



TREBALL DE FI DE GRAU

GRAU EN ENGINYERIA DE SISTEMES DE TELECOMUNICACIÓ

MEDIDA DE LA SINCRONIZACIÓN ENTRE CANALES EN UNA USRP X300

Carlos Caballero Martínez

DIRECTOR: Gonzalo Seco Granados

DEPARTAMENT DE TELECOMUNICACIÓ I ENGINYERIA DE SISTEMES

UNIVERSITAT AUTÒNOMA DE BARCELONA

Bellaterra, Juny 25, 2019

Abstract

Communication systems based on Software Defined Radios (SDR) present an economic alternative to conventional communication systems. Through the use of the Universal Software Radio Peripheral (USRP) a wide variety of possible configurations are offered to implement applications.

The USRP X300 is a transceiver that offers four channels to receive signals. The use of several channels for the same application entails the need for them to be synchronized in time, phase and frequency, so making the measurement and seeing how the device behaves is important.

Initially, in this project, the concept of SDR is introduced and the operation of the USRP devices is explained. At the same time, the necessary material is presented to assemble a platform where synchronization can be measured. In addition, the relevant concepts of communication systems and the methods to perform the measurement are explained.

Within the experimental framework, this project focuses on using the transmitter as a source of a reference signal generating system that transmits the signal to all the channels of the receiver to be processed later. Finally, the measurement is made.

Resumen

Los sistemas de comunicación basados en Radios Definidas por Software (SDR) presentan una alternativa económica a los sistemas de comunicaciones convencionales. Mediante el uso de los dispositivos *Universal Software Radio Peripheral* (USRP), se ofrece una amplia variedad de configuraciones posibles para implementar aplicaciones.

La USRP X300 es un transceptor que ofrece cuatro canales para recibir señales. El uso de varios canales para una misma aplicación conlleva la necesidad de que estén sincronizados en tiempo, fase y frecuencia, por lo que realizar la medida y ver cómo se comporta el dispositivo es importante.

Inicialmente, en este proyecto, se introduce el concepto de SDR y se profundiza en el funcionamiento de los dispositivos USRP. Al mismo tiempo, se presenta el material necesario para realizar el montaje de una plataforma donde se pueda medir la sincronización. Además, se explican los conceptos relevantes de los sistemas de comunicaciones y los métodos para realizar la medida.

Dentro del marco experimental, este proyecto se centra en utilizar el transmisor como fuente de generación de una señal de referencia que se transmitirá a todos los canales del receptor para posteriormente ser procesada y que se realice la medida de sincronización.

Resum

Els sistemes de comunicació basats en Ràdios Definides per Software (SDR) presenten una alternativa econòmica als sistemes de comunicacions convencionals. Mitjançant l'ús dels dispositius *Universal Software Radio Peripheral* (USRP), s'ofereix una àmplia varietat de configuracions possibles per implementar aplicacions.

La USRP X300 és un transceptor que ofereix quatre canals per a rebre senyals. L'ús de diversos canals per a una mateixa aplicació comporta la necessitat que estiguin sincronitzats en temps, fase i freqüència, de manera que realitzar la mesura i veure com es comporta el dispositiu és important.

Inicialment, en aquest projecte, s'introdueix el concepte de SDR i s'aprofundeix en el funcionament dels dispositius USRP. Alhora, es presenta el material necessari per a realitzar el muntatge d'una plataforma on es pugui mesurar la sincronització. A més, s'expliquen els conceptes rellevants dels sistemes de comunicacions i els mètodes per a realitzar la mesura.

Dins el marc experimental, aquest projecte se centra en utilitzar el transmissor com a font de generació d'un senyal de referència que es transmetrà a tots els canals del receptor per posteriorment ser processada i que es realitzi la mesura de sincronització.

Agradecimientos

Primero de todo, me gustaría agradecer al Prof. Gonzalo Seco Granados la oportunidad de trabajar en este proyecto, de tener paciencia conmigo desde un principio, de guiarme en todo momento, de sugerirme ideas y de corregirme cuando más me equivocaba. Ha sido un placer tenerle como director y aprender de él.

También me gustaría agradecer a Daniel Egea Roca su compromiso y dedicación durante este periodo. Su aportación ha sido clave en el desarrollo de este proyecto.

Gracias a mi familia, en especial a mi madre y a mi hermano, por apoyarme incondicionalmente a lo largo del camino y por servirme como ejemplo de lucha incansable.

Por último, pero no menos importante, gracias a mis amigos. A mi trio favorito por estar siempre ahí, aunque no nos veamos, y a mis compañeros de carrera por hacer más llevaderos estos años y por enseñarme una visión diferente de la vida.

Carlos Caballero Martínez

25 de junio, 2019

Contents

Abstract	i
Resumen	iii
Resum	v
Acknowledgment	vii
List of Figures	xiv
List of Tables	xvi
Abbreviations	xvii
1 Introducción	1
1.1 Motivación y objetivos	2
1.2 Organización	2
2 Visión general de la Radio Definida por Software	5
2.1 Pasado, presente y futuro	5
2.2 Descripción de la SDR	6
2.3 Universal Software Radio Peripheral (USRP)	7
2.3.1 USRP Hardware Driver (UHD)	9
2.3.2 GNU Radio	9

3	Plataforma de medida de la sincronización entre canales	11
3.1	Sistema de medida de la sincronización	11
3.2	Tipos de sincronización	12
3.2.1	Sincronización en tiempo y frecuencia	13
3.2.2	Sincronización en fase	13
3.3	Descripción de los componentes	14
3.3.1	HF Citrine	14
3.3.2	Switch Netgear GS105	15
3.3.3	Splitter ZA4PD-2	17
3.3.4	USRP X300	18
3.3.5	USRP N200	19
4	Modelo de sistema de comunicaciones	23
4.1	Introducción al modelo de sistema	23
4.2	Modelo transmisor	24
4.3	El canal	25
4.4	Modelo receptor	25
4.5	Caracterización matemática del modelo de sistema	25
4.6	Retardo de fase y retardo de grupo	27
4.7	Señales periódicas	27
4.7.1	Exponencial compleja	27
4.7.2	Sinusoide real	27
4.7.3	Señal triangular real	28
4.7.4	Señal triangular compleja	28
4.8	Correlación cruzada y coeficiente de correlación	28
5	Implementación de diseño en GNU Radio	31
5.1	Diseño de un transmisor de señales complejas	31

5.2	Diseño de un transmisor de señales reales	35
5.3	Diseño de un receptor de cuatro canales en una USRP	36
6	Pruebas y resultados	41
6.1	P1: Validación de la señal recibida	41
6.1.1	Descripción	41
6.1.2	Resultados	43
6.2	P2: Tasa de muestreo	45
6.2.1	Descripción	45
6.2.2	Resultados	46
6.3	P3: Medida de la sincronización en tiempo entre canales	49
6.3.1	Descripción	49
6.3.2	Resultados	49
6.4	P4: Medida de la sincronización en fase y frecuencia (1)	53
6.4.1	Descripción	53
6.4.2	Resultados	54
6.5	P5: Medida de la sincronización en fase y frecuencia (2)	57
6.5.1	Descripción	57
6.5.2	Resultados	58
6.6	P6: Desfase entre las salidas del divisor de RF	61
6.6.1	Descripción	61
6.6.2	Resultados	62
7	Conclusiones y futuras líneas de investigación	65
7.1	Conclusiones	65
7.2	Futuras líneas de investigación	68

Bibliografía

List of Figures

2.1	Diagrama de bloques general de una SDR [Fä10]	7
2.2	Componentes UHD [Etttd]	8
2.3	Ejemplo de diseño en GNU Radio Companion [GNU18]	9
3.1	Visión general del sistema de medida	12
3.2	Diagrama de bloques de los componentes	14
3.3	HF Citrine 5-15 MHz [Wen]	15
3.4	Switch Netgear GS105 [NET]	16
3.5	Esquema eléctrico del divisor [Minb] y cable SMA [Mina]	17
3.6	Esquema de la USRP X300 [Etta]	19
3.7	Esquema de un cabezal de la TwinRX [Ettg]	20
3.8	Esquema de la USRP N200 [Ettf]	20
4.1	Diagrama de bloques de un sistema de comunicaciones digitales [Pro08]	24
5.1	Propiedades del bloque <i>Options</i>	32
5.2	Propiedades del bloque <i>Signal Source</i>	33
5.3	Propiedades del bloque <i>UHD: USRP Sink</i>	34
5.4	Transmisor USRP de señales complejas	35
5.5	Propiedades del bloque <i>Null Source</i>	35
5.6	Transmisor USRP de señales reales	36
5.7	Propiedades del bloque <i>UHD: USRP Source</i>	37

5.8	Receptor USRP de cuatro canales	37
6.1	Receptor USRP de cuatro canales	42
6.2	Transmisor USRP de señales reales	42
6.3	Primeras muestras de la señal recibida con 5 dB de ganancia según el canal . . .	43
6.4	Primeras muestras de la señal recibida con 20 dB de ganancia según el canal . .	43
6.5	Primeras muestras de la señal recibida con ganancias 37 dB, 42 dB, 37 dB y 55 dB según el canal	44
6.6	Primeras muestras de la señal recibida con ganancias 47 dB, 52 dB, 47 dB y 65 dB según el canal	45
6.7	Primeras muestras en módulo con canales sincronizados de la captura 1	50
6.8	Periodo de la captura 1 con canales sincronizados para un segundo de muestras .	51
6.9	Primeras muestras en módulo con canales no sincronizados de la captura 1 . . .	53
6.10	Fase y componentes real e imaginaria de una señal triangular real sin offset . . .	55
6.11	Fases de la captura 3 para cada canal	56

List of Tables

3.1	Especificaciones de referencia externa	15
3.2	Especificaciones del divisor y de los cables SMA	18
3.3	Especificaciones de la USRP X300 con TwinRX y la USRP N200 con SBX-40	21
5.1	Tipos de datos de salida	32
6.1	Parámetros iniciales	42
6.2	Resultados para la misma tasa de muestreo en transmisión y en recepción	47
6.3	Resultados para una tasa de muestreo de 5 Msps en transmisión	47
6.4	Resultados para una tasa de muestreo de 4.55 Msps en transmisión	48
6.5	Resultados para una tasa de muestreo de 0.5 Msps en transmisión	48
6.6	Retardo entre canales sincronizados en la captura 1	52
6.7	Retardo entre canales sincronizados en la captura 2	52
6.8	Retardo entre canales no sincronizados en la captura 1	52
6.9	Retardo entre canales no sincronizados en la captura 2	52
6.10	Desfase entre los canales en radianes	56
6.11	Diferencias entre la frecuencia moduladora y la frecuencia demoduladora en Hz para cada canal	57
6.12	Desfase entre los canales en radianes utilizando la señal de los LOs de un canal como referencia	58
6.13	Diferencias entre la frecuencia moduladora y la frecuencia demoduladora en Hz para cada canal utilizando la señal de los LOs de un canal como referencia	59

6.14	Desfase entre los canales añadiendo $\pm\pi/2$ y $\pm\pi$ radianes con respecto a la captura 1	59
6.15	Desfase entre los canales en radianes con reducción de 10 dB en ganancia	59
6.16	Desfase entre los canales añadiendo $\pm\pi/2$ y $\pm\pi$ radianes con respecto a la captura 1 con reducción de 10 dB en ganancia	59
6.17	Desfase en radianes entre los canales con un coseno	60
6.18	Desfase en radianes entre los canales con una señal triangular compleja	60
6.19	Desfase en radianes entre los canales con una exponencial compleja	60
6.20	Desfases entre salidas del divisor de RF	62

Abbreviations

- **ADC** Analog to Digital Converter
- **ASIC** Application-Specific Integrated Circuit
- **BW** Bandwidth
- **DAC** Digital to Analog Converter
- **DDC** Digital Down Converter
- **DTR** Data Transfer Rate
- **DUC** Digital Up Converter
- **EDA** Electronic Design Automation
- **FPGA** Field Programmable Gate Array
- **GPP** General Purpose Processors
- **HW** Hardware
- **IoT** Internet of Things
- **IF** Intermediate Frequency
- **LO** Local oscillator
- **MUSIC** Multiple Signal Classification
- **Msp**s Megasamples per second
- **OCXO** Oven-Controlled Crystal Oscillators
- **PLL** Phase-Locked Loop
- **PPS** Pulse-Per-Second
- **RF** Radio Frequency

- **RFI** Radio Frequency Interference
- **RFIC** Radio Frequency Integrated Circuit
- **SDR** Software Defined Radio
- **SMA** SubMiniature version A
- **SW** Software
- **UHD** USRP Hardware Driver
- **USRP** Universal Software Radio Peripheral
- **VCO** Voltage-Controlled Oscillator

Chapter 1

Introducción

Cuando se realiza una comunicación inalámbrica mediante un sistema de comunicaciones, el receptor del sistema captura las señales que percibe de la antena. Estas señales están formadas por ondas de radio deseadas, que contienen un mensaje de transmisión, y por ondas interferentes, que ocupan las mismas bandas de frecuencia y se superponen a las ondas deseadas. Estas ondas interferentes, también conocidas como interferencias, son un problema existente en muchas comunicaciones inalámbricas y es recomendable evitarlas.

Con el paso de los años, han ido surgiendo técnicas que permitían el filtrado del espacio, la más conocida es el *beamforming*. Esta técnica utiliza una matriz de pesos que asigna cada peso a una antena, haciendo que varias antenas trabajen a la vez formando una agrupación y dirigiendo el haz de radiación en una dirección concreta [Van88]. Esto posibilita el uso de aplicar algoritmos de estimación de la dirección de llegada de las señales (*Direction of Arrival*, DOA) para su posterior empleo en aplicaciones como RADAR, SONAR, etc.

Uno de los algoritmos utilizados para la estimación de la dirección de llegada de las señales es el algoritmo MUSIC (*Multiple Signal Classification*). Este algoritmo maximiza las funciones cuadráticas calculadas a través de los subespacios de señal y ruido, y se utiliza para estimar la dirección de llegada de fuentes de interferencia [Hay96].

La necesidad de trabajar con una agrupación de antenas hace que se precise de sistemas receptores de radiofrecuencia (*Radio Frequency*, RF) que dispongan de tantos cabezales de frecuencia como sea posible, y que estos cabezales estén sincronizados entre sí en tiempo, fase y frecuencia para el buen uso de la agrupación. Estos cabezales de frecuencia son conocidos como *front-ends*.

1.1 Motivación y objetivos

Los sistemas de comunicación basados en Radios Definidas por Software (*Software Defined Radio*, SDR) son una alternativa a los sistemas de comunicación convencionales ya que son económicos y ofrecen una amplia variedad de configuraciones para su aplicación. Este tipo de radios implementa una parte en hardware, que se encarga de recibir y transmitir ondas de radio, y una parte en software, que se encarga de establecer la comunicación entre host y hardware y de realizar el tratamiento digital de la señal en el host [Nat19a].

La parte en hardware la realizan los dispositivos transeceptores de SDR conocidos como *Universal Software Radio Peripheral* (USRP), mientras que la parte en software la realizan programas específicos compatibles con estos aparatos.

El propósito principal de este proyecto es medir la sincronización entre canales en un receptor periférico USRP X300 para poder ser empleado en el uso de agrupaciones de antenas. Para ello, el proyecto contempla los siguientes objetivos:

- Comprobar cuántos cabezales de frecuencia pueden funcionar al mismo tiempo en el receptor USRP X300.
- Medir la velocidad máxima de muestreo que puede alcanzar el receptor USRP a través de cada cabezal de frecuencia utilizándose el máximo de cabezales disponible.
- Medir la sincronización en tiempo, fase y frecuencia entre los distintos cabezales del receptor.

Para la obtención de resultados, se utilizan señales de referencia generadas mediante un transmisor USRP N200 y enviadas a través de un canal físico a todos los cabezales de frecuencia del receptor USRP X300. Esto conllevará la captura de datos reales que se someterán al tratamiento digital correspondiente para realizar la medida de sincronización.

1.2 Organización

En esta sección se resume la organización seguida a lo largo del proyecto.

En el Capítulo 1, se introducen los sistemas de Radio Definida por Software (SDR) y se presenta la motivación y objetivos que llevan a la realización de este proyecto.

En el Capítulo 2, se proporciona una visión general más amplia sobre los sistemas de SDR y se incluye el funcionamiento de los dispositivos USRP y cómo se realiza la comunicación entre host y hardware.

En el Capítulo 3, se presentan los tipos de sincronización y el material necesario para poder montar una plataforma de medida de la sincronización entre canales.

En el Capítulo 4, se presenta el modelo de sistema de comunicaciones que sigue la plataforma y se explican algunos conceptos relacionados para la realización de las medidas.

En el Capítulo 5, se trabaja con el entorno de diseño GNU Radio con el que se realiza la comunicación entre host y dispositivos USRP.

En el Capítulo 6, se expone la descripción de las pruebas realizadas para medir la sincronización, aparte de desarrollar y comentar los resultados obtenidos.

Finalmente, el documento termina con el Capítulo 7 donde se concluye y se presentan futuras líneas de investigación.

Chapter 2

Visión general de la Radio Definida por Software

2.1 Pasado, presente y futuro

El término Radio Definida por Software tiene aproximadamente 30 años y, en todos estos años, todavía no se han puesto de acuerdo en dar una definición apropiada. Según el antiguo foro sobre SDR, conocido como *SDR Forum*, y que ahora se conoce por el nombre de *Wireless Innovation Forum* [Wir], SDR es “una radio en la que algunas o todas las funciones de la capa física están definidas por software”, dando a entender que SDR se enfoca principalmente en el procesado de la forma de onda en la capa física sin relacionarla con el cabezal de RF.

Actualmente, las SDR se han convertido en el estándar de las industrias en el mercado, cambiando así los circuitos integrados para aplicaciones específicas (*Application-Specific Integrated Circuit*, ASIC) por procesadores digitales de señales (*Digital Signal Processing*, DSP) que son programables.

La aparición y combinación de los circuitos integrados de RF (RFIC) y las FPGA (*Field-Programmable Gate Array*) de DSP satisficieron la necesidad de la radio táctica militar en su momento, por lo que el desarrollo de las SDR para su aplicación en radios militares fue financiado. Todo esto logró mejoras en herramientas de programación para la automatización de diseños electrónicos (*Electronic Design Automation*, EDA) y para compañías de semiconductores.

El progreso de las herramientas RFIC, FPGA y EDA permitió la llegada de la segunda generación de SDR que se implementó en infraestructuras 4G LTE. La tercera generación de SDR llegó al integrarse núcleos DSP en teléfonos 4G LTE, que más tarde llegarían para realizar el procesamiento de la capa física en los ASIC.

En un futuro se espera que las SDR actualmente implementadas en teléfonos 4G puedan llegar a mejorarse para su utilización en 5G e *Internet of Things* (IoT) [Nat19b].

2.2 Descripción de la SDR

SDR es un sistema de comunicaciones por radio que tiene como objetivo enviar o recibir información dependiendo de si se trata de un emisor o un receptor, donde el comportamiento del hardware viene definido por las instrucciones proporcionadas mediante software. Esto permite múltiples funciones, bandas frecuenciales, amplificaciones y modos que pueden ser controlados mediante aplicaciones. Las FPGA, los DSP o los procesadores de propósito general (*General Purpose Processors*, GPP) mejoran y agregan nuevas funciones a la radio a través de actualizaciones del sistema.

En la Figura 2.1, se puede observar el diagrama de bloques general de una SDR donde se pueden distinguir tres secciones diferentes [Fä10]. La primera sección, donde se mencionan las tarjetas secundarias o *daughterboards*, es la encargada de actuar como interfaz con el mundo analógico, emitiendo y capturando señales de RF y realizando el traslado de las señales a frecuencia intermedia (*Intermediate Frequency*, IF). La segunda sección, donde se menciona la tarjeta principal o *motherboard*, es la encargada de realizar la conversión del dominio analógico a digital y viceversa. La tercera y última sección es la encargada de realizar el procesado digital de la señal.

Haciendo un poco más de hincapié en cada sección se obtiene que:

- **La sección de RF** permite capturar o emitir señales de RF a través de los puertos SMA (*SubMiniature version A*) donde se pueden conectar antenas o cables SMA. Cada puerto o conjunto de puertos SMA está conectado a una o varias tarjetas secundarias que representan los cabezales de RF. En la Figura 2.1 se puede distinguir dos caminos.
 - **El camino superior** se dedica a la recepción de la señal de RF. Esta señal se amplifica para intentar recuperar el nivel de potencia que tenía la señal al ser transmitida, se filtra con el fin de recibir la señal deseada mientras se rechazan las señales no deseadas, y se mezcla para realizar la traslación a frecuencias IF. Este camino ajusta la señal de RF recibida para que pueda ser tratada por el conversor analógico a digital (*Analog to Digital Converter*, ADC).
 - **El camino inferior** se dedica a la emisión de la señal de RF. La amplificación y el filtrado se realizan de la misma manera que en el receptor, pero el mezclador traslada la señal de frecuencia IF a frecuencia RF. Este camino ajusta la señal de frecuencia IF procedente del conversor digital a analógico (*Digital to Analog Converter*, DAC) para ser enviada a través del puerto SMA.

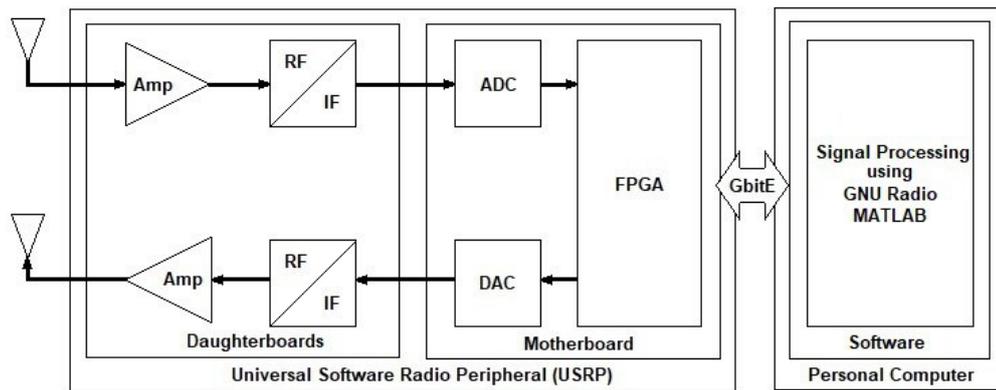


Figure 2.1: Diagrama de bloques general de una SDR [Fä10]

- **La sección de conversión de analógico a digital y viceversa** la realiza la tarjeta principal y está conectada con la tarjeta secundaria a través de los ADC y DAC. En esta sección también se distinguen dos caminos:
 - **El camino superior** se encarga de muestrear las señales analógicas mediante el uso del ADC y de realizar el mezclado a banda base a través del convertidor digital de bajada (*Digital Down Converter*, DDC) que hay dentro de la FPGA. La velocidad de ejecución depende de la tasa de muestreo del ADC.
 - **El camino inferior** se encarga de realizar la traslación de banda base a frecuencia IF a través del convertidor digital de subida (*Digital Up Converter*, DUC) y de realizar la conversión al dominio analógico a través del DAC. La velocidad de ejecución depende de la tasa de muestreo del DAC.

La cantidad de bits que se transmiten a través del cable de red Ethernet depende principalmente de la tasa de muestreo que se haya empleado en la FPGA.

- **La sección de procesamiento digital** se realiza en el ordenador personal o host con el que se esté controlando el dispositivo de radio. La conexión se realiza directamente entre la tarjeta principal y el host a través de un cable de red Ethernet. GNU Radio o MATLAB permiten el procesamiento de señales a través bloques o código.

2.3 Universal Software Radio Peripheral (USRP)

Universal Software Radio Peripheral (USRP) es el nombre con el que se conoce al dispositivo transceptor de Radio Definida por Software. Estos dispositivos ofrecen al usuario la posibilidad de transmitir y recibir todo tipo de formas de onda en rangos de frecuencia de hasta 6 GHz

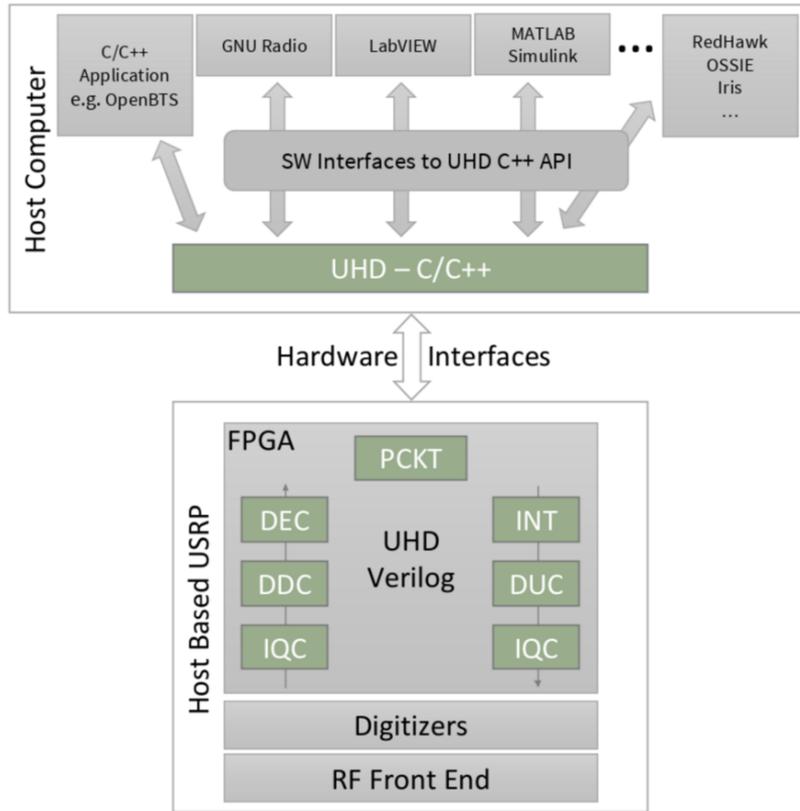


Figure 2.2: Componentes UHD [Etttd]

con anchos de banda instantáneos de hasta 56 MHz según la tarjeta secundaria instalada en su interior. Además, hacen posible la generación rápida de prototipos permitiendo la creación de aplicaciones inalámbricas avanzadas [Nat19a].

Aunque existen varios fabricantes de plataformas SDR como Ettus Research, National Instruments y FlexRadio entre otros, en este proyecto se utilizarán dos dispositivos USRP pertenecientes al fabricante Ettus Research. Toda USRP consta de dos niveles de tarjetas, una tarjeta principal conocida como *motherboard* y una o varias tarjetas secundarias conocidas como *daughterboards*. Los dos dispositivos USRP disponibles en la Universidad son la USRP N200 con una SBX-40 y la USRP X300 con dos TwinRX, más adelante se introducirán sus características.

Para establecer comunicación con el dispositivo SDR es necesario instalar unos drivers tanto en la USRP como en el host desde donde se quiere controlar. Los drivers necesarios se conocen por las siglas UHD, del inglés *USRP Hardware Driver*. Posteriormente, se necesitará instalar alguna plataforma de desarrollo software para utilizar estos controladores, en este proyecto la plataforma utilizada será GNU Radio.

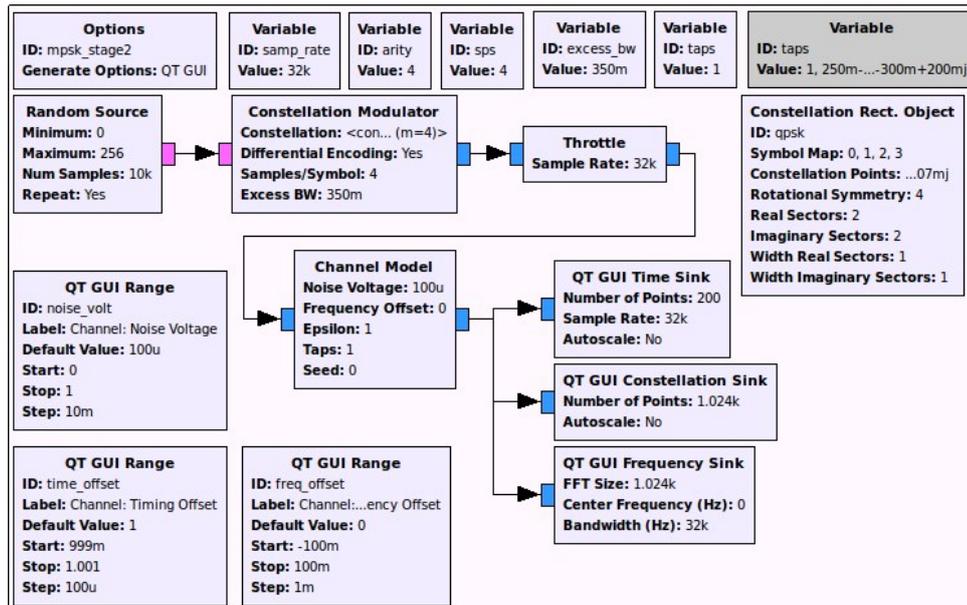


Figure 2.3: Ejemplo de diseño en GNU Radio Companion [GNU18]

2.3.1 USRP Hardware Driver (UHD)

UHD es una librería que se ejecuta en el procesador GPP del host y que permite la comunicación y el control de los dispositivos USRP. Estos dispositivos envían y reciben muestras del ordenador donde UHD ha sido instalado. Esta transmisión de muestras se efectúa a través de los protocolos IP y UDP, lo que significa que se puede tener más de un dispositivo USRP conectado al ordenador y trabajar con ellos independientemente ya que se encuentran identificados a través de la dirección IP.

Los drivers hay que instalarlos tanto en la FPGA de la USRP para que se realice la comunicación entre el ordenador y el propio dispositivo, como en el host para que el usuario pueda enviar la orden de capturar o transmitir datos a través de la conexión Ethernet. El controlador que se instala en el ordenador está escrito en código C/C++, mientras que el que se instala en la USRP está escrito en Verilog [Nat19a].

Para utilizar los controladores UHD es necesario recurrir a aplicaciones de desarrollo de software como pueden ser las indicadas en la Figura 2.2 (GNU Radio, LabVIEW, MATLAB-Simulink, etc).

2.3.2 GNU Radio

GNU Radio es una plataforma de desarrollo de software de código abierto y gratuita que proporciona un conjunto de herramientas a través de bloques de procesamiento de señal para radios

definidas por software [GNU18]. La plataforma de GNU Radio habilita al usuario una interfaz gráfica denominada GNU Radio Companion (GRC) donde se facilita al usuario dichos bloques para la generación de formas de onda, codificación de canal, elementos de sincronización, elementos de comunicación entre USRP y ordenador, elementos de almacenamiento y otros tipos de bloques disponibles (véase la Figura 2.3).

El diseño confeccionado a través de GRC se almacena como un código de ejecución Python que contiene toda la información del diseño, desde los parámetros relevantes de cada bloque (ganancia de la antena, antena que se utiliza, frecuencia de muestreo, frecuencia central de modulación, etc) hasta las interconexiones entre bloques.

Esta plataforma de desarrollo es implementable con hardware de RF externo como pueden ser los dispositivos USRP, o sin hardware en un entorno de simulación. Además, la instalación de GNU Radio proporciona al usuario varios ejemplos de aplicación que pueden ser útiles. Algunos de los ejemplos disponibles se utilizan para: capturar señal y almacenarla en un archivo (`uhd_rx_cfile.py`), analizar el espectro de la señal introducida (`uhd_fft.py`), etc.

Chapter 3

Plataforma de medida de la sincronización entre canales

Un sistema receptor con múltiples cabezales de frecuencia RF puede requerir diferentes niveles de sincronización entre canales dependiendo de la aplicación. Dado que el objetivo principal de la plataforma es medir la sincronización entre canales, es necesario utilizar una fuente que genere una señal de referencia que se reciba por todos ellos y que permita compararlos.

En este capítulo se presenta una visión del sistema propuesto, los métodos para aplicar sincronización mediante hardware y/o software y la descripción del hardware utilizado en el montaje.

3.1 Sistema de medida de la sincronización

En la Figura 3.1 se presenta una visión general del sistema de medida, en él se puede observar que está compuesto por dos bloques principales y cada uno, a su vez, se compone por dos bloques secundarios.

El primer bloque principal está formado por el transmisor de la señal de referencia, compuesto por los bloques secundarios de hardware y software. El bloque secundario de hardware contiene un dispositivo USRP N200 y la sincronización mediante hardware correspondiente. El bloque secundario de software contiene el programa utilizado de GRC para la generación de la señal de referencia y también incluye la sincronización necesaria mediante software.

El segundo bloque principal está formado por el receptor de múltiples cabezales también compuesto por los bloques secundarios de hardware y software. El bloque secundario de hardware contiene un dispositivo USRP X300 con cuatro cabezales de frecuencia RF independientes entre

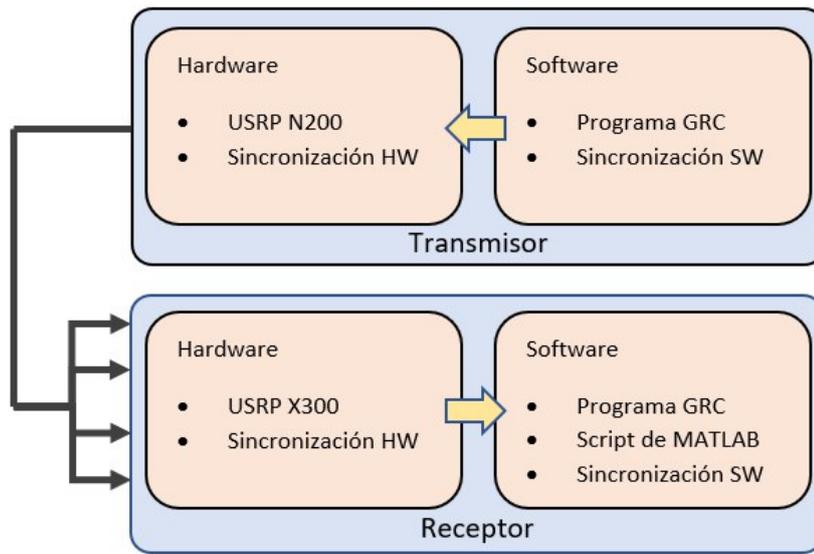


Figure 3.1: Visión general del sistema de medida

sí. Además, también incluye la sincronización mediante hardware al igual que en el caso del transmisor. El bloque secundario de software contiene el programa utilizado de GRC para la captura de la señal de referencia por los cuatro cabezales y el script MATLAB con el que se calcula la medida de sincronización. En las siguientes secciones se entrará en detalle en cada uno de los bloques secundarios.

Para alimentar al receptor por todos sus cabezales con la señal de referencia producida por el transmisor, será necesario utilizar un dispositivo divisor de potencia conocido como *splitter*. Además, para establecer la comunicación y controlar los sistemas de radio definida por software, se necesitará un ordenador personal que actuará como host, un switch para comunicar las USRPs y el host, y los correspondientes cables SMA y Ethernet para realizar el enlace. Por último, como se mencionó en el Capítulo 2, la comunicación se llevará a cabo mediante el software GNU Radio.

3.2 Tipos de sincronización

Como en cualquier otro sistema receptor con múltiples cabezales, existen distintos niveles de sincronización entre canales. Esta sincronización puede ser en tiempo, fase y frecuencia, por lo que es necesario saber que posibilidades ofrecen los dispositivos USRP para realizar la sincronización y así añadir o no hardware externo a la plataforma. A continuación, se muestran los tipos de sincronización permitidos por una USRP [Ette].

3.2.1 Sincronización en tiempo y frecuencia

Los dispositivos USRP pueden coger dos señales de referencia que les permite sincronizar los relojes y la hora. Estas señales de referencia son:

- Un reloj externo de referencia de 10 MHz que proporciona una referencia de una sola frecuencia a todos los dispositivos USRP conectados.
- Una señal de referencia de un *pulse-per-second* (PPS) que sincroniza la muestra de tiempo entre todos los dispositivos USRP conectados.

Además, existe un método alternativo donde se transmite un mensaje de tiempo codificado que permite la sincronización entre dispositivos a través de un cable MIMO.

En este proyecto, no hay interés en utilizar la señal de referencia PPS ya que el receptor no tiene porque empezar a capturar en la misma muestra de tiempo en la que el transmisor está emitiendo. Tampoco es necesario utilizar el método MIMO por el mismo motivo. Si se activara la configuración MIMO, la USRP esclava se sincronizaría en tiempo a la USRP principal, incluyendo la señal de reloj y la de hora.

Ambos métodos serían útiles si se tuviera varios dispositivos USRP para recibir, en ese caso los dispositivos sí que deberían estar alimentados por una señal de referencia PPS o mediante la configuración MIMO.

Cabe destacar el hecho de que esta sincronización se produce entre dispositivos USRP. Como se tiene una USRP para transmitir una señal de referencia y una USRP para recibir la señal que genera el transmisor, es conveniente que el reloj de ambos dispositivos esté sincronizado en frecuencia mediante el uso de un reloj externo de referencia de 10 MHz. El uso de este reloj permite que cuando el receptor capture la señal de referencia la frecuencia de demodulación sea la misma que la de modulación en el transmisor y no hayan problemas en la fase.

3.2.2 Sincronización en fase

Para conseguir alineamiento en fase entre cuatro canales en una USRP se deben cumplir tres condiciones:

- Uso de una USRP de la serie X3x0 con dos ranuras para conectar tarjetas secundarias (*daughterboards*).
- Uso de dos tarjetas secundarias iguales que habiliten la recepción por múltiples canales (SBX, UBX, TwinRX).

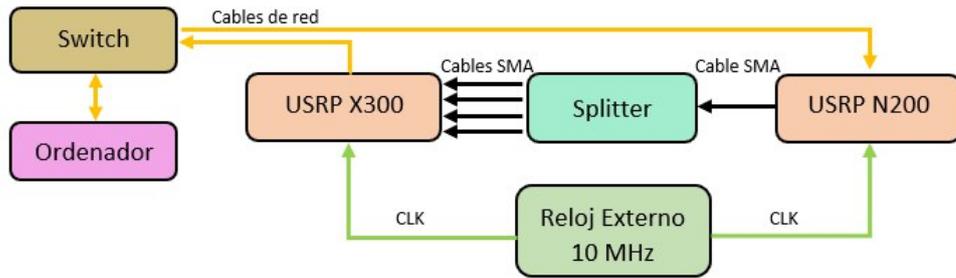


Figure 3.2: Diagrama de bloques de los componentes

- Alineamiento en frecuencia de los osciladores locales (*Local Oscillator*, LO) entre cada cabezal de frecuencia.

Después de ajustar los parámetros de cada cabezal de frecuencia, cada oscilador local puede tener un desplazamiento en fase debido a los divisores dentro de las cadenas de VCO/PLL. Este desplazamiento se mantiene constante hasta que el dispositivo se reajusta cambiando los parámetros.

Algunas consideraciones que hay que tener en cuenta son:

- El desplazamiento en fase es aleatorio entre dos cabezales de frecuencia RF.
- El desplazamiento en fase es diferente para diferentes frecuencias del LO.
- El desplazamiento en fase se mantendrá constante después de cada reajuste.

3.3 Descripción de los componentes

En la Figura 3.2 se presentan los componentes necesarios que conforman la plataforma de medida de la sincronización. A continuación, se presenta la descripción de cada componente hardware.

3.3.1 HF Citrine

El HF Citrine [Wen] es un OCXO (*Oven-Controlled Crystal Oscillators*) de frecuencia fija que puede operar a 10 MHz ofreciendo un ruido de fase bajo y estabilidad de temperatura. Este elemento presenta dos salidas por lo que es idóneo para alimentar dos dispositivos USRP como es el caso. Además, incluye un regulador de voltaje interno de bajo nivel de ruido que proporciona buen rechazo de la línea de alimentación. La Figura 3.3 muestra el componente.



Figure 3.3: HF Citrine 5-15 MHz [Wen]

Table 3.1: Especificaciones de referencia externa

Hardware	Potencia de la señal de referencia
HF Citrine	13 dBm \pm 2 dB
USRP N200	Máx 15 dBm
USRP X300	Máx 15 dBm

Las USRP N200 y X300 utilizarán las entradas de referencia de reloj externas para recibir la señal de reloj de referencia procedente del HF Citrine. Esta señal de referencia sincronizará en frecuencia el reloj de ambos dispositivos.

Algunas de las especificaciones más importantes se muestran a continuación. Para funcionar, necesita una alimentación de 12 VDC (Voltage Direct Current) \pm 5% y genera una señal de potencia de 13 dBm \pm 2 dB a 50 Ω . En la Tabla 3.1 se presenta una comparación entre el nivel de la señal de referencia externa generada por el reloj externo HF Citrine y el nivel de señal máximo que aceptan las USRP.

3.3.2 Switch Netgear GS105

El modelo switch utilizado para establecer y realizar la comunicación entre los dispositivos USRP y el ordenador personal es Netgear GS105 [NET]. Es un modelo no gestionable por el usuario que incluye 5 puertos Gigabit Ethernet. La utilización de un dispositivo de red como este es necesaria para poder controlar varios dispositivos de radio definida por software desde un host con una entrada de red Ethernet. En caso de que solo se tuviera que utilizar una USRP no sería necesario este elemento.

El ancho de banda del modelo Netgear GS105 es de 10 Gbps. En la Figura 3.4 se muestra el modelo mencionado donde se pueden observar los 5 puertos, uno se dedicará para establecer la conexión entre el ordenador y el switch, dos se dedicarán para establecer la conexión entre el

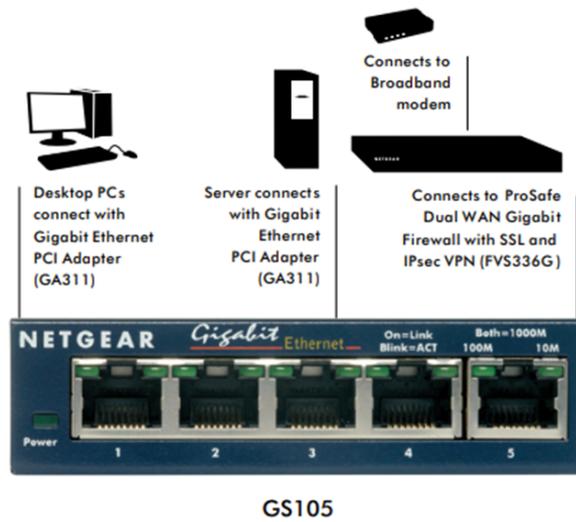


Figure 3.4: Switch Netgear GS105 [NET]

switch y cada USRP, y los dos restantes quedarán disponibles.

Con el objetivo de mantener una cierta calidad de servicio (*Quality of Service*, QoS) y evitar que haya más tráfico del que los cables Ethernet o el propio switch puedan soportar, se debe limitar la tasa de transferencia por debajo de la mínima tasa máxima de transferencia de todos los dispositivos de red. De esta manera, se evita que haya paquetes perdidos debido a la congestión [Pet12].

Los enlaces entre la USRP N200 y el switch, y entre el switch y el ordenador personal, se efectúan a través de cables Ethernet Cat5E que permiten tasas de transferencia de hasta 1 Gbps. El enlace entre la USRP X300 y el switch se efectúa a través de un cable Ethernet Cat6 que permite tasas de transferencia de hasta 10 Gbps.

El enlace que realmente limita la transferencia es el que establece la conexión entre el switch y el ordenador personal. La tasa de transferencia máxima queda limitada a 1 Gbps. Esta tasa de transferencia máxima se deberá repartir entre los dispositivos USRP. En la USRP N200 se utiliza un formato de 64 bits por muestra (Complex Float 32) para transmitir la señal de referencia mientras que en la USRP X300 se utiliza un formato de 32 bits por muestra (Complex Int 16) para recibir la señal.

El cálculo de la tasa de transferencia de datos (*Data Transfer Rate*, DTR) se realiza a través de (3.1).

$$DTR [Mbps] = SampleRate [Mps] * SampleDataFormat [bits/sample] \quad (3.1)$$

En caso de querer establecer la frecuencia máxima de muestreo para cada uno de los cuatro

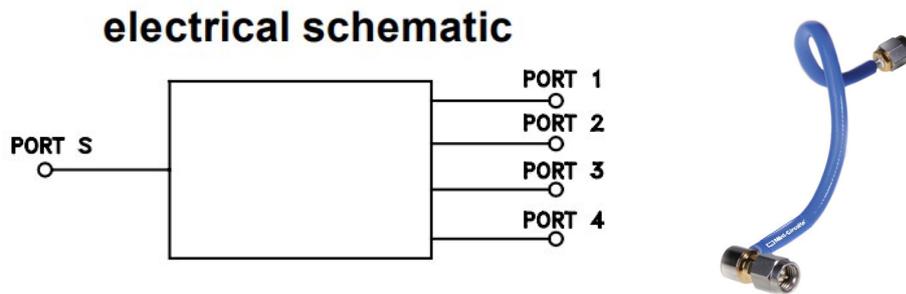


Figure 3.5: Esquema eléctrico del divisor [Minb] y cable SMA [Mina]

canales pertenecientes al receptor USRP X300, se podría considerar la no utilización de la USRP N200 como transmisora, de manera que la frecuencia máxima de muestreo sería de 7.8125 Msps por canal tal y como se indica en (3.2), donde N_{ch} indica el número de canales.

$$SampleRate_{maxch} = \frac{DTR}{SampleDataFormat * N_{ch}} = \frac{1000Mbps}{32bits/sample * 4} = 7.8125Msps \quad (3.2)$$

3.3.3 Splitter ZA4PD-2

El divisor de potencia ZA4PD-2 [Minb] es un dispositivo de radiofrecuencia pasivo con una entrada y cuatro salidas. La entrada que recibe por el puerto 'S' la divide de manera uniforme por los cuatro puertos de salida sin aplicar ganancia.

Algunas de las especificaciones más importantes se muestran a continuación. El rango de frecuencias va desde 1 GHz hasta 2 GHz, y puede variar la fase como máximo en 6 grados. Los puertos tienen una impedancia de entrada y salida de 50 Ω . La potencia máxima de entrada que acepta es de 10 W. Las aplicaciones típicas son para sistemas de comunicaciones y para GPS. A una frecuencia de 1.5 GHz la variación de fase típica es de 1.97 grados.

Los puertos se conectan a los distintos dispositivos a través de cables SMA del modelo 141 SMRSM [Mina]. Estos cables disponen de pérdidas de retorno e inserción excelentes para un rango de frecuencias desde DC a 18 GHz, lo que los hace perfectos para interconectar sistemas. Estos cables pueden soportar altos niveles de potencia de RF.

En la Tabla 3.2 se resumen las especificaciones relevantes del divisor de potencia y de los cables SMA.

Table 3.2: Especificaciones del divisor y de los cables SMA

Hardware	Potencia entrada	Rango	Impedancia	Var. Fase
ZA4PD-2	10 W	1-2 GHz	50 Ω	6°
141 SMRSM	90-546 W	DC-18 GHz	50 Ω	-

3.3.4 USRP X300

En el Capítulo 2 se definió el significado de USRP y su utilidad como interfaz entre el dominio analógico y el digital. Además, también se comentaron las distintas secciones que contiene una SDR.

Toda USRP consta de dos niveles de tarjetas: la tarjeta principal (*motherboard*) y la tarjeta secundaria (*daughterboard*). En esta sección se describirá con más detalle ambos niveles de tarjetas considerando la USRP X300 como dispositivo receptor.

3.3.4.1 Tarjeta principal

Esta tarjeta se encarga de digitalizar la señal IF recibida mediante dos ADC de doble canal (fase y cuadratura) con una resolución de 14 bits y una frecuencia de muestreo máxima de 200 Msps (modelo ADS62P48). Las muestras digitalizadas se transfieren a la FPGA (modelo XC7K325T) que realiza la traslación a banda base a través de un DDC [Ettt].

La traslación a banda base se realiza a través de un coseno y un seno para las componente I y Q respectivamente. Posteriormente, se hace un diezmo de la frecuencia de muestreo por un factor N y el resultado es la tasa de muestreo enviada al ordenador. La combinación de las componentes de salida I y Q se enviarán a través del cable de red Ethernet al ordenador personal [Fä10]. Estas muestras están en formato Complex Int 16 para mejorar el rendimiento del enlace.

La tarjeta principal, además, está controlada por un oscilador local que trabaja a la frecuencia de 96 MHz (modelo OSC-96MHZ-724821-01) y dispone también de dos ranuras para implementar tarjetas secundarias que puede usarse para transmisión o recepción. En este caso, las tarjetas secundarias implementadas son únicamente para recibir señales (véase la Figura 3.6).

3.3.4.2 TwinRX

Las ranuras disponibles en la USRP X300 se ocupan por dos TwinRX [Ettg]. Estas tarjetas secundarias se encargan de la sección de RF y contienen dos cabezales superheterodinos. Cada

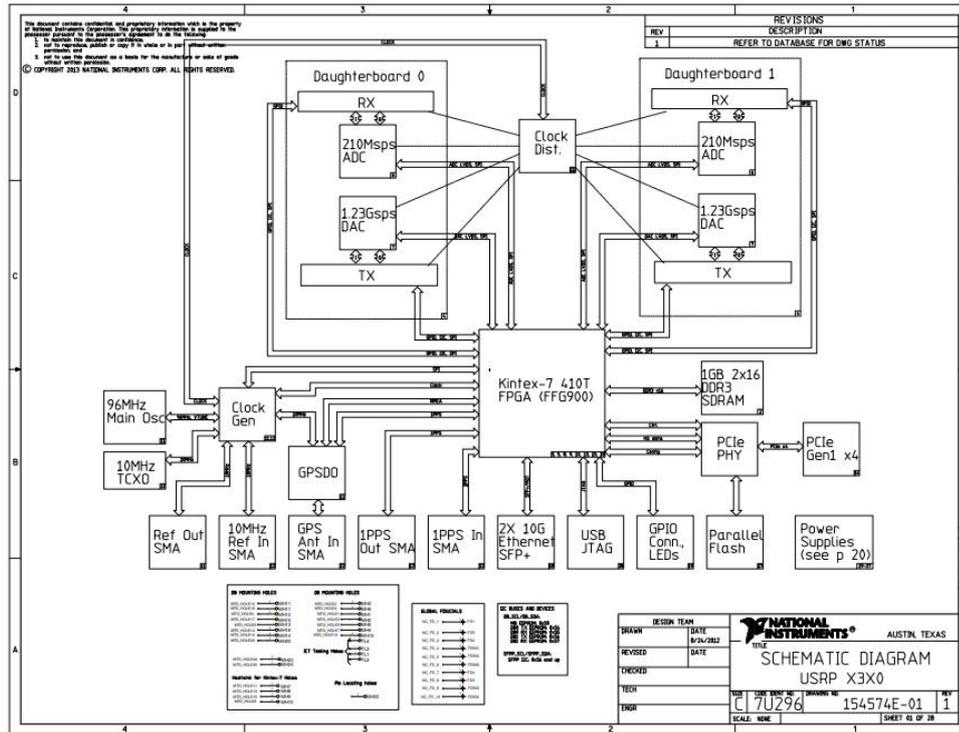


Figure 3.6: Esquema de la USRP X300 [Etta]

canal o cabezal tiene un ancho de banda de 80 MHz (160 MHz entre ambos) y el rango de frecuencia en el que trabajan es desde los 10 MHz hasta los 6 GHz. Cada canal puede introducir una ganancia de hasta 95 dB y soportan una potencia de entrada máxima de 10 dBm.

Como cada canal actúa como un receptor superheterodino, cada uno tiene una sección de RF independiente. Eso quiere decir que cada canal tiene sus preamplificadores, preseleccionadores y dos etapas de mezcla donde los osciladores locales de cada etapa son independientes.

3.3.5 USRP N200

En esta sección se describirá al detalle la tarjeta principal y la tarjeta secundaria de la USRP N200 considerando el dispositivo como transmisor.

3.3.5.1 Tarjeta principal

La tarjeta principal de la USRP N200 está controlada por el procesador FPGA (modelo Xilinx Spartan 3^a-DSP 1800). Esta tarjeta recibe las muestras desde el ordenador personal en formato Complex Float 32 asignando 32 bits a la componente I y 32 bits a la componente Q. Cada muestra es interpolada y trasladada a frecuencias IF mediante el uso del DUC [Fä10].

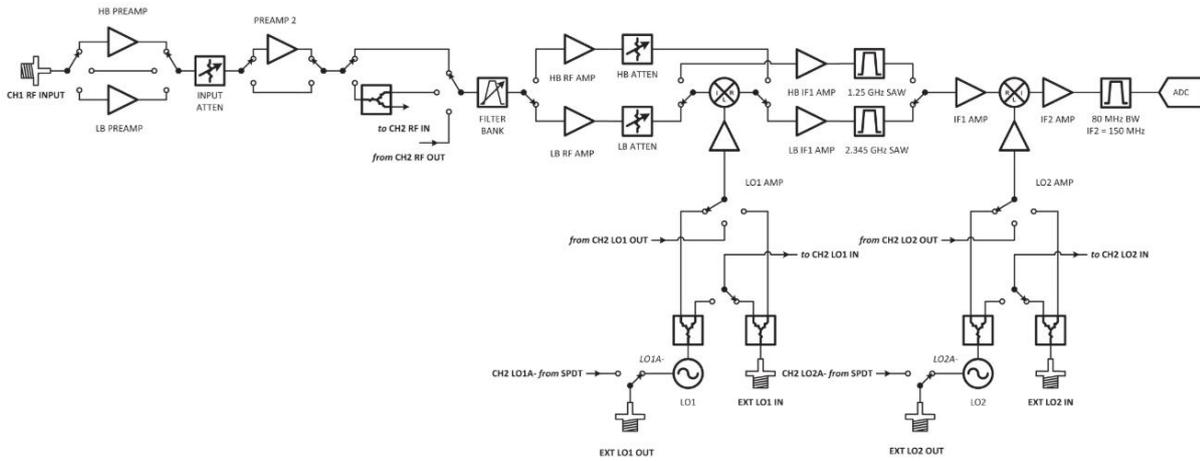


Figure 3.7: Esquema de un cabezal de la TwinRX [Ettg]

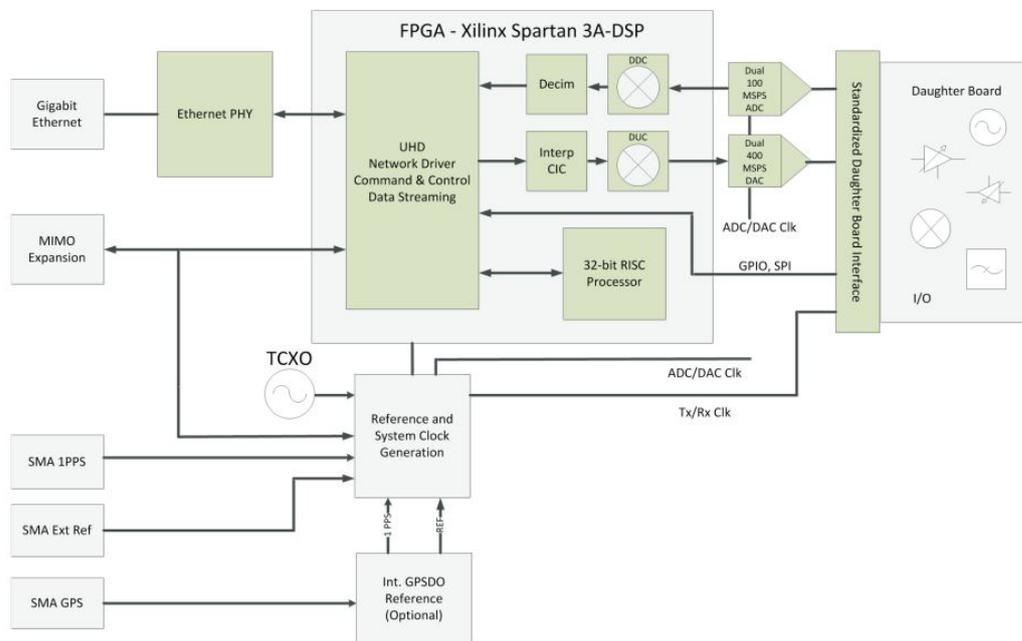


Figure 3.8: Esquema de la USRP N200 [Ettf]

Table 3.3: Especificaciones de la USRP X300 con TwinRX y la USRP N200 con SBX-40

	X300 con TwinRX	N200 con SBX-40
ADC	dual, 200 Msps 14-bit	dual, 100 Msps 14-bit
DAC	dual, 800 Msps 16-bit	dual, 400 Msps 16-bit
Máx. BW	80 MHz por canal	40 MHz
I. Eth	Dual 1G/10G, PCIe Express, ExpressCard	1G
Rango RF	10 MHz - 6 GHz	400 MHz - 4.4 GHz
Máx. Ent. RF	10 dBm	-15 dBm
Máx. Sal. RF	-	20 dBm
Ganancia	0 - 95 dB	0 - 31.5 dB
Impedancia Ent/Sal	50 Ω	50 Ω
Aplicación	RX	TX/RX

La señal se convierte del dominio digital al dominio analógico a través del DAC de doble canal (fase y cuadratura) con una resolución de 16 bits y una frecuencia de muestreo máxima de 400 Msps. Esta tarjeta está controlada por un oscilador local que trabaja a la frecuencia de 100 MHz y dispone de una ranura para implementar una tarjeta secundaria [Ettf]. En este caso, la tarjeta secundaria implementada sirve para transmitir y para recibir.

3.3.5.2 SBX-40

La tarjeta secundaria utilizada es una SBX-40 [Ettb]. Contiene dos cabezales en cuadratura (uno para transmitir y uno para recibir). Cada cabezal tiene un ancho de banda de 40 MHz y el rango de frecuencia en el que trabaja va desde los 400 MHz hasta los 4.4 GHz. Cada canal puede introducir hasta 31.5 dB de ganancia. La potencia de salida depende de la ganancia y de la frecuencia de modulación introducida [Ettc], como máximo puede llegar hasta 20 dBm [Ettb]. Para evitar dañar los puertos de entrada de la TwinRX que tienen una potencia máxima de entrada de 10 dBm, se fija una ganancia en transmisión de 5 dB. Esta ganancia a una frecuencia de modulación de 1.57542 GHz equivale a 4 dBm de potencia. Otra opción sería subir la ganancia y añadir atenuadores antes de que llegara al receptor, o bajar la ganancia y poner amplificadores a la salida del transmisor.

Chapter 4

Modelo de sistema de comunicaciones

En este capítulo se presenta el modelo de sistema de comunicaciones utilizado con el fin de comprender mejor lo que sucede desde la emisión hasta la recepción de la señal. Se comienza explicando el sistema y algunas consideraciones a tener en cuenta. Se muestran los modelos de transmisor, canal y receptor. Se presentan las expresiones matemáticas del modelo, incluyendo el significado de retardo de fase y retardo de grupo. Por último, se introducen las señales periódicas que se utilizarán para tomar las medidas de sincronización y el método mediante el cual se tomarán.

4.1 Introducción al modelo de sistema

El sistema de comunicaciones que se está tratando en este proyecto es un sistema de comunicaciones digital alámbrico y viene caracterizado por los siguientes principios:

- Existe un transmisor (USRP N200) y un receptor (USRP X300) por lo que la comunicación es unidireccional.
- No se utilizan ni codificadores/decodificadores de fuente ni de canal.
- No se utilizan moduladores/demoduladores digitales.
- El transmisor genera señales periódicas como una señal triangular o una senoide. En ambos casos las señales pueden ser complejas o reales.
- Se utilizan las ganancias en transmisión y recepción para distinguir con claridad las señales recibidas.

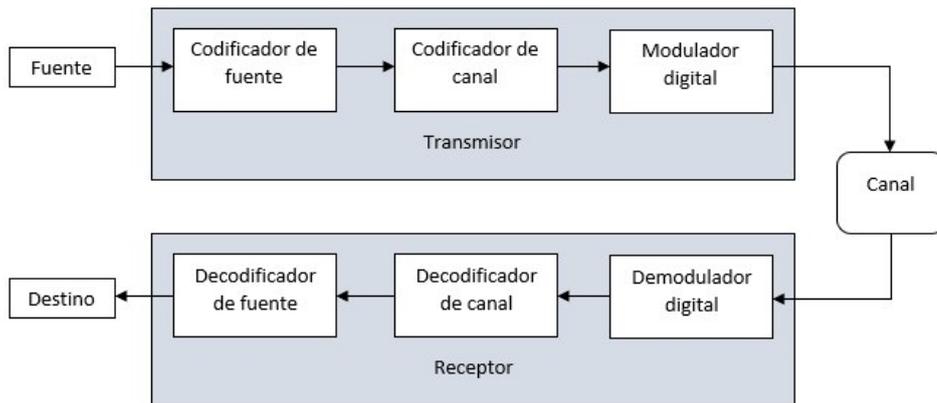


Figure 4.1: Diagrama de bloques de un sistema de comunicaciones digitales [Pro08]

- Se cumple el teorema de muestreo de Nyquist por el que cualquier señal con un ancho de banda (BW) puede ser reconstruida a partir de muestras tomadas con una tasa de Nyquist del doble del ancho de banda.

$$F_s \geq 2BW \quad (4.1)$$

En la Figura 4.1 se puede observar el diagrama de bloques de un sistema de comunicaciones digitales [Pro08]. Como ya se ha mencionado los bloques de codificación, decodificación, modulación digital y demodulación digital no son esenciales para el modelo, aunque se podrían implementar mediante software si fuera necesario debido a que la comunicación entre USRP y ordenador personal se realiza en banda base.

4.2 Modelo transmisor

El transmisor se implementa mediante la USRP N200 con una SBX-40. A continuación, se resume en orden los pasos que se llevan a cabo hasta transmitir la señal por el puerto SMA.

1. El ordenador personal genera la señal periódica y transmite las muestras a la USRP a través de GNU Radio. En el Capítulo 5 se entra en más detalle en cómo se genera el archivo ejecutable Python que incluye la generación de señal utilizando el programa GNU Radio Companion
2. La USRP utiliza la FPGA para interpolar las muestras a la frecuencia de muestreo deseada y para trasladar la señal a frecuencia IF (si fuera necesario).
3. El DAC convierte la señal digital en una señal analógica.

4. La USRP utiliza la tarjeta SBX-40 para trasladar la señal en banda base o IF a frecuencia RF, amplificarla y filtrarla para su posterior transmisión.

4.3 El canal

El canal se implementa mediante los cables SMA y el divisor de señal, y se encarga de introducir un retardo de tiempo y una variación de fase.

4.4 Modelo receptor

El receptor se implementa mediante la USRP X300 con dos TwinRX. A continuación, se resume en orden los pasos que se llevan a cabo hasta obtener los datos con las muestras de la señal en el ordenador personal.

1. La USRP utiliza las tarjetas TwinRX para captar la señal que se introduce por los puertos SMA, amplificarla, filtrarla y trasladarla a frecuencia IF o banda base en dos etapas de mezcla.
2. El ADC digitaliza las muestras con una tasa de muestreo específica.
3. La USRP utiliza la FPGA para trasladar la señal a banda base (si fuera necesario) y para adaptar las muestras a la frecuencia de muestreo del ordenador personal.
4. La USRP transmite las muestras al ordenador personal a través de GNU Radio para aplicar el procesado a la señal digital.

4.5 Caracterización matemática del modelo de sistema

Los pasos realizados en la sección 4.2, 4.3 y 4.4 se pueden caracterizar a través de expresiones matemáticas.

1. La generación de las muestras, la transmisión a la USRP, la interpolación y traslado a frecuencia IF resultan en la señal digital. La señal digital se expresa como $s[n]$.
2. Conversión del dominio digital al analógico de la señal mediante el conversor DAC. La señal en tiempo tendrá la forma:

$$s(t) = s[n] \big|_{n=tF_{s1}} \quad (4.2)$$

Donde F_{s1} es la frecuencia de muestreo del conversor DAC.

3. Traslación de frecuencia IF a frecuencia RF.

$$y(t) = s(t)e^{j2\pi f_{cMOD}t} \quad (4.3)$$

Donde f_{cMOD} es la frecuencia central de modulación.

4. Paso por el canal y captura de la señal.

$$y_x(t) = y(t - \zeta_x) = \alpha_x s(t - \zeta_x) e^{j(2\pi f_{cMOD}(t - \zeta_x) + \varphi_{yx})} \quad (4.4)$$

Donde $y_x(t)$ representa la señal temporal que llega a cada cabezal (el subíndice x indica el cabezal), que es la señal que se transmite desde el transmisor, pero con un retardo temporal de ζ_x consecuencia de la propagación a través del canal. φ_{yx} es la variación de fase introducida por el divisor de señal (normalmente, el índice de salida del divisor y coincidirá con el índice del cabezal receptor x) y α_x una atenuación. En este caso se considera un canal no ideal con distorsión lineal. El canal introduce un retardo, una distorsión de amplitud y una distorsión en fase [Hay99].

5. Traslación de frecuencia RF a IF.

$$x_x(t) = y_x(t) e^{-j(2\pi f_{cDEMx}t + \varphi_x)} = \alpha_x s(t - \zeta_x) e^{j(2\pi((f_{cMOD} - f_{cDEMx})t - \zeta_x + \varphi_{yx} - \varphi_x))} \quad (4.5)$$

Donde f_{cDEMx} es la frecuencia de demodulación en cada etapa RF, por lo que dependerá del canal en el que se realice la captura. En (4.6) se simplifica la expresión anterior para el caso que f_{cMOD} sea igual a f_{cDEMx} en todos sus canales.

$$x_x(t) = \alpha_x s(t - \zeta_x) e^{-j(2\pi f_c \zeta_x - \varphi_{yx} + \varphi_x)} \quad (4.6)$$

φ_x representa la fase introducida por el cabezal receptor al ser demodulada la señal.

6. Digitalización, traslado a banda base, diezmado y transmisión al ordenador personal.

$$x_x[n] = x_x(t) \Big|_{t=\frac{n}{F_{s2}}} \quad (4.7)$$

Donde F_{s2} es la frecuencia de muestreo a la que se digitaliza la señal en el ADC. Puede coincidir o no con F_{s1} . El archivo recibido en el ordenador personal contiene los valores para las muestras generadas a través de (4.7).

4.6 Retardo de fase y retardo de grupo

Existen dos tipos de retardo y cada uno dependerá de si varía la señal portadora o de si varía la señal deseada [Hay99].

- El retardo de fase es el desfase o retraso que se aplica a la portadora de la señal, este retardo puede ser introducido por una variación en fase o un retardo en tiempo debido al canal.
- El retardo de grupo es el retraso que se aplica sobre la señal desde que se transmite hasta que se recibe en la entrada del receptor.

4.7 Señales periódicas

En esta sección se introducen las señales con las que se probará el sistema. Estas señales son: exponencial compleja, seno real, señal triangular real y señal triangular compleja.

4.7.1 Exponencial compleja

La exponencial compleja constituye una señal ortogonal y se puede expresar mediante la relación de Euler.

$$s(t) = Ae^{2\pi ft} = A(\cos(2\pi ft) + j \sin(2\pi ft)) \quad (4.8)$$

4.7.2 Seno real

En este proyecto se utiliza un coseno como seno real. Se muestra la forma de la señal en (4.9).

$$s(t) = A \cos(2\pi ft) \quad (4.9)$$

Otra opción de representar esta forma de onda es utilizando la identidad de Euler como se ve en (4.10).

$$s(t) = \frac{A}{2} \left(e^{j2\pi ft} + e^{-j2\pi ft} \right) \quad (4.10)$$

4.7.3 Señal triangular real

En este proyecto se nombra como señal triangular real a la señal compleja transmitida cuya parte real es una señal triangular y que tiene parte imaginaria nula. La señal se representará como un sumatorio de pulsos triangulares en (4.12) a partir de un pulso triangular definido en (4.11).

$$p_T(t) = \begin{cases} 1 - \frac{|t|}{T/2}, & \text{si } |t| \leq T/2 \\ 0, & \text{si } |t| > T/2 \end{cases} \quad (4.11)$$

$$s(t) = A \sum_{k=-\infty}^{\infty} p_T(t - kT) \quad (4.12)$$

4.7.4 Señal triangular compleja

Para representar la señal triangular compleja la señal tiene que seguir el principio de ortogonalidad. En (4.13) se muestra la señal a partir del pulso triangular definido en (4.11).

$$s(t) = A \sum_{k=-\infty}^{\infty} p_T(t - kT) + jp_T\left(t - kT - \frac{T}{2}\right) \quad (4.13)$$

4.8 Correlación cruzada y coeficiente de correlación

Simplificando, cualquier señal que atraviese un canal no ideal tendrá la forma de la señal que se ha transmitido $s_{tx}(t)$, retardada en tiempo y con una fase ϑ . Entre cabezales de RF el retardo de grupo y la fase variarán por lo que se recibirán señales con un retardo en tiempo y una fase diferentes. Se añade el subíndice c para identificar de que cabezal de RF se trata.

$$s_{rxc}(t) = s_{tx}(t - \zeta_c)e^{j\vartheta_c} \quad (4.14)$$

Mediante el coeficiente de correlación y la correlación cruzada se pueden medir la diferencia entre el retardo en tiempo entre cabezales y el desfase. En (4.15) se muestra la fórmula de la correlación cruzada [Car10].

$$R_{cxcy}(\zeta) = \int_{-\infty}^{\infty} s_{cx}(t + \zeta)s_{cy}(t)^* dt = \int_{-\infty}^{\infty} s_{tx}(t - \zeta_{cx} + \zeta)e^{j\vartheta_{cx}} s_{tx}(t - \zeta_{cy})e^{-j\vartheta_{cy}} dt \quad (4.15)$$

Donde ζ es el retardo temporal entre canales y ζ_{cx} y ζ_{cy} son los retardos de grupo de cada señal. Los subíndices cx y cy definen los cabezales RF. Los retardos de grupo y las fases de cada cabezal se pueden agrupar en una sola señal, de esta manera se simplifica (4.15).

$$R_{cxcy}(\zeta) = \int_{-\infty}^{\infty} s_{tx}(t - \zeta_{cx} + \zeta_{cy} + \zeta) s_{tx}(t) e^{j(\vartheta_{cx} - \vartheta_{cy})} dt \quad (4.16)$$

El coeficiente de correlación c_{cxcy} se calcula tal que:

$$c_{cxcy} = \left| \frac{R_{cxcy}(\zeta)}{\sqrt{R_{cx}(0)R_{cy}(0)}} \right| \leq 1 \quad (4.17)$$

Donde $R_{cx}(0)$ y $R_{cy}(0)$ son las autocorrelaciones¹ cuando el retardo entre las señales es nulo y representan la energía de la señal en cada cabezal. Como son las mismas señales tenemos que:

$$R_{cx}(0) = R_{cy}(0) = \int_{-\infty}^{\infty} s_{tx}(t) s_{tx}(t) dt = R_{tx}(0) = E_{tx} \quad (4.18)$$

El valor del retardo ζ puede ser igual o diferente a cero.

- El coeficiente de correlación será máximo siempre que la diferencia en los retardos de grupo entre cabezales se vea compensado por el retardo aplicado en la correlación cruzada. Viene dado por (4.19).

$$\zeta_{cx} - \zeta_{cy} + \zeta = 0 \quad (4.19)$$

A partir de (4.16) y considerando lo dicho en (4.19) se obtiene que:

$$R_{cxcy}(0) = \int_{-\infty}^{\infty} s_{tx}(t) s_{tx}(t) e^{j(\vartheta_{cx} - j\vartheta_{cy})} dt = \int_{-\infty}^{\infty} s_{tx}^2(t) e^{j(\vartheta_{cx} - j\vartheta_{cy})} dt \quad (4.20)$$

Si se realiza la integral el resultado es:

$$R_{cxcy}(0) = R_{tx}(0) e^{j(\vartheta_{cx} - \vartheta_{cy})} = E_{tx} e^{j(\vartheta_{cx} - \vartheta_{cy})} \quad (4.21)$$

Aplicando (4.18) y (4.21) a (4.17) se obtiene que:

$$c_{cxcy} = \left| \frac{E_{tx} e^{j(\vartheta_{cx} - \vartheta_{cy})}}{\sqrt{E_{tx} E_{tx}}} \right| = 1 \quad (4.22)$$

El desfase entre señales se obtiene a partir de (4.21).

$$\vartheta_{cxcy} = \vartheta_{cx} - \vartheta_{cy} \quad (4.23)$$

¹Autocorrelación: Correlación cruzada de una señal consigo misma.

- En caso de que el retardo no compensara la diferencia entre los retardos de grupo de las señales de cada cabezal, el coeficiente de correlación c_{cxcy} no sería máximo, lo que supondría tener que aplicar una corrección en tiempo a la señal retardada con el valor del lag ζ .

Chapter 5

Implementación de diseño en GNU Radio

GNU Radio, como se explicó anteriormente, es un software de desarrollo gratuito que contiene herramientas de procesamiento de señal a través de bloques que permiten implementar Radios Definidas por Software (SDR).

La manera de implementar bloques básicos de diseño se realiza con el programa GNU Radio Companion (GRC), que ofrece al usuario la posibilidad de configurar y comunicar su ordenador personal con los dispositivos USRP mediante los UHD.

En este capítulo se implementan los esquemas básicos de diseño realizados mediante GNU Radio Companion (GRC) para la transmisión y recepción de señales. Para abrir el programa es necesario abrir un terminal y ejecutar el comando: `sudo gnuradio-companion`.

5.1 Diseño de un transmisor de señales complejas

En primer lugar, se configura el bloque *Options* (véase la Figura 5.1). Los parámetros importantes que se tienen que modificar son los siguientes:

- **ID:** Identifica el archivo Python de ejecución que se crea al generar el gráfico de flujo.
- **Generate Options:** La opción *No GUI* permite no mostrar nada en el editor gráfico. Una alternativa interesante es la opción *WX GUI* que permite analizar la salida del generador de señal dentro del propio editor gráfico.
- **Run Options:** Para la opción *No GUI* es recomendable seleccionar *Prompt for Exit* ya que interesa que, cuando se ejecute el archivo Python mediante el terminal de comandos,

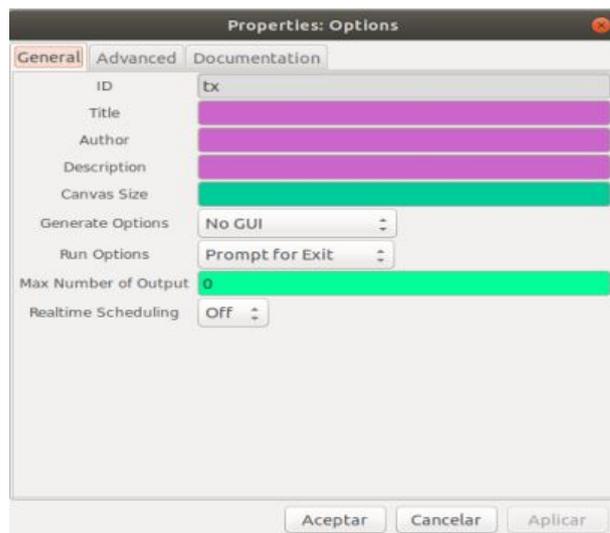
Figure 5.1: Propiedades del bloque *Options*

Table 5.1: Tipos de datos de salida

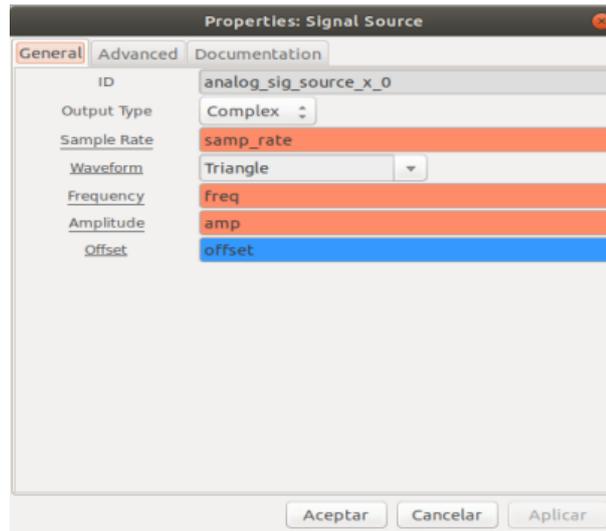
Complex	Complex float32
Float	Float32
Int	Integer32
Short	Integer16

la fuente esté transmitiendo en todo momento. En el caso de haber seleccionado la opción *WX GUI* es usual seleccionar *Autostart* ya que sólo se utiliza para analizar la salida del generador de señal.

- **Realtime Scheduling:** Seleccionar *OFF*.

Para generar una señal de referencia periódica compleja se utiliza el bloque *Signal Source*. Esta fuente se puede encontrar en el apartado *Waveform Generators* del menú que aparece a la derecha. Los parámetros a modificar aparecen en la Figura 5.2 y se comentan a continuación.

- **Output Type:** Especifica el tipo de salida (véase la Tabla 5.1). Para este caso interesa que la salida sea del tipo *Complex*, de esta manera a la parte real de la señal se le asignarán 32 bits *Float* y a la parte imaginaria otros 32 bits *Float*.
- **Sample Rate:** Especifica la frecuencia de muestreo de la señal generada.

Figure 5.2: Propiedades del bloque *Signal Source*

- **Waveform:** Permite escoger entre varios tipos de señales como: Constante, Seno, Coseno, Cuadrada, Triangular y Diente de sierra.
- **Frequency:** Especifica la frecuencia de trabajo de la señal.
- **Amplitude:** Especifica la amplitud pico en el caso del Seno y el Coseno o la amplitud pico a pico en los casos Cuadrada, Triangular y Diente de sierra. La amplitud si se escoge la forma de onda Constante debe ser 0 y se usa el parámetro *Offset*.
- **Offset:** El valor definido se añade a la forma de onda generada.

Para establecer comunicación con la USRP y transmitir la señal generada a través de la misma, es necesario un bloque que realice la comunicación UHD-GNU Radio. El bloque que se necesita se conoce como *UHD: USRP Sink*. Los parámetros a modificar aparecen en la Figura 5.3 y se comentan a continuación.

- **Input Type:** Controla el tipo de datos a la entrada del flujo. Dos opciones: Complex float32 o Complex int16. Solo la primera opción es compatible con el tipo de datos a la salida del generador.
- **Device Addr:** Identifica la USRP utilizada para transmitir a través de su dirección IP. Se elige la IP de la USRP N200: *192.168.10.3*.
- **Sync:** Permite sincronizar varios cabezales de frecuencia RF o varios dispositivos USRP. Seleccionar la opción *Don't sync* en caso de estar utilizando un cabezal de frecuencia de una USRP (es el caso de este proyecto).

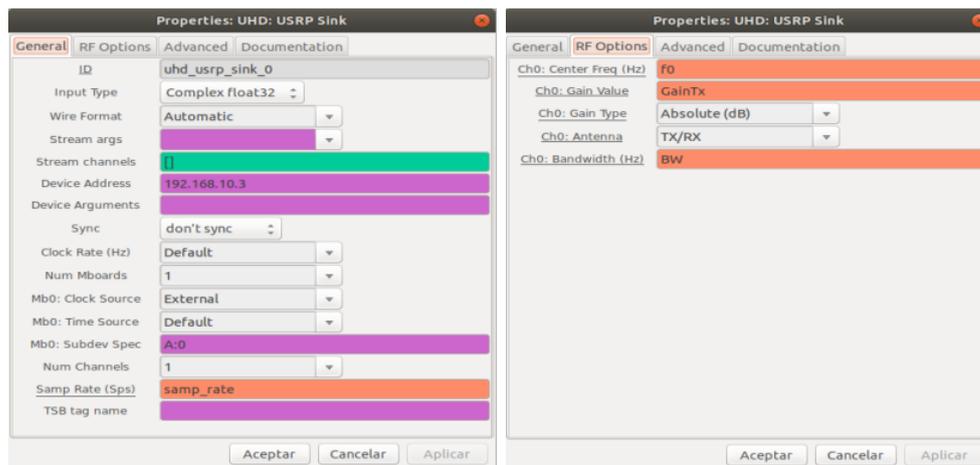


Figure 5.3: Propiedades del bloque *UHD: USRP Sink*

- **Num Mboards:** Permite seleccionar la cantidad de dispositivos USRP a utilizar. Seleccionar *1* en caso de que solo se utilice una USRP.
- **Mb0: Clock Source:** Fuente donde la *motherboard* sincroniza sus referencias de reloj. La opción *External* se refiere a la entrada de 10 MHz de un reloj externo a la USRP.
- **Mb0: Time Source:** Fuente donde la *motherboard* sincroniza sus referencias de tiempo. Se escoge por defecto.
- **Mb0: Subdev Spec:** Se define si hay más de una *daughterboard* por USRP.
- **Samp Rate (Sps):** Especifica el número de muestras por segundo, que es igual al ancho de banda en Hz que se desea observar. Debe cumplir el teorema de Nyquist presentado en el Capítulo 4. Debe ser divisor de 100 Msps.
- **Ch0: Center Freq (Hz):** Frecuencia central a la que se modula la señal.
- **Ch0: Gain Value:** Valor de ganancia deseado entre 0 y 31.5 dB de la SBX-40.
- **Ch0: Gain Type:** *Absolute* (en dB).
- **Ch0: Antenna:** Especifica el canal *TX/RX* para transmitir.
- **Ch0: Bandwidth (Hz):** Ancho de banda utilizado por el filtro anti-aliasing de la USRP. Puede dejarse por defecto poniendo un 0.

Por último, modificar los parámetros anteriormente mencionados a través de variables y unir las salidas de los bloques con las entradas de los demás para solucionar errores tal y como se muestra en la Figura 5.4, y poder generar el archivo de ejecución Python.

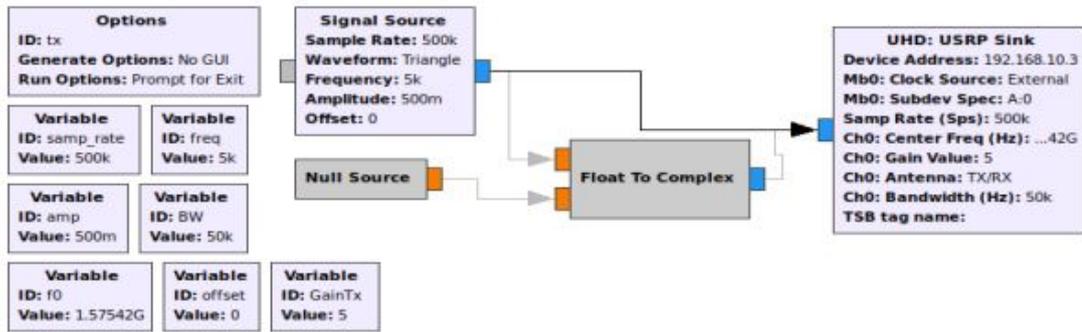
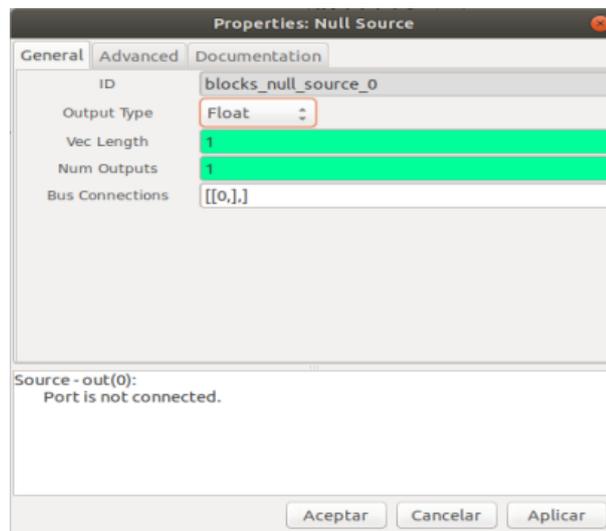


Figure 5.4: Transmisor USRP de señales complejas

Figure 5.5: Propiedades del bloque *Null Source*

5.2 Diseño de un transmisor de señales reales

Para generar un transmisor de señales reales se puede partir del diseño realizado en la sección anterior añadiendo dos bloques más y modificando uno de los ya presentes.

El bloque a modificar sería *Signal Source*, donde el tipo de salida debe ser *Float* para que se le asignen 32 bits a la parte real de la señal (véase la Tabla 5.1). El primer bloque que habría que añadir sería *Null Source*, una fuente que para cada muestra tiene como salida un cero, el formato de datos de salida es *Float* al igual que el bloque *Signal Source*. Como se ha comentado el principal parámetro a modificar es:

- **Output Type:** Especificar el tipo de salida a *Float* para que el valor sea real.

Por último, habría que añadir el bloque *Float To Complex*. Este bloque admite dos entradas, una real de 32 bits, donde se conectará la fuente de señal, y una imaginaria, donde se conectará

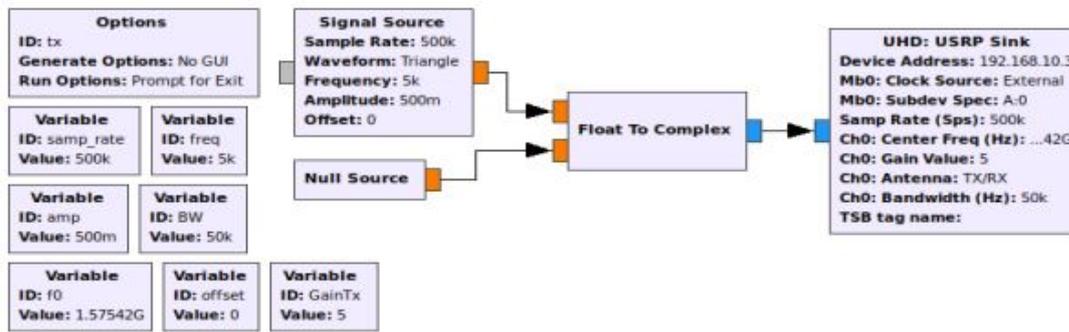


Figure 5.6: Transmisor USRP de señales reales

la fuente de nulos. La salida de este bloque es compleja con formato Complex float 32, con parte real la señal generada y parte imaginaria nula. En la Figure 5.6 se muestra el transmisor USRP final.

5.3 Diseño de un receptor de cuatro canales en una USRP

En esta sección se presenta el diseño final de un receptor de cuatro canales sincronizado con un reloj externo de 10 MHz y utilizando los osciladores locales LO1 y LO2 del primer cabezal de frecuencia de la primera *daughterboards* como referencia para los otros cabezales.

En primer lugar, se configura el bloque *Options*. Los parámetros que se tienen que modificar son:

- **ID:** Especificar un nombre para el archivo de ejecución Python.
- **Generate Options:** Seleccionar la opción *No GUI* para no mostrar nada en el editor gráfico.
- **Run Options:** Seleccionar la opción *Run to Completion* para que cuando el receptor haya captado la cantidad de muestras deseadas finalice.
- **Realtime Scheduling:** Seleccionar *OFF*.

Seguidamente, se añade el bloque que entabla comunicación con la USRP. Este bloque es similar al utilizado en la sección anterior para realizar la comunicación y transmitir, solo que en este caso se quiere recibir. El nombre del bloque es *UHD: USRP Source* y los parámetros son exactamente los mismos que para *UHD USRP Sink* (véase la Figura 5.7).

- **Input Type:** Seleccionar entre Complex float 32 o Complex int16 dependiendo de la cantidad de bits que se quieran almacenar. En este caso, se selecciona Complex int16.

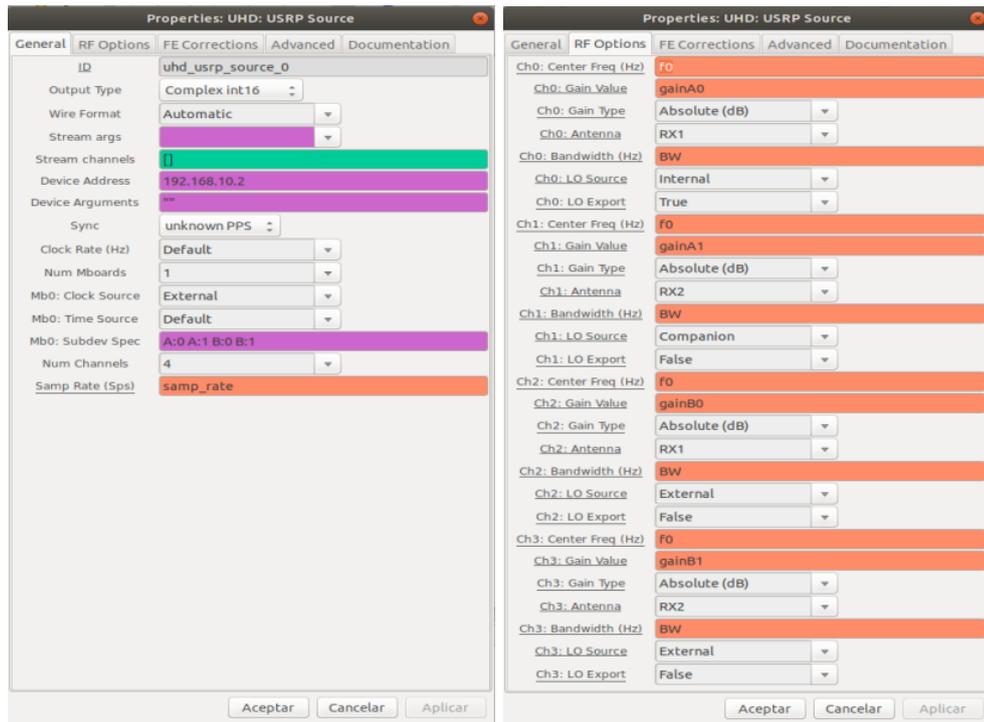


Figure 5.7: Propiedades del bloque *UHD: USRP Source*

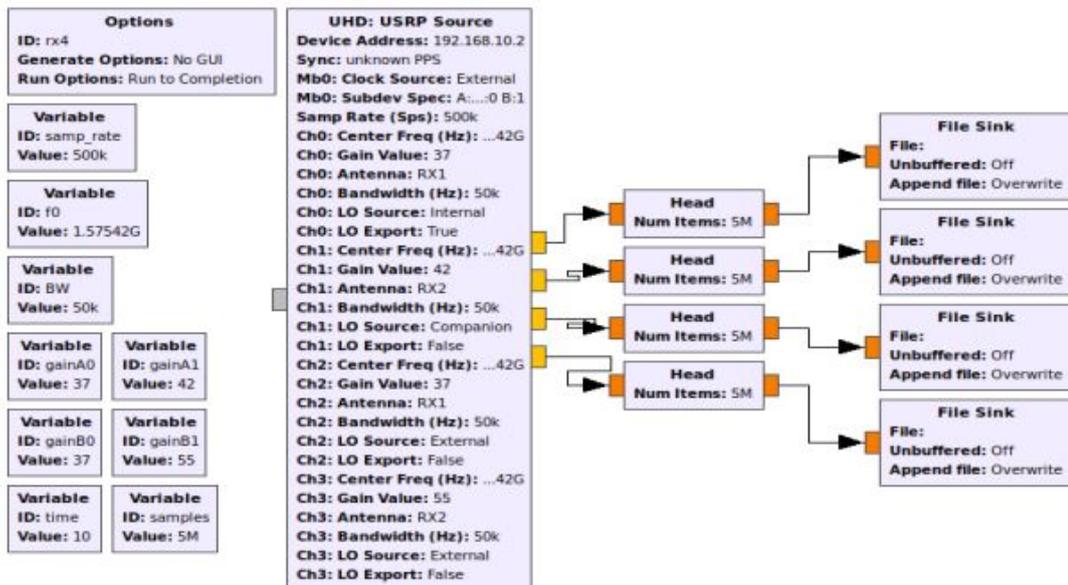


Figure 5.8: Receptor USRP de cuatro canales

- **Device Addr:** Especificar la dirección IP de la USRP utilizada para recibir. Se elige la IP de la USRP X300: *192.168.10.2*.
- **Sync:** Seleccionar la opción *Unknown PPS* para sincronizar los cabezales de frecuencia de la USRP entre ellos.
- **Num Mboards:** Seleccionar el número de dispositivos USRP con los que recibir, en este caso seleccionar *1*.
- **Mb0: Clock Source:** Seleccionar la opción *External* para que la *motherboard* sincronice sus referencias de reloj con el reloj externo de 10 MHz.
- **Mb0: Time Source:** Seleccionar la opción *Default* ya que no hay conectada una fuente para referencias de tiempo.
- **Mb0: Subdev Spec:** Especificar las *daughterboards* utilizadas y sus canales mediante el formato “A:0”, donde A representa la *daughterboard* empleada y 0 el canal utilizado. En este caso, habría que rellenar este campo con “A:0 A:1 B:0 B:1”.
- **Num Channels:** Seleccionar la opción *4* ya que se van a utilizar cuatro canales.
- **Show LO Controls:** Seleccionar la opción *Yes* para mostrar las opciones de uso de los osciladores internos de cada cabezal de frecuencia. Este apartado se encuentra en la pestaña *Advanced*.
- **Samp Rate (Sps):** Especificar el número de muestras por segundo. Se refiere a la tasa de muestreo que el ADC utilizará para digitalizar la señal. Debe ser divisor de 100 Msps.
- **ChX: Center Freq (Hz):** Especificar la frecuencia central de recepción deseada para cada canal.
- **ChX: Gain Value:** Especificar la ganancia deseada entre 0 y 95 dB en cada cabezal.
- **ChX: Gain Type:** *Absolute* (en dB).
- **ChX: Antenna:** Especificar la antena utilizada para cada canal, en este caso seleccionar *RX1* (Ch0), *RX2* (Ch1), *RX1* (Ch2) y *RX2* (Ch3).
- **ChX: Bandwidth (Hz):** Especificar el ancho de banda deseado para cada canal.
- **ChX: LO Source:** Seleccionar qué osciladores locales dentro de cada canal hacen de referencia. En este caso, este campo se rellena con “*Internal, Companion, External, External*”. El canal que contenga el valor *Internal* será el que haga de referencia, mientras que el otro canal de la misma *daughterboard* tendrá que tener el valor *Companion*. Los canales de la otra *daughterboard* tendrán el valor *External* ya que deben recibir la referencia desde la otra tarjeta.

- **ChX: LO Export:** Seleccionar qué osciladores locales dentro de cada canal exportan su señal o no. En este caso, solo el primer canal exportará su señal de referencia, por lo que este campo se rellena con “*True, False, False, False*”.

Posteriormente, se añaden bloques *Head* que limitan el número de muestras que recibe el receptor por cada canal. Para que este bloque concuerde con la salida del bloque *UHD: USRP Source* es necesario cambiar el tipo de datos a *Float*. Finalmente, se añaden los bloques *File Sink* que permiten el almacenamiento de los datos digitalizados de la USRP en un archivo binario, el lugar de almacenamiento y el nombre del archivo dependen del usuario. Los archivos almacenados contendrán la parte real y la parte imaginaria de las muestras intercaladas ($R[1]$, $I[1]$, $R[2]$, $I[2]$...). Utilizar variables simplifica el esquema y su posterior modificación.

Chapter 6

Pruebas y resultados

En este capítulo se muestran las pruebas realizadas junto con los resultados obtenidos para cada una de ellas. Las pruebas se realizan a todo el sistema presentado en el Capítulo 3 y en cada sección se menciona qué esquemas de los presentados en el Capítulo 5 y qué parámetros se utilizan.

Cabe destacar que el esquema de recepción no cambiará en cuanto al número de canales utilizado. La ganancia en transmisión se fijará a 5 dB para evitar daños en el receptor y la frecuencia central se mantendrá constante a 1.57542 GHz. Las antenas siempre tendrán la misma asignación: A0:RX1, A1:RX2, B0:RX1 y B1:RX2 (véase las Figuras 6.1 y 6.2).

6.1 P1: Validación de la señal recibida

6.1.1 Descripción

En esta primera prueba se han definido unos parámetros iniciales en los esquemas presentados en las Figuras 6.1 y 6.2. Los parámetros se muestran en la Tabla 6.1, donde la ganancia en transmisión se ajusta de manera fija a 5 dB para evitar el daño de los puertos SMA de cada cabezal receptor (soportan un máximo de 10 dBm de potencia de entrada). El tiempo durante el que se ha realizado esta prueba ha sido de 3 segundos, aunque solo se van a utilizar las primeras muestras para la comprobación.

El objetivo de esta prueba es ver si la señal que se recibe tiene la misma forma de onda que la señal triangular que se transmite. Además, permite ajustar las ganancias de cada canal en el receptor.

Algo a tener en cuenta es que la señal triangular real se genera con una media de la mitad de la amplitud, lo que significa que es una señal positiva.

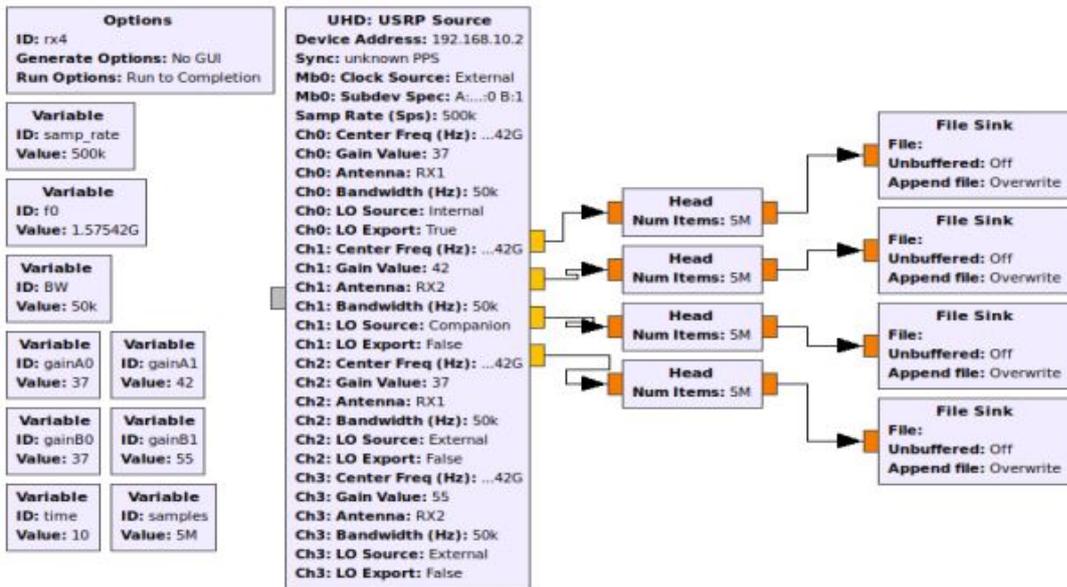


Figure 6.1: Receptor USRP de cuatro canales

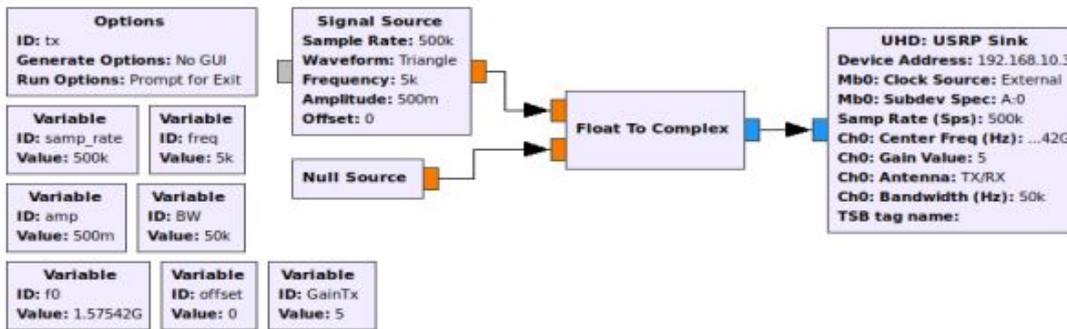


Figure 6.2: Transmisor USRP de señales reales

Table 6.1: Parámetros iniciales

Emisor									
Onda	S.Rate	Freq	Amp	Offset	Clk	Sps	Gain	BW	Sync
Tri. Real	500k	5k	0.5	0	Ext	500k	5	50k	Not Sync

Receptor						
Clk	Sps	Gain	LO Source	LO Exp.	BW	Sync
Ext	500k	5	Internal	False	50k	Unknow PPS

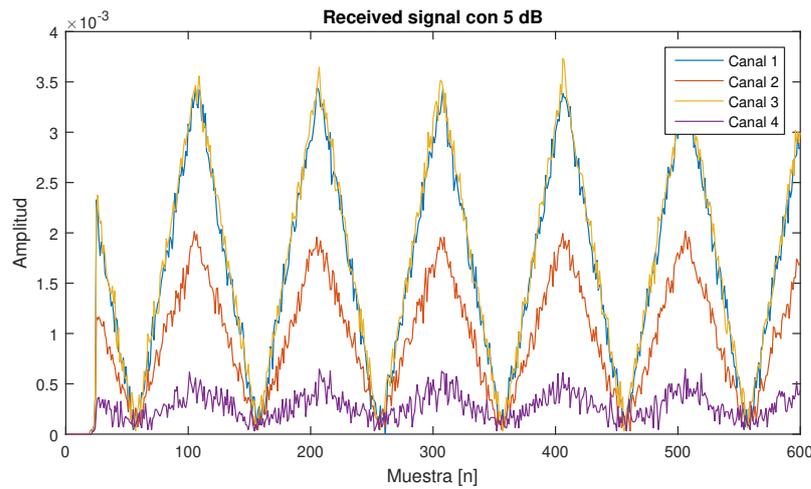


Figure 6.3: Primeras muestras de la señal recibida con 5 dB de ganancia según el canal

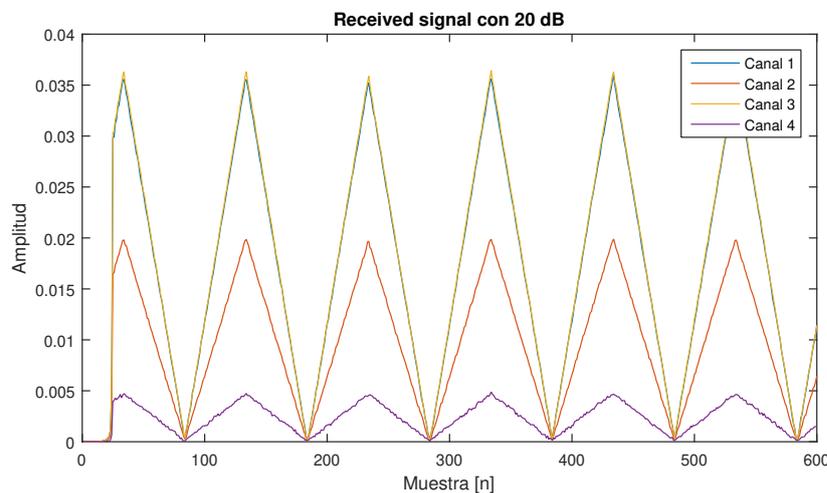


Figure 6.4: Primeras muestras de la señal recibida con 20 dB de ganancia según el canal

6.1.2 Resultados

Al asignar una ganancia en recepción y en transmisión de 5 dB y haciendo el módulo con el fin de visualizar la señal real recibida, se consigue distinguir la señal triangular, pero el nivel de potencia de la señal es demasiado bajo por lo que se ve muy afectada por el ruido del receptor. Además, se puede observar como los 5 dB de ganancia que se habían puesto en cada canal de recepción no proporcionan el mismo nivel de señal en cada uno. (véase la Figura 6.3).

Para evitar primero el ruido que afecta a la señal se procede a aumentar de 5 dB a 20 dB la ganancia en todos los canales receptores. El resultado se muestra en la Figura 6.4, en ella se observa como las señales al haber sido amplificadas no se ven tan afectadas por el ruido y tienen una forma de onda más similar a la de una onda triangular. Aun así, la señal recibida por el

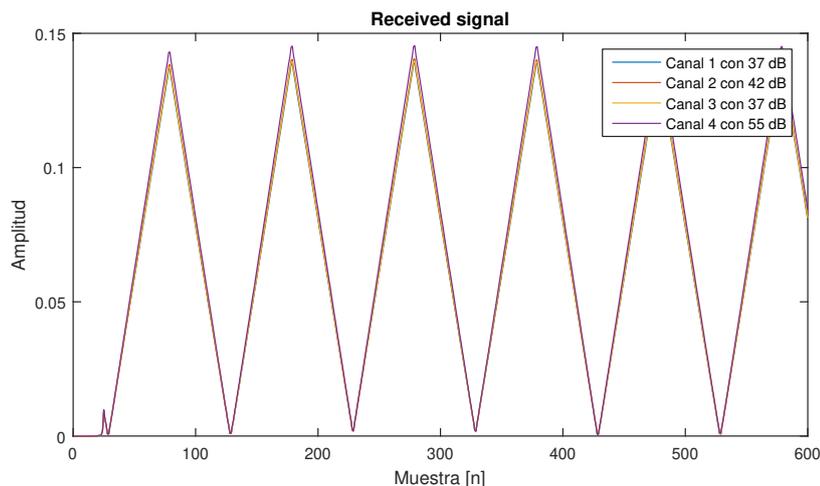


Figure 6.5: Primeras muestras de la señal recibida con ganancias 37 dB, 42 dB, 37 dB y 55 dB según el canal

canal 4 todavía no tiene un nivel de potencia lo suficientemente alto para verse bien.

Con el fin de aumentar el nivel de la señal recibida por los canales 2 y 4 para que se asemejen al nivel de la señal recibida por los canales 1 y 3, se procede a aumentar la ganancia de todos los canales, pero teniendo en cuenta que el aumento en los canales 2 y 4 será mayor (sobre todo en el canal 4 ya que la diferencia de nivel es mayor).

Las ganancias que se prueban son 37 dB, 42 dB, 37 dB y 55 dB para los canales 1, 2, 3 y 4 respectivamente. El resultado se muestra en la Figura 6.5, en ella se observa como los canales tienen un nivel de señal similar y suficientemente alto como para distinguirse la onda triangular sin ruido.

Ya solo por curiosidad, se aumenta la ganancia en todos los canales en 10 dB. El resultado se muestra en la Figura 6.6 y se observa como ahora parece que el nivel de señal ha aumentado igual en todos los canales.

Teniendo en cuenta que la señal que se envía desde el receptor tiene una cierta potencia, atraviesa los cables SMA que tienen la misma longitud y las mismas pérdidas, atraviesa el divisor donde la división de potencia es uniforme para cada salida según el *datasheet*, y llega a las distintas entradas del receptor, se supone que al indicar la misma ganancia en cada canal debería verse la misma señal con el mismo nivel prácticamente en cada canal, pero esto no es así y se necesitan ganancias diferentes.

También es interesante mencionar que las primeras muestras recibidas no contienen la forma de la señal porque el receptor ha tardado un tiempo en ponerse a funcionar.

Por último, como la longitud de los cables SMA, la circuitería interna del divisor y la cir-

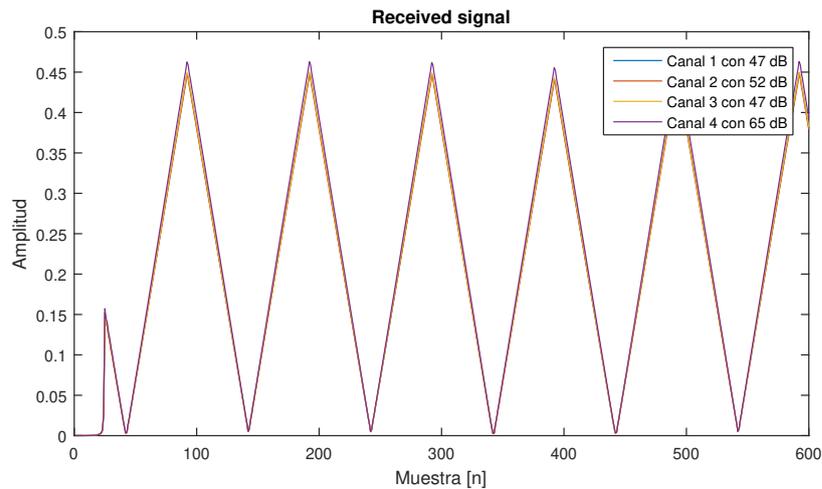


Figure 6.6: Primeras muestras de la señal recibida con ganancias 47 dB, 52 dB, 47 dB y 65 dB según el canal

cuitaría interna del receptor por cada cabezal son iguales y, además, los relojes de cada cabezal funcionan a la misma frecuencia debido a que cogen como referencia el reloj de la tarjeta principal de la X300 o el reloj externo, el retardo de grupo para las señales introducidas por cada cabezal es el mismo y, visualmente, parecen estar alineadas en tiempo.

Si se cambiase un cable de los que van del divisor a uno de los cabezales por otro de distinta longitud, se vería como la muestra inicial donde se empieza a capturar señal tendría otro valor de amplitud debido a que la onda ha llegado más temprano o más tarde según la longitud aplicada.

6.2 P2: Tasa de muestreo

6.2.1 Descripción

En esta prueba se procede a comprobar cual es la mayor tasa de muestreo que se puede utilizar en recepción sin que se sature el enlace entre el switch y el ordenador personal. Se comprueba el funcionamiento mediante la reiteración de la misma prueba 20 veces y alargando el tiempo de recepción a 30 segundos.

Una cosa a tener en cuenta son los flags que aparecen en el terminal de comandos de Linux mediante los cuales se puede saber si se han perdido muestras. Los flags son los siguientes:

- **O (Overflow):** Se produce cuando la USRP intenta transmitir más datos de los que el búfer del ordenador o el enlace pueden soportar y conlleva el descarte de muestras. El archivo recibido y procesado tiene menos muestras de las que debería.

- **U (Underrun):** Se produce cuando el flujo de muestras continuo que debe recibir la USRP del ordenador se interrumpe. El archivo recibido y procesado tiene las muestras deseadas, pero algunas con valores aleatorios.

Cabe destacar que el formato de los datos definido en este proyecto es de Complex float32 para transmisión y Complex int16 para recepción. Es útil recurrir a (3.1) donde se muestra la expresión del *Data Transfer Rate* (DTR). En (6.1) se presenta (3.1) modificada para calcular el DTR total sobre el enlace donde se incluyen los 64 bits por muestra para el DTR transmisor (DTR_{Tx}) y los 32 bits por muestra para el DTR receptor (DTR_{Rx}) en cada uno de los cuatro canales.

$$DTR [Mbps] = 64 * S.RateTx [Msps] + 128 * S.RateRx [Msps] \quad (6.1)$$

Se empezará mostrando los resultados con los cuatro canales en recepción y un canal en transmisión donde la tasa de muestreo es la misma para receptor y transmisor. Posteriormente, se muestran los resultados donde la tasa de muestreo del transmisor se va disminuyendo.

6.2.2 Resultados

6.2.2.1 Misma tasa de muestreo en transmisión y en recepción

En la Tabla 6.2 se puede observar como conforme se va disminuyendo la tasa de muestreo va disminuyendo la cantidad de bits sobre el enlace y van aumentando las realizaciones correctas.

En el caso de la tasa de 6.25 Msps el número de realizaciones correctas es de 3 sobre 20. El DTR total del enlace supera el límite de 1 Gbps y en la mayoría de casos la USRP o el switch se acaban congestionando con los datos que no han podido enviar y los acaban descartando. Excepcionalmente, se reciben los datos 3 veces y el número de muestras al procesar los archivos parece correcto.

En el caso de la tasa de 5 Msps el número de realizaciones correctas ha aumentado debido a que el DTR total está dentro del límite de 1 Gbps del enlace. Aun así, el número de realizaciones correctas no es máximo lo que significa que, o el enlace no siempre soporta este DTR total, o el tamaño del búfer del ordenador en recepción no tiene el tamaño suficiente a la hora de crear el socket¹. Esto explicaría por qué a pesar de estar transfiriendo datos por el enlace dentro de lo permitido, cada vez que se genera el *socket*, no siempre se obtiene un tamaño de búfer consecuente al DTR en recepción y se acaban perdiendo datos.

¹*Socket*: mecanismo mediante el cual dos programas se localizan entre sí y establecen una comunicación.

Table 6.2: Resultados para la misma tasa de muestreo en transmisión y en recepción

S.Rate (Msps)	DTR _{Rx} (Mbps)	DTR _{Tx} (Mbps)	DTR _T (Mbps)	Intentos
6.25	800	400	1200	3/20
5	640	320	960	15/20
4.55	582	291	873	20/20

Table 6.3: Resultados para una tasa de muestreo de 5 Msps en transmisión

S.Rate (Msps)	DTR _{Rx} (Mbps)	DTR _{Tx} (Mbps)	DTR _T (Mbps)	Intentos
6.25	800	320	1120	4/20
5	640	320	960	15/20
4.55	582	320	902	20/20

En el caso de la tasa de 4.54 Msps el número de realizaciones correctas es máximo ya que el DTR total sobre el enlace tiene un margen de más de 100 Mbps y no hay problema en el tamaño del búfer al crear el *socket*.

6.2.2.2 Tasa de muestreo en transmisión más baja que en recepción

En la Tabla 6.3 se puede observar como fijando la tasa de transmisión a 5 Msps y disminuyendo la tasa de muestreo en recepción, la DTR total disminuye permitiendo que el número de intentos satisfactorio aumente, aunque como se ha comentado anteriormente para el caso de 5 Msps en recepción, el enlace no siempre puede asegurar que se realicen transferencias cerca de 1 Gbps o que el *socket* creado genere una ventana de datos en el búfer receptor lo suficientemente ancha. En el caso de disminuir la tasa de muestreo del receptor a 4.55 Msps el número de realizaciones correctas es máximo.

En el caso de fijar la tasa de muestreo en transmisión a 4.55 Msps, los resultados son parecidos a cuando fijas una tasa de 5 Msps (véase la Tabla 6.4).

En el caso de bajar la tasa de muestreo en transmisión a 0.5 Msps se puede observar que aun habiendo una DTR total por debajo del límite del enlace y con cierto margen de seguridad sobre el límite máximo de 1 Gbps, la reiteración de pruebas no lleva a un máximo de acierto en los casos de 6.25 y 5 Msps en el receptor. El número de aciertos sigue aumentando conforme disminuye la tasa de muestreo en recepción (véase la Tabla 6.5).

Table 6.4: Resultados para una tasa de muestreo de 4.55 Msps en transmisión

S.Rate (Msps)	DTR _{Rx} (Mbps)	DTR _{Tx} (Mbps)	DTR _T (Mbps)	Intentos
6.25	800	291	1091	3/20
5	640	291	931	17/20
4.55	582	291	873	20/20

Table 6.5: Resultados para una tasa de muestreo de 0.5 Msps en transmisión

S.Rate (Msps)	DTR _{Rx} (Mbps)	DTR _{Tx} (Mbps)	DTR _T (Mbps)	Intentos
6.25	800	32	832	3/20
5	640	32	672	16/20
4.55	582	32	614	20/20

Aunque el DTR total sobre el enlace sea correcto, el *socket* generado en el ordenador no crea un tamaño de búfer lo suficientemente ancho como para recibir el DTR de recepción. Para solucionar este problema habría que redimensionar el tamaño máximo de los búferes. En Linux (que es el sistema operativo que se está usando) se pueden redimensionar los búferes de transmisión y recepción respectivamente utilizando los comandos:

```
sudo sysctl -w net.core.wmem_max=<new value>
```

```
sudo sysctl -w net.core.rmem_max=<new value>
```

Los valores implementados son: 1048576 y 110000000. Estos valores están expresados en bytes. El primer valor equivale a una ventana de 8 Mbits y el segundo valor equivale a 840 Mbits. En emisión no había problemas, el problema era en recepción y aún aumentando la ventana a casi 840 Mbits (por encima del DTR en recepción de 800 Mbits para 6.25 Msps) no se consigue que funcione durante un número de intentos prolongado ni a 5 Msps (640 Mbits en recepción).

Cabe destacar que cuando sucede un error la USRP deja de capturar datos, por lo que no es recomendable que el enlace se sature. A pesar de esto, se permite la transferencia de datos con una tasa de muestreo de 4.55 Msps para los cuatro canales funcionando al mismo tiempo y sin error. Si se quisiera trabajar solo con un canal individual, la tasa de muestreo en recepción asciende a 12.5 Msps, y si se quisiera trabajar con dos canales (uno de cada TwinRX) la tasa de muestreo en recepción asciende a 10 Msps en cada canal utilizado.

6.3 P3: Medida de la sincronización en tiempo entre canales

6.3.1 Descripción

En esta prueba se procede a medir la sincronización en tiempo entre los canales utilizando la correlación cruzada y el coeficiente de correlación.

Para ello, en el caso del transmisor se utiliza una señal triangular real con el mismo esquema que se muestra en la Figura 6.2 y con los mismos parámetros que se muestran en la Tabla 6.1.

En el caso del receptor, primero se utiliza el esquema mostrado en la Figura 6.1 indicando como *Unknown PPS* la opción de sincronización y, posteriormente, indicando la opción *No Sync*. Se mantiene en ambos casos la opción de importar el reloj externo y se utilizan las ganancias de 47 dB, 52 dB, 47 dB y 65 dB usadas en la primera prueba para los canales 1, 2, 3 y 4 respectivamente ya que conforman bien la señal. El tiempo de captura es de 10 segundos.

El objetivo de esta prueba es comprobar si hay retardos entre los canales de la USRP X300 y comentar qué es lo que sucede.

6.3.2 Resultados

6.3.2.1 Receptor con los cabezales sincronizados en tiempo

Lo primero que se hace es comprobar que la forma de onda sea triangular², que es la onda que se está transmitiendo. En la Figura 6.7 se muestran las primeras 1000 muestras de la señal en módulo y se ve como la onda recibida tiene la forma de una onda triangular.

Lo siguiente sería comprobar que el periodo de la señal es el mismo que el de la señal generada en transmisión, así se sabría que no se han perdido muestras. La señal generada en transmisión tiene una frecuencia de 5 kHz y, por tanto, un periodo de 0.2 ms. En la Figura 6.8 se muestra el valor del periodo medido entre todos los picos máximos de la señal para cada canal en un segundo. Teniendo en cuenta que el valor del intervalo en tiempo entre muestras es el inverso al de la frecuencia de muestreo utilizada ($1/F_s = 2\mu s$) y que la cantidad de muestras en un periodo es de 100 según la Figura 6.8, un periodo en recepción equivale a 0.2 ms, que es el mismo valor que tenía el periodo de la señal en transmisión.

Aunque visualmente se puede observar en la Figura 6.7 que no hay retardo entre las señales captadas por los distintos canales (ya que se superponen), se realiza la correlación del primer segundo de muestras (equivalente a 500000 muestras) utilizando la función *xcorr* de MATLAB.

²Se recuerda que la onda triangular tiene una media de la mitad de la amplitud por lo que va desde 0 hasta el valor de amplitud introducido.

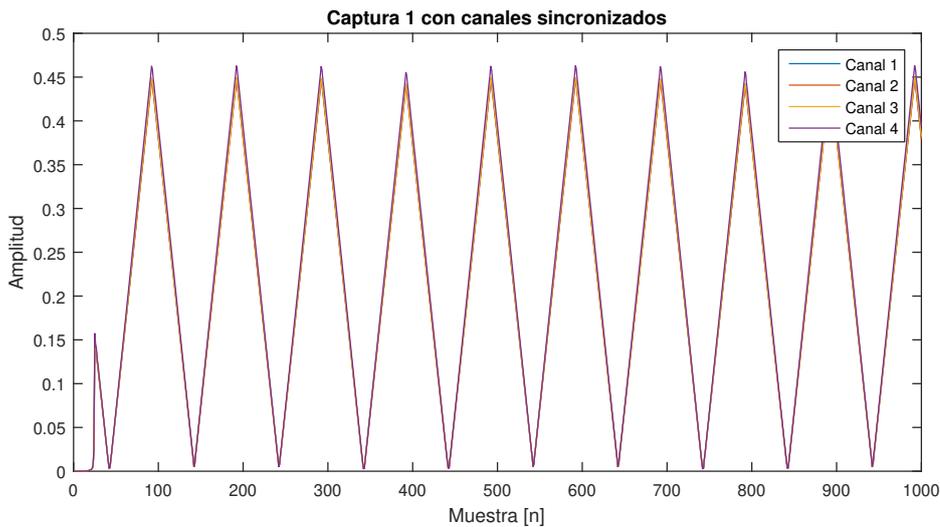


Figure 6.7: Primeras muestras en módulo con canales sincronizados de la captura 1

El resultado se muestra en la Tabla 6.6. En ella se observa que el retardo en muestras entre canales obtenido al hacer la correlación es nulo y que los coeficientes de correlación son cercanos a la unidad, pero no iguales. Hay que tener en cuenta que un retardo en muestras nulo equivale a un retardo en tiempo nulo.

Que el coeficiente de correlación no sea uno no significa que no sea máximo. El cálculo de la correlación cruzada conlleva utilizar señales que no son exactamente las mismas para cada canal y varían debido al medio atravesado y al ruido.

Se recupera la expresión de la correlación para valores de tiempo continuos en (6.2). Si se transforma la expresión para valores de tiempo discretos se obtiene (6.3) (expresión que utiliza la función `xcorr` de MATLAB para varios lags) donde $corr_{cxcy}$ es la correlación cruzada en el dominio digital, k indica el retardo en muestras, N indica el número total de muestras, ζ_{ncx} y ζ_{ncy} con sus subíndices cx y cy indican el retardo en grupo de cada canal expresado en muestras y n indica la muestra.

$$R_{cxcy}(\zeta) = \int_{-\infty}^{\infty} s_{tx}(t - \zeta_{cx} + \zeta_{cy} + \zeta) s_{tx}(t) e^{j(\vartheta_{cx} - \vartheta_{cy})} dt \quad (6.2)$$

$$corr_{cxcy}[k] = \sum_{n=0}^{N-1} s_{tx}[n - \zeta_{ncx} + \zeta_{ncy} + k] s_{tx}[n] e^{j(\vartheta_{cx} - \vartheta_{cy})} \quad (6.3)$$

Finalmente, se aplica que la señal recibida en cada canal no será exactamente la misma, por lo que se tendrán dos señales ligeramente diferentes s_{txcy} y s_{txcy} , el resultado se muestra en (6.4).

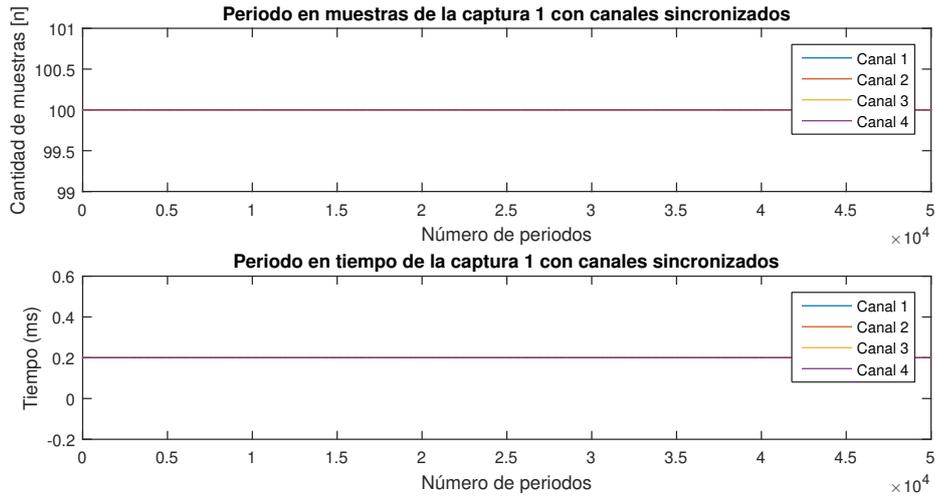


Figure 6.8: Periodo de la captura 1 con canales sincronizados para un segundo de muestras

$$corr_{cxcy}[k] = \sum_{k=0}^{N-1} s_{txcx}[n - \zeta_{ncx} + \zeta_{ncy} + k] s_{txcy}[n] e^{j(\vartheta_{cx} - \vartheta_{cy})} \quad (6.4)$$

El concepto explicado en el Capítulo 4 a través de la correlación es totalmente válido para la correlación de muestras expresada en este apartado. Por lo que si los retardos en grupo ζ_{ncx} y ζ_{ncy} son iguales, el coeficiente de correlación será máximo (igual o próximo a uno) cuando el retardo en muestras k sea nulo.

Suponiendo que los retardos en grupo para cada canal son iguales debido a que el trayecto tiene las mismas dimensiones y que los osciladores están alineados en tiempo debido a la sincronización, el retardo entre canales debe ser nulo ya que todos los canales empiezan a capturar con la misma marca de tiempo.

Para comprobar que no se trata de una casualidad y que realmente los osciladores están alineados en tiempo, se realiza otra captura de datos con los mismos parámetros. El resultado se muestra en la Tabla 6.7, donde se obtienen valores para los coeficientes de correlación cercanos a la unidad y con un retardo en muestras y en tiempo nulos.

Es necesario utilizar los cuatro canales sincronizados en tiempo ya que asegura que, sea cual sea el momento del día en el que se realice una captura, no habrá retardo entre canales.

6.3.2.2 Receptor con los cabezales no sincronizados en tiempo

En caso de que los canales no estén sincronizados, se muestra un retardo como en la Figura 6.9. Como se comentó anteriormente, el intervalo entre muestras equivale a la inversa de la frecuencia

Table 6.6: Retardo entre canales sincronizados en la captura 1

	Ch1-Ch2	Ch1-Ch3	Ch1-Ch4
Coef. Correlación	0.9995	0.9990	0.9994
Retardo en muestras	0	0	0
Retardo en tiempo (μs)	0	0	0

Table 6.7: Retardo entre canales sincronizados en la captura 2

	Ch1-Ch2	Ch1-Ch3	Ch1-Ch4
Coef. Correlación	0.9996	0.9991	0.9993
Retardo en muestras	0	0	0
Retardo en tiempo (μs)	0	0	0

Table 6.8: Retardo entre canales no sincronizados en la captura 1

	Ch1-Ch2	Ch1-Ch3	Ch1-Ch4
Coef. Correlación	0.9996	0.9985	0.9989
Retardo en muestras	0	15	15
Retardo en tiempo (μs)	0	30	30

Table 6.9: Retardo entre canales no sincronizados en la captura 2

	Ch1-Ch2	Ch1-Ch3	Ch1-Ch4
Coef. Correlación	0.9996	0.9983	0.9983
Retardo en muestras	0	-39	-39
Retardo en tiempo (μs)	0	78	78

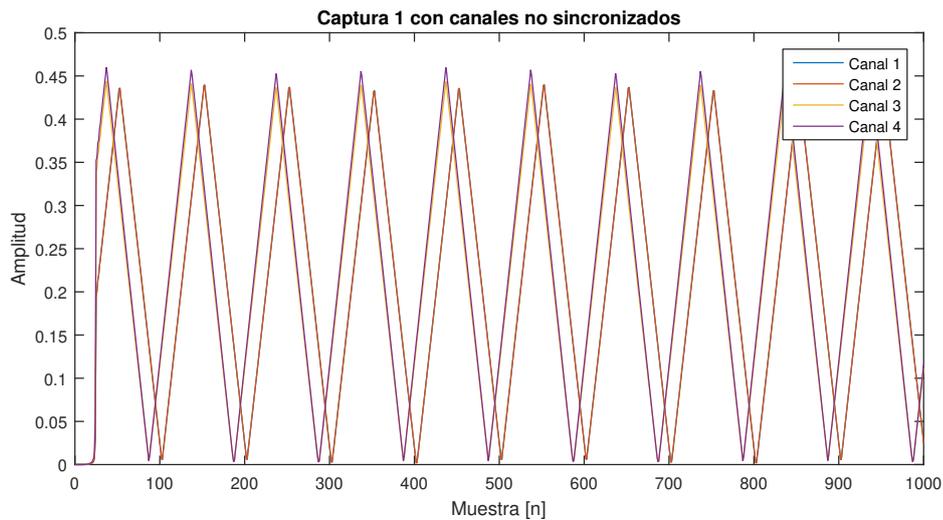


Figure 6.9: Primeras muestras en módulo con canales no sincronizados de la captura 1

de muestreo, por lo que el retardo en tiempo entre los canales es el producto entre las muestras retardadas y la inversa de la frecuencia de muestreo.

Los canales pertenecientes a la misma TwinRX están sincronizados entre sí, pero como la sincronización no se aplica entre tarjetas TwinRX, los osciladores internos de cada tarjeta empiezan a capturar con una marca de tiempo diferente. Esta diferencia de hora o marca de tiempo introduce un retardo en muestras entre los distintos canales de las diferentes TwinRX. Para la primera captura que se realiza se obtiene un retardo de 15 muestras equivalente a $30 \mu\text{s}$ (véase la Tabla 6.8).

Si se prueba con otra captura diferente el retardo no permanece constante, sino que varía, pero se sigue cumpliendo la condición de que los canales de la misma TwinRX están sincronizados entre sí. En la Tabla 6.9 se muestra como el retardo para los canales 3 y 4 respecto al canal 1 es de -39 muestras, que en tiempo equivale a $78 \mu\text{s}$.

El uso de la no sincronización temporal podría ser útil para capturar señales por dos canales que utilicen la misma TwinRX.

6.4 P4: Medida de la sincronización en fase y frecuencia (1)

6.4.1 Descripción

En esta prueba se procede a medir la sincronización en fase y frecuencia entre canales que utilizan osciladores locales independientes en la etapa de RF y están sincronizados en tiempo.

Para ello, el esquema transmisor se mantiene constante con los mismos parámetros iniciales, y el esquema receptor se mantiene igual que en la prueba anterior, pero con la opción de sincronización *Unknown PPS* seleccionada para asegurar la sincronización en tiempo.

Para esta prueba se utilizarán ocho capturas de 10 segundos cada una tomadas a lo largo de un día (mañana, tarde y noche) y una mañana, y será fundamental el uso de la correlación cruzada para medir el desfase entre canales.

6.4.2 Resultados

Para empezar, se procede a mostrar la fase junto con la parte real y la parte imaginaria de un rango de muestras tomadas de la primera captura realizada mediante el canal 1 (véase la Figura 6.10). Esta primera captura es la captura 1 utilizada en la prueba anterior con sincronización.

En la Figura 6.10 se puede observar como existen algunos picos en la fase cuando debería ser aproximadamente constante. Estos picos aparecen porque la señal generada en recepción no es exactamente igual que la generada en transmisión. En transmisión, la amplitud de la señal va desde cero hasta la amplitud definida mientras que, en recepción, la amplitud de la señal va desde un poco menos de cero hasta el máximo de amplitud, haciendo que a la hora de calcular la fase mediante atan2 , ésta cambie π radianes su valor en ciertas muestras.

Para solucionar esto se pueden utilizar señales triangulares reales que tengan un offset en amplitud. En este caso se añade un offset de 0.15 en amplitud a la señal generada en el transmisor USRP N200 (esquema mostrado en la Figura 6.2). Una vez modificado el parámetro se vuelve a mostrar la fase, pero en este caso para los cuatro canales y con todas las muestras de la captura.

En la Figura 6.11 se muestran las fases o retardos de fase de cada canal donde ya no aparecen estos picos mencionados. Estas fases contienen el retardo de grupo introducido por los cables SMA, las variaciones de fase introducidas por el divisor de potencia y las fases introducidas por los osciladores locales de la etapa de RF en cada canal. Además, se puede observar como las fases evolucionan debido a una ligera pendiente conforme avanza el número de muestra. Esta pendiente está provocada por una ligera diferencia entre las frecuencias de modulación f_{cMOD} y de demodulación f_{cDEMx} , donde la frecuencia de demodulación dependerá del canal y de su etapa RF.

En (6.5) se muestra la expresión de la señal recibida presentada en el Capítulo 4.

$$x_x(t) = y_x(t)e^{-j(2\pi f_{cDEM}t + \varphi_x)} = s(t - \zeta_x)e^{j(2\pi((f_{cMOD} - f_{cDEMx})t) - \zeta_x + \varphi_{yx} - \varphi_x)} \quad (6.5)$$

En ella se puede observar que si las frecuencias de modulación y demodulación fueran exactamente las mismas, la fase no iría variando linealmente con el tiempo, sino que sería constante,

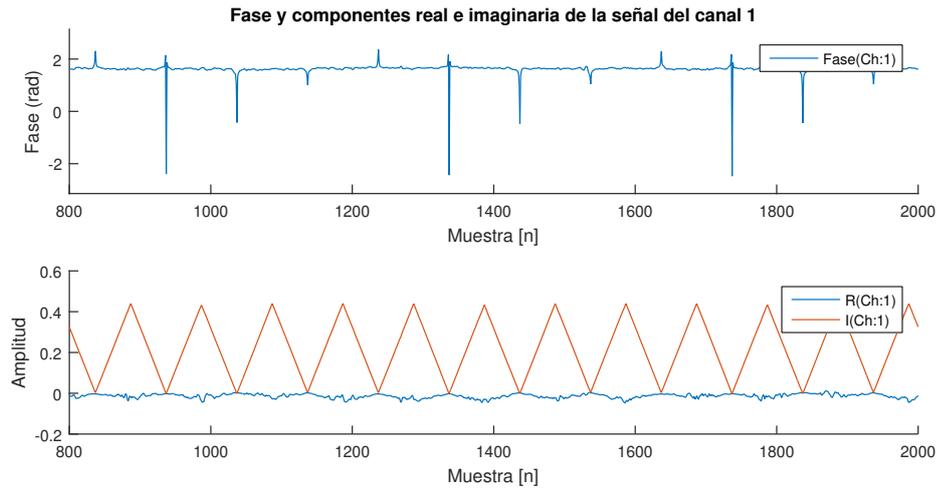


Figure 6.10: Fase y componentes real e imaginaria de una señal triangular real sin offset

pero esto no ocurre. Es por eso que es importante comprobar la sincronización en frecuencia entre canales, dado que si la frecuencia de demodulación fuera igual para todos los canales, las fases evolucionarían al mismo tiempo y se produciría alineamiento en frecuencia. Según la Figura 6.11, los canales 1 y 3 no aparentan estar alineados en frecuencia ya que las pendientes parecen distintas a simple vista.

6.4.2.1 Medida de la sincronización en fase

Para medir la sincronización en fase es necesario primero calcular los desfases entre los canales. Se toma como referencia el canal 1 y se utiliza la correlación mostrada en (6.4) particularizada para el caso en que hay sincronización en tiempo.

$$corr_{cxcy}[0] = \sum_{n=0}^{N-1} s_{txcx}[n]s_{txcy}[n]e^{j(\vartheta_{cx}[n]-\vartheta_{cy}[n])} = \sum_{n=0}^{N-1} s_{txcx}[n]s_{txcy}[n]e^{j\varphi_{cxcy}} \quad (6.6)$$

La diferencia de fases se puede medir extrayendo el exponente φ_{cxcy} de la correlación.

Sabiendo esto, se calcula la correlación cruzada entre las señales recibidas por los cabezales 1 y 2, los cabezales 1 y 3 y los cabezales 1 y 4 para las ocho capturas. Posteriormente, se obtienen los desfases extraídos de cada correlación y se muestran en la Tabla 6.10.

Las capturas 1, 2, 3, 4, 5 y 6 se han realizado con el transmisor emitiendo y el receptor encendido en todo momento. En cada captura, el receptor se ha ido activando para recibir los datos. Las capturas 7 y 8 se han realizado al día siguiente manteniendo el transmisor emitiendo, pero apagando entre captura y captura el receptor.

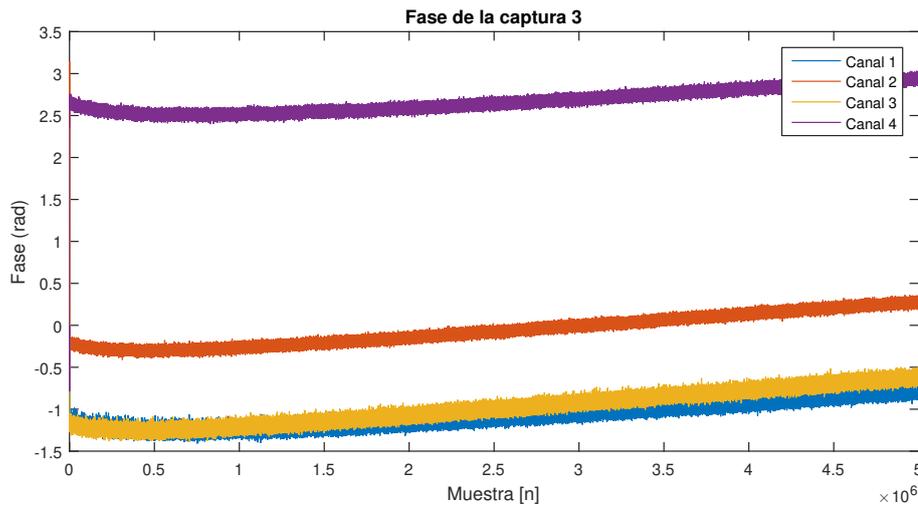


Figure 6.11: Fases de la captura 3 para cada canal

Table 6.10: Desfase entre los canales en radianes

Captura	1	2	3	4	5	6	7	8
φ_{12}	-2.1079	-0.7502	-1.0230	0.3570	-0.3158	0.5738	2.0693	-2.3548
φ_{13}	1.9924	1.0054	-0.0960	-1.4046	-2.0591	2.4069	1.0743	2.9352
φ_{14}	-0.1387	2.2227	2.5288	1.2296	2.8131	-0.3395	-2.3816	-2.3914

Aparentemente, los resultados obtenidos para el desfase entre los canales no tienen ninguna relación lógica, ni para el caso en que la USRP en recepción se mantiene encendida entre capturas, ni para el caso en que se apaga. Estos desfases sirven para alinear la fase de las señales de cada canal con el canal 1 que hace de referencia, pero habrá que contemplar la posibilidad de que las frecuencias centrales de demodulación entre canales sean diferentes entre sí, y que, por tanto, las fases evolucionen de manera distinta en cada canal.

Esto conllevaría que pasado un tiempo las fases volviesen a desalinearse produciéndose otro desfase diferente. En la Figura 6.11, las pendientes de los canales 1 y 3 parecen diferentes.

6.4.2.2 Medida de la sincronización en frecuencia

Como se acaba de comentar, es importante contemplar la posibilidad de que cada canal introduzca un error diferente en la frecuencia a la hora de demodular la señal. Este error se introduce debido a que los osciladores locales de la etapa de RF no son capaces de demodular la señal recibida exactamente con la misma frecuencia con la que ha modulado el transmisor (véase

Table 6.11: Diferencias entre la frecuencia moduladora y la frecuencia demoduladora en Hz para cada canal

Capturas	1	2	3	4	5	6	7	8
$diff_{DEM1}$	0.0118	0.0118	0.0095	0.0097	0.0094	0.0097	0.0095	0.0092
$diff_{DEM2}$	0.0110	0.0093	0.0111	0.0093	0.0089	0.0101	0.0090	0.0090
$diff_{DEM3}$	0.0121	0.0118	0.0119	0.0124	0.0118	0.0100	0.0097	0.0119
$diff_{DEM4}$	0.0113	0.0091	0.0089	0.0114	0.0090	0.0091	0.0087	0.0088

(6.5)).

Para calcular la pendiente de la fase se hace un promedio de la señal convolucionada con un pulso unitario para cada canal y evitando los primeros segundos de no linealidad. El resultado de la pendiente dependerá del tamaño del pulso unitario y del rango temporal con el que se hace la convolución.

El tamaño del pulso unitario utilizado es 10 veces más grande que el periodo de la señal expresado en muestras (1000 muestras). La pendiente o diferencia entre las frecuencias de modulación y demodulación para cada canal en Hz se recogen en la Tabla 6.11. Se puede observar como la diferencia entre las frecuencias tiene un valor parecido entre canales y a lo largo del tiempo, pero no son exactamente iguales. Esto se debe a que cada oscilador local de cada canal funciona de manera independiente y cada uno genera la frecuencia de demodulación haciendo su mejor esfuerzo.

Estas diferencias harán que si se alinean en fase las señales de los canales 2, 3 y 4 con respecto al canal 1, al cabo del tiempo volverá a producirse otro desfase que consecuentemente habrá que corregir.

6.5 P5: Medida de la sincronización en fase y frecuencia (2)

6.5.1 Descripción

En esta prueba se continúa con el objetivo de medir la sincronización en fase y frecuencia entre canales, pero en este caso se utiliza la exportación de las señales de los osciladores locales de un canal.

Si se observa el esquema presentado en la Figura 6.1, se puede ver que las opciones *LO Source* y *LO Export* están activadas con los valores para cada canal en este orden: '*Internal, True*',

Table 6.12: Desfase entre los canales en radianes utilizando la señal de los LOs de un canal como referencia

Captura	1	2	3	4	5	6
φ_{12}	-0.0725	1.5235	3.0869	1.5228	3.0728	-0.0707
φ_{13}	-1.0222	0.5580	-1.0142	2.1130	2.1214	-2.5829
φ_{14}	-0.1387	1.4219	-0.1667	1.4069	-0.1609	2.9808

'Companion, False', 'External, False' y 'External, False'. Estos serán los cambios aplicados con respecto a la prueba anterior. Todos los demás parámetros se mantienen.

Para esta prueba se utilizarán seis capturas realizadas con una señal triangular real con offset, tres capturas realizadas con una senoide real, tres con una señal triangular compleja y tres con una exponencial compleja. Además, se probará a cambiar las ganancias de cada canal en recepción para ver como varía el desfase entre canales.

6.5.2 Resultados

Aplicando los métodos igual que en las sección 6.4.2.1 y 6.4.2.2 se obtienen las Tabla 6.12 y 6.13 donde se muestran los desfases φ_{12} , φ_{13} y φ_{14} y la diferencia entre frecuencias moduladora y demoduladora en Hz para cada canal.

En la Tabla 6.12 se recogen los desfases entre las distintas capturas realizadas. En cuanto a las diferencias entre frecuencias, ahora se mantienen constantes para todos los canales por lo que la frecuencia entre canales estará alineada (véase la Tabla 6.13).

En la Tabla 6.14, para facilitar la visualización de los desfases presentados en la Tabla 6.12, se muestran los desfases a partir de la captura 1 y añadiendo $\pm\pi/2$ o $\pm\pi$ radianes.

Estas capturas han sido tomadas a lo largo de varios días manteniendo el esquema con los mismos parámetros, y manteniendo el receptor encendido entre capturas y desconectándolo de la fuente de corriente. Además, alguna ha sido tomada después de modificar los parámetros del esquema, capturar señal, y volver a poner los parámetros como estaban.

Los resultados de reducir la ganancia de cada canal en 10 dB, se muestran en la Tabla 6.15. En la Tabla 6.16 se muestran los desfases de la Tabla 6.15 a partir de la captura 1 y añadiendo $\pm\pi/2$ o $\pm\pi$ radianes.

Aparentemente, al cambiar las ganancias de cada canal, los valores de desfase cambian, por lo menos eso se aprecia en el caso del desfase entre el canal 1 y el canal 2. En los desfases entre

Table 6.13: Diferencias entre la frecuencia moduladora y la frecuencia demoduladora en Hz para cada canal utilizando la señal de los LOs de un canal como referencia

Capturas	1	2	3	4	5	6
dif_{DEM1}	0.0112	0.0107	0.0105	0.0108	0.0106	0.0092
dif_{DEM2}	0.0112	0.0107	0.0105	0.0108	0.0106	0.0092
dif_{DEM3}	0.0112	0.0107	0.0105	0.0108	0.0106	0.0092
dif_{DEM4}	0.0112	0.0107	0.0105	0.0108	0.0106	0.0092

Table 6.14: Desfase entre los canales añadiendo $\pm\pi/2$ y $\pm\pi$ radianes con respecto a la captura 1

Captura	1	2	3	4	5	6
φ_{12}	-0.0725	$-0.0473 + \frac{\pi}{2}$	$-0.0547 + \pi$	$-0.0480 + \frac{\pi}{2}$	$-0.0688 + \pi$	-0.0707
φ_{13}	-1.0222	$-1.0128 + \frac{\pi}{2}$	-1.0142	$-1.0286 + \pi$	$-1.0202 + \pi$	$-1.0121 - \frac{\pi}{2}$
φ_{14}	-0.1387	$-0.1489 + \frac{\pi}{2}$	-0.1667	$-0.1639 + \frac{\pi}{2}$	-0.1609	$-0.1608 + \pi$

Table 6.15: Desfase entre los canales en radianes con reducción de 10 dB en ganancia

Captura	1	2	3	4
φ_{12}	-0.1453	-1.7012	1.4362	3.0187
φ_{13}	-1.0162	2.1213	-1.0315	-2.5866
φ_{14}	-0.1414	-1.7212	-1.7035	-1.7173

Table 6.16: Desfase entre los canales añadiendo $\pm\pi/2$ y $\pm\pi$ radianes con respecto a la captura 1 con reducción de 10 dB en ganancia

Captura	1	2	3	4
φ_{12}	-0.1453	$-0.1304 - \frac{\pi}{2}$	$-0.1346 + \frac{\pi}{2}$	$-0.1229 + \pi$
φ_{13}	-1.0162	$-1.0202 + \pi$	-1.0315	$-1.0158 - \frac{\pi}{2}$
φ_{14}	-0.1414	$-0.1504 - \frac{\pi}{2}$	$-0.1327 - \frac{\pi}{2}$	$-0.1465 - \frac{\pi}{2}$

Table 6.17: Desfase en radianes entre los canales con un coseno

Captura	1	2	3
φ_{12}	-0.0541	-0.0496	-0.0610
φ_{13}	-1.0266	-1.0118	$-1.0256 + \pi$
φ_{14}	$-0.1653 + \pi$	$-0.1495 + \frac{\pi}{2}$	$-0.1598 + \pi$

Table 6.18: Desfase en radianes entre los canales con una señal triangular compleja

Captura	1	2	3
φ_{12}	$-0.0705 + \frac{\pi}{2}$	$-0.0718 + \frac{\pi}{2}$	-0.0746
φ_{13}	$-1.0282 - \frac{\pi}{2}$	$-1.0213 + \pi$	-1.0272
φ_{14}	$-0.1723 + \pi$	$-0.1772 + \frac{\pi}{2}$	$-0.1528 - \frac{\pi}{2}$

los canales 1 y 3 y 1 y 4 no se aprecia gran cambio. El error de estimación producido a la hora de calcular los desfases oculta la pequeña variación de fase si la hubiera.

Esto significa que cuando se definen los parámetros de diseño del receptor que afectan a la etapa de RF (ganancias, filtros y frecuencia central) los desfases que se obtienen pueden tener valores aleatorios. Una vez definidos, si se mantienen los mismos parámetros entre captura y captura, los valores de desfase se mantienen fijos variando 0, $\pm\pi/2$ o $\pm\pi$ radianes entre captura y captura debido al funcionamiento de la cadena VCO/PLL.

El hecho de que la diferencia entre la frecuencia moduladora y la frecuencia demoduladora sea la misma entre canales indica que el desfase entre canales una vez inicializado el tiempo de captura evolucionará de la misma manera y no se producirán desfases posteriores.

Ahora se cambia el tipo de señal utilizado poniendo los valores de ganancia antes de restar

Table 6.19: Desfase en radianes entre los canales con una exponencial compleja

Captura	1	2	3
φ_{12}	-0.0639	$-0.0756 + \pi$	-0.0759
φ_{13}	$-1.0183 + \pi$	-1.0336	-1.0234
φ_{14}	$-0.1772 + \pi$	$-0.1609 + \pi$	$-0.1536 - \frac{\pi}{2}$

10 dB, para ver si el desfase cambia o se mantiene constante. En la Tabla 6.17 se muestran los desfases para una senoide real como el coseno. En la Tabla 6.18 se muestran los desfases para una señal triangular compleja, y en la Tabla 6.19 los desfases para una exponencial compleja.

En todos los casos probados para las distintas señales se obtienen desfases similares a los desfases obtenidos con una señal triangular real. Hay que destacar que no son exactamente los mismos desfases debido a que existe un error de estimación producido por el método empleado para realizar el cálculo.

6.6 P6: Desfase entre las salidas del divisor de RF

6.6.1 Descripción

En esta prueba se pretende medir el desfase entre las salidas del divisor de potencia a partir de los desfases entre canales.

Para llevar a cabo esta prueba, se utilizan unas expresiones que se presentan a continuación. El desfase entre canales se mide como el desfase producido por los osciladores locales más el desfase producido por las salidas del divisor de potencia. En (6.7) se contempla el caso entre los canales 1 y 2, y en (6.8) se contempla el caso entre los canales 3 y 4.

$$\varphi_{12} = \varphi_{LO12} + \varphi_{Div12} \quad (6.7)$$

$$\varphi_{34} = \varphi_{LO34} + \varphi_{Div34} \quad (6.8)$$

Donde se tiene en cuenta que las salidas del divisor de potencia coinciden en índice con las entradas de los canales en recepción.

Si cambias el indexado haciendo que la salida 1 vaya a la entrada 2 y la salida 2 a la entrada 1 (y lo mismo para los canales 3 y 4) se obtiene que:

$$\varphi'_{12} = \varphi_{LO12} - \varphi_{Div12} \quad (6.9)$$

$$\varphi'_{34} = \varphi_{LO34} - \varphi_{Div34} \quad (6.10)$$

Para conocer el desfase entre las salidas 1 y 2 y las salidas 3 y 4 del divisor de RF se realiza la diferencia entre desfases de las señales recibidas.

Table 6.20: Desfases entre salidas del divisor de RF

Desf. capt.	Cables Directos	Cables Invertidos	φ_{Div} (rad)	φ_{Div} (grados)
φ_{12cap1}	-0.0725	-0.1157	0.0216	1.24
φ_{34cap1}	0.8835	0.8940	0.0053	0.30
φ_{12cap2}	-0.0725	-0.1379	0.0327	1.87
φ_{34cap2}	0.8835	0.8713	0.0610	0.35
φ_{12cap3}	-0.0725	-0.1420	0.0347	1.99
φ_{34cap3}	0.8835	0.8816	0.0010	0.05
φ_{12cap4}	-0.0725	-0.1191	0.0233	1.34
φ_{34cap4}	0.8835	0.8971	0.0068	0.39
φ_{12cap5}	-0.0725	-0.1177	0.0226	1.29
φ_{34cap5}	0.8835	0.8661	0.0087	0.50
φ_{12cap6}	-0.0725	-0.1377	0.0326	1.87
φ_{34cap6}	0.8835	0.8946	0.0055	0.32

$$\varphi_{Div12} = \frac{\varphi_{12} - \varphi'_{12}}{2} \quad (6.11)$$

$$\varphi_{Div34} = \frac{\varphi_{34} - \varphi'_{34}}{2} \quad (6.12)$$

Si la diferencia entre las salidas del divisor de RF fuera nula, el desfase obtenido al invertir los cables sería el mismo.

6.6.2 Resultados

Los valores obtenidos para los desfases entre los canales 1 y 2 y los canales 3 y 4 se muestran en la Tabla 6.20.

Los resultados de desfase entre salidas del divisor obtenidos son acordes con los datos proporcionados en el datasheet donde se comentaba que la variación máxima de fase introducida por cada salida era de 6 grados y para una frecuencia de 1.5 GHz era de 1.97 grados típicamente.

Esto significa que la fase en la salida 1 o en la salida 2 es muy alta, mientras que en la otra

salida es muy baja, de ahí que el desfase entre salidas sea tan próximo a 1.97 grados. En el caso de las salidas 3 y 4 tienen una variación de fase relativamente pequeña lo que significa que cada salida introduce una fase parecida.

Hay que tener en cuenta que se ha supuesto que los cables que conectan el divisor con el receptor (USRP X300) son iguales, pero esto no es así en la realidad. La frecuencia que se está utilizando es de 1.57542 GHz, lo que en longitud de onda serían 19 cm. Debido a cualquier pequeña diferencia aparentemente despreciable en la longitud de los cables, esto podría añadir una fase adicional que habría que añadir a la ecuación.

Para medir bien el desfase entre las salidas del divisor, habría que intercambiar los cables utilizados para conectar los distintos canales del divisor y el receptor. Aún así, esto tampoco aseguraría que los resultados obtenidos fueran adecuados ya que también hay que tener presente que existe un error de estimación que dificulta la medición.

Chapter 7

Conclusiones y futuras líneas de investigación

7.1 Conclusiones

El enfoque principal de este proyecto ha sido medir la sincronización en tiempo, fase y frecuencia entre los cabezales de radiofrecuencia de la USRP X300 disponible en el laboratorio. Para cumplir con el objetivo, se ha tenido que aprender a utilizar el entorno de GNU Radio y el programa GNU Radio Companion. Además, se ha tenido que investigar el funcionamiento de los dispositivos USRP y las distintas opciones que permitían para que funcionaran de acuerdo a las necesidades de este proyecto. A través de este aprendizaje se han podido diseñar los esquemas ejecutables adecuados que permitían comunicar el ordenador con los dispositivos USRP, haciendo que la USRP N200 funcionara como transmisor y la USRP X300 como receptor multicanal.

En el Capítulo 2, se ha presentado una visión general de lo que son las Radios Definidas por Software. El objetivo principal era entender el funcionamiento interno de este tipo de sistemas para, posteriormente, concretar en los dispositivos USRP disponibles en el laboratorio y que se utilizarían a lo largo de este proyecto.

En el Capítulo 3, se ha descrito el tipo de sistema necesario para medir la sincronización, mediante el cual se ha podido definir el material necesario para montar la plataforma de medida de la sincronización entre canales.

En el Capítulo 4, se ha presentado el modelo de sistema de comunicaciones que representa la plataforma de medida montada. Además, se han explicado conceptos necesarios para realizar las medidas.

En el Capítulo 5, se ha implementado el diseño mediante bloques en el programa GNU Radio

Companion que permite la comunicación entre ordenador y dispositivos USRP y que genera la señal para ser transmitida a través de un canal y recibida por la USRP X300.

Por último, en el Capítulo 6 se han presentado las diferentes pruebas realizadas para comprobar la recepción de las señales, la tasa máxima de muestreo con la que se pueden muestrear las señales recibidas sin que haya error, y las medidas de sincronización en tiempo, fase y frecuencia realizadas.

Los objetivos propuestos al inicio de este proyecto eran:

- Comprobar cuántos cabezales de frecuencia podían funcionar al mismo tiempo en la USRP X300.
- Medir la tasa máxima de muestreo que podía alcanzar la USRP X300 por cada canal utilizándose el máximo de canales disponibles.
- Medir la sincronización en tiempo, fase y frecuencia entre los canales del receptor.

Mediante la realización de este proyecto se han validado a partir de los datos capturados y de los resultados obtenidos que los objetivos se han cumplido.

La USRP X300 con las dos TwinRX puede utilizar los cuatro canales indistintamente según interese. Para el caso de utilizar un único canal y mediante el enlace Ethernet a través de la tarjeta de red de 1 Gbit instalada en el ordenador, la tasa máxima de muestreo permitida sin error en los datos es de 12.5 Msps. Si se utilizan dos canales (uno de cada TwinRX) la tasa máxima de muestreo de cada canal es de 10 Msps. Por último, si se utilizan los cuatro canales del receptor, la tasa máxima de muestreo es de 4.55 Msps.

Teniendo en cuenta que interesaba trabajar con el máximo de canales disponibles, la medida de sincronización en tiempo, fase y frecuencia se realizó para los cuatro canales del receptor.

Los resultados obtenidos de la primera prueba indican que definir la misma ganancia para todos los canales no resulta en la obtención de la señal con el mismo nivel de potencia. Esto es curioso porque la señal transmitida desde la USRP N200 sale con un nivel de potencia definido, atraviesa el canal formado por los cables y el divisor de potencia (que distribuye uniformemente la señal por todas sus salidas) y llega al receptor, por lo que, asignando los mismos valores de ganancia para cada canal, el nivel de potencia de la señal debería ser el mismo en todos los casos.

Aun así, que los valores de ganancia definidos actúen de manera diferente no es del todo crítico. Fijando la ganancia de uno de los canales y variando la de los demás, se puede llegar a calibrar la diferencia existente entre las ganancias de cada canal. Una vez calibrado, si aumentas o disminuyes la ganancia un valor constante en todos los canales, el nivel de potencia de la señal recibida por cada canal es el mismo.

Los resultados obtenidos de la tercera prueba indican que los canales que pertenecen a la misma TwinRX están sincronizados en tiempo en todos los casos. En caso de aplicar sincronización mediante la opción de GNU Radio Companion, los canales entre TwinRX no presentarán retardos entre sí, haciendo que los canales estén alineados en tiempo. En caso de no aplicar sincronización, los canales entre TwinRX presentarán un retardo aleatorio para cada captura.

De la cuarta prueba se puede concluir que los osciladores locales de la etapa de RF de cada canal funcionan independientemente entre sí, haciendo que, a la hora de demodular la señal recibida, cada canal presente un error de frecuencia diferente que, consecuentemente, derive en una evolución de las fases distinta. Este error de frecuencia hace que por mucho que intentes alinear las fases de las señales recibidas por los diferentes canales, con el tiempo se acabarán desalineando de nuevo. Esto indica que no existe alineamiento en frecuencia.

Además, también se puede concluir que, si no utilizas un canal para exportar la referencia del oscilador local a los demás, los desfases en cada captura serán totalmente aleatorios. Sin embargo, que los desfases entre captura y captura varíen no significa que no se puedan alinear en fase las distintas señales (siempre teniendo en cuenta que esta alineación durará un tiempo determinado debido a que no existe alineamiento en frecuencia).

En la quinta prueba, esto se soluciona exportando la señal del oscilador local del canal 1 a los demás canales. Rápidamente se observa como al utilizar exactamente el mismo oscilador la frecuencia de demodulación en cada canal será la misma y, por tanto, la fase de las señales recibidas evolucionará al mismo tiempo. Esto significa que existe un alineamiento en frecuencia.

Además, los desfases parecen fijarse en unos valores concretos entre capturas siempre y cuando no se modifiquen los parámetros de diseño del esquema utilizado. Estos desfases muestran una pequeña variación debido al error de estimación del método empleado, pero aun así se aprecia como el valor de desfase se mantiene entre capturas.

Hay que destacar que el funcionamiento de las cadenas VCO/PLL introducen un valor de $\pi/2$ o π en el desfase. Dicho esto, y sabiendo que los desfases variarán sea por las cadenas VCO/PLL o por el error de estimación, es necesario utilizar en toda captura y durante un corto periodo de tiempo un generador de señales de referencia con el fin de poder alinear la fase de las señales recibidas.

Por último, en la sexta prueba se mide el desfase que introducen las salidas del divisor de potencia y los cables SMA. Se puede concluir que los valores de desfase obtenidos entre cada salida aun teniendo en cuenta el error de estimación, están dentro de los valores típicos proporcionados por el datasheet del producto.

7.2 Futuras líneas de investigación

Este proyecto deja abiertos algunos temas que requerirían un mayor grado de estudio y que no se han podido implementar debido a las limitaciones de tiempo.

- En términos de comunicación entre el ordenador y los dispositivos USRP, interesaría buscar la manera de reducir más el tamaño en bits de las muestras obtenidas a partir de la digitalización de la señal para poder muestrear a una frecuencia más alta sin que se congestione el enlace. Además, estaría bien instalar los drivers de la tarjeta de red de 10 Gbit Ethernet para alcanzar la máxima frecuencia de muestreo de la USRP X300.
- En términos de sincronización, interesaría investigar porqué cuando se hace varias capturas de señal mediante el receptor multicanal, el desfase entre canales no es constante y da saltos de $\pi/2$ a lo largo del tiempo. La solución a este problema permitiría alinear en fase las señales recibidas sin necesidad de un generador de señales de referencia.
- A nivel de proyecto, se podría realizar la calibración de un array de antenas y la implementación de algoritmos de estimación de dirección de llegada para comprobar el funcionamiento del sistema.

Bibliography

- [Car10] A. Bruce Carlson, Paul B. Crilly, *Communication Systems: An Introduction to Signals and Noise in Electrical Communication*, McGraw-Hill, fifth ed., 2010.
- [Etta] Ettus Research, “Esquemas de la USRP X300”, <http://files.ettus.com/schematics/x300/x3xx.pdf>.
- [Ettb] Ettus Research, “SBX”, <https://kb.ettus.com/SBX>.
- [Ettc] Ettus Research, “SBX RF Performance Data”, http://files.ettus.com/performance_data/sbx/SBX-without-UHD-corrections.pdf.
- [Ett d] Ettus Research, “UHD”, <https://kb.ettus.com/UHD>.
- [Ette] Ettus Research, “USRP Hardware Driver and USRP Manual: Device Synchronization”, https://files.ettus.com/manual/page_sync.html.
- [Ettf] Ettus Research, “USRP N200/N210 Networked Series Datasheet”, http://www.ettus.com/wp-content/uploads/2019/01/07495_Ettus_N200-210_DS_Flyer_HR.pdf.
- [Ettg] Ettus Research, “TwinRX”, <https://kb.ettus.com/TwinRX>.
- [Ett h] Ettus Research, “X300/X310”, <https://kb.ettus.com/X300/X310>.
- [Fä10] Matthias Föhnle, “Software-Defined Radio with GNU Radio and USRP/2 Hardware Frontend: Setup and FM/GSM Applications”, pag. 92, 2010.
- [GNU18] GNU Radio, “Main Page”, https://wiki.gnuradio.org/index.php/Main_Page, 2018.
- [Hay96] Monson H. Hayes, *Statistical digital signal processing and modeling*, John Wiley & Sons, Inc., 1996.
- [Hay99] Simon Haykin, Barry Van Veen, *Signals and systems*, John Wiley & Sons, Inc., 1999.
- [Mina] Mini-Circuits, “Coaxial Cable 141-16smrsm+ Datasheet”, <https://ww2.minicircuits.com/pdfs/141-16SMRSM+.pdf>.
- [Minb] Mini-Circuits, “Power Splitter/Combiner: ZA4pd-2”, <https://ww2.minicircuits.com/pdfs/ZA4PD-2.pdf>.
- [Nat19a] National Instruments, “Radio Definido por Software USRP”, <https://www.ni.com/es-es/shop/select/usrp-software-defined-radio-device>, 2019.

-
- [Nat19b] National Instruments, “Software Defined Radio: Past, Present, and Future”, <http://www.ni.com/es-es/innovations/white-papers/17/software-defined-radio-past-present-and-future.html#section-715124239>, may 2019.
- [NET] NETGEAR, “Desktop Switches GS10xv3 Datasheet”, <http://www.downloads.netgear.com/files/GDC/datasheet/en/GS105v3-GS108v3.pdf>.
- [Pet12] Larry L. Peterson, Bruce S. Davie, *Computer Networks: A Systems Approach*, Elsevier, fifth ed., 2012.
- [Pro08] John G. Proakis, Masoud Salehi, *Digital communications*, McGraw-Hill, Boston, fifth ed., 2008.
- [Van88] B.D. Van Veen, K.M. Buckley, “Beamforming: a versatile approach to spatial filtering”, *IEEE ASSP Magazine*, Vol. 5, pags. 4–24, apr. 1988.
- [Wen] Wenzel Associates, Inc., “HF Citrine”, <http://www.wenzel.com/model/hfcitrine/>.
- [Wir] Wireless Innovation Forum, “Introduction to SDR”, https://www.wirelessinnovation.org/Introduction_to_SDR.