible plantear alternativas a futuro para poder sobrellevar estos casos en el que la información se ve alterada, afectada o filtrada debido a los ciberataques perpetrados por ciberdelicuentes mayormente dirigidos a equipos con escasa seguridad de los datos y de la información.

Actualmente, los fabricantes de este tipo de tecnologías del Internet de las cosas buscan cubrir la demanda de los equipos que se comercializan en el mercado dejando de lado un tema muy importante como es la seguridad de la información, proporcionando una brecha para que personas malintencionadas traten de vulnerar la escasa seguridad de estos dispositivos.

En busca de expandir los conocimientos a futuro, se debería adquirir más equipos IoT para interrumpir sus operaciones y habilitar nuevas alternativas de estudio en el Laboratorio de Telecomunicaciones de la Universidad Estatal Península de Santa Elena. Considerando el alto

coste de estas tecnologías IoT, se ve la necesidad de estudiar dispositivos que cumplen funciones similares y presenten costos más bajos, estos serán de gran utilidad para el análisis de los datos que son transmitidos tomando en cuenta la falta de protocolos de seguridad o encriptación proporcionando una alternativa para vulnerar sus operaciones.

1.5 Alcance

Esta investigación dirigida a las vulnerabilidades en los equipos operando en la banda ISM pretende exponer los ataques más comunes que podrían alterar o afectar su modo de operación mediante un análisis de las frecuencias interceptadas y la información o los datos que se pueden extraer para luego determinar el tipo de ataque que podría estar sufriendo el dispositivo alterado.

Para ello se debe considerar un medio que facilite captar dicha información que se produce durante una comunicación inalámbrica y un software que permita interactuar con el equipo mediante simulaciones a través de aplicaciones, de este modo se pretende ocupar dispositivos de radio definido por software como el RTL-SDR que se usará como un detector de frecuencias cuando se requiera captar señales de 500 KHz a 1.7 GHz y el dispositivo HackRF One que cumplirá la función de atacante para rangos de frecuencias de 1 MHz a 6 GHz gracias a la capacidad de frecuencias que puede transmitir, cabe recalcar que también puede cumplir la función de detector con capacidad para analizar señales en la banda ISM.

Las pruebas realizadas con las simulaciones permitirán dar un punto de vista a menor escala de los ataques que se podrían evidenciar en un entorno real donde se pueda ver afectado algún tipo de proceso que requiera de la conexión a la red para cumplir las acciones determinadas por el usuario a cargo del sistema.

La adecuación actual de un entorno controlado para el análisis de las frecuencias y el estudio de estas mediante el RTL-SDR y HackRF One servirá como bases primarias para

habilitar más áreas de investigación en el laboratorio de telecomunicaciones de la Universidad Estatal Península de Santa Elena para el beneficio de los estudiantes de la facultad de Sistemas y Telecomunicaciones y estudiantes de Maestría. Se estima para un futuro la implementación de nuevos equipos, dispositivos de medición y así proporcionar otras alternativas para examinar y realizar otras actividades referentes al tema de ciberataques y seguridades en las comunicaciones inalámbricas.

CAPITULO II

2 Fundamentación Teórica

En este apartado se realiza una revisión bibliográfica para cada uno de los temas importantes con el fin de analizar y definir conceptos que ayuden a comprender el funcionamiento y las características de los componentes que se utilizó para el desarrollo de este trabajo.

2.1 Antecedentes Investigativos

En busca de proporcionar información relacionada con el tema principal de investigación, se realizó una revisión literaria sobre artículos y publicaciones de revista que ayudarán con la comprensión de los argumentos como las amenazas en la seguridad IoT, vulnerabilidades, el análisis de las señales, detección o ejecución de ataques que serán de gran utilidad para el entendimiento del proyecto.

En el año 2019 Jaimes Rico, R., & Lazcano Salas, S. publican en la revista ReCIBE “SDR y GNU Radio como plataforma para un laboratorio de comunicaciones digitales”, en este trabajo analizan la radio definida por software como base para un laboratorio de comunicaciones digitales en programas académicos vinculados a las telecomunicaciones, que se logró gracias al análisis de temas fundamentales como el muestreo, filtrado, modulación y la viabilidad de implementarlos en un esquema SDR a través de un HackRF One en el entorno de GNU Radio (Rico & Salas, 2019).

Juan Carlos Martínez Quintero, Paola Estupiñán Cuesta E & Vanessa Rodríguez Días S. en el año 2020 publican en la revista RISTI “Mitigación del ataque replay en automóviles usando SDR y CNN”, este trabajo habla sobre el desarrollo de un detector de ataques replay que analiza y diferencia las señales retransmitidas de alta o baja ganancia por medio de tres redes neuronales convolucionales, el cual se encarga de evaluar la eficiencia de la red a través

del área bajo la curva para determinar la exactitud de estas redes que, en este caso, obtuvo resultados favorables superiores al 90% (Carlos Martínez Quintero et al., 2020).

Según Joffre J. Cartuche Calva Dixys L. Hernández Rojas Rodrigo F. Morocho Román Ciro D. Radicelli García; en un artículo publicado en la revista Hamut'ay en el año 2021 “Seguridad IoT: Principales amenazas en una taxonomía de activos”, este artículo aborda los potenciales inconvenientes relacionados con la seguridad, como las vulnerabilidades, ataques y amenazas, entre otros. Se analizan los dispositivos susceptibles de ser afectados mediante el intercambio de información en respuesta a las amenazas existentes. Estas amenazas son evaluadas considerando los niveles de riesgo e impacto en un entorno de Internet de las cosas (Cartuche Calva et al., 2021).

2.2 Espectro Electromagnético basados en rangos de vulnerabilidad IoT

La tecnología IoT al tener su variante inalámbrica utiliza como medio de comunicación el aire para enviar y recibir la información mediante ondas electromagnéticas. La mayoría de los dispositivos IoT se encuentran operando en la banda ISM, esta pertenece a las bandas no licenciadas y cuenta con un amplio rango de frecuencias para transmitir, sin embargo, la mayor parte de las tecnologías IoT centran su funcionamiento en frecuencias inferiores a 1GHz:

- **Banda ISM 433.92 MHz**

Esta banda es utilizada por los dispositivos de bajo costo, sin contar con ningún tipo de seguridad al momento de ejecutar la comunicación. El uso de esta banda de frecuencia se despliega por África, Rusia, Europa y en gran mayoría de la región 1 de la Unión Internacional de Telecomunicaciones (ITU).

- **Banda ISM 868 MHz**

A pesar de que posee una cobertura menor, lo que realmente hace importante a esta frecuencia es el bajo nivel de perturbación que existe al momento de realizarse la transmisión

de datos, debido al uso limitado que posee. Como esta banda no está disponible en todos los países, es usada únicamente por la agrupación CEPT, específicamente en Estados Unidos, Canadá, Australia, Europa, Nueva Zelanda, Perú, Israel, Sudáfrica y las Antillas Neerlandesas.

- **Banda ISM 915 MHz**

Es una banda que opera de forma similar a la banda 433.92 MHz en países americanos pertenecientes a la región 2 de la ITU y países norteamericanos como Estados Unidos. Otras bandas usadas mayormente para las comunicaciones inalámbricas entre los dispositivos IoT, son las frecuencias 2.4 GHz y 5 GHz.

- **Banda 2.4 GHz en IoT**

La banda de frecuencia de 2.4 GHz se encuentra dentro del rango de frecuencia de radio no licenciadas y es utilizada por muchos dispositivos inalámbricos. Esta banda es utilizada por estándares de comunicación inalámbrica como Bluetooth y Wi-Fi, los cuales son protocolos empleados para la interconexión de dispositivos IoT con redes locales de internet.

A pesar de ser una banda que ofrece una amplia compatibilidad puede presentar interferencias debido a la gran cantidad de dispositivos conectados, lo cual afecta el rendimiento y la calidad de la conexión. Para solventar estos problemas se ha desarrollado bandas de frecuencia como la banda de 5 GHz, la cual posee mayor ancho de banda, pero tiene un menor alcance.

- **Banda 5 GHz en IoT**

A diferencia de la banda 2.4 GHz, la banda de 5 GHz ofrece un mayor ancho de banda y menor interferencia debido a que la cantidad de dispositivos que la utilizan es reducida. Sin embargo, es importante tener en cuenta que esta banda presenta limitaciones, como la cobertura que puede ser limitada en ciertos escenarios, por lo que es imprescindible considerar su uso según las necesidades específicas de la aplicación de IoT.

Otra limitación es la compatibilidad que esta pueda tener con ciertos dispositivos, debido a que existen tecnologías antiguas o más económicas que probablemente trabajen solo en la banda de frecuencia de 2.4 GHz.

2.3 Aplicaciones de las comunicaciones inalámbricas en el rango de frecuencia 433 MHz

Hoy en día existen muchas tecnologías y dispositivos que están operando en la banda ya mencionada, desde aparatos a control remoto como juguetes con sistemas RC, los dispositivos que facilitan la apertura y el bloqueo de los automóviles, dispositivos que cuentan con un sistema de comunicación encargados de la automatización de un lugar como sensores, actuadores, relés inteligentes, entre otros componentes que se comunican a esta frecuencia o mantienen una red de monitoreo donde emplean esta banda para realizar transmisiones de datos hacia una aplicación que se encargará de recolectar estos datos o también, con la capacidad de ser manipulado a través de sus aplicaciones que se enlazan a los dispositivos implementados para el control o automatización del área, en la tabla 1 se puede evidenciar los modelos de arreglos de circuitos para el sector de radiofrecuencias que poseen algunos dispositivos IoT.

Tabla 1

Tipos de módulos de transmisión y recepción

MÓDULO		CARACTERÍSTICAS			
Tx	Rx	FRECUENCIA TRANSMITIDA	POTENCIA	ALCANCE	SENSIBILIDAD
MX-FS-03V	MX.05V	433MHz	10 Mw	350 m	-105dB
STX882	SRX882	433MHz	50 Mw	100 m	-105dB
FS1000A	XY-MK-5V	433MHz	10 Mw	352 m	-105dB

TWS-BS	RWS- 371	433.92MHz	13 Mw	353 m	-108dB
--------	-------------	-----------	-------	-------	--------

Nota: En la tabla 1, se observan ejemplos de módulos de transmisión que transmiten en una frecuencia de 433MHz.

2.4 Modulaciones de señales digitales

La modulación en la señal es un tema muy diverso en el área de las Telecomunicaciones, ya que es el proceso en el que se puede transportar información a distancias muy extensas, en este tema influyen dos puntos importantes como la señal portadora y la señal del mensaje o moduladora las cuales tienen características diferentes, en el caso de la moduladora, corresponde una señal de baja frecuencia que difícilmente puede ser propagada a una distancia muy grande debido a que no contiene una potencia necesaria para viajar, en cuanto a la señal portadora se considera como una frecuencia superior a la anterior que tiene un nivel de frecuencias elevado, lo siguiente es la modulación de la señal que corresponde a la superposición de la información de la señal moduladora con una portadora y así poder propagar la información por el medio considerando espacios más extensos (Couch, Sistema De Comunicación Digitales Y Analógicos, 2015).

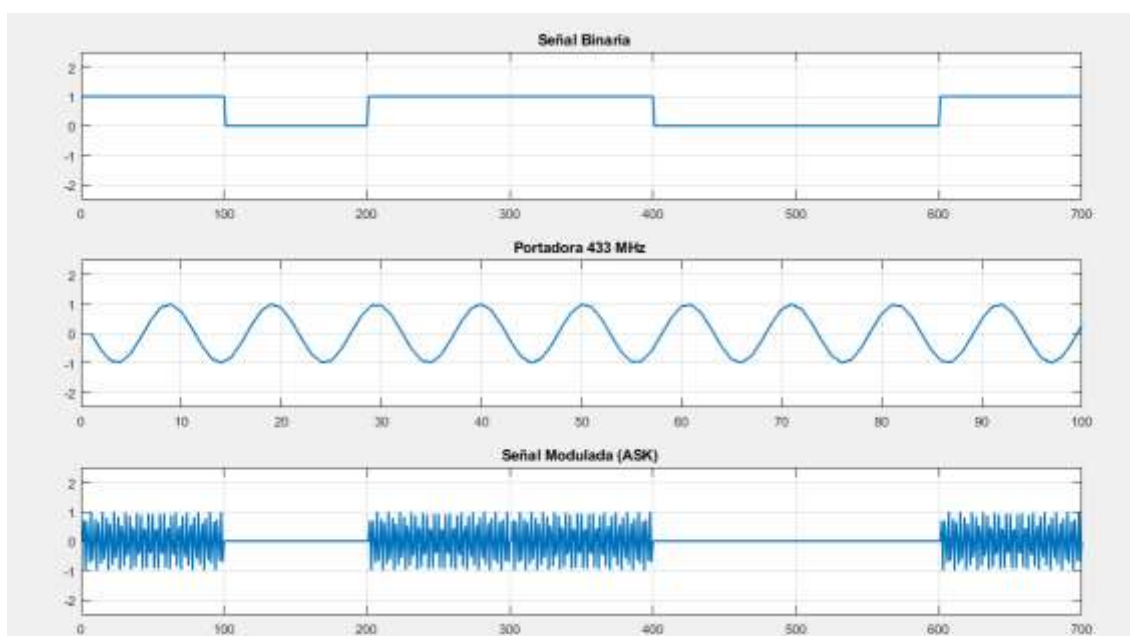
Los resultados de la trama de datos una señal digital modulada van a variar con respecto al tipo de modulación por desplazamiento que se vaya a realizar de tal forma que existe la modulación por desplazamiento de amplitud (ASK), la modulación por desplazamiento de frecuencia (FSK) y la modulación por desplazamiento de fase (PSK), de esta manera introducimos analizando el comportamiento de cada una de ellas al considerar una señal binaria que se modulará con una portadora de frecuencia de operación de 433 MHz tomando en cuenta el tipo de oscilador del módulo RF que se pretende utilizar para las practicas finales.

2.4.1 Análisis de la trama de datos en la Modulación ASK

La modulación por desplazamiento de amplitud, es una de las modulaciones más comunes que se emplean a nivel de radiofrecuencias, se la menciona como un tipo de modulación por corrimiento de amplitud y su operatividad es muy similar a la modulación del tipo OOK, para ello se considera la figura 1 que muestra un comportamiento de la señal del tipo binaria que en este caso fue “1011001” y es mostrada en la primera sección de las señales representadas en el gráfico, por consiguiente se muestra su portadora con la frecuencia de operación antes mencionada previo a su mezcla con la señal binaria, lo cual generarán distintos modos en la señal, en este caso, al emitir un código binario con nivel 1 la señal en esta parte se mezclará con la portadora obteniendo resultados que llegaran a más distancias, por otra parte están los niveles bajos o un 0, que al mezclarse con la señal portadora provocará que se suprima esta sección de la onda, de esta manera se muestra en la parte final de la figura el resultado de la señal luego del proceso de modulación que se ejecuta previo a la transmisión a través de las antenas (Couch, Modulación ASK, 2015).

Figura 1

Modulación ASK

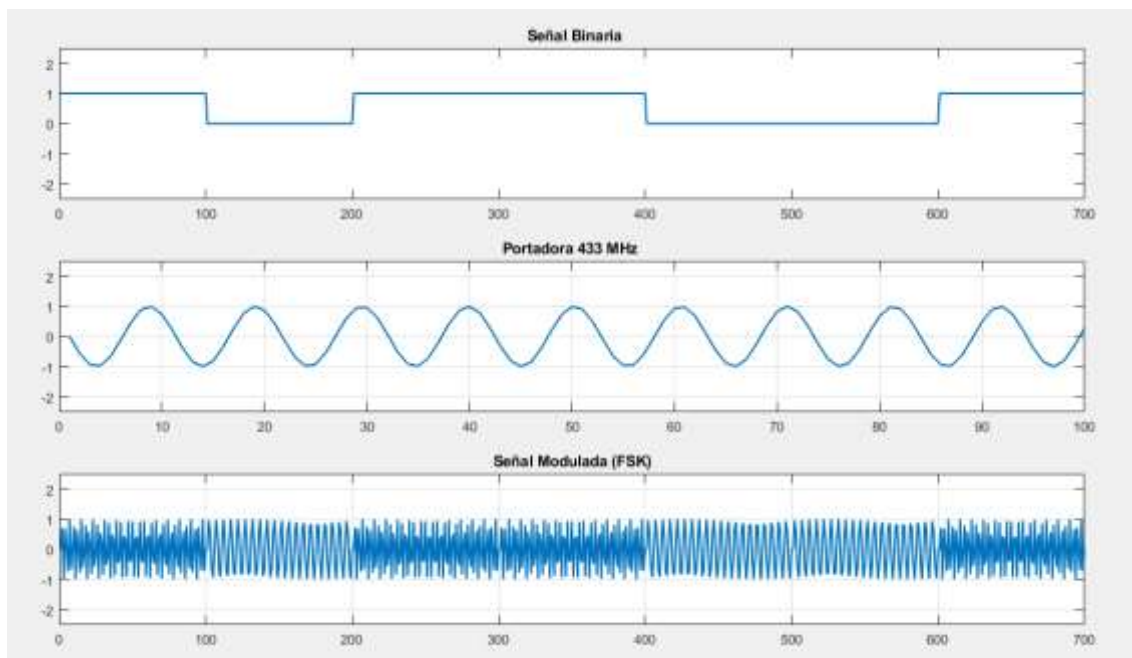


2.4.2 Análisis de la trama de datos en la Modulación FSK

A la modulación FSK se la denomina modulación por corrimiento de frecuencia y su funcionamiento se basa en que si se tiene una transmisión de un uno lógico se comportará como en la modulación anterior, sin embargo, al pasar un 0 lógico se tiene una pequeña porción por donde se cambia la señal y la frecuencia se ve afectada (Couch, Modulación FSK, 2015), tal como se muestra en la figura 2 que muestra tres gráficos, el primero corresponde a una señal binaria “1011001”, el siguiente gráfico corresponde a una señal portadora de frecuencia 433 MHz que al ser modulada con la señal binaria se obtiene el resultado mostrado al final del gráfico.

Figura 2

Modulación FSK



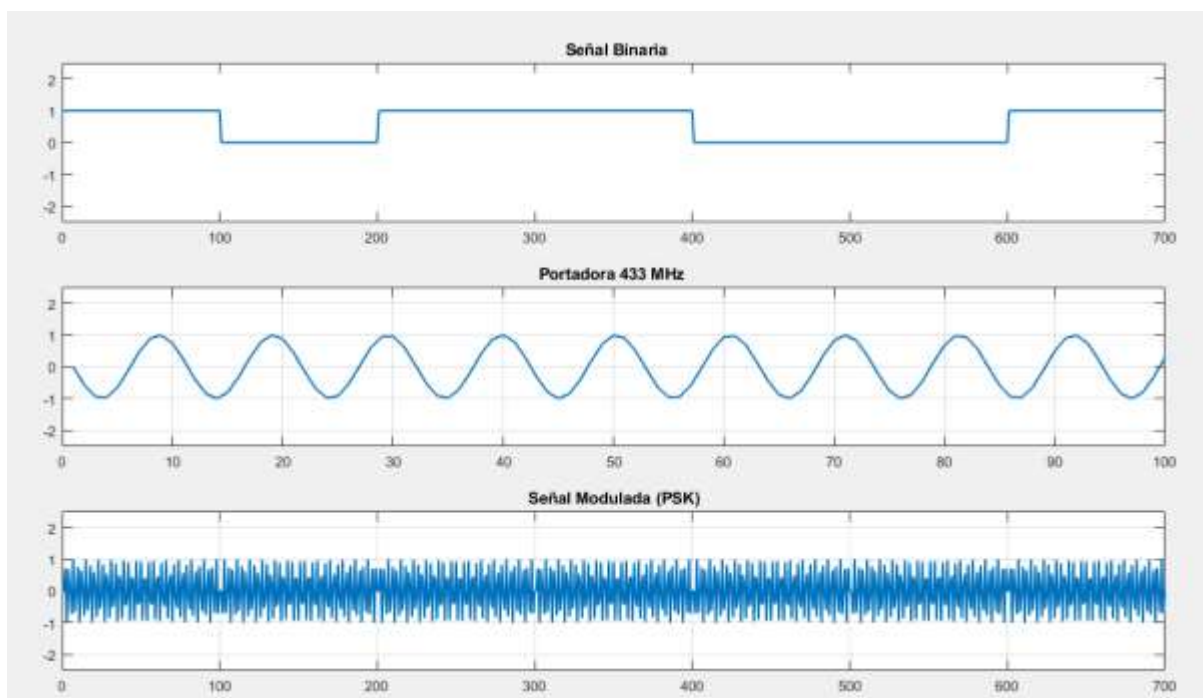
2.4.3 Análisis de la trama de datos de la Modulación PSK

La manera en cómo funcionan la modulación del tipo PSK es la siguiente, se tendrá el desplazamiento correspondiente a la fase de tal forma que si la señal binaria toma un valor de 0 este se tendrá un valor de salto de fase de 0° mientras que al llegar un 1 en la señal binaria provocará un salto de fase de 180° , por lo tanto, al tener dos puntos de modulación se la

denomina modulación BPSK. En la figura 3 se observa la señal binaria ubicada en la primera sección del gráfico en cuestión, por consiguiente, la onda senoidal con frecuencia de operación de 433 MHz que se ubica en la parte media del gráfico, finalmente la señal modulada resultante correspondiente a la mezcla entre la señal binaria y la portadora, donde se obtiene el resultado gráfico mostrado al final de la figura (Couch, Modulación PSK, 2015).

Figura 3

Modulación PSK



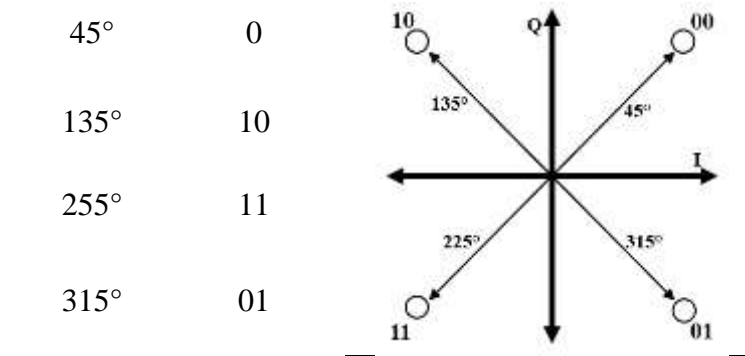
2.4.4 Análisis de la trama de datos de la Modulación QPSK

La modulación QPSK se encarga de representar una porción de datos a través de los bits que varían con respecto a los cambios de fase, en este caso, la modulación QPSK ocupa 4 cambios de fase y cada una de ellas se codifica con 2 bits (Verdecia Peña, 2018). A continuación, se muestra en la tabla 2 los cambios de fase y la cadena de bits:

Tabla 2

Cadena de bits, cambios de fase y constelación de la modulación QPSK

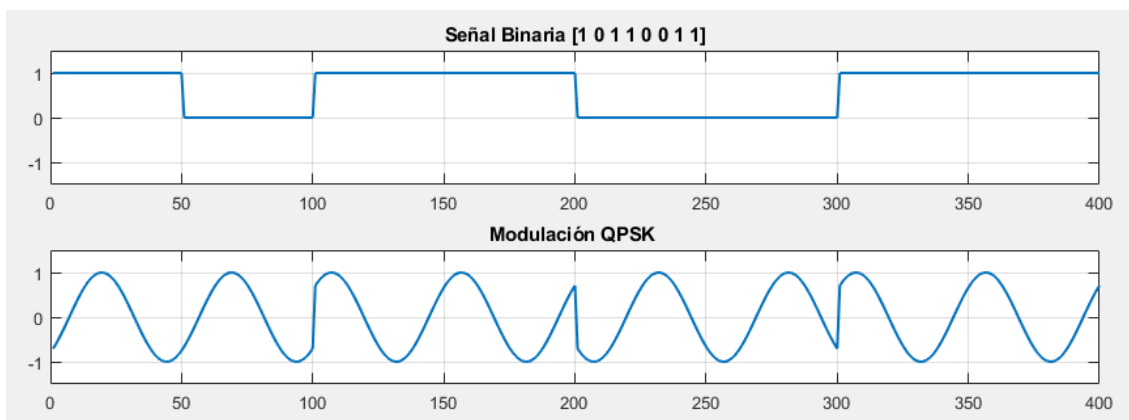
Desfase	Código	Diagrama de constelación
---------	--------	--------------------------



Tomando una señal binaria “10110011” y una señal portadora de frecuencia 433 MHz se procede a realizar una señal modulada QPSK y su diagrama de constelación el cual está representado en la siguiente figura 4.

Figura 4

Modulación QPSK

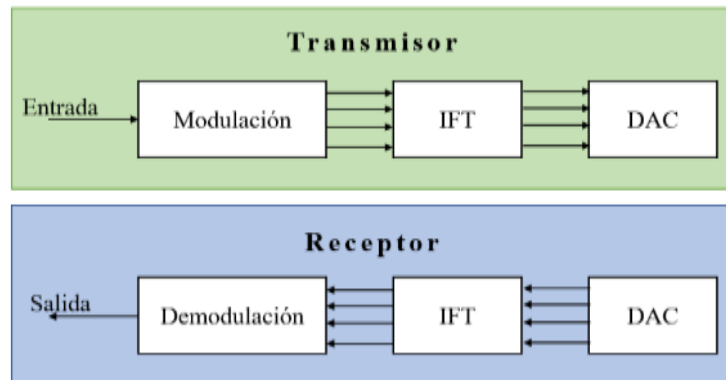


2.4.5 Análisis de la Modulación OFDM

La multiplexación por división ortogonal en frecuencia es un tipo de modulación que permite realizar una división de un canal de frecuencia en varias subportadoras con distintas bandas que serán usados para transmitir una porción de la información, una manera de explicar este proceso en un sistema OFDM de transmisión y recepción es dividir en 4 partes para cada sistema, la figura 5 representa un resumen de este proceso:

Figura 5

Modelo básico de un sistema OFDM de transmisión y recepción



Transmisor

Entrada de datos: Para una trama de datos se convertirá en una cantidad de palabras necesarias para la transmisión, si se toma la modulación QPSK se tomarán dos bits para cada palabra de dato.

Modulación: Cada palabra de datos se modula con una subportadora.

IFT: Se realiza una transformada inversa de Fourier con la finalidad de representar la señal OFDM en dominio temporal.

DAC: La finalidad de tener en la salida una IFT es proporcionar una señal analógica para transmitirla mediante radio o medio transmisor.

Receptor

ADC: La señal analógica que llega al receptor se convierte y pasa a ser digital.

FFT: La señal digital es convertida en una señal en dominio de la frecuencia con la transformada rápida de Fourier.

Demodulación: La señal demodulada proporciona la cantidad de palabras que el transmisor envió.

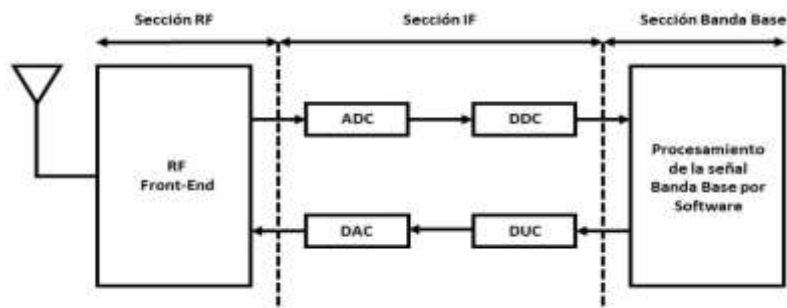
Salida de datos: El grupo de palabras obtenidas luego del proceso de modulación corresponde a la trama de datos.

2.5 Tecnología SDR

Estas tecnologías denominadas como radios definidos por software tuvieron sus inicios en los años 90, como parte de dispositivos ocupados en proyectos militares que facilitaban el proceso de digitalización de señales que finalmente se enviaban a un ordenador que culmina el procesamiento de la señal mediante el uso de softwares que permiten la interacción con equipos de hardware conectados al ordenador que operan como analizadores de señales (Navarro, Canto, & Poveda, 2018), en la figura 6 se analiza la estructura típica de diagramas de bloques de un equipo SDR y las especificaciones de las secciones se determinan en la tabla 3.

Figura 6

Diagrama de bloques de un equipo SDR



Nota: la imagen muestra una estructura típica de la tecnología SDR.

Tabla 3

Funciones de los diagramas de bloques presentes en un equipo SDR

Sección	Función
RF o de radiofrecuencia	➤ Se encarga de la transmisión y recepción de las señales RF
	➤ Realiza el paso de señal RF a señal IF
	➤ Ejecuta procesos de modulación o amplificación mediante LNA
IF o frecuencia intermedia	➤ Área que contiene convertidores (ADC/DAC) y (DDC/DUC)
	➤ Se realiza el paso de IF a banda base

	➤ Convertidor analógico digital (ADC)
	Convertidor digital analógico (DAC)
	➤ Convertidor descendente digital (DDC)
	➤ Convertidor ascendente digital (DDU)

	➤ En la transmisión, se realiza la extracción de la señal digitalizada en banda base para ser manipulada a través de la aplicación en el ordenador.
Banda Base	➤ En la recepción, se ejecuta un proceso inverso al anterior.

Nota: En la tabla 3 se puede observar una descripción breve de los bloques que intervienen en el funcionamiento de un equipo SDR.

2.5.1 Equipos tecnológicos de radio definido por software

La evolución de las tecnologías de comunicaciones de radio ha permitido a los desarrolladores crear dispositivos que agrupan un conjunto de características y funciones dejando atrás aquellas que se crearon para una función específica, de tal manera que el propio hardware compuesto por varios componentes electrónicos permita la interacción a nivel de software gracias a los convertidores, codificadores, decodificadores, moduladores de señal, filtros, memorias, entre otros que dispone el dispositivo, dando apertura a los equipos tecnológicos de radio definidos por softwares o SDR por sus siglas en inglés.

Estos dispositivos son útiles en escenarios donde se requiere un análisis de datos durante una comunicación inalámbrica, para esto es necesario que los equipos SDR sean compatibles con los diferentes programas que se encuentren disponibles en equipos de computación con el fin de asignarles tareas a nivel de programación o incluso reprogramarlo (López Cabrera Director et al., 2019).

A continuación, se presentan algunos de los equipos SDR actuales que se encuentran a la venta en el mercado.

2.5.1.1 HackRF One

Es un equipo que trabaja de modo bidireccional, es decir que permite la transmisión como la recepción de señales, pero no es posible realizar ambas acciones en un mismo tiempo. Forman parte de las nuevas tecnologías de dispositivos programables para las comunicaciones inalámbricas y es capaz de captar frecuencias en el rango de 1 MHz a 6 GHz. El HackRF One se programa para diversas simulaciones según sus fines, pero requiere estar conectado a una computadora debido a la falta de memoria interna. Esta conexión permite realizar correcciones al instante, lo que facilita su uso en el análisis de comunicaciones inalámbricas en tiempo real. A pesar de su dependencia de la computadora, la flexibilidad y capacidad de reprogramación lo convierten en una herramienta valiosa para la investigación y análisis que puede adaptarse a diferentes estándares, frecuencias y protocolos de comunicación. Ver figura 7.

Figura 7

HackRF One



2.5.1.2 RTL-SDR

Un dispositivo RTL-SDR como se puede observar en la figura 8, es empleado como receptor de frecuencias, posee una antena que permite captar señales de radio desde 500 KHz hasta 1750 MHz, cuenta con un puerto USB para la conexión directa hacia los ordenadores. También es compatible con smartphones, lo cual permite el uso de aplicaciones móviles

disponibles en la App store para el uso variado de sus funciones, cabe recalcar que el equipo no se desenvuelve como transmisor (PAZMIÑO, 2015).

Figura 8

RTL-SDR



2.5.1.3 NOOELEC NESDR SMART HF V4

Por otra parte, modelo NooELEC NESDR posee características físicas similares a una memoria USB, lo cual posibilita una conexión directa hacia un computador o incluso un Smartphone a través de un cable adaptador USB, este modelo SDR capta frecuencias desde 25 MHz hasta 1700 MHz mediante su antena incorporada y es compatible para varias aplicaciones y softwares de programación, además cuenta con la característica de poder ser reprogramado con otras tareas. En la Figura 9 se puede observar una imagen de referencia de este dispositivo.

Figura 9

NOOELEC NESDR SMART HF V4



Nota: Obtenido de (Ebay, s.f.)

2.5.1.4 Comparación de los equipos tecnológicos de Radio Definido por Software

Los equipos de radio definido por software antes mencionados serán medio principal que permitirá almacenar los datos de la información durante el proceso de comunicación inalámbrica del dispositivo IoT que se estudiará en este módulo educativo, para ello es necesario determinar las mejores opciones al momento adquirir el equipo en el mercado con la finalidad de escoger un SDR acorde a la frecuencia de operación del aparato comercial IoT y el prototipo que se desarrollará para la comparación al modelo comercial.

Como la propuesta necesita de ciertos requerimientos a nivel de hardware como que el transceiver captador de señal sea compatible con SDR y que la antena trabaje en un rango de 315MHz - 2.4GHz se procede a realizar un análisis comparativo de los periféricos captadores de señal, el cual se resumen en la tabla 4 dispuesta a continuación.

Tabla 4

Comparativa de los equipos de tecnología SDR

Características	HackRF One	RTL-SDR	NOOELEC NESDR SMART HF V4
Frecuencia de operación	1MHz - 6GHz	500KHz - 1750MHz	25MHz - 1700MHz
Modo de transmisión	Half dúplex	Simplex (Rx)	Simplex (Rx)
Compatibilidad	GNU Radio, SDR# y +	SDR#, RTL-433 HSDR, SDR-Radio, GQRX o SDR Touch en Android	SDR#, HSDR, Matlab y +
Conector de antena	SMA	SMA	SMA
Modo de alimentación	USB	USB	USB
Hardware	Código abierto	Código abierto	Código abierto
Costo	Módico	Bajo	Bajo
Compatibilidad con tecnologías inalámbricas	Total	Total	Total

A nivel general para el desarrollo de comunicaciones con equipos SDR existe una gran variedad de transceptores que permiten el tratamiento de las señales inalámbricas, es por ello que dependiendo de las necesidades del proyecto y gracias a la comparativa realizada en la tabla 4, se escogieron las siguientes tecnologías:

El RTL-SDR, este se convertirá en un detector para las frecuencias de 433 MHz y el responsable de captar la información que se evaluará a nivel de software, pese a sus limitaciones para transmitir o el rango de frecuencias de operación que puede captar se convertirá en una pieza clave al momento de hacer uso del software de programación RTL-433 quien va a posibilitar el análisis y detección de vulnerabilidades en base al análisis de protocolos que se podrían encontrar en el proceso, cabe recalcar que este análisis será posible gracias a la compatibilidad entre el hardware RTL-SDR y el software RTL-433.

Y el HackRF One quien sería el encargado de trabajar como atacante para las frecuencias de 2.4 GHz, también se podría considerar como un sustituto para las frecuencias de 433 MHz y operar como detector, sin embargo, no es compatible con el software RTL-433 quien permitiría el análisis de protocolos de comunicación por lo que no se podría usar dicho programa debido a su incompatibilidad. Sin embargo, será útil para ser un detector ante las frecuencias 2.4 GHz que emplea el dispositivo IoT al enlazarse a la red.

Cabe recalcar que los equipos se seleccionaron tomando en cuenta sus características, los buenos comentarios de la comunidad, por ser de fácil de configurar, por su amplia gama de información y ejercicios en la red, portabilidad, versatilidad y costos.

2.6 Internet de las Cosas

El termino Internet de las cosas, hace referencia al avance que se ha experimentado, en el desarrollo de las tecnologías basadas en la conexión de objetos a Internet, los cuales buscan intercambiar, agregar y procesar datos sobre el entorno físico donde se lleva a cabo la comunicación, con una mínima intervención humana. Este tipo de sistemas actúan de forma autónoma ante el reconocimiento de eventos o cambios. Por lo tanto, su finalidad es brindar una infraestructura que logre superar barreras entre objetos del medio físico y su representación en los sistemas de información (Barrio A. M., 2020).

Por otra parte, la Unión Internacional de Telecomunicaciones (ITU), define IoT como un objeto del mundo físico o del mundo digital, capaz de ser identificado e integrado en una red de comunicación. El termino IoT fue propuesto en 1999 por Kevin Ashton un pionero de la tecnología británica, en una presentación para Procter & Gamble donde describía a un sistema en el cual los objetos en el mundo físico podrían conectarse a internet a través de sensores (Barrio A. M., 2020).

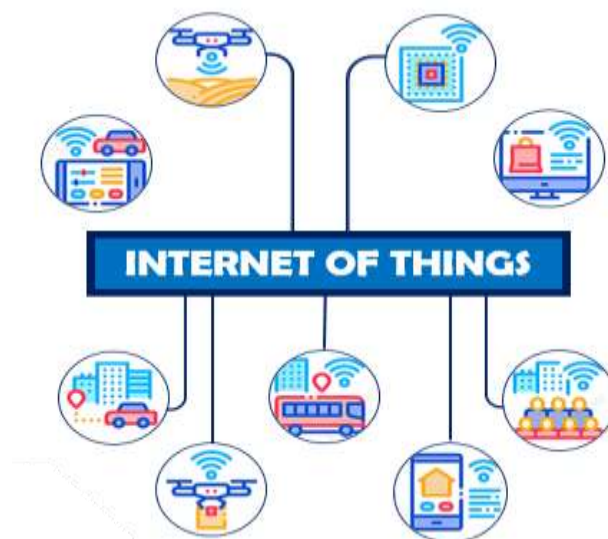
Cabe recalcar que los sistemas IoT junto a otras tecnologías como ciberseguridad, Big Data, Cloud computing y automatización se posicionaron como la base de la transformación digital. Por supuesto el uso de las tecnologías IoT ha permitido generar nuevos ingresos y reducir costos, además de contar con una mayor eficiencia en las cadenas de suministros y logísticas. Sin embargo, uno de los aportes importantes que ha brindado la IoT es la automatización de gran parte de las tareas que requerían una mayor intervención humana con el fin de simplificar procesos.

2.6.1 Aplicaciones IoT

El principal objetivo de IoT es la creación de entornos inteligentes y cosas conscientes para aplicaciones relacionadas con el clima, la alimentación, la energía, la movilidad, la sociedad digital y la salud (por ejemplo: transporte, productos, ciudades, edificios, zonas rurales, energía, salud, inteligente, etc.). Ver figura 10.

Figura 10

Campos de Internet de las Cosas



Nota: La imagen muestra los diferentes campos de aplicación en los que se desenvuelve la tecnología IoT.

2.6.2 Seguridad en sistemas de Internet de las Cosas

La seguridad en dispositivos IoT es una de las obligaciones que los fabricantes de los sistemas deben garantizar en cuanto a la gestión de datos, es decir este tipo de tecnologías deben implementar medidas de seguridad para dar protección a los datos contra los ciberataques y otros incidentes informáticos. Los dispositivos IoT no solo son víctimas de cibercriminales, sino también de vectores de ataques que pueden promover otros ataques cibernéticos, como el ocurrido en 2016 donde se efectuó un gran asalto de botnets a través de la toma de millones de electrodomésticos como grabadoras de video, cámaras de seguridad y otros dispositivos inteligentes que posteriormente fueron utilizados para ejecutar un ataque de Denegación de Servicios (DoS), este ataque afectó a varias plataformas web, incluyendo Amazon, Reddit, Spotify, CNN y Netflix. El secuestro de estos dispositivos se llevó a cabo debido a las vulnerabilidades que presentaban.

El internet de las cosas enfrenta un sinnúmero de amenazas de seguridad, en varias ocasiones para enfrentar dichos riesgos que afectan su funcionamiento en las diversas áreas de aplicación en las que se desenvuelve se debe realizar una gran inversión de recursos. Esta tecnología ayuda en la integración de las industrias en los procesos de presentación de servicios, también transmite, recopila, procesa y realiza la gestión de datos, al momento de usar plataformas que manejan grandes cantidades de datos personales de los usuarios pueden ser vulnerables debido a la importancia de dicha información.

La arquitectura de los dispositivos IoT busca garantizar la confidencialidad, validez, seguridad e integridad de la información. Las tecnologías como los sistemas de detección de intrusos, las PKI, los cortafuegos funcionan como medio de protección. El mecanismo de confidencialidad en los dispositivos IoT tiene como objetivo principal proteger el interés

público y la privacidad del usuario. Un ejemplo donde se observa el uso de estas funciones es en la obtención de información, el cual necesita un mecanismo de control para acceder a la capa perceptiva debido a la divulgación indebida de información de usuarios, lo cual no produce pérdidas, pero si genera un ataque de Sybil, DoS, entre otros, los ataques mencionados van a afectar la integridad y la validez de los datos. Los desafíos en la arquitectura IoT son la transmisión y el control masivo de datos.

2.7 Vulnerabilidades

A pesar de que existen tecnologías de protección de redes informáticas como antivirus y cortafuegos, con la frecuente aparición de los cibercriminales, los sistemas de redes se encuentran aún más expuestos por presentar vulnerabilidades de seguridad. En el campo de la seguridad de la información la vulnerabilidad se ha vuelto un tema primordial.

Hablar de vulnerabilidades hace referencia a las debilidades que presenta un sistema que permite a los cibercriminales comprometer la confidencialidad, integridad y disponibilidad de los servicios e información soportados. Adicional a ello las vulnerabilidades también pueden presentarse como resultado de una deficiencia en el diseño, implementación, operación o los controles internos en un proceso, lo cual provoca que el desarrollo no presente las mejores prácticas para proveer una buena seguridad. Otro de los factores es la limitación tecnológica, porque como es de conocimiento general, no existe un sistema totalmente seguro, también se debe a las reducciones de costes en el desarrollo de estos.

2.7.1 Vulnerabilidades en sistemas de Internet de las Cosas

Implementar una política de seguridad en la tecnología IoT puede resultar complicado debido a la falta de conocimiento, ya que podría afectar notoriamente a los consumidores, desarrolladores y fabricantes. La fundación Open Web Application Security Project's (OWASP) tiene como objetivo apoyar a organizaciones y empresas en el desarrollo, operación

y mantenimiento de aplicaciones confiables a nivel de seguridad, para promover el uso seguro de dichas tecnologías. La OWASP elaboro un listado del Top 10 de las vulnerabilidades presente en la IoT, según el informe de seguridad del año 2018 (OWASP, 2018). A continuación, se detallará cada una de las vulnerabilidades:

2.7.1.1 Debilidad en contraseñas

Esta es una de las vulnerabilidades más graves dentro de los sistemas IoT, debido a que es un blanco fácil de explotar, con los diferentes ataques. La solución a esta vulnerabilidad está en el uso de contraseñas únicas de manera que no esté embebida en el dispositivo.

2.7.1.2 Inseguridad en los servicios de red

Esta vulnerabilidad es común en dispositivos IoT, ya que se presenta en los servicios de red de comunicación que se encuentran al alcance de los usuarios.

Las causas por las que se presenta este tipo de vulnerabilidad son las siguientes:

- Utilización de protocolos obsoletos
- Dispositivos que permitan el uso de puertos abiertos
- Puertos expuestos a internet
- Vulnerabilidad ante ataques DoS
- No uso de dispositivos de protección
- Configuración de red erróneos

2.7.1.3 Interfaces inseguras en el ecosistema IoT

La configuración de las herramientas externas a dispositivos como servicios en la nube, API o interfaces web pueden ser inseguras, lo que afectaría directamente a los dispositivos. Las medidas más adecuadas para contrarrestar este tipo de problemas son filtrar entradas y salidas,

control de acceso a interfaces y asegurar la comunicación añadiendo algoritmos para la encriptación.

2.7.1.4 Falta de un mecanismo de actualización seguro

Este tipo de vulnerabilidad hace referencia a la falta de mecanismos de validación de las versiones de firmware en los dispositivos, lo que resulta inseguro debido a que no existe la autenticación o procedencia de dicho paquete antes de su instalación. por ello es necesario que exista siempre una revisión para comprobar la integridad del firmware que se va a instalar, para de esta manera evitar que versiones piratas puedan ser instaladas.

2.7.1.5 Uso de componentes desactualizados e inseguros

Los dispositivos podrían verse afectados por el uso de componentes de software y hardware obsoletos. Por lo general estos suelen utilizar librerías, componentes, sistemas operativos personalizados, por lo que es importante asegurar que dichas partes no contengan algún tipo de vulnerabilidad.

2.7.1.6 Insuficiente protección de privacidad

El manejo de datos almacenados en los dispositivos IoT es inseguro y suele realizarse sin necesidad de permisos. Una solución a esta vulnerabilidad es el establecimiento de una política para la manipulación de datos del cliente.

2.7.1.7 Falta de seguridad en el almacenamiento y transferencia de datos

Se debe utilizar algoritmos de cifrado para el manejo de datos. También se debe llevar un control de acceso a los mismos dentro del ecosistema IoT.

2.7.1.8 Gestión de dispositivos inadecuada

Los controles de seguridad en los dispositivos de producción son necesarios para llevar a cabo una buena gestión de actualizaciones y activos, políticas de desmantelamiento, monitorización de los sistemas y en la eliminación segura de los dispositivos.

2.7.1.9 Configuraciones predeterminadas inseguras

Las configuraciones por defecto de los dispositivos son inseguras, por lo que es recomendable establecer configuraciones para proteger el sistema mediante el uso de políticas de filtrado en conexiones y gestión de permisos.

2.7.1.10 Falta de seguridad física

En este tipo de vulnerabilidad no existen medidas de seguridad física ante un ataque de contacto directo. Por lo que el atacante podría acceder fácilmente a la información del dispositivo por medio de puertos, controles, interfaces, otorgando el acceso libre a los datos que podría extraer mediante diversas técnicas.

Los dispositivos que tienen este tipo de vulnerabilidad presentan las siguientes características:

- Acceso a datos a través de puertos externos (USB, serial, entre otros)
- Fácil desmontaje de los dispositivos y de sus componentes internos
- Falta de protección en la información almacenada
- Poseen puertos externos

2.8 Ataques a Internet de las Cosas

Con el paso de los años el aumento de dispositivos IoT ha sido considerable, el constante avance tecnológico ha significado mayor cantidad de dispositivos conectados a Internet, por lo que se pretende que en los próximos años esta cifra crecerá exponencialmente.

Los ciberdelincuentes aprovechan el auge de este tipo de tecnología para ejecutar ataques, basándose principalmente en las vulnerabilidades que encuentran, ya sea esta una

configuración de seguridad deficiente por parte de los usuarios. Comprometiendo no solo al equipo, sino también a otros equipos presentes en la red.

2.8.1 Ataques Pasivos

Algunos atacantes monitorean las redes de sus víctimas a través de la tecnología IoT, donde recopilan datos privados sin que el usuario note el más mínimo movimiento, extrayendo información importante como datos bancarios, credenciales de inicio de sesión, incluso el cibercriminal puede escuchar conversaciones, si las personas se encuentran en un lugar cercano al dispositivo interceptado.

Un ataque pasivo implica la interceptación del tráfico de la red para recopilar información confidencial, este se diferencia de los ataques activos, debido a que no pueden detectarse porque estos no alteran los datos interceptados, sin embargo, puede implicar el inicio de un ataque activo para un cibercriminal.

2.8.1.1 Replay

El ataque Replay o también denominado ataque de reproducción, se produce cuando el atacante intercepta y luego replica una transmisión de datos válidas para la red, esto se debe a que el atacante logro acceder a credenciales válidas para la red. Razón por la cual los protocolos de seguridad tratan a esta acción como una transmisión normal, sin que el intruso tenga problemas de realizar su cometido.

2.8.1.2 Sniffing

Es la técnica utilizada para capturar el tráfico generado en una red local, este logra recolectar información importante que puede servir para realizar otros ataques. Para realizar la captura del tráfico de la red se necesitan herramientas denominadas sniffers y analizadores de red. Un sniffer tiene la función de construir los paquetes de datos para observar sus cabeceras

y contenidos de forma estructuradas mostrando los valores de cada campo de forma ordenada y consistente, lo cual permite al administrador de una red, saber qué tipo de paquetes están siendo enviados y recibidos.

2.8.1.3 Wardriving

También conocido como driver de guerra, realiza una búsqueda de redes Wi-Fi abiertas o factibles a invasión realizada desde un vehículo en movimiento. Para cometer este acto además de necesitar un vehículo, es necesaria una placa Ethernet configurada para interceptar la lectura de paquetes de comunicación y una antena para ubicarla dentro o fuera del automóvil.

2.8.1.4 Snooping-Donwloading

Al igual que el sniffing obtiene la información sin modificarla, sin embargo, los procedimientos utilizados en este ataque son diferentes, puesto que además de realizar la interceptación del tráfico de red, el cibercriminal ingresa a los documentos, mensajes de correo electrónico para realizar un downloading, es decir crea una copia de los documentos en su propia computadora para posteriormente realizar un análisis de esta. Por otra parte, el snooping es un tipo de ataque realizado para el robo o espionaje de información.

2.8.1.5 Evil Twin

También conocido como gemelo malvado es un ataque donde el ciberdelincuente, crea una red Wi-Fi para que parezca legítima, con el fin de robar datos de las víctimas o realizar ataques Man in the middle. Una vez ejecutado el Evil twin, solo se necesitarán víctimas, que puedan conectarse a dicha red de internet, para llevar a cabo el cometido del atacante.

2.8.2 Ataques Activos

Estos ataques implican la modificación en el flujo de datos o la creación de datos falsos para cambiar el contenido de la información que se está recibiendo. El objetivo de este ataque

es pretender colapsar los servicios que puede prestar la red. Debido a la capacidad de los daños que pueden causar a nivel de hardware y software, este tipo de ataques son difíciles de prevenir. Sin embargo, al momento de que un usuario malicioso intenta afectar la información almacenada en el sistema, la víctima siempre es notificada.

2.8.2.1 Denegación De Servicio

Es un tipo de ataque cibernético donde el atacante tiene como finalidad que un ordenador u otro aparato no esté disponible para los usuarios, afectando su funcionamiento normal, los ataques de denegación de servicios sobrecargan el ordenador con solicitudes hasta que el tráfico normal no puede ser procesado. Un ataque DoS puede causar interrupciones significativas en servicios en línea, afectando la disponibilidad y rendimiento de sitios web, servicios de correo electrónico, aplicaciones y otros sistemas en Internet.

2.8.2.2 Inhibidor de Señal

Un inhibidor de frecuencia realiza ataques de denegación de servicio, en este caso dominan el espectro para que este se vuelva inaccesible. Un inhibidor puede provocar la interrupción de las comunicaciones inalámbricas entre dispositivos como sistemas de seguridad, Bluetooth, teléfonos móviles, redes Wi-Fi, entre otros. Los inhibidores pueden desconectar servicios al bloquear las señales de frecuencias específicas provocando graves consecuencia en situaciones de emergencia.

2.8.2.3 Fuerza Bruta

El objetivo de este ataque es descubrir claves de cifrado o contraseñas, utilizando todas las combinaciones posibles hasta descubrir la contraseña correcta. En otras palabras, se trata de un método de prueba y error empleado para decodificar información de carácter personal. Realizar un ataque de fuerza bruta es relativamente fácil, aunque a la hora de ejecutarlos son

muy lentos, debido a que se debe efectuar pruebas con toda la cadena posible de caracteres para cumplir su objetivo.

2.8.2.4 Spoofing

Un ataque spoofing es un tipo de estafa que se origina cuando un atacante se hace pasar por un remitente de confianza para engañar a sus víctimas con el fin de acceder a su información, este ataque puede realizarse mediante llamadas telefónicas, sitios web, direcciones IP, correos electrónico o textos. También denominado suplantación de identidad, tiene como propósito usurpar la información personal, propagar malware mediante archivos infectados, saltarse controles de acceso a una red o intentar robar bienes.

2.8.2.5 Ramsonware

Es un tipo de malware de rescate que impide a los usuarios acceder a su sistema o archivos personales, con el fin de exigir un pago de rescate para volver a acceder a dicha información. Este ataque puede infectar un ordenador de muchas formas como por ejemplo a través de un spam malicioso que son mensajes no solicitados que utilizan el correo electrónico para realizar envíos del malware.

2.9 Módulo de control relé basados en Radiofrecuencia

Los módulos de control relé son una configuración de circuitos y microcontroladores aplicados para interactuar con ellos a través de dispositivos por medio de controles remotos de bajas frecuencias o las aplicaciones disponibles para Smartphones que requieren enlazarse a la red de internet con la finalidad de tener acceso al control del dispositivo, presentando una interfaz gráfica e interactiva que contienen las opciones que permiten el cambio del status de los puertos.

Su función principal es permitir el paso de la energía eléctrica para energizar algún tipo de carga, el dispositivo se implementa entre la fuente de energía alterna y la carga, siendo un

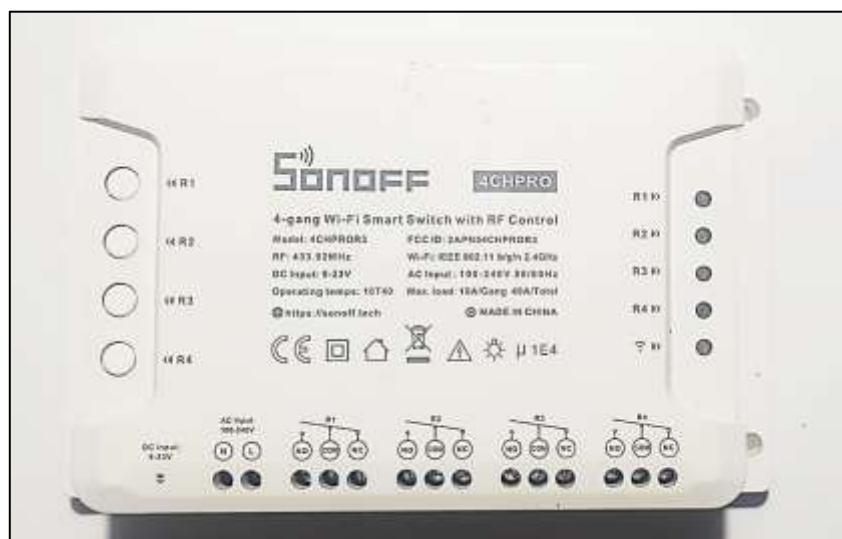
dispositivo que se coloca como intermediario para cerrar o abrir el circuito con el fin de ser controlados sin necesidad de estar físicamente, por otra parte, en este apartado se comparten las características y modelos que actualmente existen en el mercado.

2.9.1 Smart switch relé SONOFF 4CHPRO

Es un dispositivo que permite el control de artefactos o equipos conectados a la red eléctrica del hogar al ser implementados como intermediario entre la toma de corriente de la casa y el equipo que requiere ser controlado lo cual es posible a través de un control externo que transmite una señal de 433 MHz al Smart switch con una orden establecida que habilita o deshabilita el paso de la energía o también, a través de aplicaciones disponibles para teléfonos móviles donde se puede tener un control del dispositivo y establecer tareas para su activación en determinado tiempo basándose en el estándar IEEE 802.11 b/n/g empleado en Wi-Fi en un rango de operación de 2.4 GHz, en la figura 11 se puede observar una imagen de referencia de este dispositivo.

Figura 11

Smart switch Relé SONOFF 4CHPRO



2.9.2 Módulo relé Sra RF programable

Este modelo es fabricado por la marca desarrolladora de módulos electrónicos SUTAGAO de procedencia colombiana, consta de una parte transmisora para el control y una parte receptora que recibe las ordenes que determina el momento para habilitar o deshabilitar sus puertos con un alcance de 50 metros en lugares abiertos, entre los modelos que han desarrollado van desde dos, cuatro y ocho puertos, el siguiente modelo a analizar es el que dispone cuatro puertos para la activación/desactivación de estos. En la figura 12 se puede observar el modelo antes descrito.

Figura 12

Módulo Relé RF Programable



Nota: Obtenido de (Vistrónica, s.f.)

Para la transmisión y recepción de la señal se ocuparon módulos de radiofrecuencia del modelo FS1000A y XY-MK-5V, posee dos relés con borneras para facilitar las instalaciones del cableado, la placa ocupa un microcontrolador ATmega 328P y puede ser programable para variar los modos de operación en base a los requisitos propuestos por el usuario, para ello es necesario un Arduino con encapsulado extraíble para cargar los modos de uso mediante una

programación facilitada por los fabricantes del modelo, entre los modos que dispone este dispositivo están:

- Primer modo: apagado o desactivación de los relés.
- Segundo modo: en este caso los botones funcionan como pulsador, es decir, el relé estará activo si se mantiene presionado el botón y se desactivará si se deja de pulsar.
- Tercer modo: la función de interruptor se basa en que al pulsar el botón el relé se mantendrá activo hasta el momento en que se vuelva a pulsar el botón, en este punto el relé se desactivará.
- Cuarto modo: funciona como un interruptor selectivo, es decir, si en primera instancia se pulsa A en el control, el transmisor activará el relé A y si en otro momento se pulsa B en el transmisor, en el receptor se desactivará el puerto A para activarse el B.

Las características técnicas de este modelo se describen en la tabla 5:

Tabla 5

Características técnicas de un Módulo Relé Sra Rf Programable

Características	
Voltaje de entrada	12V DC
decodificador	Pt2272-M6
Alcance	50m. Línea de vista
Consumo máximo de corriente	150mA. Todos los relés activos
Modos de operación	Cuatro, se puede reprogramar
Receptor de radiofrecuencia	315 MHz
Pines de programación	RX, TX, reset, VCC, GND.
Microcontrolador	Atmega 328UP
Temperatura	-20°C -+100°C
Control	2 puertos

Nota: En la tabla 5 se describen las especificaciones técnicas de un equipo Relé Sra Rf Programable. Elaborado por el autor.

2.9.3 Switch Relé AK-RK02E

El modelo AK-RK02E de la marca Fushionsea posee dos relés en el receptor y su control remoto dispone de dos botones, es decir que son capaces de controlar dos puertos en la salida, los módulos de radiofrecuencia ocupados para la transmisión y recepción son los módulos FS1000A y XY-MK-5V, la frecuencia de trabajo de estos dispositivos es de 433 MHz, considerando una línea de vista despejada se tiene un alcance de 50 metros. Ver figura 13.

Figura 13

Switch relé AK-RK02E



Nota: Obtenido de (Amazon, s.f.)

Dispone de cuatro modos de trabajo que pueden cambiarse a través de un botón ubicado en el receptor ('Model') este permite variar las funciones de trabajo del relé, a continuación, se describen los cuatro modos que se pueden utilizar en estos dispositivos:

- Primer modo: presionando el botón Model una vez, se activa el modo pulsador, es decir que el relé se va a mantener activado si en el control se mantiene pulsado el botón que lo acciona y el relé se desactivará en el momento que se deje de pulsar.
- Segundo modo: presionando el botón Model dos veces, se activa el modo cierre de cierre, es decir que si se activa un relé se podrá desactivar haciendo uso de alguno de los dos botones.
- Tercer modo: presionando el botón Model tres veces, se habilita la función de interruptor que permite mantener el relé activo en el caso de presionar una vez y se desactivará si se presiona el mismo botón.
- Cuarto modo: presionando el botón Model cuatro veces, se presenta la función de retraso de 5 segundos, esta permite mantener el relé activo durante 5 segundos al presionar el botón que lo activa a través del control.

Las características técnicas de este modelo se describen en la tabla 6.

Tabla 6

Características del dispositivo Switch Relé AK-RK02E

Características	
Voltaje de entrada	12V DC
Alcance	50m. Línea de vista
Consumo máximo de corriente	150mA. Todos los relés activos
Modos de operación	Cuatro, se puede reprogramar
Receptor de radiofrecuencia	433 MHz
Sensibilidad del receptor	>105 dBm
Pines de programación	RX, TX, reset, VCC, GND.
Temperatura	-40°C - +80°C
Control	2 puertos

2.9.4 Smart Power Strip

A diferencia de los modelos anteriores, el dispositivo Smart Power Strip como se observa en la figura 14, es un dispositivo que se encarga de distribuir energía eléctrica a través de sus puertos los cuales se conectan directamente con conectores para tomas de corriente del tipo B y puertos USB, por otra parte, se incluye el control por asistente de voz y a través de una aplicación para Smartphoness que necesariamente debe estar conectada a una red Wi-Fi para vincular el dispositivo IoT con el teléfono móvil quien tendrá el control del mismo.

Figura 14

Smart Power Strip



Nota: Obtenido de (Amazon, s.f.)

No se puede manipular el estado de los puertos por medio de controles remotos ya que fue diseñado para trabajar bajo una conexión Wi-Fi de 2.4 GHz, por lo tanto, no es posible interactuar mediante un control de frecuencias de 4.33 MHz, ocupa dimensiones reducidas, el dispositivo se energiza mediante un cable conector para entradas tipo B, a parte de las características ya mencionadas, la tabla 7 busca ampliar las características y especificaciones técnicas establecidas por el fabricante.

Tabla 7*Dispositivo Smart Power Strip*

Características	
Voltaje de entrada	12V DC
Conexión	Vía Wi-Fi 2.4 GHz
Salida de corriente	hasta 3.1 A
Modos de operación	mediante aplicación móvil, todas las salidas activas, tiempos establecidos.
Sensibilidad del receptor	>105 dBm
Programación del tiempo	permitido
Temperatura	-10°C - +40°C
Salidas	3 tipo B, 3 tipo USB.

2.9.5 Comparación de los dispositivos comerciales

Los dispositivos IoT con tecnología Wi-Fi que se encuentran en el mercado, poseen características que pueden variar como las configuraciones de circuitos para radiofrecuencias o Wi-Fi, seguridad, alcance, estructura física, capacidad, entre otras alternativas que lo definen como una buena opción para un caso de estudio.

De este modo se considera el dispositivo Switch Relé Wi-Fi de la marca Sonoff para el análisis de las vulnerabilidades de este, tomando en consideración los componentes ocupados en la placa PCB como el módulo ESP para el Wi-Fi, los microcontroladores que almacenan las tareas definidas desde fábrica, alcance, forma física y tecnologías empleadas en el modelo comercial, con la finalidad de comparar con el modelo del prototipo que se pretende desarrollar.

A pesar de la existencia de varios dispositivos del internet de las cosas en el mercado, se toma en consideración este tipo de modelos tomando en cuenta el lugar donde comúnmente son implementados, como oficinas, empresas, hospitales, escuelas, entre otras aplicaciones que

este puede cubrir en dependencia del lugar donde trabajarán, siendo un blanco fácil para iniciar un análisis de las vulnerabilidades que podrían detectarse en el acto.

2.10 Componentes para ejemplificar el diseño del Prototipo

Para el desarrollo del prototipo es necesario definir con que piezas y componentes serán los más apropiados para su construcción donde es necesario contar con las tarjetas físicas que almacenen instrucciones y lógicas de programación, módulos para la transmisión y recepción, otros componentes como pulsadores, medios conductores y las bases para el soporte de todas las piezas que conforman el diseño.

De esta manera, se toma en consideración la búsqueda de los elementos del circuito para determinar las mejores opciones entre las variedades de piezas que realizan las mismas acciones, pero con diferentes características como la cantidad de pin de los módulos, tamaño, especificaciones técnicas o precios.

2.10.1 Tarjetas para procesos físicos de Código Abierto

Las tarjetas programables o placas de desarrollo de código abierto son dispositivos conformados por un microcontrolador reprogramable, donde se puede escribir instrucciones en un lenguaje de programación determinado, para luego ser ejecutadas por el microcontrolador. Este tipo de herramientas cuentan con entradas y salidas digitales o analógicas para permitir la comunicación con los sensores externos utilizadas en el campo de la ingeniería para el diseño y el desarrollo proyectos electrónicos.

2.10.1.1 Placas Arduino

Las tarjetas programables Arduino se encuentran entre las placas de código abierto muy usadas en la actualidad para la solución de proyectos bajo un contexto de programación para su operatividad, toda lógica de programación para estas tarjetas se realiza en la plataforma

Arduino IDE, existen varios modelos con diferentes dimensiones, características y números de pines analógicos y digitales, en este caso de estudio se analizarán los siguientes modelos:

Arduino Uno

Este modelo forma parte de las placas que más se comercializan en el mercado siendo una de las más útiles para iniciarse en la programación de estos, consta de un microcontrolador ATmega328P y se alimenta a través de un cable con conectores USB tipo A y tipo B, cuenta con 6 pines para entradas analógicas y 14 para digitales (ARDUINO UNO R3, 2021). Ver figura 15.

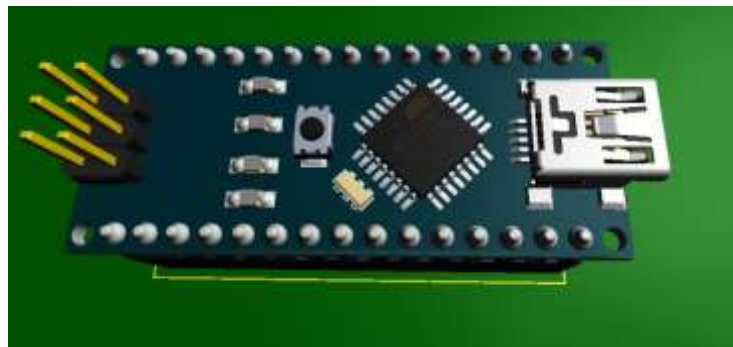
Figura 15

Arduino UNO

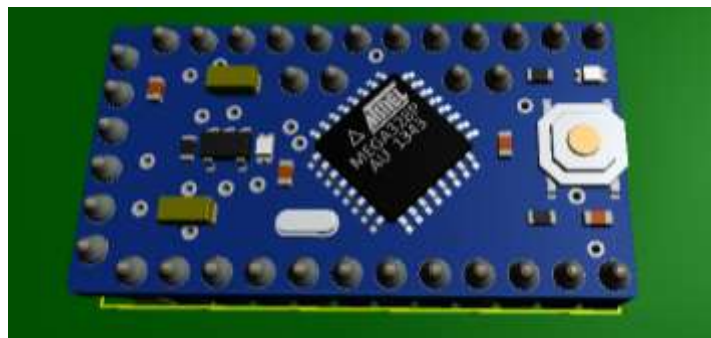


Arduino Nano

Este modelo a diferencia del anterior posee dimensiones más reducidas permitiendo ser accesible para una conexión directa en placas de pruebas Protoboard, el número de pines digitales se mantiene en 14 y 8 para la serie de pines analógicos, posee un microcontrolador modelo ATmega328 y la placa se alimenta a través de un cable USB mini-B (NANO, 2021). Ver figura 16.

Figura 16*Arduino NANO***Arduino Pro Mini**

Este modelo a diferencia del anterior posee dimensiones más reducidas permitiendo ser accesible para áreas pequeñas, dispone de 6 entradas analógicas y 14 pines destinados para entradas o salidas digitales, requiere de un conversor serial USB a serial (TTL) que permita grabar los códigos de programación por medio de un computador. Ver figura 17.

Figura 17*Arduino Pro Mini*

Especificaciones De Las Tarjetas Arduino

Estas tarjetas para procesos físicos de código abierto poseen una serie de características y especificaciones dadas por los fabricantes las cuales se resumen en la siguiente tabla 8 que busca comparar las mismas características en las placas antes vistas.

Tabla 8*Datos y especificaciones del fabricante de las tarjetas Arduino*

Especificaciones	Arduino UNO	Arduino Nano	Arduino Pro Mini
Microcontrolador	ATmega328P	ATmega328	ATmega328P
conector para carga de códigos	Conector USB tipo B	Conector Mini-USB	Coversor USB a Serial
Puertos Análogos	6	8	8
Puertos digitales Entradas/Salidas	14 (6 para salida PWM)	14 (6 para salida PWM)	14 (6 para salida PWM)
Memoria/SRAM/EPROM	32 KB/ 2 KB/ 1 Kbytes	16 KB/ 1KB/ 512 bytes	32 KB/ 2 KB/ 1 Kbytes
Frecuencia	16 MHz	16 MHz	16 MHz
Dimensiones (mm)	53x68	18x45	
Tensión de alimentación	de 5 a 12 V	de 5 a 12 V	de 5 a 12 V
Temperatura de operación	de -40 a 85 °C	de -40 a 85 °C	de -40 a 105 °C

2.10.1.2 Microcontroladores PIC

El microcontrolador PIC es un encapsulado que contiene un pequeño chip y aprovecha la tecnología de circuitos integrados programables (PIC) que a su vez, requieren de un medio que facilite su uso en cuanto a la implementación que se le pueda dar al considerar sus dimensiones tan reducidas, para ello emplea un encapsulado que dispone de pines que permite enlazarse directamente con las entradas/salidas del chip interno, facilitando la manera de implementar el microcontrolador en proyectos con tarjetas PCB o protoboards.

Los microcontroladores de este tipo requieren de la asignación de tareas que son los programas que se cargan en PIC, para ello se necesita de un software de programación que permita el desarrollo de las instrucciones que se ejecutarán al momento de implementarlos en el circuito, entre ellos están MPLab, PICKit 4, WinPic 800, PICAT 1.25, entre otros.

Sin embargo, es necesario un dispositivo externo que permita la conexión mediante un cable conectado en el ordenador que ejecuta el software donde se desarrolló el código de programación y el dispositivo que sirve como puente para llevar el programa que se grabará en

el PIC. A continuación, se describen tres microcontroladores que se encuentran disponibles en el mercado.

PIC16F84A

Este modelo de microcontroladores cuenta con 18 pines, 22.99 x 6.60 mm de largo y de ancho, se alimenta con 5V DC mediante el pin VDD, su punto a tierra mediante VSS, el pin MCLR para reinicio, y los puertos de RA0 hasta RA4 y RB0 hasta RB7 ya sean para entradas o salidas digitales, no posee una sección referente a conversión de analógico digital, por lo que no permite el libre uso de sensores analógicos (*PIC16F84A Data Sheet*, 2001). Ver figura 18.

Figura 18

Microcontrolador PIC16F84A



Nota: Obtenido de (MICROSIDE, s.f.)

A diferencia del anterior la cantidad de pines aumentó a 40, 52.45 x 14.22 mm de largo y de ancho, se alimenta con 4 a 5.5 VDC mediante el pin VDD, 33 de los pines son empleados para entradas o salidas digitales y si posee una sección para la conversión de analógico digital por lo que sí es posible interactuar con sensores analógicos (*PIC16F87XA Data Sheet*, 2003). Ver figura 19.

Figura 19

Microcontrolador PIC16F877A



Nota: Obtenido de (Ja-Bots.com, s.f.)

PIC18F452

Este microcontrolador cuenta con la misma cantidad de pines que el anterior y ocupa las mismas dimensiones largo y de ancho, se alimenta con 3.3 a 5.5 VDC mediante el pin VDD y contiene conversores analógicos digitales que posibilitan el uso de sensores analógicos conectados en el microcontrolador (*PIC18FXX2 Data Sheet, 2006*). Ver figura 20.

Figura 20

Microcontrolador PIC18F452



Nota: Obtenido de (3DBOTS, s.f.)

Especificaciones de los microcontroladores PIC

Las características mencionadas solo son una parte de las que cuentan estos circuitos integrados, por lo tanto. Con la tabla 9 se pretende ampliar los datos técnicos y más esenciales de estos modelos de PIC.

Tabla 9*Especificaciones de los microcontroladores PIC*

ESPECIFICACIONES	PIC16F84A	PIC16F877A	PIC18F452
Memoria Flash (KB)	1	8	32
Memoria RAM (Bytes)	68	368	1.5K
Memoria EEPROM (Bytes)	64	256	256
Voltaje de operación (VDC)	5	4-5.5	3.3-5.5
Frecuencia de operación (MHz)	20	20	40
Cantidad de pines	18	40	40
Puertos I/O	13	33	34
Dimensiones largo x ancho (mm)	22.99 x 6.60	40, 52.45 x 14.22	40, 52.45 x 14.22
Temp. Max.	+70°C	+70°C	+85°C
Comunicación serial	--	MSSP	MSSP, Addressable, USART
Instrucciones	35	35	75
ADC	no	si	si

2.10.1.3 Raspberry Pi

Las placas de la marca Raspberry Pi, se basan en el funcionamiento similar a los de una computadora con pequeñas dimensiones y de precios bajos que ocupa el sistema operativo de Linux, es una tarjeta programable que permite almacenar instrucciones o programas y dispone de varias características que dependen del modelo o versión, como las entradas o salidas de la placa, tamaños de memoria, puertos USB o entradas HDMI.

Inicialmente fue desarrollado con el objetivo de impulsar y fortalecer el crecimiento de la informática, donde empezó a ganar popularidad entre aficionados que se enfocan en el desarrollo electrónico, lo cual ha llevado a los desarrolladores a actualizar sus tarjetas implementando mejoras y características, de este modo se describen tres modelos de versiones que han surgido con el pasar de los años en relación con su hoja técnica de datos.

Raspberry Pi 2 model B

Entre las características físicas de este modelo, se encuentra la disponibilidad de puertos implementados en la placa como los puertos USB, los 40 pines GPIO, Módulos para la conexión vía Ethernet, salidas para la implementación de cámaras externas y displays, puerto de tarjeta Micro USB y salidas HDMI (Raspberry Pi 2 Model B, 2017). Ver figura 21.

Figura 21

Raspberry Pi 2 Modelo B



Nota: Obtenido de (Raspberry Pi, s.f.)

Raspberry Pi 3 model B+

El siguiente es una evolución del anterior modelo mencionado, implementando mejoras en las velocidades del procesador, los métodos de conexión y conservando las salidas disponibles para la implementación de cámaras o displays externos y disponibilidad de pines GPIO (RaspberryPi3ModelB+, 2018). Ver figura 22.

Figura 22

Raspberry Pi 3 model B+



Nota: Obtenido de (Raspberry Pi, s.f.)

Raspberry Pi 4 model B

Esta versión amplió la capacidad de memoria en 1,2 y 4 GB, mejoró su velocidad de operación, conservó las salidas de las cámaras y los displays externos, los 40 pines GPIO, mejoró su conectividad e implementó entradas USB 2.0 y 3.0 respectivamente (*Raspberry Pi 4 Computer Model B*, 2019). Ver figura 23.

Figura 23

Raspberry Pi 4 model B



Nota: Obtenido de (Element14, s.f.)

Especificaciones De Módulos Raspberry Pi

Considerando otros parámetros importantes sobre las versiones de tarjetas Raspberry mencionadas anteriormente, se presentan los siguientes datos técnicos más esenciales de cada una de ellas a través de la tabla 10.

Tabla 10

Especificaciones de los módulos Raspberry Pi

ESPECIFICACIONES	Raspberry Pi 2 Model B	Raspberry Pi 3 Model B+	Raspberry Pi 4 Model B
Memoria	1 GB LPDDR2	1 GB LPDDR2 SDRAM	1 GB/2 GB/4 GB LPDDR4
Puerto Micro USB	Si	Si	Si
Alimentación/conector USB-C	5 VDC/2.5 A	5 VDC/2.5 A	5 VDC/3 A
Alimentación/GPIO header	3.3 - 5 VDC	3.3 - 5 VDC	3.3 - 5 VDC
Alimentación/Ethernet	Power Over Ethernet	Power Over Ethernet	Power Over Ethernet
Procesador	Broadcom BCM2836 SoC/Cortex-A7/ 64-bit SoC 900MHz	Broadcom BCM2837B0/Cortex- A53/ 64-bit SoC 1.4 GHz	Broadcom BCM2711/quad- core Cortex-A72/ 64-bit SoC 1.5 GHz
Cantidad de pines GPIO	40	40	40
Salida video y sonido	Conector HDMI/MIPI DSI para display/MIPI CSI para cámara/conector Jack audio	Conector HDMI/MIPI DSI para display/MIPI CSI para cámara/conector Jack audio	2 conector Micro HDMI/2 MIPI DSI para display/2 MIPI CSI para cámara/conector Jack audio
Dimensiones (mm)	85 x 56	85 x 56	85 x 56
Temp. Max.	+50°C	+50°C	+50°C
Conexión	10/100 BaseT Ethernet/4 puertos USB 2.0	2.5-5 GHz IEEE 802.11 b/g/n/ac Wireless LAN/Bluetooth/4 puertos USB 2.0	2.5-5 GHz IEEE 802.11 b/g/n/ac Wireless LAN/Bluetooth/Gb Ethernet/2 puertos USB 2.0/2 puertos USB 3.0
Fecha de lanzamiento	febrero/2015	marzo/2015	junio/2019

2.10.1.4 Comparación de las tarjetas en base a la aplicación del proyecto

La amplia gama de tarjetas físicas de código abierto que disponen los fabricantes como Arduino, microcontroladores PIC o Raspberry Pi han permitido adecuar los proyectos en base a los requerimientos que este solicite, como el caso de los modelos de Raspberry Pi que poseen módulos de conexión como el HDMI, puertos USB, Ethernet, Bluetooth, Wi-Fi, GPIO que permite la comunicación con otros equipos o dispositivos, esto la convierte en una tarjeta robusta con una variedad de puertos que en el proyecto no tendrían una función en especial.

En cuanto a los microcontroladores PIC, presentan dimensiones reducidas, fácil de programar, amplios recursos en Internet, bajo coste y variedad de modelos que serían favorables para el diseño final del proyecto, sin embargo. Carece de elementos importantes como resistencias, bobinas, capacitores, osciladores, entre otros arreglos de circuitos que contribuyen en el filtrado de las señales, sincronización, conversión o amplificación de señales.

Por otra parte, los modelos de tarjetas Arduino se caracterizan por ser compactos con una amplia gama de modelos y dimensiones, contiene arreglos de circuitos que contribuyen en el filtrado de las señales, conversión o amplificación de señales, son fáciles de programar, existe una gran cantidad de información, ejemplos y sus costos dependen del modelo de la tarjeta.

De este modo, la tarjeta Arduino modelo Nano se convierte en una gran opción gracias al reducido espacio que ocupa, su facilidad de implementación, bajo costo, bajo consumo y la disponibilidad de pines que serán necesarios para interactuar con el resto de los elementos que se colocarán en la placa del prototipo destinado para la distribución de energía eléctrica controlada por radiofrecuencia y tecnología Wi-Fi.

2.10.2 Módulos de comunicación por frecuencias de radio

Los módulos de comunicación por radiofrecuencias son arreglos de circuitos capacitivos, inductivos, resistivos, osciladores y otros elementos que en conjunto forman un pequeño dispositivo para la recepción y para la transmisión de datos a nivel de radiofrecuencia que está diseñado para operar en rangos específicos, sin embargo, existen módulos que permiten el cambio de la frecuencia de operación mediante el giro de una pequeña bobina incorporada en la PCB del módulo, permitiendo variar entre las frecuencias establecidas en la hoja del fabricante.

Estos módulos de RF son elementos de gran utilidad si se configuran y conectan a través de microcontroladores capaces de ejecutar tareas específicas, entre ellas la de permitir que se establezca una comunicación entre ambos componentes por medio del aire como el medio para la propagación de la señal y va desde 20 a 100 metros con vista libre. A continuación, se mencionan algunos de los módulos RF.

2.10.2.1 Módulo TX/RX (FS1000A/XY-MK-5V)

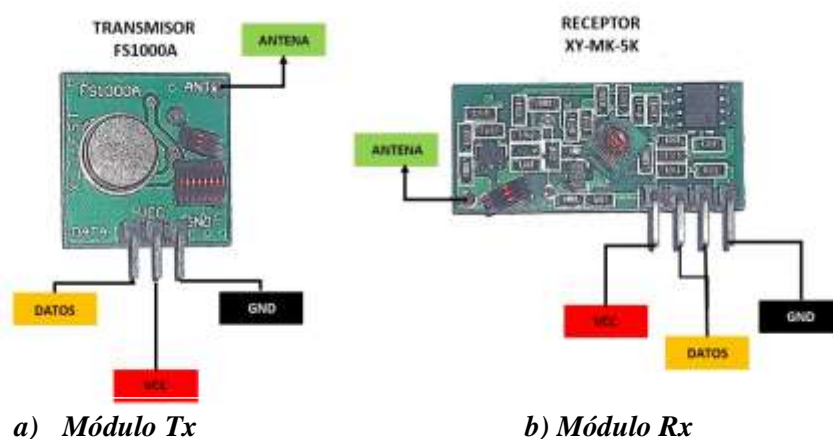
Es un arreglo de circuitos eléctricos que se divide en dos partes, transmisor (ver figura 24.a) y receptor (ver figura 24.b) los cuales operan a 433 MHz para la transmisión de datos a través de sus antenas incorporadas en ambas partes a una velocidad de 1 a 10 Kbps, emplea la modulación por desplazamiento de amplitud (ASK) o también conocida como modulación OOK, sin embargo, es posible emplear otro tipo de modulación como la FSK o PSK al considerar una programación que permita dicha acción. Es importante señalar que ambas partes se alimentan con tensiones diferentes, es decir, la parte transmisora soporta tensiones desde 3.3 V y 12 V, mientras que la parte receptora se alimenta únicamente con 5 V.

En cuanto a la disponibilidad de pines, el modelo FS1000A consta de 3 pines establecidos para datos, VCC y GND respectivamente, y es a través del pin de datos que se

conecta a la placa Arduino por donde se emiten las órdenes y la información se propaga a nivel de radiofrecuencias a través de la antena. El modelo XY-MK-5V que opera como receptor, consta de 4 pines (VCC, 2 para datos y GND), cabe recalcar que este tipo de módulos tiene igual salida de información en ambos pines de datos. Tampoco dispone de una antena, sin embargo, existe la posibilidad de incorporarla (Sertronics, 2014).

Figura 24

Módulo de Radiofrecuencia 433MHz



2.10.2.2 Módulo TX/RX (STX882/ SRX882)

Este modelo de circuitos destinados para transmisiones inalámbricas por RF es compatible para implementarlos con tarjetas para procesos físicos de código abierto, se compone por un módulo enfocado para la transmisión y otro módulo encargado a la recepción de los datos que trabajan en los rangos de frecuencia 315 a 433 MHz con un alcance de 100 m con línea de vista despejada.

Son empleados para proyectos electrónicos que requieren el envío y la recepción de medidas o lecturas tomadas a través de sensores u otro medio que provee los datos que son tratados por algún microcontrolador configurado de tal manera que exista una comunicación entre el módulo de transmisión STX822 y el de recepción SRX822. Ver figura 25.

Figura 25*Módulo STX882/SRX882*

Nota: Obtenido de (UNIT ELECTRONICS, s.f.)

Tomando en consideración esta breve introducción del modelo presentado, se pretende ampliar las características técnicas mediante la tabla 11, donde se describe de mejor forma las especificaciones y parámetros importantes como los voltajes o corrientes de alimentación, modulaciones, rangos de frecuencia de operación, entre otros datos.

Tabla 11*Características de los módulos transmisores y receptores*

Características	FS1000A	XY-MK-5V	STX822	SRX822
Frecuencia de operación	433,92/315 MHz	433/315 MHz	433/315 MHz	433/315 MHz
Modulación	ASK	ASK/OOK	ASK	ASK/OOK
Alcance	100 m		100 m	
Alimentación	3.5 a 12 VDC	5 VDC	1.2 a 6 VDC	2.4 a 5.5 VDC
Cantidad de pines	4	5	4	7
Antena	No incorporada		Incorporada	
Dimensiones (mm)	18x18	49x26	12x15.20	35x10.4
Temperatura de funcionamiento	20°C - 70°C	-30°C - 85°C	-20°C a 70°C	-30°C - 80°C
Potencia de transmisión/Sensibilidad de recepción	15 dBm	-105 dB	15 dBm	-107 dB

2.10.2.3 Módulos NODEMCU

Son tarjetas programables que almacenan procesos y funciones desarrollados a través de softwares de programación como Arduino IDE, LUA, ESP-idf o MicroPython, también cuenta con un módulo Wifi ESP32 que opera en la banda 2.4 GHz mediante el estándar 802.11. Son empleadas en proyectos referentes al Internet de las Cosas debido a sus características y facilidad de configurar, es compatible para el desarrollo de diseño de aplicaciones para PC y teléfonos móviles a través de la aplicación en línea denominada Arduino IoT Cloud.

Posee memorias donde se guardan las instrucciones, ocupa dimensiones reducidas, estándar 802.11, antenas incluidas dentro de la placa PCB y se necesita de un cable Micro USB, la figura 26 muestra una imagen de referencia de esta placa.

Figura 26

Módulo NodeMCU ESP8266



A continuación, en la tabla 12 se describen algunas características de los módulos NODE MCU ESP-32 y ESP-8266.

Tabla 12

Características de las tarjetas NodeMCU

Características	NodeMCU-32	NodeMCU-8266
Memoria	448 KB ROM/520 KB SRAM/ 16 KB SRAM En RTC	Instrucción RAM 32 KB/ datos RAM 96 KB/Flash externa 4 MB
CPU	Dual-Core Tensilica Xtensa LX6 (32 bit)	Tensilica Xtensa LX3 (32 bit)
Alimentación/conector	5 VDC/Micro USB	
Voltajes de entradas/salidas	3.3 VDC	
Cantidad de pines	30	
Pines analógicos ADC	18	1
Pines digitales GPIO	24	17
Pines PWM	16	4
Dimensiones (mm)	55x28	49x26
Temperatura de maxima funcionamiento	+50°C	
Conversores DAC	si	
Estándares	IEEE 802.11 b/g/n/e/i 2.4 GHz hasta 150 Mbit/s Bluetooth V4.2 BR	IEEE 802.11 b/g/n 2.4 GHz
Antena	Incorporada	
SoM	ESP-WROOM-32 (Espressif)	ESP-12E (Ai-Thinker)
SoC	ESP32 (Espressif)	ESP8266

2.10.2.4 Comparación de los módulos de comunicación en base a la aplicación del proyecto

El prototipo que se desarrollará busca una forma de reducir dimensiones en el diseño de la placa PCB de tal manera que la impresión del modelo en la placa sea comparable con otros diseños de los dispositivos comerciales, para ello es necesario tomar los arreglos de circuitos destinados para la comunicación por radiofrecuencia adecuados para el proyecto.

Considerando el lado del prototipo receptor, el modelo SRX882 es más largo y posee 7 pines a diferencia del modelo XY-MK-5V que posee 4 pines y es menos largo que el módulo receptor antes mencionado, lo cual se convierte en una gran opción para el sector de radiofrecuencias.

Para el sector de Wi-Fi, los dos modelos de módulos NodeMCU son similares comparando sus dimensiones, disponibilidad de pines, características físicas y tipo de software para la programación de este, sin embargo. La facilidad de encontrar el modelo NodeMCU-8266 a nivel comercial lo convierte en la opción final que se implementará en el prototipo.

2.10.3 Botones

Los botones o pulsadores permiten el paso de la energía a través de sus terminales al estar en corto circuito cuando es presionado o en circuito abierto cuando no (E-SWITCH, 2018). En el mercado existen variedad de modelos, precios, tamaños, materiales de fabricación y modos de uso como el switch sostenido o el modo de uso convencional, considerando los diversos escenarios, es recomendable utilizar un modelo acorde al diseño de implementación final.

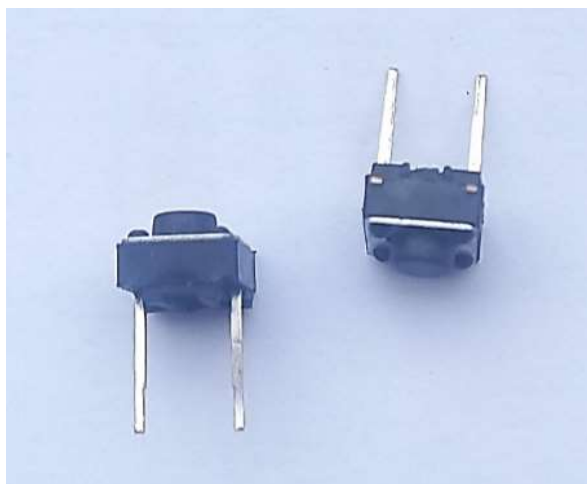
A continuación, se muestran los siguientes ejemplos de pulsadores y se mencionan las características más esenciales que los diferencian.

Pulsador 4 Pin 6X6X13

Este modelo de pulsadores ocupa espacios reducidos y son ideales para la implementación en las placas de circuitos impresos y Protoboard, sus 4 patas permiten un mejor soporte, posee un encapsulado de 6x6 mm y la altura del pulsador es de 13 mm, sin embargo, el botón se puede llegar a fracturar. En la figura 27 se observa el aspecto físico de este tipo de modelo de pulsador.

Figura 27*Botón pulsador 4 PIN 6X6X13***Pulsador 2 Pin 6X6X5**

Este pulsador posee dos pines que son amigables con el Protoboard y placas de circuito impreso, su encapsulado es de 6x6 mm y la altura del botón llega a 5 mm, ocupa poco espacio debido a sus reducidas dimensiones y son ideales para proyectos de pequeñas escalas. La figura 28 se presenta el modelo físico del pulsador.

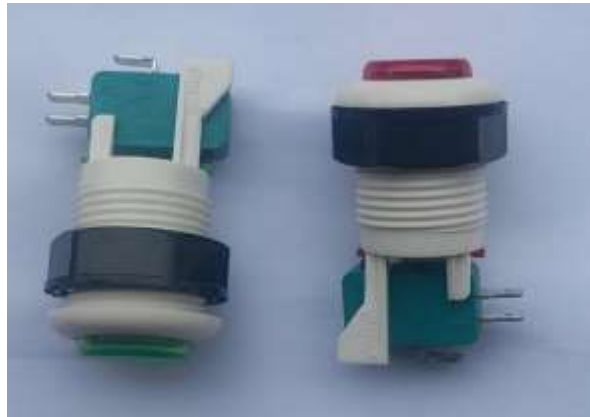
Figura 28*Botón pulsador 2 PIN 6X6X5***Pulsador Con Micro Interruptor**

Este pulsador se caracteriza por su gran tamaño y es un conjunto de tres piezas, el micro interruptor, un botón de material plástico que encaja con el micro interruptor y una tuerca plástica para sujetar en superficies. Posee un pin común (COM), uno normalmente cerrado

(NC) y uno normalmente abierto (NO). En la figura 29 se puede apreciar el micro interruptor correspondiente al encapsulado color verde, la extensión del botón de color blanco y la rosca para fijar en superficies.

Figura 29

Botón pulsador con Micro Interruptor



2.10.4 Líneas de transmisión

Los medios conductores son elementos esenciales para transportar la energía que se disipará en todos los componentes conectados por medio de este, permitiendo que el circuito cumpla su funcionamiento al tener una forma de hacer fluir las cargas eléctricas por todo el arreglo de elementos que conforman el circuito.

Generalmente, está hecho a base de cobre debido a varios factores como el bajo costo, es un buen metal conductor, flexible y se puede moldear, de tal manera que existen formas de representar un medio conductor de manera física y es a través de la elaboración del cable de cobre, un alambre forrado con una cubierta dieléctrica que evita el contacto directo con el conductor.

Por otra parte, se encuentra la representación de una delgada capa de cobre impresa sobre un material de sustrato, el cual cumple la función de un medio conductor que sustituye

el cableado de cobre, disminuyendo espacios y el volumen que ocupa en su forma de alambre, a continuación, me mencionan los más comunes para aplicaciones de baja escala.

Jumper Dupont

Cables de cobre Ideales para conexión de elementos en las placas de pruebas, de fácil manejo gracias a sus extremos que cuentan con pequeños puertos que encajan fácilmente la placa de tal manera que se pueden desmontar con facilidad, es compatible con elementos como sensores, pantallas entre otros módulos que cuentan con pines y se comercializan con diferentes medidas y conectores como se muestra en la figura 30, entre las alternativas que se encuentran disponibles están:

- Connector jumper Dupont (female - male)
- Connector jumper Dupont (female - female)
- Connector jumper DuPont (male - male)

Figura 30

Jumper Dupont



Línea Microstrip

La línea microstrip es un medio conductor que aprovecha la tecnología de circuito impreso tomando la forma de una cinta muy delgada que es la encargada de conectar los

componentes en la placa como lo haría un cable conductor de cobre para energizar todos los elementos del circuito.

En ocasiones es necesario perforar la PCB para fijar los elementos que poseen pines en sus terminales, ocupa reducidos espacios y es posible diseñarla variando sus dimensiones a través de algún software de simulación que disponga de esta herramienta como lo es Proteus Design Suite, NI Multisim, Fritzing o Easy Eda, quienes se convierten en una herramienta apropiada para el diseño de este tipo de líneas.

Cable De Cobre

El cable de cobre para el uso en el Protoboard suele ser un cable fino, ya que facilita la implementación en las placas de pruebas, los más habituales son los que se componen de varias hebras en su interior y aquellas que cuentan con una sola, brinda flexibilidad, posee un aislamiento, se puede definir la longitud del cable para ser cortado previo a su instalación.

Una de las alternativas de cable compatibles con el tamaño de los agujeros de las placas de pruebas es el cable UTP quien se compone de 8 hebras internas que mantienen un diámetro adecuado para ser colocado en las ranuras convirtiéndose en un medio ideal para este tipo de actividades donde se requiere un medio conductor para energizar los componentes distribuidos en el circuito.

2.10.5 Bases para el soporte de componentes

En este apartado se mencionan las alternativas que se pueden escoger para el desarrollo del circuito, son de gran utilidad tomando en cuenta los escenarios de aplicación de cada una de ellas, desde el desarrollo de circuitos de prueba en Protoboard o circuitos finales implementados en la PCB.

Para la construcción del tablero de distribución eléctrica como se denominó este prototipo, se deberá considerar las dimensiones que presentan los dispositivos comerciales, ya que se caracterizan por ocupar espacios reducidos, por lo que se tomará en cuenta un modelo de placa adecuado para el diseño final.

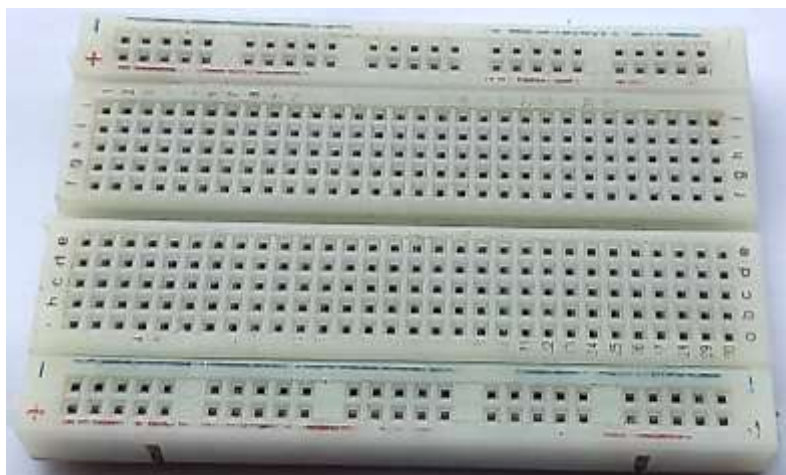
Protoboard

Un elemento indispensable en las pruebas de proyectos de electrónica ya que permite la conexión de varios elementos distribuidos en su estructura, es amigable con los pines de los elementos que se conectan y la extracción de las piezas ensambladas en él no requiere un gran esfuerzo (KONDSO, 2014).

Dispone de ranuras metálicas especialmente diseñadas para trabajar con circuitos en serie y otra sección para conexiones en paralelo, posee una carcasa plástica con agujeros que sirven para incorporar un componente electrónico y existen variedad de dimensiones para probar proyectos de pequeña y gran escala, en la figura 31 se encuentra un modelo con disponibilidad de 400 puntos. Ver figura 31.

Figura 31

Protoboard de 400 puntos



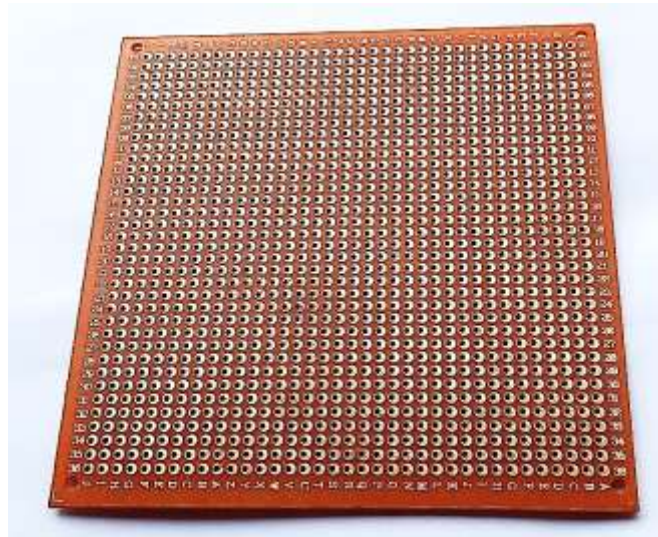
Placa PCB con Agujeros

La placa de circuito impreso con agujeros es una alternativa para evitar el cableado tipo Jumper, requiere del uso de soldadura y cables para realizar las conexiones entre los elementos colocados en los agujeros, esto garantiza firmeza ya que las piezas soldadas se mantienen adheridas a la placa debido a la fusión entre el dispositivo, la placa PCB y el estaño.

Los materiales más populares para el desarrollo de estos modelos de PCB son la fibra de vidrio, FR-4, fibra de aramida como sustrato y como material conductor se toma el cobre debido a su precio y accesibilidad, en la figura 32 se muestra un modelo de placa de material FR4 y cobre.

Es importante considerar los siguientes puntos:

- Se requiere de conocimientos en soldadura por estaño para las prácticas en este modelo de placas.
- Proporciona un amplio espacio para la colocación y la organización del circuito
- Es flexible hasta cierto punto.
- Existen variedad de dimensiones.
- Es una placa delgada.
- Fácil manejo.
- Bajo costo.

Figura 32*Placa PCB***Placa De Circuito Impreso (PCB)**

Es una de las mejores alternativas para la implementación de un circuito, se basa en una placa con líneas conductoras impresas en una plancha de cobre tomando en cuenta el diagrama del circuito previamente diseñado, esto se lo puede desarrollar aprovechando las herramientas de Software libres relacionados con el diseño de circuitos, donde es posible crear los modelos pensados para la impresión del PCB.

Para usar correctamente este modelo de placa es importante conocer los siguientes puntos:

- Es necesario tener conocimientos en soldadura por estaño.
- Evitar sobrecalentar las pistas conductoras, ya que se podría levantar y dañar la fina cinta de cobre.
- Flexible hasta cierto punto.
- Los precios varían dependiendo del diseño, modelo, dimensiones, entre otras características.

- Se obtienen mejores acabados en el diseño final.
- Es posible reducir las dimensiones finales del PCB al organizar los elementos de tal manera que se ocupe menos espacio.
- Los softwares como Proteus Design Suite, NI Multisim, Fritzing o Easy Eda, son los más populares para el desarrollo de diseños de circuitos.

2.11 Software para el diseño y desarrollo de circuitos

Una manera de poder llevar la idea del prototipo a nivel de software es a través de aplicaciones de diseños de circuitos, ya sean para el desarrollo de Scripts para colocar los componentes y conexión entre ellos, diseño PCB que permite modelar las líneas de conexión entre los componentes y extraer los archivos necesarios que se requieren para adquirir un modelo físico de la placa diseñada, entre otras características como la presentación de los resultados en 2D y 3D, para visualizar los resultados que dan una perspectiva de los resultados finales de las placas a nivel de software.

En esta sección, se realiza una búsqueda de los softwares que están disponibles de manera gratuita para el desarrollo del prototipo.

2.11.1 Proteus VSM

El software Proteus VSM, es una de las aplicaciones de libre acceso que es de gran utilidad para el desarrollo y la creación de circuitos eléctricos que resulta de la interacción de elementos disponibles en el software, VSM hace referencia a las palabras Virtual System Modelling o Sistema de Modelado Virtual en su traducción al español, ya que la acción de modelar circuitos a nivel de software de simulación es posible a través de esta aplicación. Su utilidad resulta favorable para las aplicaciones de prueba, ya que presenta una forma de simular el estado del circuito como tal, mediante alternativas que permiten examinar parámetros de medición, gráficas resultantes, estudiar su comportamiento en tiempos cortos determinados y

la posibilidad de interactuar con otros softwares de simulación relacionados a su funcionamiento. En la Figura 33 se muestra una referencia del entorno de este programa.

Figura 33

Proteus



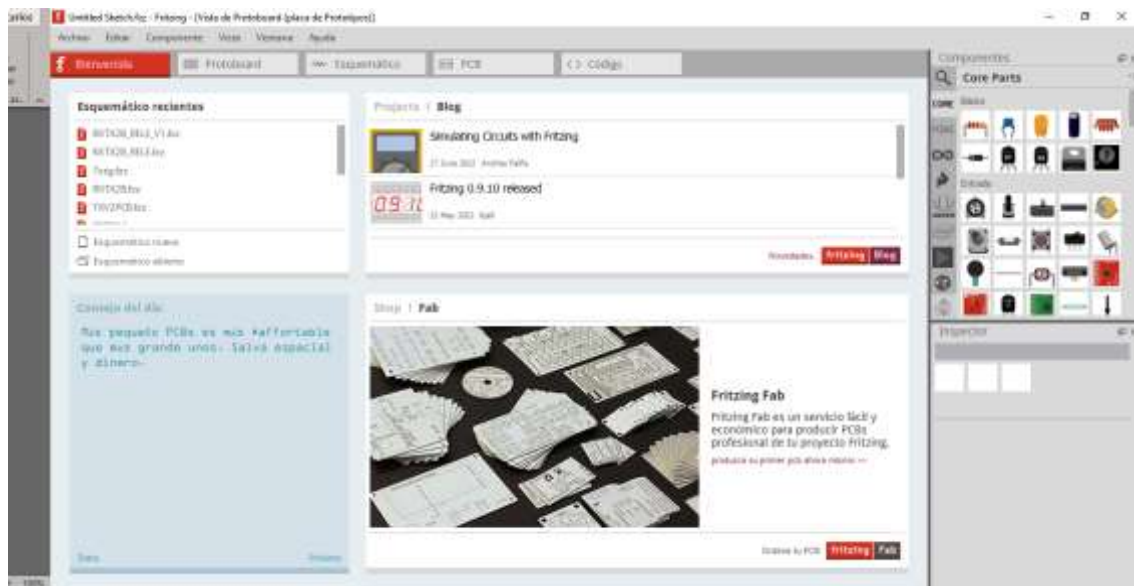
Entre las herramientas de medición que dispone el software se encuentran los siguientes, osciloscopios, analizadores de espectro, multímetros entre otros que cumplen el mismo principio de funcionamiento que ofrecen aquellos de manera física, por otra parte, es posible emplear varios modelos de microcontroladores dentro de la aplicación, en caso de no poseer un modelo en específico existe la posibilidad de solventar esa ausencia de elementos mediante actualizaciones y cargas de nuevas librerías que no se incluyen al momento de ejecutar la instalación del software en el ordenador. Se caracteriza por presentar un diseño en PCB del circuito configurado en su área de trabajo lo cual permite ser impresa para ensamblar sus componentes (Rossano, 2013).

2.11.2 Fritzing

Fritzing es un software libre que permite simular la instalación de circuitos, elementos, sensores, microcontroladores entre otros objetos en un Protoboard virtual que se encuentra disponible en la aplicación. El entorno de este programa puede observarse en la Figura 34.

Figura 34

Fritzing



Es una alternativa para mantener un orden con respecto a las líneas conductoras que se distribuyen a lo largo de la placa de pruebas, es recomendable para plasmar las conexiones planteadas, es de fácil manejo y posee una interfaz que es amigable para el rápido entendimiento.

Posee una opción que permite el diseño de placas PCB para su posterior impresión, previo a la implementación de los elementos o dispositivos incluidos en el diseño del circuito (Interaction Design Lab, 2013).

2.11.3 Autodesk Eagle

El software Autodesk Eagle es una potente herramienta que permite crear diseños electrónicos, fue desarrollada por la empresa Autodesk. Es una herramienta de gran utilidad

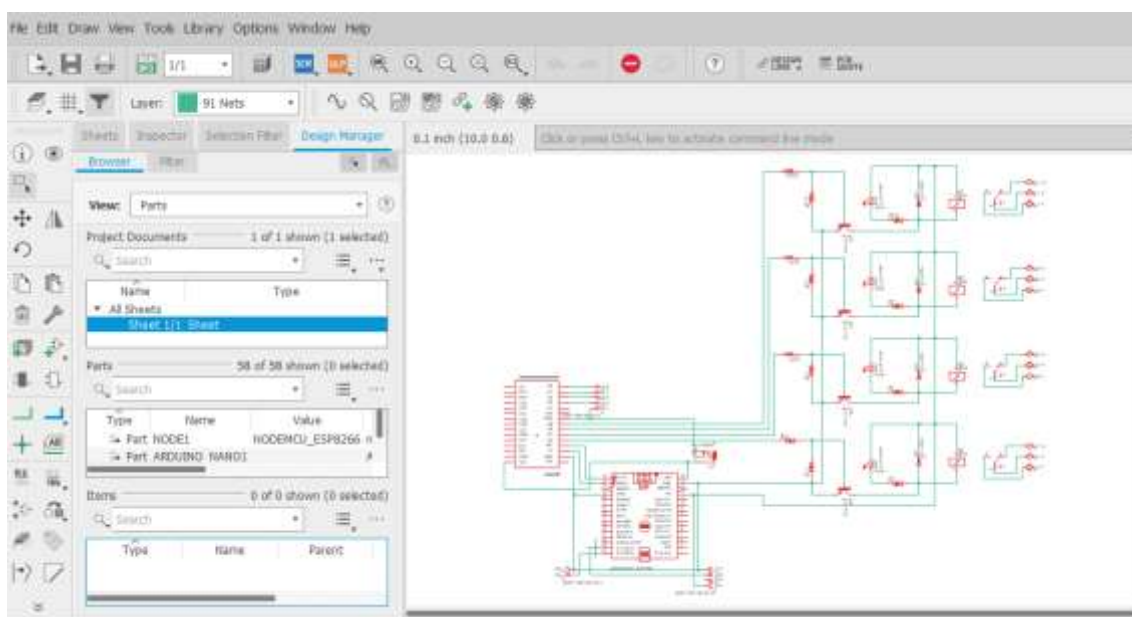
para la industria electrónica y aficionados a la electrónica para crear esquemas y diseños de circuitos impresos (PCB). Con este software se puede diseñar los esquemas de circuitos de manera gráfica y luego realizar el enrutamiento de las pistas para fabricar y ensamblar los componentes necesarios en la PCB.

La interfaz gráfica de esta aplicación es amigable para que los desarrolladores puedan esquematizar sus ideas de manera rápida y eficiente, cuenta con una amplia biblioteca de componentes y a su vez, permite el diseño de símbolos y footprints para el desarrollo de componentes personalizados.

La herramienta también proporciona características que garantizan la integridad y el funcionamiento correcto del diseño final como reglas de diseño y análisis de colisiones, cabe recalcar que este software puede trabajar con licencia gratuita. La Figura 35 muestra la interfaz de trabajo de este programa.

Figura 35

Autodesk Eagle



2.12 Software para la asignación de tareas y programación de las tarjetas

Tomando en cuenta que la tarjeta de la marca Arduino serán el principal componente para los procesos que se empleará con el prototipo, de este modo el software Arduino IDE es la mejor opción para el desarrollo de los códigos de programación que se grabará en la placa.

Con respecto a la aplicación para teléfonos móviles que servirá para interactuar entre el prototipo a través de un Smarth Phone, se desarrollará mediante el Software libre Arduino IoT Cloud ya que es compatible con los módulos de comunicación de la familia NodeMCU y es de fácil entendimiento. A continuación, se especifica de manera más amplia la utilidad de los programas.

2.12.1 Arduino IDE

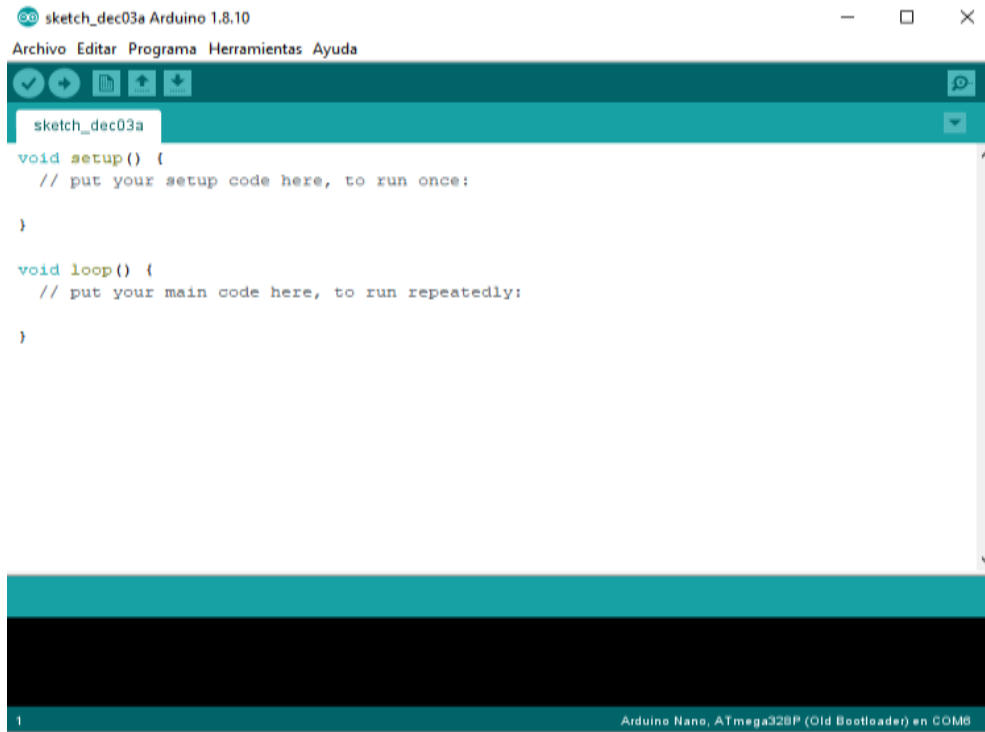
Arduino IDE (Integrated Development Environment) es una plataforma de programación disponible para sistemas operativos como GNU/Linux, Mac OSX o Microsoft Windows que permite el desarrollo de lógicas de programación basados en códigos almacenados en un sketch, posibilita la acción de verificar el manejo correcto de las sentencias respetando los algoritmos planteados, dispone de varias herramientas y librerías útiles para la construcción de proyectos.

El software está diseñado para la serie de tarjetas que Arduino ofrece en el mercado y se encarga de grabar las instrucciones que serán subidas a la memoria de la tarjeta en cuestión, presenta un entorno de trabajo de fácil manipulación y baja complejidad como se puede observar en la figura 36. También dispone de ejemplos almacenados en la plataforma para ser cargados sin errores y se considera como una de las herramientas de gran utilidad para el desarrollo de proyectos de electrónica que requieren de un software basados en entornos de simulación y programación, por otra parte, facilita la información desarrollada en archivos que pueden ser leídos por otros programas que se basan en el desarrollo de implementación virtual

de dispositivos eléctricos y electrónicos como el software Proteus, NI Multisim, entre otros (Peña, 2013).

Figura 36

Arduino IDE



2.12.2 Arduino IoT Cloud

La plataforma de desarrollo Arduino IoT Cloud, es una herramienta de gestión y administración de dispositivos del Internet de las Cosas que permite interactuar con todo tipo de sensores que son enlazados a la red con un almacenamiento en la nube donde se guarda todo tipo de información que la placa de tecnología Wi-Fi puede proporcionar, de tal manera que posibilita del desarrollo de redes de sensores que se administran y se visualizan mediante aplicaciones móviles que están configuradas en base al tipo de sensores, actuadores o módulos adicionales conectados en la red, es gratuita y proporciona algunas funcionalidades como:

Registro y autenticación: para permitir la vinculación entre dispositivos para proporcionar una comunicación segura.

Almacenamiento y procesamiento de datos: gran capacidad de almacenar y presentar datos en la nube que son generados por los dispositivos IoT.

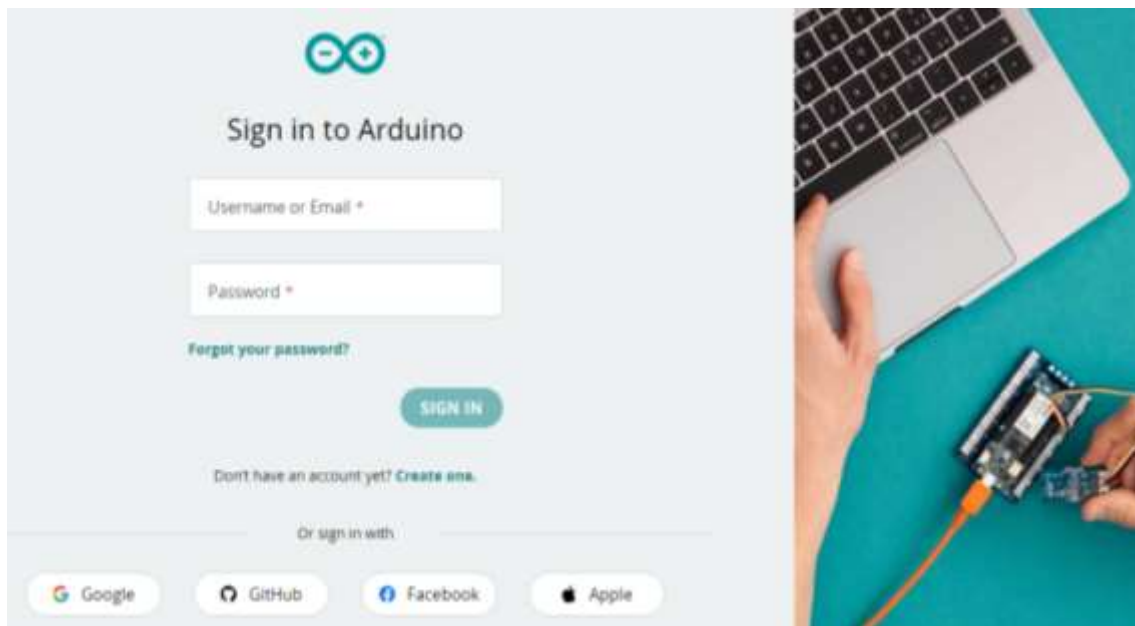
Gestión de dispositivos: con las herramientas que dispone la plataforma es posible configurar los sensores conectados al dispositivo IoT para poder monitorear el estado, actualizaciones del firmware, configuraciones o control remoto.

Conectividad y comunicación: proporciona protocolos de comunicación como MQTT, HTTP o CoAP.

Integración con otros servicios y aplicaciones: permite interactuar con los datos o funcionalidades de los dispositivos IoT con otros servicios y aplicaciones en la nube.

Seguridad: proporciona mecanismos de seguridad para la protección de los datos en las comunicaciones entre dispositivos IoT y la nube.

Las funcionalidades que dispone esta plataforma proporcionan una alternativa segura para el desarrollo de aplicaciones enlazadas a Internet gracias a las diversas herramientas, protocolos de comunicación, seguridad y fiabilidad ante el control de los datos, en la figura 37 se muestra la interfaz principal de la plataforma online, sin embargo, existe una versión de instalación que está disponible de manera gratuita para una instalación directa para equipos computacionales.

Figura 37*Interfaz gráfica de la plataforma IoT Cloud*

2.13 Software para el análisis de información en Sistemas Inalámbricos

Este apartado se comparte la recopilación de softwares que permiten el análisis de la información que se transmitirá por los módulos para ser propagada por el medio, esta captura de datos se realizará con ayuda de los dispositivos SDR, el cual será colocado en un punto medio de comunicación de los módulos para interceptar las frecuencias empleadas en los casos de aplicación del equipo comercial y el prototipo desarrollado, detectar protocolos, presentar las modulaciones de la señal, descifrar códigos, replicar información, con la finalidad de evaluar y determinar la existencia de vulnerabilidades en ambos escenarios.

2.13.1 GNU Radio

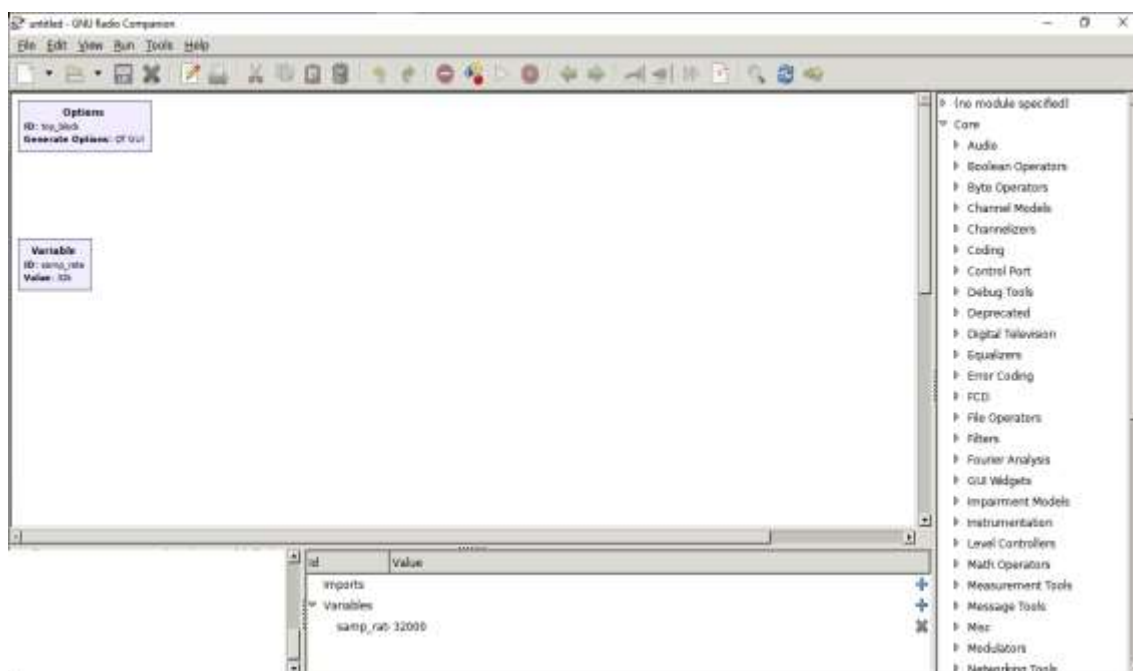
Es un software de simulación de acceso libre sin requerimientos de licencias, es una herramienta que basa su funcionamiento en el desarrollo de lógicas de bloques que contienen instrucciones que permiten el procesamiento digital de señales, se encuentra disponible para sistemas operativos populares como Windows, Linux, Mac, entre otros.

Trabaja en conjunto de dispositivos hardware conocidos como SDR que forman parte de la familia de radios definidos por softwares compatibles con el programa, estas configuraciones y agrupaciones de bloques se basan en lenguajes de programación como C++ o Python.

Resulta de gran utilidad al momento de ejecutar pruebas referentes a las modulaciones y demodulaciones, fuentes de señales, analizadores de redes, convertidores, entre otras aplicaciones (Vasan, 2019). Ver figura 38.

Figura 38

GNU Radio



2.13.2 RTL_433

Este software surgió gracias Benjamín Larsson su desarrollador, nombrado como RTL_433 es una herramienta que hoy en día sigue en constante actualización por parte de una comunidad muy amplia de personas encargadas de su mantenimiento, tiene la disponibilidad de analizar en tiempo real el tráfico de paquetes, evidenciar el tipo de modulación de la señal es detectada que es emitida por equipos con sistemas radiantes a nuestro alrededor, facilita

información de modelos de sensores hallados en el entorno como el caso del sistema TPMS que presentan los fabricantes de vehículos como Ford, Toyota, Renault, etc. El cual puede ser detectado con gran facilidad por el software.

Es un software de acceso libre y no requiere licencias, es una aplicación que requiere de un hardware de placa SDR tales como el HackRF One y el RTL-SDR que se han vuelto muy populares en estos últimos tiempos y son compatibles para el software (Gámez, 2020). Ver figura 39.

Figura 39

RTL-433

```

C:\Users\Bryan\Downloads\rtl433_win64_11112016\rtl_433_64bit_static.exe
Registering protocol [31] "Bresser Thermo-/Hygro-Sensor 3CH"
Registering protocol [32] "Springfield Temperature and Soil Moisture"
Registering protocol [33] "Oregon Scientific SL109H Remote Thermal Hygro Sensor"
Registering protocol [34] "Acurite 606TX Temperature Sensor"
Registering protocol [35] "TFA pool temperature sensor"
Registering protocol [36] "Kedsum Temperature & Humidity Sensor"
Registering protocol [37] "blyss DC5-UK-WH (433.02 MHz)"
Registering protocol [38] "Stealeate TPMS"
Registering protocol [39] "Schraeder TPMS"
Registering protocol [40] "Elro DB286A Doorbell"
Registering protocol [41] "Efengy Optical"
Registering protocol [42] "Honda Car Key"
Registering protocol [43] "Fine Offset Electronics, XC0400"
Registering protocol [44] "Radiohead ASK"
Registering protocol [45] "Kervi PIR Sensor"
Registering protocol [46] "Fine Offset WH1050 Weather Station"
Registering protocol [47] "Honeywell Door/Window Sensor"
Registered 47 out of 69 device decoding protocols
Found 1 device(s):
 0: Realtek, RTL2830UHIDIR, SN: 00000001

Using device 0: Generic RTL2832U OEM
Found Rafael Micro RB20T tuner
Exact sample rate is: 250000.000414 Hz
Sample rate set to 250000
Bit detection level set to 0 (Auto)
Tuner gain set to Auto
Reading samples in async mode...
Tuned to 433920000 Hz

```

2.13.3 Equipo computacional para la instalación de los softwares necesarios para el proyecto

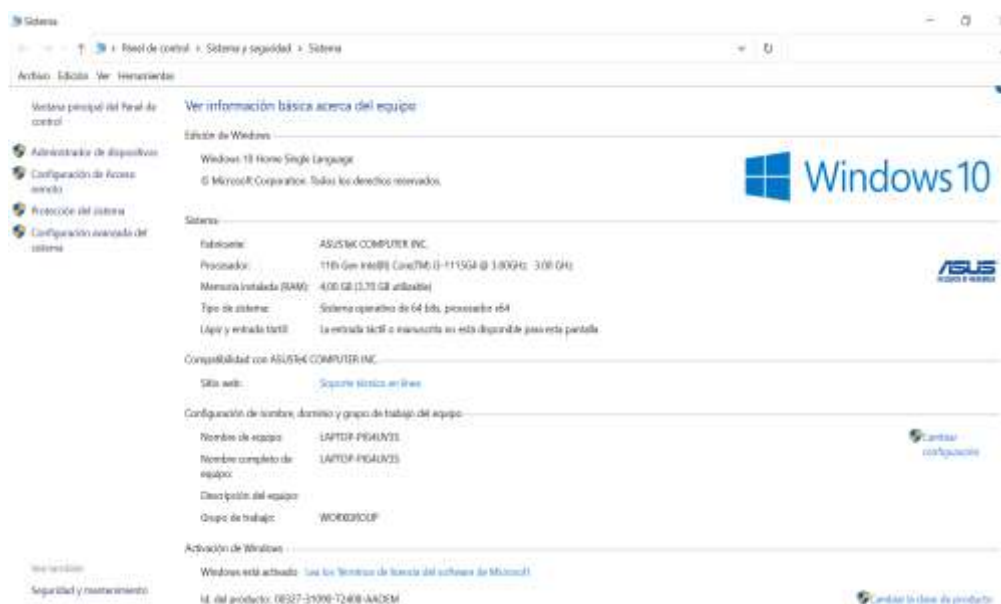
Con el fin de aprovechar al máximo el uso de la tecnología de SDR, es fundamental adquirir una computadora que cumpla con los requisitos necesarios para el funcionamiento adecuado de estos equipos. La potencia de procesamiento, la capacidad gráfica y la conectividad son aspectos fundamentales para poder manejar señales de radio en tiempo real

con el uso del SDR. Además, una buena capacidad de almacenamiento es muy importante para asegurar suficiente espacio para almacenar programas y datos asociados a esta tecnología.

Otra característica primordial es seleccionar un sistema operativo compatible con los programas especializados de SDR, como Windows, macOS o alguna distribución de Linux. Esta compatibilidad asegurará un funcionamiento sin problemas y permitirá aprovechar al máximo las funcionalidades del equipo. Si la movilidad es importante, considerar una computadora portátil podría ser beneficioso, ya que permitiría trabajar desde distintos lugares y explorar el espectro radioeléctrico en diferentes entornos. En la Figura 40 se puede observar las características de la computadora portátil que se adquirió para la instalación de los softwares necesarios para los equipos SDR.

Figura 40

Sistema operativo de ASUS VivoBook



CAPITULO III

3 METODOLOGÍA

En este apartado se describe la metodología empleada para el desarrollo de la propuesta. Además, se presentan aspectos fundamentales correspondientes a la implementación y ejecución de cada uno de los escenarios propuestos donde se realiza el diseño de la propuesta, la arquitectura general del proyecto, el desarrollo de los diagramas de bloques para la realizar los ataques, también se muestra el análisis del dispositivo IoT en cuanto a funcionamiento y nivel de aplicación, de la misma manera se presenta la construcción del hardware y software del prototipo.

3.1 Investigación Exploratoria

Se realizará la búsqueda de información en fuentes bibliográficas como libros, proyectos de tesis, artículos o revistas científicas que ayudaran a profundizar los siguientes temas: Dispositivos IoT, vulnerabilidades y ataques a IoT, tipos de tecnologías SDR, aplicaciones del software GNU Radio para SDR, Diagramas de Bloque desarrollados en GNU Radio.

3.2 Investigación Aplicada

Para este tipo de investigación, se aplicarán todos los conocimientos obtenidos en la consulta bibliográfica para la implementación del proyecto, además se hará uso de la técnica de ensayo y error, donde comprobara si el diseño del receptor en el software GNU Radio y el dispositivo RTL SDR es efectivo al momento de realizar las pruebas de captación de señal del tablero de distribución eléctrica. Para la segunda etapa de la propuesta se realizará la experimentación del sistema anteriormente mencionado en un entorno real donde se ejecutarán diferentes ataques para estudiar las vulnerabilidades de estos dispositivos IoT.

3.3 Desarrollo de la propuesta:

El desarrollo de la propuesta incluye la aplicación de cuatro fases, la primera consiste en una revisión bibliográfica donde se realiza un estudio generalizado de la tecnología IoT y el análisis de los equipos que se emplean para la construcción de la implementación, en la segunda se analiza el diseño del prototipo y el desarrollo de la lógica de bloques, en la tercera fase se analizan los datos obtenidos y las vulnerabilidades halladas, finalmente en la cuarta se realizan las pruebas necesarias para verificar el funcionamiento del prototipo y se analizan los resultados obtenidos de los ataques.

3.3.1 Fase 1: Investigación bibliográfica y elección de componentes electrónicos y softwares necesarios para el desarrollo de la implementación

En esta fase se realiza una revisión bibliográfica para comprender las vulnerabilidades presentes en IoT, además de conocer características y aplicaciones que abarca el mundo del Internet de las cosas. También se estudia el comportamiento de la tecnología SDR en trabajo conjunto con la herramienta GNU Radio en el análisis de las vulnerabilidades y desarrollo de ataques.

Gracias a esta investigación se obtiene un estudio referente a las especificaciones, características propias de los componentes electrónicos y los softwares que se emplearan en la construcción del prototipo. En el capítulo 2, se puede evidenciar las comparaciones realizadas para elegir el mejor componente al momento de empezar con el diseño del prototipo, así como también seleccionar los programas que sean compatibles con los SDR utilizados.

3.3.2 Fase 2: Diseño y configuración de equipos

En esta fase, se llevará a cabo un análisis detallado de las características del hardware y software de un dispositivo IoT comercial, Además, se procederá con el desarrollo de un nuevo dispositivo IoT que contará con funcionalidades similares y mejoradas. Una parte esencial de

la investigación consistirá en la formación de diagramas de bloques para realizar un análisis exhaustivo de los datos receptados por el dispositivo.

En particular, se utilizarán dispositivos para el análisis de los datos como el SDR HackRF One, RTL-SDR y el software GNU Radio como medio de análisis para detectar posibles vulnerabilidades del dispositivo IoT en relación con los ataques efectuados. Este enfoque innovador permitirá una comprensión más profunda del funcionamiento del dispositivo comercial y contribuirá significativamente a mejorar la seguridad y fiabilidad del nuevo dispositivo IoT desarrollado.

3.3.3 Fase 3: Análisis de datos y búsqueda de vulnerabilidades del dispositivo IoT

Una vez receptada la señal es necesario el análisis de la información, GNU Radio permite observar el comportamiento de la señal mediante bloques específicos, de esta manera se puede visualizar la señal en función del tiempo, frecuencia, tren de datos y el diagrama de constelación. Además, se puede hacer uso de la herramienta Audacity como otro medio que permite mostrar donde la señal previamente capturada para ver el tren de datos en el receptor o utilizar el software RTL-433 para realizar el escaneo de librerías o protocolos de comunicación inalámbricas que utilizan los dispositivos de radiofrecuencia y son susceptibles a sufrir ataques.

3.3.4 Fase 4: Comparación de resultados y alternativas para mitigar posibles ataques.

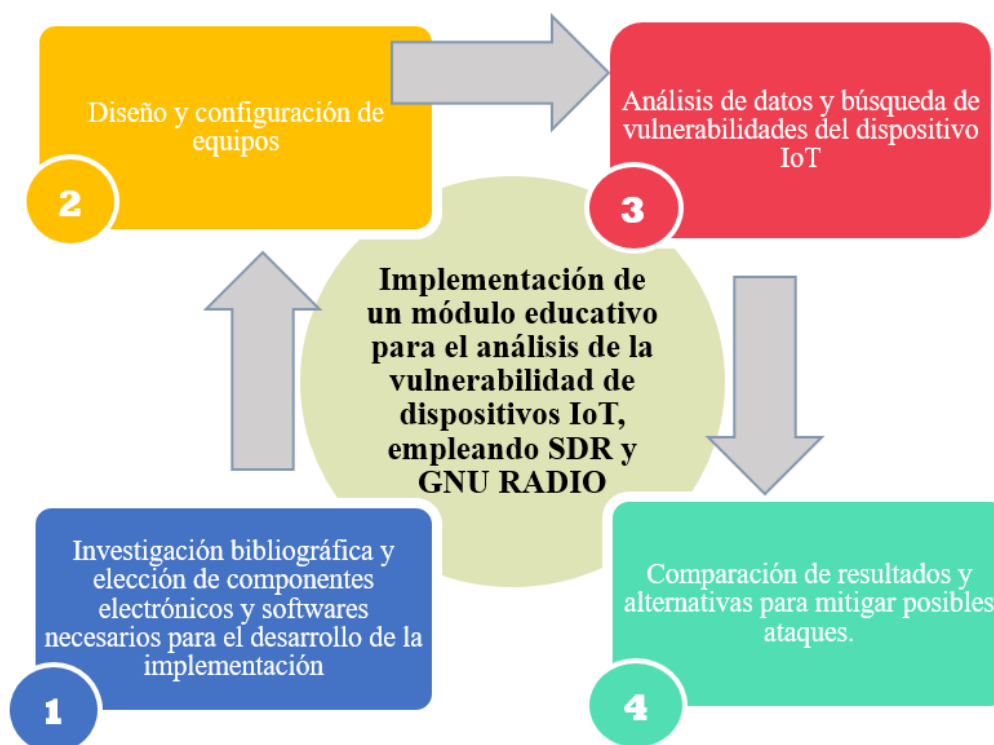
En esta última etapa se despliega la metodología aplicada, donde se realizan las pruebas necesarias para verificar el funcionamiento de cada uno de los sistemas planteados para cubrir las expectativas de la propuesta tecnológica, además se recolectan los resultados obtenidos a través de los ataques efectuados en dichos escenarios, para analizar y comparar las vulnerabilidades a las que se encuentran expuestos este tipo de sistema IoT mediante el uso de los equipos de radio definida por software HackRF One y RTL-SDR. También, se presentan

las alternativas basadas en recomendaciones que contribuirán en el cuidado de la información durante la comunicación inalámbrica y evitar posibles ataques de repetición de señales o evitar la desconexión de la red de internet ante una inhibición de señal.

A continuación, en la figura 41, se muestra un diagrama que permite interpretar de manera gráfica el proceso que se debe cumplir para lograr el desarrollo de la propuesta.

Figura 41

Diagrama general para el desarrollo de la propuesta.



3.4 Arquitectura del proyecto

La arquitectura del proyecto para el análisis de señales, vulnerabilidades y ataques en dispositivos IoT combina el uso de tecnología SDR como el HackRF One, el dispositivo IoT seleccionado, el dispositivo IoT en desarrollo y una computadora para el procesamiento y análisis de datos. A través de los softwares compatibles con tecnología SDR, se capturan y examinan las señales durante las comunicaciones inalámbricas de estos dispositivos.

La computadora alberga los softwares especializados para detectar y decodificar señales, permitiendo una evaluación exhaustiva de la seguridad de los dispositivos. Esta arquitectura busca proporcionar un medio para identificar posibles vulnerabilidades y fortalecer la seguridad en el contexto del Internet de las cosas en un entorno de laboratorio controlado, con el enfoque integral de estos componentes, se espera obtener resultados significativos en el análisis y mejora de la seguridad en dispositivos IoT, ver figura 42.

Figura 42

Arquitectura del proyecto



3.4.1 Diseño de ataque replay o reproducción

Es de gran importancia destacar que el objetivo de este trabajo no es promover actividades ilícitas, es más bien comprender los riesgos y vulnerabilidades ligadas a las comunicaciones inalámbricas. Razón por la cual se enfoca en el estudio y análisis de las señales de radiofrecuencias que operan a una frecuencia de 433 MHz con el uso de la tecnología SDR.

Para el diseño de este ataque en GNU Radio se debe tener en cuenta que el objetivo de un ataque de reproducción es interceptar la información para después retransmitirla con el fin de engañar al sistema y obtener el acceso no autorizado. El desarrollo de este se desglosará en dos partes, en la primera es necesaria la construcción de un diagrama de bloque que capture la

señal que se está transmitiendo entre el control y el prototipo, el segundo debe cumplir con la retransmisión de la señal capturada.

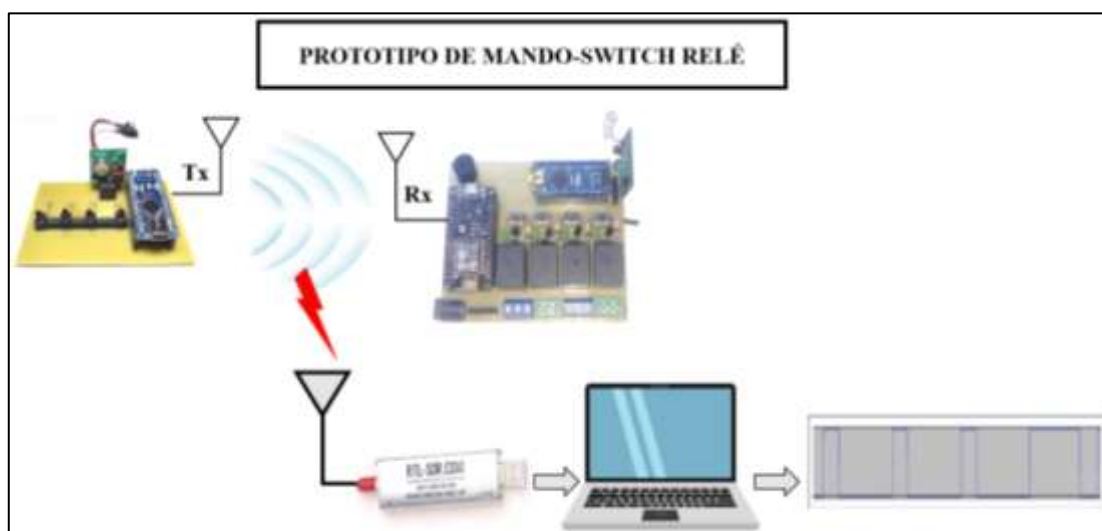
3.4.1.1 Diagrama de bloque para la captura de señal

Para el diseño del ataque de repetición, lo primero que se necesita es capturar la señal que se transmite en una determinada frecuencia, en este caso se conoce que el transmisor trabaja a 433 MHz y la modulación que utiliza es la ASK, estos datos se obtuvieron en la recolección bibliográfica estudiada en el capítulo 2. En el caso de que se desconozcan estos datos, es de gran ayuda contar con ciertas herramientas como GQRX, SDRshark o SDRConsole, para conocer la frecuencia a la que está operando el objetivo de ataque.

Aprovechando las características que tiene el RTL-SDR como receptor de señal, se emplea como capturador de señal inalámbrica en la comunicación propuesta en la figura 43, en el esquema se observa que el equipo SDR intercepta la información y posterior a ello la almacena en un fichero, para que luego sea retransmitida en la segunda parte del diseño.

Figura 43

Diagrama de conexiones para la capturar la señal con el RTL-SDR



Como el primer paso es recibir la señal FM, se utilizará el bloque RTL-SDR Source diseñado para trabajar con el transceptor RTL-SDR, en este se configura la frecuencia de trabajo que es 433.904 MHz y una frecuencia de muestreo de dos millones por segundo. Para visualizar la señal se emplea el bloque QT GUI Frequency Sink donde se define una frecuencia central de 434 MHz y un ancho de banda de 2 MHz.

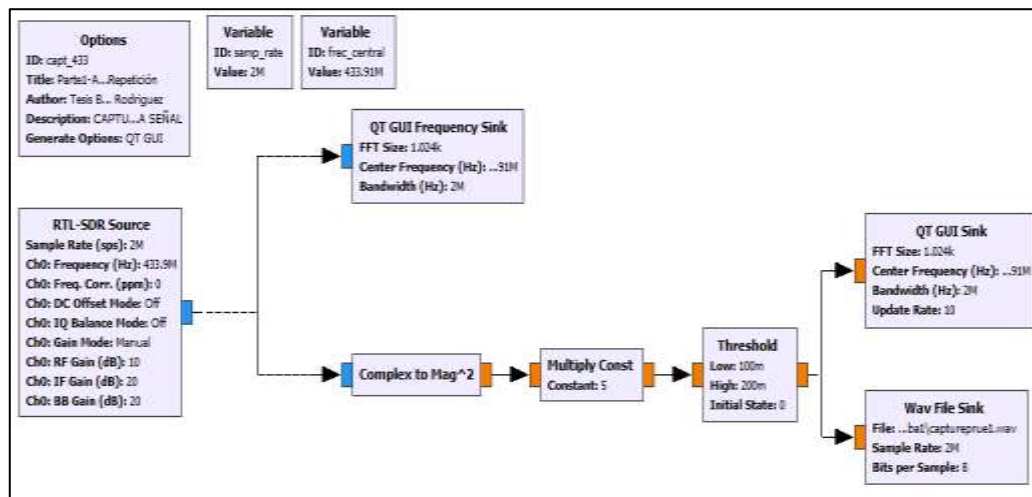
Lo siguiente es demodular y reducir el ruido de la señal AM recibida, para esto se utiliza el bloque Complex to Mag, la función de este bloque es convertir la señal captada en una magnitud real, lo que implica que la componente imaginaria se elimine, como la salida de este bloque es de tipo Float, es esencial asegurarse de que los bloques que siguen en la cadena de procesamiento también operen con valores de este tipo.

El siguiente paso es utilizar el bloque Multiply Const donde se especifica una constante de 5, para aumentar la amplitud de la señal y mediante el bloque Threshold, se configura un valor umbral bajo de 100m y alto de 200m, que de acuerdo con la sentencia se convertirá en un valor de 1 o 0, para convertir la señal en una onda rectangular.

Mediante el bloque Wav File Sink, se almacena la señal en formato wav en la ubicación establecida y finalmente se emplea el bloque QT GUI Sink para representar de forma gráfica los diagramas de constelación, el espectrograma, así como también el comportamiento de la señal tanto en función del tiempo y frecuencia. El diagrama final se observa en la Figura 44.

Figura 44

Diagrama de bloques para la captura de la señal con el RTL-SDR en GNU Radio

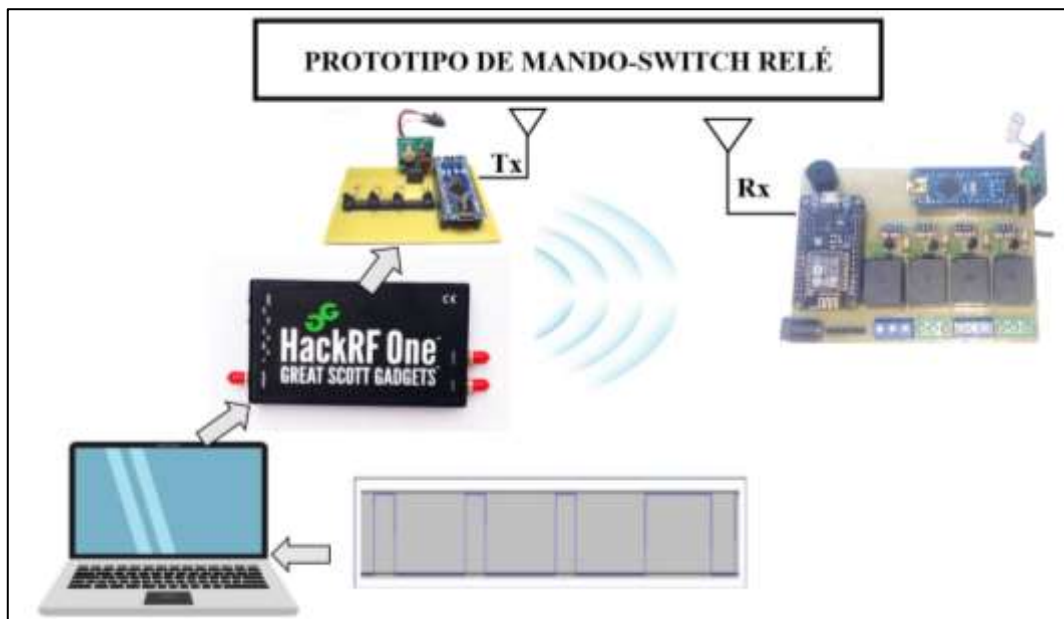


3.4.1.2 Diagrama de bloque para retransmitir la señal

Para replicar la señal se usa el equipo HackRF One, de tal modo que el dispositivo transceptor se convierta en el control que active los canales del receptor con ayuda del tren de datos previamente capturado. En la Figura 45 se observa el esquema de conexión que gráfica lo antes mencionado.

Figura 45

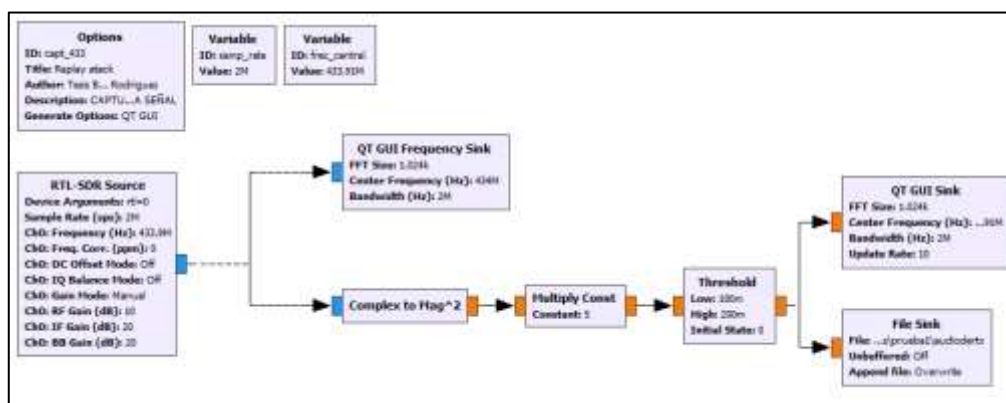
Diagrama de conexiones para retransmitir la señal con el HackRF One



Primero se genera un diagrama de bloque en el GRC que permite capturar y grabar la señal en formato IQ, también se elabora un segundo flujograma para transmitir la información con el HackRF One. El diagrama que se desarrolla para la captura de datos se elabora en base al primer diagrama presentado para el análisis de la información, difiere del primero porque este utiliza el bloque File Sink para almacenar los datos en un fichero en I y Q. La figura 46 representa el flujograma descrito.

Figura 46

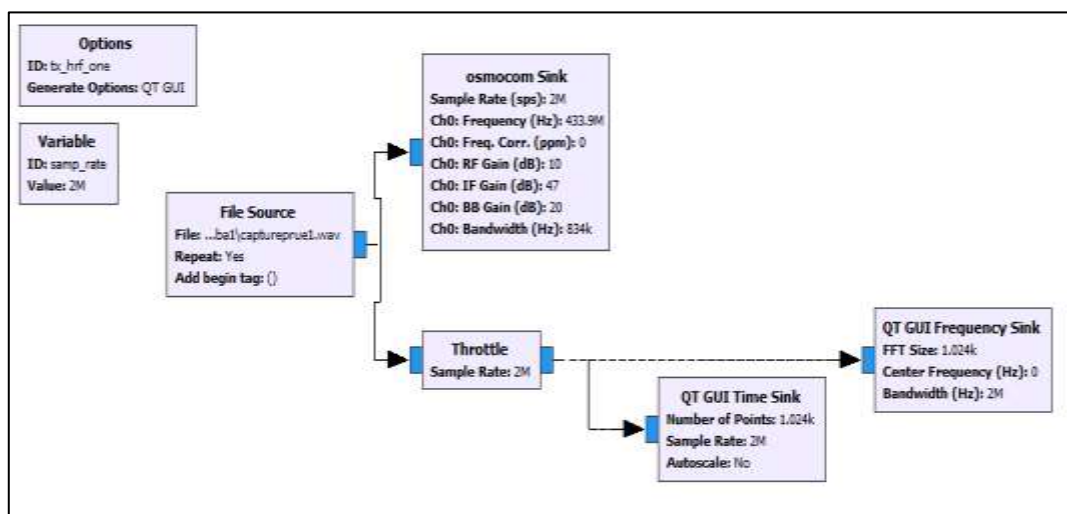
Diagrama de bloques para transmitir la señal con el HackRF One en GNU Radio



Una vez obtenida la información lo siguiente es transmitir los datos mediante el SDR, para esto se genera un segundo esquema de bloques, el objetivo de este flujograma es tomar la señal guardada en el fichero IQ con el bloque File Source, este a su vez entregara la información al HackRF One mediante el bloque Osmocom Sink, a la salida del bloque Files Source, también se conecta un bloque Throttle para que el ordenador limite sus recursos durante la simulación, para finalizar se emplean los bloques QT GUI Frequency Sink y Time Sink para visualizar la señal en dominio de la frecuencia y el tiempo. En la figura 47, se observa el esquema completo.

Figura 47

Diagrama de bloques del HackRF One



Para comprobar el funcionamiento de esta lógica de bloque, en lugar de usar el transmisor del esquema gráfico que se observa en la figura 45, se procede a ejecutar este programa que mediante el uso del HackRF One reemplazara al transmitir el tren de datos que activa los canales del prototipo de dispositivo IoT.

3.4.2 Diseño de inhibidor de señal.

Un ataque mediante un inhibidor de señal tiene como objetivo interferir deliberadamente con la señal de radiofrecuencia utilizada por dispositivos inalámbricos, sistemas de comunicación y redes, este busca saturar el canal de comunicación entre

dispositivos, provocando que se dificulte la transmisión y recepción de datos legítimos. Razón por la cual en este apartado se analizará el impacto que puede generarse mediante el uso de un equipo HackRF One y el software GNU Radio.

Una vez conectado el hardware a la computadora, lo primero que se hará es conocer la serie que tiene el SDR, una de las herramientas que proporciona GNU Radio en su instalación es el programa GQRX, con este programa se puede conocer de forma inmediata este dato, en la Figura 48 se observa cómo obtener la serie del equipo.

Figura 48

Entorno de configuración de dispositivos en el programa GQRX



Ahora para crear un flujograma correspondiente al inhibidor de señal, es necesario realizar una interferencia en un canal de comunicación determinado, entonces se usará la lógica de un inhibidor de tipo spot, el cual basa el ataque en seleccionar una frecuencia específica y una potencia ideal para que este sea efectivo.

Como primer paso se necesita identificar el canal donde opera la red, en este caso se utiliza una aplicación denominada WiFi Data, al iniciarla esta detecta las redes que se encuentran operando en la banda de frecuencia de 2,4 GHz. Ahora se identifica la red que se

necesita inhibir, al analizar se obtiene como respuesta que la red se encuentra en el canal 155 correspondiente a la frecuencia 5.775 GHz, en la figura 49 se puede observar las redes detectadas por el móvil, la interfaz muestra la información necesaria de la red como el canal en el que esta trabajando y el tipo de red.

Figura 49

Detección de redes en la aplicación WiFi Data

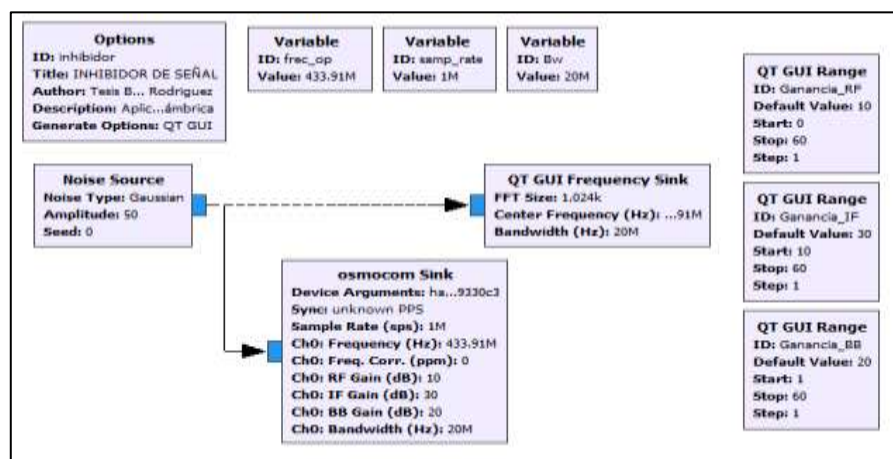
Network Name	MAC Address	Channel	Frequency	Security	Signal Strength
"UPSE"	E8:1D:A8:A2:D8:EC	CH 155	5775MHz	Open	-37dBm
"OFICINAS2"	D8:38:FC:3A:DF:98	CH 1	2412MHz	Open	-86dBm
"UPSE"	D8:38:FC:BA:DF:9C	CH 42	5210MHz	Open	-76dBm
"OFICINAS"	D8:38:FC:3A:E5:08	CH 6	2437MHz	Open	-78dBm
"DOCENTES"	D8:38:FC:7A:E5:08	CH 6	2437MHz	Open	-78dBm
"OFICINAS"	E8:1D:A8:22:D8:E8	CH 1	2412MHz	Open	-40dBm
"DOCENTES"	D8:38:FC:7A:DF:98	CH 1	2412MHz	Open	-82dBm
"OttoVera"	28:68:D2:AE:99:50	CH 6	2437MHz	Open	-86dBm

Una vez identificada la frecuencia se inicia con el diagrama de bloque, a la entrada se tiene el bloque Noise Source que produce una señal de ruido de tipo gaussiano, este se configura con el tipo de ruido ya mencionada, y con una amplitud de 50, a la salida se conectan

dos bloques el primero corresponde a QT GUI Frequency Sink, este permite obtener una representación gráfica del inhibidor de señal, y finalmente el bloque Osmocom Sink, en el cual se define el dispositivo HackRF One con la respectiva serie, también ayuda a especificar la frecuencia objetivo, de muestreo, el ancho de banda. y mediante el uso de los bloques QT GUI Range se podrá variar la ganancia IF, RF y BB mientras el programa se esté ejecutando. De esta manera se puede obtener buenos resultados al aplicar el inhibidor, debido a que si la señal de interferencia es muy débil se puede aumentar el valor de las ganancias. En la figura 50 se observa el diagrama final de lo antes mencionado.

Figura 50

Diagrama de bloques correspondiente al Inhibidor de Señal



3.5 Análisis del modelo y características del diseño final a nivel comercial

En esta sección se pretende analizar con mayor profundidad el dispositivo comercial Sonoff quien fue el seleccionado como equipo para ambientes del internet de las cosas que se usará para establecer la comunicación inalámbrica, ya sea, entre el dispositivo IoT y el Smartphone que porta la aplicación para poder interactuar al estar enlazados a la red de internet.

Para esto se deberá tomar en cuenta el funcionamiento del software que ocupa el dispositivo, compatibilidad de la App y versiones de Android o iOS, complejidad de manejo, interfaz de control, entre otras características. Del mismo modo se analiza el hardware del

dispositivo con la finalidad de identificar que tecnologías de módulos RF ocupa para las transmisiones inalámbricas, microcontroladores, tipos de antenas, dimensiones de la placa o carcasa, entre otras características.

3.5.1 Hardware del dispositivo

Para analizar el hardware de dispositivo 4CHPRO de la marca Sonoff es necesario revisar físicamente la composición de toda la estructura en general, desde el dimensionamiento del modelo de placa PCB, componentes eléctricos, arreglos de circuitos, módulos y tecnologías, y la carcasa protectora que aloja la placa.

Análisis de las medidas de la placa PCB y la carcasa protectora

La placa PCB es una parte muy importante entre los circuitos ya que, es el responsable de mantener los componentes y arreglos de circuitos en una sola pieza, en el caso del modelo de placa PCB desarrollada para este dispositivo IoT presenta un diseño compacto con los componentes montados y organizados de tal manera que ocupa un espacio reducido, de manera general tiene 85 milímetros de largo y 12 de ancho, mientras que, la carcasa protectora posee 145 milímetros de largo y 90 de ancho, está diseñada para que la placa PCB se instale de manera precisa en su interior, la manera de fijar el dispositivo en la pared puede ser a través de tornillos o también, por medio de sujetadores que el dispositivo trae consigo.

Componentes de la placa PCB

Realizando una inspección visual de la parte interna del dispositivo se pudo evidenciar la composición y los arreglos de los circuitos dispuestos en la placa, entre ellos se encontraron los siguientes componentes presentados en la tabla 13.

Tabla 13

Componentes del dispositivo comercial

Componentes	
Microcontrolador	El esp8285 es un chip Wi-Fi programable que permite almacenar instrucciones, realizar tareas o interactuar con sensores u otros dispositivos de red, en este caso contiene las instrucciones que posee el Sonoff 4CHPro
Diodos supresores	El electroimán del relé puede causar picos de voltaje (fuerza contraelectromotriz), lo suficientemente altos para dañar circuitos, el diodo permite absorber los altos picos de voltaje producidos por la fuerza contraelectromotriz
Alimentación	El dispositivo posee un Plug DC Female destinado para la alimentación de entrada (9-23 V)
Antena RF	El dispositivo Sonoff 4CHPro posee una antena de cobre tipo helicoidal que está diseñada para la recepción de señales de radio en frecuencias de 433.92 MHz
Antena Wi-fi	El dispositivo Sonoff 4CHPro lleva incorporada una antena impresa en la placa PCB para frecuencias de 2.4 GHz y estándar IEEE 802.11 b/g/n

3.5.2 Software del dispositivo a nivel comercial

El dispositivo Sonoff 4CHPro es un producto lanzado por la empresa tecnológica ITEAD Studio con sede en China que se enfoca en el desarrollo de diseños, productos del internet de las cosas y soluciones innovadoras y asequibles para la automatización o el desarrollo de escenarios personalizados. A parte de esto, ha desarrollado la plataforma eWeLink que es la aplicación para Smartphoness que se encarga del control y la administración de todos los dispositivos IoT conectados a la red de Internet.

Análisis de la interfaz de control del dispositivo

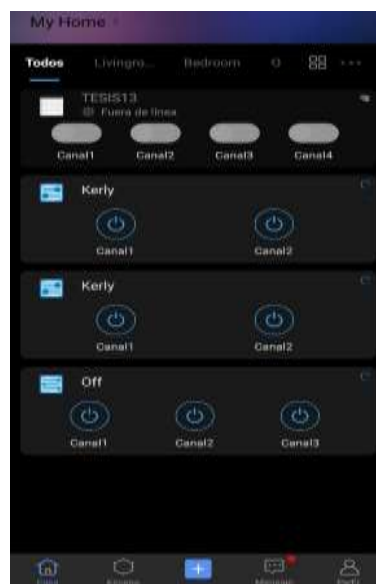
Para poder acceder al Sonoff es necesario descargar la aplicación que permitirá vincular el dispositivo con la red de Internet y así poder interactuar a través de un Smartphone, para ello es necesario descargar eWelink Smarth Home en Google Play de Android o la Play Store para sistema operativo iOS.

En este punto se debe alimentar el dispositivo para su correcto funcionamiento y después abrir la aplicación móvil para empezar a vincular el dispositivo, para ello se requiere presionar el primer botón del dispositivo por aproximadamente 10 segundos para acceder al modo de configuración, lo cual permite identificar a través del Smartphone el dispositivo Sonoff, se selecciona el dispositivo para luego ingresar los parámetros de la red Wi-Fi como el SSID y la contraseña a la que se conectará por defecto, se aceptan los cambios, se asigna un nombre al dispositivo y por último, se aceptan los cambios.

Luego de haber realizado el anterior procedimiento ya es posible acceder a la interfaz de control del dispositivo en el que se puede interactuar con el dispositivo final mediante la red Wi-Fi para sus aplicaciones en el campo del Internet de las cosas. En la figura 51 se observa la interfaz de la aplicación que se enlaza con el equipo Sonoff.

Figura 51

Interfaz de control del dispositivo Sonoff 4CH PRO



3.6 Desarrollo e implementación del prototipo

Para el desarrollo e implementación de un prototipo con funcionalidades y características similares al Sonoff 4CHPro es necesario la creación tanto del hardware como el

software necesario para el funcionamiento. En la fase de desarrollo del hardware, se diseñan y seleccionan los componentes adecuados, se realiza el diseño del circuito y se lleva a cabo el ensamblaje y conexión de estos. Por otro lado, en la etapa de desarrollo del software, se programa el firmware que controla el prototipo, permitiendo la recepción y procesamiento de señales inalámbricas y se desarrolla la aplicación móvil.

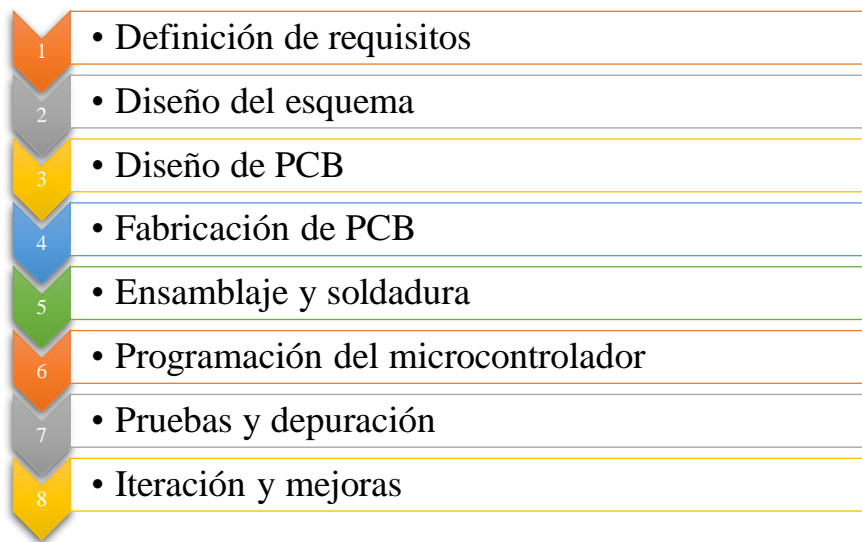
Estos dos aspectos, el hardware y el software, son fundamentales para lograr un prototipo funcional y eficiente, capaz de integrarse en sistemas de automatización del hogar, personalizar escenarios o proyectos similares. A continuación, el desarrollo.

3.6.1 Desarrollo del hardware para el dispositivo

Para el desarrollo correcto del prototipo es necesario llevar un orden de pasos que facilite la fabricación del hardware, es decir, seguir un procedimiento secuencial que permita establecer una visión clara de los objetivos y requisitos del proyecto, lo que facilita la toma de decisiones y la definición de los pasos a seguir. Para ello es necesario considerar la figura 52, que representa la serie de pasos que se pudieron considerar para facilitar el desarrollo del prototipo.

Figura 52

Pasos para el desarrollo del prototipo similar al dispositivo comercial



3.6.1.1 Definición de requisitos

En esta etapa inicial se definen los requisitos y funcionalidades específicas del prototipo. Esto incluye determinar el tipo de señales inalámbricas que se van utilizarán, como Wi-Fi, Bluetooth u otro protocolo, así como las características y compatibilidad con otros sistemas. Para ello se describen los requisitos y funcionalidades del prototipo.

Requisitos

- El control de dispositivos: Este prototipo, al igual que el dispositivo comercial, contará con la disponibilidad de cuatro puertos para dispositivos eléctricos.
- Conectividad inalámbrica: Debe contener la capacidad de comunicación inalámbrica para el control remoto, en este caso presentará dos alternativas, comunicación mediante frecuencias de radio a 433 MHz y mediante conexión Wi-Fi.
- Configuración sencilla: La configuración y vinculación con la red Wi-Fi debe ser intuitiva y fácil de usar con la red de Internet.

- Programación de horarios: La programación de horarios se debe a la automatización del control mediante tiempos establecidos para la activación de sus puertos, de tal modo que se generen escenarios de aplicación.
- Control remoto: Debe permitir el control remoto de los dispositivos, en el caso del uso de las frecuencias de radio a 433 MHz se empleará un prototipo de control transmisor que va a interactuar con el receptor principal, para el uso de 2.4 GHz se empleará una aplicación móvil que se desarrollará posteriormente, esta aplicación permitirá el control del receptor del prototipo.

Funcionalidades

- Control individual: El prototipo debe permitir controlar cada dispositivo de forma individual, así como crear grupos para controlar varios dispositivos al mismo tiempo.
- Temporizador y programación: Debe contar con la capacidad de establecer temporizadores y programar horarios para la activación de sus respectivos puertos de alimentación eléctrica de manera automática de manera controlada.
- Retroalimentación del estado: Se debe ver reflejado en tiempo real los cambios que se pueda realizar en el dispositivo en la aplicación móvil.
- Integración con asistentes de voz: Debe permitir la integración con asistentes de voz como Google Assistant o Amazon Alexa, lo que permitirá al usuario controlar los dispositivos mediante comandos de voz.

3.6.1.2 Diseño del esquema

Luego de haber establecido los requisitos y funcionalidades, se procede al diseño del esquema electrónico que será impreso en la placa de circuito. Esto implica identificar los

componentes necesarios, como microcontroladores, módulos de comunicación inalámbrica, relés u otros elementos según las necesidades del proyecto.

En el caso del receptor se tendrá en cuenta los siguientes dispositivos mencionados en la tabla 14 que serán de gran utilidad para su correcto funcionamiento.

Tabla 14

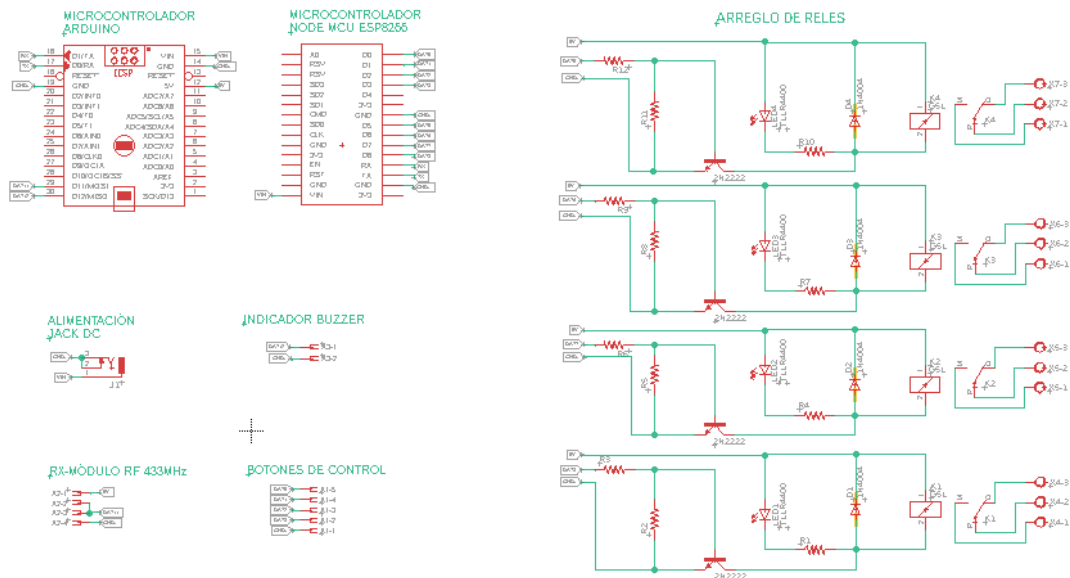
Componentes necesarios para el desarrollo del receptor

Requisito	Dispositivo	Funcionamiento
Comunicación inalámbrica	Módulos RF 433 MHz	Para que reciba las señales radio del control transmisor y poder interactuar con el dispositivo receptor
	Módulos 2.4 GHz	Permite la conexión Wi-Fi para el acceso a la interfaz de usuario de la aplicación móvil
Microcontroladores	Arduino Nano	Almacenará la programación para interactuar con el módulo RF. Proporcionará 5v que el relé necesita para ser activados.
	ESP8266	Permitirá la conexión Wi-Fi para interactuar desde la aplicación móvil
puertos físicos de conexión eléctrica	Toma de corriente tipo B	Para facilitar el uso de la energía eléctrica mediante la implementación de enchufes para sus puertos
Componentes para conexión del relé	transistores, resistencias, diodos	Destinados para el arreglo del circuito de relés
Indicadores	Leds y buzzer	Con los leds se podrá analizar de manera visual es estado de los puertos y con el buzzer poder escuchar los cambios de estado

Tomando en cuenta la disponibilidad de pines en los microcontroladores y buscando una forma organizar las conexiones de todos los elementos, se propone el siguiente modelo del esquema de conexiones para el circuito receptor en la figura 53.

Figura 53

Esquema de conexiones del receptor



En el caso de requerir el control por medio de frecuencias de radio de 433 MHz se podría considerar lo que se presenta en la siguiente tabla 15 para el desarrollo y correcto funcionamiento de un prototipo de control transmisor.

Tabla 15

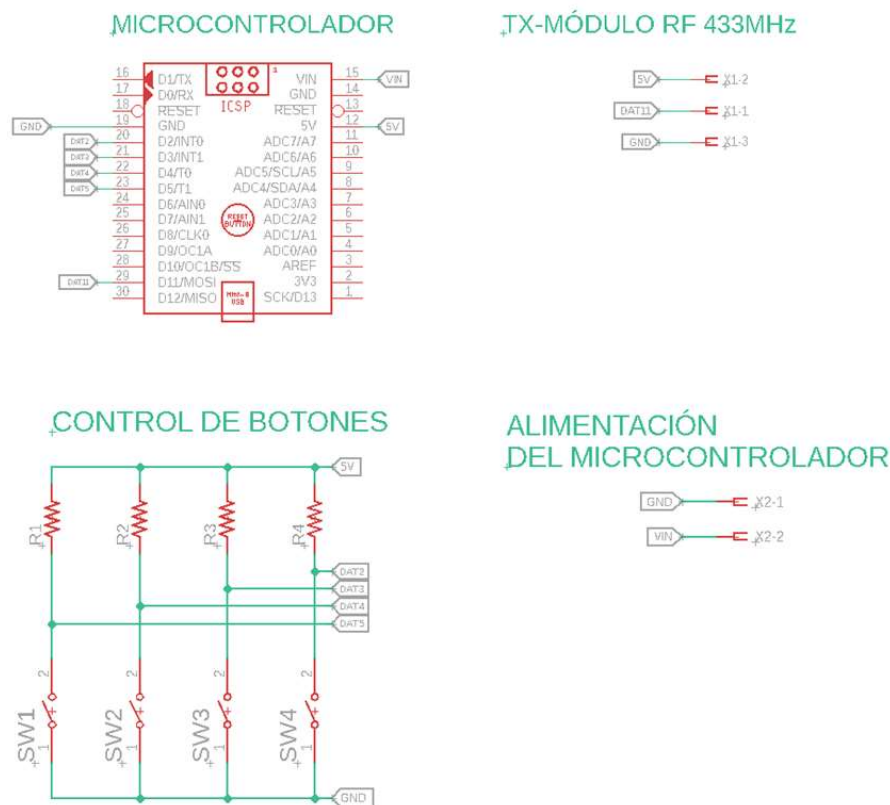
Componentes necesarios para un control RF

Requisito	Dispositivo	Funcionamiento
Microcontroladores	Arduino Nano	Almacenará la programación para interactuar con el módulo RF.
Botones	Botones 6x6x6	Pulsadores para emitir ordenes mediante ondas de radio luego de ser presionados
Comunicación inalámbrica	Módulos RF 433 MHz	Para transmitir las señales radio y poder interactuar con el dispositivo receptor

Luego de reconocer los elementos necesarios para el desarrollo de un control de radiofrecuencias, se propone el siguiente modelo de esquema de conexiones presentados en la figura 54.

Figura 54

Esquema de conexiones del transmisor RF



Cabe recalcar, que el diseño de la PCB requiere otro esquema de conexión que contiene todas las pistas y la forma de las placas que resultan del diagrama de conexión tanto para el transmisor como para el receptor, en el **Anexo A** es posible apreciar los resultados del diagrama de conexión de las pistas de los componentes del proyecto.

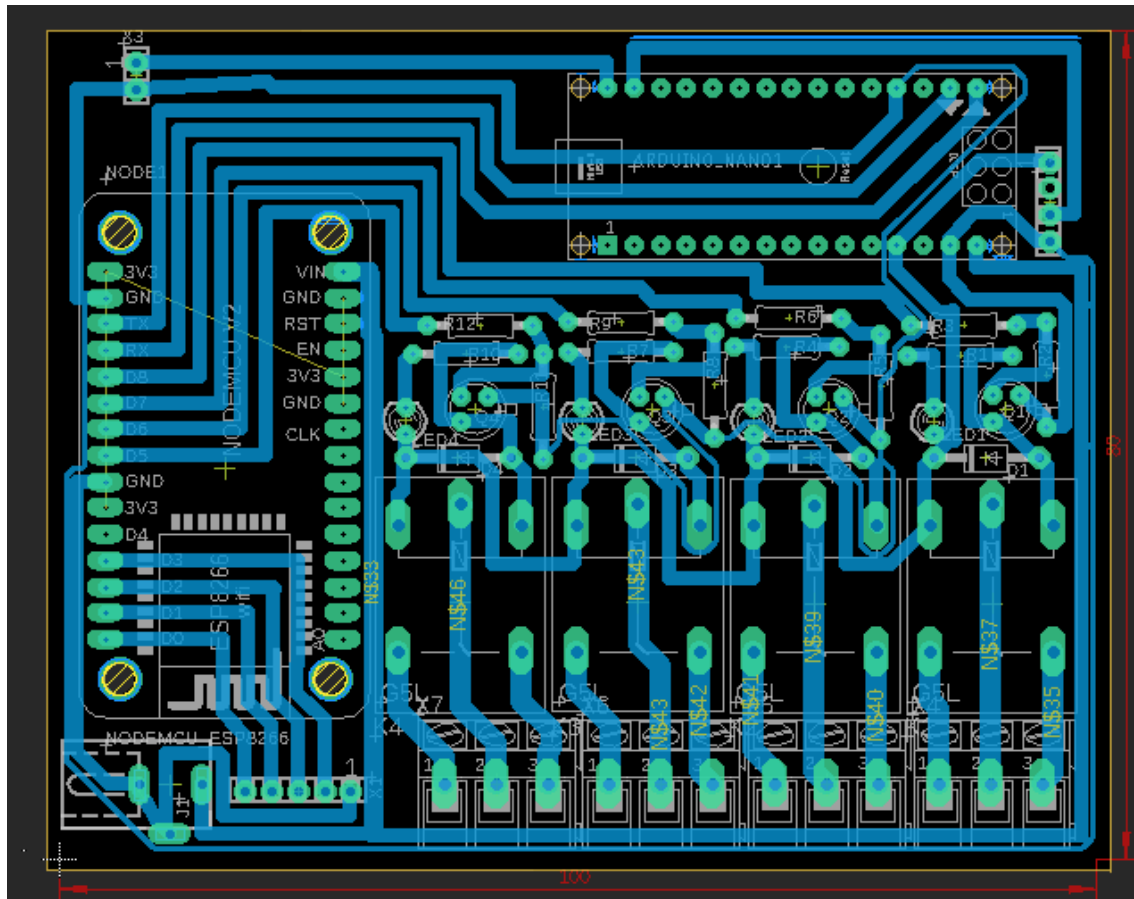
3.6.1.3 Diseño de PCB

Luego de haber definido el esquema de conexión de todos los dispositivos que formarán parte del prototipo, es necesario diseñar la placa de circuito impreso. Esto implica colocar los componentes en el diseño de la PCB y trazar las líneas de conexión, en este caso hay que tomar en cuenta el tamaño de los componentes para evitar superposiciones, realizar un correcto enrutamiento de las líneas de transmisión y aprovechar el menor espacio posible en las placas,

de este modo se estableció el siguiente diseño de PCB mostrado en la figura 55 que representa el lado del receptor.

Figura 55

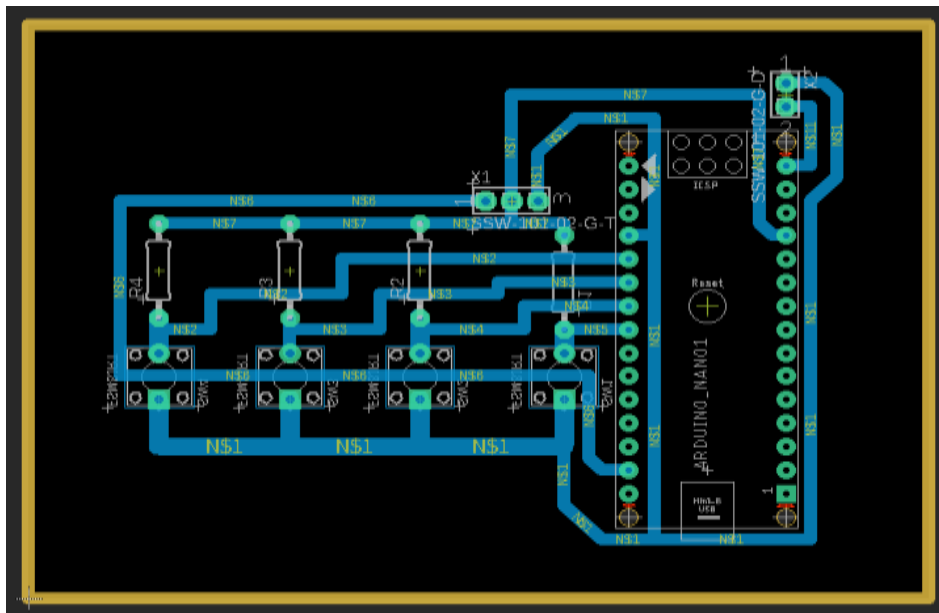
Diseño del modelo PCB del receptor



En el caso del diseño para el control de radio se presenta el siguiente diseño mostrado en la figura 56.

Figura 56

Diseño del modelo PCB del transmisor



3.6.1.4 Fabricación de PCB

El diseño establecido para el PCB presentado con anterioridad se convierte en una parte de gran importancia ya que a nivel de software se tendrá que extraer los datos de la plantilla del esquema para realizar la impresión del circuito impreso por medio de control numérico por computadora (CNC), esto permitirá crear las pistas conductoras en una placa de circuito impreso. Para ello es necesario transferir el diseño del circuito al software del control de la fresadora, esto permitirá generar las coordenadas para la fresadora de corte que cumplirá la función de extraer el exceso de cobre y crear las conexiones deseadas. La figura 57 representa parte del proceso de la impresión por CNC.

Figura 57

Fabricación de las placas PCB por medio de control numérico computarizado (CNC)

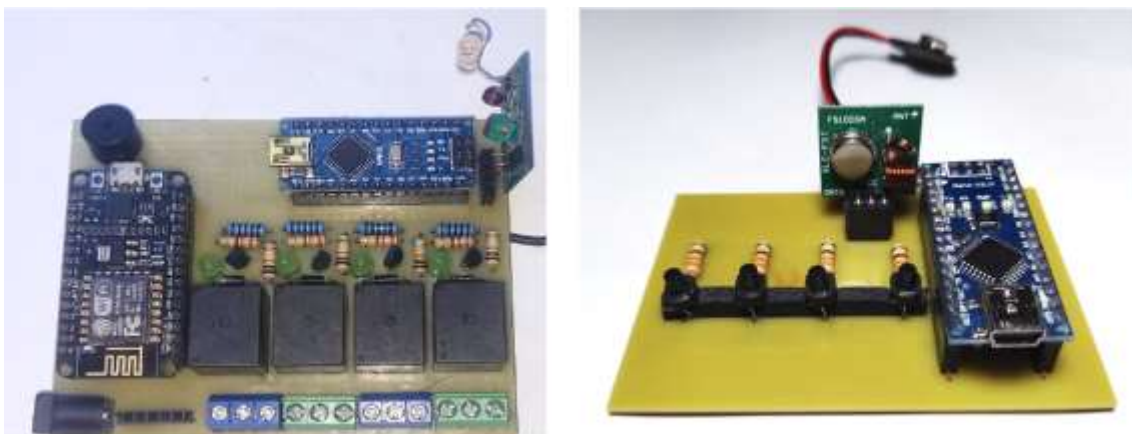


3.6.1.5 Ensamblaje y soldadura

Luego de haber extraído el modelo físico de la PCB del prototipo se procede a soldar los componentes en sus posiciones respectivas siguiendo el diseño y tomando la lista de materiales previamente establecida, en la figura 58 se muestran los resultados del ensamblaje y soldadura de los componentes.

Figura 58

Componentes ensamblados en las placas PCB del receptor y el transmisor



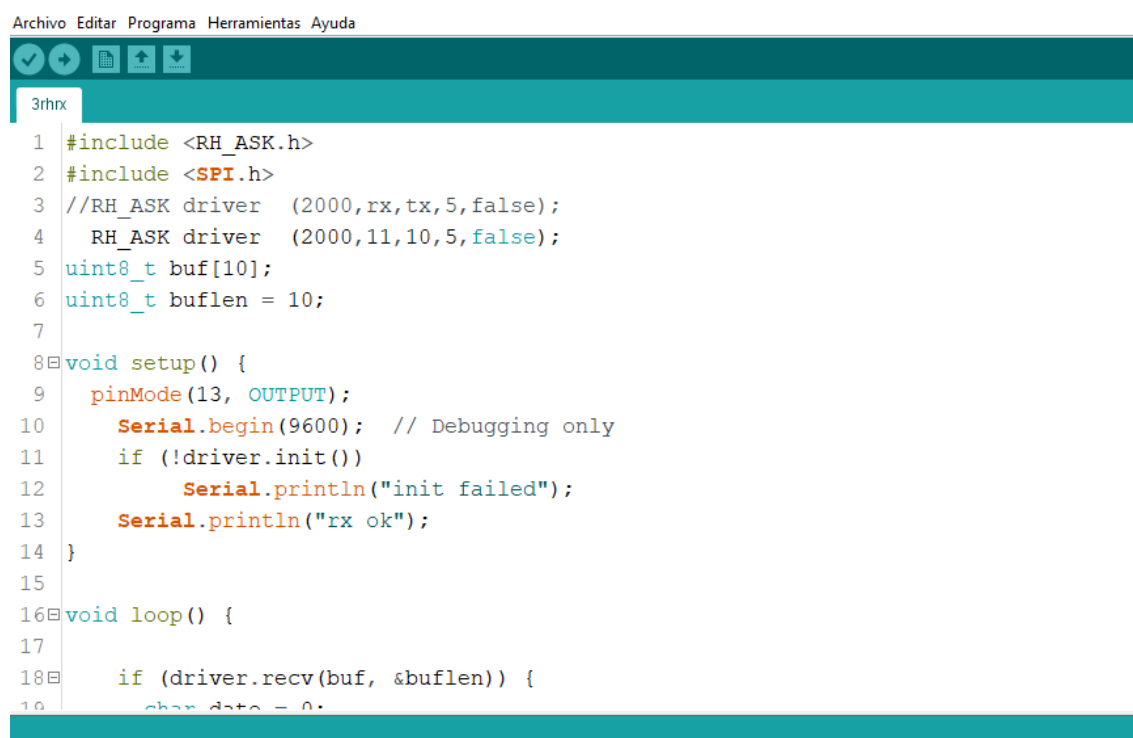
3.6.1.6 Programación del microcontrolador

El siguiente paso implica programar el microcontrolador que controlará el funcionamiento del receptor, lo cual involucra el desarrollo de un firmware personalizado en base a los requerimientos del prototipo y crear semejanzas en entre el funcionamiento del dispositivo comercial Sonoff 4CHPro y el prototipo en desarrollo. En este caso se presenta la programación que llevará el dispositivo microcontrolador Arduino Nano realizada en Arduino IDE destinada para el prototipo receptor ya que el arreglo de circuitos para activar el relé requiere de 5 V para cambiar de estado.

La programación del microcontrolador NodeMCU ESP8266 no está incluida en este apartado puesto que ese dispositivo servirá para el desarrollo de la aplicación móvil que será tratada en los siguientes temas relacionados con el desarrollo del software, a continuación, en la figura 59 se presenta las líneas de código para el receptor RF.

Figura 59

Entorno de programación del receptor en Arduino IDE



```

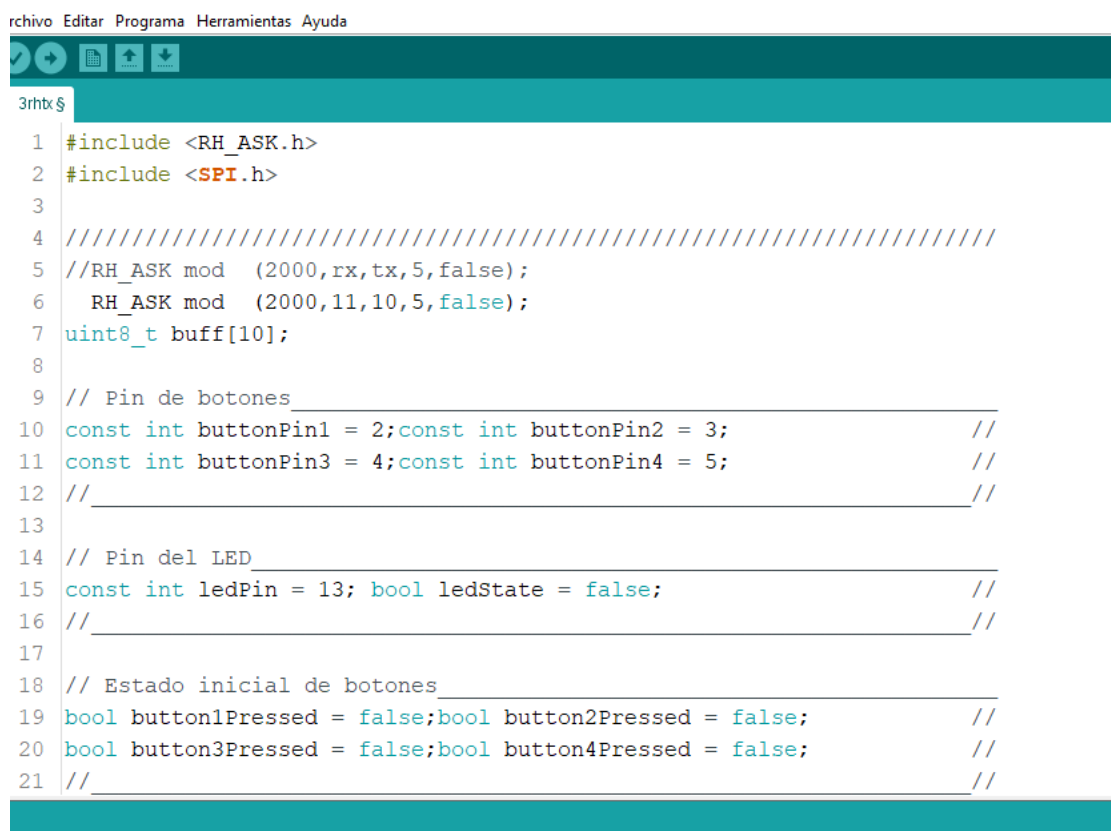
Archivo Editar Programa Herramientas Ayuda
3rhrx
1 #include <RH_ASK.h>
2 #include <SPI.h>
3 //RH_ASK driver (2000,rx,tx,5,false);
4 RH_ASK driver (2000,11,10,5,false);
5 uint8_t buf[10];
6 uint8_t buflen = 10;
7
8 void setup() {
9   pinMode(13, OUTPUT);
10   Serial.begin(9600); // Debugging only
11   if (!driver.init())
12     Serial.println("init failed");
13   Serial.println("rx ok");
14 }
15
16 void loop() {
17
18   if (driver.recv(buf, &buflen)) {
19     char data = 0;

```

El paso siguiente representa la programación del prototipo de control de radio transmisor para las frecuencias 433 MHz, ver figura 60. Cabe señalar que los códigos de ambos microcontroladores se pueden encontrar en el Anexo B.

Figura 60

Entorno de programación del transmisor en Arduino IDE



```

rchivo Editar Programa Herramientas Ayuda
3rhtx$
1 #include <RH_ASK.h>
2 #include <SPI.h>
3
4 ///////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
5 //RH_ASK mod (2000,rx,tx,5,false);
6   RH_ASK mod (2000,11,10,5,false);
7 uint8_t buff[10];
8
9 // Pin de botones
10 const int buttonPin1 = 2;const int buttonPin2 = 3;           //
11 const int buttonPin3 = 4;const int buttonPin4 = 5;           //
12 // _____ //
13
14 // Pin del LED
15 const int ledPin = 13; bool ledState = false;               //
16 // _____ //
17
18 // Estado inicial de botones
19 bool button1Pressed = false;bool button2Pressed = false;    //
20 bool button3Pressed = false;bool button4Pressed = false;    //
21 // _____ //
  
```

3.6.1.7 Pruebas y depuración

Una vez que se ha ensamblado y programado la placa de circuito, se llevan a cabo pruebas exhaustivas para verificar su funcionamiento y detectar posibles problemas o errores, lo cual implica probar la recepción y el control de señales, así como la interacción con otros sistemas o dispositivos.

Durante las pruebas se detectaron los siguientes eventos:

Tabla 16*Eventos detectados durante las pruebas*

Situación	Causa	Corrección
funcionamiento incorrecto del relé	Aplicación incorrecta del material de soldadura	Quitar excesos de aleación
Problemas para la comunicación en frecuencias 433 MHz	Construcción de la antena	Cálculos para fabricar una antena helicoidal mediante alambre de cobre considerando la frecuencia de operación, también se podría sustituir por una antena telescópica

3.6.1.8 Iteración y mejoras

En función de los resultados de las pruebas, se puede realizar ajustes y mejoras en el diseño y el firmware para optimizar el rendimiento y corregir posibles fallos o limitaciones.

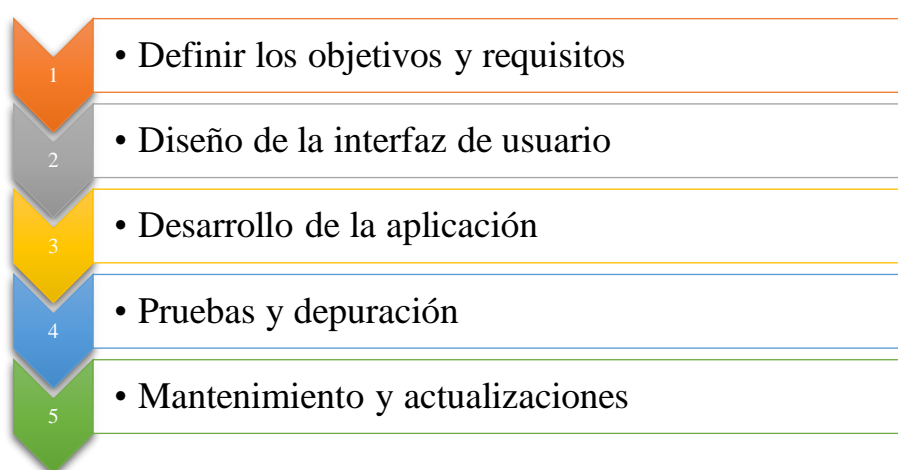
- Módulos de telefonía móvil GSM/GPRS: Estos son dispositivos de comunicación que permiten la conexión de dispositivos electrónicos a redes móviles, esto permitiría la transferencia de datos a través de la red móvil para emitir notificaciones por mensaje de texto en el caso de algún cambio de estado en el receptor.
- Sensores: A futuro para mejorar las características del prototipo se podría aplicar sensores como un micrófono para utilizar comandos de voz, sensores de humedad en caso de filtraciones de agua dentro del prototipo.
- Fortalecer la impermeabilidad: Para garantizar que una caja sea totalmente hermética para proteger una placa PCB de la humedad.
- Mejoras para el alcance del radiocontrol: módulos de radiofrecuencia para abarcar mayores distancias.

3.6.2 Desarrollo del software para el dispositivo

En este apartado se realiza el desarrollo del software del prototipo para así poder tomar el control a través de una aplicación móvil, para ello es necesario considerar una serie de pasos para crear y lanzar con éxito una aplicación para Smartphones. A continuación, en la figura 61 se describen los pasos generales involucrados en el desarrollo de una aplicación móvil:

Figura 61

Pasos para el desarrollo de la aplicación móvil del proyecto



3.6.2.1 Definir los objetivos y requisitos

En esta etapa inicial es importante definir los objetivos y requisitos de la misma. Esto implica determinar el propósito de la aplicación, las funcionalidades que se desean implementar y las plataformas móviles a las que se dirigirá.

Objetivos

- Control remoto de dispositivos: El objetivo principal sería permitir a los usuarios controlar los dispositivos conectados a través de la aplicación móvil, tanto localmente como de forma remota.
- Automatización de escenarios: La aplicación debería permitir la creación de escenas y rutinas para automatizar el funcionamiento de los dispositivos. Esto podría incluir

programar horarios de encendido/apagado, activación basada en eventos, control por voz.

- **Interfaz intuitiva:** El objetivo es proporcionar una interfaz de usuario intuitiva y fácil de usar en la aplicación móvil, que permita a los usuarios configurar y controlar los dispositivos de manera sencilla.
- **Integración con otras plataformas:** La integración con plataformas de terceros o servicios populares de automatización del hogar como Google Home o Amazon Alexa.

Requisitos

- **Desarrollo multiplataforma:** La aplicación debe ser compatible con múltiples plataformas móviles, como iOS y Android, para llegar a una audiencia más amplia.
- **Conexión y comunicación:** La aplicación debe ser capaz de establecer una conexión segura y confiable con los dispositivos Sonoff 4CH Pro a través de Wi-Fi u otras tecnologías de comunicación.
- **Autenticación y seguridad:** Es esencial implementar medidas de autenticación seguras para garantizar que solo los usuarios autorizados puedan acceder y controlar los dispositivos. Como también se deben aplicar las prácticas de seguridad para proteger la privacidad y la integridad de los datos.
- **Pruebas exhaustivas:** Para asegurarte de que la aplicación funcione correctamente en diferentes móviles, versiones de sistemas operativos y escenarios de uso

3.6.2.2 Diseño de la interfaz de usuario

Luego de que se tienen claros los requisitos y funcionalidades, se procede con el diseño de la interfaz de usuario, lo cual implica la creación de wireframes, prototipos y el diseño visual de la aplicación, es importante que la interfaz sea intuitiva, atractiva y fácil de usar para los usuarios. Para ello se propone el siguiente modelo a seguir en la figura 62 para la

implementación de los objetos necesarios que facilitarán la interacción con el dispositivo a través de una computadora o laptop y para la aplicación móvil se presenta el siguiente modelo mostrado en la figura 63.

Figura 62

Distribución de los componentes definida para el computador

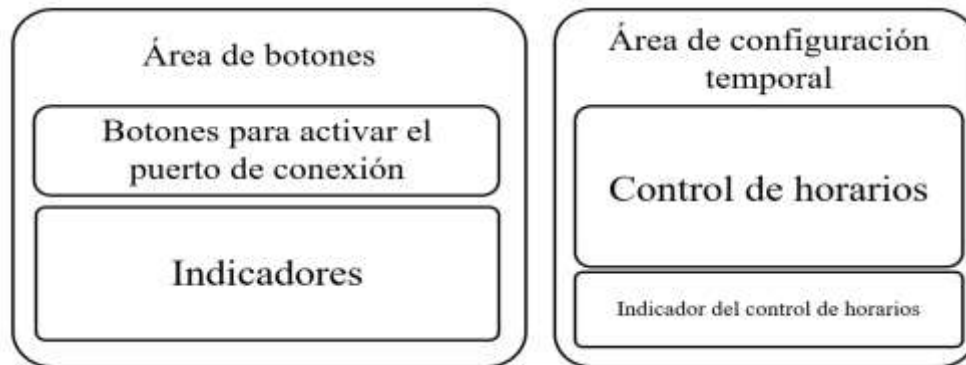
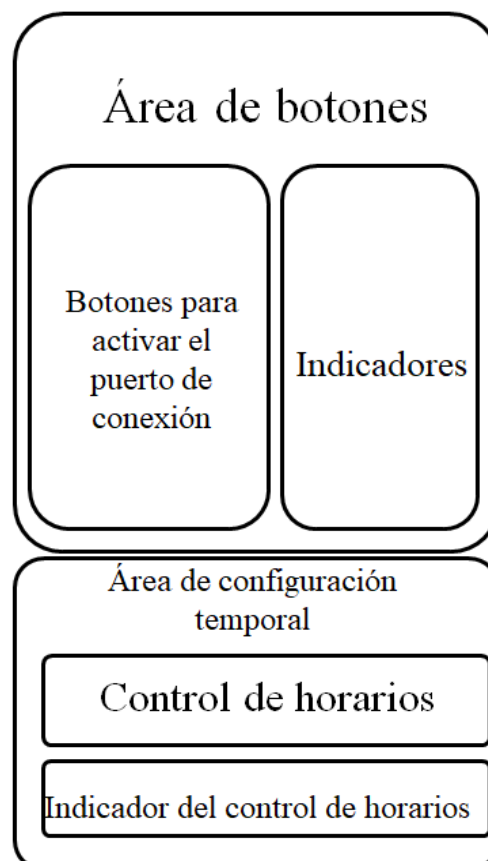


Figura 63

Distribución de los componentes definida para teléfonos móviles



De este modo se obtiene una idea de cómo se ubicarían los “widgets” que son elementos u objetos que permiten la iteración o el medio para visualizar datos en la aplicación Arduino IoT Cloud Remote. A continuación, en la tabla 17 se presentan aquellos que son relevantes para el proyecto en desarrollo.

Tabla 17

Elementos necesarios de Arduino IoT Cloud para el proyecto

Widgets	Función	Utilidad
Switch	Pulsadores	Pensado para mantener activado un punto de conexión en el tablero de distribución eléctrica hasta segunda orden
Push Button		Pensado para activar un punto de conexión en el tablero de distribución eléctrica bajo un horario establecido
Scheduler	Control de horarios	Facilita la configuración de los horarios mediante fechas de inicio y final, modo de repeticiones, tiempo de duración y selección de los días de la semana
Led	Indicador	Para identificar el estado de los puntos de conexión en el tablero de distribución eléctrica de manera visual
Chart	Registrar cambios	Para ver los cambios de estado realizados durante el día

3.6.2.3 Desarrollo de la aplicación

Como se mencionó en capítulos anteriores, la aplicación para el desarrollo se realizará mediante la aplicación Arduino IoT Cloud Remote que se puede encontrar en versiones online y para aplicación de escritorio, es una aplicación gratuita que permite el desarrollo de aplicaciones móviles compatibles con módulos Wi-Fi con el software, en este caso el módulo Node MCU esp 8266 el cual será vinculado con la aplicación para tener un control por medio de un Smartphone.

Para ingresar en el entorno de desarrollo de aplicaciones en IoT Cloud es necesario registrarse mediante el correo electrónico o simplemente vincular estos parámetros con la cuenta de Google y poder conceder el acceso para iniciar los proyectos, de este modo se

presenta en la figura 64 la primera ventana que se despliega en el desarrollador donde se encuentra la opción “Create Thing” que inicia la ventana de desarrollo.

Figura 64

Primera instancia



La figura 65 muestra ventana de desarrollo de aplicaciones, en ella se debe identificar las secciones donde van a realizar los primeros cambios las cuales se diferencian por tener las palabras grandes con una tonalidad más oscura que el resto, en las palabras **United** se debe ingresar algún nombre para identificar el proyecto, **Cloud Variables** permite la definición de las variables para el entorno de programación, **Associated Device** permite especificar el dispositivo de red conectado para la programación, en **Network** se puede ingresar los parámetros necesarios para conectarse a la red como el ID y la contraseña. De este modo se procede a iniciar con los cambios siguiendo el orden de las secciones que se identificaron con anterioridad.

Figura 65*Ventana de desarrollo*

Lo primero a realizar es ingresar como nombre “Tablero de distribución eléctrica” en el apartado **Untitled**, el siguiente procedimiento que se realizará es crear las variables de la programación que se vincularán con los widgets, para ello se debe dar clic en **Add Variable** para desplegar la ventana de creación de variables, ver figura 66.

Figura 66*Ventana para agregar las variables*

 The image shows a modal dialog box titled 'Add variable'. It contains the following fields and options:

- Name:** A text input field.
- Sync with other Things:** A toggle switch.
- Select variable type:** A dropdown menu.
- Declaration:** A text area.
- Variable Permission:** Radio buttons for 'Read & Write' (selected) and 'Read Only'.
- Variable Update Policy:** Radio buttons for 'On change' (selected) and 'Periodically'.
- Threshold:** A text input field with the value '0'.

 At the bottom of the dialog, there are two buttons: 'ADD VARIABLE' and 'CANCEL'.

Esta ventana contiene los siguientes parámetros, **Name** para identificar la variable con un nombre, **Select variable type** para desplegar las variables disponibles en la aplicación y especificar de que tipo será, en **Variable Permission** se presentan dos opciones, **Read & Write** que permite especificar si la variable trabajara tanto como entrada y salida y la opción **Read** que permite definir la variable como salida, en **Variable Update Policy** se presentan dos opciones para determinar los periodos de actualización de los cambios de la variable en la nube, **On Charge** permitirá actualizar la información de manera inmediata a diferencia de la opción **Periodically** que permite establecer horarios para la actualización del estado de la variable. Finalmente, se agregan las variables en la opción **Add Variable**. Considerando los parámetros antes mencionados se procede a incluir las siguientes variables con sus respectivas características siguiendo las especificaciones mostradas en la tabla 18.

Tabla 18

Definición de variables para la aplicación móvil

Name	Select Variable Type/Variable	Variable Permission	Variable Update Policy	Utilidad
Puerto1, Puerto2, Puerto3.	AlexaCompatible/ CloudSwitch	Read & write	On change	Se selecciona el tipo de variable compatible con Alexa para aprovechar su asistente de voz, se especifica que sea del tipo switch para vincular con un pulsador. Con esta variable se va a interactuar, por lo tanto, se concede el permiso Read & write, y se requiere la actualización de los cambios de manera inmediata, por lo tanto, se selecciona el campo On Change.
Puerto4	Time/ CloudSchedule	Read & write	On change	Se selecciona el tipo Time y se toma la opción Schedule que permite establecer horarios para controlar el estado de la variable. Al ser otra variable que sufrirá cambios se concede el permiso Read & write y se selecciona el campo On Change.

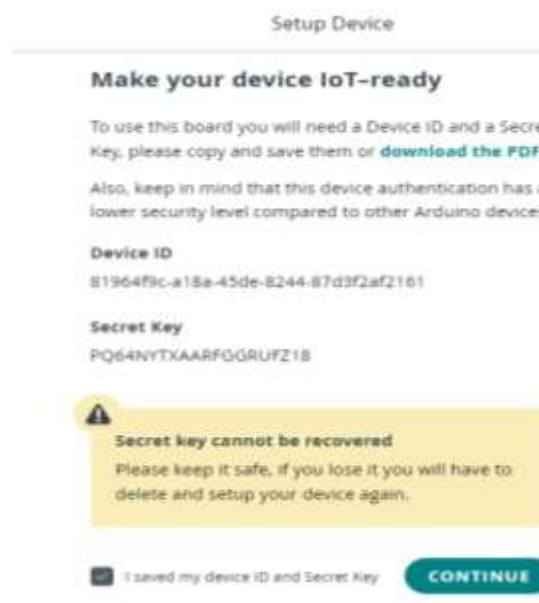
Indicador4	Basic Types/ Boolean	Read	On change	Se selecciona el tipo Basic y se toma la opción Boolean, esta variable se destina para la salida de datos, debido a esto se selecciona la opción Read, por otra parte, se selecciona la opción On Change para actualizar el estado en la nube de manera inmediata.
------------	-------------------------	------	--------------	--

El siguiente apartado que se modificará es **Associate Device**, en él se definirá el dispositivo de red que se empleará en este proyecto, para ello es necesario dar clic en el botón verde que se encuentra en esa sección y poder desplegar las opciones que permiten el registro del dispositivo de red. En esta nueva ventana que se despliega se debe dar clic en la opción **Set up new device**, luego se despliega otra ventana donde es necesario seleccionar **Third party device**, en la siguiente se especifica el modelo de microcontrolador que se va a usar, en este caso el **ESP8266** modelo **NodeMCU 1.0 (ESP-12E Module)** seguido de la opción **Continue**, se despliega una ventana para especificar el nombre del dispositivo, en este caso **ESP8266RX** seguido de la opción **Next**, este procedimiento se puede visualizar de manera gráfica en el **Anexo C**.

El anterior procedimiento era necesario para generar un ID y un código del dispositivo, es importante guardar estos parámetros ya que son necesarios para que el microcontrolador NodeMCU ESP8266 pueda conectarse a la red de Internet, para ello se puede copiar esta información o simplemente bajar el archivo PDF que se genera de manera automática, este documento presentará el nombre del dispositivo que se creó con anterioridad, el modelo de dispositivo seleccionado, fecha de creación del proyecto, el ID del dispositivo y el código secreto que se requiere para el siguiente paso que se realiza en la siguiente sección denominada **Network**, finalmente la opción **Continue**, ver figura 67.

Figura 67

ID y código del módulo NodeMCU ESP8266



Setup Device

Make your device IoT-ready

To use this board you will need a Device ID and a Secret Key, please copy and save them or [download the PDF](#).

Also, keep in mind that this device authentication has a lower security level compared to other Arduino devices.

Device ID
E1964F9c-a18a-45de-B244-87d3f2af2161

Secret Key
PQ64NYTXAARFGGRUFZ1B

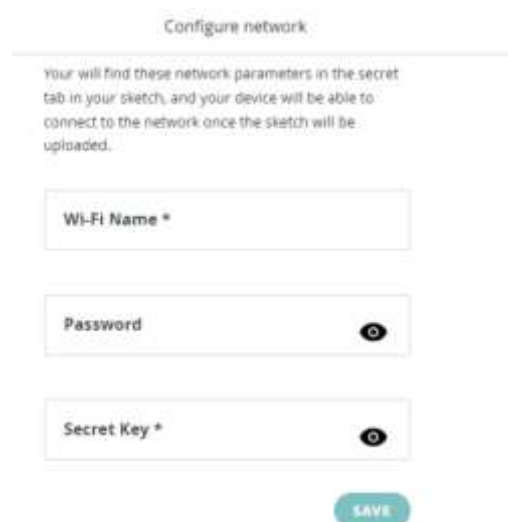
Warning: Secret key cannot be recovered. Please keep it safe, if you lose it you will have to delete and setup your device again.

I saved my device ID and Secret Key **CONTINUE**

Luego de haber obtenido los parámetros ya mencionados, se procede a realizar los cambios en la sección **Network**, aquí se debe ingresar el nombre de la red, la contraseña y el código del dispositivo de red, ver figura 68.

Figura 68

Configuración para la conexión a Internet



Configure network

Your will find these network parameters in the secret tab in your sketch, and your device will be able to connect to the network once the sketch will be uploaded.

Wi-Fi Name *

Password

Secret Key *

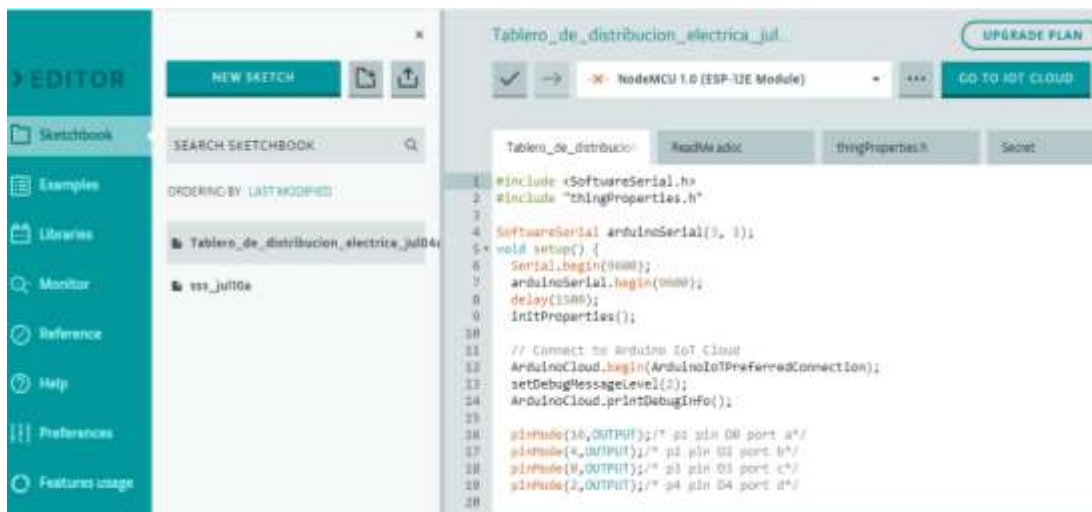
SAVE

Luego de haber cumplido con todos esos pasos se procede a ingresars en el entorno de programación, para ello es necesario situar la opción **Sketch** que se encuentra a la derecha del

nombre del proyecto, en ese punto se va a desplegar una nueva ventana donde se seleccionará la opción **Open full editor** que permite abrir la ventana que se muestra en la figura 69 para la programación de la tarjeta de red, en el **Anexo D** se puede apreciar todo el código de programación.

Figura 69

Ventana de programación de la aplicación



Finalmente, se crea el dashboard para implementación de los widgets que servirán para interactuar con la aplicación móvil, para ello es necesario situar la opción **Dashboards** que se encuentra en la parte superior de la ventana principal, seguido de la opción **Build Dashboard**, aquí se despliega la ventana de configuración donde es necesario asignar un nombre a esta interfaz en la opción **United**, para luego agregar los Widgets de los cuales se seleccionarán del tipo Switch, Scheduler y led que se encuentran al dar clic en la opción **Add**, en el **Anexo E** se encuentra este procedimiento de manera gráfica, mientras que la Figura 70 representa el esquema final del dashboard de la aplicación para el control mediante un ordenador, por otra parte, la figura 71 representa el dashboard que se puede observar en un Smartphone.

Figura 70

Componentes distribuidos para la vista desde el computador

**Figura 71**

Componentes distribuidos para la vista desde el Smartphone



3.6.2.3.1 Configuración de la App Amazon Alexa

La razón de usar la aplicación Amazon Alexa es para poder tener acceso al asistente de voz como otra alternativa de control donde se ocuparán dispositivos móviles con sistema operativo de Android para estas pruebas.

Lo primero que se debe realizar es ir a Google Play, buscar la aplicación **Amazon Alexa** y descargar la App, en este punto se requiere vincular los Widgets que se crearon en la aplicación Arduino IoT Cloud, para ello se debe situar el menú de opciones que se encuentra al abrir la aplicación, en esta se debe ubicar la opción **Más** para desplegar el contenido de esa opción, luego se debe identificar el botón **Skills y juegos**, y en el buscador que aparece escribir la palabra “Arduino” e instalar esa extensión.

Lo segundo es ubicar la opción **dispositivos** que se encuentra en la parte inferior del menú de opciones, luego identificar la opción **Editar** que aparece en esa ventana, en esta se aparecen los nombres de los Widgets que se crearon en el desarrollador de aplicaciones y se los incluye en favoritos.

Por último, se debe ubicar la opción **Inicio** en la parte inferior del menú de opciones de la App para luego hacer uso de la aplicación como tal, en este caso se tienen dos métodos de control, la primera es por medio de la opción **Interruptores** que contiene los Widgets que controlan el dispositivo, o la segunda alternativa que implica entrenar la aplicación por comandos de voz para hacer uso del asistente de voz de la aplicación donde es necesario decir el nombre del Widgets para activarlo, en la figura 72 muestra la interfaz de control que se muestra luego del procedimiento antes realizado.

Figura 72

Interfaz de control en Amazon Alexa



3.6.2.4 Pruebas y depuración

Una vez que se ha desarrollado la aplicación, se llevan a cabo pruebas exhaustivas para detectar errores y asegurarse que la aplicación funcione correctamente en diferentes dispositivos móviles y sistemas operativos. Se pueden realizar pruebas manuales y automatizadas para verificar el rendimiento, usabilidad y estabilidad de la aplicación.

3.6.2.5 Mantenimiento y actualizaciones

Una vez que la aplicación haya funcionado correctamente, es necesario realizar un seguimiento del rendimiento y atender cualquier problema o error que pueda surgir, además se pueden realizar actualizaciones periódicas para agregar nuevas funcionalidades, mejorar la experiencia del usuario y abordar cualquier problema que se haya identificado.

Para ello, se puede emplear la librería ArduinoOTA incluida en el entorno de desarrollo de Arduino que proporciona una manera sencilla de realizar las actualizaciones en el

dispositivo ESP8266, entre las funciones contenidas en la librería se encuentran las siguientes mostradas en la siguiente tabla 19.

Tabla 19

Mejoras para el dispositivo en desarrollo

Librería	Función	Utilidad
	ArduinoOTA.setPort()	Se emplea para especificar el puerto de escucha para las actualizaciones OTA
	ArduinoOTA.setHostName()	Se utiliza para establecer un nombre de host en el prototipo receptor que sirva para poder identificarlo de una única manera
	ArduinoOTA.begin()	Se utiliza para iniciar el servicio OTA (Over-The-Air) en Arduino para dispositivos ESP, esto habilita la capacidad de recibir actualizaciones de firmware a través de la red Wi-Fi
#include <ArduinoOTA.h>	ArduinoOTA.setPassword()	Se emplea para establecer una contraseña de autenticación en el servicio OTA, esto permite asegurar que solo las personas autorizadas puedan acceder y enviar actualizaciones al dispositivo
	ArduinoOta.onStart()	Es un callback que se utiliza para realizar acciones personalizadas al inicio de una actualización de firmware como, etiquetas con mensajes, el progreso de la actualización o ejecutar otras líneas de programación durante el proceso de actualización

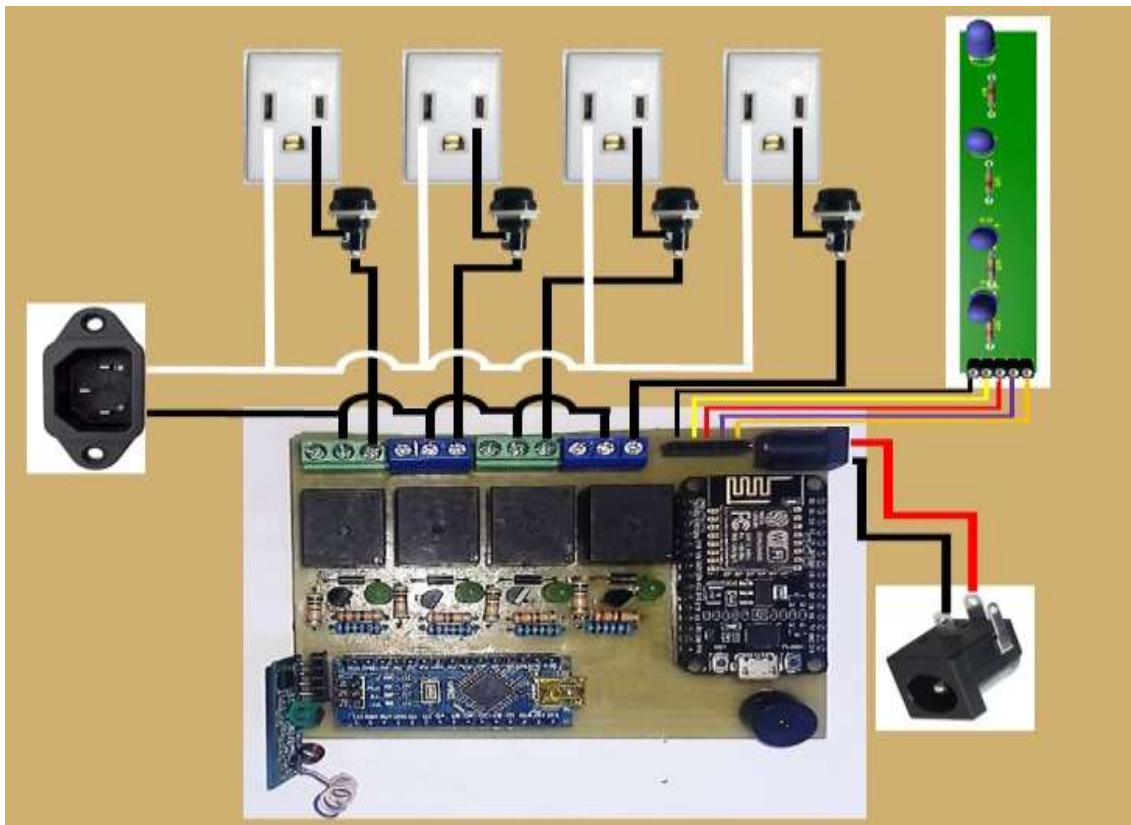
3.6.3 Implementación de componentes externos para receptor

Luego de ensamblar los componentes eléctricos en la placa PCB del receptor, se llevará a cabo la implementación de los últimos componentes para el dispositivo final que se va a constituir por el cableado eléctrico y los interruptores americanos tipo B para tener acceso directo con la salida de los puertos del PCB con respecto a los relés de este, lo cual implica considerar una alimentación de corriente alterna para el paso de la energía a través de los interruptores antes mencionados y se logrará por medio de un conector hembra para corriente alterna.

Para el cuidado de los equipos que se conectarán a la energía a través de sus puertos de conexión se toma en cuenta el componente portafusibles para limitar la corriente que pueda pasar hacia las salidas de los interruptores, caso contrario, un exceso de corriente quemará el fusible interno del componente.

se toma una serie de indicadores led para apreciar en la parte externa el estado de los puertos de conexión tal y como se implementó en el circuito PCB, y para alimentar la placa en general se toma otro Jack DC para la parte externa de la carcasa del dispositivo en desarrollo, todas las especificaciones de los componentes en mención se encuentran en la siguiente tabla.

El esquema de conexiones del dispositivo final está definido en la figura 73.

Figura 73*Esquema de conexiones del dispositivo final*

Todo el circuito final será incorporado en el interior de una caja plástica que dispondrá las ranuras necesarias para los componentes que van a sobresalir de la carcasa plástica, la parte izquierda de la figura 74 corresponde a la vista final del dispositivo y en la parte derecha el prototipo transmisor y el **ANEXO F** muestra la parte interna para visualizar las conexiones internas del dispositivo.

Figura 74

Presentación final del Prototipo de dispositivo IoT



3.6.4 Costos de implementación

El desarrollo e implementación de todo el proyecto conlleva una planificación exhaustiva de diseño tanto para el circuito a nivel de software, como para el dispositivo final y se deberá para tomar los componentes necesarios para tener resultados favorables en las pruebas finales en comparación al funcionamiento del dispositivo comercial, siendo una opción más económica que el equipo comercial y presenta mejoras para su uso gracias a la las conexiones y tomas de corrientes incorporadas en este, por otra parte la implementación de los fusibles permite asegurar la vida útil de los componentes conectados ante variaciones de corrientes, cabe señalar que el *Prototipo de dispositivo IoT*, como se denominó este proyecto podrá ser direccionado para el uso en varias áreas de trabajo como en hospitales, en la industria, en el área agrícola, etc. En las siguientes tablas se muestra el costo del dispositivo final.

Tabla 20*Costos de implementación del transmisor RF*

Componente	Precio unitario	Unidades	Precio total
Arduino nano	\$ 12,00	1,00	\$ 12,00
Resistencias	\$ 0,05	4,00	\$ 0,20
Módulo RF 433MHz TX	\$ 4,40	1,00	\$ 4,40
Botones	\$ 0,10	4,00	\$ 0,40
Precio total			\$ 17,00

Tabla 21*Costos de implementación del receptor*

Componente	Precio unitario	Unidades	Precio total
NodeMCU ESP8266	\$ 10,00	1,00	\$ 10,00
Arduino Nano	\$ 12,00	1,00	\$ 12,00
Módulo RF 433 RX	\$ 4,40	1,00	\$ 4,40
Led	\$ 0,15	8,00	\$ 1,20
Reles	\$ 0,50	4,00	\$ 2,00
Resistencias	\$ 0,05	16,00	\$ 0,80
Jack DC	\$ 0,50	2,00	\$ 1,00
Diodos 1N4004	\$ 0,05	4,00	\$ 0,20
Transistor 2n2222a	\$ 0,30	4,00	\$ 1,20
Costos de PCB	\$ 20,00	1,00	\$ 20,00
Precio total			\$ 52,80

Tabla 22*Costos totales para el tablero de distribución eléctrica*

Componente	Precio unitario	Unidades	Precio total
Portafusibles	\$ 0,75	4,00	\$ 3,00
Fusibles	\$ 0,25	4,00	\$ 1,00
Conector hembra AC	\$ 1,00	1,00	\$ 1,00
Receptor	\$ 52,80	1,00	\$ 52,80

Enchufe de corriente	\$	0,80	4,00	\$	3,20
Carcasa	\$	5,00	1,00	\$	5,00
Precio total				\$	66,00

Finalmente, los costos para el análisis de las señales y equipamiento para el desarrollo del área de pruebas del laboratorio se pueden observar en la tabla 23.

Tabla 23

Costos totales del proyecto

Componente	Precio unitario	Unidades	Precio total
HackRF One	\$ 324,95	1,00	\$ 324,95
RTL-SDR	\$ 39,00	1,00	\$ 39,00
Dispositivo IoT			
Sonoff	\$ 70,00	1,00	\$ 70,00
Costos de envíos	\$ 25,00	1,00	\$ 25,00
Tablero de distribución eléctrica	\$ 66,00	1,00	\$ 66,00
Computadora	\$ 626,00	1,00	\$ 626,00
Precio total			\$ 1.150,95

3.7 Área de laboratorio destinada para aplicación del prototipo de dispositivo IoT y el dispositivo Sonoff 4CH Pro

En el laboratorio, se procederá con la instalación y fijación de los dispositivos IoT en una superficie vertical para el soporte de estos de tal forma que tenga acceso a la energía eléctrica para que sean alimentados, tal y como se muestra en la figura 75. Además de eso, se ubicará la computadora junto a los SDR en un escritorio para llevar a cabo las pruebas de ataques y análisis de vulnerabilidades. Esta configuración permitirá un entorno controlado, para la ejecución de las actividades con fines educativos y académicos. El trabajo realizado en el

laboratorio se enfocará en el aprendizaje y la investigación, promoviendo un estudio responsable y ético en el campo de la seguridad cibernética y las tecnologías IoT.

Figura 75

Ubicación de los dispositivos IoT en el Laboratorio



Al instalar los dispositivos IoT conectados a la red eléctrica se crea una oportunidad para aprovechar esa infraestructura para alimentar otros equipos. Esta estrategia permitiría suministrar energía a dispositivos como antenas, routers, cámaras, fusionadoras, medidoras de señales, OTDR y OPM, entre otros, que requieren una carga eléctrica. Al compartir la infraestructura de alimentación, se simplifica la gestión de energía y se crea un ambiente organizado y eficiente en el laboratorio. Además, el monitoreo y control de la energía a través de los dispositivos IoT permite optimizar el consumo de energía y promover una mayor eficiencia energética en el entorno de pruebas y análisis con fines educativos y académicos.

De este modo, en la figura 76 se muestran las opciones en las que sería útil la implementación de estos dispositivos IoT en un entorno real.

Figura 76

Escenarios posibles de aplicación para los dispositivos IoT



CAPITULO IV

4 Pruebas y resultados de la propuesta

En este apartado, se presentan los resultados obtenidos mediante el uso de equipos IoT en las pruebas realizadas como parte de la propuesta de investigación. Tomando en cuenta los microcontroladores Arduino y NodeMCU ESP8266 quienes han demostrado ser herramientas valiosas para el análisis de señales inalámbricas y el estudio de vulnerabilidades en sistemas de comunicación, por otra parte.

La flexibilidad y la capacidad de los dispositivos SDR permitirán la captura y procesamiento de señales en frecuencias específicas, como 433 MHz y 2.4 GHz mediante la implementación de un entorno controlado y éticamente autorizado posibilitarán el descubrimiento de posibles vulnerabilidades, debilidades y posibles oportunidades de mejora en la protección de la privacidad y seguridad en las comunicaciones inalámbricas.

4.1 Funcionamiento de los dispositivos IoT

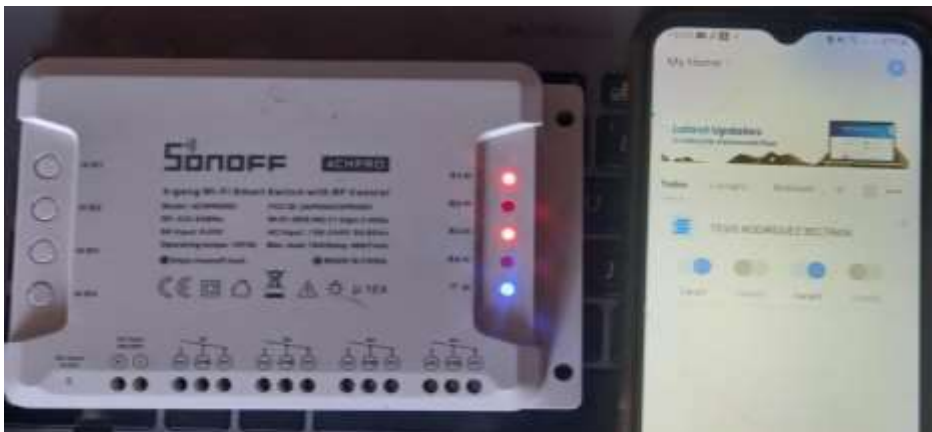
A continuación, se evalúa el funcionamiento final del dispositivo IoT comercial y el tablero de distribución eléctrica en desarrollo.

4.1.1 Pruebas de la comunicación entre un Smartphone y el Sonoff 4CH PRO

Entre las pruebas de funcionamiento del dispositivo comercial se ha logrado comprobar la respuesta inmediata ante los cambios de estados presentes en la aplicación móvil Ewelink que es la aplicación móvil dedicada al control de los dispositivos SonOff, en este caso, el modelo 4CH PRO, ver figura 77.

Figura 77

Pruebas de funcionamiento del equipo comercial



4.1.2 Pruebas de la comunicación entre aplicación diseñada y el Prototipo de dispositivo IoT

En las siguientes pruebas que se han realizado por medio de Arduino IoT Cloud se puede apreciar como ocurre la interacción entre la computadora o el Smartphone y el dispositivo final, para ello se realiza una serie de pruebas en la transmisión y determinar el uso correcto de la información que receipta el tablero de distribución eléctrica. En la figura 78 se evidencia el estado del puerto de conexión eléctrico en la placa PCB y en la aplicación vista desde una computadora. Mientras que en el **Anexo G** se muestra el dispositivo final con una carga conectada en el puerto de conexión.

Figura 78

Pruebas de funcionamiento del prototipo de dispositivo IoT



4.2 Vulnerabilidades detectadas en la comunicación

Una vulnerabilidad detectada durante la transmisión fue detectar una librería de programación que permite las comunicaciones inalámbricas a través de módulos de radiofrecuencias y presenta una escasa seguridad como es el caso de la librería *RadioHead*, que es muy utilizada al ser compatibles con módulos RF de bajo costo. Cabe señalar que es responsabilidad del desarrollador implementar medidas de seguridad dependiendo las necesidades del proyecto.

4.2.1 Protocolos de transmisión RF débiles

Mediante el dispositivo RTL-SDR es posible realizar un escaneo de las librerías que están empleando los dispositivos de radiofrecuencias a través del software RTL-433 debido a su compatibilidad con este, actualmente permite detectar alrededor de 246 protocolos y que, a su vez ha permitido identificar el tipo de librería que emplea el microcontrolador para establecer una comunicación inalámbrica como fue la librería *RadioHead*, que se implementó con fines demostrativos para poder visualizar el comportamiento del software.

Esta librería es utilizada en los dispositivos que funcionan mediante un control RF que facilita la interacción de manera remota en varios escenarios de aplicación como la activación de los mecanismos de apertura de compuertas, habilitar sensores, en el control de maquinarias o diversos escenarios donde se necesitan entrar en funcionamiento algún mecanismo luego de haber recibido una orden establecida por el control de radio que los activa, siendo un blanco fácil de atacar al ser vulnerable ante ataques de repetición, de este modo. La figura 79 presenta los resultados que se obtuvieron al estar manipulando los botones del control transmisor que fue evaluado en su frecuencia establecida 433,92 MHz.

Figura 79

Identificación de librerías vulnerables mediante el hardware RTL-SDR y el software RTL-433

```

rtl_433 version unknown inputs file rtl_tcp RTL-SDR
Trying conf file at "C:\Users\Bryan\Downloads\rtl_433_win_2019-08-19\rtl_433.conf"...
Trying conf file at "C:\Users\Bryan\AppData\Local\rtl_433\rtl_433.conf"...
Trying conf file at "C:\ProgramData\rtl_433\rtl_433.conf"...

Consider using "-H newmodel" to transition to new model keys. This will become the default someday.
A table of changes and discussion is at https://github.com/merbanan/rtl\_433/pull/986.

Registered 104 out of 134 device decoding protocols [ 1-4 8 11-12 15-17 19-21 23 25-26 29-36 38-60 63 67-71 73-100 102-1
03 108-116 119 121 124-128 131-134 ]
Found Rafael Micro 4820F tuner
Exact sample rate is: 250000.000414 Hz
[882XX] PLL not locked!
Sample rate set to 250000 S/s.
Tuner gain set to Auto.
Tuned to 433.92MHz.

time      : 2023-07-25 05:50:48
model     : Rafael ASK Data len : 40          To : 10
from      : 0          Id          : 1
flags     : 0          Payload     : 0, 0, 64, 0, 0, 1, 0, 4, 5, 13, 247, 90, 130, 125, 250, 50, 220, 215, 70, 251, 167,
220, 230, 60, 140, 15, 15, 20, 30, 61, 100, 8, 255, 2, 15, 0, 252, 0, 0, 2, 150, 220, 150, 0, 0, 0
integrity : CRC

time      : 2023-07-25 05:50:53
model     : Rafael ASK Data len : 40          To : 10
from      : 0          Id          : 1
flags     : 0          Payload     : 0, 0, 64, 0, 0, 1, 0, 4, 5, 13, 247, 90, 130, 125, 250, 50, 220, 215, 70, 251, 167,
220, 230, 60, 140, 15, 15, 20, 30, 61, 100, 8, 255, 2, 15, 0, 252, 0, 0, 2, 150, 220, 150, 0, 0, 0
integrity : CRC

```

Cabe señalar que una comunicación con este tipo de protocolos para transmisiones de radiofrecuencias puede ser susceptible a ataques de repetición de señal o modificación de datos, y se ha logrado evidenciar que también se puede generar una señal de ruido para hacer perder la comunicación entre el control y el sector RF del dispositivo desarrollado como receptor. Es importante indicar que el software no proporciona un medio para realizar un ataque de manera directa en la comunicación más bien, sirve para identificar que protocolos de comunicación se

están empleando en la transmisión para considerar la existencia de un sistema vulnerable al emplear este tipo de protocolos.

4.3 Análisis de ataques ejecutados

Es importante tener en cuenta que este análisis solo representa una situación específica en un entorno controlado de laboratorio. La seguridad en los dispositivos IoT es un desafío constante, debido a que esta tecnología está expuesta a una gran variedad de ataques, aunque en este caso puntual se ejecutaran con el fin de evaluar las vulnerabilidades presentes en esta tecnología.

4.3.1 Ataques realizados al dispositivo Sonoff 4CH PRO

Para ejecutar este ataque se hace uso de un equipo comercial Sonoff 4CHPRO para llevar a cabo experimentos de laboratorio con el objetivo de explorar posibles debilidades y evaluar la efectividad de un inhibidor como herramienta de ataque, ver figura 80.

Figura 80

Interrupción de la comunicación entre el Smartphone y el Sonoff 4CH PRO



Con este escenario se demostró que los dispositivos poseen una debilidad en su sistema ante la presencia de ataques que afectan las comunicaciones inalámbricas del sistema IoT que se está empleando, en este caso. Al iniciar el programa en GNU Radio se pudo verificar luego de un instante la pérdida de conexión del dispositivo físico, mientras que en la aplicación móvil surgió un mensaje en la parte céntrica del Smartphone que indicaba la pérdida de la señal. Cabe señalar que la reconexión de los dispositivos puede quedarse intermitente, en referencia a la conexión y desconexión de la red en los dispositivos, donde es necesario apagar el dispositivo router Wi-Fi principal para reestablecer de mejor manera la red de Internet.

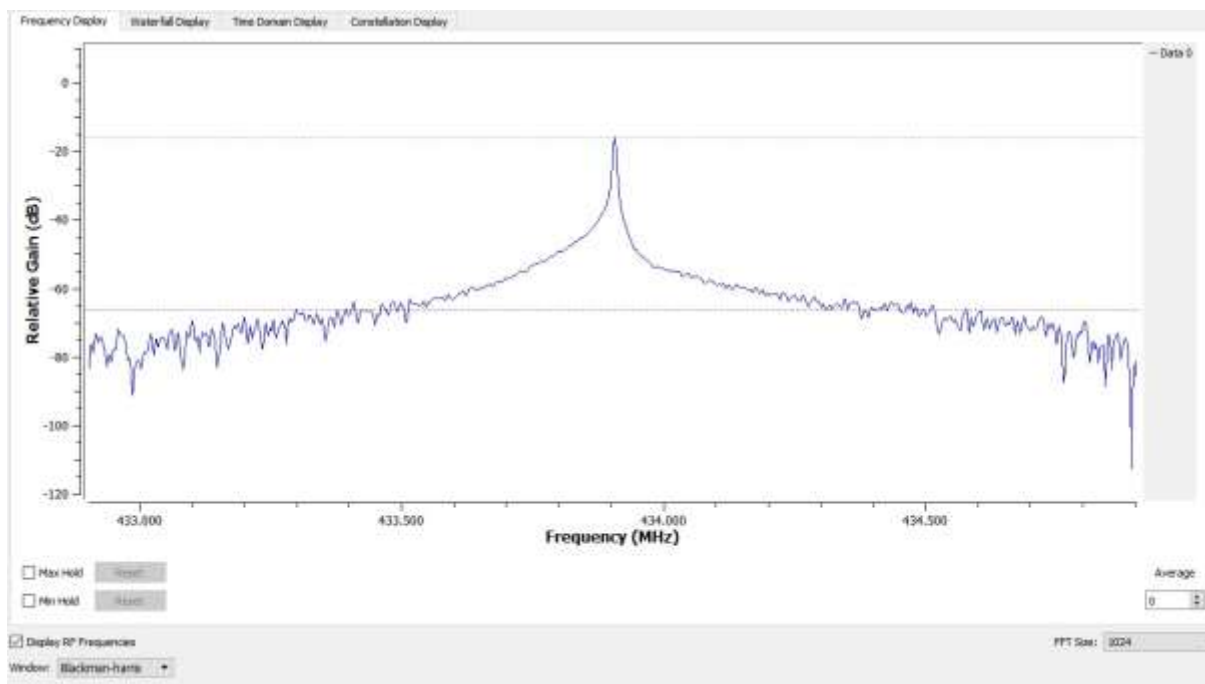
4.3.2 Ataques realizados al Prototipo de dispositivo IoT.

En este escenario se pone en funcionamiento la comunicación en frecuencias de 433MHz ya que es uno de los métodos de control del dispositivo, en este caso se realiza una captura de datos transmitidos a través del control que activará alguno de los puertos dispuestos en el prototipo. Para ello, la configuración de bloques en GNU Radio permitirá a través del hardware RTL-SDR capturar los datos que posibilitan la interacción con el prototipo y así, almacenar los datos para usarlos en algún otro momento.

Lo primero que se debe hacer es activar los puertos de conexión eléctrica mediante el control RF de manera intencionada para realizar la captura y almacenamiento de los datos durante la transmisión mediante el diagrama de bloques que se presentó en la figura 44, mientras que en la figura 81 se observa el instante en que el botón del transmisor RF es pulsado y reflejado en la gráfica del espectro a través del software. En esta imagen es posible observar el comportamiento de la señal capturada en función de la frecuencia, se puede observar que el espectro de la señal presenta ruido.

Figura 81

Señal emitida por el transmisor y receptada por el RTL-SDR

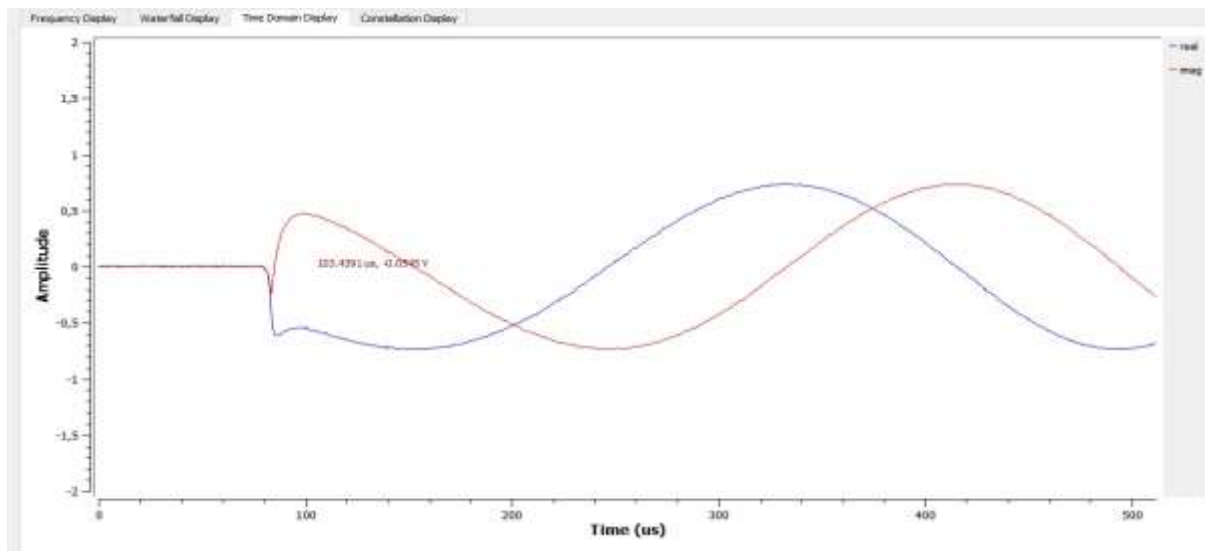


Como se observó anteriormente el flujograma permite el análisis de la señal para ver el comportamiento en frecuencia, en función del tiempo y el respectivo diagrama de constelación.

En la figura 82, se puede visualizar la señal analógica que corresponde al comportamiento de la señal receptada en función del tiempo, donde se observa la componente real representada en color azul y la componente imaginaria en color rojo.

Figura 82

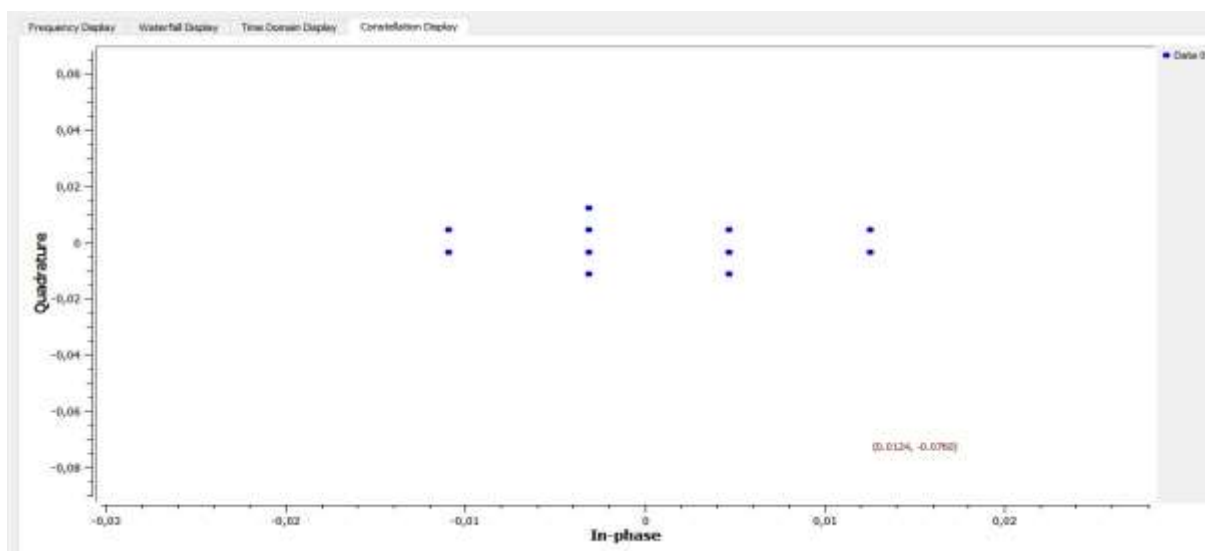
Comportamiento de la señal en función del tiempo



A continuación, en la figura 83 se observa el diagrama de constelación de la señal capturada, donde se observa una modulación 16-QAM, en la imagen se observa como la posición símbolo está dada en referencia a la distancia del punto al origen del plano, lo que define la amplitud de dicho punto y la fase se representa, mediante el ángulo que forma el punto con respecto a un eje de referencia.

Figura 83

Diagrama de Constelación de la Señal



Como este diagrama receptor permite guardar la señal en formato wav, es posible hacer uso de la herramienta Audacity para poder analizar el contenido, en la figura 84 se puede observar el tren de pulsos enviados mediante el dispositivo transmisor al presionar el botón que activa el puerto del equipo receptor.

Figura 84

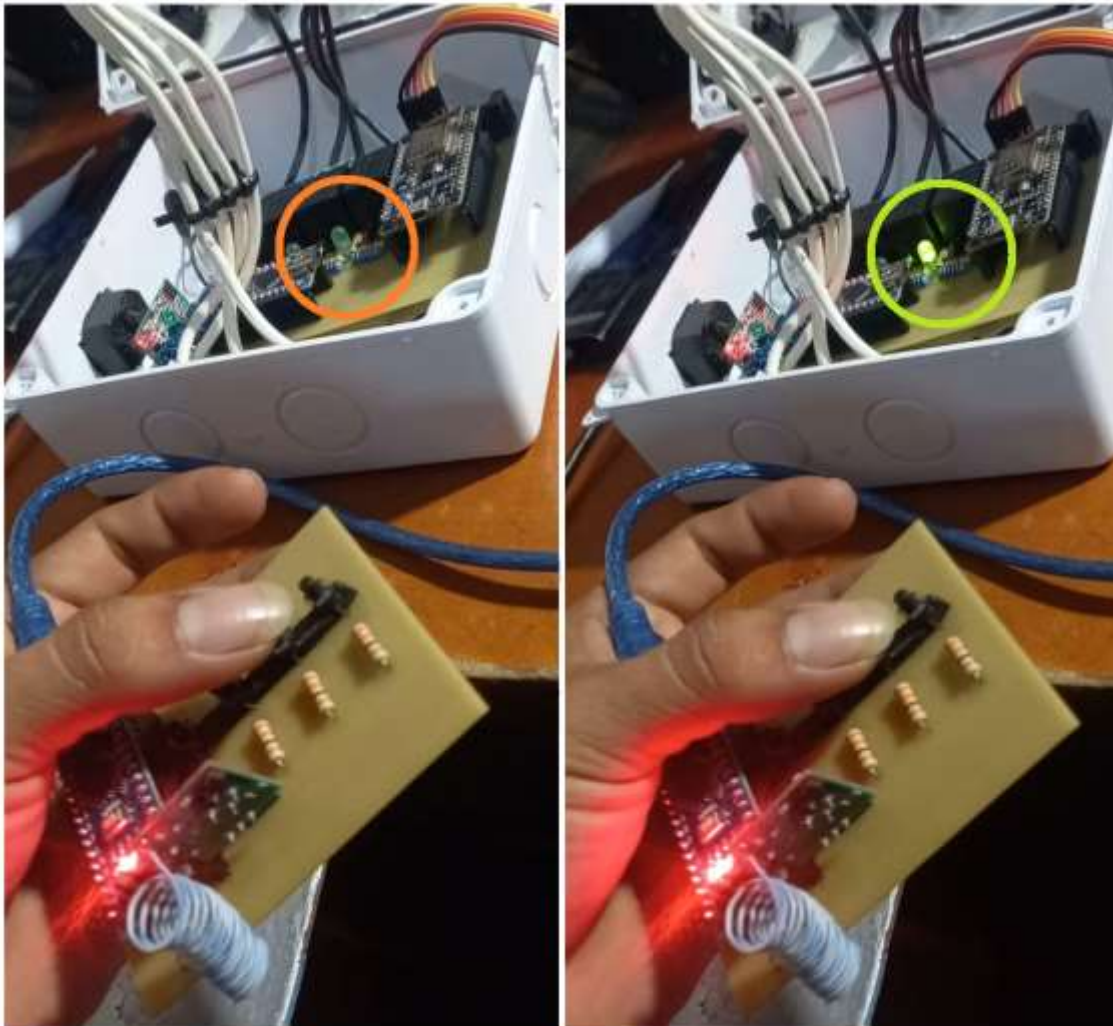
Representación de la señal mediante el tren de pulsos en Audacity



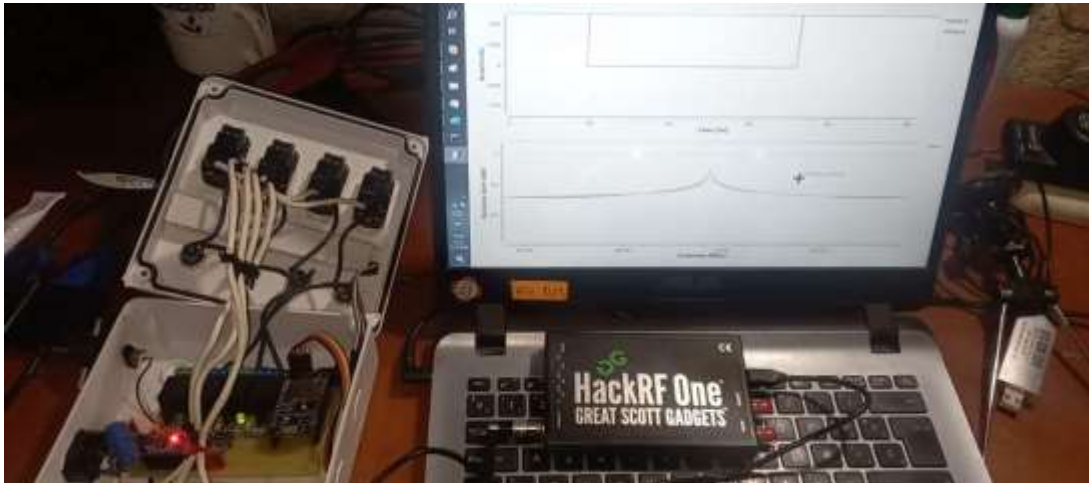
A continuación, se muestra un ejemplo que corresponde al funcionamiento del Prototipo donde se puede evidenciar que, al pulsar el botón en el control de RF, este provocará la activación del puerto en el dispositivo receptor como se muestra en la figura 85.

Figura 85

Activación del puerto de conexión mediante el control RF



Luego de presentar el funcionamiento de este prototipo, el ataque de repetición tiene como objetivo almacenar los datos que surgieron en la comunicación inalámbrica, lo cual se realizó previamente con el flujograma de captura, obteniendo como resultado un archivo que contiene la información que da apertura a los puertos en el receptor, mediante el uso del diagrama de bloques presentado en la figura 47, este se puede manipular el puerto de conexión que está vinculado con los datos grabados de la transmisión. En la figura 86 se puede observar que el puerto de conexión eléctrico se activa con el uso del HackRF One como transmisor, dando resultados exitosos en el ataque de repetición.

Figura 86*Ataque de repetición en la señal RF*

En el escenario siguiente se toma en consideración el inhibidor antes desarrollado y presentado en la figura 50 para evaluar cómo se desempeña el prototipo durante la ejecución de este tipo de ataques, en este caso de estudio se confirmó la pérdida de la conexión a la red Wi-Fi y se logró evidenciar a través de la aplicación un mensaje que indica la pérdida del acceso al control de la aplicación dejando incapacitado al dispositivo, ver figura 87.

Figura 87*Desconexión de la red y la aplicación del control del Prototipo de dispositivo IoT*

Los resultados de esta última práctica fueron favorables ya que el ataque aplicado se ejecutó exitosamente y cumplió con las expectativas de la práctica, en este caso se logró generar un ruido que interrumpe la señal Wi-Fi donde se han conectado los dispositivos finales haciendo imposible la comunicación con el dispositivo y el Smartphone que controla la interfaz.

4.4 Comparación de los resultados

En las pruebas realizadas se busca calcular la efectividad en la comunicación en lo que duró el desarrollo del proyecto, específicamente en la comunicación por radiofrecuencia, en el ataque de repetición y en la inhibición de las señales. Es por esta razón que se emplea la siguiente fórmula en relación con los escenarios de análisis:

$$Efectividad(\%) = \left(\frac{Resultado\ alcanzado}{Resultado\ esperado} \right) * 100$$

Donde:

Resultado alcanzado: Corresponde a la cantidad de una medida que se logró en el proceso.

Resultado esperado: Es la cantidad de una medida que se esperaba alcanzar en el proceso.

Porcentaje de efectividad: se logra multiplicando por 100 la cantidad que resultó en la relación del resultado alcanzado sobre el resultado esperado.

El primer análisis se evalúa con respecto a la efectividad de la comunicación en radiofrecuencias basándose en cuatro registros que fueron indispensables hasta la aplicación final del prototipo y efectuando los cálculos de la efectividad con la relación de los datos recibidos y los datos totales transmitidos por la cantidad de 100 para extraer el porcentaje. De esta manera se inició analizando la comunicación de los módulos de radiofrecuencia con la programación incluida en los ejemplos de la librería Radiohead donde se puede ejecutar

pruebas con el envío de un mensaje de texto emitido por el prototipo transmisor y recibido por el prototipo receptor.

En este caso se optó por realizar una transmisión de 10 datos por medio del circuito transmisor y a través del circuito receptor solo se obtuvieron 8 datos recibidos dando una efectividad del 80% en esa prueba. En el siguiente plano de estudio se consideró armar un arreglo de botones para el lado del transmisor, como primeras pruebas ante el uso de estos pulsadores, se efectuaron 10 ensayos en el equipo transmisor mientras que, en el lado del receptor se obtuvieron 9 datos recibidos dando una efectividad del 90% en esta etapa. En el siguiente caso de análisis se obtuvo el diagrama PCB y se implementaron los elementos electrónicos en la placa PCB para realizar nuevos cálculos con respecto a la radiofrecuencia luego de haber ensamblado todo el prototipo, en este punto se consideraron 10 datos totales transmitidos y el aparato receptor censó 9 cambios, es por este motivo que la efectividad se limita al 90%, finalmente las pruebas realizadas con la aplicación móvil donde se efectuó una muestra de 10 ensayos realizados por medio de la aplicación en Arduino IoT Cloud, se obtuvo la misma cantidad de datos recibidos por el dispositivo receptor y es por esta razón que la efectividad es del 100%, en la tabla 24 se indica de forma resumida lo antes mencionado.

Tabla 24*Análisis de Efectividad en las primeras pruebas de comunicación del Prototipo*

Registro	Pruebas de comunicación	Datos recibidos	Datos totales transmitidos	Efectividad
1	Primeras pruebas de conexión inalámbrica de los módulos	8	10	80,00%
2	Pruebas de la conexión del receptor con el transmisor mediante arreglo de botones	9	10	90,00%
3	Pruebas realizadas con radiofrecuencias luego de la implementación de los componentes en la PCB	9	10	90,00%
4	Pruebas realizadas con la aplicación luego de la implementación de los componentes en la PCB	10	10	100,00%

En este apartado se realiza el análisis de la efectividad en base al ataque de repetición efectuado al prototipo en el área de transmisión y recepción. Para el área de recepción se realizaron dos pruebas la primera consiste en capturar la señal que transmite el prototipo inicial ensamblado en Protoboard, y guardar dicha señal, para esto fue necesario realizar una prueba de 10 intentos de intrusión, obteniendo un logro de 5 intentos comprobando una efectividad del 50%. Para la segunda prueba se aplica el mismo esquema para la ejecución del mismo ataque, pero en esta ocasión se toma como objeto de análisis el prototipo ya ensamblado en la placa PCB, donde al realizar la misma prueba de captura de información se obtiene una cantidad de 9 de 10 intentos exitosos, lo que conlleva a una efectividad del 90 % en esta primera etapa de ataque.

En el caso de la aplicación de la segunda etapa del ataque que corresponde a la transmisión de la información capturada se tiene dos registros de prueba. El primer registro hace referencia a la prueba de retransmisión de la información grabada en la comunicación por radiofrecuencia del prototipo inicial, donde al igual que el primer ejemplo se obtuvo una

efectividad solo del 50%. Mientras que en el segundo registro se realizan 10 intentos exitosos en la transmisión de información obtenida de comunicación del prototipo ensamblado en PCB, por lo que muestra una efectividad del 100% al momento de que el receptor recibe la señal retransmitida por el hackrf One. En la tabla 25 se observa el resumen del análisis descrito anteriormente.

Tabla 25

Análisis de Efectividad en la aplicación del Ataque Replay al Prototipo

Registro	Área	Pruebas en el ataque de repetición	Intentos logrados	Intentos totales realizados	Efectividad
1	Recepción	Primera captura de datos en la comunicación inalámbrica con el prototipo ensamblado en protoboard	5	10	50,00%
2		Prueba de captura en el prototipo final impreso en PCB	9	10	90,00%
3	Transmisión	Prueba de retransmisión de la primera señal grabada con el prototipo ensamblado en protoboard	5	10	50,00%
4		Prueba de la transmisión de la señal grabada en la comunicación del prototipo final impreso en PCB	10	10	100,00%

A continuación, se describe la efectividad que se obtuvo al aplicar el ataque de un inhibidor de señal en una comunicación mediante radiofrecuencia y wifi, para esto se tiene el registro de dos pruebas. La primera prueba corresponde al escenario de comunicación por medio de radiofrecuencia donde se busca introducir una interferencia que evite que el prototipo pueda comunicarse en la frecuencia 433 MHz, donde se efectuaron 10 intentos de inhibición, de los cuales solo 6 fueron exitosos obteniendo una efectividad del 60%.

Para el segundo registro se realizaron pruebas de interferencia en la comunicación inalámbrica mediante Wifi, donde se aplicó el ataque al prototipo y al equipo Sonoff 4Ch Pro, aquí se verificó que en ambos casos la pérdida de señal se realizó de forma inmediata al aplicarse el ataque, en los dos casos se generaron 10 pruebas de los cuales solo 8 fueron intentos válidos, teniendo como resultado una efectividad del 80% para la ejecución de este ataque. En la tabla 26 se detalla de forma resumida lo antes expuesto.

Tabla 26

Análisis de la Efectividad del Ataque de Inhibidor de Señal

Registro	Medio	Pruebas en el ataque del Inhibidor de señal	Área	Intentos logrados	Intentos totales realizados	Efectividad
1	Radiofrecuencia	Introducir interferencias en la comunicación	Prototipo	6	10	60,00%
2	Wi-Fi	Introducir interferencias en la comunicación	Prototipo	8	10	80,00%
			Sonoff 4ch PRO	8	10	80,00%

4.5 Alternativas para mitigar posibles ataques

Durante las pruebas realizadas en el área de pruebas del laboratorio se pensó en hallar alternativas para evitar ataques y así contribuir con la seguridad en la comunicación. Por esta razón, se exponen las siguientes alternativas que podrían ayudar a mitigar los posibles ataques.

4.5.1 Conexión a redes conocidas y confiables

Una de las formas de proteger los dispositivos IoT como el Sonoff 4CH Pro o el dispositivo de distribución eléctrica que se desarrolló es conectar a redes WiFi seguras y conocidas, esto podría contribuir con el acceso controlado gracias a requerimientos de autenticación para permitir el acceso autorizado de los dispositivos, así mismo la protección de

datos sensibles que al transmitirse entre los dispositivos y la red se encripten para evitar que la información pueda ser leída.

4.5.2 Actualizaciones de sistema

Es de vital importancia mantener en actualizado el sistema, para ello se puede utilizar la librería ArduinoOTA.h que permite actualizar el firmware del dispositivo sin necesidad de tener conectado el dispositivo a la computadora y proporciona funcionalidades para que el dispositivo cumpla con tareas específicas.

4.5.3 Conexión a múltiples redes Wifi

Es recomendable registrar más de una red conocida por medio de la librería WiFiMulti.h, que proporciona una alternativa de administrar usuarios y contraseñas de red para que el sistema se pueda conectar y elegir la mejor red de Internet para la conexión del dispositivo de red del proyecto.

4.5.4 Librería Virtualwire para la comunicación inalámbricas en Arduino

Durante las pruebas realizadas en el laboratorio, se confirmó que la librería Radiohead.h es susceptible a sufrir ataques de repetición de señal al capturar la información y almacenarla para utilizarla en otro momento, sin embargo. Se descubrió que por medio de la librería VirtualWire.h no es posible llevar a cabo el ataque de repetición, ya que, al momento de recibir y retransmitir la señal guardada por el atacante, el sistema receptor no responde ante la señal almacenada con anterioridad, así mismo hay que tomar en cuenta que la manera de transmitir un mensaje es a través de la escritura por teclado para redactar el mensaje a enviar. Esto abre las puertas a nuevas alternativas para futuras investigaciones con la finalidad de hallar otras maneras de evitar ataques.

CONCLUSIONES

El desarrollo de este proyecto se destinó para el estudio de las vulnerabilidades de los dispositivos del Internet de las cosas, por esta razón se decidió estudiar el dispositivo Sonoff 4CH Pro que es muy utilizado en la industria, la medicina y en varias áreas donde se requieren automatizar procesos gracias a la facilidad que este presenta al momento de ser configurado, siendo un dispositivo que puede ser programado con tareas específicas en horarios establecidos por el usuario.

Se realizó un análisis del hardware y el software del Sonoff 4CH Pro y se estudió su composición en general para el desarrollo de un dispositivo con funcionalidades similares, pero a menor costo, para ello se hizo una revisión literaria expuesta en el capítulo 2 donde se pudo definir las mejores opciones de los componentes electrónicos en comparación a los que se emplearon en el modelo comercial mediante el estudio de las características, capacidades y propiedades de los elementos electrónicos.

Fue necesario el desarrollo de una lógica de programación que permita las comunicaciones inalámbricas en frecuencias de 433 MHz y 2.4 GHz con el uso de las tarjetas programables y los componentes transmisores/receptores respectivamente, además se realizó una aplicación móvil para interactuar directamente con el dispositivo conectado a la red.

El trabajo conjunto de GNU Radio y los equipos SDR, permitió crear una lógica de bloques para llevar a cabo un ataque de reproducción y construir un inhibidor de señal, lo cual proporciono una herramienta valiosa para evaluar y comprender las vulnerabilidades presentes, en sistemas de comunicación inalámbricos de dispositivos IoT. Esta técnica permitió evaluar y comprender de manera controlada las debilidades presentes en la seguridad de estos dispositivos. Al simular ataques, se pueden identificar vulnerabilidades y desarrollar estrategias de mitigación para fortalecer la seguridad de los sistemas IoT. Así, esta metodología contribuye

a mejorar la resistencia y confiabilidad de los dispositivos inalámbricos, garantizando una mayor protección para los usuarios y datos en un mundo cada vez más conectado.

RECOMENDACIONES

El presente estudio se desarrolló para evaluar las condiciones de las comunicaciones inalámbricas que trabajan en el rango de 433 MHz y 2,4 GHz, para ello es recomendable usar las librerías adecuadas que permitan la protección de las comunicaciones inalámbricas en el software de desarrollo de aplicaciones IoT Cloud y Arduino, aprovechando su compatibilidad en ambos escenarios de programación se recomienda hacer uso de la librería "ESP8266WIFI" que proporciona varias funcionalidades para conectarse a la red de Internet.

La implementación de protocolos de seguridad robustos y buenas prácticas de programación es esencial para proteger los datos y garantizar el correcto funcionamiento del dispositivo final, por otra parte, se sugiere implementar componentes electrónicos de buena calidad para prolongar la vida útil y evitar costos de mantenimiento, se enfatiza la importancia de la mejora continua y la adaptabilidad para mantenerse relevante en un campo en constante evolución.

Las practicas realizadas con los equipos SDR se llevaron a cabo en el laboratorio de telecomunicaciones siendo un ambiente controlado y autorizado para esta actividad en especial, no se recomienda ejecutar estas prácticas fuera de estos escenarios controlados ya que es posible afectar las comunicaciones inalámbricas en el rango de cobertura de los equipos, además se busca mejorar la comprensión de la seguridad en comunicaciones inalámbricas, asegurándose de divulgar los hallazgos de manera responsable. Las recomendaciones prácticas resultantes pueden aplicarse en la industria para fortalecer la seguridad de los sistemas inalámbricos.

BIBLIOGRAFÍA

- 3DBOTS. (s.f.). *Microcontrolador PIC18F452 Microcontrolador Dip40 Microchip*. Obtenido de 3DBOTS: <https://3dbots.co/producto/microcontrolador-pic18f452-microcontrolador-dip40-microchip/>
- Amazon. (s.f.). *Smart Power Strip, WiFi Surge Protector Work with Alexa Google Home, Smart Plug Outlets with 3 USB 3 Charging Port, Home Office Cruise Ship Travel Multi-Plug Extender, 10A, Black*. Obtenido de Amazon: https://www.amazon.com.mx/Smart-Protector-Charging-Multi-Plug-Extender/dp/B09JZ398R8/ref=pd_lpo_3?pd_rd_w=xaC1J&content-id=amzn1.sym.5ca78996-70c7-4a7b-b60c-f030ccc1aa2f&pf_rd_p=5ca78996-70c7-4a7b-b60c-f030ccc1aa2f&pf_rd_r=0C3P30FSRQ0C12KQJNYA&pd_rd_wg=
- Amazon. (s.f.). *uhppote – AC/DC12 – 48 V 2-Ch 433 mhz RF control remoto interruptor de relé transceptor 2-buttons*. Obtenido de Amazon: <https://www.amazon.com/UHPPOTE-DC12-48V-Control-Transceiver-2-Buttons/dp/B07CZB2G6Y>
- (junio de 2021). *ARDUINO UNO R3*.
- Baró, A. V. (2002). Inicios de la construcción de la primera red. En *La prehistoria de la red* (págs. 7-9). Barcelona: Ediciones Península. Recuperado el 23 de mayo de 2022, de <https://www.tdx.cat/bitstream/handle/10803/9156/Tavb02de23.pdf?sequence=3>
- Baró, A. V. (2002). Motivaciones originales de ARPANET e Internet. En *La prehistoria de la red* (págs. 17-18). Barcelona, Barcelona, España: Ediciones Península. Recuperado el 23 de mayo de 2022, de <https://www.tdx.cat/bitstream/handle/10803/9156/Tavb02de23.pdf?sequence=3>
- Barrio, A. M. (2020). *Internet de las Cosas*. Madrid, Madrid, España: REUS. Obtenido de <https://books.google.com.ec/books?id=0BE5EAAAQBAJ&pg=PA127&dq=SEGURIDAD+EN+SISTEMAS+DEL+INTERNET+DE+LAS+COSAS&hl=es&sa=X&ved=2ahUKEwjNhIGb4dn5AhX-RjABHTfsBW8Q6AF6BAgLEAI#v=onepage&q&f=true>
- Barrio, A. M. (2020). LA SEGURIDAD EN EL INTERNET DE LAS COSAS. En *Internet de las cosas* (págs. 22-25). Madrid, España: REUS. Obtenido de <https://books.google.com.ec/books?id=0BE5EAAAQBAJ&pg=PA127&dq=SEGURIDAD+EN+SISTEMAS+DEL+INTERNET+DE+LAS+COSAS&hl=es&sa=X&ved=2ahUKEwjNhIGb4dn5AhX-RjABHTfsBW8Q6AF6BAgLEAI#v=onepage&q&f=true>
- Couch, L. W. (2015). Modulación ASK. En *Sistema De Comunicación Digitales Y Analógicos* (pág. 369). Mexico: Pearson Education.
- Couch, L. W. (2015). Modulación FSK. En *Sistema de comunicación digitales y analógicos* (págs. 375-382). Mexico: Pearson Education .
- Couch, L. W. (2015). Modulación PSK. En *Sistema de comunicación digitales y analógicos* (págs. 3-6). Mexico: Pearson Education .
- Couch, L. W. (2015). *Sistema De Comunicación Digitales Y Analógicos*. Mexico: Pearson Education.
- Ebay. (s.f.). *Receptor estéreo de paquete NooElec NESDR Smart V4 -100700 [Fotografía]*. Obtenido de Ebay: <https://www.ebay.co.uk/p/2254506191>
- Element14. (s.f.). *RPI4-MODBP-8GB*. Obtenido de Element14: <https://sg.element14.com/raspberry-pi/rpi4-modbp-8gb/raspberry-pi-4-model-b-cortex/dp/3369503>

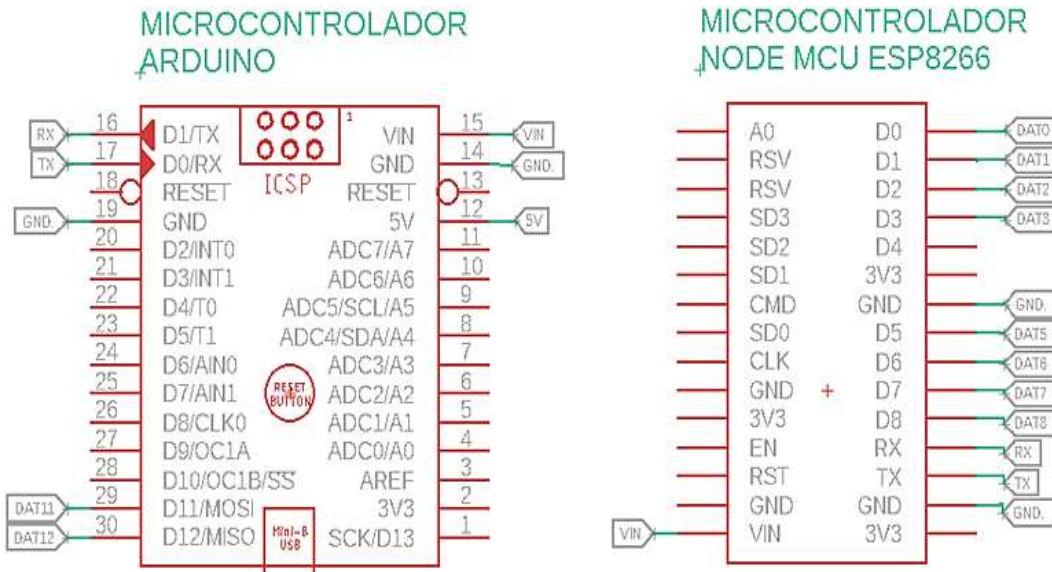
- E-SWITCH. (2018). *DATASHEET E-SWITCH TL1100F260Q*. Obtenido de <https://pdf1.alldatasheet.es/datasheet-pdf/view/639957/E-SWITCH/P001111.html>
- Gámez, A. C. (2020). *Estudio De La Seguridad Del Protocolo Tpms Vulnerabilities In The Tpms Protocol*. Madrid.
- Interaction Design Lab. (2013). *Descripción del Software Fritzing*. Berlín. Obtenido de <https://fritzing.org/>
- Ja-Bots.com. (s.f.). *Microcontrolador PIC16F877A*. Obtenido de MICROSIDE: <https://ja-bots.com/producto/microcontrolador>
- KONDSOON. (2014). *Breadboard*. Claudio Peña.
- LivingInternet. (30 de mayo de 2022). *The Internet Toaster*. Obtenido de https://www.livinginternet.com/i/ia_myths_toast.htm
- MICROSIDE. (s.f.). *Microcontrolador PIC16F84A*. Obtenido de MICROSIDE: <https://store.microside.com/productos/pic16f84a-20-p-8-bits-microchip/>
- Mora, S. L. (2001). Clientes WEB. En S. L. Mora, *Programación en Internet* (págs. 27-28). España: Editorial Clib Universitario.
- NANO, A. (2021).
- Navarro, K., Canto, F., & Poveda, H. (2018). *La Radio Definida por Software como Herramienta de Aprendizaje Educativa en Comunicaciones Inalámbricas*. Lima. Obtenido de http://www.laccei.org/LACCEI2018-Lima/full_papers/FP115.pdf
- OWASP. (2018). *Internet of things TOP 10*. Obtenido de <https://owasp.org/www-pdf-archive/OWASP-IoT-Top-10-2018-final.pdf>
- PAZMIÑO, S. J. (2015). *IMPLEMENTACION DE SISTEMAS RECEPTORES DE AM, FM Y ADB-S USANDO SDR (SOFTWARE DEFINED RADIO), HARDWARE Y SOFTWARE*. QUITO.
- Peña, C. (2013). *Arduino IDE*. Buenos Aires, Argentina: Claudio Peña . Obtenido de https://books.google.es/books?hl=es&lr=&id=Xgv2DwAAQBAJ&oi=fnd&pg=PP1&dq=arduino+ide&ots=vNDTDcTx5Z&sig=vIHS-5_dFJ3HOM_BbKTPW__H0ew#v=onepage&q=arduino%20ide&f=false
- Peréz, H. D. (12 de Febrero de 2021). El Internet de las cosas y la economía de los resultados. *La República*. Recuperado el 4 de Junio de 2022, de <https://www.larepublica.co/analisis/hernan-david-perez-3094442/el-internet-de-las-cosas-y-la-economia-de-los-resultados-3124099>
- Raspberry Pi. (s.f.). *Raspberry Pi 2 Modelo B*. Obtenido de Raspberry Pi: <https://www.raspberrypi.com/products/raspberry-pi-2-model-b/>
- Raspberry Pi. (s.f.). *Raspberry Pi 3 Modelo B+*. Obtenido de Raspberry Pi: <https://www.raspberrypi.com/products/raspberry-pi-3-model-b-plus/>
- Real Academia Española. (2014). Madrid: Aduana vieja.
- Rossano, V. (2013). *PROTEUS VSM*. Buenos Aires, Argentina: FOX ANDINA. Obtenido de https://books.google.es/books?hl=es&lr=&id=b1mivOB6_YMC&oi=fnd&pg=PA4&dq=isis+proteus&ots=PuODQjLec4&sig=C3MySX38Za5tq7ckHFVeFAInIPI#v=onepage&q=isis%20proteus&f=false

- Sanabria, J. S., & Montoya, A. F. (2020). *ESQUEMA DE SEGURIDAD DE DATOS ENTRE LOS NODOS Y EL GATEWAY EN UNA RED LoRaWAN*. Bogotá.
- Santos, P. R. (22 de Septiembre de 2020). *Breve historia del Internet de las cosas*. (Telefonía Tech) Recuperado el 2 de Junio de 2022, de <https://empresas.blogthinkbig.com/breve-historia-de-internet-de-las-cosas-iot/>
- Sertronics. (2014). 433. Claudio Peña. Obtenido de https://cdn-reichert.de/documents/datenblatt/FS1000A_DB_DE.pdf
- UNIT ELECTRONICS. (s.f.). *Módulos Inalámbricos STX882/SRX882*. Obtenido de <https://uelectronics.com/producto/modulos-inalambricos-stx882-srx882-433mhz-rx-tx/>
- Universidad Complutense Madrid. (2013). *Descripción del Software Wireshark*. Madrid. Obtenido de <https://www.ucm.es/pimcd2014-free-software/wireshark>
- Vasan, S. (2019). *Truck Connectivity Platform Using Software Defined Radios*. Stockholm.
- Vistrónica. (s.f.). *MÓDULO RELÉ SRA RF PROGRAMABLE DE 4 CANALES 50 METROS*. Obtenido de Vistrónica: <https://www.vistronica.com/comunicaciones/radiofrecuencia/modulo-rele-sra-rf-programable-de-4-canales-50-metros-sutagao-detail.html>
- Ziegler, J. L., Arn, R. T., & Chambers, W. (2017). Modulation recognition with GNU radio, keras, and HackRF. *IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)*. Baltimore, MD, USA. doi:10.1109/DySPAN.2017.7920747.

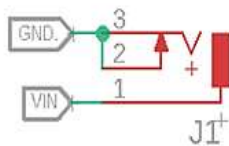
ANEXOS

Anexo A

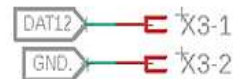
Diagrama de conexiones del Receptor



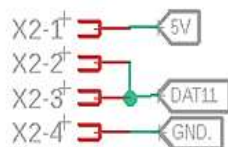
ALIMENTACIÓN JACK DC



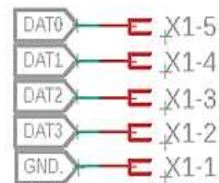
INDICADOR BUZZER



RX-MÓDULO RF 433MHZ



INDICADORES LED



ARREGLO DE RELES

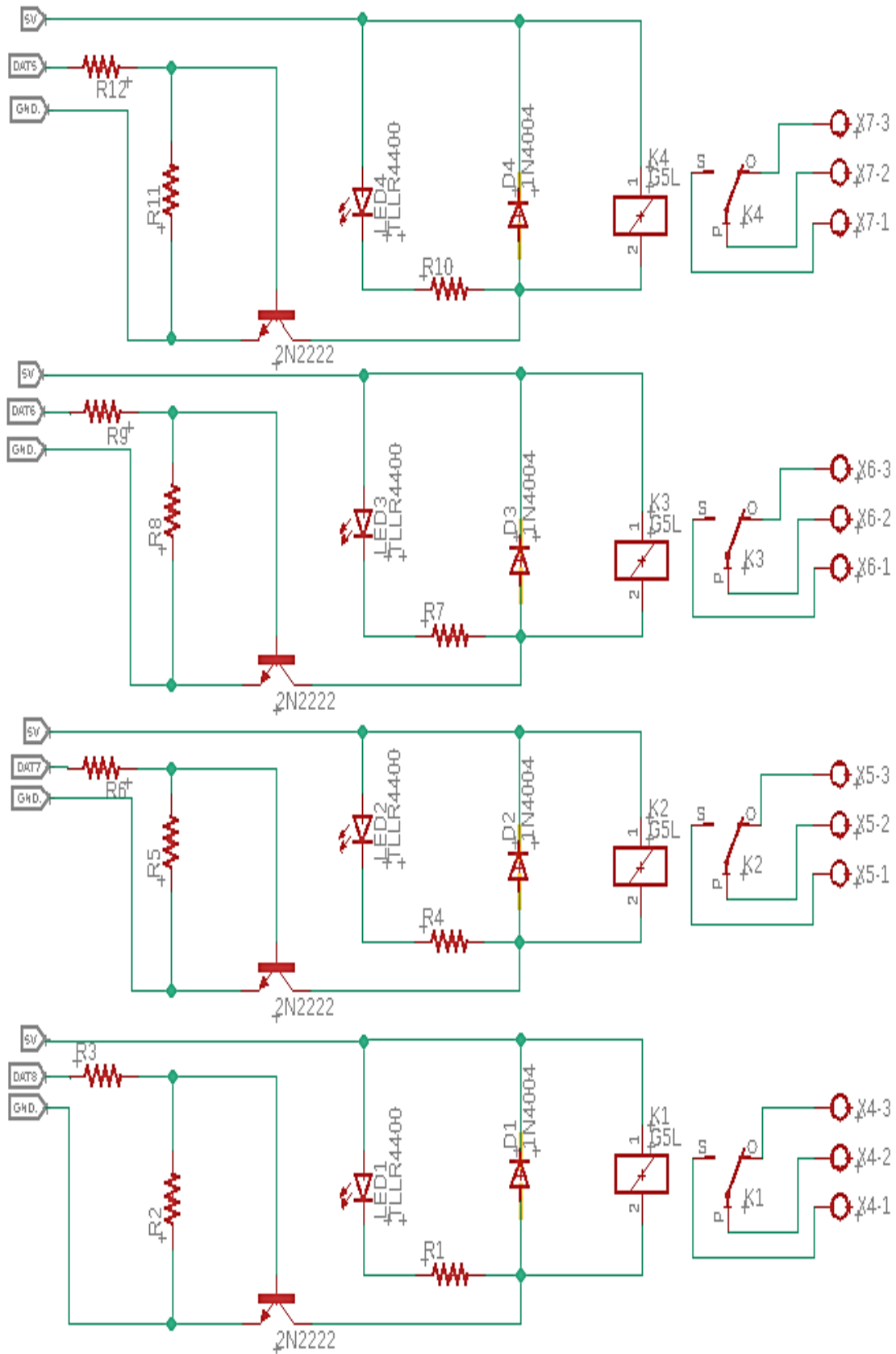
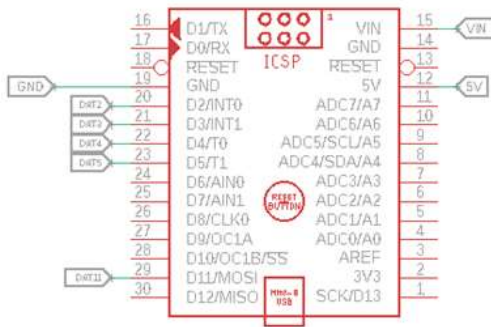


Diagrama de conexiones del transmisor

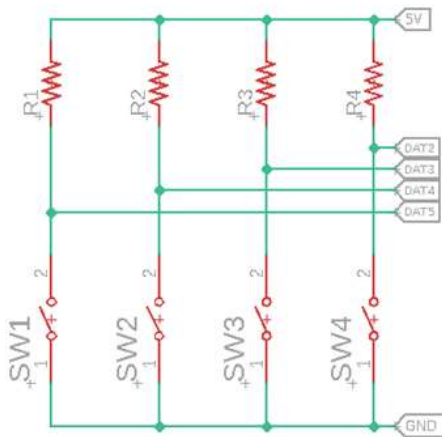
MICROCONTROLADOR



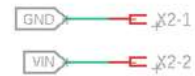
TX-MÓDULO RF 433MHZ



CONTROL DE BOTONES



ALIMENTACIÓN DEL MICROCONTROLADOR



Anexo B

Programación receptor RF a 433.92 MHz

```

3rhrx
1 #include <RH_ASK.h>
2 #include <SPI.h>
3 //RH_ASK driver (2000,rx,tx,5,false);
4   RH_ASK driver (2000,11,10,5,false);
5 uint8_t buf[10];
6 uint8_t buflen = 10;
7
8 void setup() {
9   pinMode(13, OUTPUT);
10  Serial.begin(9600); // Debugging only
11  if (!driver.init())
12    Serial.println("init failed");
13  Serial.println("rx ok");
14 }
15
16 void loop() {
17
18   if (driver.recv(buf, &buflen)) {
19     char dato = 0;
20
21
22     if (buf[0] == 1) {
23       digitalWrite(13, HIGH);Serial.println("on p1");
24       char dato = 'a';Serial.println(dato);Serial.println(); }
25     Serial.write(dato);
26     if (buf[0] == 2) {
27       digitalWrite(13, LOW);Serial.println("off P1");
28       char dato = 'b';Serial.println(dato);Serial.println(); }
29     Serial.write(dato);
30     if (buf[0] == 3) {
31       digitalWrite(13, HIGH);Serial.println("on p2");
32       dato = 'c';Serial.println(dato); }
33
34     if (buf[0] == 4) {
35       digitalWrite(13, LOW);Serial.println("off P2");
36       dato = 'd'; Serial.println(dato); }
37
38     if (buf[0] == 5) {
39       digitalWrite(13, HIGH);Serial.println("on p3");
40       dato = 'e';Serial.println(dato); }
41
42     if (buf[0] == 6) {
43       digitalWrite(13, LOW);Serial.println("off P3");
44       dato = 'f';Serial.println(dato); }
45
46     if (buf[0] == 7) {
47       digitalWrite(13, HIGH);Serial.println("on p4");
48       dato = 'g'; Serial.println(dato); }
49
50     if (buf[0] == 8) {
51       digitalWrite(13, LOW);Serial.println("off P4");
52       dato = 'h'; Serial.println(dato); }
53   }
54 }

```


Programación transmisor RF a 433.92 MHz

```

1  #include <RH_ASK.h>
2  #include <SPI.h>
3
4  ///////////////////////////////////////////////////////////////////
5  //RH_ASK mod (2000,rx,tx,5,false);
6  RH_ASK mod (2000,11,10,5,false);
7  uint8_t buff[10];
8
9  // Pin de botones
10 const int buttonPin1 = 2;const int buttonPin2 = 3;          //
11 const int buttonPin3 = 4;const int buttonPin4 = 5;          //
12 //_____//
13
14 // Pin del LED
15 const int ledPin = 13; bool ledState = false;              //
16 //_____//
17
18 // Estado inicial de botones
19 bool button1Pressed = false;bool button2Pressed = false;   //
20 bool button3Pressed = false;bool button4Pressed = false;   //
21 //_____//
22
23 // Almacenar el tiempo que se pulsa un boton
24 unsigned long button1PressStartTime = 0;                    //
25 unsigned long button2PressStartTime = 0;                    //
26 unsigned long button3PressStartTime = 0;                    //
27 unsigned long button4PressStartTime = 0;                    //
28 //_____//
29
30 void setup() {
31   Serial.println(9600);
32   pinMode(13, OUTPUT);
33   if (!mod.init()) {
34     Serial.println("Error de modulo");
35     while(1);
36   }
37   Serial.println("433-TX");
38   buff[0] = 0;
39
40 // Configuración de botones como entrada con resistencia pull-up //
41 pinMode(buttonPin1, INPUT_PULLUP);pinMode(buttonPin2, INPUT_PULLUP); //
42 pinMode(buttonPin3, INPUT_PULLUP);pinMode(buttonPin4, INPUT_PULLUP); //

```

```

43 // _____ //
44 }
45
46 void loop() {
47
48 // Leer el estado de cada botón _____ //
49   bool button1State = digitalRead(buttonPin1); //
50   bool button2State = digitalRead(buttonPin2); //
51   bool button3State = digitalRead(buttonPin3); //
52   bool button4State = digitalRead(buttonPin4); //
53 // _____ //
54
55
56 // Cuando se presiona el boton 1 Y si el estado inicial del boton 1_//
57   if (button1State == LOW && !button1Pressed) { //
58 // Boton 1 presionado y Registro del tiempo que se presionó //
59     button1Pressed = true; button1PressStartTime = millis();} //
60 // //
61 // Cuando se suelta el boton 1 Y si fue registrado su estado //
62   if (button1State == HIGH && button1Pressed) { //
63 // Indicar que se deajo de pulsar boton 1 //
64     button1Pressed = false; //
65 // Calculo del tiempo que se presionó el boton //
66     unsigned long button1PressDuration=millis()-button1PressStartTime;//
67 // Comparacion del tiempo //
68 // Si se presiona el boton 1 y es menor a 4s //
69   if (button1PressDuration <= 4000) { //
70     ledState = true;digitalWrite(ledPin, HIGH); //
71     buff[0] = 1;digitalWrite(13, HIGH); //
72     mod.send((uint8_t *)buff, 10); mod.waitPacketSent(); //
73     // Enviar y presentar 'a' por comunicación serial //
74     Serial.write('a'); Serial.println("a");} //
75 // Si se presiona el boton 1 y es mayor a 4s //
76   else { //
77     ledState = false; digitalWrite(ledPin, LOW); //
78     buff[0] = 2; digitalWrite(13, LOW); //
79     mod.send((uint8_t *)buff, 10); mod.waitPacketSent(); //
80     // Enviar y presentar 'b' por comunicación serial //
81     Serial.write('b');Serial.println("b");}} //
82 // _____ //

```

```

85 // Cuando se presiona el boton 2 Y si el estado inicial del boton 2_//
86 if (button2State == LOW && !button2Pressed) { //
87 // Boton 2 presionado y Registro del tiempo que se presionó //
88     button2Pressed = true;button2PressStartTime = millis();} //
89 // //
90 // Cuando se suelta el boton 2 Y si fue registrado su estado //
91 if (button2State == HIGH && button2Pressed) { //
92 // Indicar que se dejo de pulsar boton 2 //
93     button2Pressed = false; //
94 // Calculo del tiempo que se presionó el boton //
95     unsigned long button2PressDuration=millis()-button2PressStartTime; //
96 // Comparacion del tiempo //
97 // Si se presiona el boton 2 es menor de 4s //
98 if (button2PressDuration <= 4000) { //
99     ledState = true;digitalWrite(ledPin, HIGH); //
100     buff[0] = 3; digitalWrite(13, HIGH); //
101     mod.send((uint8_t *)buff, 10);mod.waitPacketSent(); //
102     // Enviar y presentar 'c' por comunicación serial //
103     Serial.write('c'); Serial.println("c");} //
104 // Si se presiona el boton 2 es mayor de 4s //
105 else { //
106     ledState = false;digitalWrite(ledPin, LOW); //
107     buff[0] = 4;digitalWrite(13, LOW); //
108     mod.send((uint8_t *)buff, 10);mod.waitPacketSent(); //
109     // Enviar y presentar 'd' por comunicación serial //
110     Serial.write('d');Serial.println("d");}} //
111 // _____ //
112
113
114
115 // Cuando se presiona el boton 3 Y si el estado inicial del boton 3_//
116 if (button3State == LOW && !button3Pressed) { //
117 // Boton 3 presionado y Registro del tiempo que se presionó //
118     button3Pressed = true;button3PressStartTime = millis();} //
119 // //
120 // Cuando se suelta el boton 3 Y si fue registrado su estado //
121 if (button3State == HIGH && button3Pressed) { //
122 // Indicar que se dejo de pulsar boton 3 //
123     button3Pressed = false; //
124 // Calculo del tiempo que se presionó el boton //
125     unsigned long button3PressDuration=millis()-button3PressStartTime;//
126 // Comparacion del tiempo //

```

```

144 // Cuando se presiona el boton 4 Y si el estado inicial del boton 4_//
145▣ if (button4State == LOW && !button4Pressed) { //
146 // Boton 1 presionado y Registro del tiempo que se presionó //
147     button4Pressed = true;button4PressStartTime = millis();} //
148 // //
149 // Cuando se suelta el boton 1 Y si fue registrado su estado //
150▣ if (button4State == HIGH && button4Pressed) { //
151 // Indicar que se deajo de pulsar boton 1 //
152     button4Pressed = false; //
153 // //
154 // Calculo del tiempo que se presionó el boton //
155     unsigned long button4PressDuration=millis()-button4PressStartTime;//
156 // Comparacion del tiempo //
157 // Si se presiona el boton 1 y es menor a 4s //
158▣ if (button4PressDuration <= 4000) { //
159     ledState = true;digitalWrite(ledPin, HIGH); //
160     buff[0] = 7;digitalWrite(13, HIGH); //
161     mod.send((uint8_t *)buff, 10);mod.waitPacketSent(); //
162     // Enviar y presentar 'a' por comunicación serial //
163     Serial.write('g');Serial.println("g");} //
164 // Si se presiona el boton 1 y es mayor a 4s //
165▣ else { //
166     ledState = false;digitalWrite(ledPin, LOW); //
167     buff[0] = 8;digitalWrite(13, LOW); //
168     mod.send((uint8_t *)buff, 10);mod.waitPacketSent(); //
169     // Enviar y presentar 'b' por comunicación serial //
170     Serial.write('h');Serial.println("h");} //
171 // _____ //
172 //
173 // Actualizar el estado del LED _____ //
174     digitalWrite(ledPin, ledState); //
175     delay(500);} //
176 // _____ //


```

Anexo C

Vinculación del dispositivo ESP8266 con la plataforma Arduino IoT Cloud


Associate device ×

Choose the device to associate to
TABLERO DE DISTRIBUCION ELECTRICA



NO DEVICE FOUND

SET UP NEW DEVICE 1



2

Third party device ⓘ
Arduino language (C++)

Select device type

Please select the device type and model you want to configure

ESP8266 3 ESP32 LoRaWAN

NodeMCU 1.0 (ESP-12E Module) ▼

CONTINUE 4

Give your device a name

Name your device so you will be able to recognize it.

Device Name
ESP8266RX ↻

NEXT 5

Anexo D

Programación del receptor a 2.4 GHz para la aplicación

```

Tablero_de_distribuci  ReadMe.adoc  thingProperties.h  Secret  ▼
1  #include <SoftwareSerial.h>
2  #include "thingProperties.h"
3
4  SoftwareSerial arduinoSerial(3, 1);
5  void setup() {
6      Serial.begin(9600);
7      arduinoSerial.begin(9600);
8      delay(1500);
9      initProperties();
10
11     // Connect to Arduino IoT Cloud
12     ArduinoCloud.begin(ArduinoIoTPreferredConnection);
13     setDebugMessageLevel(2);
14     ArduinoCloud.printDebugInfo();
15
16     pinMode(16,OUTPUT);/* p1 pin D0 port a*/
17     pinMode(4,OUTPUT);/* p2 pin D2 port b*/
18     pinMode(0,OUTPUT);/* p3 pin D3 port c*/
19     pinMode(2,OUTPUT);/* p4 pin D4 port d*/
20
21     pinMode(14,OUTPUT);/* p1 pin D5 port a*/
22     pinMode(12,OUTPUT);/* p2 pin D6 port b*/
23     pinMode(3,OUTPUT);/* p4 pin D7 port c*/
24     pinMode(1,OUTPUT);/* p3 pin D8 port d*/
25 }
26
27 void loop() {
28     ArduinoCloud.update();
29     // Your code here
30     ..... /*Activar puerto 4*/
31     /*     if (puerto4.isActive()){
32     .....     serialx.write ("7");
33     .....     digitalWrite(2,HIGH);
34     .....     digitalWrite(1,HIGH);
35     .....     indicador4 = true;
36
37     ..... }else{
38     .....     digitalWrite(2,LOW);
39     .....     digitalWrite(1,LOW);
40     .....     serialx.write ("8");
41     .....     indicador4 = false;}*/
42 }
43
44
45 void onPuerto1Change() {
46     // Add your code here to act upon Puerto1 change
47     .....     if(puerto1==true){
48     .....     digitalWrite(16,HIGH);
49     .....     digitalWrite(14,HIGH);
50
51     .....
52     ..... }else{
53     .....     digitalWrite(16,LOW);
54     .....     digitalWrite(14,LOW);

```

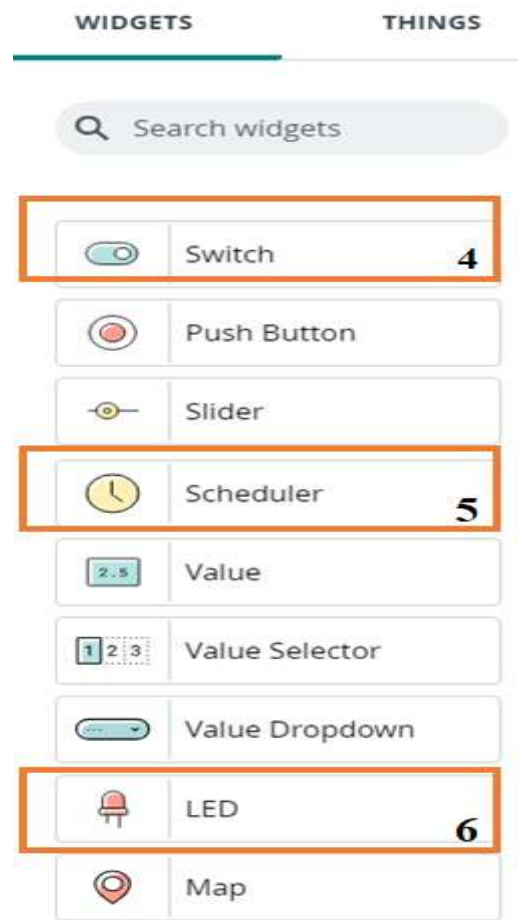
```

55 }
56
57 /*
58  Since Puerto2 is READ_WRITE variable, onPuerto2Change() is
59  executed every time a new value is received from IoT Cloud.
60  */
61 void onPuerto2Change() {
62  // Add your code here to act upon Puerto2 change
63  if(puerto2==true){
64
65      digitalWrite(04,HIGH);
66      digitalWrite(12,HIGH);
67  }else{
68      digitalWrite(04,LOW);
69      digitalWrite(12,LOW);}
70 }
71
72 /*
73  Since Puerto3 is READ_WRITE variable, onPuerto3Change() is
74  executed every time a new value is received from IoT Cloud.
75  */
76 void onPuerto3Change() {
77  // Add your code here to act upon Puerto3 change
78  if(puerto3==true){
79      digitalWrite(0,HIGH);
80      digitalWrite(3,HIGH);
81
82  }else{
83      digitalWrite(0,LOW);
84      digitalWrite(3,LOW);}
85 }
86
87 /*
88  Since Puerto4 is READ_WRITE variable, onPuerto4Change() is
89  executed every time a new value is received from IoT Cloud.
90  */
91 void onPuerto4Change() {
92  // Add your code here to act upon Puerto4 change
93  /*Activar puerto 4*/
94  if (puerto4.isActive()){
95      digitalWrite(2,HIGH);
96      digitalWrite(1,HIGH);
97      indicador4 = true;
98
99  }else{
100     digitalWrite(2,LOW);
101     digitalWrite(1,LOW);
102     indicador4 = false;}
103 }
104

```

Anexo E

Ubicación de los Widgets



Anexo F*Parte interna del prototipo de dispositivo IoT*

Parte interna del prototipo transmisor



Anexo G

Comprobación del paso de la energía por el puerto de conexión eléctrico

