



UNIVERSIDAD DE JAÉN
Escuela Politécnica Superior de Linares

Trabajo Fin de Grado

IMPLEMENTACIÓN DE UN RECEPTOR GPS USANDO SDR

Alumno: Francisco David González Martínez

Tutores: Prof. D. Damián Martínez Muñoz
Prof. D. Antonio Manuel Herrera Olmo

Depto.: Ingeniería de Telecomunicación

Septiembre, 2021



UNIVERSIDAD DE JAÉN
Escuela Politécnica Superior de Linares

Trabajo Fin de Grado

IMPLEMENTACIÓN DE UN RECEPTOR GPS USANDO SDR

Alumno: Francisco David González Martínez

Tutores: Prof. D. Damián Martínez Muñoz
Prof. D. Antonio Manuel Herrera Olmo

Depto.: Ingeniería de Telecomunicación

Septiembre, 2021

Firma del autor:

Firma de los tutores:

ÍNDICE

LISTA DE ABREVIATURAS	8
RESUMEN.....	10
1. OBJETIVOS	11
2. INTRODUCCIÓN A LA NAVEGACIÓN GLOBAL POR SATÉLITE	12
2.1. Inicio de la navegación por satélite	12
2.2. Concepto de posicionamiento en GNSS.....	14
2.2.1. Mediciones básicas	14
2.2.2. Posicionamiento basado en la distancia	16
2.2.3. Posicionamiento basado en la pseudodistancia.....	18
2.2.4. Precisión de la solución de posición	21
2.2.5. Ecuaciones de observación GNSS	21
2.2.6. Segmentos y sistemas actuales.....	22
2.3. Señales GNSS	23
2.3.1. Bandas de frecuencia y polarización	23
2.3.2. Técnica de espectro ensanchado y códigos pseudoaleatorios	24
3. EL SISTEMA DE POSICIONAMIENTO GLOBAL (GPS)	28
3.1. Segmento espacial	28
3.1.1. Diseño y gestión de la constelación	28
3.1.2. Satélites GPS	29
3.2. Segmento de control.....	30
3.3. Señales de navegación	31
3.3.1. Señales heredadas.....	31
3.3.2. Señales modernas.....	33
3.3.3. Niveles de potencia	34
3.4. Datos de navegación	35
3.5. Servicios y precisión	36
3.6. Arquitectura de un receptor GPS.....	36

3.6.1.	Antena.....	37
3.6.2.	Front-end de radiofrecuencia.....	39
3.6.3.	Conversor analógico-digital	40
3.6.4.	Procesamiento de señal	41
4.	INTRODUCCIÓN A LA RADIO DEFINIDA POR SOFTWARE (SDR)	47
4.1.	Inicio de la radio definida por software.....	47
4.2.	Concepto	48
4.3.	Ventajas e inconvenientes	50
4.4.	Dispositivos SDR.....	51
4.4.1.	bladeRF x40	52
4.4.2.	RTL-SDR Blog V3	53
4.4.3.	HackRF One.....	54
5.	DESCRIPCIÓN DEL TRABAJO REALIZADO	56
5.1.	Herramientas software.....	56
5.1.1.	GNU Radio	56
5.1.2.	GNSS-SDR	57
5.1.3.	GNSS-SDR-Monitor.....	59
5.1.4.	MATLAB	61
5.2.	Herramientas hardware	61
5.2.1.	Antena.....	61
5.2.2.	RTL-SDR Blog V3 / bladeRF x40.....	63
5.2.3.	Ordenador de propósito general	63
5.3.	Implementación del receptor GPS	63
5.3.1.	Estructura del archivo de configuración	63
6.	RESULTADOS OBTENIDOS	70
6.1.	Posicionamiento de un receptor estático con bladeRFx40.....	70
6.2.	Posicionamiento de un receptor estático con RTL-SDR Blog V3	72
6.2.1.	Recepción de la señal GPS	72
6.2.2.	Representación de los resultados de los bloques intermedios.....	75

6.2.3.	Análisis de la precisión del receptor.....	77
6.3.	Posicionamiento de un receptor estático con Galileo.....	80
6.3.1.	Recepción de la señal Galileo	80
6.3.2.	Representación de los resultados de los bloques intermedios.....	82
6.3.3.	Análisis de las posiciones obtenidas.....	83
7.	CONCLUSIONES Y LÍNEAS FUTURAS.....	87
8.	ANEXOS	89
8.1.	Anexo I: Manual de instalación	89
8.2.	Anexo II: Manual de usuario	92
8.3.	Anexo III: Archivos de configuración.....	94
8.3.1.	Anexo III.A: Archivo de configuración para recepción de señal GPS L1 C/A con bladeRF x40.....	94
8.3.2.	Anexo III.B: Archivo de configuración para la recepción de señal GPS L1 C/A con RTL-SDR Blog V3	97
8.3.3.	Anexo III.C: Archivo de configuración para la recepción de señal Galileo E1 con RTL-SDR Blog V3.....	100
	BIBLIOGRAFÍA.....	104

ÍNDICE DE FIGURAS

Figura 2.1. Evolución de las mejoras en el posicionamiento con Transit [1].	13
Figura 2.2. Principio básico para la medición de la distancia en GNSS [1].	15
Figura 2.3. Determinación de la posición mediante la intersección de tres esferas. (a) Representación tridimensional. (b) Representación bidimensional [3].	17
Figura 2.4. Efecto de la desviación del reloj del receptor en la obtención de la posición [1].	19
Figura 2.5. Bandas de frecuencia empleadas en GNSS [1].	24
Figura 2.6. (a) Acceso múltiple por división en el tiempo (TDMA), (b) Acceso múltiple por división en frecuencia (FDMA) y (c) acceso múltiple por división de código (CDMA) [1].	25
Figura 2.7. Espectro de la señal de navegación y de la secuencia de ensanchamiento [1].	27
Figura 3.1. Constelación nominal de 24 satélites GPS en 1993 [1].	28
Figura 3.2. Segmento de control GPS [4].	30
Figura 3.3. Funcionamiento del segmento de control GPS [4].	31
Figura 3.4. Modulación de espectro ensanchado por secuencia directa [1].	32
Figura 3.5. Evolución de las señales GPS [1].	33
Figura 3.6. Estructura de los datos de navegación en GPS [1].	35
Figura 3.7. Diagrama de bloques de un receptor GPS L1 de código C/A [1].	37
Figura 3.8. Diagrama de radiación típico de una antena GPS para ondas RHCP [1].	38
Figura 3.9. Front-end de radiofrecuencia para GPS [1].	39
Figura 3.10. Esquema de un ADC [1].	40
Figura 3.11. Procesamiento de señal para un canal [5].	41
Figura 3.12. Diagrama de bloques del algoritmo PCPSA [5].	44
Figura 3.13. Gráficos de adquisición para los códigos 19 y 21 [5].	44
Figura 3.14. Esquema lógico del procesamiento de señal de un receptor GPS [7].	46
Figura 4.1. Arquitectura de un SDR ideal. a) Transmisor. b) Receptor [9].	49
Figura 4.2. Arquitectura de un SDR real.	49
Figura 4.3. bladeRF x40.	52
Figura 4.4. Arquitectura de bladeRF x40 [11].	53
Figura 4.5. RTL-SDR Blog V3.	53
Figura 4.6. Arquitectura de RTL-SDR Blog V3.	54
Figura 4.7. HackRF One.	55
Figura 5.1. Interfaz de GNU Radio.	56
Figura 5.2. Diagrama de bloques de un receptor en GNSS-SDR [15].	58

Figura 5.3. Interfaz de GNSS-SDR-Monitor [16].	60
Figura 5.4. Bloques empleados por GNSS-SDR-Monitor [16].	60
Figura 5.5. Antena GPS Mike 3A de RoHS.	61
Figura 5.6. Bias-tee.	62
Figura 5.7. Nivel de señal recibida con bladeRF x40. a) Sin bias-tee. b) Con bias-tee.	62
Figura 5.8. Ejemplo de configuración del bloque Signal Source para un dispositivo RTL-SDR.	64
Figura 5.9. Ejemplo de configuración del bloque Signal Conditioner.	65
Figura 5.10. Ejemplo de configuración del bloque de adquisición.	66
Figura 5.11. Ejemplo de configuración del bloque de tracking.	67
Figura 5.12. Ejemplo de configuración del bloque PVT.	68
Figura 5.13. Ejemplo de configuración del bloque Monitor.	69
Figura 6.1. Salida del terminal para bladeRF x40.	70
Figura 6.2. Configuración de las etapas de ganancia con bladeRF-cli.	71
Figura 6.3. Gráfico de adquisición para el código PRN 1.	72
Figura 6.4. Inicio del proceso de tracking.	73
Figura 6.5. Inicio de la de la decodificación del mensaje de navegación.	73
Figura 6.6. Recepción de los mensajes de navegación.	74
Figura 6.7 Obtención de la posición del receptor.	74
Figura 6.8. Actualización de la posición del receptor.	74
Figura 6.9. Interfaz de GNSS-SDR Monitor durante la recepción.	75
Figura 6.10. Gráficos de adquisición en el canal 0 para los códigos 15 y 16.	75
Figura 6.11. Resultados del proceso de tracking en el canal 1.	76
Figura 6.12. Representación de las posiciones obtenidas en Google Earth.	77
Figura 6.13. Inicio del proceso de tracking.	81
Figura 6.14. Inicio de la decodificación del mensaje de datos.	81
Figura 6.15. Recepción de los mensajes de datos.	81
Figura 6.16. Obtención de la primera posición fija.	81
Figura 6.17. Actualización de la posición.	82
Figura 6.18. Interfaz de GNSS-SDR-Monitor durante la ejecución del receptor.	82
Figura 6.19. Gráfico de adquisición obtenido para el PRN 1.	82
Figura 6.20. Resultados del proceso de tracking para el canal 1.	83
Figura 6.21. Representación de las posiciones obtenidas en Google Earth.	83
Figura 8.1. Ajuste de preferencias en GNSS-SDR-Monitor.	92
Figura 8.2. Interfaz de Telecommand.	93

ÍNDICE DE TABLAS

Tabla 2.1. Visión general de los sistemas de navegación por satélite mundiales y regionales.....	23
Tabla 3.1 Parámetros nominales de la constelación GPS.....	28
Tabla 3.2. Características principales de los satélites GPS.....	29
Tabla 3.3. Señales GPS.	34
Tabla 3.4. Niveles mínimos de potencia para la recepción de la señal GPS [1].	34
Tabla 3.5. Contenido esencial de las palabras TLM y HOW.	36
Tabla 4.1. Dispositivos SDR.	51
Tabla 6.1. Posición real de la antena del receptor.	77
Tabla 6.2. Coordenadas de las posiciones obtenidas con el recetor.....	78
Tabla 6.3. Error cometido en las posiciones obtenidas.	79
Tabla 6.4. Error medio.	79
Tabla 6.5. Coordenadas de las posiciones obtenidas con el recetor.....	84
Tabla 6.6. Error cometido en las posiciones obtenidas.	85
Tabla 6.7. Error medio.	86

LISTA DE ABREVIATURAS

ADC	<i>Analog-to-Digital Converter</i>
ARNS	<i>Aeronautical Radio Navigation Service</i>
ASIC	<i>Application-Specific Integrated Circuit</i>
BOC	<i>Binary Offset Carrier</i>
BPSK	<i>Binary Phase Shift Keying</i>
CDMA	<i>Code Division Multiple Acces</i>
DAC	<i>Digital-to-Analog Converter</i>
DFT	<i>Discrete Fourier Transform</i>
DLL	<i>Delay-Locked Loop</i>
DOP	<i>Dilution of Precision</i>
DS-CDMA	<i>Direct Sequence Code Division Multiple Acces</i>
DSP	<i>Digital Signal Processor</i>
FDMA	<i>Frequency Division Multiple Acces</i>
FFT	<i>Fast Fourier Transform</i>
FPGA	<i>Field Programmable Gate Array</i>
GEO	<i>Geostationary Earth Orbit</i>
GLONASS	<i>Global'naya Navigatsionnaya Sputnikova Sistema</i>
GNSS	<i>Global Navigation Satellite System</i>
GPS	<i>Global Positioning System</i>
GPS	<i>Global Positioning System Disciplined Oscillator</i>
GRARR	<i>Goddard Space Flight Center Range and Range Rate</i>
HDOP	<i>Horizontal Dilution of Precision</i>
IF	<i>Intermediate Frequency</i>
IGSO	<i>Inclined Geo-Synchronous Orbit</i>
IRNSS	<i>Indian Regional Navigation Satellite System</i>
KML	<i>Keyhole Markup Language</i>
LHCP	<i>Left-Hand Circular Polarized</i>
LNA	<i>Low-Noise Amplifier</i>
MA	<i>Multiple Acces</i>
MCS	<i>Master Control Station</i>
MEO	<i>Medium Earth Orbit</i>
NASA	<i>National Aeronautics and Space Administration</i>
PCPSA	<i>Parallel Code Phase Search Acquisition</i>
PLL	<i>Phase-Locked Loop</i>
PPS	<i>Precise Positioning Service</i>

PRN	<i>Pseudo-Random Noise</i>
QZSS	<i>Quasi-Zenith Satellite System</i>
RAM	<i>Random Acces Memory</i>
RF	<i>Radio Frequency</i>
RHCP	<i>Right-Hand Circular Polarized</i>
RINEX	<i>Receiver Independent Exchange</i>
RNSS	<i>Radio Navigation Satellite Service</i>
ROM	<i>Read-Only Memory</i>
SAW	<i>Surface Acoustic Wave</i>
SDR	<i>Software Defined Radio</i>
SNR	<i>Signal-to-Noise Ratio</i>
SPS	<i>Standard Positioning Service</i>
TDMA	<i>Time Division Multiple Acces</i>
TT&C	<i>Telemetry, Tracking and Command</i>
UDP	<i>User Datagram Protocol</i>
VDOP	<i>Vertical Dilution of Precision</i>

RESUMEN

La navegación por satélite se ha convertido en una parte integral de nuestra sociedad moderna y es empleada ampliamente en todo el mundo. Concretamente, dentro de los diferentes sistemas de navegación por satélite, el sistema de posicionamiento global (GPS) se ha convertido en un modelo para toda una familia de sistemas de navegación. Paralelamente, los conceptos de procesamiento de señales y datos de los sistemas globales de navegación por satélite han madurado continuamente, permitiendo alcanzar nuevos niveles de rendimiento en términos de precisión y tiempo.

Partiendo de estas premisas surge el presente trabajo fin de grado, que pretende realizar la implantación de un receptor GPS de señal L1 C/A empleando una plataforma de radio definida por software (SDR). La motivación para realizar la implementación de este receptor mediante la tecnología SDR es la creación de un receptor flexible de arquitectura abierta, dado que esta tecnología permite el desarrollo de receptores reconfigurables en los que es posible la selección dinámica de parámetros para módulos individuales.

1. OBJETIVOS

Este trabajo fin de grado tiene como objetivo principal la implementación de un receptor GPS de señal L1 C/A basado en un dispositivo SDR. El dispositivo SDR será el encargado de la captura y acondicionamiento de la señal de radiofrecuencia, que posteriormente será procesada en un ordenador para obtener la posición del receptor. Para la consecución de este objetivo principal es necesario satisfacer los siguientes objetivos:

- Estudio del funcionamiento del sistema GPS.
- Estudio de la tecnología SDR.
- Elección de las herramientas software y hardware adecuadas que permitan realizar la recepción de la señal y su posterior procesamiento.
- Empleo de una interfaz gráfica de usuario que permita visualizar la posición del receptor y otra información relevante.
- Realización de un análisis de la precisión obtenida en el posicionamiento con el receptor implementado.

2. INTRODUCCIÓN A LA NAVEGACIÓN GLOBAL POR SATÉLITE

2.1. Inicio de la navegación por satélite

La determinación de la posición de los puntos de la superficie terrestre mediante la observación de objetos lejanos se ha llevado a cabo desde hace cientos de años. No obstante, hasta el comienzo de la era espacial no fue posible desarrollar un sistema global de navegación por satélite (*Global Navigation Satellite System*, GNSS) de alta precisión.

Varios sistemas radio fueron desarrollados para el seguimiento de satélites y la determinación de la órbita. En Estados Unidos, estos sistemas incluían el radar, el sistema GRARR (*Goddard Space Flight Center Range and Range Rate*) y el sistema *Minitrack* de la NASA.

Sin embargo, el sistema de posicionamiento por satélite más exitoso y que se solapó con el desarrollo de GPS (*Global Positioning System*) fue *Transit*, que se convirtió en el primer sistema de posicionamiento por satélite que operó a nivel mundial. Este sistema progresó con el objetivo de rastrear el satélite *Sputnik I* de la Unión Soviética, que fue el primer satélite artificial en órbita terrestre.

A través de la medición del desplazamiento doppler de las señales radio de 20 MHz, que eran recibidas en una posición conocida, fue posible calcular la órbita del satélite. Poco después, los investigadores establecieron que, si la órbita del satélite era conocida, la posición del receptor podría determinarse mediante el desplazamiento obtenido. Esta constatación fue fundamental para el desarrollo de *Transit*, cuyo primer satélite experimental fue lanzado en 1959. Simultáneamente, la Unión Soviética desarrolló un sistema de posicionamiento llamado *Tsikada*, con una versión militar denominada *Parus*. Aunque en un comienzo *Transit* fue un sistema secreto, este estuvo disponible para los civiles en 1967 y fue ampliamente empleado para la navegación y el posicionamiento de alta precisión hasta su clausura en 1996.

Para poder obtener datos de navegación con *Transit* era necesario realizar medidas del desplazamiento doppler de la señal emitida por el satélite durante una pasada completa, la cual podría tardar hasta 18 minutos de horizonte a horizonte. Al final de esta pasada era posible determinar la latitud y la longitud del receptor. Con cinco satélites operativos, el tiempo medio entre obtención de posiciones para una latitud media era, aproximadamente, una hora. Con el paso del tiempo, a medida que las órbitas de los satélites podían ser determinadas con mayor precisión, fue posible

obtener una precisión de varias decenas de metros en la posición bidimensional (2-D) con una única pasada del satélite. Para obtener posiciones tridimensionales (3-D) con precisiones próximas a un metro, era necesario el almacenamiento de datos obtenidos mediante varias pasadas durante unos pocos días desde un punto fijo en la Tierra.

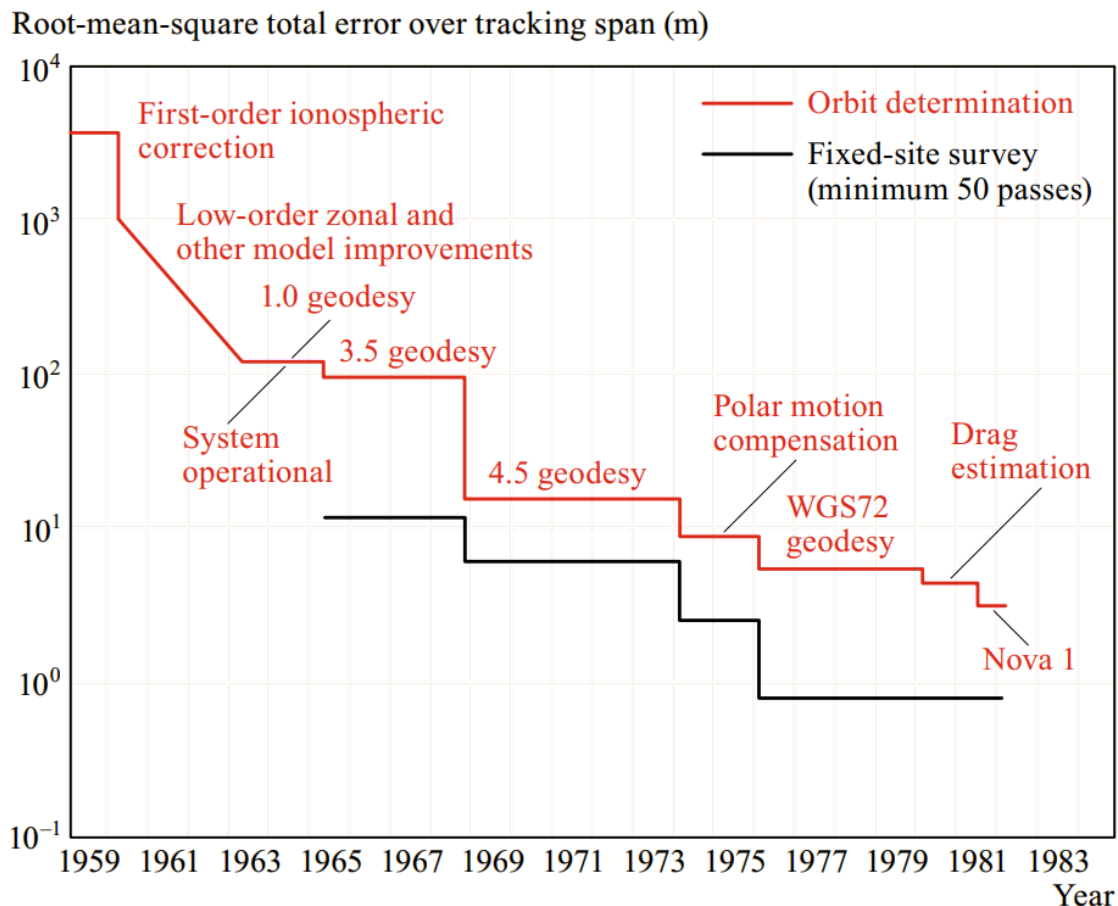


Figura 2.1. Evolución de las mejoras en el posicionamiento con Transit [1].

Los satélites *Transit* utilizaban señales de dos frecuencias distintas (150/400 MHz) con el objetivo de compensar los retrasos producidos en la ionosfera, un concepto que fue heredado por el resto de GNSSs, incluyendo GPS. Además de su uso principal como sistema de navegación, *Transit* también proporcionó una primera contribución a la geodesia y ayudó a establecer un nuevo marco de referencia global.

A finales de 1996, Transit fue retirado de servicio debido a la llegada de GPS y sus superiores prestaciones. Del mismo modo, los sistemas equivalentes rusos fueron sustituidos por el *Global'naya Navigatsionnaya Sputnikova Sistema* (GLONASS), el segundo GNSS plenamente operativo. Estos nuevos sistemas surgieron basados en el concepto de medición de distancia en lugar de las observaciones del desplazamiento doppler. Además, estos sistemas también empleaban un nuevo diseño de constelación para sus satélites que permitía ofrecer cobertura continua. Estos nuevos conceptos

permitieron una notable mejoraría en la precisión, además de un posicionamiento instantáneo en todo el mundo.

2.2. Concepto de posicionamiento en GNSS

2.2.1. Mediciones básicas

Las señales GNSS son ondas electromagnéticas que se propagan a la velocidad de la luz. Las frecuencias de estas señales se encuentran en el rango del espectro radioeléctrico comprendido entre 1.1 – 1.6 GHz (una parte de la banda L), dado que estas frecuencias permiten realizar medidas con una buena precisión, emplear equipos relativamente sencillos y evitan atenuación en la atmósfera en condiciones meteorológicas comunes [1]. Además, también se emplean señales de al menos dos frecuencias diferentes con el objetivo de compensar los retrasos ionosféricos, tal y como ocurría en *Transit*.

Una característica propia de las señales GNSS es la modulación de la onda de radiofrecuencia armónica (portadora) con un código de ruido pseudoaleatorio (*Pseudo-Random Noise*, PRN). Este código es una secuencia binaria de unos y ceros sin ningún tipo de patrón evidente que se transmite con una frecuencia típicamente comprendida entre 1 – 10 MHz, donde frecuencias más altas consiguen una mayor precisión en el posicionamiento a costa de un mayor esfuerzo en el procesamiento de la señal. El código PRN se repite continuamente en intervalos de unos pocos milisegundos a segundos, y facilita la medición del tiempo de transmisión de la señal. Además, en la mayoría de sistemas globales de navegación por satélite, este código es el único elemento que permite al receptor distinguir entre los distintos satélites individuales que transmiten a la misma frecuencia [2].

Además de la modulación realizada con este código, la portadora también se modula con un flujo de datos de navegación (mensaje de navegación) con un bajo régimen binario (por ejemplo, 50 bps en GPS) que proporciona información de la órbita del satélite transmisor y del desfase del reloj local del mismo respecto al del GNSS.

La medición básica que realiza un receptor GNSS es el tiempo necesario para que la señal GNSS se propague desde un satélite hasta el receptor, τ , que se obtiene mediante el seguimiento de la modulación del código PRN de la señal, tal y como se muestra en la Figura 2.2. Una vez recibida la señal, el receptor genera una copia local de la secuencia PRN, que se compara y alinea continuamente con la señal recibida del satélite. Este bucle de seguimiento proporciona mediciones continuas de la fase instantánea del código y, por tanto, del tiempo de transmisión correspondiente a la señal recibida en ese momento. Comparando este tiempo con el tiempo del receptor local, se

obtiene el tiempo de propagación de la señal y, multiplicado por la velocidad de la luz, la distancia entre el receptor y el satélite.

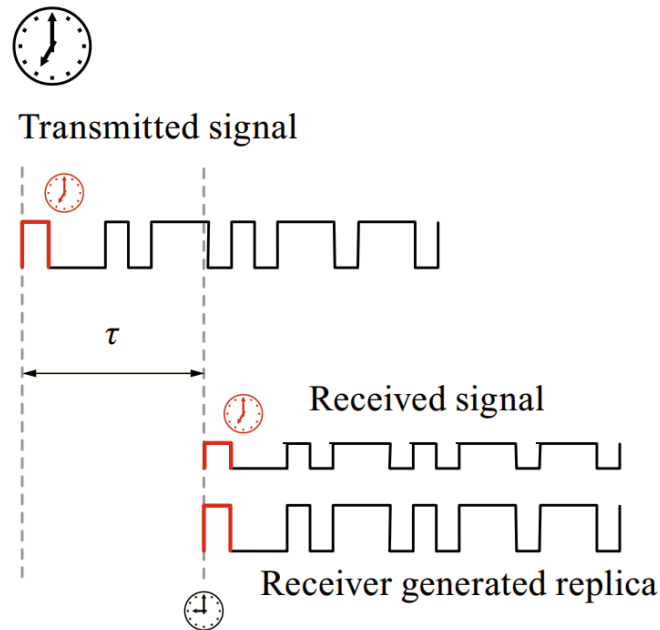


Figura 2.2. Principio básico para la medición de la distancia en GNSS [1].

Generalmente, las señales GNSS permiten tres tipos básicos de mediciones:

- Pseudodistancia (*pseudorange*). Este parámetro proporciona una medida de la diferencia entre el reloj del receptor en la recepción de la señal y el reloj del satélite en la transmisión de la señal escalada por la velocidad de la luz. Debido a la asincronía de los relojes y otros retardos producidos por la atmósfera, la medición difiere de la verdadera distancia e incluye una contribución relacionada con estos parámetros, motivo por el cual se denomina pseudodistancia.
- Fase de la portadora (*carrier phase*). Esta medición indica la fase de batido instantánea y el número acumulado de cruces por cero obtenidos tras la mezcla con una señal de referencia de la frecuencia nominal.
- Doppler. Este parámetro indica la modificación en la frecuencia recibida a causa del efecto doppler. Además, este efecto también permite medir la velocidad radial, esto es, la velocidad del receptor a lo largo de la línea visual del satélite.

La pseudodistancia, la fase de la portadora y el desplazamiento doppler proporcionan las mediciones básicas para calcular posición y velocidad, así como el desplazamiento del tiempo del receptor con respecto a la escala de tiempo del sistema GNSS.

Estas mediciones básicas se complementan con información sobre la órbita y el desfase del reloj de cada uno de los satélites GNSS, que se transmite como parte del mensaje de navegación y que permite al receptor calcular la posición y la velocidad del satélite emisor en el momento de la transmisión de la señal. Para proporcionar esta información con una precisión adecuada, el operador del GNSS debe ser capaz de determinar y predecir la órbita del satélite con antelación, de modo que pueda ser cargada en el satélite para su posterior transmisión a los usuarios. Asimismo, GNSS se basa en relojes de a bordo muy estables (relojes atómicos de cesio, rubidio e hidrogeno), cuyos desfases pueden predecirse con exactitud.

2.2.2. Posicionamiento basado en la distancia

Tal y como se ha mencionado anteriormente, la medición básica que realiza un receptor GNSS es el tiempo necesario para que la señal GNSS se propague desde la antena del satélite hasta la antena del receptor. Puesto que la señal se propaga a la velocidad de la luz, c , este intervalo de tiempo puede convertirse en una distancia multiplicándolo por c .

$$\rho_r^s(t) = c \cdot \tau_r^s(t) \quad (1)$$

Para el siguiente desarrollo se asume que los relojes del satélite y el receptor están perfectamente sincronizados y que la atmósfera (troposfera y ionosfera), que retarda ligeramente la señal, no existe. Además, también se asume que no existe ruido de medición, esto es, que no aparecen perturbaciones aleatorias en la medición, algo que afecta siempre a todas las mediciones en mayor o menor medida. Bajo estas circunstancias simplificadas, la ecuación de observación para la distancia se expresa como

$$\begin{aligned} \rho_r^s(t) &= \|r_r(t) - r^s(t - \tau)\| \\ &= \left[(x_r(t) - x^s(t - \tau))^2 + (y_r(t) - y^s(t - \tau))^2 \right. \\ &\quad \left. + (z_r(t) - z^s(t - \tau))^2 \right]^{\frac{1}{2}}, \end{aligned} \quad (2)$$

donde $r_r = (x_r, y_r, z_r)^T$ es el vector de posición de la antena receptora (posiblemente en movimiento y por lo tanto variará su valor en función del tiempo) y $r^s = (x^s, y^s, z^s)^T$ es el vector de posición de la antena del satélite transmisor que se obtiene a través del mensaje de navegación transmitido por el satélite.

Con una única medición de distancia se determinaría que la posición de receptor se encontraría en algún punto de una esfera, centrada en el satélite de radio ρ_r^1 , cuyo valor es la distancia medida. Si simultáneamente se realiza otra medición con un segundo satélite, se podría establecer que la posición de receptor se encontraría en algún punto de una esfera de radio ρ_r^2 centrada en este satélite. Estas dos esferas se cruzarían, de modo que todos los puntos de esta intersección formarían un círculo en el cual estaría situado el receptor. Una medición simultánea adicional, ρ_r^3 , proporcionaría una tercera esfera, que únicamente se cruzaría con las otras esferas en dos puntos. Uno de estos dos puntos podría descartarse para la localización del receptor, ya que se encontraría muy alejado en el espacio. Por tanto, se puede concluir que, bajo las anteriores hipótesis, únicamente se necesita la medición simultánea de la distancia con tres satélites es suficiente para determinar una posición fija tridimensional.

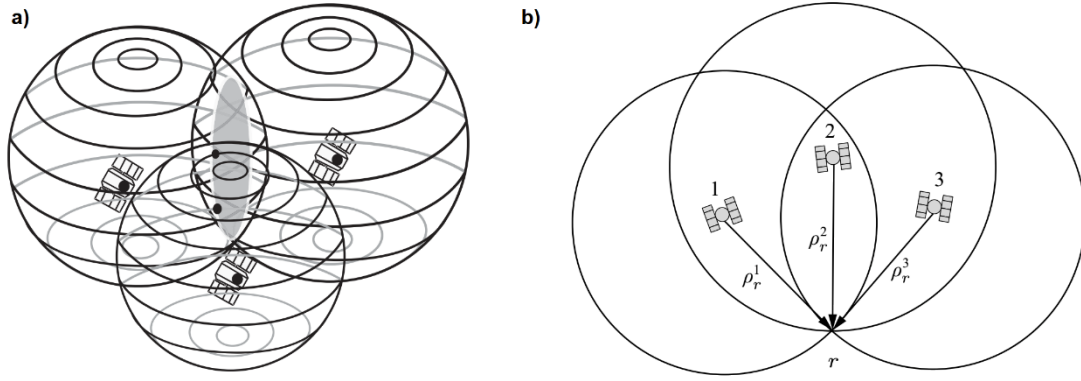


Figura 2.3. Determinación de la posición mediante la intersección de tres esferas.
(a) Representación tridimensional. (b) Representación bidimensional [3].

Computacionalmente, la posición de receptor se obtiene a partir del siguiente sistema de ecuaciones:

$$\begin{aligned}\rho_r^1 &= \sqrt{(x_r - x^1)^2 + (y_r - y^1)^2 + (z_r - z^1)^2} \\ \rho_r^2 &= \sqrt{(x_r - x^2)^2 + (y_r - y^2)^2 + (z_r - z^2)^2} \\ \rho_r^3 &= \sqrt{(x_r - x^3)^2 + (y_r - y^3)^2 + (z_r - z^3)^2}\end{aligned}\tag{3}$$

La solución de este sistema de ecuaciones $x = (x_r, y_r, z_r)^T$ se obtiene, generalmente, mediante un proceso iterativo de linealización. Expresando las distancias obtenidas como un vector $\rho = (\rho_r^1, \rho_r^2, \rho_r^3)^T$ y eliminando los índices, el sistema de ecuaciones de distancia no lineales (3) puede aproximarse al primer orden como

$$\rho = \rho_0 + A \cdot \Delta x, \quad (4)$$

donde ρ_0 es el vector de valores de distancias calculadas a partir de las coordenadas de los satélites (x^i, y^i, z^i) , $i = 1, 2, 3$, y $x_0 = (x_{r,0}, y_{r,0}, z_{r,0})^T$ es una estimación inicial de la posición del receptor. Además,

$$A = \begin{pmatrix} \frac{\partial \rho_r^1}{\partial x_r} & \frac{\partial \rho_r^1}{\partial y_r} & \frac{\partial \rho_r^1}{\partial z_r} \\ \frac{\partial \rho_r^2}{\partial x_r} & \frac{\partial \rho_r^2}{\partial y_r} & \frac{\partial \rho_r^2}{\partial z_r} \\ \frac{\partial \rho_r^3}{\partial x_r} & \frac{\partial \rho_r^3}{\partial y_r} & \frac{\partial \rho_r^3}{\partial z_r} \end{pmatrix} \quad (5)$$

con $\partial \rho_r^i / \partial x_r = (x_{r,0} - x^i) / \rho_{r,0}^i$ ($i = 1, 2, 3$), es la matriz de diseño, y $\Delta x = x - x_0$ es el incremento del vector inicial de coordenadas que debe determinarse. Nótese, que la matriz A refleja la geometría relativa de los satélites y el receptor. Despejando Δx , se obtiene

$$\Delta x = A^{-1}(\rho - \rho_0) = A^{-1}\Delta \rho, \quad (6)$$

de modo que

$$x = x_0 + \Delta x. \quad (7)$$

En función de la precisión de la aproximación de las distancias, ρ_0 , a las distancias medidas, ρ , se requiere un mayor o menor número de iteraciones para obtener un valor final de x .

2.2.3. Posicionamiento basado en la pseudodistancia

Hasta este punto se ha asumido que el reloj del receptor GNSS está perfectamente sincronizado con los relojes de los satélites, sin embargo, este supuesto es erróneo. Cuando un receptor GNSS se activa, su reloj estará generalmente desincronizado con respecto a los relojes de los satélites, en una cantidad desconocida. Además, los relojes de los satélites se sincronizan entre sí con una escala de tiempo maestra, llamada tiempo del sistema, con una precisión de, aproximadamente, un milisegundo. Las mediciones de distancia que realiza el receptor están sesgadas por los

errores en el reloj del receptor y de los satélites, dt_r y dt^s , de modo que este es el motivo por el que se denominar pseudodistancias.

$$\rho_r^s = \rho_r^s + c(dt_r - dt^s) \quad (8)$$

Sería posible sincronizar mejor los relojes del satélite mediante un envío frecuente de los comandos de ajuste desde tierra, pero se ha comprobado que estos relojes funcionan mejor si en lugar de enviar estos datos únicamente se corrigen las lecturas del reloj. Por tanto, los operadores GNSS monitorizan los relojes de los distintos satélites, determinando la desviación y la deriva respecto al tiempo del sistema. Estos parámetros se cargan posteriormente en los satélites y se transmiten como parte del mensaje de navegación [1]. Un receptor GNSS emplea esta información de los relojes de los satélites para corregir las pseudodistancias medidas.

Sin embargo, aunque sea posible compensar la desviación de los relojes de los satélites, todavía existe un error de sincronismo por parte del receptor. Debido a este error, las tres esferas con radios iguales a las pseudodistancias medidas corregidas por las desviaciones de los relojes de los satélites no se cruzarán en un punto común. No obstante, si el error en el reloj del receptor, dt_r , puede ser determinado, el valor de las pseudodistancias podrá ser corregido, de modo que será posible determinar la posición del receptor.

Por tanto, en el caso que se está abordando, existen cuatro incógnitas en la ecuación de observación de la pseudodistancia

$$\rho_r^s = \rho_r^s + c \cdot dt_r. \quad (9)$$

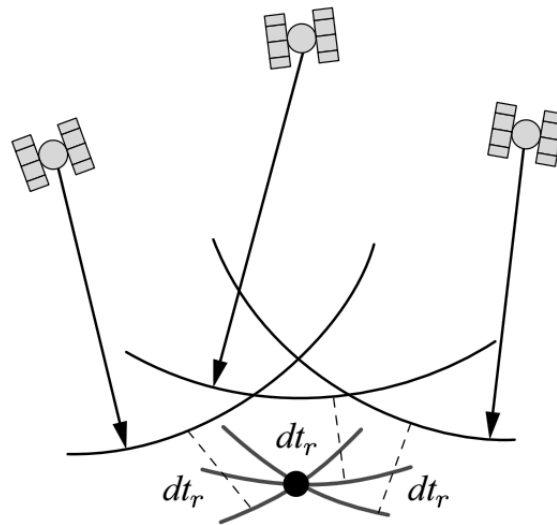


Figura 2.4. Efecto de la desviación del reloj del receptor en la obtención de la posición [1].

Estas cuatro incógnitas son las coordenadas de la posición de la antena receptora, (x_r, y_r, z_r) , y el desvío en el reloj del receptor, dt_r . Debido a este motivo se necesitan al menos cuatro pseudodistancias simultáneas (cuatro satélites) para obtener las tres coordenadas y el desvío en el reloj del receptor (expresado en unidades de distancia). Por tanto, para este supuesto, con $x = (x_r, y_r, z_r, dt_r)^T$ y la matriz de diseño

$$A = \begin{pmatrix} \frac{\partial \rho_r^1}{\partial x_r} & \frac{\partial \rho_r^1}{\partial y_r} & \frac{\partial \rho_r^1}{\partial z_r} & 1 \\ \frac{\partial \rho_r^2}{\partial x_r} & \frac{\partial \rho_r^2}{\partial y_r} & \frac{\partial \rho_r^2}{\partial z_r} & 1 \\ \frac{\partial \rho_r^3}{\partial x_r} & \frac{\partial \rho_r^3}{\partial y_r} & \frac{\partial \rho_r^3}{\partial z_r} & 1 \\ \frac{\partial \rho_r^4}{\partial x_r} & \frac{\partial \rho_r^4}{\partial y_r} & \frac{\partial \rho_r^4}{\partial z_r} & 1 \end{pmatrix} \quad (10)$$

puede aplicarse el mismo proceso iterativo descrito anteriormente.

Debido a los errores que no se consideran en el modelo (por ejemplo, los efectos atmosféricos) así como a los errores residuales en los términos modelados, es beneficioso utilizar simultáneamente todos los m satélites disponibles para estimar las coordenadas del receptor y el desfase del reloj. Esto requiere el empleo de estimación no lineal por mínimos cuadrados

$$\Delta x = (A^T W A)^{-1} A^T W \Delta \rho, \quad (11)$$

en la que A ahora tiene dimensiones de $m \times 4$ y W es una matriz de pesos que refleja la incertidumbre en las observaciones y la correlación que puede existir entre ellas. Esta matriz se puede expresar como

$$W = Q_{pp}^{-1}, \quad (12)$$

donde Q_{pp} es la matriz de covarianza de los errores de pseudodistancia. En general, la solución de un problema no lineal requiere varias iteraciones para obtener un resultado. No obstante, si el punto de linealización es lo suficiente próximo a la solución, solo se requiere una única iteración.

2.2.4. Precisión de la solución de posición

La precisión con la que pueden ser determinadas las coordenadas del receptor y el desfase del reloj depende de la matriz de covarianza de la solución Δx . Esta matriz de covarianza, denominada Q_{xx} , se obtiene, junto a (12), al aplicar la ley de propagación de errores, también conocida como ley de propagación de varianzas, a (11) y tiene la forma

$$Q_{xx} = [(A^T W A)^{-1} A^T W] Q_{pp} [(A^T W A)^{-1} A^T W]^T = (A^T Q_{pp}^{-1} A)^{-1}. \quad (13)$$

Los elementos diagonales de esta matriz son las varianzas estimadas de las de las coordenadas del receptor y del desfase del reloj, mientras que el resto de elementos (las covarianzas) indican el grado de correlación de estas estimaciones. Esta ecuación permite examinar el efecto que tendrá un diseño concreto (mediante la matriz de diseño A) y la capacidad de medición de los parámetros especificados (a través de la matriz de covarianza de medición Q_{pp}) sin realizar ninguna medición.

2.2.5. Ecuaciones de observación GNSS

Aunque el modelo descrito con las ecuaciones (8) y (9) aporta un método más óptimo para la observación de las pseudodistancias que el expuesto en (1), este aún ofrece una visión simplificada de la realidad, ya que existen varias fuentes adicionales de error que deben ser consideradas y modeladas en la ecuación de observación, por lo que la esta ecuación se puede expresar como

$$\rho_r^s = \rho_r^s + c(dt_r - dt^s) + T_r^s + I_r^s + e_r^s. \quad (14)$$

En esta ecuación, dt_r y dt^s son el desfase de los relojes del receptor y del satélite respecto al sistema de tiempo del GNSS, T_r^s es el retardo producido por la propagación en la troposfera, I_r^s indica el retardo producido por la propagación en la ionosfera, y e_r^s representa errores que no se consideran en el modelo, tales como el ruido del receptor, el multitrayecto y otros pequeños efectos.

Además de la pseudodistancia, los receptores GNSS también emplean la ecuación de observación de la fase de la portadora que es similar a la de la pseudodistancia

$$\varphi_r^s = \rho_r^s + c(dt_r - dt^s) + T_r^s - I_r^s + \lambda M_r^s + \epsilon_r^s. \quad (15)$$

Donde, además de los términos definidos anteriormente, λ es la longitud de onda de la portadora, $M_r^s = N_r^s + \delta_r - \delta^s$ es la suma de la ambigüedad de la fase de portadora N_r^s (en ciclos) y los retrasos de fase del receptor y del satélite $\delta_r - \delta^s$ y ϵ_r^s representa errores no modelados, tales como el ruido del receptor, el multitrayecto y otros pequeños efectos.

Aunque esta ecuación es similar a la de la pseudodistancia se puede observar que la magnitud del término de la ionosfera presenta signo opuesto, ya que la fase de la portadora se adelanta durante el paso de la señal por la ionosfera.

El receptor emplea estas ecuaciones de observación para obtener las coordenadas de receptor y otros parámetros relevantes. Los modelos permiten describir el retardo producido en la propagación por la troposfera, y el retardo producido por la ionosfera puede corregirse empleando los coeficientes de un modelo incluidos en el mensaje de navegación o combinado varias medidas de pseudodistancia o fase de la portadora en las dos frecuencias transmitidas.

2.2.6. Segmentos y sistemas actuales

Los sistemas de navegación por satélites (tanto globales como regionales) se componen de tres segmentos:

- El segmento espacial está compuesto por la constelación de satélites que orbitan alrededor de la superficie terrestre y transmiten señales en al menos dos frecuencias en la parte de microondas del espectro radioeléctrico.
- El segmento de control es el encargado de mantener el correcto funcionamiento del sistema mediante la monitorización de las señales emitidas y la obtención y carga de los datos de navegación necesarios en los satélites. Este segmento está formado por un grupo de estaciones de monitorización, antenas terrestres para la comunicación con los satélites y una estación maestra de control.
- El segmento de usuario se compone de los equipos de recepción de señales GNSS, tanto civiles como militares. Este segmento incluye los receptores en tierra, agua, aire e incluso en el espacio.

Las constelaciones GNSS suelen adoptar una configuración orbital específica: MEO (órbita terrestre de altitud media) para la cobertura mundial; IGSO (órbitas geosincrónicas inclinadas) y GEO (órbitas geoestacionarias) como suplementos en sistemas regionales.

Los satélites MEO suelen estar distribuidos uniformemente en órbitas inclinadas y casi circulares, espaciadas de forma excesiva en los planos orbitales, formando una constelación conocida como constelación *Walker*. La geometría de una constelación

Walker se describe mediante el triplete $t/p/f$, donde t denota el número total de satélites, p el número de planos igualmente espaciados, y f la diferencia de fase entre los planos orbitales adyacentes

En la actualidad existen un total de seis sistemas de navegación por satélite. De estos, cuatro operan a nivel global (GNSS): GPS (Estados Unidos), GLONASS (Rusia), BeiDou (China) y Galileo (Europa); y dos a nivel regional: QZSS (Japón) e IRNSS/NavIC (India).

Sistema	GPS	GLONASS	BeiDou	Galileo	QZSS	IRNSS/NavIC
Órbita	MEO	MEO	MEO, IGSO, GEO	MEO	IGSO, GEO	IGSO, GEO
Número nominal de satélites	24	24	27, 3, 5	30	3, 1	4, 3
Inicio del servicio	Diciembre 1993	Septiembre 1993	Diciembre 2012	Diciembre 2016	Octubre 2017	Abril 2018
Origen	EEUU	Rusia	China	Europa	Japón	India
Cobertura	Global	Global	Global	Global	Este asiático Oceanía	$-30^\circ < \phi < 50^\circ$ $30^\circ < \lambda < 130^\circ$
Frecuencia (MHZ)	L1 1575.42 L2 1227.60 L5 1176.45	L1 1602.00 L2 1246.00 L3 1202.025	L1 1561.098 B2 1207.14 B3 1268.52	E1 1575.42 E5a 1176.45 E5b 1207.14 E6 1278.75	L1 1575.42 L2 1227.60 L5 1176.45 E6 1278.75	L5 1176.45 S 2492.028

Tabla 2.1. Visión general de los sistemas de navegación por satélite mundiales y regionales

2.3. Señales GNSS

2.3.1. Bandas de frecuencia y polarización

La frecuencia de portadora es un aspecto crítico en las señales de navegación por satélite, ya que influye en muchos aspectos tales como el efecto doppler o los componentes hardware necesarios para la recepción. Actualmente, todos los sistemas de navegación por satélite hacen uso de una banda de frecuencia comprendida entre 1 y 2 GHz, que se denomina banda L. La banda L proporciona varias ventajas para el uso en la navegación por satélites, puesto que, para estas frecuencias, las condiciones de propagación son bastante favorables, ya que la atenuación es moderada y el impacto de los efectos atmosféricos son reducidos. Además, el tamaño de la antena para las señales de la banda L es bastante reducido, lo que permite el empleo de receptores miniaturizados y aplicaciones móviles. Otra gran ventaja del empleo de la banda L es la gran variedad de componente hardware a bajo coste disponible para estas frecuencias.

Se han establecido dos franjas diferentes del espectro, la banda L superior (1559-1610 MHz) y la banda L inferior (1164-1300 MHz), para el uso mundial de la navegación por satélite. Esta porción de espectro se denomina banda RNSS (servicio de

radionavegación por satélite). Para las aplicaciones críticas de seguridad, como las aeronáuticas, es necesaria una cierta protección de las señales de navegación contra las interferencias. Por tanto, para este tipo de servicios, se atribuye la banda ARNS (servicio de radionavegación aeronáutica) que ofrece una protección específica contra tales amenazas.

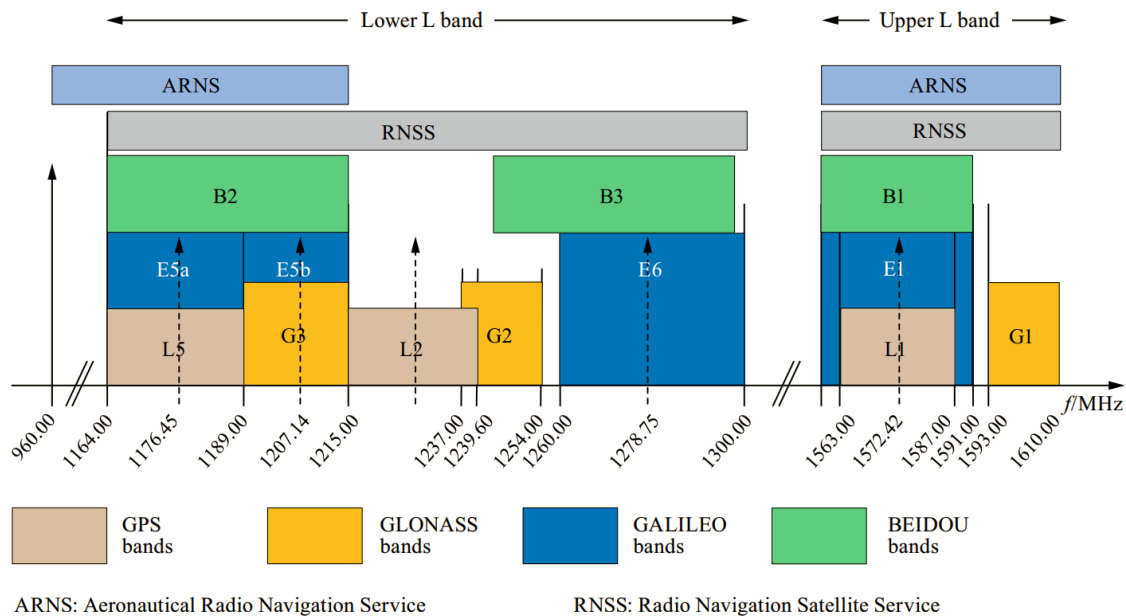


Figura 2.5. Bandas de frecuencia empleadas en GNSS [1].

Los sistemas globales de navegación por satélite emplean ondas circulares polarizadas a derechas para las señales transmitidas. Para la transmisión de señales GNSS se prefiere la polarización circular en lugar de la polarización lineal para limitar posibles pérdidas causadas por el desajuste de la orientación entre el campo electromagnético incidente y la antena receptora. La elección exclusiva de polarización a derechas (que fue primeramente adaptada en GPS) en todos los GNSSs está motivada por la intención de mejorar la interoperabilidad entre diferentes sistemas, ya que una antena receptora con polarización a derechas podría utilizarse para recibir señales de diferentes GNSSs.

2.3.2. Técnica de espectro ensanchado y códigos pseudoaleatorios

Los satélites de un GNSS deben compartir el medio de transmisión para que los usuarios puedan identificar los diferentes satélites, determinar la distancia y recibir los datos de navegación. Por tanto, los satélites deben compartir el ancho de banda disponible utilizando técnicas de acceso múltiple (MA). En general existen tres técnicas básicas de acceso múltiple:

- Acceso múltiple por división en el tiempo (TDMA).
- Acceso múltiple por división en frecuencia (FDMA).

- Acceso múltiple por división de código (CDMA).

Estas tres técnicas de acceso múltiple separan las señales en tiempo, en frecuencia o por código con el objetivo de que las señales sean divididas o incluso ortogonales. Cada una de estas técnicas puede lograr, en principio, la misma eficiencia espectral total, ya que alcanzan la misma tasa de símbolos por usuarios del canal (satélite), el mismo número de usuarios del canal (satélites) y el mismo ancho de banda total. No obstante, cada una de estas técnicas tiene diferentes ventajas y desventajas.

El principio básico de funcionamiento de TDMA, FDMA y CDMA se muestra en la Figura 2.6, donde los diferentes bloques sombreados representan los diferentes canales de usuario (satélites). En el caso de TDMA, los satélites ocupan todo el ancho de banda disponible, pero en diferentes intervalos temporales, por lo tanto, transmiten por turnos en los intervalos temporales asignados. En el caso de FDMA los satélites transmiten todos al mismo tiempo, pero ocupan diferentes divisiones dentro del ancho de banda total disponible. En el caso de CDMA, todos los satélites transmiten al mismo tiempo y utilizan todo el ancho de banda disponible, pero cada uno de ellos emplea un código distinto.

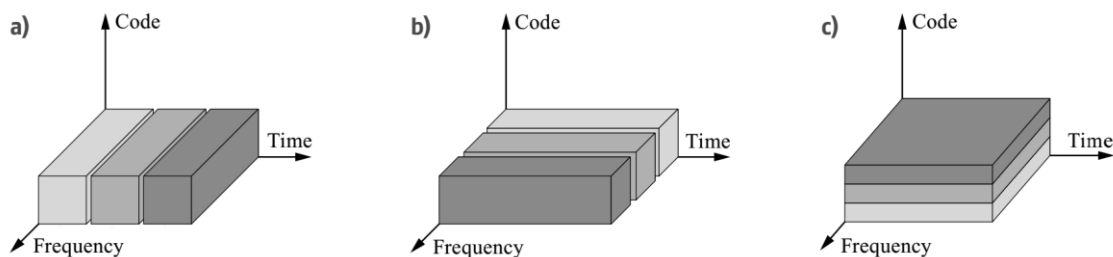


Figura 2.6. (a) Acceso múltiple por división en el tiempo (TDMA), (b) Acceso múltiple por división en frecuencia (FDMA) y (c) acceso múltiple por división de código (CDMA) [1].

Dado que para GNSS el número de satélites visibles para un receptor es bastante bajo (aproximadamente 12 como máximo) y las demandas de transmisión de datos en general no desempeñan un papel destacado, hay que tener en cuenta principalmente otras medidas de rendimiento a la hora de elegir la técnica de MA adecuada, lo que incluye, por ejemplo, la precisión en la sincronización, la eficiencia en el uso del ancho de banda o la dificultad de implementación.

Atendiendo a estos requisitos, la mayoría de GNSS emplean CDMA por secuencia directa (DS-CDMA), para el que la energía de la señal de cada satélite se distribuye de forma continua en todo el plano de tiempo-frecuencia. Los diferentes satélites transmiten sus señales en la misma banda de frecuencias al mismo tiempo, pero cada satélite utiliza un código diferente para transmitir su señal. La separación espectral entre

diferentes GNSS en la misma banda de frecuencias puede lograrse mediante el uso de diferentes esquemas de modulación.

El principio básico de esta técnica de acceso múltiple es la multiplicación del mensaje de datos de navegación por una señal de un régimen binario muy superior denominada secuencia de ensanchamiento pseudoaleatoria $c(t)$, que se puede expresar como una convolución

$$c(t) = \sum_{k=-\infty}^{\infty} d_k \sqrt{T_c} p(t - kT_c), \quad (16)$$

donde $p(t)$ denota la forma de pulso del chip (un chip se corresponde a un bit y se denomina de este modo para enfatizar que no contiene información) y T_c es el periodo de chip. Esta secuencia pseudoaleatoria puede asumirse como binaria, de media cero y cicloestacionaria en sentido amplio con $\{d_k\} \in \{-1, 1\}$ y periodo $T = N_d T_c$, $N_d \in \mathbb{N}$. La multiplicación del mensaje de datos de navegación $m(t)$ de bajo régimen binario con la secuencia pseudoaleatoria mucho más rápida $c(t)$ provoca un ensanchamiento del espectro de la señal de navegación por un factor

$$G = \frac{B}{B_m}, \quad (17)$$

donde B es el ancho de banda de la secuencia de ensanchamiento y B_m es ancho de banda del mensaje de navegación. Este factor G se denomina factor de ensanchamiento o ganancia de procesamiento, y para los sistemas DS-SS que usan chip con forma rectangular puede obtenerse como:

$$G = \frac{T}{T_c} \quad (18)$$

Una de las ventajas de los sistemas de espectro ensanchado es que proporcionan cierta capacidad de rechazo a las interferencias, puesto que las interferencias recibidas podrán ensancharse tras una correlación de la señal recibida con $c(t - \tau)$. Además, otro aspecto clave de estos sistemas para su uso en GNSS es el hecho de que cuanto mayor es el ancho de banda de la señal transmitida (el valor de G es más grande), mayor es la precisión de la sincronización.

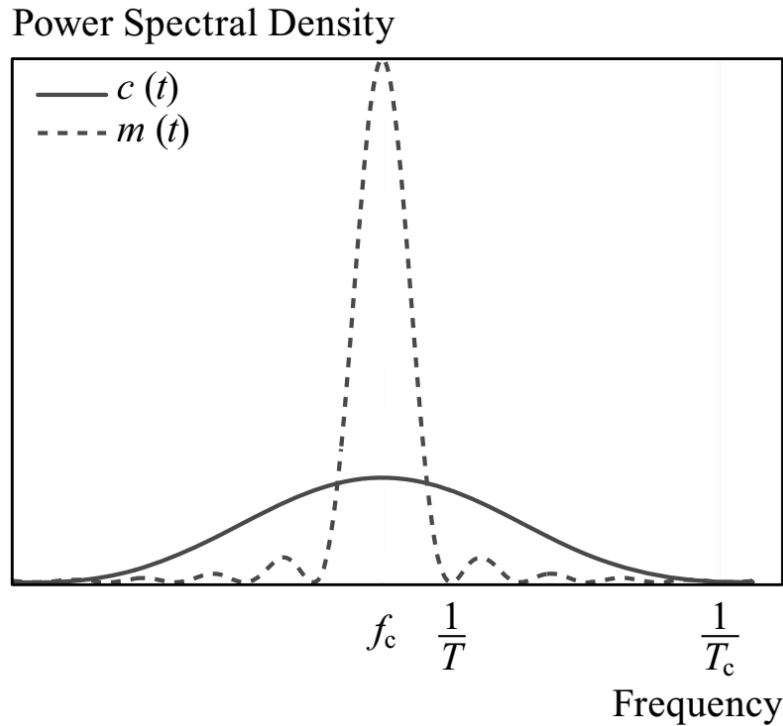


Figura 2.7. Espectro de la señal de navegación y de la secuencia de ensanchamiento [1].

Como ya se ha mencionado anteriormente, los sistemas DS-CDMA aplican secuencias de ensanchamiento o de código para separar los diferentes satélites que transmiten al mismo tiempo dentro del ancho de banda de transmisión disponible. En el caso del GNSS, estas secuencias de propagación suelen ser llamadas secuencias binarias pseudoaleatorias. Aunque estas secuencias generadas de forma determinista nunca pueden ser verdaderamente aleatorias, su diseño garantiza unas propiedades bien definidas asociadas con la aleatoriedad, de ahí que se denominan pseudoaleatorias.

Finalmente, esta señal de espectro ancho obtenida de la multiplicación del mensaje de datos de navegación y de la secuencia pseudoaleatoria debe ser modulada. Los diferentes esquemas de modulación empleados en GNSS se eligieron para obtener un buen rendimiento de la estimación del retardo temporal, además de una implementación relativamente sencilla en el satélite. Debido a estos motivos, típicamente se emplea la modulación BPSK o modulación BOC.

3. EL SISTEMA DE POSICIONAMIENTO GLOBAL (GPS)

3.1. Segmento espacial

3.1.1. Diseño y gestión de la constelación

La constelación GPS consta nominalmente de 24 satélites en órbitas circulares con un radio orbital de 26559 km. Las órbitas de los satélites están inclinadas 55° respecto al plano ecuatorial. Cuatro satélites están contenidos en cada uno de los seis planos orbitales que están igualmente espaciados con respecto a su orientación alrededor del eje de giro de la Tierra. Estos planos orbitales se identifican con letras, de la A a la F. Las 24 ubicaciones nominales de los satélites se denominan ranuras y se designan con una combinación de letra y número, por ejemplo, A1 para la primera ranura del satélite dentro del plano A.

Parámetro	Valor
Numero de satélites operacionales	24
Numero de planos orbitales	6
Numera de satélites por plano	4
Tipo de orbita	Casi circular
Excentricidad	$e < 0.02$
Inclinación	$i = 55^\circ$
Altitud nominal de la orbita	$h = 20180$ km
Periodo de revolución	$T = 11$ h 58 min

Tabla 3.1 Parámetros nominales de la constelación GPS.

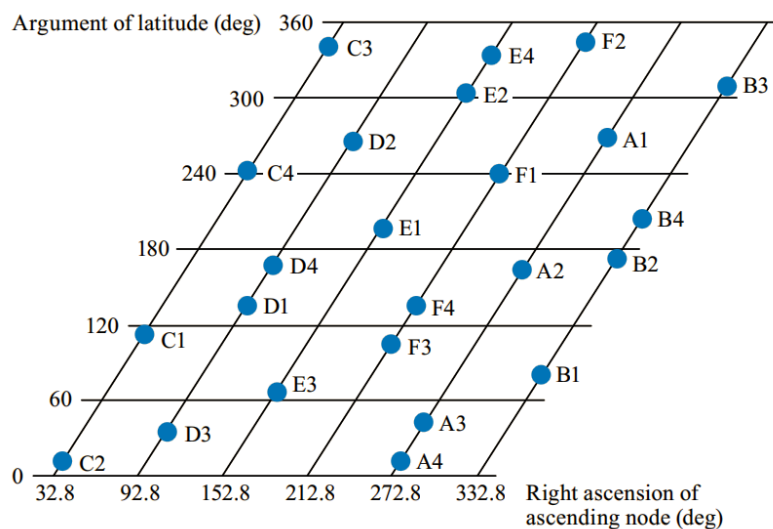


Figura 3.1. Constelación nominal de 24 satélites GPS en 1993 [1].

Observando la Figura 3.1, se puede establecer que las cuatro ranuras de cada plano orbital están espaciadas asimétricamente, puesto que este diseño proporciona robustez frente a posibles fallos en los satélites. Además, la altitud orbital fue elegida, en parte, para apoyar las primeras pruebas del sistema cuanto la constelación estaba parcialmente poblada. Esta altura orbital permite que el periodo orbital sea de media día (aproximadamente 11 h 58 min), lo que permite que la pista terrestre de los satélites se repita diariamente.

En los últimos años esta constelación se ha completado, obteniéndose 31 satélites operativos. Los 3 primeros satélites añadidos a estos 24 se colocaron en ranuras ampliables dentro de la constelación básica. El resto de satélites fueron colocados en ubicaciones adyacentes a aquellos satélites que se esperan que sean remplazados lo antes posible.

Esta constelación de satélites requiere operaciones ocasionales de mantenimiento para conservar los satélites GPS cerca de sus posiciones nominales (ranuras). Cuando un satélite GPS llega al final de su vida útil, sus señales de navegación se detienen y el satélite es elevado alrededor de 500 km en altitud hacia una órbita de eliminación.

3.1.2. Satélites GPS

Desde 1978 hasta la actualidad, 67 satélites GPS han sido puestos en órbita con éxito, de los cuales 31 se encuentran actualmente operativos. Los relojes atómicos redundantes, de rubidio y/o cesio, son componentes clave de cada satélite para que las señales se sincronicen con precisión a una escala de tiempo común. Además, las capacidades de los satélites han aumentado con el tiempo, al igual que su tamaño, peso y coste. En la Tabla 3.2 se enumeran algunas de las características principales de cada tipo de satélite.

Parámetro	Bloque I	Bloque II/IIA	Bloque IIR/IIR-M	Bloque IIF	GPS III
Satélites operativos	0	0	15	12	4
Tiempo de vida (años)	5	7.5	7.5	12	15
Primer lanzamiento	1978	1989	1997	2010	2018
Masa (kg)	450	850	1080	1630	2200
Potencia (W)	400	700	1140	2610	4480
Señales	L1, L2	L1, L2	L1, L2	L1, L2, L5	L1, L2, L5

Tabla 3.2. Características principales de los satélites GPS.

3.2. Segmento de control

Los satélites GPS son monitorizados, comandados y controlados por una red terrestre que forma el segmento de control GPS, el cual se compone de una estación maestra de control (*Master Control Station*, MCS) situada en Colorado y una red global de estaciones de monitorización y antenas terrestres.

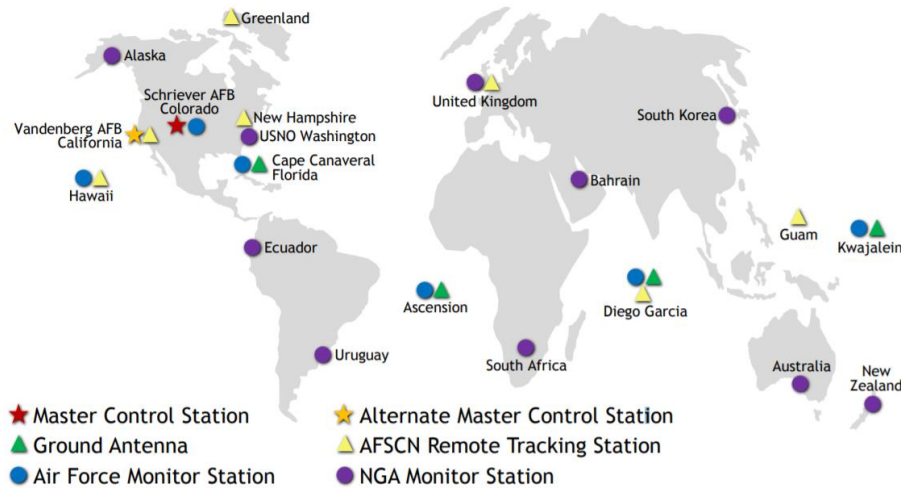


Figura 3.2. Segmento de control GPS [4].

Las estaciones de monitorización emplean receptores GPS de alta precisión que rastrean las distintas señales de navegación transmitidas por los satélites usando antenas con ganancia hemisférica (todas las direcciones por encima del horizonte). Las mediciones obtenidas se envían continuamente a la estación maestra de control.

La estación maestra de control contiene los equipos necesarios para procesar las mediciones y generar estimaciones de las posiciones, velocidades, error de los relojes y tasa de deriva de los relojes de los satélites. Esta estación se encuentra atendida en todo momento y se encarga de supervisar el estado de los satélites y gestionar las maniobras de los satélites y la carga de datos de navegación.

Las antenas terrestres de alta ganancia y altamente direccionales se emplean para leer los datos de telemetría de los satélites GPS y para proporcionar enlaces ascendentes de datos de mando y navegación. Esta función TT&C (*Telemetry, Tracking and Command*) se logra empleado un enlace ascendente con una frecuencia de 1783.74 MHz y un enlace descendente de frecuencia 2227.50 MHz. Estas antenas son relativamente grandes (aproximadamente 10 m) y deben apuntar a un único satélite GPS, por lo que se requiere una programación por parte de los operadores en la estación maestra de control.

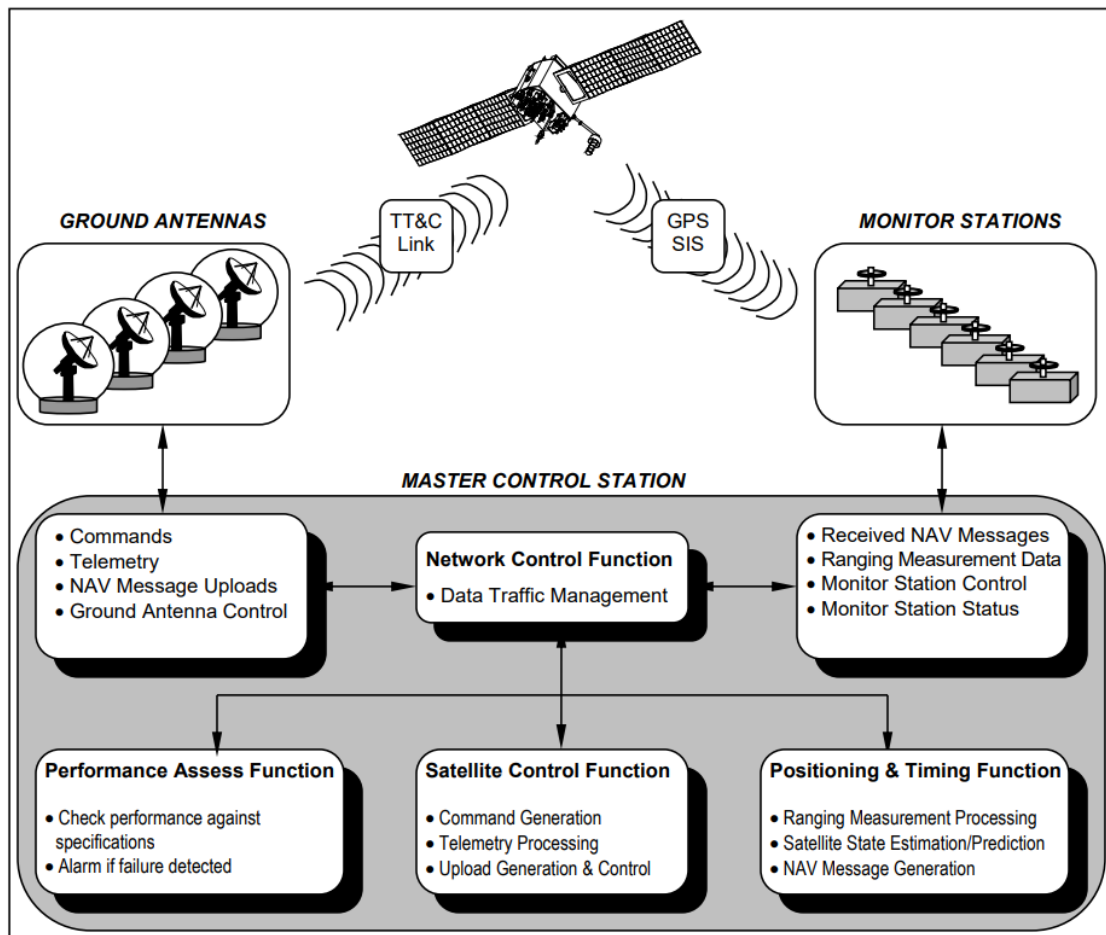


Figura 3.3. Funcionamiento del segmento de control GPS [4].

3.3. Señales de navegación

La sincronización de todos los componentes de las diferentes señales de navegación de GPS se deriva de forma coherente de un sintetizador de frecuencia a bordo impulsado por el reloj atómico activo. Este sintetizador de a bordo está diseñado para producir una frecuencia de 10.23 MHz aparente para un usuario que se encuentre en la superficie de la Tierra o cerca de ella. Un observador que se mueva junto con el satélite observaría que la frecuencia fundamental del reloj funciona a una velocidad ligeramente inferior, aproximadamente 10.229999995453 MHz, debido a los efectos combinados de relatividad especial y general.

3.3.1. Señales heredadas

Los 15 satélites más antiguos de los 31 operativos sólo emiten aquellas señales que actualmente son conocidas como señales GPS heredadas. Las señales GPS heredadas incluyen la señal de código C/A (*coarse/acquisition*) en el enlace 1 (L1), con una frecuencia de portadora de 1575.42 MHz, y la señal de código P (*precision*), tanto en L1 como en la frecuencia del enlace 2 (L2) de 1227.6 MHz. El código C/A es abierto

(sin cifrar), mientras que la señal de código P sólo está destinada a uso autorizado (militar) y normalmente está encriptada. Cuando el código P está en este modo de funcionamiento cifrado se denomina formalmente código Y. En cualquiera de los dos modos de funcionamiento, la señal se denomina comúnmente señal de código P(Y).

Ambas señales heredadas se generan empleando una modulación de espectro ensanchado por secuencia directa (modulación DSSS), para la cual se necesitan las siguientes componentes:

1. Una portadora de radiofrecuencia (RF).
2. Una señal de datos.
3. Una secuencia de ensanchamiento.

Para el código C/A o el código P(Y), la portadora de radiofrecuencia es simplemente una senoide pura de frecuencia L1 o L2. La señal de datos es una serie de pulsos binarios rectangulares contiguos de amplitud unitaria con polaridad que se transmite a 50 bps desde el satélite al usuario. La secuencia de ensanchamiento es una serie contigua de pulsos rectangulares generados mediante un código digital determinista de ruido pseudoaleatorio (PRN). El tiempo mínimo entre transiciones en la forma de onda de dispersión se denomina chip y se denota T_c , mientras que R_c es la tasa de reloj del código binario PRN binario.

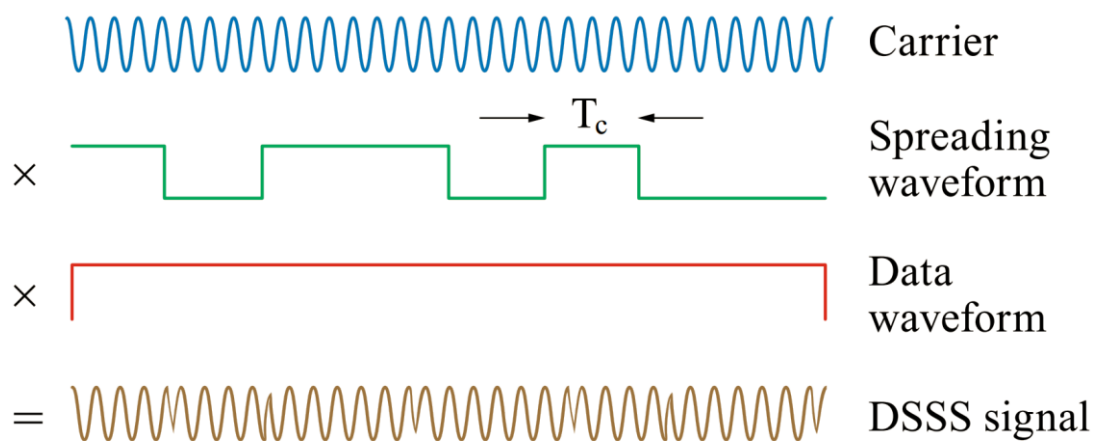


Figura 3.4. Modulación de espectro ensanchado por secuencia directa [1].

El código PRN para cada señal de código C/A se toma de la familia de códigos Gold de longitud 1023, y se genera a 1.023 MHz. Por otro lado, el código PRN para el código P(Y) se generan a 10.23 MHz utilizando un cifrado de clave privada, por lo que el algoritmo y las claves para el código P(Y) sólo están disponibles para usuarios autorizados.

3.3.2. Señales modernas

Los ocho satélites del bloque IIR-M lanzados entre 2005 y 2009 introdujeron dos nuevas señales de navegación; una señal militar en L1 y L2 denominada código M y una señal civil en L2 denominada L2C. Los satélites del bloque IIF lanzados desde 2010 introdujeron una tercera señal civil en una nueva frecuencia portadora (1176.45 MHz), ambas denominadas L5. Finalmente, los satélites GPS III introdujeron una señal civil adicional denominada L1C en la portadora L1.

En comparación con las señales heredadas, las señales GPS modernizadas tienen una serie de características de diseño avanzadas. Para todas las señales civiles modernizadas estas características incluyen la transmisión de componentes sin datos (señales piloto), códigos PRN de mayor longitud y varias mejoras en la codificación y el contenido de los datos de navegación. Además, las señales L5 y L2C emplean códigos PRN secundarios, y las señales L5 y L1C utilizan modulaciones de mayor ancho de banda.

La componente sin datos (comúnmente denominada piloto) es una porción de señal GPS que no es modulada por el mensaje de datos. El motivo por el cual esta se emplea es permitir un seguimiento más sólido de la señal por parte de un receptor en condiciones de baja relación señal-ruido (SNR).

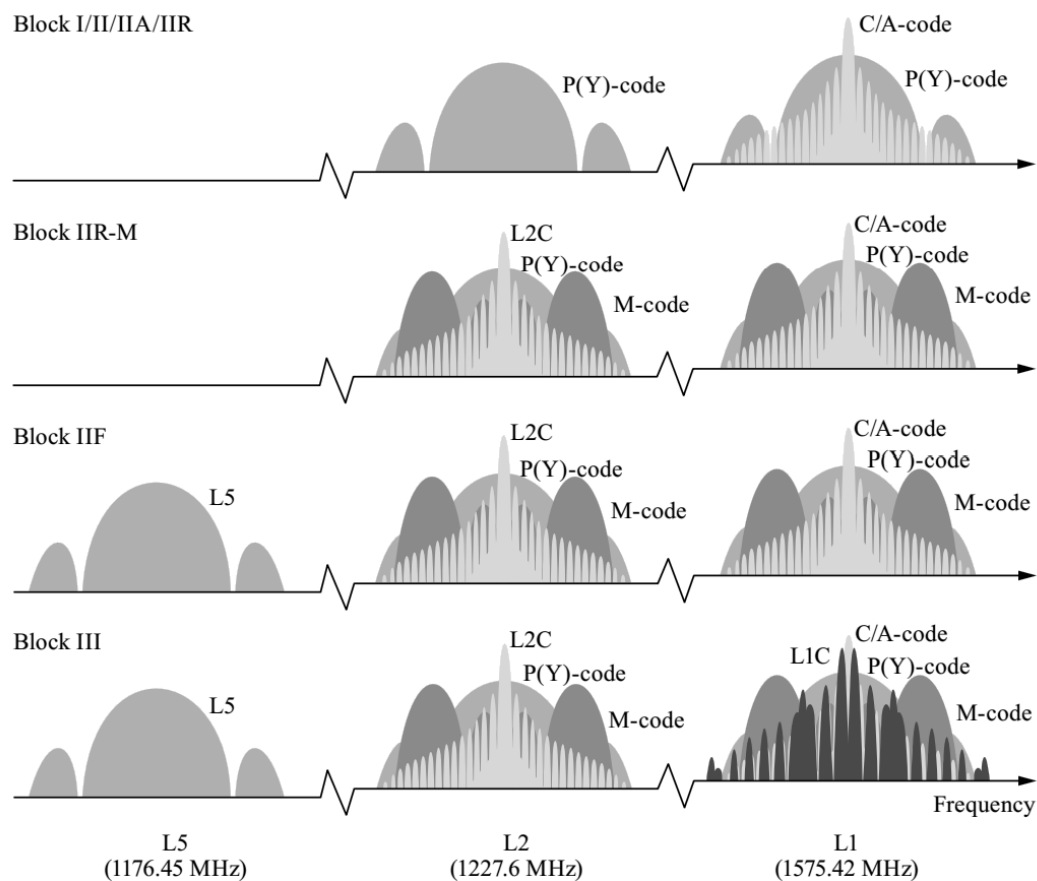


Figura 3.5. Evolución de las señales GPS [1].

Enlace	Señal	Frecuencia (MHz)	Chips del código	Frecuencia del código (MHz)	Tasa de datos (bps)	Modulación
L1	P(Y)	1575.42	N/A	10.23	50	BPSK(10)
	C/A	1575.42	1023	1.023	50	BPSK(1)
	L1C	1575.42	10230	1.023	50	TMBOC
	L1C	1575.42	10230	1.023	--	TMBOC
	M	1575.42	N/A	5.115	50	BOC _{sin}
L2	P(Y)	1227.60	N/A	10.23	50	BPSK(10)
	L2 CM	1227.60	10230	0.5115	50	BPSK(1)
	L2 CL	1227.60	767250	0.5115	50	BPSK(1)
	M	1227.60	N/A	5.115	50	BOC _{sin}
L5	I5	1176.45	10230	10.23	50	BPSK(10)
	Q5	1176.45	10230	10.23	--	BPSK(10)

Tabla 3.3. Señales GPS.

Todas las señales civiles modernizadas de GPS utilizan códigos PRN que son al menos 10 veces más largos que los utilizados para la señal civil heredada de código C/A. Estos códigos PRN de mayor longitud reducen la interferencia entre las señales cuando son procesadas por un receptor que recibe señales simultáneas de varios satélites.

3.3.3. Niveles de potencia

Los niveles de potencia mínimos especificados para las señales civiles del GPS se resumen en la Tabla 3.4. Estos niveles son aplicables para satélites con un ángulo de elevación igual o superior a 5° y suponen que el usuario está situado en la superficie de la Tierra o cerca de ella con una antena de 3 dBi de polarización lineal con la peor orientación normal.

Señal	Potencia mínima especificada en recepción (dBW)
C/A	-158.5
L2C	-160 (IIR-M/IIF), -158.5 (III)
L5	-154.9 (IIF), -154 (IIF)
L1C	-157

Tabla 3.4. Niveles mínimos de potencia para la recepción de la señal GPS [1].

3.4. Datos de navegación

Los datos de navegación se transmiten a 50 bps organizados en tramas de 1500 bits. Asimismo, estas tramas se componen de 5 subtramas de 300 bits, que comprenden diez palabras de 30 bits. De estos 30 bits, sólo 24 contienen información, ya que los 6 bits restantes se utilizan en la paridad para que el equipo del usuario pueda detectar errores de transmisión.

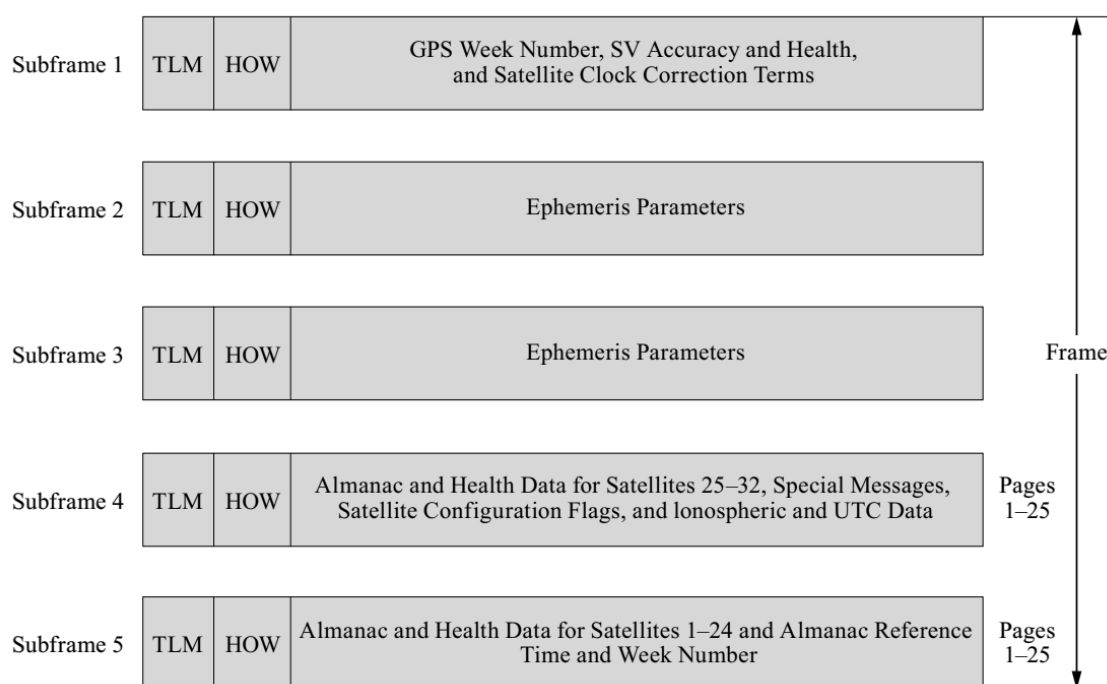


Figura 3.6. Estructura de los datos de navegación en GPS [1].

Cada subtrama comienza con una palabra de telemetría (TLM) y una palabra de traspaso (HOW) cuyo contenido esencial se resume en la Tabla 3.5. La subtrama 1 contiene datos de corrección del reloj para relacionar la hora mantenida por el reloj del satélite con la escala de tiempo común del sistema GPS. Esta subtrama también incluye información sobre el estado del satélite emisor y la precisión prevista. Las subtramas 2 y 3 proporcionan datos de efemérides, que se utilizan para determinar la posición precisa del satélite. Las subtramas 4 y 5 tienen un ciclo de 25 páginas cada una e incluyen datos de almanaque y de salud para los restantes satélites GPS, así como datos de corrección ionosférica y la diferencia horaria entre UTC y la hora del sistema GPS [5].

Palabra	Bits	Parámetro	Descripción
TLM	1-8	Preámbulo	Patrón de bits fijo para ayudar al receptor a sincronizarse con las subtramas de datos de navegación
TLM	9-22	Mensaje de telemetría	Reservado para usuarios autorizados y equipos del segmento de control
HOW	1-17	Contador de tiempo semanal	Proporciona una marca de tiempo correspondiente al inicio de la siguiente subtrama
HOW	18	Bit de alerta	Cuando está activo indica que este satélite se emplea bajo la responsabilidad del receptor
HOW	19	Bit de <i>anti-spoofing</i>	Cuando está activo indica que el código P este encriptado en el código Y
HOW	20-22	Identificación de subtrama	Indica cuál de las subtramas se transmite tras la palabra HOW.

Tabla 3.5. Contenido esencial de las palabras TLM y HOW.

3.5. Servicios y precisión

En la actualidad, GPS ofrece dos servicios: uno para los usuarios civiles, denominado servicio de posicionamiento estándar (SPS) y otro disponible sólo para usuarios autorizados (principalmente el ejército estadounidense y los ejércitos de sus aliados), denominado servicio de posicionamiento preciso (PPS).

La precisión especificada del GPS SPS es de 13 m para el posicionamiento horizontal y de 22 m para el posicionamiento vertical. Esta especificación se refiere únicamente a la señal en el espacio (es decir, no incluye los errores debidos a la atmósfera, a las trayectorias múltiples o al equipo del usuario) y se basa en una media global [4]. Los usuarios de PPS suelen tener incluso de 1 a 2 m de precisión tanto para el posicionamiento horizontal como para el vertical. Esta precisión se puede mejorar empleando técnicas diferenciales, con las que se pueden lograr precisiones superiores a 1 cm.

3.6. Arquitectura de un receptor GPS

Un receptor GPS convencional está básicamente compuesto por una parte analógica, una parte digital y las interfaces de entrada-salida. La parte analógica está

compuesta por una antena de banda L de una o varias frecuencias, un *front-end* de radiofrecuencia (que incluye el amplificador de bajo nivel de ruido, el oscilador, el convertidor descendente y los mezcladores), los filtros paso banda y el conversor analógico-digital (ADC). Asimismo, la parte digital consiste en un circuito integrado de banda base que contiene los correladores, el procesador múltiple y las unidades de memoria de sólo lectura (ROM) y de acceso aleatorio (RAM).

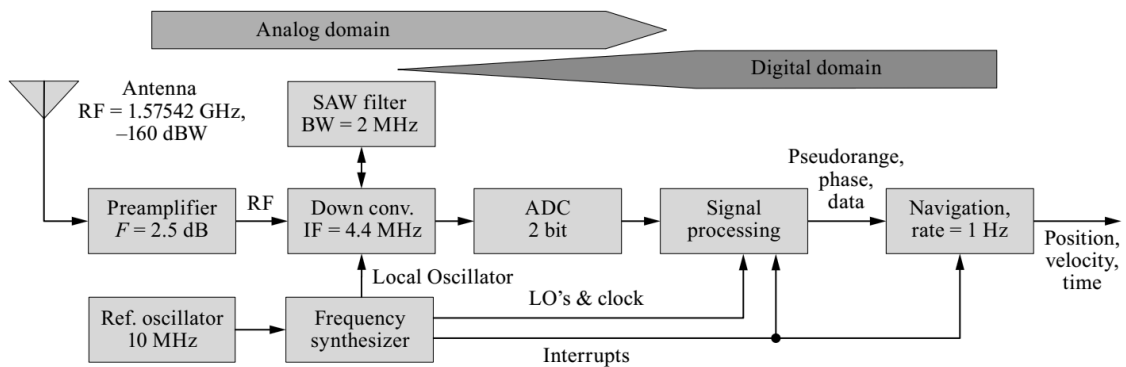


Figura 3.7. Diagrama de bloques de un receptor GPS L1 de código C/A [1].

3.6.1. Antena

La antena es el elemento encargado de recibir una onda electromagnética y convertirla en una señal eléctrica. Para el caso de un receptor GPS, existen tres parámetros fundamentales a la hora de su diseño: la frecuencia central, el ancho de banda y el diagrama de radiación.

Una antena se comporta como un filtro paso banda alrededor de la frecuencia central, por lo tanto, el ancho de banda debe ser lo suficientemente grande como para permitir el paso de las señales GPS que se desean recibir.

Adicionalmente, el objetivo del receptor GPS es trabajar con una visión completa de la constelación de satélites, de modo que todos los satélites por encima del horizonte deben ser recibidos en las frecuencias portadoras de la banda L con un ancho de banda adecuado. Para lograr este objetivo se requiere un patrón de radiación hemisférico para la recepción simultánea de la señal en la banda L, donde la ganancia máxima de la antena suele producirse para una elevación de 90°. Para elevaciones inferiores, esta ganancia se reduce suavemente hasta llegar a una elevación de 0°, donde se produce una atenuación brusca que evita la recepción para elevaciones inferiores al horizonte. Esta atenuación también permite atenuar las señales multitrayecto que se reflejan desde el suelo o la superficie de montaje de la plataforma.

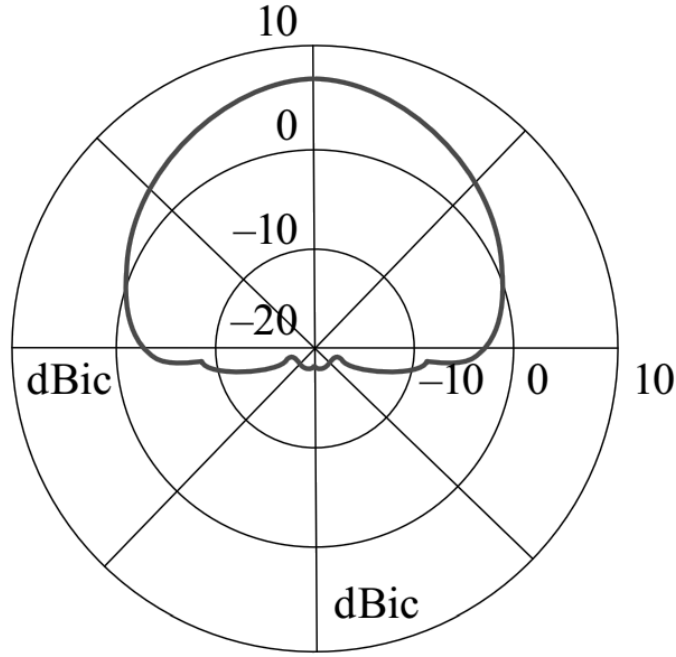


Figura 3.8. Diagrama de radiación típico de una antena GPS para ondas RHCP [1].

Además, la antena tiene que trabajar con señales polarizadas, por lo que debe proporcionar una buena ganancia para las ondas polarizadas circularmente a la derecha (RHCP) y una baja ganancia para las ondas circulares polarizadas a la izquierda (LHCP).

Finalmente, se debe realizar una diferenciación técnica entre las antenas activas y las antenas GPS pasivas. En una antena activa, el amplificador de bajo nivel de ruido (LNA) está integrado en la antena, por lo que las pérdidas del cable pueden superarse aplicando una ganancia adecuada a la salida del LNA. Esta antena necesita una alimentación remota desde el receptor a través del cable de la antena. En una antena GPS pasiva, el LNA está integrado en la unidad de RF del receptor. Normalmente, este enfoque sólo se utiliza si la distancia entre la antena y el receptor es corta (algunos cm).

Uno de los principales motivos por los que es preferible emplear una antena activa es la reducción del factor de ruido total del sistema. Este parámetro indica el ruido añadido como resultado del acondicionamiento de la señal analógica, por lo que se desea que sea lo menor posible. Denotando el factor de ruido total del sistema como F_{system} , el factor de ruido del n -ésimo componente de la cascada como F_n y la ganancia del n -ésimo componente de la cascada como G_n , el factor de ruido total del sistema se puede obtener mediante la fórmula de Friis [6]:

$$F_{\text{system}} = F_1 + \frac{F_2 - 1}{G_1} + \frac{F_3 - 1}{G_1 G_2} + \frac{F_4 - 1}{G_1 G_2 G_3} + \dots + \frac{F_n - 1}{G_1 G_2 G_3 \dots G_{n-1}} \quad (19)$$

Analizando esta ecuación se puede establecer que el primer elemento de la cascada es el que predomina en el factor de ruido del sistema resultante, siendo preferible que este sea un elemento con alta ganancia y bajo factor de ruido.

3.6.2. Front-end de radiofrecuencia

En un receptor GPS actual, la unidad de radiofrecuencia es el subsistema más crítico, porque determina el coste, el tamaño y el consumo de energía del receptor. En la Figura 3.9 se muestra el diagrama de bloques de un receptor GPS típico donde el primer subsistema necesario en la cadena para dicho *front-end* es la antena, que ya se describió en el capítulo anterior.

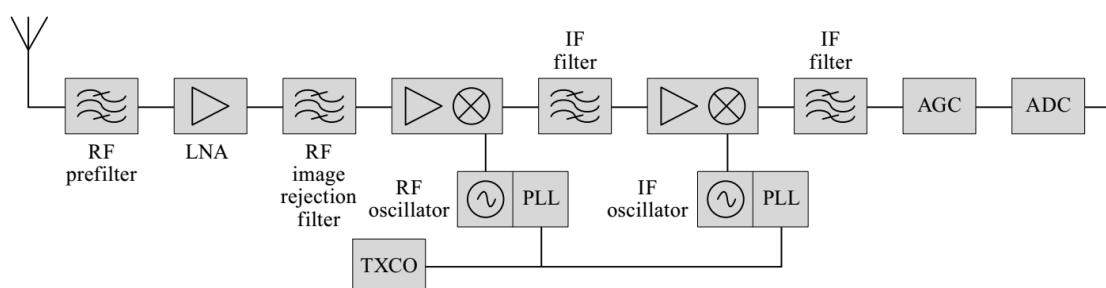


Figura 3.9. Front-end de radiofrecuencia para GPS [1].

Después de la antena, se emplea un filtro paso banda de radiofrecuencia centrado en L1, L2 o L5 que permiten eliminar las emisiones no deseadas que podrían entrar en el LNA. La selectividad necesaria de este filtro determina la tecnología y también (junto con el factor de ruido del LNA) la pérdida introducida en la banda de paso. Como se dijo en el capítulo anterior, la mayoría de las antenas se construyen como antenas activas, es decir, incluyen el LNA y el filtro de RF.

Tras este filtro se encuentra el LNA que aumenta el nivel de la señal que posteriormente se somete a otra etapa de filtrado de RF que proporciona rechazo a la frecuencia imagen (frecuencia de entrada no deseada que es capaz de producir la misma frecuencia intermedia que la señal deseada).

La combinación del mezclador y el oscilador local se emplea para convertir la frecuencia de la señal de entrada (L1, L2 o L5) a una frecuencia menor denominada frecuencia intermedia (IF), que típicamente se encuentra en el rango de 100 MHz. Para ello, la señal oscilador local, cuya frecuencia es 10 MHz, se combina con un bucle enganchado en fase (PLL) para obtener una señal de mayor frecuencia que se inserta en el mezclador, que partir de la señal de entrada y esta señal proporciona dos frecuencias a su salida: la suma y la diferencia de estas frecuencias. La razón más obvia por la que se realiza esta conversión de frecuencia es llevar la frecuencia a rangos

utilizables en los que operar con la señal, en particular para realizar la conversión analógico-digital.

Dado que a la salida del mezclador se tienen la frecuencia suma y la frecuencia diferencia, se emplea un filtro paso banda para seleccionar la frecuencia intermedia deseada (frecuencia diferencia). Seguidamente, se realiza una segunda conversión descendente de forma análoga a la explicada anteriormente que entrega la señal recibida en un rango de frecuencia de varios MHz. Finalmente, el filtro de FI posterior elimina las componentes espectrales no deseadas y también permite asegurar que el ADC cumpla el teorema de muestreo.

3.6.3. Conversor analógico-digital

El último componente de la ruta del *front-end* es el convertidor analógico-digital. Este dispositivo es responsable de la conversión de la señal analógica en muestras digitales y comprende dos funciones básicas: conversión de tiempo continuo a tiempo discreto (muestreo) y conversión de amplitud continua a amplitud discreta (cuantificación). Los parámetros clave a tener en cuenta para un conversor analógico-digital son el número de bits y la frecuencia de muestreo.

Los elementos básicos de un ADC son un oscilador local para fijar la frecuencia de muestreo, el circuito de muestreo y retención para definir el incremento de tiempo digital del receptor, un comparador de umbral para comparar la amplitud analógica (en el eje de entrada) con un número de 2^n umbrales predefinidos y un codificador para convertir el resultado de la comparación (entre los niveles de amplitud) en un determinado número de bits (n).

Puesto que la naturaleza CDMA de las señales GPS requieren muy poco rango dinámico de la señal muestreada, no es necesario emplear una gran cantidad de bits para obtener buenos resultados.

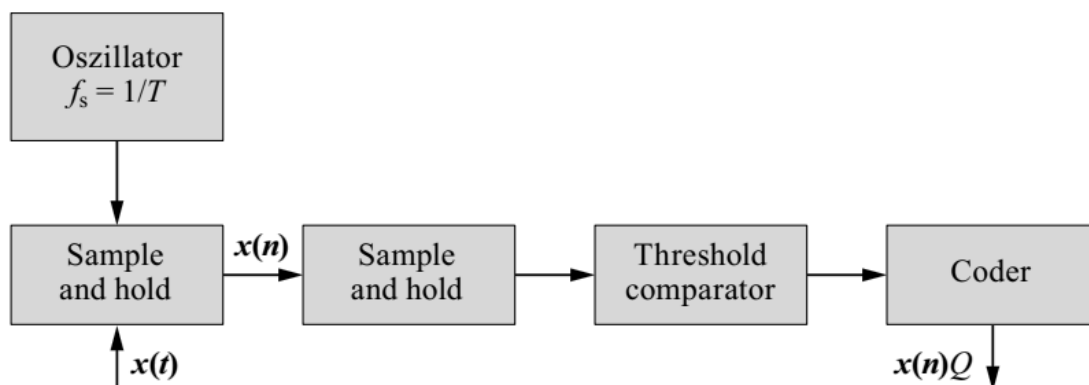


Figura 3.10. Esquema de un ADC [1].

A la hora de realizar el muestreo de la señal se debe tener en consideración el teorema de Nyquist, por el cual se demuestra que la reconstrucción exacta de una señal en banda base a partir de sus muestras, es matemáticamente posible si la señal está limitada en banda y la tasa de muestreo es superior al doble de su ancho de banda. Por tanto, a lo hará de seleccionar la frecuencia de muestreo se debe tener en encunete este requisito.

3.6.4. Procesamiento de señal

Una vez la señal de radiofrecuencia se ha convertido en una secuencia de bits se inicia el procesamiento de señal. El procesamiento de señal para los sistemas de navegación por satélite se basa en una estructura canalizada. No obstante, antes de asignar un canal a un satélite el receptor debe saber qué satélites son visibles en ese momento. Existen tres formas comunes de identificar los satélites inicialmente visibles; estas se denominan inicio en frío, inicio cálido e inicio en caliente.

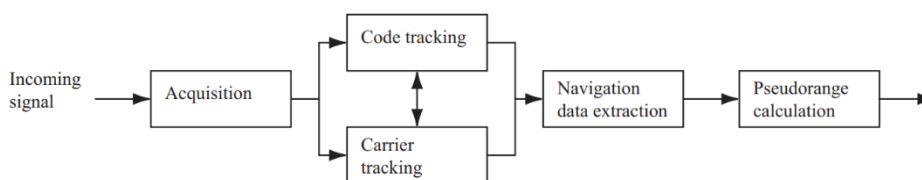


Figura 3.11. Procesamiento de señal para un canal [5].

En un inicio en caliente (*hot start*), el receptor combina la información de los datos del almanaque y efemérides almacenados y la última posición calculada por el receptor. Los datos del almanaque y de efemérides se utilizan para calcular las posiciones aproximadas de todos los satélites en el momento actual. Estas posiciones se combinan con la posición del receptor en un algoritmo que calcula qué satélites deben ser visibles. El arranque en caliente tiene dos inconvenientes principales. Si el receptor se ha alejado mucho desde la última vez que fue empleado para calcular la posición no se puede confiar en la posición del receptor y los satélites encontrados no coinciden con los satélites realmente visibles. El otro inconveniente se produce cuando los datos del almanaque y efemérides están desfasados, por lo que no pueden proporcionar buenas posiciones de los satélites. En cualquiera de los dos casos, el receptor tiene que hacer un inicio en frío.

Un inicio cálido (*warm start*), es similar a un incio en caliente en el cual no se conoce la información de efemérides. Por tanto, es necesario la obtención de esta información para determinar la posición del receptor.

En un inicio en frío (*cold start*), el receptor no se basa en ninguna información almacenada. En su lugar, empieza desde cero buscando satélites empleando un método

de búsqueda denominado adquisición que se explica posteriormente. Una búsqueda de adquisiciones a través de todos los satélites posibles requiere bastante tiempo. Este es el motivo por el que se prefiere un arranque en caliente si es posible.

Tal y como se ha dicho anteriormente, los satélites visibles se identifican mediante un proceso de adquisición. Si un satélite es visible, la adquisición debe determinar la frecuencia y la fase del código de la señal.

La frecuencia de la señal de un determinado satélite puede diferir de su valor nominal. Debido a la conversión de frecuencia realizada, la frecuencia nominal de la señal GPS se corresponde a la FI. Sin embargo, la frecuencia de la señal de radiofrecuencia recibida no se corresponde con la de una señal GPS, puesto que las señales se ven afectadas por el efecto doppler, modificando su frecuencia en un rango de ± 10 kHz para un receptor en movimiento. En caso de que el receptor se encuentre estático esta modificación de frecuencia se encontrará en el rango de ± 5 kHz. Por otro lado, la fase de código denota el momento del bloque de datos actual donde comienza el código empleado por la señal en cuestión.

La señal de entrada al proceso de adquisición contiene la información de todos los satélites visibles. Cuando se adquiere un determinado satélite, la señal entrante se multiplica por el código generado localmente correspondiente a dicho satélite. La nula correlación cruzada entre los códigos de diferentes satélites implica que las señales de otros satélites son casi eliminadas por este procedimiento. Para evitar la eliminación de la componente de la señal deseada, el código generado localmente debe estar correctamente alineado en el tiempo, es decir tener la fase de código correcta. Tras la multiplicación con el código generado localmente, la señal debe mezclarse con una portadora generada localmente para eliminar la portadora de la señal recibida. Como se ha mencionado anteriormente, la frecuencia puede variar hasta ± 10 kHz de la frecuencia nominal, por lo que deben probarse diferentes frecuencias dentro de un determinado rango.

El procedimiento de adquisición funciona como un procedimiento de búsqueda. Para cada una de las diferentes frecuencias se prueban tantas fases de código diferentes como la longitud del mismo. Cuando se han probado todas las posibilidades de fase de código y frecuencia, se busca el valor máximo. Si el valor máximo supera un umbral determinado, el satélite se adquiere con la frecuencia y el desplazamiento de fase correspondientes.

A la hora de implementar el proceso de adquisición es necesario emplear un algoritmo que permite realizar todas las combinaciones de fase de código y frecuencia del modo más rápido posible. Este es el motivo por el cual se emplea el algoritmo PCPSA (*Parallel Code Phase Search Acquisition*).

Tal y como se comentó anteriormente, el objetivo de la adquisición es realizar una correlación con la señal entrante y un código PRN. Por tanto, la forma más sencilla de realizar esta correlación es multiplicar la señal de entrada con un código PRN con 1023 fases de código diferente. No obstante, este proceso es muy lento, por lo que en lugar de emplear este método es más conveniente realizar una correlación cruzada circular entre la señal de entrada y el código PRN sin desplazar. Esta correlación circular puede obtener mediante la transformada de Fourier, tal y como se describe a continuación [6].

La transformada discreta de Fourier (DFT) de una secuencia de longitud finita $x(n)$ de longitud N se puede obtener a partir de la expresión

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-j2\pi kn/N}. \quad (20)$$

La correlación cruzada circular entre dos secuencias finitas $x(n)$ y $y(n)$, ambas de longitud N puede calcularse como

$$z(n) = \frac{1}{N} \sum_{m=0}^{N-1} x(m) y(m+n) = \frac{1}{N} \sum_{m=0}^{N-1} x(-m) y(m-n). \quad (21)$$

Omitiendo el factor de escalado $1/N$, la transformada discreta de Fourier de N puntos de $z(n)$ se puede expresar como

$$\begin{aligned} Z(k) &= \sum_{n=0}^{N-1} \sum_{m=0}^{N-1} x(-m) y(m-n) e^{-j2\pi kn/N} \\ &= \sum_{m=0}^{N-1} x(m) e^{-j2\pi km/N} \sum_{n=0}^{N-1} y(m+n) e^{-j2\pi k(m+n)/N} = X^*(k) Y(k), \end{aligned} \quad (22)$$

donde $X^*(k)$ es el complejo conjugado de $X(k)$.

La Figura 3.12 se corresponde con un diagrama de bloques del algoritmo PCPSA. Observando esta figura, se establece que la señal de entrada es multiplicada por una portadora generada localmente, la cual varía su valor en el rango comprendido entre la frecuencia IF nominal y el desplazamiento doppler máximo. Esta multiplicación da lugar a dos señales, una señal en fase (I), obtenida mediante la multiplicación de la señal de entrada y la portadora local, y una señal en cuadratura (Q), resultado de multiplicar la señal de entrada con la portadora desfasada 90° . Posteriormente, las

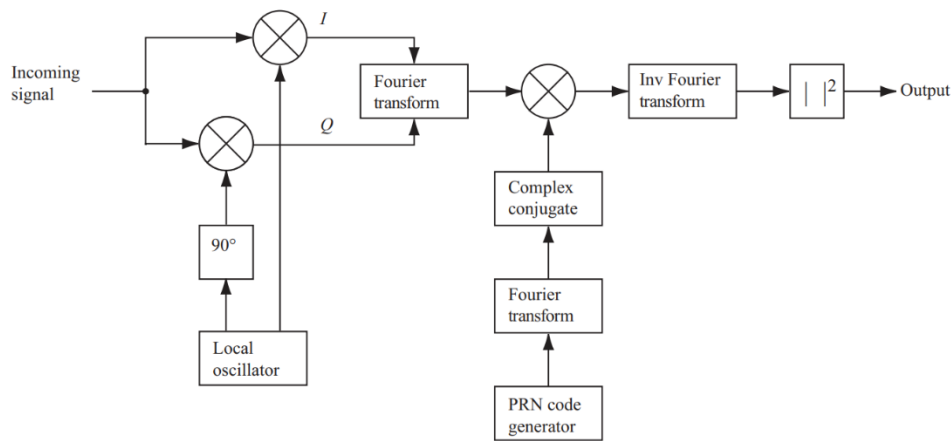


Figura 3.12. Diagrama de bloques del algoritmo PCPSA [5].

señales I y Q se combinan obteniéndose una señal compleja, $z(n) = I(n) + jQ(n)$, a la que se le aplica la DFT.

Por otro lado, el código PRN generado es transformado al dominio de la frecuencia mediante la DFT para posteriormente obtener su complejo conjugado. Esta señal se multiplica por la transformada de Fourier de la entrada. El resultado de la multiplicación se transforma al dominio del tiempo mediante una transformada de Fourier inversa, de la cual se obtiene el módulo. El módulo de la salida de la transformada inversa de Fourier representa la correlación entre la entrada y el código PRN, y su valor es comparado con un umbral para determinar si la señal recibida se corresponde con la del código generado localmente.

Finalmente, cabe destacar que esta implementación del cálculo de la correlación cruzada mediante la DFT es posible debido a la existencia del algoritmo FFT (*Fast Fourier Transform*), que permite un cálculo eficiente de la transformada de Fourier y su inversa.

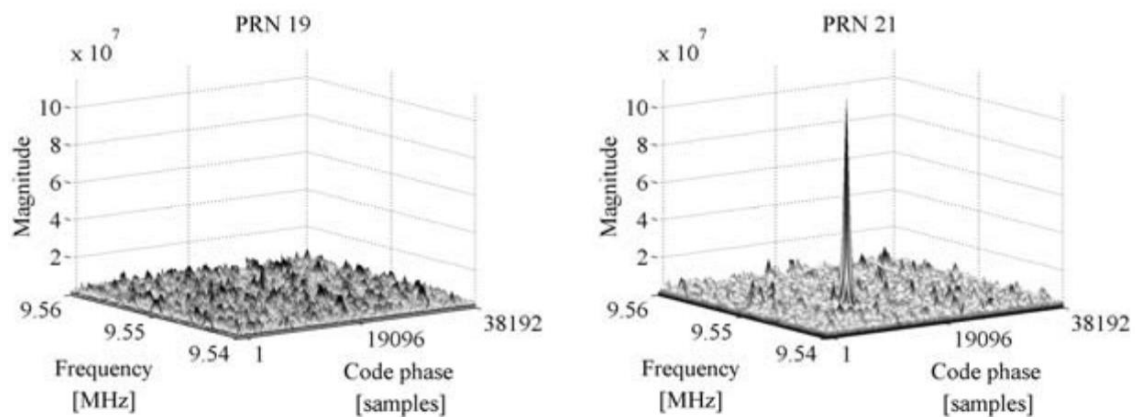


Figura 3.13. Gráficos de adquisición para los códigos 19 y 21 [5].

Observando la figura anterior se observan dos gráficos de adquisición para distintos códigos. En el caso del código 21, se muestra un gráfico de adquisición típico obtenido para un satélite visible, puesto que presenta un pico significativo que indica una alta correlación. Sin embargo, para el código 19 se muestra un gráfico de adquisición típico obtenido para un satélite que no es visible para el receptor, puesto que los valores de correlación son bajos y no hay picos significativos.

Una vez se ha identificado el satélite visible se inicia el proceso de *tracking*, cuyo objetivo principal es perfeccionar los valores de la fase y la frecuencia del código y seguirlos a medida que las propiedades de la señal cambian con el tiempo.

La precisión del valor final de la fase de código está relacionada con la precisión de la pseudodistancia calculada posteriormente. El proceso de *tracking* contiene dos partes, el seguimiento del código y el seguimiento de la portadora.

El seguimiento del código se implementa comúnmente mediante un bucle de enganche de retardo (DLL) en el que se generan tres códigos locales (réplicas) y se correlan con la señal entrante. Estas tres réplicas se denominan réplica temprana (*Early*), de entrada (*Prompt*) y tardía (*Late*) y suelen estar separadas por una longitud de medio chip respecto de la réplica alineada con la señal de entrada (*Prompt*). La otra parte del seguimiento es el de la onda portadora, el cual puede realizarse de dos formas: siguiendo la fase de la señal o siguiendo la frecuencia.

El proceso de *tracking* se realiza de forma continua para seguir los cambios de frecuencia en función del tiempo. Si el receptor pierde el rastro de un satélite, se debe realizar una nueva adquisición para ese satélite en particular.

Cuando el proceso de *tracking* se lleva a cabo correctamente, el código y la onda portadora pueden eliminarse de la señal, quedando únicamente los bits de datos de navegación. El valor de un bit de datos se obtiene integrando sobre un periodo de bit de navegación de 20 ms. Después de leer 30 s de datos (una trama), se debe identificar el principio de una subtrama para conocer el instante en que los datos fueron transmitidos desde el satélite.

En el momento que la hora de transmisión es conocida, los datos de las efemérides del satélite deben ser decodificados, puesto que estos serán utilizados posteriormente para calcular la posición del satélite en el momento de la transmisión. El último paso antes de realizar los cálculos de posición es obtener las pseudodistancias a partir de la hora de transmisión del satélite y la hora de llegada al receptor, que se encuentra en el principio de la subtrama. Una vez son conocidas las pseudodistancias se procede a calcular la posición del receptor empleando un procedimiento similar al descrito en el apartado 2.2.3 con las consideraciones adicionales expuestas en 2.2.5.

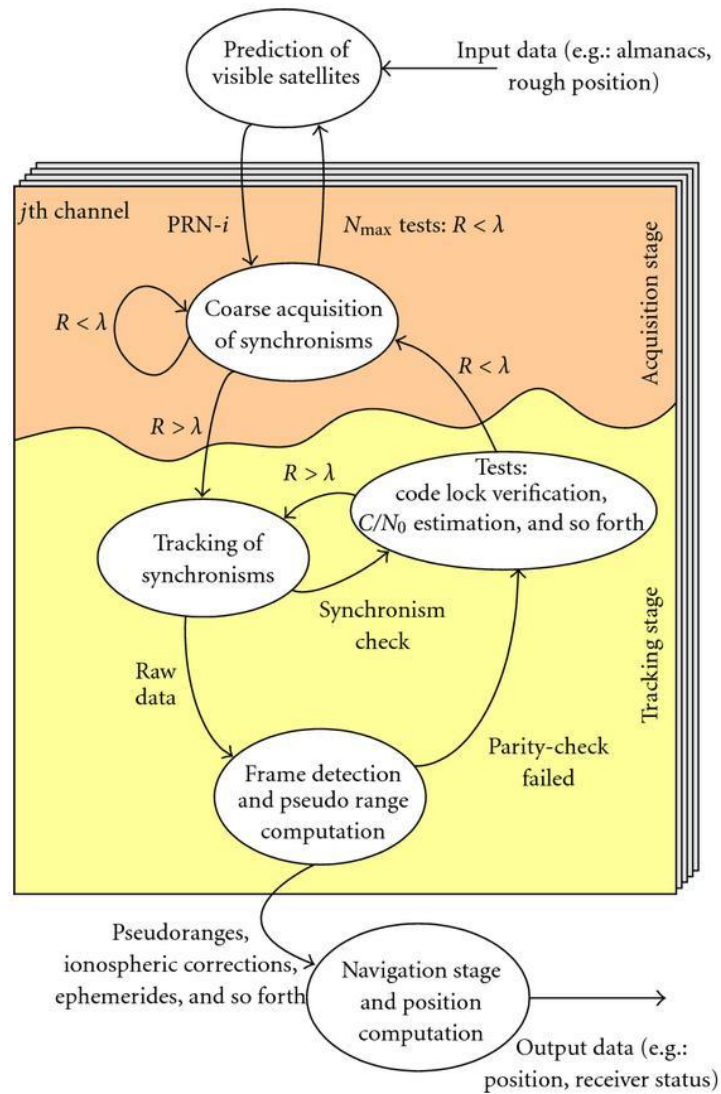


Figura 3.14. Esquema lógico del procesamiento de señal de un receptor GPS [7].

4. INTRODUCCIÓN A LA RADIO DEFINIDA POR SOFTWARE (SDR)

4.1. Inicio de la radio definida por software

Desde las primeras transmisiones inalámbricas llevadas a cabo en torno a 1890 las técnicas de transmisión por radio han evolucionado continuamente, ofreciendo a los usuarios la posibilidad de permanecer conectados con mayores velocidades de transmisión.

A pesar del crecimiento alcanzado por múltiples sistemas de radiocomunicación, una cuestión de interés y potencialmente problemática común a todos los dispositivos desarrollados hasta 1990 era que su tecnología se basaba principalmente en el hardware. Por tanto, las opciones de reprogramación o reconfiguración eran mínimas, al menos en lo que respectaba a las funciones de radio. Además, esta falta de flexibilidad provocaba que en el caso de que se produjese un fallo de hardware, firmware o software no hubiese una forma razonable de corregir el problema. De hecho, estos dispositivos estaban limitados en su funcionalidad a los componentes hardware y no podían reconfigurarse para realizar otros protocolos más allá de lo que proporciona el propio hardware. Para dar solución a estos problemas, a inicios de la década de los 90, Joseph Mitola propuso un nuevo concepto de radiocomunicación denominado radio definida por software (*Software Defined Radio*, SDR), que pretendía solucionar los problemas descritos anteriormente.

El primer SDR operativo, conocido como *SpeakEasy*, fue desarrollado por la Marina de Estados Unidos entre 1991 y 1995. No obstante, este SDR presentaba grandes limitaciones, puesto que la aplicación no podía utilizarse con otro hardware que no fuera el concebido. Además, otro aspecto negativo era el hecho de que el dispositivo ocupaba por completo la parte trasera de un vehículo de transporte. Con el paso del tiempo este SDR se desarrolló dando lugar al *SpeakEasy II*, que alcanzó un mayor éxito debido principalmente a los avances en electrónica, circuitos de comunicaciones inalámbricas y técnicas de programación reutilizables y modulares [8].

Desde el desarrollo de este primer SDR hasta la actualidad, tanto el software SDR como el hardware están disponibles a costes aceptables (de hecho, la mayoría de las implementaciones de software son gratuitas), lo que permite considerar la introducción de SDRs en las soluciones radioeléctricas.

4.2. Concepto

La radio definida por software es un paradigma de diseño de dispositivos de comunicaciones inalámbricas. Su creador, Joseph Mitola, definió el término a principios de los años 90 como un identificador de una clase de sistemas de radiocomunicaciones que podían reprogramarse y reconfigurarse mediante software. Mitola imaginó una radio definida por software ideal, cuyos componentes físicos eran sólo una antena y un convertidor analógico-digital (ADC) en el lado del receptor. Asimismo, el transmisor tendría un convertidor digital-analógico (DAC) y una antena de transmisión. El resto de las funciones serían gestionadas por procesadores reprogramables. No obstante, este tipo de sistema de radiocomunicación es imposible de implementar en la actualidad por los siguientes motivos [9]:

- La mayoría de las antenas son estructuras mecánicas y son difíciles de sintonizar dinámicamente. Un SDR ideal no debería limitar la frecuencia portadora o el ancho de banda de la forma de onda, por lo que la antena debería ser capaz de captar ondas electromagnéticas desde frecuencias muy bajas (por ejemplo, 3 MHz) hasta frecuencias muy altas (por ejemplo, 60 GHz).
- La selección de la señal deseada y el rechazo de las interferencias (selección del canal) es una característica clave del *front-end* de RF. Sin embargo, la antena y los filtros necesarios para llevar a cabo la selección del canal suelen ser estructuras electromecánicas y son difíciles de sintonizar dinámicamente.
- Sin un *front-end* de RF para seleccionar la banda de interés, toda la banda de frecuencias debe ser digitalizada. Además, siguiendo el criterio de Nyquist, la señal debe ser muestreada a una frecuencia de muestreo superior o igual a dos veces la frecuencia máxima de la señal de entrada, por lo que para el caso extremo de una señal de 60 GHz se necesitaría emplear una frecuencia de muestreo de 120 GHz, algo inverosímil para un convertidor analógico-digital actual.
- El espectro capturado contendría la señal de interés y una multitud de otras señales. Por tanto, es posible que las señales interferentes sean de mayor magnitud que la señal de interés. Para poder trabajar con estas señales de diferente magnitud, el ADC debe tener un rango dinámico muy amplio, lo cual supondría el empleo de un alto número de bits. Este hecho, combinado con una alta frecuencia de muestreo, daría lugar a un flujo de

datos de alto régimen binario que podría exceder las capacidades de los procesadores actuales.

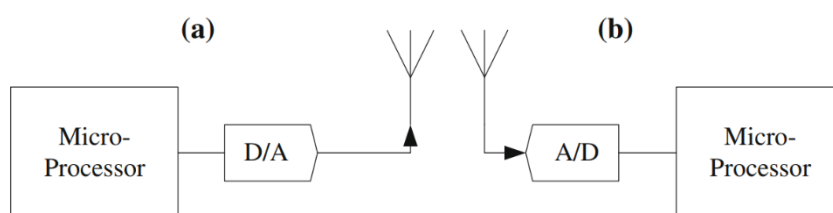


Figura 4.1. Arquitectura de un SDR ideal. a) Transmisor. b) Receptor [9].

Suponiendo que todos estos problemas técnicos fueran superados, la misma radio podría utilizarse para procesar cualquier forma de onda existente y prevista en el futuro. Sin embargo, esto no significa que esa radio sea óptima o adecuada para una aplicación determinada. El SDR ideal podría ser empleado un laboratorio de investigación, donde el tamaño físico y el consumo de energía no son un problema, pero sería completamente inapropiado para un dispositivo de mano.

Debido a que esta idea de SDR ideal no es aún realizable, y no es probable que lo sea a corto plazo, el término SDR se utiliza para describir un dispositivo de radiocomunicaciones viable en el que algunas funciones hardware están definidas por software.

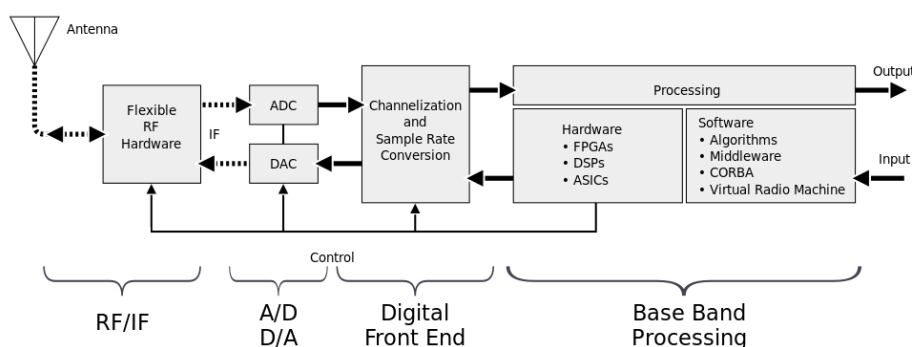


Figura 4.2. Arquitectura de un SDR real.

Esta arquitectura de un SDR real se compone de un bloque de radiofrecuencia configurable por software encargado del filtrado y conversión de frecuencia de la señal, la cual se digitaliza mediante un ADC. Posteriormente, se encuentra un front-end digital encargado de realizar la conversión de IF a banda base. Finalmente, las muestras digitalizadas en banda base son procesadas por un DSP (*Digital Signal Processor*). Cabe destacar que las operaciones descritas anteriormente por los distintos bloques del SDR son las que se llevarían a cabo si el sistema funciona como un receptor. En caso

de que este se emplease como transmisor, el funcionamiento del sistema sería inverso al descrito anteriormente [10].

Finalmente, es conveniente señalar que la etapa de procesamiento digital puede ser implementada mediante un ordenador de propósito general, ya que permite una mayor versatilidad al poder emplear distintas herramientas software, como pueden ser MATLAB o GNU Radio.

4.3. Ventajas e inconvenientes

La radio definida por software es una tecnología relativamente moderna que presenta como grandes ventajas la flexibilidad y su capacidad de reprogramación. No obstante, esta tecnología presenta otras ventajas adicionales derivadas, en mayor o menor medida, de estas ventajas principales [9].

La interoperabilidad fue una de las principales razones que provocó el interés y la financiación en la tecnología SDR por parte del ejército estadounidense, puesto que las diferentes ramas del ejército y de las fuerzas del orden emplean sistemas incompatibles, lo que dificulta la comunicación durante las operaciones conjuntas. Un SDR puede comunicarse sin problemas con sistemas de radiocomunicación incompatibles o actuar como puente entre ellos.

Otro aspecto de interés de los dispositivos SDR es el uso eficiente de los recursos en condiciones variables. De este modo, un dispositivo SDR puede adaptar la forma de onda adecuada para maximizar una métrica clave, ofreciendo una mejor experiencia al usuario.

Los sistemas SDR permiten la reutilización oportunista de frecuencias, pudiéndose aprovechar el espectro infrautilizado. Esto es posible debido a que los dispositivos SDR pueden tomar prestado una porción del espectro que en el momento de la comunicación se encuentre en desuso. Esta técnica tiene el potencial para aumentar drásticamente la cantidad de espectro disponible.

Además, esta tecnología proporciona una reducción de la obsolescencia, ya que un dispositivo SDR puede actualizarse para soportar los últimos estándares de comunicación. Esta capacidad es especialmente importante para los radios con ciclos de vida largos, como los de las aplicaciones militares y aeroespaciales.

Adicionalmente, la radio definida por software facilita una plataforma excepcional para la investigación y el desarrollo. De este modo, un dispositivo SDR puede ser empleado para implementar diferentes formas de onda para el análisis del rendimiento en tiempo real. Además, también permite realizar grandes estudios del espacio

comercial mucho más rápidos (y a menudo con mayor fidelidad) que a través de simulaciones

Tal y como se ha expuesto anteriormente, la radio definida por software proporciona una gran cantidad de ventajas derivadas de su flexibilidad. No obstante, esta tecnología también presenta ciertos inconvenientes que evitan que su uso esté más extendido.

El argumento más común contra la tecnología SDR es el coste. Este argumento es especialmente importante para sistemas relativamente simples que son ampliamente usados, ya que para estos casos es más adecuado el empleo de un circuito integrado de aplicación específica (ASIC). Otro aspecto negativo es el mayor consumo de potencia provocado por la mayor complejidad del DSP y el mayor ancho de banda de la señal.

Además de los inconvenientes comentados anteriormente, la radio definida por software requiere una complejidad adicional, puesto que es necesario un mayor esfuerzo de investigación para desarrollar sus componentes.

4.4. Dispositivos SDR

A día de hoy es posible encontrar una gran variedad de dispositivos SDR con diferentes características y prestaciones, lo que provoca que exista cierta variabilidad en el precio de estos dispositivos. En la siguiente tabla se ha elaborado una breve descripción de los principales dispositivos SDR, destacando algunas de sus características más reseñables.

Dispositivo	Rango de frecuencia	Frecuencia de muestreo	Ancho de banda en recepción	Estabilidad de frecuencia	TX/RX	Precio
Airspy HF+	0.5 KHz – 31 MHz 60 MHz – 260 MHz	36 MHz	768 kHz	1 ppb	No/Sí	~200 €
bladeRF	300 MHz – 3.8 GHz	40 MHz	28 MHz	1 ppm	Sí/Sí	350 € – 450 €
bladeRF 2.0	47 MHz – 6 GHz	61.44 MHz	56 MHz	1 ppm	Sí/Sí	400 € - 600 €
HackRF One	1 MHz – 6 GHz	20 MHz	20 MHz	20 ppm	Sí/Sí	~280 €
RTL-SDR Blog V3	500 kHz – 1.766 GHz	3.2 MHz	2.4 MHz	1 ppm	No/Sí	~28 €
USRP N200	DC – 6 GHz	100 MHz	25 MHz	0.01 ppm	Sí/Sí	~2000 €

Tabla 4.1. Dispositivos SDR.

A la hora de seleccionar un SDR para la recepción de señales GPS de frecuencia L1 (1575.42 MHz. Además, otro requisito esencial para la recepción de señales GPS es la estabilidad en frecuencia del oscilador del dispositivo, ya que se requiere una estabilidad de frecuencia menor o igual a 1 ppm (parte por millón). Por tanto, atendiendo

a estos requisitos, y a la disponibilidad de los dispositivos, se han empleado las plataformas bladeRF x40 y RTL-SDR Blog V3.

4.4.1. bladeRF x40

bladeRFx40 es una plataforma SDR desarrollada por *Nuand* que permite la exploración de los sistemas de radiocomunicación en ámbitos académicos y profesionales. Este dispositivo contiene un puerto USB 3.0 de alta velocidad que, además de proporcionar alimentación al sistema, permite el intercambio de datos con un ordenador.

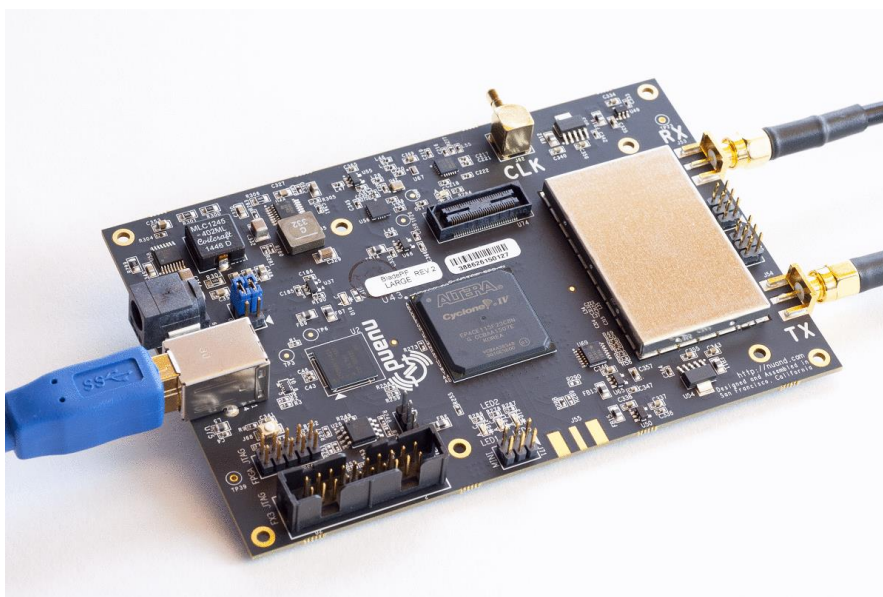


Figura 4.3. bladeRF x40.

bladeRF x40 se compone de un canal de recepción y otro de transmisión. Para ello emplea el transceptor LMS6002D fabricado por *Lime Microsystems* que permite la recepción y transmisión de señales de frecuencias comprendidas entre 300 MHz – 3.8 GHz, mientras que el ancho de banda del filtro de radiofrecuencia se encuentra entre 1.5 MHz – 28 MHz.

Además, emplea un ADC/DAC que tiene una resolución de 12 bit y una frecuencia de muestreo máxima de 40 MHz. Adicionalmente, consta de un bloque de control y procesamiento compuesto por la FPGA Cyclone IV E de *Altera*.

Asimismo, también cuenta con un bloque de temporización conformado por el generador de reloj programable Si5338 de *Silicon Labs*, un DAC y un oscilador de cristal con una estabilidad de frecuencia de 1 ppm.

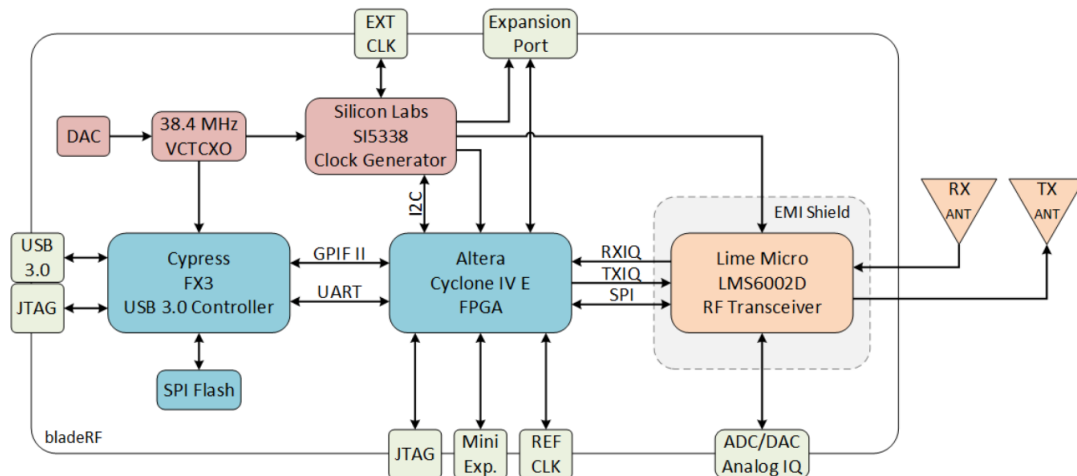


Figura 4.4. Arquitectura de bladeRF x40 [11].

4.4.2. RTL-SDR Blog V3

RTL-SDR Blog V3 es un receptor SDR compuesto por el chipset R820T2 V3 que permite la recepción de señales de radiofrecuencia comprendida entre 500 KHz - 1.766 GHz.



Figura 4.5. RTL-SDR Blog V3.

Este dispositivo presenta las siguientes características [12]:

- ADC con 8 bit de resolución.
- Recepción completa para la banda de frecuencia de 500 KHz 1.766 GHz.
- Ancho de banda de 2.4 MHz.
- Conector de antena SMA.

- Conector USB.
- Oscilador de cristal de alta precisión (1 ppm).
- *Bias-tee* configurable mediante software.
- Frecuencia máxima de muestreo de 3.2 MHz.

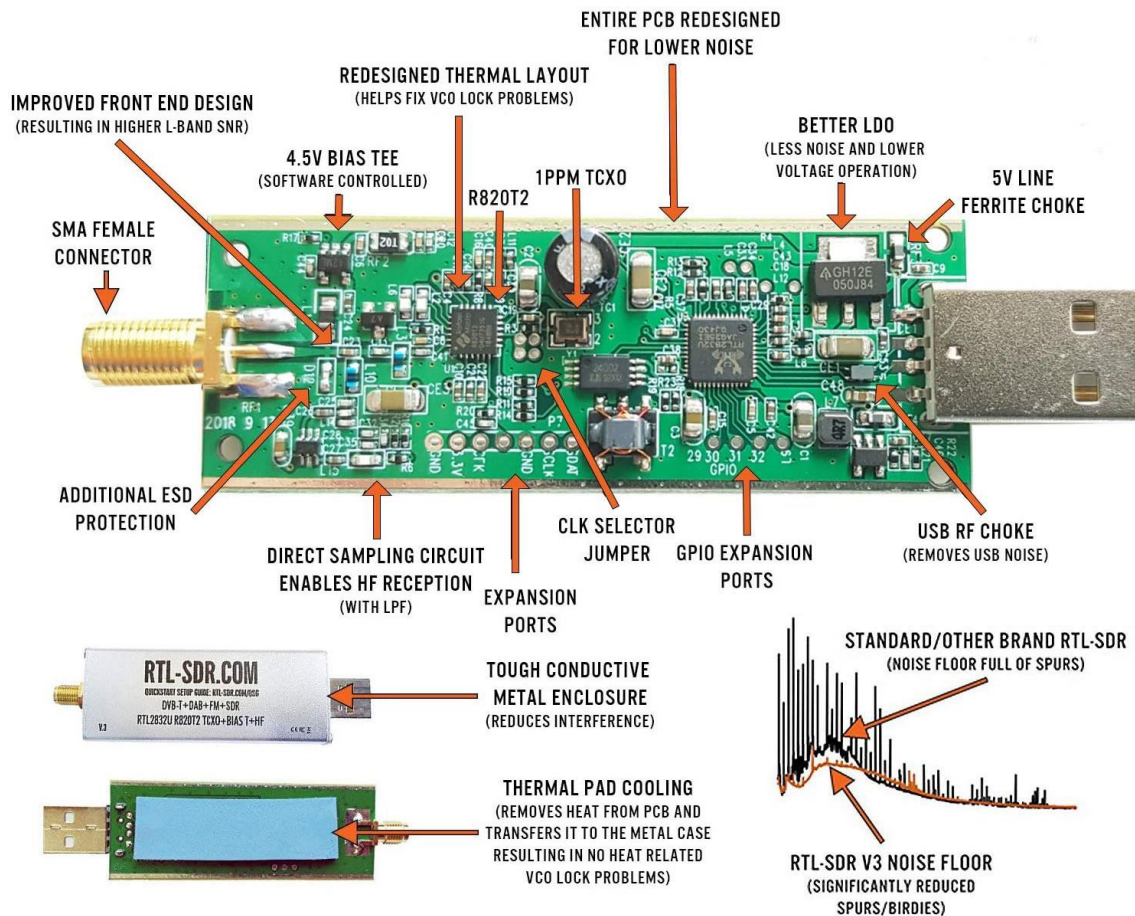


Figura 4.6. Arquitectura de RTL-SDR Blog V3.

Finalmente, cabe destacar para una frecuencia de muestreo de 3.2 MHz, el dispositivo RTL-SDR Blog V3 se comporta de forma inestable, pudiendo ocasionar una pérdida muestras. Para evitar esta pérdida de muestras se debe emplear una frecuencia máxima de muestreo de 2.56 MHz. No obstante, es posible que este valor de frecuencia de muestreo sea ligeramente superior si se emplea una conexión USB 3.0.

4.4.3. HackRF One

HackRF One es un periférico de radio definida por software fabricado por *Great Scott Gadgets* capaz de transmitir o recibir señales de radiofrecuencia comprendidas en el rango de 1 MHz - 6 GHz.



Figura 4.7. HackRF One.

A continuación, se detallarán las características principales del dispositivo, junto con algunas de las capacidades que le confieren una gran potencia y flexibilidad [13]:

- Rango de operación en frecuencia de 1 MHz - 6 GHz.
- Frecuencia de muestreo máxima de 20 MHz
- Oscilador de cristal con estabilidad en frecuencia de 20 ppm.
- Posibilidad de comunicación *half-duplex*.
- Muestreo de las señales con 8 bits en cuadratura (8 bits serán para la componente en fase, *I*, y 8 para la componente en cuadratura, *Q*).
- Conector de antena SMA hembra.
- Conector USB 2.0 de alta velocidad.

5. DESCRIPCIÓN DEL TRABAJO REALIZADO

Antes de realizar la descripción del trabajo realizado es necesario conocer las distintas herramientas hardware y software necesarias para su desarrollo. Por tanto, en primer lugar se describirán las herramientas empleadas para posteriormente indicar el trabajo realizado.

5.1. Herramientas software

5.1.1. GNU Radio

GNU Radio es un conjunto de herramientas de desarrollo de software libre que proporciona los bloques de procesamiento necesarios para la implementación de radios definidas por software empleando hardware de RF externo y procesadores básicos fácilmente disponibles. Esta herramienta es ampliamente usada en entornos de aficionados, académicos y comerciales para apoyar la investigación de las comunicaciones inalámbricas, así como para implementar sistemas radio del mundo real. Las aplicaciones de GNU Radio se implementan principalmente mediante el lenguaje de programación Python, mientras que los algoritmos críticos de bajo nivel algo y los módulos de procesamiento de señales se implementan en C++. De este modo, el desarrollador es capaz de implementar sistemas de radio de alto rendimiento en tiempo real en un entorno de desarrollo rápido y sencillo. Aunque no es principalmente una herramienta de simulación, GNU Radio soporta el desarrollo de algoritmos de procesamiento de señales utilizando datos pregrabados o generados, evitando la necesidad de hardware de RF real.

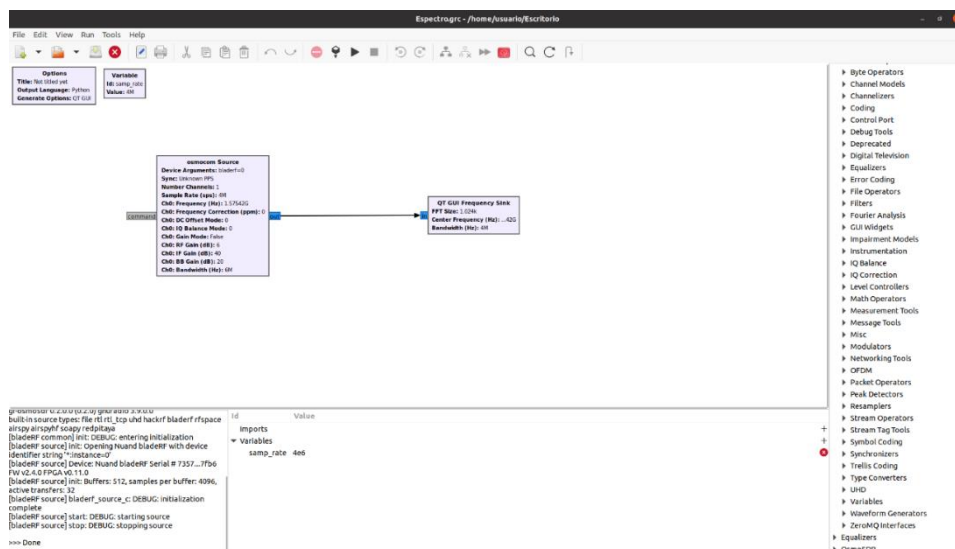


Figura 5.1. Interfaz de GNU Radio.

El proyecto GNU Radio fue iniciado hace unos veinte años por Eric Blossom, debido a las bajas prestaciones de los proyectos SDR de la época. La idea principal que está detrás de este proyecto es convertir todos los problemas de hardware en problemas de software, es decir, trasladar la complejidad de un equipo de radio del nivel de hardware al de software, acercando el software a la antena tanto como sea posible.

Empleando la interfaz de la figura anterior el usuario puede construir un receptor creando un gráfico donde los nodos son bloques de procesamiento de señal y las líneas representan el flujo de datos entre ellos. Conceptualmente, los bloques procesan infinitos datos que fluyen de sus puertos de entrada a sus puertos de salida. Estos bloques se configuran mediante una serie de atributos que incluyen el número de puertos de entrada y de salida, así como el tipo de datos que se procesan. Una vez conectados formando un diagrama de flujo, la aplicación puede ejecutarse y los datos se introducirán en el flujo.

5.1.2. GNSS-SDR

GNSS-SDR es un software libre de código abierto que proporciona una interfaz para diferentes *front-ends* de RF e implementa toda la cadena de un receptor GNSS (adquisición de la señal, seguimiento de las señales de los satélites disponibles, decodificación del mensaje de navegación y obtención de los observables necesarios para los algoritmos de posicionamiento) hasta la solución de navegación. Su diseño permite cualquier tipo de personalización, incluyendo la intercambiabilidad de las fuentes de señal, los algoritmos de procesamiento de señales, la interoperabilidad con otros sistemas, los formatos de salida, y ofrece interfaces para todas las señales, parámetros y variables intermedias. El software está desarrollado en C++ y utiliza GNU Radio para todas las funciones de procesado de señal de los receptores [14].

Una característica atractiva de este software es la posibilidad de trabajar en tiempo real con señales reales, cuando el procesador es lo suficientemente rápido, o realizar un post-procesamiento a partir de muestras almacenadas en un archivo, cuando la complejidad de la implementación impide un procesamiento en tiempo real.

El diagrama de bloques general del receptor se representa en la más adelante. Este se compone de un plano de control encargado de gestionar todo el receptor y las interacciones con el sistema operativo subyacente, las aplicaciones externas y la interfaz usuario-máquina. Además, también consta de un plano de procesamiento de señal que extrae la información de las muestras de la señal.

El plano de control se encarga de crear un diagrama de flujo mediante un archivo de configuración y de gestionar los módulos. El archivo de configuración permite al

satélite. Por lo tanto, esta división en canales permite encapsular los módulos de adquisición, *tracking* y decodificación.

La primera tarea de un receptor GNSS es detectar la presencia o ausencia de satélites a la vista. Esto se realiza mediante el proceso de adquisición del sistema, que también proporciona una estimación aproximada de dos parámetros de la señal: el desplazamiento de frecuencia con respecto a la IF nominal debido al efecto doppler, y un término de retardo que permite al receptor crear un código local alineado con el código entrante.

Cuando se declara la presencia de un satélite, los parámetros estimados por el módulo de adquisición se envían al módulo de *tracking* del receptor, que representa la segunda etapa de la unidad de procesamiento de la señal, y tiene como objetivo realizar una búsqueda local de la fase de la portadora, y seguir sus eventuales variaciones.

Seguidamente, con la información obtenidas por los bloques anteriores, la señal llega al módulo *Telemetry Decoder* (decodificador de telemetría), que se encarga de obtener los bits de datos del mensaje de navegación emitido por los satélites GNSS.

Una vez se han decodificado los bits del mensaje de navegación, el módulo *Observables* recoge los datos de sincronización procedentes de todos los canales de procesamiento y calcula, a partir de ellos, las mediciones básicas de un GNSS: pseudodistancia, fase de la portadora y desplazamiento doppler.

Finalmente, el módulo *PVT (Position-Velocity-Time)* calcula las soluciones de navegación y entrega esta información en formatos adecuados para su posterior procesamiento o representación.

A pesar de que estos son los bloques esenciales en el procesamiento de señal, existe un módulo adicional, denominado *Monitor*, que proporciona una interfaz para supervisar el estado interno del receptor en tiempo real mediante la transmisión de sus datos internos a clientes locales o remotos a través del protocolo UDP.

5.1.3. GNSS-SDR-Monitor

Uno de los inconvenientes que presenta GNSS-SDR es la ausencia de una interfaz gráfica de usuario que permita la representación de la posición del receptor y la monitorización de los parámetros básicos de las señales GNSS. Por este motivo es recomendable la utilización del software GNSS-SDR-Monitor, que proporciona una interfaz gráfica de usuario desarrollada para monitorizar el estado de GNSS-SDR en tiempo real.

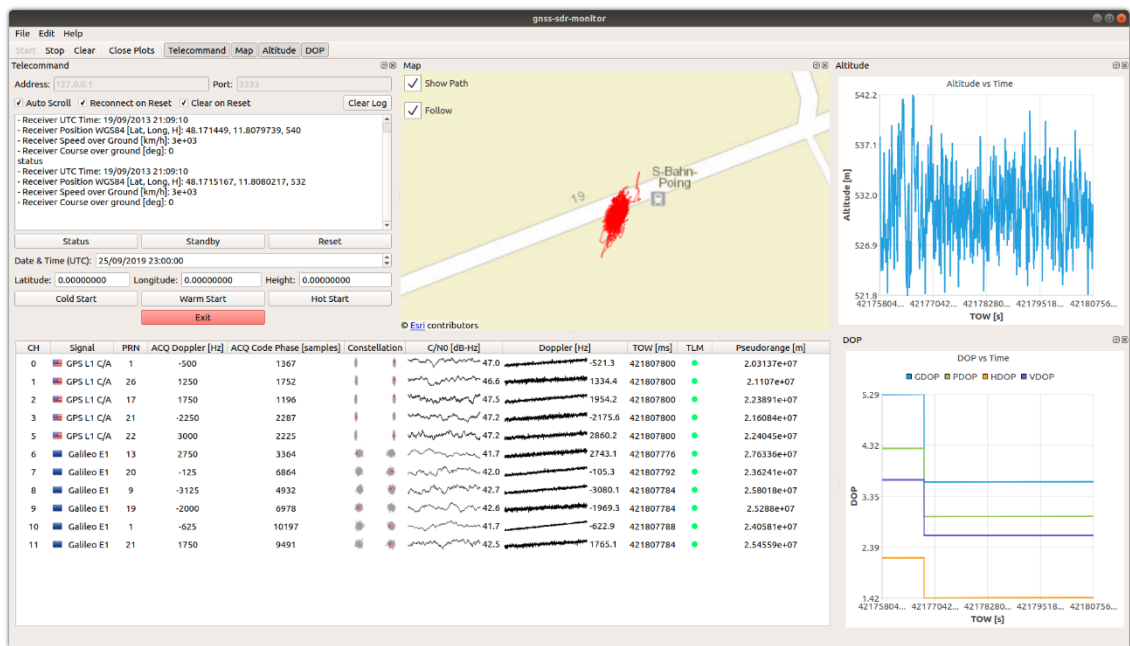


Figura 5.3. Interfaz de GNSS-SDR-Monitor [16].

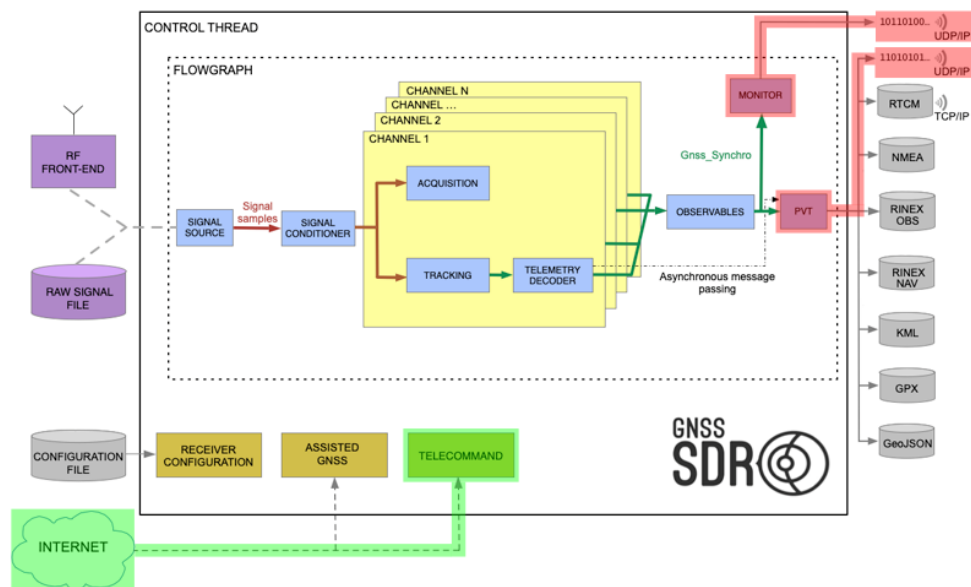


Figura 5.4. Bloques empleados por GNSS-SDR-Monitor [16].

Esta interfaz gráfica de usuario está diseñada para recibir datos en tiempo real de GNSS-SDR a través de la red, ya sea localmente o remotamente. Los datos que necesita GNSS-SDR-Monitor para funcionar provienen de GNSS-SDR a través de dos flujos de serialización sobre UDP: el flujo del bloque *Monitor* y el flujo del bloque *PVT*, los cuales son independientes entre sí y proporcionan diferentes tipos de información. Además, esta herramienta proporciona un widget para comunicarse con la interfaz interactiva del receptor (*Telecommand* vía TCP/IP), permitiendo enviar comandos especiales a GNSS-SDR.

5.1.4. MATLAB

MATLAB es una plataforma de programación que emplea un lenguaje basado en matrices. Este software proporciona un entorno ideal para el análisis y representación de datos, por lo que en el desarrollo de este trabajo MATLAB será empleado para realizar estos procesos con los distintos datos obtenidos por los bloques de GNSS-SDR, prestando especial atención a los bloques de adquisición y *tracking*.

5.2. Herramientas hardware

5.2.1. Antena

Para poder recibir correctamente señales GPS de frecuencia 1575.42 MHz se ha empleado la antena GPS Mike 3A de RoHS, que cuenta con un LNA y un filtro SAW para la atenuación de la señal fuera de banda. Esta antena presenta las siguientes características [17]:

- Conector SMA macho.
- Cable de 3 m.
- Ganancia de 3 dBiC.
- Frecuencia central de 1575.42 MHz $\pm$ 1.023 MHz.
- Ancho de banda del filtro de 10 MHz.
- Polarización RHCP.
- Ganancia del LNA de 28 dB (sin cable).
- Figura de ruido de 1.4 dB.
- Voltaje comprendido entre 3 V – 5 V.



Figura 5.5. Antena GPS Mike 3A de RoHS.

Dado que la antena empleada es una antena activa es necesario suministrar una alimentación mediante un *bias-tee*.

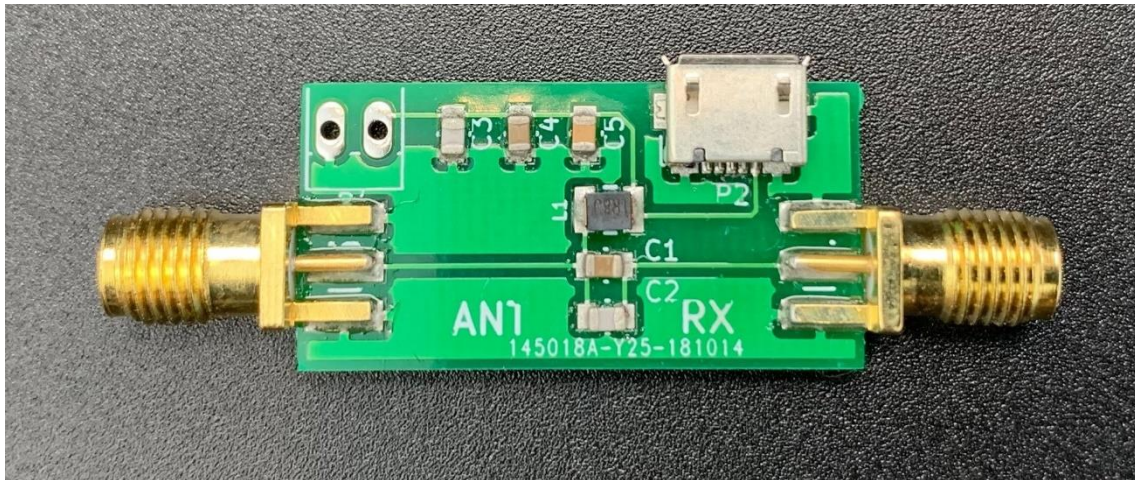


Figura 5.6. Bias-tee.

Un *bias-tee* es un componente de tres puertos (DC, RF+DC y RF) que inyecta energía continua en el cable de la antena desde el *front-end* para alimentar el amplificador de la antena [5].

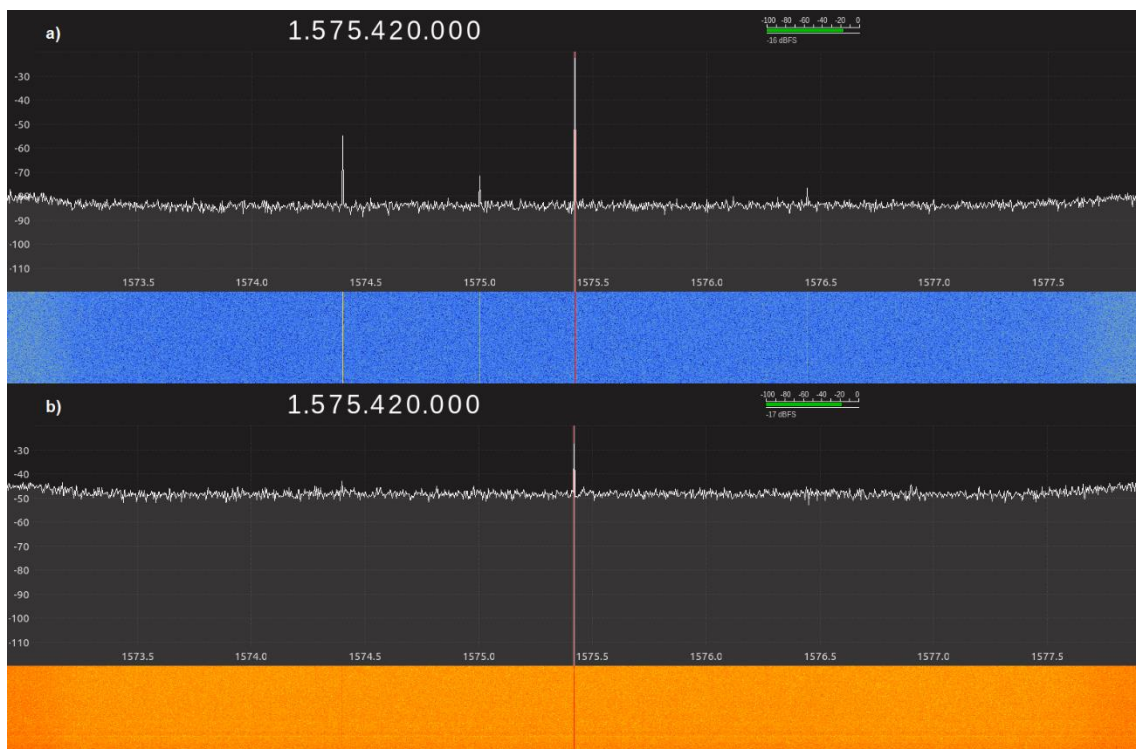


Figura 5.7. Nivel de señal recibida con bladeRF x40. a) Sin bias-tee. b) Con bias-tee.

5.2.2. RTL-SDR Blog V3 / bladeRF x40

Los dispositivos SDR bladeRF x40 o RTL-SDR Blog V3 son los encargados de realizar todas las funciones relativas al *front-end* de radiofrecuencia hasta la conversión de la señal de RF a muestras digitales en banda base que posteriormente serán procesadas por un ordenador.

Tal y como se comentó anteriormente, una ventaja de RTL-SDR Blog V3 frente a bladeRF x40 es que consta de un *bias-tee* integrado, de modo que no es necesario emplear un dispositivo para la alimentación de la antena.

5.2.3. Ordenador de propósito general

Para poder llevar a cabo toda el procesado y configuración del software es necesario el empleo de un ordenador. En concreto, para este trabajo se ha empleado el sistema operativo Ubuntu 21.04, aunque cualquier versión anterior superior a la 16.04 es adecuada para su desarrollo. El motivo por el cual se ha elegido el sistema operativo Ubuntu es la no disponibilidad del software GNSS-SDR en Windows.

Aunque ninguno de los programas descrito en la sección 5.1 presenta unos grandes requisitos, seguidamente se detallan los requisitos mínimos recomendables del ordenador de propósito general:

- Procesador de 4 núcleos de 64 bits (3 GHz).
- 4 GB de RAM.
- Interfaz USB 3.0.
- Memoria de 30 GB.

5.3. Implementación del receptor GPS

A la hora de implementar el receptor GPS se observan dos etapas claramente diferenciadas: la configuración de la plataforma SDR para la recepción de la señal GPS y el procesado digital de señal, para la obtención de la posición. Ambos procesos pueden ser implementados mediante el archivo de configuración de GNSS-SDR, el cual presenta una estructura claramente diferencia que divide la implementación en varias partes.

5.3.1. Estructura del archivo de configuración

El archivo de configuración de GNSS-SDR consta de una multitud de parámetros que permiten la recepción y el procesamiento de diferentes señales GNSS. Estos parámetros están directamente asociados a los distintos bloques de la Figura 5.2, por lo que el archivo de configuración presentara una estructura muy similar a este diagrama de bloques. A continuación, se procederá a exponer esta estructura, indicando aquellos parámetros de mayor importancia [15].

En primer lugar, antes de especificar algún parámetro es necesario el empleo de la etiqueta *[GNSS-SDR]* para que el software puede reconocer el archivo de configuración. Posteriormente, deben especificarse las opciones globales, que siguen la notación *GNSS-SDR.XX = YY*, donde *XX* representa el parámetro e *YY* indica su valor. Entre estos parámetros, existe uno que es esencial:

- *internal_fs_sps*. Frecuencia de muestreo de entrada a los canales de procesamiento, en muestras por segundo.

Seguidamente se debe configura el bloque *Signal Source*, que especifica la fuente de información además de realizar su configuración. Para este bloque se sigue la notación *SignalSource.XX = YY*, y sus parámetros más destacados son los siguientes:

- *implementation*. Este parámetro indica como se realiza la implementación de la fuente de señal. Para el caso de bladeRF x40 y RTL-SDR Blog V3 esta implementación se realiza mediante el bloque *osmosdr* de GNU Radio, por lo que debe emplearse el valor *Osmosdr_Signal_Source*.
- *freq*. Frecuencia central de recepción del *front-end*, en Hz.
- *gain*. Ganancia del *front-end* de RF para el canal de RF 0, en dB. Este valor se reparte por toda la cadena de recepción.
- *rf_gain*. Ganancia del *front-end* de RF para el amplificador de RF, en dB.
- *if_gain*: Ganancia del *front-end* de RF para el amplificador de IF, en dB.
- *sampling_frequency*. Frecuencia de muestreo, en muestras por segundo.
- *Item_type*. Establece el formato de datos de salida.
- *osmosdr_args*. Indica el SDR empleado y permite la configuración de parámetros adicionales, como el *bias-tee*.
- *telecommand_enabled*. Activa o desactiva el sistema *telecommand*.
- *telecommand_tcp_port*. Establece el puerto TCP empleado para el sistema *telecommand*.

```
##### SIGNAL_SOURCE CONFIG #####
SignalSource.implementation=Osmosdr_Signal_Source
SignalSource.item_type=gr_complex
SignalSource.sampling_frequency=2000000
SignalSource.freq=1575420000
SignalSource.gain=40
SignalSource.rf_gain=40
SignalSource.if_gain=30
SignalSource.enable_throttle_control=false
SignalSource.osmosdr_args=rtl,bias=1
```

Figura 5.8. Ejemplo de configuración del bloque *Signal Source* para un dispositivo RTL-SDR.

El siguiente paso es la configuración del *Signal Conditioner*, que comprende la configuración de tres bloques de procesamiento: *Data Type Adapter*, *Input Filter* y *Resampler*. A continuación, se detallan los parámetros más importantes:

- *SignalConditioner.implementation*. Establece si se desea implementar el bloque *Signal Conditioner*. En caso de que se desee implementar toma el valor *Signal_Conditioner*, mientras que si no se desea implementar tomará el valor *Pass_Through*.
- *DataTypeAdapter.implementation*. Implementa una conversión del formato de datos en caso de que sea necesario.
- *InputFilter.implementation*. Implementa un filtro para la eliminación de ruido y posibles interferencias en caso de que sea necesario.
- *Resampler.implementation*. Realiza un nuevo muestreo de la señal en caso de que sea necesario.

```
##### SIGNAL_CONDITIONER CONFIG #####
;## It holds blocks to change data type, filter and resample input data.
SignalConditioner.implementation=Signal_Conditioner

##### DATA_TYPE_ADAPTER CONFIG #####
;## Changes the type of input data.
;#implementation: [Pass_Through] disables this block
DataTypeAdapter.implementation=Ishort_To_Complex

##### INPUT_FILTER CONFIG #####
;## Filter the input data.
InputFilter.implementation=Fir_Filter
InputFilter.input_item_type=gr_complex
InputFilter.output_item_type=gr_complex
; ... other InputFilter parameters

##### RESAMPLER CONFIG #####
;## Resamples the input data.
Resampler.implementation=Pass_Through
```

Figura 5.9. Ejemplo de configuración del bloque *Signal Conditioner*.

Tras esto es necesario realizar la configuración de los distintos canales para la recepción de señales GPS L1 C/A. Para esto se deben emplear los siguientes parámetros:

- *Channels_1C.count*. Número de canales destinados a las señales GPS L1 C/A.
- *Channels.in_acquisition*. Número máximo de canales que realizan la adquisición de la señal al mismo tiempo.

Una vez configurados los canales se debe configurar la adquisición de la señal. Para establecer que este proceso se desear realizar a las señales GPS L1 C/A se emplea la notación *Acquisition_1C.XX = YY*, y los parámetros más destacados son:

- *implementation*. Indica el algoritmo de adquisición empleado. El algoritmo de adquisición empleado es el algoritmo PCPSA descrito anteriormente, el cual se secciona mediante el valor *GPS_L1_CA_PCPS_Acquisition*.
- *doppler_max*. Valor máximo de desplazamiento doppler empleado en la búsqueda, en Hz.
- *doppler_step*. Salto en frecuencia empleado para la búsqueda del desplazamiento Doppler, en Hz.

```
##### ACQUISITION CONFIG FOR GPS L1 CHANNELS #####
Acquisition_1C.implementation=GPS_L1_CA_PCPS_Acquisition
Acquisition_1C.doppler_max=5000
Acquisition_1C.doppler_step=250
Acquisition_1C.pfa=0.01
Acquisition_1C.coherent_integration_time_ms=1
Acquisition_1C.max_dwells=1
```

Figura 5.10. Ejemplo de configuración del bloque de adquisición.

Después del bloque de adquisición, el siguiente bloque en ser configurado es el de *Tracking*. Para ello se sigue la notación *Tracking_1C.XX = YY*, y sus principales parámetros son:

- *implementation*. Establece la implementación del proceso de *tracking*. Esta implementación se realiza del modo descrito en el apartado 3.6.4 mediante el valor *GPS_L1_CA_DLL_PLL_Tracking*.
- *extend_correlation_symbols*. Establece el número de símbolos de correlación que se extenderá después de lograr la sincronización de bits.
- *early_late_space_chips*. Establece la separación de las réplicas temprana, y tardía respecto a la réplica alineada con la señal de entrada..

```

##### TRACKING CONFIG FOR GPS L1 CHANNELS #####
Tracking_1C.implementation=GPS_L1_CA_DLL_PLL_Tracking
Tracking_1C.item_type=gr_complex
Tracking_1C.extend_correlation_symbols=20
Tracking_1C.early_late_space_chips=0.5;
Tracking_1C.early_late_space_narrow_chips=0.1;
Tracking_1C.pll_bw_hz=35;
Tracking_1C.dll_bw_hz=2.0;
Tracking_1C.pll_bw_narrow_hz=5.0;
Tracking_1C.dll_bw_narrow_hz=0.50;
Tracking_1C.fll_bw_hz=10
Tracking_1C.enable_fll_pull_in=true;
Tracking_1C.enable_fll_steady_state=false
Tracking_1C.dump=false
Tracking_1C.dump_filename=tracking_ch_

```

Figura 5.11. Ejemplo de configuración del bloque de tracking.

Tras el proceso de *tracking*, se debe configurar el bloque *Telemetry* que se encarga de la obtención de los bits de datos del mensaje de navegación. Para ello se debe emplear el siguiente parámetro:

- *TelemetryDecoder_1C.implementation*. Implementa la recepción de los bits de datos. Para realizar esta recepción de la señal GPS L1 C/A se emplea el valor *GPS_L1_CA_Telemetry_Decoder*.

Seguidamente se realiza la configuración del bloque *Observables*, que permite el cálculo de las medidas básicas. En este caso, únicamente es obligatorio el uso del siguiente parámetro:

- *Observables.implementation*. Permite la implementación del bloque *Observables* mediante el valor *Hybrid_Observables*.

A continuación, se debe configurar el bloque *PVT*, encargado de la obtención de la posición a partir de las mediciones básicas calculados anteriormente. Para este bloque se emplea la notación *PVT.XX = YY* y sus parámetros básicos son los siguientes:

- *implementation*. Implementa las librerías necesarias para el cálculo de la posición mediante el parámetro *RTKLIB_PVT*.
- *output_rate_ms*. Indica la velocidad a la que las soluciones PVT serán computadas, en ms.
- *display_rate_ms*. Indica la velocidad a la que las soluciones PVT serán mostradas en la terminal, en ms.
- *positioning_mode*. Establece el modo de posicionamiento (SPS o PPS).

Dado que este bloque también se comunica con GNSS-SDR-Monitor es necesario emplear los siguientes parámetros adicionales:

- *PVT.enable_monitor*. Activa o desactiva la transmisión de los datos del bloque a otro equipo.
- *PVT.monitor_client_addresses*. Indica la dirección IP del equipo que recibirá la información del bloque.
- *PVT.monitor_udp_port*. Selecciona el puerto UDP empleado por el receptor.

```
##### PVT CONFIG #####
PVT.implementation=RTKLIB_PVT
PVT.positioning_mode=PPP_Static
PVT.output_rate_ms=100
PVT.display_rate_ms=500
PVT.iono_model=Broadcast
PVT.trop_model=Saastamoinen
PVT.flag_rtcn_server=true
PVT.flag_rtcn_tty_port=false
PVT.rtcn_dump_devname=/dev/pts/1
PVT.rtcn_tcp_port=2101
PVT.rtcn_MT1019_rate_ms=5000
PVT.rtcn_MT1045_rate_ms=5000
PVT.rtcn_MT1097_rate_ms=1000
PVT.rtcn_MT1077_rate_ms=1000
PVT.rinex_version=2
```

Figura 5.12. Ejemplo de configuración del bloque PVT.

Finalmente, el único bloque restante para su configuración es el bloque *Monitor*, que será el encargado de tramitar la información a GNSS-SDR-Monitor. Para ello, se emplea la notación *Monitor.XX = YY*. En este caso, los parámetros fundamentales son los siguientes:

- *enable_monitor*. Activa o desactiva el bloque *Monitor*.
- *client_addresses*. Establece la dirección IP a la cual serán enviados los datos.
- *udp_port*. Establece el puerto UDP del receptor empleado para la comunicación.

```
;##### MONITOR CONFIG #####  
Monitor.enable_monitor=true  
Monitor.decimation_factor=1  
Monitor.client_addresses=127.0.0.1  
Monitor.udp_port=1234
```

Figura 5.13. Ejemplo de configuración del bloque Monitor.

6. RESULTADOS OBTENIDOS

6.1. Posicionamiento de un receptor estático con bladeRFx40

En primer lugar, se ha realizado una prueba para la obtención del posicionamiento de un receptor estático mediante bladeRF x40. Para ello se ha empleado el software GNSS-SDR con el archivo de configuración descrito en el Anexo III.A: Archivo de configuración para recepción de señal GPS L1 C/A con bladeRF x40, obteniéndose el siguiente resultado:

```
Initializing GNSS-SDR v0.0.14-git-next-ecc339e68b ... Please wait.
Logging will be written at "/tmp"
Use gnss-sdr --log_dir=/path/to/log to change that.
OsmoSdr arguments: bladerf=0
gr-osmosdr 0.2.0.0 (0.2.0) gnuradio 3.9.0.0
built-in source types: file rtl rtl_tcp uhd hackrf bladerf rfspice airspy airspyhf soapy redpitaya
[bladerf common] init: DEBUG: entering initialization
[bladerf source] init: Opening Nuand bladerf with device identifier string '*:instance=0'
[bladerf source] Device: Nuand bladerf Serial # 7357...7fb6 FW v2.4.0 FPGA v0.11.0
[bladerf source] init: Buffers: 512, samples per buffer: 4096, active transfers: 32
[bladerf source] bladerf_source_c: DEBUG: initialization complete
Actual RX Rate: 2e+06 [SPS]...
Actual RX Freq: 1.57542e+09 [Hz]...
PLL Frequency tune error: 0 [Hz]...
[WARNING @ host/libraries/libbladeRF/src/board/bladerf1/bladerf1.c:1869] bladerf1_get_gain_stage: gain stage 'LNA' invalid
Actual LNA Gain: 0 dB...
[WARNING @ host/libraries/libbladeRF/src/board/bladerf1/bladerf1.c:1869] bladerf1_get_gain_stage: gain stage 'VGA1' invalid
Actual VGA1 Gain: 0 dB...
[WARNING @ host/libraries/libbladeRF/src/board/bladerf1/bladerf1.c:1869] bladerf1_get_gain_stage: gain stage 'VGA2' invalid
Actual VGA2 Gain: 0 dB...
Actual Bandwidth: 3e+06 [Hz]...
RF Channels: 1
[bladerf source] start: DEBUG: starting source
Current receiver time: 1 s
Current receiver time: 2 s
Current receiver time: 3 s
Current receiver time: 4 s
Current receiver time: 5 s
Current receiver time: 6 s
Current receiver time: 7 s
Current receiver time: 8 s
Current receiver time: 9 s
Current receiver time: 10 s
Current receiver time: 11 s
Current receiver time: 12 s
Current receiver time: 13 s
Current receiver time: 14 s
Current receiver time: 15 s
Current receiver time: 16 s
Current receiver time: 17 s
Current receiver time: 18 s
Current receiver time: 19 s
Current receiver time: 20 s
Current receiver time: 21 s
Current receiver time: 22 s
Current receiver time: 23 s
Current receiver time: 24 s
Current receiver time: 25 s
Current receiver time: 26 s
Current receiver time: 27 s
Current receiver time: 28 s
Current receiver time: 29 s
```

Figura 6.1. Salida del terminal para bladeRF x40.

Al analizar la salida del terminal puede observarse una advertencia que indica que las distintas etapas de ganancia no son reconocidas. No obstante, esta inconveniente se soluciona mediante la configuración manual de las etapas de ganancia a través de la interfaz de línea de comandos de bladeRF (bladeRF-cli). Cabe destacar que para que esta configuración manual tenga efecto es necesario eliminar la configuración de las etapas de ganancia en el archivo de configuración de GNSS-SDR.

Para realizar la configuración de la ganancia de bladeRF x40 se debe tener en consideración que este dispositivo divide la amplificación en tres etapas. En primer lugar, se lleva a cabo una amplificación de la señal de RF mediante un LNA cuya ganancia puede tomar tres valores: 0 dB, 3 dB o 6 dB. Una vez se ha convertido la señal a banda base se realiza otra etapa de amplificación de ganancia variable comprendida en el rango 5 dB – 30 dB. En caso de ser necesario realizar una amplificación de mayor ganancia es posible emplear una etapa adicional cuya ganancia se encuentra entre 0 dB – 30 dB.

```
usuario@usuario:~$ pybombs -p TFG run bladeRF-cli -- -i
[INFO] Prefix Python version is: 3.8.6
[INFO] PyBOMBS Version 2.3.4
bladeRF> set gain rx1 lna 6

Note: This change will not be visible until the channel is enabled.
Setting RX1 lna gain to 6 dB
      lna:    6 dB (Range: [0, 6])

bladeRF> set gain rx1 rxvga1 30

Note: This change will not be visible until the channel is enabled.
Setting RX1 rxvga1 gain to 30 dB
      rxvga1: 30 dB (Range: [5, 30])

bladeRF> set gain rx1 rxvga2 18

Note: This change will not be visible until the channel is enabled.
Setting RX1 rxvga2 gain to 18 dB
      rxvga2: 18 dB (Range: [0, 30])
```

Figura 6.2. Configuración de las etapas de ganancia con bladeRF-cli.

Sin embargo, a pesar de realizar esta configuración de las ganancias la salida obtenida sigue siendo similar a la representada en la Figura 6.1. De este modo se puede observar cómo el receptor es incapaz de identificar los satélites visibles, por lo que no es posible iniciar el proceso de *tracking*. Esto es debido a que la estabilidad en frecuencia del oscilador que posee bladeRF x40 es superior al valor típico de 1 ppm proporcionado por el fabricante.

De hecho, al observar el resultado de un gráfico de adquisición, como el que se muestra en la Figura 6.3, se puede establecer que no existe un único pico claramente diferenciado, por lo que no es posible identificar la frecuencia doppler ni la fase del código.

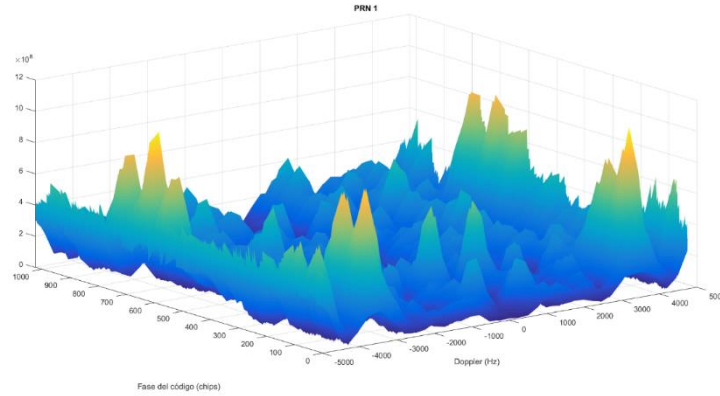


Figura 6.3. Gráfico de adquisición para el código PRN 1.

Para poder obtener una posición empleando bladeRF x40 sería necesario el adquirir un oscilador externo de mayor precisión que actuase como señal de referencia. No obstante, la adquisición de un oscilador de mayor precisión (por ejemplo, el GPSDO de *Leo Bodnar* empleado en [18]) supone un coste mayor que la adquisición del dispositivo RTL-SDR Blog V3, por lo que se ha optado por adquirir directamente este dispositivo SDR.

6.2. Posicionamiento de un receptor estático con RTL-SDR Blog V3

6.2.1. Recepción de la señal GPS

Tras no haber sido posible obtener la posición de un receptor estático con bladeRF x40 se procede a realizar este proceso mediante el dispositivo RTL-SDR Blog V3. Para ello, una vez diseñado el archivo de configuración que se muestra en el Anexo III.B: Archivo de configuración para la recepción de señal GPS L1 C/A con RTL-SDR Blog V3 se procede a su ejecución en GNSS-SDR.

En primer lugar se observa que al ejecutar el archivo se inicia el proceso de adquisición en los diferentes canales del receptor. Una vez que la adquisición se ha realizado correctamente se inicia el *tracking* de los satélites detectados en los diferentes canales.

```

Initializing GNSS-SDR v0.0.14.git-next-ecc339e68b ... Please wait.
Logging will be written at "/tmp"
Use gnss-sdr --log_dir=/path/to/log to change that.
OsmoSdr arguments: rtl,bias=1
gr-osmosdr 0.2.0.0 (0.2.0) gnuradio 3.9.0.0
built-in source types: file rtl rtl_tcp uhd hackrf bladerf rfspcse airspy airspyhf soapy redpitaya
Using device #0 Realtek RTL2838UHDIR SN: 00000001
Detached kernel driver
Found Rafael Micro R820T tuner
[R82XX] PLL not locked!
[R82XX] PLL not locked!
Actual RX Rate: 2.048e+06 [SPS]...
Actual RX Freq: 1.57542e+09 [Hz]...
PLL Frequency tune error: 0 [Hz]...
Actual RX Gain: 40.2 dB...
Actual Bandwidth: 0 [Hz]...
RF Channels: 1
TcpCmdInterface: Telecommand TCP interface listening on port 3333
Current receiver time: 1 s
Tracking of GPS L1 C/A signal started on channel 2 for satellite GPS PRN 10 (Block IIF)
Tracking of GPS L1 C/A signal started on channel 0 for satellite GPS PRN 16 (Block IIR)
Tracking of GPS L1 C/A signal started on channel 6 for satellite GPS PRN 21 (Block IIR)
Tracking of GPS L1 C/A signal started on channel 7 for satellite GPS PRN 22 (Block IIR)
Tracking of GPS L1 C/A signal started on channel 1 for satellite GPS PRN 23 (Block III)
Tracking of GPS L1 C/A signal started on channel 3 for satellite GPS PRN 27 (Block IIF)

```

Figura 6.4. Inicio del proceso de tracking.

Durante este proceso de tracking se refinan los valores de frecuencia doppler y fase del código obtenidos en el proceso de adquisición. Además de estos parámetros, también se rastrea la fase de la portadora, por lo que se realiza un proceso de sincronización a nivel de bit para posteriormente emplear un correlador de 20 ms (periodo de bit) encargado de la decodificación del mensaje de navegación

```

GPS L1 C/A tracking bit synchronization locked in channel 3 for satellite GPS PRN 27 (Block IIF)
Enabled 20 ms extended correlator in channel 3 for satellite GPS PRN 27 (Block IIF)
Current receiver time: 7 s
GPS L1 C/A tracking bit synchronization locked in channel 4 for satellite GPS PRN 01 (Block IIF)
Enabled 20 ms extended correlator in channel 4 for satellite GPS PRN 01 (Block IIF)
GPS L1 C/A tracking bit synchronization locked in channel 6 for satellite GPS PRN 21 (Block IIR)
Enabled 20 ms extended correlator in channel 6 for satellite GPS PRN 21 (Block IIR)
GPS L1 C/A tracking bit synchronization locked in channel 2 for satellite GPS PRN 10 (Block IIF)
Enabled 20 ms extended correlator in channel 2 for satellite GPS PRN 10 (Block IIF)
Current receiver time: 8 s
Current receiver time: 9 s
GPS L1 C/A tracking bit synchronization locked in channel 0 for satellite GPS PRN 16 (Block IIR)
Enabled 20 ms extended correlator in channel 0 for satellite GPS PRN 16 (Block IIR)
GPS L1 C/A tracking bit synchronization locked in channel 5 for satellite GPS PRN 08 (Block IIF)
Enabled 20 ms extended correlator in channel 5 for satellite GPS PRN 08 (Block IIF)
GPS L1 C/A tracking bit synchronization locked in channel 7 for satellite GPS PRN 22 (Block IIR)
Enabled 20 ms extended correlator in channel 7 for satellite GPS PRN 22 (Block IIR)
GPS L1 C/A tracking bit synchronization locked in channel 1 for satellite GPS PRN 23 (Block III)
Enabled 20 ms extended correlator in channel 1 for satellite GPS PRN 23 (Block III)
Current receiver time: 10 s

```

Figura 6.5. Inicio de la de la decodificación del mensaje de navegación

Pasado un tiempo, se observa como los mensajes de navegación empiezan a ser recibidos por el receptor. Además, en este momento sería posible visualizar los principales parámetros de las señales de los satélites visibles en GNSS-SDR-Monitor.

```

Current receiver time: 27 s
New GPS NAV message received in channel 2: subframe 1 from satellite GPS PRN 10 (Block IIF)
New GPS NAV message received in channel 6: subframe 1 from satellite GPS PRN 21 (Block IIR)
New GPS NAV message received in channel 0: subframe 1 from satellite GPS PRN 16 (Block IIR)
New GPS NAV message received in channel 3: subframe 1 from satellite GPS PRN 27 (Block IIF)
New GPS NAV message received in channel 4: subframe 1 from satellite GPS PRN 01 (Block IIF)
Current receiver time: 28 s

```

Figura 6.6. Recepción de los mensajes de navegación.

Llegado a este punto, los procesos descritos anteriormente se llevan a cabo de forma continua hasta la recepción estable de mensajes de navegación de 4 o más satélites. Una vez se consigue esta recepción estable de 4 o más satélite es posible obtener la posición del receptor.

```

New GPS NAV message received in channel 1: subframe 3 from satellite GPS PRN 23 (Block III)
New GPS NAV message received in channel 7: subframe 3 from satellite GPS PRN 22 (Block IIR)
New GPS NAV message received in channel 3: subframe 3 from satellite GPS PRN 27 (Block IIF)
New GPS NAV message received in channel 5: subframe 3 from satellite GPS PRN 08 (Block IIF)
First position fix at 2021-Aug-21 09:03:29.905013 UTC is Lat = 38.0433 [deg], Long = -3.09442 [deg], Height= 461.7 [n]
Position at 2021-Aug-21 09:03:30.065013 UTC using 5 observations is Lat = 38.043266838 [deg], Long = -3.094418676 [deg], Height = 454.898 [n]
Velocity: East: -0.006 [n/s], North: -0.023 [n/s], Up = 0.093 [n/s]
Current receiver time: 40 s
Position at 2021-Aug-21 09:03:30.565013 UTC using 5 observations is Lat = 38.043267321 [deg], Long = -3.094406139 [deg], Height = 454.307 [n]
Velocity: East: 0.015 [n/s], North: 0.019 [n/s], Up = -0.007 [n/s]
Position at 2021-Aug-21 09:03:31.065013 UTC using 5 observations is Lat = 38.043281229 [deg], Long = -3.094431940 [deg], Height = 458.593 [n]
Velocity: East: -0.028 [n/s], North: 0.006 [n/s], Up = 0.044 [n/s]
Position at 2021-Aug-21 09:03:31.565013 UTC using 5 observations is Lat = 38.043257118 [deg], Long = -3.094427635 [deg], Height = 467.319 [n]
Velocity: East: -0.010 [n/s], North: -0.024 [n/s], Up = 0.041 [n/s]
Current receiver time: 41 s
Position at 2021-Aug-21 09:03:32.065013 UTC using 5 observations is Lat = 38.043246126 [deg], Long = -3.094382813 [deg], Height = 456.580 [n]
Velocity: East: 0.003 [n/s], North: -0.005 [n/s], Up = 0.020 [n/s]
Current receiver time: 42 s

```

Figura 6.7 Obtención de la posición del receptor.

Tras obtener la primera posición fija, el receptor sigue funcionando de manera continuada, por lo que seguirá recibiendo los diferentes mensajes de navegación de los satélites visibles para calcular una nueva posición.

```

Current receiver time: 50 s
Position at 2021-Aug-21 09:03:40.565013 UTC using 5 observations is Lat = 38.043237591 [deg], Long = -3.094418447 [deg], Height = 474.813 [n]
Velocity: East: -0.005 [n/s], North: -0.019 [n/s], Up = 0.004 [n/s]
Position at 2021-Aug-21 09:03:41.065013 UTC using 5 observations is Lat = 38.043234350 [deg], Long = -3.094422108 [deg], Height = 470.922 [n]
Velocity: East: -0.004 [n/s], North: 0.003 [n/s], Up = 0.030 [n/s]
Position at 2021-Aug-21 09:03:41.565013 UTC using 5 observations is Lat = 38.043243498 [deg], Long = -3.094424838 [deg], Height = 471.879 [n]
Velocity: East: -0.018 [n/s], North: 0.006 [n/s], Up = 0.022 [n/s]
Current receiver time: 51 s
New GPS NAV message received in channel 3: subframe 5 from satellite GPS PRN 27 (Block IIF)
New GPS NAV message received in channel 0: subframe 5 from satellite GPS PRN 16 (Block IIR)
New GPS NAV message received in channel 6: subframe 5 from satellite GPS PRN 21 (Block IIR)
New GPS NAV message received in channel 2: subframe 5 from satellite GPS PRN 10 (Block IIF)
New GPS NAV message received in channel 7: subframe 5 from satellite GPS PRN 22 (Block IIR)
New GPS NAV message received in channel 5: subframe 5 from satellite GPS PRN 08 (Block IIF)
GPS PRN 01 (Block IIF)
New GPS NAV message received in channel 1: subframe 5 from satellite GPS PRN 23 (Block III)
Position at 2021-Aug-21 09:03:42.065014 UTC using 5 observations is Lat = 38.043254555 [deg], Long = -3.094437065 [deg], Height = 467.295 [n]
Velocity: East: -0.004 [n/s], North: -0.015 [n/s], Up = 0.003 [n/s]
Position at 2021-Aug-21 09:03:42.565014 UTC using 5 observations is Lat = 38.043228503 [deg], Long = -3.094417564 [deg], Height = 471.594 [n]
Velocity: East: 0.037 [n/s], North: 0.017 [n/s], Up = -0.144 [n/s]
Current receiver time: 52 s
Position at 2021-Aug-21 09:03:43.065014 UTC using 5 observations is Lat = 38.043219415 [deg], Long = -3.094411428 [deg], Height = 472.289 [n]
Velocity: East: -0.013 [n/s], North: -0.000 [n/s], Up = 0.041 [n/s]
Current receiver time: 53 s
Position at 2021-Aug-21 09:03:43.565014 UTC using 5 observations is Lat = 38.043234135 [deg], Long = -3.094404366 [deg], Height = 473.152 [n]
Velocity: East: -0.004 [n/s], North: -0.007 [n/s], Up = 0.049 [n/s]
Position at 2021-Aug-21 09:03:44.065014 UTC using 5 observations is Lat = 38.043209370 [deg], Long = -3.094394706 [deg], Height = 474.173 [n]
Velocity: East: -0.014 [n/s], North: -0.009 [n/s], Up = 0.043 [n/s]
Current receiver time: 54 s

```

Figura 6.8. Actualización de la posición del receptor.

Finalmente cabe destacar que de forma simultánea a la ejecución de GNSS-SDR, GNSS-SDR-Monitor mostrará en su parte inferior los distintos satélites adquiridos, detallando sus principales parámetros. Por tanto, una vez se consiga la adquisición de al menos 4 satélites, se obtendrá una posición, la cual será representada en un mapa. Además, también existe una gráfica adicional que informa de como varia la dilución de

precisión (DOP) a lo largo de la recepción. La dilución de precisión es un coeficiente que indica la contribución de la disposición geométrica de los satélites a la incertidumbre de la posición del receptor, por lo que es conveniente que su valor sea lo menor posible.



Figura 6.9. Interfaz de GNSS-SDR Monitor durante la recepción.

6.2.2. Representación de los resultados de los bloques intermedios

A través de GNSS-SDR y GNSS-SDR-Monitor es posible observar la posición del receptor u otros parámetros destacados de las señales recibidas. No obstante, es conveniente realizar una representación más exhaustiva de los resultados obtenidos en los bloques intermedios. Para ello, los archivos obtenidos en estos bloques pueden ser cargados en MATLAB, pudiéndose realizar una representación de los parámetros más destacados.

En primer lugar, se representarán los diagramas de adquisición obtenidos en el canal 0 mediante la función de MATLAB `acq_plot.m`.

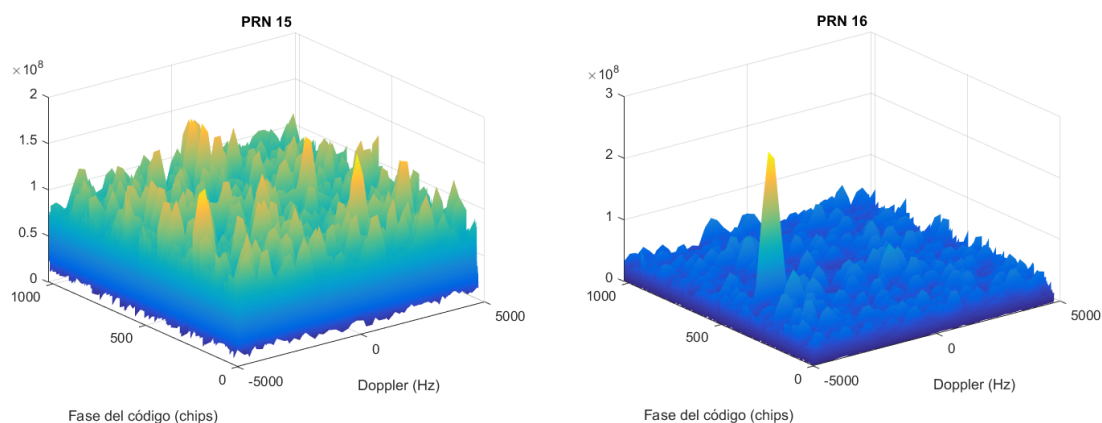


Figura 6.10. Gráficos de adquisición en el canal 0 para los códigos 15 y 16.

La Figura 6.10 muestra los gráficos de adquisición obtenidos en el canal 0. En un inicio se observa como para el PRN 15 se muestran varios picos sin que exista uno de ellos claramente diferenciado, por lo no es posible la adquisición de esta señal. Sin

embargo, también se puede observar que el gráfico de adquisición del PRN 16 si presenta un pico claramente diferenciado, por lo que en este caso si es posible la adquisición de la señal. Concretamente, este pico se produce para una frecuencia doppler de 3500 Hz y una fase de código de 443.3 chips.

Seguidamente, se procede a representar los resultados obtenidos en el proceso de tracking. Para ello se ha empleado la función *track_plot.m*.

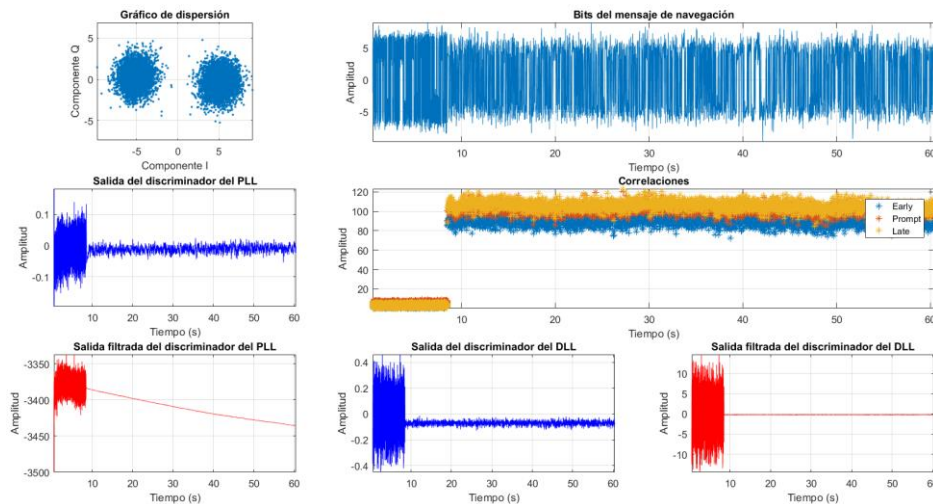


Figura 6.11. Resultados del proceso de tracking en el canal 1.

En la Figura 6.11 se muestra los resultados de *tracking* obtenidos en canal 1. En primer lugar, se representa el diagrama de dispersión de las componentes en fase (*I*) y cuadratura (*Q*). La calidad del proceso de tracking puede ser evaluada de forma directa mediante la examinación de la simetría de los grupos de concentración de este diagrama. Además, también se han representado los bits del mensaje de navegación obtenidos tras el proceso. Debajo de este gráfico, se muestran los resultados obtenidos de la correlación de la señal de entrada con los códigos *Early*, *Prompt* y *Late*. Para obtener una mayor calidad en el proceso es conveniente que el valor máximo de la correlación se obtenga con el código *Prompt*, aunque para el caso mostrado la correlación obtenida con el código *Late* es ligeramente superior. Por último, se muestra la salida de los discriminadores del PLL y del DLL, que representan el error obtenido en los osciladores. La salida directa de estos discriminadores se representa mediante el color azul, mientras que la salida filtrada se muestra en rojo.

Finalmente cabe destacar que en los primeros segundos de las representaciones realizadas se muestra una salida ruidosa que posteriormente se estabiliza. Esto es debido a que en estos instantes los bucles empleados se encuentran en el proceso de enganche, por lo que la salida del sistema es de peor calidad que la obtenida una vez se realiza el enganche.

6.2.3. Análisis de la precisión del receptor

Finalmente, se va a proceder a analizar la precisión de la posición del receptor. Para ello, se empleará el archivo KML (*Keyhole Markup Language*) obtenido en el bloque *Monitor*. KML es un lenguaje de marcas que permite la representación y almacenamiento de datos geográficos y sus asociados. Una de las ventajas del empleo de este lenguaje es la compatibilidad con una gran cantidad de aplicaciones gratuitas, entre las que se incluye *Google Earth*.

Por tanto, para representar las distintas posiciones obtenidas durante el proceso de recepción será necesario la importación de este archivo en *Google Earth*. En la Figura 6.12 se representa las distintas posiciones obtenidas por el receptor, para las que se emplea el color amarillo mientras que la posición real del receptor se representa con un marcador azul.

Después de haber representado la posición del receptor y las distintas posiciones obtenidas se procede a realizar el cálculo del error cometido en el posicionamiento. Para ello, en primer lugar, se deben conocer las coordenadas de las distintas posiciones de interés.



Figura 6.12. Representación de las posiciones obtenidas en Google Earth.

Posición	Latitud (°)	Longitud (°)	Altitud (m)
Receptor	38.04317	-3.09445	434.77275

Tabla 6.1. Posición real de la antena del receptor.

Posición	Latitud (°)	Longitud (°)	Altitud (m)
1	38.04327	-3.09442	454.89798
2	38.04328	-3.09443	458.59349
3	38.04325	-3.09438	456.58039
4	38.04325	-3.09439	469.26156
5	38.04323	-3.09434	465.82192
6	38.04324	-3.09435	478.20349
7	38.04323	-3.09434	476.36807
8	38.04321	-3.09435	480.21375
9	38.04322	-3.09439	484.74918
10	38.04321	-3.09439	483.95942
11	38.04324	-3.09441	472.46079
12	38.04323	-3.09442	470.92180
13	38.04325	-3.09444	467.29482
14	38.04322	-3.09441	472.28916
15	38.04321	-3.09439	474.17344
16	38.04321	-3.09439	481.11446
17	38.04322	-3.09439	473.55935
18	38.04321	-3.09439	473.70459
19	38.04319	-3.09439	470.86262
20	38.04318	-3.09439	474.46708
21	38.04313	-3.09439	481.60404
22	38.04316	-3.09439	473.13276

Tabla 6.2. Coordenadas de las posiciones obtenidas con el receptor.

Una vez conocidas estas coordenadas se procede al cálculo del error mediante el empleo de la fórmula del semiverseno. Esta fórmula permite calcular la distancia entre dos puntos situados en la superficie de una esfera a través de la siguiente expresión

$$d = 2 R \arcsin \left[\sqrt{\sin^2 \left(\frac{\varphi_1 - \varphi_2}{2} \right) + \cos(\varphi_1) \cos(\varphi_2) \sin^2 \left(\frac{\lambda_1 - \lambda_2}{2} \right)} \right], \quad (23)$$

donde R es el radio de la esfera, φ_1 y φ_2 indican la latitud de los puntos en radianes y, λ_1 y λ_2 son la longitud de los puntos expresadas en radianes. Por tanto, para este cálculo se asume que la Tierra es una esfera perfecta de radio 6371 km.

Teniendo en cuenta la anterior ecuación se ha implementado en MATLAB la función *pos_error.m* que permite calcular el error horizontal y vertical de un conjunto de posiciones respecto a una posición.

Posición	Error horizontal (m) (latitud y longitud)	Error vertical (m) (altitud)
1	11.30	20.13
2	12.51	23.82
3	11.28	21.81
4	11.51	34.49
5	13.93	31.05
6	13.32	43.43
7	14.22	41.60
8	12.10	45.44
9	8.68	49.98
10	7.98	49.19
11	9.14	37.69
12	7.79	36.15
13	9.50	32.52
14	6.96	37.52
15	7.54	39.40
16	8.21	46.34
17	8.70	38.79
18	8.59	38.93
19	6.77	36.09
20	6.30	39.69
21	8.62	46.83
22	7.00	38.36

Tabla 6.3. Error cometido en las posiciones obtenidas.

	Error horizontal (m) (latitud y longitud)	Error vertical (m) (altitud)
Error (media $\pm$ desviación típica)	9.63 $\pm$ 2.45	37.70 $\pm$ 8.16

Tabla 6.4. Error medio.

Analizando los resultados obtenidos se puede observar como el error horizontal se encuentra dentro de los márgenes establecidos en el estándar GPS SPS. Sin embargo, esto no ocurre en el caso del error vertical, ya que este se ve fuertemente afectado por la geometría de los satélites. De hecho, en la Figura 6.9 se puede observar como la dilución de precisión vertical (VDOP) tiene un valor aproximado de 3.5, mientras que la dilución de precisión horizontal (HDOP) toma un valor de 1.64, por lo que la precisión obtenida en esta componente será mucho mayor. Adicionalmente, cabe destacar que en el estándar [4] se establece un error vertical de ± 17.25 m en el 95 % de las posiciones obtenidas con VDOP = 2.20, por lo que es de esperar que para un valor de VDOP $\cong 3.5$ el error vertical sea considerablemente superior al especificado en el estándar.

6.3. Posicionamiento de un receptor estático con Galileo

Para poner de manifiesto la flexibilidad que permiten los dispositivos SDR y el software GNSS-SDR se ha implementado un receptor Galileo. Concretamente, se ha empleado la señal Galileo E1, que comparte frecuencia con las señales GPS L1 (1574,42 MHz), por lo que la antena empleada anteriormente puede ser reutilizada. Esta coincidencia de frecuencia no es casual, ya que uno de los objetivos de Galileo era proporcionar compatibilidad e interoperabilidad con otros sistemas de navegación global por satélite, especialmente GPS, por lo que los conceptos fundamentales del mensaje de navegación son similares (almanaque, efemérides, etc.). No obstante, existe una serie de diferencias con la señal GPS L1 C/A que se detallan a continuación:

- Además de la componente de datos, también se transmite una señal piloto, al igual que ocurre en las señales GPS modernizadas.
- Se emplean dos códigos PRN en lugar de un único código. El primer código tiene una longitud de 4092 chips, mientras que el segundo tiene una longitud de 25 chips. A pesar de esta diferencia, la frecuencia del código si coincide con la empleada en las señales GPS L1 C/A (1.023 MHz).
- El mensaje de datos se transmite a una velocidad de 250 sps.

6.3.1. Recepción de la señal Galileo

Teniendo en cuenta los diferentes aspectos expuestos anteriormente se ha realizado la implementación de un receptor de señal Galileo E1 mediante el fichero de configuración del Anexo III.C: Archivo de configuración para la recepción de señal

Galileo E1 con RTL-SDR Blog V3, cuyo funcionamiento se muestra en las siguientes figuras.

```
Current receiver time: 1 s
Tracking of Galileo E1C signal started on channel 0 for satellite Galileo PRN E01 (Block FOC-FM10)
Tracking of Galileo E1C signal started on channel 5 for satellite Galileo PRN E13 (Block FOC-FM20)
Tracking of Galileo E1C signal started on channel 6 for satellite Galileo PRN E14 (Block FOC-FM2*)
Tracking of Galileo E1C signal started on channel 1 for satellite Galileo PRN E21 (Block FOC-FM15)
Current receiver time: 2 s
Current receiver time: 3 s
Tracking of Galileo E1C signal started on channel 2 for satellite Galileo PRN E26 (Block FOC-FM3)
Current receiver time: 4 s
```

Figura 6.13. Inicio del proceso de tracking.

```
Current receiver time: 12 s
Galileo E1C secondary code locked in channel 0 for satellite Galileo PRN E01 (Block FOC-FM10)
Enabled 16 ms extended correlator in channel 0 for satellite Galileo PRN E01 (Block FOC-FM10)
Galileo E1C secondary code locked in channel 6 for satellite Galileo PRN E14 (Block FOC-FM2*)
Enabled 16 ms extended correlator in channel 6 for satellite Galileo PRN E14 (Block FOC-FM2*)
Galileo E1C secondary code locked in channel 5 for satellite Galileo PRN E13 (Block FOC-FM20)
Enabled 16 ms extended correlator in channel 5 for satellite Galileo PRN E13 (Block FOC-FM20)
Galileo E1C secondary code locked in channel 1 for satellite Galileo PRN E21 (Block FOC-FM15)
Enabled 16 ms extended correlator in channel 1 for satellite Galileo PRN E21 (Block FOC-FM15)
Current receiver time: 13 s
Current receiver time: 14 s
Current receiver time: 15 s
Galileo E1C secondary code locked in channel 2 for satellite Galileo PRN E26 (Block FOC-FM3)
Enabled 16 ms extended correlator in channel 2 for satellite Galileo PRN E26 (Block FOC-FM3)
Current receiver time: 16 s
```

Figura 6.14. Inicio de la decodificación del mensaje de datos.

```
Current receiver time: 36 s
New Galileo E1 I/NAV message received in channel 6: ephemeris from satellite Galileo PRN E14 (Block FOC-FM2*)
New Galileo E1 I/NAV message received in channel 0: ephemeris from satellite Galileo PRN E01 (Block FOC-FM10)
New Galileo E1 I/NAV message received in channel 1: ephemeris from satellite Galileo PRN E21 (Block FOC-FM15)
Current receiver time: 37 s
Current receiver time: 38 s
New Galileo E1 I/NAV message received in channel 6: iono/GST model parameters from satellite Galileo PRN E14 (Block FOC-FM2*)
New Galileo E1 I/NAV message received in channel 6: UTC model parameters from satellite Galileo PRN E14 (Block FOC-FM2*)
New Galileo E1 I/NAV message received in channel 1: iono/GST model parameters from satellite Galileo PRN E21 (Block FOC-FM15)
New Galileo E1 I/NAV message received in channel 0: iono/GST model parameters from satellite Galileo PRN E01 (Block FOC-FM10)
New Galileo E1 I/NAV message received in channel 1: UTC model parameters from satellite Galileo PRN E21 (Block FOC-FM15)
New Galileo E1 I/NAV message received in channel 0: UTC model parameters from satellite Galileo PRN E01 (Block FOC-FM10)
Current receiver time: 39 s
```

Figura 6.15. Recepción de los mensajes de datos.

```
Current receiver time: 1 min 36 s
New Galileo E1 I/NAV message received in channel 6: ephemeris from satellite Galileo PRN E14 (Block FOC-FM2*)
New Galileo E1 I/NAV message received in channel 4: ephemeris from satellite Galileo PRN E13 (Block FOC-FM20)
New Galileo E1 I/NAV message received in channel 3: ephemeris from satellite Galileo PRN E26 (Block FOC-FM3)
New Galileo E1 I/NAV message received in channel 1: ephemeris from satellite Galileo PRN E21 (Block FOC-FM15)
New Galileo E1 I/NAV message received in channel 0: ephemeris from satellite Galileo PRN E01 (Block FOC-FM10)
First position fix at 2021-Aug-21 08:52:48.510820 UTC is Lat = 38.0431 [deg], Long = -3.0944 [deg], Height= 510.641 [m]
Current receiver time: 1 min 37 s
Position at 2021-Aug-21 08:52:48.510820 UTC using 5 observations is Lat = 38.043144336 [deg], Long = -3.094384885 [deg], Height = 518.516 [m]
Velocity: East: 0.025 [m/s], North: -0.037 [m/s], Up = 0.072 [m/s]
The RINEX Navigation file header has been updated with UTC and IONO info.
Position at 2021-Aug-21 08:52:49.010820 UTC using 5 observations is Lat = 38.043121758 [deg], Long = -3.094374123 [deg], Height = 521.300 [m]
Velocity: East: -0.011 [m/s], North: 0.012 [m/s], Up = 0.093 [m/s]
Current receiver time: 1 min 38 s
Galileo E1C secondary code locked in channel 5 for satellite Galileo PRN E08 (Block FOC-FM8)
Enabled 16 ms extended correlator in channel 5 for satellite Galileo PRN E08 (Block FOC-FM8)
Position at 2021-Aug-21 08:52:49.510820 UTC using 5 observations is Lat = 38.043145634 [deg], Long = -3.094381868 [deg], Height = 506.520 [m]
Velocity: East: 0.012 [m/s], North: -0.025 [m/s], Up = 0.063 [m/s]
```

Figura 6.16. Obtención de la primera posición fija.

```

New Galileo E1 I/NAV message received in channel FOC-FM15)
0: iono/GST model parameters from satellite Galileo PRN E01 (Block FOC-FM10)
New Galileo E1 I/NAV message received in channel 1: UTC model parameters from satellite Galileo PRN E21 (Block FOC-FM15)
New Galileo E1 I/NAV message received in channel 4: UTC model parameters from satellite Galileo PRN E13 (Block FOC-FM20)
New Galileo E1 I/NAV message received in channel 0: UTC model parameters from satellite Galileo PRN E01 (Block FOC-FM10)
Position at 2021-Aug-21 08:53:20.010818 UTC using 6 observations is Lat = 38.043178397 [deg], Long = -3.094433731 [deg], Height = 480.332 [m]
Velocity: East: 0.018 [m/s], North: 0.008 [m/s], Up = 0.075 [m/s]
Current receiver time: 2 min 9 s
Position at 2021-Aug-21 08:53:20.510818 UTC using 6 observations is Lat = 38.043186233 [deg], Long = -3.094453767 [deg], Height = 480.543 [m]
Velocity: East: 0.001 [m/s], North: 0.018 [m/s], Up = 0.016 [m/s]
Position at 2021-Aug-21 08:53:21.010818 UTC using 6 observations is Lat = 38.043185048 [deg], Long = -3.094436000 [deg], Height = 480.237 [m]
Velocity: East: 0.012 [m/s], North: 0.043 [m/s], Up = 0.018 [m/s]
Current receiver time: 2 min 10 s
Position at 2021-Aug-21 08:53:21.510818 UTC using 6 observations is Lat = 38.043170530 [deg], Long = -3.094412254 [deg], Height = 486.203 [m]
Velocity: East: -0.004 [m/s], North: 0.037 [m/s], Up = -0.049 [m/s]

```

Figura 6.17. Actualización de la posición.

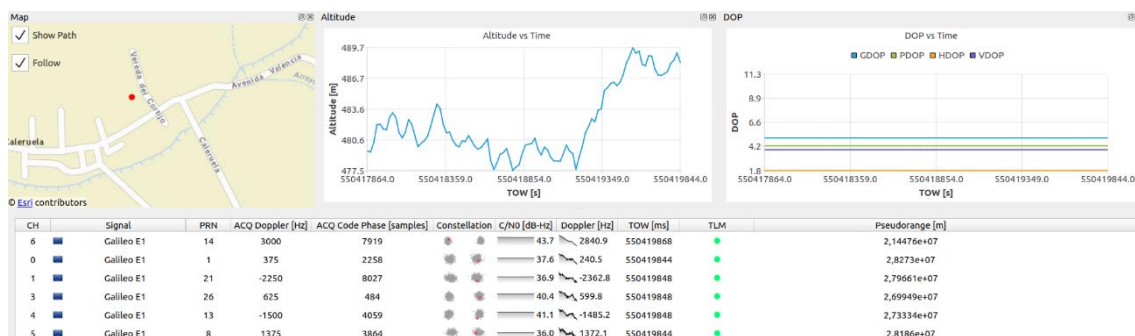


Figura 6.18. Interfaz de GNSS-SDR-Monitor durante la ejecución del receptor.

6.3.2. Representación de los resultados de los bloques intermedios

Al igual que ocurre con los resultados de las señales GPS, también es posible representar los resultados obtenidos en los bloques intermedios mediante las mismas funciones de MATLAB citadas anteriormente.

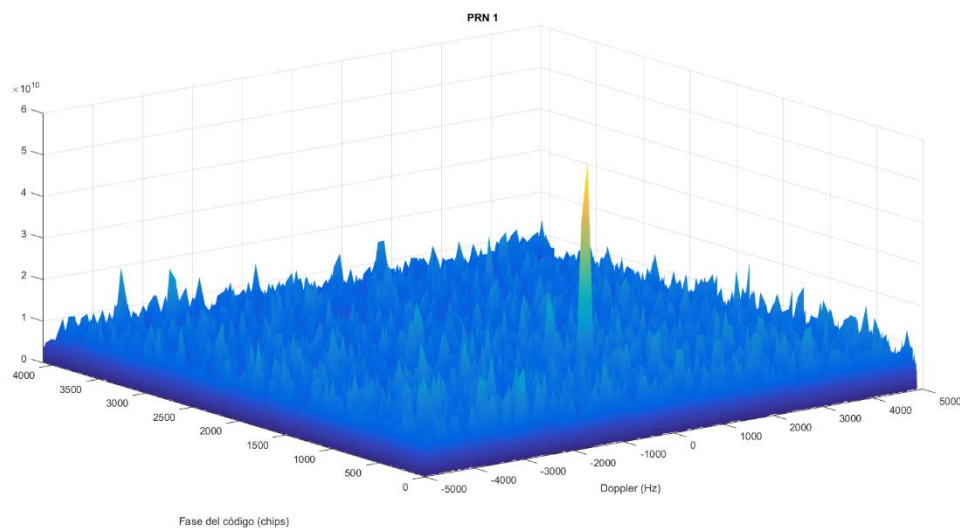


Figura 6.19. Gráfico de adquisición obtenido para el PRN 1.

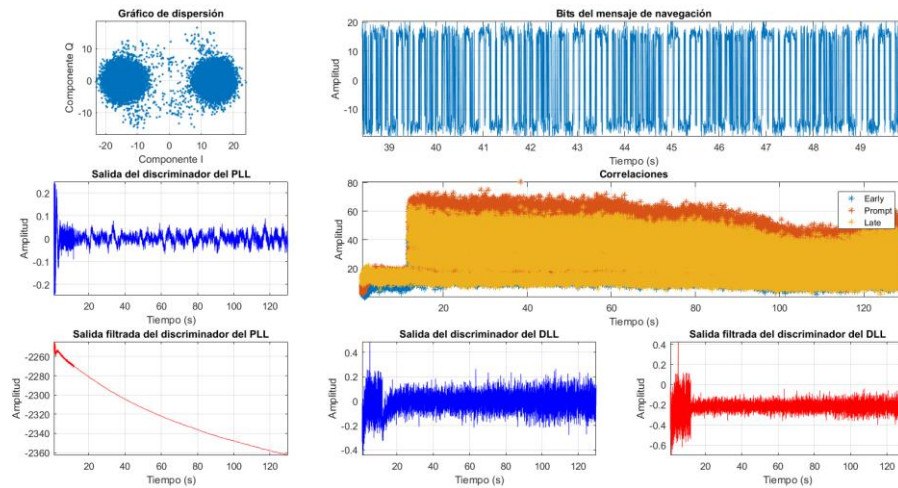


Figura 6.20. Resultados del proceso de tracking para el canal 1.

Comparando estos resultados con los obtenidos para la señal GPS se puede observar que ambos son muy similares. No obstante, cabe destacar que para este caso el resultado de la correlación de la señal de entrada con la réplica de código alineada (*Prompt*) es superior al obtenido para cualquiera de las otras réplicas, por lo que en este caso el proceso de *tracking* es de mayor calidad. Adicionalmente, cabe destacar que la representación de los símbolos del mensaje de navegación ha sido recortada en el tiempo, puesto que si se hubiese representado toda la secuencia no hubiese sido posible percibirlos.

6.3.3. Análisis de las posiciones obtenidas

Finalmente, se procederá a representar las distintas posiciones obtenidas con la señal Galileo E1, para posteriormente determinar el error en el posicionamiento siguiendo la misma metodología que en la recepción de señal GPS L1 C/A.



Figura 6.21. Representación de las posiciones obtenidas en Google Earth.

Posición	Latitud (°)	Longitud (°)	Altitud (m)
1	38.04312	-3.09437	521.30021
2	38.04314	-3.09438	504.84570
3	38.04313	-3.09437	498.65260
4	38.04316	-3.09439	486.71324
5	38.04315	-3.09440	481.64877
6	38.04315	-3.09441	479.87352
7	38.04315	-3.09442	493.37858
8	38.04315	-3.09444	499.86888
9	38.04313	-3.09444	493.20911
10	38.04312	-3.09443	493.46948
11	38.04313	-3.09442	476.88107
12	38.04314	-3.09444	464.80735
13	38.04313	-3.09444	472.99852
14	38.04314	-3.09447	471.85548
15	38.04313	-3.09445	477.78449
16	38.04314	-3.09446	470.05572
17	38.04314	-3.09446	467.38590
18	38.04315	-3.09447	463.43682
19	38.04313	-3.09445	465.06525
20	38.04312	-3.09445	463.26437
21	38.04315	-3.09445	476.74676
22	38.04316	-3.09444	464.76507
23	38.04314	-3.09443	476.11128
24	38.04318	-3.09445	457.78790
25	38.04317	-3.09443	458.06759
26	38.04318	-3.09444	464.36634
27	38.04317	-3.09444	469.75680
28	38.04318	-3.09445	464.59946
29	38.04318	-3.09444	462.16013
30	38.04318	-3.09442	488.93887
31	38.04318	-3.09445	487.26000
32	38.04318	-3.09443	480.33221
33	38.04318	-3.09444	480.23680

Tabla 6.5. Coordenadas de las posiciones obtenidas con el receptor.

Posición	Error horizontal (m) (latitud y longitud)	Error vertical (m) (altitud)
1	9.99	86.53
2	7.98	70.07
3	10.37	63.88
4	7.20	51.94
5	6.34	46.88
6	4.42	45.10
7	3.95	58.61
8	2.24	65.10
9	5.15	58.44
10	6.34	58.70
11	5.08	42.11
12	3.35	30.03
13	4.39	38.23
14	3.57	37.08
15	4.06	43.01
16	3.84	35.28
17	3.55	32.61
18	3.09	28.66
19	4.77	30.30
20	5.19	28.49
21	2.46	41.97
22	2.22	29.99
23	4.16	41.34
24	1.23	23.01
25	1.99	23.29
26	1.71	29.59
27	0.86	34.98
28	1.54	29.83
29	1.38	27.38
30	3.11	54.17
31	1.20	52.49
32	2.04	45.56
33	2.28	45.46

Tabla 6.6. Error cometido en las posiciones obtenidas.

	Error horizontal (m) (latitud y longitud)	Error vertical (m) (altitud)
Error (media $\pm$ desviación típica)	3.97 $\pm$ 2.4	43.34 $\pm$ 15

Tabla 6.7. Error medio.

Comparando estos resultados con los obtenidos para la señal GPS L1 C/A se puede observar que, aunque el error vertical es bastante similar, el posicionamiento horizontal obtenido con la señal Galileo E1 es de mejor precisión. Esto es debido a que esta señal es una señal moderna que contiene una serie de nuevas características que hace posible este posicionamiento de mayor precisión por un mismo receptor.

7. CONCLUSIONES Y LÍNEAS FUTURAS

Desde que en los años 70 el ejército estadounidense desarrolló el sistema de posicionamiento global se han realizado multitud de investigaciones en torno a los receptores GPS. En un inicio, estos receptores fueron implementados principalmente mediante hardware, pero gracias a los distintos avances tecnológicos ha sido posible el desarrollo de receptores digitales. Los receptores GPS digitales basados en software proporcionan al usuario una herramienta de gran utilidad y flexibilidad, con el añadido de una interesante característica; la capacidad de reprogramación.

Durante el desarrollo de este proyecto se ha conseguido implementar un receptor GPS L1 C/A mediante una plataforma SDR que permite realizar una representación gráfica de la posición obtenida. Para ello ha sido necesario el empleo de las herramientas software GNSS-SDR y GNSS-SDR-Monitor.

Dado que el dispositivo bladeRF x40 proporciona una estabilidad en frecuencia insuficiente para la recepción de señales GPS, ha sido necesario adquirir la plataforma RTL-SDR Blog V3, la cual, pese a su bajo coste, ha permitido una implementación exitosa del receptor. Una vez se ha implementado el receptor, se han puesto a prueba el funcionamiento del mismo bajo distintas configuraciones, obteniéndose una precisión de aproximadamente 10 m para el posicionamiento horizontal y 40 m para el posicionamiento vertical. Además, también se ha puesto en práctica la flexibilidad de los dispositivos SDR mediante la implementación de un receptor Galileo E1, que permite un posicionamiento de mayor calidad.

Aunque implementación del receptor se ha realizado correctamente existen múltiples aspectos que pueden ser abordados en el futuro. Uno de los más evidente, es el almacenamiento de los datos obtenidos en ficheros RINEX (que puede realizarse mediante GNSS-SDR), para un posterior procesamiento de la señal que permita la reducción de la influencia de las distintas fuentes de error dando lugar a un posicionamiento de mayor exactitud.

Si bien las señales GPS son ampliamente usadas para la navegación y el posicionamiento existen muchos otros ámbitos donde estas señales pueden ser utilizadas. Por tanto, otra línea de interés para el desarrollo de futuros trabajos podría ser el empleo de un receptor GPS para el estudio de la atmósfera. El principio que subyace a estas técnicas es la utilización de los efectos de la propagación atmosférica sobre las señales GNSS en su camino desde los satélites hasta los receptores en tierra. Por tanto, un análisis adecuado de las señales recibidas puede utilizarse para mapear el contenido variable de electrones de la ionosfera y la cantidad de vapor de agua en la troposfera. De hecho, varios servicios meteorológicos emplean las mediciones de vapor

de agua determinadas por un GNSS para mejorar sus previsiones meteorológicas. Además, puesto que los terremotos y los tsunamis provocan variaciones en la densidad de electrones de la ionosfera, GNSS puede convertirse en un componente importante de los sistemas de alerta de tsunamis [1].

8. ANEXOS

8.1. Anexo I: Manual de instalación

La instalación de las distintas herramientas software empleadas se ha realizado a través de la terminal del terminal de Ubuntu mediante PyBOMBS. PyBOMBS es un gestor de paquetes y un sistema de construcción para GNU Radio capaz de gestionar la construcción e instalación de GNU Radio y sus dependencias. Por tanto, en primer lugar es necesario realizar la instalación de PyBOMBS para poder instalar el resto de herramientas.

Antes de instalar PyBOMBS es necesario instalar las siguientes dependencias:

```
$ sudo apt-get install python3-six
$ sudo apt-get install python-six
$ sudo apt-get install python3-mako
$ sudo apt-get install python3-lxml
$ sudo apt-get install python3-numpy
$ sudo apt-get install python3-pip
$ sudo apt-get install python3-pybind11
$ sudo apt-get install git
$ sudo apt-get install libsndfile1-dev
```

Tras instalar las dependencias, se debe añadir directorio de instalación de PyBOMBS a la variable PATH:

```
$ nano ~/.bashrc
$ export PATH=$PATH:$HOME/.local/bin
```

Seguidamente, se procede a instalar PyBOMBS aplicando la configuración y recetas por defecto:

```
$ pip3 install --upgrade git+https://github.com/gnuradio/pybombs.git
$ pybombs auto-config
$ pybombs recipes add-defaults
```

Dado que se desea usar el paquete *gr-osmosdr* con bladeRF x40 es necesario ejecutar los siguientes comandos:

```
$ sed -i 's/git.osmocom.org/github.com\/Nuand/g'  
~/.pybombs/recipes/gr-recipes/gr-osmosdr.lwr  
$ sed -i '/- gr.\{7,\}/d' ~/.pybombs/recipes/gr-  
recipes/grosmosdr.lwr
```

A continuación, se creará un directorio en el que instalar todos los componentes de GNU Radio. En este caso, se establece el directorio `~/pybombs/TFG`. Además, también se especifica el alias, `TFG`, para facilitar referencias posteriores.

```
$ mkdir ~/pybombs/  
$ pybombs config --package gnuradio gitrev v3.9.0.0  
$ pybombs config --package gr-osmosdr gitrev dev-gr-3.9  
$ pybombs prefix init ~/pybombs/TFG -a TFG -R gnuradio-default
```

En este punto GNU radio ha sido instalado en el directorio especificado. Seguidamente, se instalarán los paquetes *gr-osmosdr* y *gr-iqbal*, además del software *gqrx*:

```
$ pybombs -p TFG install gr-iqbal gr-osmosdr gqrx
```

El paquete *gr-iqbal* se emplea para mejorar la recepción de señales *IQ*, mientras que el paquete *gr-osmosdr* permite el uso de diferentes SDRs en GNU-Radio, además de realizar la instalación de sus controladores. Por otro lado, *gqrx* permite monitorizar distintos aspectos de la señal recibida para una gran variedad de dispositivos SDR. De hecho, este es el software empleado para la obtención de la Figura 5.7.

Puesto que PyBOMBS permite a los usuarios realizar instalaciones como usuarios no root se emplea el comando `-DINSTALL_UDEV_RULES=OFF` al configurar el proyecto *bladeRF* con *CMake*. Por lo tanto, es necesario instalar manualmente las reglas *udev* para tener acceso al dispositivo. Para ello es necesario seguir los siguientes pasos:

1. Copie cada archivo [88-nuand-*.rules.in](#) en un nuevo fichero llamado `88-nuand-*.rules`.
2. Edite el fichero creado reemplazando cada aparición del parámetro `GROUP="@BLADERF_GROUP@"` por `GROUP="plugdev"`.
3. Copie este archivo al directorio de normas *udev*:

```
$ sudo cp 88-nuand-*.rules /etc/udev/rules.d/ && sudo chmod 644 /etc/udev/rules.d/88-nuand-*.rules
```

4. Recargue las normas:

```
$ sudo udevadm control --reload-rules && sudo udevadm trigger
```

Una vez se ha configurado el acceso al dispositivo bladeRF x40 es necesario instalar el firmware y cargar la FPGA:

```
$ sudo apt-get install bladerf-firmware-fx3  
$ sudo apt-get install bladerf-fpga-hostedx40
```

Seguidamente se procede a instalar GNSS-SDR. Si se instala GNSS-SDR mediante PyBOMBS no se habilita el empleo del bloque gr-osmosdr. Para solucionar esto es necesario añadir la etiqueta `-DENABLE_OSMOSDR=ON` en la receta de GNSS-SDR para después instalar el software mediante PyBOMBS:

```
$ printf "vars:\n  config_opt: \"-DENABLE_OSMOSDR=ON\"\n" >>  
$HOME/.pybombs/recipes/gr-recipes/gnss-sdr.lwr  
$ pybombs -p TFG install gnss-sdr
```

En este punto ya se encuentra instalado GNSS-SDR. No obstante, si se desean obtener las ventajas de los ajustes SIMD, se deben ejecutar los siguientes comandos para las librerías *VOLK* y *VOLK-GNSSSDR*.

```
$ volk_profile  
$ volk_gnssdr_profile
```

Finalmente, se instala GNSS-SDR-Monitor, iniciando este proceso mediante la instalación de las dependencias:

```
$ sudo apt install build-essential cmake git libboost-dev libboost-system-dev libprotobuf-dev protobuf-compiler qtbase5-dev qtdeclarative5-dev qtpositioning5-dev libqt5charts5-dev qml-module-qtquick2 qml-module-qtquick-controls2 qml-module-qtquick-window2 qml-
```

```
module-qtlocation      qml-module-qtpositioning  qml-module-qtquick-  
layouts
```

Tras esto solo queda clonar el repositorio, montarlo y proceder a su instalación:

```
$ git clone https://github.com/acebrianjuan/gnss-sdr-monitor  
$ cd gnss-sdr-monitor/build  
$ cmake ..  
$ make
```

8.2. Anexo II: Manual de usuario

Del mismo modo que ocurría en la instalación, para poder ejecutar el receptor GPS es necesario emplear la terminal de Linux.

El primer paso a realizar para la ejecución de receptor es la correcta configuración de GNSS-SDR-Monitor, puesto que si se desea utilizar este software es esencial activar los flujos de datos de serialización *Telecommnad* y *Monitor* en el archivo de configuración del receptor GNSS-SDR empleando los parámetros relacionados con estos bloques, los cuales se encuentran detallados en el capítulo 5.3.1. Después de realizar esta configuración se procede a abrir el programa:

```
$ cd gnss-sdr-monitor/build/src/  
$ ./gnss-sdr-monitor
```

Una vez se ha abierto el programa es fundamental asegurarse de que su configuración coincide con la del archivo de configuración de GNSS-SDR. Para ello en la barra del menú se accede a *Editar > Preferencias*, y se verifica que los puertos empleados son correctos.

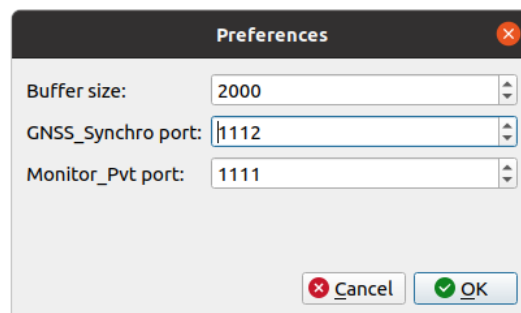


Figura 8.1. Ajuste de preferencias en GNSS-SDR-Monitor.

Además, también es necesario especificar la dirección IP y el puerto TCP empleados en el bloque *Telecommand*.

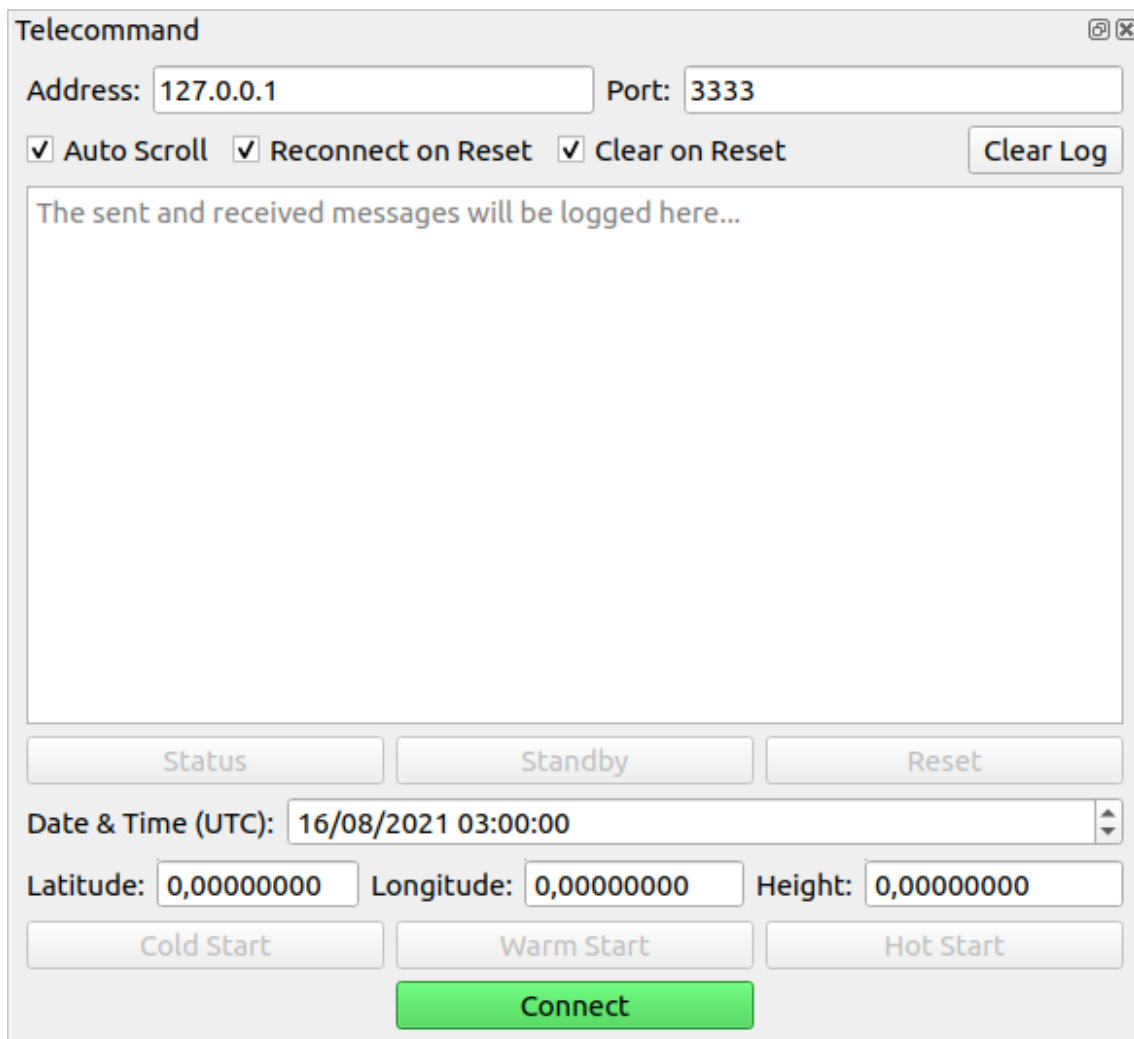


Figura 8.2. Interfaz de Telecommand.

La funcionalidad *Telecommand* permite al usuario un control intuitivo del receptor. Para activarla únicamente es necesario hacer clic sobre el botón *Connect* mientras que GNSS-SDR está en ejecución. Tras esto, es posible comprobar el estado del receptor (*Status*), pausar su recepción (*Standby*) o finalizarla borrando todos los datos obtenidos (*Reset*). Además, esta funcionalidad también permite iniciar el receptor en sus distintos modos de arranque (*Cold Start*, *Warm Start* y *Hot Start*).

Tras realizar la configuración de GNSS-SDR-Monitor es posible ejecutar el receptor a través de GNSS-SDR con el fichero de configuración creado:

```
$ source ~/pybombs/TFG/setup_env.sh
$ gnss-sdr --config_file=[directorio]/[archivo]
```

Una vez ejecutados los comandos anteriores se obtendría en el terminal la información del proceso de recepción de las señales hasta la obtención de la posición del receptor. Para finalizar esta ejecución, únicamente es necesario pulsar una vez la letra *q* y posteriormente pulsar *intro*,

8.3. Anexo III: Archivos de configuración

8.3.1. Anexo III.A: Archivo de configuración para recepción de señal GPS L1 C/A con bladeRF x40

[GNSS-SDR]

```
;##### GLOBAL OPTIONS #####
```

```
GNSS-SDR.internal_fs_sps = 2000000
```

```
;##### SIGNAL_SOURCE CONFIG #####
```

```
SignalSource.implementation = Osmosdr_Signal_Source
```

```
SignalSource.item_type = gr_complex
```

```
SignalSource.sampling_frequency = 2000000
```

```
SignalSource.freq = 1575420000
```

```
;# RF Gain: LNA Gain {0, 3, 6}
```

```
;SignalSource.gain=6
```

```
;# IF Gain: N/A
```

```
;SignalSource.rf_gain=40
```

```
;# BB Gain: RX VGA1 + VGA2 [5, 60]
```

```
;SignalSource.if_gain=48
```

```
SignalSource.AGC_enabled = false
```

```
SignalSource.samples = 0
```

```
SignalSource.repeat = false
```

```
SignalSource.osmosdr_args = bladerf=0
```

```
SignalSource.enable_throttle_control = false
```

```
SignalSource.dump = false
```

```
SignalSource.dump_filename = ./signal_source.dat
```

```

;##### SIGNAL_CONDITIONER CONFIG #####

SignalConditioner.implementation = Signal_Conditioner

;##### DATA_TYPE_ADAPTER CONFIG #####

DataAdapter.implementation = Pass_Through

;##### INPUT_FILTER CONFIG #####

InputFilter.implementation = Pass_Through

;##### RESAMPLER CONFIG #####

Resampler.implementation = Pass_Through

;##### CHANNELS GLOBAL CONFIG #####

Channels_1C.count = 8
Channels.in_acquisition = 1

;##### GPS ACQUISITION CONFIG #####

Acquisition_1C.implementation = GPS_L1_CA_PCPS_Acquisition
Acquisition_1C.item_type = gr_complex
Acquisition_1C.coherent_integration_time_ms = 1
Acquisition_1C.threshold = 2.3
Acquisition_1C.doppler_max = 5000
Acquisition_1C.doppler_step = 250
Acquisition_1C.max_dwells = 5
Acquisition_1C.dump = true
Acquisition_1C.dump_filename = ./acq_dump.dat

;##### TRACKING GPS CONFIG #####

Tracking_1C.implementation = GPS_L1_CA_DLL_PLL_Tracking

```



```

Tracking_1C.item_type = gr_complex
Tracking_1C.pll_bw_hz = 40.0
Tracking_1C.dll_bw_hz = 2.0
Tracking_1C.order = 3
Tracking_1C.early_late_space_chips = 0.5
Tracking_1C.dump = false
Tracking_1C.dump_filename = ./tracking_ch.dat

;##### TELEMETRY DECODER GPS CONFIG #####

TelemetryDecoder_1C.implementation = GPS_L1_CA_Telemetry_Decoder
TelemetryDecoder_1C.dump = false

;##### OBSERVABLES CONFIG #####

Observables.implementation = Hybrid_Observables
Observables.dump = false
Observables.dump_filename = ./observables.dat

;##### PVT CONFIG #####

PVT.implementation = RTKLIB_PVT
PVT.positioning_mode = Single
PVT.output_rate_ms = 100
PVT.display_rate_ms = 500
PVT.iono_model = Broadcast
PVT.trop_model = Saastamoinen
PVT.flag_rtcn_server = false
PVT.flag_rtcn_tty_port = false
PVT.rtcn_dump_devname = /dev/pts/1
PVT.rtcn_tcp_port = 2101
PVT.rtcn_MT1019_rate_ms = 5000
PVT.rtcn_MT1077_rate_ms = 1000

```

8.3.2. Anexo III.B: Archivo de configuración para la recepción de señal GPS L1 C/A con RTL-SDR Blog V3

[GNSS-SDR]

GNSS-SDR.AGNSS_XML_enabled = true

;##### GLOBAL OPTIONS #####

GNSS-SDR.internal_fs_sps = 2048000

GNSS-SDR.telecommand_enabled = true

GNSS-SDR.telecommand_tcp_port = 3333

;##### SIGNAL_SOURCE CONFIG #####

SignalSource.implementation = Osmosdr_Signal_Source

SignalSource.item_type = gr_complex

SignalSource.sampling_frequency = 2048000

SignalSource.freq = 1575420000

SignalSource.gain = 40.2

SignalSource.rf_gain = 40.2

SignalSource.if_gain = 29.7

SignalSource.AGC_enabled = false

SignalSource.samples = 0

SignalSource.repeat = false

SignalSource.osmosdr_args = rtl,bias=1

SignalSource.dump = false

SignalSource.dump_filename = ./data/signal_source.dat

SignalSource.enable_throttle_control = false

;##### SIGNAL_CONDITIONER CONFIG #####

SignalConditioner.implementation = Signal_Conditioner

;##### DATA_TYPE_ADAPTER CONFIG #####

```

DataAdapter.implementation = Pass_Through

;##### INPUT_FILTER CONFIG #####

InputFilter.implementation = Pass_Through

;##### RESAMPLER CONFIG #####

Resampler.implementation = Pass_Through

;##### CHANNELS GLOBAL CONFIG #####

Channels_1C.count = 8
Channels.in_acquisition = 1

;##### GPS ACQUISITION CONFIG #####

Acquisition_1C.implementation = GPS_L1_CA_PCPS_Acquisition
Acquisition_1C.item_type = gr_complex
Acquisition_1C.coherent_integration_time_ms = 1
Acquisition_1C.threshold = 2.3
Acquisition_1C.doppler_max = 5000
Acquisition_1C.doppler_step = 250
Acquisition_1C.max_dwells = 5
Acquisition_1C.dump = true
Acquisition_1C.dump_filename = ./acq_dump.dat

;##### TRACKING GPS CONFIG #####

Tracking_1C.implementation = GPS_L1_CA_DLL_PLL_Tracking
Tracking_1C.item_type = gr_complex
Tracking_1C.order = 2
Tracking_1C.carrier_aiding = true
Tracking_1C.enable_fll_pull_in = true
Tracking_1C.pull_in_time_s = 3
Tracking_1C.fll_bw_hz = 100

```

```

Tracking_1C.pll_filter_order = 2
Tracking_1C.pll_bw_hz = 100.0
Tracking_1C.dll_filter_order = 1
Tracking_1C.dll_bw_hz = 8.0
Tracking_1C.early_late_space_chips = 0.5
Tracking_1C.extend_correlation_symbols = 20
Tracking_1C.cn0_min = 15
Tracking_1C.max_lock_fail = 200
Tracking_1C.dump = true
Tracking_1C.dump_filename = ./track_ch.dat

;##### TELEMETRY DECODER GPS CONFIG #####

TelemetryDecoder_1C.implementation = GPS_L1_CA_Telemetry_Decoder
TelemetryDecoder_1C.dump = true
TelemetryDecoder_1C.dump_filename = /.telemetry.dat

;##### OBSERVABLES CONFIG #####

Observables.implementation = Hybrid_Observables
Observables.dump = true
Observables.dump_filename = ./observables.dat

;##### PVT CONFIG #####

PVT.implementation = RTKLIB_PVT
PVT.positioning_mode = Static
PVT.iono_model = Broadcast ;
PVT.trop_model = Saastamoinen ;
PVT.output_rate_ms = 500
PVT.display_rate_ms = 500
PVT.output_enabled = true
PVT.gpx_output_enabled = false
PVT.geojson_output_enabled = false
PVT.kml_output_enabled = true
PVT.xml_output_enabled = true

```

```
PVT.enable_monitor = true
PVT.monitor_client_addresses = 127.0.0.1
PVT.monitor_udp_port = 1111
```

```
;##### MONITOR CONFIG #####
```

```
Monitor.enable_monitor = true
Monitor.decimation_factor = 1
Monitor.client_addresses = 127.0.0.1
Monitor.udp_port = 1112
```

8.3.3. Anexo III.C: Archivo de configuración para la recepción de señal Galileo E1 con RTL-SDR Blog V3

```
[GNSS-SDR]
```

```
GNSS-SDR.AGNSS_XML_enabled = true
```

```
;##### GLOBAL OPTIONS #####
```

```
GNSS-SDR.internal_fs_sps = 2048000
GNSS-SDR.telecommand_enabled = true
GNSS-SDR.telecommand_tcp_port = 3333
```

```
;##### SIGNAL_SOURCE CONFIG #####
```

```
SignalSource.implementation = Osmosdr_Signal_Source
SignalSource.item_type = gr_complex
SignalSource.sampling_frequency = 2048000
SignalSource.freq = 1575420000
SignalSource.gain = 40.2
SignalSource.rf_gain = 40.2
SignalSource.if_gain = 29.7
SignalSource.AGC_enabled = false
SignalSource.samples = 0
```

```

SignalSource.repeat = false
SignalSource.osmosdr_args = rtl,bias=1
SignalSource.dump = false
SignalSource.dump_filename = ./data/signal_source.dat
SignalSource.enable_throttle_control = false

;##### SIGNAL_CONDITIONER CONFIG #####

SignalConditioner.implementation = Signal_Conditioner

;##### DATA_TYPE_ADAPTER CONFIG #####

DataAdapter.implementation = Pass_Through

;##### INPUT_FILTER CONFIG #####

InputFilter.implementation = Pass_Through

;##### RESAMPLER CONFIG #####

Resampler.implementation = Pass_Through

;##### CHANNELS GLOBAL CONFIG #####

Channels_1B.count = 8
Channels.in_acquisition = 1

;##### GALILEO ACQUISITION CONFIG #####

Acquisition_1B.implementation = Galileo_E1_PCPS_Ambiguous_Acquisition
Acquisition_1B.item_type = gr_complex
Acquisition_1B.coherent_integration_time_ms = 4
Acquisition_1B.acquire_pilot = true
Acquisition_1B.threshold = 2.5
Acquisition_1B.doppler_max = 5000
Acquisition_1B.doppler_step = 125

```

```

Acquisition_1B.bit_transition_flag = true
Acquisition_1B.dump = true
Acquisition_1B.dump_filename = ./acq_dump.dat

;##### TRACKING GALILEO CONFIG #####

Tracking_1B.implementation = Galileo_E1_DLL_PLL_VEML_Tracking
Tracking_1B.item_type = gr_complex
Tracking_1B.track_pilot = true
Tracking_1B.pll_bw_hz = 7.5
Tracking_1B.dll_bw_hz = 0.5
Tracking_1B.pll_bw_narrow_hz = 2.5
Tracking_1B.dll_bw_narrow_hz = 0.25
Tracking_1B.extend_correlation_symbols = 4
Tracking_1B.order = 3
Tracking_1B.early_late_space_chips = 0.15
Tracking_1B.very_early_late_space_chips = 0.6
Tracking_1B.early_late_space_narrow_chips = 0.15
Tracking_1B.very_early_late_space_narrow_chips = 0.30
Tracking_1B.dump = true
Tracking_1B.dump_filename = ./track_ch.dat

;##### TELEMETRY DECODER GALILEO CONFIG #####

TelemetryDecoder_1B.implementation = Galileo_E1B_Telemetry_Decoder
TelemetryDecoder_1B.dump = false

;##### OBSERVABLES CONFIG #####

Observables.implementation = Hybrid_Observables
Observables.dump = true
Observables.dump_filename = ./observables.dat

;##### PVT CONFIG #####

PVT.implementation = RTKLIB_PVT

```

```
PVT.positioning_mode = Static
PVT.iono_model = Broadcast
PVT.trop_model= Saastamoinen
PVT.output_rate_ms = 500
PVT.display_rate_ms = 500
PVT.output_enabled = true
PVT.gpx_output_enabled = false
PVT.geojson_output_enabled = false
PVT.kml_output_enabled = true
PVT.xml_output_enabled = true
PVT.enable_monitor = true
PVT.monitor_client_addresses = 127.0.0.1
PVT.monitor_udp_port = 1111
```

```
;##### MONITOR CONFIG #####
```

```
Monitor.enable_monitor = true
Monitor.decimation_factor = 1
Monitor.client_addresses = 127.0.0.1
Monitor.udp_port = 1112
```


BIBLIOGRAFÍA

- [1] Teunissen, P. J.G., Montenbruck, O. *Springer Handbook of Global Navigation Satellite Systems*. Cham: Springer, 2017.
- [2] Doberstein, D. *Fundamentals of GPS Receivers: A Hardware Approach*. New York: Springer, 2012.
- [3] Kaplan, E.D., Hegarty, C.J. *Understanding GPS/GNSS: Principles and Applications, Third Edition*. Boston: ARTECH HOUSE, 2017.
- [4] *Global Positioning System Standard Position Service Performance Standard, 5th Edition*, 2020. URL: <https://www.gps.gov/technical/ps/2020-SPS-performance-standard.pdf>
- [5] Borre, K., Akos, D.M., Bertelsen, N., Rinder, P., Jensen, S.H. *A Software-Defined GPS and Galileo Receiver: A Single-Frequency Approach*. Boston: Birkhäuser, 2007.
- [6] Bao-Yen Tsui, J. *Fundamentals of Global Positioning System Receivers: A Software Approach*. New York: John Wiley & Sons, 2000.
- [7] Principe, F., Bacci, G., Giannetti, F., Luise, M. *Software-Defined Radio Technologies for GNSS Receivers: A Tutorial Approach to a Simple Design and Implementation. International Journal of Navigation and Observation*, 2011, vol. 2011. URL: <https://www.hindawi.com/journals/ijno/2011/979815>
- [8] Machado-Fernández, J.R. *Software Defined Radio: Basic Principles and Applications. Revista Facultad de Ingeniería*, 2015, 24.
- [9] Grayver, E. *Implementing Software Defined Radio*. New York: Springer, 2013.
- [10] Collins, T.F., Getz, R., Pu, D., Wyglinski, A.M. *Software-Defined Radio for Engineers*. Boston: ARTECH HOUSE, 2018.
- [11] *bladeRF USB 3.0 Software Defined Radio*. Nuand. URL: <https://www.nuand.com/bladeRF-brief.pdf>

- [12] *RTL-SDR Blog V3 Datasheet*. URL: <https://www.rtl-sdr.com/wp-content/uploads/2018/02/RTL-SDR-Blog-V3-Datasheet.pdf>
- [13] *HackRF One Official Web Site*. URL: <https://greatscottgadgets.com/hackrf/>
- [14] Fernández-Prades, C., Arribas, J., Closas, P. *GNSS-SDR: An open source tool for researchers and developers*. ION GNSS Conference, 2011.
- [15] *GNSS-SDR Official Web Site*. URL: <https://gnss-sdr.org/>
- [16] *GNSS-SDR-Monitor GitHub Page*. URL: <https://github.com/acebrianjuan/gnss-sdr-monitor>
- [17] *Mike 3A Weather Resistant GPS Magnetic Mount Antenna*. URL: https://www.siretta.com/?smd_process_download=1&download_id=2502
- [18] Kelly, R. *bladeRF Micro SDR Setup: Setting up the bladeRF xA4 for GNU Radio and GNSS-SDR*, 2020.