



**Implementación de un receptor EWBS y aplicaciones interactivas Full-seg
utilizando una plataforma de Radio Definida por Software según el estándar ISDB-Tb**

Chanataxi Nacimba, Bryan Javier

Departamento de Eléctrica, Electrónica y Telecomunicaciones

Carrera de Ingeniería en Electrónica y Telecomunicaciones

Trabajo de titulación, previo a la obtención del título de Ingeniero en Electrónica y
Telecomunicaciones

Ing. Olmedo Cifuentes, Gonzalo Fernando

28 de julio del 2022



Tesis_Chanataxi_Bryan_rev.pdf

Scanned on: 16:7 July 27, 2022 UTC



Overall Similarity Score



Results Found



Total Words in Text

Identical Words	1106
Words with Minor Changes	246
Paraphrased Words	949
Omitted Words	0



Firma electrónica por:
GONZALO PERNANDO
OLMEDO CIFUENTES



Website | Education | Businesses



Departamento de Eléctrica, Electrónica y Telecomunicaciones

Carrera de Ingeniería en Electrónica y Telecomunicaciones

Certificación

Certifico que el trabajo de titulación: **“Implementación de un receptor EWBS y aplicaciones interactivas Full-seg utilizando una plataforma de Radio Definida por Software según el estándar ISDB-Tb”** fue realizado por el señor **Chanataxi Nacimba, Bryan Javier**; el mismo que cumple con los requisitos legales, teóricos, científicos, técnicos y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, además fue revisado y analizado en su totalidad por la herramienta de prevención y/o verificación de similitud de contenidos; razón por la cual me permito acreditar y autorizar para que se lo sustente públicamente.

Sangolquí, 28 julio de 2022

Firma:



.....
Ing. Olmedo Cifuentes, Gonzalo Fernando, PhD

C. C.: 1711696342



Departamento de Eléctrica, Electrónica y Telecomunicaciones
Carrera de Ingeniería en Electrónica y Telecomunicaciones

Responsabilidad de Autoría

Yo, **Chanataxi Nacimba, Bryan Javier**, con cédula de ciudadanía n° 1722071055, declaro que el contenido, ideas y criterios del trabajo de titulación: **Implementación de un receptor EWBS y aplicaciones interactivas Full-seg utilizando una plataforma de Radio Definida por Software según el estándar ISDB-Tb** es de mi autoría y responsabilidad, cumpliendo con los requisitos legales, teóricos, científicos, técnicos, y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, respetando los derechos intelectuales de terceros y referenciando las citas bibliográficas.

Sangolquí, 28 de julio de 2022

Firma

Chanataxi Nacimba, Bryan Javier

C.C.: 1722071055



Departamento de Eléctrica, Electrónica y Telecomunicaciones

Carrera de Ingeniería en Electrónica y Telecomunicaciones

Autorización de Publicación

Yo **Chanataxi Nacimba, Bryan Javier**, con cédula de ciudadanía n° 1722071055, autorizo a la Universidad de las Fuerzas Armadas ESPE publicar el trabajo de titulación: **Implementación de un receptor EWBS y aplicaciones interactivas Full-seg utilizando una plataforma de Radio Definida por Software según el estándar ISDB-Tb** en el Repositorio Institucional, cuyo contenido, ideas y criterios son de mi responsabilidad.

Sangolquí, 28 de julio de 2022

Firma

Chanataxi Nacimba, Bryan Javier

C.C.: 1722071055

Dedicatoria

Dedico el presente trabajo a mis padres, Jannett y Gonzalo quienes me han apoyado en todo momento a lo largo de mi vida estudiantil, sus consejos y dedicación han sido una gran motivación para lograr mis objetivos. A mis hermanos que han estado en los momentos difíciles y me han brindado su apoyo para seguir y conseguir mis objetivos.

Este logro es para ustedes.

Bryan Javier Chanataxi Nacimba

Agradecimiento

Agradezco a mi madre Jannett que con su amor incondicional ha sabido guiarme por el buen camino y con sus consejos ha hecho de mí una persona de bien, a mi padre Gonzalo que con su esfuerzo nos ha brindado las facilidades para estudiar a mis hermanos y a mí, por todo su apoyo en todos estos años los admiro por la familia que han formado.

A mis hermanos que me han apoyado en este largo camino y esas acciones han hecho que seamos unidos, esperando que ellos cumplan sus objetivos y sin duda estaré también para apoyarlos en la manera que se pueda.

A mis amigos que han estado en los momentos difíciles y me han ayudado a sobrellevar la vida universitaria, de cada uno de ellos he aprendido algo que me ha servido para seguir creciendo como persona.

A mis profesores que han compartido su conocimiento, han sido un pilar importante en mi desarrollo profesional, un agradecimiento especial al Ing. Gonzalo Olmedo mi tutor de tesis, gracias por su paciencia y apoyo a lo largo de la realización del proyecto.

Bryan Javier Chanataxi Nacimba

Índice de Contenido

Dedicatoria	6
Agradecimiento	7
Resumen.....	17
Abstract	18
CAPÍTULO I	19
Introducción.....	19
Antecedentes	19
Justificación e importancia	21
Alcance del proyecto.....	23
Objetivos.....	24
Objetivo General.....	24
Objetivos Específicos	24
Estado del arte	25
Organización	27
CAPÍTULO II	29
Marco Teórico	29
Introducción.....	29
Televisión Digital Terrestre	29
Estándares de TDT	30
Televisión Digital Terrestre en el Ecuador	31
Beneficios en el Ecuador.....	33

Estándar ISDB-Tb	33
Sistema de Recepción para TDT	34
Transport Stream.....	40
Tablas PSI/SI.....	42
Información del TMCC.....	46
Sistema de alerta de emergencia para el estándar ISDB-T	47
Emergency Warning Broadcasting System (EWBS)	48
Aplicaciones interactivas en TDT	56
GINGA	56
Norma ISO 13818-6	57
Paquetes DSM-CC dentro del Transport Stream	59
Secciones DSM-CC.....	59
Carrusel de Datos.....	61
Carrusel de Objetos.....	62
Radio Definida por Software	63
Equipo SDR Adalm Pluto	64
GNU Radio	65
CAPÍTULO III	67
Diseño e implementación del sistema de recepción EWBS y aplicaciones interactivas Full-seg	67
Introducción.....	67
Descripción del sistema	67

	10
Descripción de los elementos del sistema	68
Hardware	69
Software.....	69
Descarga de Módulos desde Git-Hub.....	70
Creación de nuevos bloques	71
Implementación del receptor Full-seg.....	74
Algoritmo para extraer el mensaje de emergencia.....	75
Bloque extractor PMT	76
Bloque Read EWBS	77
Bloque Flag EWBS.....	79
Bloque Emergency Message	81
Algoritmo para extraer los archivos de las aplicaciones interactivas.....	82
Función para extraer las secciones DSM-CC.....	85
Función para extraer el carrusel de datos	87
Función para extraer el carrusel de objetos	88
Diagrama completo del Receptor	89
PlutoSDR Source	90
PlutoSDR Sink.....	92
CAPÍTULO IV.....	95
Análisis de Resultados	95
Consideraciones Especiales.....	96

	11
Sincronización OFDM.....	96
Recepción de la constelación.....	98
Escenario 1.....	99
Transmisión para la aplicación interactiva.....	100
Transmisión para el Mensaje EWBS	106
Escenario 2.....	115
Medidas de calidad del receptor de aplicaciones interactivas y mensaje EWBS	116
CAPÍTULO V.....	127
Conclusiones y Recomendaciones	127
Conclusiones.....	127
Recomendaciones	129
Referencias.....	131
Apéndices	139

Índice de Tablas

Tabla 1	<i>Estaciones autorizadas para transmitir TDT en Ecuador</i>	32
Tabla 2	<i>Beneficios de usar TDT</i>	33
Tabla 3	<i>Parámetros del Sistema de transmisión ISDB-Tb</i>	39
Tabla 4	<i>Información del TMCC</i>	46
Tabla 5	<i>Estándares que ocupan el descriptor EWBS</i>	51
Tabla 6	<i>Código de Área por Países</i>	53
Tabla 7	<i>Señalización para la sobreimposición</i>	55
Tabla 8	<i>Codificación de las secciones DSM-CC</i>	60
Tabla 9	<i>Tipos de objeto</i>	62
Tabla 10	<i>Parámetros bloque PlutoSDR Source</i>	92
Tabla 11	<i>Parámetros bloque PlutoSDR Sink</i>	94
Tabla 12	<i>Ciudades en las que se activa el Sistema de Alerta EWBS</i>	110
Tabla 13	<i>Medidas objetivas en el rango de -90 a -60 Dbm</i>	118

Índice de Figuras

Figura 1 <i>Países latinoamericanos que usan el estándar ISDB-T</i>	30
Figura 2 <i>Sistema de Recepción para TDT</i>	35
Figura 3 <i>Procesamiento de la señal en el front end para full-seg</i>	36
Figura 4 <i>Multiplexación del Transport Stream para una programación</i>	40
Figura 5 <i>Proceso para obtener un TS</i>	41
Figura 6 <i>Formato de un TS con su respectiva cabecera</i>	42
Figura 7 <i>Estructura de la Tabla PAT</i>	44
Figura 8 <i>Estructura de la Tabla PMT</i>	45
Figura 9 <i>Diagrama general de un Sistema EWBS</i>	49
Figura 10 <i>Multiplexación de 4 servicios y mensaje EWBS</i>	50
Figura 11 <i>Descriptor EWBS</i>	52
Figura 12 <i>Código de Área para Ecuador</i>	53
Figura 13 <i>Código de Área para el cantón Cayambe</i>	55
Figura 14 <i>Estructura del estándar ISDB-T e ISDB-Tb</i>	57
Figura 15 <i>Tipos de conexiones para transmitir aplicaciones interactivas</i>	58
Figura 16 <i>Descriptor para el flujo DSM-CC</i>	59
Figura 17 <i>Transmisión Cíclica del Carrusel de Datos</i>	61
Figura 18 <i>Arquitectura genérica de la Radio Definida por Software</i>	63
Figura 19 <i>Dispositivo Adalm Pluto</i>	64
Figura 20 <i>Pantalla principal de GNU Radio</i>	65
Figura 21 <i>Diagrama General del Sistema</i>	68
Figura 22 <i>Conexión de los elementos del Sistema</i>	69
Figura 23 <i>Bloques del módulo ISDBT</i>	71
Figura 24 <i>Creación de nuevos módulos</i>	72

Figura 25 Creación de nuevos bloques.....	73
Figura 26 Declaración de entradas, salidas y funcionalidad de un bloque	73
Figura 27 Módulo y bloque creados en GNU Radio.	74
Figura 28 Diagrama de bloques del receptor Full-seg.....	75
Figura 29 Bloques necesarios para la extracción del mensaje EWBS	76
Figura 30 Diagrama de Flujo para el bloque Read EWBS	78
Figura 31 Diagrama del flujo para el bloque Flag EWBS	80
Figura 32 Diagrama de flujo para el bloque Emergency Message.....	82
Figura 33 Diagrama de bloques para las secciones DSM-CC	83
Figura 34 Diagrama del proceso para la extracción de los archivos.....	83
Figura 35 Estructura del algoritmo extractor de archivos	84
Figura 36 Diagrama de Flujo para el extractor de secciones DSM-CC	86
Figura 37 Receptor en GNU Radio	90
Figura 38 Bloque PlutoSDR Source.....	91
Figura 39 Parámetros PlutoSDR Source	91
Figura 40 Bloque Pluto SDR Sink	93
Figura 41 Parámetros bloque Pluto SDR Sink.....	93
Figura 42 Transmisor de TDT según el estándar ISDB-Tb.....	95
Figura 43 Sincronización del símbolo OFDM.....	97
Figura 44 Falta de sincronización del símbolo OFDM.....	97
Figura 45 Constelación recibida a frecuencias de muestreo altas	98
Figura 46 Constelación recibida correctamente.....	99
Figura 47 SDR Adalam-Pluto como transmisor y receptor escenario 1.....	100
Figura 48 Sincronización OFDM en la transmisión del archivo espetv.ts	101
Figura 49 Información del TMCC para la transmisión del archivo espetv.ts.....	102
Figura 50 Constelación recibida del archivo espetv.ts.....	102

Figura 51 <i>Información de la Tabla PMT del archivo espstv.ts</i>	103
Figura 52 <i>Información de las secciones DSM-CC</i>	104
Figura 53 <i>Archivos extraídos del archivo espstv.ts</i>	104
Figura 54 <i>Ejecución de la aplicación interactiva en GINGA</i>	105
Figura 55 <i>Explicación de la extracción de los archivos que complementan la aplicación interactiva</i>	106
Figura 56 <i>Espectro de la señal recibida</i>	106
Figura 57 <i>Sincronización de OFDM para la transmisión del archivo JAPON_EWBS_ESPE.ts</i>	107
Figura 58 <i>Información de TMCC del archivo JAPON_EWBS_ESPE.ts</i>	108
Figura 59 <i>Constelación de la señal recibida en la transmisión del archivo JAPON_EWBS_ESPE.ts</i>	109
Figura 60 <i>Análisis de la tabla PMT sin descriptor EWBS</i>	109
Figura 61 <i>Análisis del descriptor EWBS contenido en una tabla PMT</i>	110
Figura 62 <i>Ciudades en las cuales se activa el Sistema de Alerta EWBS</i>	112
Figura 63 <i>Análisis de la información para la sobreimposición</i>	113
Figura 64 <i>Activación de la bandera EWBS</i>	113
Figura 65 <i>Mensaje EWBS</i>	114
Figura 66 <i>Hoja de cálculo con los códigos de área</i>	114
Figura 67 <i>Funcionamiento del Receptor EWBS</i>	115
Figura 68 <i>SDR Adalam-Pluto como transmisor y receptor escenario 2</i>	116
Figura 69 <i>Medidas Objetivas en GNU Radio</i>	117
Figura 70 <i>Valores del MER obtenido</i>	119
Figura 71 <i>BER de Viterbi</i>	120
Figura 72 <i>BER de Viterbi en escala logarítmica</i>	121
Figura 73 <i>BER de Reed Solomon</i>	122

Figura 74 <i>BER de Reed Solomon en escala logarítmica</i>	123
Figura 75 <i>Constelaciones obtenidas en la recepción de la señal con variación de la potencia de recepción en el rango de -90 a -75 dBm</i>	124
Figura 76 <i>Constelaciones obtenidas en la recepción de la señal con variación de la potencia de recepción en el rango de -70 a -60 dBm</i>	125
Figura 77 <i>Reproducción de los archivos recibidos</i>	126

Resumen

La ubicación geográfica del Ecuador lo hace propenso a diversos fenómenos naturales que pueden causar daños materiales, así como la pérdida de vidas, es por esto que la implementación de un sistema que alerta temprana para notificar a la ciudadanía el inicio de un evento emergente, gracias al uso de la televisión digital se pueden implementar este tipo de sistemas con el uso de EWBS (En inglés, Emergency Warning Broadcasting System), a esto se pueden añadir aplicaciones interactivas que permitan alertar a la población.

En el presente trabajo se realizó la implementación de un receptor Full-seg según el estándar ISDB-Tb, con el uso de un SDR de bajo costo, específicamente el Adalm-Pluto. Para programar la funcionalidad del mismo fue utilizado el software GNU Radio, el que permite importar bloques de librerías existentes, así como crear nuevos, para completar una determinada tarea. Al receptor implementado fue necesario añadirle bloques que permitan analizar el flujo de transporte (TS - Transport Stream) generado para extraer la información EWBS, así como para extraer los archivos que complementan las aplicaciones interactivas que puedan venir incluidas en el flujo de datos. Los resultados obtenidos pueden verificarse con los archivos generados, así como de las medidas de calidad objetivas realizadas a la señal recibida.

Palabras clave: EWBS, ISDB-Tb, Transport Stream, Adalm-Pluto, Aplicaciones Interactivas.

Abstract

The geographical location of Ecuador makes it prone to various natural phenomena that can cause material damage and loss of life, which is why the implementation of an early warning system to notify citizens of the start of an emerging event, with the use of digital television it is possible to implement this type of system with the use of EWBS (Emergency Warning Broadcasting System), to which interactive applications can be added to alert the population.

In the present work, we have worked on the implementation of a Full-seg receiver according to the ISDB-Tb standard, with the use of a low-cost SDR, specifically the Adalm-Pluto, to program its functionality, GNU Radio has been used due to the ease of importing blocks from existing libraries as well as creating new ones to complete a specific task, it is necessary to add blocks to the implemented receiver that allow analyzing the generated Transport Stream (TS) to extract the EWBS information, as well as extracting the files that complement the interactive applications that may be included in the data flow, the results obtained can be verified with the generated files, as well as the objective quality measurements made on the received signal.

Keywords: EWBS, ISDB-Tb, Transport Stream, Adalm-Pluto, Interactive Applications.

CAPÍTULO I

Introducción

Antecedentes

La ubicación geográfica del país lo hace propenso a sufrir diversos desastres naturales en todas las regiones de su territorio, se hace especial énfasis en la Costa y la Sierra, en la Costa existen problemas de inundaciones debido a fenómenos meteorológicos, como es el caso del denominado “Fenómeno del niño”, el cual provoca intensas lluvias que a su vez traen problemas como: deslizamientos, inundaciones entre otros. En el caso de la Sierra ecuatoriana, se tienen problemas por sismos y erupciones volcánicas que a lo largo de los años han causado un sin número de pérdidas tanto materiales como vidas humanas.

Una de las principales amenazas en la Sierra fue la activación del volcán Cotopaxi en el año 2015, lo cual puede traer una serie de inconvenientes a las provincias cercanas al mismo, los deslizamientos de tierra pueden causar daños materiales significativos, la falta de herramientas para alertar de forma temprana a la ciudadanía puede tener varias causas, desde la falta de dinero para invertir en sistemas eficientes, hasta la falta de gestión de los respectivos gobiernos municipales (Córdova, 2017; Torres et al., 2018).

En el año 2016 en la Costa ecuatoriana, específicamente en Pedernales provincia de Manabí se dio un sismo de magnitud 7.8, convirtiéndose así en uno de los peores desastres naturales en los últimos años en el Ecuador, este evento causó una gran pérdida tanto económica como de vidas humanas, ya que se dio un alto índice de personas fallecidas, heridas y desaparecidas (IGPN, 2016).

El uso de la tecnología puede ser un gran paso para mejorar la respuesta que actualmente se tiene a este tipo de desastres naturales, actualmente el Ecuador se encuentra en la transición hacia la Televisión Digital Terrestre (TDT), lo que será una gran ventaja, no solo por la mejora de audio y video sino que se tendrá muchas otras ventajas como la optimización del uso del espectro radioeléctrico, además del uso de aplicaciones interactivas que ayuden a mejorar la experiencia de los usuarios; todas estas herramientas antes mencionadas pueden ser usadas para mejorar los sistemas de alerta que actualmente existen (MINTEL, 2017).

Las autoridades del país para empezar a implementar TDT a lo largo de todo el territorio ecuatoriano decidieron adoptar desde marzo del 2010 el estándar ISDB-T internacional o ISDB-Tb, que contiene mejoras brasileñas (ISDB-T, 2008). Como se mencionó anteriormente el uso de TDT ayudaría a optimizar el uso del espectro radioeléctrico (Valles et al., 2019).

El estándar para TDT ISDB-T fue estandarizado por ARIB (Association of Radio Industries and Businesses), en Japón desde diciembre del 2003, una de las principales ventajas de este estándar es la transmisión simultánea de señales para HDTV y One-seg. Los países que hacen uso del estándar forman parte del Foro Internacional ISDB-T que se encarga de añadir normativas para distintas aplicaciones que se tiene para TDT como pueden ser el Sistema de Alerta de Emergencia EWBS (Emergency Warning Broadcasting System) y aplicaciones interactivas (DiBEG, 2008).

El sistema EWBS se empezó a utilizar en Japón desde septiembre del año 1985, desde el año 2000 de forma digital para la transmisión satelital y desde el 2003 para transmisiones terrestres. En Latinoamérica el primer país que implementó este sistema fue Perú, gracias al aporte de la agencia Japan International Cooperation Agency (JICA), la cual aportó para que este sistema sea implementado con éxito; la transmisión

se la realizó a través del Canal Estatal de Perú (IRTP) (ISDB-T, 2015; JICA, 2016; PROMAX, 2017).

Las aplicaciones interactivas desarrolladas en GINGA pueden llegar a ser un aporte muy importante al estándar ISDB-Tb, brindando más opciones al usuario, presentando contenido de interactividad, entre los que se puede tener: información, ligada a la programación y aplicaciones transaccionales, todas estas características hacen que este tipo de aplicaciones ayuden de forma significativa a la ciudadanía en caso de ocurrir cualquier tipo de emergencia (Alulema, 2012; Bernal & Valencia, 2013).

Desde hace varios años, algunos profesores de la Universidad de las Fuerzas Armadas han venido trabajando con esta tecnología, para ayudar a mitigar los efectos que puede causar una eventual erupción del volcán Cotopaxi, han realizado pruebas tanto en la transmisión como en la recepción. Los resultados han sido alentadores, sin embargo en casos como el proyecto “Prototype of a centralized alert and emergency system for digital terrestrial television in Ecuador” hace falta complementar ciertas situaciones en la recepción (Olmedo, Acosta, et al., 2020; Olmedo, Sambrano, et al., 2020).

Es por esto, que la implementación de Sistemas de Alerta Temprana se vuelve esencial estos tiempos, además que se busca reducir costos es por esto que se ha decidido usar Radio Definido por Software (SDR), ocuparlo como receptor de TDT y que además pueda extraer del flujo TS información como el mensaje EWBS y los archivos necesarios para complementar las aplicaciones interactivas.

Justificación e importancia

Al encontrarnos en el “Cinturón de Fuego del Pacífico” el país se ubica en una zona propensa a sufrir desastres naturales entre los principales se tienen las amenazas

sísmicas, volcánicas e hidrometeorológicas, aunque dichos eventos no pueden ser predichos con anterioridad, sí es posible tomar acciones que permita reducir el impacto que podrían llegar a tener, es por esto que en la actualidad se ha trabajado con varios sistemas de alerta temprana, uno de ellos es el sistema de alerta EWBS que se lo puede complementar con el uso de aplicaciones interactivas, con estas herramientas se puede enviar mensajes interactivos a través de la televisión digital terrestre (TDT), este tipo de sistemas ayudan a que la ciudadanía tome acciones para disminuir los efectos de los distintos desastres naturales (Olmedo, Sambrano, et al., 2020).

En la actualidad los países de Latinoamérica están migrando hacia la Televisión Digital Terrestre (TDT), esto se ha podido lograr gracias al uso del estándar ISDB-Tb, en algunos países se ha iniciado con el apagón analógico, lo que permitirá que a los hogares de sus ciudadanos únicamente llegue la señal de TDT, esto puede ser una ventaja que debería ser usada para implementar un sistema EWBS además de poder agregar aplicaciones interactivas en GINGA, que brindará facilidades para alertar a la ciudadanía cuando suceda un evento que ponga en riesgo su seguridad, aunque ya existe codificadores que son capaces de son compatibles con el sistema EWBS, estos decodificadores pueden llegar a tener un elevado costo (Olmedo, Acosta, et al., 2020).

Al agregar aplicaciones interactivas al sistema EWBS en TDT es posible que este sistema se pueda conectar a otros sistemas de comunicación, de esta forma se podrá intercambiar el mensaje de alerta entre los diferentes sistemas, además de poder acoplar nuevas herramientas que permitan alertar de forma eficaz. Un ejemplo puede ser la implementación de una central telefónica institucional que envíe alertas a sus usuarios, otra alternativa puede ser que se implemente un servicio WEB en el que se encuentre información de cada cantón y permita determinar de qué lugar del país se produce la emisión de la señal de alerta, este tipo de proyectos se han venido

realizando en el país, se tiene que ya desarrollada la etapa de transmisión para un sistema de emergencia EWBS y aplicaciones interactivas para informar a la comunidad a través del proyecto de CEDIA durante los años 2018-2019, es por esto que se busca implementar la etapa de recepción para implementar en su totalidad el sistema de alerta temprana (Olmedo, Acosta, et al., 2020).

Por las razones expuestas anteriormente, se ha realizado la implementación de un receptor EWBS y aplicaciones interactivas en Ginga según el estándar ISDB-Tb. Para reducir costos de implementación se utilizó un sistema de Radio Definida por Software (SDR), cuyos los bloques de procesamiento fueron desarrollados en el software GNU Radio, a través del que se puede aportar a los trabajos que se han venido realizando hasta la actualidad, implementando nuevas funcionalidades.

Alcance del proyecto

El principal objetivo del presente trabajo investigativo ha sido diseñar un receptor robusto para Televisión Digital Terrestre (TDT), siguiendo las normativas del estándar ISDB-Tb, que permite analizar señales EWBS, además de descargar aplicaciones interactivas que vengan incluidas en el TS, utilizando el software GNU Radio para configurar un SDR del tipo ADALM-PLUTO.

El algoritmo de recepción implementado en el SDR es capaz de extraer la información necesaria para que el equipo funcione como un receptor Full-seg, con esto es posible la extracción del mensaje de emergencia guardando esta información en archivos de texto, para complementar el funcionamiento del receptor se han extraído los archivos necesarios para el funcionamiento de las aplicaciones interactivas, para observar el resultados de la extracción se lo hizo mediante el reproductor multimedia con GINGA. Los datos mencionados anteriormente han sido extraídos del TS.

Para lograr lo antes mencionado, el receptor funciona como un analizador del flujo de datos de TDT. Con el objetivo de detectar la señal de emergencia, se verificó la activación del bit 26 del TMCC. Adicionalmente, se validó ciertos parámetros para demostrar el correcto funcionamiento del receptor como pueden ser: la potencia de recepción, MER y BER.

Las pruebas que permitieron verificar el correcto funcionamiento del receptor se las realizó en el laboratorio de Televisión Digital de la Universidad de las Fuerzas Armadas – ESPE, el cual cuenta con los equipos necesarios para procesar el algoritmo de recepción. El mensaje EWBS se lo extrae en un archivo de texto, el que contendrá la información de EWBS a mostrar y los archivos de interactividad que serán reproducidos con GINGA, todo esto, una vez que el proceso de transmisión haya concluido.

Objetivos

Objetivo General

Implementar un receptor EWBS y aplicaciones interactivas Full-seg utilizando una plataforma de radio definida por software según el estándar ISDB-Tb.

Objetivos Específicos

- Realizar el estudio del estado de arte acerca de receptores para TDT bajo el estándar ISDB-Tb, así como de las especificaciones técnicas de los SDR de bajo costo, además de las características del software GNU Radio.
- Implementar un receptor Full-seg según el estándar ISDB-Tb en una plataforma SDR.
- Diseñar un algoritmo para la extracción de la señal EWBS y de aplicaciones interactivas en Ginga.

- Realizar pruebas de funcionamiento del sistema implementado para la detección de señales EWBS enviados a través de televisión digital terrestre.
- Evaluar mediante técnicas objetivas el desempeño del sistema, en función de la calidad de audio y video, así como de la extracción del mensaje el Sistema de Alerta Temprana e interactividad

Estado del arte

Un proyecto que ha sido realizado el 2021, es la implementación de un receptor EWBS One-seg según las normativas que se indican en el estándar ISDB-Tb, se implementó un Sistema de Alerta Temprana, esto se lo realizó en una plataforma de Radio Definida por Software (SDR) utilizando el hardware SDR ADALM-PLUTO y el software GNU Radio, con ayuda de estas herramientas se ha logrado extraer información acerca de la señal de emergencia enviada que se encuentra contenida en el Transport Stream (TS), este proyecto fue realizado con éxito y se lograron los objetivos planteados (Castillo, 2021).

Otro proyecto es el que se ha desarrollado en el 2016 en la Universidad de Chimborazo, el cual se trata del diseño de una aplicación de alerta temprana para alertar de eventos volcánicos, se lo realizó usando el middleware Ginga, para generar el Transport Stream su utilizó el servidor VillageFlow, el cual permitió configurar parámetros como: codificación, multiplexación, interactividad y modulación (Buenaño, 2006).

Para el tema de interactividad el proyecto Estudio, análisis e implementación de un software para construir un extractor y constructor de carrusel de datos incluido en el flujo de transporte mpeg-2 ts de televisión digital, desarrollado el 2016 en la Universidad de las Fuerzas Armadas, el cual hace uso de Matlab para implementar un algoritmo que permite extraer los elementos de interactividad contenidos en un TS, además de

organizar estos datos de forma jerárquica según la norma MPEG-2 ISO-13818-6: Extensiones para el Comando y Control de Almacenamiento de Medios Digital (DSM CC del inglés Digital Storage Media Command and Control) (Nuñez, 2016).

Un proyecto que se relaciona directamente con el presente, es el que ha sido realizado con la colaboración de CEDIA en el año 2020, se trata de un transmisor de EWBS y además se usan aplicaciones interactivas, este proyecto se ha llevado a cabo por profesores de la Universidad de las Fuerzas Armadas – ESPE, con colaboración de la Universidad de Cuenca y la Universidad Técnica de Machala, las pruebas de este prototipo se las ha realizado a través del canal Telesucesos (Olmedo, Acosta, et al., 2020).

Otro proyecto en el cual ha estado involucrada la Universidad de las Fuerzas Armadas es en el de implementar un receptor EWBS, pero añadiendo una central telefónica es Asterisk, esta es una idea muy innovadora ya que no siempre los usuarios se encuentran haciendo uso de la televisión, es por esto que la implementación de una central para telefonía IP ayudaría a solucionar este problema, alertando de forma alternativa al usuario cuando no esté haciendo uso de la televisión (Olmedo, Sambrano, et al., 2020).

En algunos países de Latinoamérica ha sido adoptado el estándar ISDB-T para TDT, este es el caso de Uruguay se ha realizado un proyecto interesante, se trata de un receptor One-seg y Full-seg a base de software, esto con el uso de un SDR, la creación de este módulo fue aprovechado en el 2018 en la Universidad de las Fuerzas Armadas – ESPE, se realizaron mejoras a este módulo además de añadir medidas objetivas para determinar el rendimiento del mismo (Flores, 2016; Garzón, 2018).

Organización

El presente trabajo de investigación, se encuentra distribuido en 5 capítulos, su distribución fue realizada para facilitar la interpretación y lectura del lector, su distribución es la siguiente:

Capítulo I. Introducción: Este capítulo se encuentran las siguientes secciones: antecedentes, justificación e importancia, el alcance del proyecto, los objetivos que se desea cumplir, el estado del arte acerca del tema de investigación, además de la organización del contenido que componen este proyecto.

Capítulo II. Marco Teórico: Se detalla la materia relacionada a la implementación del proyecto, se abarcarán temas generales de TDT, análisis de tablas PSI, análisis del TS, normas que regulan la extracción de los archivos multimedia, generalidades del software GINGA, componentes del SDR Adalm Pluto, generalidades del software GNU Radio.

Capítulo III. Desarrollo del receptor. Se da a conocer los detalles del proceso que se siguió para elaborar el receptor, empezando por indicar como se descarga y crea nuevos bloques de procesamiento en GNU Radio, y detallando el algoritmo que se siguió para la extracción de la señal EWBS y de los archivos que complementan la aplicación interactiva.

Capítulo IV. Análisis de Resultados. En este capítulo se presentan los resultados obtenidos con el uso del receptor, se muestra toda la información que puede ser extraída del mismo, así como las mediciones objetivas para determinar la calidad de la transmisión.

Capítulo V. Conclusiones. Se presenta las conclusiones y recomendaciones a las que se llegó luego de finalizar el proyecto, en base al funcionamiento y los contratiempos que se han presentado en el desarrollo del receptor full-seg.

CAPÍTULO II

Marco Teórico

Introducción

En este capítulo se abordarán todos los temas relevantes que intervienen a la hora de implementar el receptor full-seg. Inicia con una breve reseña de la televisión digital y el uso que se le está dando en el Ecuador; el proceso de recepción para señales de TDT; y una revisión de las tablas PSI (*Program Specific Information*) que servirán para extraer la información necesaria, en este caso se trabajará principalmente sobre las tablas PAT y PMT. Además, se aborda sobre el proceso de extracción del Sistema de Radiodifusión de Alerta de Emergencia (EWBS - *Emergency Warning Broadcasting System*), y para la parte de aplicaciones interactivas se determinará cual es el proceso adecuado para extraer todos estos archivos correctamente. Por último, se explicará sobre el hardware y el software que hará posible la implementación del receptor, en este caso el SDR Adalm Pluto y el programa GNU Radio.

Televisión Digital Terrestre

La tecnología está en constante evolución y a lo largo de los años los medios de comunicación la han ido implementando, para mejorar su desempeño y de las características que se puede ofrecer a los usuarios. En el caso de la televisión desde que se incorporó las transmisiones a color, la implementación de la Televisión Digital Terrestre (TDT), ha sido un cambio significativo ya que a diferencia de la televisión analógica, la transmisión es a través de codificación binaria, que le permite tener beneficios como: procesamiento, multiplexación, codificación y modulación digital de las señales tanto de audio, vídeo y datos (Morales, 2010).

De las características que más resaltan en TDT son: la capacidad de transmitir imágenes de alta definición, sonido de calidad, transmisión de datos de alta velocidad, transmisión de varios canales. Además, se puede prestar servicios de un solo programa de alta definición (HDTV) o de varios programas de menor calidad (SDTV) el cual presenta imágenes de menor calidad, pero significativamente mejor a las señales emitidas por la televisión analógica (Enríquez & Rivadeneira, 2018).

Estándares de TDT

En todo el mundo existen diferentes estándares para la transmisión y recepción de TDT, a continuación, se muestran cada uno de estos

Figura 1

Países latinoamericanos que usan el estándar ISDB-T



Nota. La figura muestra los países que han adoptado el estándar japonés ISDB-T.

Tomado de *Japoneses que contribuyen en otras partes del mundo*, 2016.

- Estándar Americano, ATSC (Advanced Television Systems Committee).
- Estándar Europeo, DVB (Digital Video Broadcasting).
- Estándar Japonés, ISDB-T (Integrated Services Digital Broadcasting).

- Estándar Chino, DTMB (Digital Terrestrial / Television Multimedia Broadcasting)

Los países de Latinoamérica en su mayoría han adoptado el estándar japonés, este estándar se ha ido adaptando según las necesidades de cada lugar, un claro ejemplo es la adaptación realizada en Brasil que toma el nombre de ISDB-Tb, se caracteriza por la alta definición, la posibilidad de movilidad, y la transmisión datos. La imagen de la Figura 1 muestra los países de Latinoamérica que han adoptado el estándar ISDB-T.

Televisión Digital Terrestre en el Ecuador

En el país, desde el año 2010 se empezó a implementar el estándar ISDB-Tb para televisión abierta, para lo que se firmó un contrato tecnológico con la empresa japonesa Hitachi Kokusai Electric Inc. la cual se encargó de la generación, procesamiento, transmisión de televisión digital y de alta definición; con el financiamiento del Japan Bank for International Cooperation (Poveda et al., 2017).

Durante esos años se ha buscado implementar la transición de televisión analógica hacia TDT, la que ofrece mejores características como son la calidad de audio, video y datos por un mismo canal optimizando el espectro electromagnético, pero el principal problema que se ha venido evaluando es, si todas las estaciones televisivas se encuentran preparadas para realizar la inversión para la compra de estos equipos, a pesar de que algunas de estas ya fueron aprobadas por la anterior Superintendencia de Telecomunicaciones (Poveda et al., 2017).

Las estaciones que fueron autorizadas para implementar TDT bajo el estándar ISDB-Tb en el Ecuador se muestran en la tabla 1.

Tabla 1
Estaciones autorizadas para transmitir TDT en Ecuador

ESTACIÓN AUTORIZADA	CIUDAD
Ecuadortv	Cuenca
Colortv	Latacunga
Oktv-Tevecorp	Machala
Ecuadortv	Guayaquil
Corporacion Ecuatoriana De Televisión	Guayaquil
Red Telesistema (RTS)	Guayaquil
Teleamazonas Guayaquil	Guayaquil
Cadena Ecuatoriana de Televisión	Guayaquil
Canal Uno	Guayaquil
Tv+ (Tevemas)	Guayaquil
Televisión Satelital	Guayaquil
Costanera (RTU)	Guayaquil
Oromar	Manta
Ecuadortv	Quito
Televisión Del Pacífico	Quito
Teleamazonas	Quito
Telesistema	Quito
Televisora Nacional	Quito
Televisión Satelital	Quito
Telesucesos	Quito
46 Uhf Abc (RTU)	Quito
Canal Uno	Quito
Unimax	Ambato

ESTACIÓN AUTORIZADA	CIUDAD
Teleatahualpa (RTU)	Santo Domingo
Espol Tv	Santa Elena

Nota. Esta tabla muestra las estaciones televisivas autorizadas para transmitir TDT de acuerdo a cada provincia del Ecuador. Tomada de Incidencia japonesa de la televisión digital terrestre en el Ecuador, 2017.

Beneficios en el Ecuador

Los principales beneficios de implementar la TDT son la calidad de audio y video, adicional a estos, se detallan otros adicionales en la tabla 2.

Tabla 2
Beneficios de usar TDT

Calidad	Debido a que la transmisión es digital la señal se vuelve más robusta a la interferencia y ruido lo que hace que aumente la nitidez, resolución de la imagen y la calidad del audio.
Movilidad y Portabilidad	Es posible que equipos móviles como celulares, tablets, laptops, etc., puedan recibir la señal de TDT.
Optimización del uso del espectro	Usa 6 MHz al igual que la televisión analógica, pero ofrece la versatilidad de transmitir varias programaciones de alta definición y de definición estándar, además de datos adicionales.
Interactividad	Integra los contenidos de televisión estos pueden ser publicitarios o de interés para el usuario.

Nota. Esta tabla muestra un resumen de los beneficios que presenta el uso de TDT.

Adaptada de Preguntas frecuentes sobre televisión digital, 2012.

Estándar ISDB-Tb

El estándar ISDB tiene 3 tipos de sistemas ISDB-S (Satélite), ISDB-T (Terrestre) e ISDB-C (cable); fue desarrollado en Japón con la finalidad de ofrecer flexibilidad, capacidad de expansión y difusión de los servicios de transmisión de multimedia. El estándar ISDB-T empezó a ser utilizado en diciembre de 2003 en Japón y pasaron 4

años hasta que Brasil adopte este estándar en 2007. El número de países que hace uso de este estándar creció rápidamente en Latinoamérica y entre ellos se encuentra Ecuador (Poveda et al., 2017).

El estándar ISDB-Tb es definido por el Sistema Brasileño de Televisión Digital (SBTVD), que es una adaptación de ISDB-T, donde se han ido implementando mejoras al estándar japonés, para que sea compatible con algunos de los países de la región. El principal cambio es el uso de compresión de video el estándar H.264/MPEG-4 AVC (Morales, 2010).

Para implementar cambios al estándar, se ha formado un grupo de trabajo denominado Foro internacional ISDB-T, que se encarga del desarrollo y armonización de los parámetros relacionados con: especificaciones de hardware en los receptores de televisión digital terrestre, Interactividad y Middleware, especificaciones del modelo de referencia para transmitir datos a los receptores de TDT y EWBS, donde se detalla la estructura de las señales que serán transmitidas por las estaciones y la interpretación de los receptores compatibles (Poveda et al., 2017).

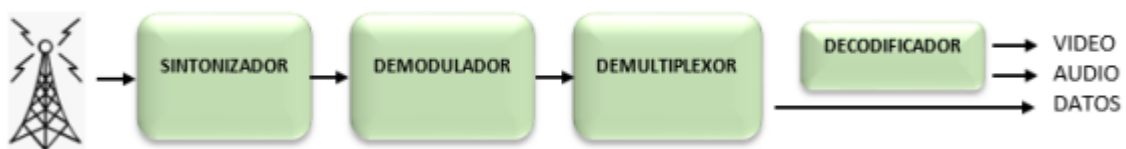
Sistema de Recepción para TDT

Para receptar las señales de TDT se debe usar equipos que sean compatibles con esta tecnología, la forma en la que se ha venido trabajando en estos últimos años hace posible que a equipos analógicos se los adapte para que sean capaces de receptar señales de audio, video y datos, es decir funcionen como receptores de TDT. Para esto se hace uso de un decodificador denominado Set-Top-Box (STB) (Castillo, 2021).

De forma general, se puede explicar el receptor de TDT de la siguiente forma: La primera etapa consta de una antena que debe ser capaz de receptar señales en las

bandas de frecuencia UHF o VHF, luego la señal recibida pasa por un circuito de FI (Frecuencia intermedia), y por el demodulador, el que se encarga de obtener la señal en banda base. En esta etapa existen varios procesos que intervienen: la demodulación OFDM, la recuperación del sincronismo, la extracción de los cuadros OFDM y la FFT (en inglés, Fast Fourier Transform); luego la señal pasa al demultiplexor que desencapsula el TS MPEG-2, separando las tramas de control de aquellas que contienen información y a través del decodificador, se reproducen las señales de audio y video. En la figura 2, se puede observar un diagrama de bloques de este proceso (Garzón, 2018).

Figura 2
Sistema de Recepción para TDT



Nota. La figura muestra el diagrama de bloques de un receptor para TDT. Tomado de Visión General del sistema de Recepción de TDT, de Implementación de un sistema de recepción de una señal de emergencia EWBS, 2021.

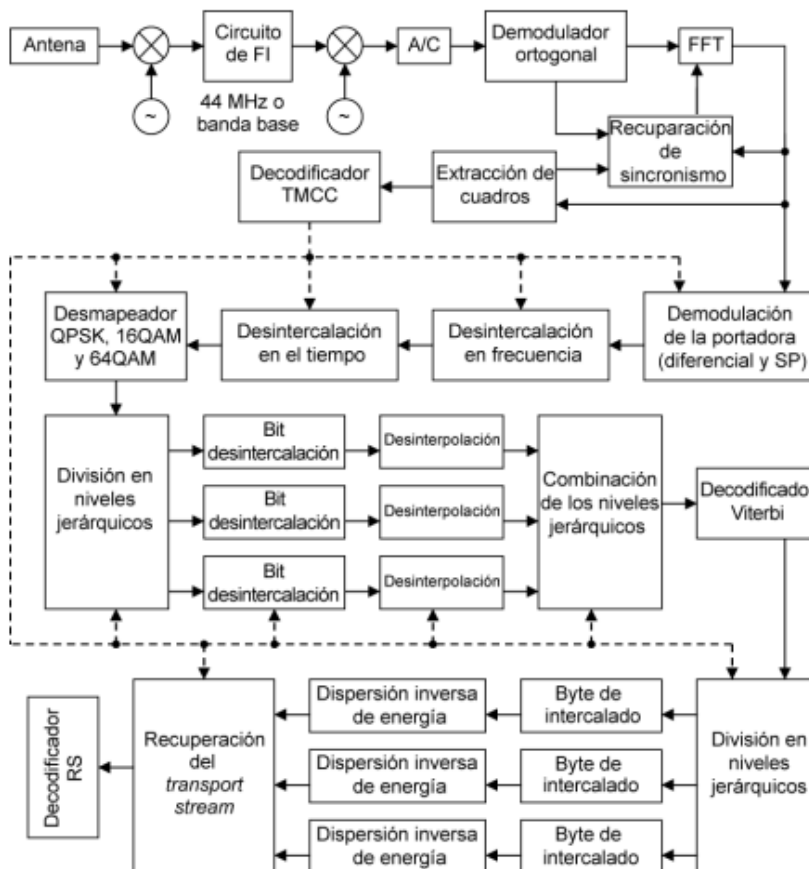
Es necesario conocer de forma más detallada de las etapas que intervienen para lograr extraer el TS de la señal emitida por un transmisor de TDT para esto se hace uso de la Norma Brasileña ABNT NBR 15604, la cual detalla las funciones que cumple cada uno de ellos, esto se detalla a continuación.

Procesamiento de la señal para un receptor de TDT

La norma ABNT NBR 15604 detalla cada uno de los bloques que intervienen en este proceso, con su respectiva función, lo que se puede observar en la figura 3.

Figura 3

Procesamiento de la señal en el front end para full-seg



Nota. La figura muestra los bloques de procesamiento de un receptor full-seg. Tomado de Norma Brasileña ABNT NBR 15604 Televisión digital terrestre -Receptores, de Procesamiento de señal en el front-end para full-seg, 2007.

- **Selección de canales:** se especifica el canal de VHF alto o UFH para receptores Full-Seg.
- **Demodulador Ortogonal:** obtiene los símbolos de OFDM que han sido recibidos.
- **Recuperación del sincronismo:** la señal demodulada a través de la recuperación del sincronismo de los símbolos OFDM y muestreo de la

frecuencia FFT se obtiene usando como parámetros al modo de recepción, así como del intervalo de guarda.

- **FFT:** esta operación se debe realizar con un periodo apropiado, es por esto que se ejecuta por un periodo correspondiente a la duración de un símbolo de OFDM.
- **Extracción de cuadros:** la extracción de la señal de sincronización del cuadro OFDM se la debe realizar a través de la señal del TMCC.
- **Decodificador del TMCC:** la información TMCC contiene información del sistema de transmisión, permite el control de varios módulos y en su estructura se encuentra la bandera de emergencia.
- **Demodulación de la portadora:** Tomando la información TMCC, se procede a demodular con una demodulación diferencial DQPSK, demodulación síncrona a través de la colocación de la señal piloto (SP) utilizando QPSK, 16QAM o 64 QAM para supervisar la amplitud y la fase de la señal.
- **Desentrelazado:** se aplica desentrelazamiento en tiempo y frecuencia para evitar efectos de desvanecimiento selectivo, movilidad y portabilidad.
- **Desmapeador:** se encarga del desmapeo de la señal puede ser QPSK, 16QAM, 64QAM, de acuerdo con la información de fase y amplitud, esta información se extrae a nivel de bits.
- **División de niveles jerárquicos:** si la información TMCC indica la transmisión jerárquica, se realiza la división cada 204 bytes entre el byte de sincronización (0x47) de los paquetes que contienen información dentro del TS y el byte de sincronización del siguiente paquete.
- **Bit de intercalación:** cada nivel jerárquico se encarga de ejecutar la desintercalación.

- **Bit de interpolación:** se realiza en cada nivel jerárquico, se hace uso de la tasa de código convolucional que se indica en la información TMCC.
- **Decodificador de Viterbi:** este bloque emplea un algoritmo para corregir errores que se pueden dar en la transmisión por los efectos del canal.
- **Byte entrelazado:** se ejecuta en desentrelazado byte a byte con el fin de mitigar los efectos de desvanecimiento.
- **Dispersión inversa de energía:** este bloque ayuda a la eliminación de ráfagas de ceros y unos, se hace uso de una compuerta XOR en cada bit de entrada, solo se exceptúa el byte de sincronismo.
- **Recuperación del TS:** el orden los paquetes que conforman el TS y la localización temporal del PCR deben ser igual a los transmitidos. En el estándar ISDB-T se define al Broadcast Transport Stream (BTS) el cual contiene 204 bytes
- **Decodificador RS:** código RS (204,188), se elimina los 16 bytes sobrantes y se obtiene un TS de 188 bytes, corrigiendo errores y eliminando la redundancia.

El correcto entendimiento de los bloques que han sido detallados anteriormente es de suma importancia para el correcto desarrollo del proyecto, puesto que a partir de estas premisas se deben elaborar o colocar los respectivos bloques para obtener un receptor que cumpla con las especificaciones que indica el estándar.

Características importantes

La información puede ser multiplexada en tres capas A, B y C, para esto se hace uso de la multiplexación OFDM, donde el prefijo cíclico debe ser expresado en función de la duración del tiempo por símbolo. Para determinar el número de portadoras, se hace referencia los modos de transmisión 1, 2 y 3; siendo 2048, 4096, 8192 portadoras

correspondientes a cada modo de transmisión. Para una transmisión y recepción full-seg se deben recibir los 13 segmentos transmitidos, donde se puede tener transmisiones de alta calidad, así como otros elementos que complementan la televisión digital, como es el caso de las aplicaciones interactivas. En la tabla 3 se detallan algunas características importantes para la codificación del canal según la ARIB STD-B31:2005 (Castillo, 2021).

Tabla 3
Parámetros del Sistema de transmisión ISDB-Tb

Parámetros	Valores
Ancho de Banda del canal	6 MHz
Número de segmentos	13
Ancho de Banda de un segmento	60 MHz/14=428.57 MHz
Ancho de banda utilizado	5.575 MHz (modo 1) 5.573 MHz (modo 2) 5.572 MHz (modo 3)
Número de portadoras activas de todo el espectro	1405 (modo 1) 2809 (modo 2) 5617 (modo 3)
Esquema de modulación	QPSK, DQPSK, 16QAM, 64QAM
Duración del símbolo activo	252 μ s (modo 1) 504 μ s (modo 2) 1008 μ s (modo 3)
Portadoras activas por segmento	96 de datos y 12 pilotos = 108 (modo 1) 192 de datos y 24 pilotos = 216 (modo 2) 384 de datos y 48 pilotos = 432 (modo 3)
Portadoras de datos por segmento	96 (modo 1) 192 (modo 2) 384 (modo 3)
Separación de portadoras	428.57 kHz/108 = 3.968 kHz (modo 1) 428.57 kHz/216 = 1.984 kHz (modo 2) 428.57 kHz/432 = 0.992 kHz (modo 3)
Duración del intervalo de guarda	1/4, 1/8, 1/16 y 1/32
Codificación de canal (Tasa de código convolucional)	1/2, 2/3, 3/4, 5/6, 7/8 0, 1, 2, 4 (modo 1)
Profundidad del entrelazado temporal	0, 2, 4, 8 (modo 2) 0, 4, 8, 16 (modo 3)

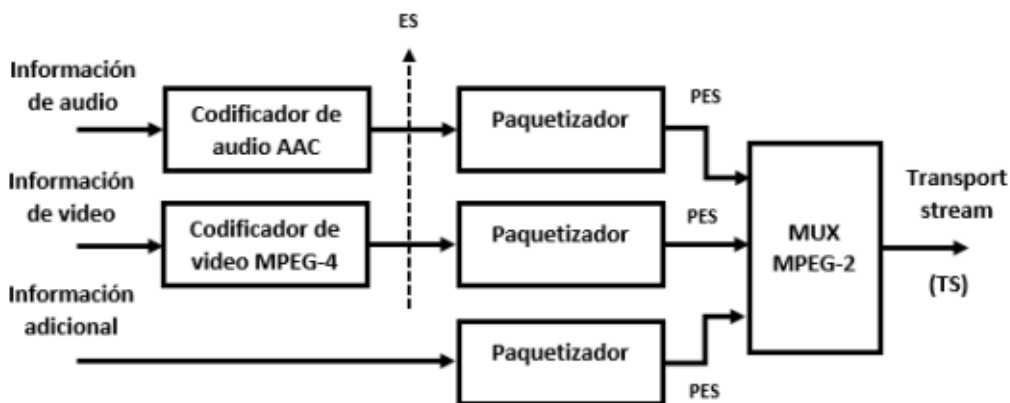
Nota. La tabla detalla algunos de los parámetros de transmisión del estándar ISDB-Tb.

Tomado de Norma Brasileña ABNT NBR 15601 Televisión digital terrestre - Sistema de transmisión ISDB-Tb, de Parámetros del sistema de transmisión, 2007.

Transport Stream

Figura 4

Multiplexación del Transport Stream para una programación



Nota. La figura muestra un ejemplo funcional de la multiplexación de los diferentes elementos que se transmiten en TDT. Adaptada de Digital Video Broadcasting Technology Standards, and Regulations, de Functional representation of the MPEG-2 systems, pp.153, 199.

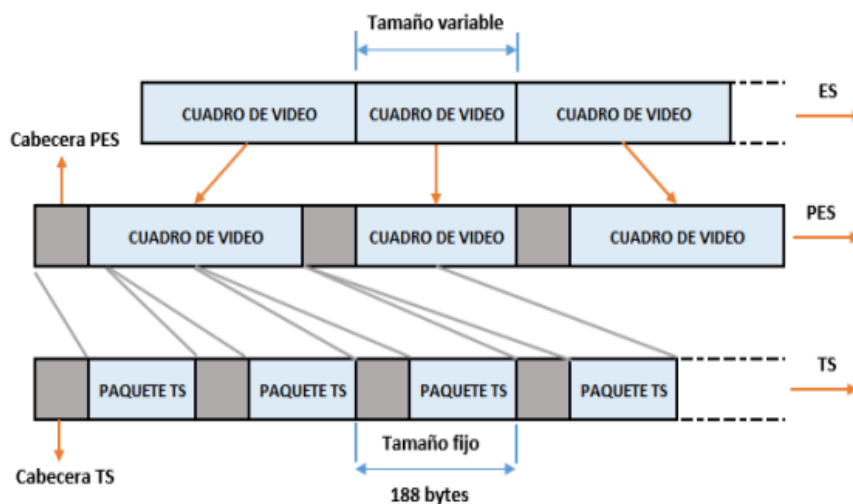
El estándar ISDB-Tb para enviar audio, video y datos usa el estándar MPEG-2 (Moving Picture Experts Group 2), que se encarga de multiplexar todos los elementos y enviarlos en un solo Transport Stream y el receptor debe ser capaz de recuperar el TS y reproducir el audio, video y ser capaz de extraer los archivos de aplicaciones interactivas que pueden venir incluidos en la transmisión, en la figura 4 se puede observar un ejemplo de cómo funciona MPEG-2 para una sola programación (De Bruin & Smits, 1999).

En la figura 4 se puede apreciar el proceso que se debe seguir para obtener un TS para transmitirlo, las señales de audio y video entran a un proceso de compresión y codificación, luego de este proceso cada uno de estos elementos forman un Flujo Elemental (ES, Elementary Stream), luego estos datos ingresan al paquetizador el cual se encarga de colocar las cabeceras necesarias para el control tanto en la transmisión

como en la recepción. En este punto se incluyen los datos extra que se desea transmitir. Cuando ya pasan por el paquetizador se obtiene un PES (Packetized Elementary Stream), una vez se tiene el PES de audio, video y datos se multiplexa esta información para finalmente generar el TS que se desea transmitir (Pozo et al., 2015).

El proceso de formación de un TS se muestra en la figura 5 para diferentes cuadros de video y de tamaño variable, los que son el resultado de la codificación MPEG-4. Con esto se obtiene el ES. Como se explicó anteriormente, estos datos entran al paquetizador que se encarga de colocar la cabeceras que servirán para el control de la transmisión y recepción, después de colocar estos datos de control, el multiplexador se encarga de formar un solo flujo de 188 bytes, esto para que el control en la detección de errores sea más simple, además de la multiplexación de varios programas en un mismo flujo (Villamari et al., 2012).

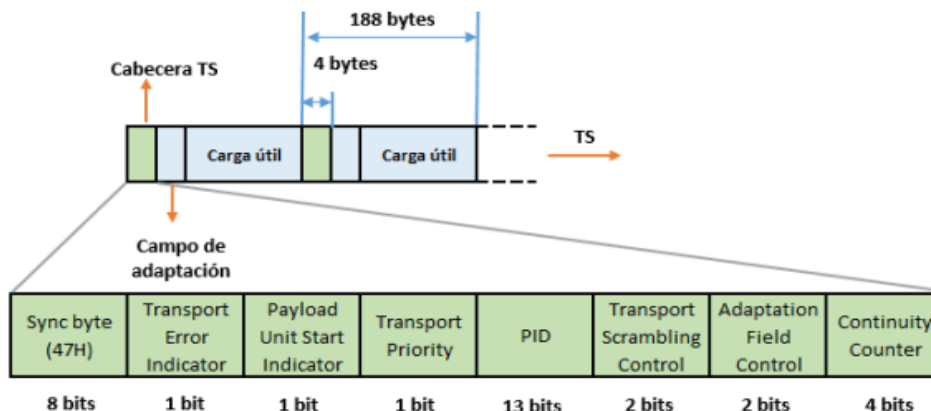
Figura 5
Proceso para obtener un TS



Nota. La figura muestra el proceso para obtener un TS para cuadros de video de tamaño variable. Tomado de Implementación de un sistema de recepción de una señal de emergencia EWBS, de Formación de ES, PES y TS, 2021.

Figura 6

Formato de un TS con su respectiva cabecera



Nota. La figura muestra la composición de un flujo TS con su cabecera. Tomado de Implementación de un sistema de recepción de una señal de emergencia EWBS, de Formato de un paquete TS con su cabecera, 2021.

De los 188 bytes que componen el TS los 4 primeros corresponden a la cabecera y los 184 restantes corresponden a la carga útil o payload, incluyendo el campo de adaptación. Dentro de la carga útil se encuentra información de: datos, servicios o información de relleno, tal como se muestra en la figura 6.

Tablas PSI/SI

Como se mencionó anteriormente, toda la información que se desea transmitir se la envía a través de un flujo único conocido como el Transport Stream. Para que los receptores sean capaces de descifrar la información que viene incluida en este flujo de información, es necesario el uso de las tablas PSI/SI (*Program Specific Information/Service Information*), con su ayuda se puede decodificar las transmisiones de programas existentes, localizar la información de audio, video y datos, además hacen posible la sincronización automática entre el transmisor y el receptor, todo esto se encuentra especificado en la norma brasileña (15603-1, 2007).

Tabla PSI

La información que contienen las tablas PSI es de gran ayuda para los receptores, que son los encargados de demultiplexar toda la información que viene incluida dentro de un TS, existen diferentes tablas que cada una cumple una función diferente, y extrae información importante. A continuación, se detalla la estructura de las tablas PAT (*Program Associate Table*) y PMT (*Program Map Table*), las que son base para el desarrollo del proyecto.

PAT (Program Associate Table)

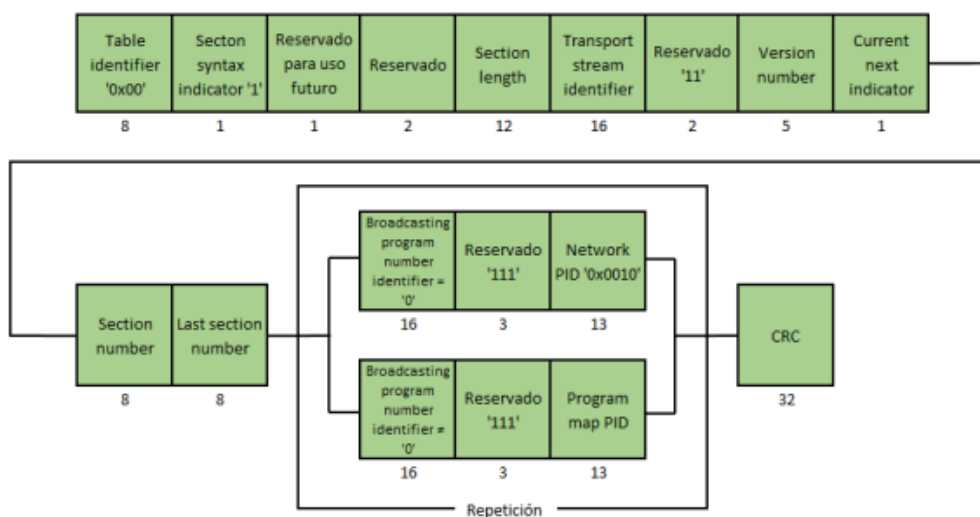
En la tabla de asociación de programa viene incluida de forma obligatoria la información acerca del número de programaciones, así como los valores del PID donde se encuentran ubicadas dentro del Transport Stream. Esta es la primera tabla en ser analizada para conseguir el propósito planteado para este proyecto, se hace uso del identificador de paquete PID, para conocer cuál de ellos transporta la información de ese programa en los 13 bits de la cabecera de la tabla PMT, la cual será mencionada más adelante.

El PID que identifica la tabla PAT es el 0x00 en hexadecimal o 0 en decimal. En la figura 7 se muestra cómo se compone esta tabla y como deberían ser analizados sus bytes, ya que cada uno de estos contienen información importante que ayuda a determinar los programas que posee un TS (Castillo, 2021).

Como se ha mencionado anteriormente, en un mismo TS pueden venir incluidas varias programaciones, los PID de cada una de estas deben obligatoriamente venir incluidas dentro de la PAT, el identificativo de cada uno se encuentra en el campo de *Network PID* o *Program Map PID*. Existen varias consideraciones para tomar en cuenta entre las más importantes se puede mencionar que: el campo *Program Number*, con un

valor de '0' debe ser utilizado únicamente en la tabla NIT con un valor de *Network PID* igual a 0x0010. El otro caso se da cuando el valor del *Program Number* es diferente de '0', el PID corresponde a la tabla PMT, y de esta forma se obtiene a través de la PAT el PID que permitirá localizar el PID de la tabla PMT para poder analizarla y tomar los datos necesarios (ABNT NBR 15602-1, 2007).

Figura 7
Estructura de la Tabla PAT



Nota. La figura muestra la estructura de datos de la tabla PAT. Tomada de la Estructura de datos de la PAT, de la Norma Brasileña ABNT NBR 15603-1 Primera edición, 2007

PMT (Program Map Table)

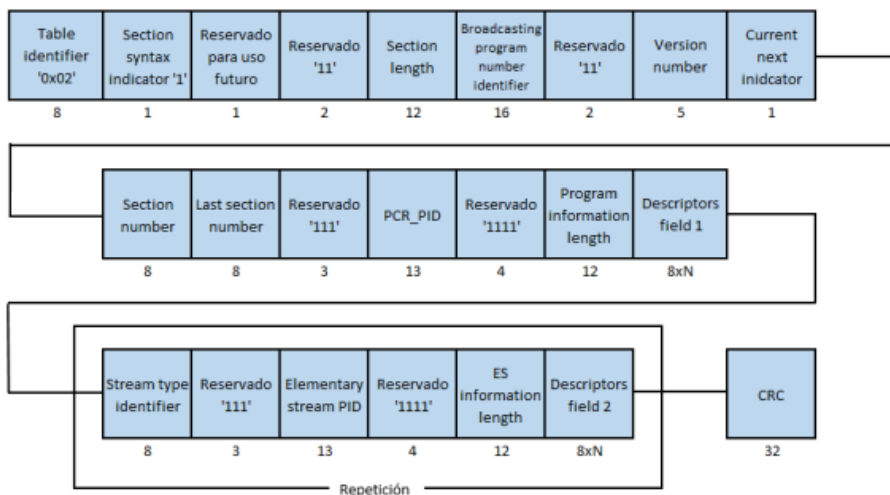
La función de la tabla de mapeo de programa es especificar el programa correspondiente al PID encontrado, esto se hace para conseguir los servicios de radiodifusión que vienen incluidos en un determinado programa, así como la referencia de hora del sistema o Program Clock Reference (PCR), como se mencionó existe una PMT para cada programa que contenga el TS, y de la misma se puede extraer información acerca de datos, audio, video y referencia de reloj, así como los

descriptores asociados para cada servicio, es por esto que el análisis de esta tabla es importante ya que de aquí se puede extraer el mensaje de emergencia EWBS, así como los archivos de la aplicación interactiva (Castillo, 2021).

En la figura 8, se muestra de forma abierta como deben ser analizados los 188 bytes que componen la PMT, en este caso y eliminando los bytes de la cabecera, se tiene que el byte identificador de la tabla o *Table identifier* debe ser 0x02. Cabe recalcar que una PMT puede o no contener descriptores para cada servicio.

Para el análisis de esta tabla es necesario ir descomponiendo los bytes para tomar los bits que son necesarios, el campo importante en este caso es el *Program Information Length*, el cual indica si existe o no un descriptor para este programa, si su valor es diferente de '0', se debe analizar el descriptor, para este caso debe existir un descriptor para el mensaje de emergencia EWBS (15603-1, 2007).

Figura 8
Estructura de la Tabla PMT



Nota. La figura muestra la estructura de datos de una tabla PMT. Tomada de la Estructura de datos de la PMT, de la Norma Brasileña ABNT NBR 15603-1 Primera edición, 2007

Información del TMCC

Para ayudar a los receptores con los parámetros de decodificación y demodulación se usa la información del TMCC, en la cual se encuentran varios datos como: el sistema de identificación, el indicador de los parámetros de conmutación del transmisor, la flag de recepción parcial, la flag de activación del servicio de emergencia que nos servirá para detectar si se ha activado o no la alarma, así como de la información actual y próxima información (ARIB, 2005).

Tabla 4
Información del TMCC

Bit Asignado	Descripción	
B ₂₀ -B ₂₁	Sistema de identificación	
B ₂₂ -B ₂₅	Indicador de los parámetros de conmutación del transmisor	
B ₂₆	Inicio de flag para la alarma de emergencia de radiodifusión	
B ₂₇	Flag de recepción parcial	
B ₂₈ -B ₄₀	Información Actual	Información de los parámetros de transmisión para la capa jerárquica A
B ₄₁ -B ₅₃		Información de los parámetros de transmisión para la capa jerárquica B
B ₅₄ -B ₆₆		Información de los parámetros de transmisión para la capa jerárquica C
B ₆₇	Flag de recepción parcial	
B ₆₈ -B ₈₀	Próxima Información	Información de los parámetros de transmisión para la capa jerárquica A
B ₈₁ -B ₉₃		Información de los parámetros de transmisión para la capa jerárquica B
B ₉₄ -B ₁₀₆		Información de los parámetros de transmisión para la capa jerárquica C
B ₁₀₇ -B ₁₀₉	Corrección del valor de desvío de fase para segmentos de transmisión conectado	
B ₁₁₀ -B ₁₂₁	Reservado	

Nota. La tabla muestra la distribución de los bits de la información del TMCC. Esta tabla muestra la información del TMCC. Tomada de TMCC Information, de TRANSMISSION

SYSTEM FOR DIGITAL TERRESTIAL TELEVISION BROADCASTING ARIB STD-B31

Versión 2.2, 2014

En la tabla 4 se muestran los bits con la correspondiente información que contiene cada uno de ellos, de los principales se puede observar que el bit 26 es el que nos indica si ha sido o no activada la señal de emergencia, además se muestra como es la distribución jerárquica de las capas para la recepción, en el desarrollo del proyecto se podrá observar esta información.

Sistema de alerta de emergencia para el estándar ISDB-T

El estándar japonés ISDB-T ha sido adoptado por varios países de centro y Sudamérica, los cuales lo han ido adaptando según las necesidades de la región, actualmente en el Ecuador se usa dicho estándar con algunas modificaciones que se han realizado en Brasil, el uso de este estándar para la transmisión de TDT ayuda a brindar un sin número de beneficios a los usuarios, desde la calidad de la imagen y sonido hasta la implementación de aplicaciones interactivas a través del Middleware GINGA, además de poder agregar mensajes de emergencia ante cualquier evento que requiera atención de la ciudadanía (Llanos, 2020).

El Sistema de Emergency Warning Broadcast System (EWBS), el cual se encuentra detallado en el estándar ISDB-Tb, se ha venido implementando en diferentes países. Un ejemplo de su buen funcionamiento se lo puede apreciar en el Perú, dicho país ha optado por usar el estándar japonés desde el 2009, y en el año 2015 realizó pruebas en un ambiente real de este sistema de alerta de emergencia, esto se logró gracias a la colaboración del Instituto de Radio y Televisión del Perú (IRTP) (Silva, 2016).

Emergency Warning Broadcasting System (EWBS)

Integrar varias tecnologías para la implementación de sistemas de alerta temprana tiene una serie de beneficios para precautelar la seguridad de la ciudadanía, y justamente esto se obtiene con el uso de los mensajes EWBS que puede ser implementado en señales de radiodifusión analógicas como digitales, el principio de esta tecnología es enviar de a través de Broadcast señales que son recibidas por los equipos terminales que se encuentran en espera de este tipo de mensajes, con esto se busca alertar de forma temprana a la ciudadanía de cualquier evento que ocurra y así puedan tomar las medidas necesarias para evitar mayores daños (Shogen, 2009).

Para controlar la transmisión y la recepción de la señal EWBS, se tienen algunos indicadores que aportarán para su correcta ejecución, entre ellos se tiene una bandera que indica la activación o el término de un evento de emergencia, esto sirve para que el receptor detecte un cambio y empiece con el proceso para mostrar el mensaje al usuario, otro parámetro importante que interviene en este proceso son los códigos de área que sirven para delimitar el lugar en el cual ocurrió la emergencia, de esta forma se evita que usuarios que se encuentran fuera del rango de mismo reciban alertas, estos códigos están regulados dentro del estándar ISDB-Tb (Llanos et al., 2017).

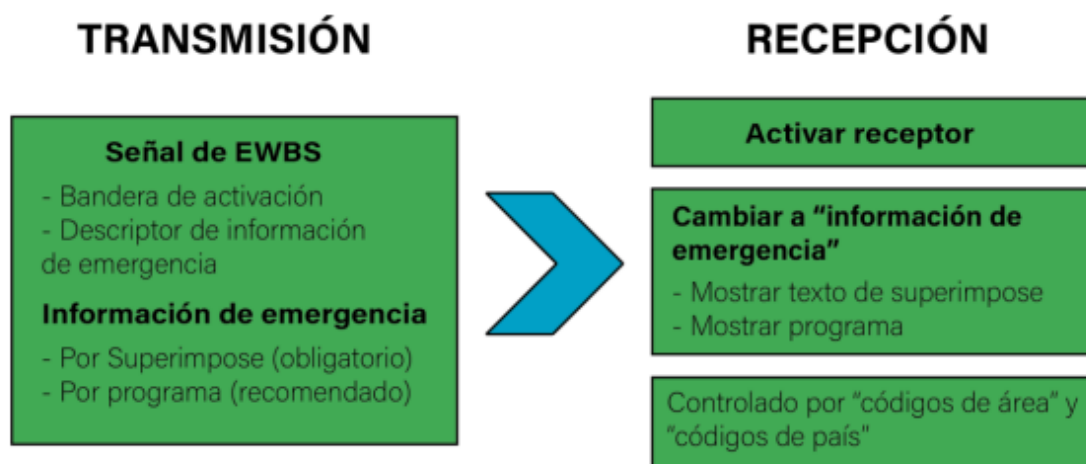
La unión de varios parámetros que vienen incluidos dentro del TS, son de suma importancia para la correcta transmisión y recepción del sistema de alerta de emergencia, se debe trabajar de forma conjunta con la bandera de activación del TMCC que se encuentra en el bit 26 del mismo y el descriptor correspondiente que se contiene en el programa analizado por la tabla PMT; dentro de los cuales se deben incluir dos elementos de suma importancia que se mencionan a continuación (ISDB-T, 2015):

- Transmisión de mensajes de texto de superimpose
- Transmisión del programa

En la figura 9, se puede apreciar en forma de diagramas de bloque el proceso que se debe seguir para la correcta transmisión y recepción de una señal de emergencia EWBS, los receptores que son capaces de recibir este tipo de señales se encuentran en stand by, esperando que se envíe la señal, al recibir la misma, se interrumpe la transmisión normal y se muestra el mensaje de alerta, como se ha mencionado anteriormente este proceso se lo realiza con sus respectivos códigos de área.

Figura 9

Diagrama general de un Sistema EWBS



Nota. La figura muestra el esquema general de un sistema EWBS. Adaptado de Funciones básicas de EWBS, de ISDB-T Documento de Armonización Parte 3: Sistema de Alerta de Emergencia EWBS (p.5), 2013

Sistema EWBS para radiodifusión digital

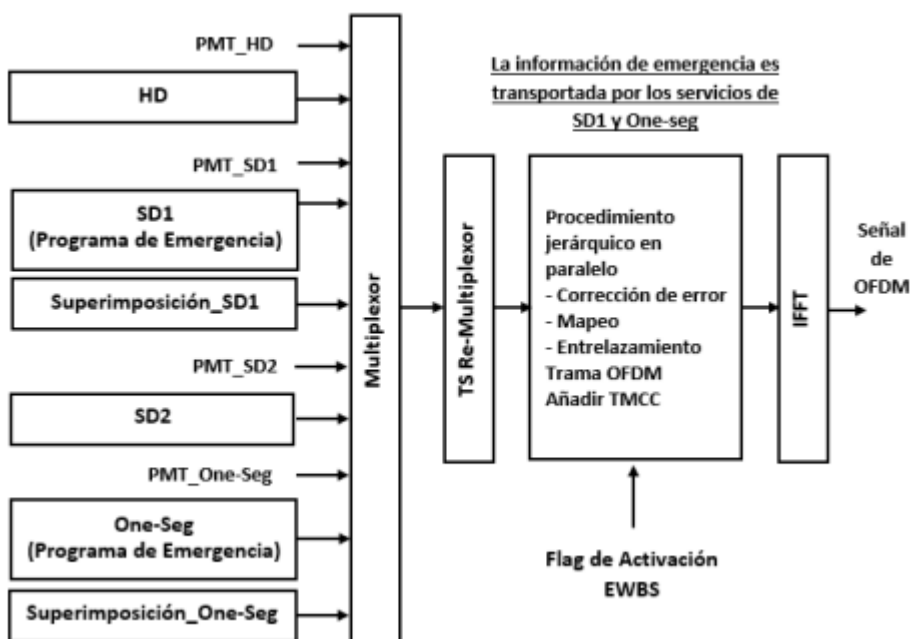
El Sistema EWBS se lo empezó a utilizar en Japón en el año 2000 en un principio para transmisiones satelitales, posteriormente en el año 2003 se lo utilizó para transmisiones de radiodifusión digital y se siguió expandiendo a los demás países que han adoptado el estándar ISDB-T. Como se ha mencionado antes una gran ventaja que

brinda el servicio de TDT, es poder transmitir varios programas en un mismo flujo TS, eso implica tener una misma transmisión pero en diferente calidad de audio y video, lo cual hace que sea más fácil poder llegar a todos los usuarios con la transmisión y así mismo con el mensaje de alerta (Poveda et al., 2017).

Como se puede tener varios programas en un mismo TS, es importante conocer un esquema en el cual se transmita un mensaje de alerta EWBS. En la figura 10, se puede observar cómo funciona la multiplexación de varios servicios para ser enviados por un mismo TS, adicional se observa cómo son incluidos los mensajes de superimposición para mostrar el mensaje de alerta si este fuera el caso.

Figura 10

Multiplexación de 4 servicios y mensaje EWBS



Nota. La figura muestra la multiplexación de 4 servicios en un mismo TS. Adaptada del Ejemplo de operación de transmisión del EWBS, de ISDB-T HARMONIZATION DOCUMENT Part 3: Emergency Warning Broadcast System EWBS, 2015

Recepción de EWBS

Existen varias formas de controlar la transmisión y recepción de las señales EWBS, para el proyecto es necesario analizar con más detenimiento el proceso de recepción, en el documento de armonización ISDB-T Parte 3, se detallan los requisitos de los dispositivos receptores de señales EWBS, entre las principales resalta que deben ser capaces de mostrar el texto de superimposición en sus pantallas, así como de colocar el código de área para reconocer si la emergencia está ocurriendo en su localidad o no.

El algoritmo seguido para recibir la señal de emergencia, se basa en el análisis del TS, el primer paso es verificar la información del TMCC. El bit 26 indica la activación del sistema EWBS ante cualquier evento de emergencia, luego se analiza la tabla PMT que contiene el descriptor EWBS, para terminar con el proceso se procede a verificar el estado del bit 26 de TMCC, este algoritmo será explicado con más detalle en el capítulo 3 (ISDB-T, 2015).

Descriptor de información de la emergencia

Las tablas PSI ayudan a realizar un análisis completo del TS el cual contiene información importante que permite recibir las señales de audio, video y datos, así como extraer toda la información que viene incluida dentro de los descriptores correspondientes a los diferentes servicios dentro de un mismo flujo TS. En la tabla 5 se indican los estándares en los que es usado el descriptor de emergencia (Shogen, 2009).

Tabla 5
Estándares que ocupan el descriptor EWBS

Estándar	Recomendación
ISDB-TSB	UIT-R BS.1114 (Sistema F)
ISDS-T	UIT-R BT.1306 (Sistema C)

Estándar	Recomendación
ISDB-TS	UIT-R BO.1408
Sistemas de radiodifusión por satélite que utilizan la banda de 2.6 GHz	UIT-R BO.1130 (Sistema E)

Nota. La tabla muestra los diferentes estándares con su respectiva recomendación para usar el descriptor de emergencia EWBS. Tomado de a la información de Handbook on Emergency Warning Broadcasting Systems, 2009.

Figura 11
Descriptor EWBS



Nota. La figura muestra el esquema del descriptor EWBS. Tomada de la Estructura de datos del descriptor de emergencia, de la Norma Brasileña ABNT NBR 15603-1 Primera edición, 2007

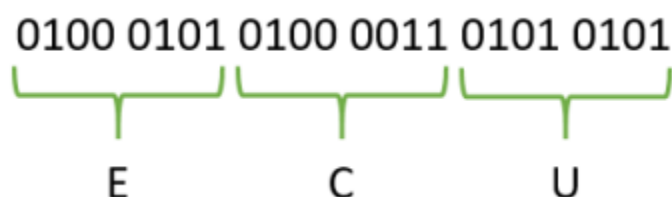
Para el estándar ISDB-Tb, en la tabla PMT se incluye un descriptor EWBS que permite obtener información importante para el sistema EWBS, el análisis de la tabla consiste en descomponer los bytes del TS según lo indica la figura 11, de aquí se puede extraer información importante, como la activación de la bandera de emergencia, además de los códigos de área, el identificador de este descriptor es 0xFC.

Códigos de área

Como se mencionó anteriormente, los códigos de área sirven para determinar los sectores involucrados en una emergencia, los mismos deben estar almacenados en el receptor para determinar si se activa o no el sistema de emergencia, estos códigos se

encuentran en las tablas PMT que contienen el descriptor EWBS. Para su organización, cada país que usa el estándar ISDB-T se encarga de generar sus propios códigos regulados por una entidad del gobierno, en el caso de Ecuador la Secretaría Nacional de Riesgos y Emergencias es la encargada de generar y dar a conocer los códigos de área que se usan dentro del país (Castillo, 2021).

Figura 12
Código de Área para Ecuador



Nota. La figura muestra el Código de Área de 3 caracteres correspondiente a Ecuador. Tomada de Codificación de Ecuador en función del código de 3 caracteres, de Implementación de un sistema de recepción de una señal de emergencia EWBS, 2021

Los códigos de área se necesitan 24 bits que representan 3 caracteres para determinar al área en cuestión, en la tabla 6 se muestran los códigos que identifican a cada país, los caracteres se codifican según la norma ISO / IEC 8859-1. En la figura 12 se tiene un ejemplo de lo mencionado para determinar los 3 caracteres para Ecuador (ARIB, 2008).

Tabla 6
Código de Área por Países

País	Código de 3 caracteres	Código de país
Argentina	ARG	0100 0001 0101 0010 0100 0111
Bolivia	BOL	0100 0010 0100 1111 0100 1100
Botswana	BWA	0100 0010 0101 0111 0100 0001

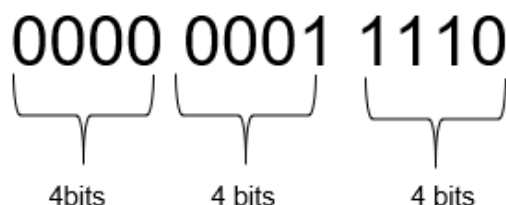
País	Código de 3 caracteres	Código de país
Brasil	BRA	0100 0010 0101 0010 0100 0001
Costa Rica	CRI	0100 0011 0101 0010 0100 1001
Chile	CHL	0100 0011 0100 1000 0100 1100
Ecuador	ECU	0100 0101 0100 0011 0101 0101
Guatemala	GTM	0100 0111 0101 0100 0100 1101
Honduras	HND	0100 1000 0100 1110 0100 0100
Japón	JPN	0100 1010 0101 0000 0100 1110
Maldivas	MDV	0100 1101 0100 0100 0101 0110
Nicaragua	NIC	0100 1110 0100 1001 0100 0011
Paraguay	PRY	0101 0000 0101 0010 0101 1001
Filipinas	PHL	0101 0000 0100 1000 0100 1100
Perú	PER	0101 0000 0100 0101 0101 0010
Sri Lanka	LKA	0100 1100 0100 1011 0100 0001
Uruguay	URY	0101 0101 0101 0010 0101 1001
Venezuela	VEN	0101 0110 0100 0101 0100 1110

Nota. Códigos de Área por Países para quienes ocupan el estándar ISDB-T. Adaptada de la tabla Código de país, de ISDB-T Documento de Armonización Parte 3: Sistema de Alerta de Emergencia EWBS, 2013.

Adicional a los códigos mostrados en la tabla 6 cada país se encarga de generar códigos internos, en el caso de Ecuador se encuentran divididos para las 24 provincias y sus respectivos cantones, donde la Secretaría Nacional de Riesgos y Emergencias es la encargada de dar a conocer estos códigos. En la figura 13 se tiene un ejemplo del código correspondiente al cantón Cayambe, de la Provincia de Pichincha.

Como se puede apreciar en la figura 13 el código para cada cantón del Ecuador fue definido un código de 12 bits. Los receptores deben ser capaces de ajustar este código ya sea por medio del GPS o de poder introducirlo de forma manual (ISDB-T, 2015).

Figura 13
Código de Área para el cantón Cayambe



Nota. La figura muestra los 12 bits correspondientes al código del Cantón Cayambe. Obtenido de Area code assignment, de ISDB-T HARMONIZATION DOCUMENT Part 3: Emergency Warning Broadcast System EWBS, 2015.

Sobreimposición

Tabla 7
Señalización para la sobreimposición

Descripción	Tipo de Flujo	Tag del componente
Sobreimposición para full-seg	0x06	0x38
Sobreimposición para one-seg		0x88

Nota. La tabla muestra el tag del componente de sobreimposición para one-seg y full-seg. Adaptada de Superimpose signaling, de ISDB-T HARMONIZATION DOCUMENT Part 3: Emergency Warning Broadcast System EWBS, 2015.

El texto de sobreimposición se encuentra definido para transmisiones one-seg y full-seg, por tratarse de un sistema de alerta de emergencia esto suele ser tratado con mayor prioridad con respecto a otros contenidos, el servicio de sobreimposición es transmitido

a través de un PES que viene incluido en la PMT con un tipo de flujo igual a 0x06, en la tabla 7 se tiene a detalle el tipo de flujo y el tag del componente, el servicio de sobreimposición se encuentra especificado en (ARIB, 2006).

Aplicaciones interactivas en TDT

La televisión digital tiene varias ventajas, una de ellas es que a través del TS es posible transmitir información adicional, por ejemplo, transmitir datos en un mismo flujo, esto es útil para trabajar con aplicaciones interactivas. Para el estándar ISDB-Tb se ha establecido que las mismas pueden ser reproducidas con el uso de la aplicación GINGA. En esta sección se detallan los componentes de las aplicaciones interactivas y el proceso que se debe seguir para extraer los archivos, dichos documentos vienen incluidos en el TS.

GINGA

Para estándar ISDB-Tb se ha determinado que es posible crear y recibir aplicaciones interactivas gracias al uso del Middleware. En la figura 14 se puede observar que en la versión brasileña del estándar se hace uso de GINGA mientras que el estándar japonés original usa ARIB. Los países sudamericanos que adoptaron el estándar con mejoras brasileñas también utilizan GINGA, middleware que se encuentra compuesto por dos subsistemas entrelazados que facilitan la interpretación de aplicaciones interactivas empleando paradigmas de programación declarativa o GINGA-NCL, imperativa o GINGA-J, también se puede dar el caso de una combinación de ambas (Pillajo et al., 2016).

GINGA fue desarrollado por el laboratorio TeleMídia de la PUC-Rio, desde el principio pensando en la inclusión social-digital, es por esto que es un software libre, lo que facilita su adaptación a las diferentes necesidades de los usuarios, ha sido utilizado para la implementación de las llamadas aplicaciones de inclusión, tales como: T-

Government, T-Health y T-Leraning. La forma en la que está constituido el desarrollo del software permite a sus usuarios intervenir para seguir mejorando las prestaciones de la misma, es por esto que existe la comunidad GINGA (GINGA, 2022).

Figura 14

Estructura del estándar ISDB-T e ISDB-Tb



Nota. La figura muestra la estructura de los estándares comparando los niveles que han sufrido cambios. Tomado de Televisión Digital Terrestre, de Sistema ISDB-T vs ISDB-TB, 2020.

Norma ISO 13818-6

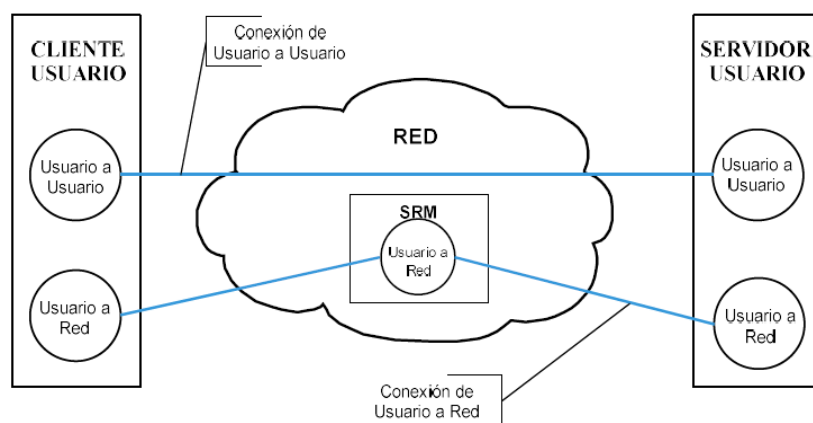
En el estándar ISDB-T, se detalla cómo es posible transmitir audio, video y datos, los archivos que constituyen las aplicaciones interactivas son transportados en paquetes que llevan la información del directorio, así como del contenido completo de las aplicaciones, para esto se los debe convertir en un carrusel de datos que se encuentra descrito en la norma MPEG-2 ISO-13818-6 (Villamarin et al., 2012).

La norma MPEG-2 ISO-13818-6: Extensiones para el Comando y Control de Almacenamiento de Medios Digital (DSM-CC, *Digital Storage Media Command and Control*) establece protocolos para las especificaciones DSM-CC, que son de ayuda para proveer soporte a las tecnologías multimedia, con esto es posible explorar, seleccionar, descargar y controlar diferentes tipos de flujo de bits, esto gracias a la Sesión, la cual es definida como el conjunto de recursos que aportan para proveer un servicio (Nuñez, 2016).

La norma define 2 modelos para transmitir y recibir aplicaciones interactivas, la primera es una conexión punto a punto denominada Usuario a Usuario, y la otra es una conexión punto a multipunto o Usuario a Red, en la figura 15 se puede apreciar un esquema de esas conexiones, para este trabajo usó la conexión Usuario a Red.

Figura 15

Tipos de conexiones para transmitir aplicaciones interactivas



Nota. La figura muestra los dos tipos de conexiones en las que se puede transmitir aplicaciones interactivas. Adaptada de Codificación genérica de imágenes en movimiento y audio asociado, 1998.

Los paquetes Usuario a Red, son transmitidos de forma que el receptor pueda controlarlos y extraerlos en cualquier momento, para esto se hace uso del protocolo de descarga, el cual se divide en 3 procesos importantes: las Secciones DSM-CC, que se encargan del encapsulamiento de los datos dentro del TS, el Carrusel del Datos que se encarga de verificar los datos que vienen incluidos en el TS, y el Carrusel de Objetos el encargado de la construcción de los directorios y los archivos (ISO/IEC, 1998).

Paquetes DSM-CC dentro del Transport Stream

Dentro de un TS, pueden venir incluidas varias programaciones, así como información de audio, video y datos, en el caso de los datos para las aplicaciones interactivas, se analizan dos tablas, la PAT y la PMT que contienen información de los servicios correspondientes a las programaciones del TS. En la figura 16 se muestra la estructura del descriptor para un DSM-CC.

Figura 16
Descriptor para el flujo DSM-CC

Campos del descriptor de flujo elemental en PMT	Stream Type Identifier	Reserved	Elementary Stream PID	Reserved	ES information length	Descriptor Information
Valores que describen al flujo de mensajes U-N DSM-CC en un TS	8 bits	3 bits	13 bits	4 bits	12 bits	8xN bits
	0x0B	'111'	PID de U-N DSM-CC	'1111'	N	-

Nota. La figura muestra los datos que debe contener el descriptor para identificar que se trata de un DSM-CC. Tomada de Implementación de un software para construir un extractor y constructor de carrusel de datos, de Descriptor de Flujo en PMT para el DSM-CC, 2016.

Secciones DSM-CC

Las secciones DSM-CC, son la herramienta que se usa para colocar los paquetes del TS en el carrusel de datos, el tamaño máximo de cada sección es de 4096

bytes, en la última sección por lo general se tiene una longitud menor a 4093 bytes, para encontrar cada una de las secciones se comprueba que, en la PMT existe un *stream type* U-N DSM-CC que tiene el valor de 0x0B, si fuera el caso de no encontrar este valor quiere decir que la transmisión no contiene aplicaciones interactivas (ISO/IEC 13818-1, 2021).

El valor que se tiene en el campo *table_id*, ayuda a determinar el tipo de bloque que se recibe, por ejemplo un valor de 0x3B, corresponden a los bloques DII (*Download Info Indication*) y DSI (*Download Server Initiate*), si se tiene un valor de 0x3C corresponde a un bloque DDB (*Download Data Message*), en la tabla 8 se muestran los valores que deben tener algunos campos del descriptor DSM-CC cuando se presentan los bloques antes mencionados (Nuñez, 2016).

Tabla 8
Codificación de las secciones DSM-CC

Mensaje	DBB	DII	DSI
table_id	0x3C	0x3B	0x3B
table_id_extension	Module Id	0x0002-0xFFFF	0x0000-0x0001
version_number	Module version	0x00	0x00
section_number	Block number (0x00 – 0xFF)	0x00	0x00
last_section_number	Max (section number)	0x00	0x00

Nota. La tabla muestra cómo es posible la codificación de las secciones DSM-CC.

Adaptada de Implementación de un software para construir un extractor y constructor de carrusel de datos, de Codificación de las secciones DSM-CC, 2016.

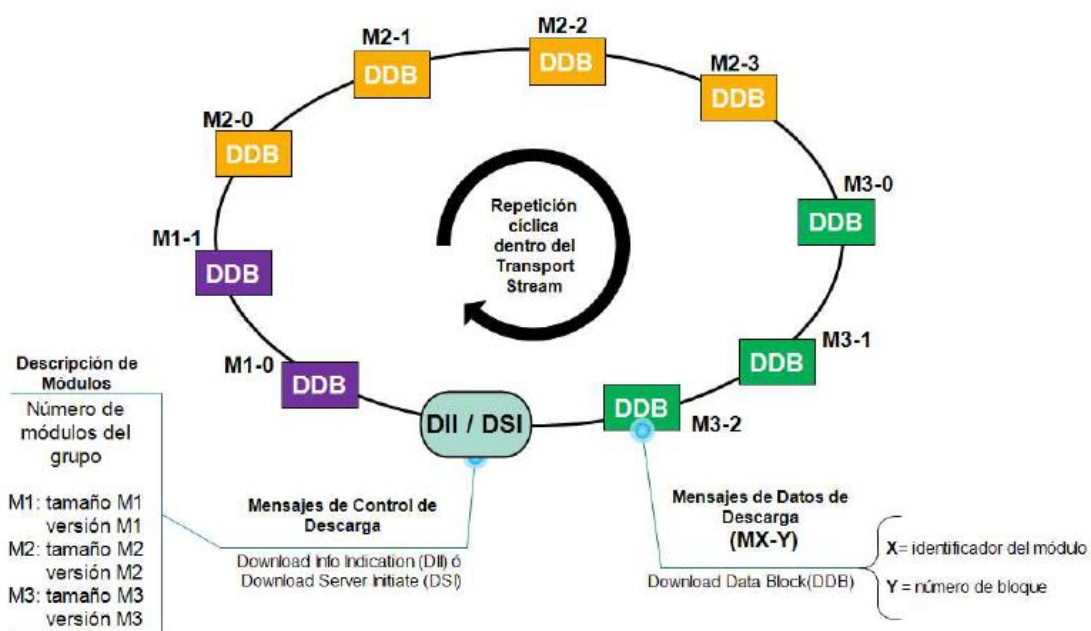
Carrusel de Datos

El carrusel de datos es usado con el fin de proveer un formato a los datos antes de transmitirlos por la red, una vez que la información ha sido convertida en un carrusel de datos, se la divide en módulos o secciones. Como se indicó anteriormente, existen 3 bloques de descarga, para identificarlos se usa el valor del campo *table_id*. Los bloques DDB son los encargados de encapsular el carrusel de datos, mientras que los bloques DDI y DSI son bloques de control que facilitan la correcta descarga de los módulos.

Transmisión del Carrusel de Datos

Figura 17

Transmisión Cíclica del Carrusel de Datos



Nota. La figura muestra la transmisión cíclica del Carrusel de Datos. Adaptada de Digital Video Broadcasting, de Transmisión cíclica de la información en el Carrusel de Datos, 2003.

La transmisión del carrusel de datos se lo hace de forma cíclica, es decir que los bloques pueden ser enviados varias veces en el TS, esto se lo hace para que el

receptor pueda descargar los datos en cualquier momento. En la figura 17 se muestra un ejemplo de la transmisión cíclica de los módulos (ETSI, 2003).

Carrusel de Objetos

Los archivos que componen las aplicaciones interactivas se transmiten a través del TS, además de recibir su contenido es importante recibir el nombre de cada archivo y su directorio, ya que se descarga el código fuente para reproducir los elementos interactivos, y en el mismo se hace el llamado a imágenes, textos y otros elementos que componen la interactividad, para esta tarea se usa el Carrusel de objetos que usa un sistema de mensajes BIOP (*Broadcast Interoperability Protocol*), que aporta para la correcta construcción del directorio (ETSI, 2003).

Tabla 9
Tipos de objeto

Tipo de Obejto	ID de Tipo	Alias
Directorio	Directory	'dir'
Archivo	File	'fil'
Puerta de Servicio	Service Gateway	'srg'
Flujo	Stream	'str'
Evento de Flujo	Stream Event	'ste'

Nota. La tabla muestra los tipos de objeto que pueden venir incluidos. Tomada de Implementación de un software para construir un extractor y constructor de carrusel de datos, de Tipo de objeto, 2016.

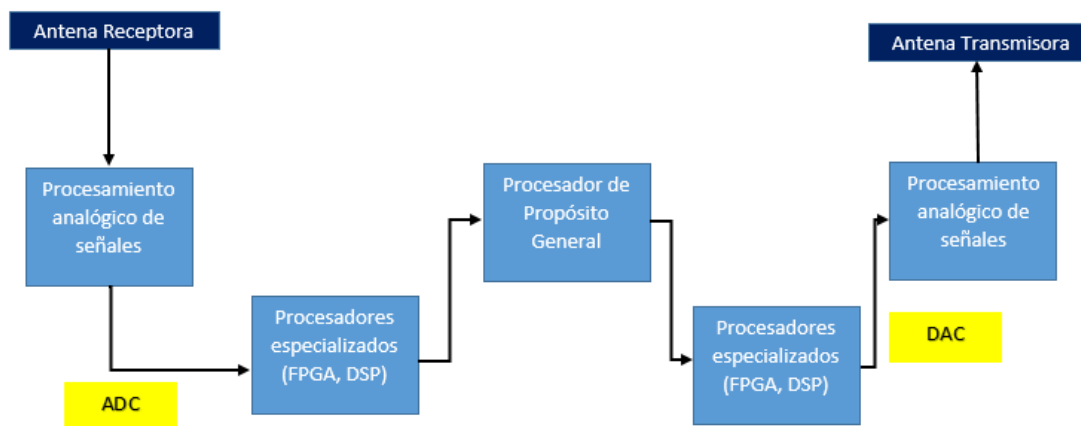
Los mensajes BIOP representan a un solo objeto, existen 5 tipos de objetos que tienen sus diferentes funciones. En la tabla 9 se muestran los tipos de objetos con los que se puede trabajar, así como de su identificativo.

Radio Definida por Software

Con el uso de la Radio Definida por Software se tiene una arquitectura flexible, al controlar el procesamiento de señales por software en lugar de hardware, permite tener una serie de herramientas para modulación, filtrado, operaciones de banda ancha o de banda estrecha y técnicas de espectro extendido, además el bajo costo que tienen estos dispositivos permiten implementarlos con relativa facilidad. La arquitectura de estos dispositivos se puede apreciar en la figura 18 (Cruz & Rodriguez, 2017).

Figura 18

Arquitectura genérica de la Radio Definida por Software



Nota. La figura muestra la arquitectura de un dispositivo de Radio Definida por Software. Adaptada de Desarrollo de aplicaciones para la radio definida por software, de Arquitectura en bloques de la Radio Definida por Software, 2017.

El bloque de procesamiento de señales en el caso de la recepción es el que se encarga de recibir las señales de radiofrecuencia para adecuarlas y convertirlas en frecuencias intermedias, en el caso de la transmisión se encarga de amplificar y modular las señales de FI adecuándolas para la transmisión en el aire. Los bloques de procesadores especializados suelen estar compuestos por FPGAs, que son un circuito integrado por varios bloques denominados lógicos configurables, los mismos que

pueden ser conectados con el uso del lenguaje de programación VHDL o Verilog (Cruz & Rodriguez, 2017).

Equipo SDR Adalm Pluto

Figura 19

Dispositivo Adalm Pluto



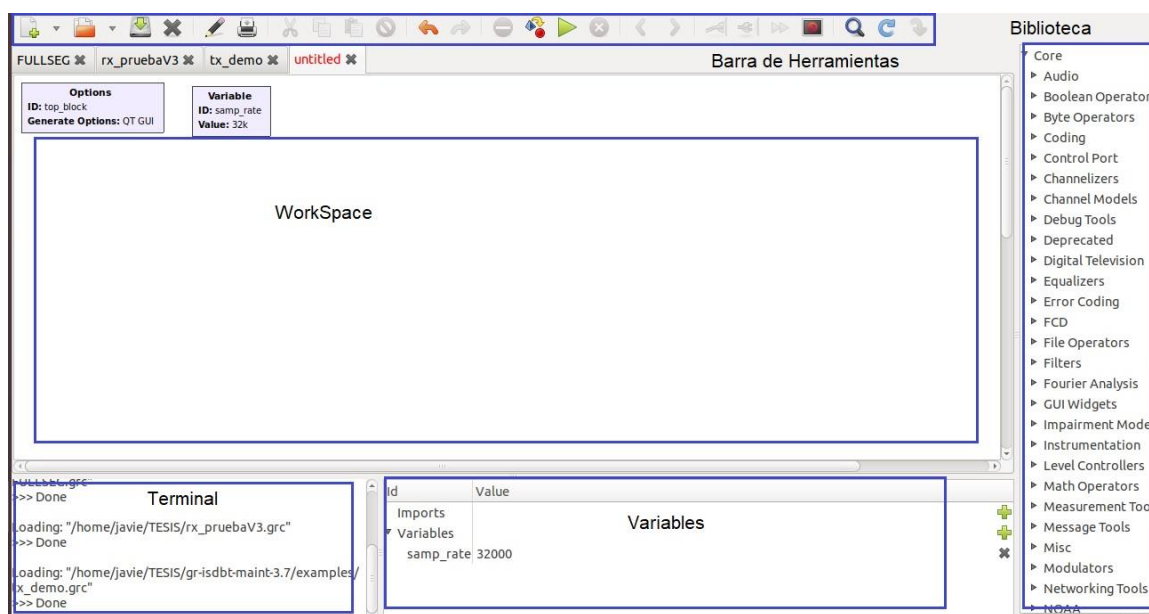
Nota. La figura muestra el SDR Adalm Pluto.

El equipo PlutoSDR, es un SDR de bajo costo que permite realizar varias actividades tanto educativas como experimentales, puede funcionar como transmisor y receptor debido a que posee dos antenas GSM, suelen ser usados para: aplicaciones de Radio Frecuencia, tecnologías de Internet inalámbrica, telefonía celular, radio cognitivo, analizador de espectro, el limitante que posee dicho equipo es que puede trabajar con frecuencias de muestreo de hasta 5 MHz, para tomar o enviar los datos de forma continua si se supera este valor los datos son enviados por ráfagas, en la figura 19 se puede apreciar el dispositivo (Castillo, 2021; Gets, 2021).

GNU Radio

El dispositivo PlutoSDR es compatible para desarrollar el software con Matlab, para programación por código, mientras que simulink permite realizarlo a través de bloques, pero GNU Radio permite la programación por módulos y a su vez facilita la creación de nuevos bloques a través de código, este software libre tiene muchas prestaciones, una de las más importantes es que se lo puede usar con dispositivos externos como el SDR que permite modelar sistemas con mayor facilidad, también se tiene la opción de realizar simulaciones (GNU Radio, 2021a).

Figura 20
Pantalla principal de GNU Radio



Nota. La figura muestra la pantalla principal de GNU Radio.

La interfaz de GNU Radio es amigable con el usuario en la figura 20 se muestra cómo está compuesta, se indica donde se ubican los componentes como: la barra de herramienta, el espacio de trabajo en el cual se colocan los bloques necesarios, la biblioteca en la cual se encuentran los módulos con sus respectivos bloques, la ventana

de variables indica todas las variables que se han declarado para un determinado programa, y el terminal, en donde se muestran los procesos al ejecutar un programa.

CAPÍTULO III

Diseño e implementación del sistema de recepción EWBS y aplicaciones interactivas Full-seg

Introducción

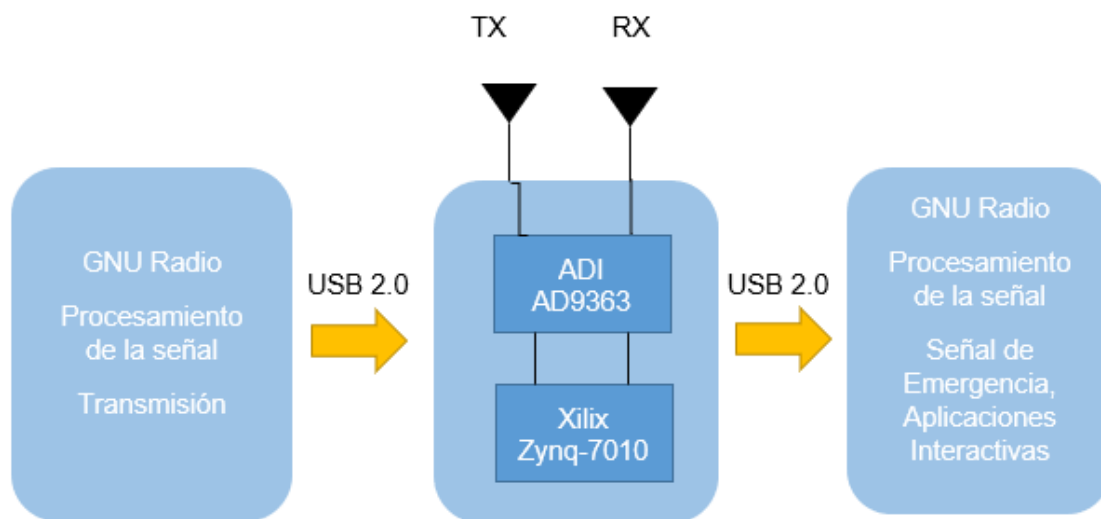
En este capítulo se indica cómo fue implementado el receptor Full-seg, el mismo que recibe señales de mensajes de emergencia EWBS y aplicaciones interactivas según el estándar ISDB-Tb, se detallan los bloques usados en el software GNU Radio, tanto los que fueron importados de proyectos anteriores, como los creados con el fin de extraer el mensaje de emergencia, fue necesario analizar el bit 26 de la información de TMCC, además de analizar la tabla PAT para encontrar el PID de la tabla PMT y de esta forma extraer sus datos. Para extraer los archivos de las aplicaciones interactivas también fue necesario analizar la tabla PMT y verificar si está contiene el descriptor DSM-CC que indica que el TS contiene aplicaciones interactivas.

Descripción del sistema

Uno de los inconvenientes presentados durante la elaboración del presente trabajo, fue que el SDR Adalm-Pluto recibía la información de forma continua con frecuencias de muestreo de hasta 5 MHz, y en el estándar se indica que para un receptor full-seg la tasa de muestreo debe ser de 8.126984 MHz, este problema ocasiona pérdida de información y no se logra generar el TS de forma correcta, aunque por especificaciones del fabricante del SDR se menciona que a partir de los 5MHz hasta los 70MHz el dispositivo puede transmitir o recibir información mediante flujos, al aumentar la frecuencia de muestreo el Adalm-Pluto presentaba un comportamiento errático ocasionando fallos en el funcionamiento del receptor.

Por la razón antes mencionada se trabajó en un transmisor compatible con la frecuencia de muestreo a la que se podía trabajar con el SDR de forma continua, el sistema con el que se realizaron las pruebas quedó tal como se muestra en la figura 21, en la parte del transmisor se tiene un SDR conectado a un computador con el software GNU Radio lo que hace posible la transmisión del TS, luego se tiene otro Adalm-Pluto que a su vez se conecta a otro host que contiene el mismo software y permite recuperar el TS el cuál fue analizado y se pudo extraer la información tanto del mensaje de emergencia como de los archivos para las aplicaciones interactivas, para probar el sistema también se puede usar el mismo computador y el mismo SDR como transmisor y receptor.

Figura 21
Diagrama General del Sistema



Nota. La figura muestra el diagrama general del Sistema.

Descripción de los elementos del sistema

Los elementos que conforman el sistema son de suma importancia, tanto en el software como en el hardware, por esta razón se hará una breve explicación de cada

uno de los elementos que intervienen en la implementación del receptor, en la figura 22, se muestra la conexión de cada uno de los elementos.

Figura 22

Conexión de los elementos del Sistema



Nota. La figura muestra los elementos conectados.

Hardware

Computador (transmisor y receptor): en este caso se usó un computador portátil HP con procesador Intel Core i7 y 1.99 GHz de procesamiento, memoria RAM de 8 GB, sistema operativo de 64 bits, 2 puertos USB 2.0 y 1 puerto USB 3.0 para establecer comunicación con el Pluto SDR.

SDR Adalm Pluto: Dispositivo de Radio Definida por Software, el cual trabaja en las frecuencias UHF (300 MHz a 3000 GHz), contiene un chip ADI AD9363 que es compatible con el estándar ISDB-T, una FPGA Xilinx Zynq-7010 (Core ARM Cortex A9) y una interfaz USB 2.0.

Software

Sistema operativo Linux: El sistema operativo escogido es Linux 18.04, para facilitar el desempeño del software GNU Radio, además de las facilidades que brinda al momento de trabajar en el lenguaje de programación python para la creación de los bloques necesarios para complementar las funcionalidades del receptor.

Drivers: Los drivers necesarios para el uso de GNU Radio son de suma importancia para que todos los elementos que intervienen sean compatibles ente si, además de permitir la programación en python y que el host sea capaz de reconocer el SDR.

Descarga de Módulos desde Git-Hub

Para la elaboración del receptor fue necesario tomar como base bloques creados anteriormente para ser utilizados en el software GNU Radio, los mismos se encuentran de forma abierta en la página web Git-Hub, estos proyectos toman como base los bloques desarrollados por Federico La Rocca en la Universidad de Uruguay junto con el grupo de investigación Git-Artes y en cada proyecto se muestra un complemento con respecto al original.

Para descargar los bloques primero fue necesario extraer los archivos directamente de Git-hub, este proceso se lo realizó a través del terminal de Linux, se recomienda crear una carpeta en la cual se guardarán todos estos archivos, una vez se tenga esto se puede ejecutar el siguiente comando, el cual permite descargar los archivos necesarios.

- `git clone https://github.com/xxxx.git`

Una vez que fueron descargados los archivos necesarios, se accede a la carpeta en la que se guardaron y se crea una nueva carpeta que servirá para compilar los ficheros e instalarlos correctamente, como se muestra en los siguientes pasos.

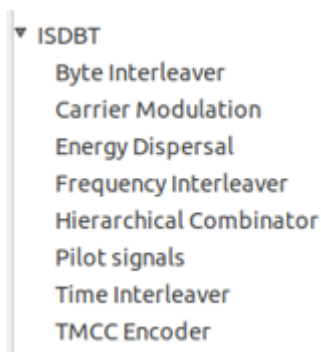
- `cd xxxx`
- `mkdir build`
- `cd build`

- `cmake ../`
- `make`
- `sudo make install`
- `sudo ldconfig`

Cuando estos comandos se han ejecutado sin errores los bloques estarán listos para ser utilizados dentro de GNU Radio, un ejemplo tenemos al instalar el módulo ISDBT, en la figura 23 se muestra los bloques contenidos en el módulo, este mismo proceso se los debe realizar con todos los módulos que se necesiten.

Figura 23

Bloques del módulo ISDBT (MEJOR LISTAR)



Nota. La figura muestra los bloques correspondientes al módulo ISDBT descargado de Git-Hub.

Creación de nuevos bloques

Para completar las funciones del receptor se crearon bloques que pueden ser usados dentro de GNU Radio, estos bloques personalizados sirvieron para el análisis del TS en tiempo real, gracias a esto se pudo extraer el mensaje de emergencia, así como de poder determinar si el flujo de datos contiene o no aplicaciones interactivas.

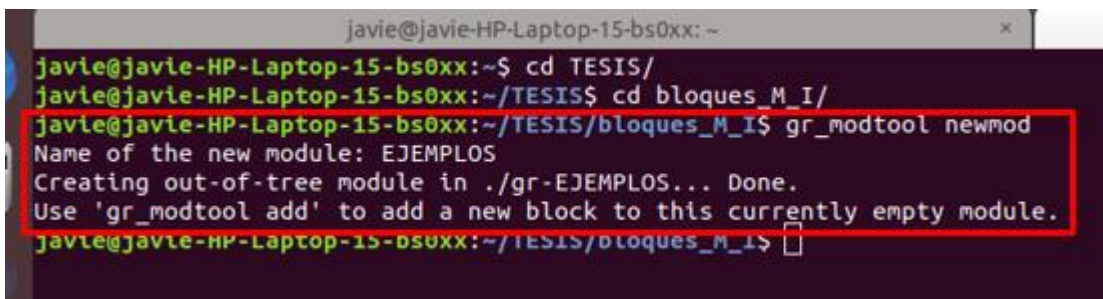
Para generar nuevos bloques se recomienda tenerlos en una carpeta específica para facilitar el manejo de los archivos generados, para crearlos se utilizó la herramienta *gr_modtool*, la cual permite la inclusión de nuevos módulos compatibles con GNU Radio, primero se debe crear el módulo en el cual se almacenará los bloques nuevos, para esto es necesario ejecutar el siguiente comando.

- `gr_modtool newmod`

Una vez se ejecuta el comando se requiere agregar el nombre que llevará el módulo creado, y se procede a verificar que todos los ficheros necesarios fueron creados dentro de la carpeta del módulo, tal como se muestra en la figura 24.

Figura 24

Creación de nuevos módulos



```

javie@javie-HP-Laptop-15-bs0xx: ~
javie@javie-HP-Laptop-15-bs0xx:~$ cd TESIS/
javie@javie-HP-Laptop-15-bs0xx:~/TESIS$ cd bloques_M_I/
javie@javie-HP-Laptop-15-bs0xx:~/TESIS/bloques_M_I$ gr_modtool newmod
Name of the new module: EJEMPLOS
Creating out-of-tree module in ./gr-EJEMPLOS... Done.
Use 'gr_modtool add' to add a new block to this currently empty module.
javie@javie-HP-Laptop-15-bs0xx:~/TESIS/bloques_M_I$

```

Nota. La figura muestra la creación del módulo EJEMPLOS.

Luego de tener creado el módulo nuevo se crearon los bloques de procesamiento, para esto se debe acceder a la carpeta en la cual se almacenan los datos del módulo y ejecutar el siguiente comando, en él se puede especificar el tipo de bloque, así como del lenguaje de programación que se usará para su programación.

- `gr_modtool add -t tipo de bloque -l lenguaje de programación`

Una vez ejecutado este comando se llena la información solicitada como se muestra en la figura 25, los datos requeridos son: el nombre del bloque, el autor de los

bloques, los argumentos que tendrá el bloque y si se desea o no crear un archivo para pruebas.

Figura 25

Creación de nuevos bloques

```

javier@javier-HP-Laptop-15-bs0xx: ~
javier@javier-HP-Laptop-15-bs0xx: ~/TESIS/bloques_M_I/gr-EJEMPLOS$ gr_modtool add -t sync -l python
GNU Radio module name identified: EJEMPLOS
Enter name of block/code (without module name prefix): Ejemplo1
Block/code identifier: Ejemplo1
Language: Python
Please specify the copyright holder: Bryan Chanataxi
Enter valid argument list, including default arguments:
Add Python QA code? [Y/n] n
Adding file 'python/Ejemplo1.py'...
Adding file 'grc/EJEMPLOS_Ejemplo1.xml'...
Editing grc/CMakeLists.txt...
javier@javier-HP-Laptop-15-bs0xx: ~/TESIS/bloques_M_I/gr-EJEMPLOS$

```

Nota. La figura muestra la creación del bloque Ejemplo 1, en lenguaje de programación Python.

Figura 26

Declaración de entradas, salidas y funcionalidad de un bloque

```

import numpy
from gnuradio import gr

class Ejemplo1(gr.sync_block):
    """
    docstring for block Ejemplo1
    """
    def __init__(self):
        gr.sync_block.__init__(self,
            name="Ejemplo1",
            in_sig=[<+numpy.float32+>],
            out_sig=[<+numpy.float32+>])

    def work(self, input_items, output_items):
        in0 = input_items[0]
        out = output_items[0]
        # <+signal processing here+>
        out[:] = in0
        return len(output_items[0])

<!-- Make one 'sink' node per input. Sub-nodes:
* name (an identifier for the GUI)
* type
* vlen
* optional (set to 1 for optional inputs) -->
<sink>
<name>in</name>
<type><!-- e.g. int, float, complex, byte, short, xxx_vector, ...--></type>
</sink>

<!-- Make one 'source' node per output. Sub-nodes:
* name (an identifier for the GUI)
* type
* vlen
* optional (set to 1 for optional inputs) -->
<source>
<name>out</name>
<type><!-- e.g. int, float, complex, byte, short, xxx_vector, ...--></type>
</source>
</block>

```

Nota. En el lado izquierdo de la imagen se muestra la funcionalidad del bloque, en el lado izquierdo los tipos de entrada.

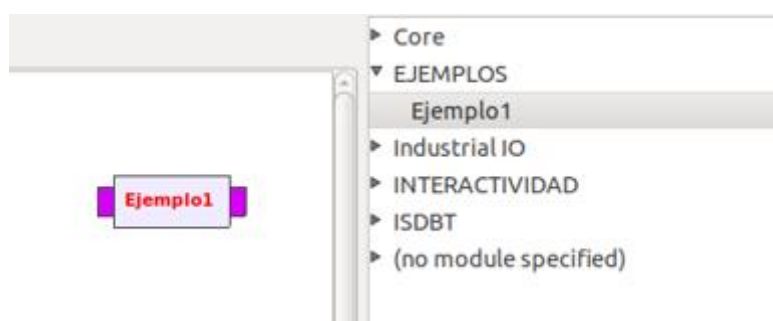
Quando este proceso haya finalizado con éxito es posible acceder a la carpeta que contiene el código en python, para especificar el tipo de datos que se tendrá a la entrada y a la salida, además de cuantas entradas y salidas tendrá el bloque, también se empieza a programar la funcionalidad. Una vez se tenga esto se debe acceder a la

carpeta `grc` para editar el archivo `xml` donde también se indica el tipo de datos que se tendrá a la entrada y a la salida, un ejemplo de lo antes mencionado se lo tiene en la figura 26.

Luego de tener los archivos mencionados con las configuraciones adecuadas se procede a crear la carpeta `build` y ejecutar los comandos que se mencionaron en la sección anterior, como resultado se tiene que en GNU Radio es posible observar el nombre del módulo que fue creado así como de los bloques que se añadan al mismo, esto se puede observar en la figura 27, es importante mencionar que si una vez creado el bloque se realiza algún cambio en su funcionalidad se deben volver a ejecutar los comandos desde la carpeta `build`, para que de esta forma el bloque se actualice.

Figura 27

Módulo y bloque creados en GNU Radio.



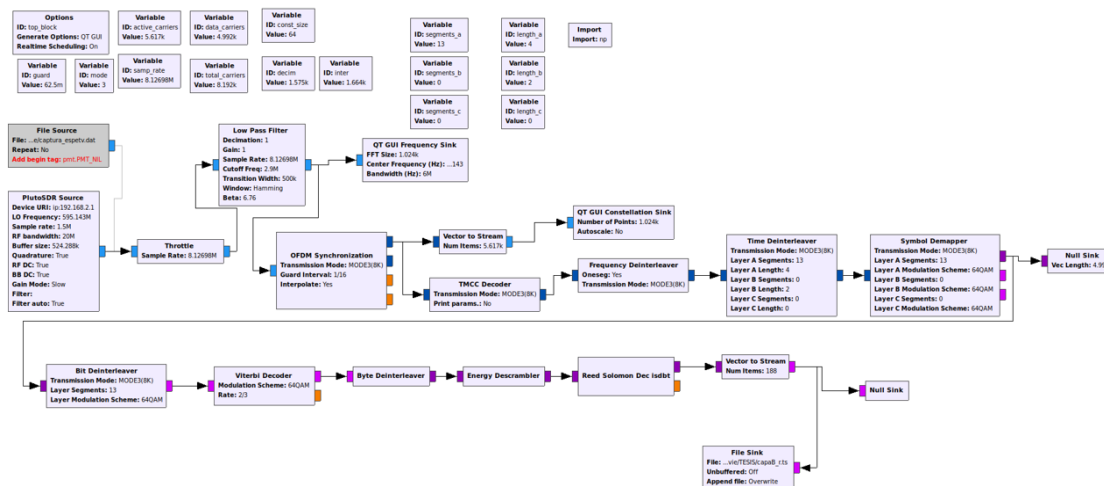
Nota. En la figura se muestra el bloque creado.

Implementación del receptor Full-seg

Una vez se instalaron los bloques necesarios se procede a la implementación del receptor, para esto se usaron los bloques disponibles en el módulo ISDBT que fue descargado desde Git-Hub y pertenece al grupo Git-Artes, otro modulo que se usó de base es el de Jordy Garzón quien dentro de los ejemplos que ha realizado tiene un receptor Full-seg, pero se tuvieron que realizar algunos cambios que serán detallados más adelante.

Como se mencionó en la sección anterior, al presentarse problemas con la frecuencia de muestreo a la que funciona el SDR, se trabajó en un transmisor compatible con el receptor desarrollado, para evitar perder datos en la transmisión. Se tomó la decisión de enviar el flujo de datos por una sola capa, en este caso la capa A, entonces para el receptor se tiene el diagrama de bloques que se muestra en la figura 28, se puede observar que se leen los datos solo de la capa antes mencionada y de esta forma se genera un Transport Stream, es importante mencionar que hasta este punto aún no se integran el extractor del mensaje de emergencia ni el de los archivos para aplicaciones interactivas.

Figura 28
Diagrama de bloques del receptor Full-seg



Nota. La figura muestra el receptor full-seg implementado en GNU-Radio.

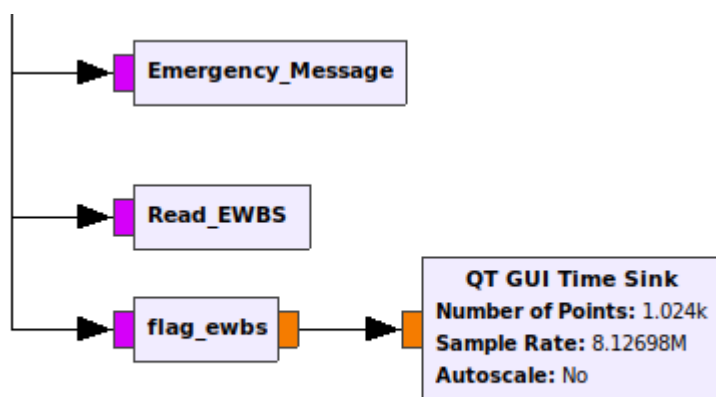
Algoritmo para extraer el mensaje de emergencia

Para elaborar el extractor del mensaje de emergencia EWBS, se tomó como base a los bloques descritos en (Castillo, 2021), a los cuales se aplicaron ciertos cambios para optimizar cada bloque, el primero fue la elaboración de un bloque anterior el cual servirá para extraer la información de la PMT y mostrarlo, se decidió hacerlo para optimizar las funcionalidades además que servirá para extraer la información del

descriptor DSM-CC, si se sobrecarga de actividades se puede llegar a saturar las funcionalidades del bloque y presente fallas en su desempeño, en la figura 29 se puede observar cómo queda el diagrama de bloques luego de realizar este cambio.

Figura 29

Bloques necesarios para la extracción del mensaje EWBS



Nota. La figura muestra el bloque del análisis de la PMT previo al análisis para obtener el mensaje EWBS.

A continuación se procede a explicar el funcionamiento de cada uno de estos bloques y como fueron realizados, haciendo especial énfasis en los cambios que se realizaron con respecto al desarrollado en (Castillo, 2021).

Bloque extractor PMT

Este bloque es el encargado de analizar la información que se recibe en tiempo real, el flujo de datos que se tiene a la entrada es de 188 bytes, este debe ser analizado buscando el identificador de la tabla PMT y determinar si contiene los descriptores tanto del mensaje de emergencia como el de aplicaciones interactivas, los pasos que se siguen para cumplir con este objetivo son los siguientes:

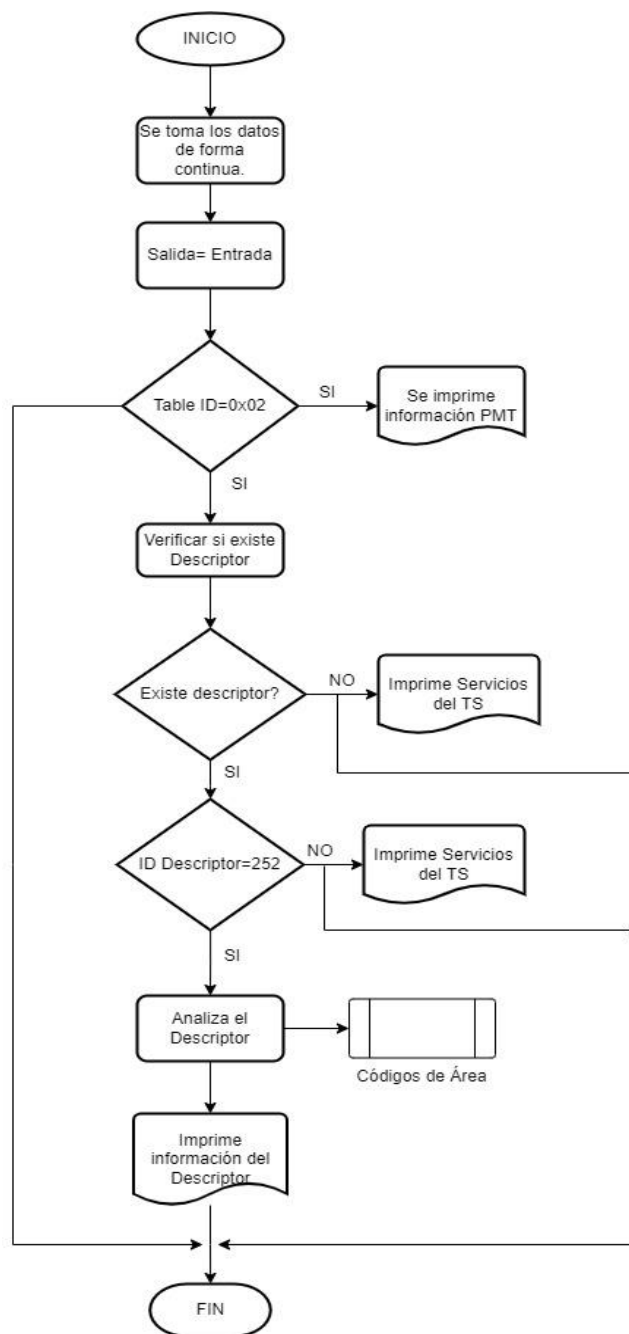
1. Recibir y enviar de forma continua el flujo de datos que ha sido limitado a 188 bytes, esto se debe hacer para que mientras el programa analiza el Transport Stream la transmisión no se vea afectada.
2. Dentro de cada paquete comprobar si se trata de una PMT, esto se lo puede hacer identificando el PID correspondiente o se puede trabajar con el identificador de la tabla para este caso es el 0x02 o 2 en decimal.
3. Mostrar la información que se puede extraer de la tabla PMT, de acuerdo a lo indicado en el capítulo II.
4. Validar el valor que se obtiene con los bits del *program length information*, si es igual a 0 se continúa en un bucle para extraer la información de *Stream type identifier*, *Elementary stream PID* y *ES information length*.

Bloque Read EWBS

Para el desarrollo de este bloque se usó como base el propuesto en (Castillo, 2021), se implementaron cambios para evitar sobrecargar de tareas al bloque, ya no se imprime la información que se obtiene de la tabla PMT, únicamente se trabajará con el descriptor EWBS, se analizará la tabla tal como se muestra en el capítulo II, con los cambios antes mencionado el algoritmo queda de la siguiente forma:

1. Recibir de forma continua el flujo de datos que ya ha sido limitado a 188 bytes.
2. Dentro de cada paquete determinar si es o no una PMT
3. Comprobar qué valor tiene el *program information length*, si es diferente de cero validar el identificador del descriptor en este caso debe ser 0xFC o 255 en decimal.

Figura 30
 Diagrama de Flujo para el bloque Read EWBS



Nota. La figura muestra el diagrama de flujo del bloque Read EWBS. Adaptada de Estructura del diagrama de flujo para la programación en Python del bloque Read EWBS, de Implementación de un sistema de recepción de una señal de emergencia EWBS, 2021.

4. Dentro del descriptor EWBS se extrae la información necesaria según la figura 11 del capítulo II, se separan los bits que contienen el código de área.
5. Luego de tener almacenados los códigos de área se llama a la función *dataAreaCode* para que con estos valores se genere un archivo que contiene las provincias y los cantones en los cuales se activó la alerta.
6. Al terminar de procesar los bytes del descriptor se termina de procesar los bytes antes del CRC, extrayendo la información necesaria. Se verifica que el *ES Information Length* sea igual a 0x6 o 6 en decimal para saber que representa el identificador de tipo de servicio de un “*Paquete PES*” que lleva el mensaje de sobreimposición, dentro del descriptor se analiza la etiqueta del componente que para este caso debe ser 0x38 por tratarse de una recepción full-seg.
7. Como salida se tiene el mismo flujo de datos que se envía a un bloque de file sink que genera el TS que puede ser reproducido con ffmpeg.

En la figura 30 se puede observar el diagrama de flujo utilizado para el análisis del descriptor EWBS, como se especifica en el capítulo II, con este análisis se busca obtener el ID de los paquetes que contiene la información para el texto de sobreimposición.

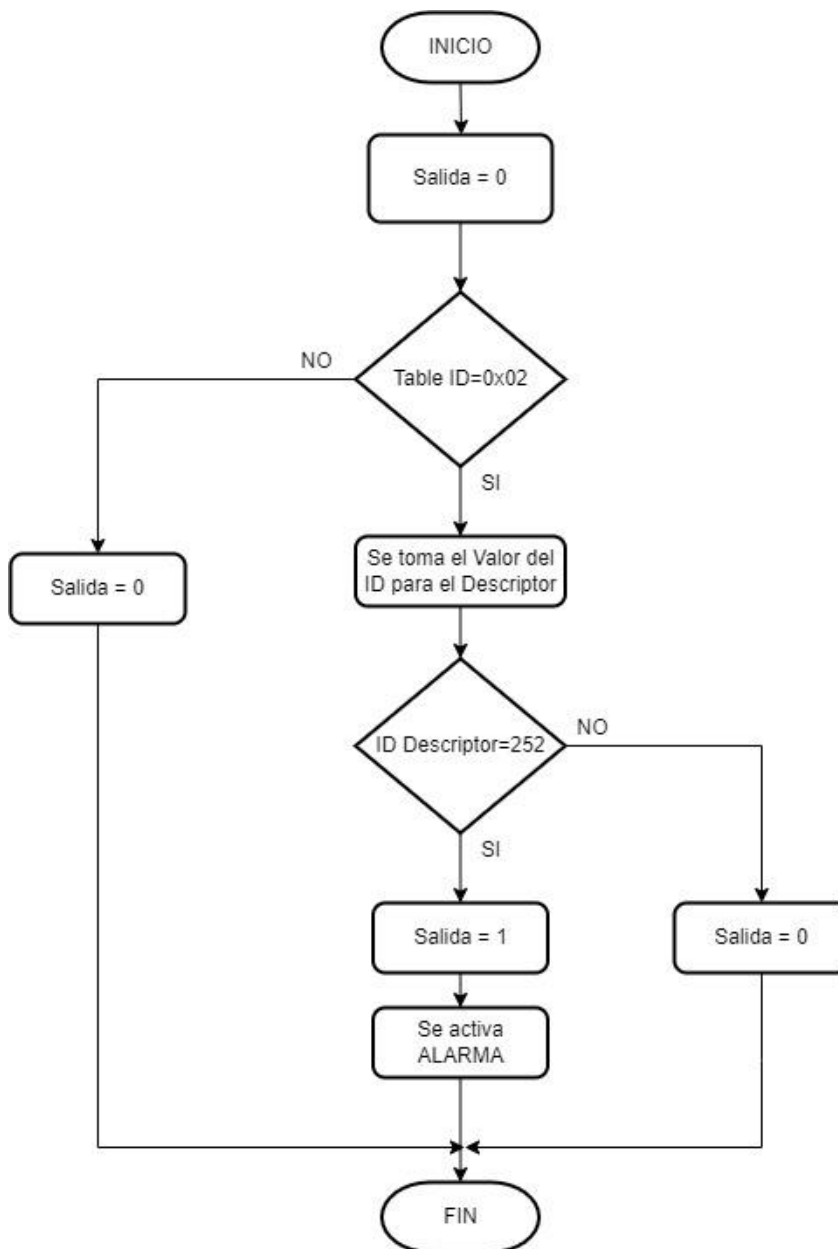
Bloque Flag EWBS

Para este bloque se siguió el mismo algoritmo que implementó (Castillo, 2021), el único cambio fue adicionar la reproducción de una sirena al momento en el cual se activa la bandera que indica la aparición de un mensaje de emergencia, el resto de funcionalidades del bloque se siguió tal como se indicaba en el trabajo antes mencionado, este bloque ayuda a recibir y percatarse de forma eficiente de la existencia

del mensaje EWBS, en la figura 31 se observa el cambio que se ha realizado en el diagrama de flujo propuesto por (Castillo, 2021).

Figura 31

Diagrama del flujo para el bloque Flag EWBS



Nota. La figura muestra el diagrama de flujo del bloque flag EWBS al que se le ha añadido una alarma.

Bloque Emergency Message

Este bloque es el encargado de extraer el mensaje de superposición que viene incluido en el TS, para esto es necesario conocer que este mensaje así como la información del lenguaje son añadidos en los paquetes vacíos luego de la primera PMT con el descriptor EWBS, el identificador de los paquetes que contienen esta información se determina con el análisis de la PMT, en ella se indica un valor de PID de los paquetes que contienen la información del lenguaje y el mensaje de superposición, en este caso se tiene valores de 534 para One-seg y 278 para Full-seg (Recalde, 2016).

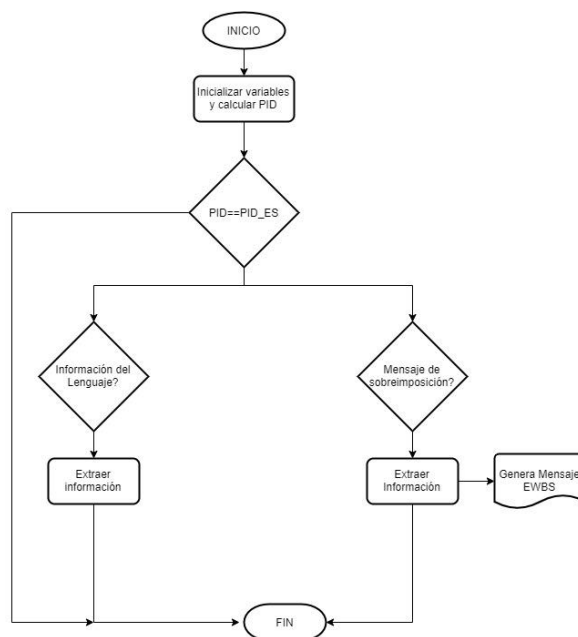
El proceso que se siguió para la extracción es el mismo que utilizó (Castillo, 2021), explicado de forma general, los pasos son los siguientes:

1. Tomar la información del TS generado en tiempo real, se toma 188 bytes en cada flujo y se los analiza.
2. Obtener el PID de cada paquete recibido para determinar si se trata o no del paquete que se busca.
3. Como se mencionó el PID es el mismo para el paquete que contiene información del lenguaje y el que contiene el mensaje de superposición por esto es importante identificar de cuál de estos dos se trata.
4. Para el mensaje de superposición se genera un archivo txt que contiene el tipo de emergencia que ocurre conjuntamente con la hora en la que se generó.

El diagrama de flujo para este bloque se lo muestra en la figura 32, el mismo que fue adaptado del propuesto por (Castillo, 2021).

Figura 32

Diagrama de flujo para el bloque Emergency Message



Nota. La figura muestra el diagrama de flujo para el bloque encargado de extraer el mensaje EWBS. Adaptada de Estructura del diagrama de flujo para la programación en Python del bloque Emergency Message, de Implementación de un sistema de recepción de una señal de emergencia EWBS, 2021

Algoritmo para extraer los archivos de las aplicaciones interactivas

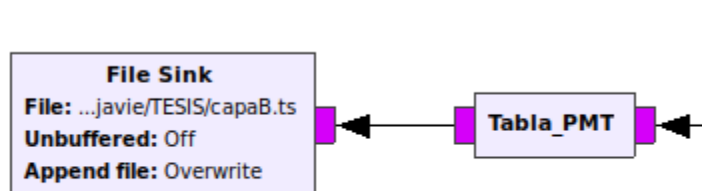
Se creó un bloque que permite analizar el TS en tiempo real para extraer la información de las cabeceras DSM-CC, para esto se hace uso de la salida del bloque que analiza la PMT como se indicó anteriormente, se analiza estos datos y se imprimen, esto se lo hace obteniendo el PID de las secciones DSM-CC y así analizando cada flujo de datos que contenga este identificador, en la figura 33, se puede ver el diagrama de bloques para este proceso.

Para extraer los archivos de las aplicaciones interactivas se tomó como base el trabajo realizado por (Nuñez, 2016), en el cual se detalla todo el proceso para extraer

los archivos con éxito. Los datos que forman parte de las aplicaciones interactivas vienen encapsulados en el TS, la forma en la que se reciben puede ser comparada con un modelo por capas similar al modelo OSI o el TCP/IP, ya que cada capa tiene cabecera y segmento de datos y cada una depende de la anterior, en la figura 34 se puede apreciar los niveles del proceso a seguir para extraer los datos.

Figura 33

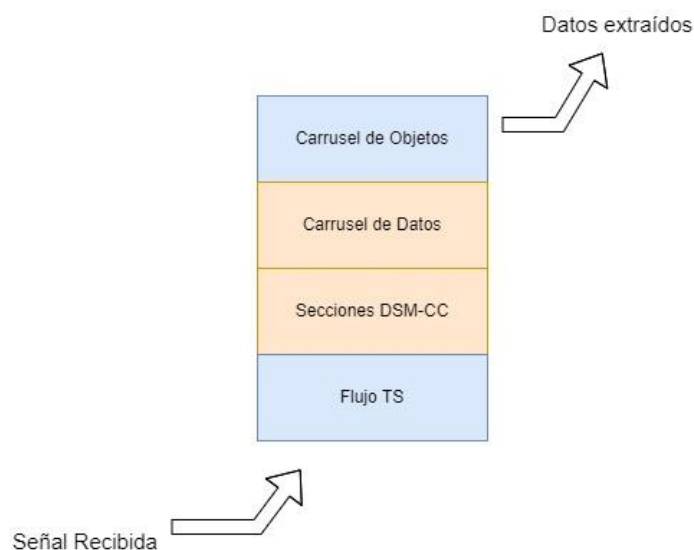
Diagrama de bloques para las secciones DSM-CC



Nota. La figura muestra el bloque encargado de analizar las tablas PMT, para el receptor de interactividad.

Figura 34

Diagrama del proceso para la extracción de los archivos



Nota. La figura muestra el proceso que se debe seguir para la extracción de los archivos que complementan la aplicación interactiva. Adaptada de Modelo por capas de

encapsulamiento de datos en el TS, de Estudio, análisis e implementación de un software para construir un extractor y constructor de Carrusel de Datos, 2016.

Para la extracción de las aplicaciones interactivas, se programó en python, la estructura principal del programa se muestra en la figura 35, donde se puede observar que a partir del TS generado, se analiza el flujo de datos con el fin de extraer la información, fueron necesarias varias funciones las cuales serán detalladas más adelante, el algoritmo del programa principal es el siguiente:

Figura 35
Estructura del algoritmo extractor de archivos



Nota. La figura muestra la estructura del programa principal para extraer los archivos de las aplicaciones interactivas.

1. Determinar el PID de las secciones DSM-CC, esto se logra a través del análisis de la tabla PMT cuyo identificador se conseguirá con el análisis de la PAT.
2. Una vez se tiene este dato se llama a la función que se encarga de extraer las secciones DSM-CC, este proceso es de suma importancia ya que, aquí es donde se eliminan los bytes de relleno que puede tener cada sección y con el archivo que se genera se continua con la extracción.

3. A la salida de esta función se puede determinar si el flujo de datos tiene o no aplicaciones interactivas, esto se lo puede verificar con el número de bloques que se detecten, si este valor es 0 indica que no posee aplicaciones interactivas.
4. Luego de comprobar si posee o no aplicaciones interactivas, se hace uso de la función que se encargará de generar el carrusel de datos, lo que implica eliminar las cabeceras de las secciones DSM-CC que se han logrado extraer en la función anterior.
5. Finalmente, los datos ingresan a la función que se encarga de des encapsular el carrusel de objetos, se hace uso del archivo generado en la función anterior para formar el directorio de los archivos que forman parte de la interactividad, es decir se encarga de asociar los ficheros con su respectivo nombre y a la carpeta que corresponden.

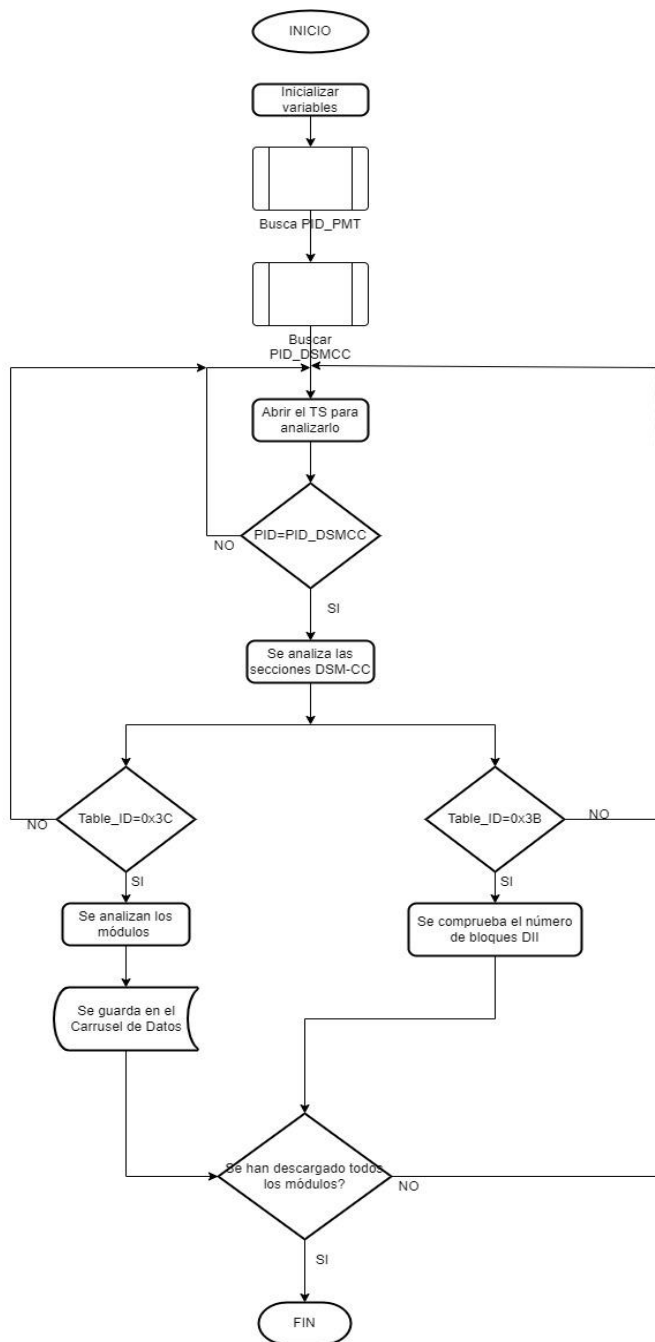
Función para extraer las secciones DSM-CC

Para la creación de esta función se usó como referencia lo realizado en (Nuñez, 2016) y según lo detallado en el capítulo II, tomando las consideraciones necesarias para extraer las secciones y eliminar los bytes de relleno, el algoritmo con el que se elaboró la función se detalla a continuación.

1. Como entrada para esta función, se tiene la ubicación del archivo que será analizado, así como del PID de las secciones DSM-CC que ha sido identificado anteriormente.
2. Se compara el PID trama por trama y si corresponde al de las secciones DSM-CC se procede con el análisis de la trama.

Figura 36

Diagrama de Flujo para el extractor de secciones DSM-CC



Nota. La figura muestra el diagrama de flujo para extraer las secciones DSM-CC.

Adaptado de Diagrama de flujo para el extractor de secciones DSM-CC, de Estudio, análisis e implementación de un software para construir un extractor y constructor de Carrusel de Datos, 2016.

3. Toma el valor del *pointer field*, el que nos ayudará a conocer la posición del *Table_id* y este a su vez ayudará a identificar si se trata de un bloque de datos o un bloque de control.
4. Tomando en cuenta si se trata de un bloque de control o de datos se empieza a extraer las secciones de acuerdo a las restricciones y casos especiales detallados en el capítulo II, este proceso es importante hacerlo de forma correcta para que al momento de extraer las secciones se eliminen correctamente los bytes de relleno que puedan existir.
5. Se detecta cuando existan bloques de control los cuales servirán como indicadores para la descarga de los ficheros, para el caso que se trabajó únicamente se encontró un bloque DII.
6. Los bloques de datos se almacenan en un documento de tipo TS que servirá para analizarlo con la siguiente función que permitirá eliminar las cabeceras de cada bloque.
7. A la salida de esta función se tiene el número de bloques encontrados tanto de datos, como de descarga.

Para un mejor entendimiento del algoritmo se puede observar el diagrama de flujo en la figura 36, el cual es una adaptación del que fue presentado en (Nuñez, 2016).

Función para extraer el carrusel de datos

Este proceso ayuda a eliminar las cabeceras de las secciones DBB que contienen datos únicamente, este proceso debería ser sencillo únicamente si en la función anterior extrajo correctamente las secciones, para lograr el objetivo en esta sección se trabajó con el número de módulos encontrados y con el *message_length* que será el parámetro que facilitará la eliminación de las cabeceras de cada sección, el algoritmo para esta función es el siguiente.

1. Tiene como entradas, la cantidad de módulos que han sido encontrados ya que a partir de este valor se irá a analizando el archivo que fue generado con la función anterior.
2. Analiza el identificador de cada módulo para comprobar que se trata de uno distinto, este proceso se lo realiza de trama en trama para ir eliminando las cabeceras de cada una de ellas.
3. De cada trama se analiza el *message_length* el cual permitirá separar las cabeceras de los datos que contiene cada una, luego de separar correctamente se lo guarda cada bloque en un nuevo archivo que servirá para la extracción del carrusel de objetos.
4. Este proceso se repite hasta terminar de analizar todos los módulos una vez se completen se sale de la función con el parámetro devuelto que es el archivo que se analizará en la siguiente función.

El diagrama de flujo para esta función se lo encuentra en el Anexo 1.

Función para extraer el carrusel de objetos

Es el proceso final realizado para extraer los datos que complementan a la interactividad del TS, en esta función se obtiene el directorio completo de la aplicación, para esto es importante reconocer tres componentes fundamentales que se tienen con los bloques DBB extraídos, dentro de cada sección se indicará si se trata de objetos de tipo fichero, de tipo directorio o de tipo puerta de servicio, siendo la principal diferencia lo que contiene cada uno de ellos, el de tipo fichero contiene los datos de cada archivo, los de tipo directorio y puerta de servicio contienen los nombres de los archivos así como de las direcciones dentro del directorio, para la elaboración de esta función se siguió el siguiente algoritmo:

1. Se toma el archivo generado en la función anterior y se obtiene el tamaño del mismo este servirá para determinar el momento en el que se terminó la extracción y se pueda empezar con la asignación de nombres y direcciones dentro del directorio.
2. Se compara bloque a bloque para determinar el tipo de cada uno, se compara con sus iniciales “file”, “dir” o “srg”.
3. Si se trata de un objeto de tipo “file”, se analiza su cabecera para extraer el contenido, así como la clave de objeto que más adelante servirá para asociarlo con su nombre y con su dirección dentro del directorio.
4. Cuando se trata de un objeto tipo “dir” o “srg” se analiza la cabecera para obtener la clave del objeto, así como el número de componentes del mismo, luego se guarda la información antecediendo de su la clave del objeto y de su clave de asociación.
5. Cuando se termina de analizar todo el carrusel de datos se procede a organizar el directorio de la aplicación, para esto se hace uso de la clave del objeto y de su clave de asociación.
6. Con ayuda de la clase de objeto y la clave de asociación se asigna a cada fichero su nombre y su ubicación dentro del directorio, al finalizar este proceso se tienen todos los archivos necesarios para las aplicaciones interactivas.

El diagrama de flujo para esta función se lo encuentra en el Anexo 2.

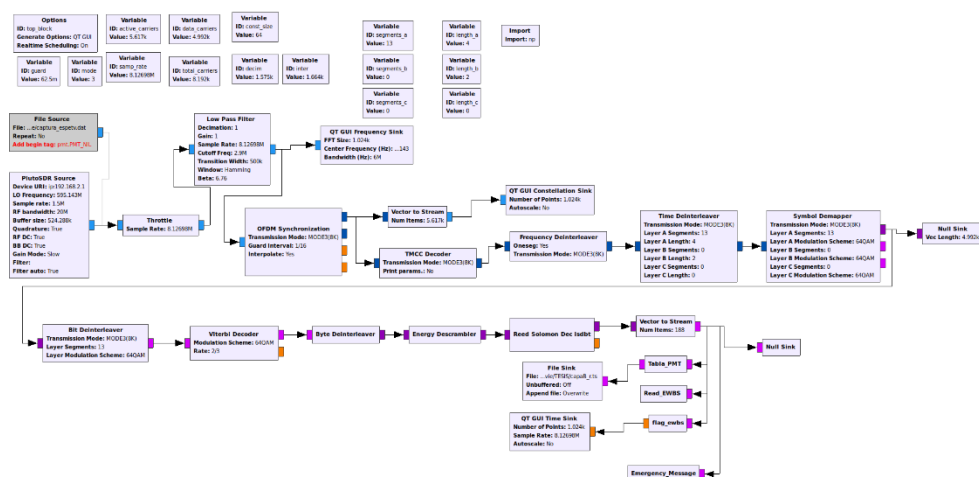
Diagrama completo del Receptor

Luego de explicar los bloques que fueron creados para extraer el mensaje EWBS y los archivos que complementan la aplicación interactiva, el diagrama de bloques queda tal como se muestra en la figura 37, los bloques se encargarán de

analizar el TS generado y mostrarán sus resultados en la pantalla, esto se puede observar en el capítulo IV.

En la figura 37 se puede observar varios bloques, los cuales fueron descargados desde Git-Hub, y se los ubico de acuerdo a los mencionado en el capítulo II, el orden en el que deben ir se puede apreciar en la figura 3, los bloques creados ya han sido explicados anteriormente y se colocan al final porque su funcionamiento se basa en el análisis del TS que se genera, el bloque de entrada sirve para capturar la señal con el SDR, este bloque y el bloque de salida del SDR serán explicados a continuación.

Figura 37
Receptor en GNU Radio



Nota. La figura muestra el diagrama del receptor completo.

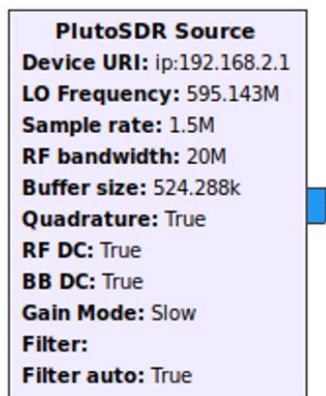
PlutoSDR Source

Este bloque viene incluido en el módulo que contiene funcionalidades para el SDR Adalm-Pluto, se lo descargó de Git-Hub, como se puede observar en la figura 38 se tiene una salida que se conecta directamente con los bloques que permiten recuperar el TS.

En la figura 38 se observa externamente el bloque en GNU Radio, son de suma importancia los datos ingresados en el bloque, esto se puede observar en la figura 39 y en la tabla 10 se detalla que función cumple cada uno de los parámetros mostrados.

Figura 38

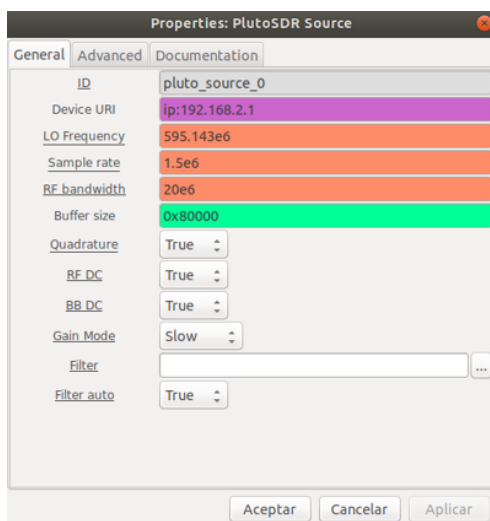
Bloque PlutoSDR Source



Nota. La figura muestra el bloque PlutoSDR Source en la interfaz de GNU Radio.

Figura 39

Parámetros PlutoSDR Source



Nota. La figura muestra los parámetros que se debe ingresar en el bloque PlutoSDR Source.

Tabla 10
Parámetros bloque PlutoSDR Source

Parámetro	Descripción
ID	Permite identificar el nombre del bloque de procesamiento.
Device URI	Se usa para la comunicación entre el PC y el SDR, estas especificaciones se pueden extraer del documento informativo de Adalm Pluto, para este caso se usó la IP:192.168.2.1
LO Frequency (MHz)	Se coloca la frecuencia a la cual trabajará el Oscilador local para recepción sus rangos van desde 70 MHz hasta 6 MHz.
Sample Rate (MSPS)	Indica la frecuencia a la que trabajará el equipo ingreso/salida de muestras, el rango es: 65.2 kSPS a 61.44 MSPS.
RF Bandwidth (MHz)	Configura los filtros analógicos RX TIA LPF y RX BB LPF que pueden tomar valores en el rango de 200kHz a 52MHz
Buffer size	Indica el tamaño del buffer interno en muestras, el bloque procesa un solo buffer, se debe ingresar el valor en decimal o hexadecimal.
Tracking	Quadrature, RF, DC, BB DC
Gain Mode	Se puede escoger entre los modos de ganancia disponibles: manual, slow, Hybrid o fast.
Manual Gain (dB)	Si se elige el tipo de ganancia manual se coloca un valor en dBs que puede ir entre 0 y 71 dB.
Filter	Permite cargar archivos con la configuración de filtros FIR.
Filter Audio	Si se habilita, carga un filtro predeterminado que permite velocidades de muestreo en banda base más bajas.

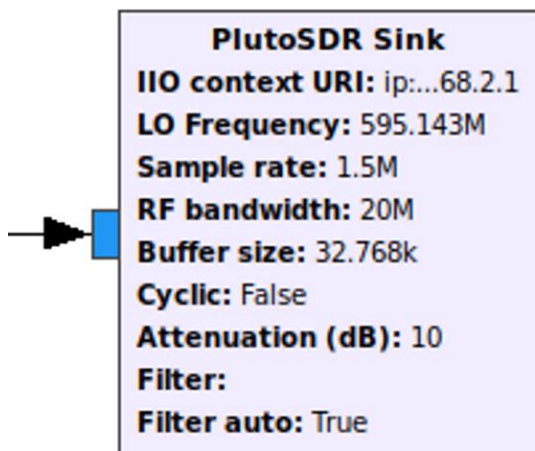
Nota. En la tabla se detalla la información que se requiere en el bloque PlutoSDR

Source. Adaptado de Analog Devices, de Using the PlutoSDR blocks, 2021b, Obtenido de: https://wiki.gnuradio.org/index.php/PlutoSDR_Source

PlutoSDR Sink

Este bloque sirve para enviar la señal mediante el Adalm-Pluto, se lo usa para el transmisor, tal como se muestra en el capítulo IV, en la figura 40 se observa el bloque en GNU-Radio, posee una entrada que recibe la señal que va a ser transmitida mediante el SDR.

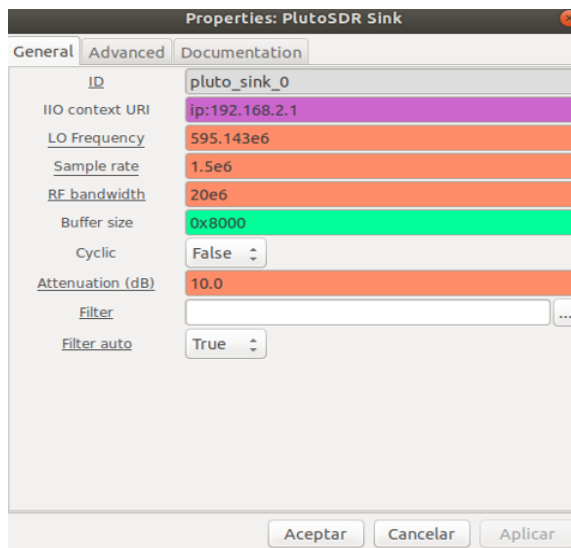
Figura 40
Bloque Pluto SDR Sink



Nota. La figura muestra el bloque PlutoSDR Sink en la interfaz de GNU Radio.

La figura anterior muestra la parte externa del bloque, al igual que el bloque Source, se deben ingresar ciertos parámetros para su funcionamiento, estos se muestran en la figura 41 y la descripción de cada uno se detalla en la tabla 11.

Figura 41
Parámetros bloque Pluto SDR Sink



Nota. La figura muestra los parámetros que requiere el bloque PlutoSDR Sink.

Tabla 11
Parámetros bloque PlutoSDR Sink

Parámetro	Descripción
ID	Identificativo del bloque de procesamiento
IIO context URI	Se usa para la comunicación entre el PC y el SDR, estas especificaciones se pueden extraer del documento informativo de Adalm Pluto, para este caso se usó la IP:192.168.2.1
LO Frequency (MHz)	Se coloca la frecuencia a la cual trabajará el Oscilador local para transmisión sus rangos van desde 70 MHz hasta 6 MHz
Sample rate (MSPS)	Indica la frecuencia a la que trabajará el equipo ingreso/salida de muestras, el rango es: 65.2 kSPS a 61.44 MSPS.
RF Bandwidth (MHz)	Configura los filtros analógicos TX BB LPF y TX Secondary LPF que pueden tomar valores en el rango de 200kHz a 52MHz
Buffer size	Indica el tamaño del buffer interno en muestras, el bloque procesa un solo buffer, se debe ingresar el valor en decimal o hexadecimal.
Ciclic	Configura si se activa o no el modo cíclico.
Attenuation (dB)	Indica la atenuación que puede tener el transmisor, el rango de valores es de 0 a -89.75 dB en pasos de 0.25 dB.
Filter	Permite cargar una configuración del filtro FIR.
Filter Auto	Cuando es habilitado, carga un filtro predeterminado.

Nota. En la tabla se detallan los parámetros que se deben ingresar en el bloque.

Adaptado de Analog Devices, de Using the PlutoSDR blocks, 2021c, Obtenido de:

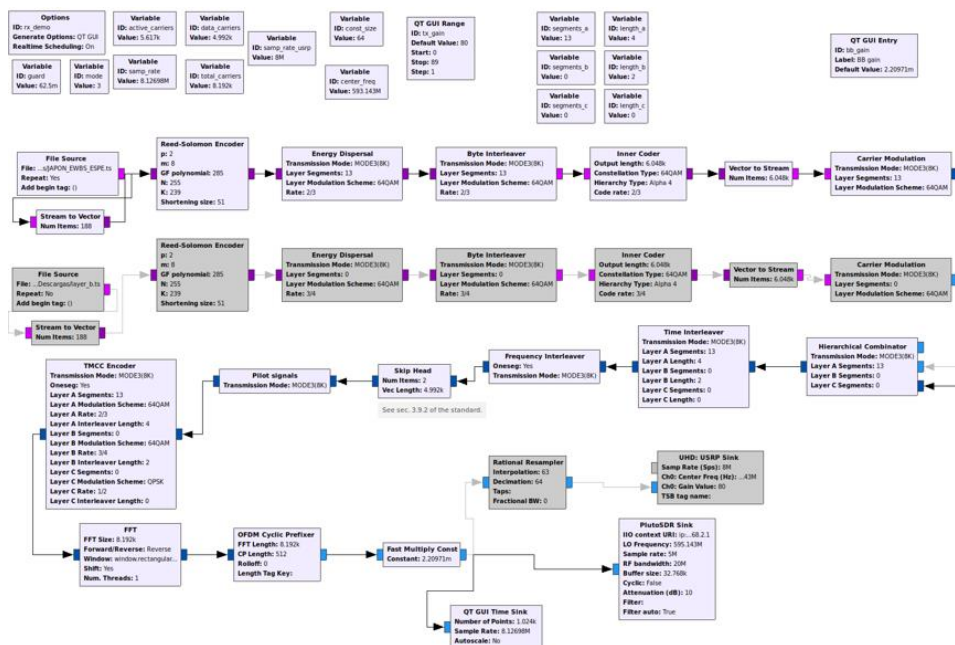
https://wiki.gnuradio.org/index.php/PlutoSDR_Sink.

CAPÍTULO IV

Análisis de Resultados

En esta sección se detallan los resultados obtenidos con el receptor, se evalúan parámetros de calidad de la señal obtenida con técnicas objetivas, así como corroborar si se cumple o no el proceso para la extracción del mensaje de emergencia y de los archivos que constituyen la aplicación interactiva, como se ha mencionado se tuvo la necesidad de implementar un transmisor en GNU Radio, el cual se lo muestra en la figura 42, para la transmisión se usa el mismo SDR que funciona como receptor, los bloques agregados fueron descargados desde Git-Hub y pertenecen al módulo ISDBT, además se basó en el diseño implementado en (Grijalva, 2022), para tomar ciertas consideraciones para que funcione correctamente.

Figura 42
Transmisor de TDT según el estándar ISDB-Tb



Nota. La imagen muestra el diagrama de bloques en GNU Radio para el transmisor de TDT.

Para realizar las pruebas de funcionamiento se evaluará dos escenarios, en ambos casos el mismo computador y el mismo SDR funcionarán como transmisor y receptor, para el primer escenario se tendrá conectado los puertos de transmisión y recepción a través del cable coaxial que se dispone en el SDR, en el escenario dos se conectarán las antenas y se reducirá las ganancias para comprobar cómo se comporta el sistema ante la presencia de ruido e interferencias.

Consideraciones Especiales

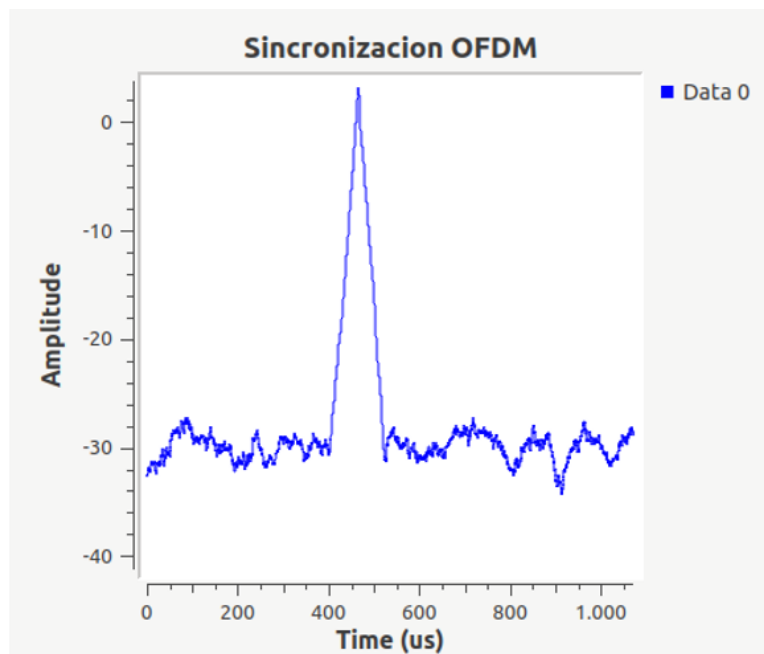
Para comprobar el correcto funcionamiento del receptor full-seg, es necesario tomar en cuenta dos gráficos que permiten visualizar, la sincronización OFDM y la constelación recibida, si estos dos parámetros son correctos se puede afirmar que los datos recibidos corresponden a una señal de TDT.

Sincronización OFDM

Con una tasa de muestreo mayor a la que indica el fabricante, se logra sincronizar el transmisor y el receptor, como se muestra en la figura 43, pero debido a que se recibe la señal por flujos existen pérdidas de información lo que ocasionaba la errónea extracción del TS y en ocasiones no se pueda visualizar la imagen y por lo tanto tampoco era posible realizar el análisis del TS, esto se puede observar en el gráfico de la constelación que se mostrará más adelante.

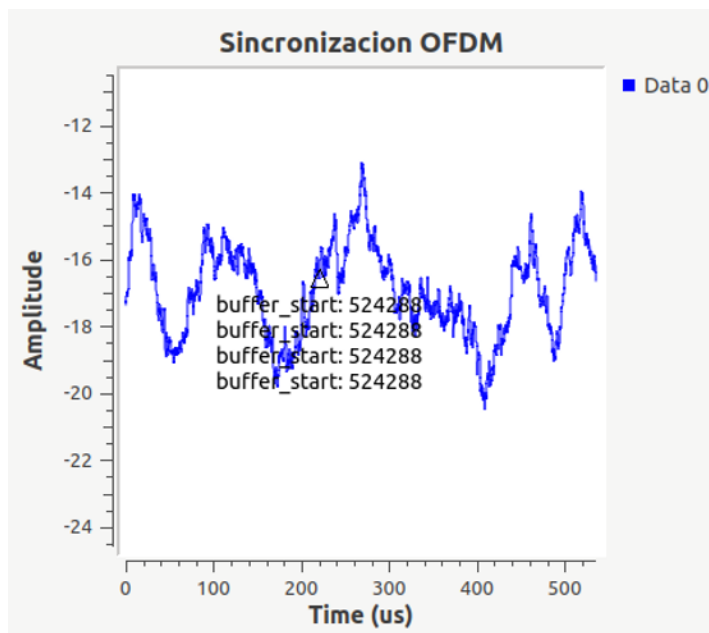
Cuando no se ingresan los datos correctos como el modo de operación o el prefijo cíclico, provoca que no exista sincronismo entre el transmisor y el receptor, un ejemplo en falla de la sincronización se lo puede ver en la figura 44.

Figura 43
Sincronización del símbolo OFDM



Nota. La figura muestra los picos de sincronización del símbolo OFDM.

Figura 44
Falta de sincronización del símbolo OFDM



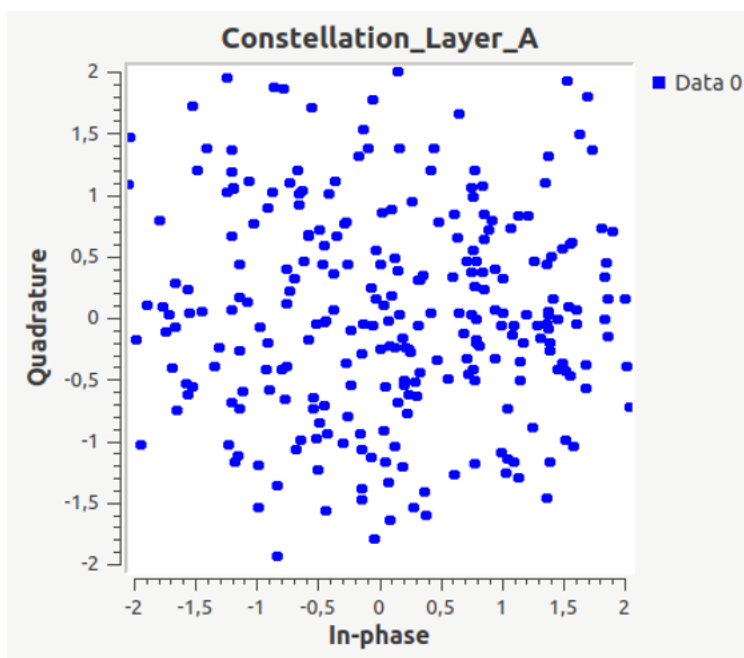
Nota. La figura muestra la falta de sincronización del símbolo OFDM.

Recepción de la constelación

Como se mencionó la tasa de muestreo máxima a la que trabaja el SDR de forma continua, ha sido un factor limitante a la hora de implementar el receptor full-seg, debido a que por especificaciones del estándar esta debe ser 8.126984 MHz, al tener este inconveniente no se podía captar la señal de un transmisor para TDT según el estándar ISDB-Tb, en la figura 45 se puede ver que no se conseguía la constelación deseada y el dispositivo Adalm-Pluto funcionaba de forma errática.

Figura 45

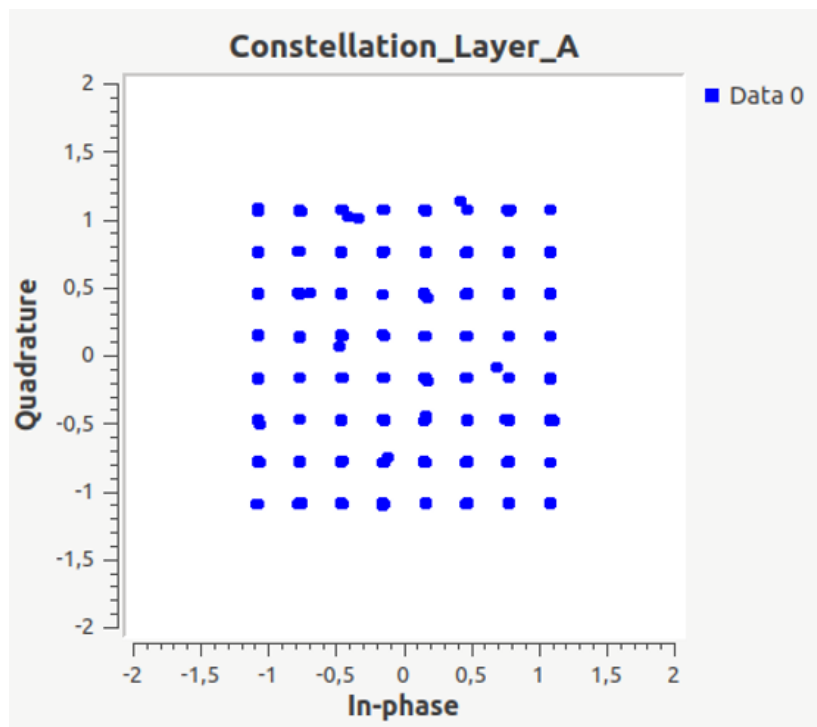
Constelación recibida a frecuencias de muestreo altas



Nota. La figura muestra la constelación recibida para frecuencias mayores a 5 MHz.

Quando se reciben los datos correctamente se tiene una constelación como la de la figura 46, en la que se puede apreciar claramente la modulación que se usa para transmitir y con esto los datos pueden seguir procesándose para extraer el TS y analizarlo.

Figura 46
Constelación recibida correctamente



Nota. La figura muestra la constelación recibida.

Escenario 1

En este primer escenario se recibe la señal de TDT a través de cable coaxial que viene incluido en el kit de equipamiento del SDR Adalm-Pluto, tal como se puede observar en la figura 47. Los archivos que se utilizarán para la transmisión son: *espetv.ts* el cual contiene los archivos necesarios para complementar la aplicación interactiva, el otro archivo que se utilizará es el *JAPON_EWBS_ESPE.ts* el cual contiene el sistema de alerta EWBS, ambos archivos van a ser transmitidos por la capa A, se utilizará una modulación de 64-QAM.

Figura 47

SDR Adalam-Pluto como transmisor y receptor escenario 1.



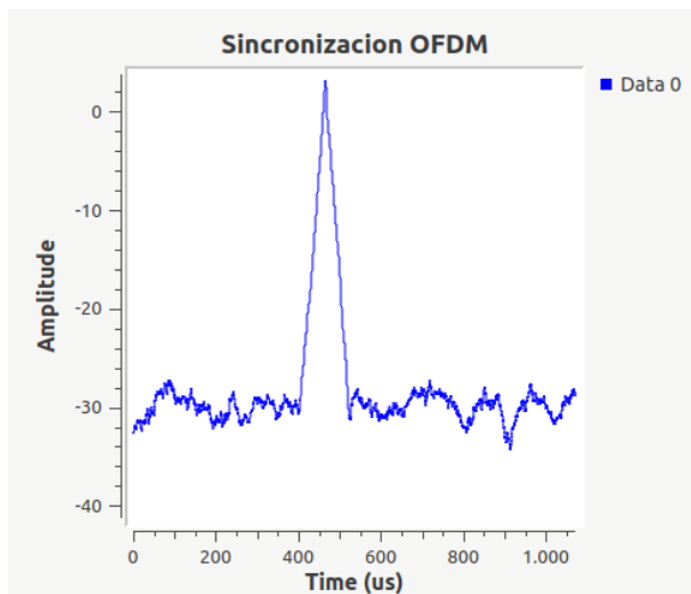
Nota. La figura muestra la conexión del cable coaxial del SDR Adalm-Pluto conectado al puerto de transmisión y al de recepción.

Transmisión para la aplicación interactiva.

Con el fin de extraer los archivos que complementan la aplicación interactiva, se transmitió el archivo *espetv.ts*, se utilizaron los siguientes bloques creados: el analizador de la tabla PMT y el analizador de las secciones DSM-CC, luego de terminar con la transmisión se realizó el análisis del archivo generado con el fin extraer los archivos que vienen empaquetados en el TS, para este caso no se realizó el análisis de las medidas de calidad objetivas debido a que al conectar los puertos directamente no existen interferencias que puedan ocasionar pérdidas de información.

Figura 48

Sincronización OFDM en la transmisión del archivo espetv.ts



Nota. La figura muestra los picos de sincronización del símbolo OFDM

El primer parámetro que se tomó en cuenta para corroborar el correcto funcionamiento del sistema, son los picos de la sincronización de símbolo OFDM en la figura 48 se puede observar que se cumple con este parámetro en la transmisión

Luego de comprobar que los parámetros ingresados en el receptor son adecuados, se procede con el análisis de la información del TMCC que se imprime en la ventana de GNU Radio, la información detallada es importante para conocer las capas que se usan para la transmisión, así como de la modulación usada en cada una de ellas, esto se puede observar en la figura 49.

Figura 49

Información del TMCC para la transmisión del archivo *espetv.ts*

```

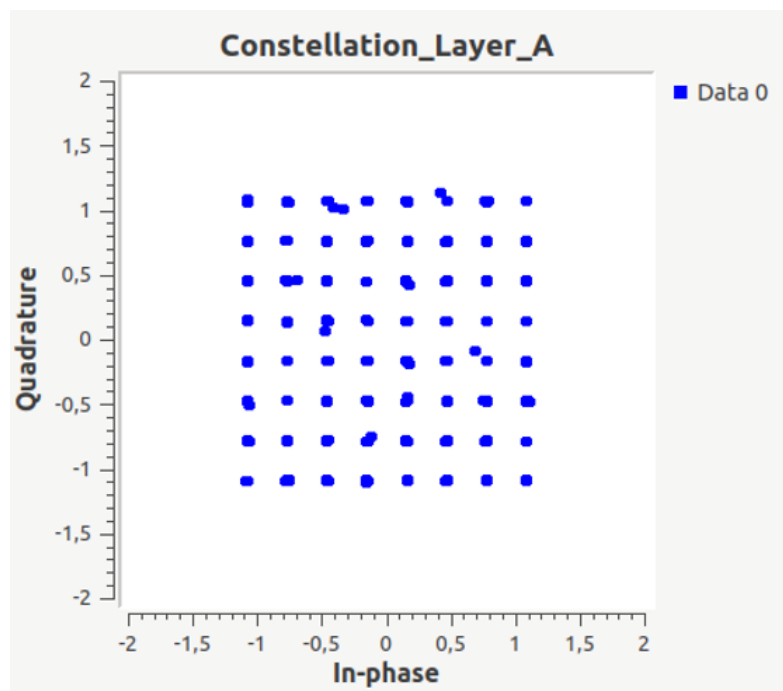
TMCC OK
-----
                                TMCC ANALYSIS
-----
Layer                            : A
Carrier Modulation Scheme         : 64QAM
Convolutional Code Rate          : 2/3
Time Interleaving Length         : 16(mode 1), 8(mode 2), 4(mode 3)
Number of segments for this layer : 13
-----
Layer                            : B
Carrier Modulation Scheme         : UNUSED
Convolutional Code Rate          : UNUSED
Time Interleaving Length         : UNUSED
Number of segments for this layer : UNUSED
-----
Layer                            : C
Carrier Modulation Scheme         : UNUSED
Convolutional Code Rate          : UNUSED
Time Interleaving Length         : UNUSED
Number of segments for this layer : UNUSED
-----

```

Nota. La figura muestra la información del TMCC.

Figura 50

Constelación recibida del archivo *espetv.ts*



Nota. La figura muestra la constelación recibida.

Una vez comprobado que la información del TMCC llega de forma correcta, se procede a visualizar la constelación mostrada en la figura 50, como se puede apreciar la modulación usada es una 64-QAM, luego de comprobar estos parámetros se continúa con los demás bloques encargados de generar el TS para su posterior análisis.

Para empezar con el análisis del TS generado en GNU Radio se revisó la información de la tabla PMT incluidas en el TS, de las cuales se puede extraer información de los servicios que vienen incluidos en el archivo de TDT, estos parámetros se pueden observar en la figura 51.

Figura 51
Información de la Tabla PMT del archivo espstv.ts

```

ES UNA PMT !!!
PID:
 1031 y en hex: 0x407
Section length: 86 y en hex: 0x56
Section syntax error: 1 entonces True
CRC-32: [192 79 103 116]
Program number: 59232 y en hex: 0xe760
Version Number: 0 y en hex: 0x0
Current next indicator: 1 entonces True
PCR_PID: 2065 y en hex: 0x811
Program information length: 0 y en hex: 0x0
-----
----- Information 1 -----
Stream Type identifier: 2 en hex: 0x2 : Video Conforme ITU Recommendation H.262
Elementary Stream PID: 2065 en hex 0x811
ES information length: 0 en hex: 0x0
-----
----- Information 2 -----
Stream Type identifier: 3 en hex: 0x3 : Audio Conforme ISO/EC 11172-3
Elementary Stream PID: 2075 en hex 0x81b
ES information length: 0 en hex: 0x0
-----
----- Information 3 -----
Stream Type identifier: 5 en hex: 0x5 : Seccion
Elementary Stream PID: 2001 en hex 0x7d1
ES information length: 12 en hex: 0xc
-----
----- Information 4 -----
Stream Type identifier: 11 en hex: 0xb : Conforme ISO/IEC 13818-6 (tipo B)
Elementary Stream PID: 2004 en hex 0x7d4
ES information length: 41 en hex: 0x29
-----
Stream type identifier: [2, 3, 5, 11]
Elementary Stream PID: [2065, 2075, 2001, 2004]
ES information length: [0, 0, 12, 41]

```

Nota. La figura muestra la información extraída de una de las tablas PMT del archivo espstv.ts

También se ha incluido un campo en el que se muestra la información de las tablas PMT que contienen secciones DSM-CC, las cuales encuentran los paquetes con el PID que corresponde y saca la información necesaria tal como se muestra en la figura 52, este es el último bloque en GNU Radio que analiza el TS generado, posterior a esto se procederá a extraer los archivos que vienen incluidos.

Figura 52

Información de las secciones DSM-CC

```

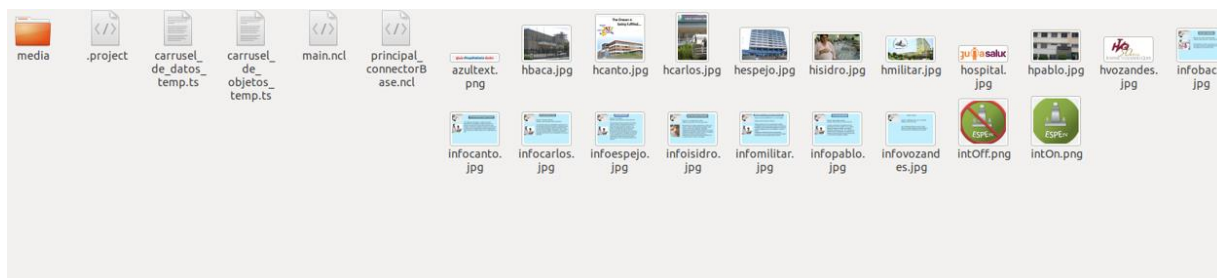
----- Information 3 -----
Stream Type identifier: 5 en hex: 0x5 : Seccion
Elementary Stream PID: 2001 en hex 0x7d1
ES information length: 12 en hex: 0xc
-----
----- Information 4 -----
Stream Type identifier: 11 en hex: 0xb : Conforme ISO/IEC 13818-6 (tipo B)
Elementary Stream PID: 2004 en hex 0x7d4
ES information length: 41 en hex: 0x29

```

Nota. La figura muestra la información de las secciones DSM-CC.

Figura 53

Archivos extraídos del archivo espetv.ts



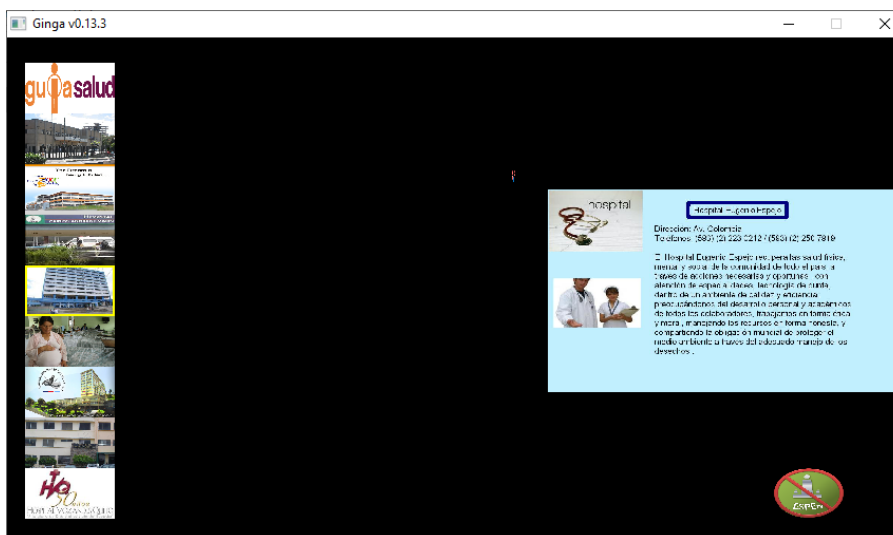
Nota. La figura muestra los archivos extraídos.

Para la extracción de los archivos que complementan la aplicación interactiva, fue necesario esperar a que el archivo sea generado completamente, debido a que los bloques de GNU Radio inicializan las variables cada que un flujo de datos ingresa a los mismos, y esto afectaba a las variables utilizadas en el proceso de extracción que ha

sido detallado en el capítulo III, luego de ejecutar el programa principal los archivos extraídos se los almacena en una sola carpeta tal como se muestra en la figura 53.

Para reproducir la aplicación interactiva es necesario el uso de GINGA, se debe abrir el archivo principal y se puede observar el funcionamiento de la aplicación interactiva, tal como se muestra en la figura 54.

Figura 54
Ejecución de la aplicación interactiva en GINGA

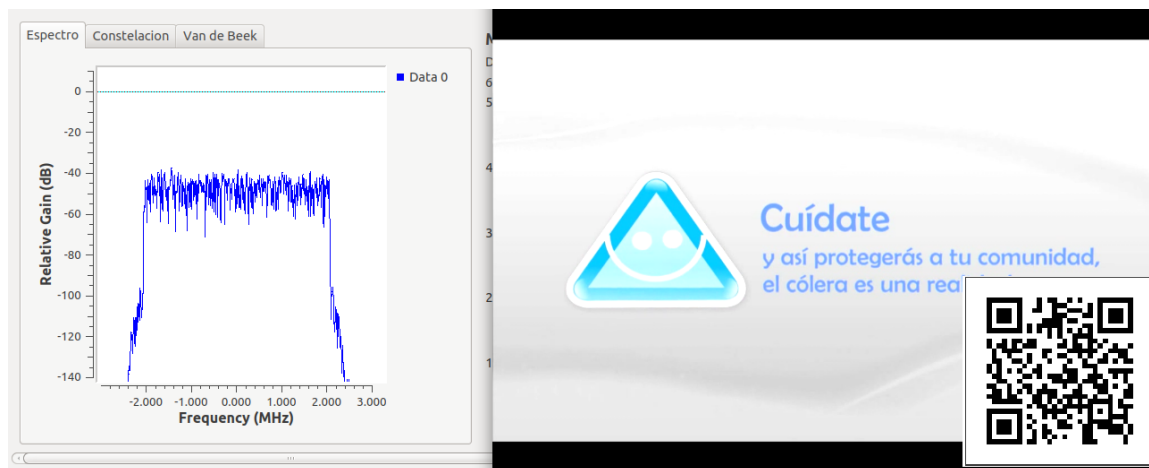


Nota. La figura muestra la ejecución de la aplicación interactiva

El funcionamiento en general del sistema de recepción y extracción de los archivos que complementan la aplicación interactiva, se lo puede apreciar en la figura 55, que contiene un código QR el cual direcciona hacia un video explicativo donde se detalla la realización y el funcionamiento del transmisor y receptor, se muestran los archivos extraídos y su ejecución en GINGA, además del respectivo análisis de las tablas PMT que se incluyen en el TS.

Figura 55

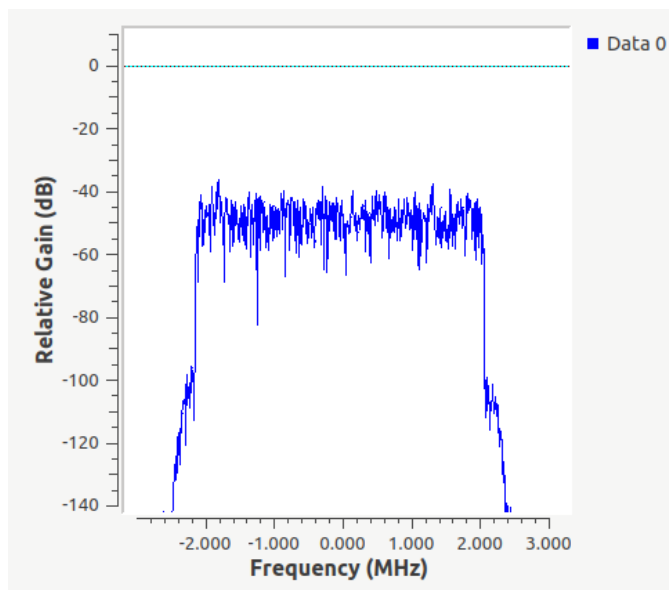
Explicación de la extracción de los archivos que complementan la aplicación interactiva



Nota. La figura muestra de forma general la ejecución del receptor de aplicaciones interactivas.

Transmisión para el Mensaje EWBS**Figura 56**

Espectro de la señal recibida



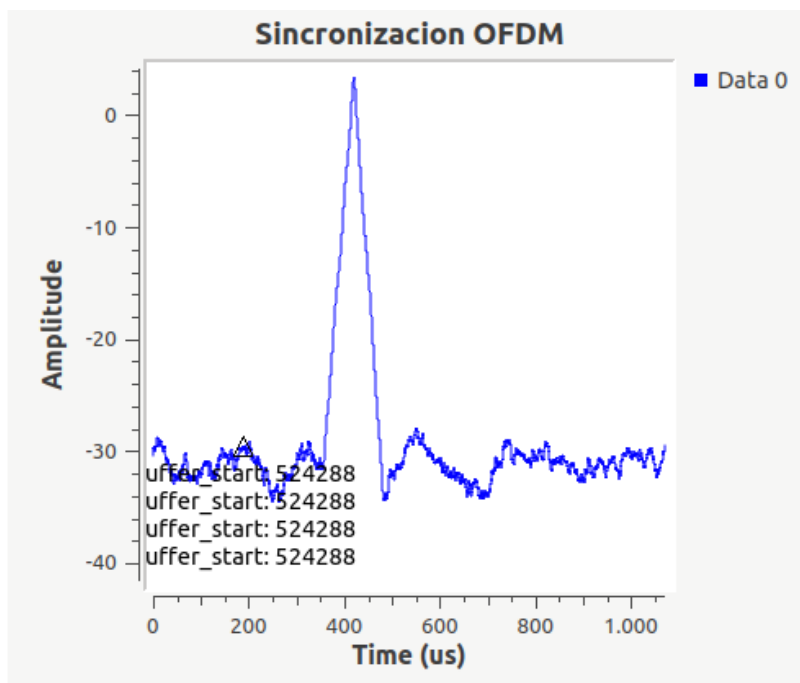
Nota. La figura muestra el espectro que se recibe a través del SDR Adalm-Pluto.

Para recibir el mensaje de alerta EWBS se usó el archivo *JAPON_EWBS_ESPE.ts* en el transmisor, de la misma forma que para el caso de las aplicaciones interactivas, fue necesario comprobar que los datos recibidos son útiles, para esto se verificó el espectro de la señal como se puede ver en la figura 56.

Otro parámetro que se tomó en cuenta es la sincronización entre el transmisor y el receptor, se lo puede verificar en los picos de sincronización de los símbolos de OFDM, tal como se muestra en la figura 57.

Figura 57

Sincronización de OFDM para la transmisión del archivo JAPON_EWBS_ESPE.ts



Nota. La figura muestra los picos de sincronización de la señal recibida.

Luego de comprobar la sincronización entre el transmisor y el receptor se procede con la verificación de la información TMCC, en la que se puede observar las capas que se están usando para la transmisión, así como de cuantos segmentos se

transmiten por cada una, la modulación que se ocupa y la tasa de código, esta información se puede observar en la figura 58.

Figura 58

Información de TMCC del archivo JAPON_EWBS_ESPE.ts

```

-----
                                TMCC ANALYSIS
-----
Layer                               : A
Carrier Modulation Scheme           : 64QAM
Convolutional Code Rate              : 2/3
Time Interleaving Length             : 16(mode 1), 8(mode 2), 4(mode 3)
Number of segments for this layer    : 13
-----
Layer                               : B
Carrier Modulation Scheme           : UNUSED
Convolutional Code Rate              : UNUSED
Time Interleaving Length             : UNUSED
Number of segments for this layer    : UNUSED
-----
Layer                               : C
Carrier Modulation Scheme           : UNUSED
Convolutional Code Rate              : UNUSED
Time Interleaving Length             : UNUSED
Number of segments for this layer    : UNUSED
-----
TMCC OK

```

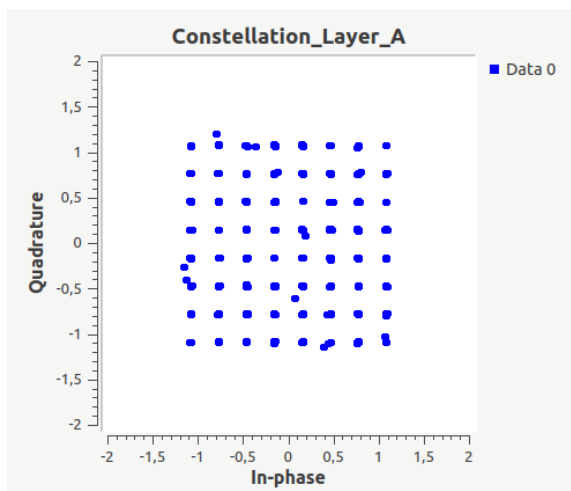
Nota. La figura muestra la información del TMCC recibida del archivo JAPON_EWBS_ESPE.

El último parámetro que se verificó de forma gráfica para la correcta recepción del flujo de datos es la constelación, en este caso se trata de una 64-QAM, tal como se muestra en la figura 59.

Una vez se comprobó que la señal captada es recibida con los parámetros adecuados se procede con la generación y análisis del TS, como primer punto se tiene el bloque *Read_EWBS*, el cual fue tomado de (Castillo, 2021), con sus respectivos cambios, el bloque se encarga del análisis de las tablas PMT contenidas en el flujo de datos e imprime esta información, así como de la información del descriptor EWBS cuando lo recibe.

Figura 59

Constelación de la señal recibida en la transmisión del archivo JAPON_EWBS_ESPE.ts



Nota. La figura muestra la correcta recepción de la constelación de la señal transmitida.

En la figura 60, se puede observar el análisis de una tabla PMT que no contiene el descriptor EWBS, y se muestra la información de los servicios que contiene el TS.

Figura 60

Análisis de la tabla PMT sin descriptor EWBS

```

-----
ES UNA PMT !!!
PID: 8136 y en hex: 0x1fc8
Section length: 33 y en hex: 0x21
Section syntax error: 1 entonces
  True
CRC-32: [113 30 140 171]
Program number: 2648 y en hex: 0xa58
Version Number: 5 y en hex: 0x5
Current next indicator: 1 entonces True
PCR_PID: 528 y en hex: 0x210
Program information length: 0 y en hex: 0x0
-----
----- Information 1 -----
Stream Type identifier: 27 en hex: 0x1b : Video conforme ITU Recommendation H.264 e ISO/IEC 14496-10
Elementary Stream PID: 529 en hex 0x211
ES information length: 3 en hex: 0x3
-----
----- Information 2 -----
Stream Type identifier: 17 en hex: 0x11 : Audio conforme ISO/IEC 14496-3
Elementary Stream PID: 530 en hex 0x212
ES information length: 7 en hex: 0x7
-----
Stream type identifier: [27, 17]
Elementary Stream PID: [529, 530]
ES information length: [3, 7]

```

Nota. La figura muestra la información de una PMT que no contiene el descriptor EWBS.

Cuando se recibe una PMT que, si contiene el descriptor EWBS, es posible apreciar como la información que se extrae de flujo de datos contiene más parámetros, como lo es el análisis del descriptor en mención, esto se lo puede observar en la figura 61.

Figura 61

Análisis del descriptor EWBS contenido en una tabla PMT

```

----- Descriptor EWBS -----
Descriptor EWBS: 252 y en hex: 0xfc
Descriptor length: 54 y en hex 0x36
Service ID: 2624 y en hex: 0xa40
Bandera EWBS: 1 entonces True
Signal Level: 0 entonces False
Area code length: 50 y en hex 0x32

```

Nota. La figura muestra el análisis del descriptor EWBS.

En el descriptor EWBS es posible encontrar información, como el código de área de las regiones en las cuales ocurre el incidente que provoca la activación del sistema de alerta EWBS, las ciudades en las que se activa en este TS se las detalla en la tabla 12.

Tabla 12

Ciudades en las que se activa el Sistema de Alerta EWBS

Provincia	Cantón
Pichincha	Cayambe
Pichincha	Mejía
Pichincha	Pedro Moncayo
Pichincha	Pedro Vicente Maldonado
Pichincha	Puerto Quito
Pichincha	Quito
Pichincha	Rumiñahui
Pichincha	San Miguel de los Bancos

Provincia	Cantón
Cotopaxi	La Maná
Cotopaxi	Latacunga
Cotopaxi	Pangua
Cotopaxi	Pujilí
Cotopaxi	Salcedo
Cotopaxi	Saquisilí
Cotopaxi	Sigchos
Napo	Archidona
Napo	Carlos Julio Arosemena
Napo	El Chaco
Napo	Quijos
Napo	Tena
Tungurahua	Ambato
Tungurahua	Baños de Agua Santa
Tungurahua	Cevallos
Tungurahua	Mocha
Tungurahua Patate	Patate

Nota. En la tabla se detallan las ciudades en las cuales se activa el sistema de alerta.

La misma información se puede observar en el análisis que se realiza en el bloque *Read_EWBS* tal como se puede observar en la figura 62.

Figura 62

Ciudades en las cuales se activa el Sistema de Alerta EWBS

```
Area Code: 30 y en hex: 0x1e
Area Code: 108 y en hex: 0x6c
Area Code: 145 y en hex: 0x91
Area Code: 146 y en hex: 0x92
Area Code: 158 y en hex: 0x9e
Area Code: 168 y en hex: 0xa8
Area Code: 172 y en hex: 0xac
Area Code: 183 y en hex: 0xb7
Area Code: 91 y en hex: 0x5b
Area Code: 97 y en hex: 0x61
Area Code: 137 y en hex: 0x89
Area Code: 159 y en hex: 0x9f
Area Code: 173 y en hex: 0xad
Area Code: 198 y en hex: 0xc6
Area Code: 202 y en hex: 0xca
Area Code: 8 y en hex: 0x8
Area Code: 27 y en hex: 0x1b
Area Code: 55 y en hex: 0x37
Area Code: 164 y en hex: 0xa4
Area Code: 211 y en hex: 0xd3
Area Code: 5 y en hex: 0x5
Area Code: 18 y en hex: 0x12
Area Code: 33 y en hex: 0x21
Area Code: 113 y en hex: 0x71
Area Code: 141 y en hex: 0x8d
```

Nota. La figura muestra las ciudades en las cuales se activa el sistema de alerta EWBS.

Además, se realiza el análisis de los servicios que contiene esta PMT, se puede evidenciar la presencia de un PES el cual contiene la información para la sobreimposición, en este caso se tiene el PID 278 el cual corresponde al paquete que contiene la información del lenguaje para el mensaje de sobreimposición, así como del mensaje en cuestión, el análisis del servicio se lo puede apreciar en la figura 63.

Figura 63
Análisis de la información para la superimposición

```

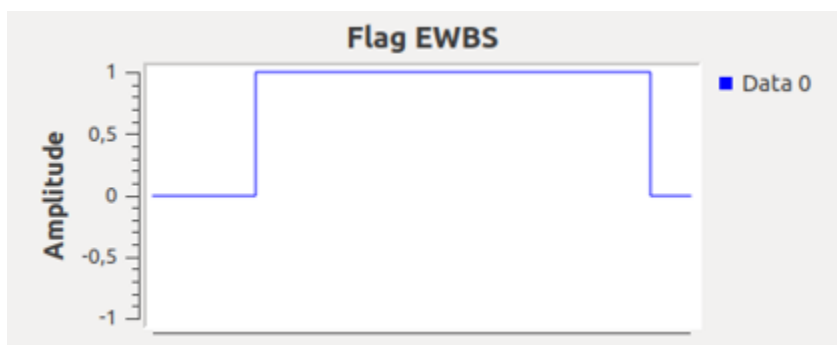
----- Information 1 -----
Stream Type identifier: 27 en hex: 0x1b : Video conforme ITU Recommendation H.264 e ISO/IEC 14496-10
Elementary Stream PID: 273 en hex 0x111
ES information length: 6 en hex: 0x6
27
----- Information 2 -----
Stream Type identifier: 6 en hex: 0x6 : Paquetes PES
Elementary Stream PID: 278 en hex 0x116
ES information length: 8 en hex: 0x8
6
*****
* ----- Descriptor de Superimposicion EWBS----- *
* Descriptor identification: 82 en hex 0x52
* Descriptor length: 1 en hex 0x1
* Component SUPERIMPOSE 56 en hex 0x38
* Descriptor component data: 253 en hex 0xfd
* Descriptor length data component: 3 en hex 0x3
* Data coding method id: 8 en hex 0x8
* Additional identifier info: 60 en hex 0x3c
*****
----- Information 3 -----
Stream Type identifier: 17 en hex: 0x11 : Audio conforme ISO/IEC 14496-3
Elementary Stream PID: 274 en hex 0x112
ES information length: 7 en hex: 0x7
17
-----

```

Nota. La figura muestra el análisis para extraer la información para la superimposición.

Otro de los bloques de análisis que han sido creados es el *flag_EWBS*, el cual se encarga de activar una sirena que indica la activación del sistema de alerta EWBS, y de forma gráfica se observa la activación de la bandera en la figura 64.

Figura 64
Activación de la bandera EWBS



Nota. Se muestra la activación de la bandera EWBS.

Por último, se tiene el bloque *Emergency_message* el cual extrae el mensaje de alerta, para esto se hace uso del PID encontrado anteriormente, este bloque se encarga de generar un archivo de texto con el mensaje encontrado, en el cual indica la hora en la que se activó el sistema de alerta como se puede evidenciar en la figura 65.

Figura 65
Mensaje EWBS

```
*****
-----Alerta de Emergencia-----
Alerta de Emergencia Cotopaxi -ESPE
2022-06-12 18:49:04
*****
```

Nota. La figura muestra el mensaje EWBS extraído.

Figura 66
Hoja de cálculo con los códigos de área

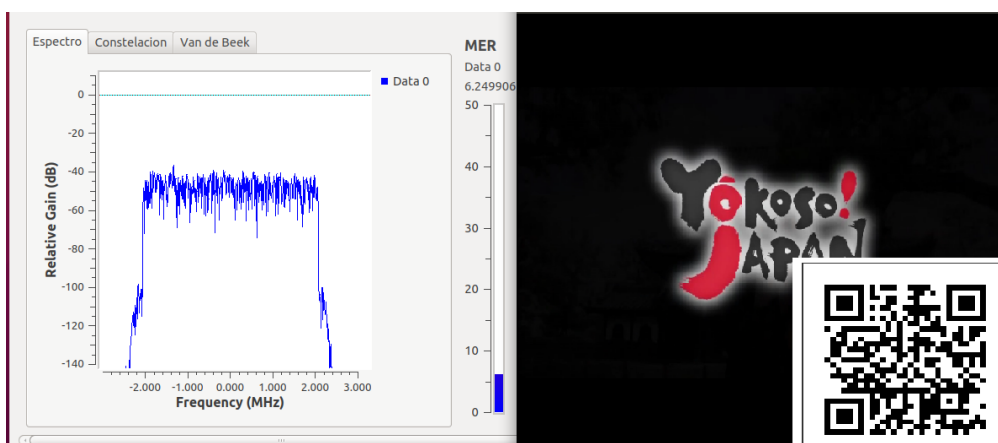
Decimal	Hexadec	Binario	Provincia	Canton
30	0x1e	00000001	PICHINCH	CAYAMBE
108	0x6c	00000110	PICHINCH	MEJIA
145	0x91	00001001	PICHINCH	PEDRO MONCAYO
146	0x92	00001001	PICHINCH	PEDRO VICENTE MALDONADO
158	0x9e	00001001	PICHINCH	PUERTO QUITO
168	0xa8	00001010	PICHINCH	QUITO
172	0xac	00001010	PICHINCH	RUMINAHUI
183	0xb7	00001011	PICHINCH	SAN MIGUEL DE LOS BANCOS
91	0x5b	00000101	COTOPAX	LA MANA
97	0x61	00000110	COTOPAX	LATACUNGA
137	0x89	00001000	COTOPAX	PANGUA
159	0x9f	00001001	COTOPAX	PUJILI
173	0xad	00001010	COTOPAX	SALCEDO
198	0xc6	00001100	COTOPAX	SAQUISILI
202	0xca	00001100	COTOPAX	SIGCHOS
8	0x8	00000000	NAPO	ARCHIDONA
27	0x1b	00000001	NAPO	CARLOS JULIO AROSEMENA
55	0x37	00000011	NAPO	EL CHACO
164	0xa4	00001010	NAPO	QUIJOS
211	0xd3	00001101	NAPO	TENA
5	0x5	00000000	TUNGURA	AMBATO
18	0x12	00000001	TUNGURA	BANOS DE AGUA SANTA
33	0x21	00000010	TUNGURA	CEVALLOS
113	0x71	00000111	TUNGURA	MOCHA
141	0x8d	00001000	TUNGURA	PATATE

Nota. La figura muestra el contenido del archivo generado con los códigos de área.

Al finalizar la ejecución del programa se puede verificar los archivos generados en la activación de la alerta EWBS, el bloque *Read_EWBS* genera una hoja de cálculo en la cual se puede observar la información de las ciudades en las cuales se activó la alerta, este archivo generado se puede observar en la figura 66.

Para verificar el funcionamiento en general del proceso de extracción para el sistema de alerta de emergencia, se muestra la figura 67, la cual tiene la reproducción del TS recibido, así como de los parámetros que indican la correcta recepción de la información, y el código QR re-direcciona hacía un video explicativo del funcionamiento.

Figura 67
Funcionamiento del Receptor EWBS



Nota. La figura muestra el funcionamiento del receptor EWBS.

Escenario 2

Para analizar las medidas de calidad objetivas del receptor se utilizaron las antenas que vienen incluidas en el SDR, tal como se puede ver en la figura 68, de esta forma se puede variar la potencia del transmisor y del receptor para obtener diferentes escenarios, y de esta forma realizar un análisis de las medidas de calidad.

Figura 68

SDR Adalam-Pluto como transmisor y receptor escenario 2.



Nota. La figura muestra el SDR usando las antenas para transmitir y recibir.

Medidas de calidad del receptor de aplicaciones interactivas y mensaje EWBS

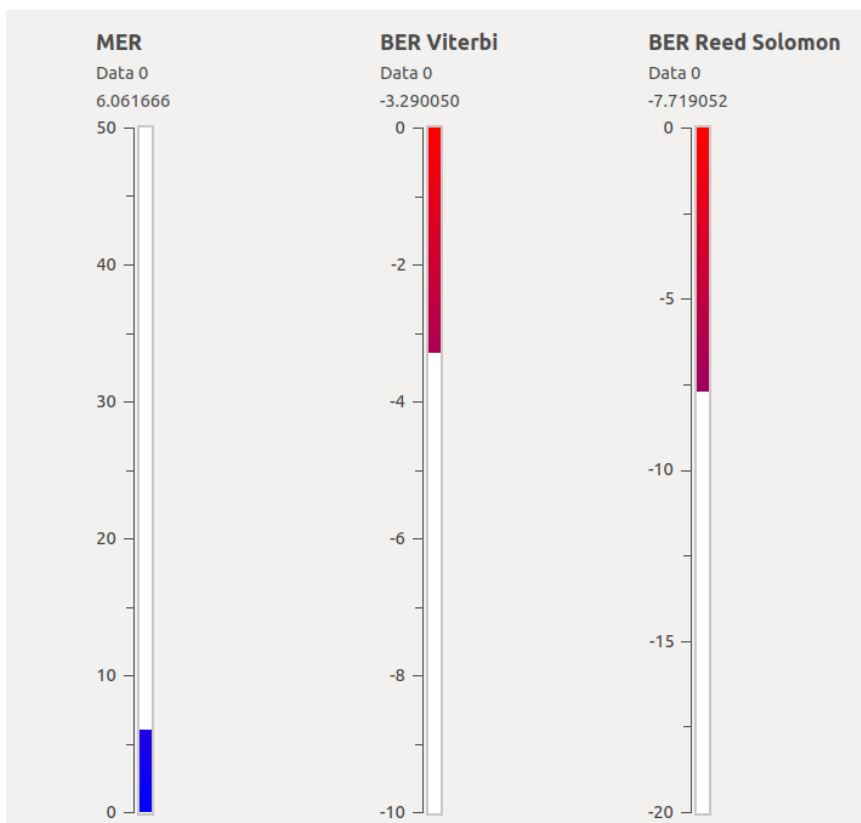
Para el análisis de las medidas objetivas fue necesario usar las antenas del kit que viene incluido en el SDR, esto se muestra en la figura 68, para lograr la variación en la potencia de recepción se realizaron cambios en los bloques Source y Sink del Adalm-Pluto, y de esta forma se generó un escenario propicio para la obtención de las medidas objetivas que se las puede realizar gracias al uso de los bloques implementados por (Garzón, 2018).

Para realizar las medidas se varió la potencia de recepción, para esto se trabajó en el rango de -90 dBm hasta -60 dBm variando 5 dBs para cada medición, los parámetros que se tomarán en cuenta para la medición objetiva de la calidad de la señal recibida será la Modulation Error Rate (MER), el BER de Viterbi y el BER de Reed

Solomon, en la pantalla de control del receptor generada en GNU Radio, se muestran las medidas objetivas obtenidas, como se muestra en la figura 69.

Figura 69

Medidas Objetivas en GNU Radio



Nota. La figura muestra la medición de calidad de la señal recibida a una potencia de -75 dBm.

Para cada una de las mediciones realizadas se obtuvo que el TS resultante reproducía el video original, es importante mencionar que según el estándar ISDB-Tb, se tiene que tener como potencia mínima en el receptor -77 dBm y como máximo -20 dBm o valores superiores, el rango en el que se va a realizar las mediciones cumple con estar dentro del rango mencionado (ABNT NBR 15604, 2007).

Los datos obtenidos en las diferentes mediciones se los puede resumir en la tabla 13, estos valores serán utilizados posteriormente para su respectivo análisis.

Tabla 13
Medidas objetivas en el rango de -90 a -60 Dbm

Potencia (Dbm)	MER (dB)	BER Viterbi	VER Reed Solomon
-60	6.74	-7.70	-18.00
-65	6.69	-6.62	-12.76
-70	6.41	-5.43	8.59
-75	6.06	-3.22	-7.72
-80	5.04	-3.13	-5.13
-85	4.12	-1.21	-3.35
-90	3.98	-1.14	-3.12

Nota. En la tabla se muestra un resumen de las mediciones obtenidas para determinar la calidad de la señal recibida.

Con los valores presentados en la tabla 13, se procede con el análisis de los diferentes parámetros de medición objetiva de la calidad de la señal recibida, se presenta a continuación cada parámetro.

Análisis del MER

Los valores obtenidos en la tabla 13, muestra que el valor máximo del MER es de 6.74 dB cuanto se tiene una potencia de recepción de -60 dBm, con este valor se obtuvieron buenos resultados al reproducir el TS generado, tanto en audio como en video, como en el análisis respectivo.

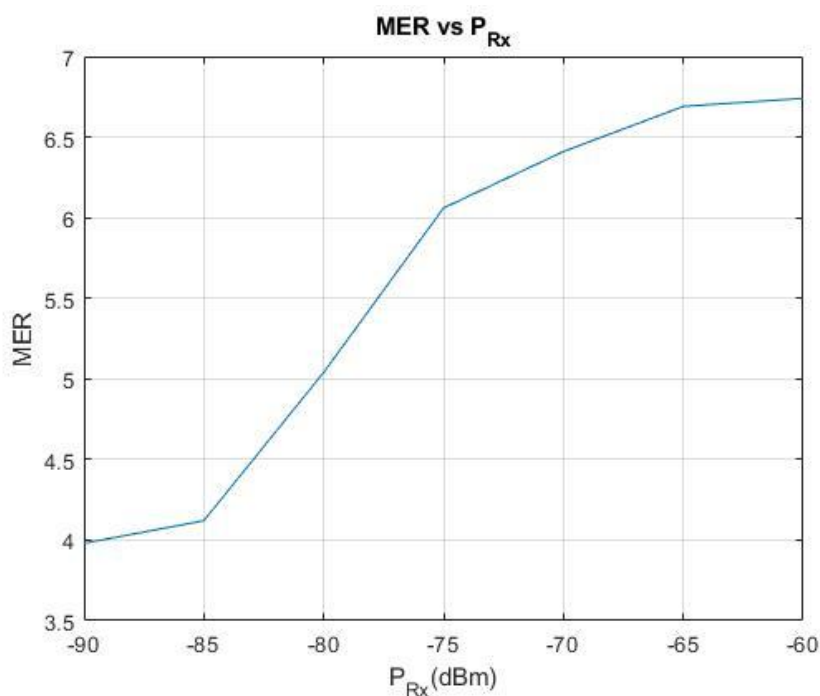
El valor más bajo de MER obtenido es de 3.98 dB este valor se obtuvo con una potencia de recepción de -90 dBm, este valor pese a ser bajo, los resultados que se obtuvieron en la reproducción del TS, son calidad de audio y video aceptables, aunque en el proceso de análisis realizado para la extracción de las aplicaciones interactivas, como para detectar el mensaje de emergencia EWBS, se tuvieron problemas, no

funcionaban adecuadamente debido a que se recibían paquetes vacíos o menores a 188 bytes lo que ocasionaba que no se los pueda analizar.

En la figura 70 se puede ver los datos obtenidos, se puede observar que mientras mayor sea la potencia de recepción mayor será el valor del MER.

Figura 70

Valores del MER obtenido



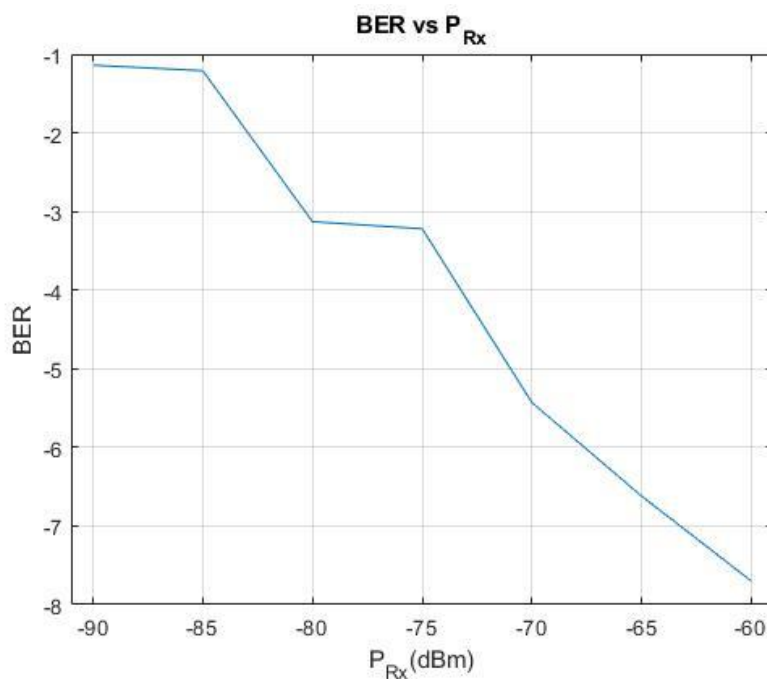
Nota. La figura muestra los valores obtenidos en la medición del MER.

Análisis del BER de Viterbi

En la tabla 13 se puede observar también los valores del BER obtenidos a la salida del bloque que se encarga de la codificación de Viterbi, se puede afirmar que para una potencia de -90 dBm se tiene un valor de -1.14, para una potencia de recepción igual a -60 dBm un valor de -7.70, es claro que mientras mejor potencia de recepción se tiene se tienen que corregir menos errores, en este caso como se usa el

mismo SDR para transmitir y recibir se tiene valores bajos debido a que no se corrige gran cantidad de errores, en la figura 71 se puede observar las mediciones obtenidas.

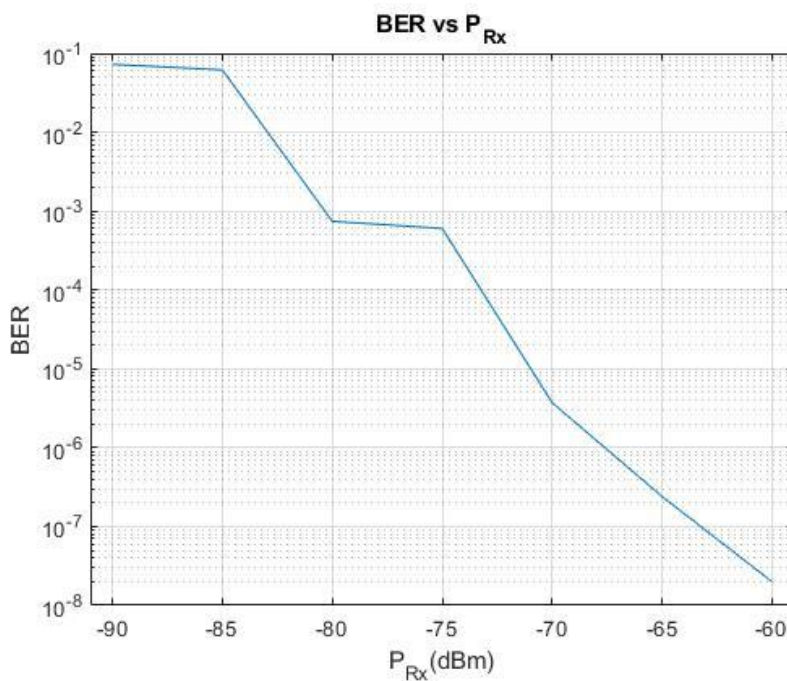
Figura 71
BER de Viterbi



Nota. La figura muestra las mediciones obtenidas para el BER de Viterbi.

Con el fin de realizar un mejor análisis de los datos obtenidos es necesario graficarlos en escala logarítmica para compararlos con los valores que debe tener un receptor de TDT según lo establecido por la UIT. Para sistemas de muy buena calidad el BER debe ser $BER < 10^{-9}$ a 10^{-10} para sistemas buenos un $BER = 10^{-3}$ a 10^{-5} para sistemas de mala calidad el $BER > 10^{-3}$, en la figura 72 se puede ver que hasta el decodificador de Viterbi el sistema implementado alcanza con una potencia de recepción valores de $BER = 10^{-7}$ lo que indica que se recibe una señal de buena calidad (UIT-R, 2015).

Figura 72
BER de Viterbi en escala logarítmica

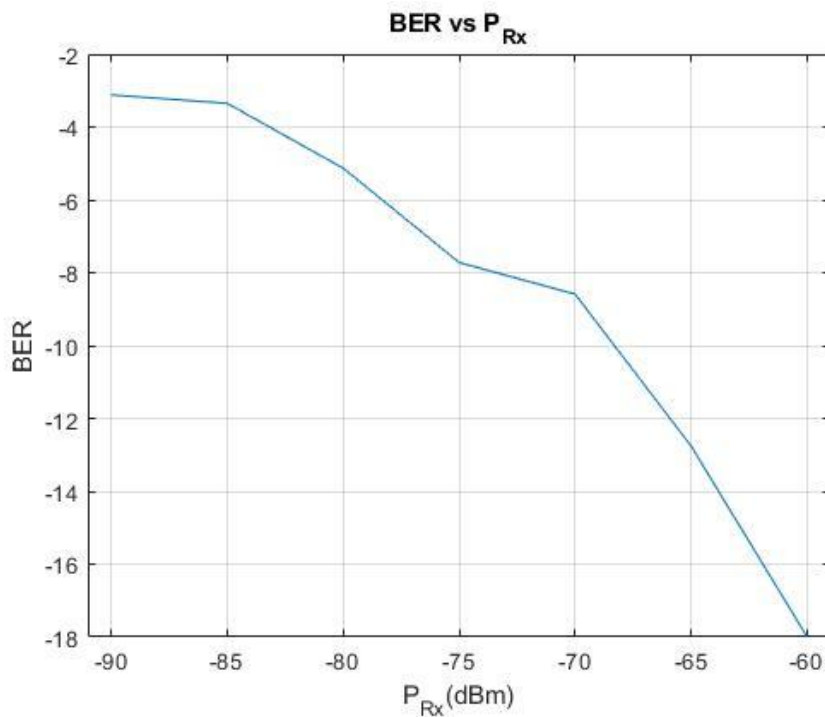


Nota. La figura muestra los resultados del BER en escala logarítmica, obtenidos al variar la potencia de recepción.

Análisis del BER de Reed Solomon

Los datos obtenidos al medir el BER a la salida del decodificador de Reed Solomon se muestran en la tabla 13, se puede observar que mientras se aumenta la potencia, el valor del BER disminuye, es así que para una potencia de recepción igual a -90 dBm se tiene un valor de -3.12 y según se aumenta la potencia de recepción este valor disminuye hasta llegar a una potencia de -60 dBm con un BER de -18, esto se puede observar de mejor forma en la figura 73.

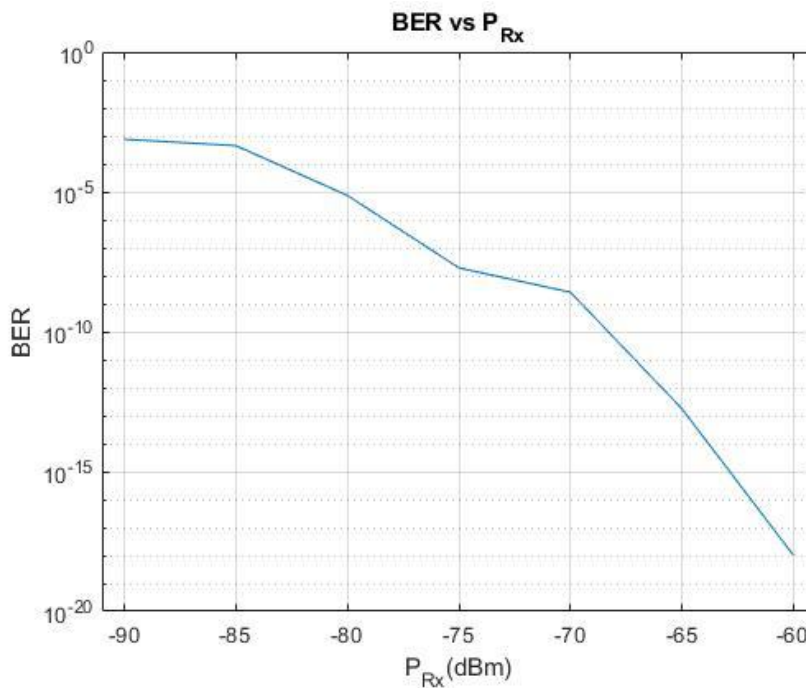
Figura 73
BER de Reed Solomon



Nota. La figura muestra los valores de BER obtenidos a la salida del decodificador de Reed Solomon.

Para poder compararlos con los valores que determina la ITU, igual que el caso anterior se procede a graficar los datos en una escala logarítmica logrando obtener un BER de hasta 10^{-18} , lo que indica que el receptor funciona de forma correcta, a la potencia de -60 dBm mientras que para una potencia de -90 dBm, se tiene un receptor que funciona de forma deficiente, lo que ocasiona que se pueda perder información, en la figura 74 se muestran los datos obtenidos graficados en una escala logarítmica.

Figura 74
BER de Reed Solomon en escala logarítmica



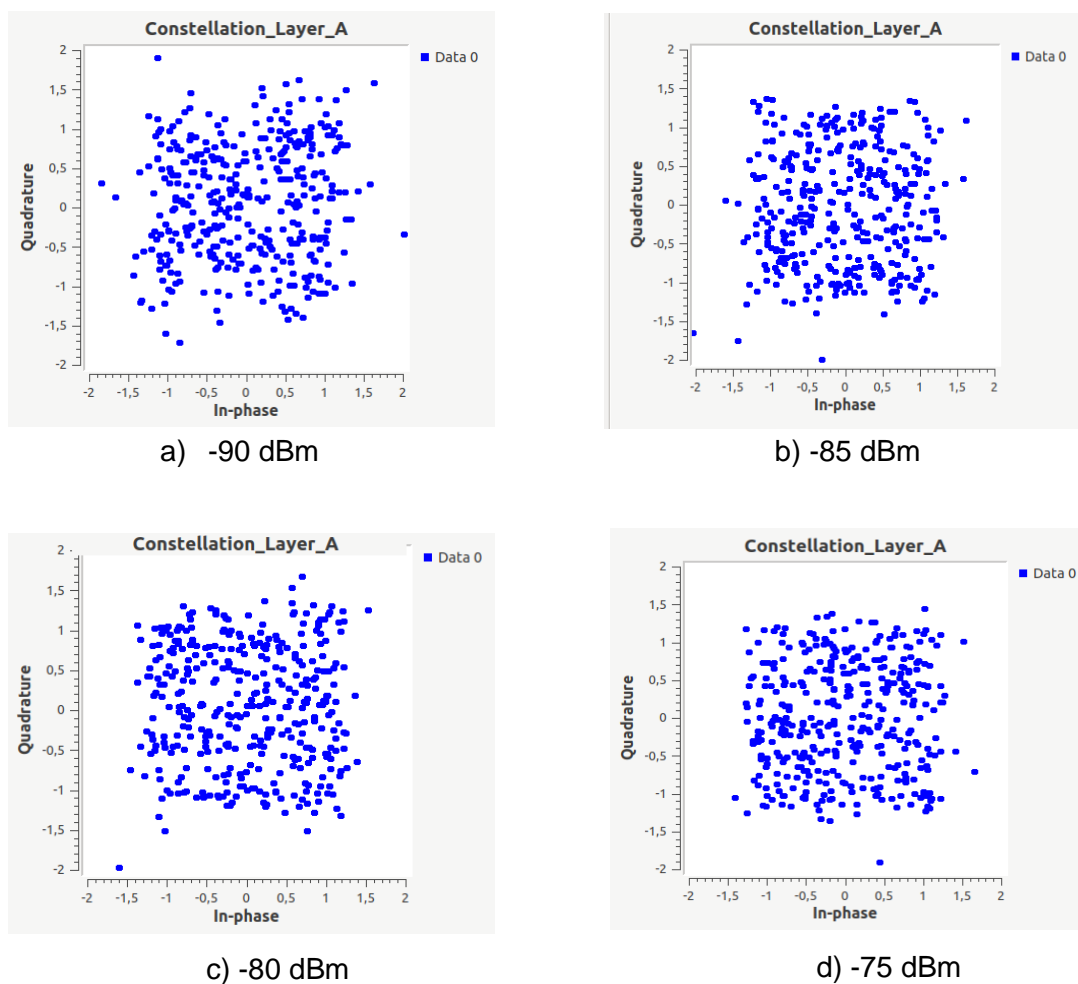
Nota. La figura muestra los datos obtenidos del BER a la salida del decodificador de Reed Solomon.

Análisis de las constelaciones recibidas

Las constelaciones recibidas se muestran en las figuras 75 y 76, en las cuales corresponden a una modulación 64 QAM, con diferentes niveles de potencia en la recepción, se puede ver como existe una mejora en la constelación al aumentar la potencia de recepción, es importante mencionar que al usar el mismo SDR para transmitir y recibir la señal no existen interferencias que puedan afectar de forma significativa la señal transmitida, para una potencia de -90 dBm se tiene una constelación con sus puntos dispersos lo que ocasiona que exista problemas en la recuperación de la señal, mientras que para una potencia de -60 dBm es posible evidenciar la constelación 64 QAM con claridad.

Figura 75

Constelaciones obtenidas en la recepción de la señal con variación de la potencia de recepción en el rango de -90 a -75 dBm

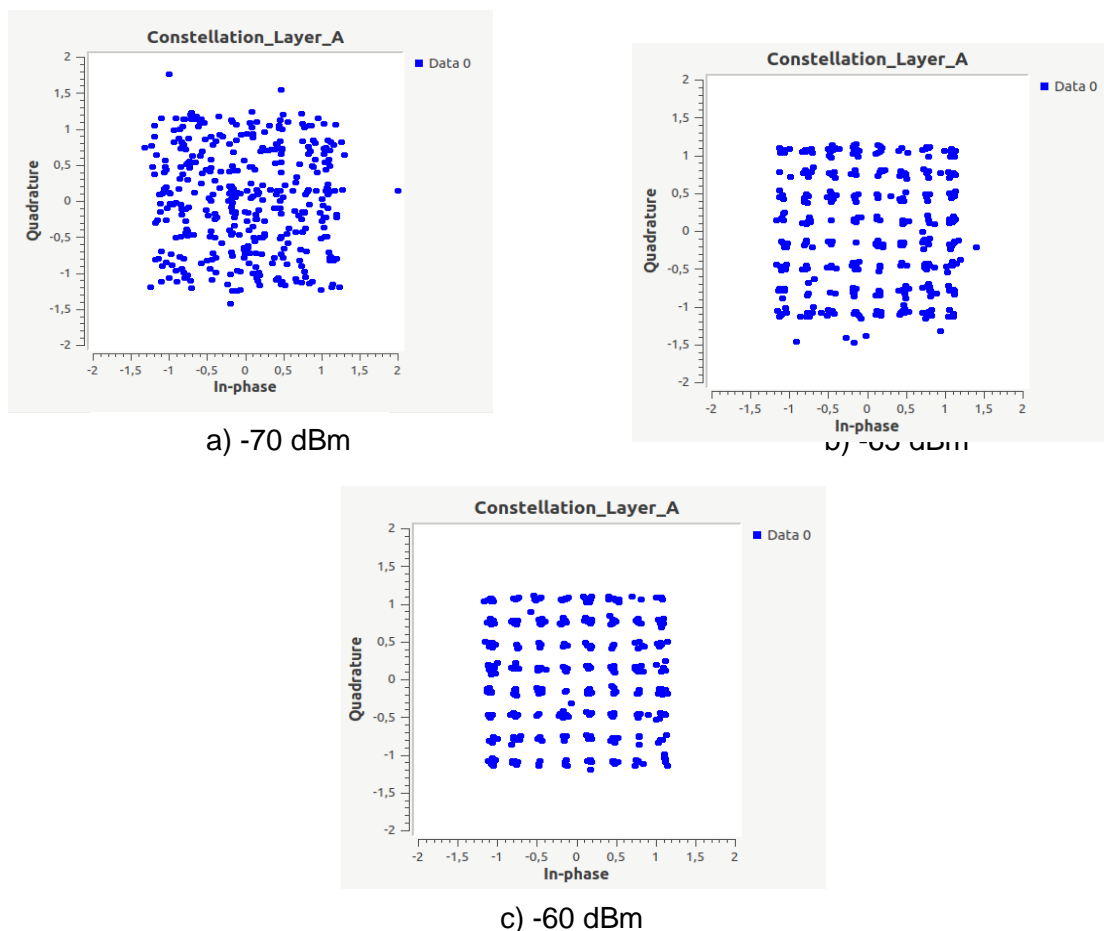


Nota. La figura muestra las constelaciones recibidas en el rango de -90 dBm hasta -75 dBm.

En la figura 75 se puede observar las constelaciones recibidas en el rango de -90 dBm hasta -75 dBm en el receptor, es posible notar como se mejora la constelación recibida según se aumenta la potencia en el receptor, aunque aún no se puede evidenciar claramente que se trata de una modulación 64 QAM.

Figura 76

Constelaciones obtenidas en la recepción de la señal con variación de la potencia de recepción en el rango de -70 a -60 dBm



Nota. La figura muestra las constelaciones obtenidas en el rango de -70 dBm hasta -60 dBm en el receptor.

En la figura 76 se observan las constelaciones obtenidas en el rango de -70 dBm hasta -60 dBm, a diferencia de la figura 75, ya se puede evidenciar que se trata de una modulación 64 QAM, de esta forma es evidente que, si se mejora la potencia de recepción, se perderán menos datos en la transmisión, así como los bloques encargados de corregir errores tendrán un mejor desempeño.

En la figura 77 se puede observar la reproducción de los archivos generados, tanto del que contiene el mensaje EWBS, para el cual en el transmisor se cargó el

archivo *JAPON_EWBS_ESPE.ts*, mientras que para extraer las aplicaciones interactivas se usó el archivo *espetv.ts*.

Figura 77

Reproducción de los archivos recibidos



a) Reproducción archivo mensaje EWBS



b) Reproducción archivo aplicaciones interactivas

Nota. La figura muestra la reproducción de los TS extraídos.

CAPÍTULO V

Conclusiones y Recomendaciones

Conclusiones

En la investigación realizada se pudo comprobar que existen varios proyectos que han trabajado en la implementación de receptores para TDT según el estándar ISDB-Tb, y a los mismos se les ha ido agregando funciones. Al ser accesibles para quien le interese hace que cada vez se puede agregar funcionalidades las cuales ayudan a mejorar su desempeño, el uso de la tecnología SDR ayuda significativamente al desarrollo de este tipo de tecnología, debido a que existe la flexibilidad de agregar bloques con diferentes aplicaciones, con este fin el software GNU Radio facilita enormemente esta actividad, puesto que se puede agregar bloques, modificarlos o utilizar los que vienen incluidos en el programa.

Existen herramientas que facilitan la creación de nuevos módulos en GNU Radio, como se mostró en la investigación fue necesario crear varios bloques que se adaptaban a las necesidades presentadas, se los podía programar en python o en C++, de acuerdo a las necesidades de cada usuario se escoge cualquiera de estos dos lenguajes de programación, para este proyecto se utilizó python por la facilidad que presenta este lenguaje para el manejo de matrices y vectores, es importante no sobrecargar de funciones en un mismo bloque para evitar que el sistema presente fallas en su funcionamiento.

El receptor de TDT que se implementó en GNU Radio funciona correctamente y permite el análisis del TS generado para extraer la información necesaria para la implementación del receptor EWBS y el de aplicaciones interactivas, se tuvieron inconvenientes con la tasa de muestreo del SDR pero se solucionó con la

implementación de un transmisor en la misma plataforma y usando el mismo dispositivo, esto ayuda a solucionar varios inconvenientes, como la sincronización en el envío de información y reducir las interferencias para obtener la señal que permita extraer los paquetes del flujo TS.

El algoritmo implementado para la extracción de la señal EWBS fue realizado en el lenguaje de programación python, se crearon 3 bloques en GNU Radio para evitar sobrecargar de funciones a los mismos, cada bloque se encarga de extraer información importante del flujo TS, el bloque Read EWBS se encarga de analizar las tablas PMT haciendo especial énfasis en las que contienen el descriptor EWBS, del cual se extrae información importante como: los códigos de las áreas en las cuales ocurrió el evento emergente, el bloque Flag_EWBS se encarga de mostrar gráficamente y activando una alarma para indicar que se ha suscitado una emergencia y finalmente el bloque Emergency_Message se encarga de extraer el mensaje de superposición del flujo TS.

Para extraer los archivos que complementan la aplicación interactiva del TS recibido, se tuvo que crear un script fuera del programa GNU Radio, esto debido a que los bloques dentro de este programa se inicializan cada vez que un flujo de datos ingresa, lo que interfiere en el correcto desarrollo del algoritmo creado para extraer los archivos, además para la reproducción de las aplicaciones interactivas es necesario tener todos los archivos que la complementan, para este propósito también se usó el lenguaje python, se crearon varias funciones para el análisis del TS generado, y se logró extraer correctamente los archivos que fueron incluidos en el flujo de información TS.

Las pruebas realizadas para la extracción de la señal EWBS y de los archivos para la aplicación interactiva han sido satisfactorias, el TS se puede analizar correctamente para extraer la información requerida, esto debido a que se usa el mismo

SDR para transmitir y recibir, esto evita que existan errores al momento de extraer la información necesaria, para comprobar esto se tiene la información leída de las tablas PMT así como de los archivos descargados que complementan la aplicación interactiva.

Al realizar el análisis con medidas objetivas de la calidad de audio y video obtenidos al generar el TS, se puede determinar que el receptor funciona de forma óptima a una potencia de -60 dBm, la cual se encuentra dentro del rango que se establece en la recomendación UIT-R BT.1368-12, en los otros valores en los cuales se realizó en análisis de la señal recibida, cumplía con el fin del proyecto ya que permitía recibir y analizar el TS para extraer la información necesaria para recibir el mensaje EWBS así como de extraer los archivos que complementan la aplicación interactiva.

Recomendaciones

El computador que se use para realizar este tipo de proyectos debe tener las características requeridas, debido a que al tratarse del envío y recepción de señales se necesita varios recursos computacionales para que este proceso se lo realice de forma adecuada, adicional se recomienda usar de forma nativa en sistema operativo Linux, puesto que si se lo utiliza a través de una máquina virtual podría presentar fallas mientras se ejecuta el programa para el receptor.

No usar varios bloques simultáneamente debido a que ocasiona que no funcionen correctamente y existan fallas en la generación del TS esto se lo pudo comprobar al usar un bloque que contenía todo el proceso final para la extracción del archivo pero presentaba fallas y al usar únicamente los bloques necesarios se solucionó este inconveniente y se obtuvieron los datos correctos.

Es importante respetar las especificaciones técnicas del SDR para evitar fallas en su funcionamiento, tal como ocurrió en el inicio del proyecto, en el cual era difícil

obtener el flujo TS debido a que se ocupaban frecuencias de muestreo mayores a las indicadas y provocaba que el SDR funcione de forma impredecible, adicional a este inconveniente puede ocurrir que al usar el dispositivo de forma errónea provocar daños en el equipo y dejarlo inutilizable.

Referencias

- 15603-1, A. N. (2007). *Televisión digital terrestre — Multiplexación y servicios de información (SI)*. http://www.telemidia.puc-rio.br/~rafaeldiniz/public_files/normas/SBTVD/es/Multiplexacion/ABNTNBR15603_2D1_2007Esp_2008.pdf
- ABNT NBR 15601. (2007). *Televisión digital terrestre - Sistema de transmisión ISDB-Tb*. http://www.telemidia.puc-rio.br/~rafaeldiniz/public_files/normas/SBTVD/es/Transmicion/15601.pdf
- ABNT NBR 15602-1. (2007). *Televisión digital terrestre-Codificación de video, audio y multiplexación*. http://www.telemidia.puc-rio.br/~rafaeldiniz/public_files/normas/SBTVD/es/Codificacion/ABNTNBR15602_2D3_2007Esp_2008.pdf
- ABNT NBR 15604. (2007). *Televisión digital terrestre - Receptores. Río de Janeiro: Norma Brasileña*. http://www.telemidia.puc-rio.br/~rafaeldiniz/public_files/normas/SBTVD/es/Receptores/ABNTNBR15604_2007Esp_2008.pdf
- Alulema, D. (2012). Desarrollo de aplicaciones para TDT con GINGA-J. *Maskay*, 2(1), 56–63. <https://doi.org/http://dx.doi.org/10.24133/maskay.v2i1.148>
- ARIB. (2005). *Transmission System for Digital Terrestrial Television Broadcasting (Vol. II)*. http://www.arib.or.jp/english/html/overview/doc/6-STD-B31v1_6-E2.pdf
- ARIB. (2006). *Digital Terrestrial Television Broadcasting Specifications for Data Broadcasting Operations (Vol. III)*. https://www.arib.or.jp/english/html/overview/doc/8-TR-B14v2_8-1p3-2-E2.pdf

- ARIB. (2008). *Service Information for Digital Broadcasting System*.
http://www.arib.or.jp/english/html/overview/doc/6-STD-B10v4_6-E2.pdf
- ARIB. (2014). *Transmission system for digital terrestrial television broadcasting*. ARIB STD-B31. http://www.arib.or.jp/english/html/overview/doc/6-STD-B31v2_2-E1.pdf
- Bernal, I., & Valencia, J. (2013). Desarrollo de Aplicaciones Interactivas para TV Digital orientadas a formar a la Población en Desastres Naturales. *Revista Politécnica*, 32. http://revistapolitecnica.epn.edu.ec/ojs2/index.php/revista_politecnica2/article/view/55
- Buenaño, D. (2006). *Analizar una aplicación de alerta temprana para desastres naturales, estimados por el COE utilizando plataforma Village flow con software Ginga, para la zona tres del Ecuador* [Tesis de pregrado, Universidad Nacional de Chimborazo]. <http://dspace.unach.edu.ec/handle/51000/2831>
- Castillo, R. (2021). *Implementación de un sistema de recepción de una señal de emergencia EWBS utilizando la Plataforma de Radio Definida por Software para el estándar ISDB-Tb* [Tesis de pregrado, Universidad de las Fuerzas Armadas]. <http://repositorio.espe.edu.ec/handle/21000/25711>
- Córdova, J. (2017). La gestión preventiva y reactiva de riesgos, para afrontar una posible erupción del Volcán Cotopaxi, por parte del COE del Gobierno Municipal de Rumiñahui. *Revista de Ciencias de Seguridad y Defensa*, 3(4). <http://geo1.espe.edu.ec/wp-content/uploads/2018/01/Art7.pdf>
- Cruz, J., & Rodriguez, Y. (2017). *Desarrollo de aplicaciones para la radio definida por software*. <http://dspace.uclv.edu.cu:8089/handle/123456789/10801>
- De Bruin, R., & Smits, J. (1999). *Digital Video Broadcasting Technology Standards, and*

Regulations. Artech House.

DiBEG. (2008). *Introduction of "ISDB-T"*. <https://www.dibeg.org/index.html>

EcuadorUniversitario. (2012). *Preguntas frecuentes sobre televisión digital*.

<http://ecuadoruniversitario.com/ciencia-y-tecnologia/preguntas-frecuentes-sobre-television-digital/>

Enríquez, P., & Rivadeneira, E. (2018). *Análisis comparativo para el servicio de televisión digital terrestre en Quito* [Tesis de Pregrado, Universidad Politécnica Salesiana]. <http://dspace.ups.edu.ec/handle/123456789/15508>

ETSI. (2003). *Digital Video Broadcasting (DVB); Implementation guidelines for Data Broadcasting*.

https://www.etsi.org/deliver/etsi_tr/101200_101299/101202/01.02.01_60/tr_101202v010201p.pdf

Flores, P. (2016). *La norma ISDB-T y un receptor implementado en SDR* [Tesis de maestría, Universidad de la Republica]. <https://hdl.handle.net/20.500.12008/20168>

Garzón, J. (2018). *Implementación de un receptor One-Seg utilizando la plataforma de Radio definida por software USRP para el estándar ISDB-Tb* [Tesis de pregrado, Universidad de las Fuerzas Armadas].

<http://repositorio.espe.edu.ec/handle/21000/13951>

Gets, R. (2021). *ADALM-PLUTO Overview*. <https://wiki.analog.com/university/tools/pluto>

GINGA. (2022). *Sobre GINGA*. <http://www.ginga.org.br/es/sobre>

GNU Radio. (2021a). *GNU Radio The Free Software & Open Software Radio Ecosystem*. <https://www.gnuradio.org/>

GNU Radio. (2021b). *PlutoSDR Sink*. https://wiki.gnuradio.org/index.php/PlutoSDR_Sink

GNU Radio. (2021c). *PlutoSDR Source*.

https://wiki.gnuradio.org/index.php/PlutoSDR_Source

Grijalva, I. (2022). *Implementación de la capa física de un sistema de transmisión ISDB - T en un dispositivo SDR de bajo costo*. Tesis de Pregrado. Universidad de las Fuerzas Armadas.

IGPN. (2016). *Terremoto 16 de abril de 2016*. <https://www.igepn.edu.ec/eq20160416-informes-noticias>

ISDB-T. (2015). *ISDB-T Documento de Armonización Parte 3: Sistema de Alerta de Emergencia EWBS*. *ISDB-T International Forum*.

https://www.dibeg.org/techp/aribstd/international_harmonization/151130_ISDB-T_Harmonization_Document_EWBS_en.pdf

ISDBT. (2013). *ISDB-T Documento de Armonización Parte 3: Sistema de Alerta de Emergencia EWBS*.

https://cdn.www.gob.pe/uploads/document/file/470102/EWBS_armonizacion.pdf

ISO/IEC. (1998). *Information technology - Generic coding of moving pictures and associated audio information*. http://www.telemedia.puc-rio.br/~rafaeldiniz/public_files/normas/ISO-13818/iso13818-6/info_isoiec13818-6.pdf

ISO/IEC 13818-1. (2021). *Information technology — Generic coding of moving pictures and associated audio information — Part 1: Systems*.

<https://www.iso.org/standard/83239.html>

JICA. (2016). *Japan International Cooperation Agency*.

<https://www.jica.go.jp/peru/espanol/office/topics/160115.html>

Llanos, A. (2020). *Sistemas de Alerta Temprana empleando Radiodifusión*.

<http://repositorio.inictel-uni.edu.pe:8080/xmlui/handle/123456789/35>

Llanos, A., Chavez, R., Zaga, A., Quenaya, Z., Jara, M., & Atacuri, D. (2017, junio).

Automatic novel system for sending emergency alert and alarm for multi events through radio broadcasting transmission system in Perú.

<https://ieeexplore.ieee.org/abstract/document/7986155>

MINTEL. (2017). *TDT - Televisión Digital Terrestre*. <https://tdtecuador.mintel.gob.ec/>

Morales, A. (2010). *Diseño de la red para interactividad en Televisión Digital terrestre e*

IPTV en el campus ESPE Sangolquí [Tesis de pregrado, Escuela Politécnica del

Ejército]. <http://repositorio.espe.edu.ec/bitstream/21000/2645/1/T-ESPE-029824.pdf>

Núñez, E. (2016). *Estudio, análisis e implementación de un software para construir un*

extractor y constructor de carrusel de datos incluido en el flujo de transporte mpeg-2 ts de televisión digital [Tesis de pregrado, Universidad de las Fuerzas Armadas].

<http://repositorio.espe.edu.ec/handle/21000/12244>

Olmedo, G. (2020). *Televisión Digital Terrestre*. Universidad de las Fuerzas Armadas ESPE.

Olmedo, G., Acosta, F., Villamari, D., Santander, F., Achig, R., & Morocho, V. (2020).

Prototype of a centralized alert and emergency system for digital terrestrial

television in ecuador. *XV Multidisciplinary International Congress on Science and*

Technology, 191–201. [https://link.springer.com/chapter/10.1007/978-3-030-68080-](https://link.springer.com/chapter/10.1007/978-3-030-68080-0_14)

[0_14](https://link.springer.com/chapter/10.1007/978-3-030-68080-0_14)

Olmedo, G., Sambrano, Y., Acosta, F., & Paredes, N. (2020). Design and

Implementation of a EWBS Gateway over an IP Telephone Network. *Iberoamerican Conference on Applications and Usability of Interactive TV*, 31–44.

https://link.springer.com/chapter/10.1007/978-3-030-81996-5_3

Pillajo, C., Ochoa, J., Acosta, F., & Olmedo, G. (2016). Herramienta multiplataforma para generación automática de aplicaciones interactivas Ginga-NCL basado en plantillas. *Maskay*, 6(1), 8–12.

[http://scielo.senescyt.gob.ec/scielo.php?script=sci_arttext&pid=S1390-](http://scielo.senescyt.gob.ec/scielo.php?script=sci_arttext&pid=S1390-67122016000100008)

[67122016000100008](http://scielo.senescyt.gob.ec/scielo.php?script=sci_arttext&pid=S1390-67122016000100008)

Poveda, G., Bazurto, M., Mayorga, J., Ortiz, S., Quijije, A., & Velasco, P. (2017, marzo).

Incidencia Japonesa De La Television Digital Terrestre En El Ecuador. *Revista Observatorio Iberoamericano de la Economía y la Sociedad del Japón*.

<https://www.eumed.net/rev/japon/29/television.zip>

Pozo, M., Moncayo, T., Bernal, I., & Mejia, D. (2015). Desarrollo de un Analizador de Flujos Únicos de Paquetes de Transporte TS Conformados de Acuerdo a la Norma de TV Digital ISDB-Tb. *Revista Politécnica*, 35(1), 59–59.

http://revistapolitecnica.epn.edu.ec/ojs2/index.php/revista_politecnica2/article/view/400

PROMAX. (2017). *EWBS: Detección de alarmas por terremoto*.

<https://www.promax.es/esp/noticias/511/ewbs-deteccion-de-alarmas-por-terremoto/>

Recalde, C. (2016). *Análisis y generación del flujo de transporte con sobreimposición de texto para alerta temprana en servicios de televisión digital terrestre* [Tesis de Pregrado. Universidad de las Fuerzas Armadas ESPE].

<http://repositorio.espe.edu.ec/handle/21000/12248>

Shogen, K. (2009). *HANDBOOK ON EMERGENCY WARNING BROADCASTING*

SYSTEMS. ABU. <https://docplayer.net/58475330-Handbook-on-emergency-warning-broadcasting-systems.html>

Silva, M. (2016). *Diseño y construcción de un receptor de señales de emergencia EWBS según el estándar ISDB-TB* [Tesis de Pregado. Universidad Nacional de Ingeniería].
https://alicia.concytec.gob.pe/vufind/Record/UUNI_b968a3f0441cb1c2102c26d0f53a0f8d/Description#tabnav

Tomodachi. (2016). *Japoneses que contribuyen en otras partes del mundo*. 8–9.
https://www.japan.go.jp/tomodachi/_userdata/pdf/2016/winter2016es/08_09es.pdf

Torres, W., Muñoz, K., Sampedro, M., Mendoza, M., Rodas, A., Almeida, D., & Almeida, M. (2018). Análisis distribucional de riesgo de los atractivos turísticos, por el lahar sur del volcán Cotopaxi. In Congreso de Ciencia y Tecnología ESPE. *Congreso de Ciencia y Tecnología ESPE*, 13(1).
<https://doi.org/http://dx.doi.org/10.24133/cctespe.v13i1.698>

UIT-R. (2015). *Criterios para la planificación, incluidas las relaciones de protección, de los servicios de televisión digital terrenal en las bandas de ondas métricas/decimétricas*. https://www.itu.int/dms_pubrec/itu-r/rec/bt/R-REC-BT.1368-12-201502-S!!PDF-S.pdf

Valles, F., Ramirez, J., Moreno, F., Ramirez, J., Veloz, D., & Martinez, G. (2019). Planificación de Frecuencias para Televisión Digital Terrestre (TDT) en el Ecuador. *Revista Perspectivas*, 1(1), 25–35.
<https://doi.org/https://doi.org/10.47187/perspectivas.vol1iss1.pp23-35.2019>

Villamari, D., Olmedo, G., Lara, R., & Illescas, M. (2012). Generación de Transport Stream con Audio, Video y Datos de Interactividad para el Sistema de Televisión

Digital Terrestre ISDB-Tb. *Maskay*, 2(1), 49–55.

<https://doi.org/https://doi.org/10.24133/maskay.v2i1.147>

Apéndices