

# Trabajo Fin de Grado

## Ingeniería de las Tecnologías de Telecomunicación

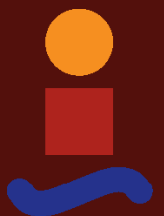
### Implementación de Software Defined Radio en sistemas de comunicaciones actuales

Autor: Álvaro Gutiérrez Rivera

Tutor: Juan Antonio Becerra González

**Dep. Teoría de la Señal y Comunicaciones**  
**Escuela Técnica Superior de Ingeniería**  
**Universidad de Sevilla**

Sevilla, 2021





Trabajo Fin de Grado  
Ingeniería de las Tecnologías de Telecomunicación

# **Implementación de Software Defined Radio en sistemas de comunicaciones actuales**

Autor:

Álvaro Gutiérrez Rivera

Tutor:

Juan Antonio Becerra González

Profesor Ayudante Doctor

Dep. Teoría de la Señal y Comunicaciones  
Escuela Técnica Superior de Ingeniería  
Universidad de Sevilla

Sevilla, 2021



Trabajo Fin de Grado: Implementación de Software Defined Radio en sistemas de comunicaciones actuales

Autor: Álvaro Gutiérrez Rivera

Tutor: Juan Antonio Becerra González

El tribunal nombrado para juzgar el trabajo arriba indicado, compuesto por los siguientes profesores:

Presidente:

Vocal/es:

Secretario:

acuerdan otorgarle la calificación de:

El Secretario del Tribunal

Fecha:



# Agradecimientos

---

Quisiera empezar los agradecimientos con especial mención a mi tutor, Juan Antonio Becerra González. Gracias, principalmente, por ofrecerme la oportunidad de realizar este proyecto; pero también por ayudar, aconsejar y tener esa paciencia y vocación en tu profesión que te caracteriza, animando y motivando a cualquier estudiante, entre los que me incluyo.

Agradecer tanto a mis amistades de siempre, como a todas y cada una de las que he sumado durante todo el grado, por compartir esos días de agobios que parecían interminables, por ofrecer ese apoyo necesario cuando todo eran malas noticias, y sobre todo, por compartir la alegría y satisfacción de poder decir, tarde o temprano: soy ingeniero. Todos habéis sido igual de importantes, pero, en especial, dar las gracias a Carmelo y Omar, pues me habéis ayudado a construir los mejores recuerdos que guardaré de esta etapa.

Por último, y no menos importante, a mi familia, mis padres. Por todo el esfuerzo que han hecho durante todo el tiempo para que pudiera estudiar lo que quisiera sin dejarlo en ningún momento. Todo enfado, agobio y celebración han sido importante para que haya podido llegar hasta aquí.

Muchas gracias a todos.

*Álvaro Gutiérrez Rivera*  
*El Puerto de Santa María (Cádiz), 2021*





# Resumen

---

El origen del término SDR (*Software Defined Radio*, o *Radio Definida por Software en español*) se remonta a las décadas de 1970 y 1980. Surge a partir del esfuerzo conjunto de varios grupos de investigación, entre los que destacan el Departamento de Defensa del gobierno estadounidense y un equipo de la División de E-Systems Inc.

Fue en 1991, cuando Joseph Mitola introdujo el término de software radio (*SR*, o *Radio de Software*), en cooperación con el equipo de E-Systems, cuyo objetivo era la construcción de un transceptor GSM basado en software (Mitola, Telesystems Conference, 1992). La plataforma SR procesa casi todos los algoritmos del transceptor como un software para un procesador.

En la actualidad, la necesidad de adoptar los principios de SDR son cada vez más y más visibles, debido a desarrollos recientes en radios multimodo y multiestándar, y las diversas aplicaciones complejas que están presentes. La flexibilidad que presenta SDR es ideal para varios requisitos de calidad de servicio (*QoS*, *Quality-of-Service*) exigidos por numerosos datos, voz y aplicaciones multimedia. Hoy en día, muchos diseños de estaciones base, emplean arquitectura SDR o, al menos, alguna tecnología basada en principios de SDR. Por otro lado, a un ritmo más lento, los proveedores de chips están adoptando los principios de diseño propios de la radio definida por software, destinados a pequeños dispositivos como teléfonos móviles y portátiles.

El objetivo de este proyecto es proporcionar información acerca de cómo funciona esta tecnología, además de profundizar en todos sus aspectos tanto a nivel físico, o de componentes, como a nivel de aplicación. Para ello se utilizarán resultados obtenidos con el kit RTL-SDR de Nooelec<sup>®</sup>. Por último, se realizará un análisis de los resultados obtenidos, incluyendo la indicación de posibles proyectos futuros.



# Abstract

---

The SDR (*Software Defined Radio*) concept has its origin in the decades of seventies and eighties. It was originally developed by the combined efforts of various research groups, in which are the US Department of Defense Laboratory and a team of E-Systems Inc.

In 1991, Joe Mitola introduced the term '*Software Radio*' (SR) in cooperation with E-Systems, which objective was the build of a software based GSM transceiver (Mitola, Telesystems Conference, 1992). The SR platform processes almost all the transceiver algorithms as software for a processor.

Nowadays, the need to adopt SDR principles is becoming more and more apparent due to recent developments in multi-mode multi-standard radios and the various complex applications that govern them. The flexibility of SDR is ideally appropriate for the various quality-of-service (QoS) requirements ordered by the numerous data, voice, and multimedia applications. Today, many base-station designs make use of SDR architecture or, at least, some technology based on SDR principles. On the other side, slowly, chipset providers are adopting SDR principles in the design of multi-mode multi-standard radios destined for cellphones and laptops, or others slight devices.

The aim of this project is to provide information about how this technology works, besides to go deeper in the physical level or component level, and application level. To do this, it will be used the results obtained with the RTL-SDR kit from the *Nooelec* company. Finally, an analysis will be carried out with the obtained results, including some possible future projects indications.



# Índice

---

<i>Resumen</i>	III
<i>Abstract</i>	V
<b>1 Introducción</b>	<b>1</b>
1.1 Objetivo	2
1.2 Estructura	2
<b>2 Radio Definida por Software</b>	<b>3</b>
2.1 SDR	3
2.1.1 Arquitectura y funcionamiento	3
2.1.2 Aspectos digitales de SDR	6
2.1.3 Mercado actual de SDR	7
Tecnologías de Internet inalámbrica y telefonía celular	7
Industria aeroespacial	7
Industria automotriz	8
Ámbito académico	8
2.2 Sistemas SDR implementados en el mercado	8
2.2.1 Dongle USB RTL-SDR	8
2.2.2 FUNcube Dongle Pro+	9
2.2.3 HackRF One	9
2.2.4 Mini Airspy	9
2.2.5 Nooelec NESDR SMArTee	9
<b>3 Materiales y Métodos</b>	<b>11</b>
3.1 Nooelec NESDR SMArTee v2	11
3.1.1 Hardware y arquitectura	11
Rafael Micro <sup>®</sup> R820T/2 radio tuner	12
Realtek RTL2832U	12
Diagrama de bloques completo	14
3.1.2 Equipamiento y aplicaciones	16
Funciones de escáner	16
Funciones de escucha	16
Funciones de seguimiento	16
Funciones de decodificación	16
Otras funciones	16

---

<b>4</b>	<b>Diseño de Experimentos</b>	<b>19</b>
4.1	Equipo de trabajo	19
4.1.1	Verificación del entorno de trabajo	19
4.2	Primer contacto con NESDR SMArTee	21
4.3	Modelo de receptor para la visualización del espectro	24
4.4	Búsqueda de estaciones de radio FM	28
4.5	Análisis de señales móviles GSM (2G)	30
4.6	Análisis de señales móviles UMTS (3G)	33
4.7	Análisis de señales móviles LTE (4G)	34
4.8	Búsqueda de señales de TV digital DVB-T	37
4.9	Buscando la actividad de un teléfono móvil en el espectro	38
4.10	Barrido del espectro	43
4.11	Decodificador y receptor de radio FM estéreo	46
<b>5</b>	<b>Conclusiones</b>	<b>53</b>
	<b>Apéndice A Códigos de Matlab</b>	<b>55</b>
	<i>Índice de Figuras</i>	63
	<i>Índice de Tablas</i>	65
	<i>Índice de Códigos</i>	67
	<i>Bibliografía</i>	69

# 1 Introducción

---

*Sólo somos una raza avanzada de primates en un planeta menor de una estrella promedio. Pero podemos entender el Universo. Eso nos hace muy especiales.*

STEPHEN HAWKING, 1988

Hoy en día y desde hace unos años, la sociedad se encuentra en pleno auge tecnológico, donde la ciencia y tecnología tienen como objetivo mejorar nuestra calidad de vida, a través de la importante tendencia de reducir el tamaño de componentes electrónicos y mejorar la simplicidad en las conexiones. El ejemplo más claro lo podemos encontrar en nuestro día a día, pues sólo hace falta echar la vista atrás un par de décadas y recordar cómo eran las dimensiones de los teléfonos móviles, televisores y ordenadores. Si retrocedemos algo menos, se observa también cómo se han simplificado las conexiones cableadas entre dispositivos, ya sea en el hogar, una oficina, etc.

Al igual que con los dispositivos más cotidianos que tenemos a nuestro alcance, esta tendencia también se observa en el sector industrial y militar. Sectores que están directamente relacionados con la principal tecnología que abarca nuestras vidas: la radiocomunicación. En los inicios, los equipos transmisores y receptores de radiocomunicaciones eran equipos construidos con componentes electrónicos analógicos en todas sus etapas (de radiofrecuencia, frecuencia intermedia, etc).

En los años ochenta, la incorporación de microprocesadores supuso un gran avance, puesto que permitía controlar funciones internas además de aportar nuevas prestaciones como la posibilidad de controlar los equipos de radio a través de un ordenador.

En los años noventa, se desarrollaron los primeros chips DSP (*Digital Signal Processor* o Procesadores Digitales de Señal en español), que permitían realizar filtrados pasobajos y de superposición de ruido mediante técnicas digitales.

Pero no fue hasta mediados de los años noventa, cuando Joseph Mitola empezó a investigar y desarrollar un nuevo concepto de equipos de radiocomunicación que empezaría a suplir a las existentes emisoras hardware. Surge el concepto de SDR (*Software Defined Radio*), naciendo los primeros equipos de radio definidos por software. Un claro ejemplo de simplicidad y ahorro en componentes electrónicos y conexiones.

SDR es también responsable del nacimiento de la *Radio Cognitiva* unos años más tarde. La Radio Cognitiva es capaz de hacer que tanto transmisor y receptor se pongan de acuerdo en elegir la mejor frecuencia a la cual pueden trabajar de manera automática. Consiguiendo, de manera inmediata, una mejora del aprovechamiento del ancho de banda, lo que da lugar a múltiples aplicaciones como se verá más adelante.

## 1.1 Objetivo

El objetivo de este documento es presentar de una manera teórico-práctica la tecnología SDR. Esta es relativamente nueva, dado que presenta numerosos campos de aplicación aún no explotados, donde permitiría un notable ahorro económicamente hablando. Además pone al alcance de cualquier usuario la Radio Definida por Software. Todo ello se trata de poner de manifiesto con el contenido de este proyecto.

## 1.2 Estructura

El presente documento está compuesto por los siguientes capítulos:

- **Introducción:** Es el capítulo actual, donde se muestra una breve información acerca del proyecto, junto a su motivación y estructura.
- **Radio Definida por Software:** Para dar una consistencia teórica al documento, este capítulo proporciona una explicación sobre la tecnología SDR, empezando por su definición, siguiendo por su arquitectura y el mercado actual y finalizando por sistemas donde se implementa esta tecnología.
- **Materiales y métodos:** En este tercer capítulo se presentará el dispositivo elegido para implementar un sistema SDR, comentando y profundizando en sus componentes, arquitectura y funcionalidades.
- **Diseño de experimentos:** Parte experimental del proyecto donde se llevarán a cabo diversas tareas con el dispositivo y el software elegido, que forman el sistema SDR montado.
- **Conclusiones:** Capítulo que se centrará en una opinión final acerca de los experimentos realizados y una reflexión de cara a un futuro cercano, tomando como base la teoría de esta tecnología.
- **Apéndice A. Códigos de Matlab:** Contiene los códigos utilizados para la implementación de algunos experimentos.



## 2 Radio Definida por Software

---

Llegados a este punto, sería conveniente aclarar que, *Software Defined Radio* y *Software Radio* son dos términos usados indistintamente en la literatura técnica. Esto se debe, principalmente, a las muchas perspectivas diferentes que tomaba la tecnología de radio software a medida que evolucionaba. El factor que realmente las diferencia es cómo de cerca están las funciones de digitalización de la antena al sistema radio.

En una radio software, el proceso de digitalización ocurre muy cerca de la antena y los demás procesos radio se realizan en el software. Por otro lado, en una radio definida por software, la digitalización ocurre más lejos de la antena. A medida que la tecnología mejore con el tiempo, la delgada línea que los diferencia irá desapareciendo. Es por ello que en las próximas páginas se asumirá estas diferencias como inexistentes, haciendo uso exclusivamente del término SDR. Además, se profundizará a nivel físico del principal componente que complementa con la radio definida por software.

### 2.1 SDR

A día de hoy, la radio definida por software es una tecnología de comunicación radio, pudiendo actuar tanto de transmisor como receptor. No obstante, a continuación se prestan dos definiciones que dejan perfectamente definido este concepto:

**Definición 2.1.1** *"Una radio que incluye un transmisor en el que los parámetros de funcionamiento del rango de frecuencia, el tipo de modulación o la potencia máxima de salida (radiada o conducida) pueden ser modificados por software sin hacer ningún cambio en componentes hardware que afectan a las emisiones de radiofrecuencia."*

(Comisión Federal de Comunicaciones, 2001)

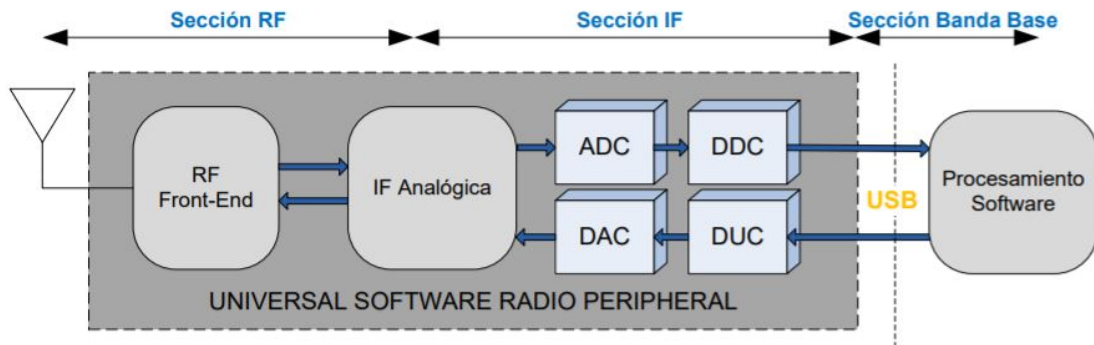
**Definición 2.1.2** *"Una radio definida por software es una colección de tecnologías software y hardware que permiten arquitecturas de sistemas reconfigurables para redes inalámbricas y usuarios terminales. Además, proporciona una eficiente y relativamente barata solución a los problemas de construcción de dispositivos inalámbricos multi-modo, multi-banda y multi-funcional que pueden ser mejorados usando actualizaciones softwares."*

(Foro SDR, 2005)

#### 2.1.1 Arquitectura y funcionamiento

Desde el punto de vista de la arquitectura, una radio definida por software es una forma de transceptor en el que idealmente todos los aspectos de su funcionamiento se determinan usando hardware versátil

de uso general, cuya configuración está bajo el control del software. Por otro lado, a pesar de que SDR haya ido evolucionando con los años, se sigue basando en el mismo esquema básico que se muestra en la Figura 2.1



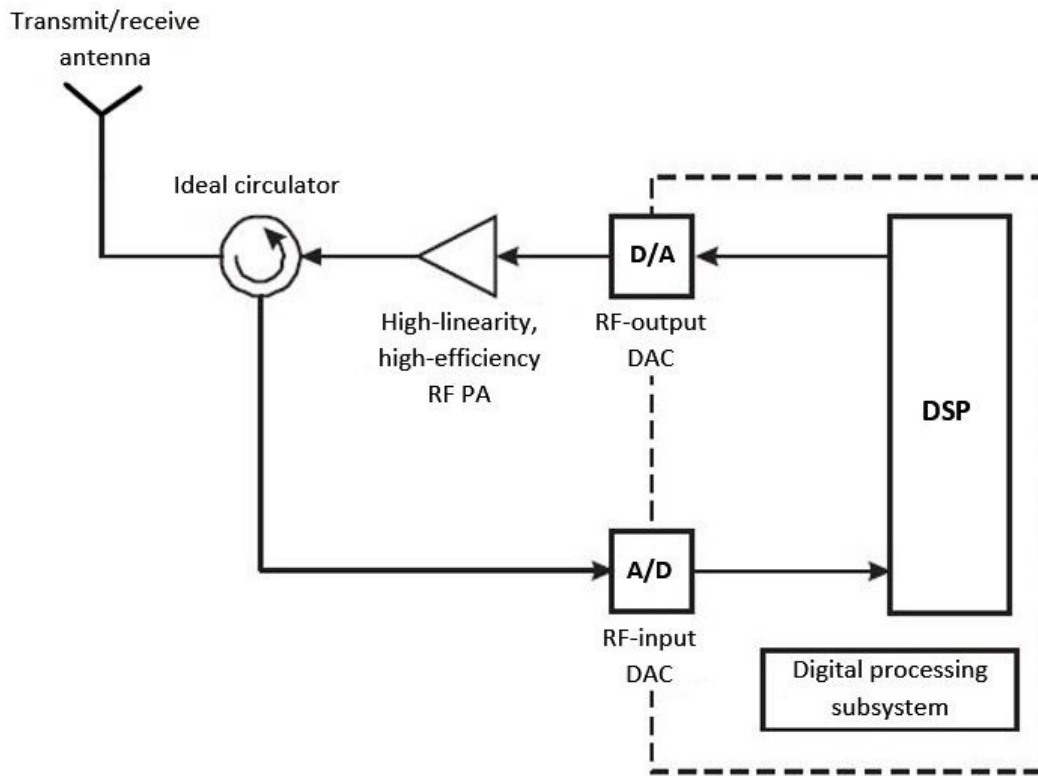
**Figura 2.1** Diagrama de bloques funcionales genérico de un dispositivo SDR.

Quedan claramente diferenciadas tres etapas funcionales: sección de RF, sección de IF y sección Banda Base. De estas tres secciones, las dos primeras se implementan en hardware, como se verá más adelante, y la tercera es la implementada en software.

Debido al comportamiento de algunos dispositivos que se basan en esta estructura, se divide la descripción del funcionamiento de las etapas dependiendo de si se trata de una recepción o transmisión de señales.

- Funcionamiento en transmisión:
  - Sección de Banda Base: es la encargada del procesamiento software de la señal, como la modulación, el análisis espectral, filtrado de señal, etc.
  - Sección IF: en esta etapa se convierte a frecuencia intermedia (IF, *Intermediate Frequency*, que generalmente será cero, es decir, del tipo *Zero-IF*), las señales de banda base. El responsable de estas conversiones es el módulo **DUC** (*Digital Up Converter*), que está acompañado de un **DAC** (*Digital to Analog Converter*) para poder convertir la señal a analógica para su envío.
  - Sección RF: conocida también como *RF Front-End*, es la responsable de transmitir las señales RF. Para ello, antes, amplifica y modula las señales de IF que recibe, ya que pueden ser de 0 Hz, como se ha mencionado.
- Funcionamiento en recepción:
  - Sección RF: en este caso es la encargada de recibir las señales y adecuarlas para la siguiente sección, convirtiéndolas de RF a IF.
  - Sección IF: recibe la señal IF de la sección anterior y la convierte a banda base gracias al módulo **DDC** (*Digital Down Converter*), encargado de disminuir la tasa de muestreo de la señal. A continuación se digitaliza la señal analógica recibida mediante un convertor **ADC** (*Analog to Digital Converter*).
  - Sección de Banda Base: por último, la señal banda base recibida se procesa mediante software, en el que se implementa el filtrado, la demodulación o el análisis espectral correspondientes.

Una vez identificadas las secciones que presenta cualquier modelo básico de SDR, estaría bien profundizar un nivel más para explicar con detalle cada bloque. Para ello, se usará como referencia el modelo ideal de una radio definida por software, mostrada en la Figura 2.2.



**Figura 2.2** Arquitectura de una radio definida por software ideal.

Hay que tener presente que se parte de la suposición de que el convertidor analógico-digital tiene un filtro anti-aliasing incorporado, y que el convertidor digital-analógico tiene otro filtro de reconstrucción.

Este esquema ideal tiene las siguientes características:

- La modulación, la canalización, los protocolos y la ecualización para la transmisión y recepción están determinadas en el software, dentro de un subsistema de procesamiento digital. Esto se refleja en la Figura 2.2 como el bloque DSP.
- El circulator ideal se usa para separar los caminos de las señales transmitidas y recibidas. Este componente, debido a que se basa en una coincidencia perfecta entre él mismo y las impedancias de la antena y amplificador, no es implementable en la práctica. Además, el circulator tendría que ser de banda ancha también, lo que no ocurre con la mayoría de los diseños actuales. Debido a las restricciones habituales de frecuencia impuestas a esta función, se usan soluciones basadas en filtros, por ejemplo, un diplexor convencional.
- El amplificador de potencia lineal asegura una transferencia ideal de la modulación de RF desde el convertidor digital-analógico hasta una señal de alta potencia preparada para la transmisión, con bajas emisiones de canales adyacentes.

En las Figuras 2.3-(a) y 2.3-(b) se pueden apreciar una arquitectura de un caso más real, como lo es un sistema transceptor de estación base (*BTS, Base-Station Transceiver System*).

Este tipo de radio se supone que es de banda ancha (multibanda o multifrecuencia). Esta suposición va implícita, ya que se abarcan múltiples aplicaciones que se dan en gran parte del espectro, como se verá más adelante.

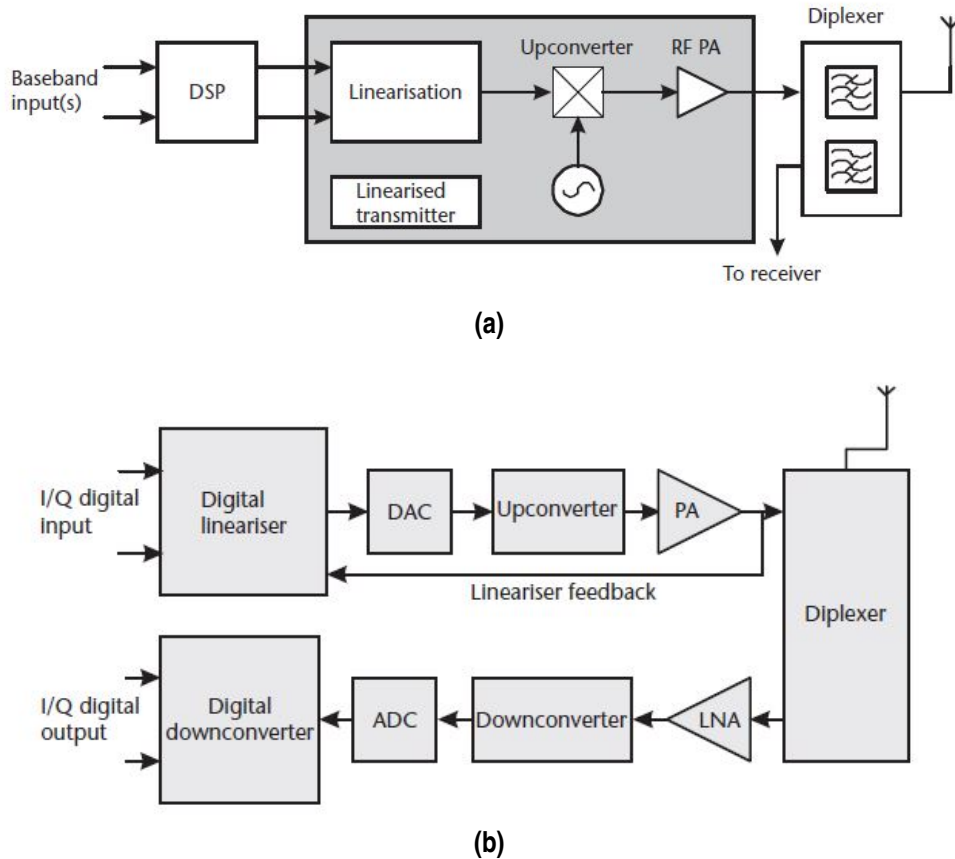


Figura 2.3 Arquitectura empleada en una BTS con SDR.

### 2.1.2 Aspectos digitales de SDR

Existe un rango de soluciones para la parte software de SDR encargada del procesamiento digital, cada una con sus propias características y áreas de aplicación.

#### DSPs

Podría decirse que los procesadores digitales de señal fueron la tecnología que originalmente habilitó a SDR. Presentan la ventaja de una flexibilidad completa y una amplia variedad de aplicaciones, siendo un dispositivo imprescindible para las comunicaciones. También se adaptan mejor a las formas de procesamiento de señales menos intensivas, desde un punto de vista computacional.

#### FPGAs

Las matrices de puertas programables (*FPGAs*, *Field-Programmable Gate Arrays*) han experimentado una revolución en los últimos años, tanto en rendimiento como en costo. También es posible convertir un FPGA en un *cuasi-ASIC*, con alto grado de confianza.

Además, ofrecen mucha más flexibilidad que la que se puede construir de manera rentable en un ASIC, por lo que se ajusta muy bien a los requisitos de SDR.

#### ASICs

El principal inconveniente de circuitos integrados de aplicaciones específicas (*ASICs*, *Application-Specific Integrated Circuits*) dentro de un sistema SDR, radica en su falta de flexibilidad, o, dicho de otra forma, en el coste adicional para añadir esa flexibilidad.

A continuación, a modo resumen y para ofrecer una visión global al lector, se ofrece la siguiente Tabla 2.1.

**Tabla 2.1** Comparación de componentes para el procesamiento digital de señal.

	ASIC	FPGA	DSP
Consumo de energía	++	–	o
Costo por grandes cantidades	++	–	o
Potencia de cálculo	++	+	o
Costo por pequeñas cantidades	--	–	o
Flexibilidad	--	+	++
Reutilización de programas	o	o	+

Los dos problemas más importantes a día de hoy son el consumo de energía y el costo de las distintas opciones, lo que, según la aplicación, servirá para optar por una tecnología u otra. Son quizás el principal inhibidor del uso generalizado de SDR en teléfonos móviles y dispositivos portátiles, no obstante, existen casos en los que estos problemas son de causa menor.

### 2.1.3 Mercado actual de SDR

Hoy en día, SDR se encuentra en sus primeras fases de implementación. El despliegue y aplicación de esta tecnología está al alcance de cualquier usuario, lo que conlleva a un aumento exponencial de su uso en distintos sectores que se mencionan a continuación. Se trata de sectores donde el coste deducido por el uso de SDR se puede ver compensado por las ventajas que ofrece.

#### Tecnologías de Internet inalámbrica y telefonía celular

Es el sector donde el concepto de SDR ha influido con más fuerza a la hora de hacer la segmentación de redes celulares y de acceso inalámbrico de banda ancha. Esta tecnología permite la elección de la forma de onda deseada para conseguir la operación simultánea de varios sistemas. Tecnologías como *WiFi* y *WiMAX* están desarrollando iniciativas donde incorporan SDR junto a la técnica de transmisión OFDM para conseguir esta simultaneidad en terminales móviles.

En redes celulares, la aplicación de SDR, posibilita el aumento de la vida útil de las estaciones base y evita la congestión del espectro. Cuando se hacen despliegues de infraestructuras por parte de las operadoras de telefonía, se hace con intención de que las estaciones base instaladas tengan una vida útil de varias décadas. Sin embargo, con la constante evolución de los terminales móviles se acorta esta vida útil, ya que la tecnología cada vez es más cambiante. Una posible solución a este inconveniente sería la incorporación de SDR en las estaciones. De este modo, se podrían incorporar los nuevos protocolos inalámbricos con tan solo una descarga en el software, aumentando la flexibilidad de la infraestructura y el ahorro de costes por parte de las operadoras.

Del mismo modo, los terminales móviles están incorporando la tecnología SDR para soportar varios protocolos (*2G*, *3G*, *WLAN*, *GPS*, *Bluetooth*, televisión móvil, etc.). Al adoptar esta tecnología se disminuye el costo en el desarrollo de nuevos chips, puesto que, de igual forma que con las estaciones base, bastaría con una descarga en el software del chip existente.

#### Industria aeroespacial

Teniendo en cuenta que el sector satelital es conservador, la idea de que un mismo hardware soporte diferentes funciones configurables por el usuario resulta muy interesante. Aún está por determinar la viabilidad y estabilidad de SDR para introducirla en este sector.

Los satélites pueden clasificarse en dos grandes grupos: satélites de comunicaciones y de investigación. En el caso de los de comunicaciones actúan como repetidores, pero los de investigación van

un paso más allá. Estos se encargan del post-procesado de la información recibida. Es por ello, que la idea de implementar SDR resulta muy tentadora a la hora de establecer una comunicación entre el satélite y la estación terrena para lectura e interpretación de telemetría, detección de la posición de la estación, envío de telecomandos..., etc.

### **Industria automotriz**

La aplicación de SDR en los automóviles mejora la experiencia de los usuarios en los viajes a través de la utilización de tecnologías de comunicaciones. Debido a la congestión en carreteras y autovías por el aumento de vehículos y a la imposibilidad de construir nuevas vías, se ha decidido optar por la telemática para solventar este problema. En Europa, la principal preocupación del gobierno para paliar este problema es la información de tráfico y navegación. Por ello, en algunas áreas se han popularizado los servicios de información de tráfico como el *Canal de Mensajes de Tráfico* usando subportadoras FM.

Los automóviles emplean sistemas ECU (*Engine Control Unit*) para el control de las funciones hidráulicas, mecánicas, y electrónicas tradicionales siendo así más efectivos en costos. A día de hoy, los vehículos más modernos emplean entre 20 y 80 ECUs. En consecuencia, esta industria está planteando minimizar este número. Además, utilizando SDR se mejoraría la comunicación entre dispositivos en el vehículo, a través de buses, aprovechando al máximo los potenciales de los sistemas ECU.

### **Ámbito académico**

Son muchas las universidades que han decidido incluir el estudio de SDR en los planes de estudio de la Ingeniería de las Tecnologías de Telecomunicación (tanto Grado como Máster) debido a la gran repercusión que está teniendo esta tecnología a nivel mundial en las comunicaciones. En estos centros, los alumnos reciben una base teórico-práctica sobre SDR. Por ejemplo, en la Universidad de Sevilla existe una asignatura en el cuarto curso de dicho Grado, en la mención de Sistemas de Telecomunicación, donde se ofrece un primer contacto de SDR. En esta asignatura, los alumnos pueden ver de una manera directa y tangible su aplicación en una transmisión por radio mediante código Python, *USRP* y *GNURadio*.

Por otro lado, *Wireless Innovation Forum* ha creado una competición, patrocinada por los desarrolladores de MATLAB<sup>®</sup>, en la que grupos de estudiante diseñan, desarrollan y testean la Radio Definida por Software orientada a resolver problemas de la tecnología inalámbrica en el mundo real.

## **2.2 Sistemas SDR implementados en el mercado**

Antes de dar paso a los siguientes capítulos, para llevar a cabo la implementación de SDR, se precisa de una parte hardware. El hardware específico que va ligado a SDR es un dispositivo con aspecto similar al de una memoria USB de tamaño considerado. Se le conoce como *dongle*.

A continuación, se mencionan y detallan los dispositivos más populares.

### **2.2.1 Dongle USB RTL-SDR**

Este receptor de señales RF fue diseñado en 2010. Se caracteriza por emplear un demodulador DVB-T, el chip Realtek RTL2832U. Se descubrió que este chip tenía ciertos registros y comandos no documentados. Con ellos, podía llevarlo a un modo de funcionamiento donde transfiere las muestras I/Q en banda base sin procesar por un puerto USB 2.0 hacia un ordenador. Estas muestras son la representación digital de la amplitud de la señal modulada (componente Q, de *Quadrature*) y su fase rotada 90 grados (componente I, de *In Phase*) para un instante de tiempo dado. Y, por medio de algún software en el ordenador, es posible demodular el flujo de muestras I/Q que provienen del sintonizador. Este modo de funcionamiento del RTL2832U es el que permitiría a estos dongles

RTL-SDR otorgar la característica de receptor SDR de banda ancha. Al tratarse del primer receptor, RTL-SDR lleva su nombre.

Por otra parte, este dispositivo se diseñó utilizando el sintonizador de frecuencia más popular del mercado, Raphael Micro<sup>®</sup> R820T, o su versión posterior R820T2.

Este receptor es capaz de captar señales en la banda de frecuencias de los 500 kHz hasta los 1.7 GHz, y admite frecuencias de muestreo de hasta 3.2 MHz. Es el dispositivo que presta las mismas prestaciones que sus competidores a un precio más reducido, acercando SDR a todos los usuarios.

Es compatible con la mayoría de herramientas software como SDRSharp, HDSDR, SDR-Touch, SeeDeR, entre otros.

### 2.2.2 FUNcube Dongle Pro+

La Comunidad de Satélites Aficionados de Reino Unido *AMSAT-UK* fue uno de los responsables del diseño de este receptor en el año 2012. Puede recibir señales desde 150 kHz hasta 260 MHz y desde 410 MHz hasta 2.05 GHz. Además se caracteriza por una frecuencia de muestreo de 192 kHz y convertidores ADC de 16 bits.

También dispone de interfaz USB, mostrando una gran compatibilidad con la amplia alternativa de softwares de configuración de SDR, como el dongle RTL-SDR. El principal inconveniente es su elevado precio.

### 2.2.3 HackRF One

En 2014, fue lanzado este periférico de la mano de Michael Ossman, siguiendo con la tendencia iniciada con el USRP1 (*Universal Software Radio Peripheral*); a partir del cual se desarrollaron diferentes modelos de transceptores, basados en SDR, para trabajar con un mismo dispositivos diversas tecnologías radio a bajo coste.

Destaca por su capacidad para transmitir y recibir señales RF en la banda de frecuencias de 1 MHz a 6 GHz con una potencia del puerto de la antena (salida) configurable desde 30 mW hasta 1mW, mediante una comunicación semidúplex. Este transceptor es compatible con GNU Radio y SDRSharp. Mencionar la ausencia de interfaz USB y su elevado precio.

### 2.2.4 Mini Airspy

Este receptor ofrece un rango de recepción continuo desde los 24 MHz a los 1.8 GHz con un ancho de banda de 6 MHz libres de espurios, lo que lo hace de los receptores mas conocidos en el mercado. Salió a la venta en 2016 de la mano de la empresa iTeard<sup>®</sup>.

Se trata de un dispositivo de alta precisión, diseñado con un conversor ADC de 12 bits, con gran compatibilidad entre los softwares de SDR por un precio más asequible que los dos modelos anteriores.

### 2.2.5 Nooelec NESDR SMArTee

Su principal ventaja frente al resto de dispositivos es su bajo precio. Este dongle de banda ancha diseñado por Nooelec, es capaz de recibir señales RF en el rango de frecuencias que comprende desde los 25 MHz hasta los 1.7 GHz, presentando unas características muy similares a las del dongle RTL-SDR.

Por lo general, todos los dongles y transceptores presentan aplicaciones similares como el escaneo de bandas de frecuencia, recepción radio Amateur, monitorización radio, etc., no presentado apenas notables diferencias, siendo el fluctuante precio, entre los principales modelos, la más significativa. Esta última diferencia es la que ayudará al usuario a elegir un modelo u otro en función de la aplicación a desarrollar.

A continuación se deja una tabla resumen de las principales características de los dongles.

**Tabla 2.2** Comparación de sistemas SDR implementados en el mercado.

<b>Modelo</b>	<b>Año</b>	<b>Rango de frecuencias</b>	<b>Demodulador</b>	<b>Sintonizador</b>	<b>Precio</b>
Dongle USB RTL-SDR	2010	500 kHz - 1.7 GHz	RTL2832U	R820T	++
FUNcube Dongle Pro+	2012	[150 kHz - 260 MHz], [410 MHz - 2.05 GHz]	PIC24FJ32 GB002	E4000	++
HackRF One	2014	1 MHz - 6 GHz	LPC4320FBD144	RFFC507	++
Mini Airspy	2016	24 MHz - 1.8 GHz	LPC4370FET100	R820T2	--
Nooelec NESDR SMArTee	2016	25 MHz - 1.7 GHz	RTL2832U	R820T2	--



## 3 Materiales y Métodos

---

Una vez concluido el capítulo donde se presenta la tecnología en la que se basa este proyecto, a continuación, se proporciona a una descripción más detallada del dongle con el que se ha realizado la parte experimental.

Se ha elegido el modelo Nooelec NESDR SMArTee v2, mencionado anteriormente entre los más populares de los sistemas que implementan SDR. Lo que ha decantado la elección de este dispositivo, principalmente, es su gran relación calidad-precio; pues este modelo comprende un gran abanico de aplicaciones gracias a sus componentes y su arquitectura. Se anticipa que guarda una estrecha relación con el dongle USB RTL-SDR.

### 3.1 Nooelec NESDR SMArTee v2

Este dongle está basado en RTL-SDR, compartiendo los principales componentes, y es diseñado, desarrollado y fabricado por Nooelec. Está diseñado desde cero con la intención de seguir en la línea de dispositivos SDR de bajo costo actuales.

#### 3.1.1 Hardware y arquitectura

El NESDR SMArTee consta de:

- Una carcasa de aluminio que asegura una mejor sensibilidad a los circuitos de RF.
- Un TCXO (en español, un *oscilador de cristal con compensación de temperatura*) de 0.5 PPM de ruido de fase, lo que garantiza la máxima estabilidad a la hora de sintonizar en cualquier entorno.
- Sintonizador Rafael Micro<sup>®</sup> R820T2.
- Demodulador Realtek RTL2832U.
- Una T de polarización (*Bias-Tee*, lo que le da nombre). Este regulador de 4.5 V ofrece una salida de CC para alimentar a los componentes electrónicos activos. Es adecuado para RF, puesto que, con menos de 10  $\mu$ VRMS de ruido en la sección de la fuente de alimentación, persigue esa búsqueda de minimizar el ruido lo máximo posible.
- Circuitos de protección para evitar daños por cortocircuitos accidentales.
- Un disipador de calor personalizado a la PCB del dispositivo, con un resultado de temperaturas de 10°C, en promedio, más bajas que el resto de versiones de este producto.
- Entrada de antena SMA hembra universal.



Figura 3.1 NESDR SMArTee v2.

A continuación se hace especial mención a los dos componentes de más importancia en el dongle, puesto que son los responsables tanto del comportamiento de este como de la relación calidad-precio.

#### Rafael Micro<sup>®</sup> R820T/2 radio tuner

El R820T en su versión actual, R820T2, aunque no predomine en la mayoría de dongles RTL-SDR, es un sintonizador de silicio usado con frecuencia. Su rango de operación inicia en los 25 MHz y alcanza hasta los 1.75 GHz, aproximadamente, y ha logrado el menor consumo de energía, además de proporcionar un notable rendimiento en sensibilidad, linealidad e inmunidad de canal adyacente, entre otras características. Además, cuenta con un detector inteligente de potencia para optimizar diferentes escenarios de potencia de entrada.

Según se muestra en la Figura 3.2, el chip se basa en un sintetizador de frecuencia que genera una señal de oscilador local (*LO*, *local oscillator*) que es responsable de convertir hacia abajo la señal RF recibida a una frecuencia intermedia (*IF*, *intermediate frequency*). También se proporciona control de ganancia, tanto en el amplificador de bajo ruido (*LNA*, *low noise amplifier*), como en la salida, a través de un amplificador de ganancia variable (*VGA*, *variable gain amplifier*). El término de control automático de ganancia (*AGC*, *automatic gain control*), se refiere al uso de un circuito o algoritmo de detección de intensidad de señal para retroalimentar la señal que controla los circuitos del detector inteligente anteriormente mencionado.

El rango de frecuencias modo de observación el R820T2 toma el papel de interfaz entre la señal RF y el SDR.

#### Realtek RTL2832U

El chip RTL2832U, por el contrario al anterior, es habitual encontrarlo en los distintos modelos de dongles USB RTL-SDR. Es un demodulador de alto rendimiento que admite interfaz USB 2.0. Admite sintonizadores en salida IF utilizando un cristal de 28.8 MHz y presenta una alta estabilidad gracias al convertidor analógico-digital avanzado que tiene integrado. Este flujo de datos muestreados resultantes de la conversión se procesa mediante una demodulación OFDM (*Orthogonal Frequency Division Multiplexing*). Además, cuenta con algoritmos patentados en Realtek, que incluyen estimación de canal superior, rechazo de interfaz de canal común, recepción de canal de eco largo y cancelación de ruido de impulso, y proporciona una solución ideal para una amplia gama de aplicaciones para ordenadores y televisiones, como el dongle USB RTL-SDR en este caso.

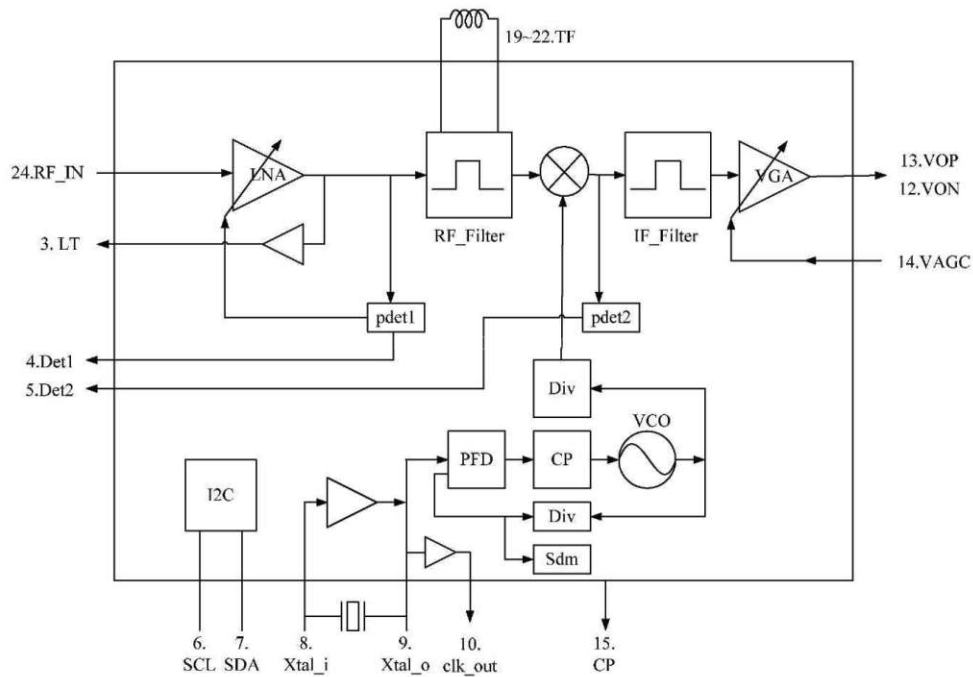


Figura 3.2 Raphael Micro<sup>®</sup> R820T radio tuner.

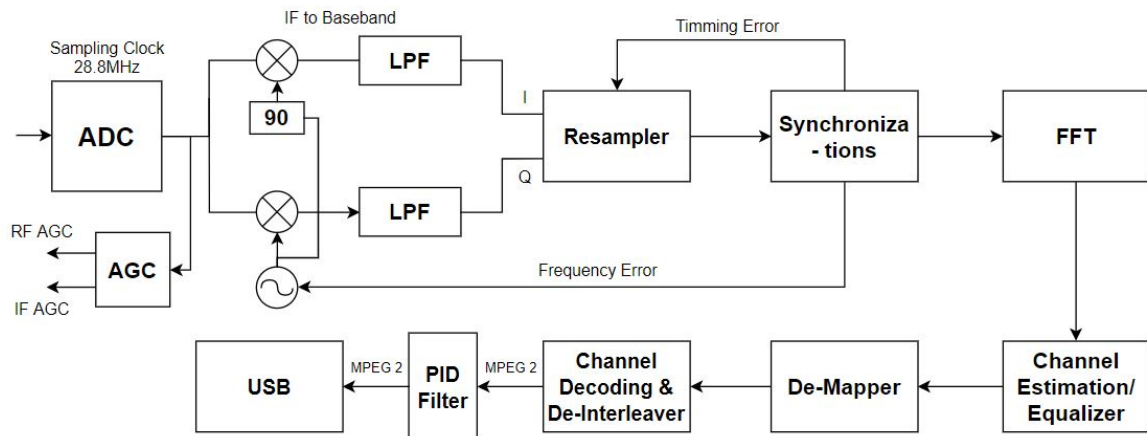


Figura 3.3 Diagrama de Bloques de Realtek RTL2832U.

Al ser un chip de continua presencia en los dongles USB RTL-SDR, se detalla una descripción funcional tomando como referencia la Figura 3.3, que muestra el diagrama de bloques:

- El convertidor interno analógico-digital (bloque ADC, *Analog-to-Digital Conversion*) puede aceptar la salida del sintonizador con varios anchos de banda (6, 7 y 8 MHz), diferentes frecuencias intermedias y entrada Zero-IF (Canales I, *In-Phase* o componente en fase, y Q, *Quadrature* o componente en cuadratura). Usando un reloj de muestreo generado por el PLL (*Phase Locked Loop* o Bucle de Enganche de Fase) interno con una frecuencia de reloj de 28.8 MHz, el RTL2832U demodula la señal recibida.
- El circuito de control de ganancia automático (bloque AGC, *Automatic Gain Control*) se usa para ajustar la fuerza de la señal recibida a un nivel moderado para el ADC. Este módulo

dispone de dos rutas de salida, y ambas son señales moduladas sigma-delta. Las señales de salida necesitan filtros de paso bajo adicionales antes de alimentar a IF y VGA en el sintonizador.

- El RTL2832U usa muestreo IF o Zero-IF para procesar las señales recibidas. El circuito de conversión digital descendente (DDC, *Digital Down Conversion*) convierte la señal de IF muestreada a una señal banda base compleja para su posterior procesamiento. La frecuencia de la conversión descendente y el filtro pasobajo pueden ser programadas acorde a las diferentes frecuencias intermedias, tasas de muestreo y anchos de banda de la señal.
- El circuito del Remuestrador (*Resampler*) cambia la señal recibida de una tasa de muestreo ADC fija a una tasa de muestreo OFDM de acuerdo con el ancho de banda de la señal.
- En modulación en el dominio del tiempo, hay un tiempo de guarda insertado entre dos señales OFDM. Es necesario quitar ese tiempo de guarda antes del procesamiento de la Transformada Rápida de Fourier (FFT, *Fast Fourier Transform*). Este módulo se utiliza para moderar el límite de símbolo OFDM para FFT de acuerdo con los resultados de la sincronización.
- El circuito FFT convierte la señal recibida en el dominio del tiempo a una señal en el dominio de la frecuencia.
- El RTL2832U puede medir y compensar una amplia gama de compensaciones de frecuencia de muestreo y compensaciones de frecuencia portadora antes de tomar una decisión sobre datos. Se utiliza un límite de símbolo moderado para evitar la interferencia entre símbolos (ISI, *Inter-Symbol Interference*).
- La respuesta estimada del canal puede usarse para ecualización, y es actualizada símbolo a símbolo para soportar variaciones de canales móviles. El encargado de esta función es el circuito de estimación de canal.
- Para soportar varias condiciones de canal, el circuito de ecualización compensa la degradación de la señal causada por diferentes perfiles de canales. El bit de datos es detectado en función de la salida del ecualizador.
- El chip incorpora un desintercalador (bloque *de-interleaver*) para reordenar el bit de datos en la secuencia correcta. Además, un decodificador FEC (*Forward Error Correction*) detecta y corrige los bits de error en la secuencia recibida.
- En último lugar, el filtro PID (*Packet Identifier*) permite la reducción del flujo de transporte a la salida del demodulador.

### **Diagrama de bloques completo**

A modo resumen, para recapitular todo lo visto, se proporciona la Figura 4.7, que proporciona una visión completa del diagrama de bloques del NESDR SMArTee v2.

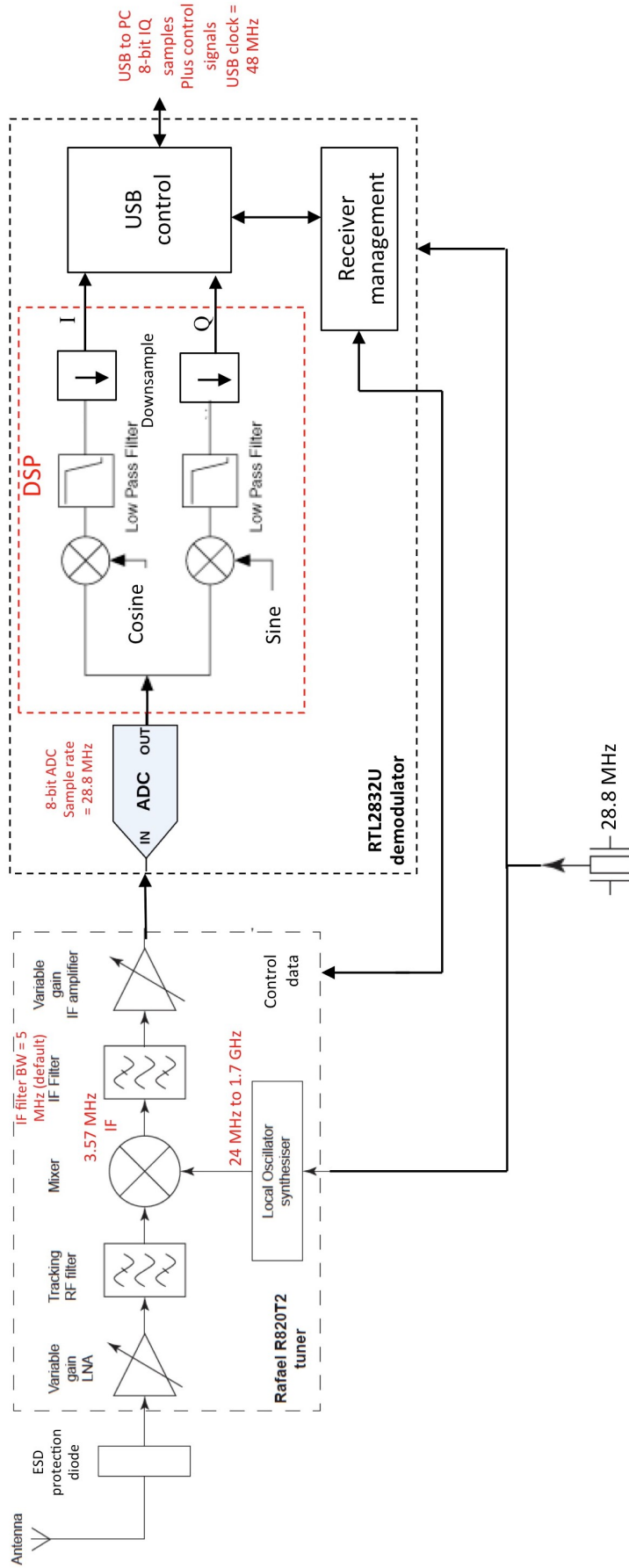


Figura 3.4 Diagrama de Bloques de Nooelec NESDR SMATee v2.

### 3.1.2 Equipamiento y aplicaciones

El dispositivo Nooelec NESDR SMARTEE viene acompañado de un conjunto de tres antenas. De esta forma se consiguen numerosas aplicaciones que generan una lista considerable, la cual seguramente siga creciendo con el tiempo.

Para una mejor visión, se engloban en cinco grupos según su función.

#### **Funciones de escáner**

- Escáner de radio de policía.
- Escaneo de conversaciones de radio de red troncal.
- Búsqueda de teléfonos inalámbricos y monitores para bebés.
- Escáner de radio portátil con un dispositivo Android y RTL-SDR.

#### **Funciones de escucha**

- Escuchar comunicaciones de EMS (*Emergency Medical Services*).
- Escuchar conversaciones de control de tráfico aéreo.
- Escuchar radio amateur VHF.
- Escuchar satélites y a la ISS (*International Space Station*).
- Escuchar radio FM y radiodifusión DAB.

#### **Funciones de seguimiento**

- Seguimiento de aeronaves, como un radar con decodificación ADSB.
- Seguimiento de embarcaciones marítimas, como un radar con decodificación AIS.
- Seguimiento de globos meteorológicos.
- Seguimiento de dispersión de meteoros.

#### **Funciones de decodificación**

- Decodificación de mensajes cortos de aeronaves.
- Decodificación de transmisiones de voz digital sin cifrar.
- Decodificación del tráfico de buscapersonas.
- Decodificación de paquetes ARPS de radioaficionados.
- Decodificación de señales de terminales de datos móviles de taxis.

#### **Otras funciones**

- Radioastronomía.
- Analizador de espectro.
- Caracterización de filtros RF.

A modo de comentario, dependiendo del modelo de RTL-SDR que se utilice se podrán ampliar estas aplicaciones. Esto es así, debido a la variante de sintonizador que se use en el dongle en particular, pues ofrecerá un rango de frecuencias diferente para cada RTL-SDR.

En la siguiente Tabla 3.1 puede apreciarse que los dongles Rafael Micro<sup>®</sup> R820T/2 y Elonics<sup>®</sup> E4000 tienen mayor rango de frecuencia.

**Tabla 3.1** Rango de frecuencias según el sintonizador.

<b>Sintonizador</b>	<b>Rango de frecuencia</b>
Rafael Micro <sup>®</sup> R820T/2	[24 - 1766] MHz
Elonics <sup>®</sup> E4000	[52 - 2200] MHz
Fitipower <sup>®</sup> FC0013	[22 - 1100] MHz
Fitipower <sup>®</sup> FC0012	[22 - 948.6] MHz
FCI <sup>®</sup> FC2580	[146 - 308] y [438 - 924] MHz





## 4 Diseño de Experimentos

---

Anteriormente se introdujo que para poder implementar un sistema SDR, además del equipo hardware, se requiere una parte software para el procesado de la señal.

Para los experimentos que se mostrarán a continuación, se ha elegido el ya conocido equipo de Nooelec, presentado en el capítulo anterior, y los programas de *MathWorks*, MATLAB (en su versión R2020a) y Simulink.

Además, hay que tener en cuenta que la evaluación de los mismos se ha llevado a cabo en un entorno urbano como lo es la ciudad de El Puerto de Santa María, Cádiz.

### 4.1 Equipo de trabajo

El equipo de Nooelec incluye:

- Mástil de antena telescópico con frecuencia variable.
- Mástil de antena de 433 MHz (*ISM*) de frecuencia fija.
- Mástil de antena *UHF* de frecuencia fija.
- NESDR SMArTee v2.
- Soporte para antena con base imantada y cable RG58 de 2m, de baja pérdida, con terminación SMA macho.
- Tarjeta de presentación con código QR para la guía de instalación.

Es conveniente destacar que todas las pruebas están realizadas en un ordenador con Windows® instalado, además es necesario tener instalado MATLAB y Simulink en dicho ordenador antes de comenzar el montaje del sistema SDR.

Una vez instalado, para llevar a cabo la parte software, se requiere la siguiente caja de herramientas (a partir de ahora, *toolbox*) de MathWorks. Estos toolboxes son complementos oficiales del software que brindan características adicionales en el entorno de desarrollo (tales como herramientas para el diseño de filtros y *scopes* en el dominio de la frecuencia). Para este caso en cuestión, se precisará del toolbox '*Communications System Toolbox*'.

#### 4.1.1 Verificación del entorno de trabajo

Antes de dar comienzo, es imprescindible asegurar la configuración previa del dispositivo. Este paso consta de un par de tareas que aseguran el correcto funcionamiento de todo el equipo de trabajo.

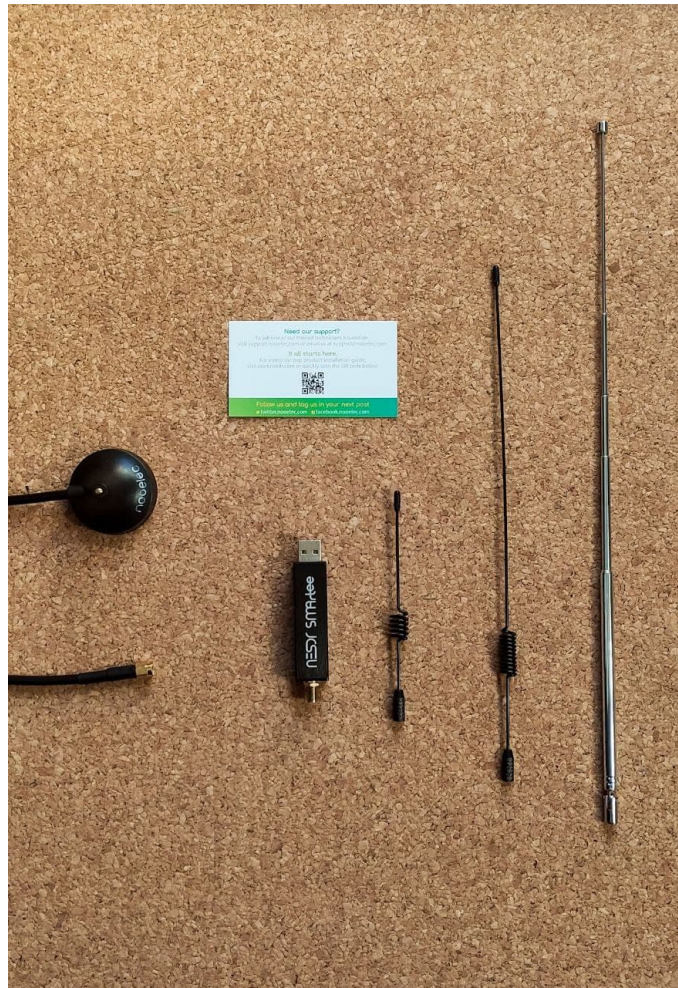


Figura 4.1 Equipo de Nooelec.

En primer lugar se debe instalar el driver del USB. Para ello, se introduce el dispositivo en un puerto USB. Automáticamente, el sistema operativo reconocerá el dispositivo y procederá a la instalación. El instalador abrirá una nueva ventana con el software *Zadig* para terminar de manera fácil e intuitiva dicha tarea.

En segundo y último lugar, hay que verificar que MATLAB se comunica correctamente con el dispositivo RTL-SDR. Para ello, se descarga el toolbox nombrado anteriormente. Este aparecerá en el apartado '*Add-ons Explorer*' de la parte superior derecha del menú, como se observa en la Figura 4.2.

Para proceder a la verificación del software, hay que hacerlo en dos pasos. El primero de ellos, se basa en la ejecución del comando *sdrrdoc* en la ventana de comandos (*Command Window*) de MATLAB, abriéndose la ventana mostrada en la Figura 4.3-(a). Esto es indicativo de que todo está bien.

El segundo paso se corresponde con la verificación del hardware a partir del software. Para ello, el dispositivo debe estar conectado a un puerto USB del ordenador. Acto seguido, en la ventana de comandos de MATLAB, se introduce `my_rtlsdr = sdrinfo`. El resultado se muestra en las Figuras 4.3-(b) y 4.3-(c).

En concreto, la Figura 4.3-(c) se corresponde con la misma comprobación realizada automáticamente desde el toolbox instalado previamente. Puede apreciarse que aparece la misma información del dispositivo.

The screenshot shows the MATLAB Add-On Explorer interface. At the top, there's a search bar and navigation options. The main content area displays the 'Communications Toolbox Support Package for RTL-SDR Radio' by MathWorks Communications Toolbox Team. It includes a description: 'Acquire RF data using RTL-SDR.' and a 'Hardware Support' icon. There are 'Learn More' and 'Install' buttons. Below this, an 'Editor's Note' states that the file was selected as 'MATLAB Central Pick of the Week' and that the package is currently unable to download third-party software for MATLAB R2017a and earlier versions. A 'Requires' section lists 'Communications Toolbox', 'DSP System Toolbox', and 'Signal Processing Toolbox'. 'MATLAB Release Compatibility' shows it was created with R2014a and is compatible with R2014a to R2020b. 'Platform Compatibility' includes Windows, macOS, and Linux. 'Categories' include 'Wireless Communications > Communications' and 'Supported Hardware - Software-Defined Radio'.

Figura 4.2 Communications System Toolbox Support Package.

Para concluir con la verificación del entorno de trabajo y confirmar que todo ha ido correctamente, hay que introducir el comando `help import_rtlsdr_data` en la ventana de comandos de MATLAB. Si ha habido algún error, mostraría `'not found'`.

## 4.2 Primer contacto con NESDR SMARTEE

Una vez validado el sistema SDR, como primera toma de contacto y para establecer un entorno familiar con dicho sistema, se mostrará el espectro de algunas señales de RF que se transmiten en la ubicación mencionada al inicio del capítulo.

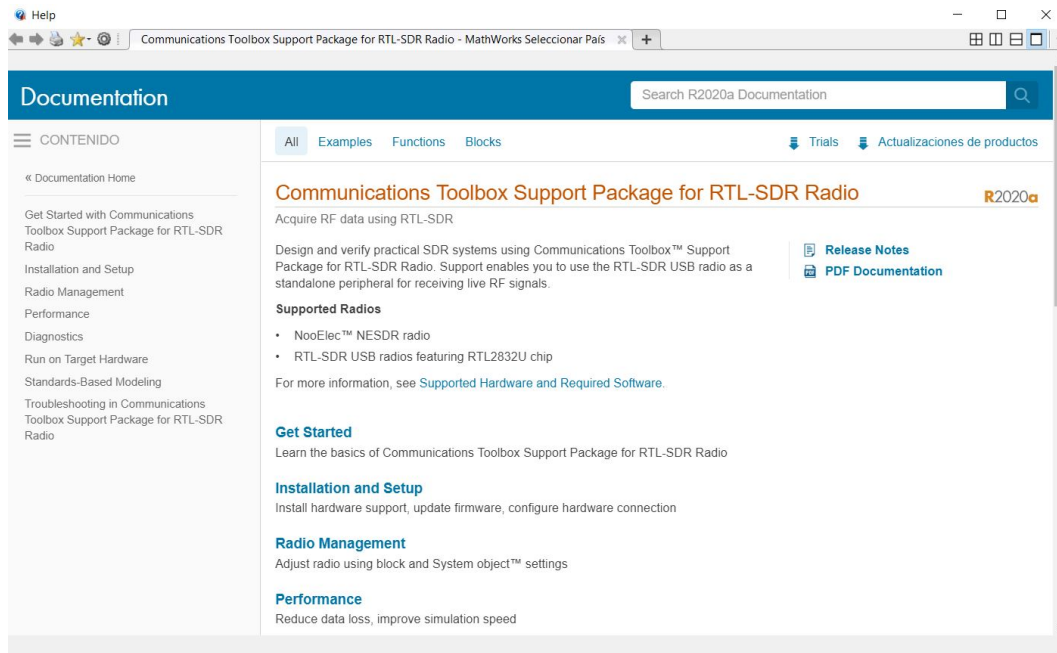
Para ello entra en juego la parte software del sistema SDR, la parte más importante, que reside en el código A.1 mostrado en el Apéndice A. Mediante dicho código se construyen los objetos que forman el sistema SDR montado. Estos objetos se definen persiguiendo una modificación fácil y cómoda de algunos parámetros como la frecuencia central del receptor, frecuencia de muestreo o ganancia, entre otros. El primero de ellos hace referencia al conjunto de NESDR SMARTEE más antena y el segundo al Analizador de Espectros (o *Spectrum Analyzer*).

Según puede apreciarse en las Figuras 4.4-(a) y 4.4-(b), las imágenes mostradas se corresponden con el resultado de ejecución del código. Puede observarse que es equivalente a lo que se obtendría en un escenario donde no se hace uso de SDR.

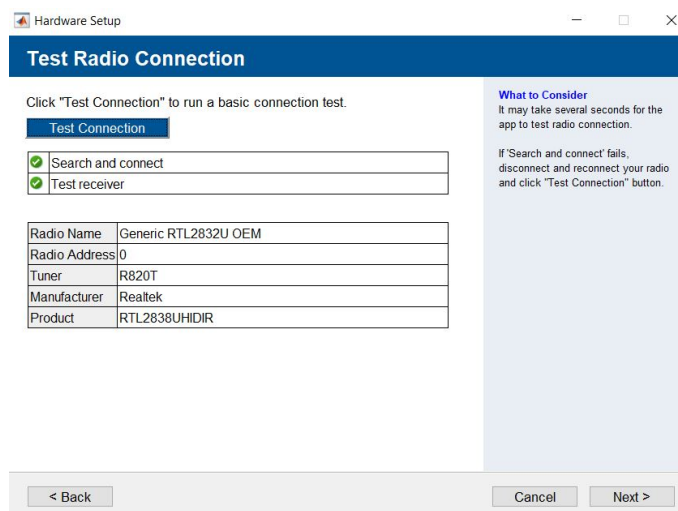
Entrando más en detalle, sería conveniente realizar un análisis acerca de la interpretación de ambas gráficas, pues van a ser imágenes que guarden relación con las resultantes en los próximos experimentos.

La primera imagen muestra una representación de las señales captadas por la antena mediante la FFT (*Fast Fourier Transform* o Transformada Rápida de Fourier). Destacar que la FFT no es más que un algoritmo basado en la DFT (*Discrete Fourier Transform* o Transformada de Fourier Discreta) que reduce notablemente el tiempo de cálculo, aumentando la eficiencia.

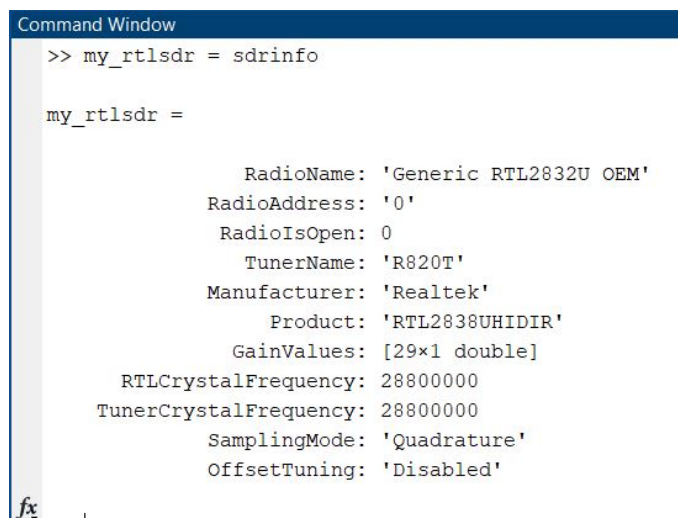
En lo que a la gráfica respecta, en el eje de ordenadas aparece representado el nivel de potencia (en dBm/Hz) del contenido espectral, y en el eje de abscisas se representa la frecuencia (en MHz) con una escala en función de la separación temporal y el número de muestras. Como apunte, la



(a)

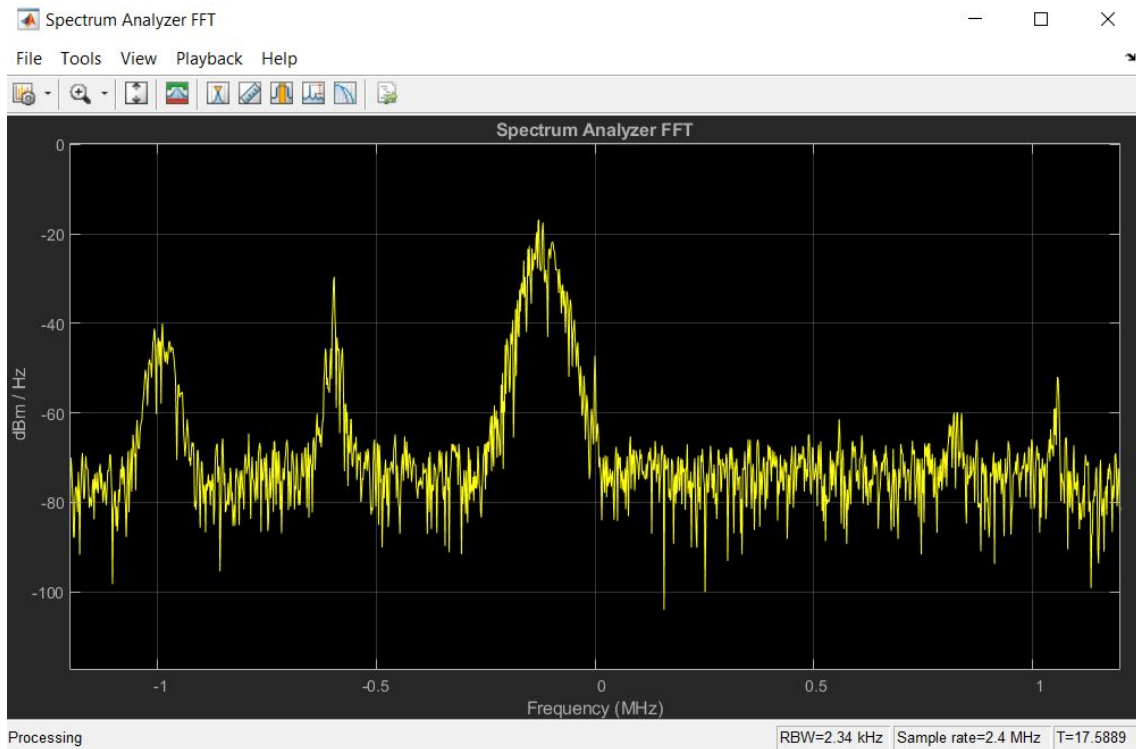


(b)

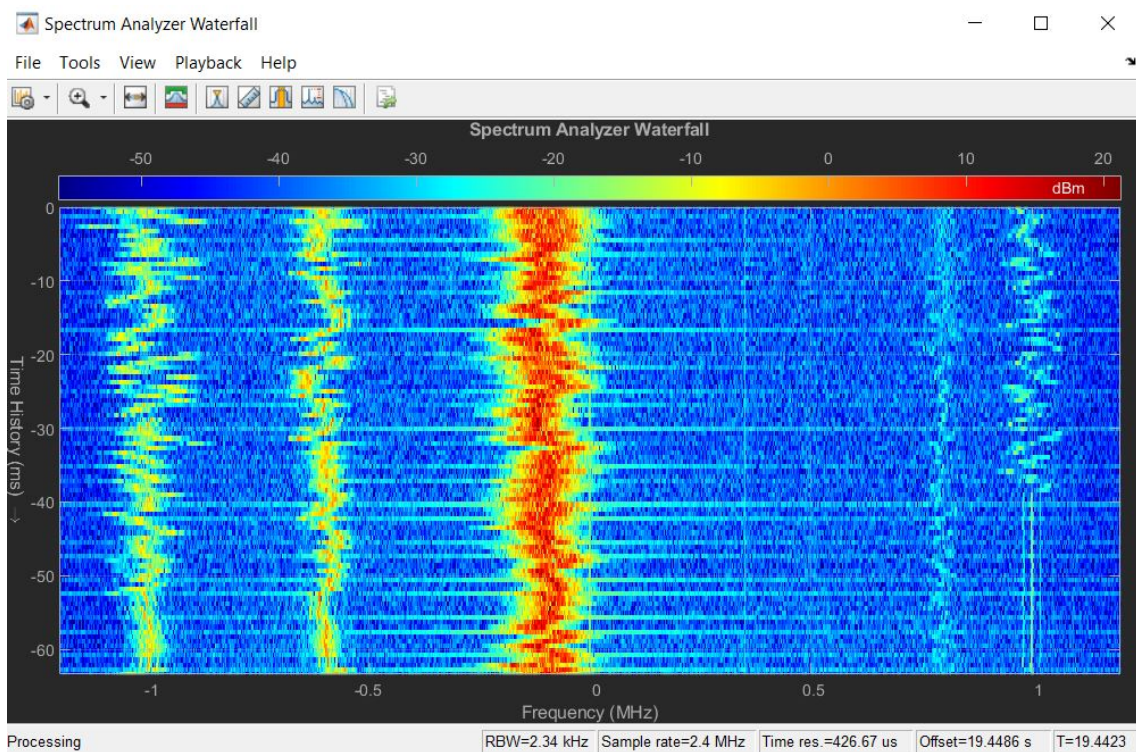


(c)

Figura 4.3 Verificación del entorno de trabajo.



(a)



(b)

Figura 4.4 Espectro de señales RF.

frecuencia correspondiente con el punto medio de la gráfica se conoce como frecuencia central (para este caso se ha fijado una frecuencia central de 100 MHz, correspondiente a la banda de radio FM, con una frecuencia de muestreo de 2.4 MHz). Sabiendo esto, y mirando el nivel de potencia, es inmediato identificar en la Figura 4.4-(a) la presencia de tres señales situadas a 1 MHz, 0.6 MHz y 0.125 MHz respecto a la frecuencia central, siendo la última de ellas la que mejor se percibe, puesto que presenta una potencia de -20 dBm.

En la segunda imagen aparece lo que se conoce como un espectrograma (o *waterfall*). No es más que una representación en tiempo real de las señales que se captan por la antena. En otras palabras, muestra la actividad espectral. El eje de ordenadas representa el historial de tiempo en milisegundos (ms) y el eje de abscisas, de nuevo, la frecuencia (en MHz). No obstante, se representa también una guía de colores que hace alusión al nivel de potencia (en dBm), siendo el color más cálido el que indica mayor nivel de potencia. A efectos prácticos y en un lenguaje coloquial, puede interpretarse que la Figura 4.4-(b) muestra el «rastros» que dejan las señales en el tiempo.

A modo de observación, se destaca la continua correspondencia entre ambas gráficas.

### 4.3 Modelo de receptor para la visualización del espectro

A modo introductorio, el espectro RF comprende una parte del espectro electromagnético que inicia en los 3kHz y abarca hasta los 300 GHz. Dicha porción tiene un uso muy extenso en la sociedad, pues se trata de la banda de frecuencias donde conviven los servicios de comunicación, mediante aplicaciones como la difusión de señales de televisión, radio, móvil y WiFi; con los sistemas de navegación y detección como radares, GPS, transpondedores, etc.

Sin mas dilatación, el siguiente experimento propone la construcción de un modelo receptor a través del cual se capacita al sistema para recibir señales RF en tiempo real además de ir viendo todo el espectro RF. A nivel funcional es igual que el ejercicio anterior, sólo que esta vez se hace a través de un modelo de Simulink, con la principal diferencia de que contiene una Interfaz Gráfica de Usuario (*GUI*) para controlar la frecuencia central y la ganancia del dongle.

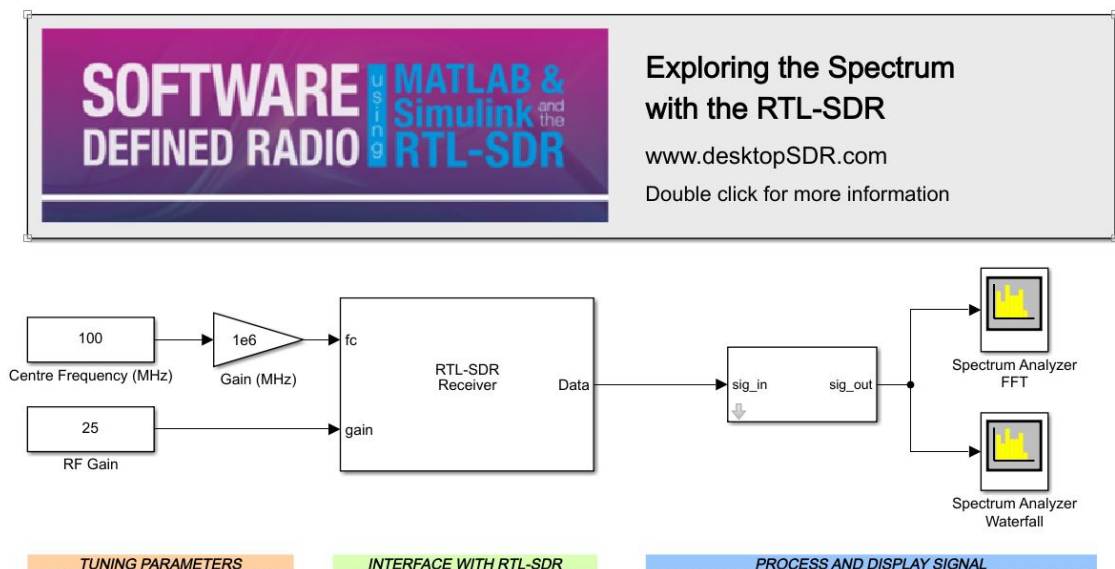


Figura 4.5 Modelo de Simulink receptor de señales RF.

En el diagrama de la Figura 4.5, el NESDR SMARTEE está sintonizado al valor del bloque *Centre Frequency (MHz)* y se aplica una ganancia de sintonizador de acuerdo con el valor de *RF Gain*. Por el bloque *RTL-SDR Receiver* salen muestras de 8 bits, las cuales pasan al bloque *Remove DC Component*. Este penúltimo bloque busca el valor medio de una trama de datos y lo resta para eliminar el pico de DC en el espectro. Por último, la señal pasa a dos bloques *Spectrum Analyzer*. El primero de ellos está configurado, al igual que el ejercicio anterior, para mostrar la FFT de una señal recibida, la cual tendrá un ancho de banda de  $f_s$  MHz, siendo  $f_s$  la frecuencia de muestreo que se ha establecido previamente. De este modo, el rango de actividad espectral que muestra este bloque comprende:

$$\text{desde } \left(f_c - \frac{f_s}{2}\right) \text{ hasta } \left(f_c + \frac{f_s}{2}\right) \text{ Hz,} \quad (4.1)$$

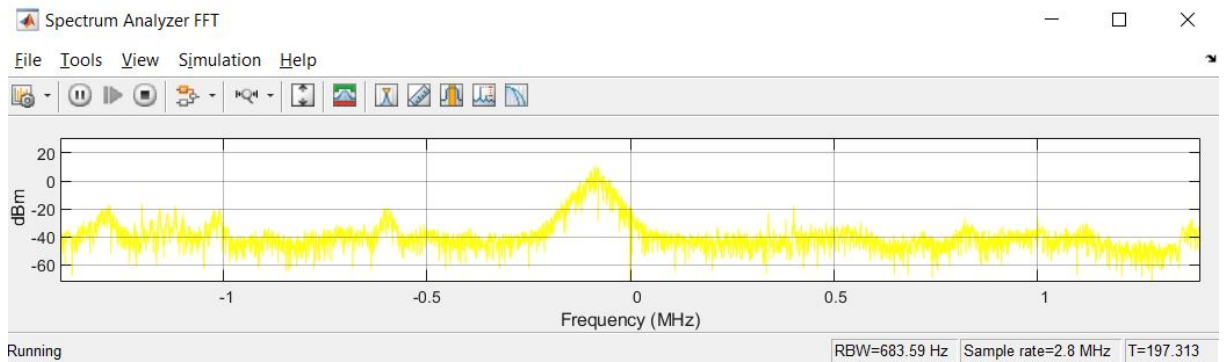
centrado al rededor de  $f_c$ . El segundo está configurado para que muestre un espectrograma con el mismo ancho de banda que el primero.

Al simular el modelo anterior, se obtienen las Figuras 4.6-(a), 4.6-(b) y 4.6-(c), que corresponden con una representación mediante FFT, un espectrograma (figuras ya conocidas de la sección anterior) y la GUI, respectivamente.

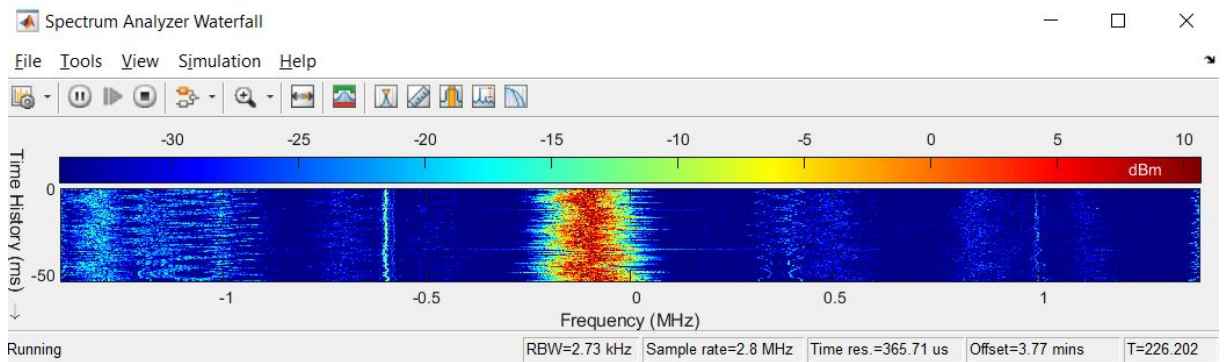
A modo de apreciación, los resultados mostrados se han ejecutado en base a una frecuencia de muestreo de 2.8 MHz. Para modificar este parámetro habría que acceder mediante doble click a la caja de *RTL-SDR Receiver*, como se muestra en la Figura 4.7, y se prueba a simular para una frecuencia de 3.2 MHz. Sin embargo, se recomienda no exceder los 2.8 MHz, ya que esta tasa asegura un muestreo preciso y confiable del espectro sin pérdida de información. El resultado de esta nueva simulación se aprecia en las Figuras 4.8-(a), 4.8-(b). Los valores de muestreo aparecen en la Tabla 4.1.

**Tabla 4.1** Frecuencias de muestreo aceptadas en el NESDR SMARTEE.

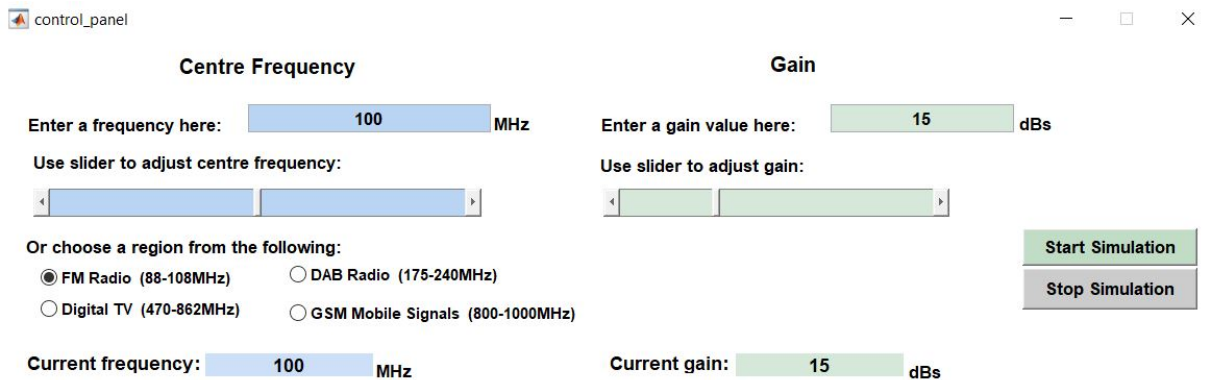
<b>Rangos de Frecuencia de Muestreo (<math>f_s</math>)</b>	
225001 Hz hasta 300001 Hz	(225.001 kHz - 300.001 kHz)
900001 Hz hasta 3200000 Hz	(900.001 kHz - 3.2 GHz)



(a)



(b)



(c)

Figura 4.6 Resultado de ejecutar el modelo de Simulink.

A través de la GUI puede sintonizarse la señal a una frecuencia diferente. Para ello el panel de control incorpora unas barras deslizadoras horizontalmente para ajustar el valor al que se desea centrar la señal captada en el Analizador de Espectro. Por defecto, la frecuencia del sintonizador es igual a 0 Hz y  $f_s$  igual a 2.8 MHz. De este modo, se puede conseguir que la señal de interés esté referenciada a 0 Hz, sin *offset*. Un ejemplo se ilustra en la Figura 4.9.



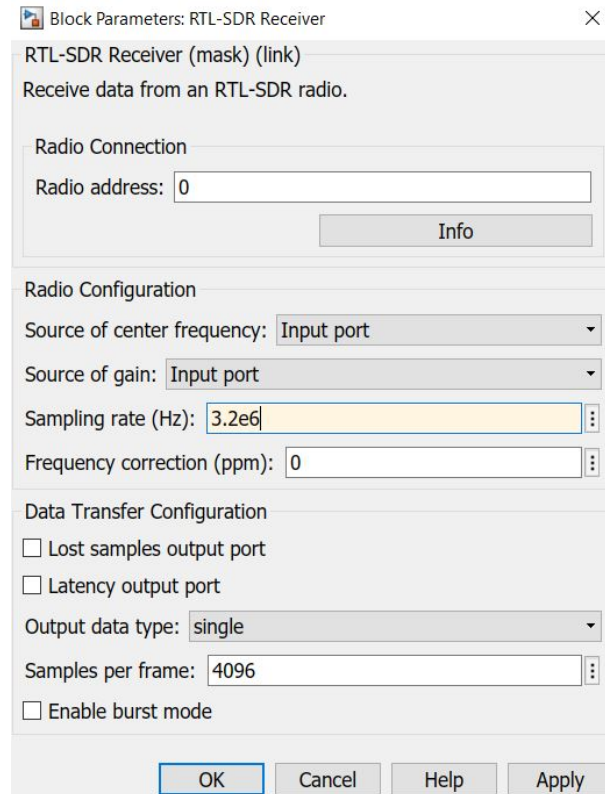
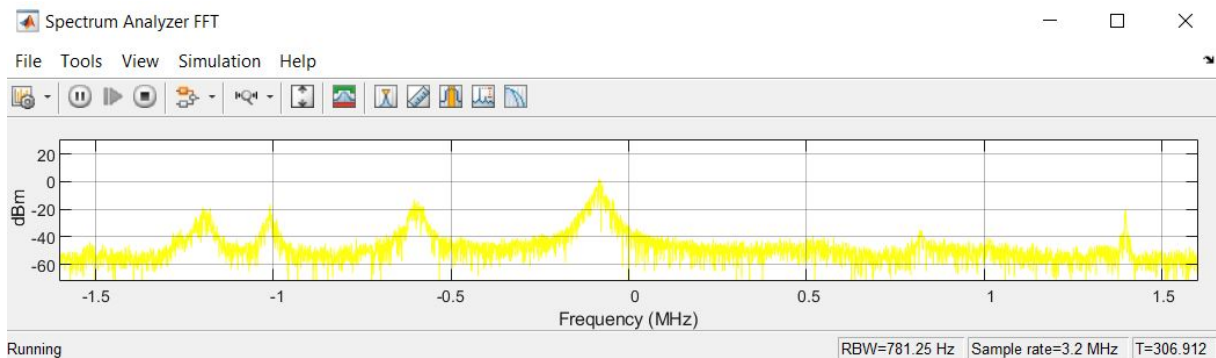
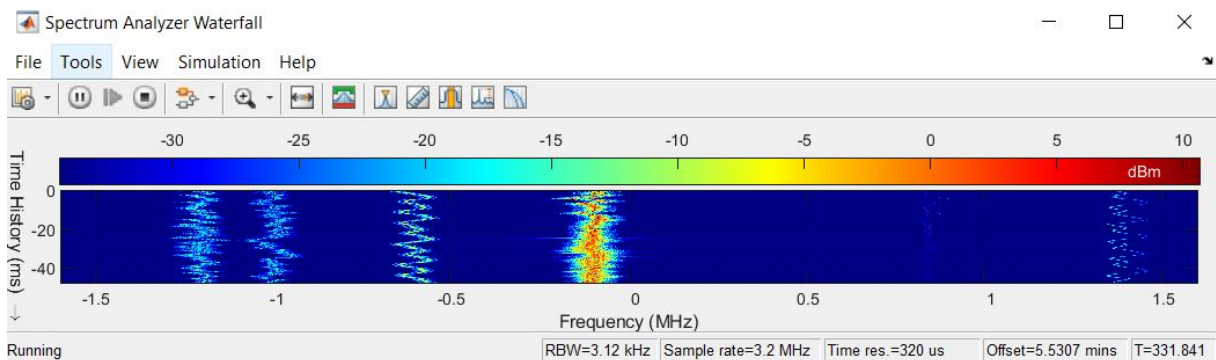


Figura 4.7 Configuración de parámetros del NESDR SMARTEE.

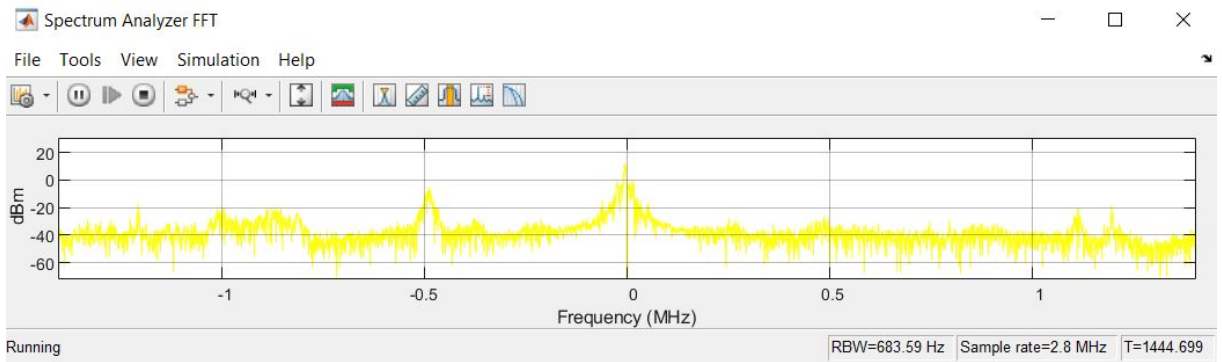


(a)

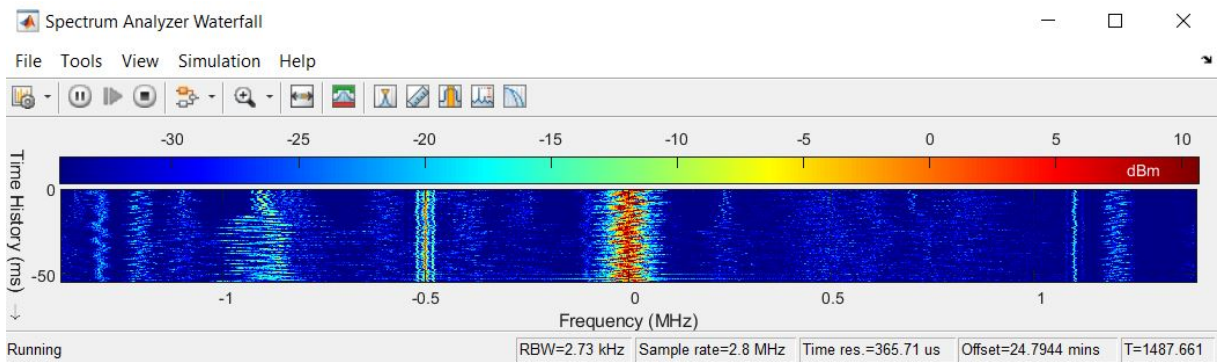


(b)

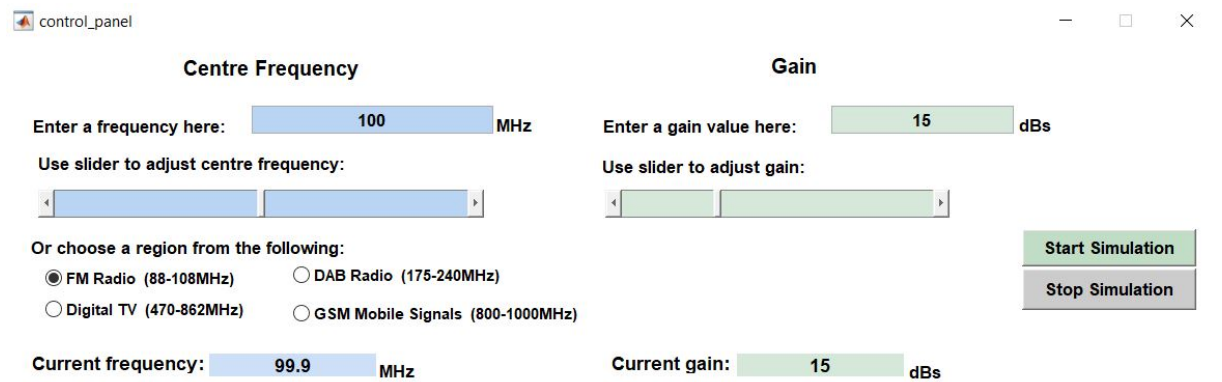
Figura 4.8 Resultados al modificar  $f_s$  de 2.8 MHz a 3.2 MHz.



(a)



(b)



(c)

Figura 4.9 Señal centrada en 100 MHz.

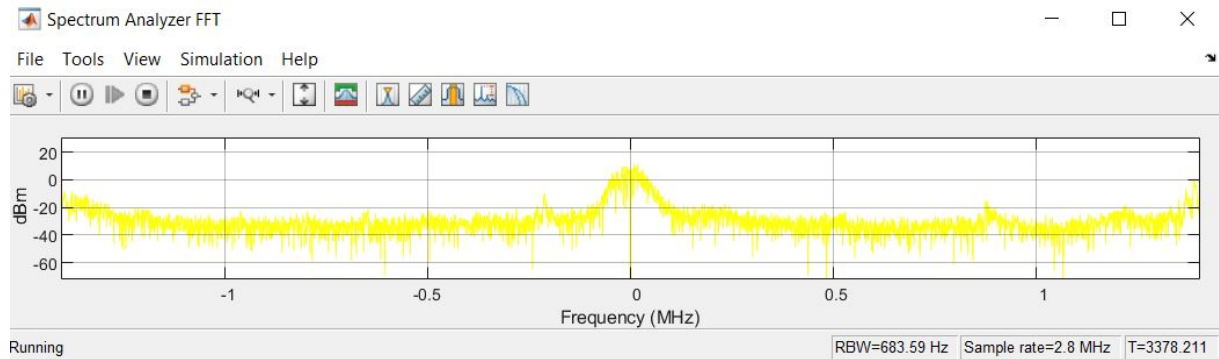
## 4.4 Búsqueda de estaciones de radio FM

Esta sección se centra, como su título indica, en la búsqueda de señales de radio FM. Estas señales se encuentran en el rango de frecuencias comprendido entre los 88 MHz y los 108 MHz, con una separación de 200kHz entre portadoras.

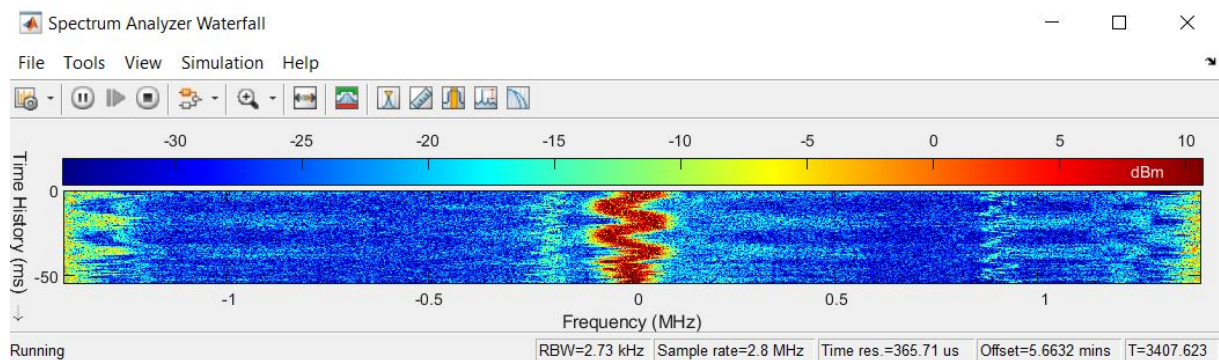
A partir de este momento entra en juego todo lo explicado en las secciones anteriores, pues se hará uso de la GUI para ir recorriendo el intervalo de frecuencias donde residen las señales FM, de los Analizadores de Espectro con su representación FFT y el espectrograma, y los conceptos de frecuencia de muestreo y ganancia.

Navegando por la banda FM, la primera señal con notable intensidad, según se muestra en

la Figura 4.10, se corresponde con una frecuencia de 89.4 MHz. Dicha señal, en la ubicación mencionada al principio del capítulo, se corresponde con la emisora 'Los 40 Principales'.



(a)



(b)

The screenshot shows a 'control\_panel' window with two main sections: 'Centre Frequency' and 'Gain'.  
 - **Centre Frequency:** Includes a text input field with '88' and 'MHz', a slider, and radio buttons for 'FM Radio (88-108MHz)', 'DAB Radio (175-240MHz)', 'Digital TV (470-862MHz)', and 'GSM Mobile Signals (800-1000MHz)'. The 'Current frequency' is displayed as '89.4 MHz'.  
 - **Gain:** Includes a text input field with '25' and 'dBs', a slider, and a 'Current gain' display showing '25 dBs'.  
 - **Buttons:** 'Start Simulation' and 'Stop Simulation' buttons are located on the right side.

(c)

Figura 4.10 Señal de radio FM.

Es sabido que estas señales FM son producidas por una modulación de la fase de una portadora que posee la información de la señal. Viendo en detalle la estación de radio elegida, con suerte se deberían apreciar los pequeños picos de subida deslizándose de lado a lado alrededor de la frecuencia central. Este movimiento se aprecia notablemente mejor en el espectrograma de la Figura 4.10-(b). Las oscilaciones que destacan se corresponden con la propagación de las señales de FM a través del aire.

## 4.5 Análisis de señales móviles GSM (2G)

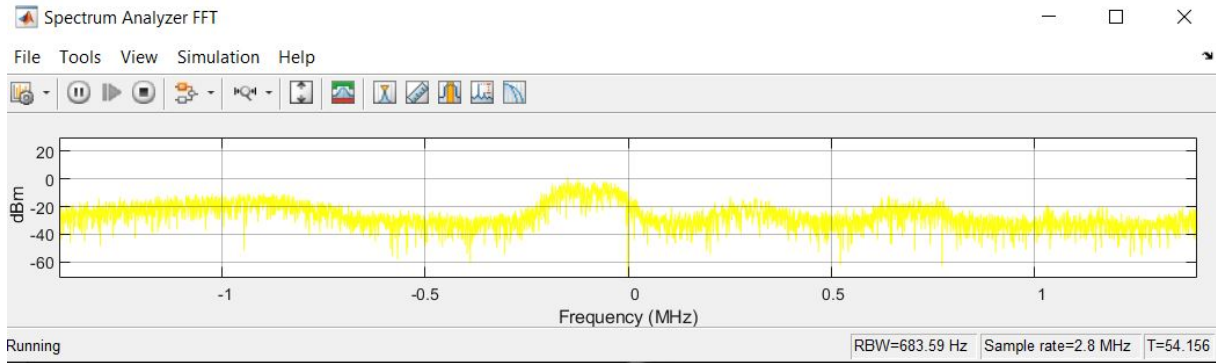
El sistema global para las comunicaciones móviles (GSM, *Global System for Mobile communications*) es, como su nombre indica, un sistema estándar de telefonía móvil digital desarrollado por el Instituto Europeo de Normas de Telecomunicaciones (ETSI, *European Telecommunications Standards Institute*). Este estándar opera entre los 890 MHz y los 960 MHz con portadoras espaciadas 200 kHz.

Las señales GSM se transmiten desde estaciones base hasta el teléfono móvil en cuestión en lo que se conoce como canal descendente o *downlink*. En Europa, el downlink comprende el intervalo de frecuencias desde los 935 MHz hasta los 960 MHz. Por el contrario, las señales que van desde el teléfono móvil a la estación base se transmiten en el canal ascendente o *uplink*, que inicia en los 890 MHz y abarca hasta los 915 MHz. Destacar que existen versiones posteriores que operan en intervalos de frecuencia ligeramente distintos, pero mantienen la estructura referida a un enlace ascendente y otro descendente.

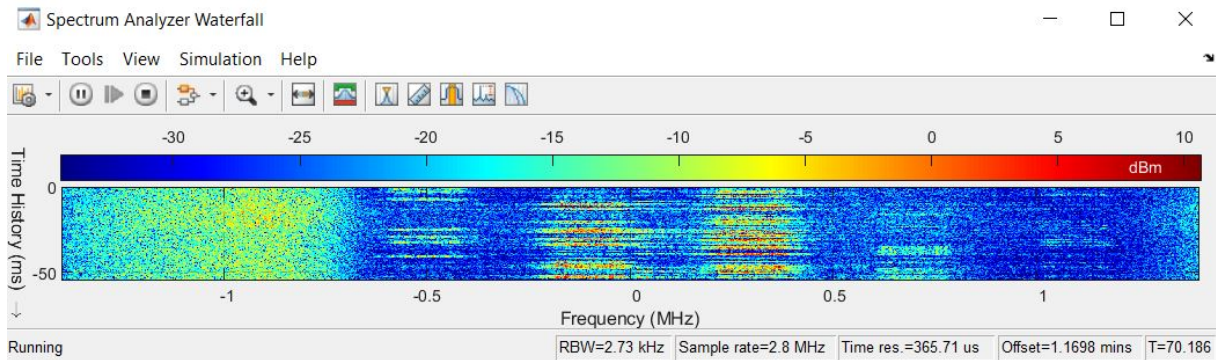
En este estándar se emplea la técnica de acceso múltiple al medio por división de tiempo (conocido como TDMA, *Time Division Multiple Access*). De este modo, se permite que múltiples usuarios de telefonía móvil tengan acceso y conectividad a través de un sólo canal dividiendo los intervalos de tiempo de acceso del canal.

A continuación se propone un análisis en dos zonas del espectro, dentro de la banda descendente, para ofrecer una mayor visualización de la actividad en dicha región. Realizando este análisis, se obtienen las Figuras 4.11 y 4.12.

Tras el análisis del espectro, puede apreciarse con minucioso detalle algunas señales GSM. Estas aparecen reunidas en pequeños grupos de canales de 200 kHz de ancho. Todas ellas operan independientemente de otras a intervalos de tiempo, en episodios repentinos y cortos. Lo comentado se puede analizar en el espectrograma de las Figuras 4.11-(b) o 4.12-(b). En la captura de pantalla se muestran las señales GSM del enlace descendente, a 955.1 MHz y 958.4 MHz, respectivamente. Este tipo de análisis es realmente útil en estas señales ya que se aprecian las ráfagas de actividad en los canales a través del tiempo. Esto está directamente relacionado con la transmisión de tramas que forman el enlace TDMA entre usuarios y estaciones móviles.



(a)



(b)

control\_panel

**Centre Frequency** **Gain**

Enter a frequency here: 940 MHz Enter a gain value here: 50 dBs

Use slider to adjust centre frequency: Use slider to adjust gain:

Or choose a region from the following:

FM Radio (88-108MHz)  DAB Radio (175-240MHz)

Digital TV (470-862MHz)  GSM Mobile Signals (800-1000MHz)

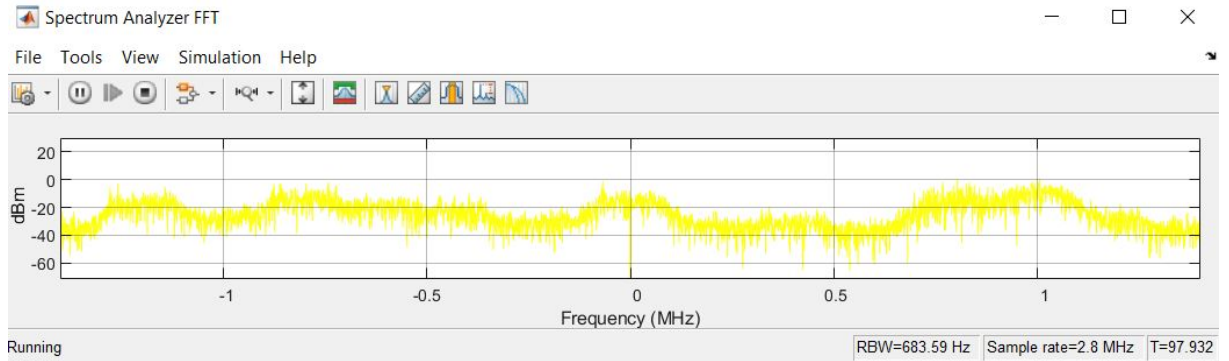
Start Simulation

Stop Simulation

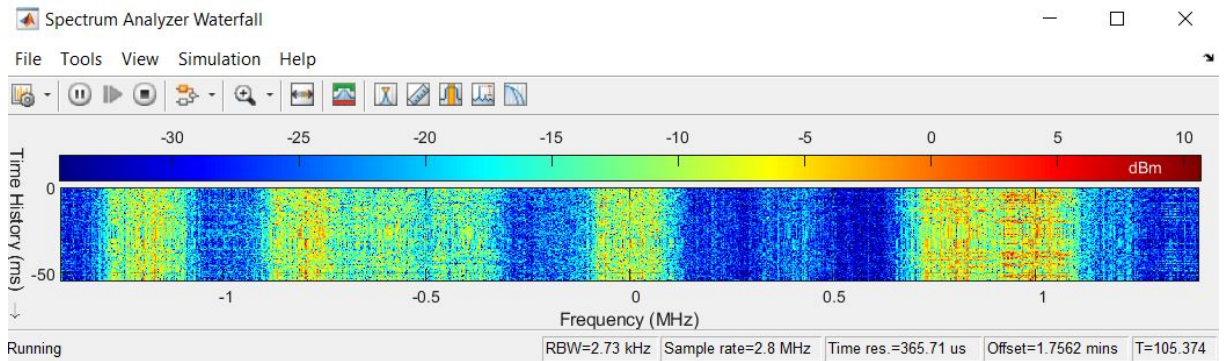
Current frequency: 955.1 MHz Current gain: 50 dBs

(c)

Figura 4.11 Actividad GSM en 955.1 MHz.



(a)



(b)

The screenshot shows a control panel titled 'control\_panel'. It has two main sections: 'Centre Frequency' and 'Gain'.  
 Under 'Centre Frequency', there is a text input field with '940' and 'MHz' next to it. Below it is a slider control. At the bottom, it says 'Current frequency: 958.4 MHz'.  
 Under 'Gain', there is a text input field with '50' and 'dBs' next to it. Below it is a slider control. At the bottom, it says 'Current gain: 50 dBs'.  
 On the right side, there are two buttons: 'Start Simulation' and 'Stop Simulation'.  
 At the bottom, there are four radio button options: 'FM Radio (88-108MHz)', 'DAB Radio (175-240MHz)', 'Digital TV (470-862MHz)', and 'GSM Mobile Signals (800-1000MHz)'. The 'GSM Mobile Signals' option is selected.

(c)

Figura 4.12 Actividad GSM en 958.4 MHz.

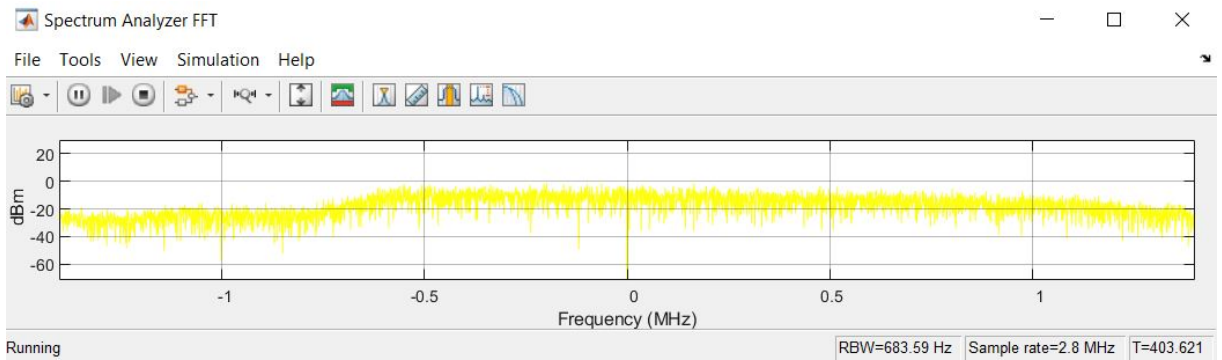
## 4.6 Análisis de señales móviles UMTS (3G)

El Sistema Universal de Telecomunicaciones Móviles (UMTS, *Universal Mobile Telecommunications System*) es el estándar usado en telefonía móvil de tercera generación y sucesor de GSM. Los canales se caracterizan por tener un ancho de 5 MHz, donde las señales son enviadas a través de toda la banda de frecuencias en forma de señales de *espectro ensanchado*. Estas características ofrecían una mayor evolución de las comunicaciones móviles, orientada para grandes cantidades de paquetes de datos.

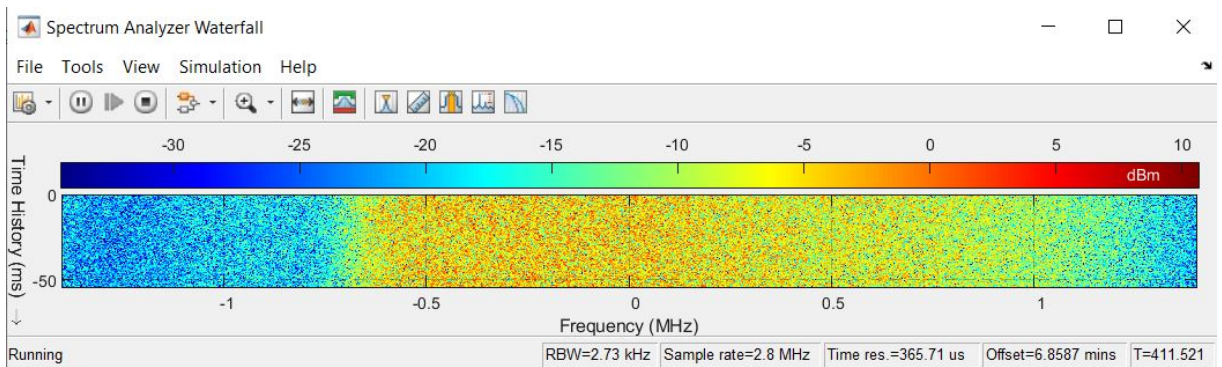
El acceso en el que está basado este estándar, es el acceso múltiple por división de código de banda ancha (conocido como WCDMA, *Wideband Code Division Multiple Access*). El ancho de banda más amplio permitió un aumento de velocidades de datos, manteniendo la eficiencia de transmisión al permitir el acceso de múltiples usuarios al mismo tiempo. En lugar de depender de los intervalos de tiempo, el teléfono móvil de cada usuario puede determinar qué señales de transmisión son para él haciendo coincidir su código individual con ellas.

La mayoría de las señales UMTS son transmitidas desde los 1700 MHz en adelante, pero aún hay un número de señales 3G que se transmiten entre los 800 MHz y los 900 MHz. En la Figura 4.13 se ilustra un ejemplo de señal UMTS.

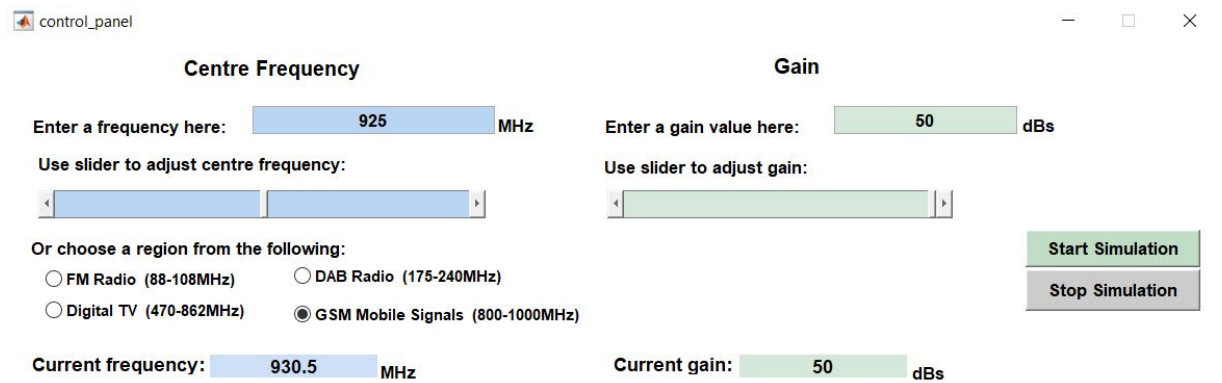
A la izquierda de la Figura 4.13-(b) se aprecia una banda de guarda de UMTS. Estas bandas no contienen ningún tipo de información cuyo único objetivo es separar la actividad entre canales. También puede apreciarse en la Figura 4.13-(a). Además, puede apreciarse que, a diferencia de GSM, no existen interrupciones a lo largo del tiempo. Esto se debe a que los usuarios son distinguidos por código y no por intervalos de tiempo, lo que se traduce en una actividad e intensidad de señal constante.



(a)



(b)



(c)

Figura 4.13 Señal UMTS.

## 4.7 Análisis de señales móviles LTE (4G)

Con los años, para satisfacer la demanda de usuarios en las comunicaciones móviles, surge uno de los estándares más recientes. Este es el conocido como 4G o LTE (*Long Term Evolution*), y se construyó sobre la infraestructura existente, centrandó la atención en incrementar las tasas de datos. Según el estándar, las velocidades de descarga de datos pueden alcanzar hasta 1 Gbps para dispositivos estáticos.

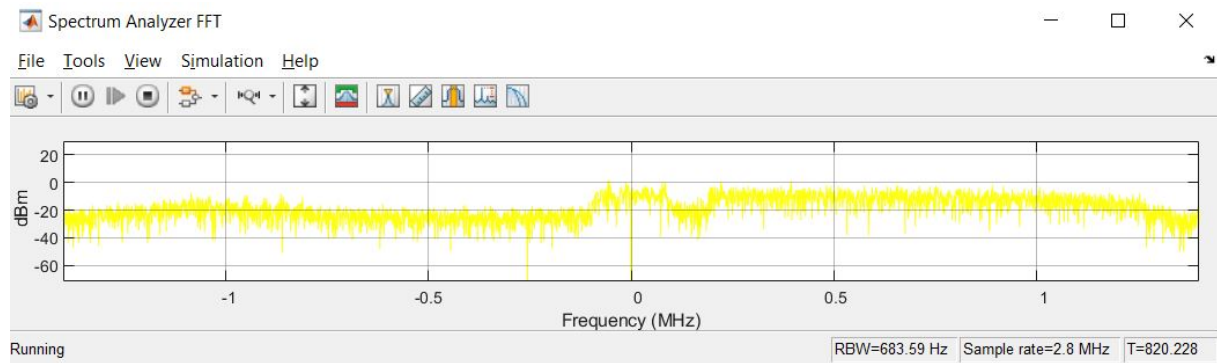
El principal objetivo de LTE es ofrecer un beneficio a largo plazo por su progresión a una técnica de multiplexión estadística conocida como acceso múltiple por división de frecuencias ortogonales (OFDMA, *Orthogonal Frequency Division Multiple Access*). Mediante la división de tiempo y frecuencia, se proporciona este acceso múltiple, asignando recursos a cada usuario, con el fin de



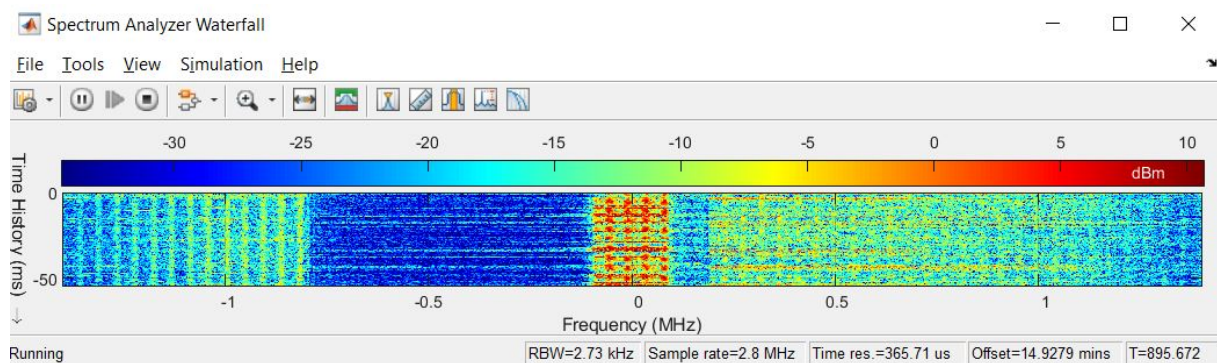
proporcionar el mayor rendimiento posible de la forma más eficiente.

Los canales de transmisión y recepción para las señales de LTE se pueden encontrar entorno a 800, 900, 1800 y 2600 MHz. Estas señales pueden ocupar varios anchos de banda que van desde los 1.4 MHz hasta los 20 MHz en el espectro.

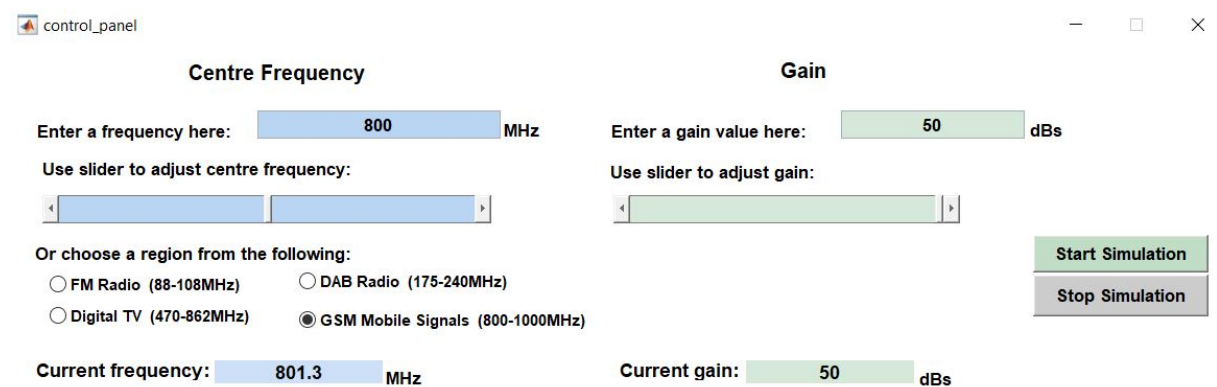
En esta sección, dadas las limitaciones del dongle, se centra el análisis de estas señales entre los 700 MHz y 900 MHz. A continuación, puede observarse en la Figura 4.14 el inicio de una señal LTE, y en la Figura 4.15 la continuación de otra señal LTE distinta (puede comprobarse viendo la frecuencia actual en las Figuras 4.14-(c) y 4.15-(c)).



(a)

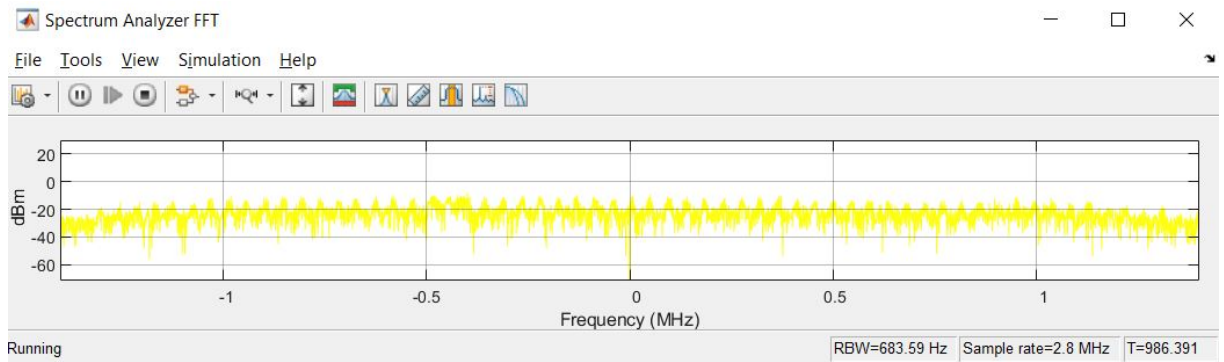


(b)

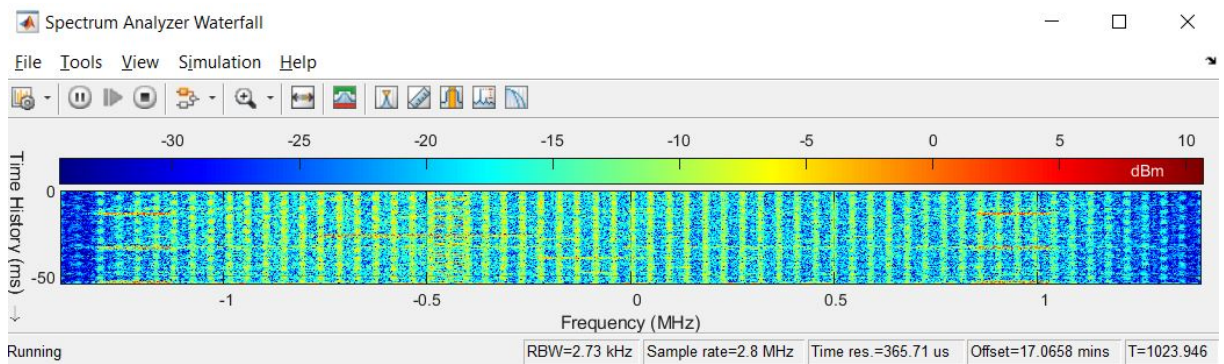


(c)

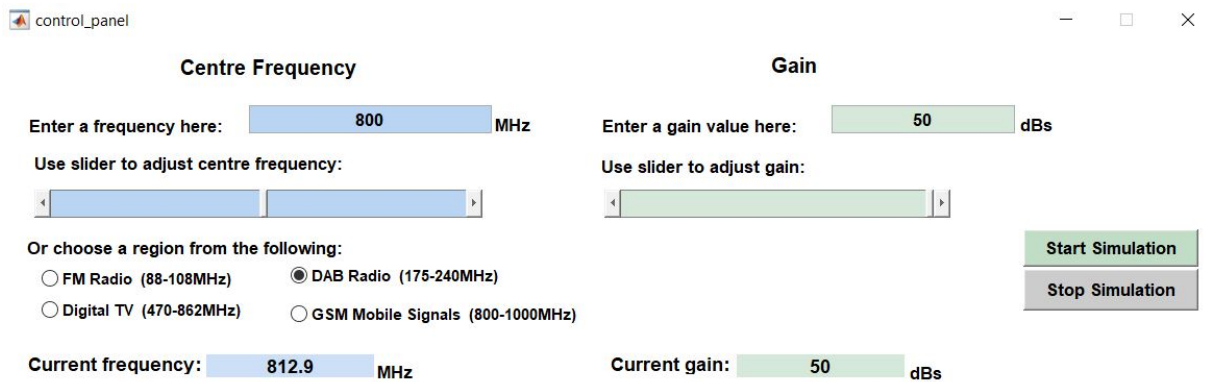
Figura 4.14 Comienzo de una señal LTE.



(a)



(b)



(c)

Figura 4.15 Continuación de una señal LTE.

Prestando atención a cualquiera de los espectrogramas de las Figuras 4.14-(b) y 4.15-(b), puede apreciarse con claridad que las señales LTE tienen numerosas portadoras OFDM de alta potencia que están igualmente espaciadas en el ancho de banda del canal. Dada la flexibilidad que presentan los canales respecto al ancho de banda, es más fácil reconocer este tipo de señales por la presencia de OFDM que por su ancho de banda. Si bien es cierto, esto sólo sería válido para los canales descendentes, puesto que en los canales ascendentes, LTE usa SC-FDMA (*Single Carrier - Frequency Division Multiple Access*).

## 4.8 Búsqueda de señales de TV digital DVB-T

Históricamente, los canales de televisión han tenido un número diferente de anchos de banda, por eso, cuando el estándar DVB-T (*Digital Video Broadcasting - Terrestrial*) fue creado, se designó para dar soporte a los canales de 5, 6, 7 y 8 MHz.

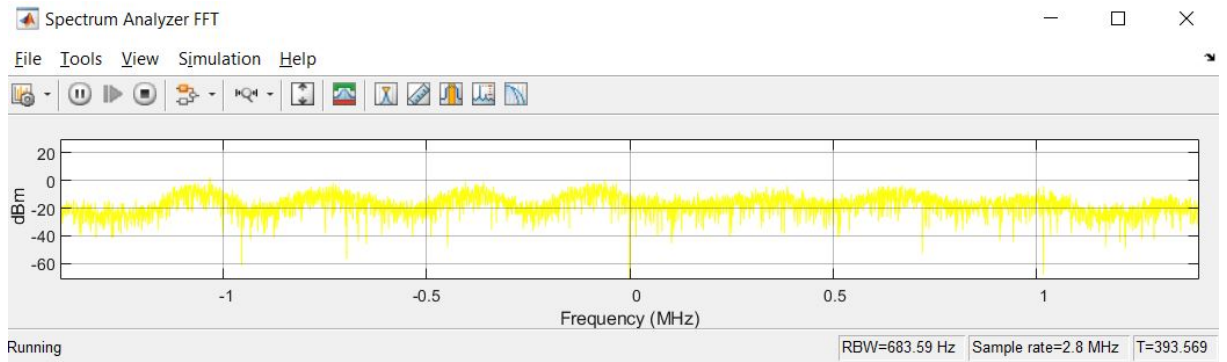
El aumento de la demanda de la alta calidad de vídeo, trajo consigo la creación de nuevas modulaciones y nuevas técnicas de multiplexación y codificación, y así poder cumplir con los requisitos de transferencia de datos que se requería. Para disminuir las tasas de errores, se añadió una gran cantidad de redundancia a cada trama de información a través de números de estados de códigos, y a pesar de que esto incrementaba los datos que debían transmitirse, añadía a las señales la corrección de errores hacia delante (FEC, *Forward Error Correction*). Este tipo es un mecanismo que permite la corrección de errores en el receptor sin necesidad de retransmisión, usado habitualmente en comunicaciones vía satélite.

Por otro lado, la información se modula usando COFDM (*Coded Orthogonal Frequency Division Multiplexing*). Esta técnica de banda ancha transmite la información del código en mil o más portadoras de RF ortogonales, lo que reducen aún más los errores resultantes de interferencias o ruido, mientras que puede transmitir grandes cantidades de información con éxito.

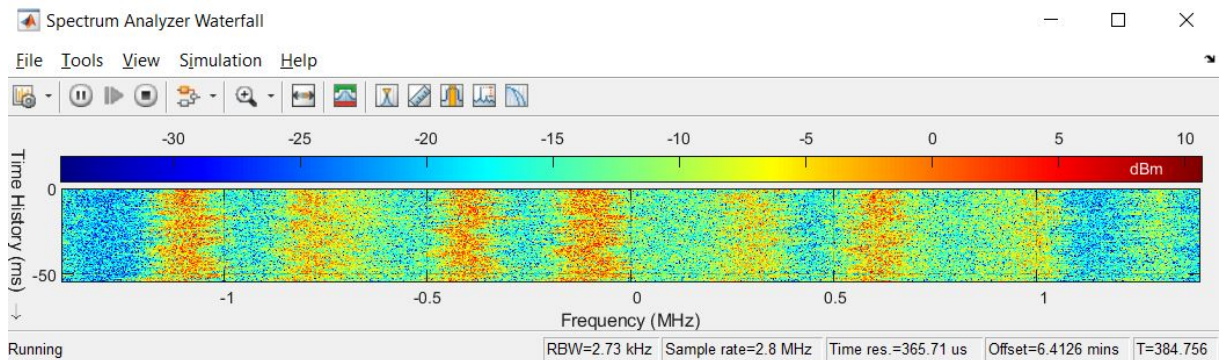
En la Figura 4.17 puede apreciarse un ejemplo de estas señales en el espectro. Concretamente, se trata de los canales de televisión que se muestran en la Figura 4.16.

1	<b>La 1</b>	22 482 MHz	SI	SI	SI	NAC	Emisión en <b>16:9</b> .	📺
1	<b>La 1 HD</b>	22 482 MHz	SI	SI	SI	NAC	Emisión en <b>HD</b> .	📺
2	<b>La 2</b>	22 482 MHz	SI	SI	SI	NAC	misión en <b>16:9</b> .	📺
2	<b>La 2 HD</b>	22 482 MHz	SI	SI	SI	NAC	Emisión en <b>HD</b> .	📺
24h	<b>24h</b>	22 482 MHz	SI	SI	NO	NAC	Emisión en <b>16:9</b> .	📺
24h	<b>24h HD</b>	22 482 MHz	SI	SI	NO	NAC	Emisión en <b>HD</b> .	-
rne	<b>Radio Nacional</b>	22 482 MHz	SI	-	-	NAC	Radio	
rne	<b>Radio 5 RNE</b>	22 482 MHz	SI	-	-	NAC	Radio	

**Figura 4.16** Canales de televisión y radio en 482 MHz. Información obtenida de <https://www.tdt1.com/canales-cadiz/>.



(a)



(b)

(c)

Figura 4.17 Señales de Televisión Digital DVB-T.

## 4.9 Buscando la actividad de un teléfono móvil en el espectro

Hasta ahora los experimentos no han ido más allá de identificar varios tipos de señales a lo largo de todo el espectro, con el fin de demostrar que, a través del equipo de trabajo seleccionado y haciendo uso de SDR, se pueden obtener e identificar señales con las que se hace un uso prácticamente a diario. Pero, a pesar de ello, no se ha podido indentificar más parámetros de la señal que los que ofrece el analizador de espectro (frecuencia, ganancia, potencia y forma de la señal). Por este motivo, en esta sección se da pie a buscar una señal forzada, con un teléfono móvil, en el espectro. Concretamente, se va a reconocer una llamada entre dos teléfonos móviles usando la red GSM (2G).

El primer paso para llevar a cabo este experimento consta de otros dos, el primero de ellos más sencillo:

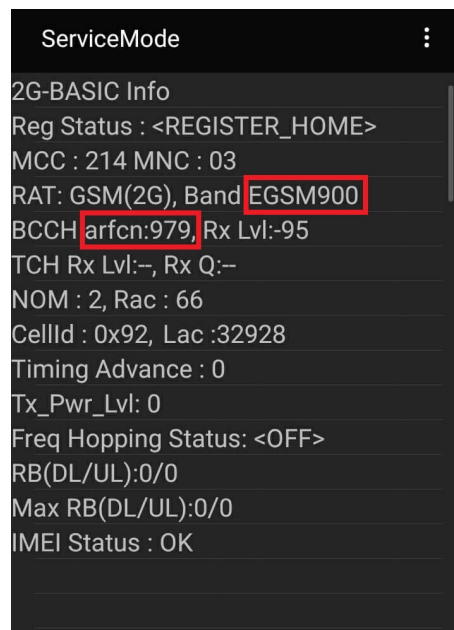
1. Configurar los teléfonos móviles con la red GSM.

*Ajustes > Redes móviles > Preferencia de red > Sólo 2G*

2. Conocer la frecuencia del teléfono móvil que realiza la llamada.

Dado que este paso es menos usual y el acceso a la configuración para conocer este dato no se realiza de igual forma en todos los dispositivos del mercado, aclarar que se ha llevado a cabo en un teléfono Samsung<sup>®</sup>.

- En primer lugar, entrar en "ServiceMode", marcando en el teclado la secuencia \*#0011#
- En la Figura 4.18 se muestra el resultado de ejecutar el punto anterior



**Figura 4.18** ServiceMode.

De la imagen se obtienen los valores "ARFCN: 979" y "BAND: EGSM900".

ARFCN es el número de canal absoluto de radiofrecuencia (*Absolute Radio-Frequency Channel Number*), que es un código que especifica un par de de frecuencias portadoras, una para la transmisión (uplink) y otra para la recepción (downlink). Por otro lado, BAND indica la banda de GSM donde opera el teléfono móvil, en este caso: E-GSM 900 (que opera entre los 880 MHz y los 914.8 MHz en el uplink y entre los 925 MHz y los 959.8 MHz en el downlink).

Obtenidos esos valores, para obtener el valor de la frecuencia de trabajo puede aplicarse la siguiente fórmula:

$$f_{gsm} = ARFCN \times (f_c + f_b + f_o) \text{ MHz} \quad (4.2)$$

donde

$f_{gsm}$  = Frecuencia central del canal GSM (MHz)

$f_c$  = Espacio del canal (MHz)

$f_b$  = Frecuencia base (MHz)

$f_o$  = Frecuencia de offset (MHz)

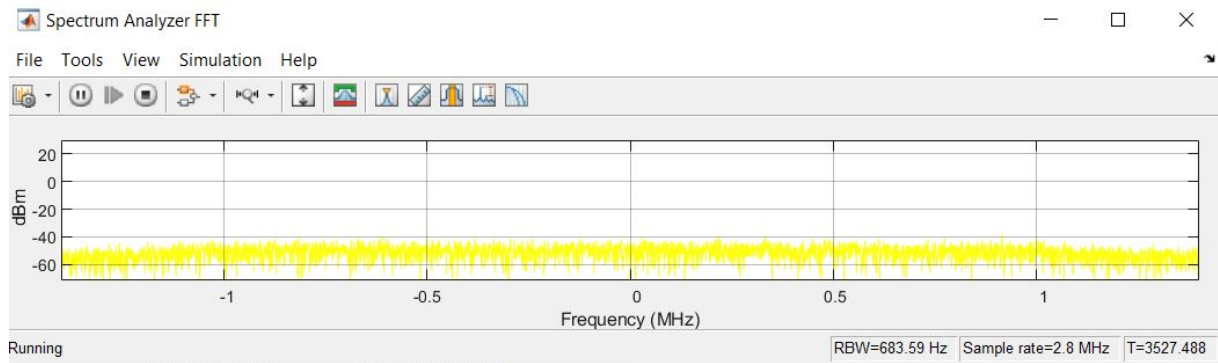
Otra alternativa más sencilla se basa en obtener las frecuencias portadoras de una tabla, mostrada en Figura 4.19. De donde se obtiene que  $f_{gsm} = 881$  MHz para el uplink.

E-GSM 900

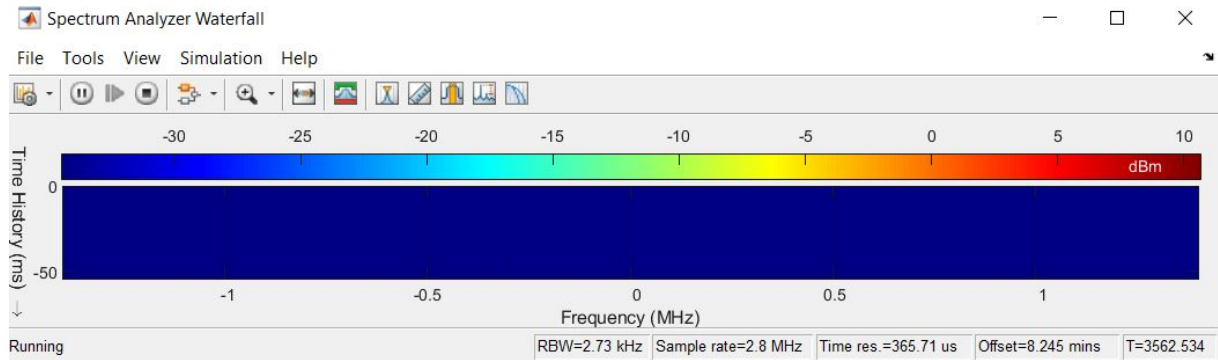
ARFCN	Frequency (MHz)	
	Downlink	Uplink
975	925.2	880.2
976	925.4	880.4
977	925.6	880.6
978	925.8	880.8
979	926	881
980	926.2	881.2
981	926.4	881.4
982	926.6	881.6

Figura 4.19 Obtenida en [https://www.sqimway.com/gsm\\_arfcn.php](https://www.sqimway.com/gsm_arfcn.php).

Una vez averiguada la frecuencia a la que opera el dispositivo móvil de interés, se fija dicha frecuencia en la GUI. Antes de realizar la llamada, como era de esperar, el espectro no tiene ninguna actividad. Así lo refleja la Figura 4.20.



(a)



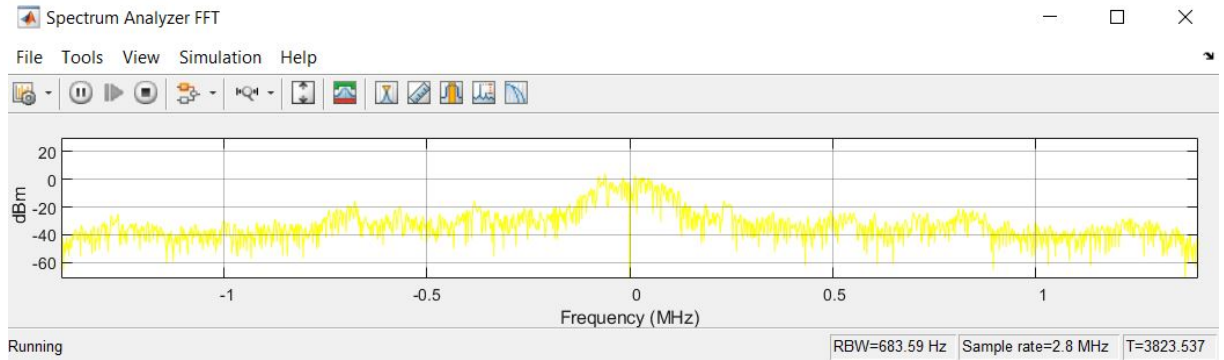
(b)

The screenshot shows the 'control\_panel' GUI. It has two main sections: 'Centre Frequency' and 'Gain'.  
 - Under 'Centre Frequency': 'Enter a frequency here:' is set to 881 MHz. Below it is a slider labeled 'Use slider to adjust centre frequency:'.  
 - Under 'Gain': 'Enter a gain value here:' is set to 40 dBs. Below it is a slider labeled 'Use slider to adjust gain:'.  
 - Below these sections, there are radio buttons for selecting a region: 'FM Radio (88-108MHz)', 'DAB Radio (175-240MHz)', 'Digital TV (470-862MHz)', and 'GSM Mobile Signals (800-1000MHz)'. The 'GSM Mobile Signals' option is selected.  
 - At the bottom, there are two buttons: 'Start Simulation' and 'Stop Simulation'.  
 - At the very bottom, 'Current frequency:' is 881 MHz and 'Current gain:' is 40 dBs.

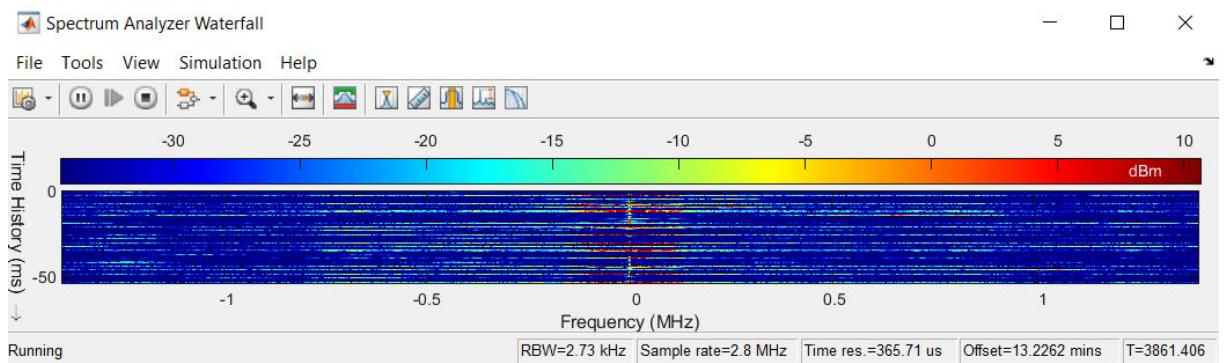
(c)

Figura 4.20 Medida del espectro antes de realizar la llamada sobre GSM.

Cuando se realiza la llamada telefónica entre dispositivos, se observa cómo aparecen las señales GSM que se identificaron en la Sección 4.5. En la Figura 4.21 se refleja la actividad resultante.



(a)



(b)

**Centre Frequency**

Enter a frequency here:  MHz

Use slider to adjust centre frequency:

**Gain**

Enter a gain value here:  dBs

Use slider to adjust gain:

Or choose a region from the following:

FM Radio (88-108MHz)   
  DAB Radio (175-240MHz)  
 Digital TV (470-862MHz)   
  GSM Mobile Signals (800-1000MHz)

Current frequency:  MHz

Current gain:  dBs

(c)

Figura 4.21 Medida del espectro durante la llamada sobre GSM.



## 4.10 Barrido del espectro

En esta sección se va a ejecutar un barrido de frecuencias con el NESDR SMArTee, para recibir todas las señales transmitidas en el rango de frecuencias donde opera su sintonizador que, como se introdujo en la Sección 3.1.1 del Capítulo 3, opera entre 25 MHz y 1750 MHz aproximadamente.

Dicho experimento se lleva a cabo sintonizando repetidamente el dongle a una frecuencia central diferente, con el fin de obtener la información espectral en todo el rango de frecuencias. Dadas las limitaciones que se comentaron al comienzo de este capítulo, en la Sección 4.3, para que no se produzca una pérdida de información y para confiar un preciso muestreo, se establece que cada frecuencia central ( $f_c$ ) será 2.8 MHz mayor que la anterior  $f_c$ , de este modo:

$$f_{c(i+1)} = f_{c(i)} + 2.8\text{MHz} \quad (4.3)$$

Para llevar a cabo lo comentado, se ejecuta el código A.2, correspondiente al Apéndice A. El resultado se muestra en las Figuras 4.22 y 4.23, barridos correspondientes a dos emplazamientos distintos dentro de la misma localidad.

Por defecto, de acuerdo a los parámetros configurados en el código, se configura para que el NESDR SMArTee muestree a 2.8 MHz. Esto se traduce en que la señal recibida se caracterizará por tener un ancho de banda de igual a la frecuencia de muestreo, conteniendo información espectral desde  $f_c - 1.4$  MHz hasta  $f_c + 1.4$  MHz, según se establecía en la Ecuación 4.1.

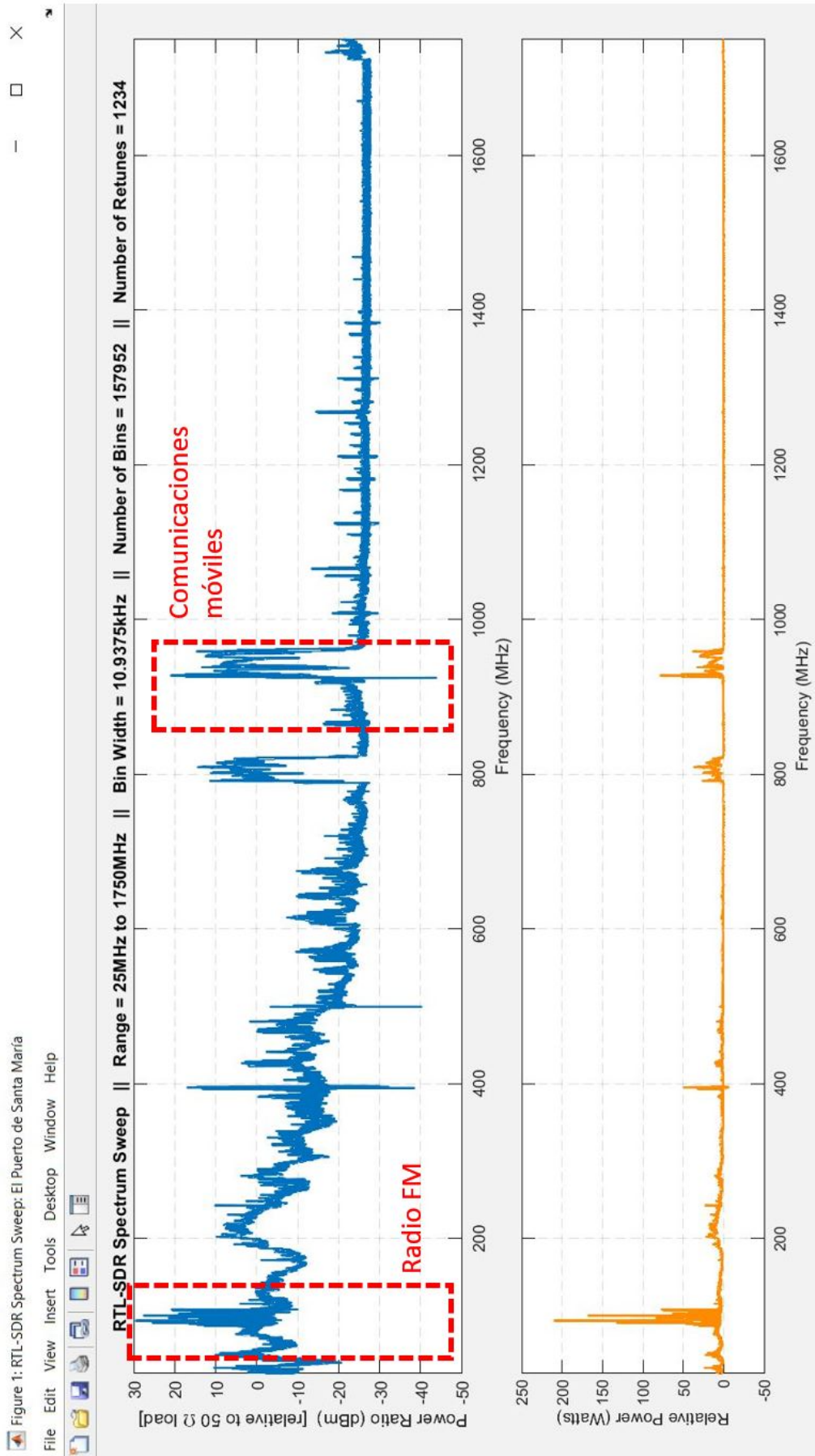


Figura 4.22 Barrido del espectro. Emplazamiento 1.

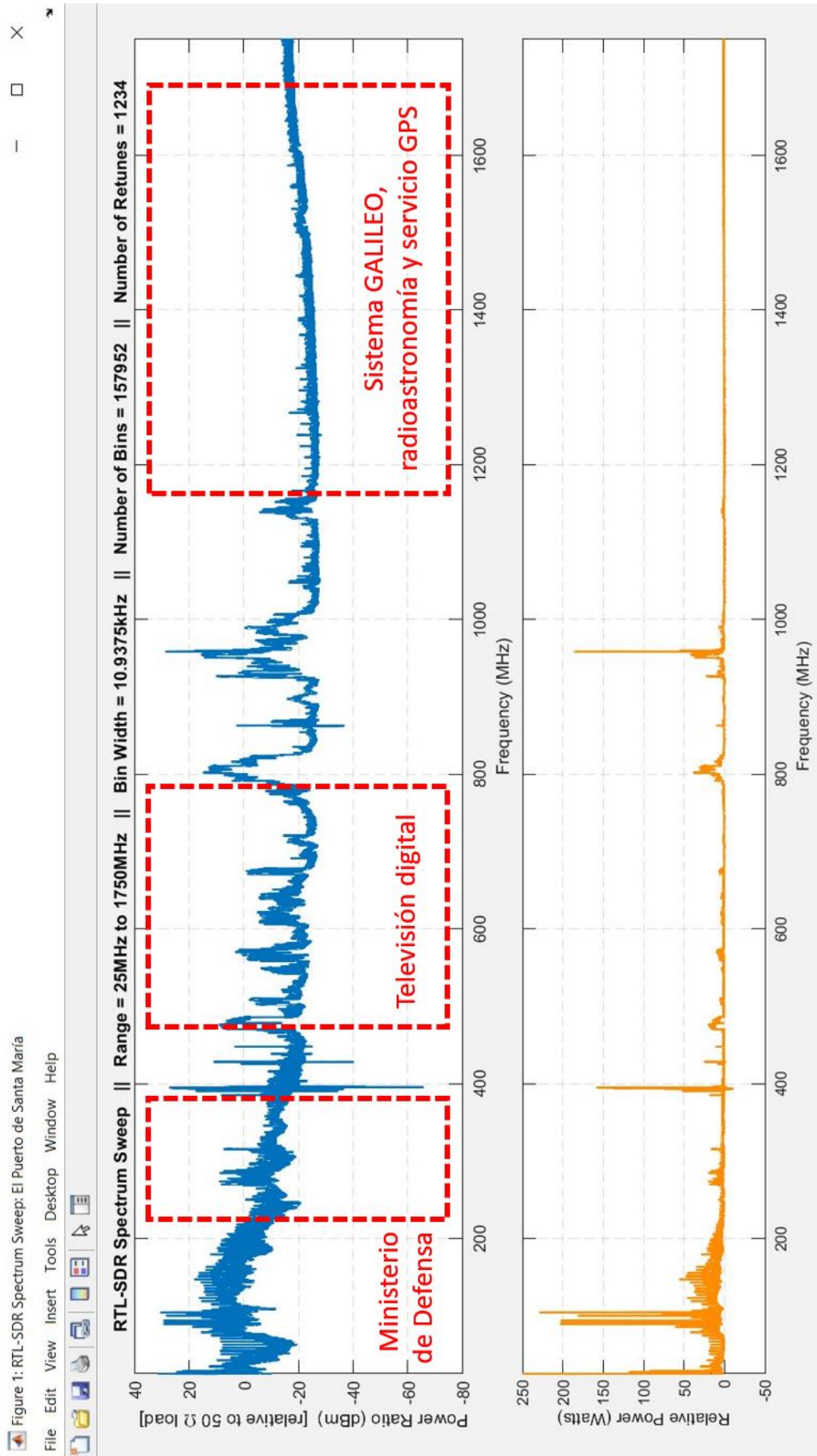


Figure 1: RTL-SDR Spectrum Sweep: El Puerto de Santa María

Figura 4.23 Barrido del espectro. Emplazamiento 2.

Para tener una mejor interpretación de las gráficas obtenidas al hacer el barrido del espectro, se recurre al Cuadro Nacional de Atribución de Frecuencias (CNAF) aprobado a través de la orden ETU/1033/2017 y que entró en vigor el 28 de Octubre de 2017 (BOE-A-2017-12318).

**Tabla 4.2** Servicios de comunicación prestados en el espectro.

<b>Frecuencia</b>	<b>Servicios de comunicación</b>
(25 - 68) MHz	Ministerio de Defensa (UN-14, UN-15), aplicaciones ICM (UN-11, UN-13), radiobúsqueda (UN-12), radioaficionados (UN-100)
(87.5 - 108) MHz	Estaciones de radio FM (UN-17)
(130 - 144) MHz	Servicio móvil aeronáutico (UN-18, UN-19)
(146 - 195) MHz	Servicio móvil terrestre, DAB (UN-26, UN-98)
(195 - 223) MHz	Radiodifusión sonora digital VHF (UN-96)
(223 -235) MHz	Ministerio de Defensa, servicio móvil terrestre (UN-27)
(235 - 399.9) MHz	Ministerio de Defensa, servicio de radionavegación aeronáutica (UN-28)
(406 - 470) MHz	Servicio móvil por satélite, radioaficionados (UN-31), aplicaciones ICM (UN-32, UN-30)
(470 - 790) MHz	Televisión digital (UN-36)
(870 - 960) MHz	Comunicaciones móviles (UN-40, UN-41)
(1164 - 1710) MHz	Sistema GALILEO (UN-122), Ministerio de Defensa (UN-45), radioastronomía (UN-47), servicio GPS (UN-99)

En la Tabla 4.2 se recogen las frecuencias correspondientes a los principales servicios de comunicación que aparecen en los barridos de las Figuras 4.22 y 4.23. Cabe destacar la presencia de más servicios que se han omitido por una cuestión de importancia de presencia en el espectro.

## 4.11 Decodificador y receptor de radio FM estéreo

En última instancia, tiene lugar un experimento que trata de exponer la utilidad y versatilidad de la tecnología SDR en un escenario tan rutinario como lo es un receptor radio FM. Para ello se va a construir un modelo en Simulink a partir del cual se proporcionará una señal de audio estéreo a los altavoces del ordenador o a unos auriculares conectados al mismo.

En primer lugar se parte del modelo receptor de radio FM mono, mostrado a continuación. Debido a que la señal multiplexada estéreo (MPX) tiene componentes de frecuencia desde los 0 kHz hasta los 53 kHz, no es posible diezmar la salida de la señal demodulada de FM del discriminador de frecuencia a una frecuencia de muestreo de 48 kHz (como se haría en el receptor mono), puesto que aparecería aliasing. Es por ello que debe escogerse un valor, como mínimo, de dos veces 53 kHz. En este caso, se ha optado por escoger un valor de 120 kHz en el filtro FIR de diezrado.

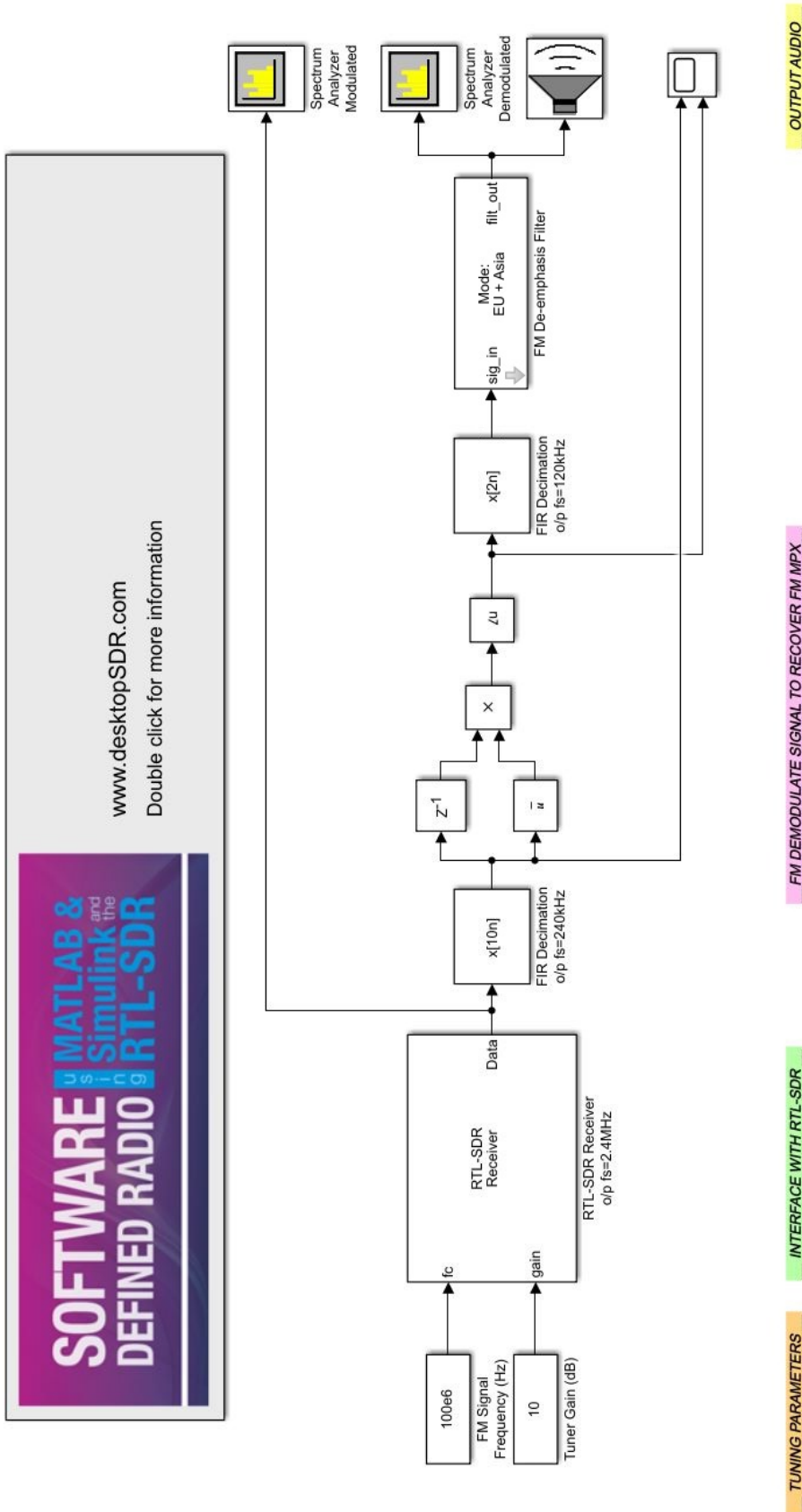


Figura 4.24 Receptor de radio FM mono.

A continuación, se incorpora el bloque *FM MPX Demultiplexer*, localizado según muestra la Figura 4.25, entre el filtro FIR modificado en el paso anterior y el filtro *FM De-emphasis*.

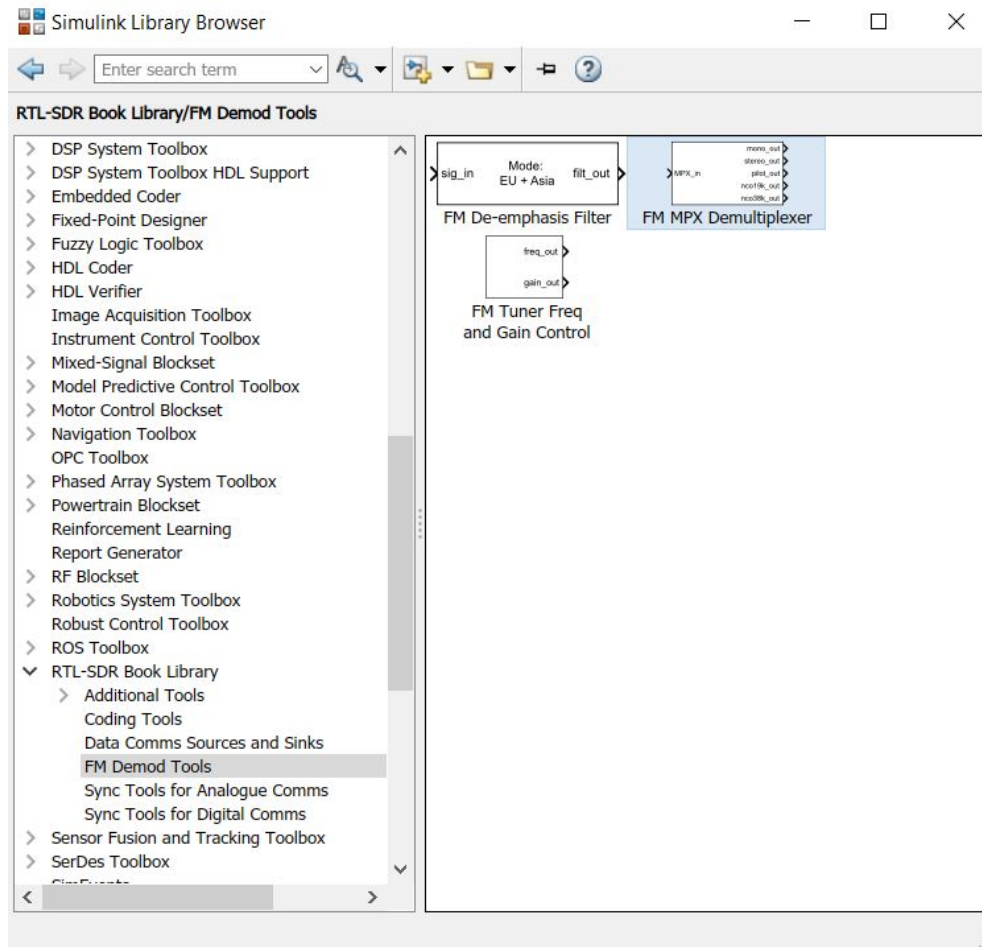


Figura 4.25 Bloque *FM MPX Demultiplexer*.

A continuación, para implementar el decodificador FM estéreo se precisan de dos bloques *FIR Rate Conversion* (Figura 4.27), dos bloques *Add* y un bloque *Matrix Concatenate*, y se disponen como muestra la siguiente figura.

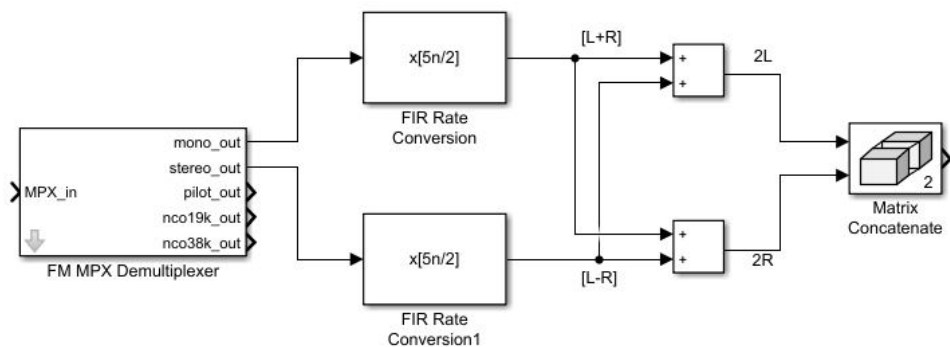
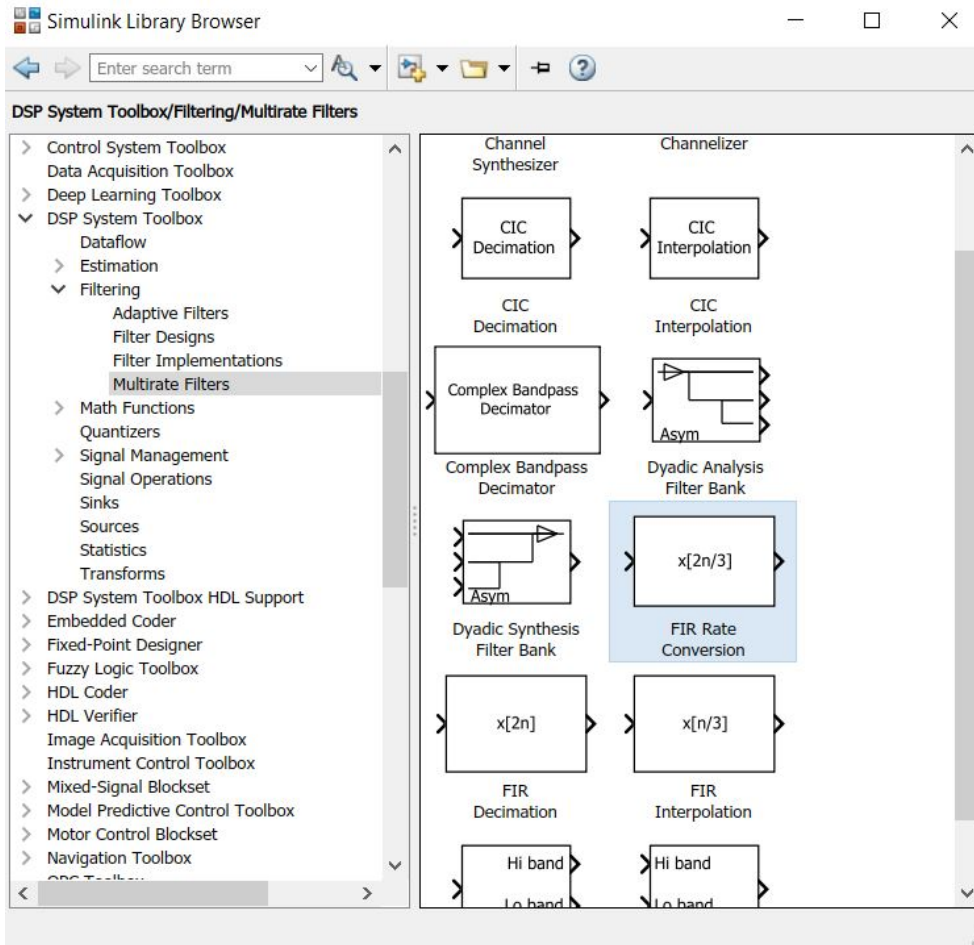
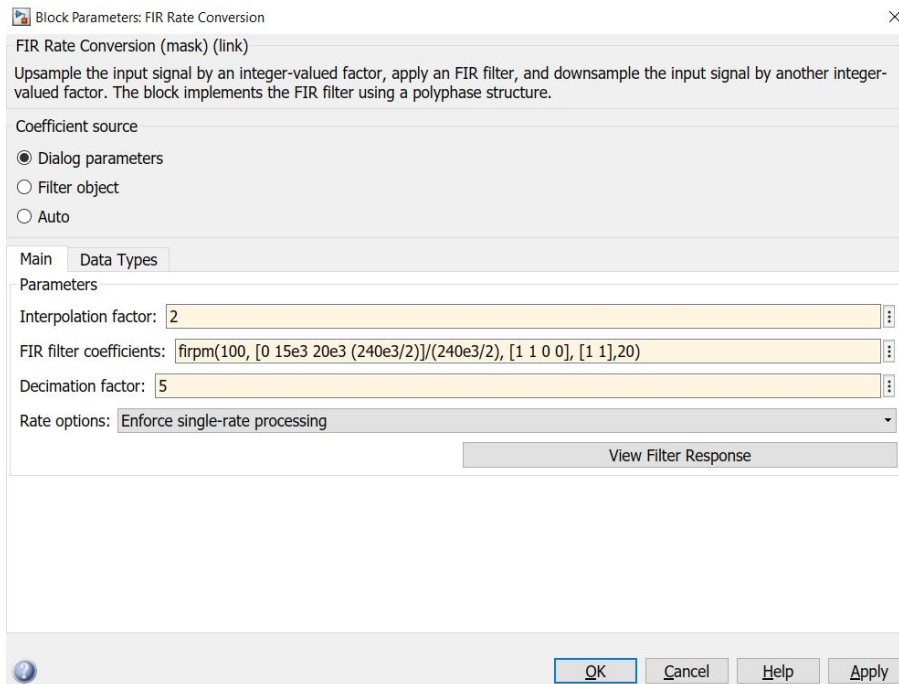


Figura 4.26 Decodificador FM estéreo.



(a)



(b)

Figura 4.27 Bloque *FIR Rate Conversion* y parámetros de configuración.

El receptor completo quedaría:

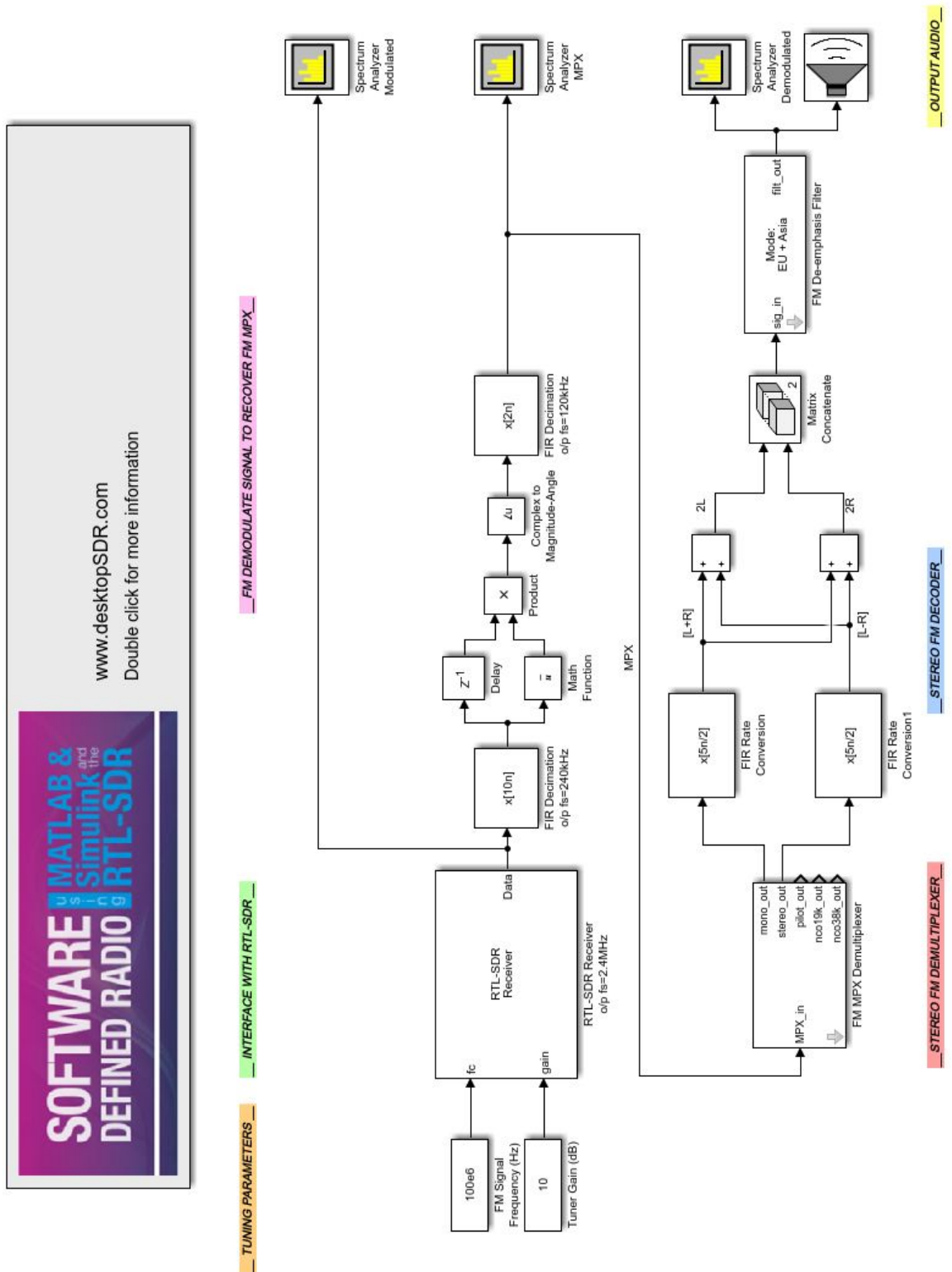
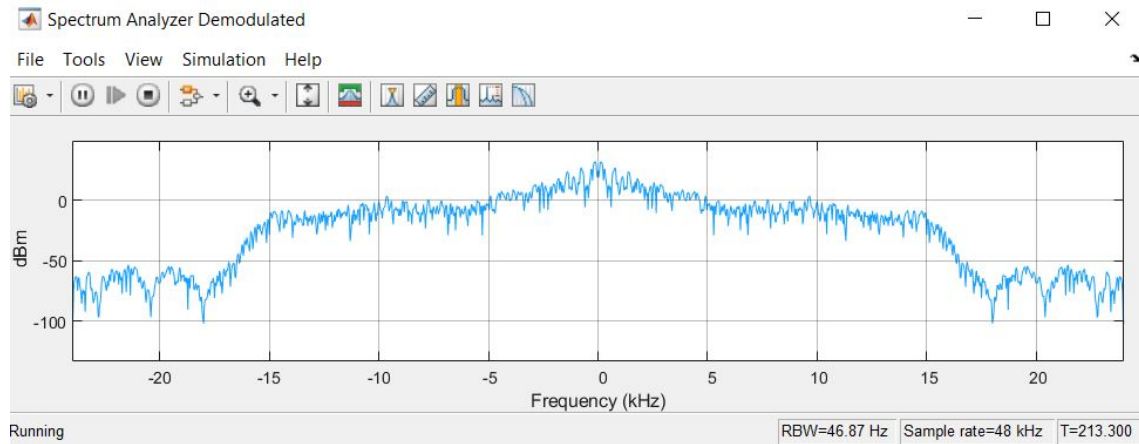


Figura 4.28 Receptor de radio FM estéreo.

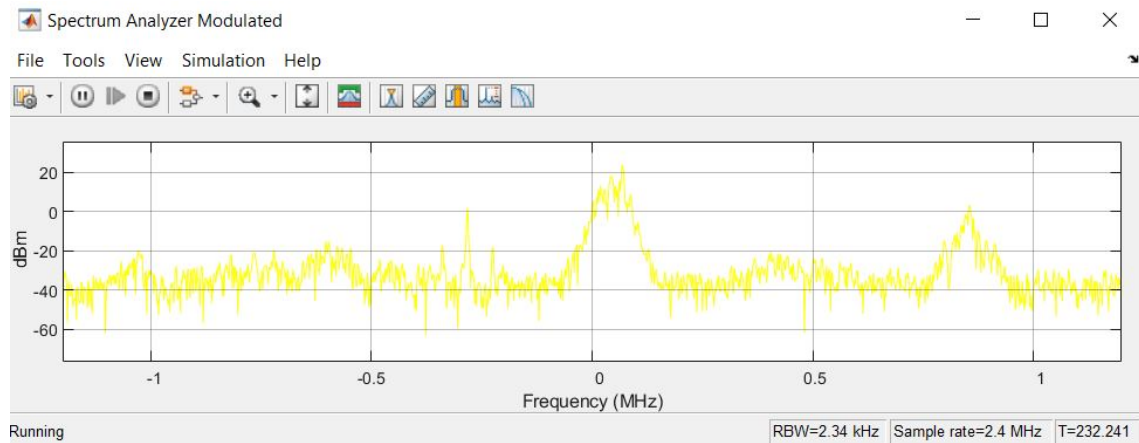


Llegados a este punto, sólo quedaría simular el diagrama de bloques completo, estableciendo, previamente,  $SimulationStopTime = inf$ .

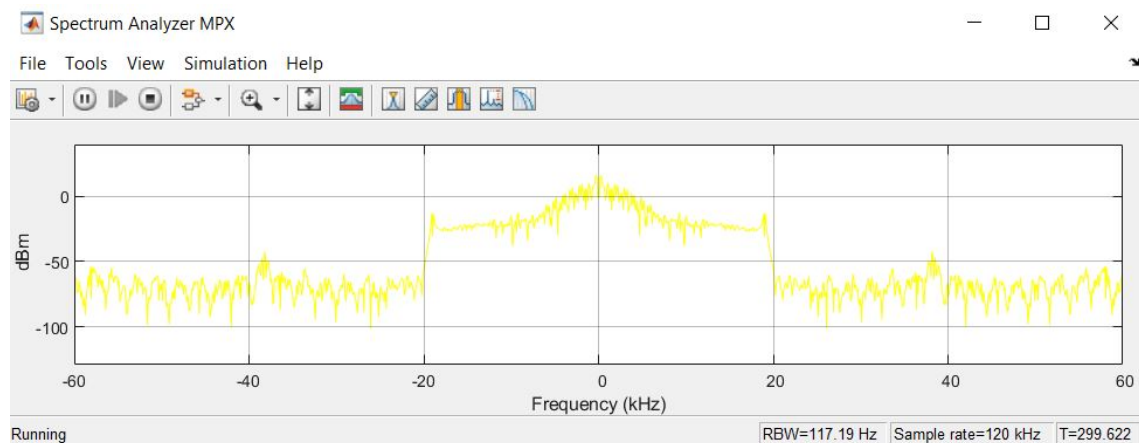
El resultado de la simulación se muestra en la Figura 4.29 donde se ha sintonizado la frecuencia 89.4 MHz (Figura 4.30), correspondiente a Los 40 Principales.



(a)



(b)



(c)

**Figura 4.29** Resultado simulación radio FM estéreo.

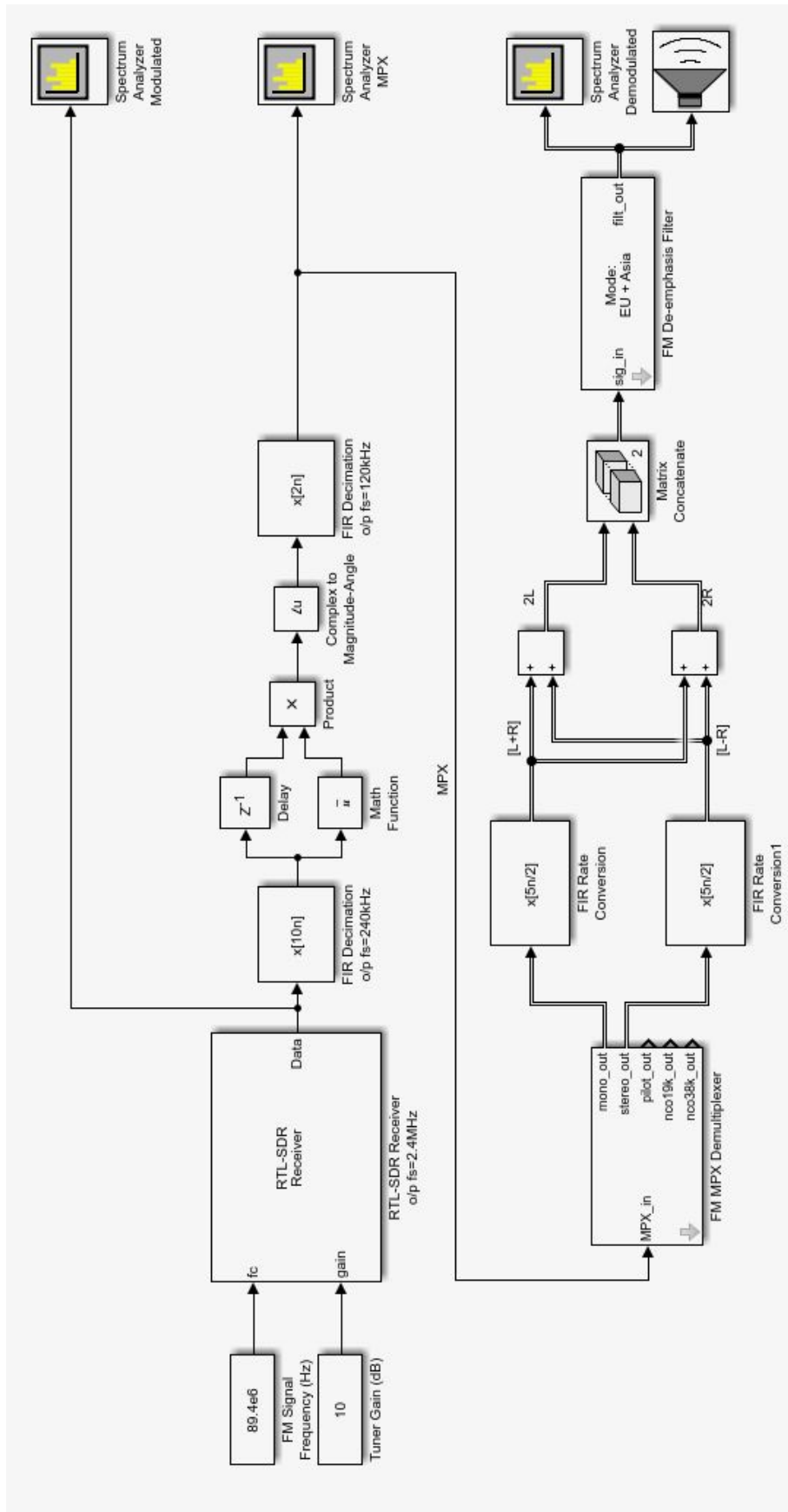


Figura 4.30 Receptor en 89.4 MHz.

## 5 Conclusiones

---

*Sin importar lo mala que parezca ser la vida, siempre hay algo que puedes hacer, y tener éxito. Mientras haya vida, hay esperanza.*

STEPHEN HAWKING, 2006

A lo largo del presente proyecto se han tratado de poner de manifiesto, aunque de manera indirecta, algunas de las muchas ventajas que posee la tecnología Software Defined Radio. Concretamente, en la Sección 2.1.3, donde se expuso el mercado actual de esta tecnología, ya se pudieron deducir algunas de ellas.

A través de un modelo receptor radio se han captado todo tipo de señales, que se corresponden con comunicaciones que están presentes en el día a día, para su análisis y comprobación teórica. Además, se ha dado un paso más allá en dos de las herramientas más usuales como lo son las comunicaciones móviles y la radio FM, tratando de ofrecer una visión más cercana de esta tecnología, que aún tiene mucho camino por delante. En otras palabras, en este documento se ha demostrado la posibilidad de implementar, prácticamente, cualquier sistema de radiofrecuencia siguiendo la filosofía de la Radio Definida por Software con sólo un dispositivo SDR y unos conocimientos básicos previos de procesado de señal y de programación.

Reflexionando sobre el trabajo realizado, se pueden deducir algunas afirmaciones en relación a las ventajas mencionadas al principio. La primera ya se adelantaba en el párrafo anterior, y es el hecho de implementar cualquier sistema RF. El principal y mayor inconveniente de este tipo de sistemas es la gran inversión económica inicial que hay que realizar, puesto que, actualmente, se trata de equipos hardware en su totalidad, lo que incrementa notablemente dicha inversión. SDR se consolida aprovechándose de esta situación, ya que usando dispositivos de bajo coste (de las características de los empleados en este proyecto) se han obtenido prestaciones satisfactorias y fiables, como se puede observar a lo largo del Capítulo 4. Si bien es cierto que hay determinadas aplicaciones que no han sido objeto de este proyecto, tales como señales satelitales, radioastronomía, comunicaciones aeronáuticas, etc., hay que señalar que podrían haberse abordado de igual forma con ayuda del equipo necesario, sin exceder mucho más económicamente.

Extrapolando esta cuestión a sectores de gran importancia en la industria, también supondría un gran ahorro en la inversión económica de los equipos de comunicaciones. Si la Radio Definida por Software estuviera extendida en su totalidad a todos los niveles, tomando como ejemplo el caso en el que una empresa de telecomunicaciones que quiera hacer una inversión de una infraestructura para ofrecer un servicio como operadora, el hecho de hacer uso de esta tecnología no sólo implicaría ahorros en el montaje (ya que parte de la misma estaría definida por software), sino que también se ahorraría en mantenimiento. No es lo mismo realizar un mantenimiento de estas infraestructuras

de manera presencial, y sabiendo que hay elementos que disponen de una vida útil limitada, a hacerlo en remoto gracias a la parte software. En un entorno más cercano, como lo sería el ámbito académico universitario, también supondría un ahorro económico notable gracias a la simulación de analizadores de espectro.

Por otro lado, otro aspecto que caracteriza esta tecnología es la flexibilidad, derivado de todo lo anterior. Cualquier persona con un mínimo de conocimientos en procesamiento de señal y con autoayuda de algún libro o tutorial, sería capaz de montar su propio equipo receptor.

Aunando todo lo anterior, se reflexiona sobre el impacto que pudiera tener SDR en algunas de las líneas futuras como IoT (*Internet Of Things*), 5G o *Software Defined Wifi* (este último con un futuro casi inmediato). En relación a IoT, existen prototipos de módulos o placas SDR para desarrolladores que ofrecen todo lo necesario para la realización de determinados diseños adaptados a aplicaciones de IoT. Todo ello con unas características a nivel de componentes bastante distintas a las que reúnen los dispositivos usados en este proyecto, pero que comparten la misma filosofía. Además, también se apoyan en SDR para estudiar algunas vulnerabilidades en dispositivos IoT.

En segundo lugar, la flexibilidad que aportaba SDR en las infraestructuras de comunicación era tal, que la principal ventaja de su uso en redes celulares era el soporte de nuevos protocolos inalámbricos con tan sólo actualizar el software. De este modo, podría allanar el camino de 5G en la sociedad actual sin necesidad de realizar costosos cambios en la infraestructura ya instalada.

Finalizando, se presenta el concepto de Software Defined Wifi, introducido por la empresa andaluza Galgus<sup>®</sup>. Según lo explican en su web, Wifi Definido por Software no es más que un paradigma novedoso respecto al despliegue de funcionalidades en una red inalámbrica. Se trata de liberar capacidades avanzadas de analítica, seguridad, gestión y optimización que hasta entonces estaban limitadas por lo que traía de serie su hardware. Siguiendo la misma filosofía que SDR, se logran notables ventajas en ámbitos cotidianos tanto en entornos de ocio y personales, además de una flexibilidad característica para muchas empresas que ofrecen un servicio.

# Apéndice A

## Códigos de Matlab

---

A continuación, se muestran los códigos de Matlab utilizados para la obtención de las representaciones gráficas expuestas en el Capítulo 4 (Diseño de Experimentos).

### Código A.1 Primer contacto con NESDR SMARTEE.

```
function rtl_sdr_rx_startup_matlab

%% PARAMETERS
rtl_sdr_id      = '0';           % RTL-SDR ID
rtl_sdr_tunerfreq = 100e6;       % RTL-SDR tuner frequency in Hz
rtl_sdr_gain     = 25;           % RTL-SDR tuner gain in dB
rtl_sdr_fs       = 2.4e6;        % RTL-SDR sampling rate in Hz
rtl_sdr_frmlen   = 4096;        % RTL-SDR output data frame size
rtl_sdr_datatype = 'single';     % RTL-SDR output data type
rtl_sdr_ppm      = 0;           % RTL-SDR tuner parts per million
    correction
sim_time         = 60;           % simulation time in seconds

%% SYSTEM OBJECTS
% rtl-sdr object
obj_rtl_sdr = comm.SDRRTLReceiver(...
    rtl_sdr_id,...
    'CenterFrequency', rtl_sdr_tunerfreq,...
    'EnableTunerAGC', false,...
    'TunerGain', rtl_sdr_gain,...
    'SampleRate', rtl_sdr_fs, ...
    'SamplesPerFrame', rtl_sdr_frmlen,...
    'OutputDataType', rtl_sdr_datatype ,...
    'FrequencyCorrection', rtl_sdr_ppm );

% spectrum analyzer objects
obj_specfft = dsp.SpectrumAnalyzer(...
    'Name', 'Spectrum Analyzer FFT',...
    'Title', 'Spectrum Analyzer FFT',...
    'SpectrumType', 'Power density',...
```

```

    'FrequencySpan', 'Full',...
    'SampleRate', rtlsdr_fs);
obj_specwaterfall = dsp.SpectrumAnalyzer(...
    'Name', 'Spectrum Analyzer Waterfall',...
    'Title', 'Spectrum Analyzer Waterfall',...
    'SpectrumType', 'Spectrogram',...
    'FrequencySpan', 'Full',...
    'SampleRate', rtlsdr_fs);

%% CALCULATIONS
rtlsdr_frmtime = rtlsdr_frmlen/rtlsdr_fs;

%% SIMULATION

% check if RTL-SDR is active
if isempty(sdrinfo(obj_rtlsdr.RadioAddress))
    error(['RTL-SDR failure. Please check connection to ',...
        'MATLAB using the "sdrinfo" command.']);
end

% reset run_time to 0 (secs)
run_time = 0;

% run while run_time is less than sim_time
while run_time < sim_time

    % fetch a frame from the rtlsdr
    rtlsdr_data = step(obj_rtlsdr);

    % update spectrum analyzer windows with new data
    step(obj_specfft, rtlsdr_data);
    step(obj_specwaterfall, rtlsdr_data);

    % update run_time after processing another frame
    run_time = run_time + rtlsdr_frmtime;

end

end

```

---

### Código A.2 Barrido del espectro.

```

function rtlsdr_rx_specsweep

% PARAMETERS (can change)
location = 'El Puerto de Santa María'; % location used for
figure name

```

```

start_freq      = 25e6;          % sweep start frequency
stop_freq       = 1750e6;       % sweep stop frequency
rtlsdr_id       = '0';         % RTL-SDR stick ID
rtlsdr_fs       = 2.8e6;       % RTL-SDR sampling rate in Hz
rtlsdr_gain     = 40;          % RTL-SDR tuner gain in dB
rtlsdr_frmlen  = 4096;        % RTL-SDR output data frame size
rtlsdr_datatype = 'single';    % RTL-SDR output data type
rtlsdr_ppm      = 0;           % RTL-SDR tuner parts per million
    correction
% PARAMETERS (can change, but may break code)
nfrmhold        = 20;          % number of frames to receive
fft_hold        = 'avg';      % hold function "max" or "avg"
nfft            = 4096;       % number of points in FFTs (2^something
    )
dec_factor      = 16;         % output plot downsample
overlap         = 0.5;        % FFT overlap to counter rolloff
nfrmdump       = 100;        % number of frames to dump after
    retuning (to clear buffer)

% CALCULATIONS
rtlsdr_tunerfreq = start_freq:rtlsdr_fs*overlap:stop_freq; % range of
    tuner frequency in Hz
if( max(rtlsdr_tunerfreq) < stop_freq ) % check the
    whole range is covered, if not, add an extra tuner freq
    rtlsdr_tunerfreq(length(rtlsdr_tunerfreq)+1) = max(rtlsdr_tunerfreq)
        +rtlsdr_fs*overlap;
end
nretunes = length(rtlsdr_tunerfreq); % calculate
    number of retunes required
freq_bin_width = (rtlsdr_fs/nfft); % create axis
freq_axis = (rtlsdr_tunerfreq(1)-rtlsdr_fs/2*overlap : freq_bin_width*
    dec_factor : (rtlsdr_tunerfreq(end)+rtlsdr_fs/2*overlap)-
    freq_bin_width)/1e6;

% create spectrum figure
h_spectrum = create_spectrum;

% run capture and plot
capture_and_plot;

% make spectrum visible
h_spectrum.fig.Visible = 'on';

% save data
filename = ['rtlsdr_rx_specsweep_',num2str(start_freq/1e6),'MHz_',
    num2str(stop_freq/1e6),'MHz_',location,'.fig'];
savefig(filename);

%% FUNCTION to create spectrum window
function h_spectrum = create_spectrum

```

```

% colours
h_spectrum.line_blue = [0.0000 0.4470 0.7410]; % spectrum
    analyzer blue
h_spectrum.line_orange = [1.0000 0.5490 0.0000]; % spectrum
    analyzer orange
h_spectrum.window_grey = [0.95 0.95 0.95]; % background light
    grey
h_spectrum.axes_grey = [0.1 0.1 0.1]; % dark grey for
    axes titles etc
h_spectrum.plot_white = [1 1 1]; % white for plot
    background

% sizes
fig_w = 1200;
fig_h = 600;
scnsize = get(0,'ScreenSize'); % find monitor 1
    size
if scnsize(3) < fig_w % if monitor is
    not fig_w wide
    fig_w = scnsize(3); % reduce fig_w
end
if scnsize(4) < fig_h % if monitor is
    not fig_h tall
    fig_h = scnsize(h); % reduce fig_h
end
fig_pos = [(scnsize(3)-fig_w)/2 (scnsize(4)-fig_h)/2 fig_w fig_h
    ]; % set to open in middle of monitor 1

% create new figure
h_spectrum.fig = figure(...
    'Color',h_spectrum.window_grey,...
    'Position',fig_pos,...
    'SizeChangedFcn',@resize_spectrum,...
    'Name',['RTL-SDR Spectrum Sweep: ',location],...
    'Visible', 'off');
h_spectrum.fig.Renderer = 'painters';

% subplot 1
h_spectrum.axes1 = axes(...
    'Parent',h_spectrum.fig,...
    'YGrid','on','YColor',h_spectrum.axes_grey,...
    'XGrid','on','XColor',h_spectrum.axes_grey,...
    'GridLineStyle','--',...
    'Color',h_spectrum.plot_white);
box(h_spectrum.axes1,'on');
hold(h_spectrum.axes1,'on');
xlabel(h_spectrum.axes1,'Frequency (MHz)');
ylabel(h_spectrum.axes1,'Power Ratio (dBm) [relative to 50 \
    Omega load] ');

```



```

xlim(h_spectrum.axes1,[start_freq/1e6,stop_freq/1e6]);

% subplot 2
h_spectrum.axes2 = axes(...
    'Parent',h_spectrum.fig,...
    'YGrid','on','YColor',h_spectrum.axes_grey,...
    'XGrid','on','XColor',h_spectrum.axes_grey,...
    'GridLineStyle','--',...
    'Color',h_spectrum.plot_white);
box(h_spectrum.axes2,'on');
hold(h_spectrum.axes2,'on');
xlabel(h_spectrum.axes2,'Frequency (MHz)');
ylabel(h_spectrum.axes2,'Relative Power (Watts)');
xlim(h_spectrum.axes2,[start_freq/1e6,stop_freq/1e6]);

% figure title
title(h_spectrum.axes1,['RTL-SDR Spectrum Sweep || Range = ',
    num2str(start_freq/1e6),'MHz to ',...
    num2str(stop_freq/1e6),'MHz || Bin Width = ',num2str(
        freq_bin_width*dec_factor/1e3),...
    'kHz || Number of Bins = ',num2str(length(freq_axis)), ' ||
        Number of Retunes = ',...
    num2str(nretunes)]);

% position axes
axes_position(fig_w,fig_h);

% link plots together for zooming
linkaxes([h_spectrum.axes1,h_spectrum.axes2],'x');

end

%% FUNCTION to calculate axes positions
function axes_position(fig_w,fig_h)

    h_spectrum.axes1.Position = [...           % dBm axes
        70/fig_w,...                           % 70px from left
        (fig_h/2)/fig_h,...                   % at centre line
        (fig_w-100)/fig_w,...                 % 100px from right
        (fig_h/2-30)/fig_h];                 % 80px from top

    h_spectrum.axes2.Position = [...           % Watts axes
        70/fig_w,...                           % 70px from left
        50/fig_h,...                           % 50px from bottom
        (fig_w-100)/fig_w,...                 % 100px from right
        (fig_h/2-100)/fig_h];                 % 100px below centre line

end

```

```

%% FUNCTION (callback) to resize axes in spectrum window
function resize_spectrum(hObject,callbackdata)

    % find current sizes
    fig_w = h_spectrum.fig.Position(3);
    fig_h = h_spectrum.fig.Position(4);

    % update axes positions
    axes_position(fig_w,fig_h);

end

%% FUNCTION to capture data from the RTL-SDR and plot it
function capture_and_plot

    % START TIMER
    tic;
    disp(' ');

    % SYSTEM OBJECTS
    % RTL-SDR system object
    obj_rtlsdr = comm.SDRRTLReceiver(...
        rtlsdr_id,...
        'CenterFrequency',    rtlsdr_tunerfreq(1),...
        'EnableTunerAGC',    false,...
        'TunerGain',         rtlsdr_gain,...
        'SampleRate',        rtlsdr_fs, ...
        'SamplesPerFrame',   rtlsdr_frmlen,...
        'OutputDataType',    rtlsdr_datatype ,...
        'FrequencyCorrection', rtlsdr_ppm );

    % FIR decimator
    obj_decntr = dsp.FIRDecimator(...
        'DecimationFactor',   dec_factor,...
        'Numerator',         fir1(300,1/dec_factor));

    % CALCULATIONS (others)
    rtlsdr_data_fft = zeros(1,nfft);           % fullsize matrix
        to hold calculated fft [1 x nfft]
    fft_reorder = zeros(length(nfrms),nfft*overlap); % matrix
        with overlap compensation to hold re-ordered ffts [navg x
        nfft*overlap]
    fft_dec = zeros(nretunes,nfft*overlap/dec_factor); % matrix with
        overlap compensation to hold all ffts [ntune x nfft*overlap/
        data_decimate]

    % SIMULATION
    % check if RTL-SDR is active
    if ~isempty(sdrinfo(obj_rtlsdr.RadioAddress))

```

```

else
    error(['RTL-SDR failure. Please check connection to ',...
          'MATLAB using the "sdrinfo" command.']);
end

% create progress variable
tune_progress = 0;

% for each of the tuner values
for ntune = 1:1:nretunes;

    % tune RTL-SDR to new centre frequency
    obj_rtlsdr.CenterFrequency = rtlsdr_tunerfreq(ntune);

    % dump frames to clear software buffer
    for frm = 1:1:nfrmdump
        % fetch a frame from the rtlsdr stick
        rtlsdr_data = step(obj_rtlsdr);
    end

    % display current centre frequency
    disp(['          fc = ',num2str(rtlsdr_tunerfreq(ntune)/1e6),
          'MHz']);

    % loop for nfrmhold frames
    for frm = 1:1:nfrmhold

        % fetch a frame from the rtlsdr stick
        rtlsdr_data = step(obj_rtlsdr);

        % remove DC component
        rtlsdr_data = rtlsdr_data - mean(rtlsdr_data);

        % find fft [ +ve , -ve ]
        rtlsdr_data_fft = abs(fft(rtlsdr_data,nfft))';

        % rearrange fft [ -ve , +ve ] and keep only overlap data
        fft_reorder(frm,( 1 : (overlap*nfft/2) )) =
            rtlsdr_data_fft( (overlap*nfft/2)+(nfft/2)+1 : end );
        % -ve
        fft_reorder(frm,( (overlap*nfft/2)+1 : end )) =
            rtlsdr_data_fft( 1 : (overlap*nfft/2) );      % +ve

    end

    % process the fft data down to [1 x nfft*overlap/
    data_decimate] from [nfrmhold x nfft*overlap/
    data_decimate]
    if strcmp(fft_hold,'avg')
        % if set to average, find mean

```

```

        fft_reorder_proc = mean(fft_reorder);

elseif strcmp(fft_hold,'max')
    % if set to max order hold, find max
    fft_reorder_proc = max(fft_reorder);

end

% decimate data to smooth and store in spectrum matrix
fft_dec(ntune,:) = step(obj_decmttr,fft_reorder_proc)';

% show progress if at an n10% value
if floor(ntune*10/nretunes) ~= tune_progress;
    tune_progress = floor(ntune*10/nretunes);
    disp(['    progress = ',num2str(tune_progress*10),'%']);
end

end

% REORDER INTO ONE MATRIX
fft_masterreshape = reshape(fft_dec',1,ntune*nfft*overlap/
    dec_factor);

% PLOT DATA
y_data = fft_masterreshape;
y_data_dbm = 10*log10((fft_masterreshape.^2)/50);
plot(h_spectrum.axes1,freq_axis,y_data_dbm,'Color',h_spectrum.
    line_blue,'linewidth',1.25);
plot(h_spectrum.axes2,freq_axis,y_data,'Color',h_spectrum.
    line_orange,'linewidth',1.25);

% STOP TIMER
disp(' ');
disp(['    run time = ',num2str(toc),'s']);
disp(' ');

end
end

```

# Índice de Figuras

---

2.1	Diagrama de bloques funcionales genérico de un dispositivo SDR	4
2.2	Arquitectura de una radio definida por software ideal	5
2.3	Arquitectura empleada en una BTS con SDR	6
3.1	NESDR SMArTee v2	12
3.2	Raphael Micro <sup>®</sup> R820T radio tuner	13
3.3	Diagrama de Bloques de Realtek RTL2832U	13
3.4	Diagrama de Bloques de Nooelec NESDR SMArTee v2	15
4.1	Equipo de Nooelec	20
4.2	Communications System Toolbox Support Package	21
4.3	Verificación del entorno de trabajo	22
4.4	Espectro de señales RF	23
4.5	Modelo de Simulink receptor de señales RF	24
4.6	Resultado de ejecutar el modelo de Simulink	26
4.7	Configuración de parámetros del NESDR SMArTee	27
4.8	Resultados al modificar $f_s$ de 2.8 MHz a 3.2 MHz	27
4.9	Señal centrada en 100 MHz	28
4.10	Señal de radio FM	29
4.11	Actividad GSM en 955.1 MHz	31
4.12	Actividad GSM en 958.4 MHz	32
4.13	Señal UMTS	34
4.14	Comienzo de una señal LTE	35
4.15	Continuación de una señal LTE	36
4.16	Canales de televisión y radio en 482 MHz. Información obtenida de <a href="https://www.tdt1.com/canales-cadiz/">https://www.tdt1.com/canales-cadiz/</a>	37
4.17	Señales de Televisión Digital DVB-T	38
4.18	ServiceMode	39
4.19	Obtenida en <a href="https://www.sqimway.com/gsm_arfcn.php">https://www.sqimway.com/gsm_arfcn.php</a>	40
4.20	Medida del espectro antes de realizar la llamada sobre GSM	41
4.21	Medida del espectro durante la llamada sobre GSM	42
4.22	Barrido del espectro. Emplazamiento 1	44
4.23	Barrido del espectro. Emplazamiento 2	45
4.24	Receptor de radio FM mono	47
4.25	Bloque FM MPX Demultiplexer	48
4.26	Decodificador FM estéreo	48

4.27	Bloque FIR Rate Conversion y parámetros de configuración	49
4.28	Receptor de radio FM estéreo	50
4.29	Resultado simulación radio FM estéreo	51
4.30	Receptor en 89.4 MHz	52

# Índice de Tablas

---

2.1	Comparación de componentes para el procesamiento digital de señal	7
2.2	Comparación de sistemas SDR implementados en el mercado	10
3.1	Rango de frecuencias según el sintonizador	17
4.1	Frecuencias de muestreo aceptadas en el NESDR SMArTee	25
4.2	Servicios de comunicación prestados en el espectro	46





# Índice de Códigos

---

A.1	Primer contacto con NESDR SMArTee	55
A.2	Barrido del espectro	56



# Bibliografía

---

- [Ecu] EcuRed, *RTLSDR Dongle*. dirección: [https://www.ecured.cu/RTLSDR\\_Dongle#:~:text=RTLSDR%20Dongle%3A%20Es%20un%20dispositivo,parecido%20al%20de%20un%20pendrive..](https://www.ecured.cu/RTLSDR_Dongle#:~:text=RTLSDR%20Dongle%3A%20Es%20un%20dispositivo,parecido%20al%20de%20un%20pendrive..)
- [Fal18] P. M. Falcón, *Algoritmos de decodificación de señales radio analógicas/digitales sobre SDR*. Universidad de Granada, 2018.
- [Fora] W. I. Forum, *Software Defined Systems Committee (SDS)*. dirección: <https://sds.wirelessinnovation.org/>.
- [Forb] W. I. Forum, *What is Software Defined Radio*. dirección: <https://www.wirelessinnovation.org/assets/documents/SoftwareDefinedRadio.pdf>.
- [Gon14] M. P. González, *Estudio piloto de los demoduladores de la serie RTL de Realtek para la Radio Definida por Software*. Universidad de Sevilla, 2014.
- [Her13] T. R. Hernández, *Implementación en Simulink de técnicas de modulación para SDR*. Universidad Central “Marta Abreu” de Las Villas, 2013.
- [Ken05] P. B. Kenington, *RF and Baseband Techniques for Software Defined Radio*. Artech House, 2005.
- [Nooa] Nooelec, *Nooelec NESDR SMARTEE v2 Bundle*. dirección: <https://www.nooelec.com/store/sdr/sdr-receivers/nedr-smartee.html>.
- [Noob] Nooelec, *Nooelec NESDR SMARTEE v2 SDR*. dirección: <https://www.nooelec.com/store/nedr-smartee-sdr.html>.
- [Org] S. Organization, *SDL Organization*. dirección: [https://online.visual-paradigm.com/app/diagrams/#diagram:proj=0&type=BlockDiagram&gallery=/repository/29540ed8-ea17-4f56-bb2a-966b247cd8e4.xml&name=SDL%20Organization%20\(Architecture%20View\)](https://online.visual-paradigm.com/app/diagrams/#diagram:proj=0&type=BlockDiagram&gallery=/repository/29540ed8-ea17-4f56-bb2a-966b247cd8e4.xml&name=SDL%20Organization%20(Architecture%20View)).
- [Pin11] I. D. Pinar, *Software defined radio USRP y plataformas de desarrollo*. Universidad de Sevilla, 2011.
- [Rad] H. Radio, *R820T Rafael Micro*. dirección: <https://radioaficion.com/cms/r820t-rafael-micro/>.
- [Rafa] I. Rafael Microelectronics, *R820T Datasheet*. dirección: [https://www.rtl-sdr.com/wp-content/uploads/2013/04/R820T\\_datasheet-Non\\_R-20111130\\_unlocked1.pdf](https://www.rtl-sdr.com/wp-content/uploads/2013/04/R820T_datasheet-Non_R-20111130_unlocked1.pdf).
- [Rafb] I. Rafael Microelectronics, *R820T2 Register Description*. dirección: [https://www.rtl-sdr.com/wp-content/uploads/2016/12/R820T2\\_Register\\_Description.pdf](https://www.rtl-sdr.com/wp-content/uploads/2016/12/R820T2_Register_Description.pdf).
- [Reaa] Realtek, *RTL2832U*. dirección: <https://www.realtek.com/en/products/communications-network-ics/item/rtl2832u>.

- [Reab] Realtek, *RTL2832U Datasheet*. dirección: <https://www.kechuan.org/r/303602>.
- [RC15] D. S. w. A. Robert W. Stewart Kenneth W. Barlee y L. H. Crockett, *Software Defined Radio using MATLAB and Simulink, and the RTL-SDR*. Strathclyde Academic Media, 2015.
- [RS] Rohde y Schwarz, *Software defined radios – overview and hardware*. dirección: [https://cdn.rohde-schwarz.com/pws/dl\\_downloads/dl\\_common\\_library/dl\\_news\\_from\\_rs/182/n182\\_radiocomunit.pdf](https://cdn.rohde-schwarz.com/pws/dl_downloads/dl_common_library/dl_news_from_rs/182/n182_radiocomunit.pdf).
- [Rou09] T. J. Roupheal, *RF and Digital Signal Processing for Software-Defined Radio. A Multi-Standard Multi-Mode Approach*. Newnes, 2009.
- [RTL a] RTL-SDR.COM, *ABOUT RTL-SDR*. dirección: <https://www.rtl-sdr.com/about-rtl-sdr/>.
- [RTL b] RTL-SDR.COM, *R820T2 CHIP DISCONTINUED*. dirección: <https://www.rtl-sdr.com/r820t2-chip-discontinued-low-cost-r820t2-rtl-sdrs-will-continue-airspy-will-redesign/>.
- [RTL c] RTL-SDR.COM, *RTL-SDR Blog V3 Datasheet*. dirección: <https://www.rtl-sdr.com/wp-content/uploads/2018/02/RTL-SDR-Blog-V3-Datasheet.pdf>.
- [Sch] D. A. Scher, *Software Defined Radio Assignment - Introduction*. dirección: [http://aaron.scher.com/Course\\_materials/SDR\\_Assignment1/SDR\\_Assignment1.html](http://aaron.scher.com/Course_materials/SDR_Assignment1/SDR_Assignment1.html).
- [Waq16] J. I. y. J. N. Waqar Hussain, *The Evolution of Software-Defined Radio: An Introduction*. Springer, 2016.
- [YB06] S. M. Youping Zhao Jeffrey H. Reed y K. K. Bael, «Overhead Analysis for Radio Environment Map-enabled Cognitive Radio Networks», *IEEE Workshop on Networking Technologies for Software Defined Radio Networks (SDR)*, sep. de 2006.