

# **ESCUELA POLITÉCNICA NACIONAL**

## **FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA**

**IMPLEMENTACIÓN DE UN SISTEMA DE RECEPCIÓN DE IMÁGENES  
METEOROLÓGICAS UTILIZANDO TECNOLOGÍA DE RADIO  
DEFINIDO POR SOFTWARE (RDS)**

**PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN  
ELECTRÓNICA Y TELECOMUNICACIONES**

**CÉSAR AUGUSTO VELASCO ARIAS**  
**[cesar\\_pm1611@hotmail.es](mailto:cesar_pm1611@hotmail.es)**

**DIRECTOR: ING. CHRISTIAN TIPANTUÑA, MSc.**  
**[christian.tipantuna@epn.edu.ec](mailto:christian.tipantuna@epn.edu.ec)**

**CODIRECTOR: ING. JULIO CÉSAR CAIZA, MSc.**  
**[cesar.caiza@epn.edu.ec](mailto:cesar.caiza@epn.edu.ec)**

**Quito, diciembre 2016**

## DECLARACIÓN

Yo, César Augusto Velasco Arias, declaro bajo juramento que el trabajo aquí descrito es de mí autoría; que no ha sido previamente presentada para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedo mis derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la ley de Propiedad Intelectual, por su Reglamento y por la normativa institucional vigente.

---

**César Augusto Velasco Arias**

## CERTIFICACIÓN

Certificamos que el presente trabajo fue desarrollado por César Augusto Velasco Arias, bajo nuestra supervisión.

---

Ing. Christian Tipantuña, MSc.  
Director del Proyecto

---

Ing. Julio César Caiza, MSc.  
Co-Director del Proyecto

## AGRADECIMIENTO

Agradezco a mi padre César, a mi madre Ivonne y a mis hermanos Christopher y Andrés por estar junto a mí, ser un apoyo y brindarme siempre su cariño.

Agradezco a mis amigos del colegio y de la universidad, por ayudarme y apoyarme a culminar una nueva etapa de mi vida y por todos aquellos buenos y malos momentos que pasamos.

Agradezco a mi director Ingeniero Christian Tipantuña por su paciencia, motivación y criterio en la realización de este proyecto.

## **DEDICATORIA**

Mi tesis la dedico a mis hermanos Christopher y Andrés.

## CONTENIDO

DECLARACIÓN .....	i
CERTIFICACIÓN .....	ii
AGRADECIMIENTO.....	iii
DEDICATORIA.....	iv
CONTENIDO.....	v
RESUMEN .....	xv
PRESENTACIÓN .....	xvi
1 CAPÍTULO 1: FUNDAMENTOS TEÓRICOS .....	1
1.1 TELEDETECCIÓN .....	1
1.1.1 ESPECTRO ELECTROMAGNÉTICO EN TELEDETECCIÓN .....	2
1.1.2 INTERACCIÓN DE LA ENERGÍA CON LA ATMÓSFERA.....	4
1.1.2.1 Absorción .....	4
1.1.2.2 Dispersión .....	5
1.1.3 INTERACCIÓN DE LA ENERGÍA CON LA SUPERFICIE DE LA TIERRA	
.....	6
1.2 SISTEMAS SATELITALES .....	6
1.2.1 CARACTERÍSTICAS DE LOS SISTEMAS SATELITALES .....	6
1.2.2 FUNCIONAMIENTO BÁSICO DE UN SATÉLITE .....	8
1.2.2.1 Carga útil o de comunicaciones .....	8
1.2.2.2 Plataforma .....	8
1.2.2.2.1 Subsistema de estructura .....	9
1.2.2.2.2 Subsistema de propulsión.....	9
1.2.2.2.3 Subsistema de control de orientación .....	9
1.2.2.2.4 Subsistema de energía .....	9
1.2.2.2.5 Subsistema de telemetría y telemando .....	10
1.2.2.2.6 Subsistema de control térmico.....	10
1.2.3 ENLACES SATELITALES .....	10

1.2.4 BANDAS DE FRECUENCIAS SATELITALES.....	11
1.2.5 TIPOS DE ÓRBITAS SATELITALES .....	13
1.2.5.1 Órbita baja (LEO, Low Earth Orbit) .....	14
1.2.5.2 Órbita media (MEO, Medium Earth Orbit) .....	14
1.2.5.3 Órbita alta (GEO, Geostationary Orbit).....	14
1.2.5.4 Órbitas de alta excentricidad (HEO, High Earth Orbit) .....	15
1.2.5.5 Órbita polar.....	15
1.2.6 TIPOS DE SATÉLITES POR SU APLICACIÓN .....	15
1.3 SENsoRES SATELITALES .....	16
1.3.1 TIPOS DE SENsoRES .....	17
1.3.1.1 Sensores activos .....	17
1.3.1.2 Sensores pasivos .....	17
1.3.2 RESOLUCIÓN DE LOS SENsoRES.....	17
1.3.2.1 Resolución espacial .....	18
1.3.2.2 Resolución espectral .....	19
1.3.2.3 Resolución radiométrica .....	19
1.3.2.4 Resolución temporal.....	20
1.4 SISTEMA SATELITAL NOAA (NATIONAL OCEANIC AND ATMOSPHERIC ADMINISTRATION).....	20
1.4.1 SATÉLITES AMBIENTALES CON OPERACIÓN POLAR (POES) .....	20
1.4.1.1 Características orbitales .....	22
1.4.1.2 Sensor AVHRR (Advanced Very High Resolution Radiometer) .....	23
1.4.1.3 Tiempo de visibilidad de un satélite NOAA.....	26
1.4.1.4 Resolución de los satélites NOAA.....	28
1.4.1.5 Formato APT de los satélites NOAA .....	29
1.4.2 ANTENAS PARA RECEPCIÓN METEoROLÓGICA .....	34
1.4.2.1 Antena Double Cross (DCA) .....	35
1.4.2.2 Antena Turnstile .....	36
1.4.2.3 Antena Cuadrifilar Helicoidal (QHA) .....	37
1.5 RADIO DEFINIDO POR SOFTWARE (RDS) .....	38
1.5.1 HISTORIA DE RDS .....	39

1.5.2 ARQUITECTURA DEL RDS.....	41
1.6 HARDWARE EMPLEADO EN EL PROTOTIPO .....	43
1.6.1 RTL-SDR .....	43
1.6.1.1 Sintonizador R820T .....	44
1.6.1.2 Demodulador RTL2832U .....	45
1.6.2 RASPBERRY PI .....	46
1.7 SOFTWARE EMPLEADO EN EL PROTOTIPO .....	48
1.7.1 RASPBIAN .....	48
1.7.2 GNU RADIO .....	48
1.7.3 WXTOIMG.....	50
1.7.4 SERVIDOR LAMP .....	51
1.7.4.1 Apache .....	52
1.7.4.2 MySQL .....	52
1.7.4.2.1 pphpMyAdmin.....	53
1.7.4.3 PHP .....	53
2 CAPÍTULO 2: DESARROLLO DEL PROTOTIPO .....	54
2.1 REQUERIMIENTOS DEL PROTOTIPO .....	54
2.1.1 INSTALACIÓN DEL SOFTWARE RTL-SDR.....	54
2.1.2 INSTALACIÓN DE GNU RADIO .....	60
2.1.3 INSTALACIÓN WXTOIMG .....	61
2.1.4 INSTALACIÓN DEL SERVIDOR LAMP .....	63
2.2 ESQUEMA Y FUNCIONAMIENTO DEL PROTOTIPO .....	68
2.3 DISEÑO E IMPLEMENTACIÓN DEL PROTOTIPO .....	70
2.3.1 BLOQUE DE LA ANTENA.....	70
2.3.1.1 Antena Turnstile .....	70
2.3.1.1.1 Cálculos de la antena turnstile .....	71
2.3.1.1.2 Construcción de la antena turnstile.....	73
2.3.1.2 Antena Cuadrifilar Helicoidal .....	74
2.3.1.2.1 Cálculos de la antena cuadrifilar helicoidal .....	74
2.3.1.2.2 Construcción de la antena cuadrifilar helicoidal .....	76

2.3.2 BLOQUE DEL MÓDULO RTL-SDR .....	79
2.3.3 BLOQUE DE ADQUISICIÓN .....	79
2.3.4 BLOQUE DE DECODIFICACIÓN.....	85
2.3.5 BLOQUE DE ALMACENAMIENTO Y VISUALIZACIÓN .....	87
2.3.5.1 Almacenamiento.....	87
2.3.5.2 Visualización .....	91
2.3.5.2.1 Inicio .....	91
2.3.5.2.2 Galería .....	92
2.3.5.2.3 Información .....	95
2.3.5.2.4 Estado del Raspberry Pi .....	96
3 CAPÍTULO 3: PRUEBAS Y RESULTADOS.....	98
3.1 PRUEBAS DE FUNCIONALIDAD DEL PROTOTIPO .....	98
3.1.1 FASE DE ADQUISICIÓN .....	103
3.1.1.1 Pruebas con la antena cuadrifilar helicoidal .....	105
3.1.1.2 Pruebas con la antena turnstile .....	108
3.1.2 FASE DE DECODIFICACIÓN .....	111
3.1.3 FASE DE ALMACENAMIENTO.....	129
3.1.4 FASE DE VISUALIZACIÓN.....	131
3.2 ANÁLISIS DE RESULTADOS .....	133
3.3 COSTO REFERENCIAL DEL PROTOTIPO .....	136
4 CAPÍTULO 4: CONCLUSIONES Y RECOMENDACIONES .....	138
4.1 CONCLUSIONES .....	138
4.2 RECOMENDACIONES.....	140
5 REFERENCIAS BIBLIOGRÁFICAS .....	142
6 ANEXOS .....	151

## ÍNDICE DE FIGURAS

Figura 1.1 Elementos de un sistema de teledetección.	1
Figura 1.2 Esquema de una onda electromagnética [3].	2
Figura 1.3 Interacción entre la radiación y la superficie terrestre [1].	6
Figura 1.4 Huella de un satélite [5].	7
Figura 1.5 Subsistemas de la plataforma de un satélite [5].	9
Figura 1.6 Enlaces satelitales [6].	11
Figura 1.7 Tipos de órbitas por la altura [11].	13
Figura 1.8 Tipos de sensores [12].	17
Figura 1.9 Resolución espacial de un satélite [12].	18
Figura 1.10 Órbita de los satélites NOAA [17].	23
Figura 1.11 Barrido del sensor AVHRR [17].	24
Figura 1.12 Sensor AVHRR [19].	24
Figura 1.13 Tiempo de visibilidad de un satélite NOAA.	27
Figura 1.14 NOAA-18 Imagen de la derecha Canal A y de la izquierda Canal B [23].	29
Figura 1.15 Modulación de la señal APT.	30
Figura 1.16 Formato de la trama de una imagen APT [21].	30
Figura 1.17 Segmentos de una imagen APT.	32
Figura 1.18 Trama de Telemetría [18].	33
Figura 1.19 Área de captura para un satélite NOAA [18].	34
Figura 1.20 Diagrama de radiación y posición de los dipolos en la antena DCA [18].	35
Figura 1.21 Antena turnstile [18].	36
Figura 1.22 Antena turnstile con reflector [18].	37
Figura 1.23 Antena QHA [18].	37
Figura 1.24 Diagrama de bloques RDS [26].	41
Figura 1.25 Tx y Rx RDS con FPGA, DSP, ASIC [29].	42
Figura 1.26 Módulo RTL-SDR.	43
Figura 1.27 Placa Raspberry Pi 2 modelo B [34].	47

Figura 1.28 Niveles de GNU Radio [26] .....	49
Figura 1.29 Pantalla principal de GRC. ....	50
Figura 2.1 Actualización de paquetes. ....	55
Figura 2.2 Instalación de git. ....	55
Figura 2.3 Instalación de cmake.....	55
Figura 2.4 Instalación libusb-1.0-0.dev.....	56
Figura 2.5 Instalación de build-essential. ....	56
Figura 2.6 Clonación del paquete rtl-sdr.....	56
Figura 2.7 Cambio de dirección a la carpeta build. ....	57
Figura 2.8 Generación del código del software RTL-SDR.....	57
Figura 2.9 Instalación del software RTL-SDR. ....	58
Figura 2.10 Habilitación de dispositivos RTL-SDR.....	58
Figura 2.11 Configuración de la regla udev.....	59
Figura 2.12 Instalación de gr-osmosdr. ....	59
Figura 2.13 Verificación de la instalación del software RTL-SDR .....	59
Figura 2.14 Instalación de GNU Radio.....	60
Figura 2.15 GNU Radio Companion.....	61
Figura 2.16 Instalación de libgl1-mesa-swx11.....	61
Figura 2.17 Instalación WXtoImg. ....	62
Figura 2.18 Comando para ejecutar el programa WXtoImg. ....	62
Figura 2.19 Configuración inicial WXtoImg.....	62
Figura 2.20 Instalación del servidor Apache.....	63
Figura 2.21 Instalación PHP.....	63
Figura 2.22 Instalación de MySQL. ....	64
Figura 2.23 Configuración de la contraseña del usuario root. ....	64
Figura 2.24 Confirmación de la contraseña del usuario root. ....	64
Figura 2.25 Instalación de PhpMyAdmin. ....	65
Figura 2.26 Configuración del servidor sobre el cual se ejecuta PhpMyAdmin. ....	65
Figura 2.27 Mensaje de configuración de PhpMyAdmin. ....	66
Figura 2.28 Configuración de la contraseña de PhpMyAdmin para acceder a MySQL. ....	66

Figura 2.29 Configuración de la contraseña para ingresar a PhpMyAdmin.....	66
Figura 2.30 Configuración del fichero Apache.....	67
Figura 2.31 Pantalla de inicio del servidor Apache.....	67
Figura 2.32 Pantalla de información de PHP.....	68
Figura 2.33 Pantalla de inicio de PhpMyAdmin.....	68
Figura 2.34 Esquema del sistema prototipo de recepción de imágenes meteorológicas.....	69
Figura 2.35 División de los dipolos.....	71
Figura 2.36 Conexión y longitudes de los cables coaxiales.....	72
Figura 2.37 Base y elementos de los dipolos.....	73
Figura 2.38 Antena Turnstile.....	74
Figura 2.39 Datos ingresados en la calculadora de Jhon Coppens.....	75
Figura 2.40 Resultados para el lazo grande.....	75
Figura 2.41 Resultados para el lazo pequeño.....	76
Figura 2.42 Brazos de la antena.....	77
Figura 2.43 Colocación de los brazos en el mástil de la antena.....	77
Figura 2.44 Brazos y cable coaxial de la antena.....	78
Figura 2.45 Conexión superior de la antena.....	78
Figura 2.46 Conexión inferior de la antena.....	79
Figura 2.47 Tiempo de adquisición de la señal en formato APT.....	81
Figura 2.48 Esquema de recepción en GNU Radio Companion.....	82
Figura 2.49 Base de datos para el sistema prototipo.....	88
Figura 2.50 Base de datos, tablas y clave foránea.....	89
Figura 2.51 Nombre e identificador único de los satélites NOAA.....	90
Figura 2.52 Sección 1 y 2 de la interfaz web Inicio.....	91
Figura 2.53 Sección 3 de la interfaz web Inicio.....	92
Figura 2.54 Galería de la aplicación web.....	94
Figura 2.55 Imágenes seleccionadas por el motor de búsqueda.....	95
Figura 2.56 Interfaz web Información.....	95
Figura 2.57 Interfaz web Estado del Raspberry Pi.....	96
Figura 2.58 Diagrama de secuencia para visualizar las imágenes.....	97

Figura 3.1 Diagrama de flujo de automatización .....	101
Figura 3.2 Fecha y hora para el satélite NOAA-15.....	104
Figura 3.3 Fecha y hora para el satélite NOAA-18.....	104
Figura 3.4 Fecha y hora para el satélite NOAA-19.....	105
Figura 3.5 Esquema para pruebas con la antena QHA.....	106
Figura 3.6 Señal APT del satélite NOAA-19.....	106
Figura 3.7 Señal APT des satélite NOAA-15.....	107
Figura 3.8 Señal APT del satélite NOAA-18.....	107
Figura 3.9 Esquema para pruebas con la antena turnstile.....	108
Figura 3.10 Señal APT del satélite NOAA-19.....	109
Figura 3.11 Señal APT del satélite NOAA-15.....	109
Figura 3.12 Señal APT del satélite NOAA-18.....	110
Figura 3.13 Imagen del canal A del satélite NOAA-19.....	112
Figura 3.14 Imagen del canal B del satélite NOAA-19.....	113
Figura 3.15 Imagen térmica proveniente de satélite NOAA-19.....	114
Figura 3.16 Imagen del canal A del satélite NOAA-15.....	115
Figura 3.17 Imagen del canal B del satélite NOAA-15.....	116
Figura 3.18 Imagen térmica proveniente de satélite NOAA-15.....	117
Figura 3.19 Imagen del canal A del satélite NOAA-18.....	118
Figura 3.20 Imagen del canal B del satélite NOAA-18.....	119
Figura 3.21 Imagen térmica proveniente de satélite NOAA-18.....	120
Figura 3.22 Imagen del canal A proveniente del satélite NOAA-19.....	121
Figura 3.23 Imagen del canal B proveniente del satélite NOAA-19.....	122
Figura 3.24 Imagen termica proveniente del satélite NOAA-19.....	123
Figura 3.25 Imagen del canal A proveniente del satélite NOAA-15.....	124
Figura 3.26 Imagen del canal B proveniente del satélite NOAA-15.....	125
Figura 3.27 Imagen termica proveniente del satélite NOAA-15.....	126
Figura 3.28 Imagen del canal A proveniente del satélite NOAA-18.....	127
Figura 3.29 Imagen del canal B proveniente del satélite NOAA-18.....	128
Figura 3.30 Imagen termica proveniente del satélite NOAA-18.....	129
Figura 3.31 Verificación del almacenamiento de las imágenes.....	130

Figura 3.32 Visualización de imágenes almacenadas del NOAA-19.....	131
Figura 3.33 Visualización de imágenes almacenadas en la del NOAA-18.....	132
Figura 3.34 Visualización de imágenes almacenadas del NOAA-15.....	132
Figura 3.35 Resultado del motor de búsqueda cuando no existen imágenes. ....	133

## ÍNDICE DE TABLAS

Tabla 1.1 Absorción de la atmósfera [4].....	5
Tabla 1.2 Procesos de Dispersión [4].....	5
Tabla 1.3 Bandas de frecuencias satelitales [8].....	12
Tabla 1.4 Fecha de lanzamiento y estado de operatividad de los satélites NOAA [1]. .....	21
Tabla 1.5 Características de los satélites NOAA operativos [16]. .....	22
Tabla 1.6 Características y aplicaciones del sensor AVHRR/3 [1]. .....	25
Tabla 1.7 Resolución de los satélites NOAA.....	28
Tabla 1.8 Información de los segmentos de una imagen APT.....	31
Tabla 1.9 Valores del Slot 16 [18]. .....	34
Tabla 1.10 Características de las antenas para recepción meteorológica [18], [24], [25]. .....	38
Tabla 1.11 Precios de dispositivos RDS [31], [32].....	43
Tabla 1.12 Características del chip R820T [33].....	45
Tabla 3.1 Señal APT en dB y dBm.....	135
Tabla 3.2 Costo referencial del prototipo.....	137

## ÍNDICE DE CÓDIGOS

Código 2.1 Conexión, autenticación y almacenamiento a la base de datos para el caso de la imagen del canal A. ....	90
Código 2.2 Código para mostrar todas las imágenes almacenadas del NOAA-15. ...	92
Código 2.3 Motor de búsqueda. ....	93
Código 2.4 Código para mostrar las imágenes con el motor de búsqueda. ....	93
Código 3.1 Código del web scraping para el NOAA-15.....	103
Código 3.2 Comando para la decodificación de la señal en formato APT.....	111

## RESUMEN

En el presente proyecto se realiza la implementación de un sistema prototipo de adquisición de imágenes meteorológicas provenientes de los satélites NOAA (*National Oceanic and Atmospheric Administration*) utilizando tecnología de Radio Definido por Software (RDS), hardware y software libre.

Inicialmente, se revisan conceptos de teledetección y sistemas satelitales. En cuanto al sistema satelital NOAA, se presentan las características de aquellos satélites que pertenecen a la constelación POES (*Polar Operational Environmental Satellite*) y su formato de transmisión APT (*Automatic Picture Transmission*).

Posteriormente, se describen las características técnicas de la placa Raspberry Pi y del módulo RTL-SDR utilizados en el presente proyecto, así como también, las características del software GNU Radio Companion (GRC) y WXtoImg.

Con esta información se procede a realizar la implementación del prototipo, el cual se encuentra constituido por cinco bloques: bloque de la antena, bloque del módulo RTL-SDR, bloque de adquisición, bloque de decodificación y bloque de almacenamiento y visualización. Cada uno de estos bloques, interactúan entre sí para la correcta adquisición y visualización de las imágenes meteorológicas.

Para el almacenamiento y visualización de las imágenes, se realiza la implementación de un servidor LAMP. Este servidor corre sobre la placa Raspberry Pi, y los módulos que lo componen son los siguientes: sistema operativo Raspbian “Jessie”, servidor web Apache, base de datos MySQL y lenguaje de programación PHP.

Además, mediante el lenguaje de programación Python y la técnica de *Web Scraping*, se ha desarrollado un código que permite obtener las imágenes meteorológicas de forma automática.

Finalmente, se realizan las pruebas de funcionamiento del sistema prototipo, se presentan los resultados obtenidos y se indica un costo referencial del proyecto.

## PRESENTACIÓN

En el presente proyecto se presenta la implementación de un sistema prototipo para la adquisición de imágenes meteorológicas utilizando tecnología de Radio Definido por Software (RDS). Para cumplir con los objetivos planteados, se ha dividido el proyecto en los siguientes capítulos.

En el capítulo uno, se establecen los fundamentos teóricos del presente proyecto. Primero, se realiza una revisión sobre la teledetección y sistemas satelitales. En segundo lugar, se revisa el sistema satelital NOAA (*National Oceanic and Atmospheric Administration*) haciendo énfasis en los satélites pertenecientes a la constelación POES (*Polar Operational Environmental Satellite*) y a su formato de transmisión APT (*Automatic Picture Transmission*). Finalmente, se revisa el fundamento teórico del *hardware* y *software* libre que es utilizado en el sistema prototipo.

En el capítulo dos se realiza el desarrollo del prototipo, aquí en base a los objetivos planteados para el presente proyecto se describen los requisitos que debe cumplir. Así también, se detalla el proceso de implementación del sistema prototipo. Además, se presenta la implementación de una aplicación web que permite visualizar las imágenes meteorológicas obtenidas.

En el capítulo tres se realizan las pruebas de funcionalidad del sistema prototipo y se analizan los resultados. Finalmente, se presenta el costo referencial del prototipo realizado.

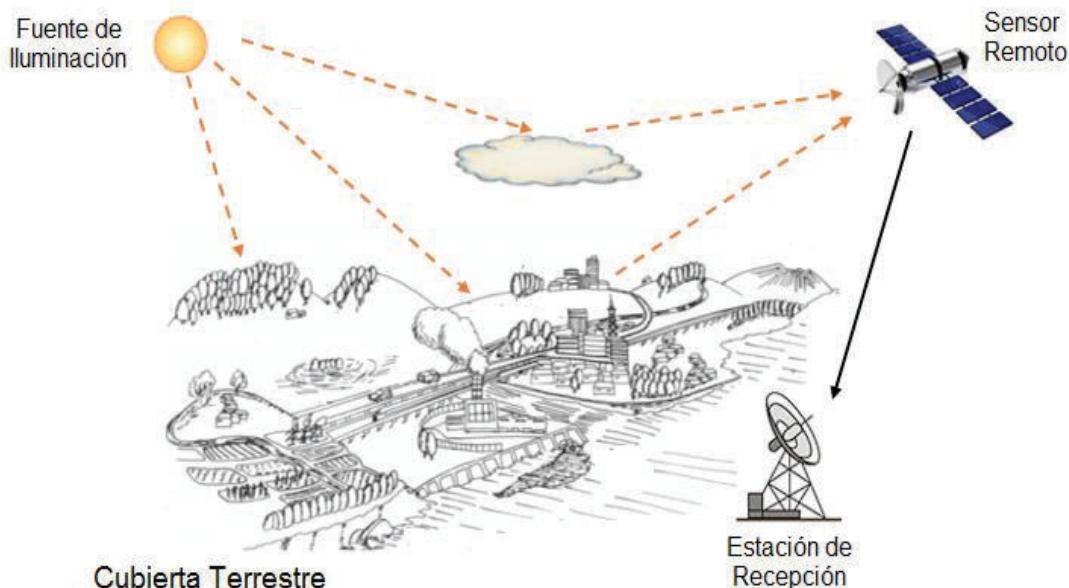
En el capítulo cuatro se presentan las conclusiones que han sido obtenidas durante el desarrollo del presente proyecto, así también se presentan recomendaciones referentes al prototipo y aplicables para futuro proyectos.

# CAPÍTULO 1: FUNDAMENTOS TEÓRICOS

## 1.1 TELEDETECCIÓN [1]

La teledetección o percepción remota puede definirse como la adquisición de información de un objeto, área o fenómeno, mediante el análisis de datos obtenidos por un dispositivo remoto; algunos ejemplos aplicativos de esta tecnología son la realización de los mapas de la temperatura superficial del mar, altura del mar, salinidad, etc. La obtención de estos datos se realiza por medio de la interacción que existe entre los diferentes materiales de la superficie terrestre y las propiedades de la radiación electromagnética.

Un sistema de teledetección está conformado principalmente por los elementos que se observan en la Figura 1.1.



**Figura 1.1** Elementos de un sistema de teledetección\*.

A continuación, se describen los elementos mostrados en la Figura 1.1.

---

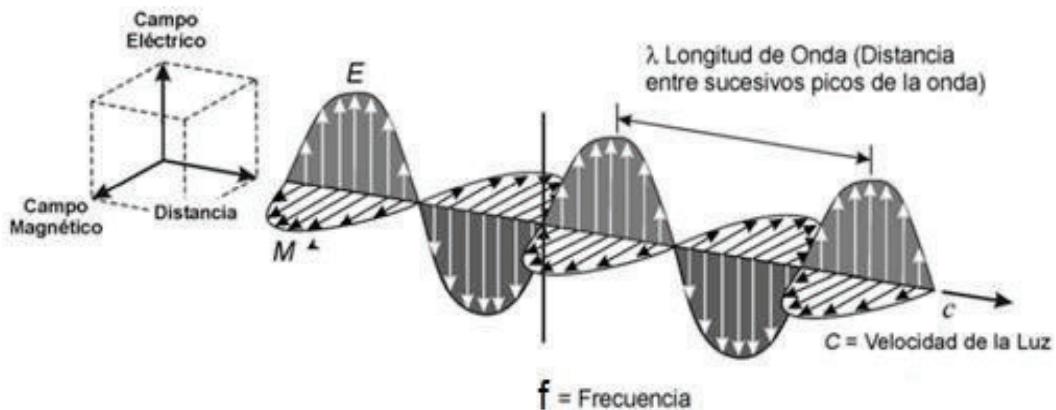
\* Todas las imágenes y tablas que no contengan referencia fueron generadas por el autor de este trabajo.

- **Fuente de iluminación:** Hace referencia a la fuente del flujo energético, que generalmente es la radiación solar.
- **Cubierta terrestre:** Son todos los elementos de la superficie terrestre, tales como vegetación, montañas, ríos, lagos, etc.
- **Sensor remoto:** Es el sensor propiamente dicho junto con la plataforma que lo sustenta; es el encargado de capturar la luz proveniente de los elementos de la superficie.
- **Estación de recepción:** Es el conjunto de equipos que se encargan del procesamiento, almacenamiento y distribución de la información proveniente del sensor remoto.

### 1.1.1 ESPECTRO ELECTROMAGNÉTICO EN TELEDETECCIÓN [2], [3]

En teledetección, el flujo energético que permite la interacción entre la cubierta terrestre y el sensor remoto constituye una forma de radiación electromagnética, la cual, en base a la teoría ondulatoria, se propaga en forma armónica y sinusoidal a la velocidad de la luz ( $c = 3 \times 10^8 \text{ m/s}$ ), conteniendo dos campos de fuerza ortogonales entre sí; el campo eléctrico y el campo magnético.

En la Figura 1.2 se puede observar la propagación de una onda electromagnética, así como también, el campo eléctrico y el campo magnético que la conforman.



**Figura 1.2** Esquema de una onda electromagnética [3].

La radiación electromagnética puede ser descrita por su longitud de onda o por su frecuencia, tal como se muestra en la ecuación (1.1) [2].

$$\lambda = \frac{c}{f} \quad (1.1)$$

Donde:

- $\lambda$  : Longitud de Onda.
- $f$  : Frecuencia.
- $c$ : Velocidad de la luz en el vacío ( $3 \times 10^8 \text{ m/s}$ ).

Para la teledetección es de vital importancia conocer e identificar, en función de la longitud de onda o frecuencia, el comportamiento que tienen los diferentes elementos de la superficie terrestre frente a la radiación electromagnética; ya que esto permitirá determinar cuáles son las bandas más idóneas para la realización de los diferentes procesos de teledetección.

Las bandas espectrales más frecuentemente usadas por los sensores a bordo de los satélites de teledetección, denominadas ventanas atmosféricas, son las siguientes:

- **Espectro visible (0.4 – 0.7  $\mu\text{m}$ )**: Se denomina así debido a que es la única radiación electromagnética que pueden ser percibida por el sistema visual humano. Principalmente, se distinguen tres bandas elementales azul (0.4 – 0.5  $\mu\text{m}$ ), verde (0.5 – 0.6  $\mu\text{m}$ ) y rojo (0.6 – 0.7  $\mu\text{m}$ ). Su principal aplicación en teledetección es para distinguir diferentes características de vegetación y penetración del agua.
- **Infrarrojo cercano (0.7 – 1.3  $\mu\text{m}$ )**: También denominado infrarrojo reflejado, puede detectarse a partir de *films*<sup>1</sup> dotados de emulsiones especiales. La principal aplicación en esta banda consiste en la discriminación de concentraciones de humedad y masas vegetales.

---

<sup>1</sup> *Film*, es una película de poliestireno utilizada para procesos de teledetección.

- **Infrarrojo medio ( $1.3 - 8.0 \mu\text{m}$ )**: En esta banda se entremezclan los procesos de reflexión de la luz solar y de emisión de la superficie terrestre. Permite distinguir la cantidad de vegetación, zonas áridas, nieve, etc.
- **Infrarrojo lejano o térmico ( $8.0 - 14.0 \mu\text{m}$ )**: Incluye la porción emisiva del espectro terrestre. Permite la diferenciación de temperatura, descargas térmicas e inercia térmica.
- **Microondas (a partir de  $1 \text{ mm}$ )**: Es de gran interés por ser un tipo de energía bastante transparente a la cubierta nubosa, polvo e incluso la lluvia, lo que permite la detección de diferentes fenómenos bajo casi todas las condiciones ambientales.

### **1.1.2 INTERACCIÓN DE LA ENERGÍA CON LA ATMÓSFERA [1], [3], [4]**

El sol es la principal fuente de flujo energético que se utiliza en teledetección; este flujo tiene que viajar a través de la atmósfera hasta la superficie terrestre, luego, tiene que volver y atravesar la atmósfera en sentido ascendente para poder ser detectado por los sensores a bordo de los satélites.

El flujo energético proveniente del sol al pasar por la atmósfera, pierde gran parte de su energía inicial, debido a diferentes condiciones como la trayectoria, longitud de onda involucrada, y principalmente por procesos como dispersión y absorción.

#### **1.1.2.1 Absorción**

Como consecuencia de la absorción en la atmósfera, la observación espacial se reduce a determinadas bandas espectrales, debido a que la atmósfera se comporta como un filtro selectivo a distintas longitudes de onda.

Las bandas espectrales que producen una escasa absorción se denominan ventanas atmosféricas, y son precisamente en estas bandas en las que los sensores remotos trabajan.

En la Tabla 1.1 se muestran los tres principales elementos que absorben la radiación solar.

**Tabla 1.1** Absorción de la atmósfera [4].

Elemento	Características
<b>Ozono (<math>O_3</math>)</b>	Absorbe la radiación ultravioleta proveniente del sol, sin embargo, es insignificante en comparación a la absorción total de la radiación solar.
<b>Vapor de Agua (<math>H_2O</math>)</b>	Varía enormemente en la parte baja de la atmósfera y absorbe la radiación de onda larga, infrarrojo térmico y microondas.
<b>Dióxido de Carbono (<math>CO_2</math>)</b>	Absorbe la radiación electromagnética en regiones cercanas al infrarrojo térmico e infrarrojo medio entre 2.5 y 4.5 micrones.

### 1.1.2.2 Dispersión

La interacción entre la radiación, los gases y partículas atmosféricas que se encuentran en suspensión se conoce como dispersión, difusión o *scattering*. En la Tabla 1.2 se muestran los principales procesos de dispersión y sus características.

**Tabla 1.2** Procesos de Dispersión [4].

Proceso de Dispersión	Tamaño de las partículas	Características
<b>Rayleigh</b>	Mucho menor que la longitud de onda incidente	Es la de mayor influencia en teledetección. Es causada por las moléculas de nitrógeno y oxígeno presentes en la atmósfera.
<b>Mie</b>	Mucho mayor que la longitud de onda incidente	Se produce en la parte baja de la atmósfera (0-5 Km) debido al vapor de agua y a las partículas de polvo.
<b>No selectiva</b>	Gran tamaño	Afectando por igual a diferentes longitudes de onda. El color blanco de las nubes se debe en gran parte a este fenómeno.

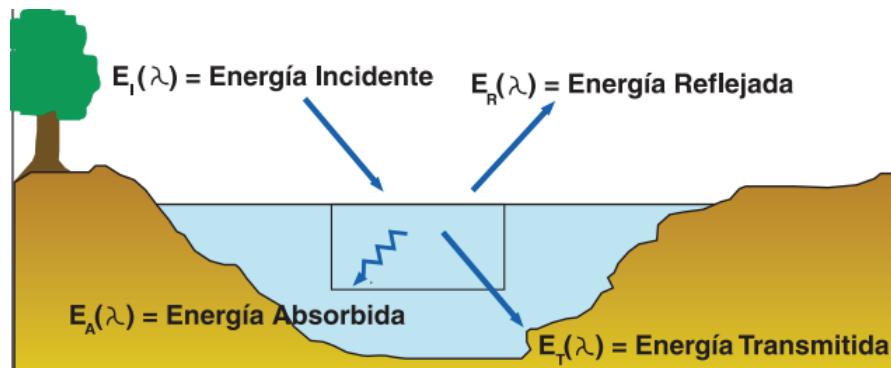
### 1.1.3 INTERACCIÓN DE LA ENERGÍA CON LA SUPERFICIE DE LA TIERRA [1], [3]

Para identificar objetos y procesos sobre la superficie terrestre, es necesario conocer la interacción que existe entre la energía incidente y la superficie terrestre. Esta energía incidente puede descomponerse en tres componentes: uno de reflexión  $E_R(\lambda)$ , uno de absorción  $E_A(\lambda)$  y uno de transmisión  $E_T(\lambda)$ , tal como se observa en la Figura 1.3.

Aplicando el principio de conservación de la energía se tiene la ecuación (1.2) [1].

$$E_1(\lambda) = E_R(\lambda) + E_A(\lambda) + E_T(\lambda) \quad (1.2)$$

Los tres tipos de energía reflexión, absorción y transmisión varían por los distintos elementos de terreno, dependiendo del tipo de material y de su condición. Así también, estas energías variarán a las diferentes longitudes de onda.

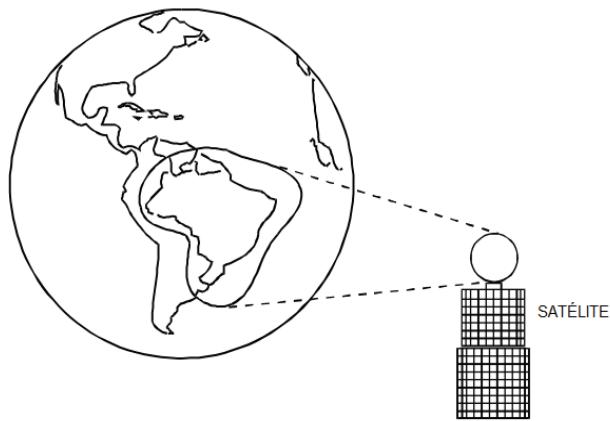


**Figura 1.3** Interacción entre la radiación y la superficie terrestre [1].

## 1.2 SISTEMAS SATELITALES [5], [6]

### 1.2.1 CARACTERÍSTICAS DE LOS SISTEMAS SATELITALES

Un satélite puede definirse como un repetidor activo que se encuentra en el espacio, cuya función consiste en recibir, amplificar y trasladar señales de radiofrecuencia, para luego ser recibidas con una intensidad similar y simultáneamente en cualquier punto de una gran superficie geográfica, tal como se muestra en la Figura 1.4.



**Figura 1.4** Huella de un satélite [5].

Un sistema satelital está conformado básicamente por tres componentes, un *transponder*<sup>2</sup>, una o varias estaciones terrenas y una red de usuario.

Un satélite puede contener uno o varios *transponder*, los cuales tienen la función de amplificar la señal recibida, cambiarla de frecuencia, invertir la polaridad y por último retransmitirla a la zona de cobertura correspondiente.

Las estaciones terrenas permiten realizar el seguimiento, control y supervisión, tanto del satélite como de los sistemas de comunicación. Las funciones de control y supervisión pueden subdividirse de la siguiente forma:

- Telecontrol y telemedida del satélite.
- Pruebas periódicas en órbita de los parámetros principales del módulo de comunicación del satélite.
- Supervisión de la calidad y características de las señales que se emiten a través del satélite.

Adicionalmente, dado que no es posible realizar cambios físicos al satélite una vez que este se encuentre en órbita, para asegurar un correcto funcionamiento ante la posibilidad de fallas en alguno de los equipos, es necesario tener un determinado nivel de redundancia sobre estos equipos.

---

<sup>2</sup> Transponder, dispositivo utilizado en telecomunicaciones que realiza la función de recepción y transmisión de una señal.

En la red de usuario se encuentran los equipos que interactúan directamente con los usuarios finales, los cuales pueden ser fijos o móviles.

A pesar de que los operadores satelitales y empresas privadas son los encargados de poner en marcha los procesos de diseño, fabricación y puesta en órbita de los satélites, no necesariamente son estos quienes brindan servicios a los usuarios finales, sino que, venden o alquilan cierta capacidad de comunicación de los satélites a diferentes prestadores de servicios móviles y de difusión.

### **1.2.2 FUNCIONAMIENTO BÁSICO DE UN SATÉLITE [5]**

Un satélite consta de dos componentes fundamentales para su operación:

- Carga útil o de comunicaciones.
- Plataforma.

Para un adecuado funcionamiento del satélite debe existir una interacción precisa entre ambos componentes, y deben ser monitoreados y controlados en todo momento.

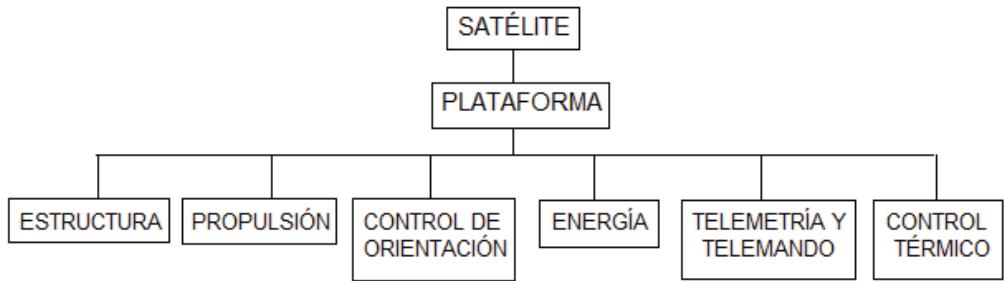
#### **1.2.2.1 Carga útil o de comunicaciones**

Es el conjunto de equipos, antenas, radares y electrónica que procesan las señales de comunicación. La carga útil es diferente para cada satélite, por ejemplo, para un satélite climático la carga útil está constituida por cámaras; mientras que para un satélite de comunicaciones esta carga está constituida principalmente por antenas que transmiten señales hacia la tierra.

#### **1.2.2.2 Plataforma**

La plataforma de un satélite se divide en varios subsistemas, los cuales apoyan la operación satisfactoria de la carga útil.

En la Figura 1.5 se observa la división de la plataforma en sus diferentes subsistemas.



**Figura 1.5** Subsistemas de la plataforma de un satélite [5].

#### 1.2.2.2.1 *Subsistema de estructura*

Es aquel que sirve de soporte, tanto para la carga útil, como para los demás subsistemas de la plataforma. Debe tener la suficiente resistencia para soportar las cargas estáticas y vibraciones del lanzamiento, generalmente está construido con aleaciones metálicas ligeras y compuestos químicos resistentes al deterioro en el espacio.

#### 1.2.2.2.2 *Subsistema de propulsión*

Este subsistema puede tener dos funciones, la primera, mediante un motor de apogeo permitir que el satélite llegue a su órbita destino después de ser liberado por el vehículo de lanzamiento, y la segunda, mediante pequeños impulsores realizar correcciones a las desviaciones que presente el satélite en su órbita.

#### 1.2.2.2.3 *Subsistema de control de orientación*

Este subsistema permite conservar la precisión del apuntamiento de la emisión y recepción de las antenas del satélite, para ello cuenta con sensores y dispositivos giroscópicos como referencia, y con un procesador digital con algoritmos de corrección.

#### 1.2.2.2.4 *Subsistema de energía*

Este subsistema se encarga de la alimentación energética del satélite durante su vida útil. Generalmente, se encuentra constituido por celdas solares que alimentan los

circuitos eléctricos de la nave, baterías que aseguran el suministro durante eclipses o cuando los rayos solares no pueden alcanzar al satélite. Además, de dispositivos de regulación y adaptación que permiten evitar variaciones de tensión o la introducción de pulsos en las líneas de alimentación, evitando así daños en los equipos.

#### 1.2.2.2.5 Subsistema de telemetría y telemando

La telemetría permite conocer el estado de todos los demás subsistemas, mediante la utilización de sensores, los cuales miden diferentes parámetros como la variación de la temperatura, presión, voltaje, corriente eléctrica, etc.

El Telemando es un *software* que permite el envío de órdenes desde una estación de control hacia el satélite por medio de un canal de comunicaciones dedicado. Estas órdenes enviadas al satélite pueden tener efecto tanto en la carga útil como en la plataforma, es por esta razón que solo son admitidas mediante la verificación a través de códigos de seguridad.

#### 1.2.2.2.6 Subsistema de control térmico

Este subsistema emplea conductos de calor y radiadores, que permiten controlar las extremas variaciones de temperatura que existe en el espacio, logrando que los componentes del satélite trabajen dentro de los márgenes tolerables para su adecuado funcionamiento y prolongando su duración.

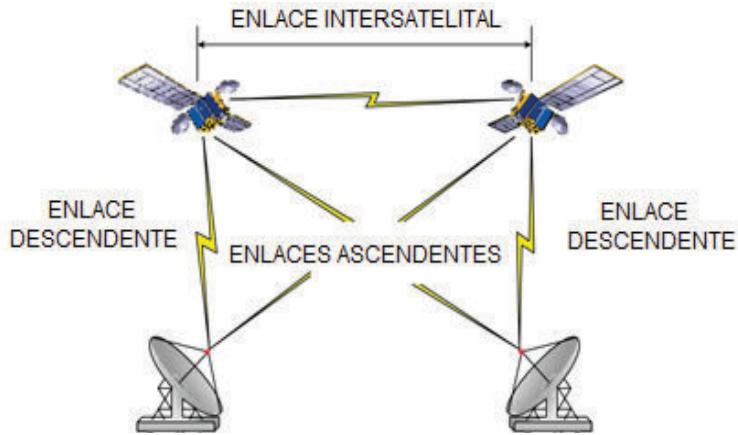
### 1.2.3 ENLACES SATELITALES [6], [7]

Un satélite al ser un repetidor, actúa como enlace entre dos puntos separados geográficamente, permitiendo el intercambio de información entre dichos puntos. Así, en base al sentido de transmisión respecto al satélite, existen dos tipos de enlaces, cada uno de ellos trabajando a frecuencias diferentes para evitar interferencias:

- 1) Enlace ascendente (*Uplink, UL*), para el sentido tierra-espacio.
- 2) Enlace descendente (*Downlink, DL*), para el sentido espacio-tierra.

En la Figura 1.6 se observan los enlaces ascendentes y descendentes. Adicionalmente, se puede observar otro tipo de enlace, el cual se establece entre una

estación satelital transmisora y una estación satelital receptora, sin estaciones terrenas intermedias.



**Figura 1.6** Enlaces satelitales [6].

Los enlaces entre satélites son explotados comercialmente por sistemas globales que necesitan intercambio satelital, como el Sistema de Posicionamiento Global (GPS – *Global Positioning System*), el cual permite determinar la posición de un objeto (una persona, un vehículo, etc.) en la tierra, o como el sistema de la *International Maritime Satellite Organization* (INMARSAT) que ofrece una cobertura global de comunicaciones móviles independientes punto a punto, entre otros servicios.

Las características que pueden determinar un enlace realizado por satélite son:

- Clase de información a administrar, la cual puede ser voz, imágenes y datos.
- Posición de los puntos a enlazar, que pueden ser fijos y móviles.
- Tipos de enlace: punto a punto, punto a multipunto y multipunto a multipunto.

#### 1.2.4 BANDAS DE FRECUENCIAS SATELITALES [6], [8]

La Unión Internacional de Telecomunicaciones (UIT<sup>3</sup>) es la encargada de determinar las bandas de frecuencias que pueden utilizar los satélites, ya sea de forma exclusiva o compartida con otros servicios.

---

<sup>3</sup> Unión Internacional de Telecomunicaciones (UIT), es el organismo especializado de la Organización de las Naciones Unidas para regular, gestionar y desarrollar las tecnologías de la información y la comunicación a nivel internacional.

Las bandas de frecuencia más comunes utilizadas para el servicio satelital se encuentran asociadas con letras y trabajan en el orden de los GHz, tal como se muestra en la Tabla 1.3; cada una de estas bandas puede ser utilizada simultáneamente en varios países, tomando las debidas precauciones a fin de evitar interferencias.

Al trabajar a altas frecuencias, se presenta como principal desventaja que la mayor parte de las ondas son absorbidas por objetos como las hojas y gotas de lluvia, lo que conlleva a necesitar equipos transmisores con mayor potencia o antenas más enfocadas, provocando que el costo del satélite aumente.

**Tabla 1.3** Bandas de frecuencias satelitales [8].

Banda	Rango de Frecuencia [GHz]	Servicio	Usos
<b>VHF</b>	0.03 – 0.3	Fijo	Telemetría.
<b>UHF</b>	0.3 – 1	Móvil	Navegación, Militar.
<b>L</b>	1 – 2	Móvil	Emisión de audio, radiolocalización.
<b>S</b>	2 – 4	Móvil	Navegación.
<b>C</b>	4 – 8	Fijo	Voz, Datos, Video, Emisión de video.
<b>X</b>	8 – 12	Fijo	Militar.
<b>Ku</b>	12 – 18	Fijo	Voz, Datos, Video, Emisión de video.
<b>K</b>	18 – 27	Fijo	Emisión de video, comunicación intersatelital.
<b>Ka</b>	27 - 40	Fijo	Emisión de video, comunicación intersatelital.

### 1.2.5 TIPOS DE ÓRBITAS SATELITALES [9], [10]

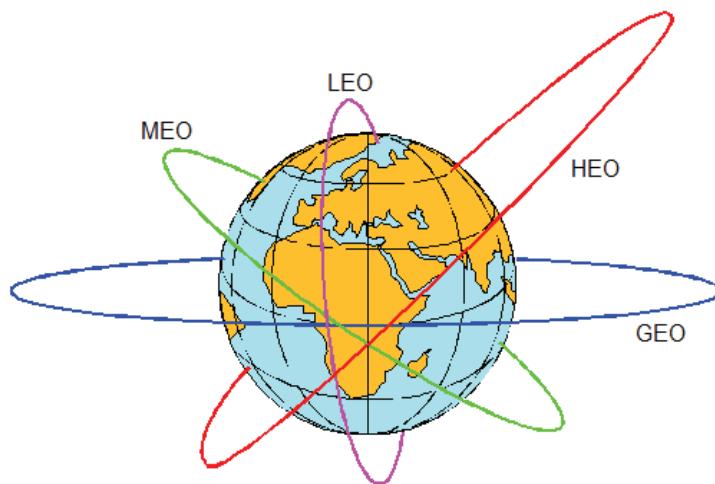
La órbita es el recorrido o trayectoria que describe un satélite a través del espacio bajo la influencia de las fuerzas de atracción y repulsión de la tierra.

Las propiedades de las órbitas satelitales se resumen en las 3 leyes de Kepler, las cuales manifiestan lo siguiente:

- 1) Los planetas describen órbitas elípticas, con el Sol en uno de sus focos.
- 2) El vector de posición de cualquier planeta, relativo al Sol, barre áreas iguales de su elipse, en tiempos iguales.
- 3) La relación de la distancia media,  $d$ , de un planeta al sol, elevada al cubo, dividida por el cuadrado de su periodo orbital,  $t$ , es una constante, tal como se muestra en la ecuación (1.3) [9].

$$\frac{d^3}{t^2} = \text{constante} \quad (1.3)$$

La clasificación más importante en cuanto al tipo de órbita que describe un satélite está en función de la distancia a la que se encuentra sobre la superficie terrestre, tal como se muestra en la Figura 1.7, pero no es la única, también puede clasificarse por la forma de la trayectoria de la órbita en circular o elíptica y por su recorrido en polar, ecuatorial e inclinada.



**Figura 1.7** Tipos de órbitas por la altura [11].

#### 1.2.5.1 Órbita baja (LEO, *Low Earth Orbit*)

La altura de los satélites en esta órbita oscila típicamente entre los 200 y 1600 Km. Su periodo orbital varía entre los 90 y 120 minutos, presentando como principal ventaja la velocidad de transmisión (cientos de Mbits/s) y una reducida latencia (centésimas de segundos); el principal inconveniente que se presenta en esta órbita es que se necesita alrededor de 50 satélites para proporcionar una cobertura global de la tierra.

La inclinación que presenta esta órbita respecto al plano ecuatorial se encuentra entre los 45 y 90 grados, y dependerá si su punto de observación es la tierra o el espacio exterior.

#### 1.2.5.2 Órbita media (MEO, *Medium Earth Orbit*)

Los satélites en esta órbita se encuentran entre los 10.000 y 20.000 Km de altitud. Su periodo orbital es alrededor de 6 horas y se necesita de aproximadamente 10 satélites para tener una cobertura global. La inclinación que presenta respecto al plano ecuatorial es de 45 a 90 grados.

Los satélites que operan en esta órbita lo hacen de manera similar a como lo hacen los satélites de órbita baja, pero, presentando un movimiento relativo a la superficie de la tierra mucho más lento, por lo que son utilizados principalmente por los Sistemas de Posicionamiento Global (GPS - *Global Positioning System*), comunicaciones móviles y aplicaciones meteorológicas.

#### 1.2.5.3 Órbita alta (GEO, *Geostationary Orbit*)

Esta órbita es también conocida como Clark<sup>4</sup>, los satélites en esta órbita se encuentran a aproximadamente 36.000 Km de altitud. Su periodo orbital es de 24 horas, por lo que parecen estar fijos sobre la superficie de la tierra. Teóricamente se necesita de tan solo 3 satélites para tener una cobertura global.

---

<sup>4</sup> Denominada así por el escritor Británico Arthur C. Clark quien publicó un artículo titulado “*Extra-terrestrial relays*” en la revista *Wireless Word* en 1945.

Los satélites que se encuentran en esta órbita necesitan tener posiciones orbitales específicas, las cuales deben estar separadas 1 ó 2 grados (800 o 1600 Km), para las bandas C y K respectivamente.

En la actualidad la vida útil planeada para los satélites en esta órbita es de 10 años o más, al terminar su vida útil deben ser desactivados y colocados en una órbita por lo menos 150 Km más alejada, quedando a la deriva y sin control, junto con otros satélites, cohetes y fragmentos.

#### **1.2.5.4 Órbitas de alta excentricidad (HEO, *High Earth Orbit*)**

Los satélites en este tipo de órbita presentan dos distancias respecto a la tierra. La primera, cuando el satélite se encuentra más allá de los 40.000 Km y la segunda, cuando se encuentra a unos 500 Km; a estas distancia se las denomina apogeo y perigeo respectivamente. Su periodo orbital es de 12 horas y con varios satélites espaciados se puede mantener una cobertura global.

#### **1.2.5.5 Órbita polar**

Los satélites en esta órbita se encuentran a unos 800 Km de altitud. Su inclinación respecto al plano ecuatorial es superior a los 90 grados. Al no estar alineados con el eje terrestre, son utilizados cuando se necesita obtener información de la tierra o de un fenómeno en particular de forma repetitiva.

### **1.2.6 TIPOS DE SATÉLITES POR SU APLICACIÓN [9]**

Se estima que desde el lanzamiento del *Sputnik I* en 1957 se han enviado al espacio más de 15.000 satélites. Actualmente existen alrededor de 1000 satélites operativos, cada uno de ellos diseñado y construido para una misión específica, por lo que parámetros como, la altura a la que debe ser ubicado, frecuencia de operación, tipo de órbita, entre otros, son analizados minuciosamente en base a objetivos y aplicaciones, las cuales se basan en la recepción, almacenamiento y reenvío de información.

Las principales aplicaciones de los satélites se tienen a nivel de navegación, teledetección, comunicación, aplicaciones militares, meteorológicas y científicas.

- **Satélites científicos:** Son aquellos construidos para el estudio de la tierra y del espacio exterior, se centran en la observación solar, investigación de fuentes estelares, etc. Algunos de los principales fenómenos en estudio son los realizados sobre los cinturones de radiación que rodean la tierra (cinturones de Van Allen), la densidad de la atmósfera de la tierra y estudios geodésicos.
- **Satélites meteorológicos:** Son satélites especializados que presentan una órbita geoestacionaria o polar, permiten el monitoreo del clima y la elaboración del pronósticos del tiempo. Sus principales objetivos pueden ser:
  - Comprensión de la física/dinámica de la atmósfera.
  - Comportamiento de las masas nubosas o el movimiento del aire frío o caliente.
  - Conocimiento de la temperatura de la atmósfera.
- **Satélites militares:** Este tipo de satélites son construidos para brindar apoyo en operaciones militares de ciertos países.
- **Satélites de comunicación:** Este tipo de satélites que por excelencia tienen una órbita geoestacionaria son los que hacen posible la comunicación de la sociedad moderna; gracias a ellos es posible tener alrededor del mundo televisión de alta definición, acceso a Internet, teleconferencias en tiempo real, etc.
- **Satélites de teledetección:** Son utilizados para la observación terrestre mediante el uso de sensores multiespectrales. Suelen tener órbitas bajas y polares, a menudo sincronizadas con el sol.
- **Satélites de navegación:** Estos satélites trabajan a frecuencias bajas y abiertas al público y son utilizados como sistemas de posicionamiento global (GPS).

### 1.3 SENORES SATELITALES [12]

Son los dispositivos capaces de detectar y recibir radiación electromagnética proveniente de la superficie de la Tierra en cualquier rango de longitud de onda, en la cual el sensor sea sensible. Sin embargo, por razones técnicas y atmosféricas los

rangos en los cuales trabajan son en las denominadas ventanas atmosféricas, las cuales fueron mencionadas en la sección 1.1.1.

### 1.3.1 TIPOS DE SENSORES

Los sensores se pueden dividir según la fuente del flujo energético que reciben, en activos y pasivos, tal como se muestra en la Figura 1.8.

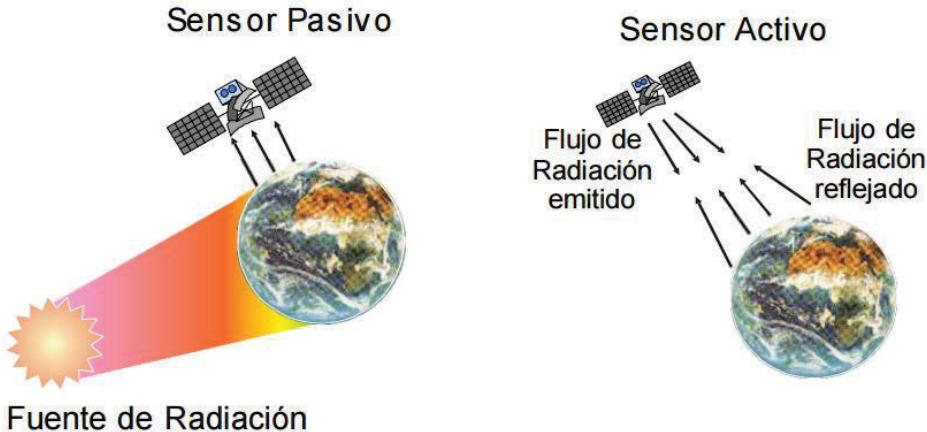


Figura 1.8 Tipos de sensores [12].

#### 1.3.1.1 Sensores activos

Son aquellos que no dependen de una fuente externa para irradiar la superficie terrestre, es decir, emiten su propio flujo de radiación hacia la tierra y captan el flujo de energía reflejado. Ejemplos de este tipo de sensor son los radares y radiómetros de microondas.

#### 1.3.1.2 Sensores pasivos

Este tipo de sensores no pueden emitir su propio flujo de radiación, sino que dependen de una fuente de iluminación externa como el sol, es decir, detectan el flujo energético reflejado o emitido por la superficie terrestre que proviene de una fuente externa.

### 1.3.2 RESOLUCIÓN DE LOS SENSORES [13]

Se puede definir a la resolución como la capacidad para registrar o discriminar la información en forma detallada. Este factor se encuentra estrechamente relacionado

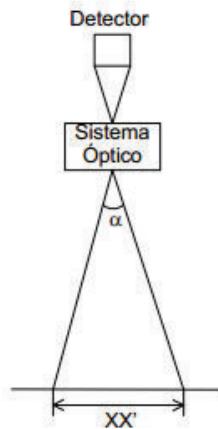
con el número de pixeles, es decir, a mayor número de pixeles por unidad de superficie mayor resolución del sensor.

En definitiva, el concepto de resolución implica al menos cuatro manifestaciones: espacial, espectral, radiométrica y temporal.

### 1.3.2.1 Resolución espacial

Se puede definir a la resolución espacial como el área de la superficie que puede ser vista por un sensor a bordo de un satélite a una altura dada, es decir, el objeto más pequeño que puede ser distinguido en una imagen.

El área vista por el satélite puede ser medida de dos formas distintas, mediante un ángulo ( $\alpha$ ) de apertura del sistema óptico-electrónico medido en radianes y denominada campo de visión instantáneo IFOV (*Instantaneous Field of View*), o como la distancia equivalente en la superficie ( $XX'$ ), tal como se muestra en la Figura 1.9.



**Figura 1.9** Resolución espacial de un satélite [12].

El ángulo de apertura así como la distancia equivalente en la superficie dependen de varios factores como: la altura orbital, velocidad de exploración y número de detectores. Debido a que ningún satélite posee una órbita perfecta, el parámetro de resolución espacial se da siempre a una altura nominal del satélite.

### 1.3.2.2 Resolución espectral

La resolución espectral está estrechamente relacionada con la aplicación y objetivos del satélite, ya que de estos dependerá el número y la anchura de las bandas espectrales que puede discriminar el sensor que se encuentra a bordo del satélite.

Mientras más bandas posea un satélite, con mayor exactitud podrá discriminar lo que está analizando, sin embargo el aumentar el número de bandas implica en primer lugar el aumento de costos que en ocasiones hacen inviables su desarrollo, en segundo lugar el mayor número de componentes mecánicos y/o electrónicos provoca que el nivel de ruido se amplifique notablemente, y en tercer lugar el crecimiento considerable de la información que se debe procesar y enviar conlleva nuevamente a costos adicionales, debido al aumento de componentes de procesamiento.

En cuanto a la anchura de las bandas conviene que estas sean lo suficientemente estrechas, dado que si no lo son se obtendrá un valor promedio, el cual puede llegar a encubrir la diferencia espectral entre objetos de interés.

### 1.3.2.3 Resolución radiométrica

Hace mención a la sensibilidad del sensor, es decir, el número de niveles digitales empleados para expresar los datos recogidos. Estos niveles hacen referencia a la intensidad de radiación electromagnética que es capaz de medir sin saturarse, por lo que el rango del espectro electromagnético en el cual debe trabajar el sensor, dependerá del tipo de monitoreo que se realice.

Los satélites que presentan mayor resolución radiométrica permiten tener una mejor interpretación de las imágenes, sin embargo, hay que tener en cuenta el tamaño del paso de un nivel digital de radiancia<sup>5</sup> a otro, ya que este no puede ser menor que el nivel de ruido o de lo contrario no es posible discriminar entre un cambio real de nivel o un cambio en la magnitud del ruido.

---

<sup>5</sup> Radiancia es el flujo radiante por unidad de ángulo sólido, que sale de una superficie emisora en una dirección dada, por unidad de área en esa dirección.

#### 1.3.2.4 Resolución temporal

Se refiere al ciclo o periodicidad con la cual el sensor adquiere datos de la misma porción de la superficie terrestre, este ciclo dependerá del tipo de evento o fenómeno a ser monitoreado. La resolución temporal va en función de las características orbitales como la altitud, velocidad e inclinación y principalmente del ángulo de observación.

### 1.4 SISTEMA SATELITAL NOAA (NATIONAL OCEANIC AND ATMOSPHERIC ADMINISTRATION) [13], [14], [15]

La NOAA es uno de los programas espaciales atmosféricos más antiguos y con mayor tradición en el mundo, y se encarga de la administración de varios satélites meteorológicos. Estos satélites dependiendo de sus funciones y características pueden pertenecer a una de las tres constelaciones que administra la NOAA, las cuales son:

- 1) Satélites Ambientales con Operación Geoestacionaria (GOES – *Geostationary Operational Environmental Satellite*)
- 2) Programa de Satélites Meteorológicos de Defensa (DMSP – *Defense Meteorological Satellite Program*)
- 3) Satélites Ambientales con Operación Polar (POES – *Polar Operational Environmental Satellite*)

En el presente proyecto, los satélites de los cuales se obtendrán las imágenes meteorológicas presentan una órbita polar, por lo tanto, estos satélites pertenecen a la constelación POES.

#### 1.4.1 SATÉLITES AMBIENTALES CON OPERACIÓN POLAR (POES)

Estos satélites ofrecen una cobertura global diaria para el pronóstico del tiempo, con órbitas matutinas y vespertinas. El primer satélite meteorológico destinado a la recolección de datos de la tierra, perteneciente a la constelación POES fue el TIROS-N, el cual fue lanzado en Octubre de 1978.

Actualmente, los satélites pertenecientes a esta constelación son conocidos como NOAA-n, donde “n” es un número asignado al satélite cuando este se encuentra en órbita, por ejemplo NOAA-18 y NOAA-19.

En la Tabla 1.4 se muestran los diferentes satélites puestos en órbita hasta la actualidad, así también, su fecha de lanzamiento y estado de operatividad.

**Tabla 1.4** Fecha de lanzamiento y estado de operatividad de los satélites NOAA [1].

Satélite	Fecha de Lanzamiento	Operatividad
TIROS-N	13/10/1978	Inactivo
NOAA-6	27/06/1979	Inactivo
NOAA-7	23/06/1981	Inactivo
NOAA-8	28/03/1983	Inactivo
NOAA-9	12/12/1984	Inactivo
NOAA-10	17/09/1986	Inactivo
NOAA-11	24/09/1988	Inactivo
NOAA-12	14/05/1991	Inactivo
NOAA-13	09/08/1993	Inactivo
NOAA-14	30/12/1994	Inactivo
NOAA-15	15/05/1998	Activo
NOAA-16	21/09/2000	Inactivo
NOAA-17	24/06/2002	Inactivo
NOAA-18	20/05/2005	Activo
NOAA-19	06/02/2009	Activo

Los principales sensores a bordo de los satélites que se encuentran operativos son los siguientes:

- a) *Advanced Very High Resolution Radiometer* (AVHRR), utilizado para la observación de radiancias y temperaturas globales tanto de día como de noche.

- b) *Tiros Operational Vertical Souder* (TOVS), utilizado para la obtención de perfiles verticales de temperatura y vapor de agua en la atmósfera terrestre.
- c) *Space Environment Monitor* (SEM), utilizado para la medición del flujo de electrones y protones cerca de la Tierra.

La información exacta y detallada sobre los satélites operativos NOAA es suministrada diariamente por el USSC (*United States Space Command*) y se encuentra disponible para el público en general.

Para la obtención de las imágenes meteorológicas en el presente proyecto se utilizará la información generada por el sensor AVHRR/3, el cual se encuentra a bordo de los tres satélites NOAA que actualmente se encuentran operativos.

En la Tabla 1.5 se muestran las principales características de estos tres satélites.

**Tabla 1.5** Características de los satélites NOAA operativos [16].

Satélite	NOAA-15	NOAA-18	NOAA-19
Altura aproximada	870 Km	854 Km	870 Km
Inclinación de la órbita	98.86°	98.74°	98.730°
Transmisión APT	137.620 MHz	137.9125 MHz	137.1 MHz
Transmisión HRPT	1702.5 MHz	1707 MHz	1698 MHz
Sensor	AVHRR/3	AVHRR/3	AVHRR/3

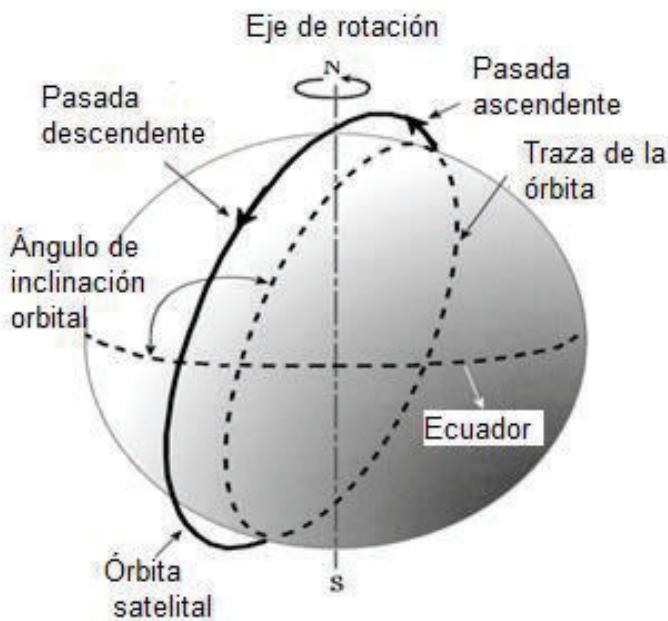
#### 1.4.1.1 Características orbitales [17]

Los satélites NOAA describen una órbita polar heliosíncrona, es decir, que orbitan de polo a polo a una frecuencia que se encuentra sincronizada con el sol, lo que les permite pasar sobre un área determinada aproximadamente a la misma hora del día, situación que resulta útil para análisis comparativos de datos multitemporales.

Su periodo orbital es de 101 minutos y se encuentran a una altura de alrededor de 865 Km. Como se observa en la Figura 1.10, la órbita de los satélites NOAA se encuentra

ligeramente inclinada respecto al eje de rotación de la tierra, esta inclinación toma valores entre  $98.7^{\circ}$  y  $98.9^{\circ}$  respecto al plano ecuatorial.

Los satélites pueden registrar datos en dos sentidos, cuando se dirigen de Norte a Sur (descendente) recogen datos de Este a Oeste y cuando se dirigen de Sur a Norte (ascendente) recogen datos de Oeste a Este, en este caso las imágenes obtenidas se presentan invertidas.



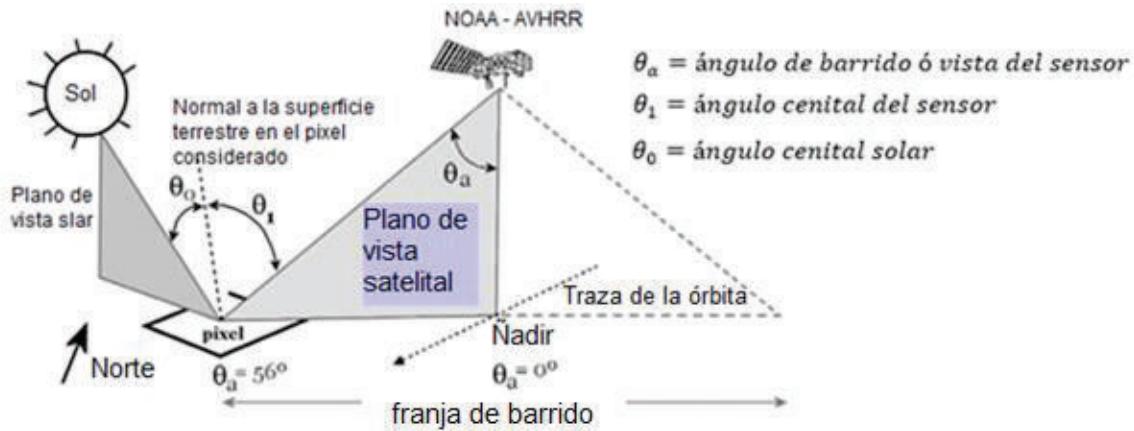
**Figura 1.10** Órbita de los satélites NOAA [17].

#### 1.4.1.2 Sensor AVHRR (Advanced Very High Resolution Radiometer) [17], [18]

Es un sensor de tipo barredor, capaz de medir radiación en diferentes zonas del espectro electromagnético. El ángulo de barrido varía entre  $0^{\circ}$  en el nadir<sup>6</sup> y aproximadamente  $\pm 56^{\circ}$  en los extremos de las imágenes, con una franja de barrido de aproximadamente 2700 Km, tal como se muestra en la Figura 1.11.

---

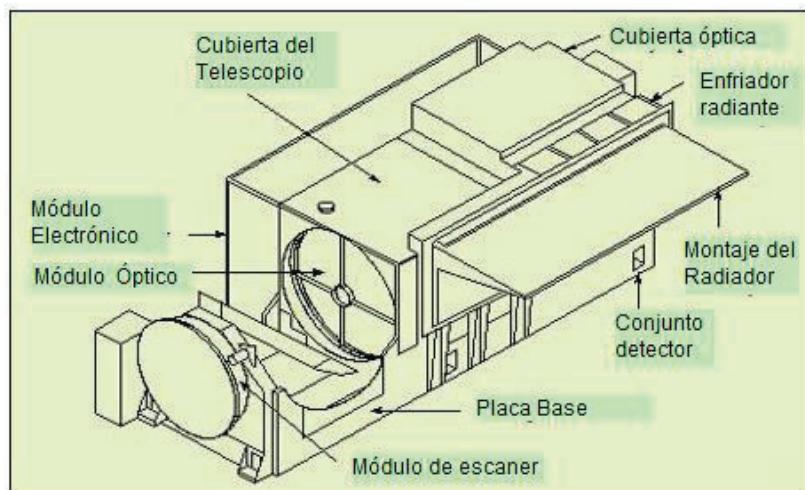
<sup>6</sup> Nadir: Pixel central o subsatélite, ubicado en la línea imaginaria que resulta de proyectar la órbita satelital sobre la tierra.



**Figura 1.11** Barrido del sensor AVHRR [17].

La primera versión de este sensor fue el AVHRR/1 el cual podía captar información en 4 bandas y se encontraba presente en los NOAA-6,-8,-10; la segunda versión fue el AVHRR/2 que podía captar información en 5 bandas y se encontraba presente en los NOAA-7,-9,-11,-12,-14 y la última versión es el AVHRR/3 el cual puede captar información en 6 bandas y se encuentra presente en los NOAA-15,-16,-17,-18,-19.

El sensor AVHRR/3 presente en los satélites NOAA operativos, es un sistema óptico compuesto por cinco módulos: módulo de escáner, módulo electrónico, enfriador radiante, sistema óptico y la unidad de placa base, tal como se muestra en la Figura 1.12.



**Figura 1.12** Sensor AVHRR [19].

Los datos obtenidos por los 6 bandas del sensor AVHRR/3 son muestreados simultáneamente a 40 KHz, para posteriormente convertirse en valores binarios de 10 bits. La combinación de las 6 diferentes bandas permite generar imágenes en falso color<sup>7</sup> para destacar elementos de interés.

En la Tabla 1.6, se muestran las diferentes bandas del sensor AVHRR/3 y su principal aplicación.

**Tabla 1.6** Características y aplicaciones del sensor AVHRR/3 [1].

Banda	Rango Espectral	Longitud de onda ( $\mu\text{m}$ )	Aplicación
1	Visible	0.58 – 0.68	Mapeo diurno de nubes y de superficie.
2	Infrarrojo Cercano	0.75 – 1.00	Límite tierra- agua.
3 <sup>a</sup>	Infrarrojo Medio	1.58 – 1.64	Detección de hielo, nieve y nubes.
3B	Infrarrojo Térmico	3.55 – 3.93	Mapeo nocturno de nubes, temperatura superficial del mar y detección de fuego.
4	Infrarrojo Térmico	10.3 – 11.3	Mapeo nocturno de nubes y temperatura superficial del mar.
5	Infrarrojo Térmico	11.5 – 12.5	Temperatura superficial del mar.

El sensor AVHRR/3 es probablemente el sensor más utilizado, tanto en estudios científicos como en aplicaciones operacionales oceanográficas.

---

<sup>7</sup> Falso color: las bandas se combinan de manera muy específica para crear una imagen de composición en color. El color rojo se asigna a la banda del infrarrojo cercano, el verde a la banda de rojo visible y el azul a la banda visible del verde. Así, la vegetación aparecerá rojiza, el agua azulosa y el suelo en tono gris y marrón.

La información del sensor AVHRR/3 es procesada y separada en diferentes tramas, las cuales son transmitidas desde el satélite hacia las diferentes estaciones terrenas. Los formatos de transmisión son las siguientes:

- a) *High Resolution Picture Transmission* (HRPT): se transmite de forma continua y se trata de datos de alta resolución.
- b) *Local Area Coverage* (LAC): son también datos de alta resolución, pero se almacenan a bordo del satélite y son enviados a tierra posteriormente.
- c) *Global Area Coverage* (GAC): se obtienen considerando solo una de cada tres filas de datos LAC y promediando a su vez dentro de cada fila los valores de 4 píxeles adyacentes.
- d) *Automatic Picture Transmission* (APT): es una transmisión continua de información en formato análogo.

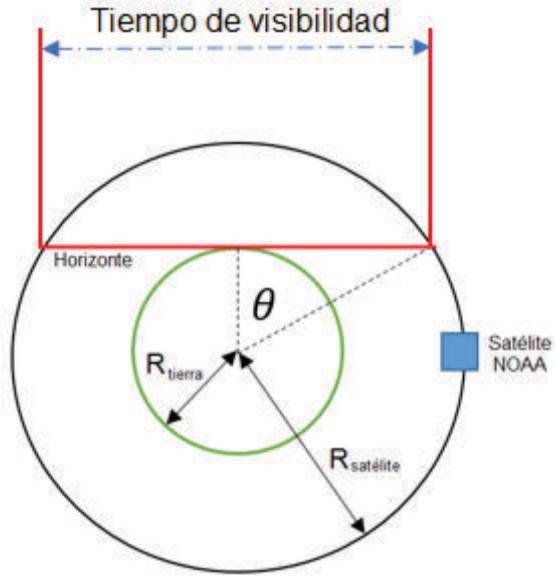
En el presente proyecto, las imágenes meteorológicas que se obtendrán vienen dadas en formato APT (*Automatic Picture Transmission*).

#### 1.4.1.3 Tiempo de visibilidad de un satélite NOAA [20]

Los satélites NOAA, al describir una órbita polar heliosíncrona, no se encuentran visibles en todo momento sobre un punto específico de la tierra; solo se encuentran visibles por un determinado tiempo.

En la Figura 1.13, se puede observar que un satélite NOAA podrá ser visible para un punto específico de la tierra durante el tiempo que el satélite demore en recorrer la porción de órbita que se encuentra sobre la línea roja, dicha línea roja representa el horizonte.

También se puede observar en la Figura 1.13, que mediante el uso de la geometría se puede calcular el tiempo de visibilidad, haciendo uso del ángulo  $\theta$ , teniendo en consideración que el periodo orbital de 101 minutos de los satélites NOAA representa un ciclo completo igual a  $2\pi$ .



**Figura 1.13** Tiempo de visibilidad de un satélite NOAA.

Para el cálculo del tiempo de visibilidad de un satélite NOAA se establece la ecuación (1.4) [20]:

$$\cos \theta = \frac{R_{tierra}}{R_{tierra} + R_{polar}} \quad (1.4)$$

Donde:

- $R_{tierra}$ : Es el radio de la Tierra, 6378 Km [20].
- $R_{polar}$ : Es la altura aproximada a la que se encuentran los satélites NOAA, 865 Km.
- $\theta$ : Es el ángulo que representa la mitad del tiempo de visibilidad.

Se remplaza valores en la ecuación 1.4 y se calcula el arco-coseno, tal como se muestra a continuación:

$$\cos^{-1} \theta = \frac{6378}{6378 + 865}$$

$$\theta = 0.49 \text{ radianes}$$

$$2\theta = 0.98 \text{ radianes}$$

Puesto que el ángulo que representa el tiempo de visibilidad ( $2\theta$ ), se encuentra en radianes se hace uso de una regla de tres simple para el cálculo del tiempo de visibilidad en minutos.

$$\frac{2\pi}{0.9874} \quad \frac{101 \text{ minutos}}{t \text{ minutos}}$$

$$t = \frac{0.98 * 101}{2\pi}$$

$$t = 15.7 \text{ minutos}$$

El tiempo calculado de 15.7 minutos es una caso ideal, en la que una estación receptora, específicamente la antena, pueda captar la señal en formato APT desde el inicio hasta el fin del horizonte.

#### 1.4.1.4 Resolución de los satélites NOAA

Como se mencionó anteriormente en la sección 1.3.2, la resolución de un sistema satelital implica al menos cuatro manifestaciones, así también, está estrechamente relacionada con el número de pixeles del sensor utilizado.

En la Tabla 1.7 se muestra la resolución de sensor AVHRR/3 de los satélites NOAA, de los cuales se obtendrán las imágenes meteorológicas del presente proyecto.

**Tabla 1.7** Resolución de los satélites NOAA.

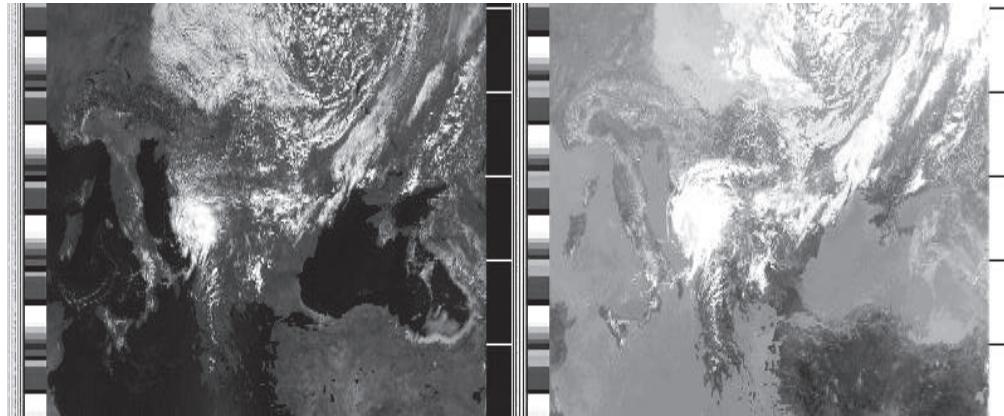
Resolución	Parámetro
Espacial	<b>Alta resolución - 1.1 Km</b> <b>Baja resolución – 4 Km</b>
Espectral	6 bandas
Radiométrica	Alta resolución - 10 bits Baja Resolución – 8 bits
Temporal	12 horas

#### 1.4.1.5 Formato APT de los satélites NOAA [18], [21], [22]

La transmisión en modo analógico de baja resolución de los satélites NOAA es el formato APT (*Automatic Picture Transmission*), cuya transmisión se realiza mediante la utilización de dos de los seis canales que posee el sensor AVHRR/3.

Las imágenes que se obtienen en formato APT son de luminancia, también conocida como escala de grises, en consecuencia los tonos oscuros representan las regiones cálidas y los tonos brillantes las regiones frías, tal como se muestra en la Figura 1.14.

Estas imágenes constan de dos canales, A y B. El canal A durante el día hace uso de la banda del espectro visible y en la noche utiliza la banda del infrarrojo medio, en tanto que el canal B, utiliza la banda del infrarrojo térmico constantemente durante el día y la noche.



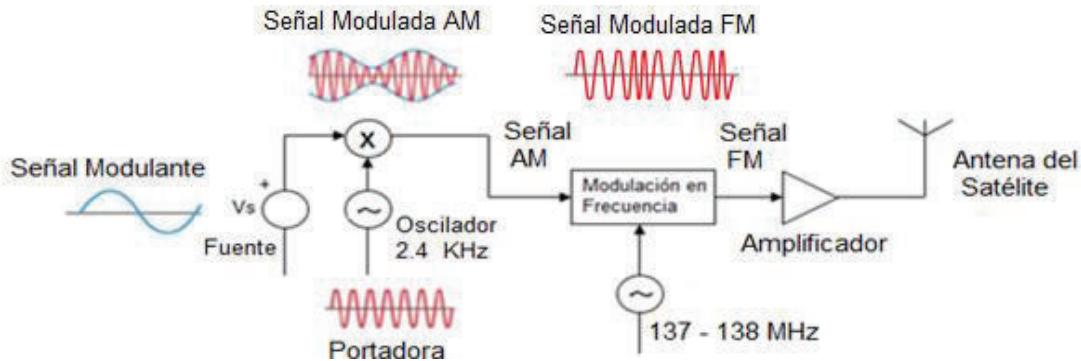
**Figura 1.14** NOAA-18 Imagen de la derecha Canal A y de la izquierda Canal B [23].

Para la transmisión de estas imágenes en formato APT, el flujo de datos es modulado en amplitud (Modulación AM) a una frecuencia de 2400 Hz, dando como resultado una señal que varía en amplitud dependiendo de la información proveniente del sensor AVHRR/3. Luego, es modulada en frecuencia (Modulación FM) entre los 137 y 138 MHz presentando una desviación pico máxima<sup>8</sup> ( $\Delta f$ ) de 17 KHz, y por último la señal

---

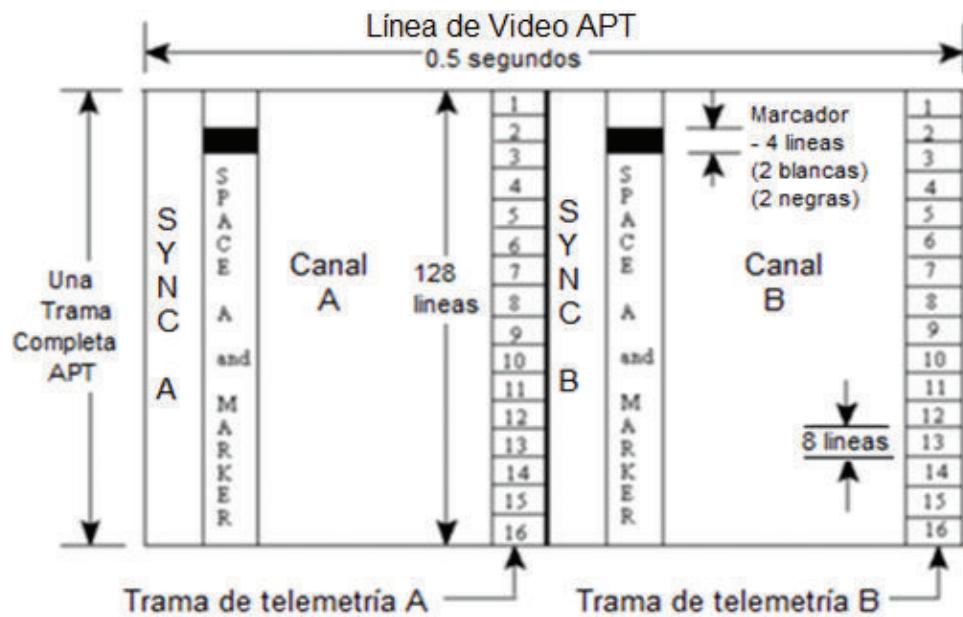
<sup>8</sup> Desviación pico máxima ( $\Delta f$ ), es el máximo cambio de frecuencia pico que puede experimentar la frecuencia de la señal portadora.

es amplificada y transmitida hacia la tierra. El procesamiento de la señal se observa en la Figura 1.15.



**Figura 1.15** Modulación de la señal APT.

El sensor AVHRR/3 puede enviar 360 líneas de video por minuto para transmisiones de alta resolución, dado que el formato APT es de baja resolución una estación receptora podrá recibir 1/3 de esta información, es decir, 120 líneas de video por minuto. Como se puede observar en la Figura 1.16, cada una de estas líneas de video se divide en dos segmentos; un segmento para cada canal, cada uno con información necesaria para formar la imagen y con una duración de 0.25 segundos.



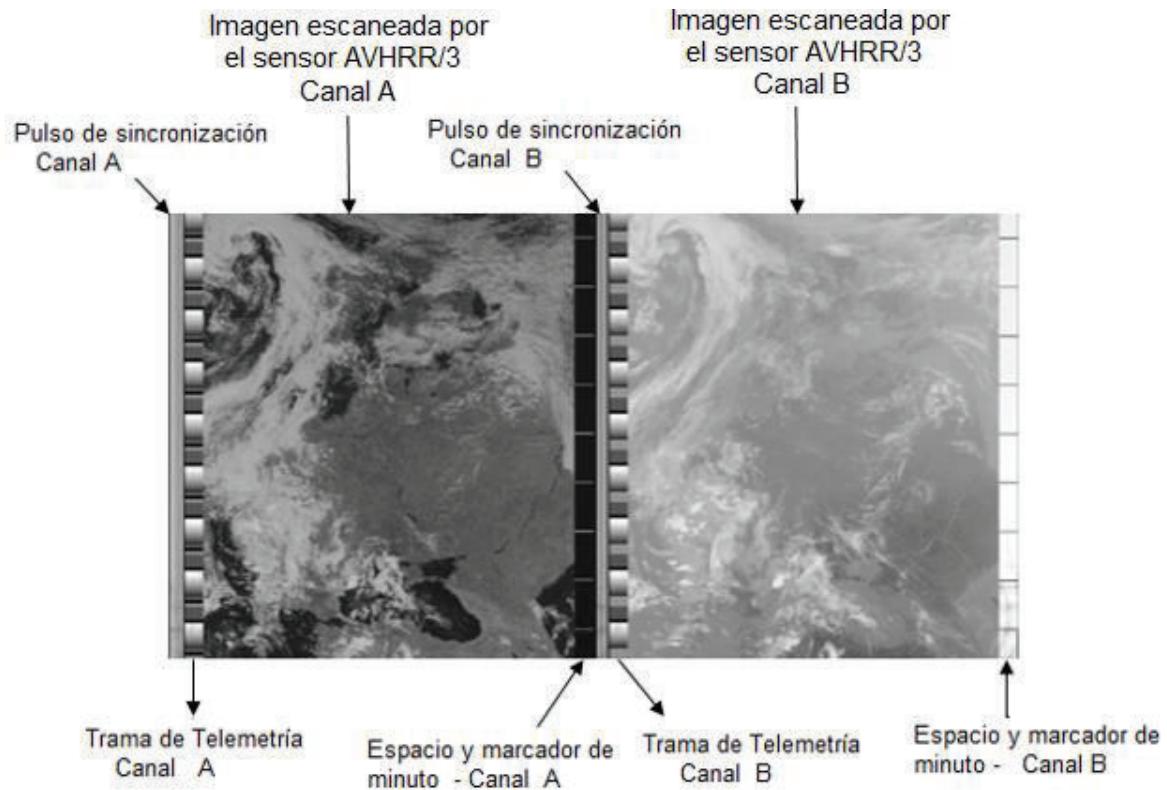
**Figura 1.16** Formato de la trama de una imagen APT [21].

En la Tabla 1.8 se muestra la información que contiene cada segmento para cada canal.

**Tabla 1.8** Información de los segmentos de una imagen APT.

	Canal A	Canal B
<b>Pulsos de sincronización</b>	7 pulsos de secuencia de sincronización a una frecuencia de 1040 Hz (SYNC A).	7 pulsos de secuencia de sincronización a una frecuencia de 832 Hz (SYNC B).
<b>Espacio y marcador de minuto</b>	El espacio hace referencia a una línea negra que se repetirá durante un lapso de 60 segundos y el marcador de minuto es una línea blanca horizontal.	El espacio hace referencia a una línea blanca que se repetirá durante un lapso de 60 segundos y el marcador de minuto es una línea negra horizontal.
<b>Imagen escaneada por el sensor AVHRR/3</b>	Es la imagen propiamente dicha.	Es la imagen propiamente dicha.
<b>Trama de telemetría.</b>	Transmite niveles de modulación y amplitud que forman la escala de grises y también la calibración de la temperatura de las imágenes.	

En la Figura 1.17 se puede apreciar de mejor manera los pulsos de sincronización, el espacio y marcador de minuto y la trama de telemetría.

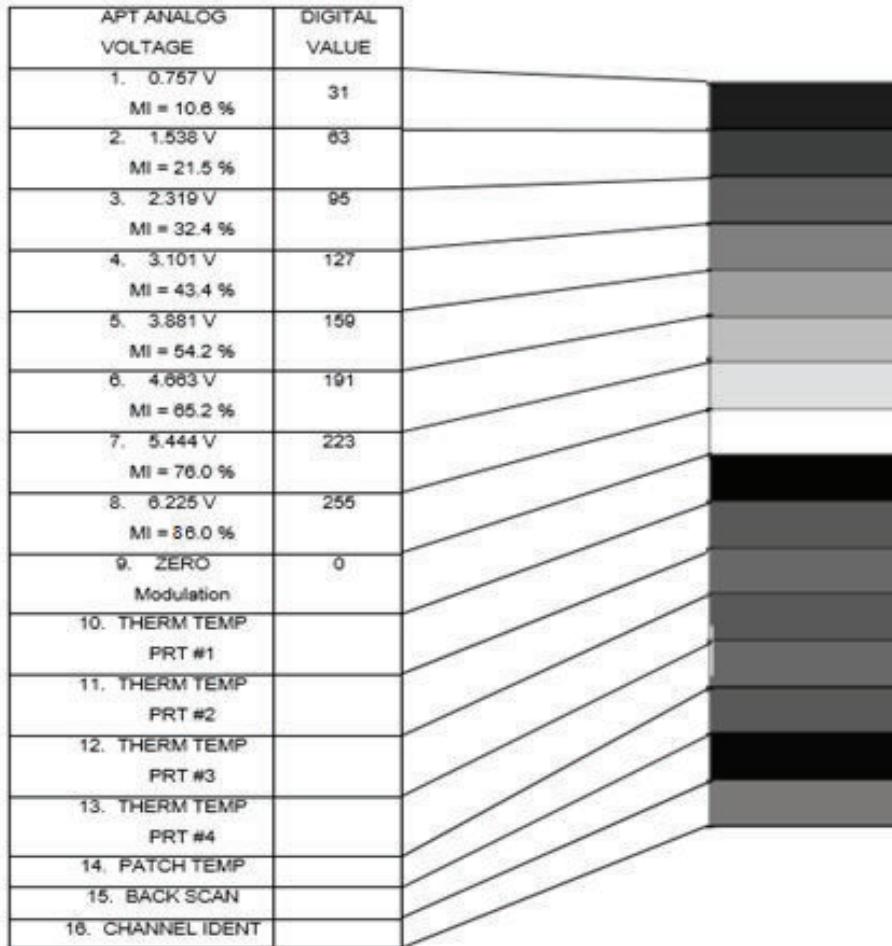


**Figura 1.17** Segmentos de una imagen APT.

Una línea de video completa tiene 2080 pixeles de largo, de los cuales 1818 son utilizados para formar la imagen, ver Figura 1.14, el resto son utilizados para telemetría y sincronización.

Los datos de telemetría, ver Figura 1.16, son de vital importancia, puesto que permiten la correcta calibración tanto de los canales usados para la observación de la Tierra, como de la información que llega hasta las estaciones receptoras. En la Figura 1.18 se muestra la trama de telemetría, la cual consta 16 slots, cada uno de los cuales a su vez se encuentra constituido por ocho líneas de video sucesivas, así también se muestra la escala de grises que se utiliza para las imágenes.

Esta información de telemetría es transmitida constantemente durante toda la órbita del satélite, garantizando que una estación receptora pueda obtener esta información varias veces.



**Figura 1.18** Trama de Telemetría [18].

- *Slots* de 1-8: Son valores de voltaje y porcentajes de índices de modulación asociados a un valor digital.
- *Slot* 9: No contiene ninguna señal o información, representa la señal ZERO de la trama de telemetría.
- *Slots* de la 10-13: Proporcionan información para determinar el valor de radiación de un cuerpo.
- *Slot* 14: Informa el menor valor que puede ser captado por la banda del infrarrojo térmico.
- *Slot* 15: El *back scan* se realiza cuando un espejo que se encuentra a bordo del sensor AVHRR/3 observa al espacio.

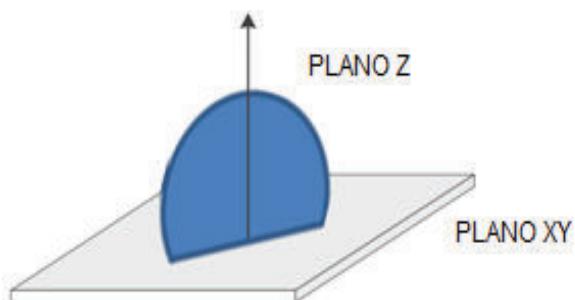
- *Slots* 16: permite identificar cuál de las 6 bandas del sensor AVHRR/3 están siendo utilizadas para la transmisión. En la Tabla 1.9 se muestran los valores que puede tomar el *slot* dependiendo de la banda en uso.

**Tabla 1.9** Valores del *Slot* 16 [18].

Valor	Banda de Uso
1	1
2	2
3	3 A
6	3 B
4	4
5	5

#### 1.4.2 ANTENAS PARA RECEPCIÓN METEOROLÓGICA [18]

Para la recepción de imágenes meteorológicas como las provenientes de los satélites NOAA, es necesario una antena que tenga un área de captura muy grande, es decir, que permita sin la necesidad de moverla poder recibir la señal del satélite sobre ella. La antena debe presentar un área de captura como la que se muestra en la Figura 1.19, permitiendo la recepción en todos los puntos del plano z homogéneamente mientras el satélite se encuentre visible en el horizonte.



**Figura 1.19** Área de captura para un satélite NOAA [18].

Adicionalmente al área de captura, para una correcta recepción de la señal proveniente de los satélites se debe tener en cuenta las siguientes consideraciones:

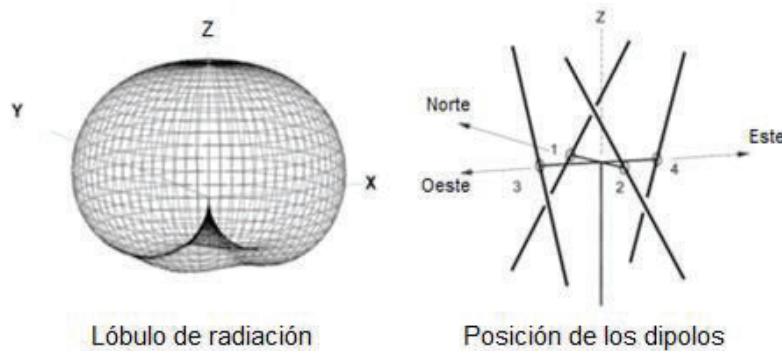
- Las dimensiones de la antena vienen dadas por la frecuencia, en el caso de los satélites NOAA la banda de operación es VHF<sup>9</sup>.
- Debe tener el mismo tipo de polarización que presenta la señal del satélite, en el caso de los satélites NOAA debe ser polarización circular derecha.
- Debe tener una ganancia que permita una recepción libre de ruido.

Luego de considerar la frecuencia de operación y la polarización para el formato APT existen disponibles [18], algunos diseños de antenas omnidireccionales y de baja ganancia que pueden ser construidas con conocimientos básicos de radiofrecuencia y a un bajo costo.

A continuación se muestran algunos modelos de antenas para captar la señal en formato APT proveniente de los satélites NOAA.

#### 1.4.2.1 Antena Double Cross (DCA)

La antena doublé cross está constituida por dos antenas de dipolos cruzados, que al ser alimentados correctamente logran obtener un diagrama de radiación omnidireccional, el cual resulta óptimo para la recepción de las señales provenientes de los satélites NOAA, tal como se puede observar en la Figura 1.20.



**Figura 1.20** Diagrama de radiación y posición de los dipolos en la antena DCA [18].

---

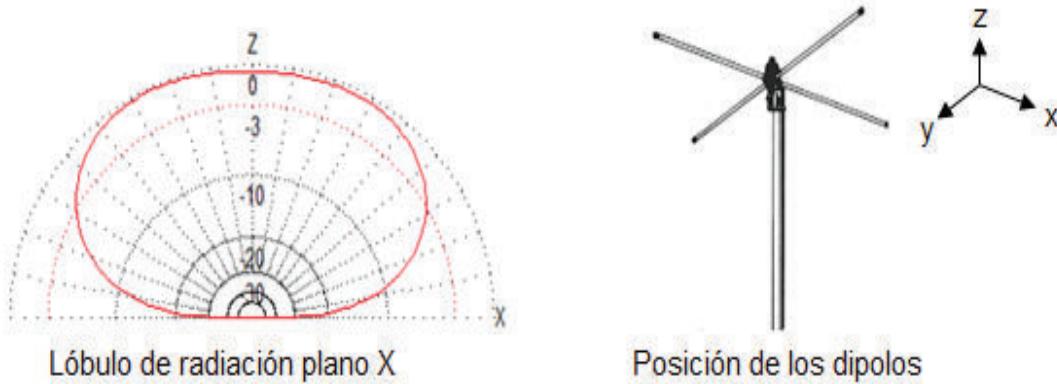
<sup>9</sup> VHF: Banda Very High Frequency del espectro electromagnético comprendida entre 30-300 MHz

Dado que el funcionamiento de esta antena está ligado al dipolo simple de longitud de media onda, cada uno de los dipolos cruzados deben tener una longitud de media onda, la cual es calculada en base a la frecuencia en la que transmiten los satélites NOAA. Para conseguir el requisito de polarización es necesario que dos de los dipolos se encuentren desfasados en 90 grados con respecto a la otra pareja, este desfasaje se consigue haciendo que una de las parejas esté conectada a una línea de un cuarto de longitud de onda mayor que la de los otros dos dipolos.

La construcción de la antena puede realizársela con varillas metálicas y tubería PVC [18].

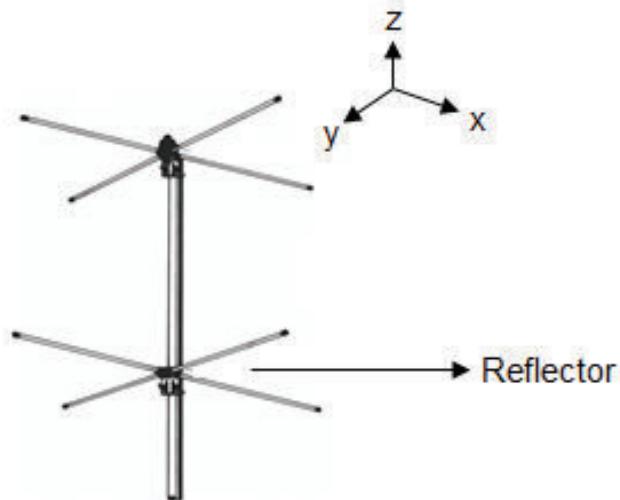
#### 1.4.2.2 Antena Turnstile

Es una de las antenas más simples y menos costosas que se pueden utilizar para la recepción de señales en formato APT. Presenta un diagrama de radiación omnidireccional y para su construcción se necesitan dos dipolos de una longitud de media onda, la cual, al igual que la antena anterior es calculada en base a la frecuencia a la que transmiten los satélites NOAA, ver Figura 1.21.



**Figura 1.21** Antena turnstile [18].

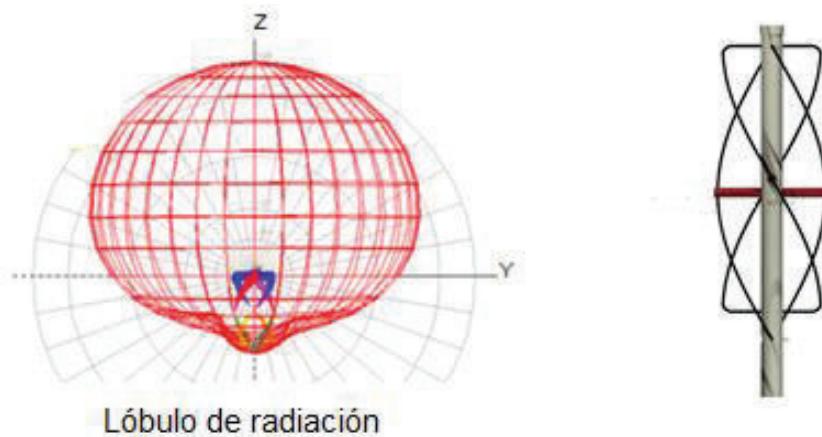
Su principal inconveniente es su poca ganancia, por lo que en ocasiones se suele colocar un reflector, como se observa en la Figura 1.22.



**Figura 1.22** Antena turnstile con reflector [18].

#### 1.4.2.3 Antena Cuadrifilar Helicoidal (QHA)

Es un tipo especial de antena omnidireccional, que cuenta con un mejor diagrama de radiación que la antena turnstile. Se encuentra constituida de dos anillos, los cuales forman una especie de cilindro, tal como se muestra en la parte derecha de la Figura 1.23.



**Figura 1.23** Antena QHA [18].

Para obtener la polarización circular las corrientes que circulan por cada uno de los anillos deben estar desfasadas 90 grados, para lo cual se tiene dos opciones, la primera es utilizar un desfasador y la segunda es utilizar anillos de diferente longitud.

En la Figura 1.23, para lograr el desfase se utiliza la segunda opción, logrando así que el anillo largo reaccione inductivamente y el anillo corto capacitivamente.

Su construcción es ligeramente más complicada que las antenas anteriormente mencionadas y puede construirse con tubo de cobre o cable coaxial, teniendo en cuenta siempre que la conexión debe ser simétrica.

En la Tabla 1.10 se muestra un resumen de algunas de las características de las antenas mencionadas para recepción meteorológica.

**Tabla 1.10** Características de las antenas para recepción meteorológica [18], [24], [25].

	<b>Antena Double Cross</b>	<b>Antena Turnstile</b>	<b>Antena Cuadrifilar Helicoidal</b>
<b>Ganancia</b>	Baja	Muy baja	Baja
<b>Diagrama de radiación</b>	Omnidireccional en el plano Z	Omnidireccional en el plano Z	Omnidireccional en el plano Z
<b>Polarización</b>	Circular	Circular	Circular
<b>Frecuencia de operación</b>	137 – 138 MHz	137 – 138 MHz	137 – 138 MHz
<b>VSWR [valor típico]</b>	1.7	1.4	1.5

## 1.5 RADIO DEFINIDO POR SOFTWARE (RDS) [26], [27], [28]

Uno de los principales problemas en los sistemas de comunicación inalámbricas en cierto modo es el hecho de llevar embebido plataformas de hardware estáticas, las cuales son diseñadas para realizar una o algunas funciones específicas, lo que implica una inversión adicional y complejidad cuando se requieran nuevas funcionalidades, nuevos sistemas o nuevas tecnologías.

En la actualidad, la demanda de dispositivos inalámbricos han crecido exponencialmente, tanto en cantidad debido a la gran demanda que existe por parte de los usuarios, como en complejidad debido a las nuevas aplicaciones y sistemas de modulación, de ahí la necesidad de contar con una tecnología más flexible a la hora de poder realizar el intercambio de información inalámbrica.

La tecnología Radio Definido por Software, SDR por sus siglas en inglés *Software Defined Radio*, permite tener esta flexibilidad a la hora de realizar un sistema de radiocomunicación, ya que permite implementar componentes como mezcladores, amplificadores, transmisores, receptores, etc. en software en lugar de hardware, permitiendo que los parámetros de estos componentes puedan ser configurables dinámicamente y por consiguiente no exista la necesidad de cambiar el hardware para cada sistema de radiocomunicación que se desee implementar.

La tecnología RDS no cuenta con un estándar oficial, pero existe el grupo de trabajo llamado *Wireless Innovation Forum* [26], el cual promueve el desarrollo de estándares, especificaciones y la divulgación de la tecnología RDS para diferentes aplicaciones del tipo académico, militar y comercial.

Para establecer una definición de la tecnología RDS coherente y con una visión clara de sus beneficios asociados el *Wireless Innovation Forum* en colaboración con el *Institute of Electrical and Electronic Engineers (IEEE) P1900.1 group* han establecido la siguiente definición:

*“Radio en el que parte o la totalidad de las funciones de la capa física son definidas por software.”*

### **1.5.1 HISTORIA DE RDS [27], [29]**

El origen de la tecnología de Radio Definido por Software está ligada al ámbito militar, más precisamente al proyecto SPEAKEasy, el cual fue desarrollado por el Departamento de Defensa de los Estados Unidos a inicios de la década de los 90. Este proyecto en un principio operaba en un rango entre los 2 MHz y 200 MHz y albergaba 10 tipos de tecnologías de comunicaciones inalámbricas.

Para el año de 1996 Joseph Mitola funda el *SDR Forum*, hoy en día conocido como *Wireless Innovation Forum*, que junto con otras empresas como la NASA, Motorola Solutions, MIT Lincoln Laboratory, se encargaron de introducir al mercado la tecnología de Radio Definido por Software y Radio Cognitiva<sup>10</sup>.

A mediados de 1997 el ejército americano a fin de terminar con la problemática de interoperabilidad que existía entre los diferentes sistemas de radio creó un programa para el desarrollo de sistemas de radiocomunicación, los cuales fueron creados para ser reconfigurables vía software, este programa fue denominado *Joint Tactical Radio System*, permitiendo especialmente a los aviones de combate tener la capacidad de realizar una transmisión convergente de voz, datos y video.

En el año 2000 empiezan los primeros proyectos de origen no militar orientados a radioaficionados, e inician foros donde se informa de las pruebas realizadas, nuevas funciones encontradas y nuevas problemáticas a las que se trata de dar solución. Paralelamente comienzan proyectos GNU<sup>11</sup> para implementar sistemas RDS, el más conocido de ellos es la herramienta de software libre llamada GNU Radio.

Los avances en la microelectrónica y en plataformas de desarrollo para RDS lograron que para el año 2007 el mercado de SDR sobrepase los 5.3 billones de dólares. Uno de los proyectos más ambiciosos fue el OpenBTS, el cual consiste en implementar una estación base de telefonía GSM en software, apoyándose con Asterisk<sup>12</sup> para el servicio de VoIP [26].

Actualmente la adaptabilidad que presenta la Radio Definida por Software a diversos escenarios, sin la necesidad de alterar el hardware del equipo, lo hace idónea para la realización de investigaciones y estudios de las comunicaciones inalámbricas.

---

<sup>10</sup> Radio Cognitiva, es un paradigma de las comunicaciones inalámbricas en la cual tanto las redes como los mismos nodos inalámbricos cambian los parámetros particulares de transmisión o de recepción para ejecutar su contenido de forma eficiente sin inferir con los usuarios autorizados.

<sup>11</sup> GNU es un acrónimo recursivo de “GNU no es Unix”.

<sup>12</sup> Asterisk es un programa bajo licencia GPL que proporciona funcionalidades de una central telefónica (PBX).

### 1.5.2 ARQUITECTURA DEL RDS [26], [29], [30]

El esquema básico de la tecnología RDS se muestra en la Figura 1.24, el cual consta de tres bloques funcionales: sección de RF (Radiofrecuencia), sección de IF (Frecuencia Intermedia) y sección de Banda base.

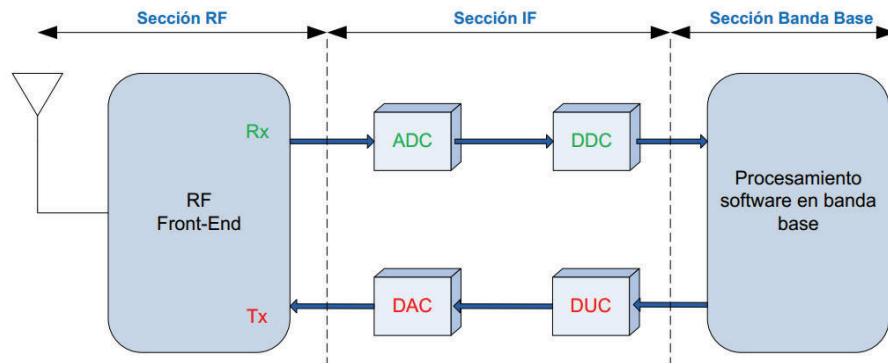


Figura 1.24 Diagrama de bloques RDS [26].

La sección RF es la encargada de trasmitir y recibir las señales de radiofrecuencia. En el caso de recepción convertir la señal en frecuencia intermedia y en el caso de transmisión modular y amplificar las señales de frecuencia intermedia.

En la sección IF se da lugar el concepto de *Zero-IF*, es decir, que la señal en RF es directamente convertida a la señal en banda base o viceversa. Adicionalmente, se realiza la conversión analógico-digital (Recepción) y digital-analógico (Transmisión), para lo cual, se hace uso de los módulos ADC (*Analog to Digital Converter*) y DAC (*Digital to Analog Converter*).

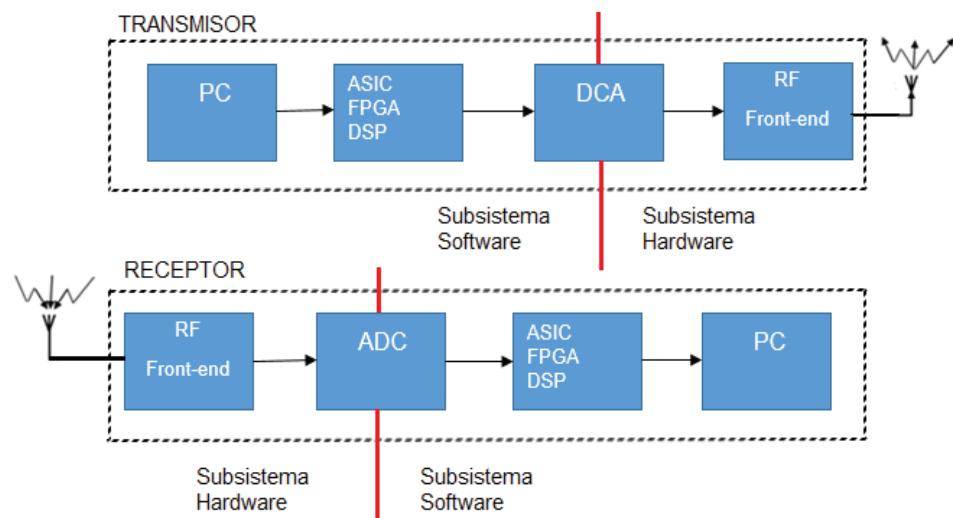
El convertidor análogo-digital (ADC), como su nombre lo indica es el encargado de realizar el paso de señales analógicas a digitales. Para poder reconstruir fielmente la señal analógica a la salida del ADC, un parámetro crítico en el convertidor es la elección de la frecuencia de muestreo, la cual, según la teoría de Nyquist debe ser igual o mayor al doble del ancho de banda de la señal en banda base.

El convertidor digital-análogo (DAC) presenta características similares al ADC, siendo capaz de recibir información digital en forma de una palabra de  $n$  bits y transformándola a una señal analógica en un determinado tiempo de muestreo.

Los módulos DDC (*Digital Down Converter*) y DUC (*Digital Up Converter*) son los encargados de bajar y subir respectivamente la tasa de muestreo entre las secciones de IF y banda base.

Por último, la sección de banda base que es la encargada del procesamiento como modulación/demodulación, análisis espectral de la señal, etc. En esta sección la señal es totalmente digital, por lo que el procesamiento se lo realiza solo a nivel de Software, y es esta la característica que permiten a los dispositivos RDS tener una flexibilidad a la hora de implementar sistemas de radiocomunicación.

En ocasiones los requerimientos de hardware exceden las especificaciones de los procesadores de propósito general que son utilizados en los RDS, en especial en aplicaciones de alta frecuencia y esquemas de modulación complejos, por lo que para minimizar los problemas que conlleva realizar las operaciones que tiene que soportar la unidad de procesamiento, se utiliza una mezcla apropiada de circuitos especializados (ASIC – *Application Specific Integrated Circuits*), Procesadores Digitales de Señales (DSP – *Digital Signal Processor*), Microprocesadores de propósito general (GPP – *General Purpose Register*) y Arreglos de Compuertas Programables en Campo (FPGA – *Field Programmable Gate Array*). El diagrama de bloques de un transmisor y un receptor RDS con FPGA, DSP y ASIC se muestra en la Figura 1.25.



**Figura 1.25** Tx y Rx RDS con FPGA, DSP, ASIC [29].

## 1.6 HARDWARE EMPLEADO EN EL PROTOTIPO

### 1.6.1 RTL-SDR [31]

El módulo RTL-SDR que se utilizará para la realización del presente proyecto se muestra en la Figura 1.26. Este módulo consta principalmente de dos chips, el demodulador RTL2832U y el Sintonizador R820T.



**Figura 1.26** Módulo RTL-SDR.

El módulo RTL-SDR de la Figura 1.26, fue seleccionado por el amplio rango de frecuencias de operación que presenta y principalmente por el bajo costo frente a otros módulos RDS, tal como se puede observar en la Tabla 1.11.

**Tabla 1.11** Precios de dispositivos RDS [31], [32].

Dispositivo	Frecuencia de operación [MHz]	Precios [USD]
<b>RTL-SDR</b>	24 – 1766	20
<b>Airspy</b>	24 – 1800	100
<b>SDPlay</b>	0.1 – 2000	149
<b>HackRF</b>	30 – 6000	300
<b>USRP B210</b>	70 – 6000	1100

Entre otras características del módulo RTL-SDR se tienen las siguientes:

- La entrada para la antena es de tipo SMA hembra, con una impedancia de 50 ohmios y una pérdida menor a los 0.2 dB.
- La máxima frecuencia de muestreo que presenta es 3.2 MHz, aunque se recomienda que sea 2.8 MHz.
- Tiene una resolución de 8 bits.
- Puede conectarse a cualquier tipo de ordenador, gracias a su puerto USB 2.0.

El módulo RTL-SDR al poder barrer un amplio rango de frecuencias se lo utiliza en varios tipos de aplicaciones, entre ellas se pueden mencionar:

- Escuchar conversaciones de control de tráfico de aeronaves.
- Escuchar radio aficionado.
- Recepción y decodificación de señales GPS.
- Escuchar radio FM.
- Triangulación de la fuente de una señal.
- Búsqueda de fuentes de ruido de RF.
- Recepción de señales de sensores inalámbricos.
- Recepción de señales GSM.
- Decodificación de transmisiones de voz digital cifrada.
- Escuchar conversaciones de la policía y ambulancias.

#### 1.6.1.1 Sintonizador R820T [27]

Es un chip sintonizador híbrido de silicio creado por la empresa Rafael Microelectronics, Inc. En principio fue creado para la TV digital, por lo que soporta estándares como el ATSC (*Advanced Television Systems Committee*), DTMB (*Digital Terrestrial Multimedia Broadcast*) e ISDB-T (*Integrated Services Digital Broadcasting*).

Actualmente, su campo de desarrollo se ha ido incrementando considerablemente, debido al rendimiento superior que ofrece en linealidad, sensibilidad, inmunidad de canal adyacente, etc.

En la Tabla 1.12 se muestran las principales características del chip sintonizador.

**Tabla 1.12** Características del chip R820T [33].

Parámetro	Valor Típico	Unidad
Voltaje de entrada	3.3	V
Consumo de corriente	<178	mA
Temperatura de operación	25	°C
Rango de Frecuencia	42-1002	MHz
Figura de Ruido	3.5	dB
Potencia de entrada máxima	+10	dBm
Rechazo de imagen	65	dBc <sup>13</sup>

#### 1.6.1.2 Demodulador RTL2832U [27]

Es un chip creado por la empresa Realtek Semiconductor Corp. este chip posee algoritmos propios y se vende como una solución para un amplio rango de aplicaciones. En las características técnicas se indica que es un demodulador COFDM<sup>14</sup> para la televisión digital terrestre, incluye soporte para radio FM, DAB y es la pieza clave que hace que un receptor de televisión pase a convertirse en un receptor RDS.

Presenta las siguientes características:

- Soporta múltiples frecuencias intermedias.
- Soporta entrada de Zero-IF.
- Su señal de reloj es generada por un cristal de bajo costo.
- Presenta detección de intervalos de guarda.
- Tiene circuitos de cancelación de ruido de impulsos.
- Recuperación de portadora automática con un amplio rango de *offset* ( $\pm 800$  KHz).
- Contiene filtros de hardware MPEG-2 PID.

<sup>13</sup> dBc, se usa para expresar el nivel de potencia de una señal con respecto a la portadora.

<sup>14</sup> COFDM: Coded Orthogonal Frequency Division Multiplexing, es una técnica de modulación de banda ancha utilizada para transmitir información digital a través de un canal de comunicaciones.

- Circuito de rechazo de interferencia co-canal e interferencia adyacente.
- Alimentación externa de 3.3 Voltios.

En el Anexo D (digital) se adjuntan los *datasheets* del chip demodulador RTL2832U y del chip sintonizador R820T.

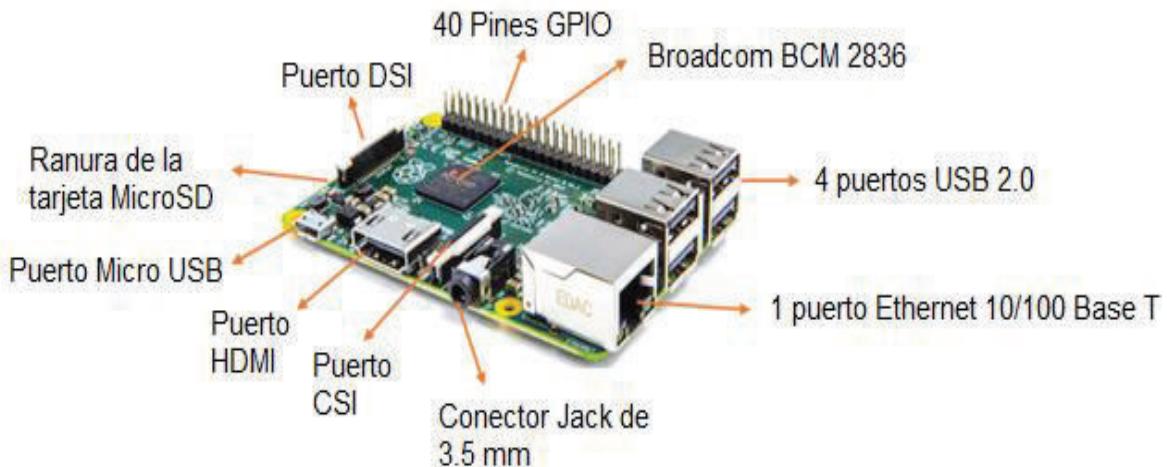
### 1.6.2 RASPBERRY PI [34], [35]

La placa Raspberry Pi es una microcomputadora de arquitectura ARM (*Advanced RISC Machine*) de 32 bits, que puede trabajar con aplicaciones donde el bajo consumo sea la prioridad y ha sido precisamente esta característica la que actualmente le ha permitido a la placa Raspberry Pi posicionarse en el mercado.

La mayoría de computadoras de escritorio o laptops utilizan procesadores AMD o Intel y estos utilizan un conjunto de instrucciones x86, por lo que el *software* utilizado por estos procesadores no es compatible con la Raspberry Pi, de ahí que se han creado y se han adaptado sistemas operativos y aplicaciones para su funcionamiento con el hardware disponible de la Raspberry Pi. Los principales sistemas operativos disponibles para la Raspberry Pi son los siguientes:

- Raspbian
- Ubuntu Mate
- Snappy Ubuntu Core
- Windows 10
- OpenELEC
- RISC OS
- Arch Linux
- Pidora
- Xbian
- NetBSD

Para la realización del presente proyecto, se utilizará la placa Raspberry Pi 2 modelo B, el cual se observa en la Figura 1.27 y presenta las siguientes características físicas y técnicas:



**Figura 1.27** Placa Raspberry Pi 2 modelo B [34].

- Un procesador Broadcom BMC2836 ARM7 Quad Core a 900 MHz, es el encargado de realizar el procesamiento gráfico, comunicación con las interfaces, procesamiento de audio y video, etc.
- 1 GB de memoria RAM
- 4 puertos USB 2.0, para la conexión de diferentes dispositivos como teclados, mouse, dispositivos de almacenamiento, etc.
- 1 puerto Ethernet 10/100 Mbps para conexión a Internet.
- 1 ranura para tarjetas micro SD. En la tarjeta micro SD se almacenan los datos, aplicaciones y el sistema operativo. Se recomienda que la capacidad mínima de la tarjeta micro SD sea de 4 GB [34].
- 1 puerto HDMI para la salida de audio y video a 1080 p.
- 1 conector Jack de 3.5mm para salida de audio.
- 1 puerto CSI (*Camera Serial Interface*) para conectar una cámara.
- 1 puerto DSI (*Display Serial Interface*) para conectar una pantalla táctil.
- 1 conector micro USB, que es utilizado como fuente de alimentación para el Raspberry PI. La fuente de alimentación utilizada debe proporcionar mínimo un voltaje de 5 Voltios y una corriente de 1.8 Amperios.
- 40 pines GPIO (General Purpose Input/Output), los cuales pueden ser utilizados como conexiones de entrada y salida de información.

## 1.7 SOFTWARE EMPLEADO EN EL PROTOTIPO

Para la realización del presente proyecto se utilizará como sistema operativo de la placa Raspberry Pi una distribución de Linux llamada Raspbian, como programa principal la aplicación GNU Radio Companion (GRC) y como programa adicional WXtoImg [36].

En cuanto a la aplicación Web, esta será implementada sobre un servidor LAMP<sup>15</sup>.

### 1.7.1 RASPBIAN [34], [37]

Raspbian es un sistema operativo basado en Debian, fue creado y diseñado para la placa reducida Raspberry Pi en el año 2012, pese a no ser el único sistema operativo disponible, desde su lanzamiento se convirtió en una de los más utilizadas y populares gracias a la enorme comunidad de usuarios y a los alrededor de 35.000 paquetes disponibles, optimizados y listos para usarse desde un principio en la Raspberry Pi, de igual manera es considerado por la Raspberry Pi Foundation como el paquete más completo y el que mejor rendimiento ofrece.

En el presente proyecto se utilizará la versión “Jessie”, la cual es la versión 8.0 del sistema operativo Raspbian. Esta versión de sistema operativo presenta un mejor rendimiento y flexibilidad respecto a otras versiones, teniendo como principales aplicaciones LibreOffice y Claws Mail y nuevos entornos de programación como BlueJ y Greenfoot. El sistema operativo Raspbian “Jessie” puede descargarse de la página oficial de la Raspberry Pi [34].

### 1.7.2 GNU RADIO [26], [38]

GNU Radio es un conjunto de diferentes archivos y aplicaciones agrupados en librerías para el procesamiento digital de señales, es un software de libre distribución licenciado bajo la GNU General Public Licence (GPL) versión 3<sup>16</sup> y corre sobre sistemas operativos GNU/Linux, pero también es posible su instalación en MAC y Windows.

---

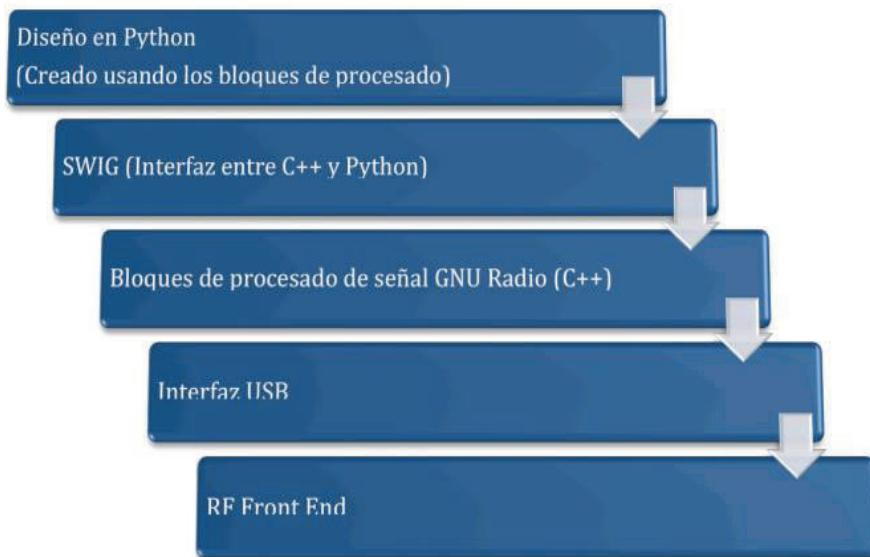
<sup>15</sup> LAMP, acrónimo para describir las tecnologías que definen la infraestructura de un servidor web.

<sup>16</sup> GPL versión 3, es una licencia que garantiza a los usuarios finales la libertad de usar, estudiar, compartir y modificar el software.

GNU Radio utiliza como lenguaje de programación Python, el cual, es un lenguaje orientado a objetos interpretado, es decir, el sistema operativo lo ejecuta directamente.

Para construir un sistema de radio se debe crear un grafo, donde los bloques utilizados representan los nodos y sus conexiones representan el flujo de datos.

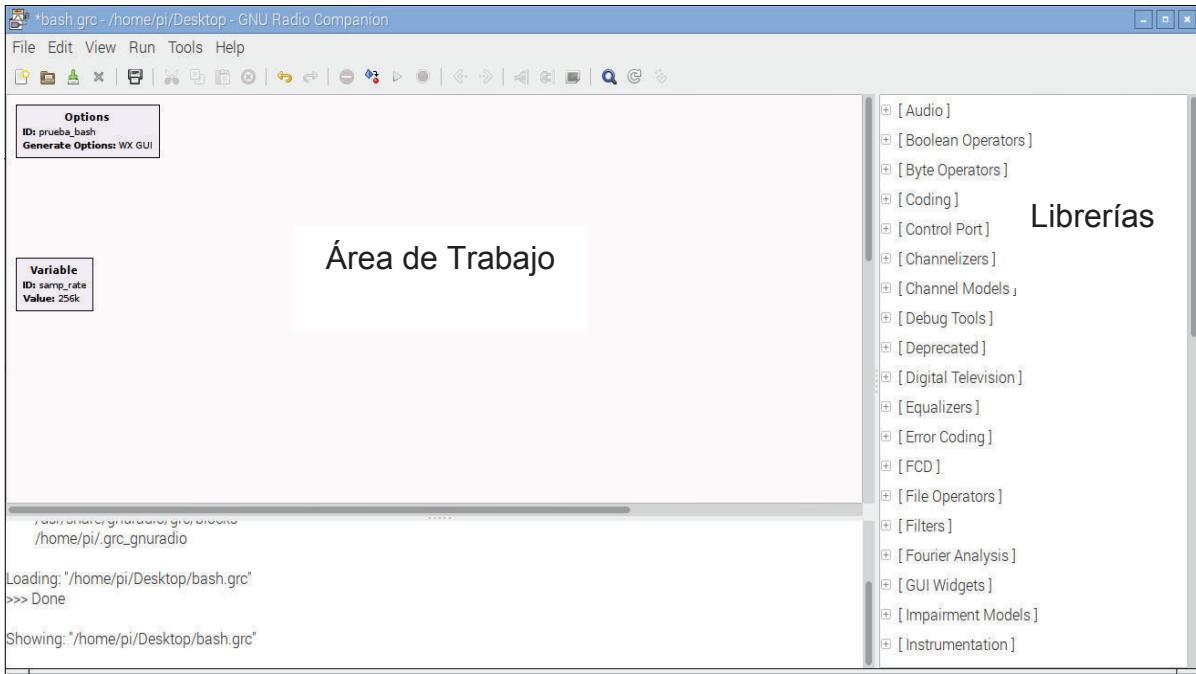
En la Figura 1.28 se observa la interacción que existe entre los diferentes niveles de GNU Radio, donde los bloques utilizados son implementados en C++ y portados a Python a través del programa SWIG (*Simplified Wrapper and Interface Generator*), para luego pasar a la interfaz USB y por ultimo a la sección de radiofrecuencia externa.



**Figura 1.28** Niveles de GNU Radio [26].

En el presente proyecto se utilizará la aplicación GNU Radio Companion (GRC), el cual, es el entorno gráfico de GNU Radio. Esta aplicación es una herramienta gráfica basada en bloques que permite crear gráficos de flujos de señal y generar códigos fuentes del flujo gráfico. Su programación es muy similar a la de Simulink en Matlab™.

En la Figura 1.29 se observa el entorno gráfico de la aplicación GRC.



**Figura 1.29** Pantalla principal de GRC.

La ventaja principal del manejo de bloques es la rápida implementación de los sistemas. El flujo de datos que procesa un bloque desde un puerto de entrada hasta un puerto de salida se realiza de manera continua y pueden ser real, flotante o complejo.

Actualmente este software es usado por investigadores, estudiantes y aficionados, ya que al ser usado conjuntamente con hardware externo de radiofrecuencia, permite la simulación de diferentes sistemas de comunicación inalámbricos.

### 1.7.3 WXTOIMG [36]

Es un software que permite la grabación y decodificación de las señales procedentes de satélites meteorológicos así como la edición y visualización de las imágenes.

Las señales admitidas por el software pueden estar en formato APT o en formato WEFAX<sup>17</sup>, las cuales provienen de satélites de órbita polar y órbita geoestacionaria respectivamente. El software WXtolmg dispone de diversas facilidades de procesado

---

<sup>17</sup> WEFAX, transmisión de imágenes en forma analógica por parte de sistemas meteorológicos geoestacionarios, es un formato derivado del APT.

de imágenes, pero también puede entregar imágenes en formato raw<sup>18</sup> para su posterior procesado. Los formatos en los que puede obtener las imágenes son: JPEG, PNG, BMP y PBM.

Para que el *software* WXtolmg realice la decodificación de la señal en formato APT necesita que la señal modulada en amplitud (AM), se encuentre muestreada a 11025 Hz, el cual es un valor que se encuentra configurado por defecto en el *software*. Puesto que la imagen que se obtiene es de luminancia, para obtener la imagen el *software* WXtolmg asigna un color dentro de una escala de grises en base al índice de modulación que presenta la señal.

Para la realización del presente proyecto se utilizará una versión de WXtolmg diseñada especialmente para la arquitectura y *hardware* disponible en la Raspberry Pi y se la puede descargar de la página oficial [www.wxtoimg.com/download](http://www.wxtoimg.com/download).

#### **1.7.4 SERVIDOR LAMP [39]**

Un servidor LAMP es un conjunto de paquetes de *software* que contiene un ambiente de desarrollo web completo, que facilita el desarrollo e implementación de aplicaciones web de una manera rápida, segura y eficiente.

El acrónimo LAMP hace referencia a los diferentes módulos que lo componen, los cuales son los siguientes:

- a) Sistema operativo Linux.
- b) Servidor Web Apache.
- c) Base de Datos MySQL o MariaDB.
- d) Lenguaje de Programación PHP, Perl o Phyton.

En el presente proyecto como se mencionó en la sección 1.7.1, el sistema operativo que se utilizará es Raspbian “Jessie”, los demás módulos que se utilizarán para el servidor LAMP son los siguientes:

- a) Servidor Web Apache.

---

<sup>18</sup> raw: es un formato de archivo digital de imágenes que contiene la totalidad de los datos de la imagen tal y como ha sido captada.

- b) Base de datos MySQL.
- c) Lenguaje de Programación PHP.

Adicionalmente, se utilizará la herramienta phpMyAdmin para la administración de la base de datos.

#### **1.7.4.1 Apache [40]**

Apache es un servidor web de código abierto y uno de los más populares y utilizados a nivel mundial, debido a su gran estabilidad y seguridad, tanto en páginas estáticas como en dinámicas. Entre sus principales características se tienen las siguientes:

- Utiliza el protocolo HTTP<sup>19</sup>.
- Es un servidor web multiplataforma.
- Es un software Open Source y con licencia GPL.
- Puede dar soporte a diferentes lenguajes de programación.
- Tiene soporte de seguridad SSL/TLS<sup>20</sup>.

Actualmente, Apache sigue siendo desarrollado por la comunidad que trabaja bajo la tutela de la *Apache Software Foundation*.

#### **1.7.4.2 MySQL [41]**

Es un sistema de gestión de base de datos relacional, multihilo y multiusuario. Puede adaptarse a diferentes entornos de desarrollo, permitiéndole interactuar con diferentes lenguajes de programación como PHP, Java, entre otros. Entre sus principales características se tienen las siguientes:

- Escrito en C y C++.
- Es un software Open Source.
- Su configuración y administración es relativamente simple.
- Utiliza un lenguaje de consulta estructurado.

---

<sup>19</sup> HTTP: Protocolo de Transferencia de HyperTexto.

<sup>20</sup> SSL (Secure Socket Layer) y TLS (Transport Layer Security) son protocolos utilizados para establecer comunicaciones seguras a través de Internet.

#### 1.7.4.2.1 *pphpMyAdmin*

Es una herramienta Web utilizada para manejar y administrar una base de datos MySQL. Se encuentra disponible bajo una licencia GPL y corre bajo un lenguaje de programación PHP.

Con esta herramienta se puede crear o eliminar base de datos, crear, eliminar o modificar tablas, ejecutar sentencias SQL, exportar datos en varios formatos, etc.

#### 1.7.4.3 PHP [41]

Es un lenguaje de programación utilizado para la creación de aplicaciones o contenido dinámico para sitios web. Este lenguaje es de código abierto interpretado, de alto nivel, embebido en páginas HTML<sup>21</sup> y ejecutado en el servidor.

Entre sus principales características se tienen las siguientes:

- Alto rendimiento.
- Permite la conexión con diferentes servidores de base de datos.
- Facilidad de aprendizaje y uso.
- Es multiplataforma.
- Posee una gran variedad de funciones integradas.

---

<sup>21</sup> HTML: Lenguaje de marcas de hipertexto, es utilizado para la elaboración de páginas web.

## CAPÍTULO 2: DESARROLLO DEL PROTOTIPO

### 2.1 REQUERIMIENTOS DEL PROTOTIPO

Los requisitos que debe cumplir el prototipo acorde a los objetivos planteados para la realización del presente proyecto son los siguientes:

- a) Adquisición de la señal en formato APT transmitida por los satélites NOAA-15,18 y 19.
- b) Decodificación de la señal a un formato de imagen.
- c) Almacenamiento de las imágenes.
- d) Visualización de las imágenes mediante una aplicación web.

Adicionalmente, para el cumplimiento de cada uno de los requisitos antes mencionados, el prototipo necesitará ciertos requerimientos de *software*, los cuales son los siguientes:

- 1) Software RTL-SDR.
- 2) Aplicación GNU Radio Companion.
- 3) Programa WXtoImg.
- 4) Servidor LAMP.

Estos requisitos de *software* correrán sobre el sistema operativo Raspbian “Jessie”, cuya instalación y configuración se detallada en el Anexo A.

Previamente para el cumplimiento de los requisitos a, b, c, y d, es necesario que en el prototipo se encuentren instalados y configurados todos los requisitos de *software*. A continuación se explicará la instalación y configuración de estos requisitos.

#### 2.1.1 INSTALACIÓN DEL SOFTWARE RTL-SDR [42], [43], [44]

Este *software* tiene la mayoría de sus paquetes basados en la librería **librtlsdr**, la que a su vez necesita de la librería **libusb** para poder comunicarse con el dispositivo, así también, el *software* presenta una serie de herramientas propias para línea de

comandos, como **rtl\_test**, **rtl\_tcp**, **rtl\_sdr** que permiten identificar a los dispositivos conectados y realizar funciones básicas de transferencia de información.

Para la instalación y el correcto funcionamiento del módulo RTL-SDR con la Raspberry Pi se tiene que realizar el procedimiento que se menciona a continuación.

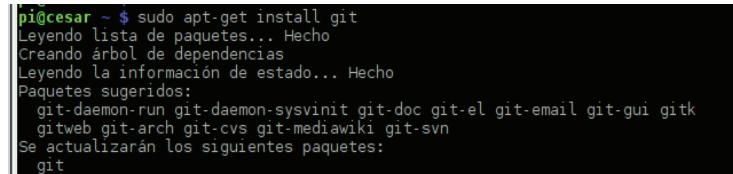
1. Realizar la actualización de todos los paquetes que se encuentran instalados en la Raspberry Pi mediante el comando “`sudo apt-get update`”, tal como se muestra en la Figura 2.1.



```
pi@cesar ~ $ sudo apt-get update
Des:1 http://mirrordirector.raspbian.org jessie InRelease [15,0 kB]
Des:2 http://archive.raspberrypi.org jessie InRelease [13,2 kB]
Des:3 http://mirrordirector.raspbian.org jessie/main armhf Packages [8.963 kB]
Des:4 http://archive.raspberrypi.org jessie/main Sources [48,8 kB]
Des:5 http://archive.raspberrypi.org jessie/ui Sources [5.197 B]
Des:6 http://archive.raspberrypi.org jessie/main armhf Packages [143 kB]
Des:7 http://archive.raspberrypi.org jessie/ui armhf Packages [7.639 B]
Ign http://archive.raspberrypi.org jessie/main Translation-es_ES
```

**Figura 2.1** Actualización de paquetes.

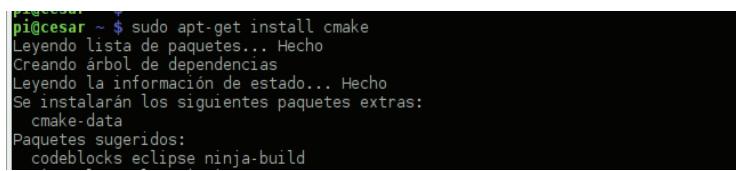
2. Instalar el *software* de control de versiones `git`, tal como se muestra en la Figura 2.2. Este *software* permite trabajar con repositorios remotos.



```
pi@cesar ~ $ sudo apt-get install git
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Paquetes sugeridos:
  git-daemon-run git-daemon-sysvinit git-doc git-el git-email git-gui gitk
  gitweb git-arch git-cvs git-mediawiki git-svn
Se actualizarán los siguientes paquetes:
  git
```

**Figura 2.2** Instalación de `git`.

3. Instalar la herramienta multiplataforma de generación de código, `cmake`, ver Figura 2.3.



```
pi@cesar ~ $ sudo apt-get install cmake
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes extras:
  cmake-data
Paquetes sugeridos:
  codeblocks eclipse ninja-build
```

**Figura 2.3** Instalación de `cmake`.

4. Instalar el paquete para aplicaciones de programación USB, libusb-1.0-0.dev, tal como se muestra en la Figura 2.4.

```
pi@cesar ~ $ sudo apt-get install libusb-1.0-0.dev
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Note, selecting 'libusb-1.0-0-dev' for regex 'libusb-1.0-0.dev'
Se instalarán los siguientes paquetes extras:
  libusb-1.0-doc
Se instalarán los siguientes paquetes NUEVOS:
  libusb-1.0-0-dev libusb-1.0-doc
```

**Figura 2.4** Instalación libusb-1.0-0.dev.

5. Instalar el paquete “essential” para la creación de paquetes Debian, ver Figura 2.5.

```
pi@cesar ~ $ sudo apt-get install build-essential
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
build-essential ya está en su versión más reciente.
0 actualizados, 0 nuevos se instalarán, 0 para eliminar y 182 no actualizados.
```

**Figura 2.5** Instalación de build-essential.

6. Una vez instalado los prerequisitos, se clona el paquete rtl-sdr de un repositorio git, tal como se muestra en la Figura 2.6. El paquete clonado pertenece al proyecto osmocom (*Open Source Mobile Communications*), proyecto que se encarga de realizar *software* y herramientas de desarrollo para comunicaciones móviles.

```
pi@cesar ~ $ git clone git://git.osmocom.org/rtl-sdr.git
Cloning into 'rtl-sdr'...
remote: Counting objects: 1617, done.
remote: Compressing objects: 100% (711/711), done.
remote: Total 1617 (delta 1182), reused 1213 (delta 898)
Receiving objects: 100% (1617/1617), 345.89 KiB | 67.00 KiB/s, done.
Resolving deltas: 100% (1182/1182), done.
Checking connectivity... done.
```

**Figura 2.6** Clonación del paquete rtl-sdr.

- Una vez terminada la clonación, se cambia de directorio hacia la nueva carpeta rtl-sdr, luego, se crea la carpeta *build* y nuevamente se cambia de directorio hacia esta nueva carpeta creada, tal como se muestra en la Figura 2.7.

```
pi@cesar ~ $ cd rtl-sdr/
pi@cesar ~/rtl-sdr $ mkdir build
pi@cesar ~/rtl-sdr $ cd build/
pi@cesar ~/rtl-sdr/build $
```

**Figura 2.7** Cambio de dirección a la carpeta *build*.

- Se genera y se compila el código del *software*, mediante los comandos “cmake .. /” y “make”, tal como se muestra en la Figura 2.8.

```
pi@cesar ~/rtl-sdr/build $ cmake ../
-- The C compiler identification is GNU 4.9.2
-- Check for working C compiler: /usr/bin/cc
-- Check for working C compiler: /usr/bin/cc -- works
-- Detecting C compiler ABI info
-- Detecting C compiler ABI info - done
-- Build type not specified: defaulting to release.
-- Extracting version information from git describe...
-- Found PkgConfig: /usr/bin/pkg-config (found version "0.28")
-- checking for module 'libusb-1.0'
--   found libusb-1.0, version 1.0.19

pi@cesar ~/rtl-sdr/build $ make
Scanning dependencies of target convenience_static
[ 5%] Building C object src/CMakeFiles/convenience_static.dir/convenience/c
nience.c.o
Linking C static library libconvenience_static.a
[ 5%] Built target convenience_static
Scanning dependencies of target rtlsdr_shared
[10%] Building C object src/CMakeFiles/rtlsdr_shared.dir/librtlsdr.c.o
[15%] Building C object src/CMakeFiles/rtlsdr_shared.dir/tuner_e4k.c.o
[20%] Building C object src/CMakeFiles/rtlsdr_shared.dir/tuner_fc0012.c.o
[25%] Building C object src/CMakeFiles/rtlsdr_shared.dir/tuner_fc0013.c.o
[30%] Building C object src/CMakeFiles/rtlsdr_shared.dir/tuner_fc2580.c.o
[35%] Building C object src/CMakeFiles/rtlsdr_shared.dir/tuner_r82xx.c.o
```

**Figura 2.8** Generación del código del software RTL-SDR.

- Una vez finalizada la compilación del *software* se realiza la instalación mediante el código “sudo make install”, luego, mediante el código “sudo ldconfig” se crean los enlaces y cache necesarios a las bibliotecas compartidas, tal como se muestra en la Figura 2.9.

```
pi@cesar:~/rtl-sdr/build$ sudo make install
[ 5%] Built target convenience_static
[ 35%] Built target rtl_sdr_shared
[ 40%] Built target rtl_adsb
[ 45%] Built target rtl_eeprom
[ 50%] Built target rtl_fm
[ 55%] Built target rtl_power
[ 60%] Built target rtl_sdr
[ 65%] Built target rtl_tcp
[ 70%] Built target rtl_test
[100%] Built target rtl_sdr_static
```

```
pi@cesar:~/rtl-sdr/build$ sudo ldconfig
```

**Figura 2.9** Instalación del software RTL-SDR.

10. Se edita el archivo `/etc/modprobe.d/raspi-blacklist.conf`, añadiéndole tres líneas para habilitar diferentes módulos RTL-SDR, tal como se muestra en la Figura 2.10. Luego, para que estos cambios surjan efecto es necesario reiniciar la Raspberry Pi.

```
GNU nano 2.2.6 Fichero: /etc/modprobe.d/raspi-blacklist.conf
blacklist dvb_usb_rtl28xxu
blacklist rtl2832
blacklist rtl2830
```

**Figura 2.10** Habilitación de dispositivos RTL-SDR.

11. Se conecta el dispositivo RTL-SDR y se escribe en la línea de comandos “`lsusb`”, con la información obtenida se crea el archivo `/etc/udev/rules.d/20.rtlsdr.rules`, donde se escribirán las líneas que se muestran en la Figura 2.11, de esta manera se crea un archivo de reglas `udev`<sup>22</sup> para acceder al dispositivo como un usuario no root.

---

<sup>22</sup> Archivo de reglas udev, es un archivo utilizado por los sistemas operativos Unix para que los dispositivos conectados puedan ser reconocidos y utilizados.

The figure consists of three separate terminal windows. The top window shows the command `lsusb` output, which identifies a Realtek Semiconductor Corp. RTL2838 DVB-T device. The middle window shows the command `sudo nano /etc/udev/rules.d/20.rtlsdr.rules` being run. The bottom window shows the contents of the file `/etc/udev/rules.d/20.rtlsdr.rules`, which contains a udev rule for the RTL-SDR device.

```
pi@cesar ~ $ lsusb
Bus 001 Device 009: ID 0bda:2838 Realtek Semiconductor Corp. RTL2838 DVB-T

pi@cesar ~ $ sudo nano /etc/udev/rules.d/20.rtlsdr.rules

GNU nano 2.2.6    Fichero: /etc/udev/rules.d/20.rtsdr.rules

SUBSYSTEM=="usb", ATTRS{idVendor}=="0bda",
ATTRS{idProduct}=="2838", GROUP="adm", MODE="0666",
SYMLINK+="rtl_sdr"
```

**Figura 2.11** Configuración de la regla udev.

12. Se instala el soporte del RTL-SDR para GNU Radio, tal como se muestra en la Figura 2.12.

The figure shows a terminal window where the command `sudo apt-get install gr-osmosdr` is being run. The output indicates that the package is being installed, with files being read from lists and dependencies being created.

```
pi@cesar ~ $ sudo apt-get install gr-osmosdr
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
```

**Figura 2.12** Instalación de gr-osmosdr.

13. Finalmente para comprobar que la instalación se realizó correctamente y que el dispositivo RTL-SDR está listo para ser utilizado, se escribe en la línea de comandos “`sudo rtl_test -t`”, tal como se muestra en la Figura 2.13.

The figure shows a terminal window where the command `sudo rtl_test -t` is run. The output provides information about the found device, its tuner type, supported gain values, and other relevant parameters.

```
pi@cesar ~ $ sudo rtl_test -t
Found 1 device(s):
  0: Realtek, RTL2838UHIDIR, SN: 00000001

Using device 0: Generic RTL2832U OEM
Found Rafael Micro R820T tuner
Supported gain values (29): 0.0 0.9 1.4 2.7 3.7 7.7 8.7 12.5 14.4 15.7 16.6 19.7
20.7 22.9 25.4 28.0 29.7 32.8 33.8 36.4 37.2 38.6 40.2 42.1 43.4 43.9 44.5 48.0
49.6
[R82XX] PLL not locked!
Sampling at 2048000 S/s.
No E4000 tuner found, aborting.
```

**Figura 2.13** Verificación de la instalación del software RTL-SDR.

En la Figura 2.13 se observa que la instalación se realizó correctamente, mostrando algunas de las características que posee el módulo, entre ellas se muestran los dos componentes principales que son, el chip sintonizador R820T y el chip demodulador

RTL2832U, así también los diferentes valores de ganancia que soporta y la velocidad de muestreo.

### 2.1.2 INSTALACIÓN DE GNU RADIO [38], [42]

El programa principal que se utilizará en el presente proyecto es la aplicación GNU Radio Companion, la cual puede ser instalada mediante un liveCD, archivos pre-compilados y mediante la compilación de archivos fuentes.

En la página oficial del proyecto GNU Radio [38] se recomienda que la instalación se la realice mediante la última opción, es decir, la descarga y ejecución del archivo ***build-gnuradio***, debido a que este archivo es un *script* que asegura la instalación de la última versión disponible, así también, con esta opción se realiza la verificación de que los prerequisitos de la aplicación se encuentren instalados y en el caso de no encontrarlos estos serán instalados.

Debido a que el archivo fuente ***build-gnuradio*** no presenta soporte para la versión del sistema operativo que se utilizará en el presente proyecto, se procedió a seleccionar la segunda opción de instalación, es decir, mediante archivos pre-compilados.

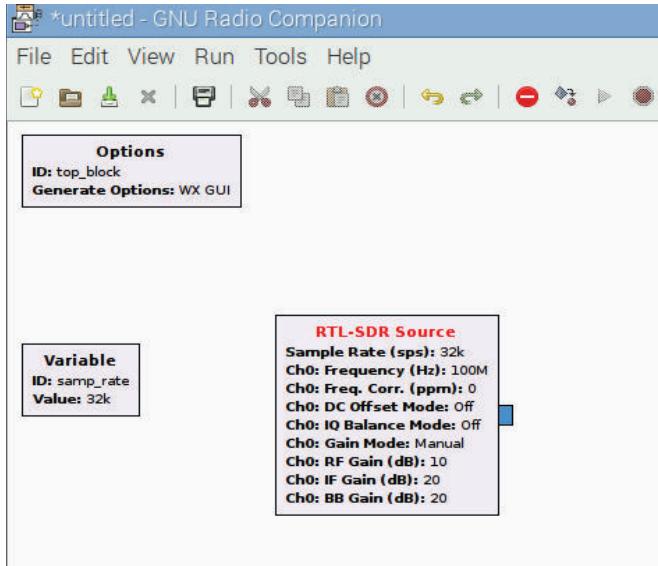
A continuación se describe el procedimiento de instalación.

1. Al igual que la instalación del *software* RTL-SDR el primer paso es la actualización de los paquetes que se encuentran en la Raspberry Pi.
2. Utilizando la línea de comandos, tal como se muestra en la Figura 2.14, se instala la aplicación GNU Radio Companion (GRC).

```
pi@cesar ~ $ sudo apt-get install gnuradio gnuradio-dev
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes extras:
  fonts-liberation fonts-lyx freeglut3 graphviz libboost-serialization1.55.0
  libboost-test1.55.0 libcdt5 libcgraph6 libcomedi0 libdrm-amdgpu1 libdrm-dev
  libdrm-exynos1 libdrm-freedreno1 libdrm-nouveau2 libdrm omap1 libdrm-radeon1
  libdrm-tegra0 libdrm2 libglib1-mesa-dev libglib1-mesa-glx libglade2-0
  libglapi-mesa libglu1-mesa-dev libgnuradio-analog3.7.5 libgnuradio-atsc3.7.5
  libgnuradio-audio3.7.5 libgnuradio-blocks3.7.5 libgnuradio-channels3.7.5
  libgnuradio-comedi3.7.5 libgnuradio-digital3.7.5 libgnuradio-dtv3.7.5
```

**Figura 2.14** Instalación de GNU Radio.

3. Una vez finalizada la instalación, la aplicación está lista para ser utilizada. A manera de verificación de que la aplicación GRC y el soporte de RTL-SDR se instalaron correctamente, en las librerías de GNU Radio se debe encontrar disponible el bloque RTL-SDR Source, tal como se muestra en la Figura 2.15.



**Figura 2.15** GNU Radio Companion.

4. Por último se debe realizar la instalación de la biblioteca OpenGL (*Open Graphics Library*) que permitirá la visualización de las gráficas 2D y 3D generada por la aplicación GRC, tal como se muestra en la Figura 2.16.

```
pi@cesar:~ $ sudo apt-get install libgl1-mesa-swx11
Reading package lists... Done
Building dependency tree... 81%
```

**Figura 2.16** Instalación de libgl1-mesa-swx11.

### 2.1.3 INSTALACIÓN WXTOIMG [36]

Una vez adquirida la señal en formato APT mediante la aplicación GNU Radio Companion, será necesario decodificar dicha señal para así poder visualizar las imágenes, para ello se utilizará el software WXtolmg.

Para la instalación, en primer lugar se tiene que descargar el paquete Linux ARM .deb que sea compatible con la Raspberry Pi, este paquete se puede descargar de la URL <http://www.wxtoimg.com/beta/>. Una vez descargado el paquete se debe realizar el siguiente procedimiento.

1. Se instala el paquete .deb mediante línea de comandos, tal como se muestra en la Figura 2.17.

```
pi@cesar ~/Downloads $ sudo dpkg -i wxtoimg-armhf-2.11.2-beta.deb
Seleccionando el paquete wxtoimg previamente no seleccionado.
(Leyendo la base de datos ... 139852 ficheros o directorios instalados actualmen
te.)
```

**Figura 2.17** Instalación WXtolImg.

2. Una vez finalizada la instalación, para poder abrir el programa se debe escribir el comando “wxtoimg”, tal como se muestra en la Figura 2.18.

```
pi@cesar ~ $ wxtoimg
```

**Figura 2.18** Comando para ejecutar el programa WXtolImg.

3. Ya abierta la aplicación se tendrán que configurar los parámetros de la ciudad y país, en este caso será Quito y Ecuador respectivamente, tal como se muestra en la Figura 2.19.



**Figura 2.19** Configuración inicial WXtolImg.

Una vez finalizado este procedimiento el programa está listo para ser utilizado.

#### 2.1.4 INSTALACIÓN DEL SERVIDOR LAMP [45], [46]

Para el almacenamiento y visualización de las imágenes meteorológicas obtenidas, se utilizarán los módulos que conforman un servidor LAMP, tal como se mencionó en la sección 1.7.4. Cada uno de estos módulos del servidor LAMP serán instalados y correrán sobre la Raspberry Pi.

A continuación se describirá el procedimiento de instalación y configuración inicial del servidor.

1. En primer lugar, actualizar los paquetes que se encuentran en la Raspberry Pi.
2. Se instala el servidor Apache y las librerías necesarias para su funcionamiento, tal como se muestra en la Figura 2.20.

```
pi@cesar ~ $ sudo apt-get install apache2 apache2-doc apache2-utils
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes extras:
  apache2-bin apache2-data libapr1 libaprutil1 libaprutil1-dbd-sqlite3
  libaprutil1-ldap liblua5.1-0 ssl-cert
Paquetes sugeridos:
  apache2-suexec-pristine apache2-suexec-custom openssl-blacklist
Se instalarán los siguientes paquetes NUEVOS:
  apache2 apache2-bin apache2-data apache2-doc apache2-utils libapr1
  libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-ldap liblua5.1-0 ssl-cert
0 actualizados, 11 nuevos se instalarán, 0 para eliminar y 182 no actualizados.
```

**Figura 2.20** Instalación del servidor Apache.

3. Se instalan las librerías de PHP conjuntamente con sus dependencias, tal como se muestra en la Figura 2.21.

```
pi@cesar ~ $ sudo apt-get install libapache2-mod-php5 php5 php-pear php5-xcache
php5-curl php5-gd
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes extras:
  libonig2 libperl4-corelibs-perl libqdbm14 lsof php5-cli php5-common
  php5-json php5-readline
Paquetes sugeridos:
  php5-dev php5-user-cache
Se instalarán los siguientes paquetes NUEVOS:
  libapache2-mod-php5 libonig2 libperl4-corelibs-perl libqdbm14 lsof php-pear
  php5 php5-cli php5-common php5-curl php5-gd php5-json php5-readline
  php5-xcache
```

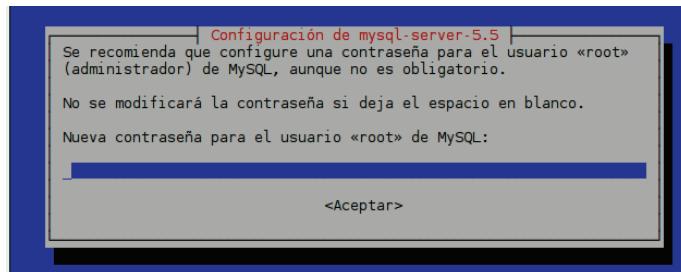
**Figura 2.21** Instalación PHP.

4. Se instala el gestor de base de datos MySQL, conjuntamente con las librerías de soporte para PHP, tal como se muestra en la Figura 2.22.

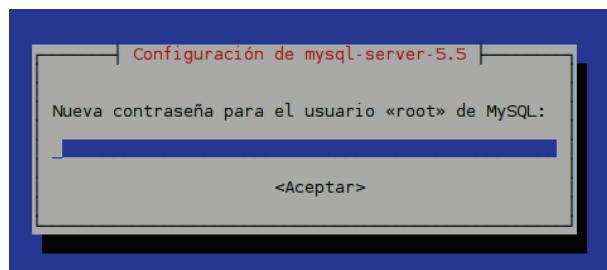
```
pi@cesar ~ $ sudo apt-get install php5-mysql mysql-server mysql-client
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes extras:
  libaio1 libdbd-mysql-perl libdbi-perl libhtml-template-perl libmysqlclient18
  libterm-readkey-perl mysql-client-5.5 mysql-common mysql-server-5.5
  mysql-server-core-5.5
Paquetes sugeridos:
  libclone-perl libmldb-perl libnet-daemon-perl libsql-statement-perl
  libipc-sharedcache-perl mailx tinyca
Se instalarán los siguientes paquetes NUEVOS:
  libaio1 libdbd-mysql-perl libdbi-perl libhtml-template-perl libmysqlclient18
  libterm-readkey-perl mysql-client mysql-client-5.5 mysql-common mysql-server
  mysql-server-5.5 mysql-server-core-5.5 php5-mysql
0 actualizados, 13 nuevos se instalarán, 0 para eliminar y 182 no actualizados.
Se necesita descargar 8.245 kB de archivos.
```

**Figura 2.22** Instalación de MySQL.

Durante este paso de instalación se abrirá una ventana, como la que se muestra en la Figura 2.23, donde se tendrá que ingresar una contraseña para el usuario root que administrará la base de datos, posteriormente aparecerá una nueva ventana donde se debe confirmar la contraseña que se introdujo anteriormente, tal como se muestra en la Figura 2.24.



**Figura 2.23** Configuración de la contraseña del usuario root.



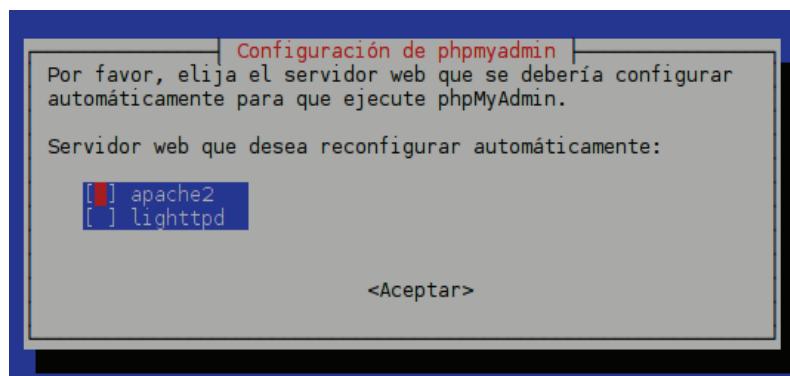
**Figura 2.24** Confirmación de la contraseña del usuario root.

5. Se instala el administrador gráfico web phpMyAdmin para la base de datos, tal como se muestra en la Figura 2.25.

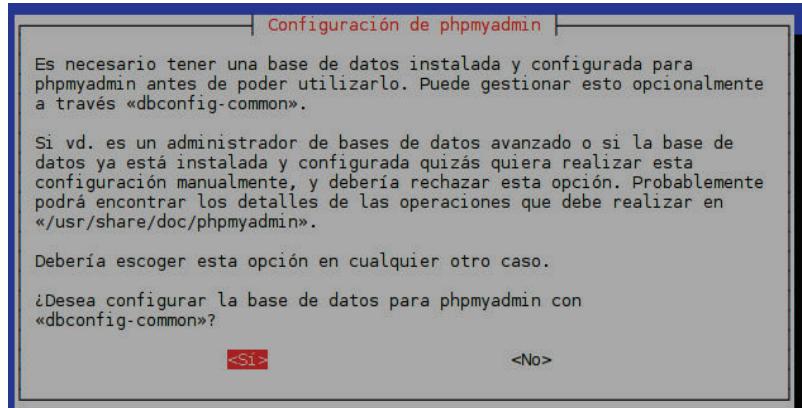
```
pi@cesar ~ $ sudo apt-get install phpmyadmin
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes extras:
  dbconfig-common libjs-sphinxdoc libjs-underscore libmcrypt4 php-gettext
  php-tcpdf php5-mcrypt
Paquetes sugeridos:
  libmcrypt-dev mcrypt php5-imagick
Se instalarán los siguientes paquetes NUEVOS:
  dbconfig-common libjs-sphinxdoc libjs-underscore libmcrypt4 php-gettext
  php-tcpdf php5-mcrypt phpmyadmin
0 actualizados, 8 nuevos se instalarán, 0 para eliminar y 182 no actualizados.
Se necesita descargar 15,4 MB de archivos.
```

**Figura 2.25** Instalación de PhpMyAdmin.

Durante el proceso de instalación del administrador gráfico web phpMyAdmin aparecerá una ventana, en la cual se debe seleccionar el servidor en el cual se va a ejecutar PhpMyAdmin, para el caso del presente proyecto es el servidor Apache, ver Figura 2.26; después de seleccionar el servidor, aparecerá una nueva ventana donde se indica que es necesario tener una base de datos instalada y configurada para poder comenzar a utilizar phpMyAdmin, ver Figura 2.27, en este caso se escogerá la opción “Si”, dado que la base de datos aún no se encuentra configurada.

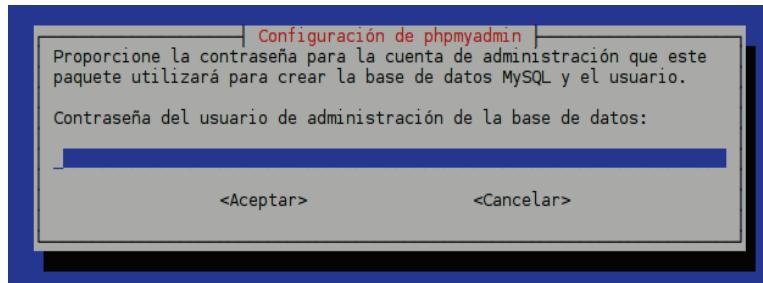


**Figura 2.26** Configuración del servidor sobre el cual se ejecuta PhpMyAdmin.

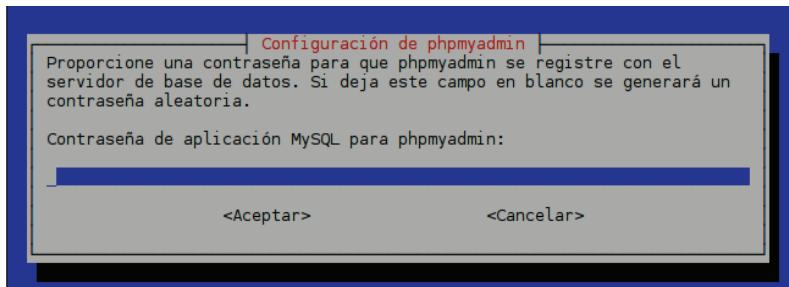


**Figura 2.27** Mensaje de configuración de PhpMyAdmin.

Posteriormente a los pasos mencionados, aparecerán dos ventanas más, en la primera como se muestra en la Figura 2.28 se tiene que introducir la contraseña de usuario root que se utilizó en el paso 4, para que phpMyAdmin pueda acceder a la base de datos MySQL; y en la segunda ventana que se muestra en la Figura 2.29 se debe introducir una contraseña para ingresar a phpMyAdmin.



**Figura 2.28** Configuración de la contraseña de PhpMyAdmin para acceder a MySQL.



**Figura 2.29** Configuración de la contraseña para ingresar a PhpMyAdmin.

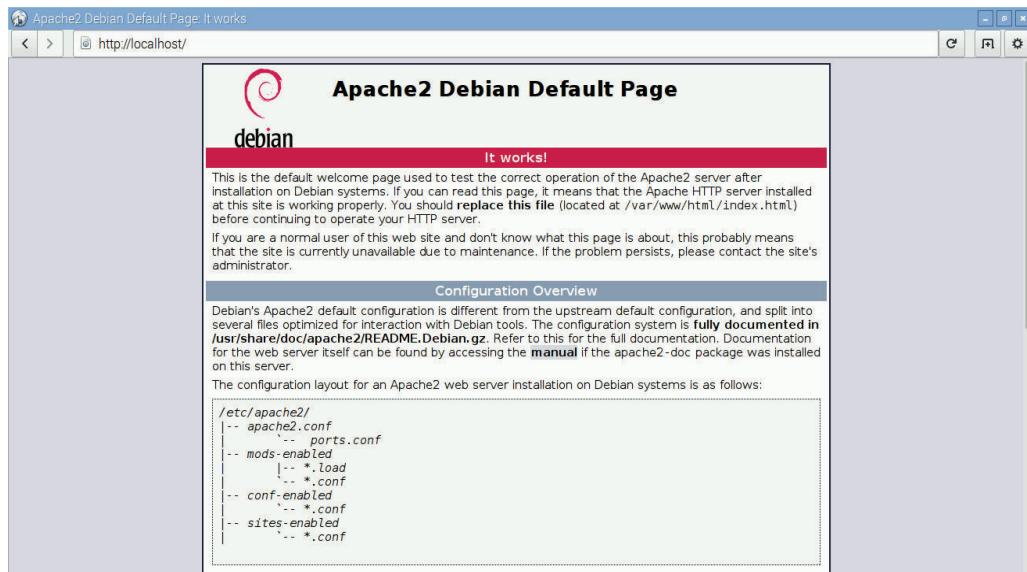
6. Se abre el fichero de configuración de Apache `/etc/apache2/apache2.conf` y se agrega la instrucción para que Apache pueda trabajar con el administrador gráfico phpMyAdmin, tal como se muestra en la Figura 2.30.

```
Include /etc/phpmyadmin/apache.conf
```

**Figura 2.30** Configuración del fichero Apache.

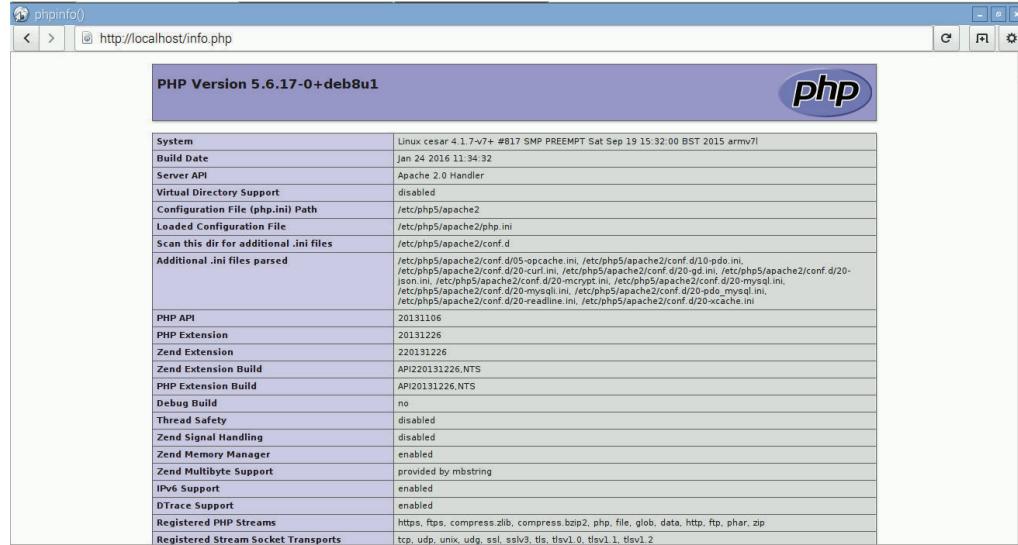
7. Finalmente, para comprobar que cada una de las herramientas utilizadas para la implementación de la aplicación web fueron instaladas correctamente se realiza lo siguiente:

- Escribir `localhost` en un *web browser* para verificar que el servidor se encuentre levantado, tal como se muestra la Figura 2.31.



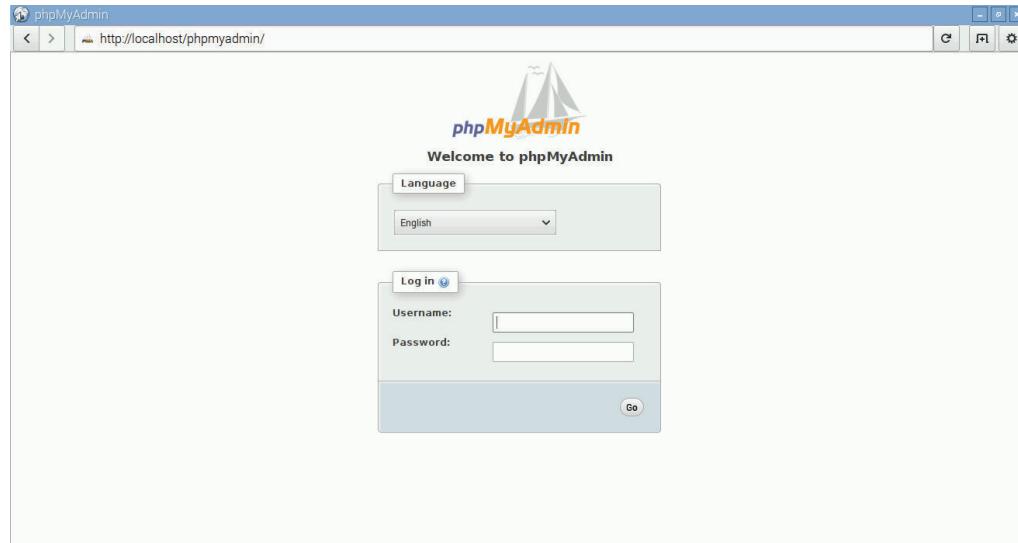
**Figura 2.31** Pantalla de inicio del servidor Apache.

- Escribir `localhost/info.php` en un *web browser* para verificar que se puede utilizar *scripts* en PHP, tal como se muestra en la Figura 2.32.



**Figura 2.32** Pantalla de información de PHP.

- Escribir *localhost/phpmyadmin* en un web browser para verificar que el administrador gráfico funciona correctamente, ver Figura 2.33.

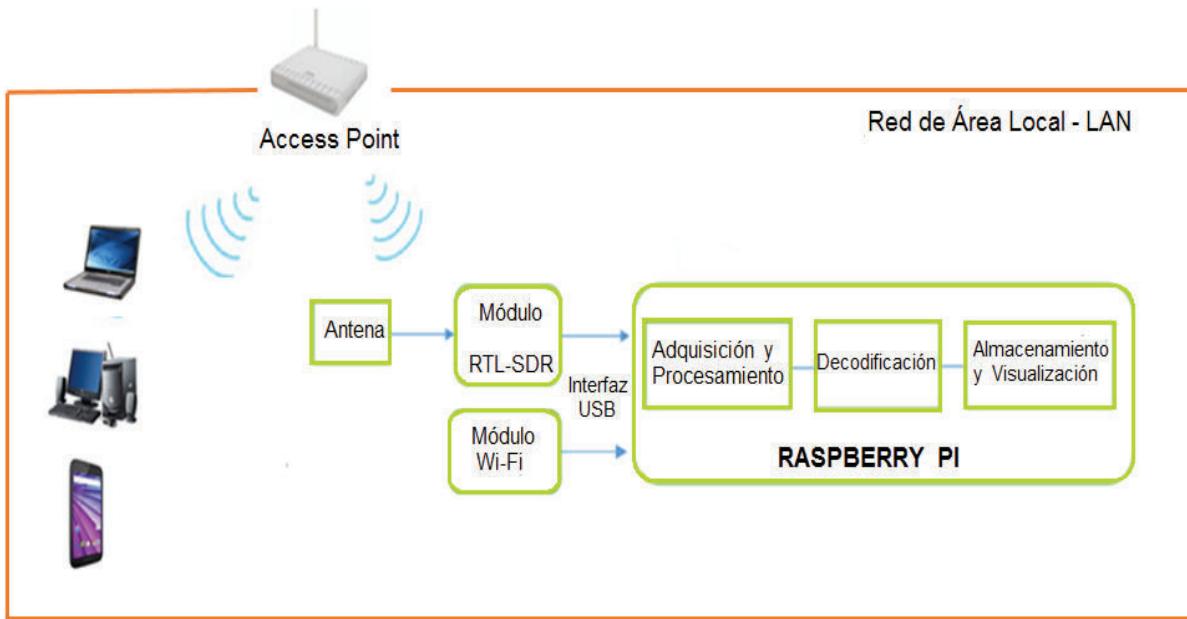


**Figura 2.33** Pantalla de inicio de PhpMyAdmin.

## 2.2 ESQUEMA Y FUNCIONAMIENTO DEL PROTOTIPO

En la Figura 2.34 se puede observar el entorno de red en el que se encuentra el prototipo, así también, se puede observar que este se encuentra formado por 6 bloques

que interactuarán entre sí para la correcta recepción y visualización de imágenes meteorológicas.



**Figura 2.34** Esquema del sistema prototipo de recepción de imágenes meteorológicas.

El funcionamiento del prototipo para la recepción de las imágenes meteorológicas, en base al esquema de la Figura 2.34, consiste en primer lugar de una antena, la cual será capaz de captar la señal proveniente de los satélites NOAA, luego, el módulo RTL-SDR trabajará conjuntamente con la aplicación GNU Radio Companion (GRC) a través de la interfaz USB de la Rapberry Pi para procesar y almacenar la señal captada por la antena, una vez que se haya obtenida la señal, con la ayuda del software WXtolmg se decodificará dicha señal para obtener la imagen.

Por último la imagen será almacenada en una base de datos, la cual trabajará conjuntamente con la aplicación web. Para la visualización de las imágenes por medio de la aplicación web la placa Raspberry Pi, a través del módulo Wi-Fi, se conectará a una red de área local inalámbrica (WLAN).

La conexión a la red de área local se realizará a través de un AP (Access Point), ver Figura 2.34; por lo tanto, todas aquellas computadoras, tablets, teléfonos celulares, etc. que se encuentren conectados al AP pertenecerán a la red de área local y podrán

acceder a las imágenes a través de sus navegadores web colocando la dirección IP de la Raspberry Pi.

## **2.3 DISEÑO E IMPLEMENTACIÓN DEL PROTOTIPO**

La implementación del prototipo se realizará y describirá en base a cada uno de los bloques del esquema de la Figura 2.34.

### **2.3.1 BLOQUE DE LA ANTENA**

Como se mencionó en la sección 1.4.2, existen varios tipos de antenas que se pueden utilizar para la recepción de imágenes meteorológicas. En el presente proyecto se construirán y se compararán dos de las antenas mencionadas en dicha sección, las cuales son:

- 1) Antena Turnstile.
- 2) Antena Cuadrifilar Helicoidal (QHA).

Puesto que en los objetivos del presente proyecto no se consideró el diseño de la antena, la construcción de las dos antenas mencionadas se realizará con los diseños tomados de [25], [47] para la antena turnstile y de [20], [24] para la antena cuadrifilar helicoidal. Para cada una de estas antenas se mencionará de manera general los cálculos realizados y el proceso de construcción.

Dado que los satélites transmiten a frecuencias que se encuentran entre los 137 y 138 MHz, la frecuencia que se utilizará para los cálculos en ambas antenas será de 137.5 MHz.

Una vez finalizada la construcción de las antenas, las mismas serán probadas y sus resultados serán comparados, a fin de determinar que antena permite captar de mejor manera la señal en formato APT proveniente de los satélites NOAA.

#### **2.3.1.1 Antena Turnstile**

Se seleccionó la antena turnstile por las buenas características que presenta para la recepción de la señal en formato APT (*Automatic Picture Transmission*) proveniente de los satélites NOAA y además por su facilidad de construcción [18].

### 2.3.1.1.1 Cálculos de la antena turnstile

Debido a que esta antena consta de dos dipolos, en primer lugar se calculará la longitud de cada uno de estos dipolos, para lo cual se utiliza la ecuación (2.1) [48].

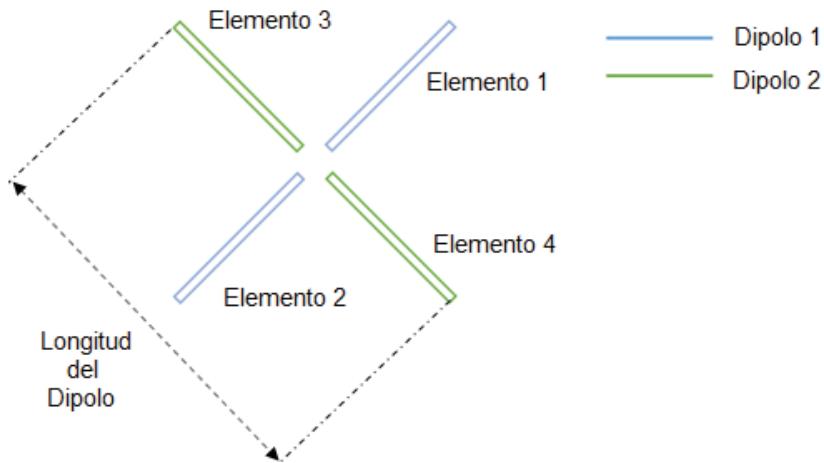
$$L = \frac{142.5}{F} \quad (2.1)$$

Donde:

- 142.5: Es una constante que resulta de dividir la velocidad de la luz por 2 y multiplicar por un factor de velocidad de 0.95 [48].
- F: es la frecuencia en MHz.
- L: Longitud del dipolo expresada en metros.

$$\text{Longitud del dipolo} = \frac{142.5}{137.5 \text{ MHz}} = 1.036 \text{ m}$$

Cada uno de los dipolos tendrá una longitud de 1.036 metros y cada dipolo se dividirá en dos elementos, tal como se muestra en la Figura 2.35.



**Figura 2.35** División de los dipolos.

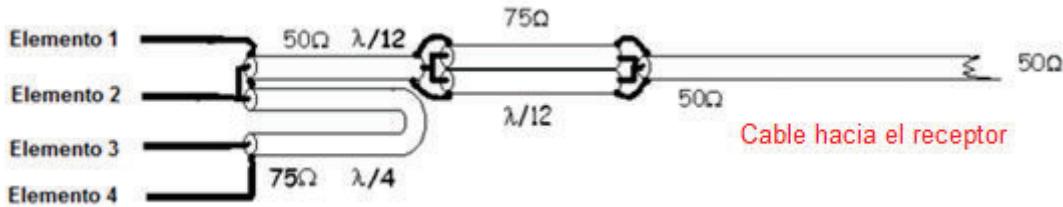
Para el montaje de la antena se necesitará una estructura donde apoyar los dipolos, dicha estructura será una tabla triplex de 10x10 cm.

Una vez realizados los cálculos para los dipolos se procede a realizar los cálculos de los cables coaxiales que permitirán tanto la conexión entre los dipolos, como la

conexión hacia el módulo RTL-SDR. Se debe tener en consideración que la impedancia característica del conector que posee del módulo RTL-SDR es de  $50 \Omega$ , tal como se mencionó en la sección 1.6.1, por lo que para asegurar una máxima transferencia de potencia, el cable coaxial de la antena que se conectará al módulo también debe presentar este valor de impedancia.

En la Figura 2.36 se pueden observar las diferentes longitudes de cable coaxial que se necesitarán, así como también la impedancia característica de cada cable utilizado. Cabe mencionar que para el cálculo de las longitudes de los cables se necesitará conocer el valor de la constante de la velocidad de propagación que posee cada cable coaxial.

- Constante de la velocidad de propagación Cable RG58 – 0.66 [49].
- Constante de la velocidad de propagación Cable RG59 – 0.82 [50].



**Figura 2.36** Conexión y longitudes de los cables coaxiales.

A continuación, se muestran los cálculos para  $\lambda$ .

$$\lambda = \frac{c}{f} = \frac{3 \times 10^8}{137.5 \times 10^6} = 218.18 \text{ cm}$$

$$\frac{\lambda}{4} = \frac{2.1818}{4} \times 0.82 = 44.7 \text{ cm}$$

$$\frac{\lambda}{12_{50\Omega}} = \frac{2.1818}{12} \times 0.66 = 12 \text{ cm}$$

$$\frac{\lambda}{12_{75\Omega}} = \frac{2.1818}{1} \times 0.82 = 15 \text{ cm}$$

### 2.3.1.1.2 Construcción de la antena turnstile

Para la construcción de la antena turnstile se necesitarán los siguientes elementos.

- Cable coaxial RG58, 300 cm.
- Cable coaxial RG59, 100 cm.
- Tubo PVC de 4 cm de diámetro y 1.5 metros de largo para el mástil.
- Una tabla de 10x10 cm, para la base de los dipolos.
- Cautín y estaño para soldar las conexiones.
- Conector BNC macho.

Los pasos para la construcción son:

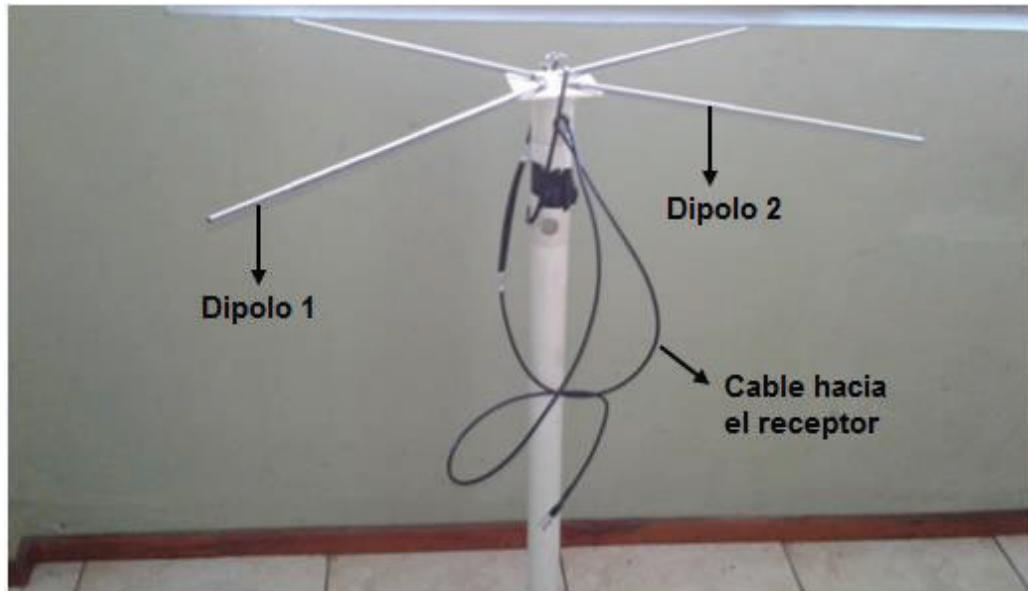
1. Cortar los dipolos y los cables coaxiales en base a las longitudes calculadas anteriormente.
2. Realizar perforaciones en la tabla, para luego colocar los dipolos, tal como se muestra en la Figura 2.37.



**Figura 2.37** Base y elementos de los dipolos.

3. Soldar las conexiones entre los cables coaxiales y los 4 elementos de la antena, así como también las conexiones entre cables coaxiales, tal como se muestra en la Figura 2.36. Colocar el conector BNC macho en el extremo del cable coaxial que va hacia el módulo RTL-SDR.

Una vez realizada todas las conexiones la antena estará lista para poder realizar las pruebas. En la Figura 2.38 se observa la antena ya terminada.



**Figura 2.38** Antena Turnstile.

### 2.3.1.2 Antena Cuadrifilar Helicoidal

Se seleccionó este tipo de antena como segunda opción debido a que presenta un mejor diagrama de radiación que la antena turnstile en el plano horizontal [18].

La principal característica de este tipo de antena es que consta de dos lazos resonantes de diferente longitud, donde, si aumenta el tamaño de un lazo se debe reducir el tamaño del otro.

#### 2.3.1.2.1 Cálculos de la antena cuadrifilar helicoidal

Para la realización de los cálculos de este tipo de antena se utilizará una aplicación web, la cual fue utilizada en [20] y cuyos resultados comparados a los obtenidos en [24] son muy similares. La aplicación web es conocida como calculadora de Jhon Coppens y se encuentra disponible en la página <http://jcoppens.com/ant/qfh/calc.en.php>.

En primer lugar, se deben ingresar los parámetros de la antena, tal como se muestra en la Figura 2.39.

Frecuencia de diseño	137.5	MHz
Cantidad de vueltas	0.5	
Largo de una vuelta	1	largos de onda
Radio del doblado	15	mm
Diámetro del conductor	5	mm (valor óptimo: 20.5 mm)
Relación diámetro/altura	0.44	
		Calcular

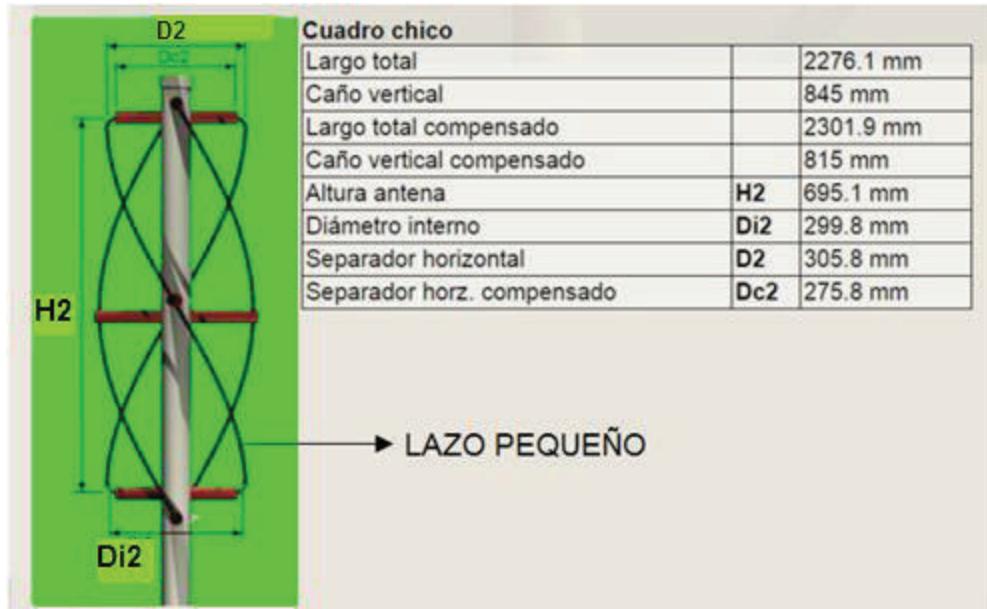
**Figura 2.39** Datos ingresados en la calculadora de Jhon Coppens.

En el Anexo B, se encuentra a detalle cada uno de los parámetros ingresados que se muestran en la Figura 2.39.

Una vez ingresados todos estos valores se genera una serie de parámetros tanto para el lazo grande como para el lazo pequeño, lo cual permitirá la construcción de la antena, tal como se muestra en la Figura 2.40 y en la Figura 2.41.



**Figura 2.40** Resultados para el lazo grande.



**Figura 2.41** Resultados para el lazo pequeño.

#### 2.3.1.2.2 Construcción de la antena cuadrifilar helicoidal

Para la construcción de la antena se han utilizado los datos de la Figura 2.40 y de la Figura 2.41, por lo tanto, los materiales de construcción son los siguientes:

- Cable coaxial RG58, 700 cm.
- Tuvo PVC de 4 cm de diámetro y 200 cm de largo para el mástil.
- Tuvo PVC de 1 cm de diámetro y 200 cm de largo para los brazos.
- Cautín y estaño para soldar las conexiones.
- Conector BNC macho.

Se utilizará cable coaxial RG58 puesto que este presenta una impedancia característica de  $50 \Omega$ , lo que permite que exista una máxima transferencia de potencia hacia el módulo RTL-SDR.

Los pasos a seguir para la construcción son:

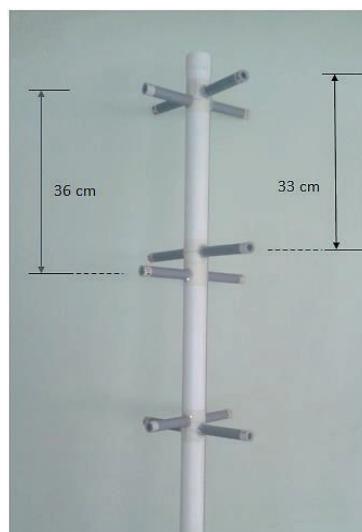
1. La antena necesitará un total de 12 brazos, tal como se muestra en la Figura 2.42, los cuales se dividen de la siguiente manera:

- 4 brazos para los extremos de la antena (lazo grande) con una medida de 12.5 cm.
- 4 brazos para los extremos de la antena (lazo pequeño) con una medida de 11.7 cm.
- 2 brazos intermedios de 13.5 cm para el lazo grande.
- 2 brazos intermedios de 12.7 cm para el lazo pequeño.



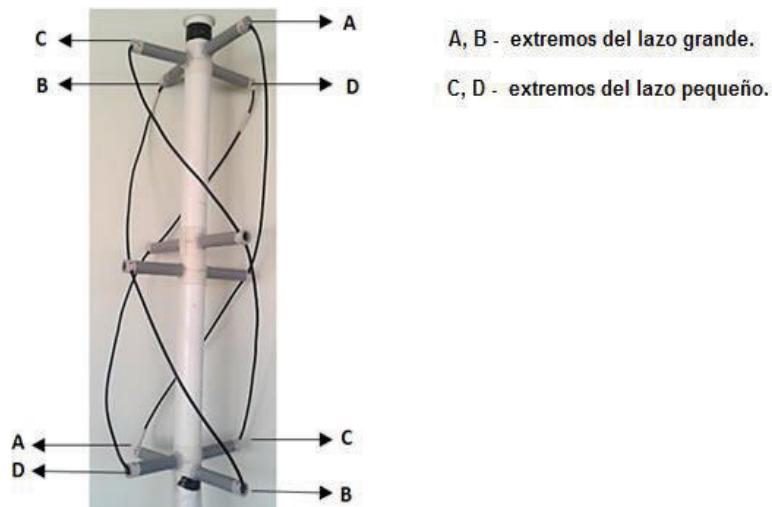
**Figura 2.42** Brazos de la antena.

2. Luego, se procede a medir y perforar el mástil en base a los datos que se muestran en la Figura 2.40 y en la Figura 2.41. Posteriormente cada uno de los brazos tienen que ser colocados en el mástil, tal como se muestra en la Figura 2.43. Por último en los brazos intermedios se deben realizar orificios a 1 cm de los extremos para que atraviese el cable coaxial.



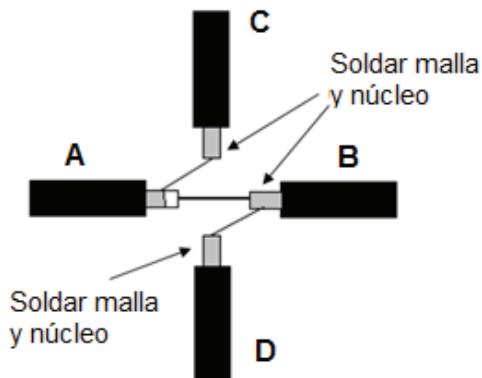
**Figura 2.43** Colocación de los brazos en el mástil de la antena.

3. Una vez colocados los brazos se procede a colocar el cable coaxial, para lo cual, el lazo grande es dividido en dos partes de igual longitud, mientras que al lazo pequeño no se le realiza ningún corte. Los extremos del lazo grande se asociarán con las letras A y B, y los extremos del lazo pequeño con las letras C y D. Una vez colocados los cables de conexión la antena lucirá tal como se muestra en la Figura 2.44.

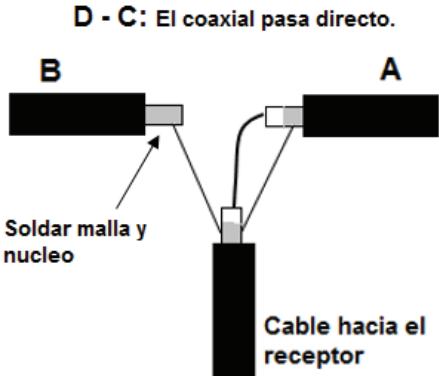


**Figura 2.44** Brazos y cable coaxial de la antena.

4. Una vez colocados los cables se deben soldar las conexiones que existen tanto en la parte superior como en la parte inferior de la antena, tal como se muestra en la Figura 2.45 y en la Figura 2.46 respectivamente. En el extremo del cable que va hacia el módulo RTL-SDR se debe colocar el conector BNC macho.



**Figura 2.45** Conexión superior de la antena.



**Figura 2.46** Conexión inferior de la antena.

Una vez realizadas las conexiones se observa que los tramos B, C y D forman un conductor único, debido a que la malla y el núcleo del cable coaxial se sueldan.

### 2.3.2 BLOQUE DEL MÓDULO RTL-SDR

Este bloque hace referencia al módulo RTL-SDR que se utilizará para la realización del presente proyecto y cuyas características fueron mencionadas en la sección 1.6.1. Así también, en dicha sección se mencionó que este dispositivo consta principalmente de dos chips, el primero es el sintonizador R820T y el segundo es el demodulador RTL2832U cuyas características técnicas se mencionaron en la sección 1.6.1.1 y 1.6.1.2 respectivamente.

Al módulo RTL-SDR no se le realizará ningún tipo de configuración directamente, tan solo será necesario conocer los diferentes valores de ganancia y que además pueda ser reconocido y que trabaje conjuntamente con la aplicación GNU Radio Companion, para la cual su instalación y configuración, detallada en la sección 2.1.1, debió ser realizada correctamente.

### 2.3.3 BLOQUE DE ADQUISICIÓN

Este bloque con la ayuda de la antena y del módulo RTL-SDR permitirá el cumplimiento del requisito a (ver sección 2.1).

Previamente a la realización del esquema de recepción de imágenes meteorológicas en la aplicación GNU Radio Companion, se revisó y comparó la documentación presentada en [21], [51] y [52].

Posteriormente a la revisión de la documentación, se concluyó que se debe determinar el ancho de banda que se necesitará para la adquisición de la señal en formato APT que se encuentra modulada en frecuencia, para lo cual, en primer lugar se debe calcular el índice de modulación ( $\beta$ ). Mediante la ecuación (2.2) [53], se calcula el índice de modulación que presenta la señal APT modulada en frecuencia.

$$\beta = \frac{\Delta f}{f_m} \quad (2.2)$$

Donde:

- $\Delta f$ : Es el máximo cambio de frecuencia pico.
- $f_m$ : Es la frecuencia moduladora máxima.
- $\beta$ : Índice de modulación.

Como se mencionó en la sección 1.4.1.5, el cambio máximo de frecuencia pico que presenta la señal APT modulada en frecuencia es de 17 KHz, y la frecuencia moduladora máxima es la subportadora de 2400 Hz. Reemplazando estos datos en la ecuación (2.2) se tienen lo siguiente:

$$\beta = \frac{17 \text{ KHz}}{2.4 \text{ KHz}} = 7.08$$

Dado que el índice de modulación calculado es mayor que uno, se hace uso de la regla de Carson para el cálculo del ancho de banda, esta regla asegura que pasará la suficiente potencia de las bandas laterales, asegurando la inteligibilidad plena de la señal. La regla de Carson establece lo siguiente [53]:

$$AB = 2(\Delta f + f_m) \quad (2.3)$$

Donde:

- $\Delta f$ : Es el máximo cambio de frecuencia pico.
- $f_m$ : Es la frecuencia moduladora máxima.

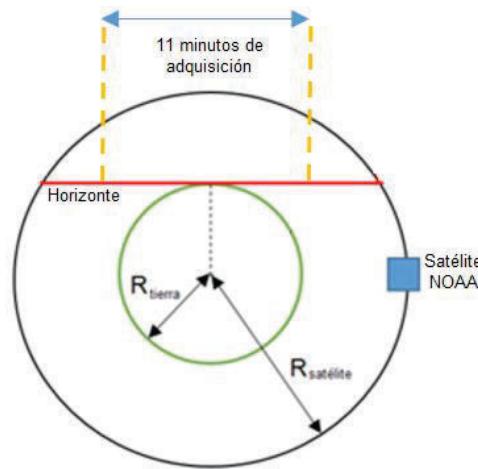
Reemplazando los valores de la señal APT antes mencionados en la ecuación (2.3), se tienen lo siguiente:

$$AB = 2(17 \text{ KHz} + 2.4 \text{ KHz}) = 38.8 \text{ KHz}$$

El ancho de banda calculado para la adquisición de la señal en formato APT es de 38.8 KHz. Para el caso del presente proyecto, se redondeará este valor calculado al inmediato superior, es decir, 39 KHz. Luego de adquirida la señal modulada en frecuencia se tiene que demodularla para obtener la señal modulada en amplitud (AM), la cual, contiene la información del sensor AVHRR/3. Por último, se almacenará la señal AM para su posterior decodificación.

Como se mencionó en la sección 1.4.1.3 el tiempo de visibilidad calculado de 15.7 minutos para un satélite NOAA es un caso ideal, por lo tanto, para el presente proyecto se realizará la adquisición de la señal en formato APT durante un tiempo de 11 minutos.

Este tiempo ha sido seleccionado teniendo en consideración que en el inicio del horizonte, cuando el satélite empieza a ser visible, y en el fin del horizonte, cuando el satélite deja de ser visible no se podrá captar adecuadamente la señal en formato APT. En la Figura 2.47 se puede observar el tiempo de adquisición (11 minutos), en referencia a la porción de órbita que recorre el satélite NOAA sobre el horizonte.



**Figura 2.47** Tiempo de adquisición de la señal en formato APT.

En la Figura 2.48 se muestra el esquema realizado en la aplicación GNU Radio Companion.

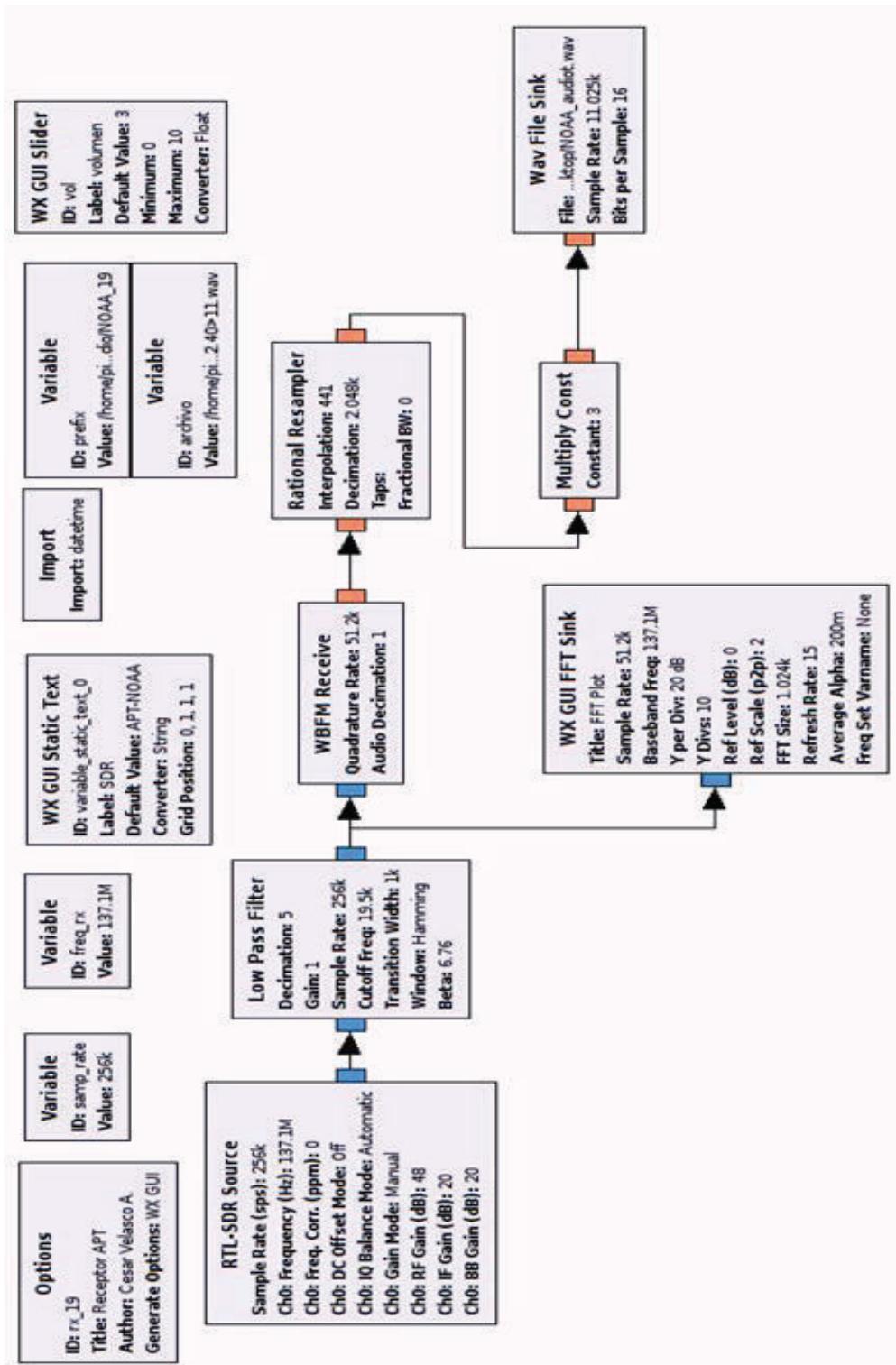


Figura 2.48 Esquema de recepción en GNU Radio Companion

A continuación, se describe la configuración realizada en los diferentes bloques presentes en el esquema de la Figura 2.48.

- **RTL-SDR Source:** Este bloque se encarga de realizar la comunicación entre la aplicación GNU Radio Companion y el módulo RTL-SDR que se encuentra conectado a la Raspberry Pi.

En este bloque se configuraron los siguientes parámetros:

- Frecuencia de muestreo: La frecuencia de muestreo configurada en este bloque es la frecuencia de muestreo práctica a la cual trabaja el chip RTL2832U, el cual se encuentra embebido en el módulo RTL-SDR; esta frecuencia en base a [54] debe encontrarse en el rango comprendido entre 225 – 300 KHz y 900 - 3200 KHz para obtener un correcto funcionamiento del chip. Para el caso del presente proyecto, se ha seleccionado 256 KHz, puesto que este valor es un múltiplo entero de 11025 Hz, que es la frecuencia que necesita el software WXtoImg. Además, esta frecuencia de muestreo cumple con el criterio de Nyquist, siendo mayor al doble del ancho de banda de la señal en banda base (39 KHz).
  - Frecuencia central: Es la frecuencia de transmisión de la señal en formato APT de los satélites NOAA, ver Tabla 1.5; por lo tanto este parámetro puede tomar 3 diferentes valores dependiendo del satélite del cual se desee captar la señal.
  - Ganancia: Debido a que las antenas que se utilizarán en el presente proyecto son de baja ganancia, se seleccionó un valor de ganancia alto del módulo RTL-SDR, el cual es 48 dB, ver Figura 2.13.
  - **Low Pass Filter:** Un filtro pasa bajos que permite suprimir aquellas frecuencias que no son necesarias.
- En este bloque se configuraron los siguientes parámetros:
- Frecuencia de muestreo: La misma que el bloque anterior (bloque RTL-SDR Source), 256 KHz.

- Frecuencia de corte: Dado que los datos de entrada y salida de este bloque son de tipo complejo, el valor configurado en este parámetro debe ser la mitad del ancho de banda necesario para la adquisición de la señal en formato APT, es decir, 19.5 KHz. Con esto se obtiene 19.5 KHz para la parte real y 19,5 KHz para la parte imaginaria; obteniendo así, los 39 KHz calculados anteriormente con la ecuación (2.3).
  - Anchura de transición: 1 KHz
  - Ventana: En este parámetro se configuró el tipo de ventana que se usa en el filtro. Para el caso del presente proyecto, se ha seleccionado la ventana de Hamming, ya que esta ventana presenta buenas características, permitiendo reducir el rizado en la banda pasante y en la banda suprimida.
  - Diezmado: 5, este valor permite obtener a la salida del bloque una frecuencia de muestreo dividida por 5, es decir 51.2 KHz. Este diezmado se realiza a fin de obtener posteriormente una frecuencia de muestreo de 11025 Hz, la cual es la frecuencia de muestreo que necesita el *software* WXtolmg.
  - **WBFM Receive:** Este bloque es un demodulador FM por cuadratura<sup>23</sup>, permite realizar la demodulación de la señal en formato APT modulada en frecuencia transmitida por los satélites NOAA y así obtener la señal modulada en amplitud (AM).
- En este bloque se configuraron los siguientes parámetros:
- Tasa de cuadratura: Es la frecuencia de muestreo que existe a la salida del bloque anterior (bloque Low Pass Filter), es decir, 51.2 KHz.
  - **Rational Resampler:** Este bloque permitirá realizar un re-muestreo<sup>24</sup> de la señal AM, obteniendo una tasa de muestreo de 11025 Hz a partir de 51.2 KHz,

---

<sup>23</sup> Demodulador FM por cuadratura, este demodulador extrae la señal de información mediante la multiplicación de la señal modulada FM recibida y la señal en cuadratura de esta señal modulada (desfasada 90°).

<sup>24</sup> Re-muestreo, hace referencia a la obtención de una nueva frecuencia de muestreo a partir otra que fue multiplicada por un factor de interpolación y dividida por un factor de diezmado.

esto debido a que el software WXtoIMG necesita que la señal AM se encuentre muestreada a esta frecuencia para poder realizar la decodificación.

- **Multiply Const:** La función de este bloque es la de controlar el volumen del audio, el cual tiene un valor mínimo de 1 y máximo de 10, y un valor por defecto de 3.
- **Wav File Sink:** Este bloque es un sumidero de audio, el cual permitirá almacenar la señal AM en un archivo de audio, cuya extensión es .WAV<sup>25</sup>. Este archivo de audio almacenado es aquel que posteriormente será decodificado.

#### 2.3.4 BLOQUE DE DECODIFICACIÓN

Este bloque permitirá el cumplimiento del requisito b, el cual consiste en realizar la decodificación a un formato de imagen. Esta decodificación se realizará a la señal modulada en amplitud (AM), la cual ha sido almacenada previamente por el bloque de adquisición.

El *software* WXtolmg permitirá realizar la decodificación mediante la línea de comandos o mediante la interfaz gráfica del *software*.

En el presente proyecto se escogió utilizar la línea de comandos, para lo cual se dispone del siguiente formato, el cual se ejecutará en la *shell*<sup>26</sup> del Raspberry Pi:

“wxtoimg [-a] [-b] [-e option] input.wav imagen.png”

A continuación, se describirá cada una de las opciones que presenta el formato mencionado:

- -N: Muestra una imagen donde aparece el canal A y el canal B.
- -a: Muestra una imagen del canal A.
- -b: Muestra una imagen del canal B.
- -e option: Es una opción de mejora y siempre tiene que ir acompañado de una de las siguientes opciones.

---

<sup>25</sup> WAVE form audio file format, es un formato de audio digital normalmente sin compresión de datos que admite diversas resoluciones y velocidades de muestreo.

<sup>26</sup> Shell o intérprete de comandos, es un programa informático que provee una interfaz de usuario para acceder a los servicios del sistema operativo.

- ZA: Esta opción permite tener una mejora mediante el aumento en el contraste saturado de las regiones cuyas temperaturas son muy bajas y muy altas.
- MB: Esta opción permite mostrar de mejor manera dónde se producen las precipitaciones más fuertes en las tormentas.
- MD: Es una modificación de la opción MB, presentando mejoras dentro de los grises.
- BD: Es una opción de mejora para huracanes, permite destacar ciertas temperaturas de la tormenta que están relacionadas con la intensidad del huracán.
- HF: Es una opción utilizada para la identificación de nubes asociadas con tormentas eléctricas.
- JF: Es una opción híbrida, utilizada para resaltar temperaturas tanto de la superficie del mar como de aquellas nubes asociadas a tormentas eléctricas.
- NO: Esta opción mejora el contraste de las regiones más oscuras de la tierra, del mar y el color de las nubes frías, permitiendo tener detalles más finos.
- HVC: Esta opción crea una imagen en falso color.
- HVCT: Similar a la opción anterior, pero con colores más indicativos de la temperatura de la tierra.
- MCIR-precip: Crea una imagen en falso color, aclarando aquellas zonas donde existen precipitaciones.
- sea: Crea una imagen en falso color basado en el la temperatura superficial del mar. Las nubes altas y la tierra aparece de color negro.
- therm: Crea una imagen en falso color basado en el la temperatura. Proporciona una buena forma de visualizar las temperaturas de las nubes.
- invert: Crea un negativo en escala de grises, estableciendo el negro a blanco y el blanco al negro.
- bw: crea una imagen en blanco y negro.

- Input.wav: Es el archivo de extensión .WAV que fue previamente almacenado por la aplicación GNU Radio Companion.
- imagen.png: Es la imagen que se obtendrá luego de realizar la decodificación, y cuya extensión será .png.

Dado que existen diferentes variantes de las imágenes, para el presente proyecto se han seleccionado 3 de estas variantes, las cuales son:

- -a
- -b
- -e therm

Se seleccionaron las variantes –a y –b con el fin de obtener por separado las imágenes de luminancia tanta del canal A como del canal B, así también, se seleccionó la opción de -e therm, la cual entrega una imagen térmica.

La razón por la cual se seleccionó mostrar una imagen térmica, es debido a que la temperatura es una de las magnitudes más utilizadas para describir el estado de la atmósfera.

### **2.3.5 BLOQUE DE ALMACENAMIENTO Y VISUALIZACIÓN**

Este bloque permitirá el cumplimiento de los requisitos c y d, ver sección 2.1.

#### **2.3.5.1 Almacenamiento**

Para el cumplimiento del requisito c, se utilizará el gestor de base de datos MySQL, la herramienta web phpMyAdmin y *scripts* desarrollados en Python<sup>27</sup>, PHP y HTML.

En primer lugar, se creará una base de datos denominada Imagen\_NOAA, la cual estará basada en un modelo relacional<sup>28</sup>. Dentro de la tabla Imagen\_NOAA se crearán

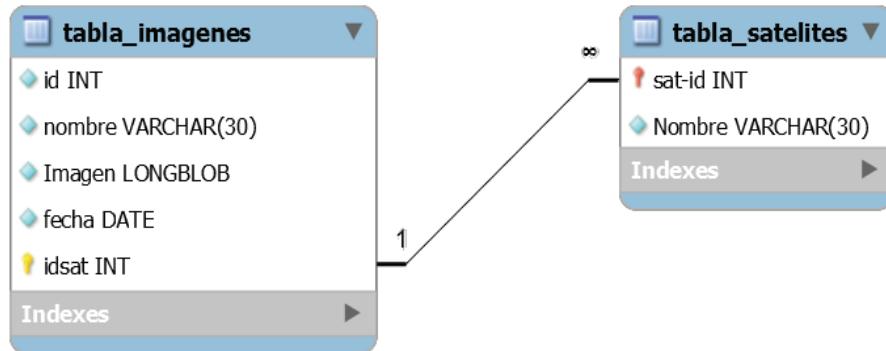
---

<sup>27</sup> Python, es un lenguaje de programación con un enfoque simple pero efectivo a la programación orientada a objetos.

<sup>28</sup> Modelo relacional, es un modelo que permite establecer interconexiones y relacionar datos de diferentes tablas.

dos tablas denominadas `tabla_imagenes` y `tabla_satelites`, y por medio de una clave foránea<sup>29</sup> se establecerá una relación entre las dos tablas antes mencionadas.

En la Figura 2.49 se observa la base de datos que se utilizará en el presente proyecto.



**Figura 2.49** Base de datos para el sistema prototipo.

La tabla `tabla_satelites` contendrá el nombre de los tres satélites NOAA de los cuales se obtendrá las imágenes, los cuales estarán asociados a un identificador único. Los campos que componen esta tabla son los siguientes:

- **sat-id**: Identificador tipo *INT*, único para cada satélite y clave foránea de la tabla `tabla_imagenes`.
- **Nombre**: Es el nombre de los satélites y es de tipo *VARCHAR*.

La tabla `tabla_imagenes` contendrá las imágenes provenientes de cada uno de los tres satélites NOAA. Los campos que componen esta tabla son los siguientes:

- **id**: Identificador tipo *INT* auto incrementable, que permitirá mostrar las imágenes en la aplicación web de una manera ordenada (ascendente).
- **nombre**: Es el nombre que tendrá la imagen y será de tipo *VARCHAR*.
- **Imagen**: En este campo se guardará la imagen y será de tipo *LONGBLOB*.
- **fecha**: Este campo contendrá la fecha en la cual fue almacenada la imagen y será de tipo *DATE*.

---

<sup>29</sup> Clave foránea (*foreign key*), identifica una columna o un grupo de columnas en una tabla que se refiere a una columna o un grupo de columnas en otra tabla.

- **idsat:** Es el identificador único de cada satélite, el cual es asignado en el campo **sat-id** de la tabla tabla\_satelites.

Como se observa en la base de datos de la Figura 2.49 el campo **sat-id** de la tabla tabla\_satelites es la clave foránea del campo **idsat** de la tabla tabla\_imagenes, por lo tanto, el campo **idsat** solo podrá tomar aquellos valor que se encuentren configurados en el campo **sat-id**.

A continuación, se describirá el proceso de implementación para el almacenamiento.

Mediante la herramienta web phpMyAdmin se crea la base de datos, las tablas y la relación entre estas, para lo cual, se debe ingresar al navegador web de la Raspberry Pi y colocar “*localhost/phpmyadmin*”. Luego, ingresar como usuario *root* utilizando la contraseña ingresada en el paso 5 de la sección 2.1.4.

En la Figura 2.50 se muestra la base de datos y las tablas creadas dentro de esta, así como también, la clave foránea.



**Figura 2.50** Base de datos, tablas y clave foránea.

Luego, dentro de la tabla tabla\_satelites en el campo **Nombre** se ingresan los nombres “NOAA\_15”, “NOAA\_18” y “NOAA\_19” y en el campo **sat-id** los identificadores únicos “1”, “2” y “3”, tal como se muestra en la Figura 2.51.

		+ Opciones		sat-id	Nombre
		← →	▼		
<input type="checkbox"/>	Editar	Copiar	Borrar	1	NOAA_15
<input type="checkbox"/>	Editar	Copiar	Borrar	2	NOAA_18
<input type="checkbox"/>	Editar	Copiar	Borrar	3	NOAA_19

**Figura 2.51** Nombre e identificador único de los satélites NOAA.

Utilizando el lenguaje de programación Python se ha desarrollado un código que permite realizar la conexión, autenticación y almacenamiento a la base de datos de cada una de las tres imágenes que han sido previamente adquiridas y decodificadas por los bloques anteriores.

En el Código 2.1 se muestra, para el caso de la imagen del canal A, como se realiza la conexión, autenticación y almacenamiento a la base de datos.

En primer lugar mediante los comandos **open**, **read** y **close** se abre, se lee y se cierra la imagen respectivamente, luego, mediante el comando **MySQLdb.connect** se realiza la conexión y autenticación a la base de datos Imagen\_NOAA, posteriormente en la variable **cursor** se establece un puntero a la base de datos, y por último mediante los comandos **execute** y **commit** se almacena y efectúan los cambios en la base de datos. En el caso de un almacenamiento exitoso retornará la frase “Ingreso exitoso de la imagen del Canal-A” y en el caso de un fallo en el almacenamiento retornará “fallo”.

```
##### lectura de la primera imagen #####
fin = open("canala.png", "rb") #abre la imagen
img = fin.read()# lee la imagen
fin.close # cierra la imagen
#####conexion y almacenamiento a la base de datos #####
bd = MySQLdb.connect("localhost", "root", "201010782", "Imagen_NOAA" )
cursor = bd.cursor() # puntero a la base de datos
sql = "INSERT INTO tabla_imagenes (nombre, Imagen, fecha, idsat)
VALUES ('%s', '%s', '%s', '%s') "% (nom1, dbapi.escape_string(img), fecha, sat)
try:
    cursor.execute(sql) # se hace la modificacion enla base de datos
    bd.commit() # efectua los cambios
    time.sleep(2)
    print ("Ingreso exitoso de la imagen del Canal-A")
except:
    time.sleep(2)
    print ("fallo")
```

**Código 2.1** Conexión, autenticación y almacenamiento a la base de datos para el caso de la imagen del canal A.

### 2.3.5.2 Visualización

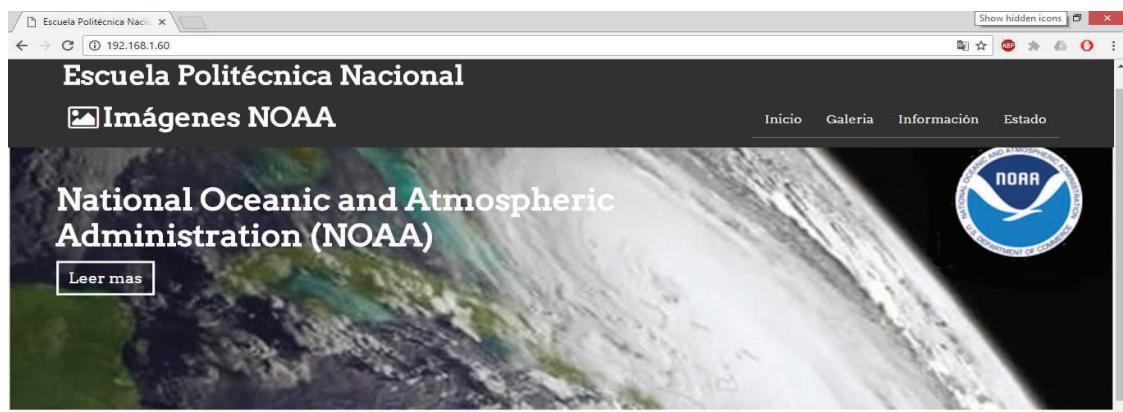
Para el cumplimiento del requisito d, se implementará una aplicación web sobre un servidor Apache. Esta aplicación constará de 4 Interfaces, las cuales son las siguientes:

- Inicio
- Galería
- Información
- Estado del Raspberry Pi

#### 2.3.5.2.1 Inicio

Esta opción de la aplicación web se divide en 3 secciones. En la primera de ellas se encuentra un *link* que re-direccionará hacia la página web oficial de la NOAA, en la segunda sección se encuentra una breve descripción del presente proyecto, y en la tercera y última sección, se muestran las principales características de cada uno de los satélites NOAA de los cuales se obtendrán las imágenes meteorológicas.

Así también, en la tercera sección se puede re-direccionar hacia una página web que realiza el *tracking online* de los satélites NOAA en tiempo real. A continuación se muestra la interfaz web **Inicio** implementada. En la Figura 2.52 se muestra la sección 1 y 2, y en la Figura 2.53 la sección 3.



#### INTRODUCCION

En el presente proyecto se realiza la implementación de un sistema prototípico de adquisición de imágenes meteorológicas provenientes de los satélites NOAA (National Oceanic and Atmospheric Administration) utilizando tecnología de Radio Definido por Software (RDS), hardware y software libre. Como dispositivo RDS se utiliza un módulo RTL-SDR compuesto por el chip demodulador RTL2832U y el sintonizador R820T; como hardware se utiliza la placa Raspberry Pi y como software se utiliza GNU Radio Companion y VVXtoIMG.

**Figura 2.52 Sección 1 y 2 de la interfaz web Inicio.**



**Figura 2.53** Sección 3 de la interfaz web Inicio.

#### 2.3.5.2.2 Galería

En esta opción de la aplicación web se puede seleccionar el satélite del cual se desea ver las imágenes meteorológicas, para lo cual se ha desarrollado un código escrito en lenguaje PHP embebido en HTML.

Por ejemplo, para el satélite NOAA-15, el Código 2.2 mediante los comandos **query** y **fetch\_assoc** permite mostrar cada una de las imágenes que se encuentren almacenadas dentro de la tabla `tabla_imagenes` y que estén asociadas con el identificador único del satélite NOAA-15.

```
$query = "SELECT * FROM tabla_imagenes WHERE idsat = 1"; /* Selecciona las imagenes que idsat=1 */
$resultado = $conexion->query($query);
while($row= $resultado->fetch_assoc()){ /* mientras exista imagenes con idsat=1 realizar*/
?><tr>
<td> <?php echo $row['nombre'];?></td> <!-- Muestra el contenido del campo nombre--&gt;
<!-- muestra el contenido del campo Imagen --&gt;
&lt;td&gt; &lt;img height="175px" src="data:image/jpg;base64,&lt;?php echo base64_encode($row['Imagen']); ?&gt;"&gt;&lt;/td&gt;
<!-- Link de descarga de la imagen --&gt;
&lt;td&gt; &lt;a href="data:image/jpg;base64,&lt;?php echo base64_encode($row['Imagen']); ?&gt;" target="_blank"&gt;Descarga&lt;/a&gt;&lt;/td&gt;
&lt;td&gt;&lt;?php echo $row['fecha'];?&gt;&lt;/td&gt;<!-- muestra el contenido del campo fecha --&gt;
&lt;/tr&gt;&lt;?php
?&gt;</pre>

```

**Código 2.2** Código para mostrar todas las imágenes almacenadas del NOAA-15.

Adicionalmente, dado que todas las imágenes se almacenan con su respectiva fecha, se implementa un motor de búsqueda por fechas, el cual permite mostrar las imágenes que se encuentren dentro del rango de fechas ingresadas por los usuarios.

En primer lugar, se crea una forma utilizando el lenguaje de programación HTML, para lo cual se ha desarrollado el Código 2.3. Esta forma permite a los usuarios ingresar el rango de fechas de la búsqueda.

```
<form name="busqueda" action="buscar3.php" method="post">
<input type="date" name="desde"/>
<input type="date" name="hasta"/>
<input type="submit" name="buscar" value="BUSCAR"/>
</form>
```

**Código 2.3** Motor de búsqueda.

En cuanto a la selección y visualización de las imágenes, para el caso del satélite NOAA-15, que se encuentren dentro del rango de fechas seleccionado, se ha desarrollado el Código 2.4.

```
/*selecciona las imagenes con idsat=1 y el campo fecha haga match entre las variables $desde y $hasta y ordena ascendente */
$query = "SELECT * FROM tabla_imagenes WHERE fecha BETWEEN '$desde' AND '$hasta' AND idsat = 1 ORDER BY id ASC";
$resultado = $conexion->query($query);
while($row= $resultado->fetch_assoc()){/*mientras exista imagenes con idsat=1*/
?
<tr>
<td> <?php echo $row['nombre'];?></td><!-- Muestra el contenido del campo nombre-->
<!-- muestra el contenido del campo Imagen: -->
<td> </td>
<!-- Link de descarga de la imagen -->
<td> <a href="data:image/jpg;base64,<?php echo base64_encode($row['Imagen']); ?>" target=_blank>Descargar</a></td>
<td><?php echo $row['fecha'];?></td><!-- muestra el contenido del campo fecha -->
</tr>
<?php
}
```

**Código 2.4** Código para mostrar las imágenes con el motor de búsqueda.

En la Figura 2.54 se muestra implementada la interfaz web **Galería**, donde se observa que existen tres opciones: Imágenes NOAA-15, Imágenes NOAA-18 e Imágenes NOAA-19; cada una de estas opciones permite seleccionar las imágenes de los satélites NOAA-15, NOAA-18 y NOAA-19 respectivamente.



**Figura 2.54** Galería de la aplicación web.

Una vez que se seleccione una de las tres opciones que se muestran en la Figura 2.54, se desplegarán todas las imágenes que se encuentren almacenadas en la tabla tabla\_imagenes y que pertenezcan al satélite seleccionado.

Para utilizar el motor de búsqueda, en primer lugar se debe ingresar el rango de fechas y luego pulsar el botón **Buscar**, de esta manera se desplegarán todas aquellas imágenes que se encuentren dentro del rango ingresado.

En la Figura 2.55 se muestra implementada la interfaz web para el caso en el que se utilice el motor de búsqueda por fechas.



**Figura 2.55** Imágenes seleccionadas por el motor de búsqueda.

#### 2.3.5.2.3 Información

En esta opción de la aplicación web se muestra el esquema del prototipo receptor de imágenes meteorológicas, los requisitos que cumple y una explicación de su funcionamiento.

En la Figura 2.56 se muestra implementada la interfaz web **Información**.



**Figura 2.56** Interfaz web Información.

#### 2.3.5.2.4 Estado del Raspberry Pi

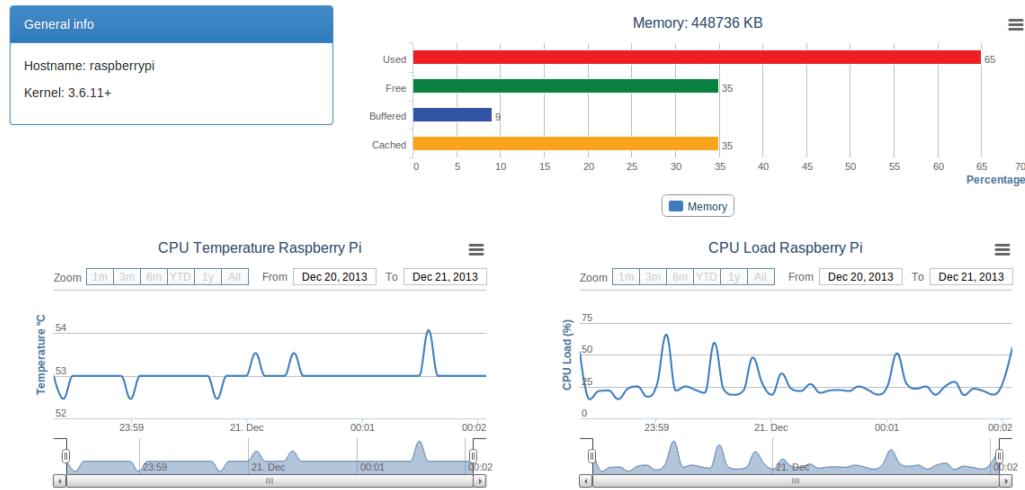
En esta opción de la aplicación web los usuarios pueden observar en tiempo real algunos parámetros de la Raspberry Pi, tales como:

- La versión del *kernel*<sup>30</sup>.
- La temperatura del CPU.
- La distribución de la memoria.
- El procesamiento del CPU.

Para la implementación de esta interfaz web se ha utilizado el código que se encuentra disponible en [55], el cual utiliza *scripts* desarrollados en HTML y NODE.js<sup>31</sup>.

En el Anexo C se encuentra detallado la instalación y configuración de los paquetes necesarios para el correcto funcionamiento de NODE.js.

En la Figura 2.57 se muestra el panel de monitorización que se podrá observar en la interfaz web **Estado del Raspberry Pi**.

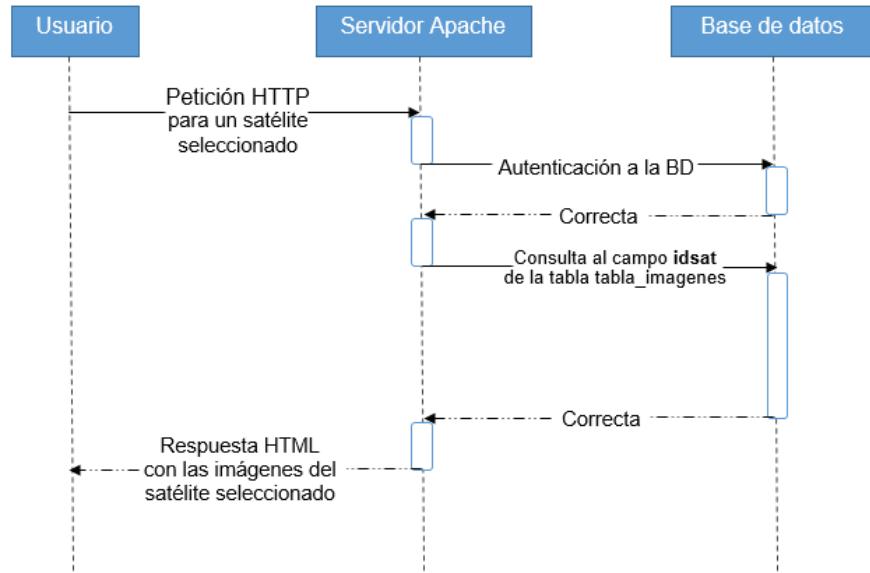


**Figura 2.57** Interfaz web Estado del Raspberry Pi.

<sup>30</sup> Kernel, es un software que constituye una parte fundamental del sistema operativo, facilita a los distintos programas acceso seguro al hardware.

<sup>31</sup> NODE.js, es un entorno de ejecución multiplataforma de código abierto que utiliza como lenguaje de programación JavaScript.

De acuerdo al esquema de la Figura 2.34, la placa Raspberry Pi se encuentra en un entorno de red de área local inalámbrico (WLAN), por lo que todos los usuarios que se encuentran en la misma red y deseen ingresar a la aplicación web para observar las imágenes, tendrán que realizar el procedimiento de la Figura 2.58.



**Figura 2.58** Diagrama de secuencia para visualizar las imágenes.

## CAPÍTULO 3: PRUEBAS Y RESULTADOS

### 3.1 PRUEBAS DE FUNCIONALIDAD DEL PROTOTIPO

Las pruebas de funcionalidad del prototipo han sido realizadas en base a los requisitos mencionados en la sección 2.1, por lo tanto, las pruebas se han dividido en cuatro fases.

- 1) Fase de adquisición.
- 2) Fase de decodificación.
- 3) Fase de almacenamiento.
- 4) Fase de visualización.

Para la realización de las pruebas en las diferentes fases, los componentes de *hardware* y *software* que se utilizarán son los siguientes:

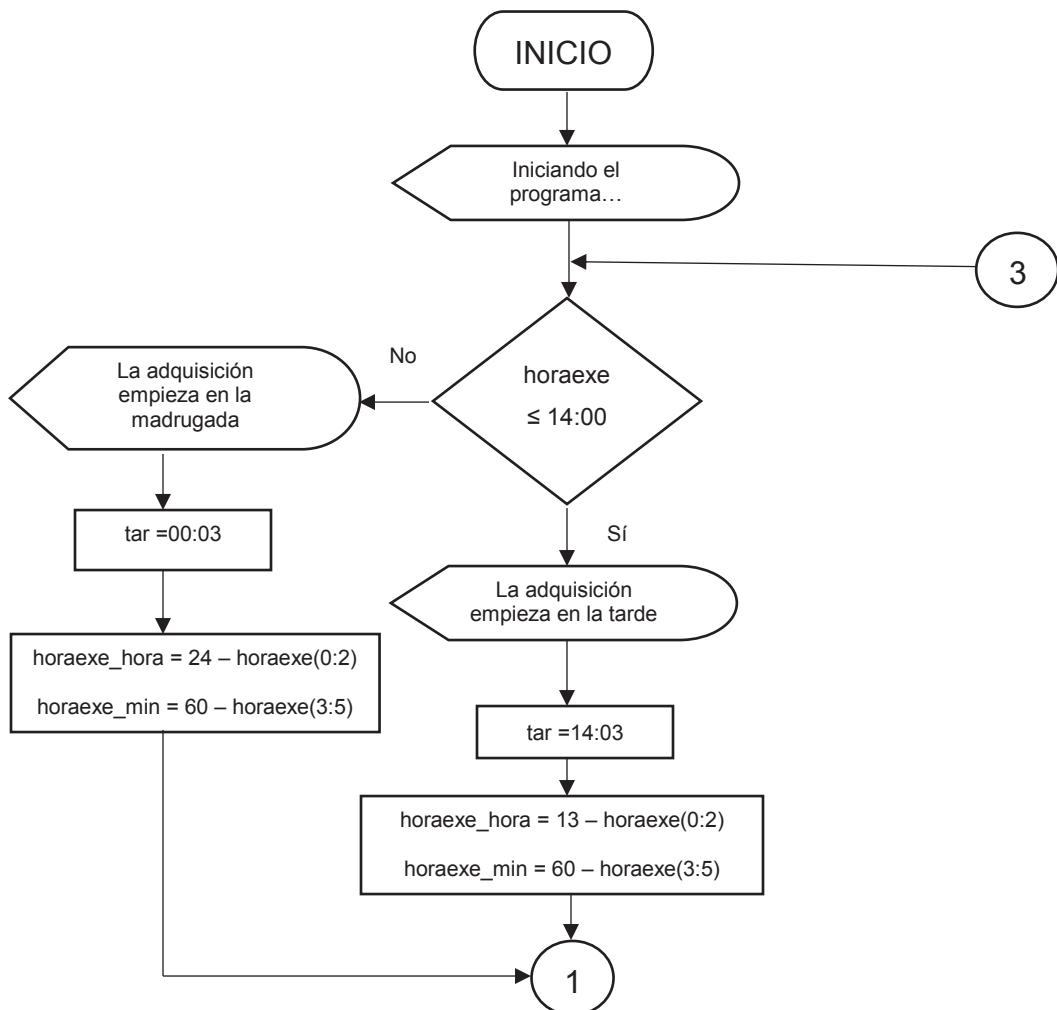
- Componentes de Hardware
  - ✓ Antena (Turnstile o Cuadrifilar Helicoidal)
  - ✓ *Pigtail* BNC hembra a SMA macho
  - ✓ Módulo RTL-SDR
  - ✓ Placa Raspberry Pi
  - ✓ Mouse
  - ✓ Pantalla
  - ✓ Teclado
  - ✓ *Access Point*
- Componentes de Software
  - ✓ GNU RADIO Companion
  - ✓ WXtoImg
  - ✓ Servidor LAMP
  - ✓ Python

Para la realización de las fases de adquisición, decodificación y almacenamiento se ha desarrollado un código utilizando el lenguaje de programación Phyton, el cual permitirá

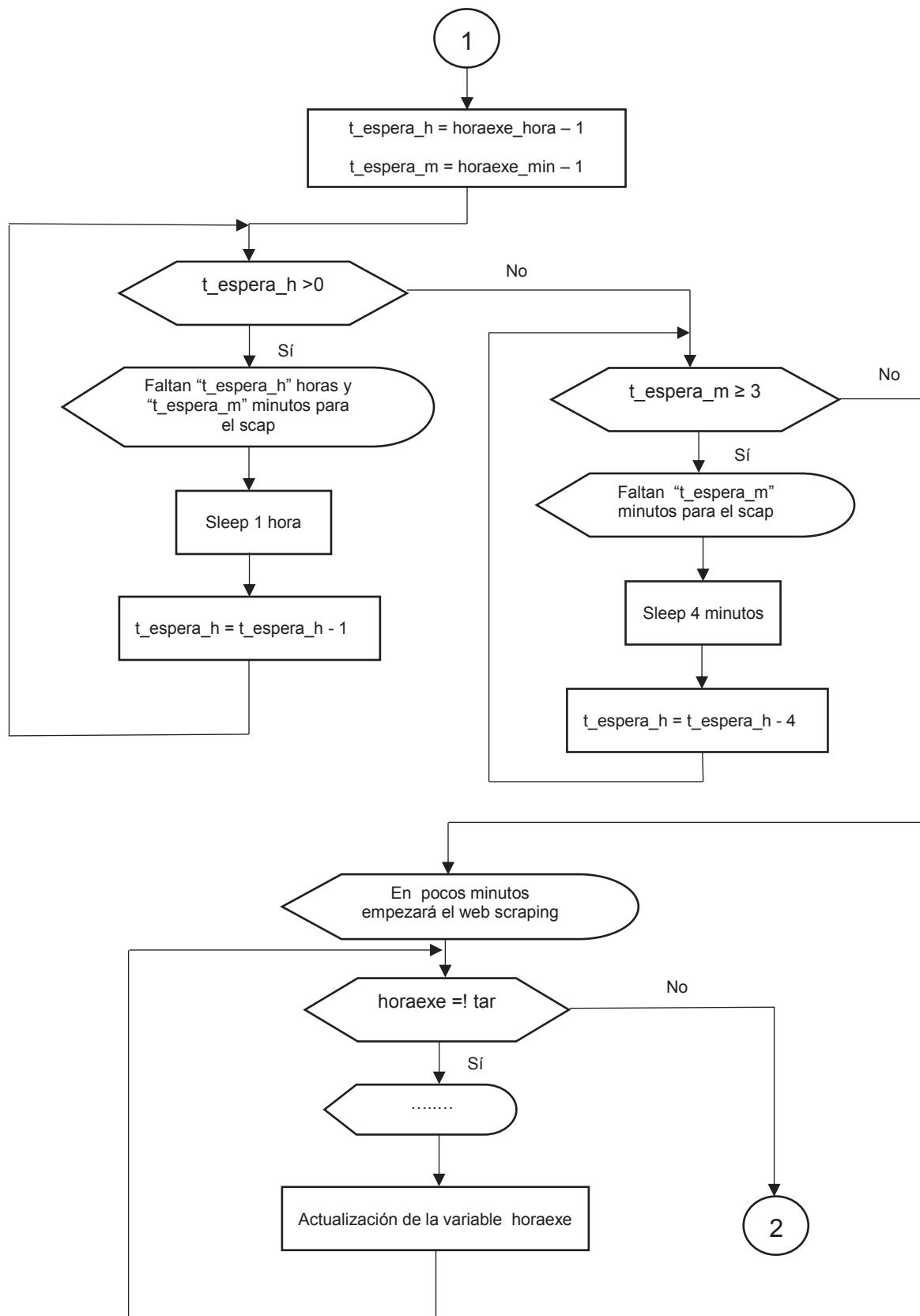
realizar las pruebas de dichas fases de manera automática. El código completo se encuentra en el Anexo E (digital).

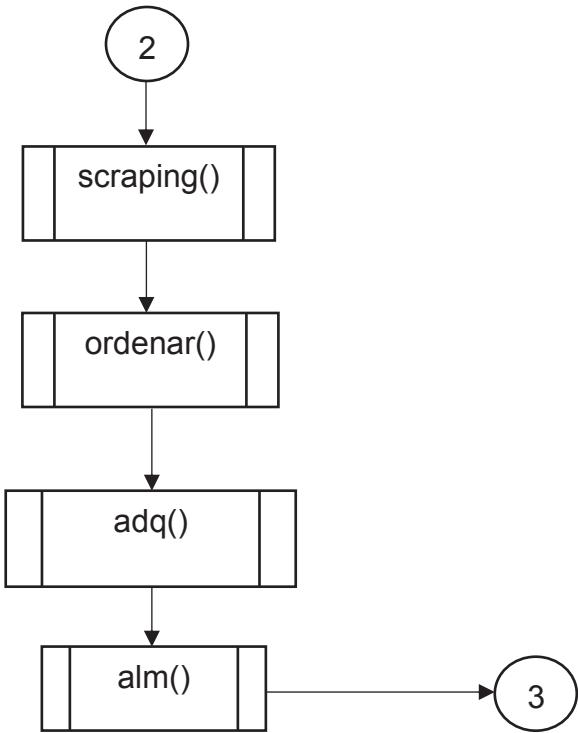
Dado que la hora exacta a la cual los satélites NOAA transmiten sobre el territorio continental ecuatoriano es diferente para cada día, se utiliza la técnica de *web scraping*<sup>32</sup> para extraer la información de la hora exacta de transmisión para cada uno de los satélites NOAA. La página web de la cual se extraen dichas horas es [www.n2yo.com](http://www.n2yo.com).

En la Figura 3.1 se puede observar el diagrama de flujo del código desarrollado para la automatización de las fases de adquisición, decodificación y almacenamiento.



<sup>32</sup> Web scraping, es una técnica utilizada mediante programas de software para extraer información de sitios web.





**Figura 3.1** Diagrama de flujo de automatización.

A continuación se describen de manera general las variables y los procesos que se muestran en el diagrama de flujo de la Figura 3.1.

- Al iniciar el programa, en la variable **horaexe** se almacena la hora del sistema de la Raspberry Pi a la cual se ejecutó el programa, por ejemplo, si el programa se ejecuta a las 11:00 am, en la variable **horaexe** se almacena dicha hora. Luego, en el caso de que la hora almacenada en la variable **horaexe** sea menor a 14:00 se establece que la variable **tar** sea igual a 14:03; caso contrario la variable **tar** es igual a 00:03.
- Las variables **horaexe\_hora**, **horaexe\_min**, **t\_espera\_h** y **t\_espera\_m** permiten establecer cuantas horas y cuantos minutos faltan para realizar el *web scraping*. Por ejemplo, si la variable **horaexe** es igual a 11:00, entonces, la variable **tar** es igual a 14:03; y en pantalla se observará “Faltan 2 horas y 59 minutos para el *web scraping*”. Este tiempo restante para empezar con el *web scraping* se actualiza cada hora, y cuando falte menos de una hora se actualiza cada 4 minutos.

- Cuando falte entre 1 y 3 minutos para la hora almacenada en la variable **tar** aparecerá en pantalla “En pocos minutos empezará el *web scraping*”. Luego, se actualiza la variable **horaexe** cada 25 segundos y se compara con la variable **tar**, y cuando estas dos variables sean iguales se pasa al proceso de **scraping**.
- En el proceso **scraping** se obtiene para cada uno de los satélites NOAA la hora en la cual se encontrarán transmitiendo sobre el territorio continental ecuatoriano, es decir, en este proceso se obtienen tres diferentes horas, cada una de ellas correspondiente a cada uno de los tres satélites NOAA.
- En el proceso **ordenar** se ordena de menor a mayor las tres horas obtenidas en el proceso anterior mediante la técnica de *web scraping*. Este ordenamiento se realiza a fin de que los siguientes procesos se ejecuten de forma ordenada.
- Luego de tener las tres horas ordenadas de menor a mayor, en el proceso **adq** se establece cuantas horas y minutos faltan para pasar el proceso **alm**, esto se establece en base a la hora actual del sistema de la Raspberry Pi y las tres horas obtenidas.
- El proceso **alm** se ejecuta en tres ocasiones, es decir, en cada una de las tres horas obtenidas y ordenadas por los procesos anteriores. En el proceso **alm** en primer lugar se ejecuta el archivo de GNU Radio Companion que realiza la adquisición de la señal en formato APT, ver Figura 2.48, luego, realiza la decodificación mediante la ejecución del comando del software WXtoImg establecido anteriormente en la sección 2.3.4, y por último se realiza el almacenamiento de las imágenes dentro de la base de datos.

Las imágenes se encuentran disponibles en la aplicación web implementada aproximadamente 15 minutos después de ejecutarse el proceso **alm**.

Una vez finalizado por completo el proceso **alm**, se repite todo el procedimiento mencionado, es decir que el código implementado permite obtener las imágenes meteorológicas todos los días, sin la necesidad de ejecutarlo cada día.

### 3.1.1 FASE DE ADQUISICIÓN

Las pruebas a realizarse en esta fase tienen como objetivo la comprobación de que la señal en formato APT (*Automatic Picture Transmission*) transmitida por los satélites NOAA está siendo adquirida correctamente.

Como se mencionó en la sección 2.3.1, para el presente proyecto se construyeron 2 antenas, por lo tanto se realizarán las pruebas tanto con la antena cuadrifilar helicoidal como con la antena turnstile.

Cabe recalcar que cada satélite NOAA transmite la señal en formato APT a una frecuencia diferente, ver Tabla 1.5, por lo tanto, para la realización de las pruebas el parámetro “Frecuencia central” del bloque “RTL-SDR Source” tomará el valor correspondiente a la frecuencia de transmisión de la señal APT de cada satélite, tal como se mencionó en la sección 2.3.3.

Previamente a la adquisición de la señal en formato APT se obtendrá la hora a la cual los satélites NOAA transmiten sobre el territorio continental ecuatoriano; para el caso del satélite NOAA-15 dicha hora se obtiene mediante el Código 3.1, el cual es un fragmento del código de automatización.

```

url15 = "http://www.n2yo.com/passes/?s=25338&a=1"
req15 = requests.get(url15)
# guarda el código de respuesta de la url15
conexion15 = req15.status_code
a = 1
while a < 6:#en caso de que no exista internet lo intenta 5 veces
    #la petición del navegador se ha completado con éxito
    if conexion15 == 200:
        print("Si hay respuesta de la web para el NOAA-15")
        try:
            browser = webdriver.Firefox()
            #ejecuta scripts y funciones mediante Firefox
            data = browser.get(url15)
            # se obtiene el dato de la columna y fila 1 de la tabla con id = passestable
            datos3 = browser.find_element_by_xpath("""//*[@id="passestable"]/tbody/tr[1]/td[1]""").text
        except:
            #Cuando no existe conexión a internet
            print("Error al realizar web scraping del NOAA-15")
        dig1 = time.strftime("%d")
        if int(dig1) < 10 :#para fechas menores a 10
            fecha3 =(datos3)[0:5]#obtiene la fecha
            hora3 = (datos3)[6:13]#obtiene la hora
        else:#para fechas mayores a 10
            fecha3 = (datos3)[0:6]#obtiene la fecha
            hora3 = (datos3)[7:14]#obtiene la hora
        print(fecha3+"-->"+hora3)
        browser.close()# cierra el navegador Firefox
        break
    else:
        print("Revisar la conexión de Internet")
    a=a+1

```

**Código 3.1** Código del *web scraping* para el NOAA-15.

En la Figura 3.2 se puede observar la página web y la hora que se obtendrá luego de ejecutar el Código 3.1, para el caso del satélite NOAA-15. Así también, en la Figura 3.3 y Figura 3.4 se muestra la página web y las horas que se obtendrán para los satélites NOAA-18 y NOAA-19 respectivamente. Dichas horas son en las que cada uno de los satélites NOAA se encontrarán transmitiendo sobre el territorio continental ecuatoriano.

N2YO.com Tracking 17857 objects as of 24-October-2016 HD Live streaming from Space Station objects crossing your sky now

**10-DAY PREDICTIONS**

All passes		AM/PM time			UTC			Visible passes		
		Start ↑			Max altitude			End ↓		
Date, Local time	Az	Local time	Az	Elt	Local time	Az	Mag	Info		
24-Oct 18:27	S 178°	18:34	W 262°	49°	18:41	NNW 336°	+6.1	<a href="#">Map and details</a>		
25-Oct 05:25	NE 33°	05:32	E 100°	29°	05:38	S 171°	+6.8	<a href="#">Map and details</a>		
25-Oct 18:02	S 156°	18:09	E 90°	83°	18:16	N 349°	+5.6	<a href="#">Map and details</a>		
	NE		E			NE				

Hora de transmisión para el NOAA-15

Figura 3.2 Fecha y hora para el satélite NOAA-15.

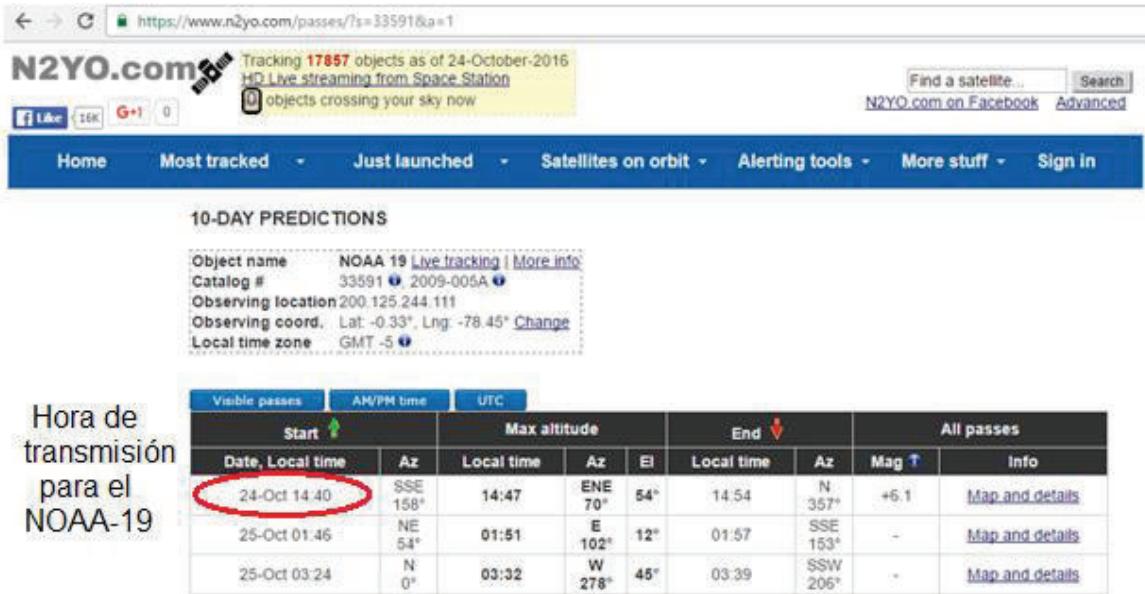
N2YO.com Tracking 17857 objects as of 24-October-2016 HD Live streaming from Space Station objects crossing your sky now

**10-DAY PREDICTIONS**

All passes		AM/PM time			UTC			Visible passes		
		Start ↑			Max altitude			End ↓		
Date, Local time	Az	Local time	Az	Elt	Local time	Az	Mag	Info		
24-Oct 18:42	S 174°	18:50	WSW 254°	62°	18:57	NNW 340°	+5.4	<a href="#">Map and details</a>		
25-Oct 05:47	NE 34°	05:54	E 100°	30°	06:01	S 172°	+6.4	<a href="#">Map and details</a>		
25-Oct 18:30	S 140°	18:38	SW 220°	84°	18:45	N 348°	+5.1	<a href="#">Map and details</a>		
	NE		E			NE				

Hora de transmisión para el NOAA-18

Figura 3.3 Fecha y hora para el satélite NOAA-18.



**Figura 3.4** Fecha y hora para el satélite NOAA-19.

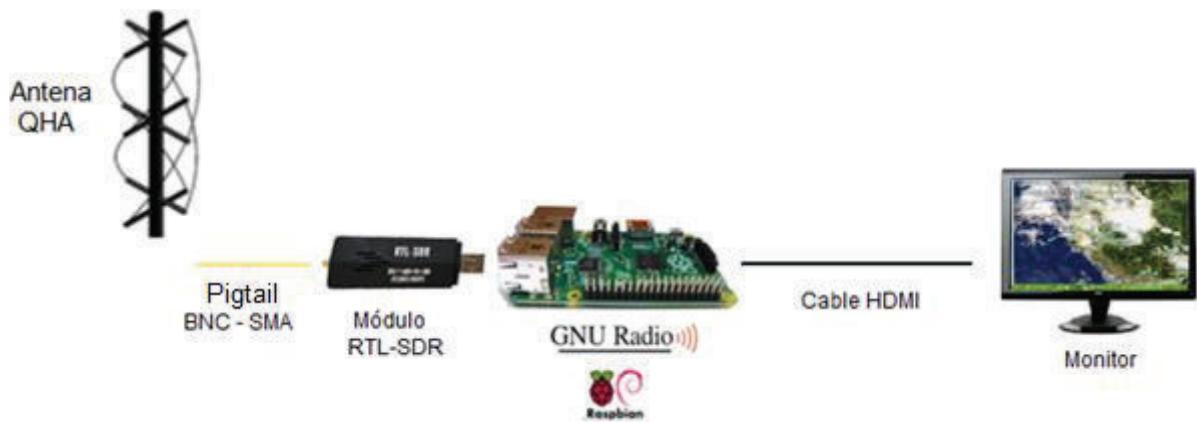
Cabe mencionar, que la placa Raspberry Pi necesita una conexión a Internet para que se pueda obtener el dato de la hora de cada uno de los satélites NOAA.

### 3.1.1.1 Pruebas con la antena cuadrifilar helicoidal

Como se puede observar en la Figura 3.2, Figura 3.3 y Figura 3.4 las tres horas correspondientes a la transmisión sobre el territorio continental ecuatoriano de los satélites NOAA-15, NOAA-18 y NOAA-19 respectivamente para el 24 de Octubre del 2016 son las siguientes:

- NOAA-15: 18:27
- NOAA-18: 18:42
- NOAA-19: 14:40

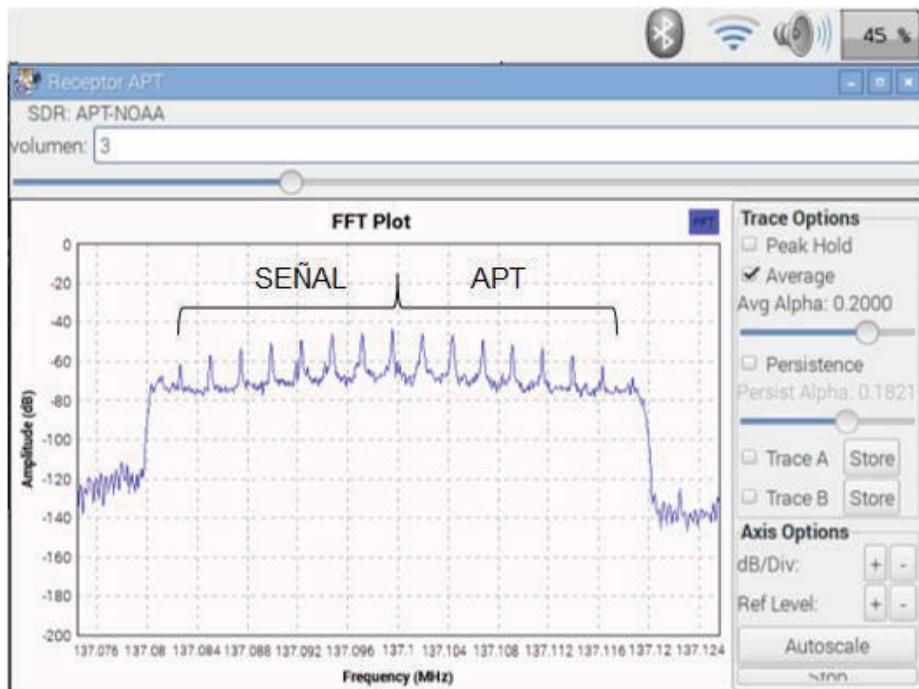
En la Figura 3.5 se muestra el esquema de conexión de los diferentes elementos que conforman el prototipo para la realización de las pruebas con la antena cuadrifilar helicoidal.



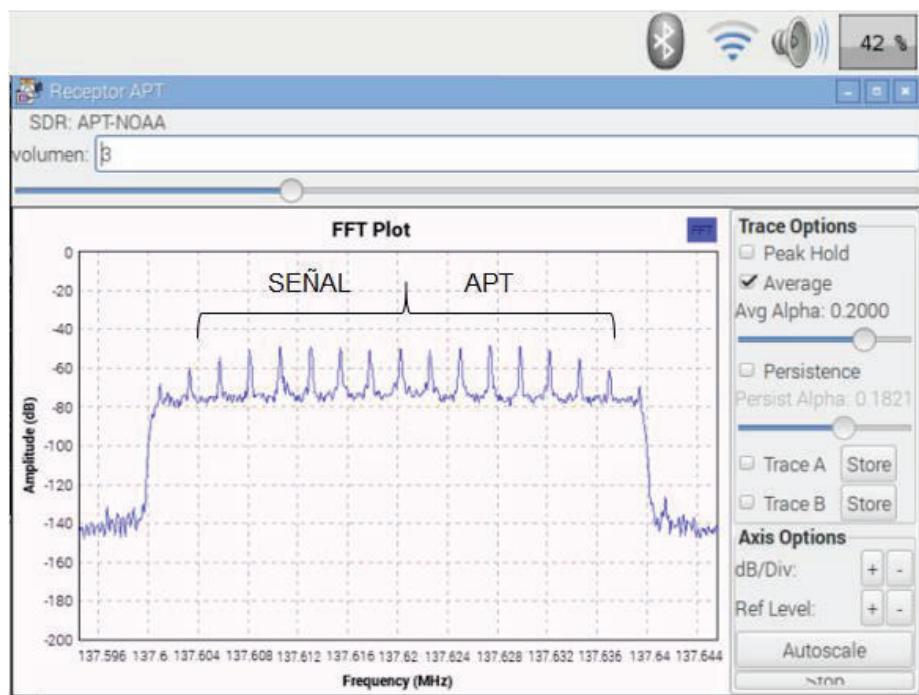
**Figura 3.5** Esquema para pruebas con la antena QHA.

Mediante la ejecución del esquema realizado en la aplicación GNU Radio Companion, ver Figura 2.48, se comprueba si la señal APT de los satélites NOAA está siendo adquirida correctamente.

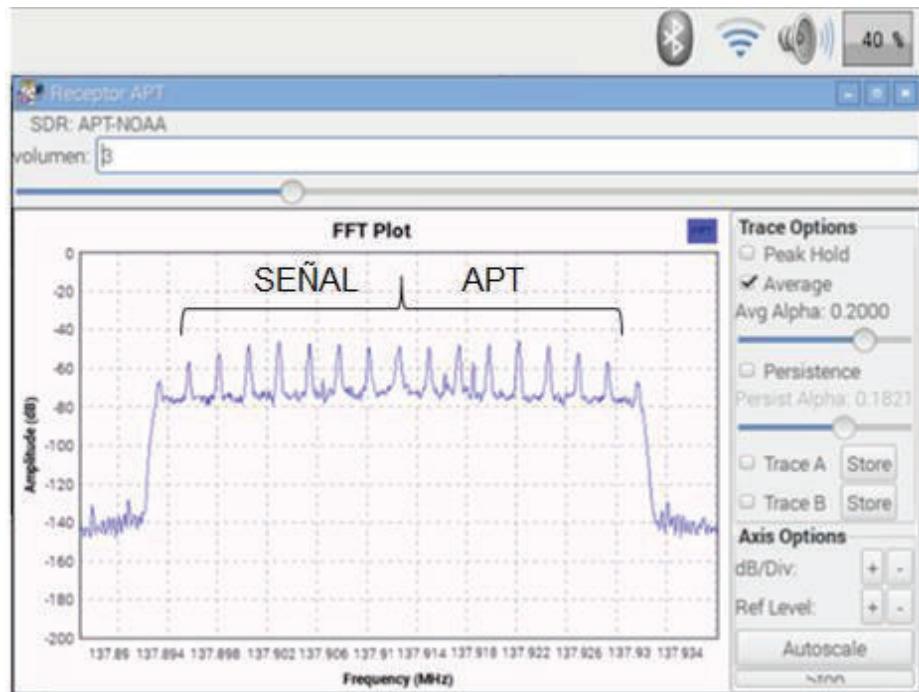
En la Figura 3.6, Figura 3.7 y Figura 3.8 se muestra la señal que está siendo adquirida de los satélites NOAA-19, NOAA-15 y NOAA-18 respectivamente; estas imágenes se obtienen luego de que la señal pasa por el filtro pasa bajos.



**Figura 3.6** Señal APT del satélite NOAA-19.



**Figura 3.7** Señal APT des satélite NOAA-15.



**Figura 3.8** Señal APT del satélite NOAA-18.

Como se puede observar en la Figura 3.6, Figura 3.7 y Figura 3.8 la señal en formato APT está siendo adquirida adecuadamente por el sistema prototipo.

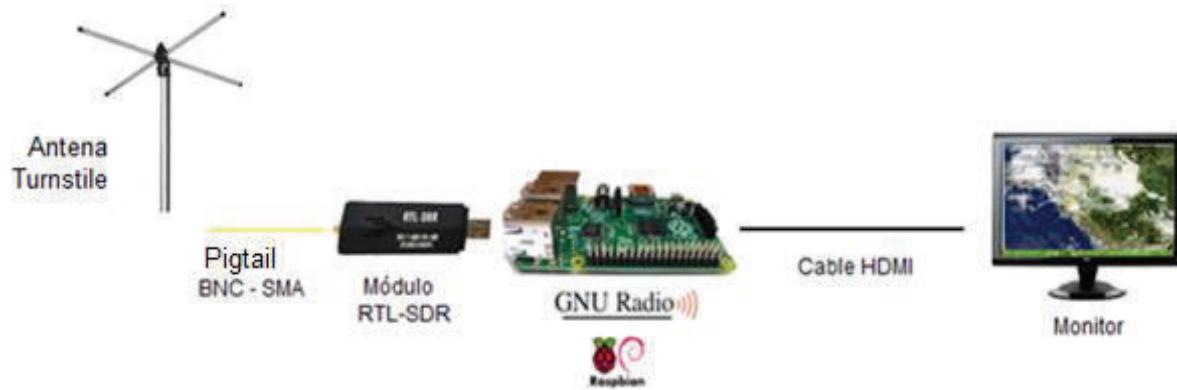
### 3.1.1.2 Pruebas con la antena turnstile

Debido a que existe un solo prototipo, las pruebas con la antena turnstile se realizaron en un día diferente de las pruebas realizadas con la antena cuadrifilar helicoidal.

Las tres horas obtenidas para el 26 de Octubre del 2016 para cada uno de los tres satélites NOAA son las siguientes:

- NOAA-19: 14:17
- NOAA-18: 18:19
- NOAA-15: 19:18

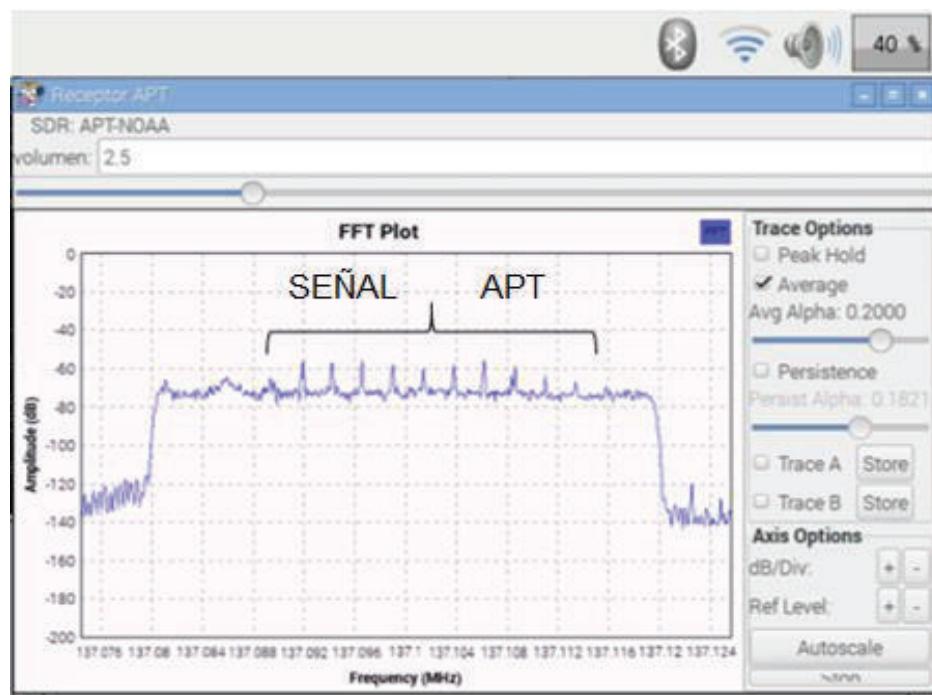
En la Figura 3.9 se muestra el esquema de conexión de los elementos que conforman el prototipo para las pruebas a realizarse con la antena turnstile.



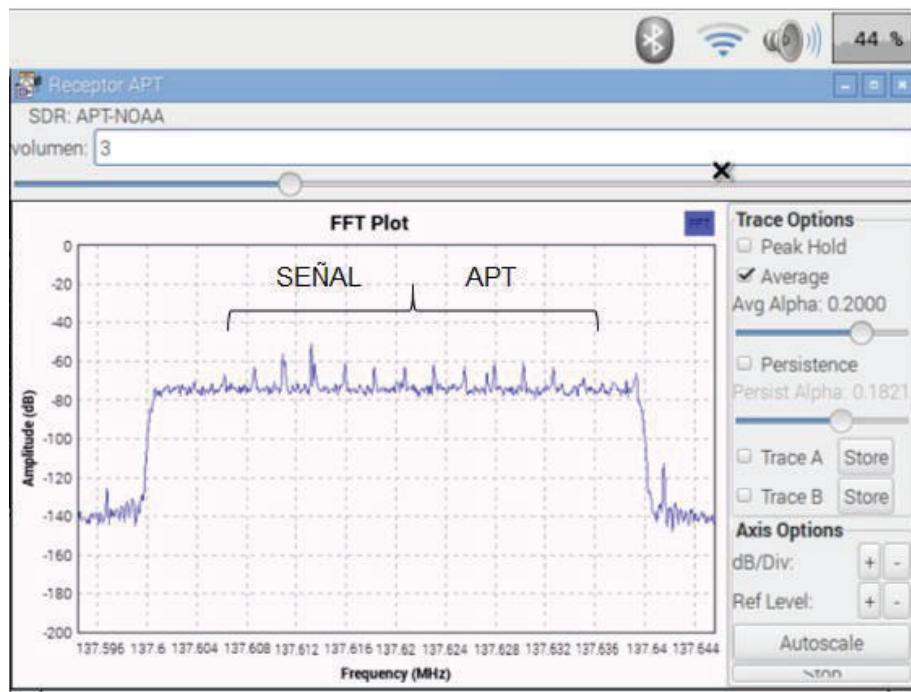
**Figura 3.9** Esquema para pruebas con la antena turnstile.

Al igual que en las pruebas con la antena cuadrifilar helicoidal la comprobación de que la señal APT está siendo adquirida correctamente se realiza mediante la ejecución del esquema realizado en la aplicación GNU Radio Companion, ver Figura 2.48.

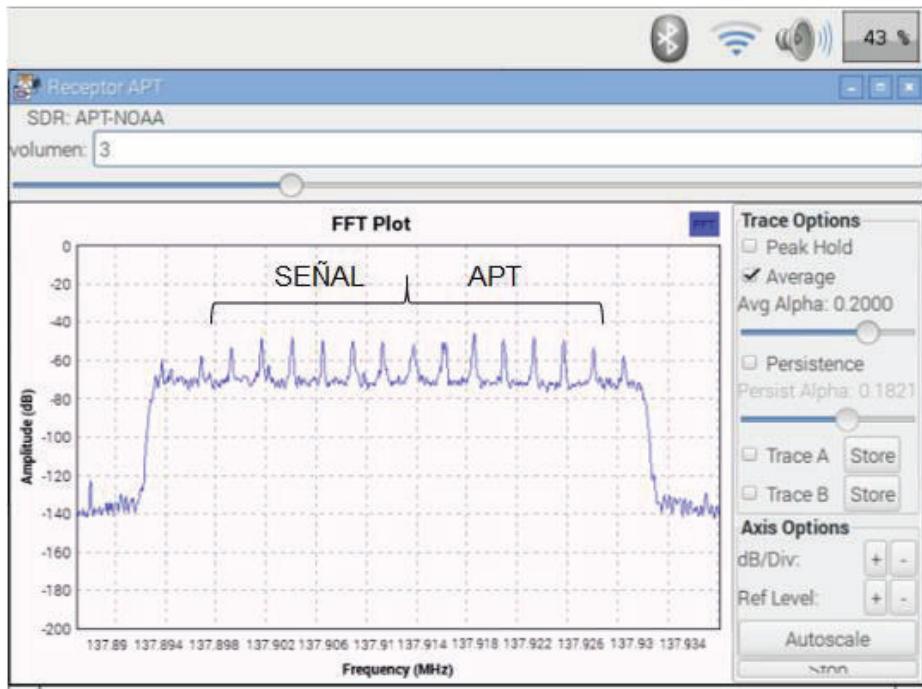
En la Figura 3.10, Figura 3.11 y Figura 3.12 se muestra la señal que está siendo adquirida de los satélites NOAA-19, NOAA-15 y NOAA-18 respectivamente, estas imágenes se obtienen luego de que la señal pasa por el filtro pasa bajos.



**Figura 3.10** Señal APT del satélite NOAA-19.



**Figura 3.11** Señal APT del satélite NOAA-15.



**Figura 3.12** Señal APT del satélite NOAA-18.

Como se puede observar en la Figura 3.10, Figura 3.11 y Figura 3.12 la señal en formato APT está siendo adquirida, pero con mayor degradación que con la antena cuadrifilar helicoidal.

Cabe mencionar que las imágenes de la Figura 3.6, Figura 3.7, Figura 3.8, Figura 3.10, Figura 3.11 y Figura 3.12, fueron capturadas a los 6 minutos de haber empezado a realizar la adquisición de la señal en formato APT, es decir, aproximadamente a la mitad del tiempo total en el que se realiza la adquisición, tal como se mencionó en la sección 2.3.3.

Cabe recalcar, que en base al esquema de la Figura 2.48, la señal modulada en amplitud (AM) que contiene la información obtenida por el sensor AVHRR/3 está siendo almacenada como un archivo de audio cuya extensión es .WAV, su frecuencia de muestreo es de 11025 Hz y tiene un tamaño en promedio de 15 MB, por lo tanto, otra forma de comprobar que la señal en formato APT está siendo adquirida es mediante la reproducción de este archivo de audio.

### 3.1.2 FASE DE DECODIFICACIÓN

El objetivo de esta fase es la decodificación del archivo de audio de extensión .wav que contiene la información del sensor AVHRR/3 y que ha sido adquirido previamente por la aplicación GNU Radio Compaion.

En primer lugar, se realiza la decodificación para las señales adquiridas con la antena cuadrifilar helicoidal y posteriormente con la antena turnstile.

El Código 3.2 es otro fragmento del código de automatización, el cual se ejecuta en la *shell* de la Raspberry Pi y permite mediante el *software* WXtoImg obtener la imagen del canal A, la imagen del canal B y la imagen térmica, tal como se mencionó en la sección 2.3.4.

```
# decodificacion para el canal A
os.system("wxtoimg -a NOAA_audio.wav canala.png")
# decodificacion para el canal B
os.system("wxtoimg -b NOAA_audio.wav canalb.png")
# decodificacion para mostrar la imagen termica
os.system("wxtoimg -e therm NOAA_audio.wav canale.png")
```

**Código 3.2** Comando para la decodificación de la señal en formato APT.

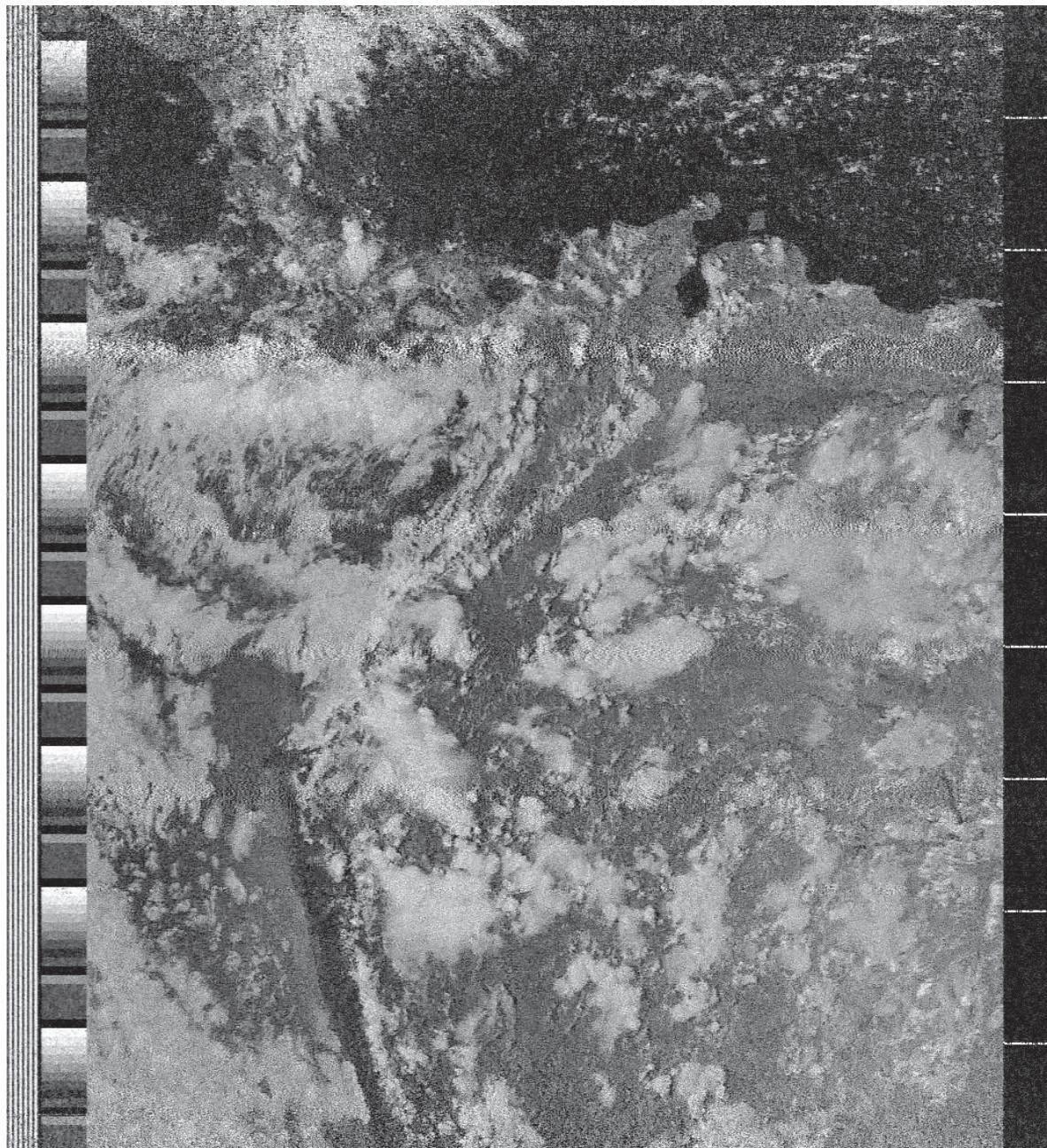
Las imágenes obtenidas son de extensión .png debido a que este es un formato en el que no se realiza una compresión de los datos y no se presenta pérdidas de información.

Cabe recalcar que la señal en formato APT utiliza 1/3 de la información generada por el sensor AVHRR/3 siendo un formato de baja resolución, tal como se mencionó en la sección 1.4.1.4.

Las imágenes obtenidas tanto con la antena cuadrifilar helicoidal como con la antena turnstile, que se muestran en el presente proyecto, son de calidad SD (*Standard Definition*).

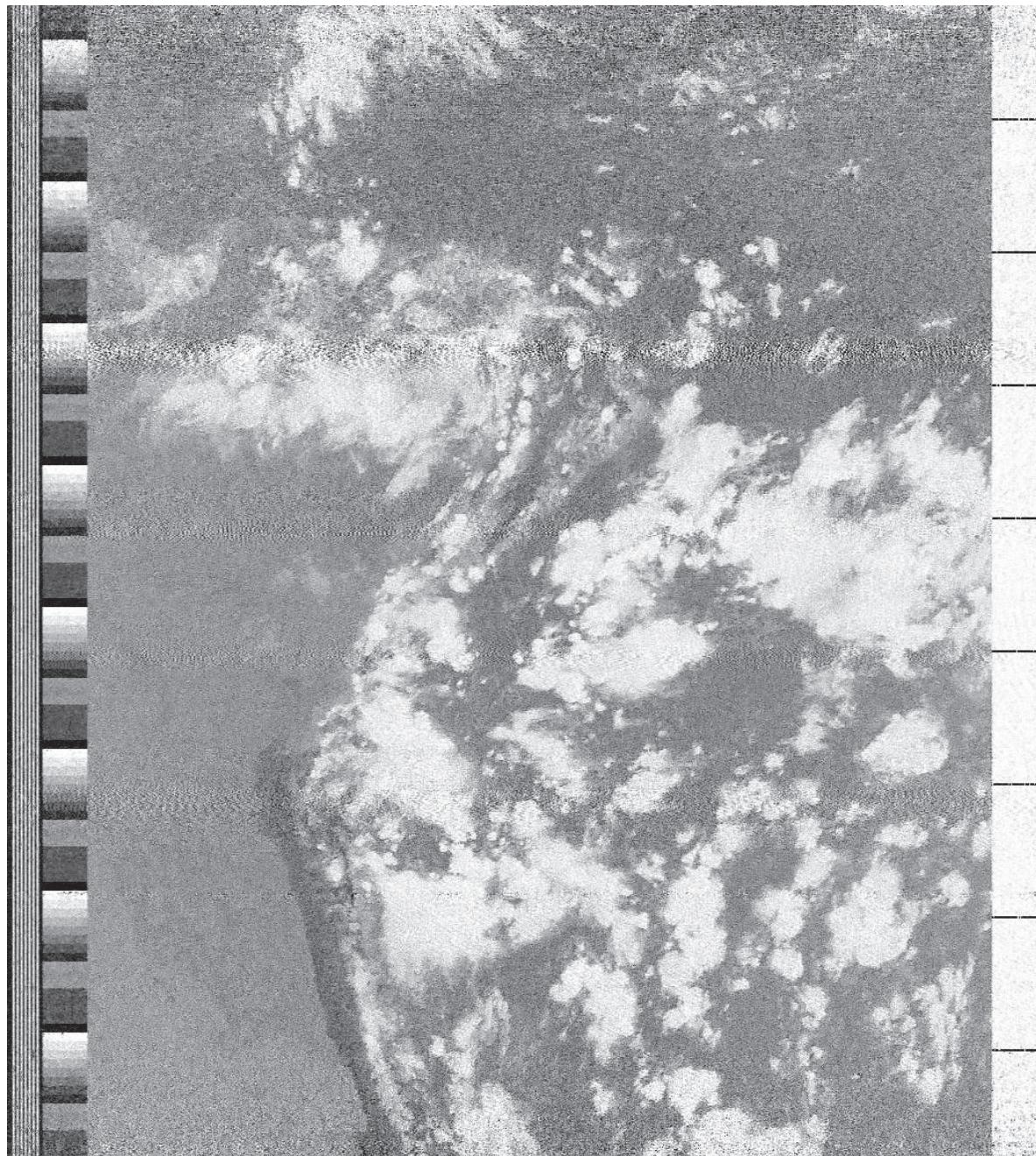
A continuación, se muestran las imágenes generadas luego de realizar la decodificación de las tres diferentes señales adquiridas con la antena cuadrifilar helicoidal y empleando el esquema de la Figura 2.48.

En la Figura 3.13 se muestra una imagen del canal A proveniente del satélite NOAA-19.



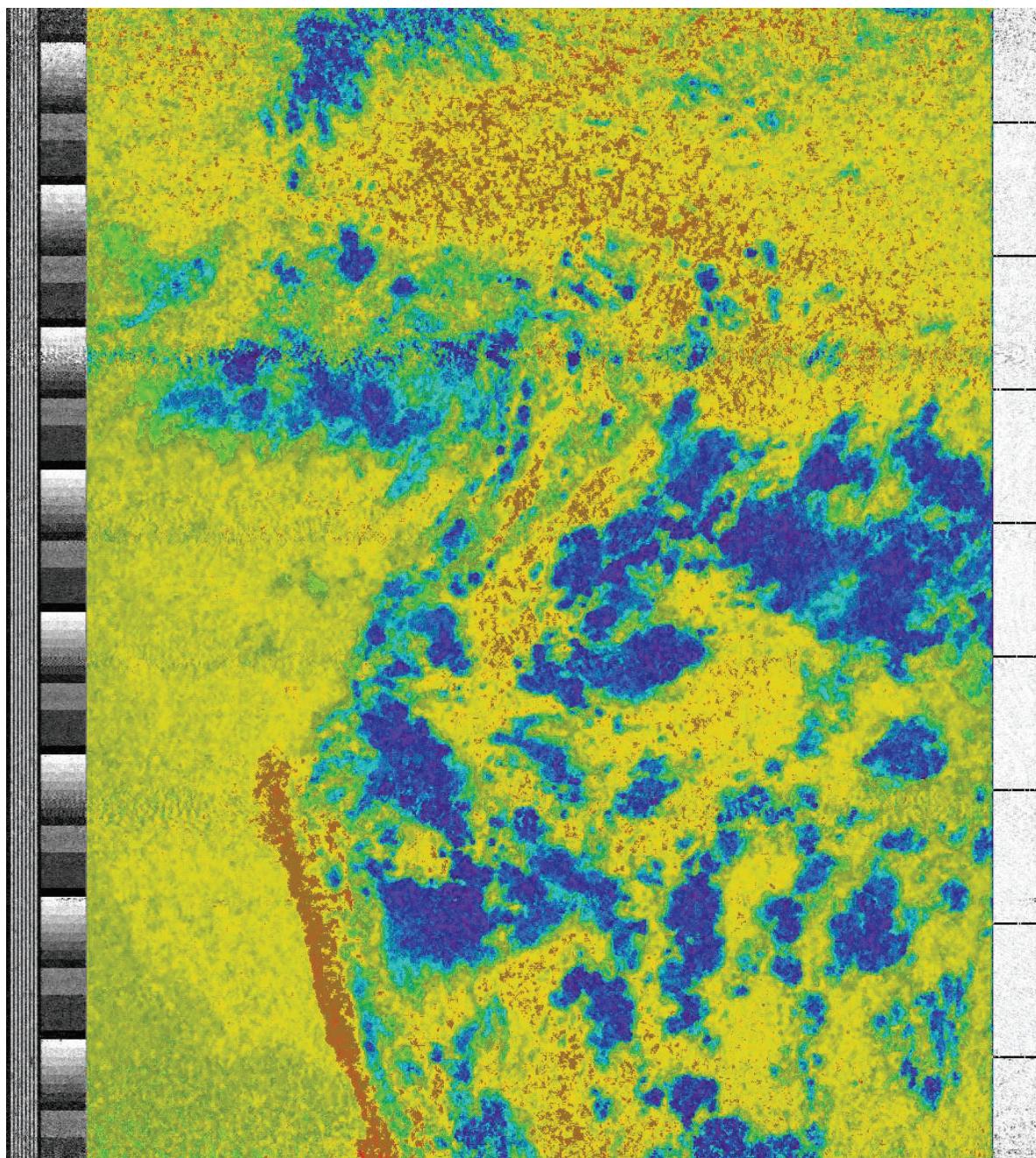
**Figura 3.13** Imagen del canal A del satélite NOAA-19.

En la Figura 3.14 se muestra una imagen del canal B proveniente del satélite NOAA-19.



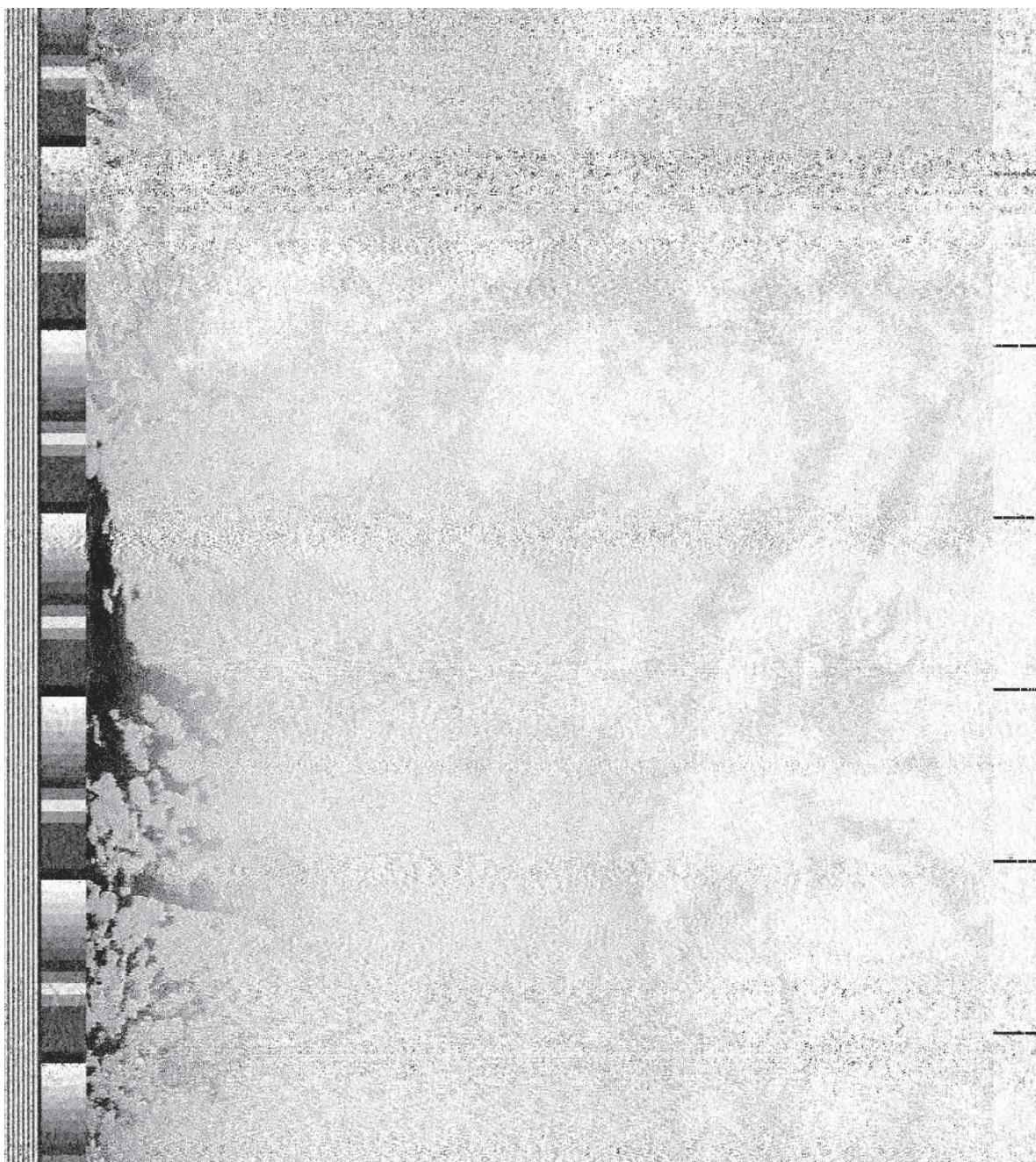
**Figura 3.14** Imagen del canal B del satélite NOAA-19.

En la Figura 3.15 se muestra una imagen térmica proveniente del satélite NOAA-19.



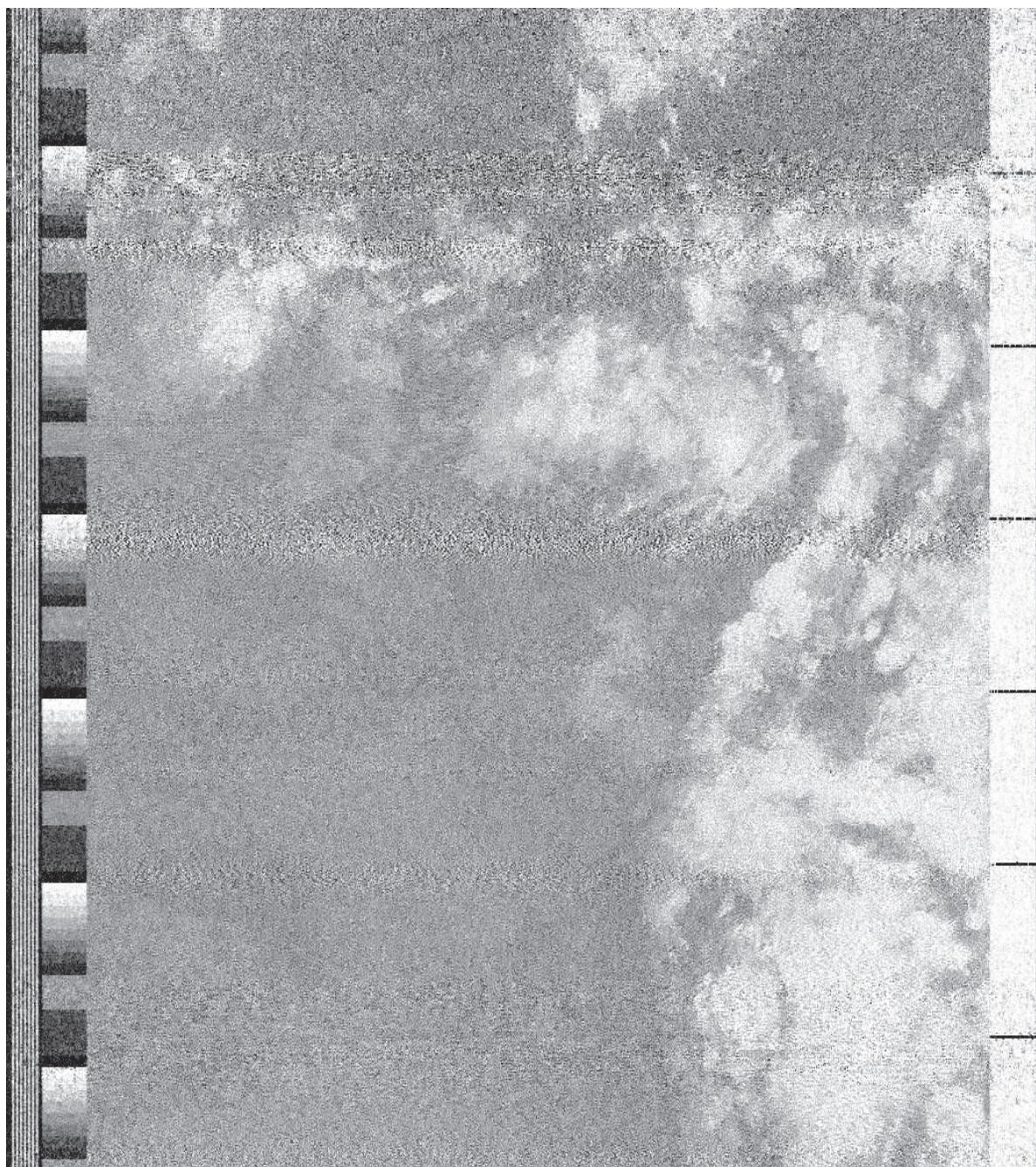
**Figura 3.15** Imagen térmica proveniente de satélite NOAA-19.

En la Figura 3.16 se muestra una imagen del canal A proveniente de satélite NOAA-15.



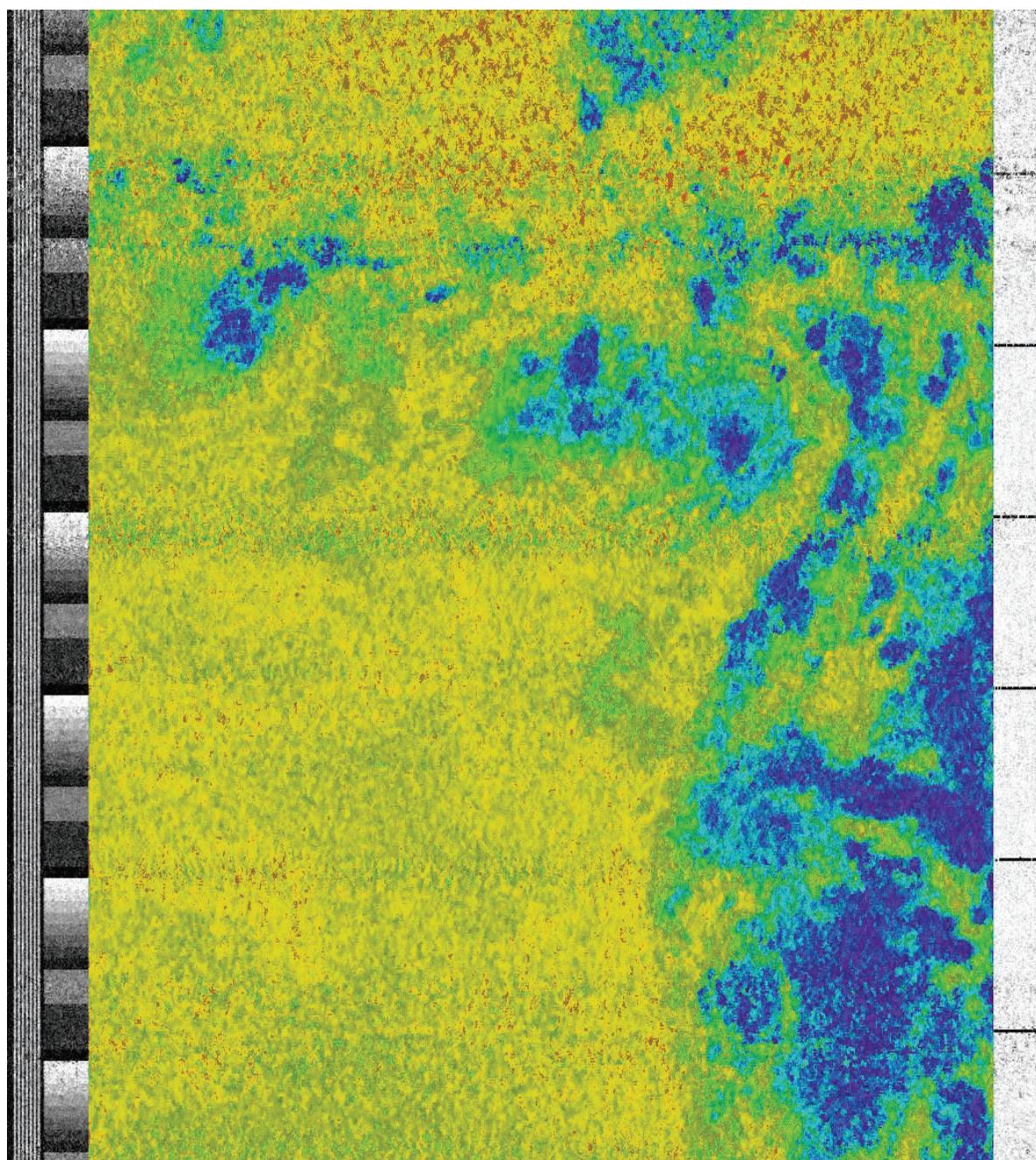
**Figura 3.16** Imagen del canal A del satélite NOAA-15.

En la Figura 3.17 se muestra una imagen del canal B proveniente del satélite NOAA-15.



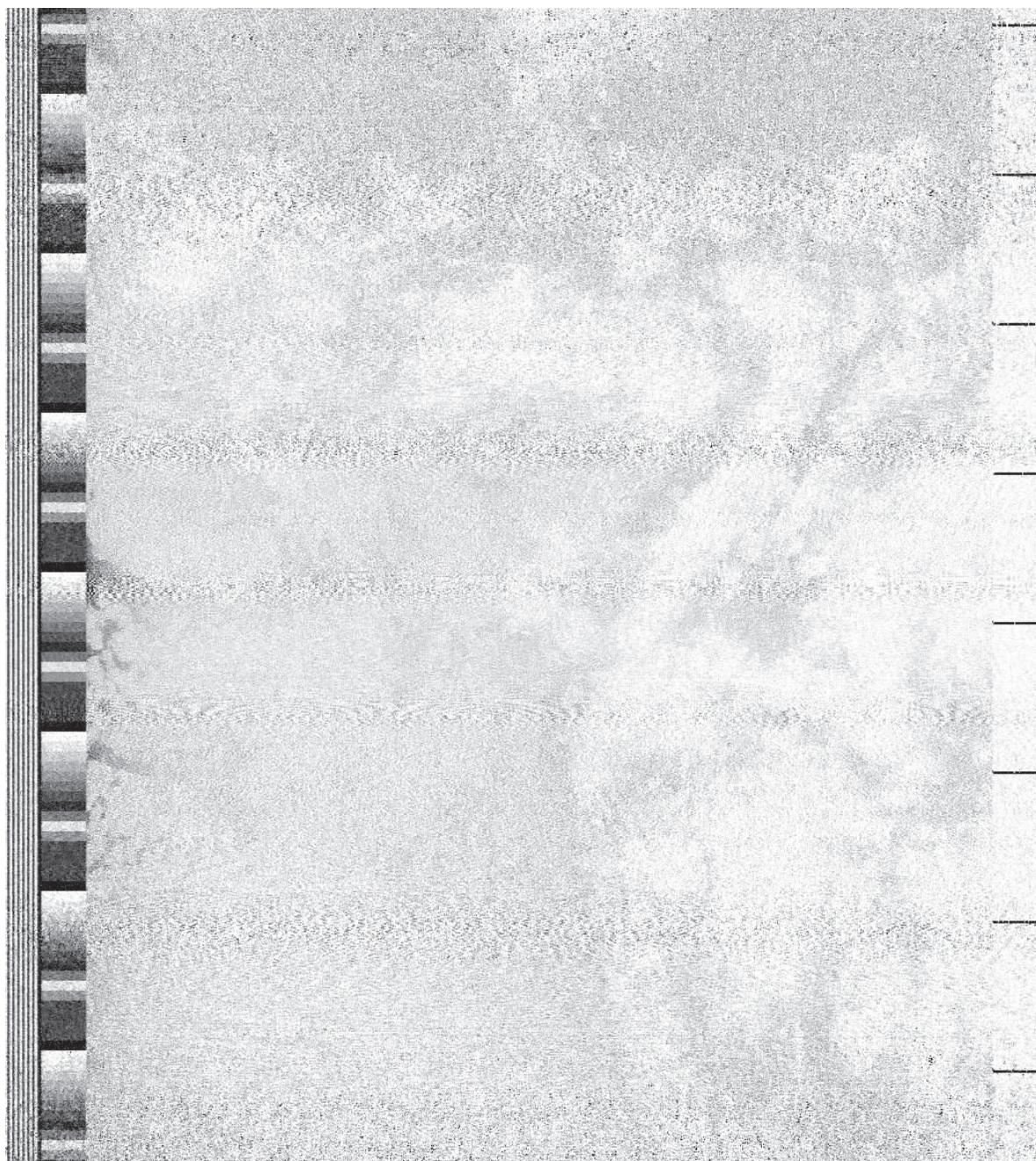
**Figura 3.17** Imagen del canal B del satélite NOAA-15.

En la Figura 3.18 se muestra una imagen térmica proveniente del satélite NOAA-15.



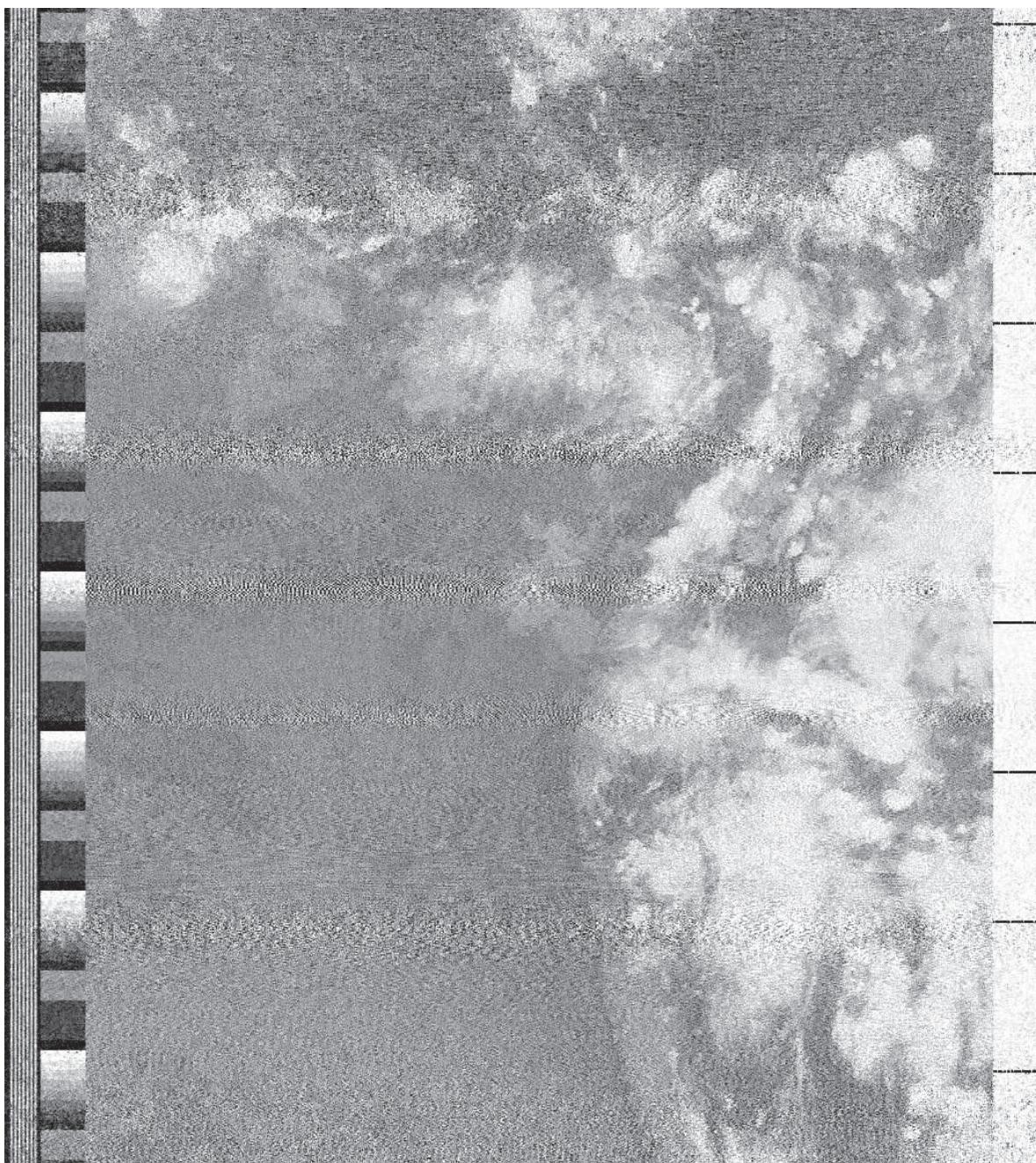
**Figura 3.18** Imagen térmica proveniente de satélite NOAA-15.

En la Figura 3.19 se muestra una imagen del canal A proveniente del satélite NOAA-18.



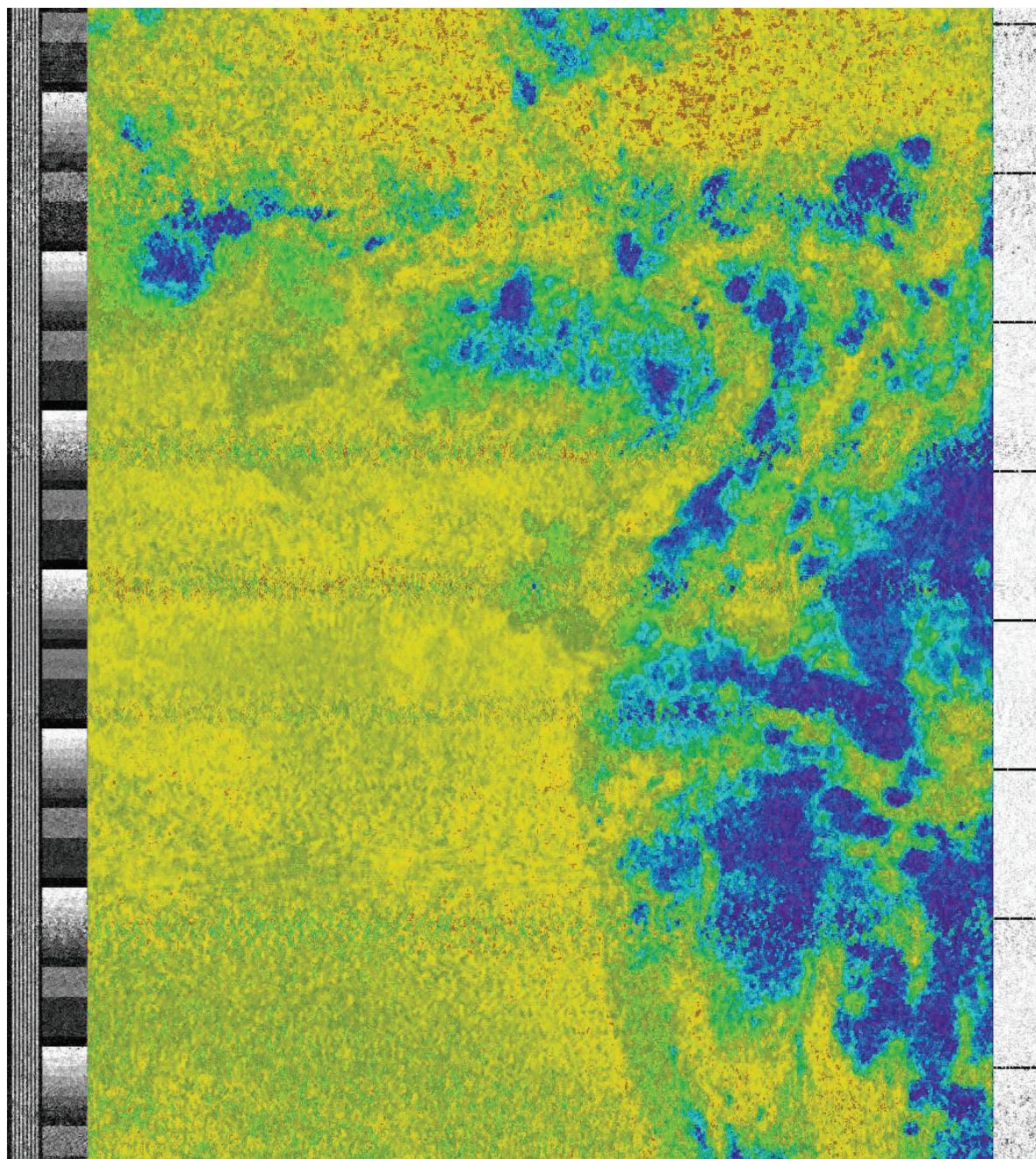
**Figura 3.19** Imagen del canal A del satélite NOAA-18.

En la Figura 3.20 se muestra una imagen del canal B proveniente del satélite NOAA-18.



**Figura 3.20** Imagen del canal B del satélite NOAA-18.

En la Figura 3.21 se muestra una imagen térmica proveniente del satélite NOAA-18.



**Figura 3.21** Imagen térmica proveniente de satélite NOAA-18.

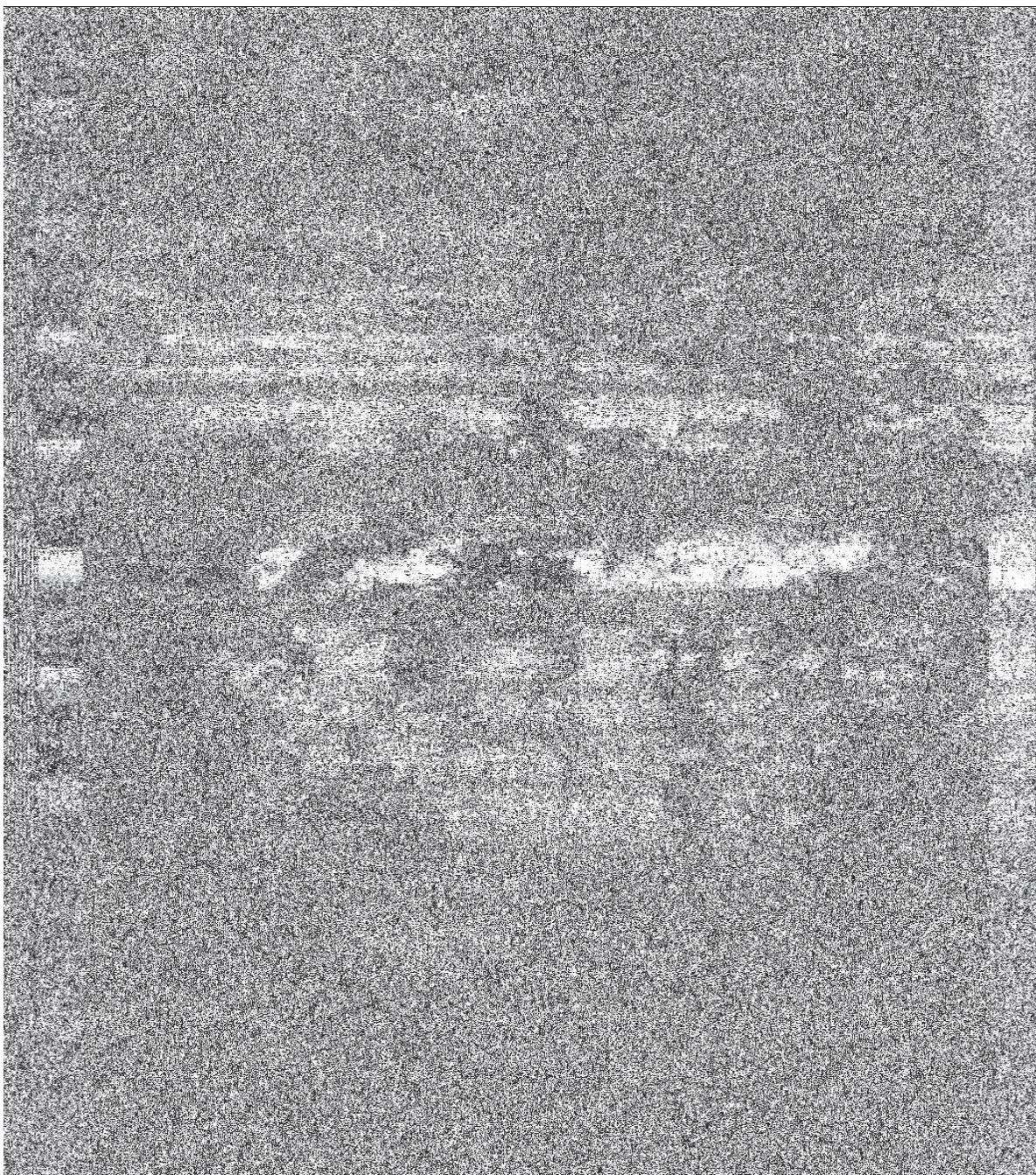
A continuación se muestran las imágenes generadas luego de realizar la decodificación de las señales adquiridas con la antena turnstile conjuntamente con el esquema de la Figura 2.48.

En la Figura 3.22 se muestra una imagen del canal A proveniente del satélite NOAA-19.



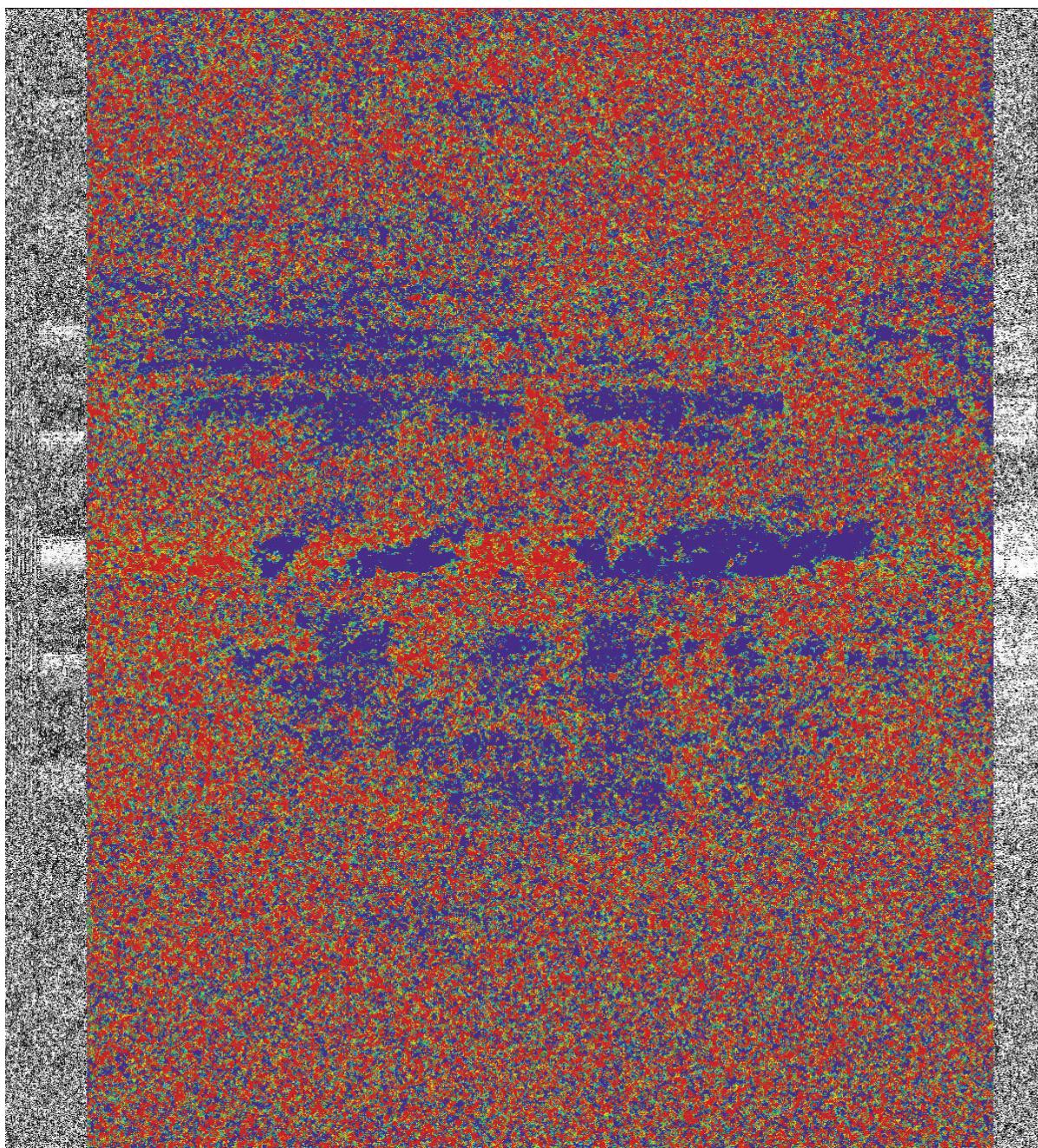
**Figura 3.22** Imagen del canal A proveniente del satélite NOAA-19.

En la Figura 3.23 se muestra una imagen del canal B proveniente del satélite NOAA-19.



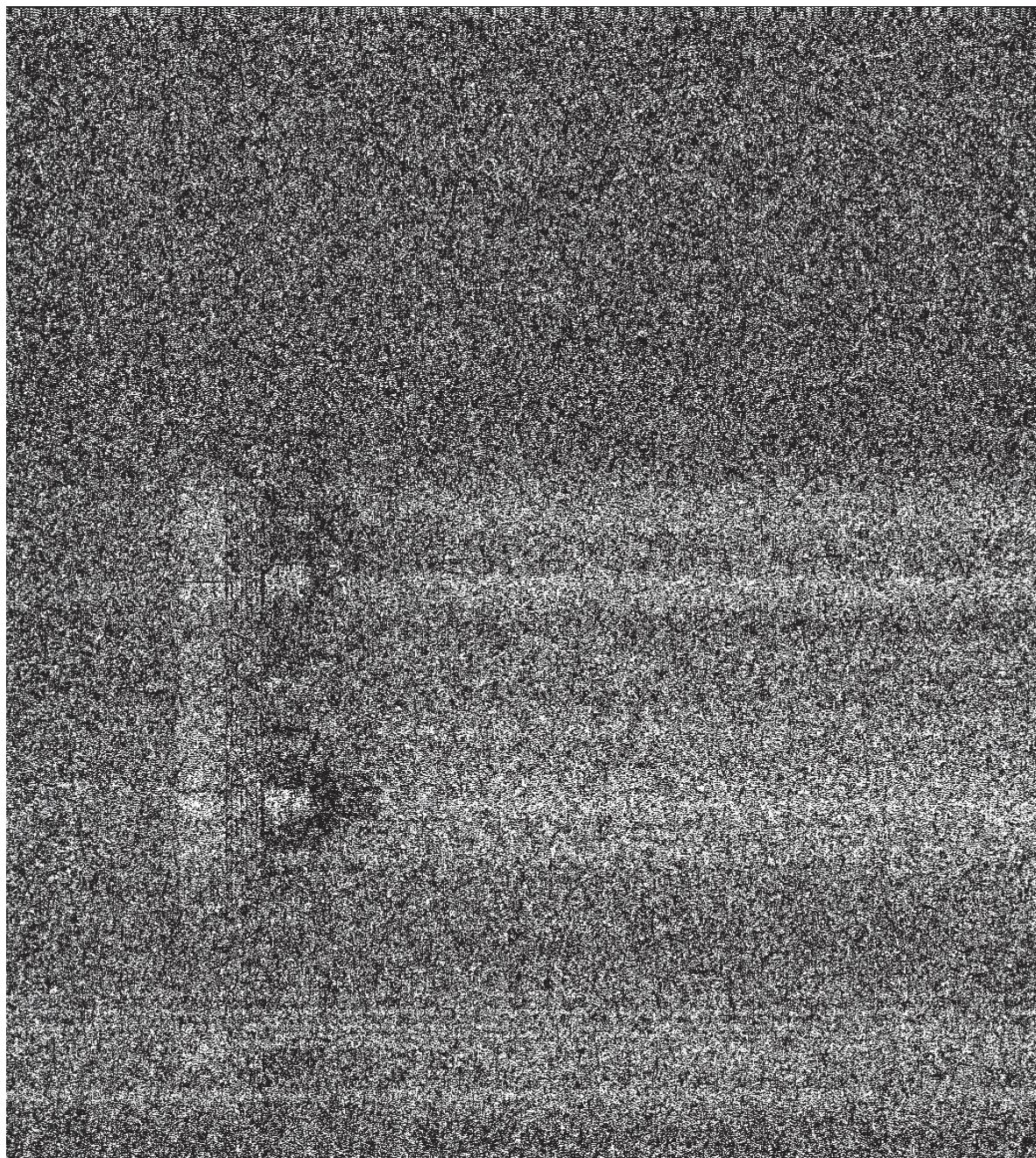
**Figura 3.23** Imagen del canal B proveniente del satélite NOAA-19.

En la Figura 3.24 se muestra una imagen térmica proveniente del satélite NOAA-19.



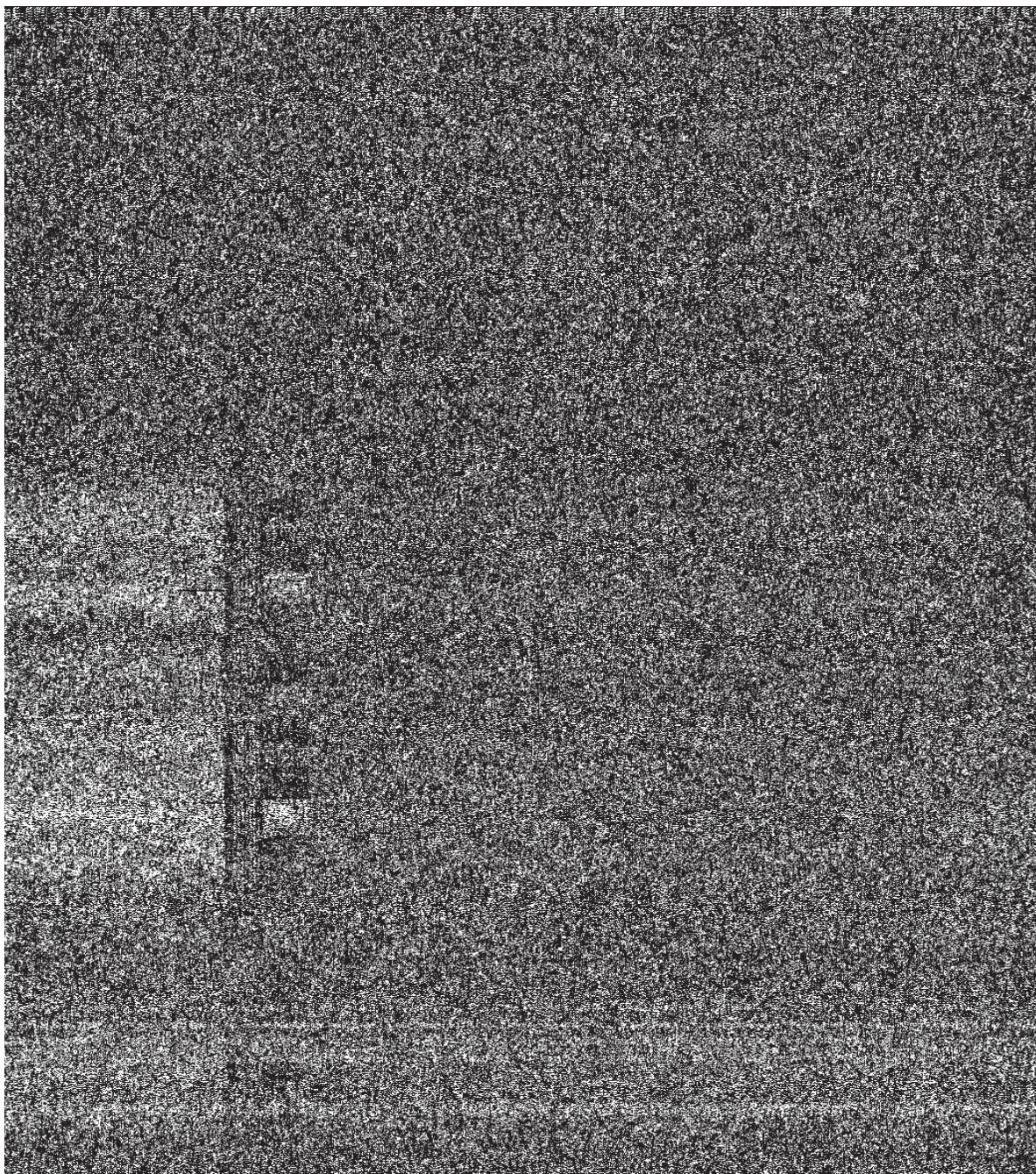
**Figura 3.24** Imagen termica proveniente del satélite NOAA-19.

En la Figura 3.25 se muestra una imagen del canal A proveniente del satélite NOAA-15.



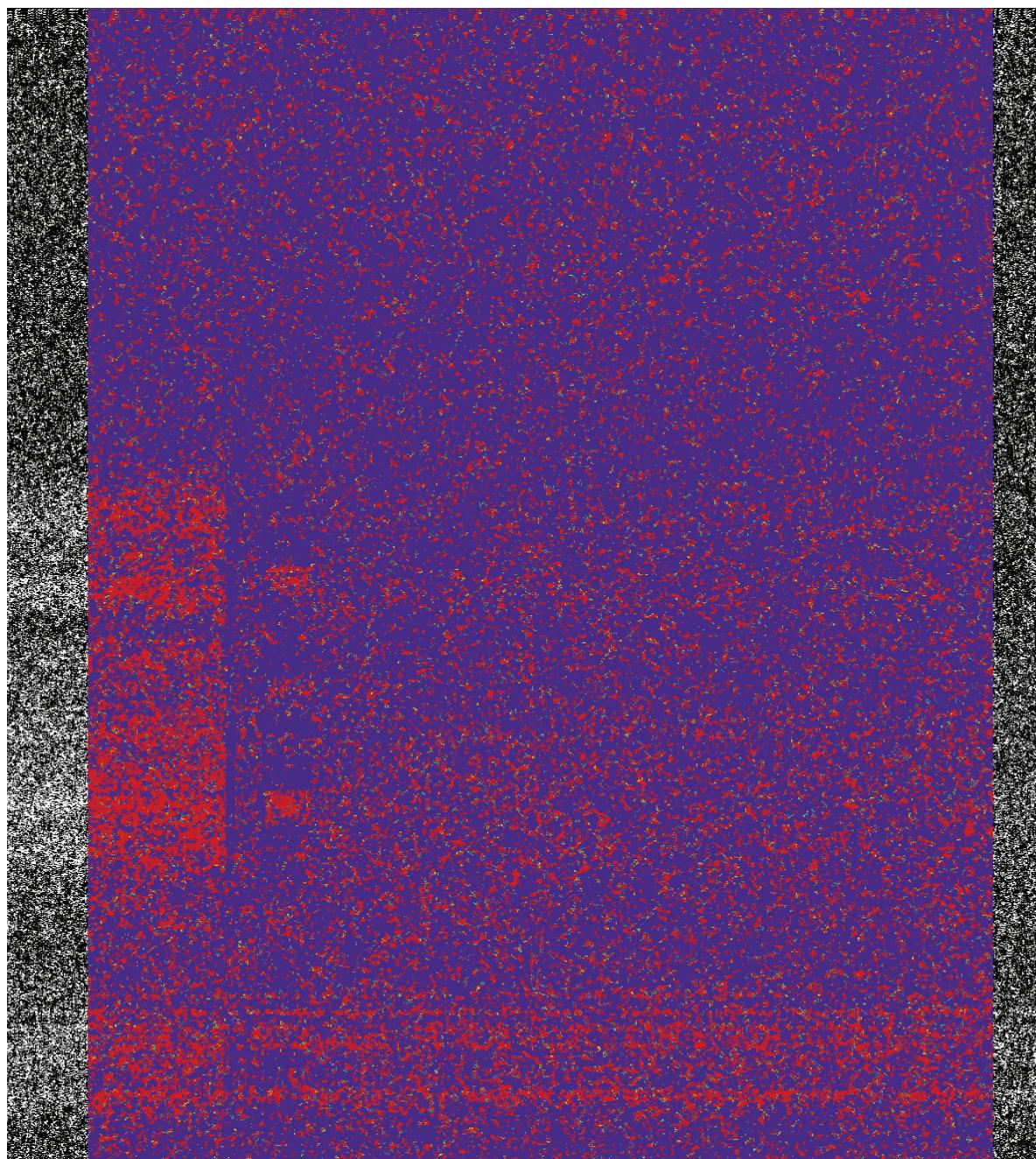
**Figura 3.25** Imagen del canal A proveniente del satélite NOAA-15.

En la Figura 3.26 se muestra una imagen del canal B proveniente del satélite NOAA-15.



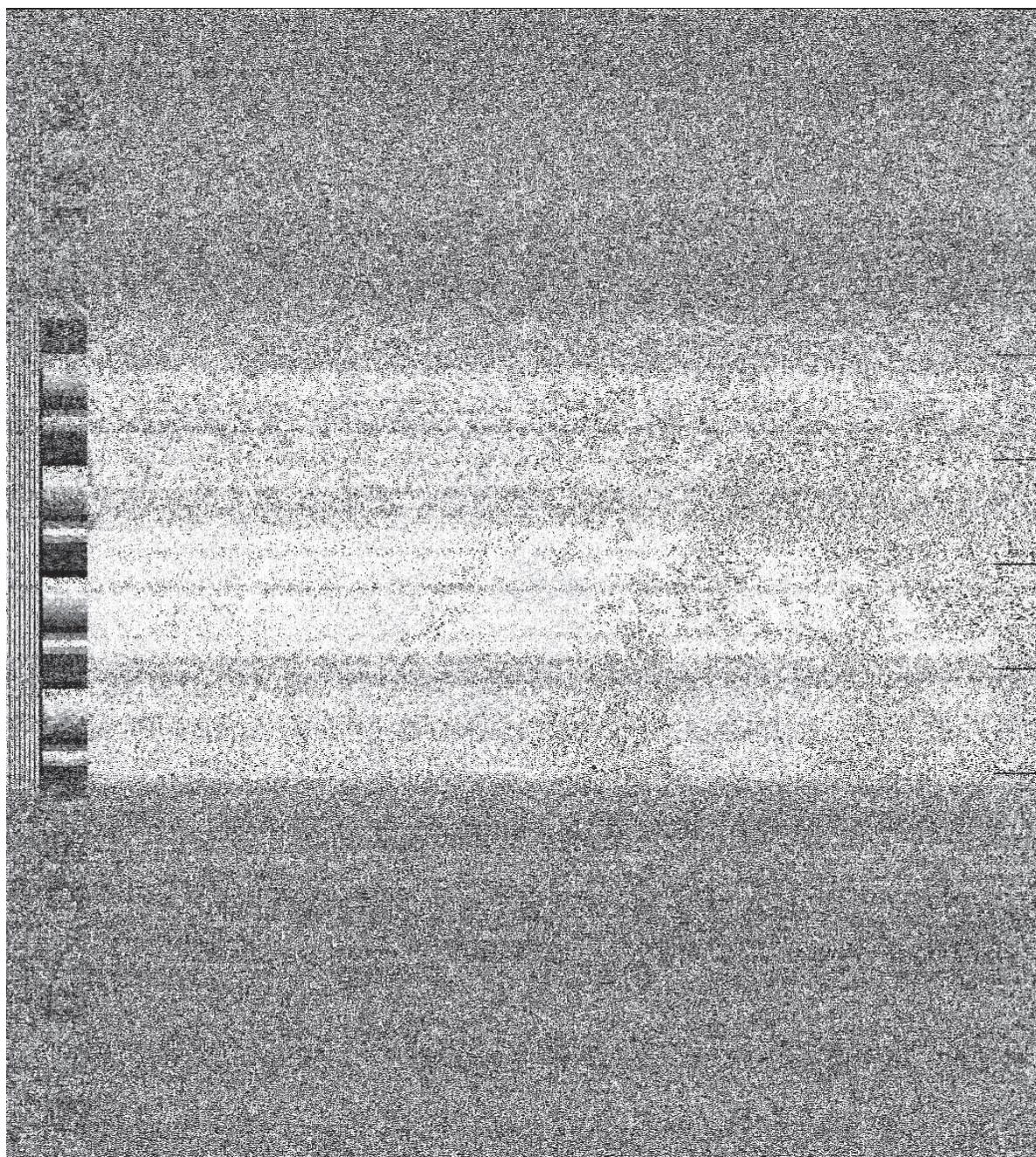
**Figura 3.26** Imagen del canal B proveniente del satélite NOAA-15.

En la Figura 3.27 se muestra una imagen térmica proveniente del satélite NOAA-15.



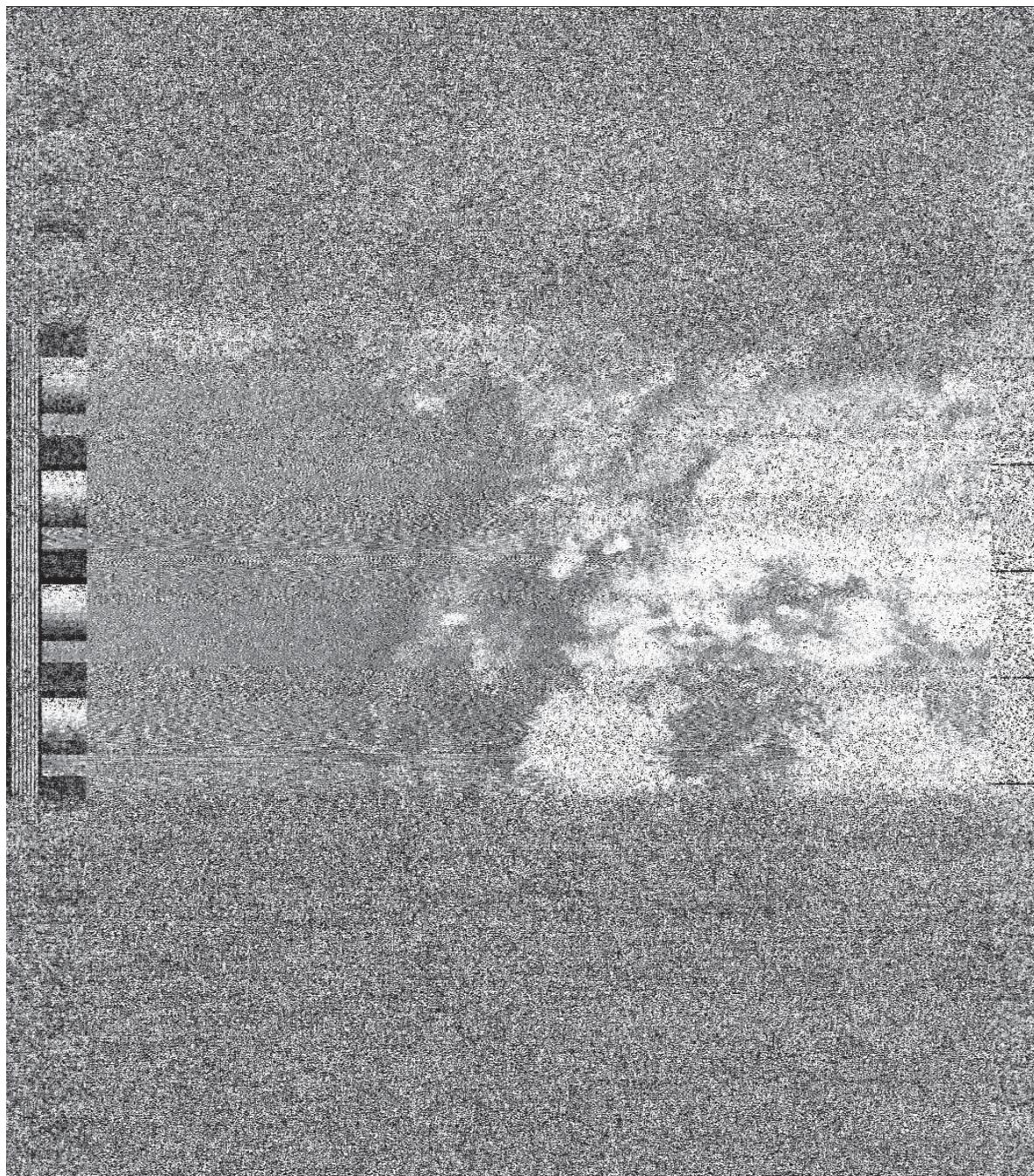
**Figura 3.27** Imagen termica proveniente del satélite NOAA-15.

En la Figura 3.28 se muestra una imagen del canal A proveniente del satélite NOAA-18.



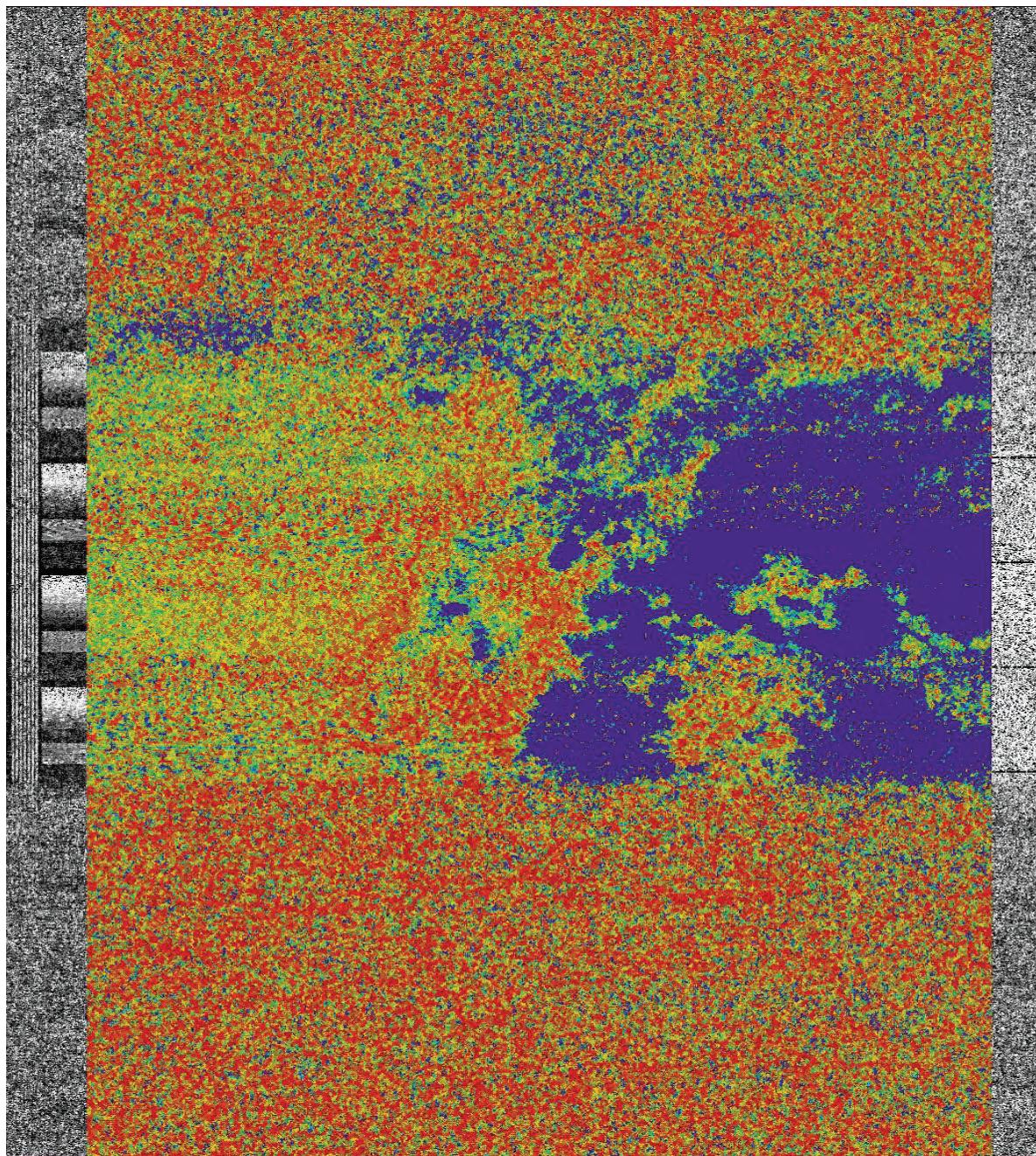
**Figura 3.28** Imagen del canal A proveniente del satélite NOAA-18.

En la Figura 3.29 se muestra una imagen del canal B proveniente del satélite NOAA-18.



**Figura 3.29** Imagen del canal B proveniente del satélite NOAA-18.

En la Figura 3.30 se muestra una imagen térmica proveniente del satélite NOAA-18.



**Figura 3.30** Imagen termica proveniente del satélite NOAA-18.

### 3.1.3 FASE DE ALMACENAMIENTO

EL objetivo de esta fase es comprobar que las imágenes que fueron previamente decodificadas, sean almacenadas en la base de datos.

El almacenamiento de todas las imágenes se realizará en la tabla tabla\_imagenes, donde el campo **idsat** tomará uno de los tres posibles valores configurados en la llave foránea, es decir, podrán tener el valor de 1, de 2 ó de 3, ver Figura 2.51.

Para realizar las pruebas de almacenamiento con cada uno de los satélites se utiliza el Código 2.1, el cual se encuentra incluido en el código de automatización. Luego, se verifica mediante la herramienta web phpMyadmin que efectivamente las imágenes se encuentran almacenadas.

En la Figura 3.31 se observa que las imágenes obtenidas de los satélites NOAA-19, NOAA-15 y NOAA-18 con la antena cuadrifilar helicoidal fueron almacenadas correctamente.

	<input type="checkbox"/>	Editar	Copiar	Borrar	<b>id</b>	<b>nombre</b>	<b>Imagen</b>	<b>fecha</b>	<b>idsat</b>
	<input type="checkbox"/>	Editar	Copiar	Borrar	139	Canal-A --> 14:55	[BLOB - 932.1 KB]	2016-10-24	3
	<input type="checkbox"/>	Editar	Copiar	Borrar	140	Canal-B --> 14:55	[BLOB - 898.9 KB]	2016-10-24	3
	<input type="checkbox"/>	Editar	Copiar	Borrar	141	Canal-therm --> 14:55	[BLOB - 2.2 MB]	2016-10-24	3
	<input type="checkbox"/>	Editar	Copiar	Borrar	142	Canal-A --> 18:41	[BLOB - 708.1 KB]	2016-10-24	1
	<input type="checkbox"/>	Editar	Copiar	Borrar	143	Canal-B --> 18:41	[BLOB - 721.1 KB]	2016-10-24	1
	<input type="checkbox"/>	Editar	Copiar	Borrar	144	Canal-therm --> 18:41	[BLOB - 1.8 MB]	2016-10-24	1
	<input type="checkbox"/>	Editar	Copiar	Borrar	145	Canal-A --> 18:56	[BLOB - 799.6 KB]	2016-10-24	2
	<input type="checkbox"/>	Editar	Copiar	Borrar	146	Canal-B --> 18:56	[BLOB - 829.8 KB]	2016-10-24	2
	<input type="checkbox"/>	Editar	Copiar	Borrar	147	Canal-therm --> 18:56	[BLOB - 2 MB]	2016-10-24	2

**Figura 3.31** Verificación del almacenamiento de las imágenes.

Como se puede observar en la Figura 3.31 el tamaño máximo que ocupa una imagen es de 2.2 MB y el mínimo es de 708.1 KB. Tomando en cuenta las 9 imágenes almacenadas, en promedio una imagen dentro de la base de datos ocupa 1.2 MB.

Así también se observa que el nombre con el cual se almacena las imágenes es un nombre por defecto acompañado de la hora a la cual se realiza el almacenamiento, es decir, para las imágenes del canal A siempre serán “Canal-A-->”, para las imágenes del canal B siempre serán “Canal-B-->” y para las imágenes térmicas siempre serán “Canal-threm -->” junto con la hora de almacenamiento.

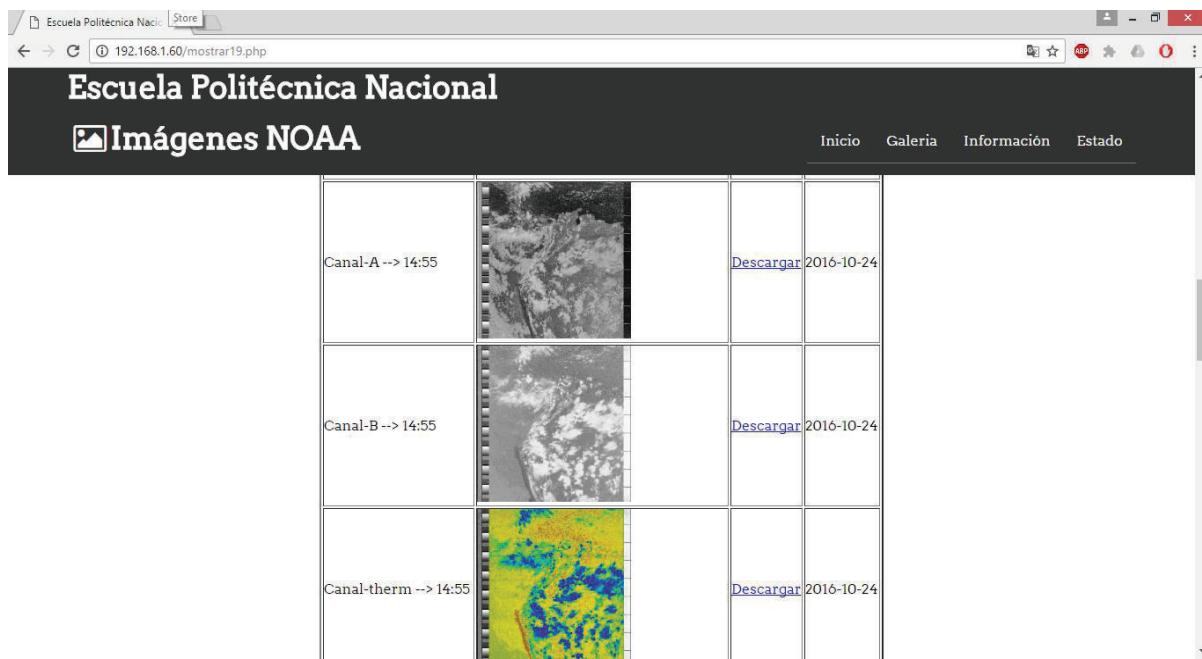
### 3.1.4 FASE DE VISUALIZACIÓN

El objetivo de esta fase es comprobar la presentación de las imágenes en la aplicación web implementada. La dirección de red utilizada para las pruebas de esta fase de visualización es 192.168.1.0/26.

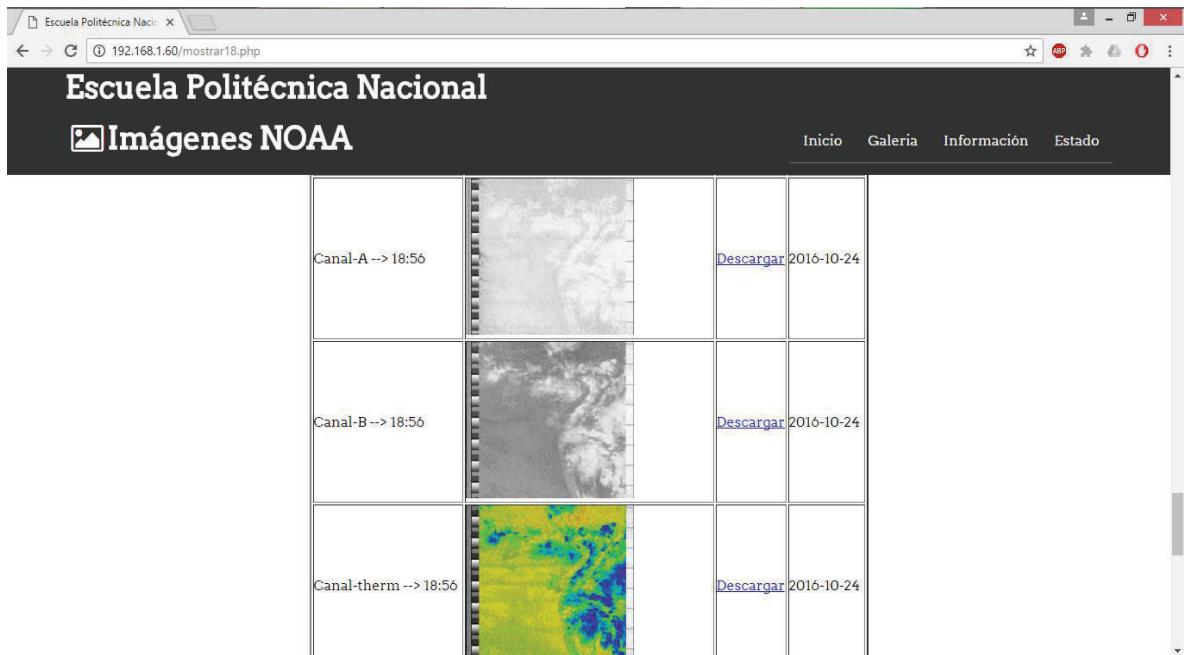
En primer lugar, se configura una dirección IP estática para el módulo WI-FI de la placa Raspberry Pi, la cual es 192.168.1.60; y esta dirección IP es la misma para el servidor web.

Luego, desde un terminal que se encuentre en la misma red que la placa Raspberry Pi se realiza una petición HTTP al servidor web colocando la dirección IP antes mencionada en un navegador web. Posteriormente, se selecciona la interfaz web **Galería** y se selecciona el satélite NOAA del cual se desean observar las imágenes.

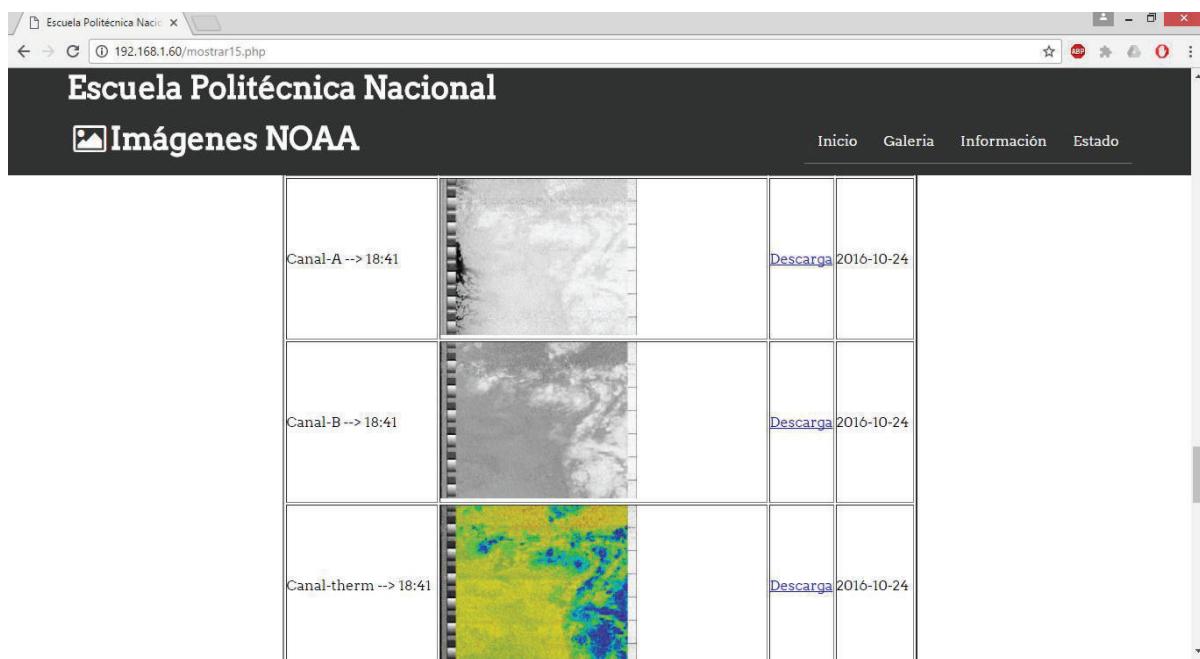
En la Figura 3.32, Figura 3.33 y Figura 3.34 se observa las imágenes almacenadas de los satélites NOAA-19, NOAA-18 y NOAA-15 respectivamente.



**Figura 3.32** Visualización de imágenes almacenadas del NOAA-19.



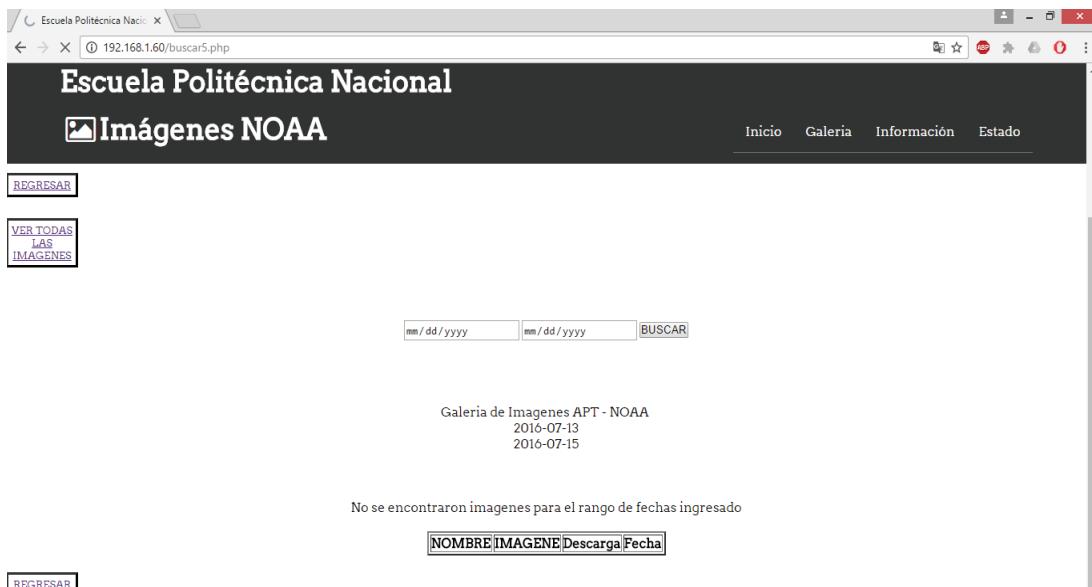
**Figura 3.33** Visualización de imágenes almacenadas en la del NOAA-18.



**Figura 3.34** Visualización de imágenes almacenadas del NOAA-15.

Para el caso de visualizar las imágenes almacenadas utilizando el motor de búsqueda, estas se mostrarán de forma ascendente, es decir, desde las imágenes más antiguas

hasta las actuales. En el caso de que no existan imágenes para el rango de fechas ingresadas aparecerá el mensaje “No se encontraron imágenes para el rango de fechas ingresado”, tal como se observa en la Figura 3.35.



**Figura 3.35** Resultado del motor de búsqueda cuando no existen imágenes.

En el Anexo F (digital) se encuentran dos videos, cada uno de ellos muestra el funcionamiento del prototipo con cada uno de las antenas realizadas en el presente proyecto.

### 3.2 ANÁLISIS DE RESULTADOS

En las pruebas realizadas con la antena cuadrifilar helicoidal y el esquema realizado en la aplicación GNU Radio Companion, ver Figura 2.48, las señales emitidas por cada uno de los tres satélites NOAA se obtuvieron satisfactoriamente, en tanto que con la antena turnstile la señal adquirida fue mucho más degradada. Estos resultados apoyan los datos analizados en la sección 1.4.2, donde se indicó que la antena cuadrifilar helicoidal presenta un mejor diagrama de radiación que la antena turnstile.

Durante de la ejecución del esquema de la Figura 2.48 para todas las pruebas de adquisición realizadas, el porcentaje promedio de procesamiento del CPU de la Raspberry Pi fue del 42%, tal como se puede observar la Figura 3.6, Figura 3.7, Figura

3.8, Figura 3.10, Figura 3.11 y Figura 3.12. Así también, en los videos del Anexo F (digital) se puede apreciar que durante la obtención de las tres horas en las que los satélites NOAA transmiten sobre el territorio continental ecuatoriano mediante la técnica de *web scraping* el procesamiento de CPU de la Raspberry Pi osciló entre el 25% y 35%; y durante la fase de decodificación el procesamiento fue en promedio 26%.

Antes de analizar y comentar los resultados de la Figura 3.6, Figura 3.7, Figura 3.8, Figura 3.10, Figura 3.11 y Figura 3.12, cabe mencionar, que el espectro que se muestra en las figuras mencionadas, es un espectro de amplitud. Por lo tanto, la escala logarítmica (dB) utilizada en el eje de las ordenadas es calculada en base a un voltaje RMS ( $V_{RMS}$ ) [56].

Para realizar un análisis en base a la potencia de la señal APT en recepción, se debe pasar de una escala logarítmica de amplitud (dB) a una escala logarítmica de potencia (dBm). Para lo cual, se hace uso de la ecuación (3.1) [57]:

$$P = 20 \log_{10} V + 13 \quad (3.1)$$

Donde:

- V: es el voltaje RMS.
- P: es la potencia expresada en dBm.

A continuación, se muestra un ejemplo del cálculo de la potencia de la señal APT en dBm. En primer lugar se debe calcular el voltaje RMS, para lo cual, se debe conocer su valor en dB. En base a la Figura 3.6, la señal APT en promedio, tiene sus picos máximos en -47 dB, por lo tanto el voltaje RMS es el siguiente:

$$V_{RMS} = 10^{-47/20} = 4.47 \text{ mV}$$

Remplazando el voltaje RMS calculado, en la ecuación (3.1), se tiene lo siguiente:

$$P_{(dBm)} = 20 \log_{10}(4.47 \times 10^{-3}) + 13$$

$$P_{(dBm)} = -33.9 \text{ dBm}$$

En la Tabla 3.1 se muestran los valores promedios, tanto en dB como en dBm, de la señal APT obtenida con cada una de las antenas, para cada uno de los satélites NOAA.

**Tabla 3.1** Señal APT en dB y dBm.

<b>Señal APT adquirida con la antena Cuadrifilar Helicoidal</b>		
<b>Satélite</b>	Amplitud de la señal [dB]	Potencia de la señal [dBm]
<b>NOAA-19</b>	-47	-33.9
<b>NOAA-15</b>	-50	-37
<b>NOAA-18</b>	-49	-35.99
<b>Señal APT adquirida con la antena Turnstile</b>		
<b>NOAA-19</b>	-62	-49
<b>NOAA-15</b>	-61	-48
<b>NOAA-18</b>	-51	-37.99

Como se puede observar en la Tabla 3.1, la potencia recibida de la señal APT es mejor con la antena cuadrifilar helicoidal.

También se puede observar, que existe una diferencia de 15.1 dBm entre la potencia recibida con la antena cuadrifilar helicoidal y la antena turnstile para el satélite NOAA-19; para el satélite NOAA-15 esta diferencia es de 11 dBm y para el satélite NOAA-18 la diferencia es de 2 dBm.

En las pruebas realizadas con la antena cuadrifilar helicoidal la reproducción de los archivos .wav permitió escuchar que existe un ruido considerable en la adquisición de la señal en formato APT. Dicho ruido, puede ser apreciado en las imágenes y puede deberse principalmente a las conexiones soldadas de la antena.

A pesar que la señal en formato APT está siendo adquirida con la antena turnstile, como se puede observar en la Figura 3.10, Figura 3.11 y Figura 3.12, la diferencia que existe entre el ruido y la señal no permite que el software WXtoImg pueda distinguir

entre la señal y el ruido, lo que conlleva a que las imágenes que fueron decodificadas aparezcan como si solo existiera ruido.

La antena cuadrifilar helicoidal permite obtener una señal de mejor calidad que con la antena turnstile, lo que conlleva también a una mejor calidad de imagen. Para posteriores adquisiciones se decidió utilizar la antena cuadrifilar helicoidal.

El software WXtolmg permitió realizar satisfactoriamente la decodificación de la señal modulada en amplitud (AM), obteniendo así la correspondiente imagen.

Al realizar el almacenamiento de las imágenes en la misma placa Raspberry Pi, el espacio disponible que se tiene para dicho almacenamiento se encuentra limitado por la capacidad de la tarjeta MicroSD; para el caso del presente proyecto la capacidad de la tarjeta MicroSD es de 64 GB, de los cuales quedan disponibles aproximadamente 48 GB, luego de realizar las instalaciones de software y del sistema operativo.

Como se mencionó en la sección 3.1.3 una imagen que se encuentra almacenada en la base de datos ocupa en promedio 1.2 MB, por lo tanto, se tiene lo siguiente:

$$1.2 \text{ MB}/\text{imagen} * 18 \text{ imágenes}/\text{día} * 30 \text{ días} = 648 \text{ MB}/\text{mes}$$

$$\frac{48 \text{ GB}}{648 \text{ MB}/\text{mes}} \approx 74 \text{ meses}$$

En base al cálculo realizado, en aproximadamente 74 meses el espacio para el almacenamiento de las imágenes se terminará.

La interfaz de la aplicación web implementada que permite visualizar las diferentes imágenes que han sido adquiridas, es intuitiva y fácil de manejar. Además, su motor de búsqueda permite a los usuarios realizar una búsqueda específica de imágenes.

### **3.3 COSTO REFERENCIAL DEL PROTOTIPO**

A continuación, en la Tabla 3.2 se muestra un costo referencial del sistema prototipo implementado en el presente proyecto. En cuanto al precio de los equipos adquiridos en el Anexo G (digital) se adjuntan las facturas.

Para el cálculo del precio de las horas de ingeniería, se tomó en consideración que un ingeniero gana en promedio USD 1200 al mes, esto dividido para el número de horas de trabajo en un mes (240 horas) [58], da como resultado un valor de 5 dólares por hora.

**Tabla 3.2** Costo referencial del prototipo.

Equipo	Cantidad	Costo Unitario [USD]	Costo Total [USD]
<b>Kit Raspberry Pi*</b>	1	69.99	69.99
<b>Módulo RTL-SDR*</b>	1	24.95	24.95
<b>Pigtail*</b>	1	3.50	3.50
<b>PVC 4cm diámetro</b>	2	6.60	13.20
<b>PVC 1cm diámetro</b>	2	1.30/metro	2.60
<b>Cable RG58</b>	10	0.32	3.20
<b>Cable RG59</b>	1	0.68	0.50
<b>Conector BNC</b>	2	0.60	1.20
<b>Tarjeta MicroSD 64 GB*</b>	1	21.99	21.99
<b>Costos Varios**</b>			102.78
<b>Mano de obra</b>			
<b>Horas de Ingeniería</b>	200	5.00	1000
		<b>Subtotal</b>	1243,91
		<b>IVA (14%)</b>	174.14
		<b>Total</b>	1418.05

\*Elementos comprados en el exterior.

\*\*En los costos varios se incluyen impuestos, nacionalización, seguros y costos de envío de todos los elementos comprados en el exterior.

## CAPÍTULO 4: CONCLUSIONES Y RECOMENDACIONES

### 4.1 CONCLUSIONES

- El módulo RTL-SDR utilizado en el presente proyecto sintonizó correctamente cada una de las tres frecuencias (137.1 MHz, 137.9125 MHz y 137.62 MHz) de transmisión en formato APT de los satélites NOAA y la aplicación GNU Radio Companion (GRC), permitió visualizar satisfactoriamente la señal en formato APT modulada en frecuencia.
- Se realizó con éxito la demodulación de la señal en formato APT modulada en frecuencia mediante la utilización del Bloque WBFM Receive, el cual realiza una demodulación por cuadratura, obteniendo así la señal modulada en amplitud (AM) que contiene la información escaneada por el sensor AVHRR/3. Ademas, mediante la utilización del bloque Wav File Sink se almacenó dicha señal modulada en amplitud (AM) para su posterior decodificación.
- La placa Raspberry Pi funciona adecuadamente para la implementación del sistema prototipo realizado en el presente proyecto, presentando su mayor consumo de procesamiento durante la fase de adquisición con un 42% de consumo de CPU. Así también, presentó un consumo máximo del 35% durante la realización del *web scraping*, y un 27% durante la fase de decodificación.
- En base a las pruebas realizadas con las antenas construidas en el presente proyecto, se concluye que la antena cuadrifilar helicoidal presenta mejores características que la antena turnstile para la adquisición de la señal en formato APT, por lo tanto, las imágenes adquiridas con la antena cuadrifilar helicoidal son de mejor calidad que las imágenes obtenidas con la antena turnstile.
- Las imágenes obtenidas con la antena cuadrifilar helicoidal en el presente proyecto podrán ser utilizadas por alumnos, docentes y entidades públicas o

privadas principalmente para el estudio, monitoreo o pronóstico del tiempo. Así también, la información relacionada al proyecto y recopilada en el mismo puede ser empleada en un proyecto similar o más complejo.

- El software WXtolmg, permitió realizar satisfactoriamente la decodificación de las señales moduladas en amplitud (AM) a un formato de imagen; para el caso del presente proyecto a una imagen de extensión .png.
- La base de datos utilizada en el presente proyecto es la adecuada, ya que permite un almacenamiento eficiente y una visualización ordenada de las imágenes meteorológicas en la aplicación web implementada. Además, este diseño es escalable, ya que si se desea añadir más satélites será necesario solo agregar el nombre del satélite y un identificador único.
- La técnica de *Web Scraping* facilitó conocer la hora a la cual los satélites NOAA se encuentran transmitiendo sobre el territorio continental ecuatoriano, y de esta manera saber en qué momento empezar a realizar la adquisición de la señal en formato APT.
- Mediante la utilización de la técnica de *Web Scraping* y el lenguaje de programación Python se desarrolló el código para la automatización del sistema prototipo realizado en el presente proyecto. Como resultado, se tiene un sistema de adquisición de imágenes meteorológicas satelitales totalmente automático, evitando así posibles errores humanos y la necesidad de un operador para la realización de las tareas de adquisición, decodificación y almacenamiento.
- La interfaz gráfica de GNU Radio Companion (GRC) es una herramienta intuitiva, fácil de manejar e idónea para estudiantes, docentes e investigadores en la realización de sistemas de comunicación, así como también en el análisis de señales.

- La tecnología RDS se vuelve cada vez más fuerte dentro de la industria de las telecomunicaciones, ya que permite la implementación de una gran variedad de sistemas de comunicación, bajo la funcionalidad de diseño en software.

## 4.2 RECOMENDACIONES

- Se recomienda para un futuro proyecto realizar el diseño y construcción de una antena para recepción de imágenes meteorológicas en el rango de 137 MHz a 138 MHz, en el que se incluyan simulaciones y mediciones de la antena.
- Se recomienda para un futuro proyecto realizar un sistema prototipo para la adquisición de la señal de alta definición HRPT (*High Resolution Picture Transmission*) proveniente los satélites NOAA.
- Para que el sistema prototipo no se apague o se reinicie por fallas del sistema eléctrico, se recomienda para un futuro proyecto realizar la implementación de un sistema de autonomía energética; asegurando de esta manera que la adquisición de las imágenes no se vea interrumpida.
- Para la conexión entre el módulo RTL-SDR y la antena, se recomienda utilizar conectores, *pigtail* o cable con la impedancia característica adecuada, evitando así pérdidas innecesarias y logrando obtener una máxima transferencia de potencia.
- Al implementar un servidor web en la placa Raspberry Pi se recomienda otorgar los permisos necesarios de lectura, escritura y ejecución a las carpetas donde se encuentren almacenados los archivos escritos en lenguaje PHP, HTML y CSS.

- Para un correcto funcionamiento del código de automatización del sistema prototipo, se recomienda que antes de ejecutarlo se verifique que la placa Raspberry Pi posea una conexión a Internet.
- Para acceder a la aplicación web implementada se recomienda establecer una dirección IP estática válida dentro la red en la se encuentre la Raspberry Pi. Esta dirección IP estática puede ser configurada tanto en el módulo Wi-Fi como en la interfaz LAN de la Raspberry Pi.

## REFERENCIAS BIBLIOGRÁFICAS

- [1] M. Raboli y A. Gulich, «Conocimientos Básicos sobre Teleobservación,» Septiembre 2005. [En línea]. Available: [https://2mp.conae.gov.ar/descargas/Materiales%20/Conocimientos\\_Basicos\\_sobre\\_Teleobservacion\\_Satelites\\_NOAA\\_CONAE.pdf](https://2mp.conae.gov.ar/descargas/Materiales%20/Conocimientos_Basicos_sobre_Teleobservacion_Satelites_NOAA_CONAE.pdf). [Último acceso: 6 Junio 2016].
- [2] J. Martínez y A. Díaz, «Fundamentos de Teledetección Espacial,» Junio 2005. [En línea]. Available: <http://siga.cna.gob.mx/SIGA/Percepcion/Fundamentos%20de%20teledetecci%C3%B3n%20espacial.PDF>. [Último acceso: 6 Junio 2016].
- [3] O. Rodriguez y H. Arredondo, «Pontificia Universidad Javeriana Bogota, Departamento de Ingeniería Civil,» 2005. [En línea]. Available: [www.javeriana.edu.co/biblos/tesis/ingenieria/tesis123.pdf](http://www.javeriana.edu.co/biblos/tesis/ingenieria/tesis123.pdf). [Último acceso: 6 Junio 2016].
- [4] M. Arbelo, «Tutorial de Teledetección,» [En línea]. Available: <https://marbelo.webs.ull.es/rs2.pdf>. [Último acceso: 6 Junio 2016].
- [5] C. Rosado, «Comunicación por Satélite, Universidad de Humboldt,» 2000. [En línea]. Available: <https://uahc.files.wordpress.com/2014/01/comunicaciones-por-satelite.pdf>. [Último acceso: 8 Junio 2016].
- [6] «Universidad Técnica del Norte,» [En línea]. Available: <http://repositorio.utn.edu.ec/bitstream/123456789/1102/4/04%20ISC%20071%20CAPITULO%20DOS.pdf>. [Último acceso: 8 Junio 2016].
- [7] J. Luque, «Comunicaciones por Satélite, Revista Digital de ACTA,» 2013. [En línea]. Available:

- [http://www.acta.es/medios/articulos/comunicacion\\_e\\_informacion/016001.pdf](http://www.acta.es/medios/articulos/comunicacion_e_informacion/016001.pdf). [Último acceso: 9 Junio 2016].
- [8] J. Ramírez, «Quórum,» [En línea]. Available: file:///C:/Users/Usuario/Downloads/comunicacion%20via%20satelite.pdf. [Último acceso: 9 Junio 2016].
- [9] J. Peris, «Localización y Seguimiento de Satélites Artificiales,» [En línea]. Available: [http://www.bubok.es/downloads/download\\_libro\\_gratis?book=MTkwNjEyLTlwMTYwMjA1LQ==&tipo\\_portada=6&clave=Y2VzYXJfcG0xNjExQGhvdG1haWwuZXM=&verificado=985e1336285977960cce1e44544848bb93774c2a543d102c1f11a4d7d8d3d73](http://www.bubok.es/downloads/download_libro_gratis?book=MTkwNjEyLTlwMTYwMjA1LQ==&tipo_portada=6&clave=Y2VzYXJfcG0xNjExQGhvdG1haWwuZXM=&verificado=985e1336285977960cce1e44544848bb93774c2a543d102c1f11a4d7d8d3d73). [Último acceso: 10 Junio 2016].
- [10] R. Anguizaca, A. Remache, A. Rodriguez y S. Santana, «Centro de Información Bibliotecaria - ESPOL,» [En línea]. Available: [www.cib.espol.edu.ec/digipath/d\\_tesis\\_pdf/d-20740.pdf](http://www.cib.espol.edu.ec/digipath/d_tesis_pdf/d-20740.pdf). [Último acceso: 10 Junio 2016].
- [11] «Blog El Espacio,» [En línea]. Available: <http://guilledambrauskas.blogspot.com/>. [Último acceso: 10 Junio 2016].
- [12] D. Pérez, «Laboratorio de Tectónica Andina, Universidad de Buenos Aires,» [En línea]. Available: [http://aviris.gl.fcen.uba.ar/Curso\\_SR/Guia\\_Curso\\_sr\\_2007.pdf](http://aviris.gl.fcen.uba.ar/Curso_SR/Guia_Curso_sr_2007.pdf). [Último acceso: 11 Junio 2016].
- [13] P. Hernández, «Análisis espacio-temporal de la desertificación usando sensores NOAA,» [En línea]. Available: <ftp://tesis.bbtk.ull.es/ccppytec/cp146.pdf>. [Último acceso: 11 Junio 2016].
- [14] J. González, «Universidad de Colima,» Septiembre 2012. [En línea]. Available: [http://digeset.ucol.mx/tesis\\_posgrado/Pdf/Maestr%C3%ADA%20en%20Cienci](http://digeset.ucol.mx/tesis_posgrado/Pdf/Maestr%C3%ADA%20en%20Cienci)

- as%20de%20la%20Tierra%20TESIS%20ARTURO%20GONZALEZ.pdf. [Último acceso: 13 Junio 2016].
- [15] E. Pernía, «Sensores Espaciales para Estudios Ambientales, a las Puertas del Siglo XXI,» *Revista Forest*, nº 2, pp. 207-22, 1999.
- [16] «Departamento de Geología, Geografía y Medio Ambiente, Universidad de Alcalá,» [En línea]. Available: <https://geogra.uah.es/noaa/galerianoaa/index.html>. [Último acceso: 13 Junio 2016].
- [17] J. Brava, «OceanDocs Repository of Ocean Publications,» 2004. [En línea]. Available: [www.oceandocs.org/bitstream/handle/1834/1467/Bava\\_2004.pdf?...3](http://www.oceandocs.org/bitstream/handle/1834/1467/Bava_2004.pdf?...3). [Último acceso: 13 Junio 2016].
- [18] C. Alvarez, «Escuela de Ingeniería de Telecomunicaciones, Universidad Politécnica de Cataluña,» [En línea]. Available: [upcommons.upc.edu/bitstream/handle/2099.1/14115/PFC\\_Carlo\\_Alvarez\\_Antena\\_DCA.pdf](http://upcommons.upc.edu/bitstream/handle/2099.1/14115/PFC_Carlo_Alvarez_Antena_DCA.pdf). [Último acceso: 15 Junio 2016].
- [19] «Sharing Earth Observation Resources,» [En línea]. Available: <https://directory.eoportal.org/web/eoportal/satellite-missions/n/noaa-poes-series-5th-generation>. [Último acceso: 16 Junio 2016].
- [20] B. J. Vera Navides, «Universidad Politécnica de Cataluña, Diseño y construcción de una antena cuadrifilar helicoidal para recepción de imágenes transmitidas por satélites de órbita polar,» Febrero 2012. [En línea]. Available: <http://upcommons.upc.edu/handle/2099.1/15026>. [Último acceso: 5 Julio 2016].

- [21] V. Dascal, P. Dolea, O. Cristea y T. Palade, «Advanced VHF Ground Station For NOAA Weather Satellite APT Image Reception,» *ACTA TECHNICA NAPOCENSIS Electronics and Telecommunications*, vol. 53, nº 3, pp. 1-7, 2012.
- [22] N. Crisan y L. Cremene, «NOAA Signal Decoding and Image Processing Using GNU-RADIO,» *ACTA TECHNICA NAPOCENSIS Electronics and Telecommunications*, vol. 51, nº 1, pp. 1-5, 2012.
- [23] «NOAA APT Receiver,» 5 Junio 2013. [En línea]. Available: <http://kwos-iz0qwm.blogspot.com/2013/06/noaa-apt-receiver.html>. [Último acceso: 14 Junio 2016].
- [24] R. W. Hollander, «Working Group Satellites, Resonant Quadrifilar Helix Antenna,» Septiembre 1999. [En línea]. Available: <http://www.kunstmanen.net/WKfiles/Techdocs/RQHA/RQHA1999-1eng.pdf>. [Último acceso: 5 Julio 2016].
- [25] L. B. Cebik, «A simple Fixed antenna for VHF/UHF Satellite Work, Asociación Amateur Radio de la Región de Lisboa,» Agosto 2001. [En línea]. Available: file:///C:/Users/Usuario/Downloads/VHFUHFSatelite.pdf. [Último acceso: 5 Julio 2016].
- [26] I. Pinar y J. Murrillo, «Departamento Teoría de la Señal y Comunicaciones, Universidad de Sevilla,» 2011. [En línea]. Available: [alojoptico.us.es/murillo/LibroSDRV7USv8.pdf](http://alojoptico.us.es/murillo/LibroSDRV7USv8.pdf). [Último acceso: 16 Junio 2016].
- [27] M. González, «Teoría de la Señal y Comunicaciones, Universidad de Sevilla,» 2014. [En línea]. Available: [bibing.us.es/proyectos/abreprojy/12174/fichero/pfc\\_Manuel\\_Puerto.pdf](http://bibing.us.es/proyectos/abreprojy/12174/fichero/pfc_Manuel_Puerto.pdf). [Último acceso: 16 Julio 2016].
- [28] M. Poveda, H. Poveda y F. Merchan, «Latin American and Caribbean Consortium of Engineering Institutions,» Julio 2015. [En línea]. Available:

- http://www.laccei.org/LACCEI2015-SantoDomingo/RefereedPapers/RP039.pdf. [Último acceso: 16 Junio 2016].
- [29] J. P. Montero, «Ingeniería de Telecomunicaciones, Universidad Autónoma de Madrid,» Enero 2014. [En línea]. Available: http://arantxa.ii.uam.es/~jms/pfcsteleco/lecturas/20140130JuanPabloMonteroHidalgo.pdf.. [Último acceso: 17 Junio 2016].
- [30] J. Amador y N. Torres, «RDS (Radio Definido por Software) consideraciones para su Implementación de Hardware,» *Revista Telemática*, vol. 12, nº 2, pp. 56-68, 2013.
- [31] R.-S. BLOG, «The Hobbist's Guide to RTL-SDR,» [En línea]. Available: http://www.qsl.net/yo4tnv/docs/The%20Hobbyists%20Guide%20To%20RTL-SDR%20-%20Carl%20Laufer.pdf. [Último acceso: 17 Junio 2016].
- [32] «RTL-SDR.COM,» [En línea]. Available: http://www rtl-sdr.com/buy-rtl-sdr-dvb-t-dongles/. [Último acceso: 17 Junio 2016].
- [33] «R820T-DATASHEET,» [En línea]. Available: http://superkuh.com/gnuradio/R820T\_datasheet-Non\_R-20111130\_unlocked.pdf. [Último acceso: 27 Abril 2016].
- [34] «Raspberrypi.org,» 2015. [En línea]. Available: https://www.raspberrypi.org. [Último acceso: 20 Junio 2016].
- [35] J. Rayo y J. Chimbo, «Centro de Información Bobliotecario, ESPOL,» 2013. [En línea]. Available: www.cib.espol.edu.ec/Digipath/D\_Tesis\_PDF/D-97061.pdf. [Último acceso: 20 Junio 2016].
- [36] «WXtoImg software to decode APT,» 2013. [En línea]. Available: http://www.wxtoimg.com/support/es\_wxgui.pdf. [Último acceso: 20 Junio 2016].

- [37] «Wiki-Debian,» Diciembre 2015. [En línea]. Available: <https://wiki.debian.org/RaspberryPi>. [Último acceso: 27 Abril 2016].
- [38] «GNURadio THE FREE & OPEN SOFTWARE RADIO ECOSYSTEM,» 2015. [En línea]. Available: <http://gnuradio.org/redmine/projects/gnuradio/wiki/EnEspanol>. [Último acceso: 21 Junio 2016].
- [39] «Tecnopedia,» [En línea]. Available: <http://www.tecnopedia.net/wp-content/uploads/2015/07/Mini-Guia-B%C3%A1sica-Servidores-WAMP-LAMP-MAMP1.pdf?eb27c9>. [Último acceso: 20 Junio 2016].
- [40] «Culturacion - Servidores Web,» 2014. [En línea]. Available: <http://culturacion.com/que-es-apache/>. [Último acceso: 20 Junio 2016].
- [41] A. Pérez, «Desarrollo de herramientas web de gestión docente, Universidad Politécnica de Cartagena,» Diciembre 2007. [En línea]. Available: <http://repositorio.upct.es/bitstream/handle/10317/179/pfc2475.pdf;jsessionid=06FB09EF72A6415EFCF87B9E941282CA?sequence=1>. [Último acceso: 20 Junio 2016].
- [42] «Designspark, Taking the Raspberry Pi 2 for a test drive with Gnu Radio,» 2 Febrero 2015. [En línea]. Available: <http://www.rs-online.com/designspark/electronics/eng/blog/taking-the-raspberry-pi-2-for-a-test-drive-with-gnu-radio-2>. [Último acceso: 22 Junio 2016].
- [43] «ZR6AIC, Raspberry Pi as SDR server,» 13 Febrero 2014. [En línea]. Available: <http://zr6aic.blogspot.com/2013/02/setting-up-my-raspberry-pi-as-sdr-server.html>. [Último acceso: 22 Junio 2016].
- [44] «Osmocom-SDR,» [En línea]. Available: <http://sdr.osmocom.org/trac/wiki/rtl-sdr>. [Último acceso: 22 Junio 2016].

- [45] «Vensign, Como instalar un servidor LAMP,» Noviembre 2009. [En línea]. Available: <http://vensign.com/como-instalar-servidor-lamp-ubuntu-paso-a-paso/>. [Último acceso: 27 Junio 2016].
- [46] «Geekland Blog de Tecnología, Servidores WEB,» Junio 2015. [En línea]. Available: <http://geekland.eu/instalar-un-servidor-web-lamp/>. [Último acceso: 27 Junio 2016].
- [47] «Turnstile Antenna-Forum,» [En línea]. Available: <https://www.circuitsonline.net/forum/view/41715>. [Último acceso: 5 Julio 2016].
- [48] C. Serres, «EA1URO.COM,» 29 Julio 2012. [En línea]. Available: <http://www.ea1uro.com/pdf/manualDipolos.pdf>. [Último acceso: 31 Agosto 2016].
- [49] «Pasternack The Engineers RF Source,» [En línea]. Available: <https://www.pasternack.com/images/productpdf/rg58c-u.pdf>. [Último acceso: 31 Agosto 2016].
- [50] «West Penn Wire Cable Confidence,» [En línea]. Available: [http://www.westpenn-wpw.com/pdfs/coaxial\\_spec\\_pdfs/CCTV/816.pdf](http://www.westpenn-wpw.com/pdfs/coaxial_spec_pdfs/CCTV/816.pdf). [Último acceso: 31 Agosto 2016].
- [51] K. P. P. Rajan y G. A. S. Sundaram, «Weather Pattern Monitoring Using SDR,» *International Journal of Advanced Electrical and Electronics Engineering*, vol. II, nº 2, pp. 63-68, 2013.
- [52] K. Eric, R. Mark, V. Vijo y D. Bill, «Jayhawk Engineering Laboratory, Weather Satellite Imaging System,» 21 Mayo 2006. [En línea]. Available: [http://project.markroland.com/weather-satellite-imaging/WxSat\\_report.pdf](http://project.markroland.com/weather-satellite-imaging/WxSat_report.pdf). [Último acceso: 7 Julio 2016].
- [53] C. Pérez, «Departamento de Ingeniería de Comunicaciones, Universidad de Cantabria,» Febrero 2015. [En línea]. Available:

- http://personales.unican.es/perezvr/pdf/CH6ST\_Web.pdf. [Último acceso: 30 Noviembre 2016].
- [54] «RadioReference.com,» Enero 2015. [En línea]. Available: <http://forums.radioreference.com/software-defined-radio/305969-rtl-sdr-sample-rate-calculations.html>. [Último acceso: 5 Septiembre 2016].
- [55] P. Mario, «GeekyTheory Raspberry-Pi-Status,» 1 Abril 2015. [En línea]. Available: <https://github.com/GeekyTheory/Raspberry-Pi-Status>. [Último acceso: 11 Julio 2016].
- [56] G. Heinzel, A. Rudiger y S. R, «Holometer,» 15 Febrero 2002. [En línea]. Available: [https://holometer.fnal.gov/GH\\_FFT.pdf](https://holometer.fnal.gov/GH_FFT.pdf). [Último acceso: 5 Diciembre 2016].
- [57] «Lab. Communications Theory and Systems, University of Victoria, Canada,» [En línea]. Available: [https://www.ece.uvic.ca/~elec350/lab\\_manual/data/Theory\\_Spectrum\\_Analyzer.pdf](https://www.ece.uvic.ca/~elec350/lab_manual/data/Theory_Spectrum_Analyzer.pdf). [Último acceso: 5 Diciembre 2016].
- [58] «Ministerio Coordinador de Producción, Empleo y Competitividad,» [En línea]. Available: [http://www.produccion.gob.ec/wp-content/uploads/2013/07/Acuerdo\\_Ministerial\\_006\\_Reglamento-de-Horas-Suplementarias-y-Extras.pdf](http://www.produccion.gob.ec/wp-content/uploads/2013/07/Acuerdo_Ministerial_006_Reglamento-de-Horas-Suplementarias-y-Extras.pdf). [Último acceso: 1 Diciembre 2016].
- [59] «Raspberry Pi NOOBS Setup,» [En línea]. Available: <https://www.raspberrypi.org/documentation/installation/noobs.md>. [Último acceso: 3 Mayo 2016].
- [60] «Github Raspberry Noobs,» 2013. [En línea]. Available: <https://github.com/raspberrypi/noobs/blob/master/README.md>. [Último acceso: 3 Mayo 2016].

- [61] «Ridgecorp Consultants Ltd,» [En línea]. Available: [www.ridgecrop.demon.co.uk/index.htm?guiformat.htm](http://www.ridgecrop.demon.co.uk/index.htm?guiformat.htm). [Último acceso: 3 Mayo 2016].

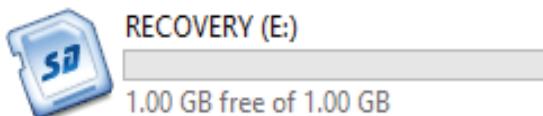
# ANEXOS

## ANEXO A: INSTALACIÓN DE RASPBIAN [59], [60]

Para la instalación de Raspbian “Jessie” se utiliza NOOBS (New Out of the Box Software), que es un paquete donde se encuentran algunos sistemas operativos, que le permiten tener a la Raspberry Pi un multi-boot la primera vez que se enciende.

El primer paso es descargar en la PC la imagen de NOOBS, que se encuentra disponible en la página oficial de la Raspberry Pi [www.raspberrypi.org/downloads/noobs/](http://www.raspberrypi.org/downloads/noobs/). La descarga se realiza en un archivo comprimido Zip, por lo que una vez finalizada se tiene que extraer los archivos comprimidos, para posteriormente copiarlos en la tarjeta Micro SD.

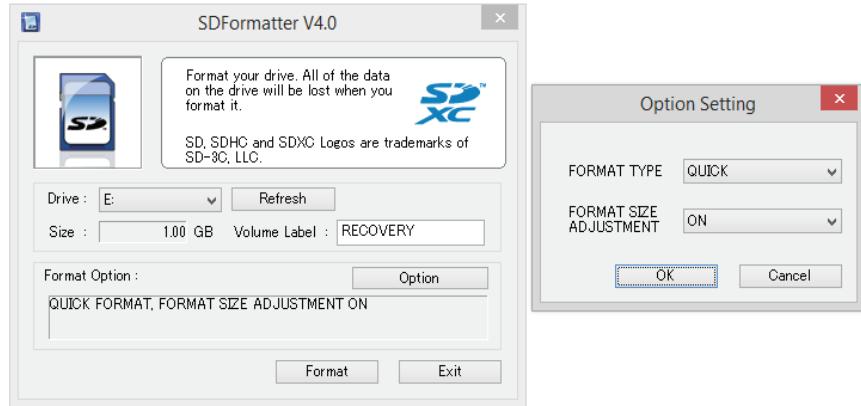
Luego, se ingresa la tarjeta Micro SD, con su respectivo adaptador, en el lector de tarjetas de la PC para poder conocer la letra que se asigna a la tarjeta, en este caso como se observa en la Figura A1 se asignó la letra E.



**Figura A1** Letra asignada a la tarjeta Micro SD.

Una vez conocida la letra que se asignó, se tiene que formatear a la tarjeta Micro SD en su formato predeterminado FAT32, para ello se utiliza el programa SDFormatter V4.0, el cual se puede descargar en la página [www.sdcards.org/downloads/formatter\\_4/](http://www.sdcards.org/downloads/formatter_4/).

Luego, terminada la descarga del programa se ejecuta como administrador; una vez abierto, se pasa a escoger la letra del dispositivo de almacenamiento, como ya se mencionó es la letra E, luego se da clic en el botón “Option” donde aparecerá una nueva ventana y en la opción “FORMAT SIZE ADJUSTEMENT” escoger la opción “ON”, ver Figura A2.

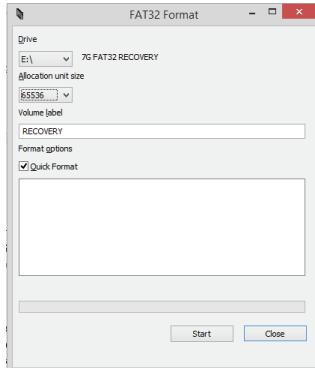


**Figura A2** Programa SDFormatter V4.0.

Debido a que el gestor de arranque integrado en la GPU de la Raspberry Pi solamente tiene soporte para lectura de sistemas de archivos FAT (FAT16 y FAT32) el procedimiento que se acaba de mencionar se lo realiza en el caso de que la tarjeta Micro SD tenga una capacidad máxima de hasta 32 GB, debido que hasta esta cantidad el formato predeterminado es FAT32.

En el caso de que se utilice una tarjeta Micro SD cuya capacidad de almacenamiento sea mayor (64 GB y 128 GB), se debe tener en cuenta que estas utilizan un sistema de archivos exFAT, por lo que a pesar de ser formateadas con el programa SDFormatter V4.0, la Raspberry Pi no podrá arrancar ninguna imagen contenida en estas.

Para lograr que la tarjeta Micro SD cuyo formato predeterminado exFAT pase a FAT32 se utilizará el programa FAT32 Format, el cual se encuentra disponible en [61]. Una vez descargado, se lo ejecuta como administrador, luego, se escogerá la letra que se le asignó a la tarjeta Micro SD y su cantidad de almacenamiento en Bytes como se muestra en la Figura A3. Una vez finalizado este procedimiento la tarjeta Micro SD tendrá un sistema de archivos FAT32, por lo que ya se puede copiar una imagen para ser arrancada en la Raspberry Pi.



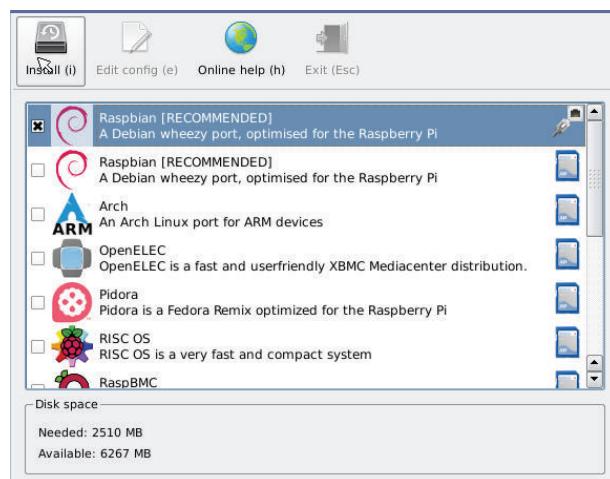
**Figura A3** Programa FAT32 Format.

Por último, cuando la tarjeta Micro SD ya este formateada se pasa a copiar todos los archivos descomprimidos de la imagen NOOBS. En la Figura A4 se observa una comparación luego de formatear la Micro SD y copiar la imagen NOOBS.



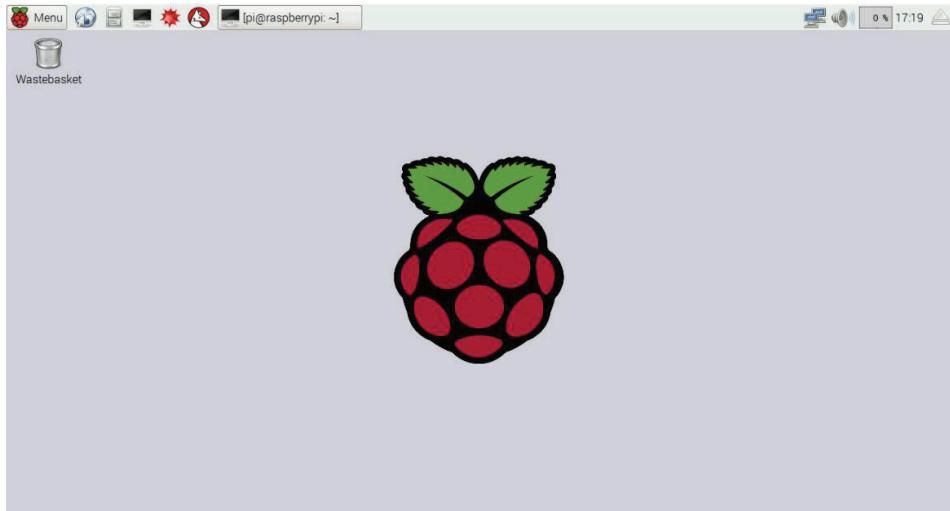
**Figura A4** Comparación de la tarjeta Micro SD.

Una vez finalizado el procedimiento de formatear y copiar la imagen NOOBS, se expulsa la tarjeta Micro SD de la PC y se la coloca en la ranura correspondiente de la Raspberry Pi, para así comenzar con el proceso de arranque del sistema operativo Raspbian “Jessie”, como se muestra en la Figura A5.



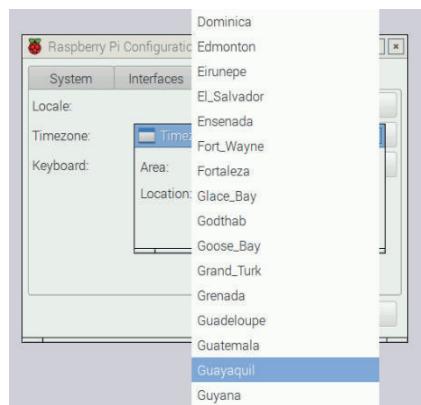
**Figura A5** Pantalla de BOOT.

Ya culminado el proceso de arranque se mostrará automáticamente la interfaz gráfica del sistema operativo, como se muestra en la Figura A6, con lo cual la Raspberry Pi esta lista para empezar a ser utilizada.



**Figura A6** Pantalla de inicio Raspberry Pi.

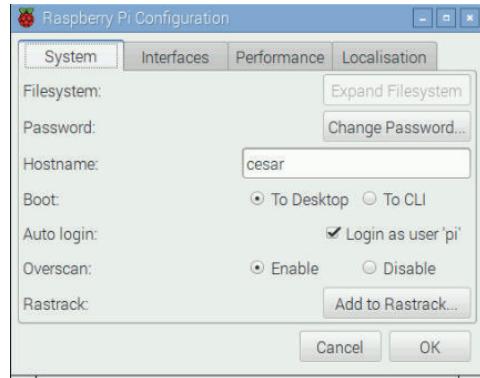
En primera instancia, se debe realizar la configuración de la localización geográfica más cercana, en este caso Guayaquil, el tipo de teclado que se utilizará, el *hostname* y un *password*, el cual se utiliza en el caso de realizar una conexión SSH. Todas estas configuraciones se las puede realizar mediante la herramienta `raspi-config` o mediante la aplicación *Raspberrypi Configuration*, en este caso las configuraciones se realizaron mediante la segunda opción, como se muestra en las siguientes Figuras.



**Figura A7** Configuración de Ubicación.



**Figura A8** Configuración del Teclado.



**Figura A9** Configuración de Hostname y Password.

En cuanto al tipo de salida de audio, se selecciona dando clic derecho en el icono de audio que se encuentra en la esquina superior derecha de la pantalla, como se muestra en la Figura A10, pudiendo escogerse entre el puerto HDMI y el Jack de 3.5 mm.



**Figura A 10** Configuración de la salida de audio.

Una vez finalizado, para que todas las configuraciones realizadas tomen efecto es necesario reiniciar la Raspberry, de lo contrario permanecerán las configuraciones por defecto.

## ANEXO B: PARÁMETROS DE LA ANTENA CUADRIFILAR HELICOIDAL [20], [24]

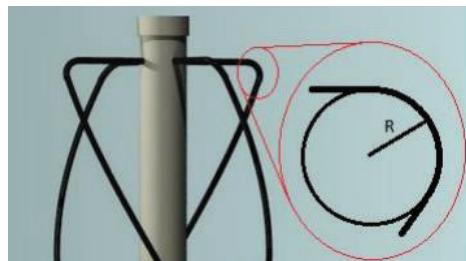
En la Figura B1 se muestran los datos ingresados en la aplicación web de la calculadora de Jhon Coppens.

Frecuencia de diseño	137.5	MHz
Cantidad de vueltas	0.5	
Largo de una vuelta	1	largos de onda
Radio del doblado	15	mm
Diámetro del conductor	5	mm (valor óptimo: 20.5 mm)
Relación diámetro/altura	0.44	
	Calcular	

**Figura B1** Datos ingresados en la calculadora de Jhon Coppens.

A continuación, se describen cada uno de los parámetros ingresados.

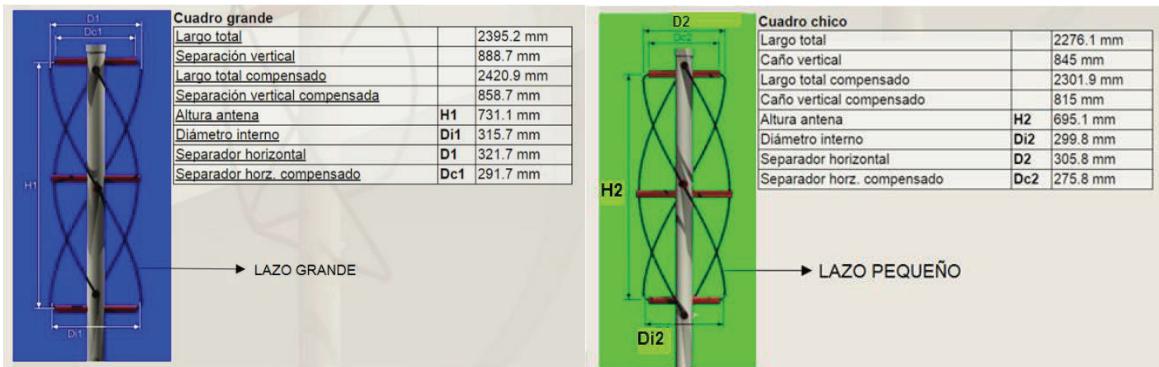
- **Frecuencia de diseño:** Dado que la frecuencia de transmisión de los satélites NOAA se encuentra entre los 137 MHz y 138 MHz, la frecuencia ingresada corresponde al promedio de dichas frecuencias, es decir, 137.5 MHz.
- **Cantidad de vueltas:** Este valor hace referencia a la torsión que tendrá la espira. En base a simulaciones realizadas en [20], el valor óptimo es 0.5.
- **Largo de una vuelta:** Este valor hace referencia a la longitud total de la espira, cuyo único valor puede ser de una longitud de onda.
- **Radio de doblado:** Este valor hace referencia al doblado presente al final de los brazos superiores e inferiores de la antena cuadrifilar helicoidal, tal como se muestra en la Figura B2. En base al estudio realizado en [24] este valor debe ser de 15 mm.



**Figura B2** Radio de doblado de la antena cuadrifilar helicoidal.

- **Diámetro del conductor:** Este valor hace referencia al diámetro del cable utilizado para la construcción de la antena. Para el caso del presente proyecto, se utilizó cable RG58 cuyo diámetro total es de 5 mm [49].
- **Relación diámetro/altura:** El parámetro diámetro, hace referencia al diámetro del cilindro imaginario que se forma debido a la torsión y forma de la antena cuadrifilar helicoidal; y el parámetro altura, hace referencia a la altura de la antena. En base a las simulaciones realizadas en [20], el valor más adecuado es 0.44.

Los resultados obtenidos con los datos ingresados anteriormente se muestran en la Figura B3.



**Figura B3** Resultados de la calculadora de Jhon Coppens.

A continuación, se describen dos de los resultados obtenidos

- **Largo total:** Largo total de cada una de las espiras tomando en consideración el radio de doblado.
- **Separación vertical:** Largo de uno de los conductores verticales.
- **Largo total compensado:** Largo total que debe tener cada uno de los lazos, sin contar con el radio de doblado que existe al final de los brazos superiores e inferiores de la antena.
- **Separación vertical compensada:** Largo de uno de los conductores verticales, sin contar con el radio de doblado.

## ANEXO C: INSTALACIÓN PARA MOSTRAR EL ESTADO DE LA RASPBERRY PI [55]

A continuación se describe el proceso de instalación de las librerías necesarias para mostrar el estado del Raspberry Pi en la interfaz web **Estado** de la Raspberry Pi.

1. Se realiza la actualización de los paquetes que se encuentran en la placa Raspberry Pi, tal como se muestra en la Figura C1.

```
pi@cesar:~ $ sudo apt-get update
```

**Figura C1** Actualización de los paquetes en la Raspberry Pi.

2. Se instala el paquete `npm`, tal como se muestra en la Figura C2. Este paquete es el manejador por defecto para Node.js.

```
pi@cesar:~ $ sudo apt-get install npm
```

**Figura C2** Instalación de npm.

3. Se clona el paquete Raspberry-Pi-Status de un repositorio `git`, tal como se muestra en la Figura C3.

```
pi@cesar:~ $ git clone https://github.com/GeekyTherory/Raspberry-Pi-Status.git
```

**Figura C3** Clonación del paquete Raspberry-Pi.Status.

4. Una vez terminada la clonación, se cambia de directorio hacia la nueva carpeta `Raspberry-Pi-Status`, tal como se muestra en la Figura C4.

```
pi@cesar:~ $ cd Raspberry-Pi-Status
```

**Figura C4** Cambio de directorio hacia Raspberry-Pi-Satatus.

5. Se realiza la instalación de `npm`, tal como se muestra en la Figura C5.

```
pi@cesar:~ $ npm install
```

**Figura C5** Instalación de npm.

6. Se ejecuta el servicio mediante el comando “nodejs server.js”, tal como se muestra en la Figura C6

```
pi@cesar:~ $ nodejs server.js
```

**Figura C6** Comando para ejecutar el servicio de Raspberry-Pi-Status.