PROYECTO FIN DE CARRERA

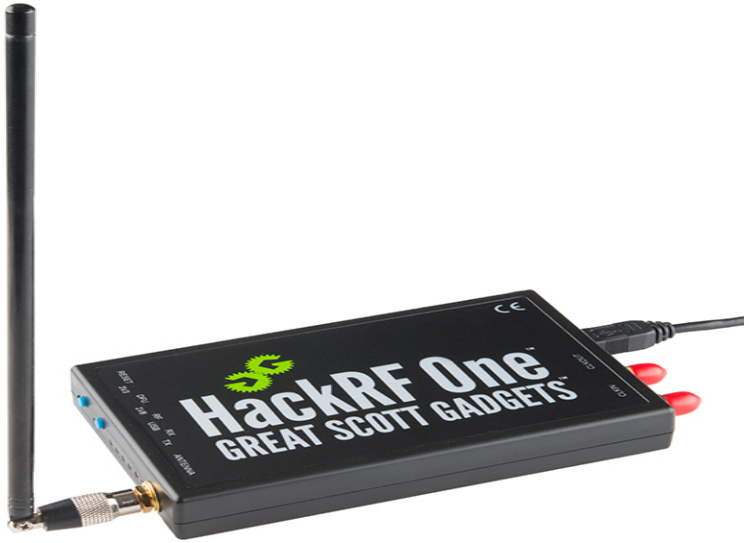
Análisis software y hardware del SDR HackRF One

Jorge Rodríguez de Haro

INGENIERÍA DE TELECOMUNICACIÓN

2016/17

El propósito principal de este proyecto es el análisis y estudio del dispositivo SDR HackRF One, tanto a nivel de aplicaciones como a nivel de diseño hardware. Finalmente se podrá proponer el diseño de un dispositivo SDR a nivel de esquemático con capacidad de operación full-duplex. A diferencia de HackRF One que sólo puede operar en modo half-duplex.



Andrés María Roldán Aranda es el profesor ingeniero a cargo del presente proyecto, así como el tutor del alumno. Actualmente es profesor del departamento de Electrónica y Tecnología de Computadores de la Universidad de Granada.

UNIVERSIDAD DE GRANADA

Ingeniería de Telecomunicación

Análisis software y hardware del SDR HackRF One

Jorge Rodríguez de Haro

Año académico 2016/2017

Tutor: Andrés María Roldán Aranda

INGENIERÍA DE TELECOMUNICACIÓN

PROYECTO FINAL DE CARRERA

"Análisis software y hardware del SDR HackRF One"

CURSO: 2016/2017

Jorge Rodríguez de Haro

INGENIERÍA DE TELECOMUNICACIÓN

"Análisis software y hardware del SDR HackRF One"

REALIZADO POR:

Jorge Rodríguez de Haro

DIRIGIDO POR:

Andrés María Roldán Aranda

DEPARTAMENTO:

Electrónica y Tecnología de los Computadores

D. Andrés María Roldán Aranda, Profesor del departamento de Electrónica y Tecnología de los Computadores de la Universidad de Granada, como director del Proyecto Fin de Carrera de D. Jorge Rodríguez de Haro,

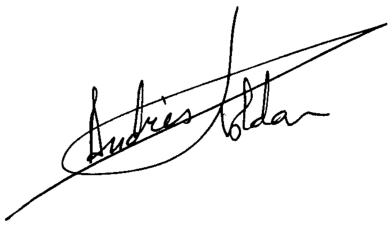
Informa:

que el presente trabajo, titulado:

"Análisis software y hardware del SDR HackRF One"

ha sido realizado y redactado por el mencionado alumno bajo nuestra dirección, y con esta fecha autorizo a su presentación.

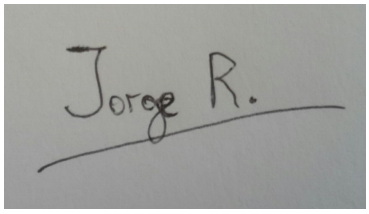
Granada, a 11 de Septiembre de 2017



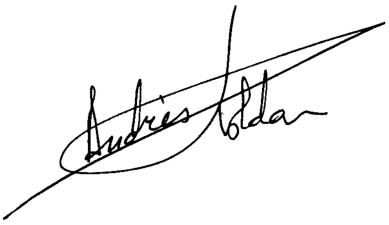
Fdo. Andrés María Roldán Aranda

Los abajo firmantes autorizan a que la presente copia del Proyecto Final de Carrera se ubique en la Biblioteca del Centro y/o departamento para ser libremente consultada por las personas que lo deseen.

Granada, a 11 de Septiembre de 2017



Fdo. Jorge Rodríguez de Haro



Fdo. Andrés María Roldán Aranda

A la memoria de Manuel de Haro Martín

Agradecimientos:

En primer lugar, me gustaría agradecer a mis abuelos Manuel y Carmen su constante guía y los múltiples consejos que siempre me han proporcionado a lo largo de mi vida, que han influido notoriamente en que hoy sea la persona que soy y a los que siempre tendré como referentes. A mi madre Mª del Carmen porque siempre intentó hacer lo mejor para mí en cada momento y siempre quiso que no me conformase con lo fácil. A mi padre Jorge por transmitirme su ilusión por la ciencia y el descubrir nuevas ideas y conceptos. A mis abuelos paternos por su comprensión y sencillez. A mi tía Eva por su apego y lealtad, a mi tía Silvia por hacerme ver otra forma de vivir la vida y a mi tío Manuel. Finalmente entre mis amigos, a Miguel Ángel, por su confianza y sus ganas de buscar siempre nuevas metas.

También quiero agradecer a mis compañeros y amigos de la facultad, quiénes me han ayudado en cada momento que lo he necesitado en mis años universitarios. Haciendo una especial mención a David Gómez Molino, Jose Montes Cañete y en estos últimos meses a Pablo Sánchez Garrido compañero en el laboratorio.

Por último, agradezco a mi tutor, Andrés María Roldán Aranda su tiempo dedicado y comentarios instructivos, no sólo para este proyecto, sino para mi futura vida profesional.

ÍNDICE

ÍNDICE DE FIGURAS

CAPÍTULO

1

INTRODUCCIÓN

El presente proyecto final de carrera finaliza los estudios en Ingeniería de Telecomunicación. El objetivo de este proyecto es el diseño de un transceiver full-dúplex que implemente tecnología software-defined-radio. Para ello, se ha tomando como base para el diseño un dispositivo SDR de hardware libre; el ampliamente extendido entre la comunidad de seguridad informática y actividades hacking, HackRF One.

Figura 1.1 – *Imagen del dispositivo HackRF One.*

1.1 Antecedentes

1

El proyecto surgió con la propuesta del profesor Andrés Roldán Aranda de realizar el diseño en Altium del SDR HackRF One. Este dispositivo permite la transmisión o recepción de señales en el rango de 10 MHz a 6 GHz, lo que le da una gran flexibilidad y potencia a la hora de realizar cualquier envío o recepción de información dentro de toda la banda de frecuencias comercial de uso diario. Esta opción fue elegida, debido a su amplia aceptación en el mercado y a su buena relación calidad/precio.
Dentro de ese rango de frecuencias, se encuentran las bandas para FM, DAB, TDT, Televisión por Satélite, GPRS(2.5G), UMTS(3G), LTE(4G), 5G, Wi-Fi (2.4 y 5 GHz) y Bluetooth. Lo que evidencia, la gran cantidad de aplicaciones comerciales en las que puede operar HackRF One.
Posteriormente la propuesta varió, tras mi proposición de hacer un diseño full-dúplex del mismo, de forma que el dispositivo pudiese enviar y recibir señales en todo el rango de forma simultánea, este hecho hacía que el aprendizaje y el reto de diseño tanto a nivel de esquemático y layout, como a nivel tecnológico fuese aún mayor, haciendo que el trabajo de ingeniería requiriese la selección de nuevos ICs que harían más complejo el funcionamiento del dispositivo, de forma que la componente de 'ingenii' y desarrollo del mismo tuviese un cariz más íntimo y de desarrollo tanto personal como profesional.

Figura 1.2 – *Logo de GranaSat.*

El presente proyecto se engloba dentro del conjunto de proyectos que forman parte del grupo de alumnos de la Universidad de Granada asociado a la ESA, Granasat. Este grupo tiene como meta, el situar en el espacio un CubeSat diseñado, construido y ensamblado por alumnos de la Univerdad de Granada. El profesor Andrés Roldán Aranda es el promotor y director del mismo, gestionando los PFCs, TFGs y TFMs que ha lo largo de los últimos 4 años se han desarrollado dentro del seno del mismo.
Personalmente, el hecho de trabajar con el profesor Andrés Roldán Aranda, tanto como el resto de compañeros que han formado y forman parte del mismo, ha sido una aventura constante de aprendizaje, tanto en el plano personal, como en el laboral.

1.2 Introducción a SDR

Un SDR puede ser descrito como un sistema de radiocomunicaciones donde gran parte de los componentes son implementados usando software en lugar de usar una implementación hardware, usando para ello un dispositivo embebido que trata la información y la transmite a un computador.
La principal ventaja de estos sistemas, es la posibilidad del uso de un microprocesador de propósito general para el procesamiento de señal, lo que reduce la complejidad del sistema al no ser necesaria la implementación hardware del bloque completo, lo que permite a su vez la reducción del coste total del dispositivo. Además, un sistema basado en software defined radio, es más flexible debido al posible uso de una gran variedad de configuraciones del mismo, lo que nos brindaría una amplia gama de posibilidades. Como corolario, podemos decir que todos los sistemas SDR están compuestos de dos elementos principales: un dispositivo hadware que recibe las señales, y un software que configura el dispositivo (fijando el modo de demodulación, la ganancia y la banda de frecuencia).

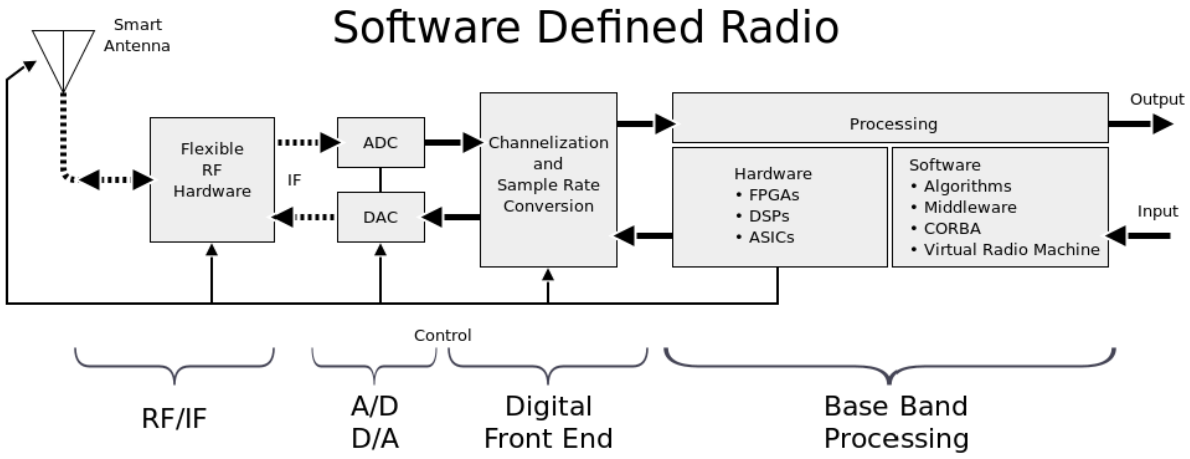


Figura 1.3 – *Diagrama de bloque genérico.*

El proyeto GranaSat usa dispositivos SDR para la recepción de señales de los satélites meteorológicos del NOAA, entre otras aplicaciones. Algunos de los ejemplos de dispositivos que tiene el proyecto GranaSat, son el RTL-SDR y el FUNcube Dongle, los cuáles son dispositivos de bajo coste; por otro lado, también usa un dispositivo de coste medio como es el HackRF One, que se desarrollará en el presente trabajo.

RTL-SDR es un dispositivo basado en un dongle para recepción de TDT, DAB y FM, conectado a un computador a través de un puerto USB. Opera en el rango entre los 22 y los 2200 MHz, y tiene un ancho de banda de canal máximo de 3.2 MHz. Usando un software configurado especialmente para el funcionamiento SDR, se puede conseguir que el dongle opere como un SDR, consiguiendo así un dispositivo low cost. Por supuesto, las prestaciones de este dongle no serán iguales a las de un SDR dedicado, pero tiene un rendimiento realmente notable, lo que lo hace válido para el proyecto GranaSat.

Figura 1.4 – *Imagen del RTL-SDR.*

Por otro lado, FUNcube Dongle es un hardware diseñado para una implementación muy sencilla, con las únicas conexiones de un puerto USB y una antena. Esto hace que el mismo carezca de controles físicos, de forma que todas las funciones del dongle son controladas desde un computador vía software. Opera en el rango entre los 150 KHz y 260 MHz, y los 410 y 2050 MHz, con una frecuencia de muestreo de 192 KHz.
El FUNcube Dongle es muy similar al RTL-SDR, con la excepción de que cuenta con un cristal de cuarzo de alta precisión, que tiene una variación de 1.5 ppm; lo que sería equivalente a una variación de 1.5 Hz, por cada MHz de la señal de salida del cristal.

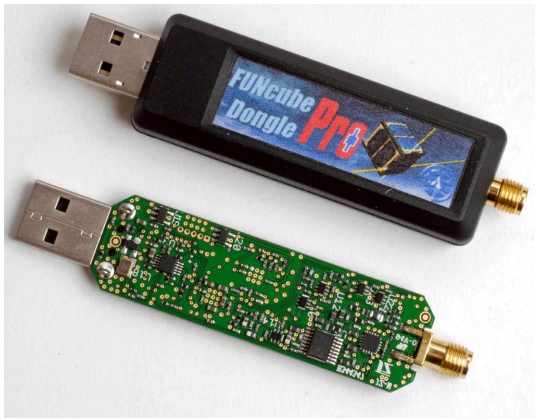


Figura 1.5 – *Imagen del FUNcube Dongle.*

1.3 Introducción al Software para SDR

Como se ha especificado anteriormente durante este capítulo, un dispositivo SDR necesita usar un software que lo configure. Es posible encontrar en la red gran cantidad de programas que cumplen esta función, la mayoría de los cuales son gratuitos. En esta sección del capítulo 1, hablaremos de los dos programas que usaremos en la presente memoria.
SDR# (SDR Sharp) es probablemente el programa más popular en Windows para uso en dispositivos SDR como el RTL-SDR, el FUNcube Dongle y el HackRF One. Su popularidad se debe a su alto rendimiento usando algoritmos de procesado digital de señales, su facilidad

de uso, rapidez y su gratuidad. Además, hay gran cantidad de información y tutoriales disponibles que hacen que la gran mayoría de SDRs del mercado sean compatibles con este software.

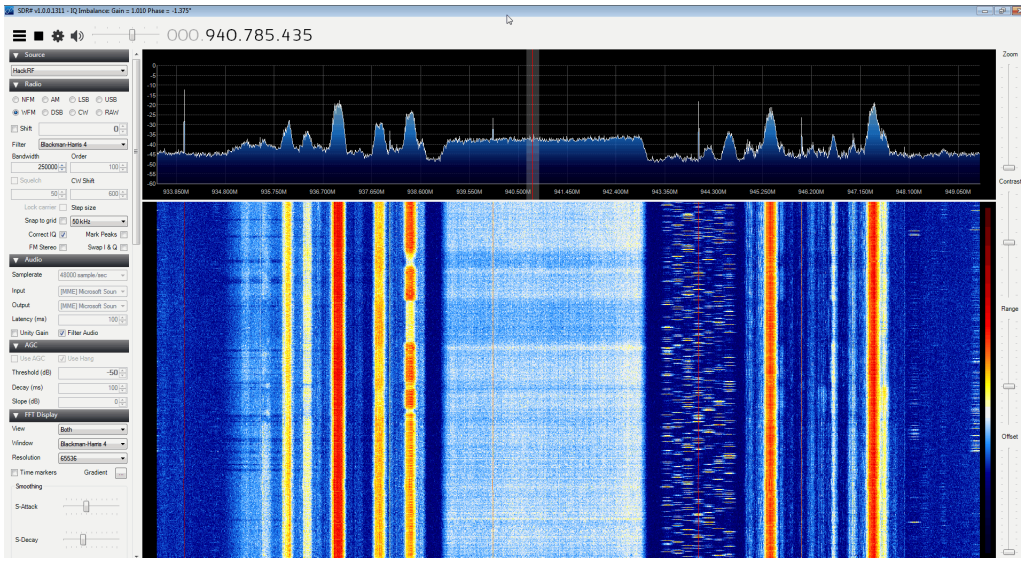


Figura 1.6 – *Imagen de SDR#.*

En la imagen ulterior a modo de ejemplo, podemos ver como usando SDR# con HackRF One se sintoniza la banda de 940 MHz, para demodular señales digitales GSM para comunicación móvil. Donde cada uno de los lóbulos que aparecen en el espectro, son señales correspondientes a diferentes radio emisoras con aplicaciones diferentes.

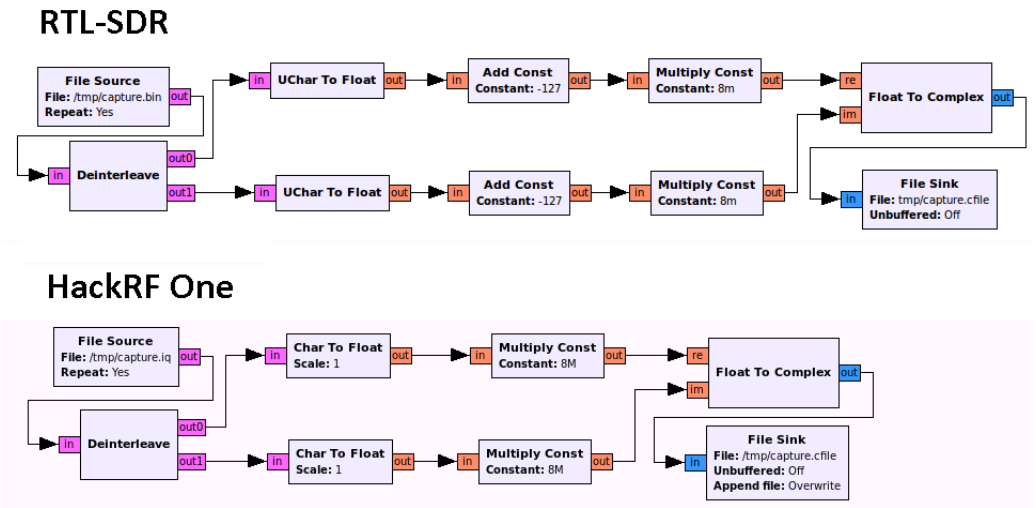


Figura 1.7 – *Imagen de GNU Radio.*

También se empleará el software para SDRs más usado en Linux, GNU Radio. Este

software a diferencia de SDR# no es un entorno gráfico con controles y botones al modo usual de un programa, sino que se basa en un sistema de programación gráfica al modo de Simulink en Matlab, LabView o Scada.
En la imagen superior, podemos ver como usando una misma plantilla podemos hacer tratamiento de datos usando RTL-SDR y HackRF One simultáneamente. En este caso vemos como se usa un deinterleaver, para desentrelazar cadenas de bits que corresponderán a la componente en fase y a la componente en cuadratura de las las señales recibidas, que posteriormente serán almacenadas en un archivo usando un tipo de dato especial para números complejos.

1.4 Objetivos del proyecto

Los principales objetivos de este proyecto final de carrera son:

- Conocer y analizar las necesidades reales de un dispositivo SDR a nivel comercial.
- Extraer los requisitos principales y secundarios así como los diferentes subsistemas que conformarán el producto, valorar la viabilidad de cada uno de ellos y proponer nuevos puntos de vista y opciones.
- Realizar un análisis de tecnologías disponibles mediante las cuales se puedan solucionar las necesidades de cada uno de los requisitos.
- Diseñar un esquema electrónico del producto para su desarrollo posterior e implementación.
- Realizar los test de validación de funcionamiento del producto y las pruebas necesarias para determinar que la fase de prototipado ha sido sido exitosa.
- Acercar al alumno al trabajo real con empresas tecnológicas.
- Formar de al alumno en el diseño de productos electrónicos.
- Poner de manifiesto los conocimientos adquiridos por el alumno principalmente durante los estudios de ingeniería de telecomunicación y reforzar los adquiridos por cuenta ajena en materia de diseño de productos electrónicos.
- Superar la asignatura de Proyecto Fin de Carrera con éxito.

1.5 Estructura del proyecto

El proyecto se divide en 6 capítulos y un anexo que describen cada una de las partes del proceso de desarrollo del producto propuesto. Estos capítulos pretenden describir de un modo lógico y cronológico el trabajo llevado a cabo durante la duración del proyecto.

Los capítulos que conforman el presente documentos son:

- El presente capítulo, numerado como 1, pretende ser una introducción al proyecto tanto en su faceta de proyecto final de carrera como en su faceta de relación con el proyecto GranaSat.
- El capítulo 2 es un breve resumen de la información extraída de las necesidades requeridas por HackRF One para ser un producto comercial. De ahí se extrae una idea global de las necesidades principales y secundarias del producto a diseñar en forma de requisitos.
- El capítulo 3 hace un análisis completo de las características técnicas de HackRF One y se presentarán varias aplicaciones posibles usando el dispositivo; por último se analizará su rendimiento en ciertas bandas de interés para su uso en comunicaciones satélite, requeridas para el proyecto GranaSat.
- A continuación, el capítulo 4 detalla el esquemático de HackRF One y el desarrollo del producto además de los detalles de implementación que resultan relevantes para describir en profundidad el proceso.
- Tras realizar el análisis y siguiendo la metodología propuesta, se presentan en el capítulo 5 el diseño del esquemático para crear una variante SDR de HackRF One que tenga la capacidad de funcionar en modo full-duplex. Donde se pondrá de manifiesto que el producto diseñado durante el capítulo cumple con los requerimientos técnicos, y por ende con los requisitos impuestos por el cliente en el capítulo 2.

1

CAPÍTULO

2

DEFINICIÓN DE REQUISITOS

Tras introducir el tema de estudio del presente proyecto, en este segundo capítulo se definirán cuales son los requisitos principales y secundarios sobre los que el trabajo discurrirá.

2.1 Requisitos comerciales del producto

Este conjunto de requisitos responde a las necesidades de mercado y está basado en el análisis realizado por Michael Ossmann y Jared Boone, creadores de la empresa Great Scott Gadgets y diseñador principal y secundario de HackRF One. Adicionalmente, se han añadido una serie de características propias para el posible diseño del esquemático de una versión HackRF One Full-duplex. Estas características fueron acordadas con el profesor Andrés Roldan Aranda tras la búsqueda de los integrados adecuados a nivel comercial, y a la comparativa de las prestaciones que ofrecían los diferentes fabricantes.
A continuación se exponen los requisitos tal y como los expuso el cliente en su petición inicial, los **principales** son:

1. El dispositivo fruto del presente proyecto debe tener la capacidad de operación en modo half-duplex y full-duplex.

2. El sistema debe de actuar como un transceiver, delegando el procesado digital de señal en el procesador principal del PC al que se conecta y no en el procesador nativo del dispositivo.

3. Debe ser capaz de enviar los datos al PC al que se conecta, de forma que el mismo pueda almacenarlos e interpretarlos consecuentemente.

4. El sistema debe mantener la arquitectura del transceiver TDD(Time Division Duplexing) en modo half-duplex, soportando así las capacidades del dispositivo inicial en cuanto al tipo y características de las señales a procesar.

5. El dispositivo debe ser compacto y ligero, de forma que le permita ser portable y pueda desplazarse con nuestro PC de manera conjunta.

6. Se requiere que el dispositivo al ser alimentado de manera externa desde un computador, preferiblemente portátil, tenga un consumo extremadamente bajo y limitado a la alimentación que puedan proporcionar las interfaces entre el dispositivo y el PC.

7. El producto con vista a una futura fabricación, deberá cumplir los requisitos necesarios, para una posterior implementación de diferentes librerías que le permitan ser compatible con todos los programas con los que puede usarse HackRF One.

8. El producto debe mantener una relación calidad/precio equivalente a la del dispositivo original, intentando que el coste de la nueva versión HackRF One Full-duplex sea similar, aunque ligeramente superior al de HackRF One.

Por otra parte, de forma **secundaria**, de los aspectos citados anteriormente se deduce que los requerimientos darán lugar a una serie de modificaciones básicas que son necesarias para que el dispositivo pueda desarrollarse, en las que se basan las siguientes plausibles consideraciones:

1. El sistema deberá sustituir el microcontrolador de arquitectura ARM a 204 MHz(dual core, Cortex-M4/M0), al menos por un microcontrolador de la misma familia LPC43XX pero con mayor número de pines y mayores prestaciones en cuanto a interfaces y memoria.

2. Deberá añadirse una nueva antena y un nuevo bloque de transmisión que replique al de HackRF One, manteniendo así la arquitectura de doble conversión del sistema principal originario.

3. El dispositivo, deberá usar una nueva interfaz para poder tener capacidad full-duplex, lo que podría requerir la adición de otro puerto USB 2.0, u optar por la elección de un puerto USB 3.0.

4. El sistema deberá replantear la fuente de alimentación del mismo, manteniendo la portabilidad. Lo que requerirá el uso de una fuente con topología Buck que soporte un mayor nivel de intensidad, y en consecuencia se deberá usar una interfaz de comunicación que soporte ese consumo.

5. La transmisión de datos sobre la interfaz a usar deberá mantener el tipo de dato usado para la transmisión de información al PC, de forma que el dispositivo pueda ser fácilmente compatible con las librerías ya creadas para HackRF One.

6. Por último, deberá tenerse en cuenta que todos los plausibles cambios anteriores, no supongan un incremento sustancial del precio del producto en base a una posible futura fabricación.

2

CAPÍTULO

3

ANÁLISIS Y APLICACIONES DE HACKRF ONE

En este tercer capítulo se llevará a cabo un análisis pormenorizado de las características técnicas de HackRF One y se presentarán algunas de sus posibles aplicaciones. Para lo cuál se usará el software introducido en el capítulo 1 para el uso de dispositivos SDR, tanto en sistema operativo Windows como Linux. Por otro lado, se mostrarán algunas medidas realizadas a la antena del dispositivo usando un analizador de espectros para comprobar su correcto funcionamiento en las diferentes bandas dentro de su rango de operación.

Figura 3.1 – *Imagen de HackRF One operativo.*

3.1 Análisis de HackRF One

HackRF One es un periférico Software Defined Radio capaz de transmitir o recibir señales de radio desde 1 MHz hasta 6 GHz fabricado por Great Scott Gadgets. Fue diseñado para facilitar el desarrollo y testeo(tanto auditoría como hacking) de tecnologías de comunicación radio tanto actuales(soporta LTE), como en desarrollo para las nuevas generaciones de tecnologías radio y sus correspondientes protocolos. HackRF One es una plataforma de hardware libre que puede ser usada como un periférico vía USB, o programada para operar de forma autónoma.

A continuación se detallarán las características principales del dispositivo, junto con algunas de las capacidades que le confieren una gran potencia y flexibilidad:

- Tiene un rango de operación en frecuencia desde 1 MHz hasta 6 GHz.
- Es un transceiver con capacidad de operación half-duplex.
- Tiene una capacidad de muestreo de hasta 20 millones de muestras por segundo, pudiéndose alcanzar las 21,5 en función del tipo de controlador USB 2.0 HS que incluya el computador al que se conecta.
- Muestreo de las señales con 8 bits en cuadratura, donde 8 bits serán para la componente en fase I y 8 serán para la componente en cuadratura Q.
- Es compatible con los principales programas para SDR tanto en Windows como en Linux, lo que incluye tanto a SDR# como a GNU Radio.
- Puede configurar vía software los amplificadores de ganancia, con 3 etapas dedicadas para recepción y 2 etapas para transmisión.
- Puede configurar vía software los filtros de señal en banda base, con un máximo de ancho de banda de señal de 28 Mhz, y con una caída de 3 dB hasta 30 MHz.
- Permite controlar vía software la potencia suministrada al puerto de la antena, con hasta 50 mA a 3.3 V.
- Conector de antena SMA hembra.
- Un conector SMA hembra para sincronizar el reloj, tanto a la entrada como a la salida, lo que permite conectar varios HackRF One y hacerlos trabajar conjuntamente, reduciendo los problemas de jitter.
- Botones para configurar convenientemente el dispositivo.
- Cabeceras de pines internos para una posible expansión de la placa usando shields.
- Uso de la interfaz USB 2.0 High Speed.

- Todo el sistema se alimenta a través de la conexión USB, sin necesidad de añadir una fuente de alimentación externa, la que le confiere una gran portabilidad.
- Es una plataforma de hardware libre.

El dispositivo HackRF One, viene dentro de un encapsulado moldeado por inyección plástica e incorpora un cable macho de USB a Micro USB tipo B, el cable tiene una longitud pequeña, y es que dada la tasa de transferencia de datos que puede alcanzar el dispositivo, en caso de necesitar un conector de mayor longitud éste debería incorporar un núcleo de ferrita para el filtrado del plausible ruido inducido.
Para poder operar el dispositivo será necesario el uso de una antena. Se recomienda el uso de ANT 500 que opera desde 75 MHz hasta 1 GHz, o ANT700 que opera desde 300 MHz hasta 1,1 GHz.
HackRF One es un equipo para prueba y monitorización de sistemas de radio frecuencia, por lo que no ha sido diseñado para cumplir con las regulaciones de transmisión de señales de radio, dando lugar a que su uso en ciertas bandas quede bajo la responsabilidad del usuario.

3.2 Uso de HackRF One en SDR#

A continuación se ilustrará con una aplicación el uso de HackRF One en sistema operativo Windows usando SDR#, para lo cuál se configurará el programa para poder recibir señales de radio en FM, tal y como se nos muestra en la imagen siguiente:

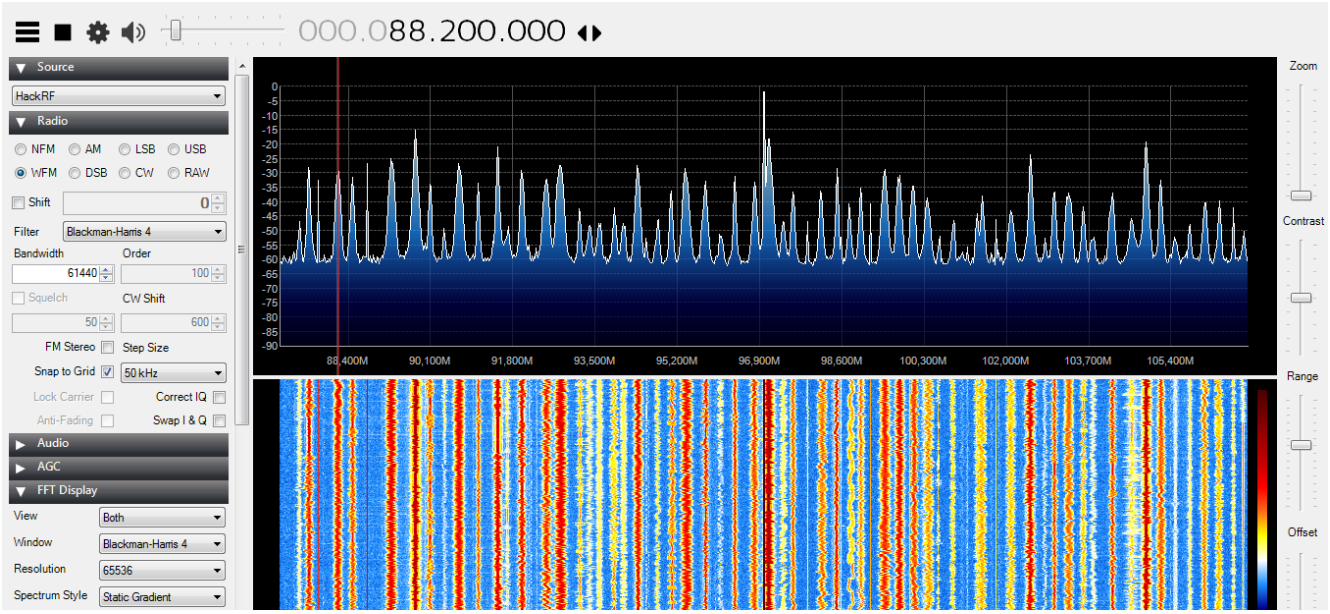


Figura 3.2 – *Recepción FM en SDR#.*

Se comenzará con la instalación de los drivers pertinentes una vez conectado HackRF One a nuestro PC, a través de la ejecución del programa 'Zadig' que listará los puertos de

nuestro PC, reconociendo a HackRF One. Posteriormente se ejecutará SDR# y se seleccionará el dispositivo a usar(en nuestro caso HackRF One).
Se configurará la frecuencia a la que operaremos(en el caso de la imagen superior a 88.2 MHz donde vemos la barra roja centrada en el lóbulo), la tasa de muestreo del convertidor analógico-digital(MAX5864) con un límite máximo de 21,5 Msamples/s, se fijará la ganancia proporcionada por los tres amplificadores de recepción de HackRF One. Dos de los amplificadores (VGA y LNA, integrados en el MAX2837) son variables, mientras que el amplificador final LNA es un integrado independiente y solo puede estar en estado activo o apagado. Posteriormente se seleccionará el modo de recepción para nuestro demodulador FM, donde se elegirá la opción de WFM(Wideband FM), ya que en caso de usar NFM(Narrowband FM) no se escuchará nada debido a que esta técnica no se usa para transmisión de señal en FM comercial. El ancho de banda seleccionado para el filtro de recepción digital será 50 KHz, y éste usará el algoritmo 'Blackman-Harris 4'. Por último en el display de FFT(Fast Fourier Transform) deberá seleccionarse una resolución mínima de 65535 para poder apreciar los lóbulos de señal en las respectivas bandas con claridad.

3.3 Uso de HackRF One en GNURadio

A continuación se ilustrará con una aplicación el uso de HackRF One en sistema operativo Linux usando GNURadio, para lo cuál se configurará el programa para poder recibir señales de radio en FM, tal y como se nos muestra en la imagen siguiente:

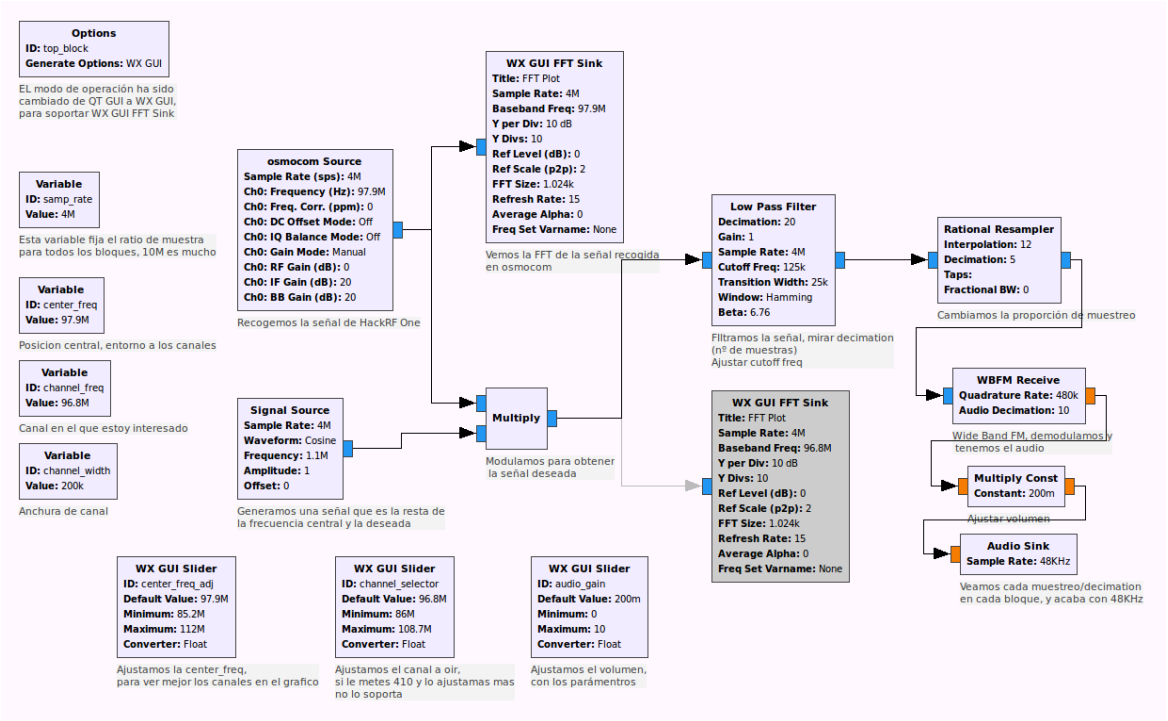


Figura 3.3 – *Recepción FM en GNURadio.*

Como se puede apreciar en la imagen, GNURadio es un programa que usa bloques al igual que simulink en Matlab para diseñar un sistema que sea capaz de procesar un conjunto de datos de acuerdo a un propósito concreto, que en este caso es la recepción de radio FM usando HackRF One.
En este caso, el conjunto de bloques del receptor FM hace operar a HackRF One de acuerdo al desempeño del sistema; es decir, que parte del tratamiento de la señal será configurada en HackRF One, mientras que otra será realizada por el procesador nativo de nuestro computador. Podemos ver que se usa un bloque 'osmocom Source' que configura el 'Sample Rate' de 4 Msps, 'Ch0:Frecuency' a 97.9 MHz y la amplificación de los tres amplificadores de RF(Radio Frecuency) a 0 dB, IF(Intermediate Frecuency) a 20 dB y BB(Base Band) a 20 dB. Posteriormente se usa un multiplicador con una 'Signal Source' de 'Sample Rate' a 4 Msps y una 'Frecuency' para bajar a banda base la señal de FM(debe aclararse que las frecuencias pueden variarse usando un slider). A continuación se usará un filtro paso baja para suprimir los armónicos superiores no deseados y se hará uso de un 'Rational Resampler' que ajustará la frecuencia de muestreo multiplicándola por el factor de interpolación y dividiéndola por el factor de decimación. Finalmente se demodulará la señal usando el bloque 'WBFM Receive' con una frecuencia final de 48 KHz(calidad de audio CD estéreo)y ésta será multiplicada por un valor constante de audio que acabará en un sumidero de salida.

A continuación se ilustrará una aplicación del uso de HackRF One en sistema operativo Linux usando GNURadio, para lo cuál se configurará el programa para poder transmitir señales de radio en FM, tal y como se nos muestra en la imagen siguiente:

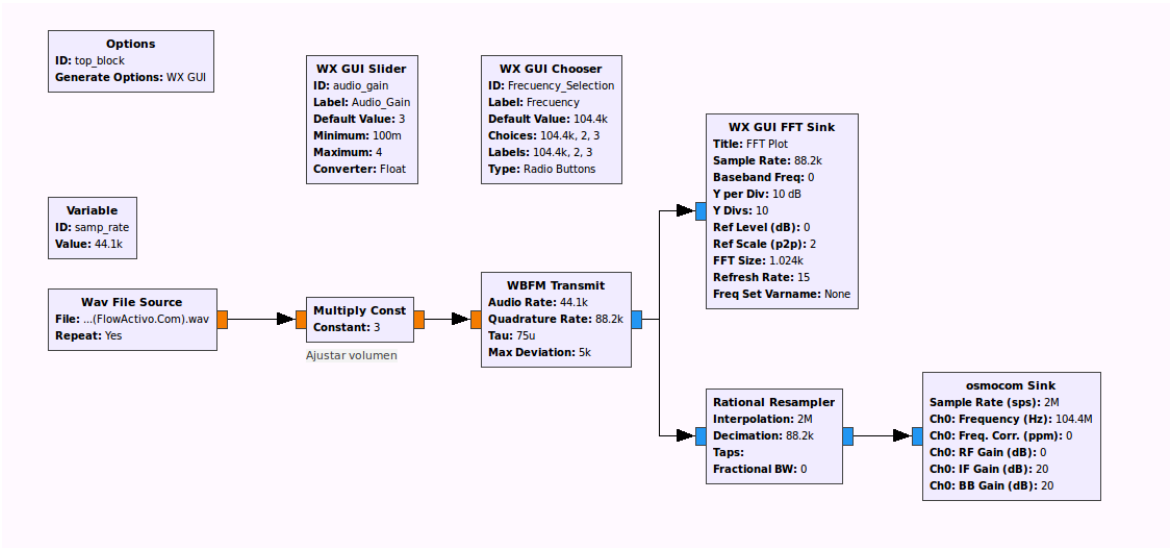


Figura 3.4 – *Transmisión FM en GNURadio.*

En la imagen superior podemos ver un diagrama de bloques de un transmisor FM para GNURadio. El conjunto de bloques es opuesto al de un receptor FM, comenzando con una fuente de archivos .wav 'Wav File Source', seguida de un multiplicador para ajustar

el volumen de la muestra, que posteriormente pasa por un transmisor de WBFM 'WBFM Transmit' donde un 'Rational Resampler' cambia la frecuencia de muestreo a la salida y que finalmente deriva en un 'osmocom Source' que emite la señal y donde hay que configurar tanto el 'Sample Rate' a 4 Msps, como el 'Ch0:Frecuency'(frecuencia donde se desee emitir) y la amplificación de los tres amplificadores de RF(Radio Frecuency) a 0 dB, IF(Intermediate Frecuency) a 20 dB y BB(Base Band) a 20 dB.

3.4 Análisis del espectro de recepción con ANT500

A continuación se hará un análisis del espectro recibido por HackRF One, para ello usaremos la antena telescópica ANT500 que tiene una longitud variable de 20 a 88 cm, opera entre 75 MHz y 1 GHz, usa un conector SMA macho y está diseñada para tener una resistencia de antena de 50 Ohmios, de forma que pueda ser adaptada a una pista con impedancia característica de 50 Ohmios como carga minimizando las pérdidas:

Figura 3.5 – *Imagen de ANT500.*

Para hacer las medidas necesarias para la caracterización del espectro recibido en el rango de operación de la antena, se usará el analizador de espectros MS2830A-041 que opera en el rango de frecuencias entre 9 KHz y 6 GHz:

Figura 3.6 – *Imagen del MS2830A-041.*

Como configuración para el MS2830A-041 hemos fijado como nivel de referencia de señal 0 dBm y una atenuación intrínseca del analizador de 0 dB a la entrada, de forma que no

varíe la potencia de señal recibida.
Comenzaremos mostrando una imagen del espectro que se puede recibir con la antena desde 1MHz a 1 GHz:

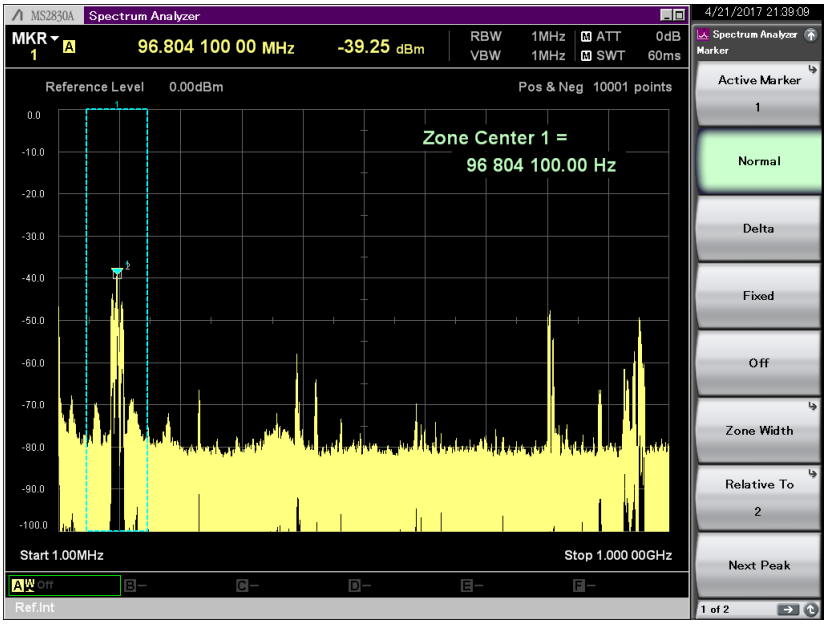


Figura 3.7 – *Imagen del espectro completo.*

Como se puede ver en la figura anterior, se reciben todas las señales esperadas para ese rango de frecuencia a nivel comercial. A modo de ejemplo se mostrará de manera más precisa la recepción de señal en el rango de radio FM:

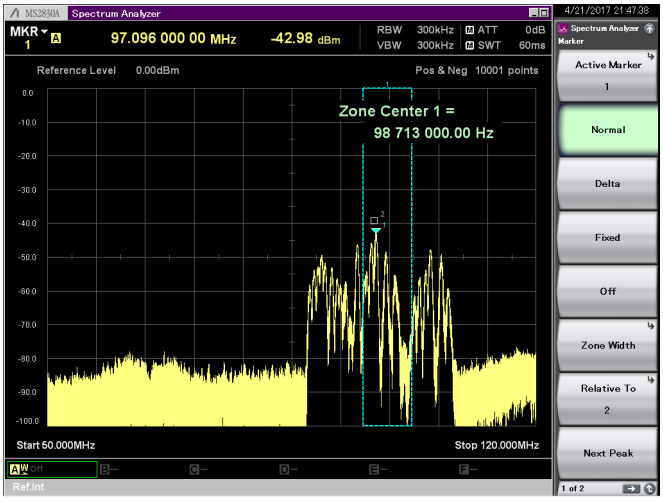


Figura 3.8 – *Imagen del rango FM.*

3.5 Análisis de consumo de HackRF One

Como parte final del capítulo se mostrarán las mediciones realizadas de HackRF One, en condiciones de máximo consumo tanto en transmisión como en recepción con todos los amplificadores activados y con máxima ganancia, de forma que usando el medidor siguiente se pudiese obtener el voltaje (V) e intensidad (I) que necesitaba HackRF One para su correcto desempeño:

Figura 3.9 – *Imagen del medidor de V/A.*

- Stand-by Mode: 5 V 0.23 A
- Transmission Mode: 5 V 0.40 A
- Reception Mode: 5 V 0.44 A

Finalmente podemos ver, que independientemente del modo de operación por el que se opte, el consumo siempre se mantendrá por debajo de los 0.5 A de alimentación que es capaz de proporcionar el protocolo USB 2.0, y que se alcanza el máximo consumo en el modo de recepción, ya que este además de tener un amplificador de RF e IF, cuenta además con un amplificador extra en BB(Banda Base).

CAPÍTULO

4

ANÁLISIS DEL ESQUEMÁTICO DE HACKRF ONE

En el presente capítulo se especificarán los detalles de diseño e implementación de cada uno de los componentes del esquemático de HackRF One que se presentarán a continuación:

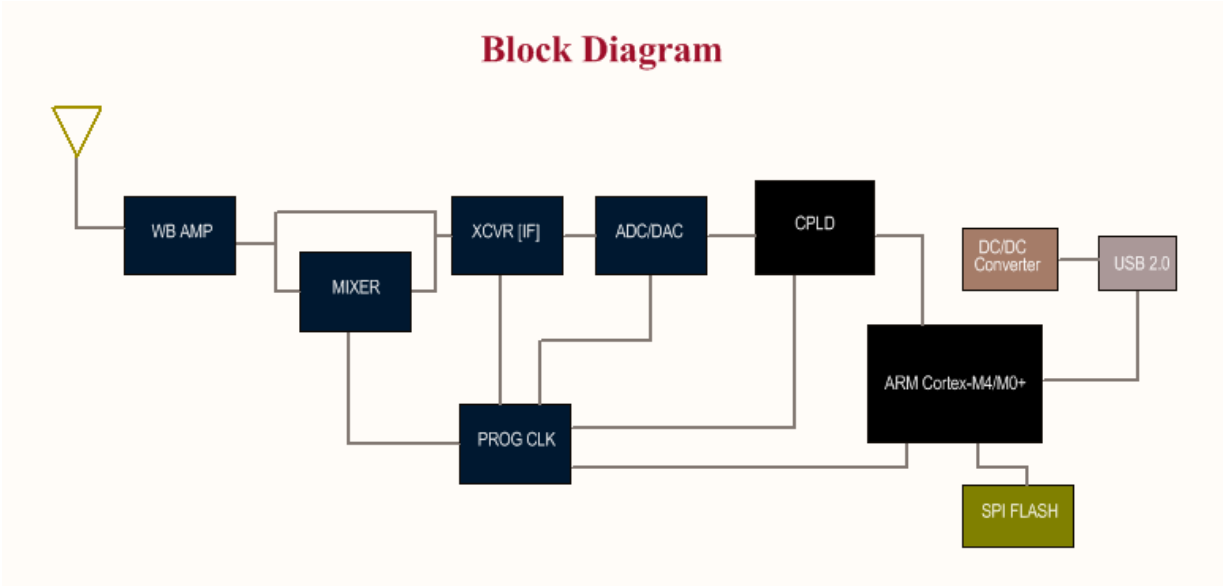


Figura 4.1 – *Diagrama de bloques de HackRF One.*

Partiendo de la imagen anterior como diagrama de bloques fundamental, se citará y explicará cada uno de los bloques que componen el diagrama de forma que el lector pueda llegar a tener una imagen global del sistema, y pueda enlazar y entender el comportamiento de cada una de las partes incluidas en el mismo que serán detalladas a continuación:

- WB AMPL: es un amplificador de banda ancha(Wideband Amplifier) para radiofrecuencia, que se usa como etapa final de amplificación. Esta implementado tanto en la opción de receptor , como en la de transmisor, por lo que en el esquemático aparecerán como 2 amplificadores independientes, cada uno en sentido opuesto.

- MIXER: es un mezclador, que se utiliza tanto en modo de recepción como de transmisión para poder manejar señales fuera del rango de 2.15 a 2.75 GHz(en cuyo caso la conexión con el bloque siguiente se establecerá de forma directa como se puede apreciar en el diagrama), con lo que una señal recibida fuera del rango anterior puede ser trasladada hacia 2.15 y 2.75 GHz usando una portadora local generada en el mezclador; empleándose la operación inversa para el caso de transmisión, desplazando esa señal desde 2.15 a 2.75 GHz hasta cualquier frecuencia distinta entre 1 MHz y 6 GHz, usando una portadora local.

- XCVR [IF]: es un transceiver que puede recibir o transmitir señales desde o hacia la banda de 2.15 a 2.75 GHz directamente desde banda base, por lo que en esta aplicación se usa como un conversor a frecuencia intermedia(IF) para posteriormente pasar a la frecuencia final usando el mezclador, siempre que la frecuencia destino no se encuentre en la banda indicada anteriormente, en cuyo caso no será necesario su uso. Por lo que podemos decir que la aplicación usa una arquitectura de doble conversión de modo genérico, aunque entre 2.15 y 2.75 GHz se opte por usar una conversión directa.

- ADC/DAC: es un conversor dual que se compone de un ADC(Analog to Digital Converter) y un DAC(Digital to Analog Converter), además tiene capacidad de operación full-duplex, por lo que ambos conversores pueden funcionar simultáneamente.

- CPLD: es un dispositivo lógico programable complejo, que se usa a modo de interfaz entre el conversor ADC/DAC y el procesador del dispositivo para hacer de puente lógico entre los protocolos de recepción de datos de ambos bloques.

- ARM Cortex-M4/M0+: es un microcontrolador de 2 núcleos con arquitectura ARM perteneciente a la familia Cortex-M, que se caracteriza por ser la que tiene un menor consumo dentro de la gama Cortex, siendo el Cortex-M0+ el núcleo con menor consumo de todos los que ARM tiene bajo licencia.

- PROG CLK: es un integrado que nos permite generar varios relojes simultáneamente con una frecuencia independiente para cada nuevo reloj, a partir de un único cristal como referencia fundamental.

- SPI FLASH: es una memoria flash conectada al microcontrolador ARM, que puede ser reconfigurada, y que nos permite la configuración del microcontrolador principal y de la CPLD, cada vez que se inicie HackRF One.

- DC/DC Converter: es un conversor reductor de voltaje(topología Buck) con dos salidas a tensiones diferentes que forma la fuente de alimentación de la placa, y está alimentado por la tensión de entrada del puerto USB 2.0.

- USB 2.0: es la interfaz encargada de suministrar la alimentación al dispositivo HackRF One y de permitir el paso de datos entre nuestro computador y la placa de circuito impreso tanto en modo de operación receptor, como transmisor.

A continuación de presentará la imagen del layout del dispositivo HackRF One, totalmente construido e implementado:

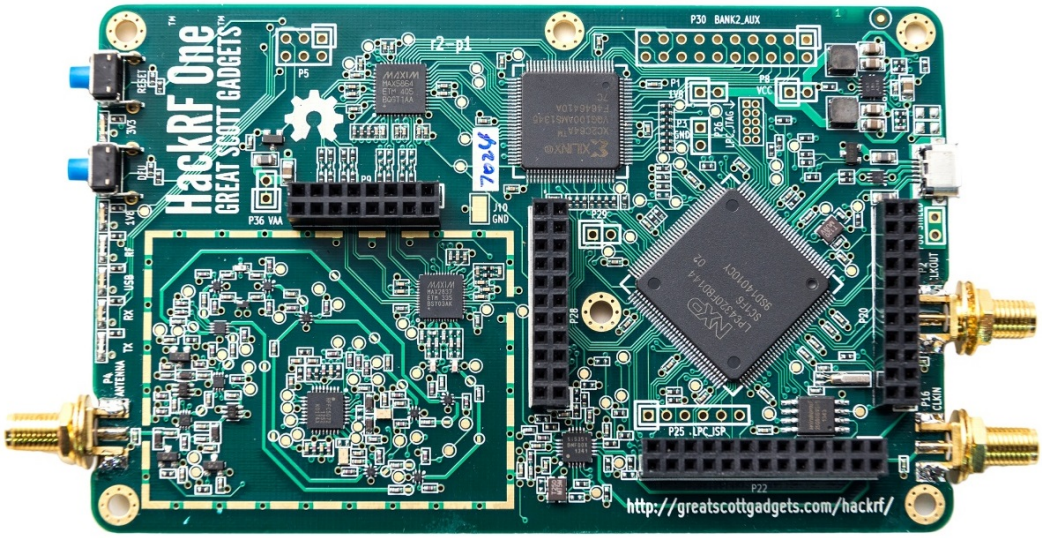


Figura 4.2 – *Placa de circuito impreso de HackRF One.*

4.1 Implementación del esquemático en Altium

El dispositivo HackRF One fue diseñado usando el software de edición de placas de circuito impreso KidCad, dado que al principio del proyecto se planteó la creación en Altium de HackRF One se crearon los esquemáticos en este software para poder implementarlo. Posteriormente tras sugerir el diseño de una versión HackRF Full-duplex, se partió de los esquemáticos del anterior dispositivo que serán expuestos a continuación.

El esquemático de HackRF One consta de tres cartelas diferentes que contienen todos los componentes que forman el dispositivo:

1. La primera cartela muestra el 'FRONTEND' del dispositivo, es decir la zona de Radio Frecuencia.

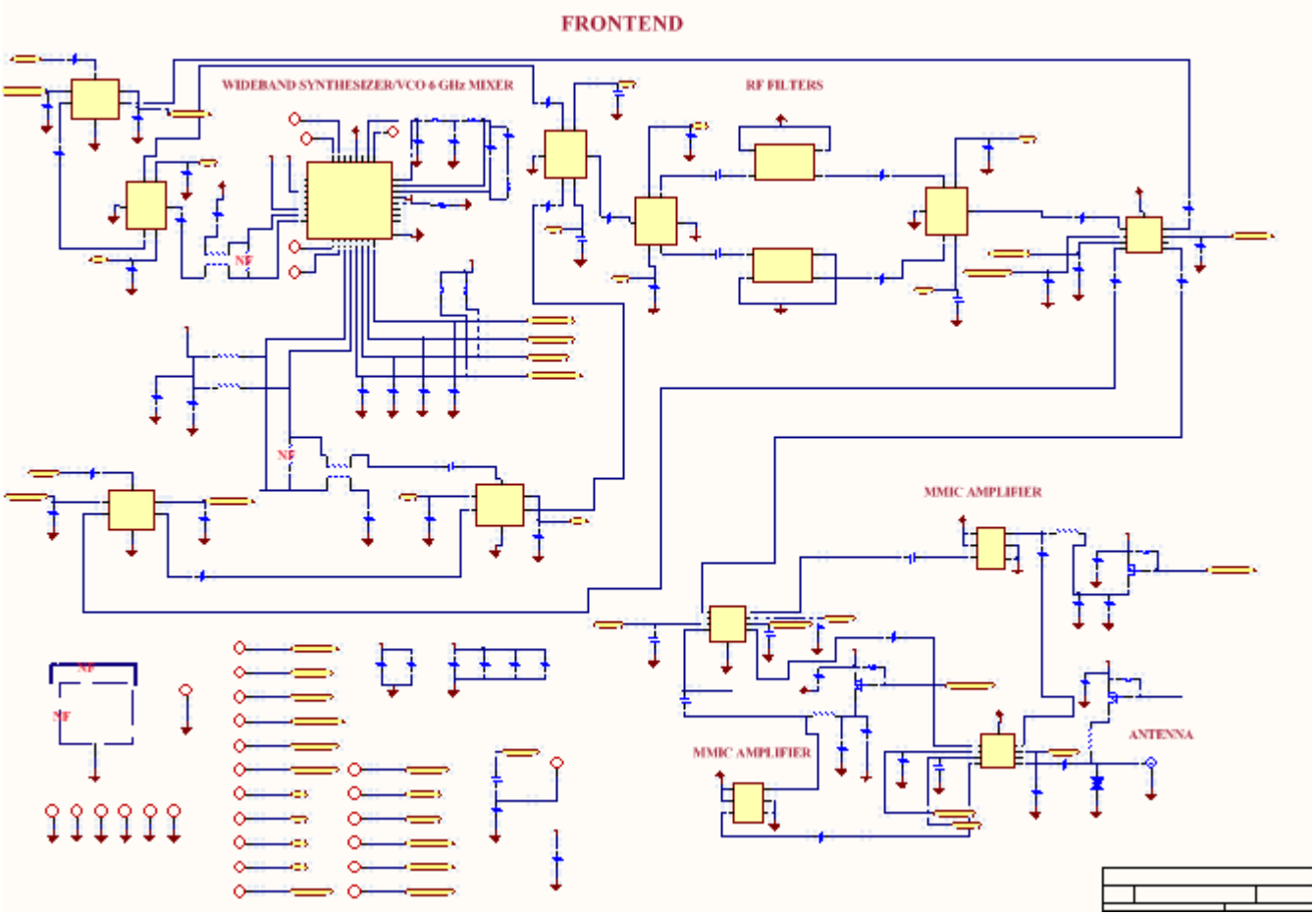


Figura 4.3 – *Esquemático Frontend de HackRF One.*

2. La segunda cartela muestra el 'BASEBAND', donde está la zona de frecuencia intermedia y banda base.

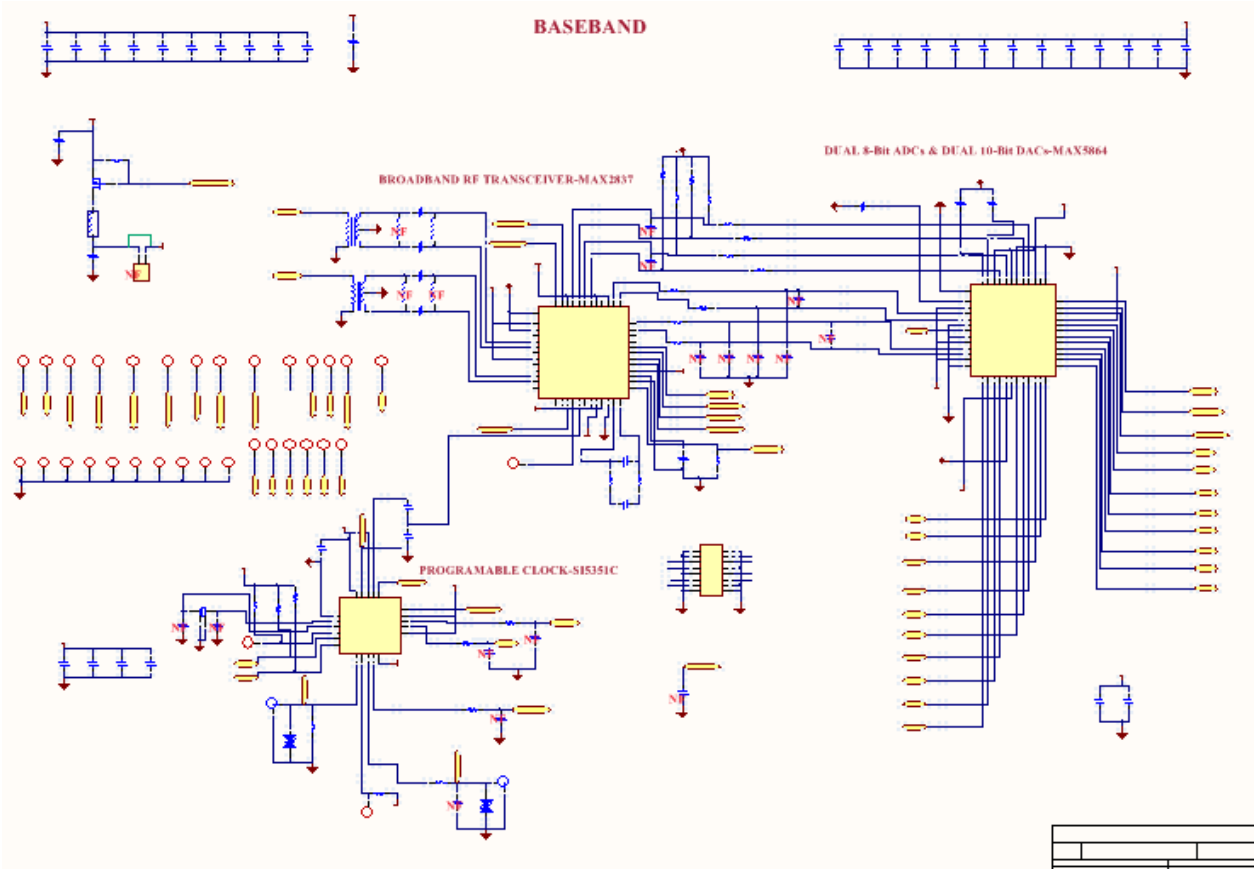


Figura 4.4 – *Esquemático Baseband de HackRF One.*

3. La tercera cartela muestra el lugar del procesador ARM, la CPLD y la alimentación vía USB 'ARM/CPLD'.

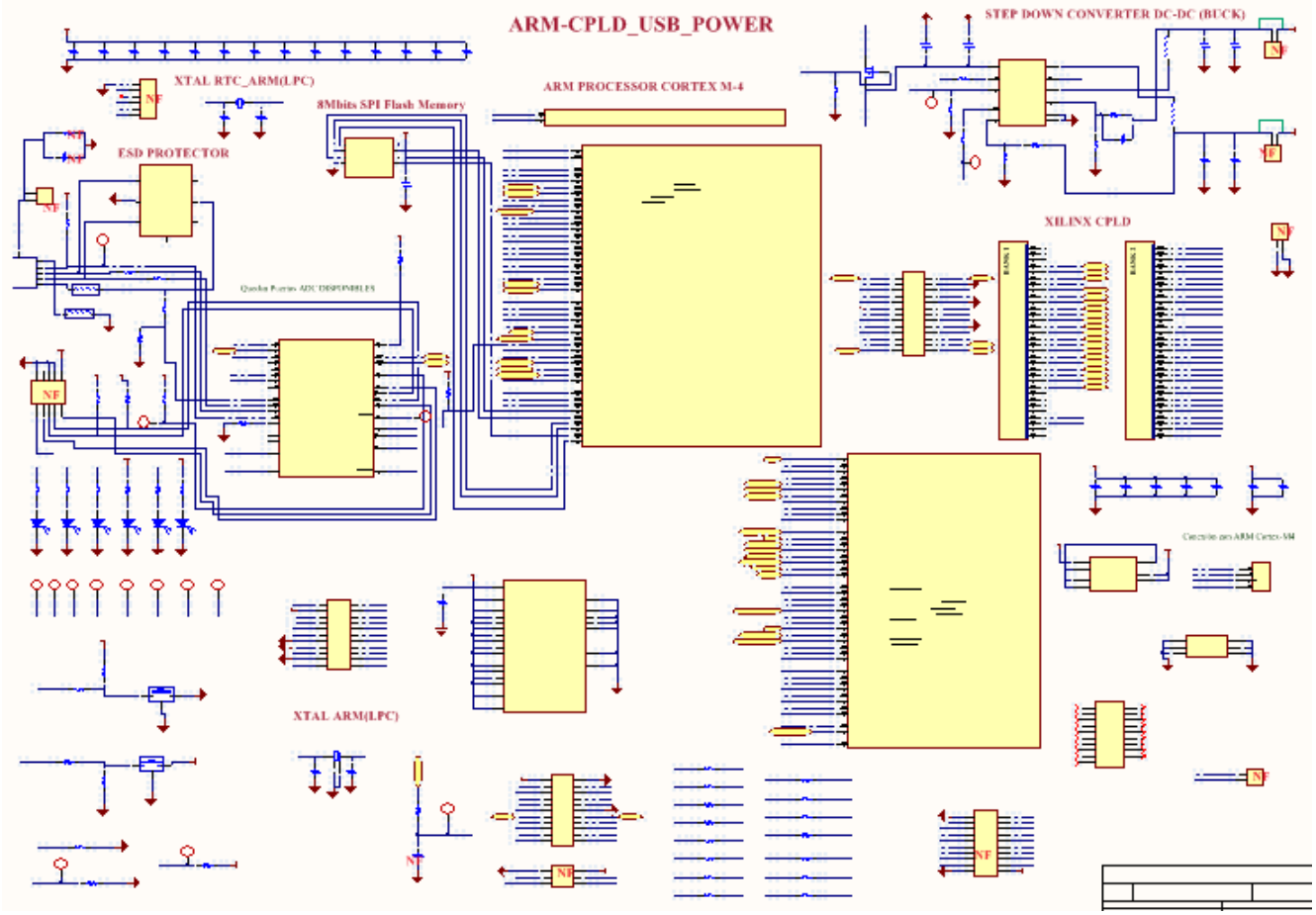


Figura 4.5 – *Esquemático ARM-CPLD de HackRF One.*

4.2 Desarrollo esquemático del presente capítulo

El esquemático de HackRF One, debido a su extensión se desarrolló para cartelas en formato A1, A2 y A3; por lo que su tamaño es demasiado grande para poder incluirse en el presente proyecto final de carrera de forma precisa. Ante este inconveniente, será necesario desarrollar cada bloque del esquemático de forma independiente, así como cada uno de los componentes que lo componen, de forma que pueda alcanzarse una correcta asimilación de los contenidos propuestos para el presente capítulo.

A su vez, también se incluirán imágenes de los modelos 3D creados para una mejor visualización de los componentes, las huellas o footprints usadas para crear un layout de circuito impreso, y las figuras usadas para representar los componentes en los esquemáticos.

4.3 Etapa de Frontend

Comenzaremos comentando el esquemático desde el 'Frontend', para comenzar a hacer un análisis profundo sobre la parte del sistema que comienza por la antena y se encarga de recibir o transmitir las señales en esta arquitectura half-duplex, y que será representada en la siguiente imagen:

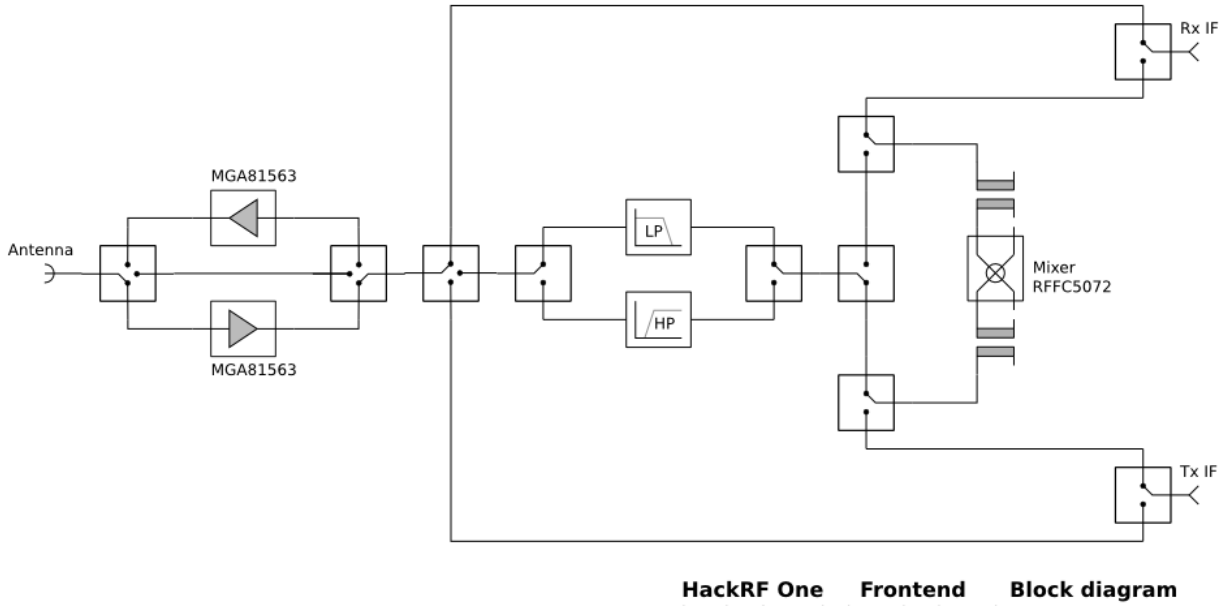


Figura 4.6 – *Diagrama de Bloques.*

El Frontend comienza con el conector SMA de antena, para el uso de la antena ANT500 o ANT700 como elemento recomendado, posteriormente tendremos un switch de tres salidas que da la opción de usar el amplificador de RF según el sentido en el que opere el dispositivo(en modo receptor o transmisor), o simplemente cortocircuitarlo en caso de que no sea necesario su uso, posteriormente se usará otro switch de tres salidas, según si vamos a usar el mezclador para desplazarnos en frecuencia, o sino está la opción de ir a la etapa de frecuencia intermedia para el caso de recepción o de transmisión.

Figura 4.7 – *Conector SMA.*

A continuación introduciremos una imagen del esquemático, donde se puede apreciar de manera más detallada la zona de la antena, el conmutador o switch y el amplificador de la etapa final de radio frecuencia:

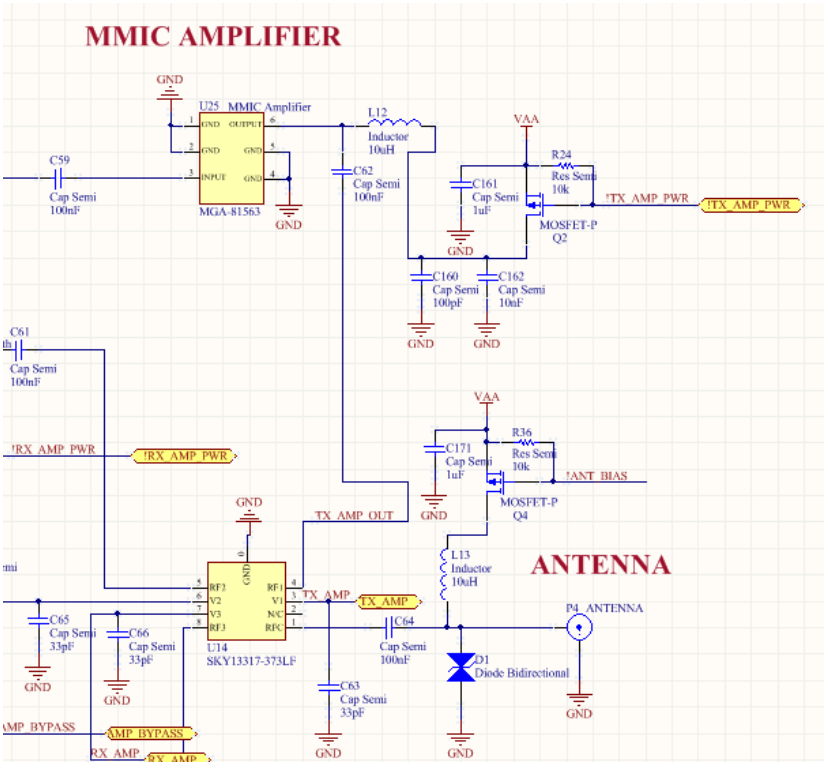


Figura 4.8 – *Zona de la Antena.*

El conmutador o switch que se usa en esta zona del esquemático es el *SKY13317*, tiene un rango de operación de 20 MHz a 6 GHz, una tensión de control de 0 / 1.8-5 V, una pérdida por inserción máxima de 0.8 dB a 6 GHz, un alto aislamiento de hasta 25 dB, y un punto de compresión a 1dB(P1dB) de hasta +29 dBm:

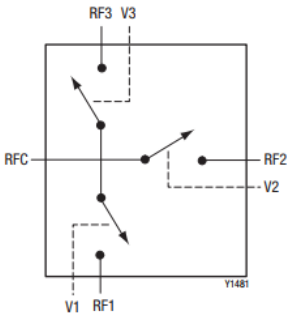


Figura 4.9 – *Diagrama del conmutador de 3 salidas.*

El amplificador de microondas que se usa es el *MGA81563* que opera en el rango de 100 MHz a 6 GHz, que es incondicionalmente estable, tiene una ganancia de 12.4 dB a 2 GHz, una tensión de control de 3 V, una figura de ruido(Noise Figure, NF) de 2.8 dB a 2 GHz, y un punto de compresión a 1dB(P1dB) de 14.8 dBm a 2 GHz.

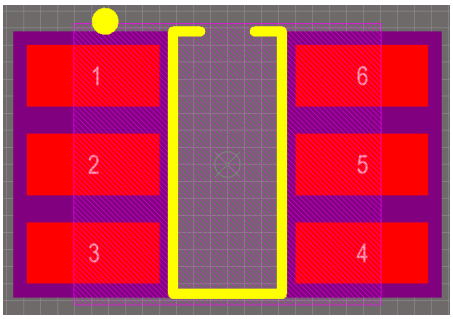


Figura 4.10 – *Footprint del amplificador de microondas.*

A continuación de la etapa de amplificación de RF, tenemos tres conexiones de las que dos van a dos conmutadores(uno para recepción y otro para transmisión) de dos salidas(una para la amplificación directa y otra para el mezclador) que dan lugar al esquemático siguiente que se trata de la etapa de frecuencia intermedia(IF); alternativamente la otra conexión va a dar lugar a dos filtros para frecuencias imagen(uno paso alta y en paralelo otro paso baja), que posteriormente serán seguidos por el mezclador. Este camino sólo se usará como se ha reiterado durante el texto, cuando se necesite variar la frecuencia desde o hacia una frecuencia fuera del rango de entre 2.15 y 2.75 GHz:

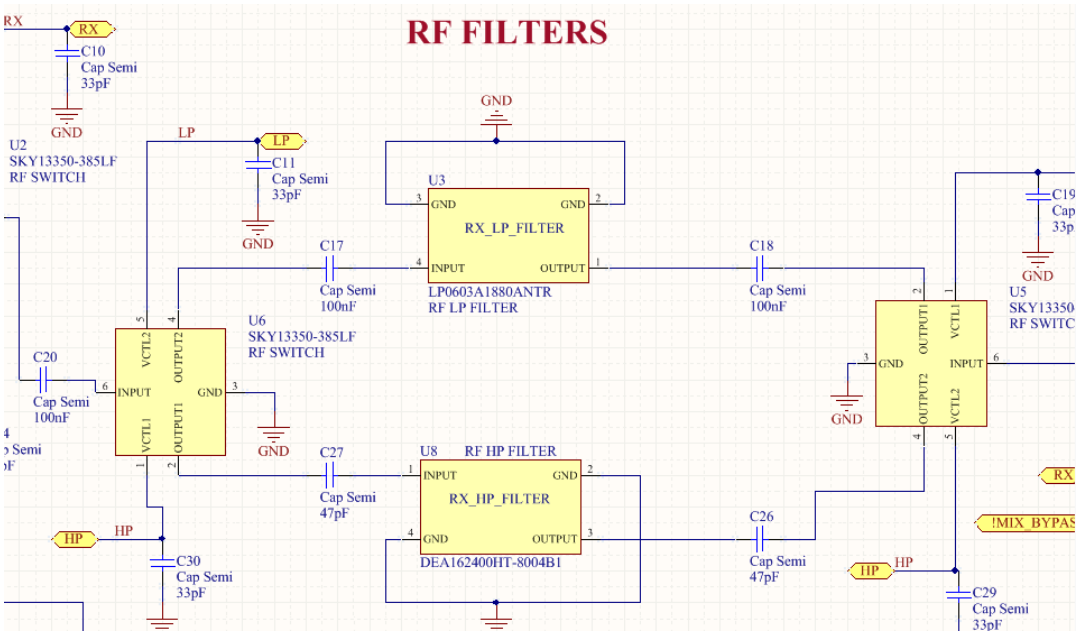


Figura 4.11 – *Filtros de frecuencia imagen.*

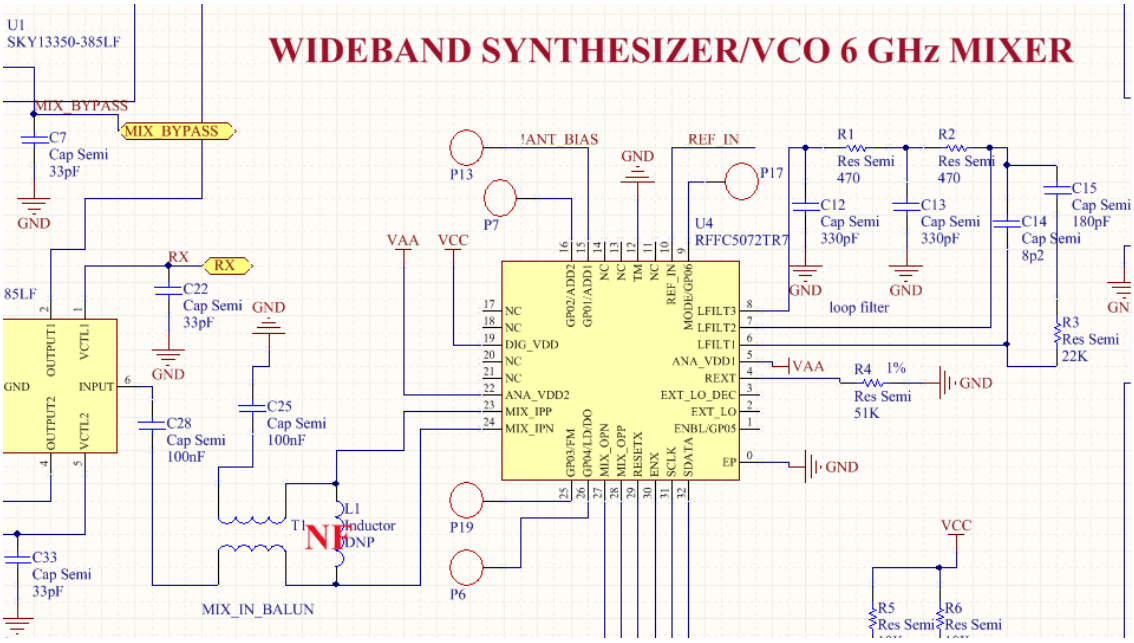


Figura 4.12 – *Esquematico del mezclador.*

El conmutador o switch que se usa en esta zona del esquemático es el *SKY13350*, tiene un rango de operación de entre 10 MHz a 6 GHz, una tensión de control de 0 / 1.6-5 V, una pérdida por inserción máxima de 0.35 dB a 3 GHz, un alto aislamiento de hasta 25 dB a 3 GHz, y un punto de compresión a 0.5dB(P0.5dB) de hasta +30 dBm:

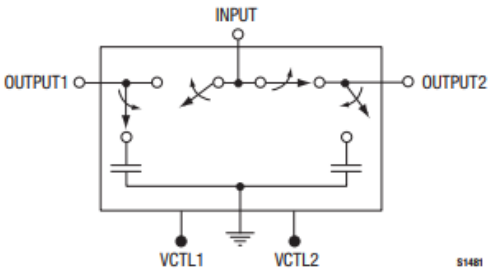


Figura 4.13 – *Diagrama del conmutador de 2 salidas.*

El mezclador que se usa en el esquemático es el *RFFC5072*, con un oscilador local de un rango entre 85 y 4200 MHz, con un sintetizador N-fraccional que puede conseguir pasos de hasta 1.5 Hz en cada frecuencia, lo que consigue mover la frecuencia de de entrada usando el mezclador entre 10 MHz y 6000 MHz. Además el mezclador tiene una alta linealidad, con un punto de intercepción de tercer orden de +23 dBm, tiene una interfaz serial de 3 o 4 conexiones, y está diseñado para tener un consumo muy reducido.

A continuación se mostrará el diagrama de bloques del mezclador, que muestra de forma global el modo de operación del mismo:

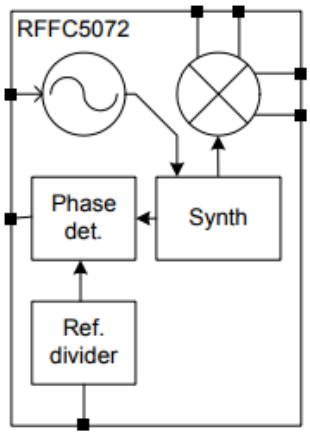


Figura 4.14 – *Diagrama del RFFC5072.*

También se mostrará el footprint o huella creado para el RFFC5072, así como el model 3D que fue creado para mostrar su tamaño y forma en un futuro diseño de un layout para circuito impreso:

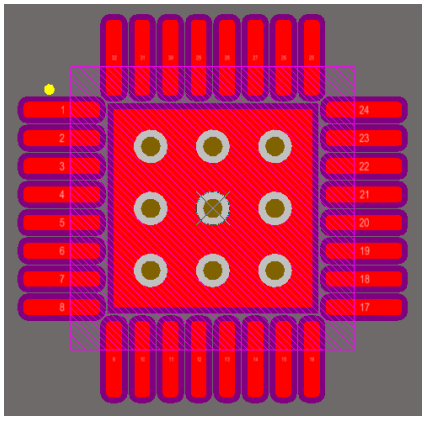


Figura 4.15 – *Footprint del RFFC5072.*

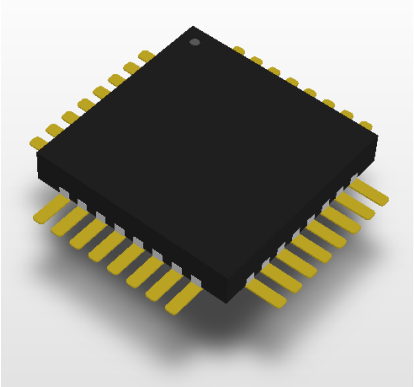


Figura 4.16 – *Modelo 3D del RFFC5072.*

4.4 Etapa de Baseband

El siguiente bloque del esquemático es el de Baseband, que recibe la señal del bloque de Frontend a través de los conmutadores de dos salidas del bloque anterior. En este bloque el transceiver recibe o transmite la señal(ya que es half-duplex y no soporta FDD, sólo TDD) al rango de 2.15 a 2.75 GHz, y posteriormente va al ADC/DAC dual que efectúa el cambio de domino de señales de analógico a digital y de digital a analógico, para pasar al bloque siguiente de la CPLD. En este esquemático también se incluye el reloj programable de múltiples salidas que se encarga de dar la señal de reloj, tanto al mezclador, como al tranceiver, al ADC/DAC y la CPLD(e incluso al microcontrolador maestro si se quiere); y que además también capacidad para sincronizarse con un reloj externo o hacer de maestro proporcionando un reloj externo a otro dispositivo.

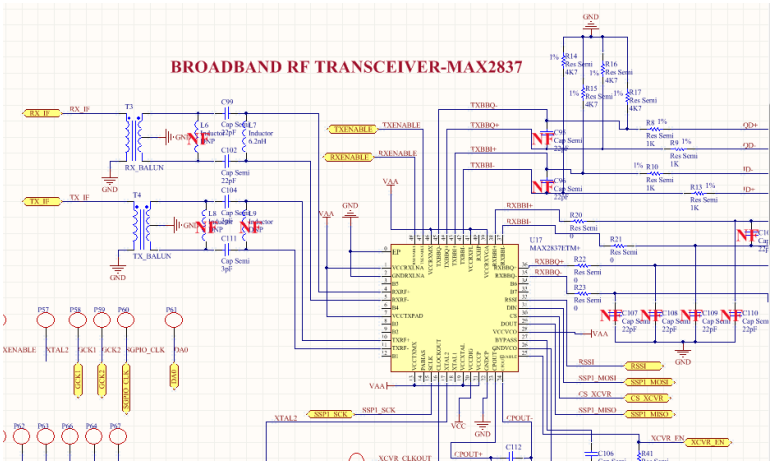


Figura 4.17 – *Zona del transceiver.*

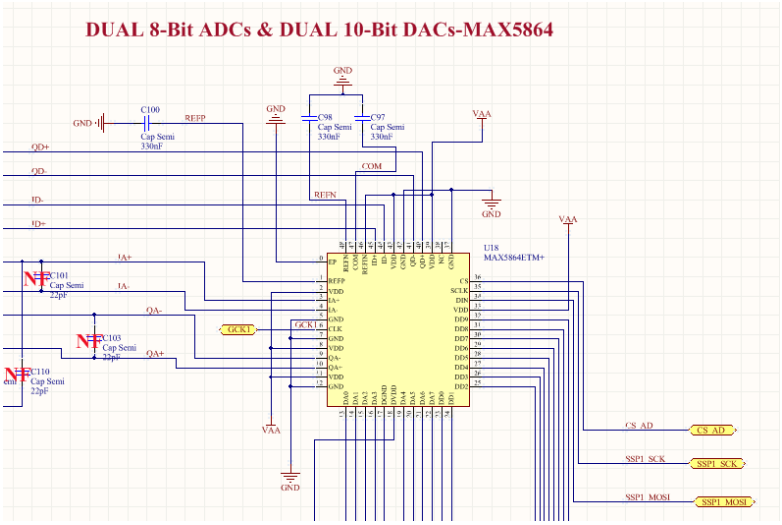


Figura 4.18 – *Zona del ADC/DAC.*

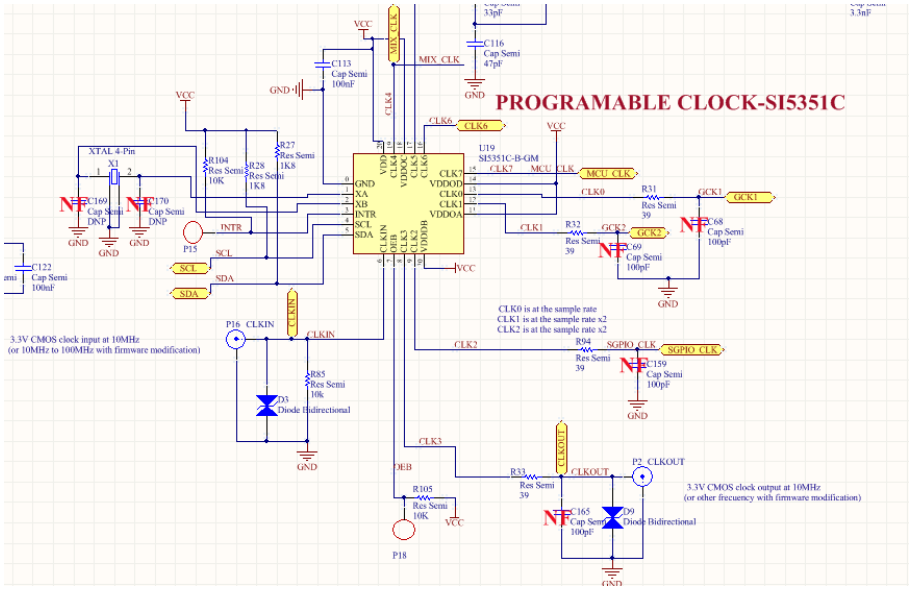


Figura 4.19 – *Zona del CLK Programable.*

El transceiver que se usa en el esquemático es el *MAX2837*, es un conversor directo de IF a banda base y viceversa con una serie de filtros configurables paso banda de hasta 28 MHz, tiene capacidad para un consumo muy reducido, contiene un amplificador de bajo ruido(LNA) para transmisión y recepción, así como un amplificador de ganancia variable extra para recepción(VGA). Opera transmitiendo las señales como componentes en fase y cuadratura en modo diferencial, por lo que para transmitir una señal debe usar 4 pistas. Es configurable usando 4 conexiones basadas en el protocolo SPI(Serial Peripheral Interface).

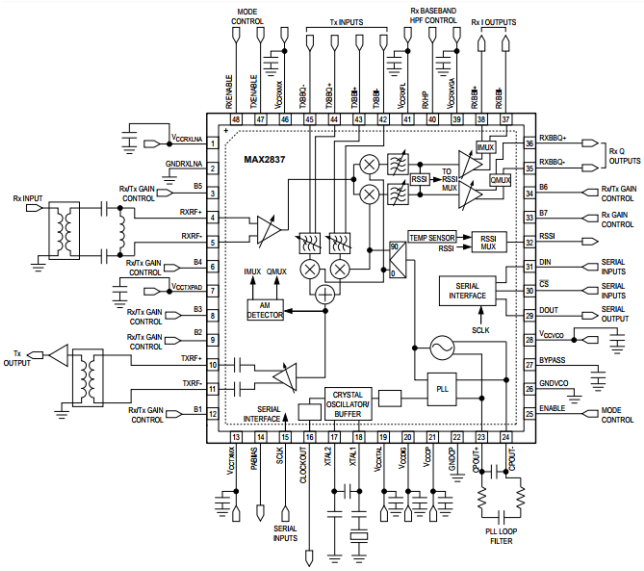


Figura 4.20 – *Zona del CLK Programable.*

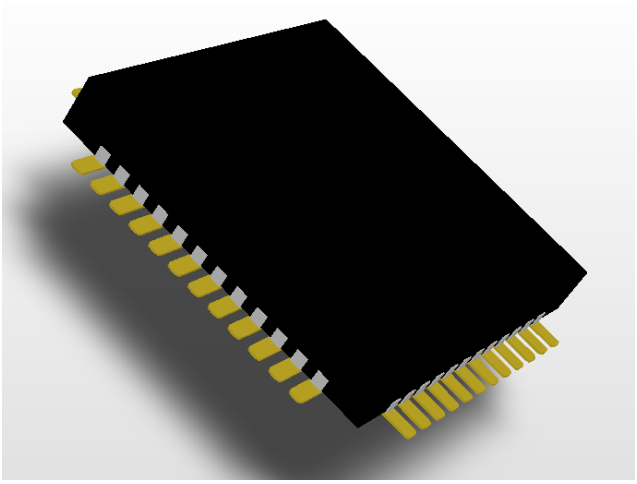


Figura 4.21 – *Modelo 3D del transceiver.*

El conversor ADC/DAC que se usa en el esquemático es el *MAX5864* capaz de funcionar hasta 22 MSPS, es un conversor dual con conversión analógico-digital de 8 bits en cuadratura y digital-analógico con 10 bits en cuadratura. Opera transmitiendo las señales como componentes en fase y cuadratura en modo diferencial, por lo que para transmitir una señal debe usar 4 pistas. Es configurable usando 4 conexiones basadas en el protocolo SPI(Serial Peripheral Interface). Finalmente tiene un consumo muy reducido, de tan sólo 42 mW a 22 MSPS.

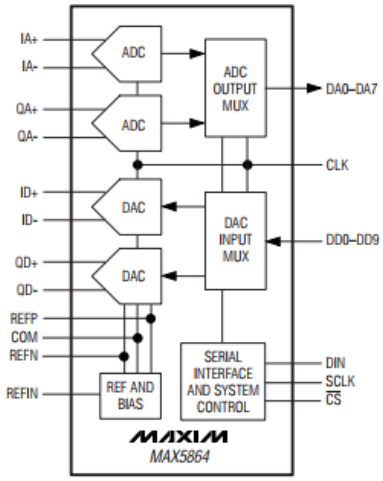


Figura 4.22 – *Diagrama del conversor ADC/DAC.*

Finalmente en este esquemático el reloj programable que se usa es el *SI5351C*, capaz de generar hasta 8 señales individuales de reloj de entre 2.5 KHz hasta 200 MHz de forma no interrelacionada, aunque sólo se pueden tener dos señales que superen los 114 MHz a la salida simultáneamente. Puede operar usando un cristal como referencia de entre 25 y 27 MHz. Tiene un error de 0 partes por millón(ppm) e integra un circuito de compensación por

variación de la temperatura. Sus salidas son configurables a 1.8, 2.5 y 3.3 V. Es configurable usando el protocolo I2C, y tiene un consumo muy reducido en potencia.

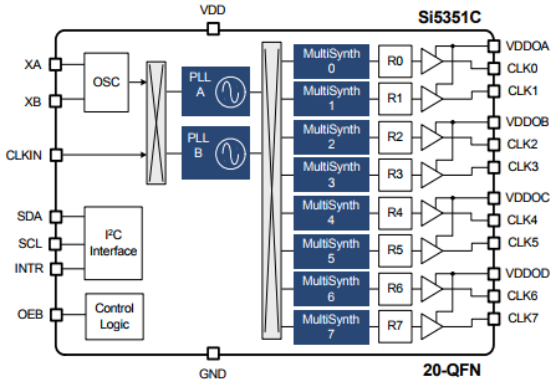


Figura 4.23 – *Diagrama del CLK programable.*

4.5 Etapa de ARM-CPLD

En esta última etapa del esquemático, la CPLD recibe o transmite la señal al convertidor ADC/DAC, una vez esta ha sido enviada desde el microcontrolador principal, de modo que la CPLD actúe como puente entre diferentes tipos de interfaces; por otro lado cabe destacar que la señal llegará a través del puerto USB 2.0 High Speed que se conecta al computador principal. El microcontrolador se encargará de comunicarse con el computador y de controlar que cada uno de los integrados del sistema operen adecuadamente. La configuración del dispositivo y la transmisión de las señales deseadas será llevada a cabo por el computador a través del puerto USB 2.0 sobre HackRF One.

El microcontrolador principal es el *LPC4320FBD144* de NXP, pertenece a la familia Cortex-M y consta de dos núcleos a 32 bits, un M4 y un M0+(de ultra bajo consumo), ambos puedes alcanzar frecuencias de hasta 204 MHz. Este procesador tiene una memoria SRAM de hasta 264 KB y soporta una gran cantidad de protocolos de comunicaciones entre los que se encuentran el SPI, I2C, USB 2.0 HS, SGPIO, EMC o Ethernet.

Figura 4.24 – *Imagen del encapsulado del LPC4320FBD144.*

La CPLD es un circuito lógico programable complejo, que se encarga de trasladar los datos entre el microcontrolador que funciona con cadenas de 8 bits que se comunican con el puerto USB 2.0, y el conversor ADC/DAC que opera con flujos de datos de 8 y 10 bits en cuadratura, por lo que se necesita un circuito lógico que adecúe los datos para poder ser tratados en ambos extremos. Para ello se usará la CPDL de la familia Cool-Runner II *XC2C64A*, que tiene 1500 puertas lógicas en conjuntos de 64 macroceldas. Además incorpora 3 relojes globales, tiene un retardo pin a pin de tan solo 4.6 ns(por ello se eligió una CPLD en lugar de una FPGA) y un consumo muy reducido.

Para poder configurar tanto el microcontrolador como la CPLD, es necesario el uso de una memoria flash que guarde la configuración de operación del dispositivo cada nuevo arranque. Para ello se optó por el uso del *W25Q80BV* que es una memoria flash de 8Mbits con interfaz serie SPI. Cabe destacar que para la configuración de la CPDL, ésta se conectará al microcontrolador a través de la interfaz JTAG.

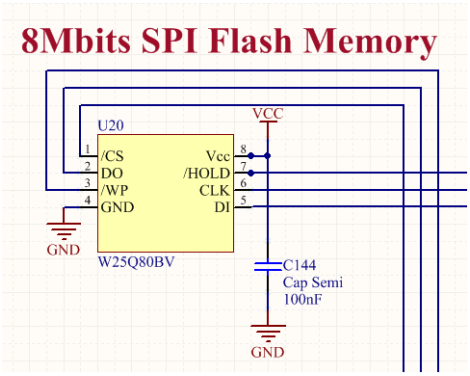


Figura 4.25 – *Zona de la memoria flash.*

Finalmente tenemos la fuente de alimentación de nuestra placa de circuito impreso:

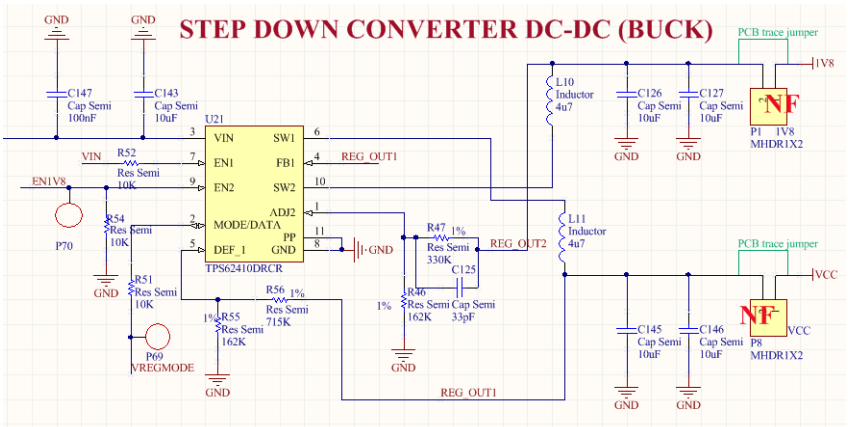


Figura 4.26 – *Zona de la fuente de alimentación.*

Esta fuente de alimentación conmutada se basa en el integrado *TPS62410* de Texas Instruments, que es un convertidor DC/DC reductor(topología Buck) de dos salidas, una a 3.3 V y otra a 1.8 V con una corriente máxima de salida en cada una de 800 mA. Tiene una eficiencia que puede alcanzar el 95% de la energía suministrada con una frecuencia de operación nominal de 2.25 MHz. Como tensión de entrada se utilizará la VDD del puerto USB 2.0 que tiene una tensión de 5 V y puede suministrar una corriente de hasta 500 mA.

4

CAPÍTULO

5

PROPUESTA DE ESQUEMÁTICO PARA HACKRF FULL-DUPLEX

En el presente capítulo se especificarán los bloques que compondrían una versión full-duplex de HackRF One y los cambios correspondientes en su esquemático:

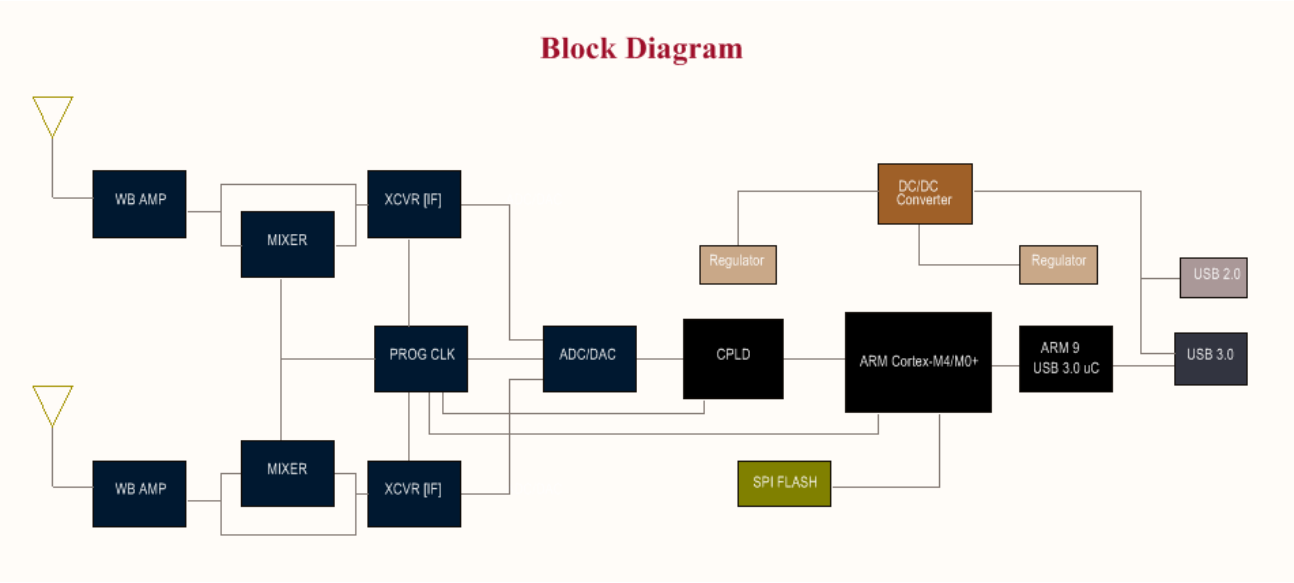


Figura 5.1 – *Diagrama de bloques de HackRF Full-duplex.*

Partiendo de la imagen anterior como diagrama de bloques fundamental, se citará y explicará cada uno de los bloques que componen el diagrama de forma que el lector pueda llegar a tener una imagen global del sistema, y pueda enlazar y entender el comportamiento de cada una de las partes incluidas en el mismo que serán detalladas a continuación:

- WB AMPL: es un amplificador de banda ancha(Wideband Amplifier) para radiofrecuencia, que se usa como etapa final de amplificación. Esta implementado tanto en la rama de recepción , como en la de transmisor.
- MIXER: es un mezclador, que se utiliza para poder manejar señales fuera del rango de 2.15 a 2.75 GHz(en cuyo caso la conexión con el bloque siguiente se establecerá de forma directa como se puede apreciar en el diagrama), con lo que una señal recibida fuera del rango anterior puede ser trasladada desde o hacia 2.15 y 2.75 GHz usando una portadora local generada en el mezclador; desplazando esa señal desde o hacia 2.15 a 2.75 GHz hasta cualquier frecuencia distinta entre 1 MHz y 6 GHz, usando una portadora local.
- XCVR [IF]: es un transceiver que puede recibir o transmitir señales desde o hacia la banda de 2.15 a 2.75 GHz directamente desde banda base, por lo que en esta aplicación se usa como un conversor a frecuencia intermedia(IF) para posteriormente pasar a la frecuencia final usando el mezclador, siempre que la frecuencia destino no se encuentre en la banda indicada anteriormente, en cuyo caso no será necesario su uso. Por lo que podemos decir que la aplicación usa una arquitectura de doble conversión de modo genérico, aunque entre 2.15 y 2.75 GHz se opte por usar una conversión directa.
- ADC/DAC: es un conversor dual que se compone de un ADC(Analog to Digital Converter) y un DAC(Digital to Analog Converter), además tiene capacidad de operación full-duplex, por lo que ambos conversores pueden funcionar simultáneamente.
- CPLD: es un dispositivo lógico programable complejo, que se usa a modo de interfaz entre el conversor ADC/DAC y el procesador del dispositivo para hacer de puente lógico entre los protocolos de recepción de datos de ambos bloques.
- ARM Cortex-M4/M0+: es un microcontrolador de 2 núcleos con arquitectura ARM perteneciente a la familia Cortex-M, que se caracteriza por ser la que tiene un menor consumo dentro de la gama Cortex, siendo el Cortex-M0+ el núcleo con menor consumo de todos los que ARM tiene bajo licencia.
- PROG CLK: es un integrado que nos permite generar varios relojes simultáneamente con una frecuencia independiente para cada nuevo reloj, a partir de un único cristal como referencia fundamental.
- SPI FLASH: es una memoria flash conectada al microcontrolador ARM, que puede ser reconfigurada, y que nos permite la configuración del microcontrolador principal y de la CPLD, cada vez que se inicie HackRF Full-duplex.

- DC/DC Converter: es un conversor reductor de voltaje(topología Buck), que forma la fuente de alimentación de la placa, y está alimentado por la tensión de entrada del puerto USB 3.0, que será complementada a su vez por la recibida por el USB 2.0.
- Linear Regulator: es un regulador de voltaje que permite obtener un voltaje fijo a la salida inferior al de la entrada manteniendo la estabilidad del mismo.
- ARM 9 USB 3.0 uC: es un integrado que opera como conversor entre interfaces, de forma que permite añadir una interfaz USB 3.0 a un IC maestro que cuente con interfaces RAM o SRAM nativas para su uso.
- USB 3.0: es una interfaz encargada de suministrar la alimentación que necesita el dispositivo HackRF Full-duplex, además tiene un canal para recibir datos y otro para transmitirlos a una frecuencia de hasta 5 Gbps, lo que la hace la interfaz idónea para implementar un dispositivo full-duplex.
- USB 2.0: es una interfaz encargada de suministrar la alimentación extra que necesita el dispositivo HackRF Full-duplex.

5.1 Frontend de HackRF Full-duplex

Para poder realizar el Frontend de HackRF Full-duplex, se decidió por separar cada camino del transceiver para recepción y emisión en dos etapas completamente separadas e independientes que nos diesen la posibiliada de poder operar el dispositivo en modo half-duplex(añadiendo un switch de alimentación a cada rama controlado por el microcontrolador, de forma que cuando una no se usase se suprimiese la alimentación) o full-duplex según se desease. Para ello se reorganizaron las conexiones de los conmutadores que seguían siendo necesarias por ejemplo para configurar el uso del los amplificadores de RF finales o no, que filtro de supresión de frecuencia imagen usar, o si se usaba el mezclador o no para una determinada señal. Además se optó por configurar la interfaz de comunicación de 3 conexiones seriales con los dos mezcladores de forma unívoca, quedando estos como esclavos del microcontrolador bajo una misma interfaz.

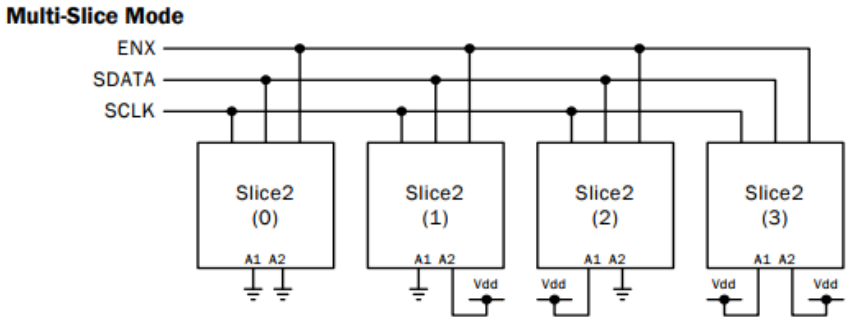


Figura 5.2 – *RFFC5072 en modo esclavo.*

A continuación se mostrarán las imágenes del Frontend de HackRF Full-duplex y de cada uno de sus 2 ramas transmisora y receptora:

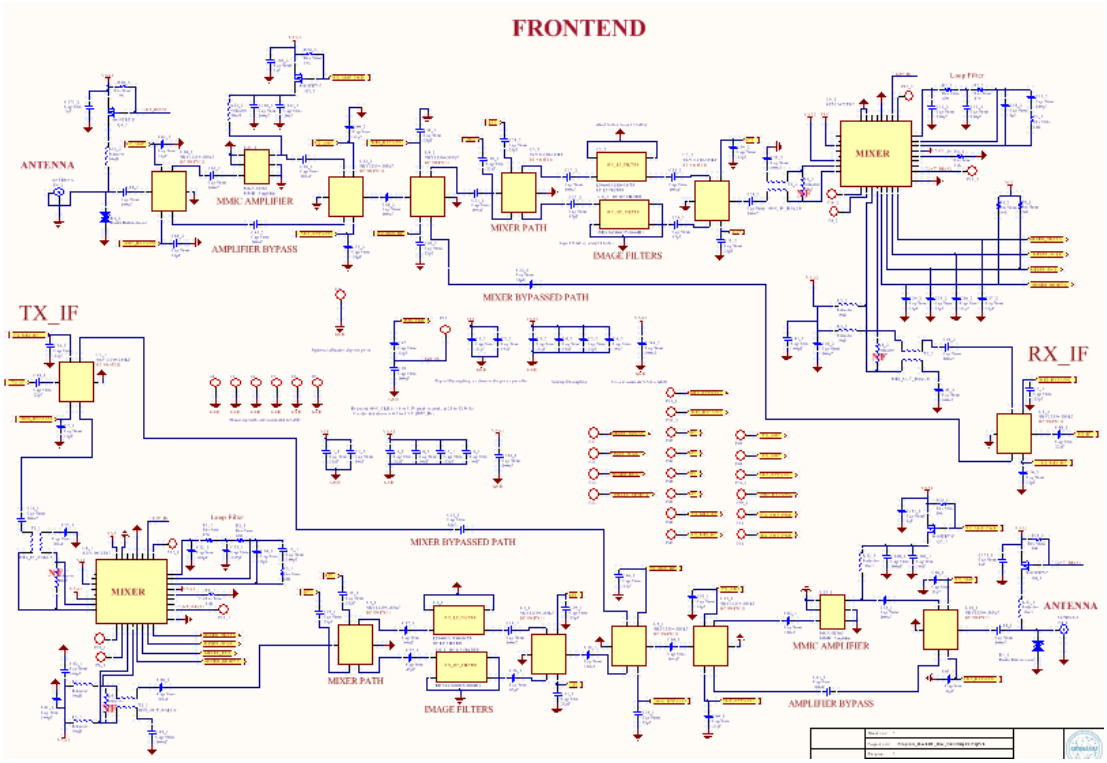


Figura 5.3 – *Frontend de HackRF Full-duplex.*

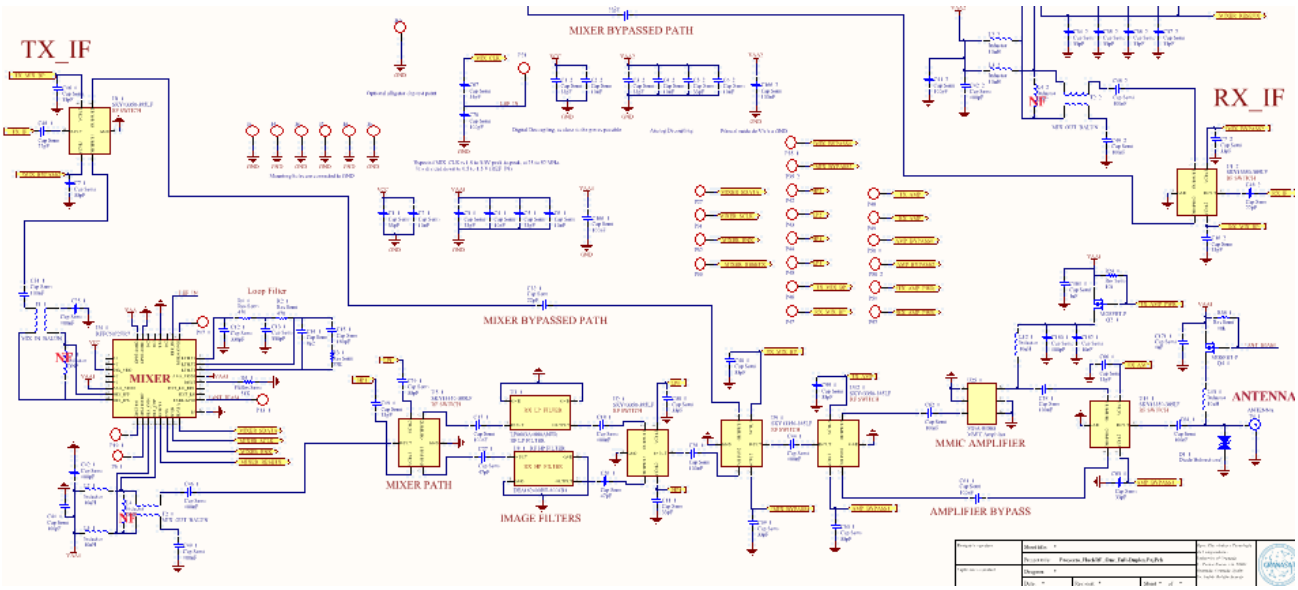


Figura 5.4 – *Transmisor de HackRF Full-duplex.*

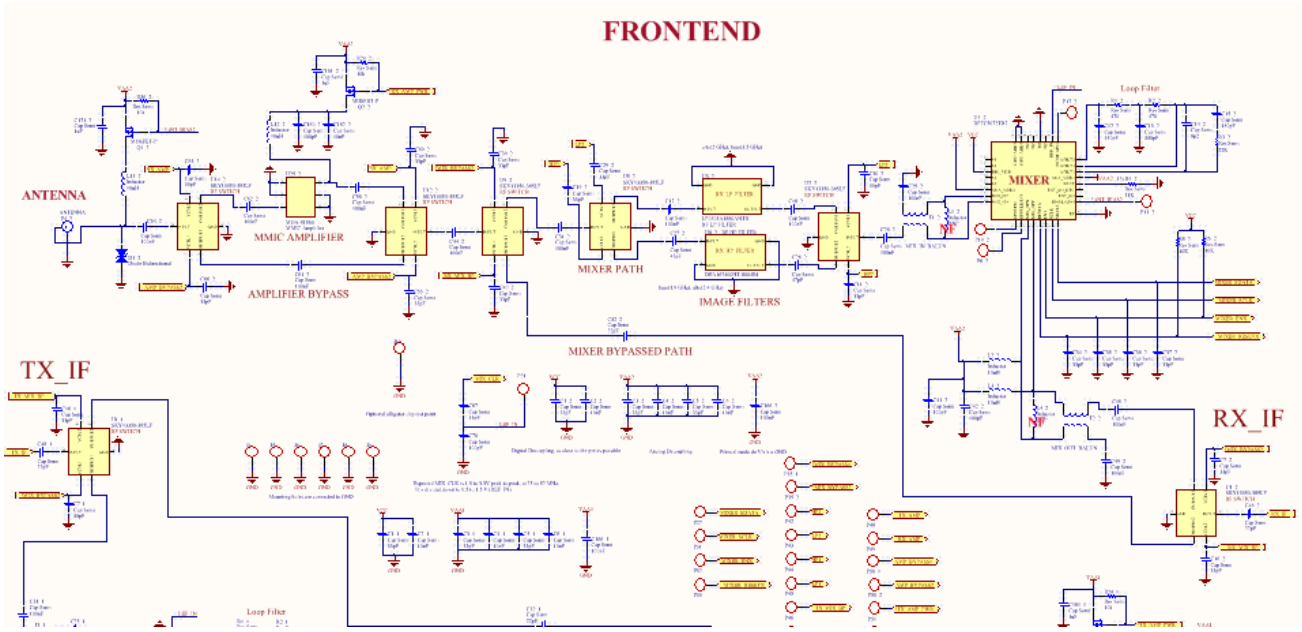


Figura 5.5 – *Receptor de HackRF Full-duplex.*

5.2 Baseband de HackRF Full-duplex

En la sección Baseband de HackRF Full-duplex, se decidió usar dos transceivers MAX2837 en vez de usar un transmisor y un receptor para cada rama. El circuito integrado MAX2837 ya integra un transmisor y un receptor, pero sólo pueden ser usados bajo un esquema TDD(Time Division Duplex), lo que impediría un funcionamiento simultáneo del dispositivo como transmisor y receptor. Ante lo cual, como se ha citado anteriormente se podría optar por buscar integrados específicos para cada funcionalidad; sin embargo para una primera versión del dispositivo esto implicaría cambiar gran parte del código software para su uso, y el profesor Andrés me sugirió usar una opción más conservadora.

Para una versión definitiva de HackRF Full-duplex, sin embargo si sería recomendable usar transmisores y receptores ad hoc para cada rama, a poder ser eligiéndolos de la compañía Maxim Integrated por su compatibilidad a nivel de envío y recepción de datos con el ADC/DAC MAX5864 y sus 4 líneas en modo diferencial con componente en fase y en cuadratura.

Como veremos en la imagen a continuación, para cada transceiver sólo se han usado los pines de alimentación correspondientes a su funcionalidad en cada rama, a diferencia de la conexión total de pines que se daba en HackRF One. Todas la conexiones con el MAX5864(conversor ADC/DAC) permanecen conectadas, pues éste puede operar sin ningún inconveniente en modo full-duplex.

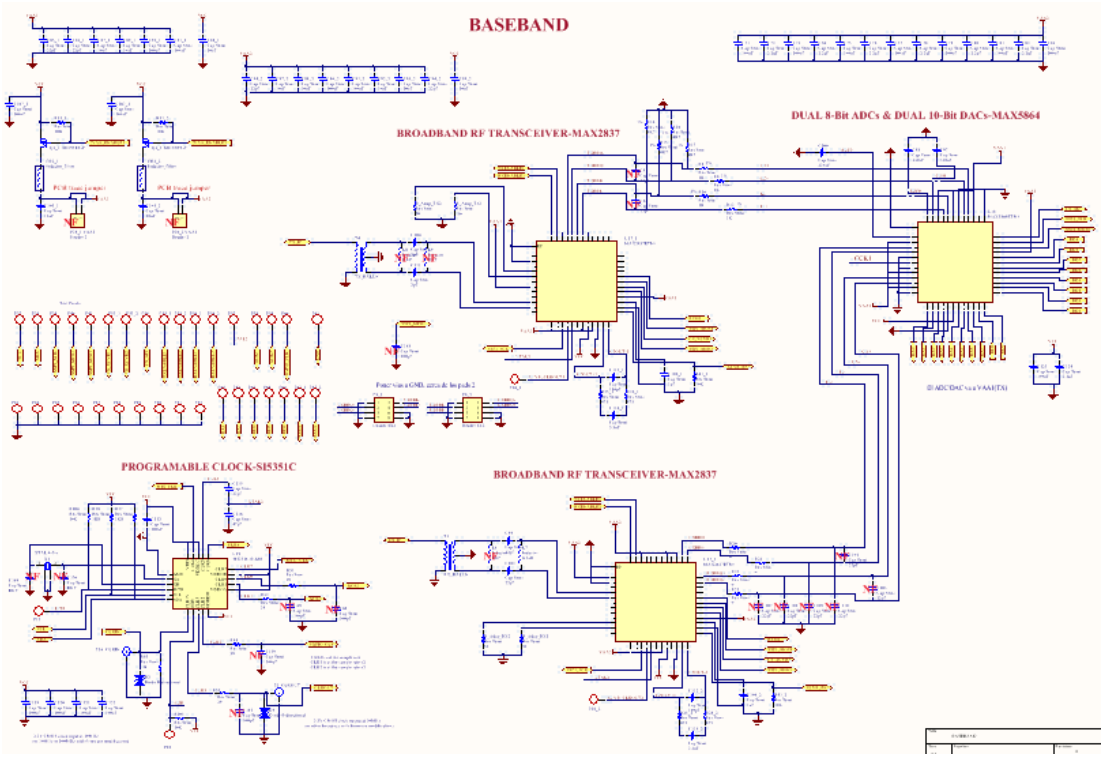


Figura 5.6 – *Baseband de HackRF Full-duplex.*

Debe añadirse a lo anterior, que ante la necesidad de implementar dos señales de reloj extra para un nuevo mezclador y un nuevo transceiver, se optó por hacer una bifurcación de las pistas de las señales de reloj existentes, tal y como se hizo en el layout de HackRF One, para sincronizar la CPLD y el conversor ADC/DAC:

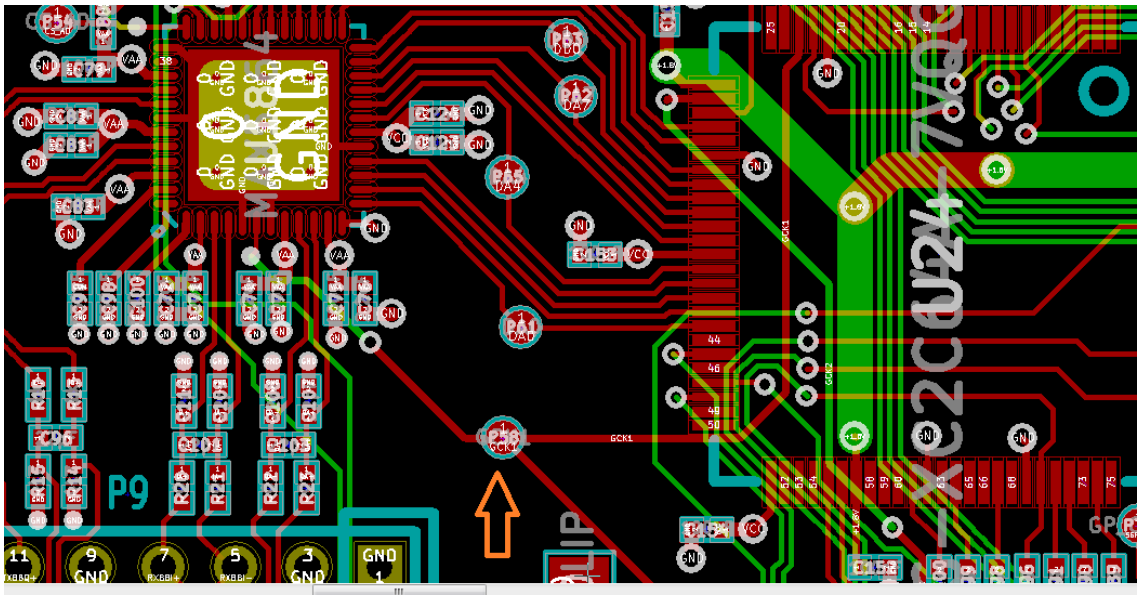


Figura 5.7 – *División reloj en el layout.*

Para comprobar, que en el caso de hacer las divisiones de las pistas de reloj en el layout no habría ningún inconveniente en el caso de los mezcladores y los transceivers, se simuló el comportamiento que tendría el reloj en cada pista dividida haciendo un análisis de transitorios con el software de simulación ADS(Advanced Design System). La simulación se llevó a cabo para un peor caso(las pistas divididas tenían una longitud mayor de la posible prevista en un layout). Además en las pistas divididas se mantuvo una impedancia característica de 50 Ohmios, tal y como se recomensaba en el Application Note del SI5351C.

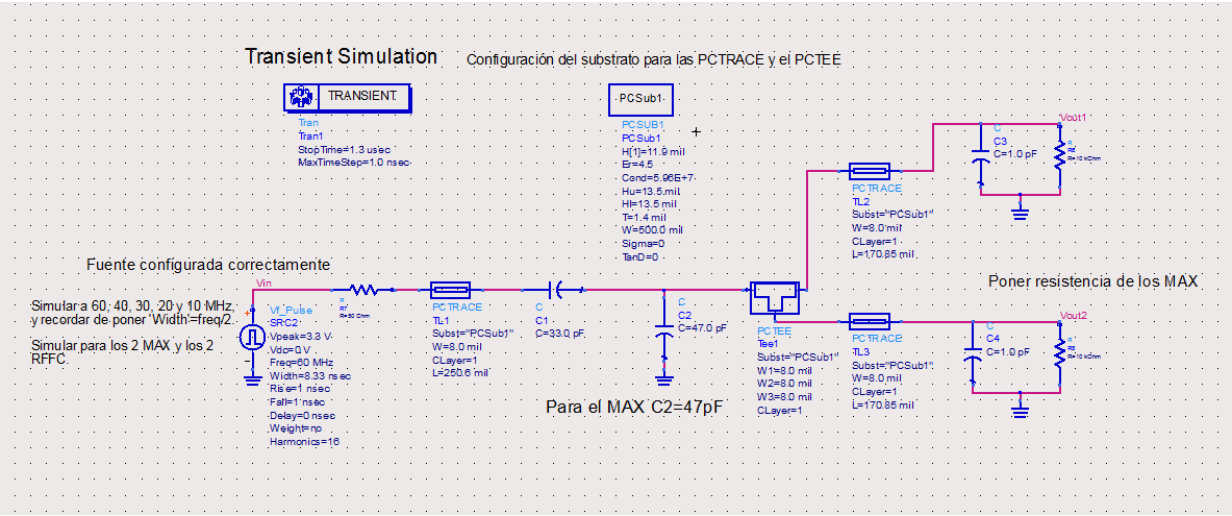


Figura 5.8 – *Esquema de simulación para el transceiver.*

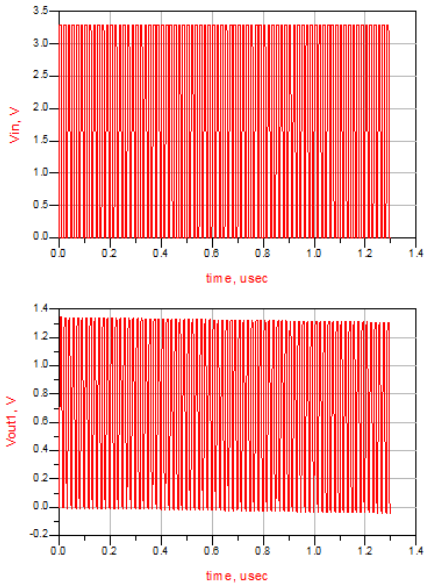


Figura 5.9 – *Resultado de la simulación.*

En la imagen superior, el resultado que se obtiene en la simulación es correcto y válido para nuestro propósito, ya que el transceiver MAX2837 requiere una tensión de referencia en la patilla del reloj de 1.2 V, que es la tensión que aparece en la imagen.

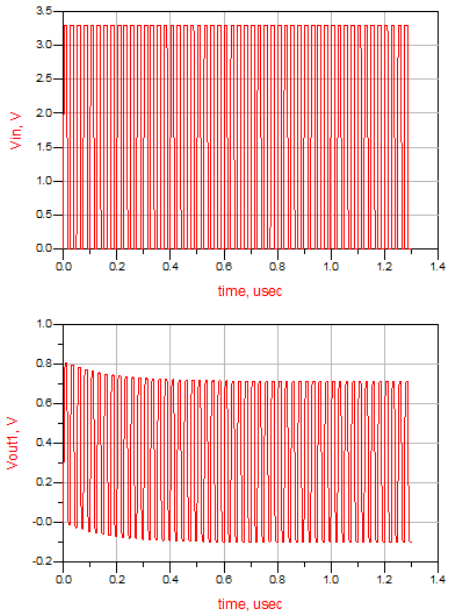


Figura 5.10 – *Resultado de la simulación.*

Para el caso de los mezcladores, vemos que también se cumple el requisito de operación correcta, ya que la tensión necesaria en el pin de reloj es de 0.8 V aproximadamente, tal y como aparece en la imagen superior.

5.3 ARM-CPLD de HackRF Full-duplex

La sección ARM-CPLD no llegó a implementarse a nivel de esquemático, debido a que no se había alcanzado un acuerdo sobre que dirección tomar de manera definitiva para implementar la interfaz USB 3.0, donde la opción de usar el integrado CYUSB3014 de Cypress podría llegar a generar problemas de incompatibilidad con el nuevo microcontrolador propuesto(se sustituía el LPC4320FBD144 por el LPC4357JBD208 por tener más pines disponibles y además una mayor memoria sram), debido a que sus interfaces no fuesen completamente compatibles(CYUSB3014 suele usarse conectado a una FPGA).

Otra posible opción sería la de conectar nuestra interfaz a la CPLD en lugar de al microcontrolador, pero todo ello supondría un enfoque distinto a nivel de diseño y un cambio en la orientación de todo el sistema.

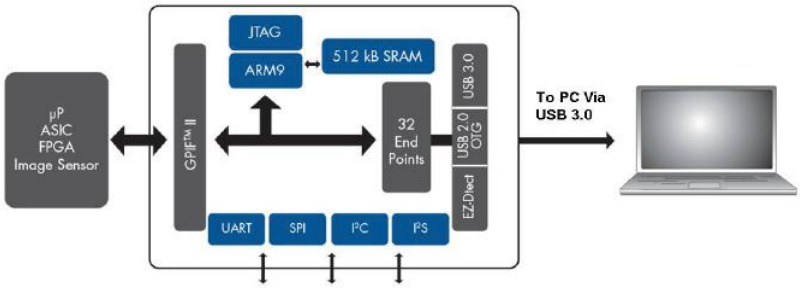


Figura 5.11 – *Diagrama del CYUSB3014.*

Por otro lado, se podría considerar la opción de usar un procesador de mayor potencia que incluyese de forma nativa la interfaz USB 3.0 y que simplificase sobre manera el conjunto del sistema. Ya que el uso de este tipo de procesadores de las familias de ARM Cortex-A15 y superiores podrían a su vez realizar la función de la CPLD debido a sus alta frecuencias de operación de hasta 1 GHz, lo cual tendría el inconveniente de incrementar de manera notable el consumo de la placa, comprometiendo así su portabilidad y su alimentación sobre la interfaz USB 3.0 en el peor de los casos reforzada usando una conexión adicional para alimentación a través de otra interfaz USB 2.0. De entre este tipo de opciones, el procesador que más destaca sería el LS1012A de la familia QorIQ de NXP.

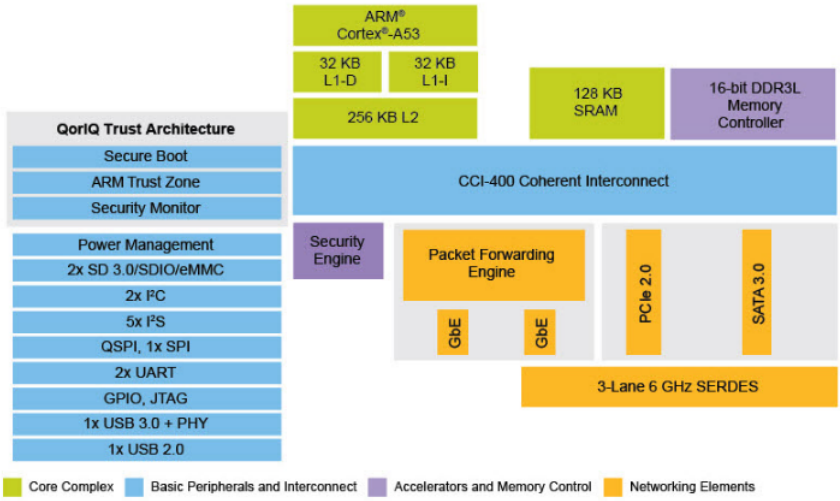


Figura 5.12 – *Arquitectura del LS1012A.*

5

REFERENCIAS

[1] Página oficial de HackRF One. URL:https://greatscottgadgets.com/hackrf/

[2] Proyecto HackRF One en Github. URL: https://github.com/mossmann/hackrf

[3] Definición de SDR. URL: https://en.wikipedia.org/wiki/Software-defined$_radio$

[4] Microcontrolador LPC4357JBD208 de NXP. URL: https://www.nxp.com/docs/en/data-sheet/LPC435X$_3X_2X_1X$.*pdf*

[5] Conmutador de 3 salidas SKY13317. URL: http://www.skyworksinc.com/uploads/documents/SKY13317$_3$73$LF_2$00914K.*pdf*

[6] Microcontrolador LPC4320FBD144 de NXP. URL: https://www.nxp.com/docs/en/data-sheet/LPC4350$_3$0$_2$0$_1$0.*pdf*

[7] Conmutador de 2 salidas SKY13350. URL: http://www.skyworksinc.com/uploads/documents/SKY13350$_3$85$LF_2$01174G.*pdf*

[8] Amplificador MMIC de 0.1 a 6 GHz MGA-81563. URL: http://www.efo.ru/components/avago/catalog/files/pdf/AV01$_0$190$EN.PDF$

[9] Filtro paso baja LP0603A1880ANTR. URL: http://datasheets.avx.com/lp0603.pdf

[10] Filtro paso alta DEA162400HT. URL: https://product.tdk.com/info/en/documents/data$_sheet/rf_hpf_dea162400ht - 8004b1_en.pdf$

[11] Es el mezclador RFFC5072. URL: http://www.qorvo.com/products/p/RFFC5072

[12] Balun a 2.5 GHz. URL: http://datasheetz.com/data/Transformers/Balun/712-1047-

1-datasheetz.html

[13] Introducción al SDR con SDR#. URL: https://wiki.gnuradio.org/index.php/Guided$_{T}utorials$

[14] Tutoriales de GNURadio. URL: https://hacking-etico.com/2016/05/05/introduccion-al-sdr/

[15] Página de ANT500. URL: https://greatscottgadgets.com/ant500/

[16] CPLD XC2C64A CoolRunner-II de Xilinx. URL: https://www.xilinx.com/support/documentation/data$_{s}heets/ds$311.*pdf*

[17] Cristal del SI5351C. URL: http://www.datasheetlib.com/datasheet/680816/cx3225ca30000d0hssz1$_{a}vx$ − *corporation.html*

[18] Transceiver MAX2837. URL: https://datasheets.maximintegrated.com/en/ds/MAX2837.pdf

[19] Conversor ADC/DAC MAX5864. URL: https://datasheets.maximintegrated.com/en/ds/MAX5864.pdf

[20] Reloj SI5351C. URL: https://www.silabs.com/documents/public/data-sheets/Si5351-B.pdf

[21] Conversor DC/DC de 2 salidas. URL: http://www.ti.com/lit/ds/slvs737a/slvs737a.pdf

[22] Memoria SPI Flash. URL: https://cdn-shop.adafruit.com/datasheets/W25Q80BV.pdf

[23] El reloj del RTC AB26TRQ. URL: http://www.mouser.com/ds/2/3/AB26TRQ-470991.pdf

[24] Cristal del microcontrolador 7V-12.000MAHE-T. URL: https://www.digikey.com/product-detail/en/txc-corporation/7V-12.000MAHE-T/887-1793-1-ND/3585984

[25] El controlador USB 3.0 CYUSB3014. URL: http://www.cypress.com/file/140296/download

[26] Conversor DC/DC de 2 salidas. URL: http://www.ti.com/lit/ds/symlink/tlv62065.pdf

[27] Regulador lineal para circuitos digitales. URL: http://www.ti.com/lit/ds/symlink/lp5907.pdf

[28] Regulador lineal de bajo ruido. URL: http://www.ti.com/lit/ds/symlink/lp3990.pdf