



UNIVERSIDAD TÉCNICA DEL NORTE

FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CARRERA DE INGENIERÍA EN ELECTRÓNICA Y REDES DE COMUNICACIÓN

TRABAJO DE GRADO PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO

EN ELECTRÓNICA Y REDES DE COMUNICACIÓN

TEMA:

“DISEÑO DE UN SISTEMA DE RECEPCIÓN DE SEÑALES LTE 4G UTILIZANDO RADIO DEFINIDO POR SOFTWARE, DIRIGIDO A LA CARRERA DE INGENIERÍA EN TELECOMUNICACIONES (CITEL) DE LA UNIVERSIDAD TÉCNICA DEL NORTE, PARA EL DESARROLLO DE PRÁCTICAS DE LABORATORIO DE LA ASIGNATURA DE RADIO DEFINIDO POR SOFTWARE (SDR)”

AUTOR: ROSERO CARAGOLLA ROMEL RICARDO

DIRECTOR: MSc. VÁSQUEZ AYALA CARLOS ALBERTO

IBARRA – ECUADOR

2023



UNIVERSIDAD TÉCNICA DEL NORTE

BIBLIOTECA UNIVERSITARIA

AUTORIZACIÓN DE USO Y PUBLICACIÓN

A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE

1. IDENTIFICACIÓN DE LA OBRA

En cumplimiento del Art. 144 de la Ley de Educación Superior, hago la entrega del presente trabajo a la Universidad Técnica del Norte para que sea publicado en el Repositorio Digital Institucional, para lo cual pongo a disposición la siguiente información:

DATOS DE CONTACTO			
CÉDULA DE IDENTIDAD:	1003104443		
APELLIDOS Y NOMBRES:	Rosero Caragolla Romel Ricardo		
DIRECCIÓN:	Atuntaqui, Andrade Marín, Calle Dalmau y Junín		
EMAIL:	rrroseroc@utn.edu.ec		
TELÉFONO FIJO:	2 530 204	TELÉFONO MÓVIL:	0960290842

DATOS DE LA OBRA	
TÍTULO:	“Diseño de un sistema de recepción de señales LTE 4G utilizando Radio Definido por Software, dirigido a la

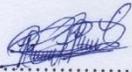
	carrera de Ingeniería en Telecomunicaciones (CITEL) de la Universidad Técnica del Norte, para el desarrollo de prácticas de laboratorio de la asignatura de Radio Definido por Software (SDR)”
AUTOR (ES):	Rosero Caragolla Romel Ricardo
FECHA DE APROBACIÓN: DD/MM/AAAA	16/02/2022
PROGRAMA:	<input checked="" type="checkbox"/> PREGRADO <input type="checkbox"/> POSGRADO
TITULO POR EL QUE OPTA:	Ingeniero en Electrónica y Redes de Comunicación
ASESOR /DIRECTOR:	MSc. Carlos Alberto Vásquez Ayala

2. CONSTANCIAS

El autor manifiesta que la obra objeto de la presente autorización es original y se la desarrolló, sin violar derechos de autor de terceros, por lo tanto, la obra es original y que es el titular de los derechos patrimoniales, por lo que asume la responsabilidad sobre el contenido de la misma y saldrá en defensa de la Universidad en caso de reclamación por parte de terceros.

Ibarra, a los 13 días del mes de diciembre de 2023

EL AUTOR:



.....
Romel Ricardo Rosero Caragolla

CI: 1003104443



UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CERTIFICACIÓN

MAGÍSTER CARLOS VÁSQUEZ, DIRECTOR DEL PRESENTE TRABAJO DE
TITULACIÓN CERTIFICA:

Que el presente trabajo de titulación: "DISEÑO DE UN SISTEMA DE RECEPCIÓN DE SEÑALES LTE 4G UTILIZANDO RADIO DEFINIDO POR SOFTWARE, DIRIGIDO A LA CARRERA DE INGENIERÍA EN TELECOMUNICACIONES (CITEL) DE LA UNIVERSIDAD TÉCNICA DEL NORTE, PARA EL DESARROLLO DE PRÁCTICAS DE LABORATORIO DE LA ASIGNATURA DE RADIO DEFINIDO POR SOFTWARE (SDR)", ha sido desarrollado por el señor Romel Ricardo Rosero Caragolla bajo mi supervisión.

Es todo cuanto puedo certificar en honor a la verdad.

MSc. Carlos Alberto Vásquez Ayala

CI: 1002424982

DEDICATORIA

Este proyecto de investigación está dedicado a Dios y a la Virgen, por brindarme la fortaleza y sabiduría necesarias para llevar a cabo este exigente proceso de estudio. Nunca me permitieron desistir, y por ello les estoy profundamente agradecido.

A mis queridos padres, José Rosero y María Caragolla, quienes han sido pilares fundamentales en mi crecimiento personal y académico. Su apoyo incondicional ha sido esencial para alcanzar mis metas. También agradezco a mi hermana Fernanda Rosero y a mis sobrinos Alejo y Steffano, quienes han estado a mi lado día a día, siendo mi constante fuente de motivación y superación.

Finalmente, a toda mi familia, quienes también me han brindado su apoyo y sabios consejos en todo momento.

Con gratitud,

Romel Ricardo Rosero Caragolla

AGRADECIMIENTO

Agradezco a Dios y a la Virgen, quienes me han otorgado la oportunidad de vivir este momento significativo. Su fortaleza para ha sido mi guía en este camino, y con su bendición, encuentro motivos diarios para ser feliz.

Agradezco de todo corazón a mi padre, José Rosero y a mi madre, María Caragolla, por ser mi apoyo constante. Su paciencia y amor incondicional han sido fundamentales en este trayecto. Son mi ejemplo de constancia, esfuerzo y perseverancia. Gracias por impulsarme cada día y luchar a mi lado para alcanzar mis sueños.

A mis sobrinos Alejo y Stéffano, por todos los momentos que he podido compartir desde que llegaron a mi mundo y en especial por su apoyo y admiración hacia mí en todo este tiempo.

A mi hermana Fernanda, quién me ha sido un ejemplo de superación y motivación tanto en los desafíos como en los triunfos. Agradecido por tu cariño y paciencia.

Un agradecimiento especial a mis colegas Will, Axel, Guano, Fredy, Máx y Adonis con quiénes compartí día con día este proceso académico y de formación. Me llevo las experiencias, tanto las buenas como las difíciles y, sobre todo, la determinación de no rendirnos hasta

*alcanzar nuestros objetivos. Así como también a mi persona especial, que, aunque ya no está;
gracias por confiar, ayudarme a creer en mí y mis capacidades.*

*A todos mis familiares y amigos, que de una u otra manera estuvieron siempre presentes,
brindándome apoyo, valiosos consejos y aliento constante a largo del camino para lograr esta
meta.*

*Mi más sincero agradecimiento al director de mi tesis, el MSc. Carlos Vásquez, por su
guía, paciencia, dedicación de tiempo y por compartir su invaluable conocimiento para el
desarrollo y conclusión de este proyecto de titulación.*

*De manera especial, al MSc. Luis Suárez, por su tiempo, consejos, asesoramiento y
contribución significativa en la elaboración del este proyecto de investigación.*

Con cariño,

Romel Ricardo Rosero Caragolla

Índice de contenido

1. Capítulo I: Antecedentes	28
1.1. Planteamiento Del Problema.....	28
1.2. Objetivos	29
<i>1.2.1. Objetivo General</i>	<i>29</i>
<i>1.2.2. Objetivos Específicos.....</i>	<i>29</i>
1.3. Alcance	29
1.4. Justificación	33
1.5. Contexto	34
2. Capítulo II: Fundamento Teórico	36
2.1. Long Term Evolution (LTE)	36
<i>2.1.1. Reseña Histórica De LTE</i>	<i>36</i>
<i>2.1.2. Definición De Long Term Evolution (LTE)</i>	<i>39</i>
<i>2.1.3. Principios Básicos y Características De LTE</i>	<i>40</i>
2.1.3.1. Técnicas De Acceso Múltiple Por División De Frecuencia Ortogonal	
OFDMA	42
2.1.3.2. Técnica De Acceso Múltiple Por División De Frecuencia De	
Portadora Única SC-FDMA.....	43
2.2. Arquitectura De La Tecnología LTE	44
<i>2.2.1. Access Network (Red de Acceso) – E-UTRAN</i>	<i>46</i>

2.2.2. Evolved Packet Core (Núcleo de Red) – EPC.....	48
2.2.2.1. Mobility Management Entity (MME)	49
2.2.2.2. Serving Gateway (S-GW)	52
2.2.2.3. Packet Data Network Gateway (P-GW).....	54
2.2.2.4. Policy and Charging Rules Function (PCRF).....	55
2.2.2.5. Home Subscriber Server (HSS)	56
2.2.3. User Equipment (UE)	57
2.3. Interfaces de red E-UTRAN.....	59
2.3.1. Interfaces S1.....	59
2.3.2. Interfaces X2	65
2.4. Radio Definido Por Software (SDR).....	66
2.4.1. Definición de Radio Definido por Software (SDR)	66
2.4.2. Características De Radio Definido Por Software (SDR).....	67
2.5. LTE Hardware ME1130 Essentials of LTE y LTE-Advanced	69
2.5.1. LTE Radio – Access Node (eNodeB) & Core Networks Nodes (EPC)	69
2.5.2. LTE User Equipment (UE).....	71
2.5.3. GNU Radio con SDR	71
2.6. Metodología Iterativa.....	72
2.7. Software SDR.....	74
2.7.1. GNU Radio Companion.....	75

2.7.2.	<i>OpenAirInterface</i>	76
3.	Capítulo III: Desarrollo De La Propuesta (Diseño)	77
3.1.	Análisis De La Situación Actual	77
3.1.1.	<i>Análisis De Los resultados</i>	78
3.2.	Requerimientos Del proyecto	79
3.2.1.	<i>Construcción De Atributos De Los Requerimientos</i>	80
3.2.1.1.	Nomenclatura Para Uso De Los Requerimientos	80
3.2.1.2.	Requerimientos De Stakeholders	80
3.2.1.3.	Requerimientos del Sistema	82
3.2.1.4.	Requerimientos De Arquitectura	85
3.3.	Selección De Hardware y De Software	89
3.3.1.	<i>Selección De Hardware</i>	89
3.3.1.1.	Radio Definido Por Software Para LTE	90
3.3.1.2.	Radio Definido Por Software Como Analizador LTE	92
3.3.1.3.	LTE UE-Huawei E3372	94
3.3.2.	<i>Selección De Software</i>	95
3.3.2.1.	Aplicación para el análisis del espectro de RF para dispositivos SDR	
	96	
3.4.	Definición De Arquitectura	98
3.4.1.	<i>Bloque De Adquisición De Datos</i>	100

3.4.1.1.	Diagrama de Flujo Del Bloque De Adquisición De Datos	100
3.4.1.2.	Diagrama De Conexión Del Bloque De Adquisición de Datos	102
3.4.2.	<i>Bloque De Recepción Y Transmisión</i>	102
3.4.2.1.	Diagrama de Flujo Del Bloque De Recepción Y Transmisión	103
3.4.2.2.	Diagrama De Conexión Del Bloque De Recepción y Transmisión	105
3.4.3.	<i>Bloque De Procesamiento y Comunicación</i>	105
3.4.3.1.	Diagrama de Flujo Del Bloque De Procesamiento Y Comunicación	106
3.4.3.2.	Diagrama de Conexión Del Bloque De Procesamiento Y Comunicación	108
3.4.4.	<i>Bloque De Gestión</i>	108
3.4.4.1.	Diagrama de Flujo Del Bloque De Gestión	109
3.4.4.2.	Diagrama de Conexión Del Bloque De Gestión	111
3.4.5.	<i>Bloque De Visualización</i>	111
3.4.6.	<i>Bloque De Alimentación</i>	113
4.	Capítulo IV: Implementación	114
4.1.	Implementación De Hardware.....	114
4.1.1.	<i>Bloque De Adquisición De Datos</i>	114
4.1.2.	<i>Bloque De Recepción y Transmisión</i>	115
4.1.3.	<i>Bloque De Visualización</i>	116

4.1.4. Bloque De Alimentación.....	117
4.2. Implementación De Software	119
4.2.1. Bloque De Procesamiento y Comunicación.....	119
4.2.1.1. Configuración de OpenAirInterface.....	120
4.2.1.2. Configuración de eNodeB.....	120
4.2.2. Bloque De Gestión	123
4.2.2.1. Configuración de HSS.....	124
4.2.2.2. Configuración de MME.....	125
4.2.2.3. Configuración de SPGW	128
4.2.3. Bloque De Adquisición De Datos	130
4.2.3.1. Configuración de UE (Dongle LTE).....	131
5. Capítulo V: Pruebas De Funcionamiento Y Guías	134
5.1. Funcionamiento De Red LTE.....	134
5.1.1. Funcionamiento de EPC	134
5.1.1.1. Ejecución de HSS	135
5.1.1.2. Ejecución de MME.....	136
5.1.1.3. Ejecución del SPGW	138
5.2. Funcionamiento De Conexión De UE (Dongle LTE) a eNodeB	140
5.3. Guía 1: Software Osciloscopio	144
5.3.1. Comando para ejecución de EPC: HSS.....	146

5.3.2.	<i>Comando para ejecución de EPC: MME</i>	146
5.3.3.	<i>Comando para ejecución de EPC: S-PGW</i>	147
5.3.4.	<i>Comando para ejecución del eNodeB</i>	147
5.3.5.	<i>Verificación de software osciloscopio</i>	148
5.3.6.	<i>Verificación de estadísticas de la capa física y MAC</i>	151
5.4.	Guía 2: Mediciones De Red	153
5.4.1.	<i>Comandos para ejecución de EPC: HSS, MME y S-PGW</i>	155
5.4.2.	<i>Comandos para ejecución de eNodeB: 25, 50 y 100 PRB</i>	155
5.4.3.	<i>Pruebas de Velocidad</i>	156
5.4.4.	<i>Índice de Esquema de Codificación y Modulación (MCS)</i>	158
5.4.5.	<i>Comparación de velocidad medida y estimada</i>	159
5.5.	Guía 3: Análisis Protocolo LTE	164
5.5.1.	<i>Instalación y configuración de Wireshark</i>	166
5.5.2.	<i>Inicio de red LTE</i>	168
5.5.3.	<i>Análisis de protocolo de conexión del UE</i>	169
5.5.3.1.	Captura De Tráfico De La Interfaz Radio	169
5.5.3.2.	Captura De Mensajes Bloque de Información del Sistema Tipo 1 - System Information Block Type 1 (SIB1)	170
5.5.3.3.	Captura De Tráfico En La Interfaz S1	173
5.5.3.4.	Captura De Tráfico Protocolo de Tunnelización GPRS (GTP)	174

5.6. Guía 4: Familiarización GNU Radio y LimeSDR Mini	176
5.6.1. <i>Instalación De LimeSuite y GNU Radio Companion</i>.....	178
5.6.1.1. Verificación De Instalación De Python 2.7.16.....	183
5.6.2. <i>Instalación Del Entorno Pothos SDR</i>	184
5.6.2.1. Agregación De Pothos SDR a La Ruta Del Sistema	193
5.6.2.2. Verificación De Instalación de GNU Radio	196
5.6.3. <i>Creación De Diagrama De Flujo Para Transmisor QPSK</i>.....	198
5.6.4. <i>Conexión De LimeSDR Mini a PC</i>.....	207
5.6.5. <i>Generación y Ejecución De Señal De Banda Base Modulada QPSK</i>	210
5.7. Guía 5: Analizador De Espectro De Radio Definido Por Software	214
5.7.1. <i>Creación De Diagrama De Flujo Para Analizador De Espectro SDR</i>	216
5.7.2. <i>Generación y Ejecución Del Analizador De Espectro SDR</i>.....	224
Conclusiones	229
Recomendaciones	230
6. Bibliografía	233
7. Anexos	237
7.1. Anexo 1. Encuesta de Stakeholders	237
7.2. Anexo 2. Tabulación de Encuesta	240
7.3. Anexo 3. Fichero de configuración del eNodeB	249
7.4. Anexo 4. Fichero de configuración del HSS.....	255

7.5.	Anexo 5. Fichero de configuración del MME	257
7.6.	Anexo 6. Fichero de configuración del S-PGW	264

Índice de Figuras

Figura 1 Configuración y verificación de hardware	31
Figura 2 Componente lógicos e interfaces en la configuración.....	31
Figura 3 Esquema de acceso múltiple LTE (OFDMA)	43
Figura 4 Esquema de acceso múltiple LTE (SC - FDMA).....	44
Figura 5 Arquitectura de la red LTE.....	45
Figura 6 Conexiones eNodeB a otros nodos lógicos y funciones principales	47
Figura 7 Reparto funcional entre E-UTRAN y EPC	48
Figura 8 Arquitectura del sistema para red E-UTRAN	49
Figura 9 Conexiones MME a otros nodos lógicos y funciones principales.....	51
Figura 10 Conexiones S-GW a otros nodos lógicos y funciones principales.....	53
Figura 11 Conexiones P-GW a otros nodos lógicos y funciones principales.....	54
Figura 12 Conexiones PCRF a otros nodos lógicos y funciones principales	56
Figura 13 Equipo de Usuario.....	58
Figura 14 Pila de protocolos del plano de control con la conexión de un UE a una PDN	59
Figura 15 Pila de protocolos del plano de control hacia las puertas de enlace.....	63
Figura 16 Pila de protocolos del plano de usuario en EPS	64
Figura 17 Pila de protocolos de control y plano de usuario para la interfaz X2.....	65
Figura 18 Diagrama de bloques de un transceptor digital genérico	68
Figura 19 LTE eNodeB & EPC	70
Figura 20 LTE UE	71
Figura 21 GNU Radio con SDR	72

Figura 22	Representación gráfica del modelo iterativo SDLC.....	73
Figura 23	Arquitectura general del sistema	99
Figura 24	Diagrama de flujo del bloque de adquisición de datos.....	100
Figura 25	Diagrama de conexión del UE.....	102
Figura 26	Diagrama de flujo del bloque de recepción y transmisión	103
Figura 27	Diagrama de conexión de LimeSDR.....	105
Figura 28	Diagrama de flujo del bloque de recepción y transmisión	106
Figura 29	Diagrama de conexión de máquina física (OAI) eNodeB y LimeSDR	108
Figura 30	Diagrama de flujo del bloque de gestión.....	109
Figura 31	Diagrama de conexión de máquina física (OAI) EPC y LimeSDR	111
Figura 32	Diagrama de conexión de monitor con máquina física (MiniPC).....	112
Figura 33	Diagrama de conexión de alimentación de mini PC, LimeSDR y periféricos de entrada.....	113
Figura 34	Conexión de dongle, cables SMA y atenuadores	114
Figura 35	Conexión de dongle, cables SMA, atenuadores al LimeSDR.....	115
Figura 36	Conexión HDMI Mini PC y HDMI Monitor	116
Figura 37	Alimentación Mini PC y monitor	118
Figura 38	Adaptador Ethernet para dongle LTE	131
Figura 39	Página web Dongle.....	131
Figura 40	Configuración APN.....	132
Figura 41	Admitir redes LTE habilitado.....	133
Figura 42	Ejecución de HSS.....	135
Figura 43	Inicialización con éxito de HSS	135

Figura 44 Ejecución de MME.....	136
Figura 45 Estadísticas de conexiones del MME	137
Figura 46 Ejecución de SPGW	138
Figura 47 Inicialización con éxito de SPGW.....	139
Figura 48 Ejecución de eNodeB	140
Figura 49 Conexión a la red LTE del Dongle LTE	141
Figura 50 Conexión exitosa del UE (Dongle LTE) al eNodeB	142
Figura 51 Gráfica generada software osciloscopio.....	148
Figura 52 Gráfica generada de estadísticas.....	151
Figura 53 Prueba de velocidad con 25 PRB	157
Figura 54 Modulation and TBS index table for PDSCH.....	159
Figura 55 Modulation, TBS index and redundancy version table for PUSCH.....	161
Figura 56 Habilitación de subdirectores heurísticos en Wireshark	166
Figura 57 Habilitación de protocolos LTE en Wireshark.....	167
Figura 58 Tráfico de red en la interfaz radio NAS y RRC	169
Figura 59 Tráfico de red en la interfaz radio RLC y MAC	170
Figura 60 Parámetros paquete SIB1	171
Figura 61 Tráfico capturado interfaz S1 (plano de control)	173
Figura 62 Diagrama de flujo tráfico interfaz S1 (plano de control)	173
Figura 63 Tráfico capturado interfaz S1 (plano de usuario protocolo GTP).....	174
Figura 64 Tráfico capturado plano de usuario sin encapsular protocolo GTP	175
Figura 65 Selección de instalador MSI Windows x86-64	178
Figura 66 Selección de instalación para todos los usuarios.....	179

Figura 67 Directorio de destino Python27.....	179
Figura 68 Personalización de Python.....	180
Figura 69 Proceso de instalación de Python	181
Figura 70 Instalación de Python finalizada.....	182
Figura 71 Verificación de instalación de Python 2.7.16.....	183
Figura 72 Selección de instalador Pothos SDR	184
Figura 73 Ventana de inicio de Instalación de Pothos SDR	185
Figura 74 Acuerdo de licencia Pothos SDR	186
Figura 75 Selección de opción agregar Pothos SDR a la ruta del sistema	187
Figura 76 Directorio de destino Pothos SDR.....	188
Figura 77 Selección de carpeta del menú inicio	189
Figura 78 Selección de componentes Pothos SDR.....	190
Figura 79 Proceso de instalación de Pothos SDR.....	191
Figura 80 Instalación de Pothos SDR finalizada	192
Figura 81 Ventana propiedades del sistema	194
Figura 82 Ventana variables del entorno	195
Figura 83 Creación de variable de entorno PothosSDR	195
Figura 84 Ejecución de GNU Radio e instalación automática de librerías	196
Figura 85 Interfaz gráfica de usuario GNU Radio.....	197
Figura 86 Bloque options y Variable en GNU Radio	198
Figura 87 Bloque modulador de constelación	199
Figura 88 Bloque generador de fuente aleatoria.....	199
Figura 89 Bloque objeto de constelación.....	200

Figura 90 Bloque LimeSuite Sink	201
Figura 91 Bloque QT GUI Range.....	202
Figura 92 Bloque Multiply const.....	203
Figura 93 Bloque QT GUI Time Sink	204
Figura 94 Bloque QT GUI Constellation Sink	205
Figura 95 Bloque QT GUI Frequency Sink.....	205
Figura 96 Diagrama de flujo completo transmisor QPSK final.....	206
Figura 97 Ventana administrador de dispositivos LimeSDR Mini no disponible.....	207
Figura 98 Instalación correcta del controlador para LimeSDR Mini	208
Figura 99 Verificación de funcionamiento de controlador para LimeSDR Mini	209
Figura 100 Barra de herramientas GNU para, generar y ejecutar diagrama de flujo	210
Figura 101 Mensajes presentados en el terminal al generar el diagrama de flujo	210
Figura 102 Mensajes presentados en el terminal al ejecutar el diagrama de flujo	211
Figura 103 Gráfica de la forma de onda en el dominio del tiempo	212
Figura 104 Gráfica del espectro de frecuencia en tiempo real	212
Figura 105 Gráfica diagrama de constelación de la señal modulada.....	213
Figura 106 Bloques Options, Variable y configuración de tasa de muestreo.....	216
Figura 107 Bloque QT GUI Range 1 (frecuencia)	217
Figura 108 Bloque QT GUI Range 2 (ganancia).....	218
Figura 109 Bloque LimeSuite Source.....	219
Figura 110 Bloque QT GUI Tab Widget.....	220
Figura 111 Bloque Variable para tamaño de la transformada de Fourier.....	221
Figura 112 Bloque QT GUI Frequency Sink.....	221

Figura 113 Bloque QT GUI Waterfall Sink.....	223
Figura 114 Diagrama de flujo completo analizador de espectro de SDR.....	223
Figura 115 Mensajes mostrados en el terminal al generar el diagrama de flujo	225
Figura 116 Mensajes mostrados en el terminal al ejecutar el diagrama de flujo.....	225
Figura 117 Gráfica de la señal de radio capturada.....	226
Figura 118 Gráfica del espectro de frecuencia	227
Figura 119 Gráfica del espectrograma (Waterfall).....	228
Figura 120 Gráfico de tabulación de pregunta 1.....	240
Figura 121 Gráfico de tabulación de pregunta 2.....	241
Figura 122 Gráfico de tabulación de pregunta 3.....	241
Figura 123 Gráfico de tabulación de pregunta 4.....	242
Figura 124 Gráfico de tabulación de pregunta 5.....	243
Figura 125 Gráfico de tabulación de pregunta 6.....	244
Figura 126 Gráfico de tabulación de pregunta 7.....	245
Figura 127 Gráfico de tabulación de pregunta 8.....	246
Figura 128 Gráfico de respuestas de pregunta 9.....	248

Índice de Tablas

Tabla 1. Evolución de las especificaciones Universal Mobile Telecommunications System - Sistema Universal de Telecomunicaciones Móviles (UMTS).....	37
Tabla 2 Abreviatura de Requerimientos	80
Tabla 3 Requerimientos de Stakeholders.....	81
Tabla 4 Requerimientos del Sistema.....	83
Tabla 5 Requerimientos de Arquitectura	86
Tabla 6 Valor establecido para requerimientos.....	90
Tabla 7 Selección de dispositivos SDR para LTE	90
Tabla 8 Características técnicas de Lime SDR	91
Tabla 9 Selección de dispositivos SDR como analizador LTE	92
Tabla 10 Características técnicas de Lime SDR Mini	93
Tabla 11 Selección de dispositivos módem para LTE UE.....	94
Tabla 12 Características técnicas de módem para LTE UE Huawei E3372	94
Tabla 13 Selección de aplicación para análisis del espectro de RF para dispositivos SDR	96
Tabla 14 Características técnicas de OpenAirInterface	97
Tabla 15 Velocidad de descarga	157
Tabla 16 Velocidad de carga.....	158
Tabla 17 Valores máximos de MCS	159
Tabla 18 Velocidad de descarga estimado, medido y diferencia.....	161
Tabla 19 Velocidad de carga estimado, medido y diferencia	163
Tabla 20 Parámetros del paquete SIB1	170

RESUMEN

Este proyecto se basa en la investigación, desarrollo y despliegue de un sistema de recepción de señales LTE 4G mediante el uso de Radio Definido por Software (SDR). Logrando establecer prácticas en tecnologías LTE y SDR, brindando una base sólida para el aprendizaje y la investigación en este campo.

El desarrollo del proyecto implica un análisis exhaustivo del estado actual de la tecnología LTE 4G y de los sistemas de recepción de señales, considerando sus parámetros, aspectos clave y criterios de funcionamiento, con la exploración de tecnologías que posibiliten la comunicación en frecuencia, ancho de banda, modulación y velocidad de transferencia, con el objetivo de obtener el despliegue del sistema de recepción de señales en base a las tecnologías LTE y SDR.

Para llevar a cabo este proyecto, se aplica la metodología Iterativa o Incremental, que es ideal en el desarrollo de software. Esta metodología consta de varias fases: Requisitos, Diseño, Implementación, Pruebas y Revisión. Durante la fase de Requisitos, se definirán las características esenciales que el sistema de recepción de señales debe poseer, alineadas con los objetivos de la materia de SDR. Se recopilarán y estudiarán cuidadosamente los requisitos, considerando la posibilidad de nuevos requisitos o modificaciones de los existentes.

En la etapa de diseño, se detalla la arquitectura del proyecto, dividiéndola en dos subsistemas de trabajo: E-UTRAN (Evolved Universal Terrestrial Radio Access Network – Red Universal de Acceso Radio Terrestre Evolucionada), o red de acceso radio y el EPC (Evolved

Packet Core – Núcleo de Red en Conmutación de Paquetes Evolucionado), o núcleo de red. Esta estructura sirve como base para la implementación de la tecnología SDR.

En la implementación se incluye procedimientos y prácticas destinados a familiarizar a los estudiantes con el sistema de recepción de señales LTE 4G. Las guías están diseñadas para fomentar el desarrollo académico y práctico en áreas relevantes, de acuerdo al plan de estudios de la materia de SDR, permitiendo que los estudiantes adquieran conocimientos sobre esta tecnología.

Finalmente, en la fase de pruebas se verifica la funcionalidad del sistema, para ello se elaboran guías de prácticas que abarcarán la familiarización con el sistema, la configuración de equipos SDR.

ABSTRACT

This project is based on the research, development, and deployment of a 4G LTE signal reception system using Software Defined Radio (SDR). It aims to establish expertise in LTE and SDR technologies, providing a solid foundation for learning and research in this field.

The development of the project involves a comprehensive analysis of the current state of 4G LTE technology and signal reception systems, considering their parameters, key aspects, and operational criteria. This includes the exploration of technologies enabling communication in terms of frequency, bandwidth, modulation, and transfer speed. The ultimate goal is to achieve the deployment of the signal reception system based on LTE and SDR technologies.

To carry out this project, an Iterative or Incremental methodology is applied, which is ideal in software development. This methodology consists of several phases: Requirements, Design, Implementation, Testing, and Review. During the Requirements phase, the essential features that the signal reception system must possess, aligned with the objectives of the SDR subject, will be defined. Requirements will be collected and carefully studied, considering the possibility of new requirements or modifications to existing ones.

In the design stage, the project architecture is detailed, dividing it into two working subsystems: E-UTRAN (Evolved Universal Terrestrial Radio Access Network) or radio access network, and EPC (Evolved Packet Core) or core network. This structure serves as the foundation for the implementation of SDR technology.

The implementation phase includes procedures and practices aimed at familiarizing students with the 4G LTE signal reception system. The guides are designed to encourage

academic and hands-on development in relevant areas, according to the curriculum of the SDR subject, enabling students to acquire knowledge about this technology.

Finally, in the testing phase, the functionality of the system is verified. For this purpose, practical guides are developed covering familiarization with the system and the configuration of SDR equipment.

1. Capítulo I: Antecedentes

1.1. Planteamiento Del Problema

Los constantes avances tanto a nivel tecnológico, han traído a lo largo del tiempo una gran cantidad de retos por solucionar, en cuanto a los usuarios de los servicios de telecomunicaciones móviles que han experimentado falencias en la prestación del servicio, debido a la insuficiencia de infraestructura en zonas rurales, además de tener problemas de cobertura en sitios como parqueaderos subterráneos, ascensores y, en general, edificaciones robustas, donde las pérdidas de la potencia de la señal son bastante altas y la conectividad con la red móvil se pierde. (Chica Pedraza et al., 2020). Por lo antes mencionado y enfocado a la Universidad Técnica del Norte, en la actualidad, no cuenta con toda la infraestructura de Telecomunicaciones que permita a los estudiantes de la Carrera de Ingeniería en Telecomunicaciones realizar prácticas de laboratorio en el área de la telefonía móvil, en la asignatura de Radio Definido por Software (SDR).

En sí, se busca definir los parámetros para el diseño de un sistema de recepción de señales LTE 4G con el uso de Radio Definido por Software que presenten una implementación de software compatible con los estándares de código abierto de LTE y LTE Advanced.

El sistema de recepción de señales LTE 4G al tener como principio el uso de Radio Definido por Software, permite replicar de una forma didáctica y educativa, todo el proceso de generación y recepción de servicios básicos de Telecomunicaciones móviles. (Codon, 2019)

Los sistemas SDR (Radio Definido por Software) se muestran como una solución para evaluar problema de cobertura móvil, ya que permite la implementación y el desarrollo de la infraestructura de servicios móviles, que orientados hacia la Universidad Técnica del Norte generan fortalezas en las competencias sobre el contexto tecnológico y educativo.

1.2. Objetivos

1.2.1. Objetivo General

Diseñar un Sistema de Recepción de Señales LTE 4G, mediante el uso de Radio Definido por Software, para la realización de prácticas de laboratorio de SDR para la Carrera de Ingeniería en Telecomunicaciones (CITEL) basados en contenidos de la materia

1.2.2. Objetivos Específicos

- Realizar un estudio del estado del arte en la tecnología LTE 4G, referente a parámetros, características de diseño, junto con su arquitectura de red de Radio Definido por Software (SDR)
- Establecer un levantamiento de requerimientos de hardware y software, para el diseño del sistema de recepción de señales LTE 4G
- Desplegar el diseño del sistema de recepción de señales, basándose en la metodología iterativa, cumpliendo con todos los parámetros planteados en el sistema de recepción de señales para la realización de prácticas de redes
- Simular el diseño planteado mediante el uso de radio definido por software y herramientas de software libre
- Elaborar guías que permitan a los estudiantes realizar prácticas en base a la temática indicada en el sílabo de la materia, para la comprensión de conceptos y operatividad sobre las tecnologías LTE 4G mediante SDR

1.3. Alcance

El presente proyecto tiene como finalidad desarrollar un sistema de recepción de señales LTE 4G, mediante el uso de Radio Definido por Software, con el fin de establecer prácticas, en

tecnologías LTE y SDR en la materia de SDR, para desarrollar criterios de aplicación de estas tecnologías para su aprendizaje e investigación.

Para el desarrollo del proyecto se realizará un análisis del estado del arte en sistemas de recepción de señales junto con sus parámetros, aspectos y criterios de funcionamiento.

Considerando tecnologías que permitan la comunicación utilizando cualquier frecuencia, ancho de banda, modulación y velocidad de transferencia con el objetivo de generar un sistema de recepción de señales con soporte en las tecnologías LTE y SDR.

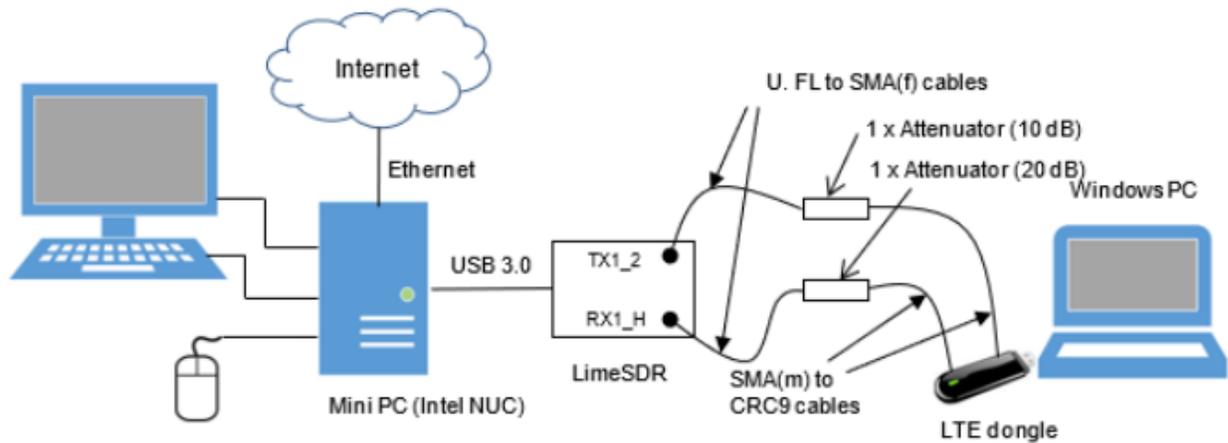
De igual manera, paralelamente al desarrollo se utilizará la metodología Iterativa o Incremental (Fase de Requisitos, Fase de Diseño, Fase de Implementación, Fase de Pruebas y Fase de Revisión), la cual es ideal en el desarrollo de software, donde se indica los pasos a realizar para la elaboración del sistema de recepción de señales basándose en los requisitos.

Para la fase de Requisitos, se definirán las características que el sistema de recepción de señales debe tener para proporcionar los servicios fundamentados en los objetivos de la materia de SDR, es decir este requisito de iteración se deberá recopilar y estudiar cuidadosamente tomando en cuenta que dicho requisito podrá ser un nuevo o una extensión del mismo que ya ha sido construido.

En la etapa de diseño, se especificará en la arquitectura del proyecto indicando la red la cual estará fragmentada en dos subsistemas de trabajo, el primero el E-UTRAN (Evolved Universal Terrestrial Radio Access Network) la cual es la subdivisión de la red encargada del acceso y la segunda, que hace énfasis en el núcleo de la misma denominada EPC (Evolved Packet Core).

Figura 1

Configuración y verificación de hardware

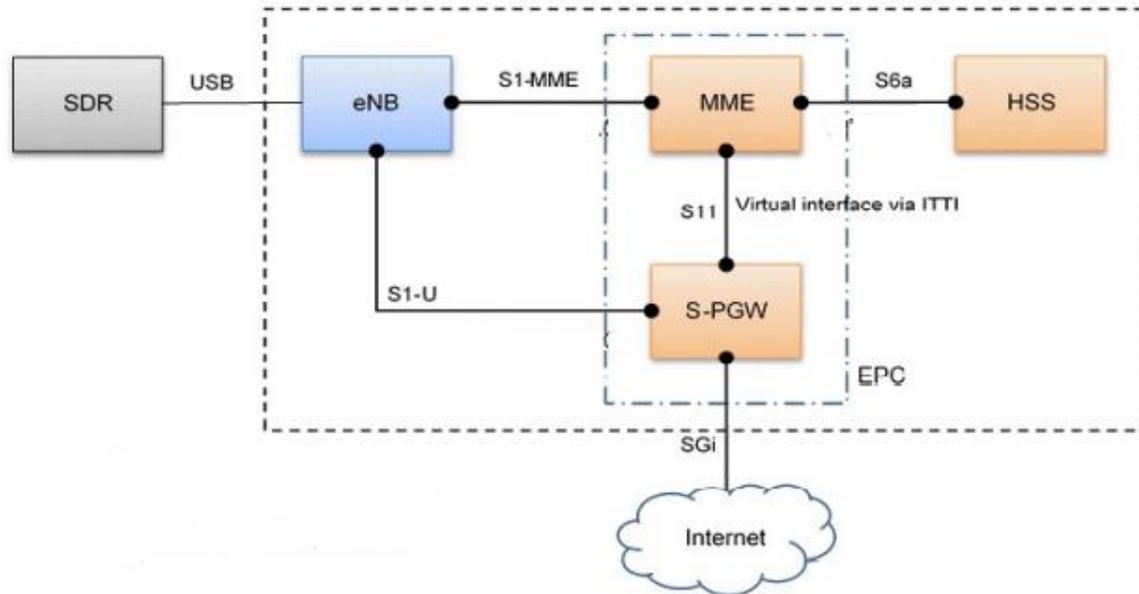


Nota. Tomado de *Load eNB and EPC configurations* (p. 9), de (DreamCatcher, 2020), *ME1130 Essentials of LTE and LTE - A courseware. Quick Start Guide.*

Para la fase de implementación o desarrollo, una vez establecidos los requerimientos de hardware y software, metodología y procedimientos a realizar se procede a aplicar cada uno de estos recursos hacia la formación del sistema de recepción de señales. Como primer paso la gestión del hardware mediante el software SDR, siendo la base para el establecimiento de las redes SDR, que permitirán la generación de un sistema de recepción de señales con soporte en las tecnologías LTE y SDR.

Figura 2

Componente lógicos e interfaces en la configuración



Nota. Adaptado de Load eNB and EPC configurations (p. 3), de (DreamCatcher Courseware , 2020), University Courseware Teaching Lab Exercises about LTE Technologies.

Parte del despliegue del sistema será fragmentar el mismo en dos subsistemas, el E-UTRAN (Evolved Universal Terrestrial Radio Access Network) que es la subdivisión de la red encargada del acceso y el EPC (Evolved Packet Core), el cual se refiere al núcleo de la misma, siendo base para el establecimiento de la tecnología SDR

Dentro de esta implementación se establecerán procedimientos y prácticas para que los estudiantes puedan familiarizarse y realizar sus prácticas con el sistema de recepción de señales LTE 4G. Las guías en sí estarán enfocadas a la fomentación de criterios académicos y de desarrollo sobre temas de aplicación revisados en el sílabo de la materia, permitiendo al estudiante formarse en los aspectos de estas tecnologías

Finalmente, en la fase de pruebas se implementará la funcionalidad de dicho sistema, donde basado en lo contenidos y objetivos de la materia de SDR se establecerán guías de

prácticas que incluyan, tanto la familiarización del sistema y tecnologías SDR, ambientación y configuraciones de equipos SDR, además de generación de topologías con SDR.

1.4. Justificación

A lo largo del tiempo, Ecuador se ha desarrollado en servicios e infraestructura de telecomunicaciones de manera que los habitantes puedan comunicarse en tiempo real sin importar el lugar donde se encuentren ubicados. Por lo tanto, es importante que se logre cubrir áreas donde la población no ha tenido acceso a la comunicación telefónica o donde se presenta fallas en servicios de voz y datos debido a antiguas infraestructuras instaladas, ya que a diario la tecnología sorprende con nuevos avances, es por este motivo que el diseño de nuevas estaciones base dentro de una red celular tiene como finalidad proveer a los usuarios con buena calidad y cobertura tanto para tecnologías de UMTS (Universal Mobile Telecommunication System) y LTE (Long Term Evolution). (Escobar Chamba, 2021)

Mediante una tecnología más avanzada como las redes de cuarta generación LTE 4G, la misma que favorecerá en gran manera el desarrollo tanto comercial como educativo, asegurando a los usuarios soporte necesario para la movilidad y compatibilidad entre los sistemas, (conectividad con otras redes), con lo cual se podrá utilizar el servicio de banda ancha en cualquier momento y lugar. (Tafur Arenazas, 2017)

En el país y a nivel internacional, existen estudios en universidades del país como lo es la Universidad de las Fuerzas Armadas (Ecuador) y la Universidad Santo Tomas (Colombia) donde ha existido investigaciones sobre diseño de una estación base con la finalidad de cubrir todas las localidades con una aceptable calidad de servicio y proveer cobertura mediante la red UMTS y LTE brindando solución al área que presenta problemas. Dando a notar que en el país si hay una tendencia hacia la investigación y aplicación de estas tecnologías ya sea en ambientes de

producción y de manera especial en ambientes de investigación académico. (Escobar Chamba, 2021), (Chica Pedraza et al., 2020)

Por lo cual se establece una necesidad de profesionales capacitados en dichas áreas y por consiguiente una necesidad de que las carreras afines tomen en consideración agregar estas tecnologías a su currículo para poder satisfacer dicha demanda, ya que, al ser carreras tecnológicas, estar a la par con la tecnología ofrece una ventaja a los graduados en estas áreas

La propuesta de este proyecto es brindar a los estudiantes de la Universidad Técnica del Norte un diseño de un sistema donde se pueda formar y capacitar a las nuevas generaciones de ingenieros en telecomunicaciones en áreas de tecnología LTE Y SDR para el fomento de su aprendizaje e investigación.

1.5. Contexto

Tomando en consideración la tesis **“DISEÑO DE UNA ESTACIÓN BASE PARA SU INTEGRACIÓN EN UNA RED CELULAR CON TECNOLOGÍA UMTS Y LTE EN UNA ESTACIÓN REFERENCIAL”** De la Ing. Andrea Escobar. Donde su enfoque es diseñar una estación base que tenga como finalidad debido a que Ecuador ha ido mejorando el servicio de telecomunicaciones por lo tanto es importante cubrir todas las localidades o áreas donde la población no ha tenido acceso a la comunicación telefónica o donde se presenta fallas en servicios de voz y datos, por lo que es motivo que el diseño de nuevas estaciones base dentro de una red celular tiene como finalidad proveer a los usuarios obtener una aceptable calidad de servicio y así proveer cobertura mediante una red UMTS y LTE.

De la misma manera en base al trabajo **“IMPLEMENTACIÓN DE ESTACIÓN BASE GSM RECEPCIÓN DE SEÑALES LTE APLICANDO RADIO DEFINIDO POR**

SOFTWARE” De Gustavo Chica, Díaz Ángel, Mónica Espinosa y David Angulo. Donde dicho trabajo hace referencia a la Universidad Santo Tomás que al no contar con toda la infraestructura de telecomunicaciones tiene como finalidad la implementación de una estación base de telefonía móvil celular (TMC), GSM 2G y el desarrollo de un sistema de recepción de señales LTE 4G, todo esto mediante el uso de radio definido por software y herramientas de software libre. Dicha implementación de la estación base genera y gestiona el recurso inalámbrico emitido por ella misma, con el fin de aprovisionar servicios de mensajería de texto corto y llamada de voz. Por otra parte, el sistema de recepción de señales LTE 4G al ser desarrollado en el software GNURadio, que permite replicar de una forma didáctica y educativa, todo el proceso de generación y recepción de servicios básicos de telecomunicaciones móviles.

El proyecto pretende establecer un diseño de un sistema de recepción de señales LTE 4G, mediante el uso de Radio Definido por Software, que permita a los estudiantes de la carrera de Ingeniería en Telecomunicaciones, poner en práctica su aprendizaje y desarrollo de tecnologías SDR y LTE. Mediante el establecimiento de un sistema basado en estas mismas tecnologías con el fin de que los docentes puedan establecer casos de estudio para que los estudiantes desarrollen distintos escenarios y cumplir con los requerimientos, fortaleciendo su desarrollo e investigación académica.

2. Capítulo II: Fundamento Teórico

2.1. Long Term Evolution (LTE)

LTE se dio inicio como un estudio en el 3rd Generation Partnership Project - Proyecto Asociación de Tercera Generación (3GPP), como un taller realizado en noviembre del 2004 en Canadá. Las primeras presentaciones del taller se basaron tanto en los requisitos esperados para el trabajo como también enfocados a las tecnologías que se adoptarán. En el mismo, se hicieron contribuciones tanto desde el punto de vista del operador como del proveedor. Como resultado del taller realizado, se dio paso a la aprobación del estudio para LTE en diciembre del 2004, para definir los requisitos y luego en base a los grupos de trabajo para discusiones técnicas detalladas para el protocolo de acceso múltiple, soluciones y arquitectura. (Rumney, 2013)

2.1.1. Reseña Histórica De LTE

“El desarrollo de LTE se encuentra en etapas conocidas como lanzamientos de 3rd Generation Partnership Project - Proyecto Asociación de Tercera Generación (3GPP). Los proveedores de hardware y los desarrolladores de software utilizan estas versiones como parte de su hoja de ruta de desarrollo” (Huawei Technologies Co., Ltd, 2010, pág. 6).

Las especificaciones LTE están escritas por 3GPP, que es una asociación de organizaciones de desarrollo de estándares (SDOs). Los socios organizacionales que componen 3GPP son la Asociación Japonesa de Industrias y Negocios de Radio (ARIB), la Alianza de EE. UU. para Soluciones de la Industria de Telecomunicaciones (ATIS), la Asociación de Estándares de Comunicaciones de China (CCSA), el Instituto Europeo de Estándares de Telecomunicaciones (ETSI), la Asociación de Tecnología de Telecomunicaciones de Corea (TTA) y Comité de Tecnología de Telecomunicaciones de Japón (TTC). (Rumney, 2013, pág. 3)

Tabla 1.

Evolución de las especificaciones Universal Mobile Telecommunications System - Sistema

Universal de Telecomunicaciones Móviles (UMTS)

Release	Fecha	Característica principal de UMTS del Release
Rel-99	Marzo 2000	W-CDMA básico de 3,84 Mcps (FDD y TDD)
Rel-4	Marzo 2001	1.28 Mcps TDD (TD-SCDMA)
Rel-5	Junio 2002	HSDPA
Rel-6	Marzo 2005	HSUPA (E-DCH)
Rel-7	Diciembre 2007	HSPA+ (64QAM downlink, MIMO, 16QAM uplink) Estudio de factibilidad LTE y SAE
Rel-8	Diciembre 2008	Elemento de trabajo LTE: interfaz aérea OFDMA/SC-FDMA, Elemento de trabajo SAE: nueva red central de IP, Dual-carrier HSDPA
Rel-9	Diciembre 2009	Home BS, MBMS, multi-standard radio, dual-carrier HSUPA, dual-carrier HSDPA con MIMO, dual-cell HSDPA
Rel-10	Marzo 2011	LTE-Advanced (agregación de portadores, 8x DL MIMO, 4x UL MIMO, retransmisión, interferencia entre celdas mejorada coordinación (eICIC)), HSDPA de 4 portadoras
Rel-11	Septiembre 2012	Además, eICIC, transmisión multipunto coordinada (CoMP), escenarios de agregación de portadoras, HSDPA de 8 portadoras
Rel-12	Marzo 2013	Mayor coordinación de interferencias, portadora entre sitios agregación, otros TBD incluyendo TDD dinámico y LTE-D

Nota. Esta tabla resume la evolución de las especificaciones 3GPP UMTS hacia LTE. Cada versión de las especificaciones 3GPP representa un conjunto definido de características. Tomado de: (Rumney, 2013)

Según (Rumney, 2013) después del Release 99, 3GPP dejó de mencionar a dichas versiones por el año y seleccionó un nuevo esquema iniciando a partir del Release 4.

El Release 4 introdujo la versión de banda estrecha de 1,28 Mcps de Wideband Code Division Multiple Access - Acceso Múltiple por División de Código de Banda Ancha W-CDMA, denominada Time Division Synchronous Code Division Multiple Access - Acceso Múltiple por División de Código Síncrono por División de Tiempo (TD-SCDMA).

El Release 5, High Speed Downlink Packet Access - Acceso de Paquetes de Enlace Descendente de Alta Velocidad (HSDPA) incluyó servicios de datos basados en paquetes para UMTS de igual forma que el General Packet Radio Service - Servicio General de Radio por Paquetes (GPRS) para Global System For Mobile Communication - Sistema Global para las Comunicaciones Móviles (GSM) en el Release 97 (1998).

El Release 6 dio paso a la finalización de los paquetes de datos para UMTS junto con la adición de High-Speed Uplink Packet - Access a Paquetes de Enlace Ascendente de Alta Velocidad (HSUPA), tecnología conocida como Enhanced Dedicated Channel - Canal Dedicado Mejorado (E-DCH). High Speed Downlink Packet Access - Acceso Descendente de Paquetes de Datos a Alta Velocidad (HSDPA) y HSUPA y definido finalmente como High-Speed Packet Access - Acceso a Paquetes de Alta Velocidad (HSPA).

Además, para el Release 7 se incluía el primer trabajo en LTE en conjunto con la finalización de los estudios de factibilidad, y además se realizaron mejoras adicionales a HSPA, como Multiple Input-Multiple Output - Múltiple Entrada Múltiple Salida (MIMO), 64QAM en el enlace descendente y 16QAM en el enlace ascendente.

El Release 8, HSPA continuó evolucionando con la adición de un sin número de funciones más pequeñas, como HSDPA de doble portadora y 64QAM con MIMO. HSUPA de dos operadores se introdujo en el Release 9, HSDPA de cuatro operadores en el Release 10 y HSDPA de ocho operadores en el Release 11. (pág. 4)

2.1.2. Definición De Long Term Evolution (LTE)

Se conoce a LTE como un estándar para redes de comunicaciones móviles el cual fue desarrollado por el 3GPP, donde se considera la evolución de las redes tanto de segunda generación (2G), en conjunto con el Sistema Global para Comunicaciones Móviles (GSM), además del Sistema Global de Conmutación de Paquetes por Radio (GPRS), seguido de la Tasa de Velocidad Mejorada para la Evolución de GSM (EDGE) y finalmente de las redes de Tercera Generación (3G) con el Servicio Universal de Telecomunicaciones Móviles (UMTS) y el Acceso por Conmutación de Paquetes de Alta Velocidad (HSPA). (Holma y Toskala, 2009)

Por lo tanto, según lo afirma (Demel, 2012) en LTE se utiliza Orthogonal Frequency Division Multiplexing - Multiplexación por División de Frecuencia Ortogonal (OFDM) para la transmisión. Las altas tasas de datos deseadas dan como resultado una duración de símbolo muy corta para la transmisión de una sola portadora, lo que significa que la interferencia entre símbolos que se produce como resultado de la propagación por trayectos múltiples tiene un impacto negativo particularmente fuerte. La ventaja de OFDM es que las transmisiones de alta

velocidad se dividen en flujos de datos paralelos de baja velocidad, con una duración de símbolo OFDM.

(Sesia et al., 2009) afirman, que previo el análisis de los requisitos para el sistema LTE llevó a cabo a la creación de un estudio oficial en 3GPP con el objetivo principal de evolucionar la tecnología de acceso de radio (3GPP) garantizando la competitividad en un período aproximado de 10 años. Dando como finalización en junio de 2005. Y los mismos se pueden detallar de la siguiente forma:

- retrasos reducidos, establecimiento de conexión y latencia de transmisión
- mayores tasas de datos de usuario
- mayor velocidad binaria en el borde de la celda, para uniformidad en la prestación del servicio
- costo por bit reducido, lo que implica una eficiencia espectral mejorada
- mayor flexibilidad en el uso del espectro, tanto en bandas nuevas como preexistentes
- arquitectura de red simplificada
- movilidad continua, entre diferentes tecnologías de acceso por radio
- consumo de energía moderada para el terminal móvil. (p. 7)

2.1.3. Principios Básicos y Características De LTE

LTE se basa en una serie de requisitos en cuanto a la velocidad de datos, capacidad, eficiencia espectral y la latencia. Por lo tanto, (Cardona et al., 2011) han afirmado lo siguiente:

Para cumplimiento de dichos requisitos, LTE se enfoca en un esquema de acceso múltiple en la interfaz aire; para ello utiliza Orthogonal Frequency Division Multiple Access - Acceso Múltiple por División de Frecuencia Ortogonal (OFDMA) en enlace

Downlink - Descendente (DL) y Single Carrier – Frequency Division Multiple Access - Portadora Única - Acceso Múltiple por División de Frecuencia (SC-FDMA) en enlace Uplink - Ascendente (UL).

En la normalización de LTE al inicio del proceso, se identificaron los requisitos principales para el diseño de un sistema con estas características. (p. 10)

Tal como la afirman, (Holma y Toskala, 2009) estas soluciones de acceso múltiple brindan ortogonalidad entre los usuarios, reduciendo la interferencia y mejorando la capacidad de la red, es decir:

- La asignación de recursos en el dominio de la frecuencia se realiza con una resolución de bloques de recursos de 180 kHz tanto en el enlace ascendente como en el descendente
- La dimensión de la frecuencia en la programación de paquetes es una de las razones de la alta capacidad de LTE
- La asignación específica del usuario del enlace ascendente es continua para permitir la transmisión de una sola portadora, mientras que el enlace descendente puede usar bloques de recursos libremente de diferentes partes del espectro
- La solución de portadora única de enlace ascendente también está diseñada para permitir un diseño eficiente del amplificador de potencia del terminal, lo cual es relevante para la vida útil de la batería del terminal
- La solución LTE permite flexibilidad de espectro donde el ancho de banda de transmisión se puede seleccionar entre 1,4 MHz y 20 MHz dependiendo del espectro disponible

- El ancho de banda de 20 MHz puede proporcionar una velocidad de datos de usuario de enlace descendente de hasta 150 Mbps con 2×2 MIMO y 300 Mbps con 4×4 MIMO
- La velocidad máxima de datos del enlace ascendente es de 75 Mbps. (p. 5)

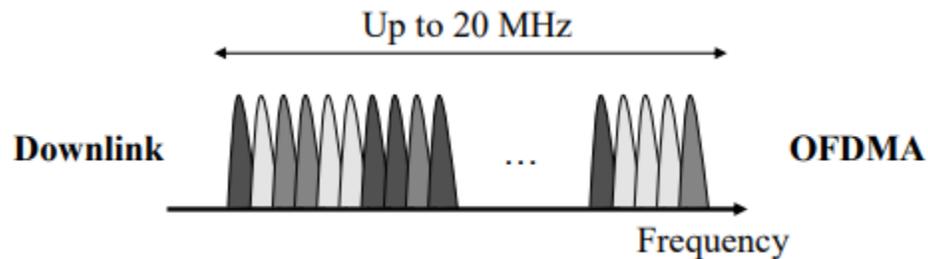
2.1.3.1. Técnicas De Acceso Múltiple Por División De Frecuencia Ortogonal OFDMA

Para el enlace descendente, LTE usa una variante de OFDM conocido como acceso múltiple por división de frecuencia ortogonal (OFDMA). Según (Rumney, 2013) con el estándar OFDM, las asignaciones de subportadoras son fijas para cada usuario por lo tanto el rendimiento puede presentar interferencia de banda estrecha.

OFDMA incorpora elementos de acceso múltiple por división de tiempo (TDMA) para que las subportadoras puedan asignarse dinámicamente entre los diferentes usuarios del canal. Esto da como resultado un sistema robusto con mayor capacidad. Esta capacidad se deriva de la eficiencia de la troncalización obtenida al multiplexar usuarios en un canal más amplio de baja velocidad proporcionando capacidad dinámica, y la solidez surge de la capacidad de programar a los usuarios por frecuencia para evitar la interferencia de banda estrecha y el desvanecimiento de trayectos múltiples, la figura 3 muestra el esquema de múltiple acceso al medio OFDMA. (p. 62)

Figura 3

Esquema de acceso múltiple LTE (OFDMA)



Nota. La figura muestra el esquema de acceso múltiple por división de frecuencia ortogonal (OFDMA) para el enlace descendente. Adaptado de *LTE multiple access schemes* (p. 6), de (Holma y Toskala, 2009), *LTE for UMTS – OFDMA and SC-FDMA Based Radio Access*. John Wiley & Sons Ltd.

2.1.3.2. Técnica De Acceso Múltiple Por División De Frecuencia De Portadora Única SC-FDMA

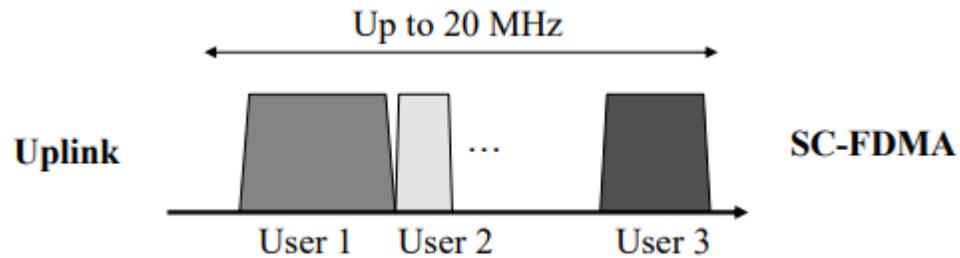
En cuanto a la técnica de acceso múltiple por división de frecuencia de portadora única SC-FDMA, (Rumney, 2013) afirma que:

Debido a la alta relación de la potencia pico a promedio, por sus siglas (PAPR) asociada con OFDM, condujo a 3GPP a investigar un esquema de modulación alternativo para el enlace ascendente LTE.

Por lo que se selecciona a SC-FDMA debido a que combina las técnicas PAPR bajas de los sistemas de transmisión de una sola portadora, tal como GSM y CDMA, con la resistencia de multi ruta y además a la asignación de frecuencia flexible de OFDMA, la figura 4 muestra el esquema de múltiple acceso al medio SC - FDMA. (p. 62)

Figura 4

Esquema de acceso múltiple LTE (SC - FDMA)



Nota. La figura muestra el esquema de acceso múltiple por división de frecuencia de portadora única (SC - FDMA) para el enlace ascendente. Adaptado de *LTE multiple access schemes* (p. 6), de (Holma y Toskala, 2009), *LTE for UMTS – OFDMA and SC-FDMA Based Radio Access*. John Wiley & Sons Ltd.

2.2. Arquitectura De La Tecnología LTE

La capacidad alta de la red requiere de una arquitectura eficiente, así como también de las funciones de radio avanzadas. Por lo que (Holma y Toskala, 2009) afirman que:

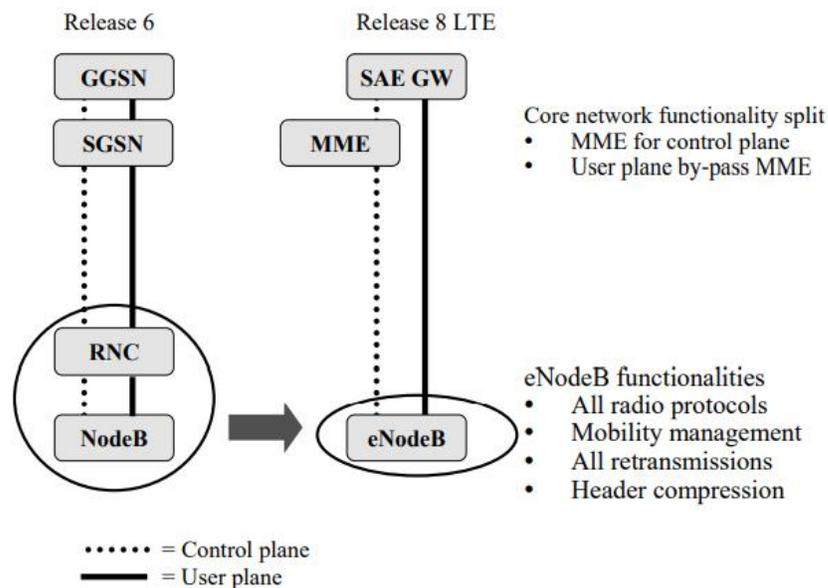
Dentro del objetivo del Release 8 de 3GPP era mejorar la escalabilidad de la red para aumentar el tráfico y a su vez minimizar la latencia de un extremo a otro con la reducción del número de elementos de la red. Los protocolos de radio, la gestión de la movilidad, la compresión de encabezados y todas las retransmisiones de paquetes se encuentran en las estaciones base denominadas evolved NodeB (eNodeB).

Por lo que, eNodeB incluye todos aquellos algoritmos que se encuentran en Radio Network Controller – Controlador de la Red Radio (RNC) en la arquitectura 3GPP del Release 6. Además, la red central se simplifica al separar el usuario y los planos de control.

La Mobility Management Entity - Entidad de Gestión de Movilidad (MME) es solo el elemento del plano de control, mientras que el plano de usuario pasa por alto la MME directamente a la puerta de enlace (GW) del System Architecture Evolution - Evolución de la Arquitectura del Sistema (SAE). La evolución de la arquitectura se ilustra en la Figura 5. Esta red central del Release 8 también se conoce como Evolved Packet Core - Núcleo de Paquete Evolucionado (EPC), mientras que para todo el sistema también se usa el término Evolved Packet System – Sistema de Paquetes Evolucionado (EPS). (p. 6)

Figura 5

Arquitectura de la red LTE



Nota. La figura muestra la evolución de la arquitectura de la red LTE. Tomado de *LTE network architecture* (p. 6), por (Holma y Toskala, 2009), *LTE for UMTS – OFDMA and SC-FDMA Based Radio Access*. John Wiley & Sons Ltd.

2.2.1. Access Network (Red de Acceso) – E-UTRAN

En la E-UTRAN (Evolved Universal Terrestrial Radio Access Network – Red Universal de Acceso Radio Terrestre Evolucionada) al único nodo se conoce como el Nodo B de la E-UTRAN (eNodeB), por lo tanto, se lo define como una estación base de radio que se encarga de controlar todas las funciones relacionadas con la radio en el sistema.

Las estaciones base, como el eNodeB, por lo general se distribuyen a lo largo del área de cobertura de las redes y cada eNodeB reside en un lugar cercano de las antenas de radio reales.

En cuanto al funcionamiento, eNodeB trabaja como un puente de capa 2 entre el User Equipment – Equipo de Usuario (UE) y el EPC, siendo el punto final de los protocolos de radio hacia el UE y retransmitiendo datos entre la conexión de radio y la conectividad basada en IP hacia el EPC.

En esta función, el eNodeB cifra y descifra los datos UP, simultáneamente comprime y descomprime el encabezado IP para así evitar el reenvío de los mismos datos o datos de secuencia en el mismo encabezado. (Holma y Toskala, 2009, pág. 27)

Por lo tanto y en base a lo mencionado anteriormente, se determinan las siguientes funciones importantes realizadas por el eNodeB, según (Holma y Toskala, 2009) los cuales afirman lo siguiente:

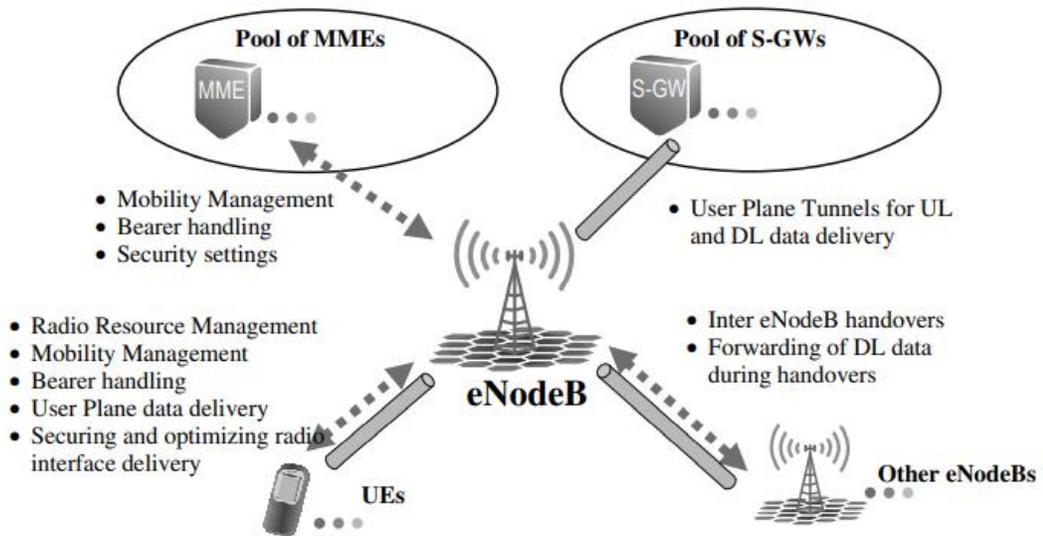
- Controla el uso de la interfaz de radio, en conjunto con la responsabilidad de la Radio Resource Management - Gestión de Recursos de Radio (RRM), es decir, asigna recursos en función de las solicitudes, prioriza y programa el tráfico en base al requerimiento de la Quality of Service - Calidad de Servicio (QoS) junto con el monitoreo constante del uso de los recursos.

- Además, es parte fundamental en la Mobility Management - Gestión de la Movilidad (MM). Se encarga de realizar un control y análisis de las mediciones obtenidas del nivel de la señal de radio del UE, con el fin de determinar el traspaso entre celdas a los UE y la señalización de traspaso de eNodeB y el MME.
- Cuando un nuevo UE se activa bajo el eNodeB y solicita conectarse a la red, el eNodeB es el encargado de enrutar la solicitud al MME anterior que sirvió al UE, o seleccionar un nuevo MME.

La Figura 6 muestra las conexiones que tiene eNodeB con los nodos lógicos circundantes y resume las funciones principales en estas interfaces.

Figura 6

Conexiones eNodeB a otros nodos lógicos y funciones principales



Nota. La figura muestra las conexiones y la relación que pueden tener sea de uno a muchos o de muchos a muchos. El eNodeB puede dar servicio a varios UE en su área de cobertura, pero cada UE está conectado a un solo eNodeB. Tomado de *eNodeB connections to other logical nodes*

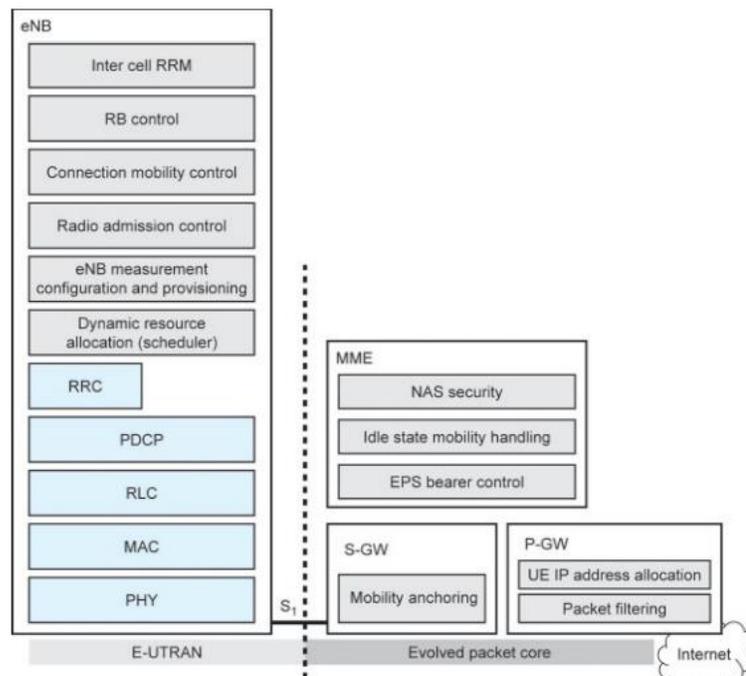
and main functions (p. 28), por (Holma y Toskala, 2009), LTE for UMTS – OFDMA and SC-FDMA Based Radio Access. John Wiley & Sons Ltd.

2.2.2. Evolved Packet Core (Núcleo de Red) – EPC

En cuanto a la arquitectura del EPC (Evolved Packet Core – Núcleo de Red en Conmutación de Paquetes Evolucionado), está compuesta por las entidades de la red e interfaces, las entidades de red son las que forman el núcleo y las entidades de red e interfaces se encargan del control del servicio de la conectividad. En la figura 7, se muestra el núcleo de la red troncal EPC compuesta por las entidades: MME (Mobility Management Entity), Serving Gateway – Puerta de Enlace de Servicio (S-GW) y Packet Data Network Gateway – Puerta de Enlace de Red de Paquetes de Datos (P-GW). (Xiang, 2018, pág. 24)

Figura 7

Reparto funcional entre E-UTRAN y EPC



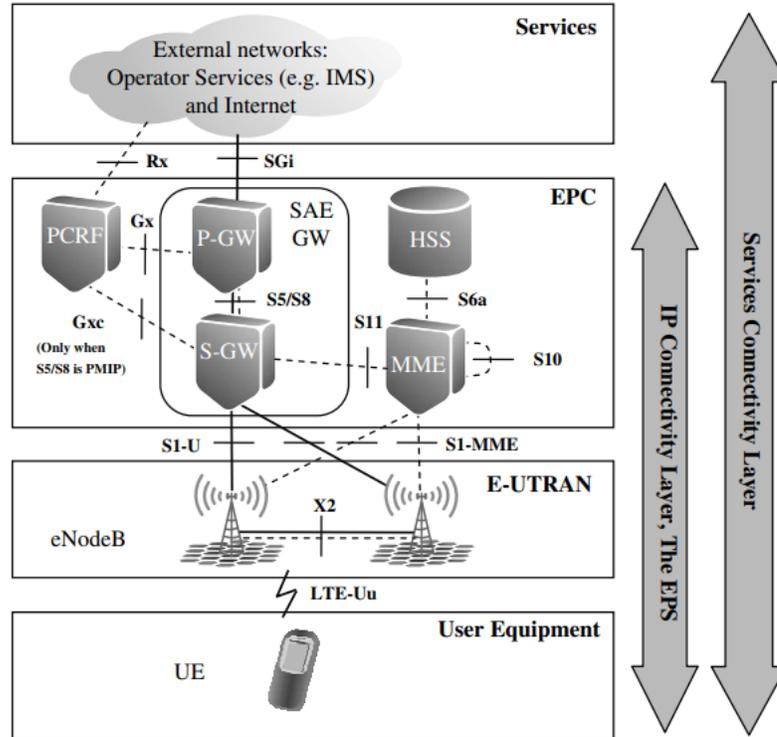
Nota. La figura muestra la arquitectura de la red troncal EPC formada por sus principales entidades. Tomado de *Functional split between E-UTRAN and EPC* (p. 206), por (Rumney, 2013), *LTE and the Evolution to 4G Wireless Design and Measurement Challenges*, Second Edition. John Wiley & Sons, Ltd.

2.2.2.1. Mobility Management Entity (MME)

La Entidad (MME) de Gestión de la Movilidad, es el elemento fundamental de control en el EPC. Generalmente, el MME es un servidor ubicado en el operador del plano de control (CP). Así como también de las interfaces que terminan en el MME en la arquitectura, tal como se muestra en la Figura 8.

Figura 8

Arquitectura del sistema para red E-UTRAN



Nota. La figura muestra la arquitectura del sistema E-UTRAN y las interfaces en el MME además la conexión de CP directamente con el UE, usada como canal de control principal entre el UE y la red. Tomado de *System architecture for E-UTRAN only network* (p. 25), por (Holma y Toskala, 2009), LTE for UMTS – OFDMA and SC-FDMA Based Radio Access. John Wiley & Sons Ltd.

Las funciones principales de MME en la configuración básica de la arquitectura, se detallan a continuación:

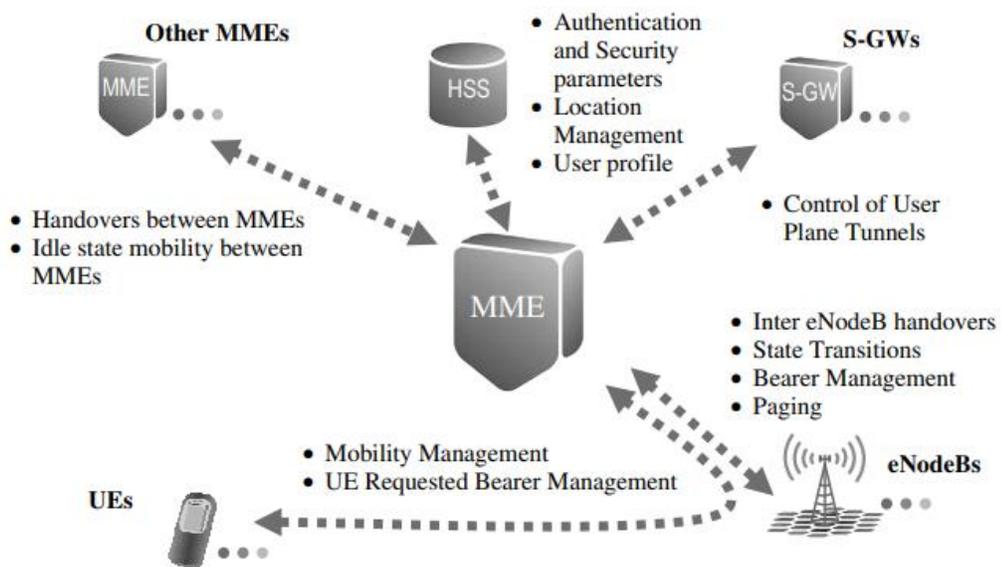
- **Autenticación y seguridad:** al registrarse por primera vez un UE, el MME inicia la autenticación, es decir revela la identidad permanente del UE de la red visitada previamente o del propio UE. Esto da paso a la solicitud del Home Subscriber Server - Servidor de Suscripción Doméstico (HSS) en la red del UE a los vectores de autenticación que comprenden los parámetros de desafío de autenticación o respuesta y procede a enviar el desafío al UE con el fin de comparar la respuesta recibida del UE con la de la red doméstica, dicha función sirve para garantizar que el UE sea quien dice ser. En ese sentido el MME puede repetir la autenticación si es necesario o de manera periódica.
- **Gestión de Movilidad:** En este aspecto el MME procede a realizar un seguimiento de todos los UE que están ubicados en su área de servicio, al realizar su primer registro en la red por parte del UE, el MME crea una entrada para el UE y se encarga de señalar la ubicación al HSS en la red local del UE. Es decir, el MME elabora una solicitud para que se configuren los recursos adecuados en el eNodeB. Luego, el MME continúa rastreando la ubicación del UE en el nivel de eNodeB y en caso de

permanecer conectado el UE o en constante comunicación activa, o en el nivel en modo inactivo ya no es necesario mantener una ruta de datos conectados.

En la figura 9 se muestra las conexiones que tiene MME con los nodos lógicos circundantes y resume las funciones principales de estas interfaces. (Holma y Toskala, 2009, pág. 28)

Figura 9

Conexiones MME a otros nodos lógicos y funciones principales



Nota. La figura muestra las conexiones MME a otros nodos lógicos, el MME se puede conectar a otro MME del sistema, pero la conectividad se limita a la red de un solo operador. La conexión entre MME con MME vecinos se utiliza en traspasos. Tomado de *MME connections to other logical nodes and main functions* (p. 30), por (Holma y Toskala, 2009), LTE for UMTS – OFDMA and SC-FDMA Based Radio Access. John Wiley & Sons Ltd.

2.2.2.2. Serving Gateway (S-GW)

La función de la puerta de enlace de servicio (S-GW) es la gestión y conmutación del túnel UP. El S-GW es la puerta de enlace que termina la interfaz EPC hacia la E-UTRAN, es parte de la infraestructura de red mantenida centralmente en las instalaciones de operación.

El S-GW está implícito de manera secundaria en las funciones de control, es responsable de sus propios recursos y los designa en función de las solicitudes del MME, P-GW o Policy and Charging Rules Function – Función de Reglas de Política y Cobro (PCRF), estos que a su vez actúan necesariamente para configurar, modificar o liberar portadores para el UE. Si la solicitud se recibe del P-GW o PCRF, S-GW también transmitirá el comando al MME para que tome control del túnel al eNodeB.

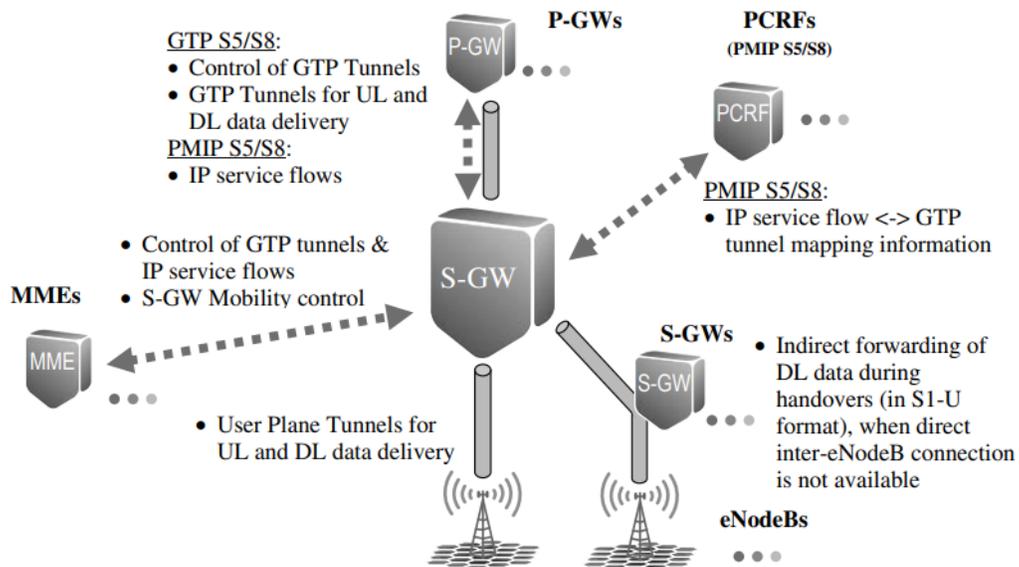
El MME señala al S-GW que modifique el túnel de un eNodeB a otro, también puede solicitar al S-GW que genere recursos de tunelización cuando sea necesario para el reenvío de datos desde el eNodeB de origen al eNodeB de destino durante el tiempo que el UE realiza el traspaso de radio.

En la figura 10 se muestra la conexión del S-GW a otros nodos lógicos y las funciones principales de estas interfaces, por lo que se requiere una configuración uno a muchos. Dicho S-GW solo puede prestar servicio a un área geográfica específica con un conjunto limitado de eNodeB y, de manera similar, puede tener un conjunto limitado de MME que controlen esa área. El S-GW debe poder conectarse a cualquier P-GW en toda la red, porque el P-GW no modificará en movimiento, mientras que el S-GW puede trasladarse cuando el UE se mueva. Para las conexiones que involucren al UE, el S-GW siempre señalará con un solo MME y el UP apuntará a un eNodeB a la vez.

Además, en la misma figura se da a conocer el caso de reenvío indirecto de datos UP los cuales se reenvían entre los eNodeB mediante los S-GW. No existe un nombre de interfaz específico asociado a la interfaz entre S-GW debido a que el formato es el mismo que en la interfaz S1-U, y los S-GW relacionados pueden considerar que se están comunicando directamente con un eNodeB. Este puede ser el caso en el que si el reenvío indirecto de datos se realiza a través de un solo S-GW, es decir, ambos eNodeB se pueden conectar al mismo S-GW. (Holma y Toskala, 2009, pág. 30)

Figura 10

Conexiones S-GW a otros nodos lógicos y funciones principales



Nota. Tomado de *S-GW connections to other logical nodes and main functions* (p. 31), por (Holma y Toskala, 2009), LTE for UMTS – OFDMA and SC-FDMA Based Radio Access. John Wiley & Sons Ltd.

2.2.2.3. Packet Data Network Gateway (P-GW)

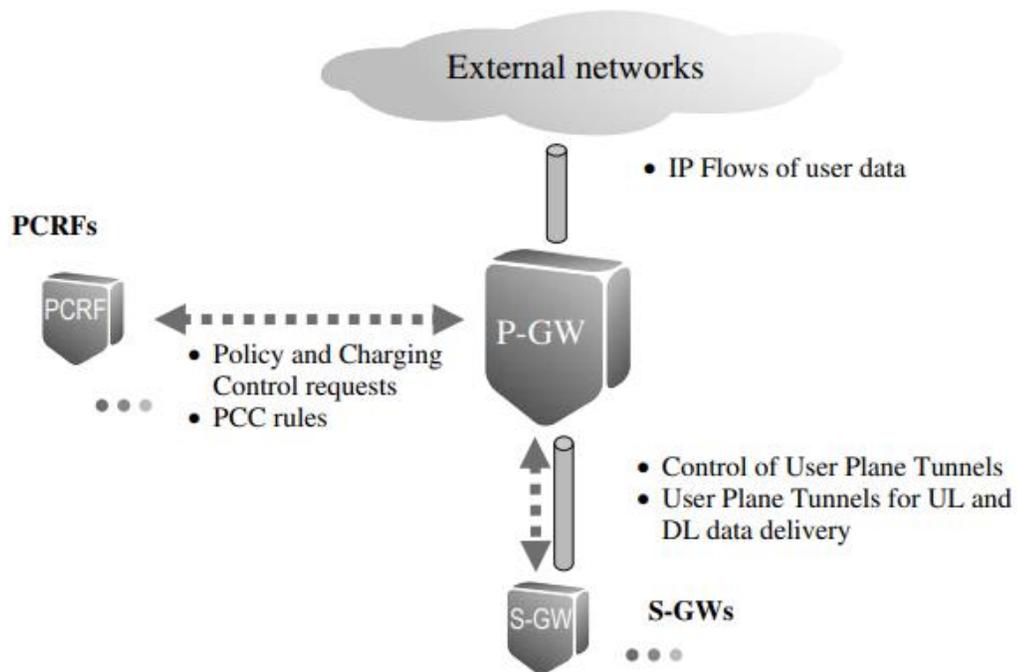
La puerta de enlace de la red de paquetes de datos (P-GW), también conocida como (PDN-GW) es el router de borde entre el EPS y la red externa de paquetes de datos. Este es el ancla de movilidad de nivel superior en el sistema, cuando un UE va de un S-GW a otro, los portadores deben cambiarse en el P-GW, el cual recibirá una señal para cambiar los flujos del nuevo S-GW. Generalmente trabaja como punto de conexión IP para el UE y realiza funciones de filtrado de tráfico y control.

Normalmente, el P-GW asigna una dirección IP al UE, la cual es usada para la comunicación con otros hosts IP en redes externas. También se da el caso en el que la PDN externa conectada al UE asigne la dirección que utilizará y por tanto la P-GW tuneliza todo el tráfico a la red. El P-GW realiza la función del Protocolo de configuración dinámica de host (DHCP), o a su vez consulta en un servidor DHCP externo para entregar la dirección al UE. El P-GW establece portadores en función de la solicitud a través de PCRF o del S-GW, que transmite información desde el MME.

En la Figura 11 se pueden ver las conexiones de los nodos lógicos circundantes que tiene el P-GW, donde un mismo nodo puede tener una o más conexiones al S-GW, PCRF o red externa. (Holma y Toskala, 2009, pág. 31)

Figura 11

Conexiones P-GW a otros nodos lógicos y funciones principales



Nota. Tomado de *P-GW connections to other logical nodes and main functions* (p. 33), por (Holma y Toskala, 2009), *LTE for UMTS – OFDMA and SC-FDMA Based Radio Access*. John Wiley & Sons Ltd.

2.2.2.4. Policy and Charging Rules Function (PCRF)

El PCRF es el encargado del control de las políticas y normas de función de carga, denominadas reglas (PCC), se basa en la toma de decisiones para el manejo de los servicios en términos de Calidad de Servicio (QoS).

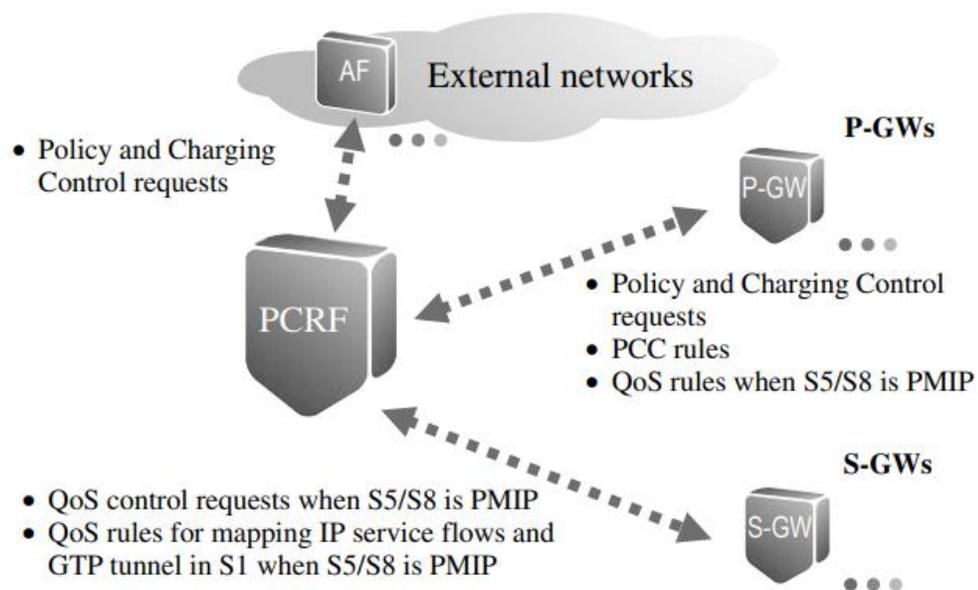
Cada vez que se establezca un nuevo portador la PCRF enviará las reglas de PCC.

Por lo tanto, es requerida la configuración del portador, cuando el UE se conecta inicialmente a la red y se configurará el portador predeterminado. La PCRF proporciona reglas de PCC en función de la solicitud del P-GW y del S-GW.

Los nodos conectados en conjunto con la PCRF se muestran en la Figura 12 donde se puede observar que cada PCRF puede tener relación con uno o más P-GW y S-GW, donde existe solo una PCRF relacionada a cada conexión PDN del UE. (Holma y Toskala, 2009, pág. 32)

Figura 12

Conexiones PCRF a otros nodos lógicos y funciones principales



Nota. Tomado de *PCRF connections to other logical nodes and main functions* (p. 33), por (Holma y Toskala, 2009), LTE for UMTS – OFDMA and SC-FDMA Based Radio Access. John Wiley & Sons Ltd.

2.2.2.5. Home Subscriber Server (HSS)

El Servidor de Abonado (HSS) se lo conoce como un servidor de base de datos ubicado en las instalaciones del operador local. Por lo tanto, (Penttinen, 2015) afirma lo siguiente:

El HSS al ser un servidor contiene una base de datos de todos los suscriptores del operador, esta base de datos tiene información tanto de las identidades de los suscriptores como los detalles de las suscripciones (registro de ubicación de inicio, funcionalidad). El HSS tiene la responsabilidad de generar claves de seguridad usadas por el MME con el fin de autenticar a los suscriptores, para el cifrado y la protección de la integridad (Centro de Autenticación, AuC).

En el transcurso de la conexión del terminal de usuario a la red, el MME se comunica con el HSS central para confirmar la identidad del suscriptor y determina las opciones de comunicación autorizadas, por ejemplo, el perfil de calidad de servicio. Por ende, el HSS debe tener conexiones con cada MME en la red del operador y también puede realizar un rastreo de la posición del terminal del usuario por el MME. (p. 427)

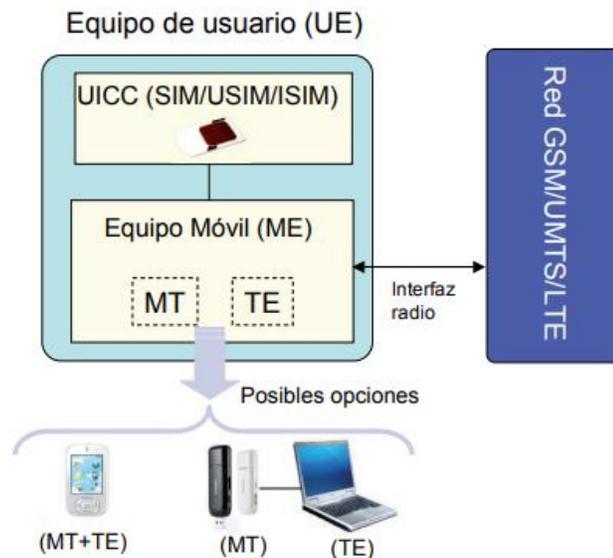
2.2.3. User Equipment (UE)

Para poder tener acceso a la red LTE por la interfaz radio, se utilizan equipos de usuario (UE). En sí, es un dispositivo que el usuario final utiliza para la comunicación. Generalmente, este terminal puede ser un dispositivo portátil como un teléfono inteligente o una tarjeta de datos, como las que se usan en 2G y 3G, o a su vez podría estar integrado por ejemplo a una computadora portátil. El equipo de usuario consta de las siguientes partes importantes, como el Módulo de identidad del suscriptor universal (USIM) que es un módulo separado del resto del UE, también conocido como Equipo terminal (TE), se encuentra en una tarjeta inteligente extraíble denominada Tarjeta de circuito integrado universal (UICC). USIM identifica y autentica al usuario con el fin de obtener claves de seguridad para la protección de la transmisión de la interfaz de radio.

El equipo de usuario, a nivel funcional se describe como una plataforma para aplicaciones de comunicación con la red, para el establecimiento, mantenimiento y eliminación de los enlaces de comunicación requerido por el usuario final. En conjunto con funciones de gestión de movilidad en base a la ubicación de los terminales donde opera el UE de acuerdo con las instrucciones de la red. La figura 13 muestra la arquitectura funcional de un equipo de usuario con sus elementos básicos. (Holma y Toskala, 2009, pág. 26)

Figura 13

Equipo de Usuario



Nota. Tomado de *Equipo de Usuario* (p. 107), de (Agusti Comes et al., 2010), LTE: Nuevas Tendencias en Comunicaciones Móviles. Fundación Vodafone España.

2.3. Interfaces de red E-UTRAN

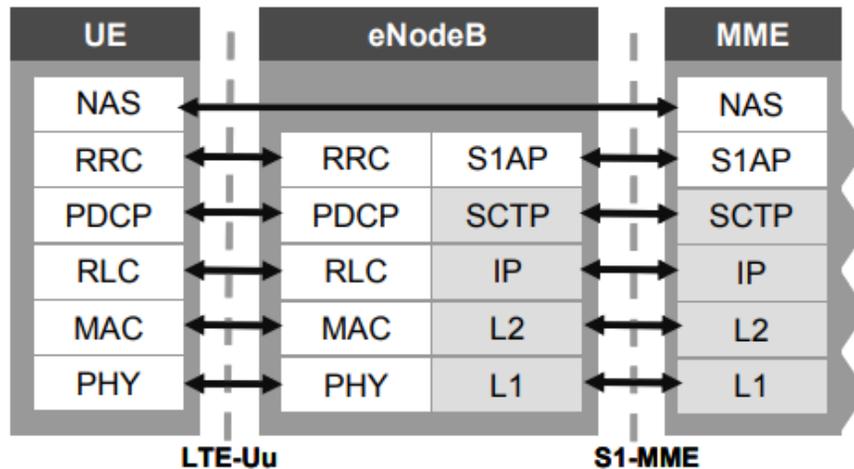
Dentro del desarrollo de 3GPP en el Release 8, se ha establecido la definición del soporte para la autoconfiguración de las interfaces S1 y X2. El proceso se comprende, donde el eNodeB una vez encendido y con conexión IP se conectará basado en la dirección IP conocida para así obtener parámetros adicionales en términos de otros elementos de la red para conectarse y también para la descarga del software eNodeB, así como parámetros iniciales para la operación, como la banda de frecuencia de operación y qué otros parámetros añadir para los canales de transmisión.

2.3.1. Interfaces S1

Por medio de la interfaz S1, el eNodeB se conecta a la red troncal EPC. En este sentido el CP (Control Plane – Plano de Control), conocido también como S1-MME, es utilizado para soportar tanto funciones como procedimientos de control entre eNodeBs y a su vez a la red troncal de la entidad MME. Por lo tanto, en la Figura 14 se observa los protocolos del CP (Plano de Control) que están relacionados con la conexión del UE (User Equipment – Equipo de Usuario) a una PDN (Packet Data Networks – Red de Paquete de Datos). Las interfaces de un único MME muestra los protocolos hacia la E-UTRAN y el UE.

Figura 14

Pila de protocolos del plano de control con la conexión de un UE a una PDN



Nota. Adaptado de *Control plane protocol stack in EPS* (p. 36), de (Holma y Toskala, 2009), LTE for UMTS – OFDMA and SC-FDMA Based Radio Access. John Wiley & Sons Ltd.

La capa superior en el plano de control es el NAS por sus siglas, que significa (Non Access Stratum – Estrato De No Acceso), la misma que está compuesto por dos protocolos separados que se llevan a cabo en el transporte de señalización directo entre el UE y el MME. En los protocolos de la capa NAS, el contenido no es visible para el eNodeB, por lo que quiere decir que el eNodeB no forma parte de ese proceso por cualquier otro medio. Además de estar a cargo del transporte de los mensajes y de brindar indicaciones adicionales referente a la capa de transporte en conjunto con los mensajes de ser el caso. Los protocolos de la capa NAS tienen otras características que se detallan a continuación:

- **Gestión de Movilidad EPS (EMM):** El protocolo EMM es el que está a cargo de la gestión de la movilidad del UE en el sistema. Además, tiene funciones para realizar la conexión y desconexión de la red y realizar constantes actualizaciones de ubicación en el medio, denominada TAU (Tracking Area Updating - Actualización del área de seguimiento) por lo que ocurre en un estado inactivo. Y debe tomar en cuenta que

los trasposos en el modo conectado son controlados por los protocolos de la capa inferior, pero la capa EMM tiene funciones para reactivar el UE estando en el modo inactivo.

- **Gestión de sesión EPS (ESM):** se utiliza a este protocolo para la gestión específicamente de portadores entre el UE y el MME, además es utilizada para los procedimientos de gestión de portadores de E-UTRAN. Tomar en consideración que el propósito no es utilizar los procedimientos ESM si el contexto del portador ya está disponible en la red y los procedimientos E-UTRAN se pueden ejecutar de manera inmediata.

Como se mencionó anteriormente, la interfaz S1 es la encargada de conectar la E-UTRAN al EPC, por lo tanto, involucra los siguientes protocolos:

- **S1 Application Protocol - Protocolo de aplicación S1 (S1AP):** es la que conduce las conexiones en el plano de control (CP) y en el plano de usuario (UP) del UE entre E-UTRAN y EPC, incluyendo la intervención en el traspaso cuando EPC está relacionado.

- **SCTP/IP Signalling Transport - Transporte de señalización SCTP/IP:** El Stream Control Transmission Protocol - Protocolo de Transmisión de Control de Flujo (SCTP) y el Internet Protocol - Protocolo de Internet (IP) son métodos de transporte IP estándar para los mensajes de señalización. Proporciona funciones seguras de transporte y entrega secuenciada.

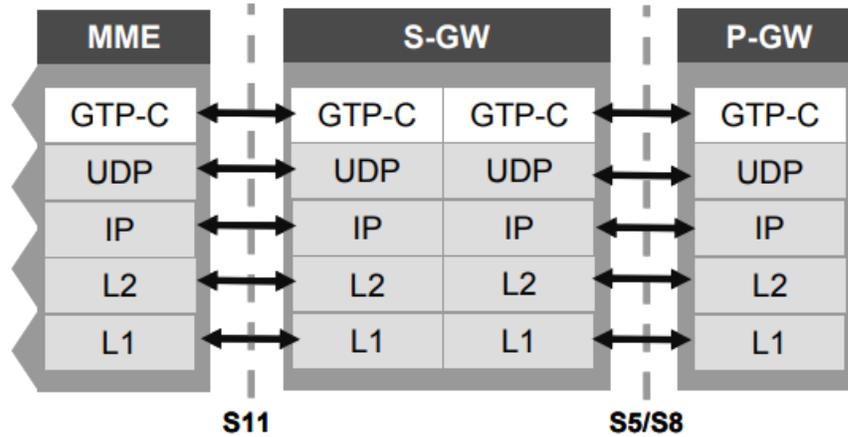
Por otra parte, los protocolos que intervienen en la interfaz radio entre el equipo de usuario y el eNodeB, se describen a continuación:

- **Radio Resource Control - Control de recursos de radio (RRC):** este protocolo es el encargado de controlar el uso de recursos de radio, es decir gestiona las conexiones de datos y la señalización del UE añadiendo también funciones para el traspaso.
- **Packet Data Convergence Protocol - Protocolo de convergencia de paquetes de datos (PDCP):** las funciones principales están enfocadas en la compresión de encabezado IP (UP), el cifrado y la protección de la integridad, solo para el plano de control (CP).
- **Radio Link Control - Control de enlace de radio (RLC):** es el responsable de la segmentación y concatenación de las PDCP-PDU para la transmisión de la interfaz de radio, así como también realiza la corrección de errores con el método de Automatic Repeat Request – Solicitud de Repetición Automática (ARQ).
- **Medium Access Control - Control de acceso al medio (MAC):** la capa MAC es la responsable de programar los datos en base a las prioridades y multiplexa los datos a los bloques de transporte de la capa física, proporciona corrección de errores con Hybrid ARQ.
- **Physical Layer - Capa física (PHY):** denominada capa 1 de la interfaz de radio LTE-Uu y se ocupa de las funciones de la capa de acceso múltiple por división de código en secuencia directa (DS-SS)

En la figura 15, se muestran los protocolos hacia las puertas de enlace y representan tecnologías estándar de Internet que se utilizan para el transporte en EPS. En el EPC, hay dos protocolos para la interfaz S5/S8 y están involucrados cuando se usa GPRS Tunneling Protocol - Protocolo de Tunelización GPRS (GTP).

Figura 15

Pila de protocolos del plano de control hacia las puertas de enlace



Nota. Adaptado de *Control plane protocol stack in EPS* (p. 36), de (Holma y Toskala, 2009), LTE for UMTS – OFDMA and SC-FDMA Based Radio Access. John Wiley & Sons Ltd.

- **GPRS Tunneling Protocol, Control Plane - Protocolo de Tunelización**

GPRS, Plano de Control (GTP-C): se ocupa de las conexiones enfocadas en el plano de usuario en el EPC, añadiendo la señalización de la calidad del servicio (QoS) junto con otros parámetros. Al utilizar GTP en la interfaz S5/S8, controla los túneles del plano de usuario de GTP-U. De esta forma, GTP-C realiza funciones de gestión de la movilidad, por ejemplo, cuando los túneles GTP-U de un UE necesitan transferirse de un nodo a otro.

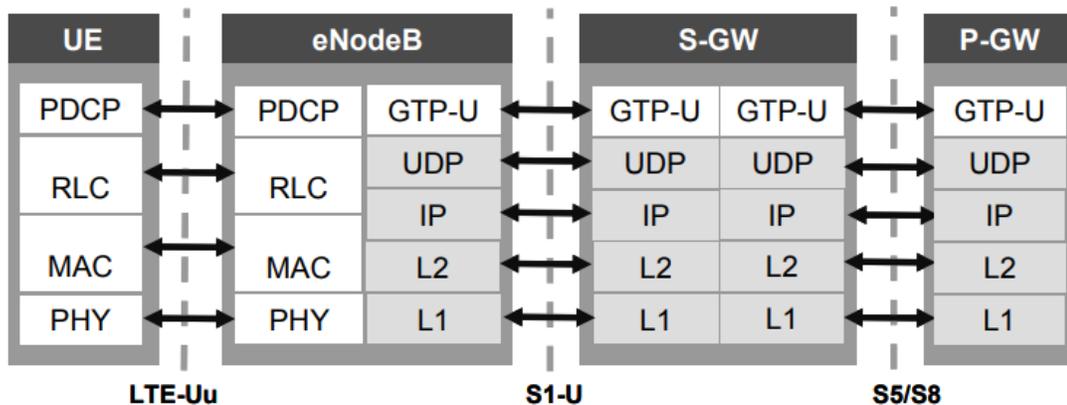
- **UDP/IP Transport - Transporte UDP/IP:** El protocolo de datos de la unidad (UDP) e IP son utilizados como transporte IP estándar y básico. Se utiliza UDP en lugar del Protocolo de Control de Transmisión (TCP) porque las capas superiores son capaces de proporcionar un transporte fiable con recuperación de errores y retransmisión.

Estos paquetes IP en EPC se pueden enviar a través de diferentes tecnologías de Capa 2 y Capa 1, como el Ethernet.

La estructura del protocolo plano de usuario (UP) para el equipo de usuario (UE) que se conecta a la puerta de enlace de la red de paquetes de datos (P-GW), se ilustra en la Figura 16. El UP incluye las capas debajo de la IP del usuario final, es decir, estos protocolos forman la Capa 2 usada para transportar paquetes IP del usuario final.

Figura 16

Pila de protocolos del plano de usuario en EPS



Nota. Tomado de *User plane protocol stack in EPS* (p. 38), de (Holma y Toskala, 2009), LTE for UMTS – OFDMA and SC-FDMA Based Radio Access. John Wiley & Sons Ltd.

La estructura del protocolo es semejante al plano de control (CP). Esto enfatiza en el hecho de que todo el sistema está diseñado para el transporte de datos en paquetes genéricos, por lo tanto, la señalización CP como los datos UP son, en definitiva, datos en paquetes y se diferencian solo en los volúmenes. La mayoría de los protocolos se han descrito anteriormente, excepto los dos siguientes del conjunto de protocolos en la interfaz S5/S8:

- **GPRS Tunnelling Protocol, User Plane - Protocolo de tunelización**

GPRS, Plano de Usuario (GTP-U): se utiliza cuando las interfaces S5/S8 se basa en GTP. Y crea un túnel GTP-U, que se emplea para el envío de paquetes IP de usuarios finales pertenecientes al portador EPS. Es utilizado en la interfaz S1-U y en S5/S8 si el CP hace uso de GTP-C.

- **Generic Routing Encapsulation - Encapsulación de Enrutamiento**

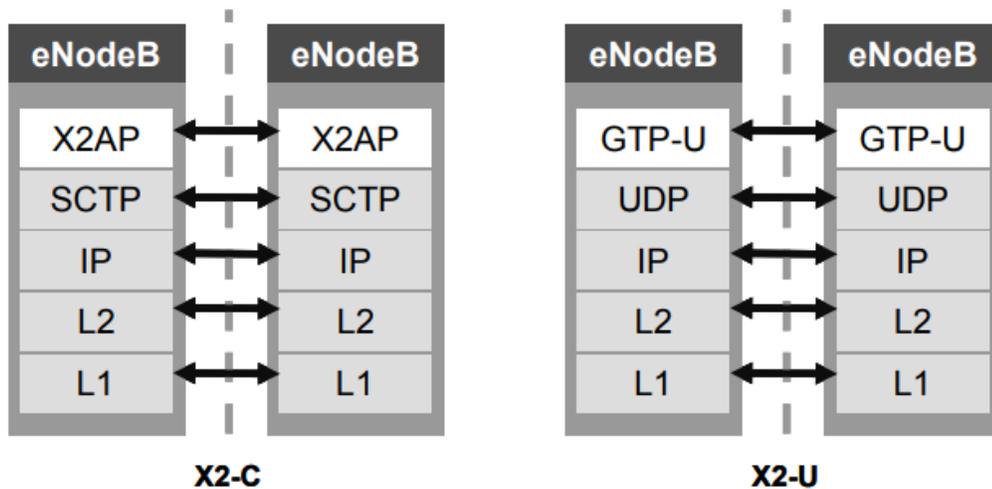
Genérico (GRE): es empleada en la interfaz S5/S8, se forma un túnel IP para el traslado de todos los datos relacionados con la conexión del UE a su propia PDN. GRE está encima de IP y no se utiliza UDP.

2.3.2. Interfaces X2

La estructura del protocolo de la interfaz X2 se asemeja a la de la interfaz S1, esto se ilustra en la figura 17. Con la diferencia que el Protocolo de aplicación del plano de control (CP) es diferente.

Figura 17

Pila de protocolos de control y plano de usuario para la interfaz X2



Nota. Tomado de *Control and user plane protocol stacks for X2 interface* (p. 38), de (Holma y Toskala, 2009), LTE for UMTS – OFDMA and SC-FDMA Based Radio Access. John Wiley & Sons Ltd.

La interfaz X2 se enfoca en la movilidad entre los eNodeBs y el X2 Application Protocol – Protocolo de Aplicación X2 (X2AP) introduce las funciones para la preparación del traspaso y para el mantenimiento a nivel general de la relación entre los eNodeB vecinos. En la interfaz X2 el UP es usado para el reenvío de datos en un estado temporal durante el traspaso, cuando la interfaz de radio se desconecta del origen y no ha continuado en el destino. Dicho reenvío de datos se realiza para Downlink - Enlace Descendente (DL), debido a que el UE regula de manera efectiva el Uplink – Enlace Ascendente (UL). (Holma y Toskala, 2009, pág. 35)

2.4. Radio Definido Por Software (SDR)

La constante evolución en el campo SDR ha conllevado a la escalada del desarrollo de protocolos y un amplio espectro de aplicaciones. Es decir, SDR tiene una gran importancia tanto en campos como la academia y la industria. Por lo que el objetivo para los diseñadores de SDR es simplificar la realización de protocolos de comunicación al tiempo que permiten a los investigadores experimentar con prototipos en la red desplegada. Para la definición de Radio Definido por Software, se considera al SDR Fórum.

2.4.1. Definición de Radio Definido por Software (SDR)

El SDR Fórum en colaboración junto con el grupo P1900.1 del Instituto de Ingenieros Eléctricos y Electrónicos (IEEE), define al Software Defined Radio - Radio Definido por Software (SDR) como: “Radio en la cual algunas o todas las funciones de la capa física son definidas mediante Software” (SDR Forum, 2007, pág. 1).

Mencionando al radio como un dispositivo el cual transmite o recibe señales de forma inalámbrica en la parte de radiofrecuencia (RF) del espectro electromagnético con el fin de facilitar la transferencia de información.

Los dispositivos de radio tradicionales basados en hardware tiene limitaciones en la funcionalidad cruzada y solo pueden modificarse mediante intervención física. Por lo que conlleva a costos de producción más altos y una flexibilidad mínima para admitir múltiples estándares de forma de onda. Por lo que, la tecnología de radio definida por software proporciona una solución eficiente y económica, lo que permite a dispositivos inalámbricos como multimodo, multibanda o multifuncionales que pueden mejorarse mediante actualizaciones de software. (SDR Forum, 2007)

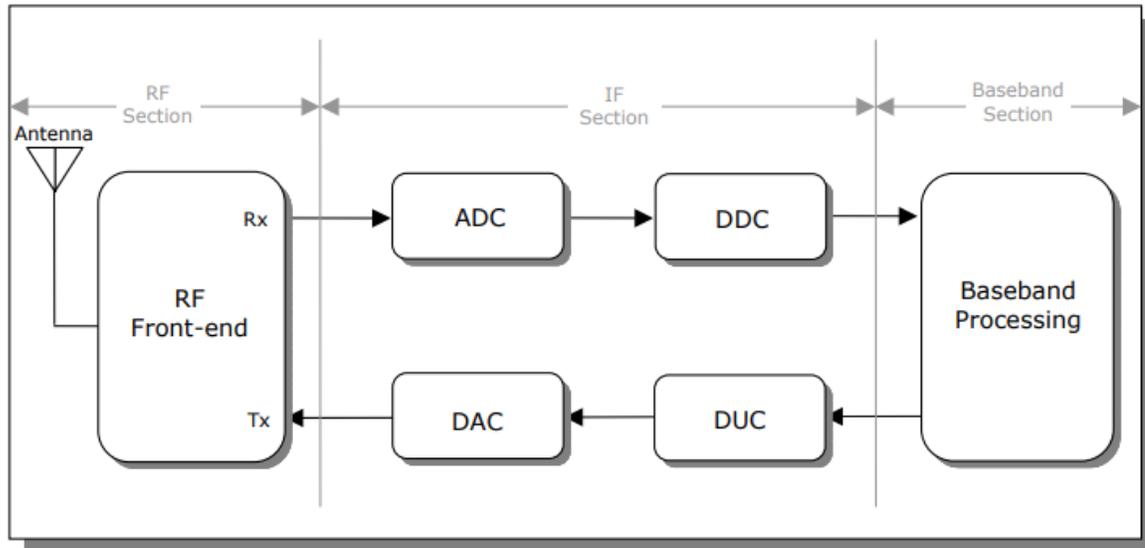
2.4.2. Características De Radio Definido Por Software (SDR)

El SDR se refiere a la tecnología en la que los módulos de software que se ejecutan en una plataforma de hardware genérica, la misma que consta de procesadores de señales digitales (DSP) y microprocesadores de uso general, se utilizan para implementar funciones de radio, como la generación de señales transmitidas (modulación) en el transmisor y la sintonización o detección de la señal de radio recibida (demodulación) en el receptor.

En el esquema básico para SDR, el cual se muestra en la figura 18, detalla los bloques funcionales en un sistema genérico de transceptor de radio digital (transmisor/receptor). El sistema de radio digital consta de tres bloques: la sección de RF, la sección de IF y la sección de Banda Base. En la sección de RF consta de módulos de hardware analógicos, mientras que las secciones de IF y de banda base contienen módulos de hardware digitales.

Figura 18

Diagrama de bloques de un transceptor digital genérico



Nota. Tomado de *Block diagram of a generic digital transceiver* (p. 6), de (Wipro Technologies, 2002), White Paper: Software-Defined Radio - A Technology Overview.

La sección de RF denominada también Front-End de RF, es la encargada de transmitir/recibir la señal de radiofrecuencia (RF) de la antena por medio del acoplador y así convertir la señal de RF en una señal de Frecuencia Intermedia (IF).

Los bloques ADC/DAC son responsables de realizar la conversión de analógico a digital en la ruta de recepción y de la conversión de digital a analógico en la ruta de transmisión. En cuanto a los bloques DDC/DUC realizan una conversión descendente digital en la ruta de recepción y una conversión ascendente digital en la ruta de transmisión. Por otro lado, los bloques DUC/DDC normalmente realizan operaciones de módem, es decir, modulan la señal en la ruta de transmisión y demodulan la señal en la ruta de recepción.

Finalmente, la sección de banda base realiza las operaciones de establecimiento de conexión, ecualización, salto de frecuencia, recuperación de temporización, correlación e implementa el protocolo de capa de enlace (capa 2) en el modelo OSI.

En cuanto al sistema de Radio Definido por Software (SDR) tanto el procesamiento de banda base como los módulos DDC/DUC se pueden programar. La disponibilidad de antenas inteligentes, Front-End de RF, tecnologías ADC/DAC de banda ancha y la capacidad de procesamiento (MIPS) de DSP y microprocesadores, han impulsado al desarrollo de sistemas que utilizan tecnología SDR. En un sistema SDR, los protocolos de la capa de enlace y las operaciones de modulación/demodulación se implementan en el software. (Wipro Technologies, 2002)

2.5. LTE Hardware ME1130 Essentials of LTE y LTE-Advanced

Para la descripción de los equipos de hardware LTE, el fabricante (DreamCatcher) menciona lo siguiente:

2.5.1. LTE Radio – Access Node (eNodeB) & Core Networks Nodes (EPC)

El hardware ME1130 utiliza un enfoque el cual se basa en software para la implementación de un LTE el cual sea compatible con el estándar para el equipo de usuario (UE), eNodeB, MME, HSS, SGw y PGw, mediante una computadora basada en el sistema operativo Linux, con una previa instalación en una mini PC Intel i7. A su vez este hardware está incorporado con las características que se mencionan a continuación:

- Radio Definido por software (SDR) como Front-End de RF
- Cobertura de espectro continuo de 100 kHz a 3,8 GHz
- Admite operación dúplex completo
- 2x2 MIMO con hasta 60 MHz de ancho de banda

El kit de capacitación ME1130 es un sistema versátil que puede ser configurado para funcionar como un sistema fuera de línea para la simulación o a su vez como un sistema en tiempo real utilizado para transmisión y prueba por aire. Una de sus características principales es que el sistema se puede configurar como una red LTE de manera independiente y se puede usar el equipo de usuario (UE) comercial lista para usar (COTS), como, por ejemplo, el módem o un teléfono inteligente) para acceder a la red. Esto permite una mayor comprensión de las señales y los protocolos en tiempo real de una red LTE.

La implementación basada en software de código abierto de LTE Core (EPC) y Radio-Access Network (eNodeB) está preinstalada en una mini PC. El software compatible con el estándar 3GPP garantiza la interoperabilidad con otros equipos comerciales de código cerrado en cualquier parte de la red. Por ejemplo, el dispositivo LTE COTS estándar puede conectarse al software LTE para una demostración rápida de prueba de concepto de los protocolos LTE en tiempo real.

Figura 19

LTE eNodeB & EPC



Nota. Tomado de *LTE eNodeB & EPC* (p. 2), de (DreamCatcher, 2020), ME1130 Essentials of LTE and LTE-Advanced.

2.5.2. LTE User Equipment (UE)

Para el equipo de usuario de LTE se utiliza un módem USB Huawei 4G LTE de 150 Mbps con SIM de prueba para proporcionar una conexión LTE de extremo a extremo, el cual consta de las siguientes características:

- **LTE FDD:** Cat4 DL: 150Mbps / UL: 50 Mbps @20 M BW
- **UMTS:** DCHSPA+: 42 / 5.76 Mbps; 21 M / 5.76 Mbps; 14 M / 5.76 M
- **HSUPA:** 7.2 M / 5.76 M
- **2G:** EDGE packet data service of up to 236.8 kbps

Figura 20

LTE UE



Nota. Tomado de Huawei 4G Dongle E3372 de (Huawei Device Co., Ltd., s.f.).

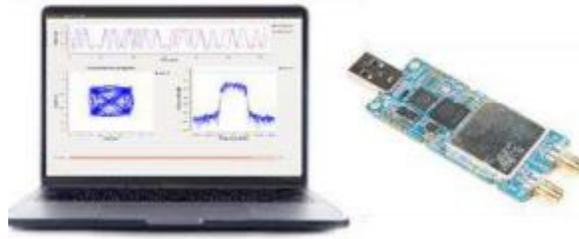
2.5.3. GNU Radio con SDR

El marco de desarrollo de software GNU Radio, permite a los usuarios diseñar, simular e implementar Radio Definidas por Software SDR del mundo real de alta capacidad. Ofrece un

enfoque de diseño gráfico además de admitir el desarrollo en Python y C++. Está configurado con un SDR mini para proporcionar análisis de espectro LTE. (DreamCatcher, 2020)

Figura 21

GNU Radio con SDR



Nota. Tomado de GNU Radio with SDR (p. 2), de (DreamCatcher, 2020), ME1130 Essentials of LTE and LTE-Advanced.

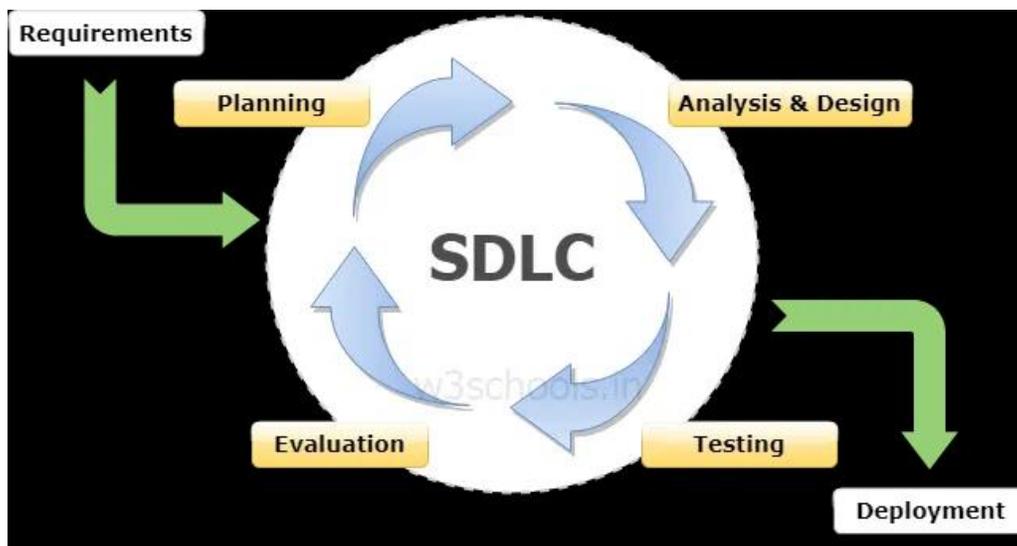
2.6. Metodología Iterativa

El proyecto se basa en la utilización de la “Metodología Iterativa” la cual se enfoca principalmente en el crecimiento y diseño preliminar, para con ello ganar impulso de una manera no acelerada con más complejidad, para cumplir con los requisitos hasta que el software final se determine por completo. Es decir, este modelo es un enfoque para segmentar cualquier proceso de desarrollo de software en partes pequeñas. Por lo tanto, este modelo no tiene como objetivo el establecimiento de un plan de especificación completo, ya que, como alternativa, está diseñado para dar inicio con requisitos mínimos que detallan e implementan solo una parte del software. Para que luego al revisar más a fondo se pueda conocer los requisitos adicionales. El modelo de ciclo de vida iterativo comprende repetir las fases mencionadas a continuación como una secuencia, estas son: fase de requisitos, fase de diseño, fase de implementación, fase de pruebas y fase de revisión. (W3schools® of Technology, 2009)

Este proyecto tiene como finalidad desarrollar un sistema de recepción de señales LTE 4G, mediante el uso de Radio Definido por Software, con el fin de establecer prácticas, en tecnologías LTE y SDR en la materia de SDR, para desarrollar criterios de aplicación de estas tecnologías para su aprendizaje e investigación. En la figura 22 se detallan los conceptos de cada una de las fases de la metodología, junto con los requerimientos que interviene en las mismas.

Figura 22

Representación gráfica del modelo iterativo SDLC



Nota. Tomado de Graphical Representation of the SDLC Iterative Model, de (W3schools® of Technology, 2009), SDLC Iterative Model.

La metodología Iterativa comprende en repetir las fases como una secuencia, a continuación, se detallan cada una de ellas:

- **Metodología Iterativa - Fase de requisitos:** en la fase de requisitos del desarrollo de software, es la que se encarga de recopilar y analizar la información

relacionada con el sistema. Estos requisitos recopilados se planifican en consecuencia para el desarrollo de este. (W3schools® of Technology, 2009)

- **Metodología Iterativa - Fase de diseño:** la segunda fase es la de diseño, donde la solución de software se prepara para satisfacer las necesidades del diseño, el cual puede ser uno nuevo, así como también, la ampliación de uno anterior. (W3schools® of Technology, 2009)
- **Metodología Iterativa - Fase de implementación:** con la planificación y el análisis definidos, se da paso para comenzar el proceso real de implementación y codificación. Las fases de planificación, especificación y diseño se codifican e implementados en esta iteración inicial del proyecto. (Airbrake, 2016)
- **Metodología Iterativa - Fase de pruebas:** una vez que esta iteración de compilación actual ha sido codificada e implementada, la siguiente fase es realizar una serie de procedimientos de prueba para identificar y localizar posibles errores o problemas que se hayan presentado. (Airbrake, 2016)
- **Metodología Iterativa - Fase de revisión:** la fase de revisión es donde se estima y verifica el software según el requisito actual. Luego, los requisitos adicionales se analizan y revisan para proponer una actualización en la próxima iteración. (W3schools® of Technology, 2009)

2.7. Software SDR

Dentro del contexto es importante también analizar el software a usar para el desarrollo del proyecto, por lo tanto, se detalla a GNU Radio y OpenAirInterface. Mediante el software, se realiza la programación y el control de los dispositivos SDR, que se ejecuta ya sea en estaciones de trabajo u ordenadores. En sí, el software se encarga de traducir las opciones y operaciones

seleccionadas y realizadas por el usuario en los comandos con los que se programa el dispositivo y actual sobre el mismo. (Martínez García, 2018)

2.7.1. GNU Radio Companion

GNU Radio Companion es conocida como una herramienta libre y de código abierto con el que se puede desarrollar aplicaciones SDR mediante bloques. Esta herramienta además puede usarse tanto con dispositivos SDR comerciales como sin ellos, como un entorno de simulación. Su entorno de uso puede ser desde enfoques industriales como de investigación, académicos y personales para el desarrollo de sistemas de comunicación inalámbricas y sistemas de radiofrecuencia.

GNU Radio es libre y de código abierto, es por esta razón que el usuario puede extender y modificar sus funcionalidades. (GNU Radio, 2023)

En GNU Radio, el usuario podría realizar la implementación de aplicaciones mediante bloques, interconectados entre sí, los cuales puede estar ya establecidos o ser definidos por el usuario. Dichos bloques permiten que GNU Radio realice el procesamiento de las señales de RF. Debido a que existen numerosos bloques ya implementados, es por esta razón que GNU Radio puede usarse para implementar aplicaciones SDR, como transmisores y receptores de Amplitude Modulation – Modulación de Amplitud (AM), Frequency Modulation – Modulación en Frecuencia (FM), analizadores de espectro, sistemas de difusión digital, entre otras. GNU Radio utiliza los lenguajes de programación C++ y Python para la programación de los bloques. Y para el control de los dispositivos SDR de igual manera se realiza mediante bloques de aplicación. (GNU Radio, 2023)

2.7.2. OpenAirInterface

OpenAirInterface (OAI), es una plataforma de código abierto que implementa las especificaciones de LTE del 3GPP, mantenida por la OpenAirInterface Software (OSA). Proporciona una implementación de la red de acceso radio (E-UTRAN) y una implementación del núcleo de red (EPC) en dos módulos de software distintos.

openairinterface5g: encargado de la red de acceso radio (E-UTRAN). Es decir, proporciona software para el despliegue de un eNodeB y la emulación de un equipo de usuario (UE).

openair-cn: encargado del núcleo de red (EPC). Proporciona software para el HSS, el MME y el P-GW. (OpenAirInterface, 2023)

OAI permite desplegar un escenario completamente modularizado para prototipos de red 4G-LTE. Esto debido a que es un código abierto y libre, además, posee una comunidad extensa de desarrolladores, por lo que la plataforma se puede mantener actualizada en cuanto a características y correcciones de errores. Cabe resaltar, que OAI sirve como base para el desarrollo futuro de prototipos de red 5G, con la inclusión de nuevas características y la colaboración en el desarrollo de redes móviles a futuro. (Martínez García, 2018)

3. Capítulo III: Desarrollo De La Propuesta (Diseño)

En este capítulo se presenta el desarrollo de la propuesta y diseño en base al análisis de la situación actual al añadir la tabulación de los resultados en base a la encuesta aplicada tanto a algunos docentes de la carrera de la Ingeniería en Telecomunicaciones (CITEL) y en especial al docente de la materia de Radio Definido por Software (SDR), el Ing. Fabián Cuzme y al Técnico Docente de Laboratorio, el Ing. Henry Farinango, esto se presenta en la ubicación del anexo 2 y detalla como las mismas ayudan a resolver las preguntas planteadas para el sistema con el uso de SDR como interfaz de RF del LTE para el fortalecimiento de conocimientos. A continuación, se realiza una descripción del sistema donde se muestra a sus beneficiarios y posibles limitaciones, además se presenta el análisis de requerimientos de stakeholders, arquitectura y funciones.

3.1. Análisis De La Situación Actual

En el análisis de la situación actual, es el cual en el que se busca lograr la recopilación de información con el objetivo clave de responder a las preguntas planteadas en base al diseño del sistema, para que de esta manera se pueda contribuir al desarrollo del proyecto, dicha información se obtiene mediante la encuesta aplicada tanto al docente de la asignatura de SDR, así como también al Técnico Docente de la carrera de CITEL, esto para que en base a sus conocimientos se llegue a determinar los requerimientos de usuarios.

Sin embargo, en el enfoque de la investigación y según lo expresa el autor (Tafur Arenazas, 2017), se establece que: mediante una tecnología más avanzada como las redes de cuarta generación LTE 4G, la misma que favorecerá en gran manera el desarrollo tanto comercial, como educativo, asegurando a los usuarios soporte necesario para la movilidad y compatibilidad entre los sistemas, (conectividad con otras redes), con lo cual se podrá utilizar el servicio de banda ancha en cualquier momento y lugar.

En ese contexto y según lo expresa el autor (Escobar Chamba, 2021), se establece que: enfocado en nuestro país Ecuador y a nivel internacional, existen estudios en universidades como la Universidad de las Fuerzas Armadas (Ecuador) y la Universidad Santo Tomás (Colombia) donde se ha evidenciado investigaciones sobre diseño de una estación base con la finalidad de cubrir las localidades con una aceptable calidad de servicio, así como también proveer de cobertura mediante la red UMTS y LTE. Evidenciando que en el país existe una tendencia hacia la investigación y aplicación de estas tecnologías ya sea en ambientes de producción y de manera especial en ambientes de investigación académico. (Chica Pedraza et al., 2020)

Por lo tanto, es importante recalcar que la propuesta de este proyecto es brindar a los estudiantes de la Universidad Técnica del Norte un diseño de un sistema donde se pueda formar y capacitar a las nuevas generaciones de ingenieros en telecomunicaciones en áreas de tecnología LTE Y SDR para el fomento de aprendizaje e investigación.

3.1.1. Análisis De Los resultados

Una vez aplicada la encuesta mencionada anteriormente, se logra obtener resultados que permitan al desarrollo del presente proyecto. A continuación, se indica el análisis obtenido de la tabulación de las encuestas como se establece en el anexo 2.

Estas preguntas realizadas ayudan a la obtención de los requerimientos del usuario para implementar en el sistema, logrando así de esta manera, definir una serie de estrategias con respecto a las necesidades de los beneficiarios. En relación, con que haría falta más herramientas, prototipos, aplicaciones educativas y didácticas para la materia de (SDR) enfocadas en prácticas de laboratorio se determina una aceptación ya que esto a su vez tendría la disponibilidad de un funcionamiento continuo para su desarrollo. Dicho sistema y con el uso de SDR como interfaz de RF (Radio Frecuencia) de LTE, se obtiene una respuesta afirmativa debido a que será útil para el

fortalecimiento de conocimientos y la investigación como características fundamentales de LTE y la arquitectura general del sistema.

En relación al tema del hardware se considera afirmativo que se pueda dar uso tanto para simulación o como implementación en tiempo real y se logre conocer como configurar y conectar un equipo de usuario a la red LTE. Reforzando los conocimientos enfocados en temas de la asignatura, como en el análisis de señales en dominios de tiempo y frecuencia

Por otra parte, en cuanto al tema del uso del sistema en conjunto con herramientas de software libre como por ejemplo GNU Radio u OpenAirInterface con SDR, se considera que, si es acertado desarrollar y también comprender el procesamiento de señales con el fin de proporcionar capacidades de análisis de espectro LTE, esto en un ámbito didáctico, de fácil entendimiento y brinde al usuario una comprensión práctica en la tecnología LTE.

Finalmente, se obtiene una aceptación total en cuanto al nivel de aprendizaje que proporcionaría el sistema, ya que los usuarios podrán comprender de mejor manera los protocolos y a su vez conozcan como se construye una red LTE de manera independiente, esto en conjunto con que los usuarios puedan configurar y conectar un Equipo de Usuario a la red LTE obteniendo información sobre el procesamiento de señales.

3.2. Requerimientos Del proyecto

Mediante el análisis del proyecto en conjunto con la información obtenida para el sistema se logra establecer los elementos necesarios y que se cumplan con las necesidades planteadas. Por tal razón, se pone a evaluación a los requerimientos implicados tanto de usuario, como los requerimientos de sistema y además los requerimientos de arquitectura.

3.2.1. Construcción De Atributos De Los Requerimientos

Para la construcción de los atributos se realiza en base a tres requerimientos establecidos: stakeholders, sistema y arquitectura, por lo tanto, en cada requerimiento se debe cumplir para que sea verificable, para que de esta forma se pueda efectuar, medir y limitar el propósito del proyecto.

3.2.1.1. Nomenclatura Para Uso De Los Requerimientos

Para que cada uno de los requerimientos se efectúen, es fundamental definirlos usando una abreviatura la cual pueda permitir un manejo adecuado de los datos. En la tabla 2 se detalla las abreviaturas que van a ser utilizadas.

Tabla 2

Abreviatura de Requerimientos

Abreviaturas	Requerimientos
SRSH	Hardware y Software
StSR	Stakeholders
SySR	Sistema

Fuente: elaborado por el autor

3.2.1.2. Requerimientos De Stakeholders

El uso de estos requerimientos tiene por finalidad establecer los requisitos del sistema en base a las necesidades del usuario en temas de LTE y Radio Definido por Software (SDR). Para lograr cumplir las expectativas tanto del docente y técnico docente de la carrera CITEL, es decir se debe asegurar la relación entre lo que esperan también los estudiantes. Por lo tanto, en la tabla 3 se examinarán los requerimientos tanto operacionales como también de usuarios.

Tabla 3

Requerimientos de Stakeholders

StSR				
#	REQUERIMIENTO	PRIORIDAD		
		<i>ALTA</i>	<i>MEDIA</i>	<i>BAJA</i>
REQUERIMIENTOS OPERACIONALES				
StSR1	El sistema debe tener una capacidad de funcionamiento continuo durante las prácticas de laboratorio de la asignatura de SDR	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
StSR2	El sistema debe ayudar a describir las principales características de LTE y la arquitectura general del sistema de la red de acceso de radio y la red central	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
StSR3	El sistema ayudará a analizar los protocolos LTE y a configurar una red LTE experimental de extremo a extremo	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
StSR4	El sistema debe proporcionar un analizador de señales OFDM/SC-FDMA en dominios de tiempo, frecuencia, medirá y analizará señales LTE	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
StSR5	El sistema junto con GNU Radio o OpenAirInterface con SDR puede ser configurada para ser un analizador de señales para proporcionar capacidades de análisis de espectro LTE	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
REQUERIMIENTOS DE USUARIO				
StSR6	El sistema brindará al usuario una comprensión profunda y experiencia práctica en la tecnología LTE	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
StSR7	La implementación basada en la PC de eNodeB y EPC de LTE es una	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

	plataforma eficaz para que los usuarios exploren los protocolos LTE y a su vez aprendan a construir una red LTE de manera independiente			
StSR8	Los usuarios aprenderán cómo configurar y conectar un UE estándar comercial como un teléfono inteligente a la red LTE y obtendrán más información sobre el procesamiento de señales en tiempo real y los protocolos de LTE	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
StSR9	El usuario no deberá manipular ningún componente del eNodeB y EPC de LTE	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
StSR10	El usuario no debe modificar ningún parámetro de la configuración de eNodeB y EPC	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Nota. Detalle de requerimientos operacionales y de usuario con el valor de prioridad alta, media y baja. Fuente: Autoría

Estos requerimientos han sido definidos mediante la encuesta realizada en este caso al Docente y Técnico Docente de la asignatura de Radio Definido por Software de la carrera de CITEL, además, se toma en cuenta la experiencia para la ejecución de este tipo de proyecto. Dando como resultado una relación acorde entre los usuarios y el sistema, estableciendo un concepto general del producto final.

3.2.1.3. Requerimientos del Sistema

Los requerimientos del sistema se establecen en base a las funciones y limitantes de desempeño del sistema de recepción de señales LTE usando Radio Definido por Software. Por lo tanto, es necesario analizar los requerimientos de interfaz, uso, performance, modo estado y físicos, los cuales tengan relación con los requerimientos de Stakeholders. En la tabla 4 se detalla

la información de los requerimientos del sistema, mencionando también la funcionalidad del mismo en conjunto con la prioridad del requerimiento, por lo que a su vez están relacionados con los requerimientos de Stakeholders.

Tabla 4

Requerimientos del Sistema

SySR				
#	REQUERIMIENTO	PRIORIDAD		
		<i>ALTA</i>	<i>MEDIA</i>	<i>BAJA</i>
REQUERIMIENTOS DE INTERFAZ				
SySR1	El sistema debe tener interacción entre los módulos del sistema	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SySR2	La interfaz debe tener interconexión entre los módulos del sistema implementada por GNU Radio u OpenAirInterface	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SySR3	Conexión de un Equipo de Usuario (UE) estándar comercial como un teléfono inteligente o dongle a la red LTE	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SySR4	El mini PC debe tener interacción con el monitor para la visualización de información de configuraciones	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
REQUERIMIENTOS USO				
SySR5	El analizador de señales en dominios de tiempo y frecuencia, junto con el analizador de protocolos LTE deben ser obtenidos de manera adecuada por el sistema	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
SySR6	El sistema permitirá monitorizar en tiempo real parámetros de nivel físico en la interfaz radio	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
SySR7	El sistema debe tener una disponibilidad de forma continua	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

	en las prácticas de laboratorio de la asignatura SDR			
SySR8	Fuente de alimentación de entrada 100 - 240 V y salida de 19 V; 11.8 A; 230 W (Mini PC Intel NUC)	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SySR9	Los módulos del sistema deben ubicarse en un lugar adecuado dentro del laboratorio de la carrera de CITEL	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

REQUERIMIENTOS DE PERFORMANCE

SySR10	El sistema debe tener una emulación adecuada de red LTE en configuración y despliegue	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
SySR11	Se debe asegurar que el sistema realice el análisis de espectro, protocolo y rendimiento de la red LTE utilizando GNU Radio y SDR	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SySR12	El sistema no debe saturarse y debe tener un funcionamiento continuo durante las prácticas de laboratorio	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
SySR13	El sistema debe proporcionar una visualización gráfica del espectro de la red LTE	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SySR14	Facilidad de administración de los recursos de la red LTE (gestionar y controlar eficientemente los recursos de la red LTE, incluyendo el espectro de frecuencias, la capacidad de la red y el ancho de banda)	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

REQUERIMIENTOS DE MODO Y ESTADO

SySR15	El sistema deberá poder configurarse con diferentes parámetros con el objetivo de evaluar distintos casos de uso y pruebas	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
--------	--	--------------------------	-------------------------------------	--------------------------

SySR16	El sistema debe mantener su operatividad durante todo el tiempo en el desarrollo de las prácticas de laboratorio	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SySR17	El sistema indicará mediante la configuración e implementación de eNodeB, EPC y UE la emulación de red LTE	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SySR18	El sistema mostrará errores en el terminal cuando existan eventos imprevistos en los módulos de interacción del sistema (En el sistema operativo Linux)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

REQUERIMIENTOS FÍSICOS

SySR19	El sistema debe ser de un adecuado entendimiento en cuanto a la implementación en base a la conexión del UE a la red LTE	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
SySR20	El sistema debe estar implementado en un ambiente idóneo como en los laboratorios de la carrera de CITEL	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SySR21	Los equipos deben montarse en un lugar adecuado que permita su fácil instalación, movilidad y que permita proteger de daños en los componentes	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SySR22	El sistema debe estar correctamente conectado a su atenuador de RF, cable RF SMA y adaptador de corriente	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Nota. Detalle de requerimientos de interfaz, uso, performance, de modo/estado y físicos con el valor de prioridad alta, media y baja. Fuente: Autoría

3.2.1.4. Requerimientos De Arquitectura

En cuanto a los requerimientos de arquitectura, se basan en los componentes tanto de hardware y de software, así como las características basándose en el funcionamiento que deberá

tener el sistema. Por lo tanto, se considera como parte importante a los requerimientos lógicos, diseño, hardware, software y eléctricos. En la tabla 5 mostrada a continuación se detallan los requerimientos de arquitectura.

Tabla 5

Requerimientos de Arquitectura

SRSH				
#	REQUERIMIENTO	PRIORIDAD		
		ALTA	MEDIA	BAJA
REQUERIMIENTOS DE DISEÑO				
SRSH1	El sistema requiere de una implementación de LTE mediante una PC, con un sistema operativo Ubuntu 16.04 LTS	☒	☐	☐
SRSH2	El sistema debe tener en su implementación basada en PC de eNodeB y EPC de LTE	☒	☐	☐
SRSH3	El sistema debe tener un UE estándar comercial, configurado y conectado a la red LTE	☐	☒	☐
SRSH4	El sistema debe ser accesible para los usuarios en las prácticas de laboratorio	☐	☒	☐
SRSH5	Sistema de bajo costo energético	☐	☒	☐
REQUERIMIENTOS LÓGICOS				
SRSH6	El sistema requiere de Radio Definido por Software para LTE con un rango de frecuencia de 100 kHz – 3.8 GHz.	☒	☐	☐
SRSH7	El sistema requiere de Radio Definido por Software para LTE con un ancho de banda 61.44 MHz.	☒	☐	☐
SRSH8	El sistema requiere de Radio Definido por Software para LTE con una profundidad de muestreo	☒	☐	☐

	de 12 bit y tasa de muestreo de 61.44 MSPS.			
SRSH9	El sistema requiere de Radio Definido por Software como analizador de LTE con un rango de frecuencia de 10 MHz – 3.5 GHz	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SRSH10	El sistema requiere de Radio Definido por Software como analizador de LTE con un ancho de banda de 30.72 MHz.	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
REQUERIMIENTOS DE HARDWARE				
SRSH11	El sistema requiere de una PC con un procesador Intel Core i7, memoria de 8GB y disco duro de 120 GB SSD	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SRSH12	El sistema requiere de Radio Definido por Software para LTE con canales de transmisión y recepción de 2x2.	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SRSH13	El sistema requiere de Radio Definido por Software para LTE con transmisión full dúplex.	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
SRSH14	El sistema requiere de Radio Definido por Software como analizador de LTE con una profundidad de muestreo de 12 bit y tasa de muestreo de 30.72 MSPS.	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SRSH15	El sistema requiere de Radio Definido por Software como analizador de con canales de transmisión y recepción de 1x1.	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
SRSH16	El sistema requiere de un equipo de usuario LTE	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SRSH17	El sistema requiere de Radio Definido por Software como analizador de LTE con transmisión full dúplex	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
REQUERIMIENTOS DE SOFTWARE				

SRSH18	Plataforma de desarrollo de sistemas de comunicación inalámbrica de código abierto (GNU Radio u OpenAirInterface)	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SRSH19	Soporte con software de simulación y emulación de redes inalámbricas	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SRSH20	El sistema requiere de una implementación de LTE compatible con el estándar para el equipo de usuario (UE), eNodeB, MME, HSS, SGw y PGw, mediante una computadora en el sistema operativo Linux Ubuntu 16.04 LTS	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SRSH21	El sistema requiere de Radio Definido por software (SDR) como Front-End de RF (Radio Frecuencia)	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SRSH22	El sistema requiere de una implementación basada en software de código abierto como GNU Radio u OpenAirInterface de LTE Core (EPC) y Radio-Access Network (eNodeB) preinstalada en la PC	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

REQUERIMIENTOS ELÉCTRICOS

SRSH23	Atenuador RF (Radio Frequency - Frecuencia de radio)	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SRSH24	Cable RF SMA (Sub-Miniatura A)	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SRSH25	Cable RF SMA (Sub-Miniatura A) a CRC9 (conector)	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SRSH26	Cable USB	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SRSH27	Adaptador de corriente	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Nota. Detalle de requerimientos de diseño, lógicos, hardware, software y eléctricos con el valor de prioridad alta, media y baja. Fuente: Autoría

Los requerimientos de arquitectura al ser analizados establecen una relación en conjunto con las necesidades planteadas tanto por los usuarios como con las del sistema, es decir, se lograría establecer un sistema óptimo de hardware y software libre.

3.3. Selección De Hardware y De Software

Para la selección de hardware y software, se basa en la relación de la metodología que permita realizar el proceso de selección de manera adecuada y con una comparativa técnica de los elementos, es decir para la elección del hardware se hace uso de los datasheets u hojas de datos de los elementos y para el caso de la elección del software se considera los requerimientos establecidos tanto de stakeholders, sistema y arquitectura, previo un análisis comparativo de cada para así poder obtener el que mejor se adapte al implementar en el proyecto.

3.3.1. Selección De Hardware

La selección de hardware se realiza a partir del establecimiento de parámetros a considerar, los mismos que se enfocan específicamente en el fundamento teórico en la sección 2.2 del capítulo 2, que están enfocados en la arquitectura de la tecnología LTE, estos tipos de equipos fueron seleccionados ya que se basan en la disponibilidad y precio de los mismos en el mercado internacional, además, que los mismos en conjunto se pueden considerar como un paquete para enseñar para una comprensión profunda y experiencia práctica en tecnología LTE.

Para la selección, se realiza en base a una tabla comparativa en conjunto con las especificaciones de cada dispositivo a usar el mismo que debe cumplir con los requerimientos de arquitectura, por lo tanto, se asignará un valor referencial para detallar si cumple o no cumple con el requerimiento. En la tabla 6 se observa cómo se define el valor de “1” para si cumple y “0” para no cumple.

Tabla 6

Valor establecido para requerimientos

Descripción	Valor referencial
Si cumple	1
No cumple	0

Fuente: Autoría

3.3.1.1. Radio Definido Por Software Para LTE

En el mercado internacional existen dispositivos SDR enfocados para LTE, como el Lime SDR de la compañía Lime Microsystems, que es una plataforma de hardware útil para diseños digitales y de RF basados en alto rendimiento, orientado al despliegue de tecnologías inalámbricas, como redes de comunicaciones móviles (GSM, UMTS, LTE), así como también para redes de sensores del Internet de las cosas (IoT), o radio y televisión en difusión digital. Esta selección se realiza mediante una tabla comparativa de las especificaciones de cada dispositivo a usar verificando que cumpla con los requerimientos, por lo tanto, en la tabla 7 se detallan algunos equipos considerados para la selección.

Tabla 7

Selección de dispositivos SDR para LTE

Hardware	Requerimientos										Puntaje	
	S1SR2	S1SR3	S1SR4	S1SR6	S1SR7	SySR10	SySR11	SySR17	SRSH6	SRSH7	SRSH8	
Lime SDR	1	1	1	1	1	0	1	1	1	1	1	10

Fuente: Autoría

En la Tabla 8 se muestra las principales características técnicas del LimeSDR.

Tabla 8

Características técnicas de Lime SDR

Lime SDR	
Rango de Frecuencia	100 kHz - 3,8 GHz
Ancho de banda	61,44 MHz
Profundidad / velocidad de muestreo	12 bits / 61,44 MSPs
Canales TX y RX	2 x 2
Dúplex	Full
Puertas Lógicas Programables	40k
Código abierto	Full
Interfaz	USB 3.0

Fuente: adaptado de (MyriadRF, 2023). Lime SDR-USB

Elección: Para la elección del dispositivo de Radio Definido por Software para LTE, se ha seleccionado el LimeSDR de la compañía Lime Microsystems, debido a que es una plataforma de hardware útil para diseños digitales y de RF basados en alto rendimiento, como en

este caso orientado al despliegue de tecnologías inalámbricas, como redes de comunicaciones móviles LTE. Sumado a esto, LimeSDR es open source (código abierto) y se posiciona dentro de los dispositivos con precio denominado medio y superior, con un amplio rango de frecuencias y ancho de banda elevado para su desarrollo.

3.3.1.2. Radio Definido Por Software Como Analizador LTE

Para la selección del Radio Definido por Software como analizador LTE, se considera a otro dispositivo de Lime Microsystems, como lo es el LimeSDR Mini el cual es una versión más pequeña, que de igual manera se puede usar para el despliegue del mismo tipo de aplicaciones, con la diferencia respecto a sus características, dicha selección debe cumplir con los requerimientos, por lo tanto, en la tabla 9 se detallan algunos dispositivos considerados para la selección.

Tabla 9

Selección de dispositivos SDR como analizador LTE

Hardware	Requerimientos								Puntaje
	S _t SR5	S _y SR5	S _y SR6	S _y SR13	SRSH9	SRSH10	SRSH14	SRSH15	
Lime SDR Mini	1	1	1	0	1	1	1	1	7
HackRF One	1	1	1	0	0(por el costo)	1	1	1	6

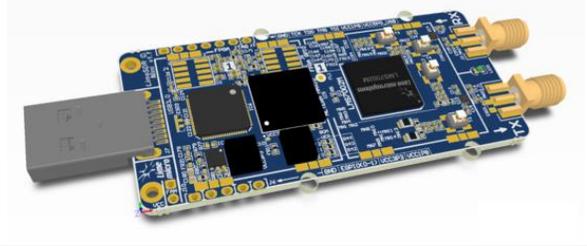
Fuente: Autoría

En la Tabla 10 se muestra las principales características técnicas del LimeSDR.Mini

Tabla 10

Características técnicas de Lime SDR Mini

Lime SDR Mini	
Rango de Frecuencia	10 MHz – 3.5 GHz
Ancho de banda	30.72 MHz
Profundidad / velocidad de muestreo	12-bits / 30.72MSPs
Canales TX y RX	1 x 1
Dúplex	Full
Puertas Lógicas Programables	16k
Código abierto	Full
Interfaz	USB 3.0



Fuente: adaptado de (MyriadRF, 2023). Lime SDR-Mini

Elección: Para la elección del dispositivo de Radio Definido por Software como analizador de LTE, se ha seleccionado el LimeSDR Mini considerado de tamaño más pequeño, corresponde a la misma compañía Lime Microsystems y al igual que el LimeSDR, los dos pueden ser utilizados para el despliegue de el mismo tipo de aplicaciones o despliegue de tecnologías inalámbricas, como LTE en redes de comunicaciones móviles, con la diferencia que poseen respecto a sus características. De igual manera, el LimeSDR Mini es de código abierto con un rango de frecuencias y ancho de banda adecuados para su desarrollo.

3.3.1.3. LTE UE-Huawei E3372

Para la selección del UE y poder tener acceso a la red LTE por la interfaz radio, se utilizan equipos de usuario (UE). En sí, es un dispositivo que el usuario final utiliza para la comunicación. Generalmente, este terminal puede ser un dispositivo portátil como un teléfono inteligente o una tarjeta de datos, como las que se usan en 2G y 3G, o a su vez podría estar integrado por ejemplo a una computadora portátil. En este caso se analizará en base a los módems con SIM de prueba para proporcionar una conexión LTE de extremo a extremo, dicha selección debe cumplir con los requerimientos, por lo tanto, en la tabla 11 se detallan algunos dispositivos considerados para la selección

Tabla 11

Selección de dispositivos módem para LTE UE

Hardware	Requerimientos							Puntaje
	S _t SR8	S _y SR3	S _y SR15	S _y SR19	SRSH3	SRSH16	SRSH20	
Huawei E3372	1	1	1	1	1	1	1	7
ZTE MF823	1	1	0	1	1	1	1	6

Fuente: Autoría

En la Tabla 12 se muestra las principales características técnicas del módem Huawei E3372

Tabla 12

Características técnicas de módem para LTE UE Huawei E3372

Huawei E3372



LTE FDD	Cat4
Downlink (DL)	150 Mbps
Uplink (UL)	50 Mbps
UMTS DCHSPA+	42 / 5.76 Mbps
	21 M / 5.76 Mbps
	14 M / 5.76 M
HSUPA	7.2 M / 5.76 M

Fuente: Adaptado de (DreamCatcher, 2020). LTE UE. ME1130 Essentials of LTE and LTE-Advanced.

Elección: Para la elección del UE (Equipo de Usuario) y así acceder a la red LTE por la interfaz radio, se ha seleccionado el módem USB Huawei E3372, el cual es un dispositivo que el usuario final utilizará para la comunicación. Este dispositivo tiene en sus características 150 Mbps con SIM de prueba como equipo de usuario, ideal para proporcionar una conexión LTE de extremo a extremo.

3.3.2. Selección De Software

En cuanto a la selección del software, que servirá para el desarrollo del proyecto se debe considerar tanto los requerimientos de software, arquitectura y a su vez deben tener relación con los requerimientos de hardware especificados con anterioridad. Esto debido a que tanto la

Fuente: Autoría

En la Tabla 14 se muestra las principales características técnicas de la aplicación para análisis del espectro de RF para dispositivos SDR, OpenAirInterface

Tabla 14

Características técnicas de OpenAirInterface

OpenAirInterface	
Lenguaje	C / C++
Dificultad de instalatación	Alta
Legibilidad de código	Buena
Implementación de UE	Si
Implementación de eNodeB	Si
Implementación de EPC	Si
Mediciones	Si
Licencia	OAI 5G / Apache V2.0
Soporte de Hardware	USRP / LimeSDR / ExpressMIMO2

Fuente: Adaptado de (Gutierrez et al., 2020). Proyectos de código abierto para aplicaciones LTE con SDR. Measurement of 4G LTE Cells with SDR Technology

Elección: Para la elección de la aplicación para el análisis del espectro de RF para dispositivos SDR, se ha seleccionado a OpenAirInterface ya que es una plataforma de Código abierto y a su vez implementa las especificaciones LTE del 3GPP, de igual manera está enfocada para la realización de tareas específicas como la creación de redes de comunicaciones móviles. Proporcionando una implementación de la red de acceso radio (E-UTRAN) y una implementación del núcleo de red (EPC). Y a su vez permite la conexión de UE comerciales,

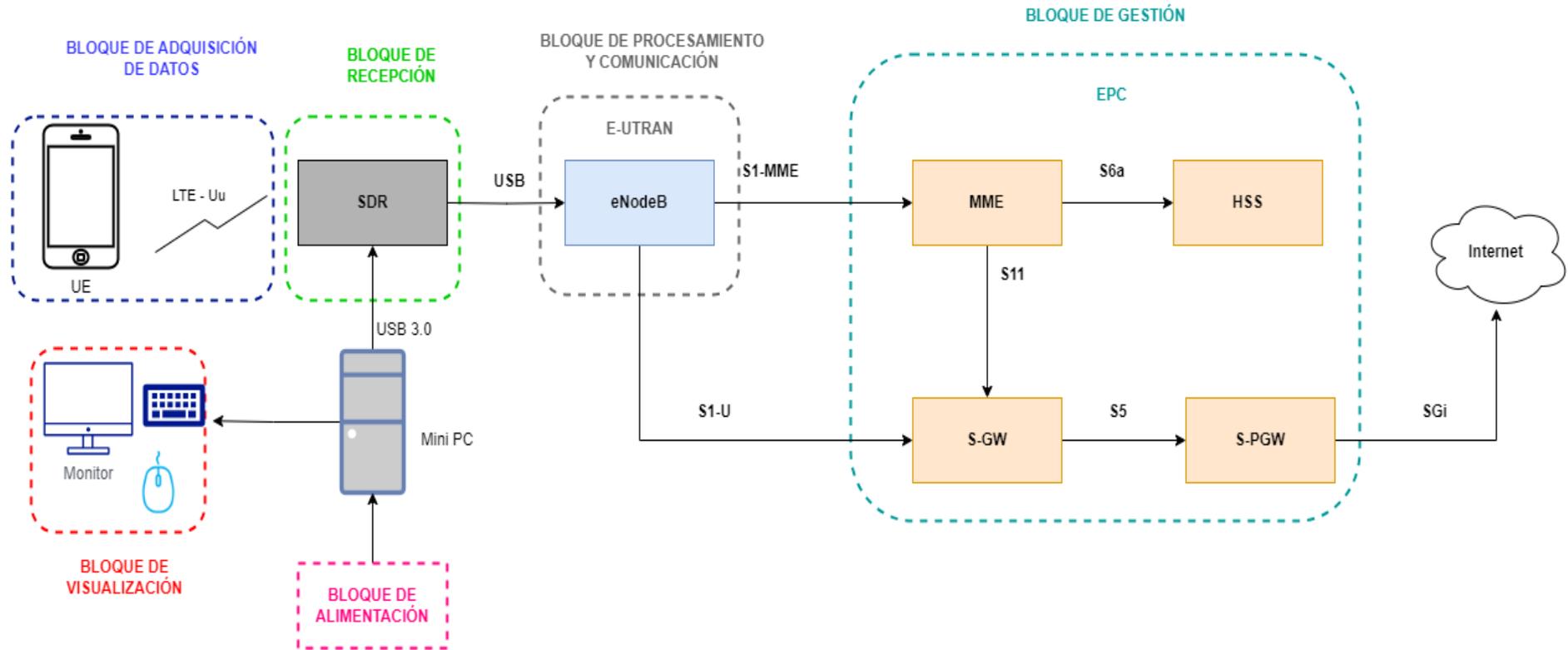
proporcionando su propia solución para la emulación de terminales de usuario, lo que lo hace adecuado para el desarrollo.

3.4. Definición De Arquitectura

La arquitectura se ha dividido en 6 bloques elementales para el funcionamiento del sistema, por lo que a continuación, se describe cada uno de estos bloques, dando a conocer tanto su diagrama de flujo como su diagrama de conexión.

Figura 23

Arquitectura general del sistema



Fuente: Elaborado por el autor

3.4.1. Bloque De Adquisición De Datos

En este bloque se encuentra el UE o Equipo de Usuario, la función del UE conectado al eNodeB por la interfaz radio, implica tanto la sincronización como la búsqueda y selección de la mejor celda disponible, el proceso de autenticación y el establecimiento de la conexión con el eNodeB correspondiente, así como también la recepción y decodificación de señales de control y la recepción y procesamiento de datos transmitidos por el eNodeB.

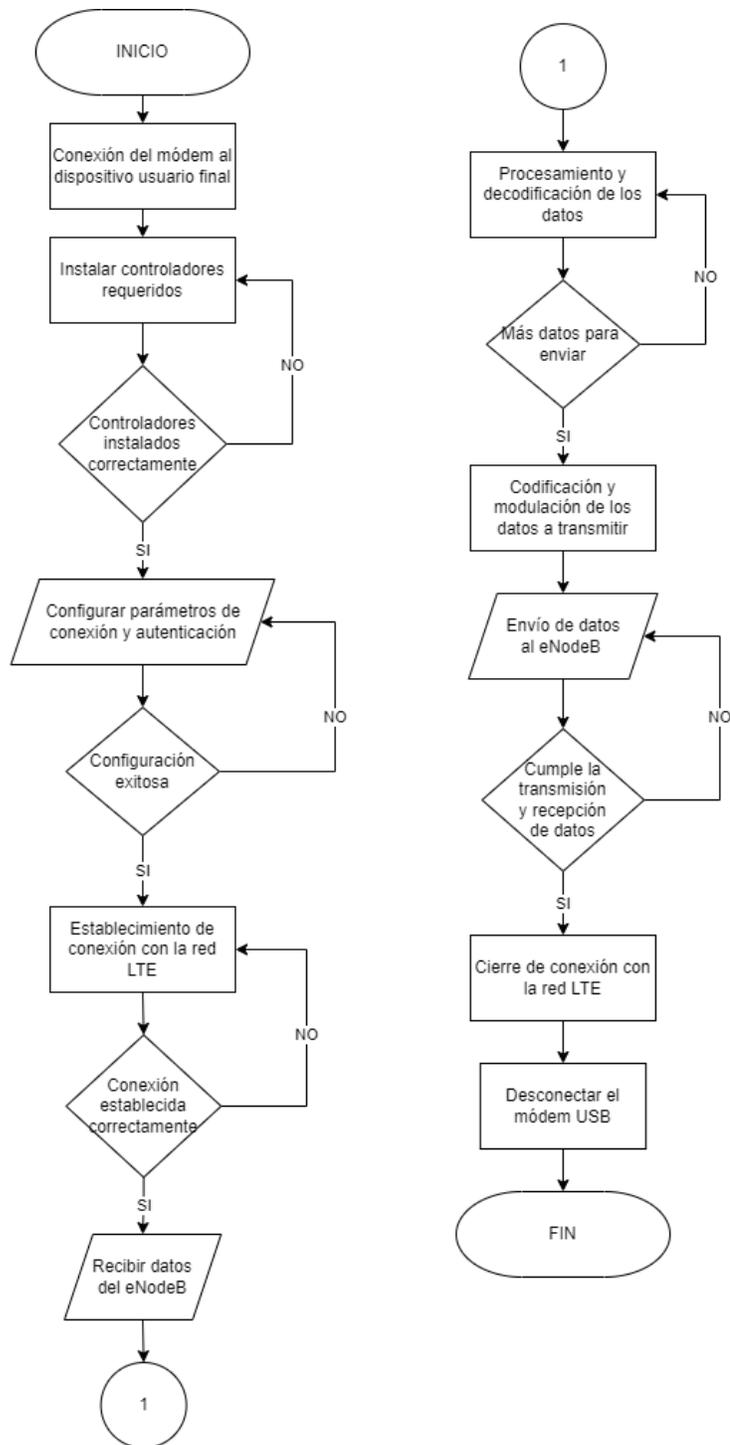
Para el bloque de adquisición de datos el cual consta del dispositivo de usuario final (computadora) y el módem USB que es el UE para el establecimiento de la conexión con la red LTE, el mismo que requiere de controladores o denominados software o firmware para garantizar la comunicación entre el usuario final y el módem USB.

3.4.1.1. Diagrama de Flujo Del Bloque De Adquisición De Datos

El diagrama de flujo muestra el ciclo de funcionamiento que se debe seguir para el correcto funcionamiento. Donde el primer paso es conectar el módem USB al dispositivo, instalar los controladores requeridos y así poder establecer la conexión con la red LTE. En la figura 24 se muestra el diagrama de flujo para el bloque de adquisición de datos.

Figura 24

Diagrama de flujo del bloque de adquisición de datos



Fuente: elaborado por el autor

Para el diseño del bloque de adquisición de datos, está conformado por dispositivo de usuario utilizado por el usuario final (computadora) y el equipo de usuario, el módulo Huawei

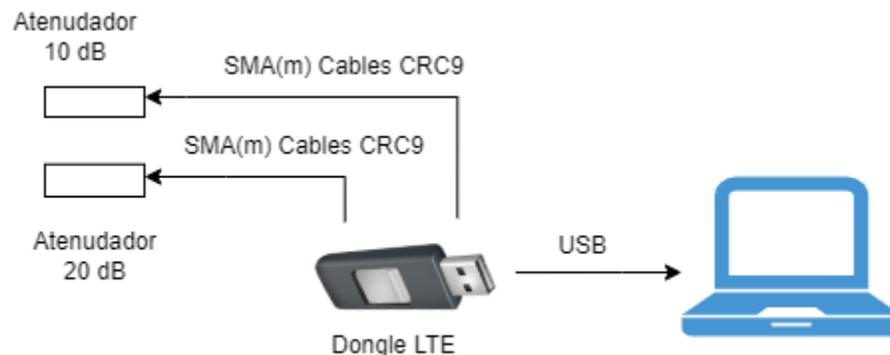
LTE 4G. El cual actúa como UE para el establecimiento de la conexión con la red LTE y adquirir los datos, de igual manera los controladores son parte fundamental para la comunicación entre el módem USB y el dispositivo de usuario.

3.4.1.2. Diagrama De Conexión Del Bloque De Adquisición de Datos

El dispositivo de usuario se relaciona con el UE como en este caso el dongle USB LTE o módem Huawei para la adquisición de datos, el módem a su vez se conecta al dispositivo de usuario mediante el puerto USB con el fin de proporcionar la capacidad de conectividad de red LTE. En la figura 25 se muestra la distribución y la conexión.

Figura 25

Diagrama de conexión del UE



Fuente: elaborado por el autor

3.4.2. Bloque De Recepción Y Transmisión

Este bloque se interconecta con el dongle LTE y se utiliza un dispositivo SDR de la compañía Lime Microsystems, enfocado al despliegue de tecnologías inalámbricas como en este caso LTE, el mismo que se conectará mediante USB al local host eNodeB y servirá tanto para la transmisión como para la recepción de señales LTE, esto debido al rango de frecuencias de

operación adecuadas, a su vez también por las distintas bandas de frecuencia LTE, por su ancho de banda instantáneo, siendo idóneo para el despliegue y a su vez por la compatibilidad con OpenAirInterface.

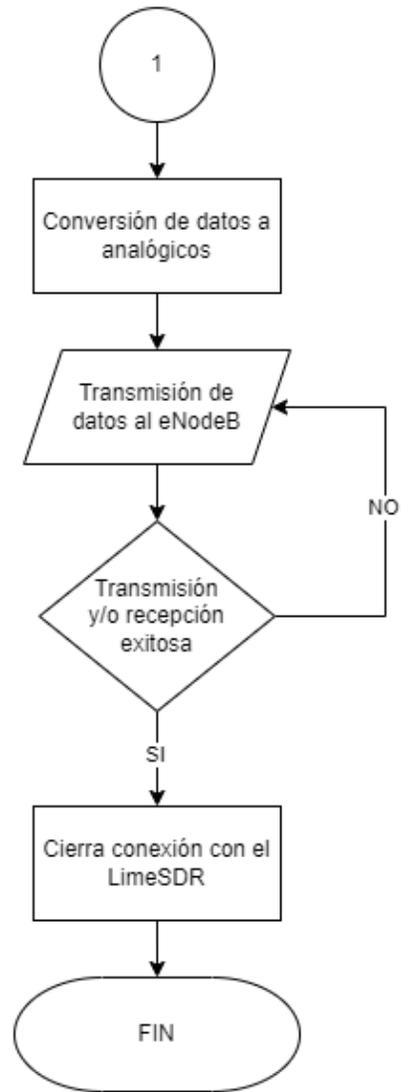
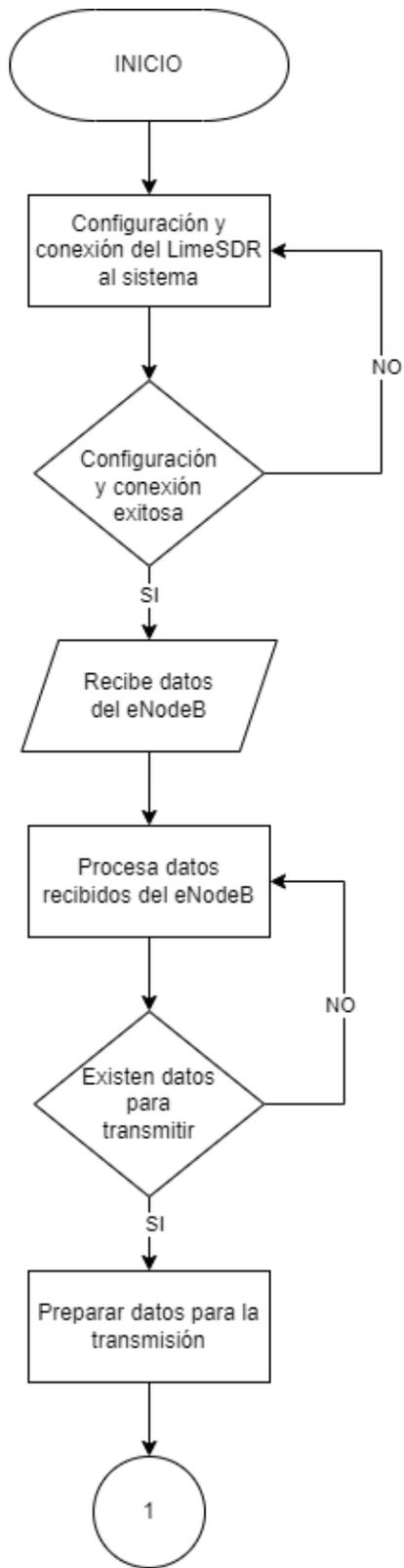
Este bloque se encarga de la recepción y procesamiento de las señales LTE, el LimeSDR se encarga de sintonizar y capturar las señales de radio LTE que son transmitidas por el eNodeB, a su vez estas señales se convierten de analógicas a digitales con el fin de extraer información como los datos del usuario, el control de la conexión, sincronización de tiempo. Para posterior tomar las señales analógicas y transmitir las en la banda de frecuencia LTE.

3.4.2.1. Diagrama de Flujo Del Bloque De Recepción Y Transmisión

El diagrama de flujo propuesto para el bloque de recepción y transmisión de señales, como se puede observar en la figura 26 se muestra el proceso en el cual se inicia con la configuración del LimeSDR, seguido de la recepción y procesamiento de señales LTE para preparar los datos para la transmisión, realizar la conversión a formato analógico y poder transmitir las señales LTE.

Figura 26

Diagrama de flujo del bloque de recepción y transmisión



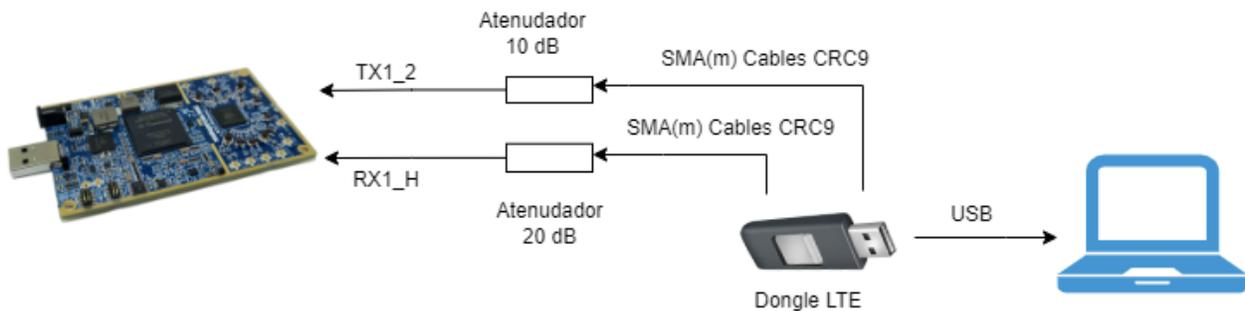
Fuente: elaborado por el autor

3.4.2.2. Diagrama De Conexión Del Bloque De Recepción y Transmisión

Para el diseño del bloque de recepción y transmisión se ha establecido el uso del LimeSDR, debido a que cumple con los requerimientos establecidos, en cuanto a frecuencias de operación, bandas de frecuencias LTE y ancho de banda. Este dispositivo se interconecta con el Dongle LTE, este dispositivo LimeSDR se conecta mediante USB al eNodeB para la transmisión y recepción de señales LTE. En la figura 27 se muestra la distribución y conexión.

Figura 27

Diagrama de conexión de LimeSDR



Fuente: elaborado por el autor

3.4.3. Bloque De Procesamiento y Comunicación

Este bloque forma parte de la red de acceso radio E-UTRAN, la cual se conoce como una red distribuida de estaciones base, denominadas Evolved NodeB (eNodeB, eNB) de LTE.

A su vez, este bloque será el encargado de realizar las operaciones de procesamiento de la señal, como la modulación y codificación de los datos, al igual que la planificación y gestión de la transmisión de los paquetes en el enlace DL y UL.

Está formado por el OAI eNodeB que implementa las funcionalidades del eNodeB de LTE, conocido dentro del software OpenAirInterface en base a los repositorios como openairinterface-5g. Y el dispositivo LimeSDR que se encarga de la transmisión y recepción de señales de radiofrecuencia.

Este bloque se encarga del procesamiento y comunicación, el eNodeB actúa como un puente de capa de enlace de datos entre el Equipo de Usuario (UE) y el EPC, es decir, es el punto final de los protocolos de radio hacia el UE y retransmitiendo datos entre la conexión de radio y la conectividad basada en IP hacia el EPC.

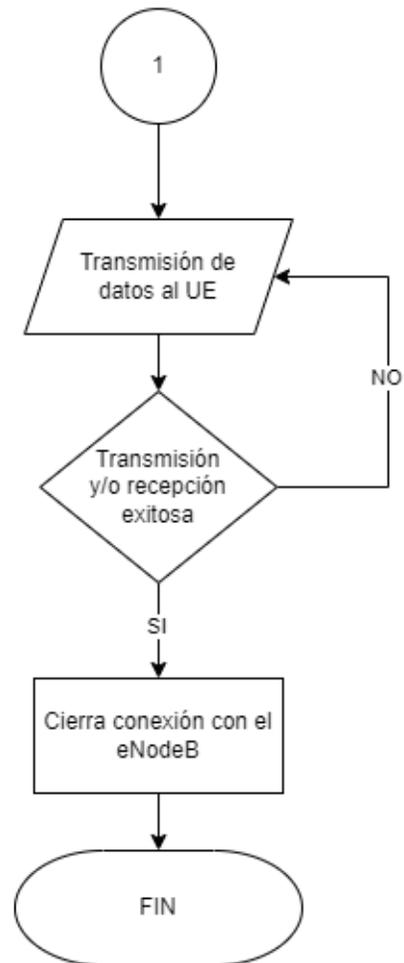
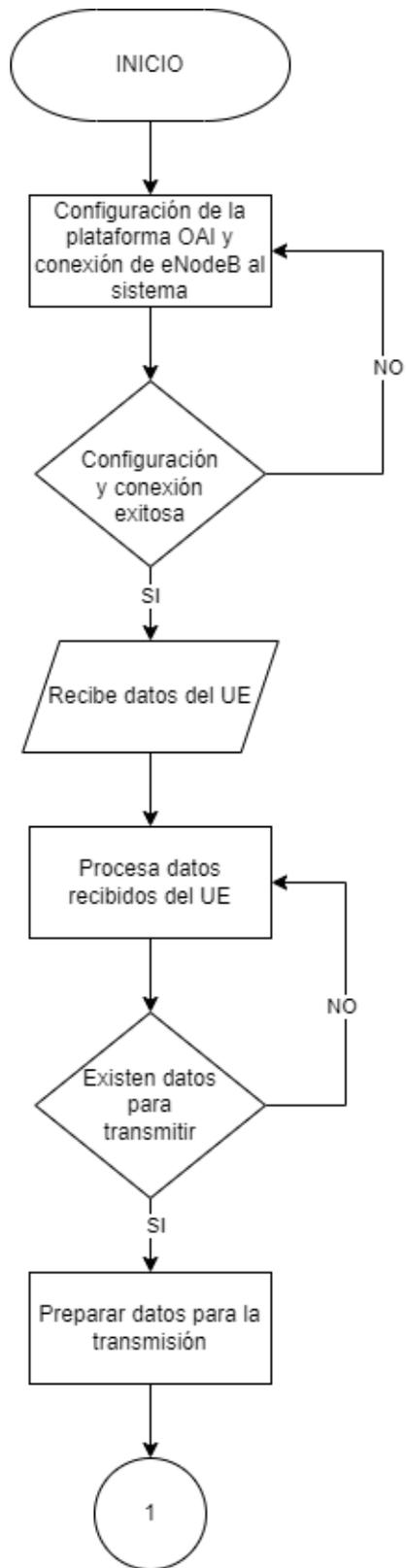
El eNB tiene dos interfaces para el MME y SPGW. La interfaz S1-MME es la interfaz de control para intercambiar mensajes de señalización del Protocolo de aplicación S1 (S1AP) con MME, y S1-U es la interfaz del plano de datos para paquetes de datos.

3.4.3.1. Diagrama de Flujo Del Bloque De Procesamiento Y Comunicación

El diagrama de flujo propuesto para el bloque de procesamiento y comunicación, como se puede observar en la figura 28 se muestra el proceso en el cual se inicia con la configuración de la plataforma OAI, en base a los parámetros para el eNodeB, seguido de la recepción de señales donde el eNodeB recibe las señales de radio, los mismos que se procesan usando el software de la plataforma OAI.

Figura 28

Diagrama de flujo del bloque de recepción y transmisión



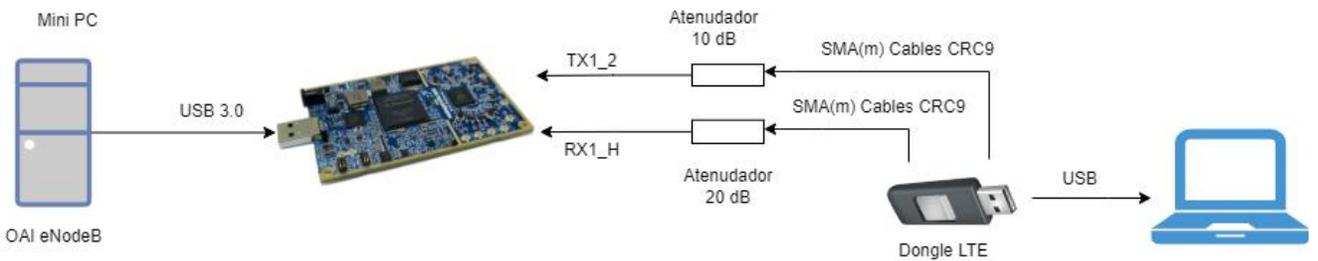
Fuente: elaborado por el autor

3.4.3.2. Diagrama de Conexión Del Bloque De Procesamiento Y Comunicación

Para el diseño del bloque de procesamiento y comunicación se ejecuta en un único local host, es decir una máquina física (Mini PC), que ejecuta la plataforma OpenAirInterface (OAI) eNodeB, esto debido a que cumple con los requerimientos establecidos. Esta máquina física se interconecta con el LimeSDR. este dispositivo se conecta mediante USB al eNodeB. En la figura 29 se muestra la distribución y conexión.

Figura 29

Diagrama de conexión de máquina física (OAI) eNodeB y LimeSDR



Fuente: elaborado por el autor

3.4.4. Bloque De Gestión

Este bloque forma parte del núcleo de red EPC, la cual se conoce como una red de servidores, enrutadores, que se encargan de la gestión del tráfico de red.

A su vez, este bloque será el encargado de trabajar en conmutación de paquete con el protocolo IP para el nivel de red. De igual manera que el bloque de procesamiento y comunicación está formado por el OAI EPC que implementa las funcionalidades del EPC de LTE, conocido dentro del software OpenAirInterface en base a los repositorios como openair-cn.

Este bloque se encarga de la gestión, el EPC implementa los siguientes componentes como el MME, que se encarga de las funcionalidades del MME del núcleo de red de LTE, así como también el HSS que implementa las funcionalidades del HSS del núcleo de red de LTE y el S-PGW que implementa las funcionalidades del S-GW y P-GW del núcleo de red de LTE.

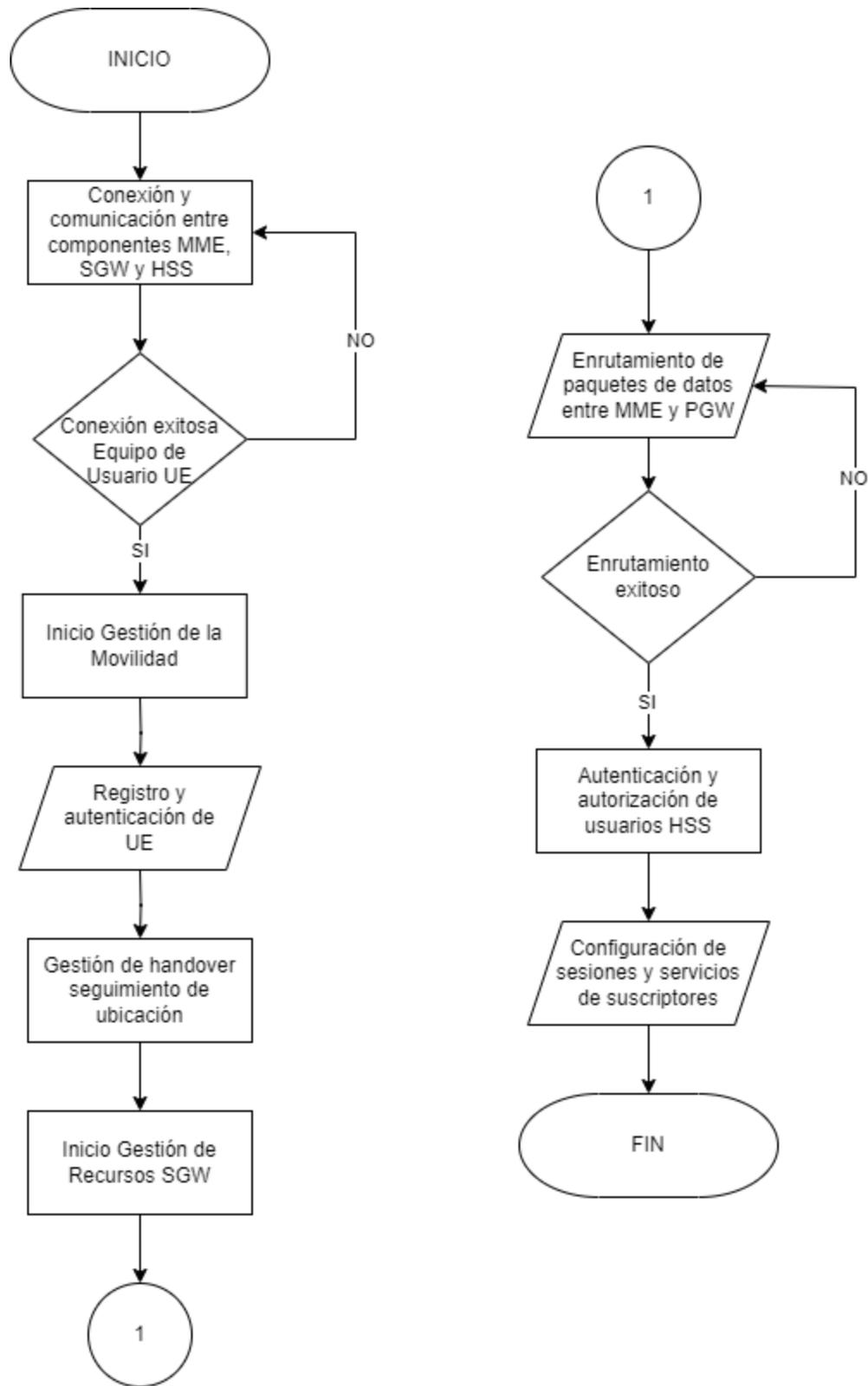
Dentro del EPC tanto el MME y SPGW es un solo ejecutable en OAI, se comunican entre sí a través de la cola ITTI (interfaz de tareas internas). El HSS en cambio es un ejecutable independiente que se comunica con MME a través de la interfaz S6a.

3.4.4.1. Diagrama de Flujo Del Bloque De Gestión

El diagrama de flujo propuesto para el bloque de gestión, se puede observar en la figura 30 donde se muestra el proceso en el cual se inicia con la conexión, configuración de los componentes donde se definen los parámetros de red, direcciones IP y configuración de seguridad, así como también los componentes de gestión de la movilidad MME, gestión de recursos SGW con el enrutamiento de paquetes de datos, seguido de la autenticación y autorización HSS, en la plataforma OAI.

Figura 30

Diagrama de flujo del bloque de gestión



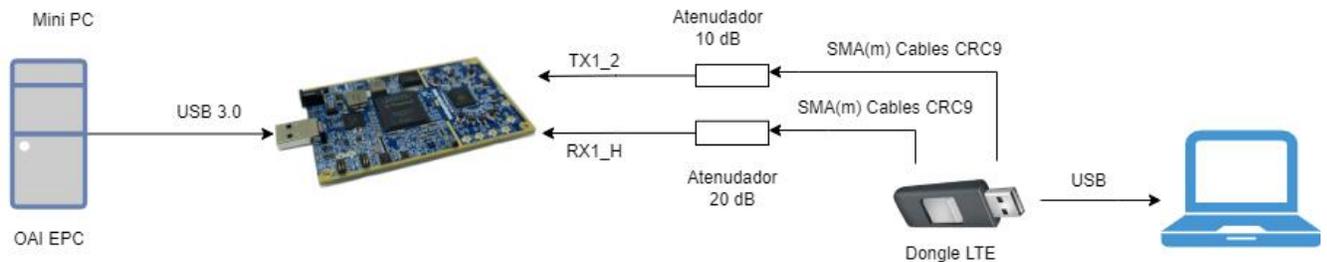
Fuente: elaborado por el autor

3.4.4.2. Diagrama de Conexión Del Bloque De Gestión

Para el diseño del bloque de gestión se ejecuta igual que el bloque de procesamiento y comunicación en un único local host, es decir una máquina física, que ejecuta la plataforma OpenAirInterface (OAI) EPC, esto debido a que cumple con los requerimientos establecidos. Esta máquina física se interconecta con el LimeSDR. este dispositivo se conecta mediante USB al eNodeB. En la figura 31 se muestra la distribución y conexión.

Figura 31

Diagrama de conexión de máquina física (OAI) EPC y LimeSDR



Fuente: elaborado por el autor

3.4.5. Bloque De Visualización

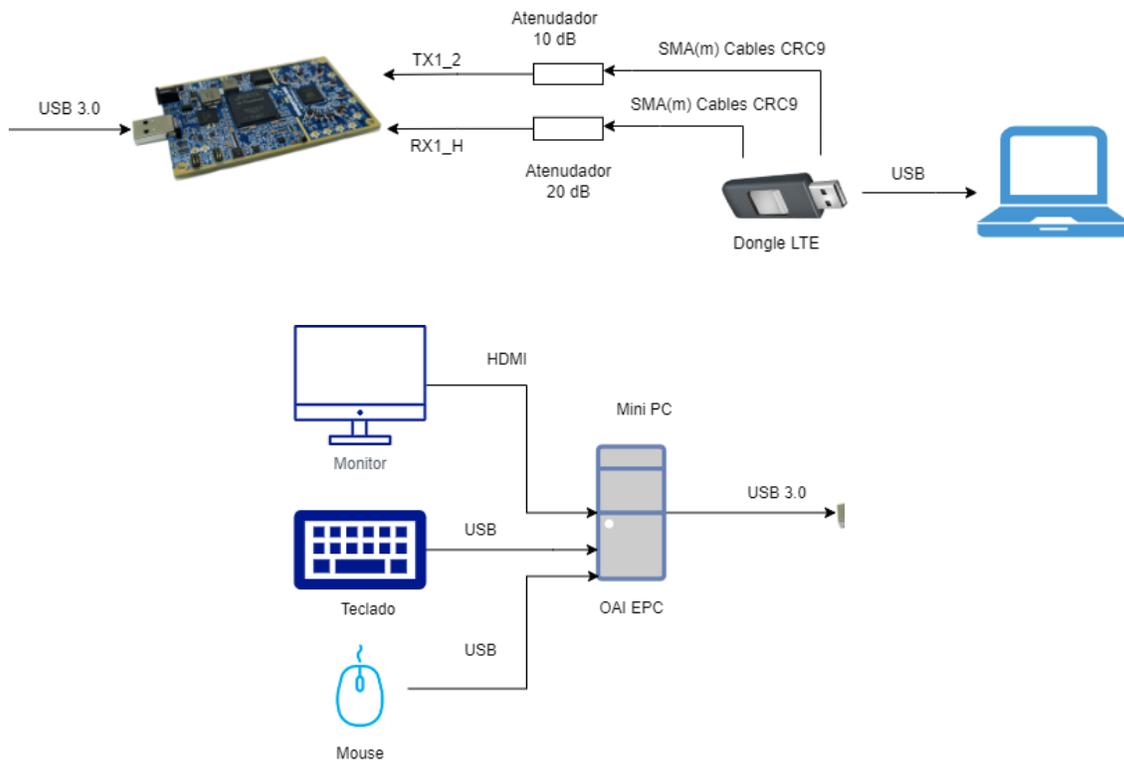
En este bloque se permite al usuario realizar un monitoreo de la capa física (PHY) tanto para el eNodeB como para el UE, en base a una traza de la potencia de la señal recibida, la respuesta del impulso del canal, la respuesta de la frecuencia del canal, el rendimiento y los componentes de las constelaciones en base a la fase y cuadratura (I/Q).

Por otra parte, las ventanas de estadísticas de la capa física PHY y MAC dan a conocer el estado de la red, ruido, potencia de la señal (mediciones de eNodeB).

Para el diseño del bloque de visualización se muestra en un monitor el cual se puede observar tanto para el eNodeB como para el UE, parámetros de la potencia de la señal recibida, la respuesta del impulso, la frecuencia del canal, así como también estadísticas sobre el estado de la red, ruido, potencia de la señal en base al eNodeB y el osciloscopio de UE que da a conocer datos del enlace ascendente y diagrama de constelación. Este monitor se conecta con el MiniPC mediante HDMI. Donde también se puede observar la conexión de los periféricos de entrada como el teclado y mouse. En la figura 32 se muestra la distribución y conexión.

Figura 32

Diagrama de conexión de monitor con máquina física (MiniPC)



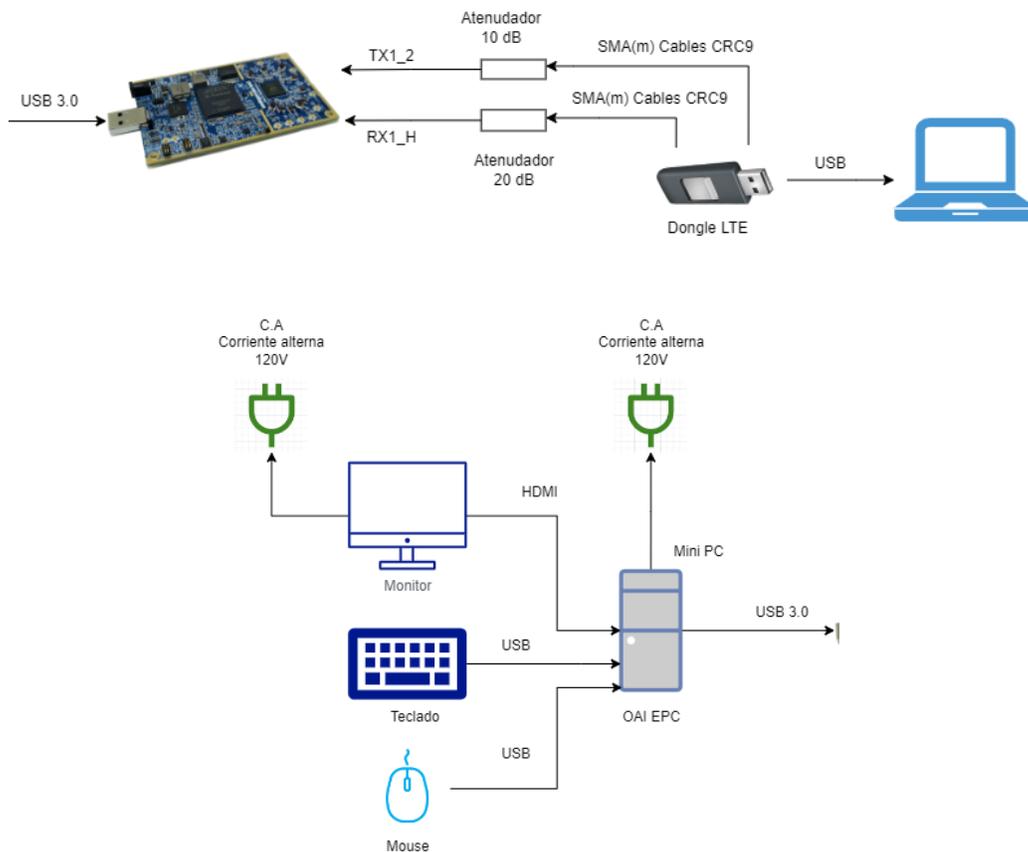
Fuente: elaborado por el autor

3.4.6. Bloque De Alimentación

La alimentación para los dispositivos como el monitor, la Mini PC, los cuales se basan en la corriente alterna, tomando en cuenta la fuente de poder tanto del monitor como de la mini PC, la cual se encargará de la conversión de la electricidad de corriente alterna a corriente directa, es decir estas fuentes de poder de cada dispositivo brindarán el suministro de energía a cada componente del sistema incluyendo a los periféricos de entrada.

Figura 33

Diagrama de conexión de alimentación de mini PC, LimeSDR y periféricos de entrada



Fuente: elaborado por el autor

4. Capítulo IV: Implementación

En esta sección, se lleva a cabo la implementación tanto a nivel de hardware como de software en cada uno de los bloques establecidos. Esto culmina en la obtención del sistema final como resultado.

4.1. Implementación De Hardware

Para la implementación de hardware se realiza la conexión de cada uno de los dispositivos que forman parte del sistema. A continuación, se detalla cada bloque con su conexión planteada.

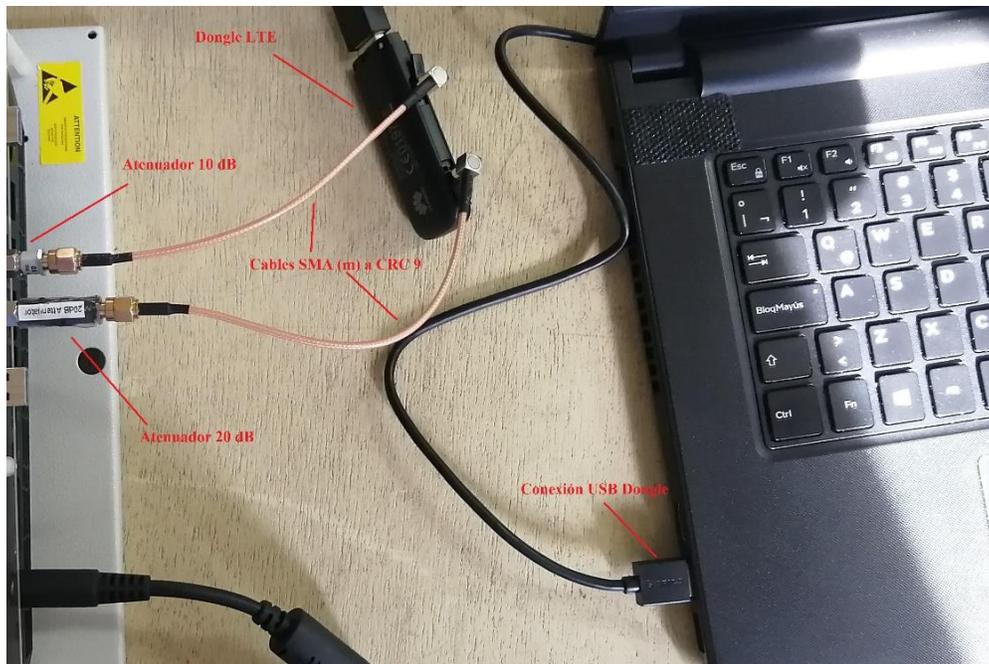
4.1.1. Bloque De Adquisición De Datos

Para la parte de adquisición de datos el cual corresponde al dispositivo de usuario final (computadora) y el dongle USB que es el UE para el establecimiento de la conexión con la red LTE, donde se puede verificar la conexión del dongle por USB a la computadora y a su vez, la conexión del dongle con los cables SMA (Subminiatura Versión A) (m) a CRC 9 los cuales se usan para la conexión entre dispositivos electrónicos como módems USB, en este caso el dongle. Los cables SMA permiten adaptar la conexión entre un conector SMA macho (SMA m) y un conector CRC 9, con el objetivo de mejorar la recepción de la señal.

Así también, por otra parte, a los cables SMA se conectan dos atenuadores, en RX de 20 dB y en TX de 10 dB respectivamente, que sirven para reducir la potencia de la señal de radio frecuencia RF transmitida a través del cable. En la figura 34 se observa la conexión del dongle, los cables SMA (m) a CRC 9 y los atenuadores.

Figura 34

Conexión de dongle, cables SMA y atenuadores



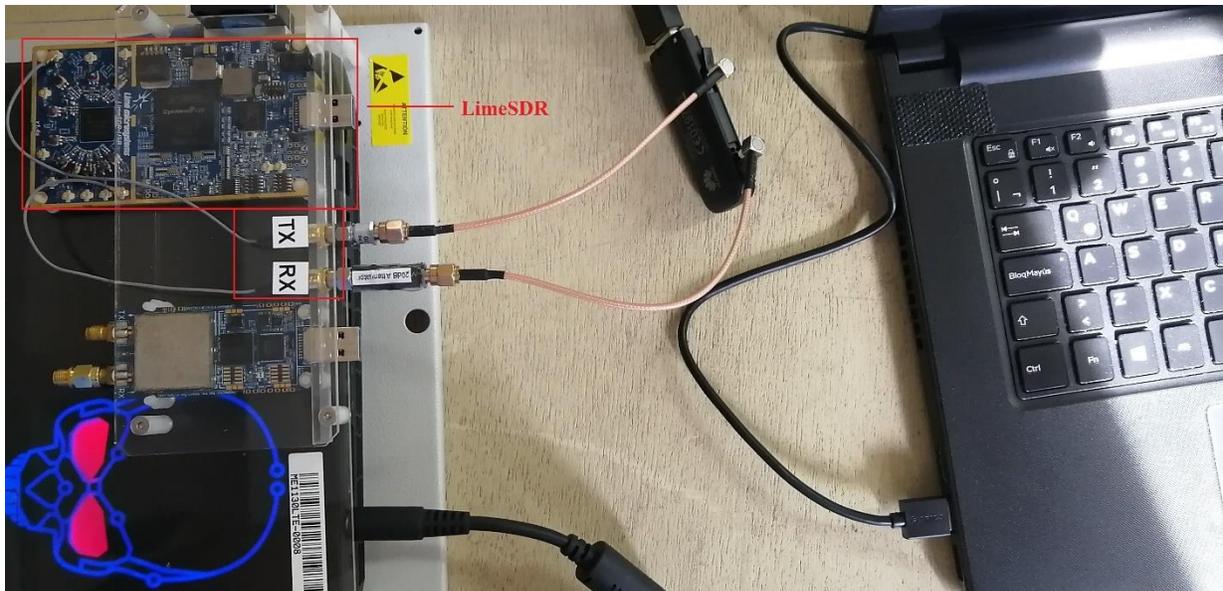
Fuente: elaborado por el autor

4.1.2. Bloque De Recepción y Transmisión

Para el bloque de recepción y transmisión una vez establecida la conexión del Dongle LTE, los cables SMA (m) a CRC 9 y los atenuadores, se realiza la conexión al LimeSDR hacia la mini PC por USB, destacando nuevamente que en el LimeSDR el TX corresponde al atenuador de 10 dB y en el RX el atenuador de 20 dB, esto es posible ya que los atenuadores al tener conectores de tipo SMA macho y el LimeSDR conectores U.FL (conectores IPX o MHF) en su placa permiten la conexión directa con los atenuadores. Este dispositivo SDR es el encargado de la transmisión y recepción de las señales de radiofrecuencia. En la figura 35 se muestra la conexión mencionada anteriormente.

Figura 35

Conexión de dongle, cables SMA, atenuadores al LimeSDR



Fuente: elaborado por el autor

4.1.3. Bloque De Visualización

Para el bloque de visualización se realiza mediante la conexión mediante cable HDMI desde la máquina física (Mini PC) hacia el monitor donde se observará los parámetros de potencia de la señal que es recibida, así como la respuesta del impulso, la frecuencia del canal y estadísticas del estado de la red, ruido, potencia de la señal en base al eNodeB. Y el osciloscopio del UE que muestra datos del enlace ascendente y diagrama de constelación. En la figura 36 se identifica la conexión respectiva para la visualización. Además, se presenta la conexión de los periféricos de entrada como el teclado y mouse que servirán para manejo del Mini PC.

Figura 36

Conexión HDMI Mini PC y HDMI Monitor



Fuente: elaborado por el autor

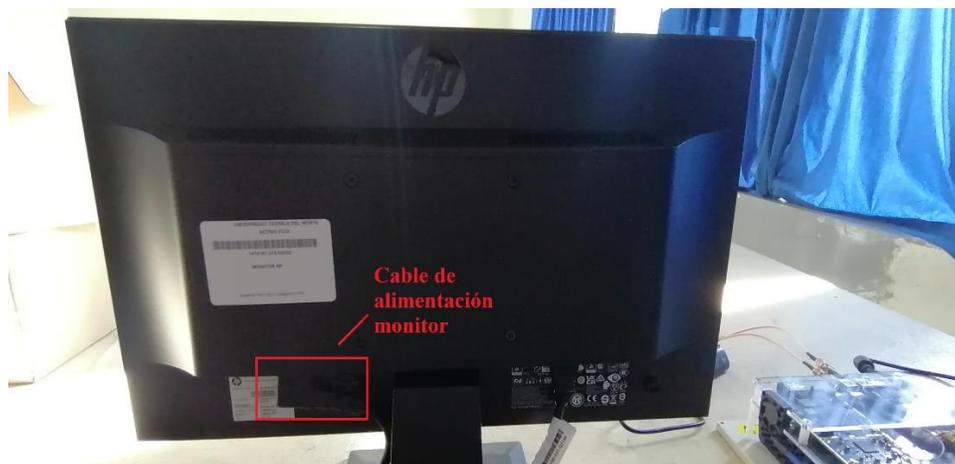
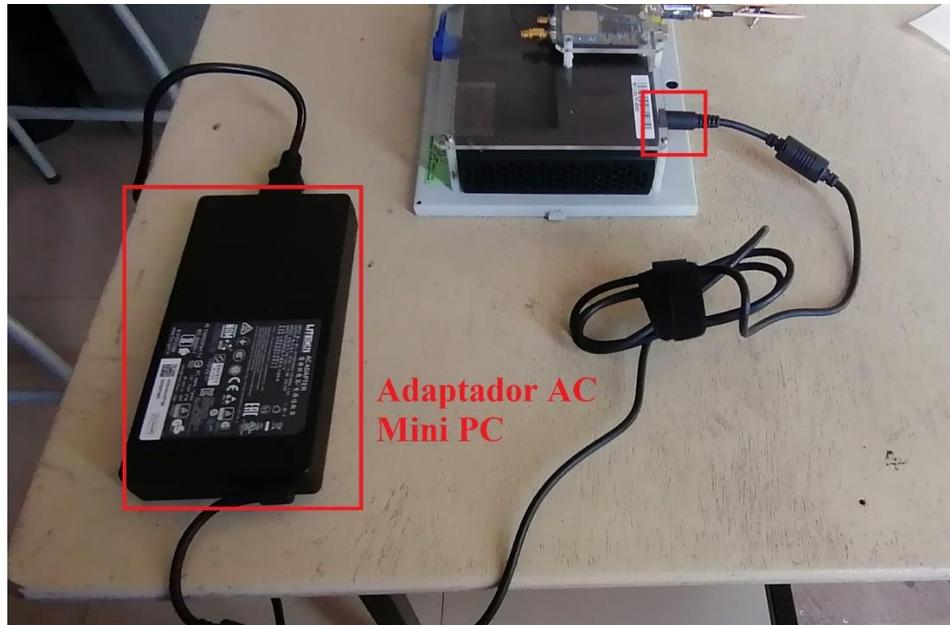
4.1.4. Bloque De Alimentación

Para el bloque de alimentación se realiza la conexión tanto del monitor y el Mini PC, los cuales se basan en la corriente alterna y con las características de cada equipo se determina el consumo de corriente que tendrá cada uno, en este caso para el Mini PC el voltaje de salida es de 19.5 V, su consumo de corriente es de 11.8 A y su consumo de energía es de 230 W. En cuanto

al monitor se obtiene un consumo de energía de 8 Wh (4 W), dando como resultado el consumo total de energía de 234 W. En la figura 37 se observa la conexión del Mini PC y del monitor.

Figura 37

Alimentación Mini PC y monitor



Fuente: elaborado por el autor

4.2. Implementación De Software

Para la implementación del software se realizan las configuraciones del, E-UTRAN y EPC en OpenAirInterface. Para el E-UTRAN (eNodeB) el cual tiene dos interfaces para MME y SPGW. En cuanto al núcleo de red (EPC) el MME y SPGW es un solo ejecutable en OAI y el HSS es un ejecutable independiente. Finalmente se realiza la configuración del UE (Dongle USB LTE).

Para estas configuraciones se procede a utilizar los equipos adquiridos y disponibles de la Universidad, como es el ME1130 para LTE y LTE-Advanced de (DreamCatcher, 2020).

4.2.1. Bloque De Procesamiento y Comunicación

Para el bloque de procesamiento y comunicación el E-UTRAN (eNodeB) como se mencionó anteriormente consta de dos interfaces para MME y SPGW. Es decir, la interfaz de control S1-MME para el intercambio de mensajes de señalización del protocolo de aplicación S1 (S1AP) con MME. Y la interfaz del plano de datos S1-U paquetes de datos. Para el eNodeB y MME en la interfaz S1-MME el direccionamiento es 127.0.0.20/8. Así mismo para el eNodeB y SPGW en la interfaz S1-U el direccionamiento es 127.0.0.30/8. Mientras que el eNodeB tendrá el direccionamiento 127.0.0.10/8.

Además, se debe considerar que la Mini PC donde se encuentra el software OAI para eNodeB y EPC debe estar previamente conectado a internet por el cable Ethernet y el Wi-Fi debe estar deshabilitado.

4.2.1.1. Configuración de OpenAirInterface

Para la configuración de OAI en primera instancia se debe señalar las tareas que se deben realizar para el funcionamiento y así poder realizar el despliegue del prototipo de red. Esta configuración se lleva a cabo con los ficheros de configuración en forma de texto o scripts.

Para la correcta configuración de la plataforma de OAI, se requiere configurar tanto el E-UTRAN (eNodeB) como el núcleo de red EPC.

4.2.1.2. Configuración de eNodeB

Para la configuración de los distintos parámetros del eNodeB, se realiza en el fichero de texto. Donde la plataforma OAI es la encargada de brindar scripts o archivos de configuración en este caso para el eNodeB, los cuales se pueden encontrar en la carpeta

~/openairinterface5g/targets/PROJECTS/GENERIC-LTE-EPC/CONF/

En este caso, para esta configuración, se selecciona de los archivos disponibles para bandas LTE, en base al ancho de banda y requisitos de hardware. Y se procede a modificar el archivo el mismo que ya contiene parámetros de configuración adecuados al LimeSDR, este archivo corresponde al “enb.ban7.tml.25PRB.lmssdr.conf” usado para la respectiva configuración del eNodeB, el cual trabaja en la banda 7 (2600 MHz) y usa 25 Bloques de Recursos Físicos - Physical Resource Block (PRB) establecido como la unidad mínima de recursos que un eNodeB asigna a un terminal móvil o UE. El LimeSDR actúa como interfaz de RF.

Del archivo mencionado, se modifica los parámetros descritos a continuación:

- Parámetros de identificación:
 - *Active_eNBs*: especifica el nombre del eNodeB

- *tracking_area_code*: identificador del área de seguimiento (código de área)
- *mobile_country_code*: Mobile Country Code – Código Móvil de País (MCC)
- *mobile_network_code*: Mobile Network Code – Código Móvil de Red (MNC)
- Parámetros de nivel físico:
 - *frame_type*: configuración del tipo para el duplexado (FDD o TDD)
 - *eutra_band*: banda de operación
 - *downlink_frequency*: enlace descendente frecuencia de operación (frecuencia dentro del rango de la banda de operación)
 - *uplink_frequency_offset*: desplazamiento en frecuencia para el enlace ascendente en base al enlace descendente
 - *Nid_cell*: número de identificación de la celda
 - *N_RB_DL*: número máximo de bloques de recursos físicos en enlace descendente
 - *nb_antenas_tx*: número de antenas para transmisión
 - *nb_antenas_rx*: número de antenas para recepción
 - *tx_gain*: ganancia en dB para transmisión para dispositivo LimeSDR
 - *rx_gain*: ganancia en dB para recepción para dispositivo LimeSDR
- Parámetros de dirección IP del MME:
 - *mme_ip_address_ipv4*: dirección IP del MME
 - *preference*: selección de preferencia de IP
- Parámetros de interfaces de red:
 - *ENB_INTERFACE_NAME_FOR_S1_MME*: nombre de la interfaz de red para S1-MME especificada por OAI

- *ENB_IPV4_ADDRESS_FOR_S1_MME*: dirección IP del eNodeB en interfaz S1-MME
- *ENB_INTERFACE_NAME_FOR_S1U*: nombre de la interfaz de red para la interfaz S1-U
- *ENB_IPV4_ADDRESS_FOR_S1U*: dirección IP del eNodeB en interfaz S1-U

Una vez definidos los parámetros, se detalla los valores establecidos, como se muestra a continuación:

- Parámetros de identificación:
 - *Active_eNBs*: eNB_LTEBox
 - *eNB_name*: eNB_LTEBox
 - *tracking_area_code*: 1
 - *mobile_country_code*: 901 (Anteriormente Clementvale Baltic OÜ)
 - *mobile_network_code*: 70 (Anteriormente Clementvale Baltic OÜ) (ITU, 2008)
- Parámetros de nivel físico:
 - *frame_type*: FDD
 - *eutra_band*: banda 7
 - *downlink_frequency*: 2680000000L
 - *uplink_frequency_offset*: -120000000
 - *Nid_cell*: 0
 - *N_RB_DL*: 25
 - *nb_antenas_tx*: 1
 - *nb_antenas_rx*: 1
 - *tx_gain*: 70 dB

- *rx_gain*: 103 dB
- Parámetros de dirección IP del MME:
 - *mme_ip_address ipv4*: 127.0.0.20
 - *preference*: ipv4
- Parámetros de interfaces de red:
 - *ENB_INTERFACE_NAME_FOR_S1_MME*: lo (interfaz local)
 - *ENB_IPV4_ADDRESS_FOR_S1_MME*: 127.0.0.10/8
 - *ENB_INTERFACE_NAME_FOR_S1U*: lo
 - *ENB_IPV4_ADDRESS_FOR_S1U*: 127.0.0.10/8

Los parámetros no definidos se establecieron con sus valores por defecto. El fichero completo de configuración del eNodeB se puede observar en el Anexo 3.

4.2.2. Bloque De Gestión

Para el bloque de gestión, el núcleo de red (EPC) consta de dos interfaces para MME, SPGW y HSS. Es decir, el HSS es un ejecutable independiente que se comunica con el MME por la interfaz S6a el direccionamiento es 127.0.0.0/8. En cuanto al MME y SPGW representan un solo ejecutable en OAI y se comunican entre sí con la interfaz S11 en este caso por medio de la interfaz de tareas internas.

Para la configuración de los parámetros del EPC, respectivamente de cada componente, se realiza de la misma manera como la configuración del eNodeB mediante ficheros de texto. En este caso la plataforma OAI es la encargada de brindar scripts o archivos de configuración para el EPC, estos se pueden encontrar en la carpeta *~/openair-cn/etc/*.

4.2.2.1. Configuración de HSS

El procedimiento de configuración del HSS se lleva a cabo mediante la modificación del fichero de texto de la configuración general el cual está localizado en la carpeta *cd/usr/local/etc/oai* y para editar se ejecuta el comando *sudo nano hss.conf*.

A continuación, se detalla los parámetros a modificar:

- Parámetros obligatorios de MySQL:
 - *MYSQL_server*: dirección IP servidor MySQL, dirección de enlace HSS S6a.
 - *MYSQL_user*: identificación de usuario en el servidor MySQL, inicio de sesión del servidor
 - *MYSQL_pass*: contraseña de identificación del servidor MySQL
- Parámetros de HSS:
 - *OPERATOR_key*: clave del operador. Se establece por defecto, de la siguiente manera “”

Con los parámetros ya definidos, se detalla los valores establecidos, quedando de la siguiente manera:

- Parámetros obligatorios de MySQL:
 - *MYSQL_server*: 127.0.0.1 (localhost)
 - *MYSQL_user*: root
 - *MYSQL_pass*: lte
- Parámetros de HSS:
 - *OPERATOR_key*: “”

Los parámetros no mencionados se dejan con sus valores por defecto. El fichero completo de configuración del HSS se observa en el Anexo 4.

4.2.2.2. Configuración de MME

Para la configuración del MME se realiza con la modificación del fichero de texto de la configuración general el cual está localizado en la carpeta `cd/usr/local/etc/oai` y para editar se ejecuta el comando `sudo nano mme.conf`.

De la misma forma, se detalla los parámetros importantes:

- Parámetros MME:
 - *REALM*: dominio del sistema
 - *MAXENB*: máximo número de eNodeB
 - *MAXUE*: máximo número de terminales móviles de conexión al prototipo de red
 - *RELATIVE_CAPACITY*: capacidad relativa del sistema
- Parámetros GUMMEI LIST: lista de GUMMEI (Globally Unique MME Identifier – Identificador Global y Único del MME). Dichos parámetros deben ser igual que los establecidos con el eNodeB anteriormente.
 - *MCC*: Código Móvil de País
 - *MNC*: Código Móvil de Red
 - *MME_GID*: Código GID
 - *MME_CODE*: Código MME
- Parámetros TAI LIST: lista de TAI (Tracking Area Identity – Identidad del Área de Seguimiento). Estos parámetros deben coincidir con los del eNodeB.
 - *MCC*: Código Móvil de País
 - *MNC*: Código Móvil de Red

- *TAI*: Código área de seguimiento
- Parámetros de interfaces de red:
 - *MME_INTERFACE_NAME_FOR_S1_MME*: nombre de la interfaz de red MME para la interfaz S1-MME de LTE
 - *MME_IPV4_ADDRESS_FOR_S1_MME*: dirección IP del MME para la interfaz S1 de LTE
 - *MME_INTERFACE_NAME_FOR_S11_MME*: nombre de la interfaz de red MME para la interfaz S11 de LTE (MME y SPGW un solo ejecutable)
 - *MME_IPV4_ADDRESS_FOR_S11_MME*: dirección IP del MME para la interfaz S11 de LTE
 - *MME_PORT_FOR_S11_MME*: puerto MME para S11
- Parámetros de IP del S-GW:
 - *SGW_IPV4_ADDRESS_FOR_S11*: dirección IP al SPGW para la interfaz S11 de LTE

Una vez definidos los parámetros importantes, se detalla los valores establecidos de la siguiente manera:

- Parámetros MME:
 - *REALM*: OpenAir5G.Alliance
 - *MAXENB*: 2 (por defecto)
 - *MAXUE*: 16 (por defecto)
 - *RELATIVE_CAPACITY*: 10 (por defecto)

- Parámetros GUMMEI LIST: lista de GUMMEI (Globally Unique MME Identifier – Identificador Global y Único del MME). Dichos parámetros deben ser igual que los establecidos con el eNodeB anteriormente.
 - *MCC*: 901
 - *MNC*: 70
 - *MME_GID*: 4 (por defecto)
 - *MME_CODE*: 1

- Parámetros TAI LIST: lista de TAI (Tracking Area Identity – Identidad del Área de Seguimiento). Estos parámetros deben coincidir con los del eNodeB.
 - *MCC*: 901
 - *MNC*: 70
 - *TAI*: 1

- Parámetros de interfaces de red:
 - *MME_INTERFACE_NAME_FOR_S1_MME*: lo (interfaz local)
 - *MME_IPV4_ADDRESS_FOR_S1_MME*: 127.0.0.20/8
 - *MME_INTERFACE_NAME_FOR_S11_MME*: lo
 - *MME_IPV4_ADDRESS_FOR_S11_MME*: 127.0.0.20/8
 - *MME_PORT_FOR_S11_MME*: 2123

- Parámetros de IP del S-GW:

SGW_IPV4_ADDRESS_FOR_S11: 127.0.0.30/8

Los parámetros no establecidos se dejan con los valores por defecto. El fichero completo de configuración del MME se observa en el Anexo 5.

4.2.2.3. Configuración de SPGW

Para la realización de la configuración del SPGW se procede con la modificación del fichero de texto el cual se encuentra en la carpeta *cd/usr/local/etc/oai* y para editar se ejecuta el comando *sudo nano spgw.conf*.

Y se detalla los parámetros más importantes del fichero:

- Parámetros de interfaces de red S-GW:

Interfaz vinculada a S-GW para comunicación S11 (GTPV2-C), en caso de no seleccionar ninguna, se usa la interfaz de mensajes ITTI (Interfaz de tareas internas)

- *SGW_INTERFACE_NAME_FOR_S11*: nombre de interfaz S-PGW para S11 de LTE (conexión MME con S-PGW) interfaz local
- *SGW_IPV4_ADDRESS_FOR_S11*: dirección IP del S-PGW para S11 de LTE, relación con el valor configurado en MME

Interfaz vinculada a S-GW para la comunicación S1-U (GTPV1-U) podría ser una interfaz ethernet, ethernet virtual, excepto interfaces inalámbricas

- *SGW_INTERFACE_NAME_FOR_S1U_S12_S4_UP*: nombre de interfaz S-PGW para S1 (S1-U) de LTE (conexión con eNodeB)
- *SGW_IPV4_ADDRESS_FOR_S1U_S12_S4_UP*: dirección IP del S-PGW en interfaz S1 (S1-U) de LTE
- *SGW_IPV4_PORT_FOR_S1U_S12_S4_UP*: puerto IPv4 SGW para interfaz S1 (S1-U) de LTE

Interfaz vinculada a S-GW para comunicación S5 o S8 (no implementada)

- *SGW_INTERFACE_NAME_FOR_S5_S8_UP*: nombre de interfaz S-PGW para conexión S5 o S8
- *SGW_IPV4_ADDRESS_FOR_S5_S8_UP*: dirección IP del S-PGW para S5 o S8
- Parámetros de interfaces de red P-GW:

Interfaz vinculada a P-GW para comunicación S5 o S8 (no implementada)

- *PGW_INTERFACE_NAME_FOR_S5_S8*: nombre de interfaz P-GW para interfaz S5 o S8

Interfaz vinculada a P-GW para tráfico de internet de salida/entrada (SGI)

- *PGW_INTERFACE_NAME_FOR_SGI*: nombre de interfaz del S-PGW para SGI de LTE, salida a internet al EPC
- *PGW_MASQUERADE_SGI*: especificación si el S-PGW realiza NAT (Network Address Translation – Traducción de Direcciones de Red)

- Parámetros de dirección DNS a UE:
 - *DEFAULT_DNS_IPV4_ADDRESS*: servidor DNS (Domain Name System – Sistema de Nombres de Dominio) de terminal móvil para Internet
 - *DEFAULT_DNS_SEC_IPV4_ADDRESS*: servidor DNS secundario

Una vez establecidos los parámetros, se detalla los valores establecidos de la siguiente manera:

- Parámetros de interfaces de red S-GW:
 - *SGW_INTERFACE_NAME_FOR_S11*: lo (interfaz local)
 - *SGW_IPV4_ADDRESS_FOR_S11*: 127.0.0.30/8
 - *SGW_INTERFACE_NAME_FOR_S1U_S12_S4_UP*: lo

- *SGW_IPV4_ADDRESS_FOR_S1U_S12_S4_UP*: 127.0.0.30/8
- *SGW_IPV4_PORT_FOR_S1U_S12_S4_UP*: 2152
- *SGW_INTERFACE_NAME_FOR_S5_S8_UP*: none (ninguno)
- *SGW_IPV4_ADDRESS_FOR_S5_S8_UP*: 0.0.0.0/24
- Parámetros de interfaces de red P-GW:
 - *PGW_INTERFACE_NAME_FOR_S5_S8*: none (ninguno)
 - *PGW_INTERFACE_NAME_FOR_SGI*: eno1
 - *PGW_MASQUERADE_SGI*: yes (si)
 - *UE_TCP_MSS_CLAMPING*: no
- Parámetros de dirección DNS a UE:
 - *DEFAULT_DNS_IPV4_ADDRESS*: 8.8.8.8
 - *DEFAULT_DNS_SEC_IPV4_ADDRESS*: 8.8.4.4

Los parámetros no establecidos no cambian por lo que se dejan con los valores por defecto. El fichero completo de configuración del S-PGW se puede observar en el Anexo 6.

4.2.3. Bloque De Adquisición De Datos

Para el bloque de adquisición de datos, se realiza la configuración del UE (dongle USB LTE) el cual se conecta al eNodeB. Para ello se deben seguir los pasos mencionados a continuación.

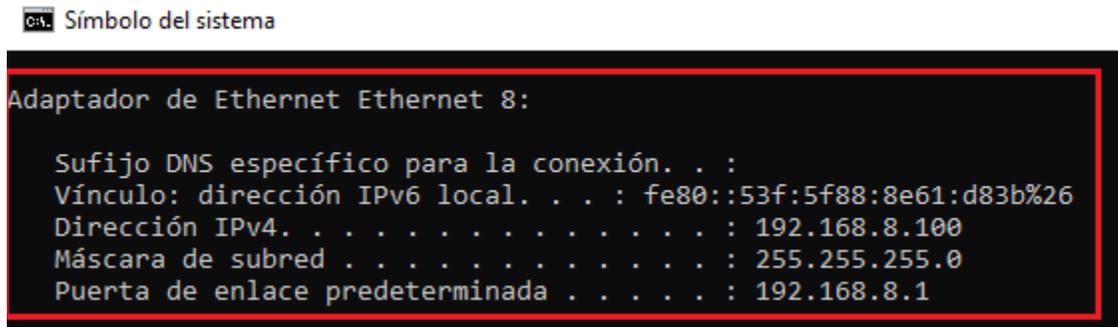
Antes de empezar con la configuración del Dongle USB LTE es necesario mencionar que NO se debe actualizar el firmware del Dongle a su última versión, ya que caso contrario dejaría de funcionar, debido a que las versiones de firmware recientes no poseen el Application Programming Interface – Interfaz de Programación de Aplicaciones (API) para el manejo del módem.

4.2.3.1. Configuración de UE (Dongle LTE)

- Conectar el dongle USB LTE en la computadora portátil, al realizar la conexión se verifica el Adaptador Ethernet creado con la dirección IP 192.168.8.100. Esto se puede comprobar en la computadora al ingresar al símbolo del sistema (CMD) y ejecutar el comando `ipconfig /all`. En la figura 38 se muestra el adaptador Ethernet creado.

Figura 38

Adaptador Ethernet para dongle LTE



```
cmd. Símbolo del sistema

Adaptador de Ethernet Ethernet 8:

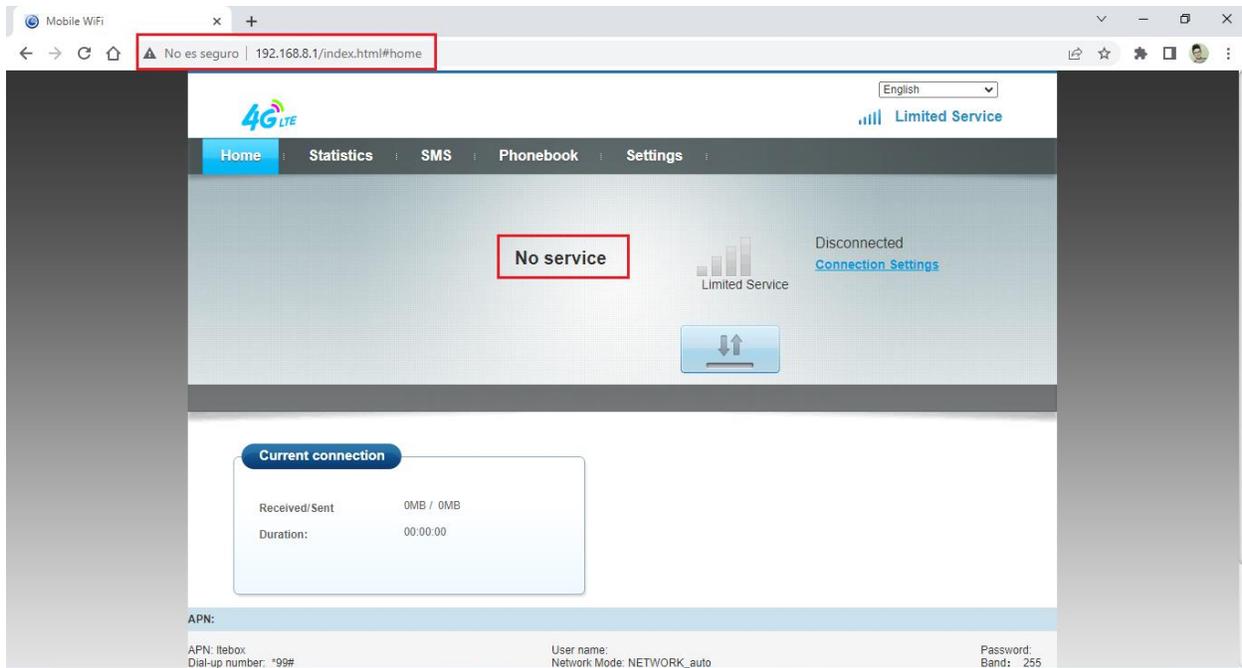
Sufijo DNS específico para la conexión. . . :
Vínculo: dirección IPv6 local. . . . . : fe80::53f:5f88:8e61:d83b%26
Dirección IPv4. . . . . : 192.168.8.100
Máscara de subred . . . . . : 255.255.255.0
Puerta de enlace predeterminada . . . . . : 192.168.8.1
```

Fuente: elaborado por el autor

- Abrir el navegador e ingresar la dirección `http://192.168.8.1` del dongle. En la figura 39 se observa que el estado se encuentra en No Service (Sin Servicio) que significa que el UE no está conectado a la red LTE.

Figura 39

Página web Dongle



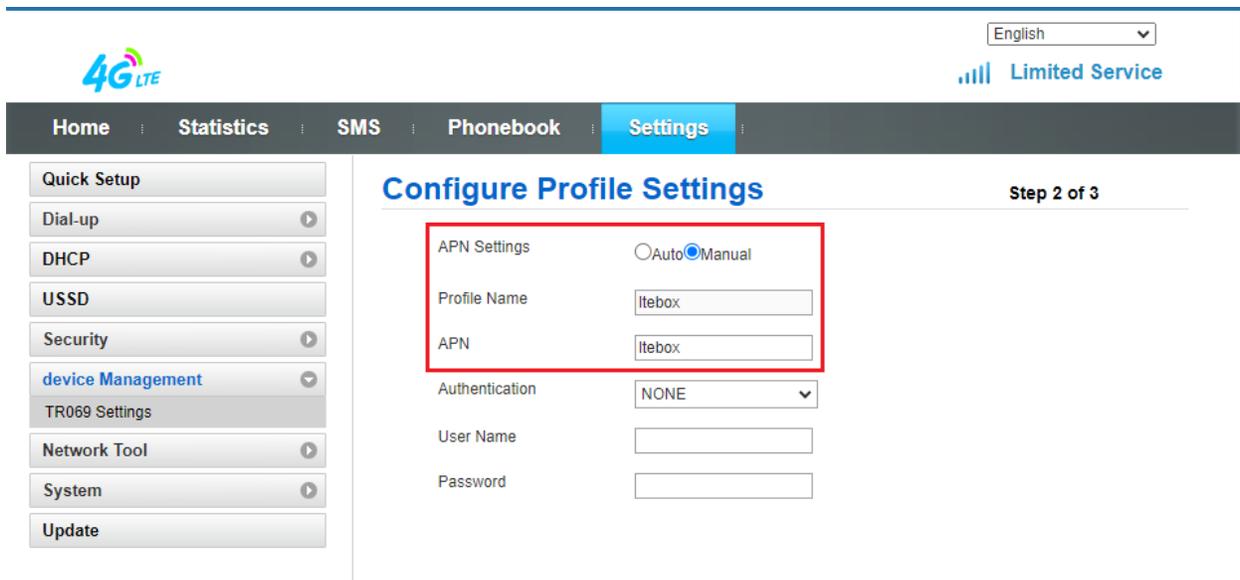
Fuente: elaborado por el autor

- En la página del dongle LTE ir al menú en “Configuración” y verificar que la opción de “Datos móviles” se encuentre habilitada. A continuación, en el menú en “Gestión de perfiles” crear un Nuevo Perfil y configurar los parámetros tanto del nombre del perfil como del Access Point Name – Nombre del Punto de Acceso (APN).
 - *Profile name:* ltebox
 - *APN:* ltebox

Los parámetros de MCC 901, MNC 70 se establecen igual que en el prototipo de red configurado anteriormente en el MME. Y para los parámetros de autenticación, nombre de usuario y contraseña, se dejan por defecto. En la figura 40 se puede verificar las configuraciones respectivas del APN.

Figura 40

Configuración APN

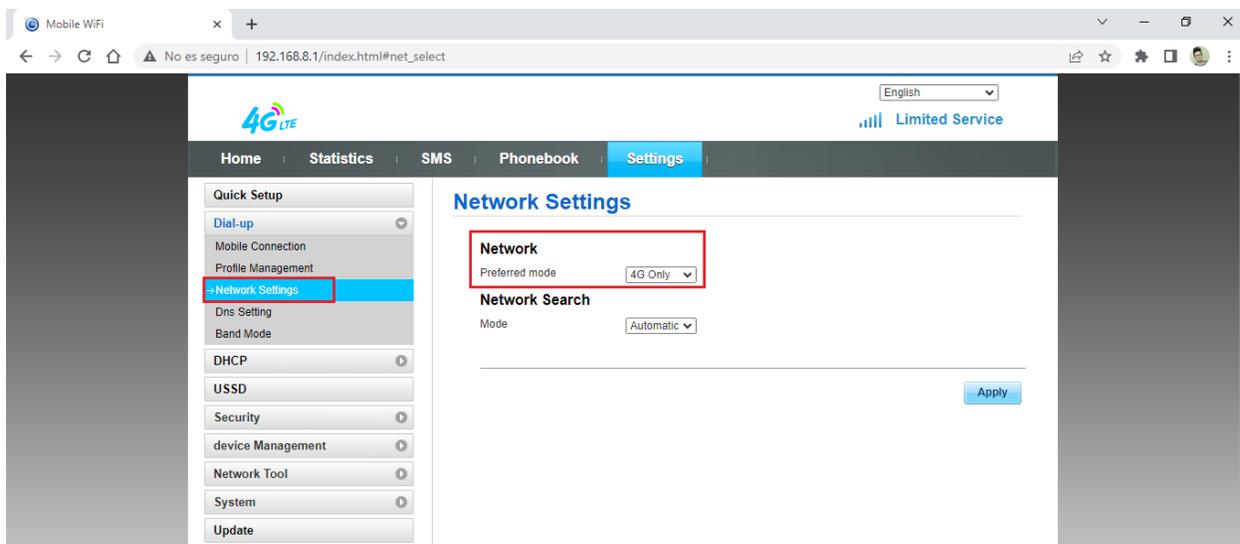


Fuente: elaborado por el autor

- Finalmente, verificar en la “Configuración de red” que se encuentre habilitada la opción de Admitir redes LTE. En la figura 41 se puede verificar la habilitación del parámetro.

Figura 41

Admitir redes LTE habilitado



Fuente: elaborado por el autor

5. Capítulo V: Pruebas De Funcionamiento Y Guías

El presente capítulo se refiere a los resultados obtenidos con la funcionalidad tanto del E-UTRAN (eNodeB) como del EPC (HSS, MME, SPGW) y del UE (Dongle LTE) mediante el establecimiento del desarrollo de guías de prácticas, logrando al usuario llegar a familiarizarse con el sistema, desarrollando tareas y configuraciones como para el análisis del protocolo LTE, análisis de tráfico de red y mediciones de red.

5.1. Funcionamiento De Red LTE

Al finalizar con la configuración respectiva de los parámetros del eNodeB, como también del EPC (HSS, MME y SPGW) detallados anteriormente mediante la modificación en los respectivos ficheros de texto, los cuales son proporcionados por la plataforma OAI encargada de brindar los scripts o archivos de configuración respectivos para cada caso. Y adicional con la configuración del dongle LTE con la creación del APN, se procede a ejecutar los módulos del software de OAI.

En la Guía 1: Software Osciloscopio, se puede observar los detalles de los comandos para la ejecución del EPC y eNodeB respectivamente junto con las especificaciones para la visualización del osciloscopio y ventanas de datos para monitorización del nivel físico.

5.1.1. Funcionamiento de EPC

Primeramente, se procede a ejecutar el núcleo de red (EPC) en el siguiente orden: HSS, MME y SPGW. Para poder ejecutar el EPC, se usan los scripts de OAI en el subdirectorio

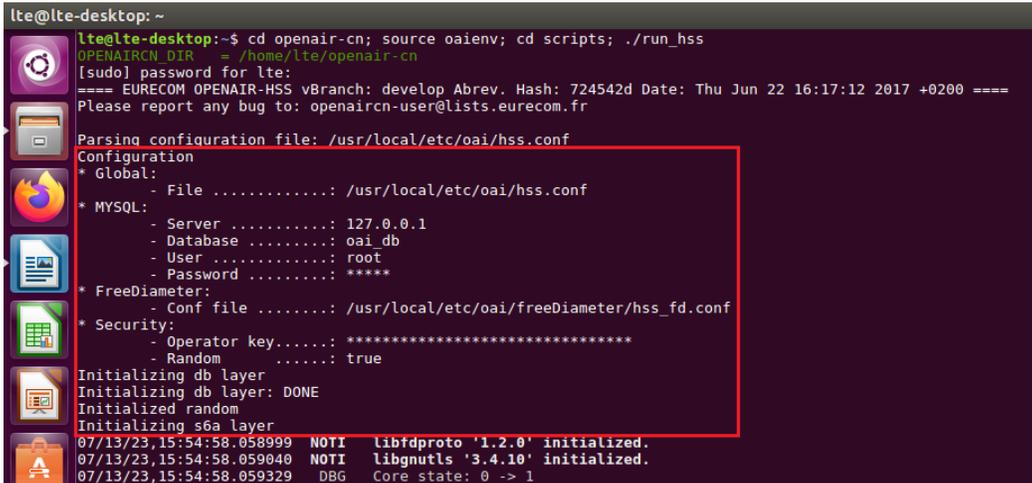
cd/usr/local/etc/oai

5.1.1.1. Ejecución de HSS

Para iniciar el HSS se debe abrir un Terminal para ejecutar el comando respectivo para su funcionamiento, al dar inicio al HSS se muestran los parámetros modificados en el script, respecto al servidor de MySQL con la dirección IP, el usuario en el servidor, con su contraseña respectiva de identificación y la inicialización de la interfaz s6a encargada de la comunicación con el MME. En la figura 42 se puede observar los detalles mencionados.

Figura 42

Ejecución de HSS



```
lte@lte-desktop: ~
lte@lte-desktop:~$ cd openair-cn; source oaienv; cd scripts; ./run_hss
OPENAIRCN_DIR = /home/lte/openair-cn
[sudo] password for lte:
==== EURECOM OPENAIR-HSS vBranch: develop Abrev. Hash: 724542d Date: Thu Jun 22 16:17:12 2017 +0200 ====
Please report any bug to: openaircn-user@lists.eurecom.fr

Parsing configuration file: /usr/local/etc/oai/hss.conf
Configuration
* Global:
  - File .....: /usr/local/etc/oai/hss.conf
* MYSQL:
  - Server .....: 127.0.0.1
  - Database .....: oai_db
  - User .....: root
  - Password .....: *****
* FreeDiameter:
  - Conf file .....: /usr/local/etc/oai/freeDiameter/hss_fd.conf
* Security:
  - Operator key.....: *****
  - Random .....: true
Initializing db layer
Initializing db layer: DONE
Initialized random
Initializing s6a layer
07/13/23,15:54:58.058999 NOTI libfdproto '1.2.0' initialized.
07/13/23,15:54:58.059040 NOTI libgnutls '3.4.10' initialized.
07/13/23,15:54:58.059329 DBG Core state: 0 -> 1
```

Fuente: elaborado por el autor

Si el HSS se inicia con éxito se muestra el mensaje de confirmación de inicialización de la interfaz S6a en estado DONE (Hecho). En la figura 43 se puede observar el estado ejecutado con éxito.

Figura 43

Inicialización con éxito de HSS

```
07/13/23,15:54:58.073651 NOTI Local endpoints ..... : Default (use all available)
07/13/23,15:54:58.073654 NOTI Local applications ..... : (none)
07/13/23,15:54:58.073657 NOTI Flags : - IP ..... : Enabled
07/13/23,15:54:58.073661 NOTI - IPv6 ..... : DISABLED
07/13/23,15:54:58.073664 NOTI - Relay app ..... : DISABLED
07/13/23,15:54:58.073667 NOTI - TCP ..... : Enabled
07/13/23,15:54:58.073670 NOTI - SCTP ..... : DISABLED
07/13/23,15:54:58.073674 NOTI - Pref. proto .. : TCP
07/13/23,15:54:58.073677 NOTI - TLS method ... : Separate port
07/13/23,15:54:58.073680 NOTI TLS : - Certificate .. : /usr/local/etc/oai/freeDiameter/hss.cert.pem
07/13/23,15:54:58.073683 NOTI - Private key .. : /usr/local/etc/oai/freeDiameter/hss.key.pem
07/13/23,15:54:58.073687 NOTI - CA (trust) ... : /usr/local/etc/oai/freeDiameter/hss.cacert.pem (1 certs)
07/13/23,15:54:58.073690 NOTI - CRL ..... : (none)
07/13/23,15:54:58.073694 NOTI - Priority ..... : (default: 'NORMAL')
07/13/23,15:54:58.073697 NOTI - DH bits ..... : 1024
07/13/23,15:54:58.073700 NOTI Origin-State-Id ..... : 1689281698
07/13/23,15:54:58.073706 NOTI Loaded extensions: '/usr/local/lib/freeDiameter/acl_wl.fdx'[/usr/local/etc/oai/freeDiameter/acl.conf], loaded
07/13/23,15:54:58.073709 NOTI Loaded extensions: '/usr/local/lib/freeDiameter/dict_nas_mipv6.fdx'[[no config file]], loaded
07/13/23,15:54:58.073713 NOTI Loaded extensions: '/usr/local/lib/freeDiameter/dict_s6a.fdx'[[no config file]], loaded
07/13/23,15:54:58.073723 DBG Core state: 1 -> 2
07/13/23,15:54:58.074152 NOTI Local server address(es): 192.168.35.252{---L-}
07/13/23,15:54:58.074178 DBG Core state: 2 -> 3
Initializing s6a layer: DONE
```

Fuente: elaborado por el autor

5.1.1.2. Ejecución de MME

Para iniciar el MME se debe abrir un nuevo terminal para ejecutar el comando respectivo para su funcionamiento, al dar inicio al MME se muestran los parámetros modificados en el script, respecto al dominio del sistema, la lista de GUMMEI en base al MCC, MNC, además de la lista TAI, las interfaces de red y la dirección IP del SGW. En la figura 44 se pueden observar los detalles mencionados.

Figura 44

Ejecución de MME

```

lte@lte-desktop:~$ cd openair-cn; source oaienv; cd scripts; ./run_mme
[sudo] password for lte:
Initializing OAI Logging
Initializing MSC Logs
000005 00000:445640 7F903DC56700 INFO CONFIG nair-cn/src/mme_app/mme_config.c:0701
:17:12 2017 +0200 -----
000006 00000:445650 7F903DC56700 INFO CONFIG nair-cn/src/mme_app/mme_config.c:0702
000007 00000:445655 7F903DC56700 INFO CONFIG nair-cn/src/mme_app/mme_config.c:0703
000008 00000:445659 7F903DC56700 INFO CONFIG nair-cn/src/mme_app/mme_config.c:0704
000009 00000:445660 7F903DC56700 INFO CONFIG nair-cn/src/mme_app/mme_config.c:0705
000010 00000:445664 7F903DC56700 INFO CONFIG nair-cn/src/mme_app/mme_config.c:0706
000011 00000:445669 7F903DC56700 INFO CONFIG nair-cn/src/mme_app/mme_config.c:0707
000012 00000:445677 7F903DC56700 INFO CONFIG nair-cn/src/mme_app/mme_config.c:0708
000013 00000:445681 7F903DC56700 INFO CONFIG nair-cn/src/mme_app/mme_config.c:0709
000014 00000:445686 7F903DC56700 INFO CONFIG nair-cn/src/mme_app/mme_config.c:0710
000015 00000:445689 7F903DC56700 INFO CONFIG nair-cn/src/mme_app/mme_config.c:0711
000016 00000:445693 7F903DC56700 INFO CONFIG nair-cn/src/mme_app/mme_config.c:0712
000017 00000:445697 7F903DC56700 INFO CONFIG nair-cn/src/mme_app/mme_config.c:0713
000018 00000:445701 7F903DC56700 INFO CONFIG nair-cn/src/mme_app/mme_config.c:0714

===== EURECOM MME vBranch: develop Abrev. Hash: 724542d Date: Thu Jun 22 18

Configuration:
- File .....: /usr/local/etc/oai/mme.conf
- Realm .....: OpenAir5G.Alliance
- Run mode .....: TEST
- Max eMBS .....: 2
- Max UES .....: 16
- IMS voice over PS session in S1 .....: false
- Emergency bearer services in S1 mode ..: false
- Location services via epc .....: false
- Extended service request .....: false
- Unauth IMSI support .....: false
- Relative capa .....: 10
- Statistics timer .....: 10 (seconds)

- S1-MME:
  port number .....: 36412
- IP:
  s1-mme iface .....: lo
  s1-mme ip .....: 127.0.0.20
  s11 mme iface ....: lo
  s11 mme port .....: 2123
  s11 mme ip .....: 127.0.0.20
- ITTI:
  queue size .....: 200000 (bytes)
  log file .....: (null)
- SCTP:
  in streams .....: 8
  out streams .....: 8
- GUMMEIS (PLMN|MMEGI|MMEC):
  901.70 |4|1
- TAIs : (mcc,mnc,tac)
  TAI list type one PLMN consecutive TAcs
  901.70:1
- S6A:
  conf file .....: /usr/local/etc/oai/freediameter/mme_fd.conf
- Logging:
  Output .....: CONSOLE
  Output thread safe ..: true
  UDP log level.....: TRACE
  GTPV1-U log level...: INVALID_LOG_LEVEL
  GTPV2-C log level....: TRACE

```

Fuente: elaborado por el autor

Si el MME se inicia con éxito, en el terminal, como se puede observar en la figura 45 se muestran las estadísticas de las conexiones, las mismas que tienen un valor de 0, esto debido a que el UE (Dongle LTE) no se encuentra conectado al eNodeB.

Figura 45

Estadísticas de conexiones del MME

```

000205 00001:496126 7F8FBA7C700 DEBUG S6A te/openair-cn/src/s6a/s6a_peer.c:0136 S6a peer connection attempt 1 / 8
000206 00001:496134 7F8FBA7C700 DEBUG S6A te/openair-cn/src/s6a/s6a_peer.c:0149 Peer hss.OpenAir5G.Alliance is now connected...
000207 00010:462967 7F9034C4B700 DEBUG MME-AP src/mme_app/mme_app_statistics.c:0033 STATISTICS =====
=====
000208 00010:462971 7F9034C4B700 DEBUG MME-AP src/mme_app/mme_app_statistics.c:0034
e last display | Current Status| Added since last display| Removed sinc
000209 00010:462972 7F9034C4B700 DEBUG MME-AP src/mme_app/mme_app_statistics.c:0036 Connected eNBs | 0 | 0 |
000210 00010:462974 7F9034C4B700 DEBUG MME-AP src/mme_app/mme_app_statistics.c:0038 Attached UEs | 0 | 0 |
000211 00010:462976 7F9034C4B700 DEBUG MME-AP src/mme_app/mme_app_statistics.c:0040 Connected UEs | 0 | 0 |
000212 00010:462977 7F9034C4B700 DEBUG MME-AP src/mme_app/mme_app_statistics.c:0042 Default Bearers| 0 | 0 |
000213 00010:462978 7F9034C4B700 DEBUG MME-AP src/mme_app/mme_app_statistics.c:0044 S1-U Bearers | 0 | 0 |

```

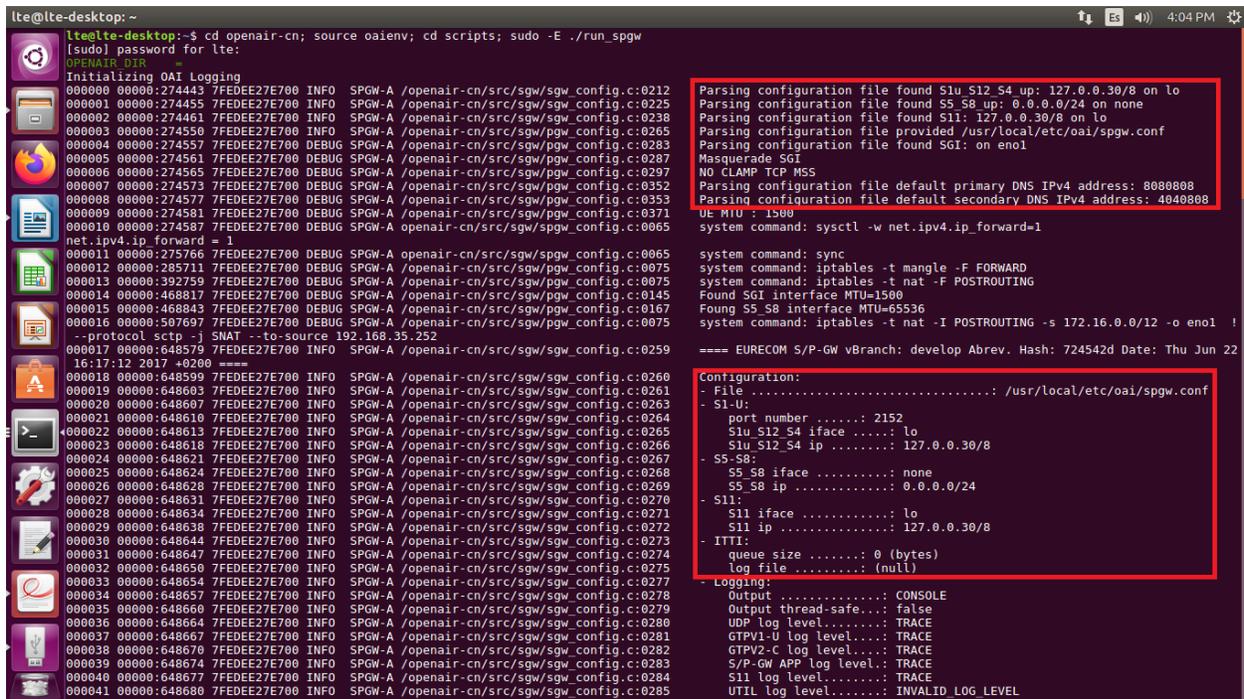
Fuente: elaborado por el autor

5.1.1.3. Ejecución del SPGW

Para proceder con el inicio del SPGW se debe abrir otro terminal para ejecutar el comando respectivo para su funcionamiento, al dar inicio al SPGW se muestran los parámetros modificados en el script, respecto a las interfaces de red S-GW para la comunicación S11 de LTE (conexión MME con S-PGW), además de la interfaz S1-U (conexión con eNodeB), seguido de la interfaz vinculada a S-GW para comunicación S5 o S8. Así como también los parámetros de interfaces de red P-GW con la interfaz para comunicación S5 o S8 y la interfaz para tráfico de internet SGI. Y finalmente, los parámetros de dirección DNS primario y secundario a UE. En la figura 46 se pueden observar los parámetros mencionados.

Figura 46

Ejecución de SPGW



```
lte@lte-desktop: ~
└─$ cd openair-cn; source oaienv; cd scripts; sudo -E ./run_spgw
[sudo] password for lte:
Initializing OAI Logging
000000 00000:274443 7FEDEE27E700 INFO  SPGW-A /openair-cn/src/sgw/sgw_config.c:0212 Parsing configuration file found S1u_S12_S4 up: 127.0.0.30/8 on lo
000001 00000:274455 7FEDEE27E700 INFO  SPGW-A /openair-cn/src/sgw/sgw_config.c:0225 Parsing configuration file found S5_S8 up: 0.0.0.0/24 on none
000002 00000:274461 7FEDEE27E700 INFO  SPGW-A /openair-cn/src/sgw/sgw_config.c:0238 Parsing configuration file found S11: 127.0.0.30/8 on lo
000003 00000:274550 7FEDEE27E700 INFO  SPGW-A /openair-cn/src/sgw/pgw_config.c:0265 Parsing configuration file provided /usr/local/etc/oai/spgw.conf
000004 00000:274557 7FEDEE27E700 DEBUG SPGW-A /openair-cn/src/sgw/pgw_config.c:0283 Parsing configuration file found SGI: on eno1
000005 00000:274561 7FEDEE27E700 DEBUG SPGW-A /openair-cn/src/sgw/pgw_config.c:0287 Masquerade SGI
000006 00000:274565 7FEDEE27E700 DEBUG SPGW-A /openair-cn/src/sgw/pgw_config.c:0297 NO CLAMP TCP MSS
000007 00000:274573 7FEDEE27E700 DEBUG SPGW-A /openair-cn/src/sgw/pgw_config.c:0352 Parsing configuration file default primary DNS IPv4 address: 80808080
000008 00000:274577 7FEDEE27E700 DEBUG SPGW-A /openair-cn/src/sgw/pgw_config.c:0353 Parsing configuration file default secondary DNS IPv4 address: 40408080
000009 00000:274581 7FEDEE27E700 DEBUG SPGW-A /openair-cn/src/sgw/pgw_config.c:0371 UE MTU : 1500
000010 00000:274587 7FEDEE27E700 DEBUG SPGW-A /openair-cn/src/sgw/pgw_config.c:0065 system command: systemctl -w net.ipv4.ip_forward=1
net.ipv4.ip_forward = 1
000011 00000:275766 7FEDEE27E700 DEBUG SPGW-A /openair-cn/src/sgw/spgw_config.c:0065 system command: sync
000012 00000:285711 7FEDEE27E700 DEBUG SPGW-A /openair-cn/src/sgw/pgw_config.c:0075 system command: iptables -t mangle -F FORWARD
000013 00000:392759 7FEDEE27E700 DEBUG SPGW-A /openair-cn/src/sgw/pgw_config.c:0075 system command: iptables -t nat -F POSTROUTING
000014 00000:468817 7FEDEE27E700 DEBUG SPGW-A /openair-cn/src/sgw/pgw_config.c:0145 Found SGI interface MTU=1500
000015 00000:468843 7FEDEE27E700 DEBUG SPGW-A /openair-cn/src/sgw/pgw_config.c:0167 Found S5_S8 interface MTU=65536
000016 00000:507697 7FEDEE27E700 DEBUG SPGW-A /openair-cn/src/sgw/pgw_config.c:0075 system command: iptables -t nat -I POSTROUTING -s 172.16.0.0/12 -o eno1 !
--protocol sctp -j SNAT --to-source 192.168.35.252
000017 00000:648579 7FEDEE27E700 INFO  SPGW-A /openair-cn/src/sgw/sgw_config.c:0259 ===== EURECOM S/P-GW vBranch: develop Abrev. Hash: 724542d Date: Thu Jun 22
16:17:12 2017 40200 =====
000018 00000:648599 7FEDEE27E700 INFO  SPGW-A /openair-cn/src/sgw/sgw_config.c:0260
000019 00000:648603 7FEDEE27E700 INFO  SPGW-A /openair-cn/src/sgw/sgw_config.c:0261
000020 00000:648607 7FEDEE27E700 INFO  SPGW-A /openair-cn/src/sgw/sgw_config.c:0263
000021 00000:648610 7FEDEE27E700 INFO  SPGW-A /openair-cn/src/sgw/sgw_config.c:0264
000022 00000:648613 7FEDEE27E700 INFO  SPGW-A /openair-cn/src/sgw/sgw_config.c:0265
000023 00000:648618 7FEDEE27E700 INFO  SPGW-A /openair-cn/src/sgw/sgw_config.c:0266
000024 00000:648621 7FEDEE27E700 INFO  SPGW-A /openair-cn/src/sgw/sgw_config.c:0267
000025 00000:648624 7FEDEE27E700 INFO  SPGW-A /openair-cn/src/sgw/sgw_config.c:0268
000026 00000:648628 7FEDEE27E700 INFO  SPGW-A /openair-cn/src/sgw/sgw_config.c:0269
000027 00000:648631 7FEDEE27E700 INFO  SPGW-A /openair-cn/src/sgw/sgw_config.c:0270
000028 00000:648634 7FEDEE27E700 INFO  SPGW-A /openair-cn/src/sgw/sgw_config.c:0271
000029 00000:648638 7FEDEE27E700 INFO  SPGW-A /openair-cn/src/sgw/sgw_config.c:0272
000030 00000:648644 7FEDEE27E700 INFO  SPGW-A /openair-cn/src/sgw/sgw_config.c:0273
000031 00000:648647 7FEDEE27E700 INFO  SPGW-A /openair-cn/src/sgw/sgw_config.c:0274
000032 00000:648650 7FEDEE27E700 INFO  SPGW-A /openair-cn/src/sgw/sgw_config.c:0275
000033 00000:648654 7FEDEE27E700 INFO  SPGW-A /openair-cn/src/sgw/sgw_config.c:0277
000034 00000:648657 7FEDEE27E700 INFO  SPGW-A /openair-cn/src/sgw/sgw_config.c:0278
000035 00000:648660 7FEDEE27E700 INFO  SPGW-A /openair-cn/src/sgw/sgw_config.c:0279
000036 00000:648664 7FEDEE27E700 INFO  SPGW-A /openair-cn/src/sgw/sgw_config.c:0280
000037 00000:648667 7FEDEE27E700 INFO  SPGW-A /openair-cn/src/sgw/sgw_config.c:0281
000038 00000:648670 7FEDEE27E700 INFO  SPGW-A /openair-cn/src/sgw/sgw_config.c:0282
000039 00000:648674 7FEDEE27E700 INFO  SPGW-A /openair-cn/src/sgw/sgw_config.c:0283
000040 00000:648677 7FEDEE27E700 INFO  SPGW-A /openair-cn/src/sgw/sgw_config.c:0284
000041 00000:648680 7FEDEE27E700 INFO  SPGW-A /openair-cn/src/sgw/sgw_config.c:0285

Configuration:
- File ..... /usr/local/etc/oai/spgw.conf
- S1-U:
  port number ..... 2152
  S1u_S12_S4 iface ..... lo
  S1u_S12_S4 ip ..... 127.0.0.30/8
- S5-S8:
  S5_S8 iface ..... none
  S5_S8 ip ..... 0.0.0.0/24
- S11:
  S11 iface ..... lo
  S11 ip ..... 127.0.0.30/8
- ITTI:
  queue size ..... 0 (bytes)
  log file ..... (null)
- Logging:
  Output ..... CONSOLE
  Output thread-safe... false
  UDP log level..... TRACE
  GTPV1-U log level... TRACE
  GTPV2-C log level... TRACE
  S/P-GW APP log level... TRACE
  S11 log level..... TRACE
  UTIL log level..... INVALID_LOG_LEVEL
```

```

lte@lte-desktop: ~
000041 00000:648680 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/sgw_config.c:0285 UTIL log level.....: INVALID LOG LEVEL
000042 00000:648683 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/sgw_config.c:0286 MSC log level.....: INVALID LOG LEVEL (MeSsage Chart)
000043 00000:648687 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/sgw_config.c:0287 ITTI log level.....: INVALID LOG LEVEL (InTer-Task Interface)
000044 00000:648690 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/pgw_config.c:0383 ===== EURECOM S/P-GW vBranch: develop Abrev. Hash: 724542d Date: Thu Jun 22

16:17:12 2017 +0200 =====
000045 00000:648694 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/pgw_config.c:0384 Configuration:
000046 00000:648697 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/pgw_config.c:0385 - File .....: /usr/local/etc/oai/sgpw.conf
000047 00000:648701 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/pgw_config.c:0387 - S5-S8:
000048 00000:648704 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/pgw_config.c:0388   S5_S8 iface .....: lo
000049 00000:648708 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/pgw_config.c:0389   S5_S8 ip (read).....: 127.0.0.1
000050 00000:648711 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/pgw_config.c:0390   S5_S8 MTU (read)....: 65536
000051 00000:648714 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/pgw_config.c:0391 - SGI:
000052 00000:648717 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/pgw_config.c:0392   SGI iface .....: eno1
000053 00000:648721 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/pgw_config.c:0393   SGI ip (read).....: 192.168.35.252
000054 00000:648724 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/pgw_config.c:0394   SGI MTU (read).....: 1500
000055 00000:648728 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/pgw_config.c:0396 - MSS clamping: .....: 0
000056 00000:648731 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/pgw_config.c:0397 - Masquerading: .....: 1
000057 00000:648734 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/pgw_config.c:0398 - Push PCO: .....: 0
000058 00000:648743 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/pgw_config.c:0259 ===== EURECOM S/P-GW vBranch: develop Abrev. Hash: 724542d Date: Thu Jun 22

16:17:12 2017 +0200 =====
000059 00000:648746 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/sgw_config.c:0260 Configuration:
000060 00000:648750 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/sgw_config.c:0261 - File .....: /usr/local/etc/oai/sgpw.conf
000061 00000:648753 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/sgw_config.c:0263 - S1-U:
000062 00000:648756 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/sgw_config.c:0264   port number .....: 2152
000063 00000:648760 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/sgw_config.c:0265   S1u_S12_S4 iface ....: lo
000064 00000:648763 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/sgw_config.c:0266   S1u_S12_S4 ip .....: 127.0.0.30/8
000065 00000:648766 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/sgw_config.c:0267 - S5-S8:
000066 00000:648769 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/sgw_config.c:0275   log file .....: null
000067 00000:648773 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/sgw_config.c:0269   S5_S8 iface .....: none
000068 00000:648776 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/sgw_config.c:0270   S5_S8 ip .....: 0.0.0.0/24
000069 00000:648779 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/sgw_config.c:0271 - S11:
000070 00000:648783 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/sgw_config.c:0272   S11 iface .....: lo
000071 00000:648786 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/sgw_config.c:0273   S11 ip .....: 127.0.0.30/8
000072 00000:648789 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/sgw_config.c:0274 - ITTI:
000073 00000:648792 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/sgw_config.c:0275   queue size .....: 0 (bytes)
000074 00000:648797 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/sgw_config.c:0277   log file .....: null
000075 00000:648800 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/sgw_config.c:0278 - Logging:
000076 00000:648804 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/sgw_config.c:0279   Output .....: CONSOLE
000077 00000:648807 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/sgw_config.c:0280   Output thread-safe...: false
000078 00000:648810 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/sgw_config.c:0281   UDP log level.....: TRACE
000079 00000:648813 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/sgw_config.c:0282   GTPV1-U log level....: TRACE
000080 00000:648816 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/sgw_config.c:0283   GTPV2-C log level....: TRACE
000081 00000:648820 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/sgw_config.c:0284   S/P-GW APP log level.: TRACE
000082 00000:648823 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/sgw_config.c:0285   S11 log level.....: TRACE
000083 00000:648826 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/sgw_config.c:0286   UTIL log level.....: INVALID LOG LEVEL
000084 00000:648830 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/sgw_config.c:0287   MSC log level.....: INVALID LOG LEVEL (MeSsage Chart)
000085 00000:648833 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/sgw_config.c:0383   ITTI log level.....: INVALID LOG LEVEL (InTer-Task Interface)
000086 00000:648836 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/pgw_config.c:0384 ===== EURECOM S/P-GW vBranch: develop Abrev. Hash: 724542d Date: Thu Jun 22

16:17:12 2017 +0200 =====
000086 00000:648836 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/pgw_config.c:0384 Configuration:
000087 00000:648839 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/pgw_config.c:0385 - File .....: /usr/local/etc/oai/sgpw.conf
000088 00000:648843 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/pgw_config.c:0387 - S5-S8:
000089 00000:648846 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/pgw_config.c:0388   S5_S8 iface .....: lo
000090 00000:648849 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/pgw_config.c:0389   S5_S8 ip (read).....: 127.0.0.1
000091 00000:648852 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/pgw_config.c:0390   S5_S8 MTU (read)....: 65536
000092 00000:648856 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/pgw_config.c:0391 - SGI:
000093 00000:648859 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/pgw_config.c:0392   SGI iface .....: eno1
000094 00000:648862 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/pgw_config.c:0393   SGI ip (read).....: 192.168.35.252
000095 00000:648866 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/pgw_config.c:0394   SGI MTU (read).....: 1500
000096 00000:648869 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/pgw_config.c:0396 - MSS clamping: .....: 0
000097 00000:648872 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/pgw_config.c:0397 - Masquerading: .....: 1
000098 00000:648875 7FEDEE27E700 INFO SPGW-A /openair-cn/src/sgw/pgw_config.c:0398 - Push PCO: .....: 0

Initializing MSC logs
Initializing MSC logs Done
000099 00000:662080 7FEDEE27E700 DEBUG UDP /src/udp/udp_primitives_server.c:0356 Initializing UDP task interface
000100 00000:662134 7FEDEE27E700 DEBUG UDP /src/udp/udp_primitives_server.c:0364 Initializing UDP task interface: DONE
000101 00000:662156 7FEDEE27E700 DEBUG S11 lte/openair-cn/src/s11/S11_gsw.c:0259 Initializing S11 interface
000102 00000:662280 7FEDEE27E700 INFO GTPV2- 2-c/nwgtpv2c-0.11/src/NwGtpv2c.c:0218

```

Fuente: elaborado por el autor

Si el SPGW se inicia con éxito, en el terminal, como se puede observar en la figura 47 se muestran el mensaje de confirmación de inicialización en estado DONE (Hecho).

Figura 47

Inicialización con éxito de SPGW

```
000133 00000:662388 7FEDE27E700 INFO GTPv2- 2-c/nwgtpv2c-0.11/src/NwGtpv2c.c:0249 *-----*
-----*
000134 00000:663488 7FEDE27E700 DEBUG S11 lte/openair-cn/src/s11/s11_sqw.c:0244 Tx UDP INIT IP addr 127.0.0.30
000135 00000:663504 7FEDE27E700 DEBUG S11 lte/openair-cn/src/s11/s11_sqw.c:0301 Initializing S11 interface: DONE
000136 00000:663509 7FEDE27E700 DEBUG SPGW-A te/openair-cn/src/sgw/sgw_task.c:0148 Initializing SPGW-APP task interface
000137 00000:663512 7FEDE27E700 DEBUG GTPv1- air-cn/src/gtpv1-u/gtpv1u_task.c:0096 Initializing GTPV1U interface
000138 00000:663519 7FEDEA34F700 DEBUG UDP /src/udp/udp_primitives_server.c:0126 Creating new listen socket on address 127.0.0.30 and port 2123
000139 00000:663560 7FEDEA34F700 DEBUG UDP /src/udp/udp_primitives_server.c:0171 Inserting new descriptor for task 6, sd 31
000140 00000:663579 7FEDEA34F700 DEBUG UDP /src/udp/udp_primitives_server.c:0187 Received 1 events
rmmod: ERROR: Module gtp is not currently loaded
000141 00000:670519 7FEDE27E700 NOTIC GTPv1- -cn/src/gtpv1-u/gtp_mod_kernel.c:0081 Using the GTP kernel mode (genl ID is 29)
000142 00000:685406 7FEDE27E700 DEBUG GTPv1- -cn/src/gtpv1-u/gtp_mod_kernel.c:0104 Setting route to reach UE net 172.16.0.0 via gtp0
000143 00000:686142 7FEDE27E700 NOTIC GTPv1- -cn/src/gtpv1-u/gtp_mod_kernel.c:0111 GTP kernel configured
000144 00000:687252 7FEDE27E700 DEBUG GTPv1- air-cn/src/gtpv1-u/gtpv1u_task.c:0124 Initializing GTPV1U interface: DONE
000145 00000:735489 7FEDE27E700 DEBUG SPGW-A te/openair-cn/src/sgw/sgw_task.c:0208 Initializing SPGW-APP task interface: DONE
```

Fuente: elaborado por el autor

5.2. Funcionamiento De Conexión De UE (Dongle LTE) a eNodeB

Una vez iniciado el EPC con éxito, se ejecuta el eNodeB, para ello se usa el archivo de texto o el script de la carpeta `~/openairinterface5g/targets/PROJECTS/GENERIC-LTE-EPC/CONF/`

Se selecciona de los archivos disponibles para bandas LTE, en base al ancho de banda y requisitos de hardware, el cual trabaja en la banda 7 (2600 MHz) y usa 25 Bloques de Recursos Físicos (PRB).

Para iniciar el eNodeB se debe abrir un nuevo terminal y aplicar los comandos respectivos para su funcionamiento, al dar inicio al eNodeB se muestran los parámetros modificados en el script, respecto a la identificación o nombre del eNodeB, el MCC, el MNC, así como también parámetros a nivel físico del tipo para el duplexado FDD, la banda de operación 7, el número máximo de bloques de recursos físicos (PRB), así como también la dirección IP del MME y las interfaces de red para S1-MME junto con la dirección IP del eNodeB en la interfaz S1-MME y la interfaz para la interfaz S1-U junto con la dirección IP del eNodeB para la interfaz. En la figura 48 se pueden observar los detalles mencionados.

Figura 48

Ejecución de eNodeB

```
Terminal File Edit View Search Terminal Help
Lte@lte-desktop:~$ sudo bash
[sudo] password for lte:
root@lte-desktop:~# cd openairinterface5g; source oaienv
root@lte-desktop:~/openairinterface5g# ./targets/bin/lte-softmodem.Rel14 -W -0 ./targets/PROJECTS/GENERIC-LTE-EPC/CONF/enb.band7.tm1.25PRB.lmssdr.conf --rf-confi
g-file ./targets/ARCH/LMSSDR/LimeSDR_above_1p8GHz_lv4.ini
# /dev/cpu_dma_latency set to 0us
Log init done
Enabling OPT for wireshark for local interfacenum component carrier 1
WARNING: setting 'gtpu_log_level' not found in configuration file
WARNING: setting 'gtpu_log_verbosity' not found in configuration file
WARNING: setting 'udp_log_level' not found in configuration file
WARNING: setting 'udp_log_verbosity' not found in configuration file
WARNING: setting 'osa_log_level' not found in configuration file
WARNING: setting 'osa_log_verbosity' not found in configuration file
-----
ENB CONFIG FILE CONTENT LOADED (TBC):
-----
ENB CONFIG for instance 0:
eNB name: eNB_LTEBox:
eNB ID: 3584:
Cell type: CELL_MACRO_ENB:
TAC: 1:
MCC: 901:
MNC: 70:

node function for CC 0: eNodeB_3GPP:
node_timing for CC 0: synch_to_ext_device:
node_synch_ref for CC 0: 0:
eutra_band for CC 0: 7:
downlink_freq for CC 0: 2680000000:
uplink_freq_offset for CC 0: -120000000:

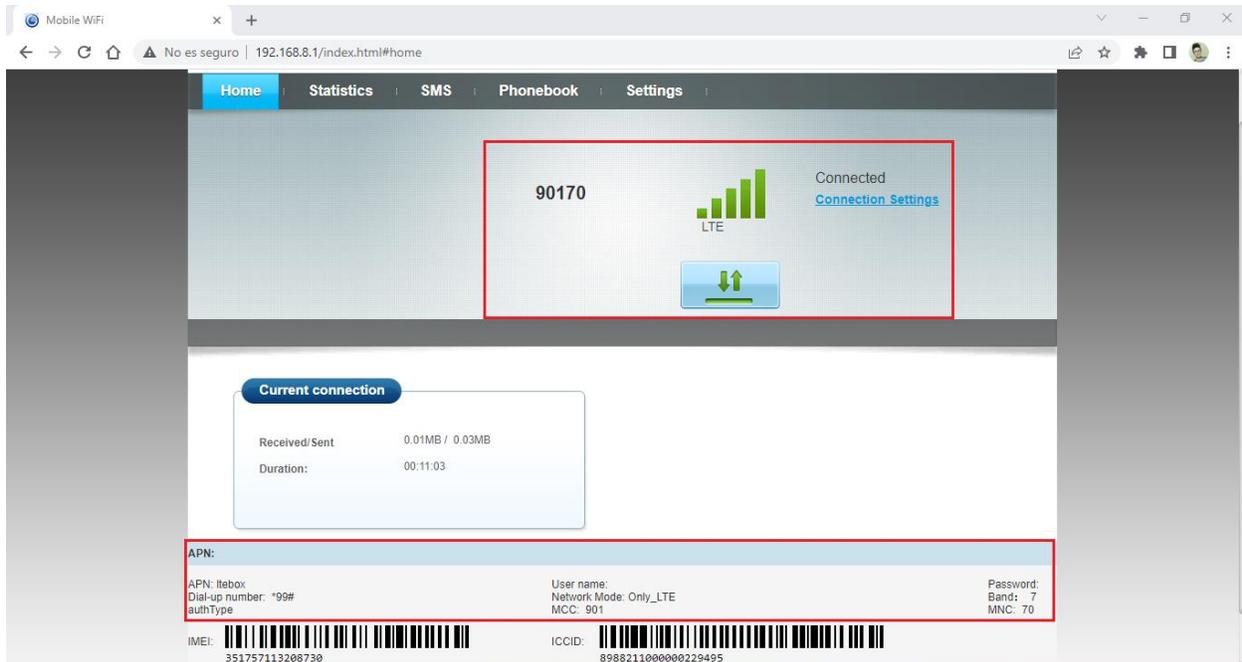
Cell ID for CC 0: 0:
N_RB_DL for CC 0: 25:
nb_antenna_ports for CC 0: 1:
nb_antennas_tx for CC 0: 1:
nb_antennas_rx for CC 0: 1:
rach_numberOfRA_Preambles for CC 0: 15:
rach_preambleGroupAConfig for CC 0: 0:
rach_powerRampingStep for CC 0: 2:
rach_preambleInitialReceivedTargetPower for CC 0: 6:
rach_preambleTransMax for CC 0: 6:
rach_raResponseWindowSize for CC 0: 7:
rach_macContentionResolutionTimer for CC 0: 5:
rach_maxHARQ_Msg3Tx for CC 0: 4:
bcch_modificationPeriodCoeff for CC 0: 0:
pcch_defaultPagingCycle for CC 0: 2:
```

Fuente: elaborado por el autor

Al estar en correcto funcionamiento el EPC y el eNodeB, el UE puede conectarse a la red. Para ello, mediante la configuración realizada en el apartado 4.2.3.1 correspondiente a la configuración del UE (Dongle LTE) con la creación del APN junto con los parámetros definidos, se puede observar en la figura 49 que el Dongle LTE se encuentra conectado a la red LTE 90170, lo cual significa que se puede navegar por Internet.

Figura 49

Conexión a la red LTE del Dongle LTE

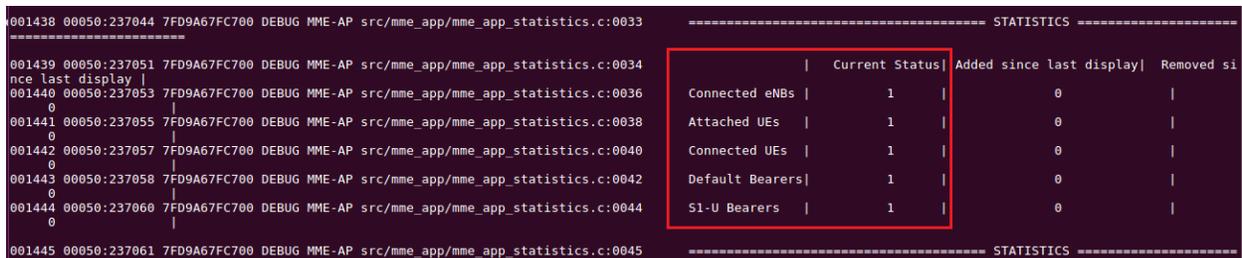


Fuente: elaborado por el autor

Además, en el terminal de ejecución del MME, en las estadísticas los valores realizan un cambio de 0 a 1, confirmando que el UE (Dongle LTE), se ha conectado correctamente al eNodeB, esto se puede evidenciar en la figura 50.

Figura 50

Conexión exitosa del UE (Dongle LTE) al eNodeB



Fuente: elaborado por el autor

Finalmente, una vez verificado el funcionamiento de la red LTE mediante la ejecución del EPC (HSS, MME y SPGW), así como el funcionamiento de la conexión del UE (Dongle LTE) al eNodeB, se procede a establecer las guías de prácticas, las mismas que tienen como objetivo familiarizarse con el sistema, en temas basados en el análisis del protocolo LTE, análisis de tráfico de red y mediciones de red.

5.3. Guía 1: Software Osciloscopio

Esta guía presentará como se mencionó en el apartado 5.1 Funcionamiento de red LTE y 5.2 Funcionamiento de conexión de UE (Dongle LTE) a eNodeB, los comandos para la ejecución del EPC y eNodeB respectivamente junto con las especificaciones para la visualización del software osciloscopio. Para monitoreo de la capa física para eNodeB y UE.

UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS
CARRERA DE INGENIERÍA EN ELECTRÓNICA Y REDES DE
COMUNICACIÓN



TEMA: Software Osciloscopio

AUTOR: ROMEL RICARDO ROSERO

Esta guía se enfoca en la ejecución del EPC y eNodeB para posterior realizar la visualización del software osciloscopio, para realizar un monitoreo de la capa física para eNodeB y UE. Así como también la visualización de las estadísticas de la capa física y MAC sobre el estado de la red mediante mediciones de eNodeB.

Objetivos:

- Familiarizarse con el inicio de la red LTE con los comandos para el funcionamiento
- Comprender mediante la visualización del osciloscopio el monitoreo de la capa física del eNodeB y del UE
- Analizar las estadísticas de la capa física y MAC del estado de la red

5.3.1. Comando para ejecución de EPC: HSS

Para dar inicio a la red LTE, como se mencionó en el apartado 5.1.1 se debe empezar con el EPC en el siguiente orden: HSS, MME y S-PGW para luego continuar con el eNodeB y se deben ejecutar los siguientes comandos usando los scripts de OAI en los subdirectorios correspondientes.

Se da inicio con la ejecución del HSS, para ello se abre un terminal y se aplica el siguiente comando:

```
cd openair-cn; source oaienv; cd scripts;
```

Es decir, se cambia al directorio actual de *openair-cn*, luego se carga y ejecuta los comandos del archivo *oaienv* y finalmente se cambia al directorio *scripts* dentro del directorio de *openair-cn*. Al realizar el establecimiento del directorio, se ejecuta el componente del EPC en este caso el HSS, esto mediante el siguiente comando:

```
./run_hss
```

Si el HSS se inicia con éxito como se muestra en la figura 43 del apartado 5.1.1.1 con el mensaje de confirmación de inicialización de la interfaz S6a en estado DONE (Hecho).

5.3.2. Comando para ejecución de EPC: MME

A continuación, se realiza la ejecución del MME. Para ello se abre un nuevo terminal y de igual manera se cambia al directorio actual de *openair-cn*, se carga y ejecuta los comandos del archivo *oaienv* y se cambia nuevamente al directorio *scripts* dentro del directorio de *openair-cn*. Realizado dicho establecimiento de directorio, se ejecuta el componente del EPC en este caso el MME, mediante el comando:

```
cd openair-cn; source oaienv; cd scripts; ./run_mme
```

Si el MME se inicia con éxito como se muestra en la figura 45 del apartado 5.1.1.2 se muestran las estadísticas de las conexiones.

5.3.3. Comando para ejecución de EPC: S-PGW

Finalmente, para terminar con la ejecución del EPC, se abre otro terminal y con los comandos para el cambio de directorio de *scripts* de *openair-cn*, se añade además el *sudo -E* para tener los permisos como super usuario y con el *-E* para que se mantengan las variables actuales, para concluir se añade el componente de S-PGW y el comando se muestra a continuación:

```
cd openair-cn; source oaienv; cd scripts; sudo -E ./run_spgw
```

Si el SPGW se inicia con éxito, en el terminal, como se puede observar en la figura 47 del apartado 5.1.1.3 se observa el mensaje de confirmación de inicialización en estado DONE (Hecho).

5.3.4. Comando para ejecución del eNodeB

Para la ejecución del eNodeB, es necesario abrir un nuevo terminal y ejecutar los siguientes comandos para el componente software del eNodeB:

```
sudo bash  
  
cd openairinterface5g; source oaienv  
  
./targets/bin/lte-softmodem.Rel14 -d -O <eNB config file> --rf-config-file  
./targets/ARCH/LMSSDR/LimeSDR_above_1p8GHz_1v4.in
```

Al ejecutable *lte-softmodem* se puede detallar las siguientes opciones:

- **-O:** especifica la ruta hacia el fichero de configuración del eNodeB
- **-d:** permite habilitar el osciloscopio de software con interfaces gráficas interactivas para monitorizar el nivel físico
- **<eNB config file>:** especifica el archivo de configuración de eNodeB

En este caso se usará el archivo de configuración de eNodeB

(enb.band7.tm1.25PRB.lmssdr.conf) en la banda 7 (2600 MHz) con 25 Bloques de Recursos Físicos (PRB). El LimeSDR actúa como interfaz de RF. El comando final se muestra a continuación:

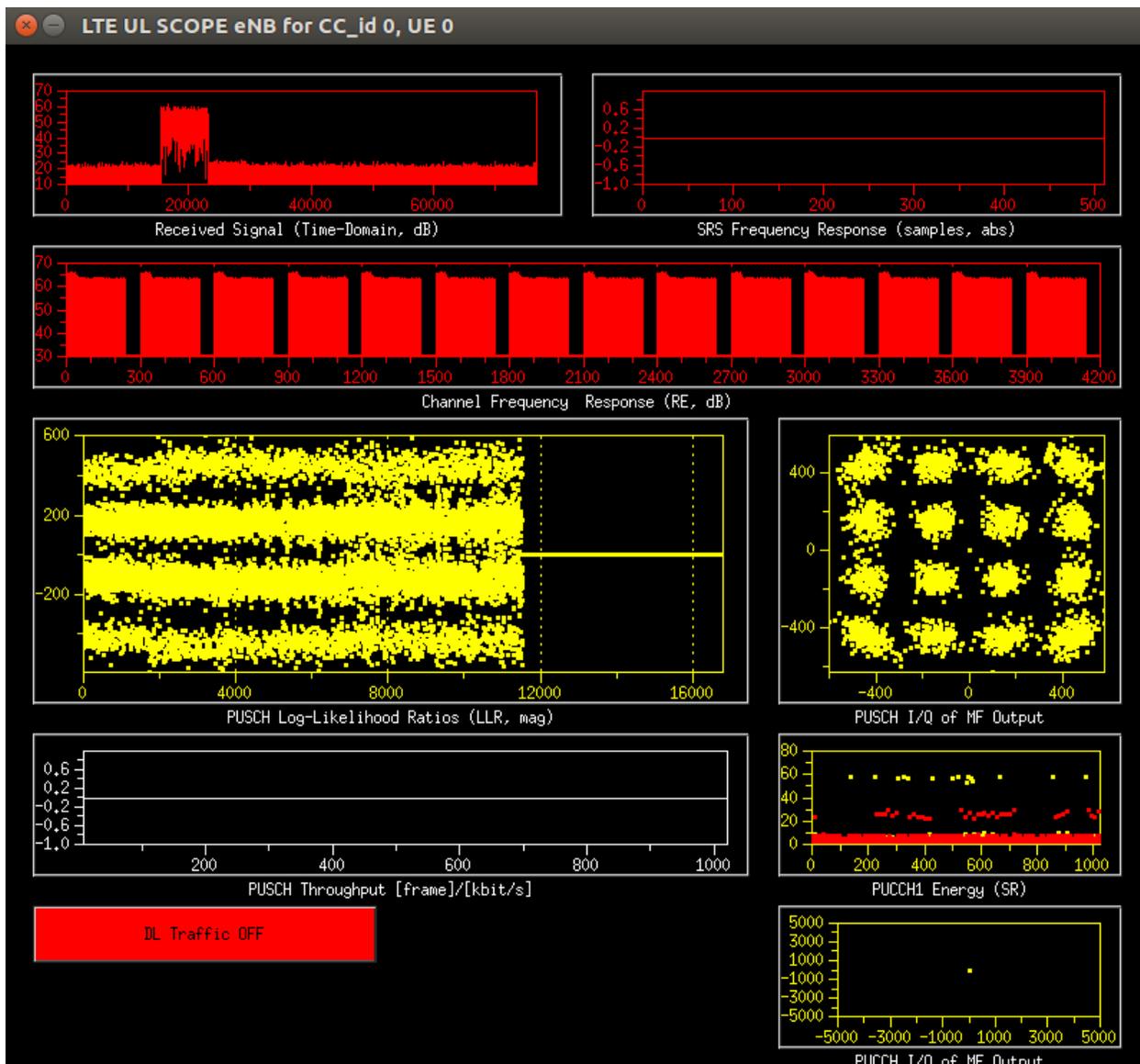
```
./targets/bin/lte-softmodem.Rel14 -d -O ./targets/PROJECTS/GENERIC-  
LTE-EPC/CONF/enb.band7.tm1.25PRB.lmssdr.conf --rf-config-file  
./targets/ARCH/LMSSDR/LimeSDR_above_1p8GHz_1v4.in
```

5.3.5. Verificación de software osciloscopio

Una vez ejecutado mediante los comandos tanto el EPC: HSS, MME y S-PGW, como también el eNodeB, se muestran las ventanas del software osciloscopio. En la figura 51 mostrada a continuación, se puede observar la gráfica del software osciloscopio generado.

Figura 51

Gráfica generada software osciloscopio



Fuente: elaborado por el autor

En la gráfica se puede realizar un monitoreo de la interfaz física en tiempo real en base a los parámetros de:

- *Received Signal - Señal Recibida (dominio del tiempo, dB):* muestra la variación de la señal recibida en el dominio del tiempo y la amplitud en decibelios (dB).

- *Sounding Reference Signals (SRS) Frequency Response – Respuesta de Frecuencia Señal de Referencia de Sondeo (muestras, absoluto):* da a conocer como el Sistema de Referencia de Señal responde a frecuencias en valores de amplitud absoluta, evalúa la calidad de señal de referencia y la eficiencia del enlace Downlink – Descendente (DL)
- *Channel Frequency Response - Respuesta de Frecuencia del Canal (RE, dB):* muestra la respuesta en frecuencia del canal de comunicación en dB, en función de los Recursos Espectrales (RE).
- *Physical Uplink Shared Channel Log-Likelihood Ratios (PUSCH LLR) – Canal Compartido de Enlace Ascendente Físico Coeficientes de Verosimilitud Logarítmica (LLR, magnitud):* muestra la calidad de la señal en el enlace Uplink – Ascendente (UL) en el canal PUSCH.
- *PUSCH I/Q de MF (Matched Filter) Output:* muestra la señal recibida en el enlace ascendente mediante los componentes en fase (I) y cuadratura (Q) de la señal después del filtro de recepción (MF).
- *PUSCH Throughput (Rendimiento) - (frame/kbits):* tasa de transferencia efectiva de datos en el canal PUSCH
- *Physical Uplink Control Channel Energy (PUCCHI Energy) – Energía del Canal de Control de Enlace Ascendente Físico (SR):* potencia o energía de la señal en el canal físico PUCCH medido en (Signal to Noise Ratio – Relación Señal Ruido)
- *PUCCH I/Q de ME (Minimum Equalizer) Output:* representación de la señal recibida en el canal PUCCH, en base a los componentes de fase y cuadratura después del Ecuador Mínimo en el receptor (ME)

En resumen, la figura 51 permite mediante el Soft Scope de OAI realizar un monitoreo del eNodeB como del UE para el enlace ascendente (Uplink), esta herramienta muestra parámetros importantes basados en la potencia de la señal recibida en el dominio del tiempo (dB), la respuesta de impulso del canal, la respuesta frecuencial del canal (dB), PUSHC Log-Likelihood, el rendimiento del PUSCH (Throughput) y los componentes de fase (I) y cuadratura (Q) de la constelación del PUSCH observando que la presentada en la figura corresponde a la constelación 16-QAM.

5.3.6. Verificación de estadísticas de la capa física y MAC

De igual manera al ejecutar mediante los comandos para el EPC y eNodeB, se muestran las ventanas de estadísticas de la capa física y MAC. En la figura 52, se puede observar la gráfica de las estadísticas generadas.

Figura 52

Gráfica generada de estadísticas

```

l2 stats
Reset Stats
eNB 0 CC 0 Frame 84: Active UEs 1, Available PRBs 25, nCCE 0, Scheduling decisions 0, Missed Deadlines 0
BCH , NB_TX_MAC = 89653, transmitted bytes (TTI 15, total 1613760) MCS (TTI 2)
DLSCH bitrate (TTI 0, avg 0) kbps, Transmitted bytes (TTI 0, total 0), Transmitted PDU (TTI 0, total 0)
ULSCH bitrate (TTI 1, avg 72729) kbps, Received bytes (TTI 161, total 7727501), Received PDU (TTI 0, total 56120)
[MAC] UE 0 (DLSCH), status RRC_CONNECTED, RNTI d8c3 : CQI 15, MCS1 28, MCS2 0, RB (tx 2, retx 18, total 155), nCCE (tx 0, retx 0)
[MAC] DLSCH bitrate (TTI 0, avg 62), Transmitted bytes (TTI 4, total 6895), Total Transmitted PDU 32, Overhead (TTI 2, total 4366, avg 41)
[MAC] UE 0 (ULSCH), Status RRC_CONNECTED, Failure timer 0, RNTI d8c3 : rx power (normalized -65, target -85), MCS (pre 20, post 20), RB (rx 0, retx 0, total 170520),
[MAC] ULSCH bitrate (TTI 1, avg 72729), received bytes (total 7727501), Total received PDU 56120, Total errors 0
[MAC] Received PHR PH = 40 (db)
[MAC] Received BSR LCGID[0][1][2][3] = 0 0 0 0
[RLC] DCCH Mode AM, NB_SDU_TO_TX = 3 (bytes 61) NB_SDU_TO_TX_DISC 0 (bytes 0)
[RLC] DCCH Mode AM, NB_TX_DATA = 12 (bytes 146) NB_TX_CONTROL 4 (bytes 8) NB_TX_RETX 0 (bytes 0) NB_TX_RETX_BY_STATUS = 0 (bytes 0)
[RLC] DCCH Mode AM, NB_RX_DATA = 4 (bytes 85) NB_RX_CONTROL 3 (bytes 6) NB_RX_DUPL 0 (bytes 0) NB_RX_DROP = 0 (bytes 0) NB_RX_OUT_OF_WINDOW = 0 (bytes 0)
[RLC] DCCH Mode AM, RX_REORDERING_TIMEOUT = 0 RX_POLL_RET_TIMEOUT 0 RX_PROHIBIT_TIME_OUT 0
[RLC] DCCH Mode AM, NB_SDU_TO_RX = 4 (bytes 77)
[RLC] DTCH Mode UM, NB_SDU_TO_TX = 1634882592 (bytes 1768780654) NB_SDU_TO_TX_DISC 1684370548 (bytes 1954112032)
[RLC] DTCH Mode UM, NB_TX_DATA = 673215333 (bytes 541676628) NB_TX_CONTROL 0 (bytes 0) NB_TX_RETX 0 (bytes 0) NB_TX_RETX_BY_STATUS = 0 (bytes 0)
[RLC] DTCH Mode UM, NB_RX_DATA = 1768780654 (bytes 1684370548) NB_RX_CONTROL 3 (bytes 6) NB_RX_DUPL 740896800 (bytes 1634882592) NB_RX_DROP = 143054032
[RLC] DTCH Mode UM, RX_REORDERING_TIMEOUT = 0 RX_POLL_RET_TIMEOUT 0 RX_PROHIBIT_TIME_OUT 0
[RLC] DTCH Mode UM, NB_SDU_TO_RX = 1948269520 (bytes 1818326127)

```

```

stats
Reset Stats
eNB 0/1 Frame 94: RX Gain 103 dB, I0 -91 dBm (12,0) dB
PRB 10 (0,0,0,0): -117 -105 -100 -105 -102 -105 -105 -102 -102 -102 -102 -100 -100 -105 -105 -99 -102 -102 -100 -100 -102 -102 -100 -100
PERFORMANCE PARAMETERS
Total DL-SCH 1 kbits / 85 Frames Total DL-SCH throughput 0 kbps Total DL-SCH trans 11 / 85 frames
UE 0 (d8c3) Power: (65,0) dB, Po_PUSCH: (-65,0) dBm, Po_PUCCH (-52/-96) dBm, Po_PUCCH1 (-103,-66) dBm, PUCCH1 Thres -105 dBm
DL MCS 0, UL MCS 20, UL RB 3, delta_TF 1108, Wideband CQI: (0,0) dB
DL TM 1, DL_CQI 15, DL_pmi_single 0 Timing advance 30 samples (7.16Ts), update 12 Mode = PUSCH(3) UE_id_mac = 0, RRC status = 4
SR received/total: 3475/109102 (diff 105627)
DL Subband CQI: 15 15 15 14 15 15 14
UL-SCH errors/attempts per harq (per round):
  harq 0: 5/3683 (fer 0) (23/3683, 11/23, 6/11, 5/6)   harq 1: 3/3595 (fer 0) (4/3595, 3/4, 3/3, 3/3)
  harq 2: 3/3937 (fer 0) (14/3937, 5/14, 4/5, 3/4)   harq 3: 6/3590 (fer 0) (14/3590, 12/14, 9/12, 6/9)
  harq 4: 7/9672 (fer 0) (22/9672, 14/22, 8/14, 7/8) harq 5: 4/3578 (fer 0) (8/3578, 4/8, 4/4, 4/4)
  harq 6: 4/9912 (fer 0) (15/9912, 6/15, 5/6, 4/5) harq 7: 3/3610 (fer 0) (13/3610, 8/13, 6/8, 3/6)
UL-SCH errors/attempts total 35/53537 (113/53537, 63/113, 45/63, 35/45)
DL-SCH errors/attempts per harq (per round):
  harq 0: 0/4 (4/0/4, 0/0/0, 0/0/0, 0/0/0)   harq 1: 0/1 (1/0/1, 0/0/0, 0/0/0, 0/0/0)
  harq 2: 0/2 (2/0/2, 0/0/0, 0/0/0, 0/0/0)   harq 3: 0/1 (1/0/1, 0/0/0, 0/0/0, 0/0/0)
  harq 4: 0/1 (1/0/1, 0/0/0, 0/0/0, 0/0/0)   harq 5: 0/0 (0/0/0, 0/0/0, 0/0/0, 0/0/0)
  harq 6: 0/1 (1/0/1, 0/0/0, 0/0/0, 0/0/0)   harq 7: 0/1 (1/0/1, 0/0/0, 0/0/0, 0/0/0)
DL-SCH errors/attempts total 0/11 (0/11, 0/0, 0/0, 0/0)
DL-SCH total bits from MAC: 0kbit DL-SCH total bits ack'ed: 1kbit DL-SCH Average throughput (100 frames): 0kbit

```

Fuente: elaborado por el autor

Las estadísticas principales se detallan a continuación:

- *[MAC] UE 0 (DL-SCH), status RRC connected:* mensaje de registro en la capa MAC del UE conectado a la estación base en el enlace descendente
- *Downlink Shared Channel – Canal Compartido de Enlace Descendente (DL-SCH)*
 - Tasa de bits transmitidos correctamente
 - Rendimiento (Throughput)
- *UE*
 - Potencia PUSCH
 - Potencia PUCCH
 - Potencia PUCCH1
- *Modulation and Coding Scheme – Esquema de Modulación y Codificación (MCS)*
- *Procesos Hybrid Automatic Repeat Request – Petición de Repetición Automática Híbrida (HARQ)*

5.4. Guía 2: Mediciones De Red

La presente guía muestra el funcionamiento de la red LTE para realizar mediciones de red usando programas de prueba de velocidad con el fin de obtener valores de las velocidades de carga y descarga usando los archivos de configuración de 25, 50 y 100 PRB.

UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS
CARRERA DE INGENIERÍA EN ELECTRÓNICA Y REDES DE
COMUNICACIÓN



TEMA: Mediciones de Red

AUTOR: ROMEL RICARDO ROSERO

Esta guía se enfoca en la ejecución del EPC y eNodeB con los archivos de configuración en banda 7 con 25, 50 y 100 PRB con el fin de realizar mediciones de red en la velocidad de carga y descarga. Y obtener un registro del esquema de codificación y modulación (MCS) en el enlace ascendente y descendente.

Objetivos

- Iniciar la red LTE con los comandos respectivos y con los archivos de configuración de banda 7 para 25, 50 y 100 PRB
- Realizar las pruebas de velocidad en carga, descarga y registrar los valores de (MCS)
- Obtener los índices de (TBS), comparar la velocidad medida, estimada y su diferencia

5.4.1. Comandos para ejecución de EPC: HSS, MME y S-PGW

Se procede a aplicar los comandos mencionados en la Guía 1: Software Osciloscopio para iniciar la red LTE en el orden establecido y con las consideraciones detalladas, empezando con el EPC y luego el eNodeB. Los comandos respectivos se detallan a continuación:

```
cd openair-cn; source oaienv; cd scripts; ./run_hss
```

```
cd openair-cn; source oaienv; cd scripts; ./run_mme
```

```
cd openair-cn; source oaienv; cd scripts; sudo -E ./run_spgw
```

5.4.2. Comandos para ejecución de eNodeB: 25, 50 y 100 PRB

Para poder realizar las mediciones de red se ejecuta primero el archivo de configuración de configuración de eNodeB (enb.band7.tm1.25PRB.lmssdr.conf) en la banda 7 (2600 MHz) con 25 Bloques de Recursos Físicos (PRB)

```
sudo bash
```

```
cd openairinterface5g; source oaienv
```

```
./targets/bin/lte-softmodem.Rel14 -d -O ./targets/PROJECTS/GENERIC-LTE-EPC/CONF/enb.band7.tm1.25PRB.lmssdr.conf --rf-config-file
```

Para detener la red LTE, se puede cerrar cada una de las terminales de la MiniPC una vez cerradas se procede con el reinicio de la misma.

Una vez realizado el reinicio, se ejecutan los comandos del EPC y luego del eNodeB con la variación siguiente, debido a que se debe usar el archivo de configuración de eNodeB (enb.band7.tm1.50PRB.lmssdr.conf) en la banda 7 (2600 MHz) con 50 Bloques de Recursos Físicos (PRB).

```
./targets/bin/lte-softmodem.Rel14 -d -O ./targets/PROJECTS/GENERIC-  
LTE-EPC/CONF/enb.band7.tm1.50PRB.lmssdr.conf --rf-config-file  
./targets/ARCH/LMSSDR/LimeSDR_above_1p8GHz_1v4.in
```

Repetir el proceso de cierre de las ventanas del terminal y a su vez de reinicio de la Mini PC, se inicia la red LTE, con la ejecución de los comandos del EPC y del eNodeB, en este caso con la modificación ya que se debe usar el archivo de configuración de eNodeB (enb.band7.tm1.100PRB.lmssdr.conf) en la banda 7 (2600 MHz) con 100 Bloques de Recursos Físicos (PRB).

```
./targets/bin/lte-softmodem.Rel14 -d -O ./targets/PROJECTS/GENERIC-  
LTE-EPC/CONF/enb.band7.tm1.100PRB.lmssdr.conf --rf-config-file  
./targets/ARCH/LMSSDR/LimeSDR_above_1p8GHz_1v4.in
```

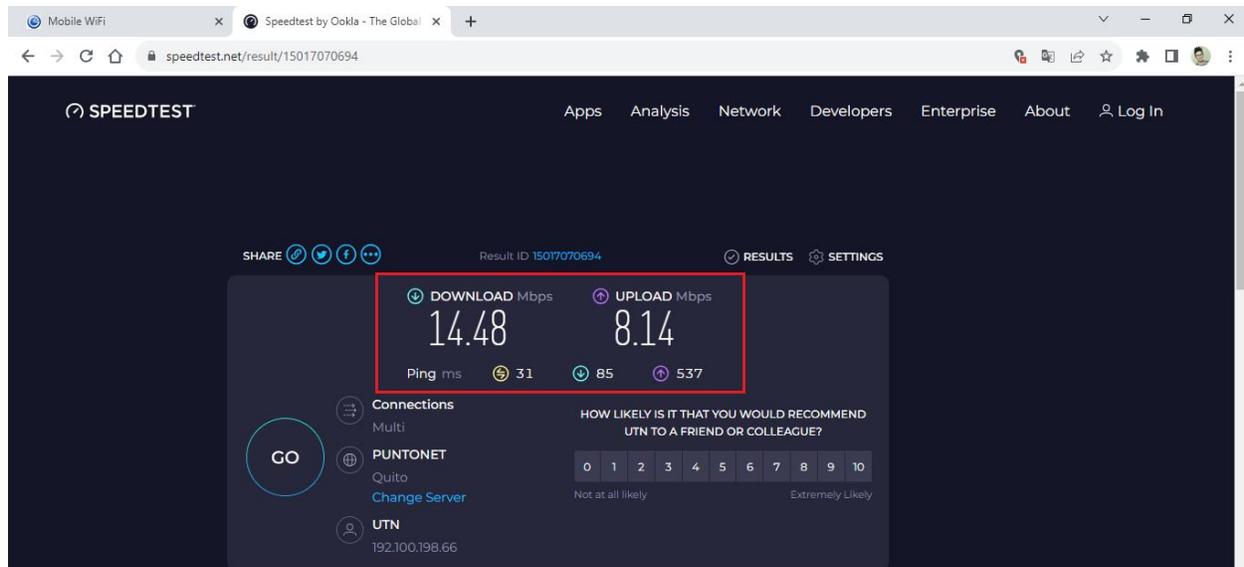
5.4.3. Pruebas de Velocidad

En la máquina donde está conectado y funcionando el Dongle LTE, es donde se realizan las pruebas de velocidad, existen varios programas en línea en internet, en este caso se usa el siguiente <https://www.speedtest.net/>. Se deben realizar 5 pruebas para cada archivo de configuración, para registrar los valores tanto en carga como en descarga para obtener un

promedio de la velocidad. En la figura 53 mostrada a continuación se puede observar un ejemplo de la prueba de velocidad con la ejecución del archivo de configuración de eNodeB con 25 PRB.

Figura 53

Prueba de velocidad con 25 PRB



Fuente: elaborado por el autor

A continuación, en la tabla 15 se detallan los valores obtenidos de velocidad de descarga en base a los archivos de 25, 50 y 100 PRB respectivamente.

Tabla 15

Velocidad de descarga

PRB	25	50	100
Test 1 (Mbps)	13,17	26,59	32,63
Test 2 (Mbps)	14,48	26,63	32,40
Test 3 (Mbps)	14,39	26,65	32,55

Test 4 (Mbps)	14,30	26,27	32,68
Test 5 (Mbps)	14,44	26,61	33,28
Promedio (Mbps)	14,15	26,55	32,70

Fuente: Elaborado por el autor

De igual manera, en la tabla 16 se detallan los valores obtenidos de velocidad de carga en base a los archivos de 25, 50 y 100 PRB respectivamente.

Tabla 16

Velocidad de carga

PRB	25	50	100
Test 1 (Mbps)	8,09	18,35	17,86
Test 2 (Mbps)	8,14	18,31	17,28
Test 3 (Mbps)	8,13	18,30	18,17
Test 4 (Mbps)	7,97	18,32	17,40
Test 5 (Mbps)	7,95	18,35	17,66
Promedio (Mbps)	8,05	18,32	17,67

Fuente: Elaborado por el autor

5.4.4. Índice de Esquema de Codificación y Modulación (MCS)

En base a la ventana de estadísticas de la figura 52 del apartado 5.3.6 de la Guía 1, se registran los valores de MCS para la descarga (DL MCS) y para la carga (UL MCS). Estos valores servirán para obtener los índices Transport Block Size – Tamaño de Bloque de Transporte (TBS). En la tabla 17, se puede observar los valores máximos obtenidos de MCS.

Tabla 17

Valores máximos de MCS

PRB	25	50	100
DL MCS (Máximo)	27	27	28
UL MCS (Máximo)	20	20	20

Fuente: Elaborado por el autor

5.4.5. Comparación de velocidad medida y estimada

Para realizar la comparación de la velocidad medida y estimada, es importante señalar que el UE Dongle LTE (Huawei E3372) según la tabla de características detallado en el apartado 3.3.1.3 se menciona que es un dispositivo LTE FDD Categoría 4 y su velocidad de Downlink (Descarga) es de 150 Mbps y de Uplink (Carga) es de 50 Mbps, destacando que esos valores se consiguen con un ancho de banda de 20 MHz, con modulación 64-QAM y MIMO 2x2.

En este caso, al usar transmisión 1x1. Se basa en el índice de esquema de modulación y codificación (MCS) y el índice Transport Block Size – Tamaño de Bloque de Transporte (TBS) como se observa en la figura 54 mostrada a continuación, el índice MCS asignado es el más alto correspondiente a 28 MCS con 64-QAM como codificación mínima, por lo tanto, el índice TBS asignado es de 26.

Figura 54

Modulation and TBS index table for PDSCH

Table 7.1.7.1-1: Modulation and TBS index table for PDSCH

MCS Index I_{MCS}	Modulation Order Q_m	Modulation Order Q'_m	TBS Index I_{TBS}
0	2	2	0
1	2	2	1
2	2	2	2
3	2	2	3
4	2	2	4
5	2	4	5
6	2	4	6
7	2	4	7
8	2	4	8
9	2	4	9
10	4	6	9
11	4	6	10
12	4	6	11
13	4	6	12
14	4	6	13
15	4	6	14
16	4	6	15
17	6	6	15
18	6	6	16
19	6	6	17
20	6	6	18
21	6	6	19
22	6	6	20
23	6	6	21
24	6	6	22
25	6	6	23
26	6	6	24
27	6	6	25
28	6	6	26/26A
29	2	2	reserved
30	4	4	
31	6	6	

Nota: La figura muestra la tabla de modulación de índices de modulación y TBS para PDSCH.

Tomado de *Modulation order and redundancy version determination* (p.79), por (3GPP TS 36.213, 2017), LTE; Evolved Universal Terrestrial Radio Access (E-UTRA); Physical layer procedures. European Telecommunications Standards Institute.

Por lo que, para obtener las velocidades máximas de descarga, se usa la tabla 7.1.7.2.1-1: Transport block size table (dimension 38×110) de (3GPP TS 36.213, 2017) página 83-87 y se analiza el Tamaño de Bloque de Transporte (TBS) establecido anteriormente en 26 y de acuerdo al número de PRB tanto de 25, 50 y 100, se obtienen las velocidades mencionadas.

En la tabla 18 se observan los valores estimados, los valores medidos en promedio obtenidos de la tabla 15 y así determinar un valor de diferencia porcentual.

Tabla 18

Velocidad de descarga estimado, medido y diferencia

PRB	25	50	100
Medido (Mbps)	14,15	26,55	32,70
Estimado (Mbps)	18,33	36,69	75,37
Diferencia (%)	22,80	27,63	56,61

Fuente: Elaborado por el autor

De igual manera, para la velocidad de carga, como se observa en la figura 55 mostrada a continuación, el índice MCS asignado es el más alto correspondiente a 20 MCS sin 64-QAM como codificación mínima, por lo tanto, el índice TBS asignado es de 19.

Figura 55

Modulation, TBS index and redundancy version table for PUSCH

Table 8.6.1-1: Modulation, TBS index and redundancy version table for PUSCH

MCS Index I_{MCS}	Modulation Order Q_m	TBS Index I_{TBS}	Redundancy Version r_{vdx}
0	2	0	0
1	2	1	0
2	2	2	0
3	2	3	0
4	2	4	0
5	2	5	0
6	2	6	0
7	2	7	0
8	2	8	0
9	2	9	0
10	2	10	0
11	4	10	0
12	4	11	0
13	4	12	0
14	4	13	0
15	4	14	0
16	4	15	0
17	4	16	0
18	4	17	0
19	4	18	0
20	4	19	0
21	6	19	0
22	6	20	0
23	6	21	0
24	6	22	0
25	6	23	0
26	6	24	0
27	6	25	0
28	6	26	0
29	reserved		1
30			2
31			3

Nota: La figura muestra la tabla de versiones de modulación, índice TBS y redundancia para PUSCH. Tomado de *Modulation order and redundancy version determination* (p.267), por (3GPP TS 36.213, 2017), LTE; Evolved Universal Terrestrial Radio Access (E-UTRA); Physical layer procedures. European Telecommunications Standards Institute.

Para obtener las velocidades máximas de carga, se usa igualmente la tabla 7.1.7.2.1-1: Transport block size table (dimension 38×110) de (3GPP TS 36.213, 2017) página 83-87 y en base al índice TBS establecido en 19 y de acuerdo al número de PRB tanto de 25, 50 y 100, se obtienen las velocidades mencionadas.

En la tabla 19 se observan los valores estimados, los valores medidos en promedio obtenidos de la tabla 16 y el valor de diferencia porcentual.

Tabla 19

Velocidad de carga estimado, medido y diferencia

PRB	25	50	100
Medido (Mbps)	8,05	18,32	17,67
Estimado (Mbps)	10,68	21,38	43,81
Diferencia (%)	24,62	14,31	59,66

Fuente: Elaborado por el autor

5.5. Guía 3: Análisis Protocolo LTE

La presente guía tiene como finalidad la realización del filtrado y captura del tráfico de la red LTE con la herramienta Wireshark, para su posterior análisis del protocolo de conexión del UE.

UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS
CARRERA DE INGENIERÍA EN ELECTRÓNICA Y REDES DE
COMUNICACIÓN



TEMA: Análisis protocolo LTE

AUTOR: ROMEL RICARDO ROSERO

Esta guía se enfoca en la ejecución del EPC y eNodeB, con el fin de analizar los parámetros del protocolo LTE con previa la instalación de la herramienta Wireshark y el comando respectivo para la habilitación de los mensajes de nivel L2 para el análisis del tráfico

Objetivos

- Instalar y configurar Wireshark para el análisis del tráfico
- Iniciar la red LTE, junto con la habilitación de los mensajes L2 para el filtrado y captura del tráfico
- Analizar los parámetros del protocolo de conexión del UE

5.5.1. *Instalación y configuración de Wireshark*

Para poder capturar y analizar el tráfico de red, es necesario instalar Wireshark. Por lo que se ejecutan los siguientes comandos en el terminal del sistema. Estos comandos servirán para la instalación de las dependencias necesarias para el funcionamiento automático.

```
sudo add-apt-repository ppa:wireshark-dev/stable
```

```
sudo apt-get update
```

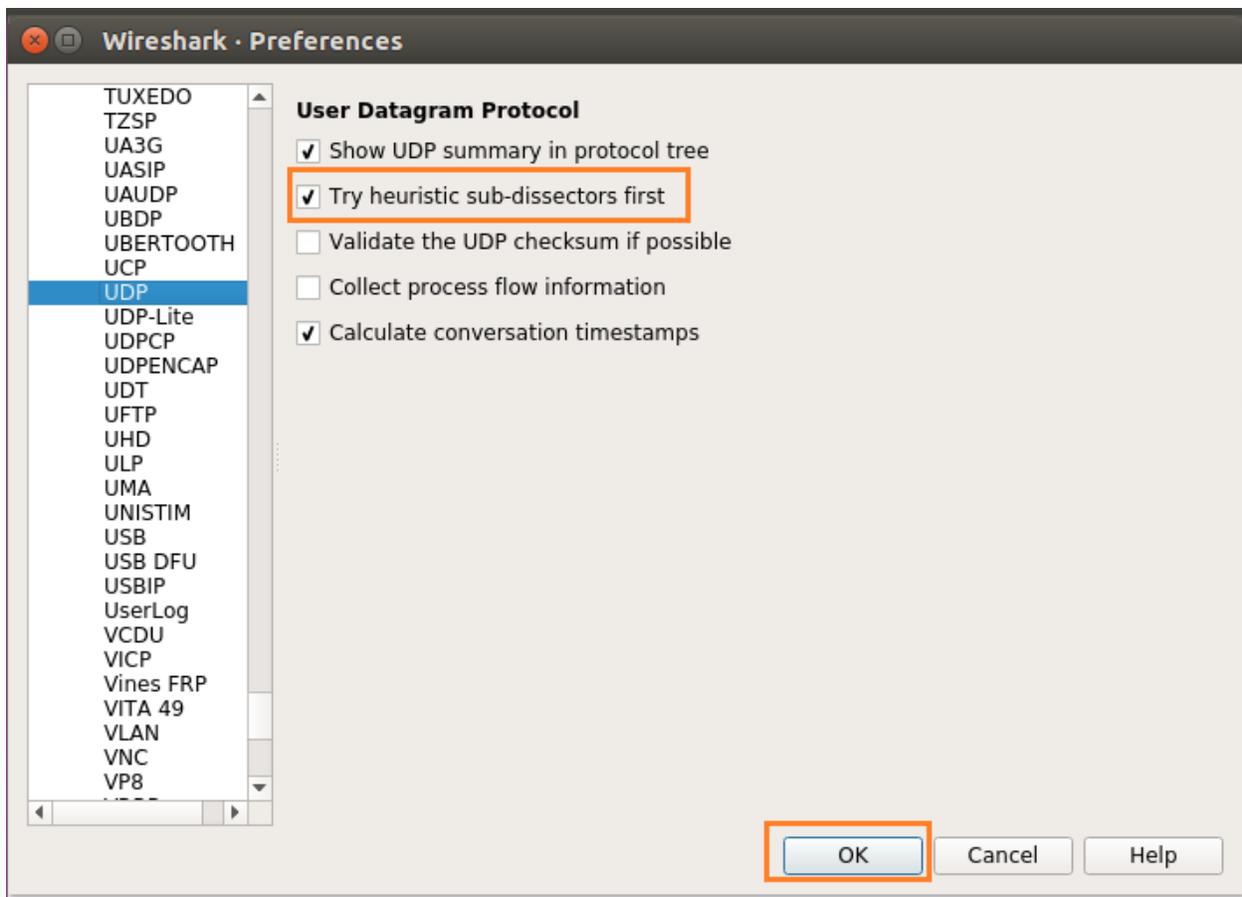
```
sudo apt-get install wireshark
```

Por esta razón, (OpenAirInterface5G, 2016) menciona que OAI lleva a cabo la captura de tráfico de la interfaz MAC de LTE con sockets UDP. Pero antes de poder capturar tráfico es necesario configurar Wireshark para que se pueda comprender la información de estos protocolos.

Primero es necesario configurar, el protocolo UDP para permitir el análisis heurístico de los dissectores de los protocolos LTE. Se debe abrir el menú *Edit* → *Preferences* y en la pestaña *Protocols*, se selecciona el protocolo UDP y se habilita la opción *Try heuristic sub-dissectors first*. En la figura 56 se puede observar la habilitación de la opción de probar primero los sub-dissectores heurísticos.

Figura 56

Habilitación de subdirectores heurísticos en Wireshark

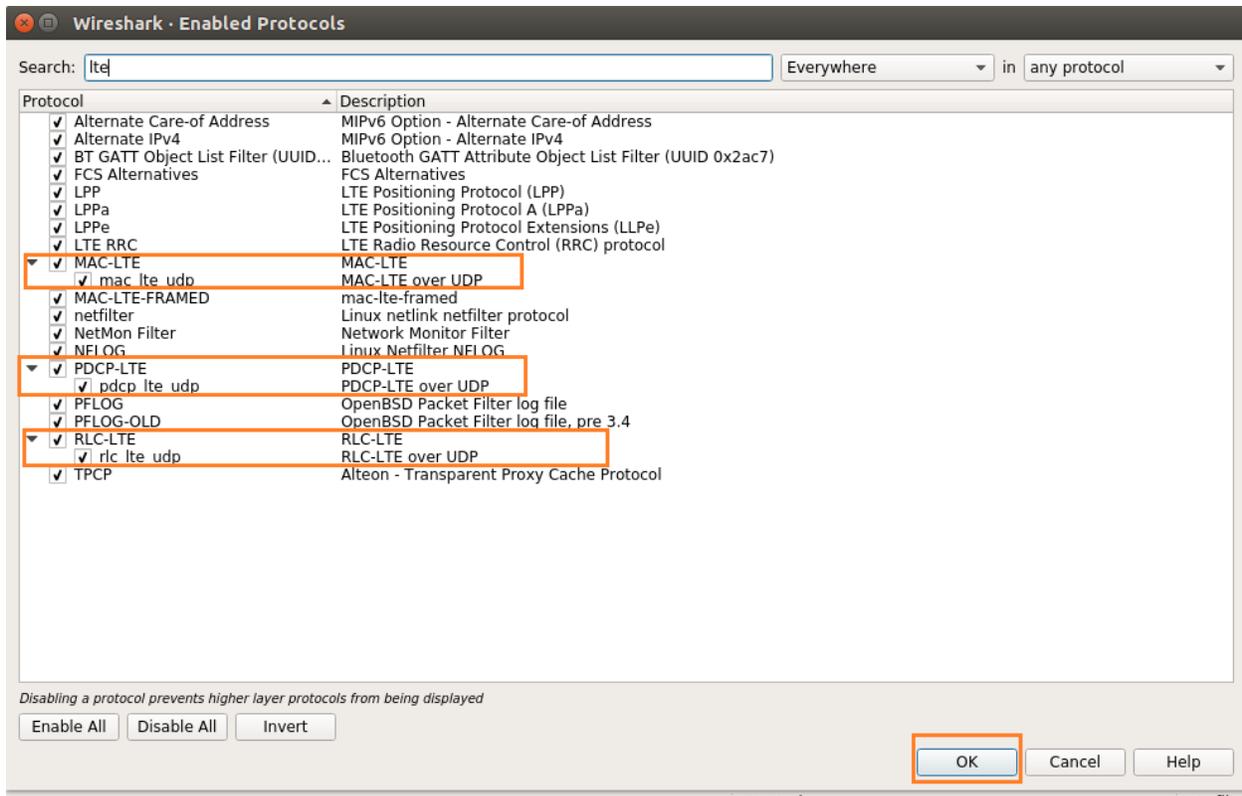


Fuente: Elaborado por el autor

A continuación, se abre el menú *Analyze* → *Enabled protocols* y en el cuadro de búsqueda de protocolos habilitados se escribe *lte*. Y se procede a habilitar los protocolos LTE RRC, MAC-LTE (*mac_lte_udp*), PDCP-LTE (*pdcp_lte_udp*) y RLC-LTE (*rlc_lte_udp*). En cuanto a los paquetes S1AP, GTP, no se requiere de configuración. En la figura 57 mostrada a continuación se visualiza la activación de los protocolos LTE.

Figura 57

Habilitación de protocolos LTE en Wireshark



Fuente: Elaborado por el autor

5.5.2. Inicio de red LTE

Una vez realizada la instalación y configuración de Wireshark para capturar y mostrar paquetes de red LTE. Se inicia la red LTE con los comandos establecidos anteriormente, tanto para el EPC y eNodeB, pero en este último se realiza la siguiente modificación.

```
./targets/bin/lte-softmodem.Rel14 -W -O ./targets/PROJECTS/GENERIC-
LTE-EPC/CONF/enb.band7.tm1.25PRB.lmssdr.conf --rf-config-file
./targets/ARCH/LMSSDR/LimeSDR_above_1p8GHz_1v4.in
```

- **-W**: permite habilitar los mensajes de nivel (L2) en Wireshark en el host local

5.5.3. Análisis de protocolo de conexión del UE

Una vez realizada la instalación y configuración de Wireshark se inicia con la captura de los paquetes en la interfaz de Loopback (lo), para ello se debe dar inicio a la red LTE (EPC y eNodeB), una vez que el dongle LTE se conecta correctamente y el servicio de red está disponible, se puede detener la captura de los paquetes de Wireshark. Y se usa el siguiente filtro **!(mac-lte.bch.pdu)** para eliminar los paquetes SI (System Information – Información del Sistema) y SIB1 (System Information Block Type 1 – Bloque de Información del Sistema Tipo 1)

5.5.3.1. Captura De Tráfico De La Interfaz Radio

En la figura 58, se observa el tráfico capturado de la interfaz radio con los mensajes Non Access Stratum – Estrato de No Acceso (NAS) y Radio Resources Control – Control de Recursos de Radio (RRC), pertenecientes a los procedimientos de registro, autenticación, activación de contextos, conexión y desconexión del UE a la red.

Figura 58

Tráfico de red en la interfaz radio NAS y RRC

No.	Time	Source	Destination	Protocol	Length	Info
185	42.129909440	127.0.0.1	127.0.0.1	LTE RRC UL_CCCH	82	RRCConnectionRequest
186	42.130938442	127.0.0.1	127.0.0.1	LTE RRC DL_SCH	79	SystemInformationBlockType1
188	42.151005729	127.0.0.1	127.0.0.1	LTE RRC DL_SCH	79	SystemInformationBlockType1
189	42.170960681	127.0.0.1	127.0.0.1	LTE RRC DL_SCH	79	SystemInformationBlockType1
192	42.179867518	127.0.0.1	127.0.0.1	LTE RRC UL_DCCH/NAS-EPS	121	RRCConnectionSetupComplete, Tracking area update request
195	42.180926073	127.0.0.1	127.0.0.1	LTE RRC DL_SCH	94	SystemInformation [SIB2 SIB3]
197	42.182025553	127.0.0.1	127.0.0.1	LTE RRC DL_DCCH/NAS-EPS	83	DLInformationTransfer, Tracking area update reject (UE identity)
199	42.191018228	127.0.0.1	127.0.0.1	LTE RRC DL_SCH	79	SystemInformationBlockType1
207	42.210990961	127.0.0.1	127.0.0.1	LTE RRC DL_SCH	79	SystemInformationBlockType1
208	42.230832236	127.0.0.1	127.0.0.1	LTE RRC DL_SCH	79	SystemInformationBlockType1
210	42.250827621	127.0.0.1	127.0.0.1	LTE RRC DL_SCH	79	SystemInformationBlockType1
216	42.260893622	127.0.0.1	127.0.0.1	LTE RRC DL_SCH	94	SystemInformation [SIB2 SIB3]
224	42.270820487	127.0.0.1	127.0.0.1	LTE RRC DL_SCH	79	SystemInformationBlockType1
226	42.290843489	127.0.0.1	127.0.0.1	LTE RRC DL_SCH	79	SystemInformationBlockType1
228	42.310854139	127.0.0.1	127.0.0.1	LTE RRC DL_SCH	79	SystemInformationBlockType1
230	42.330859348	127.0.0.1	127.0.0.1	LTE RRC DL_SCH	79	SystemInformationBlockType1
231	42.340955171	127.0.0.1	127.0.0.1	LTE RRC DL_SCH	94	SystemInformation [SIB2 SIB3]
237	42.350854747	127.0.0.1	127.0.0.1	LTE RRC DL_SCH	79	SystemInformationBlockType1
245	42.370898381	127.0.0.1	127.0.0.1	LTE RRC DL_SCH	79	SystemInformationBlockType1
251	42.386803324	127.0.0.1	127.0.0.1	LTE RRC DL_DCCH	75	RRCConnectionRelease [cause=other]
252	42.390875692	127.0.0.1	127.0.0.1	LTE RRC DL_SCH	79	SystemInformationBlockType1

Fuente: Elaborado por el autor

Además, en la misma captura de tráfico es posible observar mensajes pertenecientes a los protocolos Radio Link Control – Control de Radio Enlace (RLC) y Medium Access Control – Control de Acceso al Medio (MAC) de LTE, los cuales se presentan en la figura 59.

Figura 59

Tráfico de red en la interfaz radio RLC y MAC

No.	Time	Source	Destination	Protocol	Length	Info
198	42.190992749	127.0.0.1	127.0.0.1	MAC-LTE	121	UL-SCH: (SFN=266 , SF=1) UEId=0 (Long BSR) (Padding:remainder)
200	42.191116322	127.0.0.1	127.0.0.1	MAC-LTE	71	RAR (RA-RNTI=1, SFN=266 , SF=5) (RAPID=17: TA=0, UL-Grant=13644, Temp C-RNTI=25390)
201	42.191856994	127.0.0.1	127.0.0.1	MAC-LTE	121	UL-SCH: (SFN=266 , SF=2) UEId=0 (Long BSR) (Padding:remainder)
202	42.192918049	127.0.0.1	127.0.0.1	MAC-LTE	121	UL-SCH: (SFN=266 , SF=3) UEId=0 (Long BSR) (Padding:remainder)
203	42.194005732	127.0.0.1	127.0.0.1	MAC-LTE	121	UL-SCH: (SFN=266 , SF=4) UEId=0 (Long BSR) (Padding:remainder)
204	42.194877310	127.0.0.1	127.0.0.1	MAC-LTE	121	UL-SCH: (SFN=266 , SF=5) UEId=0 (Long BSR) (Padding:remainder)
205	42.197880600	127.0.0.1	127.0.0.1	MAC-LTE	121	UL-SCH: (SFN=266 , SF=8) UEId=0 (Long BSR) (Padding:remainder)
206	42.205019913	127.0.0.1	127.0.0.1	MAC-LTE	67	RACH Preamble chosen for UE 0 (RAPID=183, attempt=0)
209	42.231943240	127.0.0.1	127.0.0.1	MAC-LTE	67	RACH Preamble chosen for UE 0 (RAPID=183, attempt=0)
211	42.256026222	127.0.0.1	127.0.0.1	RLC-LTE	75	[DL] [AM] SRB:1 [DATA-SEGMENT] (P) sn=0 S0=0 [1-byte..
212	42.256842465	127.0.0.1	127.0.0.1	RLC-LTE	75	[DL] [AM] SRB:1 [DATA-SEGMENT] sn=0 S0=1 ..1-byte..
213	42.257888829	127.0.0.1	127.0.0.1	RLC-LTE	75	[DL] [AM] SRB:1 [DATA-SEGMENT] sn=0 S0=2 ..1-byte..
214	42.258945625	127.0.0.1	127.0.0.1	RLC-LTE	75	[DL] [AM] SRB:1 [DATA-SEGMENT] sn=0 S0=3 ..1-byte..
215	42.259843201	127.0.0.1	127.0.0.1	RLC-LTE	75	[DL] [AM] SRB:1 [DATA-SEGMENT] sn=0 S0=4 ..1-byte..
217	42.260904978	127.0.0.1	127.0.0.1	RLC-LTE	75	[DL] [AM] SRB:1 [DATA-SEGMENT] sn=0 S0=5 ..1-byte..
218	42.261957752	127.0.0.1	127.0.0.1	RLC-LTE	75	[DL] [AM] SRB:1 [DATA-SEGMENT] sn=0 S0=6 ..1-byte..
219	42.262843843	127.0.0.1	127.0.0.1	RLC-LTE	75	[DL] [AM] SRB:1 [DATA-SEGMENT] sn=0 S0=7 ..1-byte..
220	42.263932904	127.0.0.1	127.0.0.1	MAC-LTE	67	RACH Preamble chosen for UE 0 (RAPID=183, attempt=0)
221	42.263949417	127.0.0.1	127.0.0.1	RLC-LTE	75	[DL] [AM] SRB:1 [DATA-SEGMENT] sn=0 S0=8 ..1-byte..
222	42.264799768	127.0.0.1	127.0.0.1	RLC-LTE	75	[DL] [AM] SRB:1 [DATA-SEGMENT] sn=0 S0=9 ..1-byte..
223	42.265860031	127.0.0.1	127.0.0.1	RLC-LTE	75	[DL] [AM] SRB:1 [DATA-SEGMENT] (P) sn=0 S0=10 ..1-byte]
225	42.275997153	127.0.0.1	127.0.0.1	MAC-LTE	71	RAR (RA-RNTI=1, SFN=275 , SF=0) (RAPID=60: TA=1, UL-Grant=13644, Temp C-RNTI=15162)

Fuente: Elaborado por el autor

5.5.3.2. Captura De Mensajes Bloque de Información del Sistema Tipo 1 - System Information Block Type 1 (SIB1)

Para mostrar los paquetes SystemInformation y SystemInformationBlockType1, se usa el filtro **frame.len==79** obtener una lista solo con los paquetes mencionados. El SIB1 tiene información importante al realizar una evaluación si un UE puede acceder a una celda. Por lo tanto, se plantea la tabla 20 con los parámetros del paquete SIB1.

Tabla 20

Parámetros del paquete SIB1

Parámetro SIB1	Valor o Dato	Descripción
plmn-Identity (plmn-Identidad)	MCC=901 MNC=70	Plmn id es el número que identifica una red. Detalla al código de país móvil (MCC) y el código de red móvil (MNC)
trackingAreaCode (código de área)	1	Código único que el operador asigna a su área de seguimiento
cellIdentity (identidad celular)	3584	Identifica una celda dentro de PLMN
cellBarred (celda prohibida)	notBarred (no prohibido)	Estado de la celda
cellReservedForOperatorUse (celda reservada para uso del operador)	notReserved (no reservado)	Reserva de celular
freqBandIndicator (indicador de banda de frecuencia)	7	Banda operativa

Fuente: Elaborado por el autor

En la figura 60, se observa la aplicación del filtro para la captura de los parámetros del paquete SIB1.

Figura 60

Parámetros paquete SIB1

25PRB.pcapng

Archivo Edición Visualización Ir Captura Analizar Estadísticas Telefonía Wireless Herramientas Ayuda

frame.len == 79

No.	Time	Source	Destination	Protocol	Length	Info
20	50.875647542	127.0.0.1	127.0.0.1	LTE RRC DL_SCH	79	SystemInformationBlockType1
22	50.894940619	127.0.0.1	127.0.0.1	LTE RRC DL_SCH	79	SystemInformationBlockType1
23	50.914953729	127.0.0.1	127.0.0.1	LTE RRC DL_SCH	79	SystemInformationBlockType1
24	50.934945518	127.0.0.1	127.0.0.1	LTE RRC DL_SCH	79	SystemInformationBlockType1

> Frame 20: 79 bytes on wire (632 bits), 79 bytes captured (632 bits) on interface lo, id 0
 > Ethernet II, Src: 00:00:00_00:00:00 (00:00:00:00:00:00), Dst: 00:00:00_00:00:00 (00:00:00:00:00:00)
 > Internet Protocol Version 4, Src: 127.0.0.1, Dst: 127.0.0.1
 > User Datagram Protocol, Src Port: 45428, Dst Port: 9999
 > MAC-LTE BCH PDU (15 bytes, on DL-SCH transport)
 > [Context (RNTI=65535)]
 [Transport channel: DL-SCH (4)]
 > LTE Radio Resource Control (RRC) protocol
 > BCCH-DL-SCH-Message
 > message: c1 (0)
 > c1: systemInformationBlockType1 (1)
 > systemInformationBlockType1
 > cellAccessRelatedInfo
 > plmn-IdentityList: 1 item
 > Item 0
 > PLMN-IdentityInfo
 > plmn-Identity
 > mcc: 3 items
 > Item 0
 MCC-MNC-Digit: 9
 > Item 1
 MCC-MNC-Digit: 0
 > Item 2
 MCC-MNC-Digit: 1
 > mnc: 2 items
 > Item 0
 MCC-MNC-Digit: 7
 > Item 1
 MCC-MNC-Digit: 0

> Frame 20: 79 bytes on wire (632 bits), 79 bytes captured (632 bits) on interface lo, id 0
 > Ethernet II, Src: 00:00:00_00:00:00 (00:00:00:00:00:00), Dst: 00:00:00_00:00:00 (00:00:00:00:00:00)
 > Internet Protocol Version 4, Src: 127.0.0.1, Dst: 127.0.0.1
 > User Datagram Protocol, Src Port: 45428, Dst Port: 9999
 > MAC-LTE BCH PDU (15 bytes, on DL-SCH transport)
 > [Context (RNTI=65535)]
 [Transport channel: DL-SCH (4)]
 > LTE Radio Resource Control (RRC) protocol
 > BCCH-DL-SCH-Message
 > message: c1 (0)
 > c1: systemInformationBlockType1 (1)
 > systemInformationBlockType1
 > cellAccessRelatedInfo
 > plmn-IdentityList: 1 item
 > Item 0
 > PLMN-IdentityInfo
 > plmn-Identity
 > cellReservedForOperatorUse: notReserved (1)
 trackingAreaCode: 0001 [bit length 16, 0000 0000 0000 0001 decimal value 1]
 cellIdentity: 0000e000 [bit length 28, 4 LSB pad bits, 0000 0000 0000 0000 1110 0000 0000 decimal value 3584]
 cellBarred: notBarred (1)
 intraFreqReselection: notAllowed (1)
0. csg-Indication: False
 > cellSelectionInfo
 freqBandIndicator: 7
 > schedulingInfoList: 1 item
 si-WindowLength: ms20 (5)
 systemInfoValueTag: 0

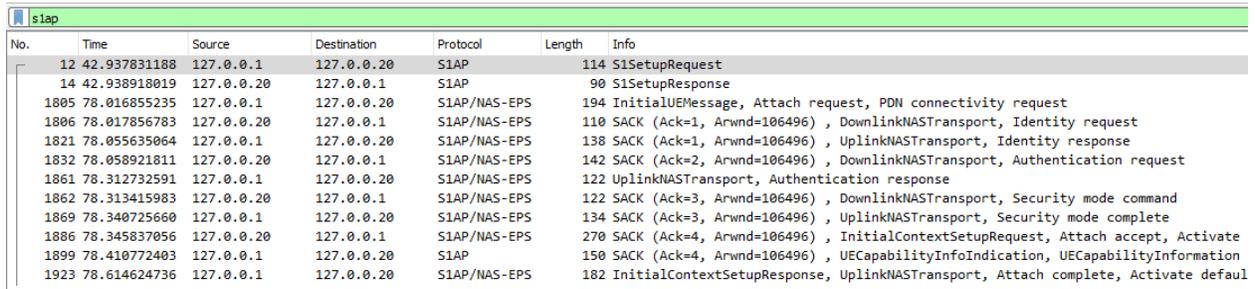
Fuente: Elaborado por el autor

5.5.3.3. Captura De Tráfico En La Interfaz S1

En este proceso, se puede observar el tráfico capturado en la interfaz S1 de LTE implementado por OAI, en el plano de control (señalización, perteneciente a la interfaz S1-MME) como también para el plano de usuario (datos, perteneciente a la interfaz S1-U). El tráfico en la interfaz S1-MME (señalización) usa el protocolo S1AP en el nivel de aplicación. En la figura 61 mostrada a continuación, se puede observar además la información NAS intercambiada del eNodeB y MME, en base a los parámetros mostrados en la interfaz radio como el registro, autenticación, activación de contextos, desconexión.

Figura 61

Tráfico capturado interfaz S1 (plano de control)



No.	Time	Source	Destination	Protocol	Length	Info
12	42.937831188	127.0.0.1	127.0.0.20	S1AP	114	S1SetupRequest
14	42.938918019	127.0.0.20	127.0.0.1	S1AP	90	S1SetupResponse
1805	78.016855235	127.0.0.1	127.0.0.20	S1AP/NAS-EPS	194	InitialUEMessage, Attach request, PDN connectivity request
1806	78.017856783	127.0.0.20	127.0.0.1	S1AP/NAS-EPS	110	SACK (Ack=1, Arwnd=106496), DownlinkNASTransport, Identity request
1821	78.055635064	127.0.0.1	127.0.0.20	S1AP/NAS-EPS	138	SACK (Ack=1, Arwnd=106496), UplinkNASTransport, Identity response
1832	78.058921811	127.0.0.20	127.0.0.1	S1AP/NAS-EPS	142	SACK (Ack=2, Arwnd=106496), DownlinkNASTransport, Authentication request
1861	78.312732591	127.0.0.1	127.0.0.20	S1AP/NAS-EPS	122	UplinkNASTransport, Authentication response
1862	78.313415983	127.0.0.20	127.0.0.1	S1AP/NAS-EPS	122	SACK (Ack=3, Arwnd=106496), DownlinkNASTransport, Security mode command
1869	78.340725660	127.0.0.1	127.0.0.20	S1AP/NAS-EPS	134	SACK (Ack=3, Arwnd=106496), UplinkNASTransport, Security mode complete
1886	78.345837056	127.0.0.20	127.0.0.1	S1AP/NAS-EPS	270	SACK (Ack=4, Arwnd=106496), InitialContextSetupRequest, Attach accept, Activate
1899	78.410772403	127.0.0.1	127.0.0.20	S1AP	150	SACK (Ack=4, Arwnd=106496), UECapabilityInfoIndication, UECapabilityInformation
1923	78.614624736	127.0.0.1	127.0.0.20	S1AP/NAS-EPS	182	InitialContextSetupResponse, UplinkNASTransport, Attach complete, Activate default

Fuente: Elaborado por el autor

Para comprender mejor el intercambio de información de control, se presenta un diagrama de flujo, como se observa en la figura 62, donde la dirección IP 127.0.0.1 corresponde al eNodeB y la dirección IP 127.0.0.20 corresponde a la MME del EPC.

Figura 62

Diagrama de flujo tráfico interfaz S1 (plano de control)

Intervalo	127.0.0.1	127.0.0.20	Comentario
42.937831188	35931	S1SetupRequest → 36412	S1AP: S1SetupRequest
42.938918019	35931	← S1SetupResponse 36412	S1AP: S1SetupResponse
78.016855235	35931	InitialUEMessage, Attach request → 36412	S1AP/NAS-EPS: InitialUEMessage, Attach request...
78.017856783	35931	SACK (Ack=1, Arwnd=106496) → 36412	S1AP/NAS-EPS: SACK (Ack=1, Arwnd=106496) ...
78.055635064	35931	SACK (Ack=1, Arwnd=106496) → 36412	S1AP/NAS-EPS: SACK (Ack=1, Arwnd=106496) ...
78.058921811	35931	SACK (Ack=2, Arwnd=106496) → 36412	S1AP/NAS-EPS: SACK (Ack=2, Arwnd=106496) ...
78.312732591	35931	UplinkNASTransport, Authentication → 36412	S1AP/NAS-EPS: UplinkNASTransport, Authenticat...
78.313415983	35931	SACK (Ack=3, Arwnd=106496) → 36412	S1AP/NAS-EPS: SACK (Ack=3, Arwnd=106496) ...
78.340725660	35931	SACK (Ack=3, Arwnd=106496) → 36412	S1AP/NAS-EPS: SACK (Ack=3, Arwnd=106496) ...
78.345837056	35931	SACK (Ack=4, Arwnd=106496) → 36412	S1AP/NAS-EPS: SACK (Ack=4, Arwnd=106496) ...
78.410772403	35931	SACK (Ack=4, Arwnd=106496) → 36412	S1AP: SACK (Ack=4, Arwnd=106496) , UECapa...
78.614624736	35931	InitialContextSetupResponse, Up → 36412	S1AP/NAS-EPS: InitialContextSetupResponse, Upl...

Fuente: Elaborado por el autor

5.5.3.4. Captura De Tráfico Protocolo de Tunelización GPRS (GTP)

En la interfaz S1-U el tráfico del plano de usuario se encapsula mediante el protocolo GTP, el tráfico mencionado puede observarse en la figura 63 mostrada a continuación.

Figura 63

Tráfico capturado interfaz S1 (plano de usuario protocolo GTP)

No.	Time	Source	Destination	Protocol	Length	Info
2814	82.732744131	172.16.0.2	8.8.4.4	GTP <DNS>	112	Standard query 0x6747 A oneclient.sfx.ms
2817	82.752751983	172.16.0.2	40.126.7.35	GTP <TCP>	90	56306 → 443 [RST, ACK] Seq=2789 Ack=12998 Win=0 Len=0
2820	82.772764825	172.16.0.2	8.8.8.8	GTP <DNS>	112	Standard query 0x6747 A oneclient.sfx.ms
2823	82.780732104	172.16.0.2	8.8.4.4	GTP <DNS>	112	Standard query 0x6747 A oneclient.sfx.ms
2835	82.829087413	8.8.4.4	172.16.0.2	GTP <DNS>	206	Standard query response 0x6747 A oneclient.sfx.ms CNAME oneclient.sfx.ms.
2841	82.862725738	172.16.0.2	96.17.20.38	GTP <TCP>	102	56330 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1360 WS=256 SACK_PERM
2842	82.863264492	96.17.20.38	172.16.0.2	GTP <TCP>	102	443 → 56330 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460 SACK_PERM WS=
2844	82.865329461	8.8.4.4	172.16.0.2	GTP <DNS>	206	Standard query response 0x6747 A oneclient.sfx.ms CNAME oneclient.sfx.ms.
2848	82.882728289	172.16.0.2	96.17.20.38	GTP <TCP>	90	56330 → 443 [ACK] Seq=1 Ack=1 Win=131840 Len=0
2854	82.918708677	172.16.0.2	96.17.20.38	GTP <TLSv1.2>	291	Client Hello
2855	82.919197661	96.17.20.38	172.16.0.2	GTP <TCP>	90	443 → 56330 [ACK] Seq=1 Ack=202 Win=30336 Len=0
2860	82.922380417	8.8.8.8	172.16.0.2	GTP <DNS>	206	Standard query response 0x6747 A oneclient.sfx.ms CNAME oneclient.sfx.ms.
2880	83.062694188	172.16.0.2	8.8.4.4	GTP <DNS>	122	Standard query 0x793d A utneduec-my.sharepoint.com
2883	83.090653615	172.16.0.2	8.8.8.8	GTP <DNS>	122	Standard query 0x793d A utneduec-my.sharepoint.com
2884	83.090664323	172.16.0.2	8.8.4.4	GTP <DNS>	122	Standard query 0x793d A utneduec-my.sharepoint.com
2893	83.173323544	8.8.4.4	172.16.0.2	GTP <DNS>	444	Standard query response 0x793d A utneduec-my.sharepoint.com CNAME utnedue
2895	83.175309233	8.8.4.4	172.16.0.2	GTP <DNS>	444	Standard query response 0x793d A utneduec-my.sharepoint.com CNAME utnedue
2901	83.202744856	172.16.0.2	13.107.136.8	GTP <TCP>	102	56335 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1360 WS=256 SACK_PERM
2902	83.203256608	13.107.136.8	172.16.0.2	GTP <TCP>	102	443 → 56335 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460 SACK_PERM WS=
2907	83.222748022	172.16.0.2	13.107.136.8	GTP <TCP>	90	56335 → 443 [ACK] Seq=1 Ack=1 Win=131840 Len=0

Fuente: Elaborado por el autor

De igual manera se puede observar el tráfico del plano de usuario sin encapsular mediante GTP, este tráfico corresponde al S-PGW de OAI, en el cual se elimina la encapsulación GTP antes de dirigir el tráfico hacia internet por la interfaz SGi. El tráfico mencionado se presenta en la figura 64, donde se identifica el campo Protocol de Wireshark en cual ya no tiene el parámetro GTP. Además, en los campos Source y Destination se puede identificar a las direcciones IP de origen y destino.

Figura 64

Tráfico capturado plano de usuario sin encapsular protocolo GTP

No.	Time	Source	Destination	Protocol	Length	Info
1340	64.836326182	127.0.0.1	127.0.0.1	MAC-LTE	121	UL-SCH: (SFN=935 , SF=3) UEId=0 (Long BSR) (Padding:remainder)
1341	64.837205771	127.0.0.1	127.0.0.1	RLC-LTE	86	[UL] [AM] SRB:1 [DATA] sn=1 ..19-bytes..
1342	64.837365690	127.0.0.1	127.0.0.20	SIAP/NAS-EPS	174	InitialUEMessage, Attach request, PDN connectivity request
1343	64.837999629	127.0.0.1	127.0.0.1	DIAMETER	354	cmd=3GPP-Authentication-Information Request(318) flags=RP-- appl=3GPP S6a/S6d(16777251)
1344	64.838008489	127.0.0.1	127.0.0.1	TCP	66	3868 -> 3386 [ACK] Seq=301 Ack=577 Win=510 Len=0 TSval=1479369409 TSecr=1479369409
1345	64.838234132	127.0.0.1	127.0.0.1	TCP	184	40944 -> 3386 [PSH, ACK] Seq=1 Ack=1 Win=512 Len=118 TSval=1479369409 TSecr=1479244076 [
1346	64.839216506	127.0.0.1	127.0.0.1	TCP	381	3386 -> 40944 [PSH, ACK] Seq=1 Ack=119 Win=512 Len=315 TSval=1479369410 TSecr=1479369409
1347	64.839225507	127.0.0.1	127.0.0.1	TCP	66	40944 -> 3386 [ACK] Seq=119 Ack=316 Win=510 Len=0 TSval=1479369410 TSecr=1479369410
1348	64.839339670	127.0.0.1	127.0.0.1	MAC-LTE	121	UL-SCH: (SFN=935 , SF=6) UEId=0 (Long BSR) (Padding:remainder)
1349	64.839373655	127.0.0.1	127.0.0.1	TCP	217	40944 -> 3386 [PSH, ACK] Seq=119 Ack=316 Win=512 Len=151 TSval=1479369410 TSecr=1479369410
1350	64.839975912	127.0.0.1	127.0.0.1	TCP	147	3386 -> 40944 [PSH, ACK] Seq=316 Ack=270 Win=512 Len=81 TSval=1479369411 TSecr=1479369411
1351	64.840005428	127.0.0.1	127.0.0.1	TCP	176	40944 -> 3386 [PSH, ACK] Seq=270 Ack=397 Win=512 Len=110 TSval=1479369411 TSecr=1479369411
1352	64.840277218	127.0.0.1	127.0.0.1	TCP	147	3386 -> 40944 [PSH, ACK] Seq=397 Ack=380 Win=512 Len=81 TSval=1479369411 TSecr=1479369411
1353	64.840347886	127.0.0.1	127.0.0.1	DIAMETER	366	cmd=3GPP-Authentication-Information Answer(318) flags=-P-- appl=3GPP S6a/S6d(16777251)
1354	64.840911551	127.0.0.20	127.0.0.1	SIAP/NAS-EPS	142	SACK (Ack=3, Arwnd=106496) , DownlinkNASTransport, Authentication request
1355	64.841331339	127.0.0.1	127.0.0.1	LTE RRC DL_DCCH/NAS-EPS	113	DLInformationTransfer, Authentication request
1357	64.846246396	127.0.0.1	127.0.0.1	MAC-LTE	121	UL-SCH: (SFN=936 , SF=3) UEId=0 (Long BSR) (Padding:remainder)
1358	64.849262396	127.0.0.1	127.0.0.1	MAC-LTE	67	RACH Preamble chosen for UE 0 (RAPID=244, attempt=0)
1361	64.880611510	127.0.0.1	127.0.0.1	TCP	66	40944 -> 3386 [ACK] Seq=380 Ack=478 Win=512 Len=0 TSval=1479369451 TSecr=1479369411
1362	64.880611481	127.0.0.1	127.0.0.1	TCP	66	36864 -> 3868 [ACK] Seq=577 Ack=601 Win=512 Len=0 TSval=1479369451 TSecr=1479369411
1364	64.885721451	127.0.0.1	127.0.0.1	MAC-LTE	67	RACH Preamble chosen for UE 0 (RAPID=244, attempt=0)

Fuente: Elaborado por el autor

5.6. Guía 4: Familiarización GNU Radio y LimeSDR Mini

La presente guía tiene como finalidad brindar instrucciones detalladas para la instalación de GNU Radio y crear paso a paso de un diagrama de flujo para un transmisor QPSK. Todo esto mediante el uso de GNU Radio en conjunto con el dispositivo LimeSDR Mini.

UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS
CARRERA DE INGENIERÍA EN ELECTRÓNICA Y REDES DE
COMUNICACIÓN



TEMA: Familiarización GNU Radio y LimeSDR Mini

AUTOR: ROMEL RICARDO ROSERO

Esta guía se enfoca en la instalación de GNU Radio como implementación de aplicaciones de Radio Definido por Software con la elaboración de un transmisor QPSK con el funcionamiento de GNU Radio y LimeSDR Mini

Objetivos

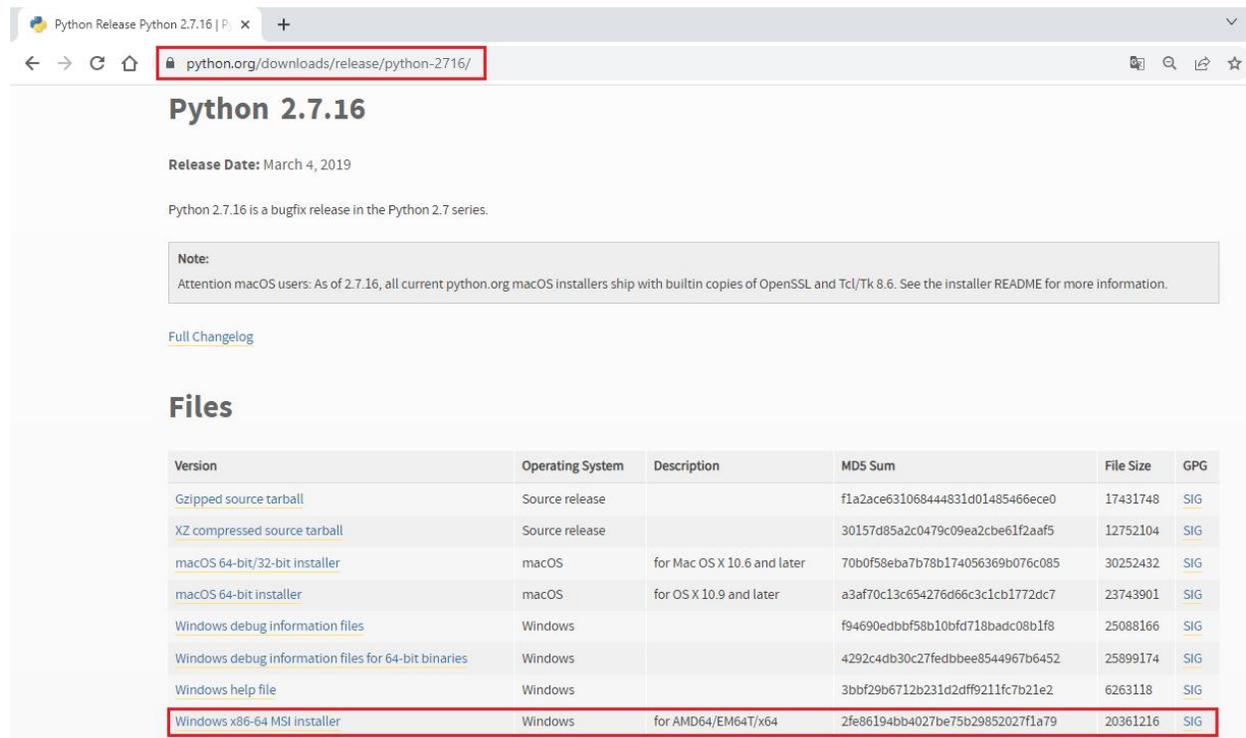
- Comprender el proceso de instalación de GNU Radio para su implementación en aplicaciones de Radio Definido por Software (SDR)
- Identificar los bloques apropiados para construir el diagrama de flujo
- Desarrollar un transmisor QPSK utilizando las capacidades de GNU Radio en conjunto con LimeSDR Mini

5.6.1. Instalación De LimeSuite y GNU Radio Companion

El primer paso a realizar es instalar Python 2.7.16, para ello se debe ir a la página oficial de Python, el siguiente enlace <https://www.python.org/downloads/release/python-2716/> dirige hacia la versión 2.7.16 donde se debe seleccionar el instalador *MSI de Windows x86-64* para descargar el archivo de instalación. En la figura 65, se observa la selección del instalador MSI de Windows

Figura 65

Selección de instalador MSI Windows x86-64



The screenshot shows the Python 2.7.16 release page. The browser address bar is highlighted with a red box, showing the URL [python.org/downloads/release/python-2716/](https://www.python.org/downloads/release/python-2716/). The page title is "Python 2.7.16". Below the title, it says "Release Date: March 4, 2019" and "Python 2.7.16 is a bugfix release in the Python 2.7 series." There is a "Note" section for macOS users. Below that is a "Full Changelog" link. The "Files" section contains a table with the following data:

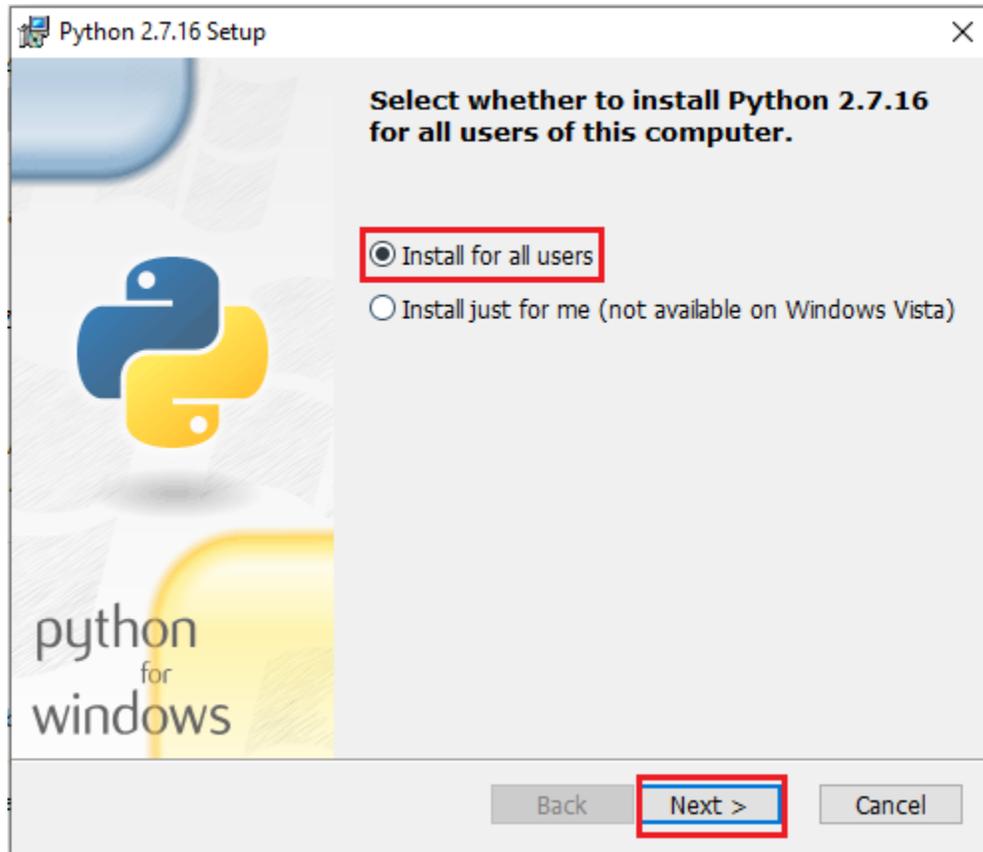
Version	Operating System	Description	MD5 Sum	File Size	GPG
Gzipped source tarball	Source release		f1a2ace631068444831d01485466ece0	17431748	SIG
XZ compressed source tarball	Source release		30157d85a2c0479c09ea2cbe61f2aaf5	12752104	SIG
macOS 64-bit/32-bit installer	macOS	for Mac OS X 10.6 and later	70b0f58eba7b78b174056369b076c085	30252432	SIG
macOS 64-bit installer	macOS	for OS X 10.9 and later	a3af70c13c654276d66c3c1cb1772dc7	23743901	SIG
Windows debug information files	Windows		f94690edbbf58b10bfd718badc08b1f8	25088166	SIG
Windows debug information files for 64-bit binaries	Windows		4292c4db30c27fedbbe8544967b6452	25899174	SIG
Windows help file	Windows		3bbf29b6712b231d2dff9211fc7b21e2	6263118	SIG
Windows x86-64 MSI installer	Windows	for AMD64/EM64T/x64	2fe86194bb4027be75b29852027f1a79	20361216	SIG

Fuente: Elaborado por el autor

Una vez descargado, se procede a instalar Python 2.7.16 en la PC. Y en el cuadro mostrado, se selecciona la opción Install for all users – Instalar para todos los usuarios y presionar en siguiente, la figura 66 presenta la selección de la instalación.

Figura 66

Selección de instalación para todos los usuarios

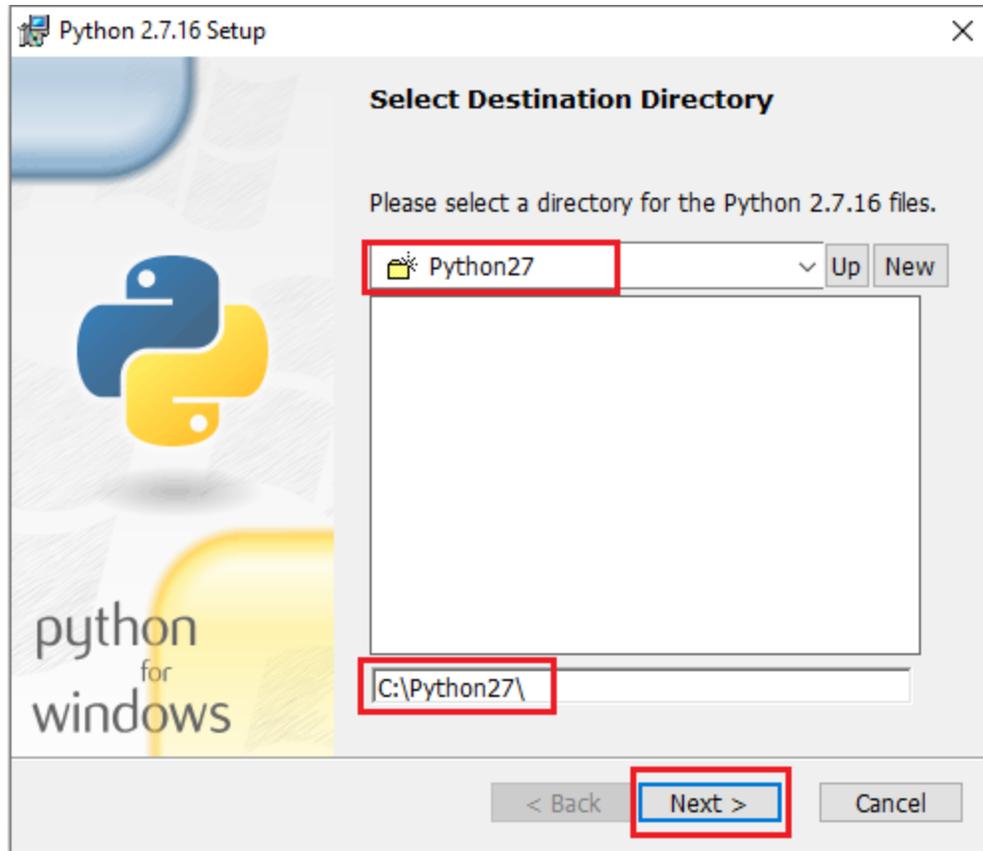


Fuente: Elaborado por el autor

El siguiente paso es seleccionar el directorio de destino, el cual se deja por defecto en el disco local C en la carpeta Python 27 y se continúa con la instalación al hacer clic en siguiente, la figura 67 muestra el cuadro de diálogo con el directorio de destino.

Figura 67

Directorio de destino Python27



Fuente: Elaborado por el autor

A continuación, se procede a personalizar Python, seleccionando la forma en la que se desea que se instalen las funciones, en la figura 68 se observa los íconos en el árbol en la forma en la que instalarán las funciones, no se realiza ningún cambio y se presiona en siguiente.

Figura 68

Personalización de Python

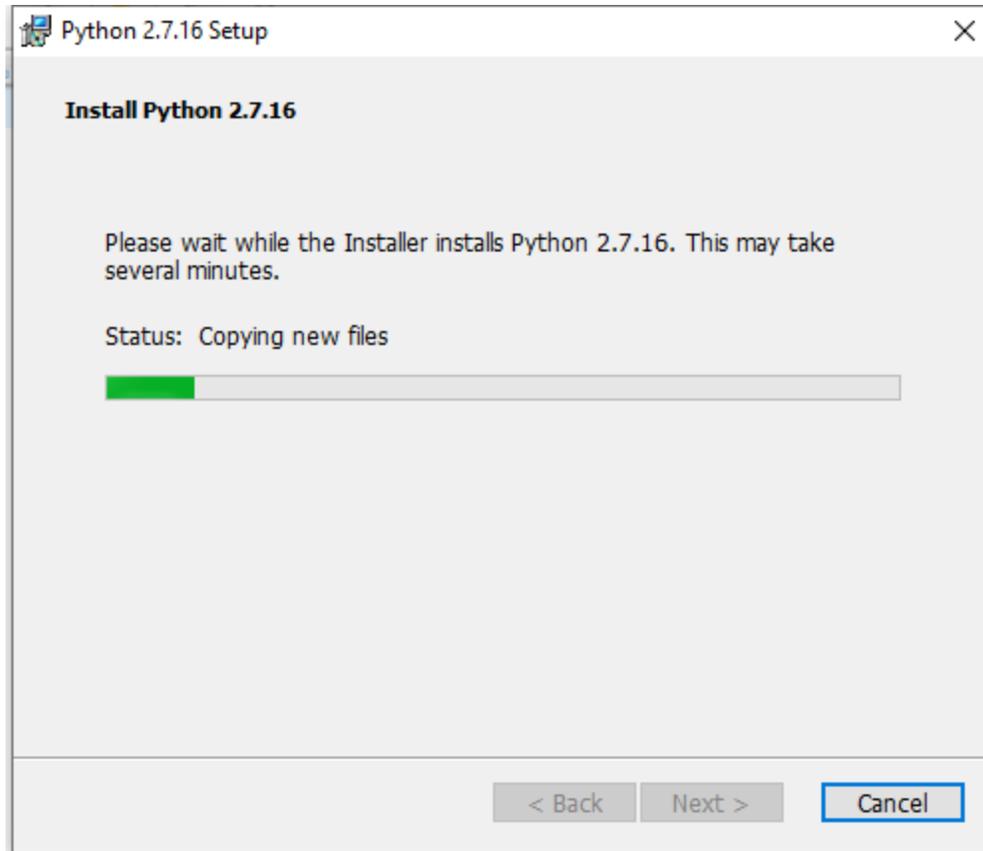


Fuente: Elaborado por el autor

La ventana de configuración mostrará que la instalación de Python 2.7.16 se está realizando con éxito, en la figura 69 se evidencia el proceso de instalación con el estado del mismo.

Figura 69

Proceso de instalación de Python



Fuente: Elaborado por el autor

El proceso de instalación puede tomar algunos minutos y al finalizar se observa el mensaje que la instalación de Python 2.7.16 se ha completado correctamente. En la figura 70 se muestra la ventana de configuración realizada con éxito y se debe dar clic en Finalizar para salir del instalador.

Figura 70

Instalación de Python finalizada



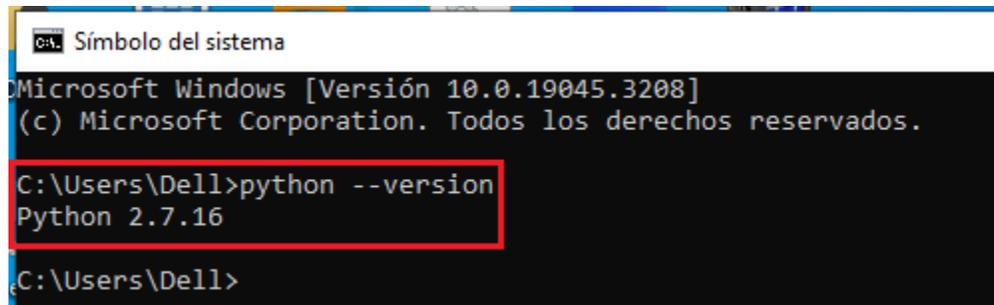
Fuente: Elaborado por el autor

5.6.1.1. Verificación De Instalación De Python 2.7.16

Para realizar la comprobación de la instalación y la versión de Python, en la computadora al ingresar al símbolo del sistema (CMD) se ejecuta el comando `python --version`. En la figura 71 se muestra que la versión instalada de Python es la 2.7.16 realizada correctamente.

Figura 71

Verificación de instalación de Python 2.7.16



```
Símbolo del sistema
Microsoft Windows [Versión 10.0.19045.3208]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\Dell>python --version
Python 2.7.16

C:\Users\Dell>
```

Fuente: Elaborado por el autor

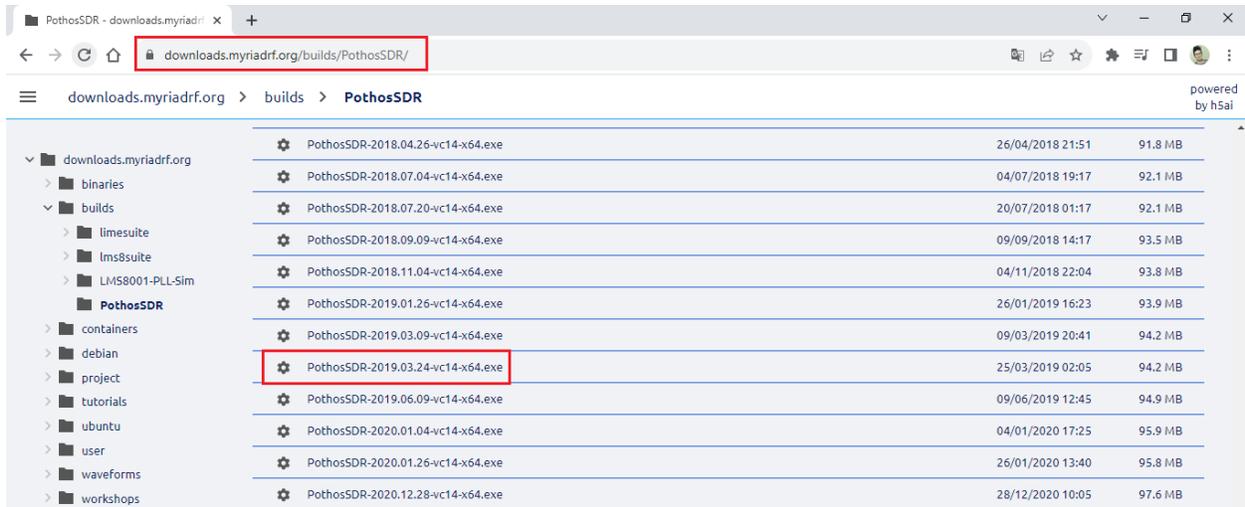
5.6.2. Instalación Del Entorno Pothos SDR

Una vez instalado GNU Radio se procede a realizar la instalación del entorno de desarrollo Pothos SDR. Este entorno brinda a los usuarios de Windows la posibilidad de explorar y desarrollar con hardware y software SDR convencionales, sin la necesidad de descargar y configurar paquetes de software.

El primer paso a realizar es descargar Pothos SDR, para ello se dirige a la página de MyriadRF, en el siguiente enlace <https://downloads.myriardf.org/builds/PothosSDR/> y buscar la versión 2019.03.24 donde se debe seleccionar el instalador del entorno *PothosSDR-2019.03.24-vc14-x64.exe* para descargar el archivo de instalación. En la figura 72, se observa la selección del instalador del entorno de Pothos SDR

Figura 72

Selección de instalador Pothos SDR



Fuente: Elaborado por el autor

Una vez descargado, se procede a instalar el entorno de Pothos SDR en la PC. Se abre el asistente de configuración y en el cuadro mostrado, se muestra un mensaje de recomendación de cerrar las demás aplicaciones antes de iniciar con la instalación, se da clic en siguiente para continuar, en la figura 73 se presenta la bienvenida al asistente de instalación de Pothos SDR.

Figura 73

Ventana de inicio de Instalación de Pothos SDR

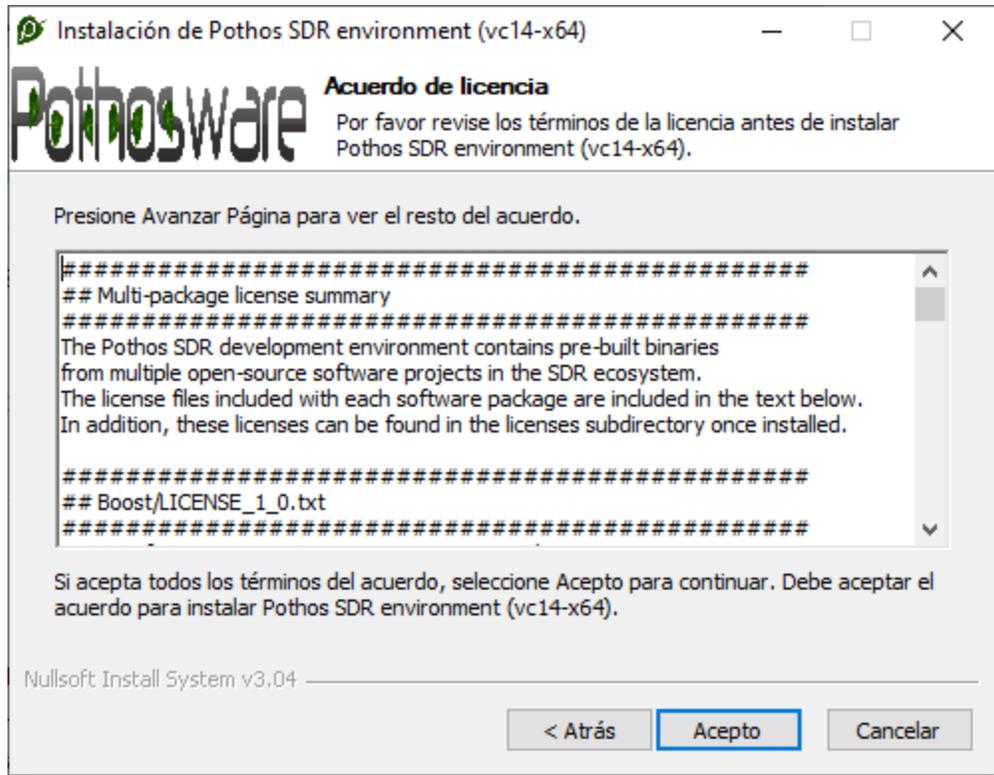


Fuente: Elaborado por el autor

El siguiente paso, corresponde al acuerdo de licencia, al estar conforme con los términos del acuerdo, se selecciona Acepto para continuar con la instalación de Pothos SDR. En la figura 74 se observa el acuerdo de licencia.

Figura 74

Acuerdo de licencia Pothos SDR

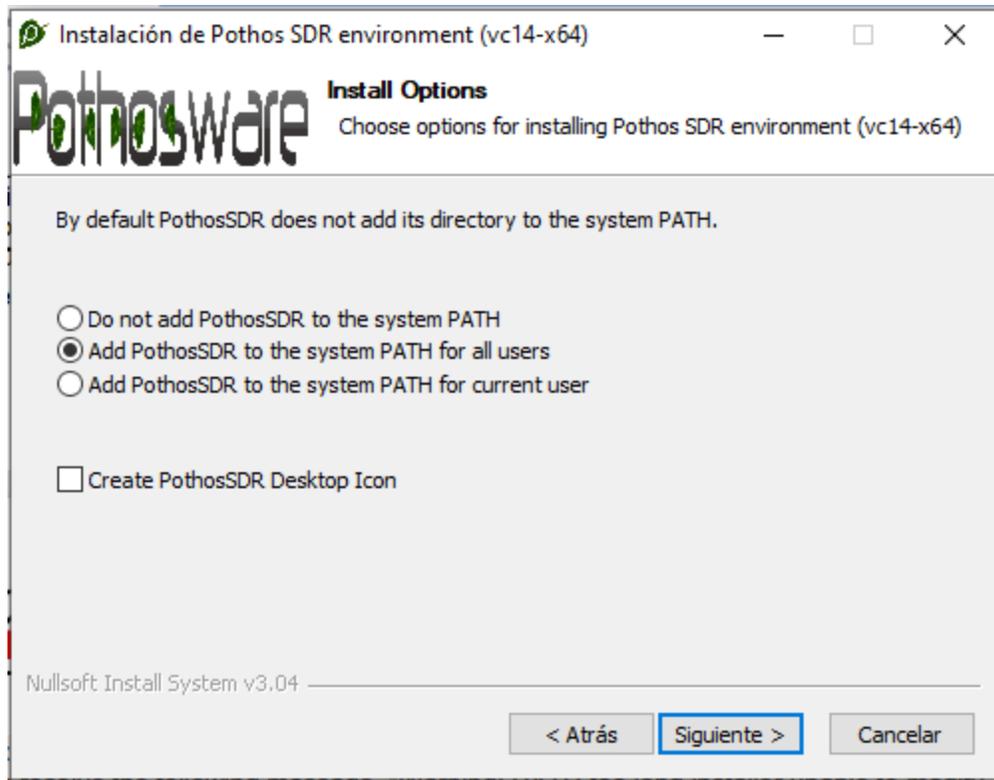


Fuente: Elaborado por el autor

Una vez aceptado el acuerdo de licencia, el siguiente paso es seleccionar la opción de instalación, por defecto Pothos SDR no agrega el directorio a la ruta del sistema. En la figura 75 se puede observar la selección de Add PothosSDR to the system PATH for all users – Agregue Pothos SDR a la ruta del sistema para todos los usuarios.

Figura 75

Selección de opción agregar Pothos SDR a la ruta del sistema

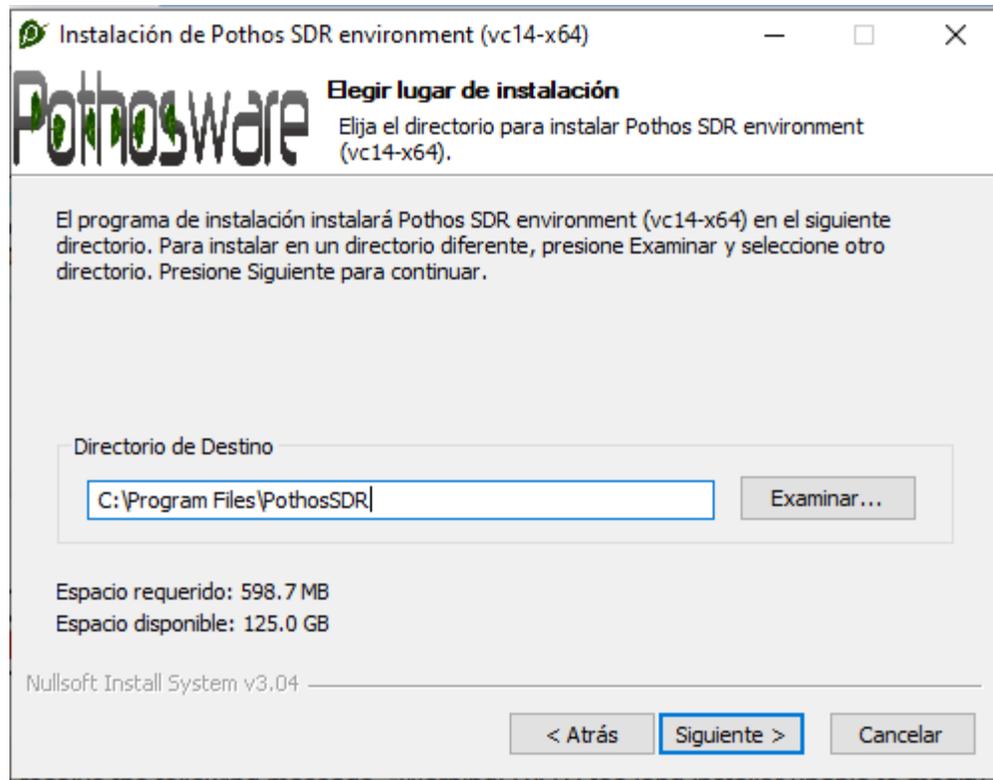


Fuente: Elaborado por el autor

El siguiente paso consiste en seleccionar el directorio de destino, el cual se deja por defecto en el disco local C en la carpeta Program Files - PothosSDR y se continúa con la instalación al hacer clic en siguiente, la figura 76 muestra el directorio de destino para la instalación de Pothos SDR.

Figura 76

Directorio de destino Pothos SDR

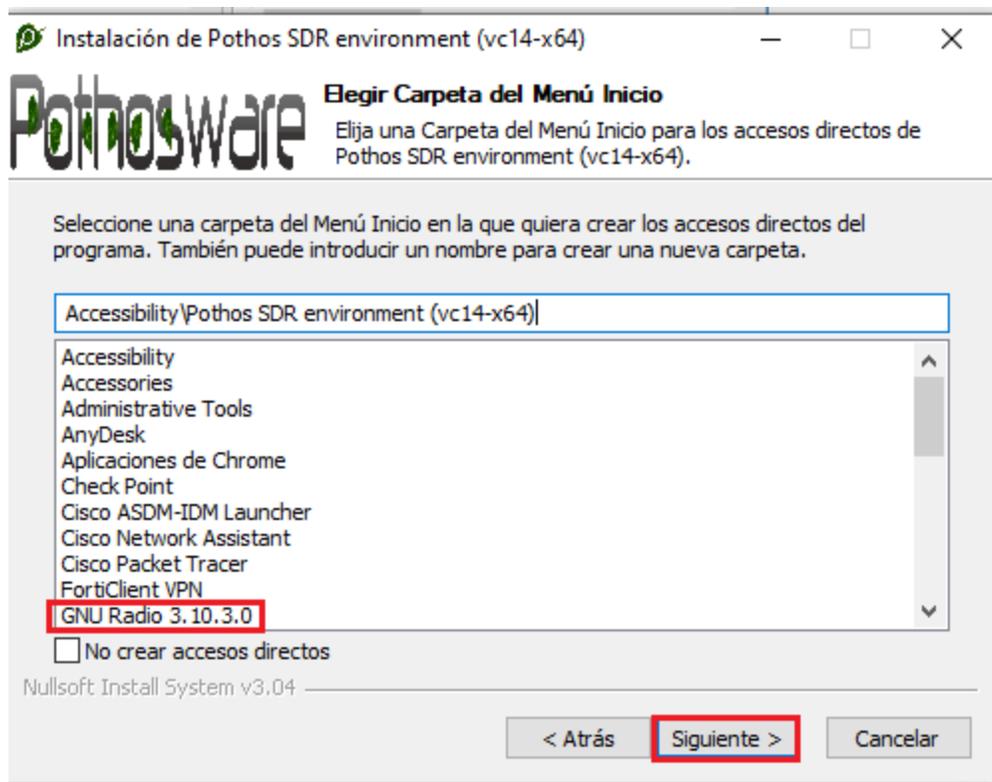


Fuente: Elaborado por el autor

A continuación, se procede a elegir la carpeta del menú de inicio para la creación de los accesos directos del programa. La cual se debe seleccionar la opción de GNU Radio 3.10.3.0. En la figura 77 se presenta la carpeta del menú de inicio.

Figura 77

Selección de carpeta del menú inicio

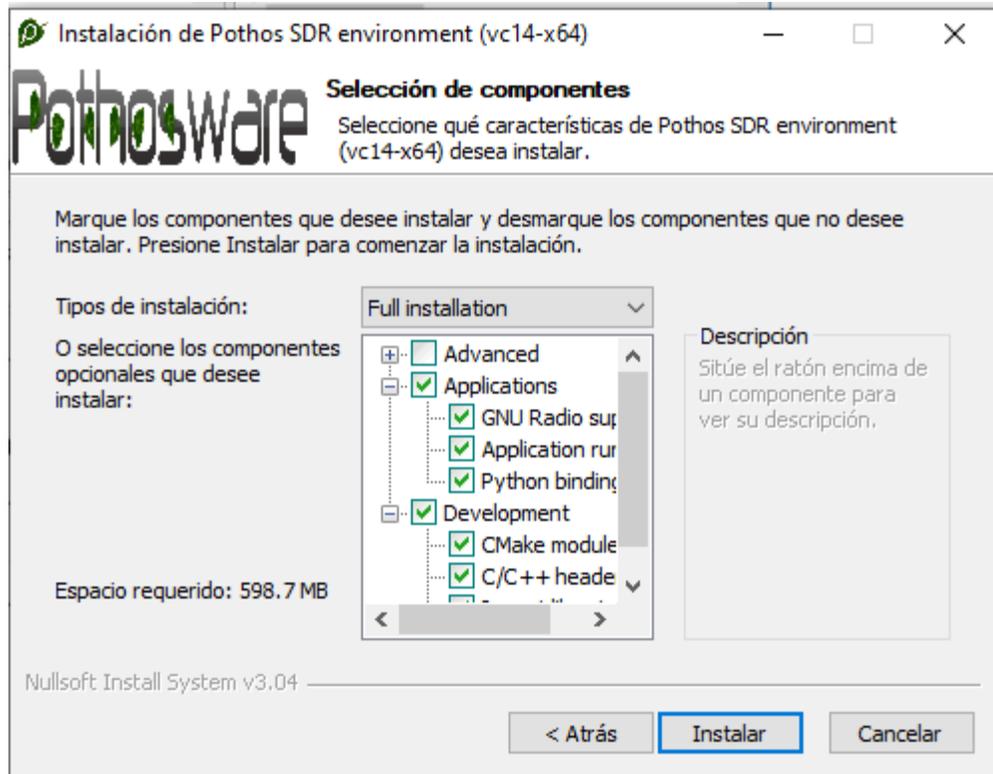


Fuente: Elaborado por el autor

Ahora se debe seleccionar los componentes, es decir, que características del entorno de Pothos SDR se desea instalar, en la figura 78 se observa los componentes marcados en el árbol en la forma en la que se realizará la instalación, se deja por defecto con los componentes establecidos y se presiona en Instalar.

Figura 78

Selección de componentes Pothos SDR

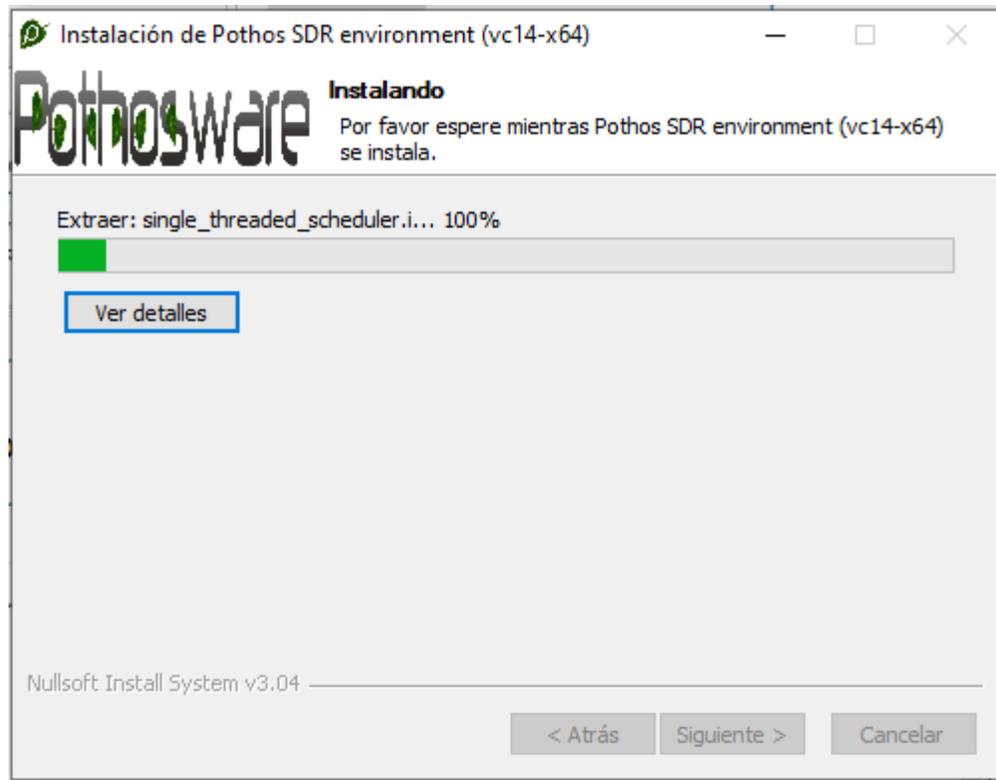


Fuente: Elaborado por el autor

La ventana de configuración indicará que la instalación de Pothos SDR se está llevando a cabo de manera exitosa, en la figura 79 se puede apreciar el proceso de instalación junto con su estado actual.

Figura 79

Proceso de instalación de Pothos SDR

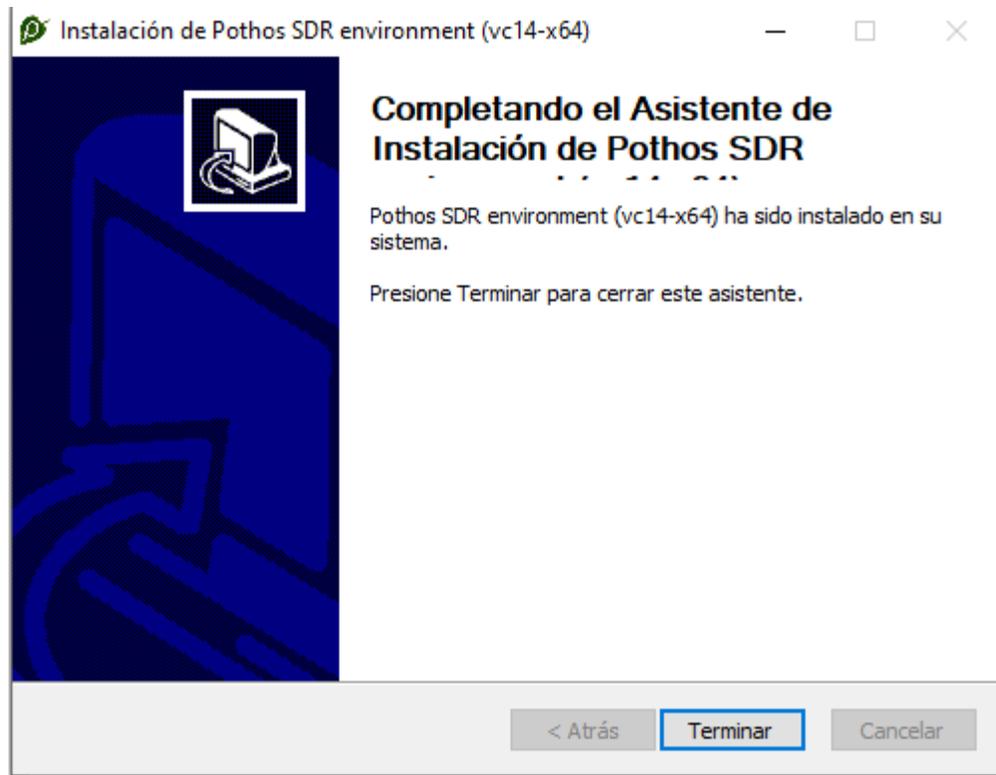


Fuente: Elaborado por el autor

La instalación puede tardar varios minutos y una vez finalizada, aparecerá un mensaje confirmando que Pothos SDR se ha completado correctamente. La figura 80 se ilustra la ventana de configuración exitosa y para cerrar el asistente de instalación, se debe dar clic en Terminar.

Figura 80

Instalación de Pothos SDR finalizada



Fuente: Elaborado por el autor

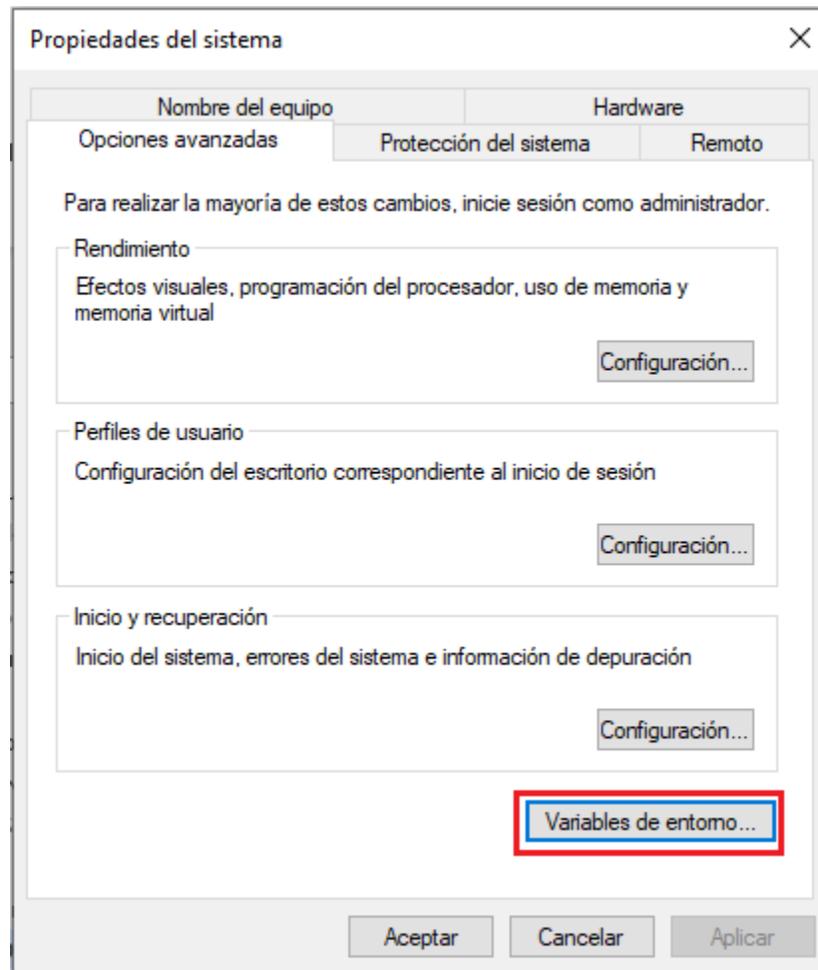
5.6.2.1. Agregación De Pothon SDR a La Ruta Del Sistema

Para que Pothon SDR pueda funcionar correctamente, es necesario agregar a la ruta (PATH) del sistema en la PC, es decir, con esto se permite que el sistema operativo pueda encontrar los ejecutables y bibliotecas necesarias al ingresar comandos relacionados con Pothon SDR en la línea de comandos, con el fin de simplificar el proceso y trabajar de una manera más fluida con Pothon SDR y poder realizar tareas de desarrollo y procesamiento de señales.

Para ello en el buscador junto a la barra de tareas, escribir la siguiente orden *Editar las variables de entorno del sistema*, la cual abre una ventana de Propiedades del sistema, en la figura 81 se aprecia la ventana en la cual se debe seleccionar la opción de Variables de entorno.

Figura 81

Ventana propiedades del sistema

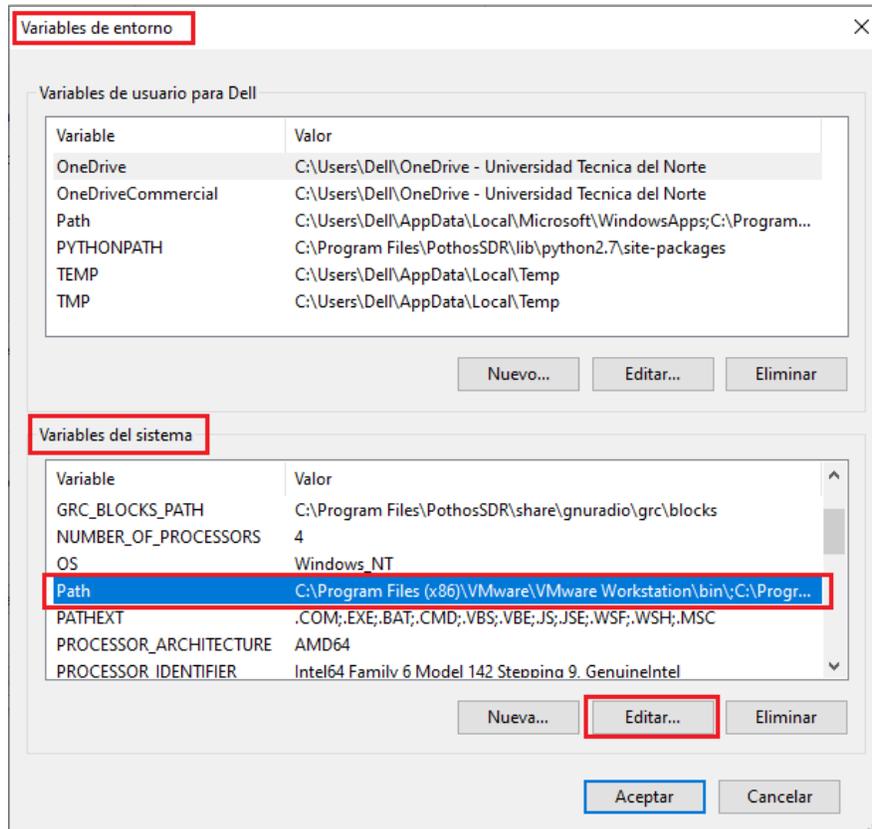


Fuente: Elaborado por el autor

Al seleccionar la opción de variables de entorno se abre una nueva ventana, en la cual nos dirigimos al cuadro denominado variables del sistema, en la variable Path (Ruta) y se selecciona en Editar. En la figura 82 se observa el cuadro con las variables del entorno tanto las variables de usuario como las variables del sistema.

Figura 82

Ventana variables del entorno

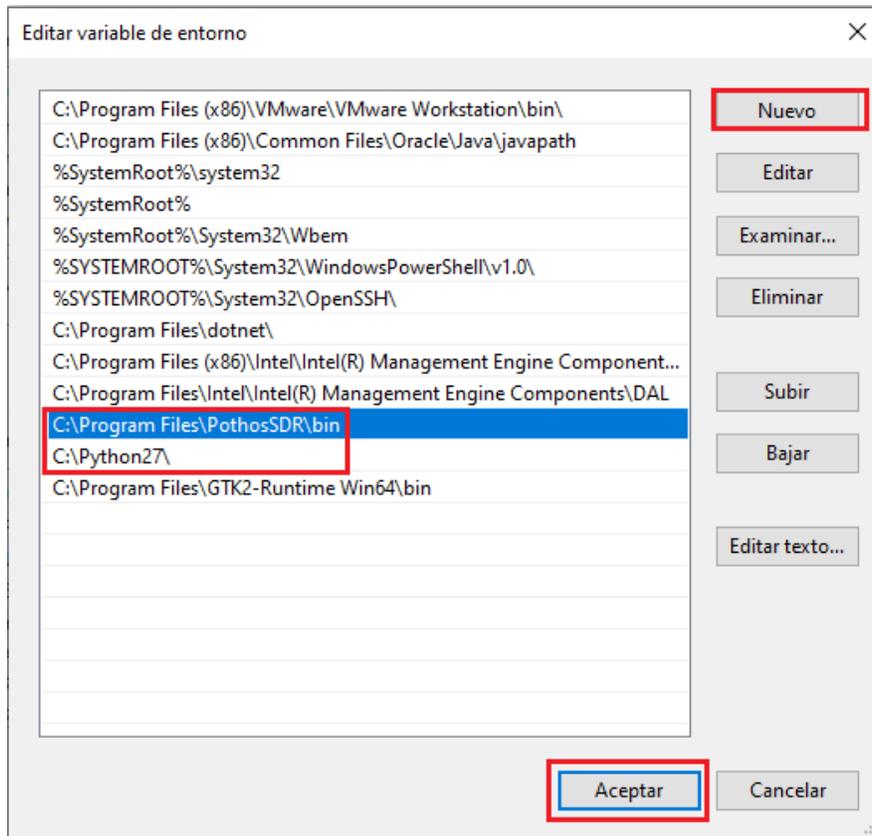


Fuente: Elaborado por el autor

Y finalmente, en la ventana de editar variable de entorno en la opción Nuevo, se agrega a Pothos SDR, de la siguiente manera, *C:\Program Files\PothosSDR\bin*. En la figura 83 se muestra la creación de la variable de entorno y se da clic en Aceptar para terminar.

Figura 83

Creación de variable de entorno PothosSDR



Fuente: Elaborado por el autor

5.6.2.2. Verificación De Instalación de GNU Radio

Una vez realizada la instalación de GNU Radio y además añadir Pothos SDR a la ruta del sistema, se realiza la verificación, para ello se debe dar doble clic en el ícono creado para la ejecución de GNU Radio. En la figura 84 se muestra el arranque exitoso de GNU Radio junto con la instalación automática de librerías adicionales.

Figura 84

Ejecución de GNU Radio e instalación automática de librerías

```
C:\Python27\python.exe
Successfully installed PyOpenGL-accelerate-3.1.1
You are using pip version 18.1, however version 20.3.4 is available.
You should consider upgrading via the 'python -m pip install --upgrade pip' command.
Done!
Handling issues for IMPORT_GR...
Error: GNURadio modules missing from PYTHONPATH

Current search path:
* C:\Program Files\PothosSDR\bin
* C:\Program Files\PothosSDR\lib\python2.7\site-packages
* C:\Windows\SYSTEM32\python27.zip
* C:\Python27\DLLs
* C:\Python27\lib
* C:\Python27\lib\plat-win
* C:\Python27\lib\lib-tk
* C:\Python27
* C:\Python27\lib\site-packages

Current PYTHONPATH: ''
Adding C:\Program Files\PothosSDR\lib\python2.7\site-packages to the PYTHONPATH

The PYTHONPATH for the current user has been modified
Open a new command window and re-run this script...

Changes made! Please re-run this script in a new terminal.

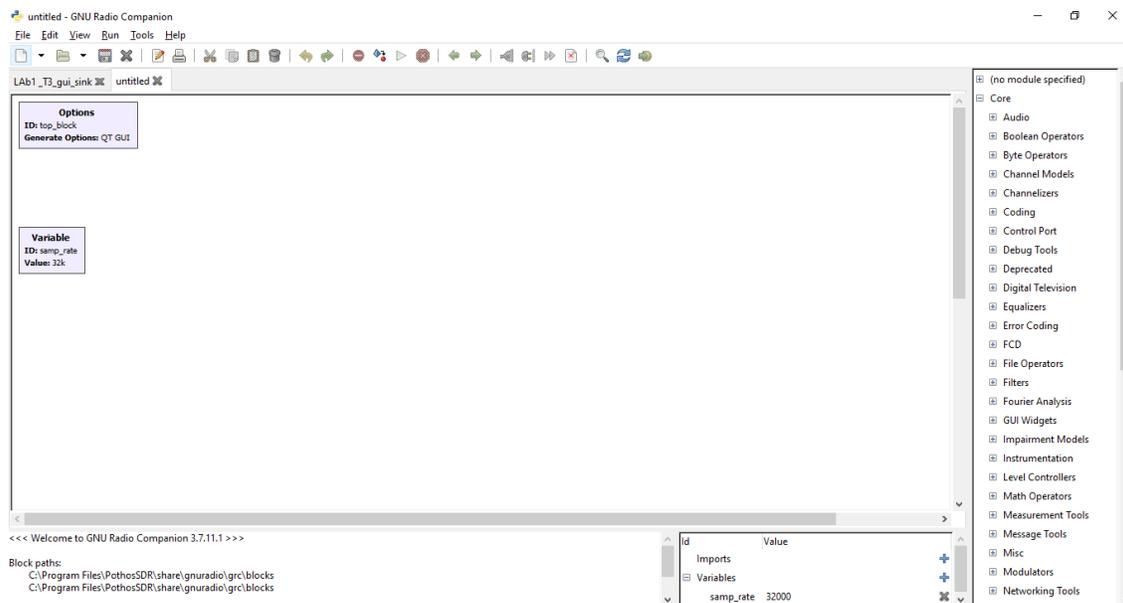
Presione una tecla para continuar . . .
```

Fuente: Elaborado por el autor

Después de finalizar la instalación de las librerías, se puede observar en la figura 85 la interfaz gráfica de usuario de GNU Radio, confirmando la instalación exitosa.

Figura 85

Interfaz gráfica de usuario GNU Radio



Fuente: Elaborado por el autor

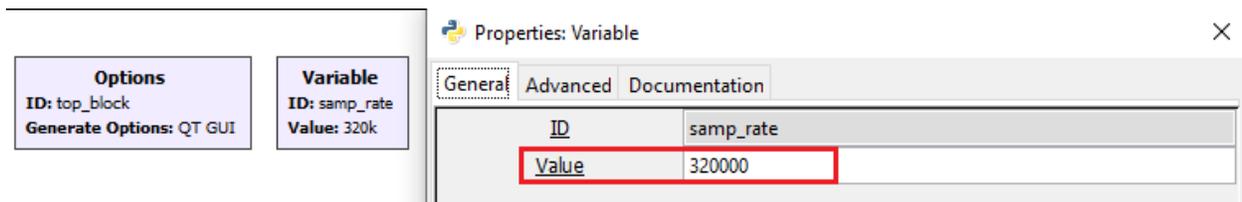
5.6.3. Creación De Diagrama De Flujo Para Transmisor QPSK

Ahora se procede a desarrollar un transmisor QPSK utilizando las capacidades de GNU Radio en conjunto con LimeSDR Mini el cual usa el complemento gr-limesdr para el software GNU Radio. De modo que, las fuentes (sources) y los receptores (sinks) de hardware sean bloques LimeSuite en el entorno gráfico de GNU Radio.

El primero paso consiste en crear un nuevo diagrama de flujo, en el cual se presentan de forma predeterminada dos bloques: el bloque llamado “Options” y el bloque “Variable”. En este último bloque es necesario ajustar el parámetro *samp_rate* (tasa de muestreo) en 320 kHz, para ello en el bloque Variable al hacer doble clic se abre una ventana de propiedades de la variable, en la cual se ajusta el parámetro. La figura 86 ilustra los bloques Options” y “Variable”.

Figura 86

Bloque options y Variable en GNU Radio



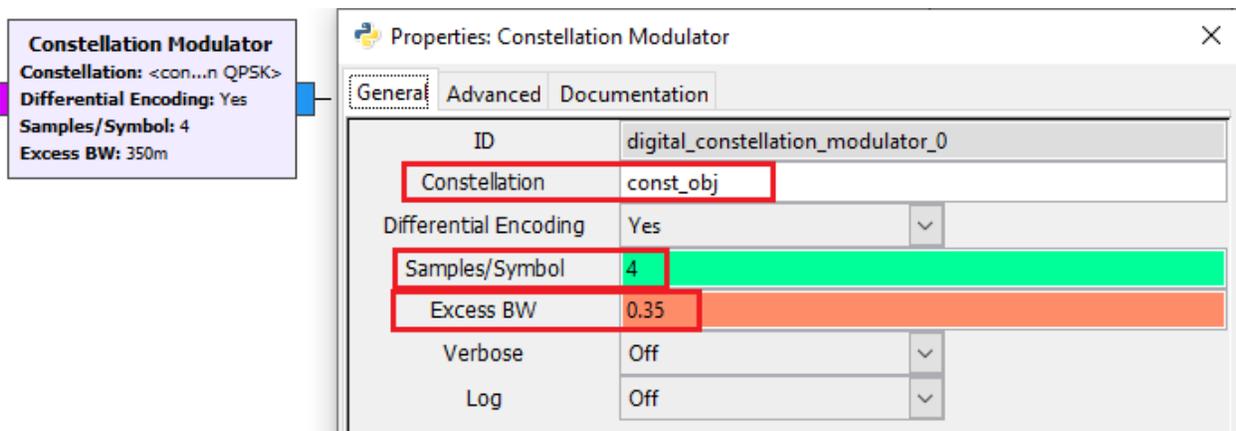
Fuente: Elaborado por el autor

El siguiente paso es insertar en el diagrama de flujo un bloque denominado “Constellation Modulator” (Modulador de Constelación) para realizar la modulación digital. Al hacer doble clic en el bloque se realizan las siguientes modificaciones: *Constellation* (constelación): *const_obj* y *Samples/Symbol* (Muestras/Símbolo): 4.

El bloque y las modificaciones realizadas en base a los parámetros establecidos se pueden observar en la figura 87. Además, de la propiedad “*Excess BW*” referente al factor de caída del filtro con el valor predeterminado en 0.35. Este debido a que el modulador de constelación usa un filtro de conformación de pulso de raíz cuadrada - Root Raised Cosine (RRC).

Figura 87

Bloque modulador de constelación



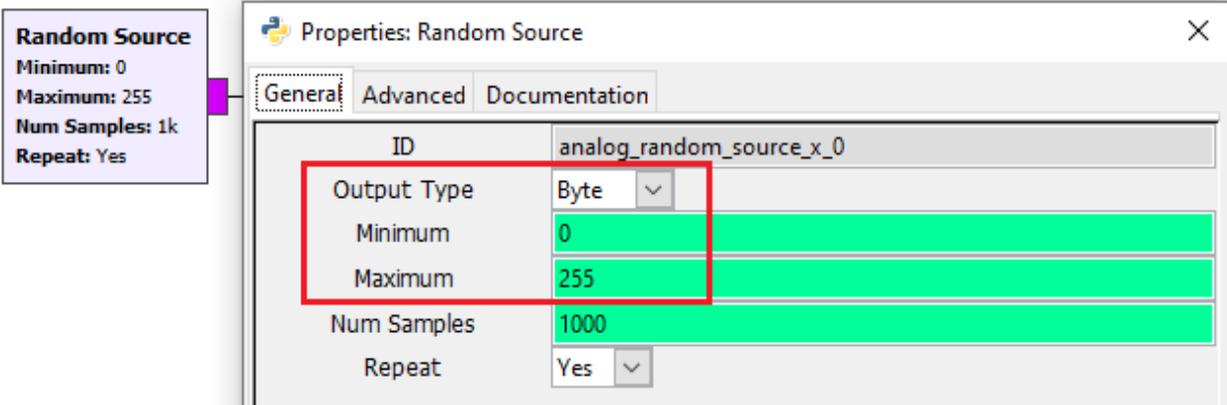
Fuente: Elaborado por el autor

A continuación, se debe agregar el bloque “*Random Source*” el cual es un generador de fuente aleatoria, debido a que el modulador de constelación requiere de bytes empaquetados. El generador proporciona bytes con valores de 0 a 255 por lo que se ubica antes del modulador de constelación. Al hacer doble clic en el bloque se realizan los siguientes cambios, los cuales se observan en la figura 88, estas modificaciones son las siguientes: *Output type (Tipo de salida):*

Byte, Minimum (Mínimo):0 y Maximum (Máximo): 255

Figura 88

Bloque generador de fuente aleatoria

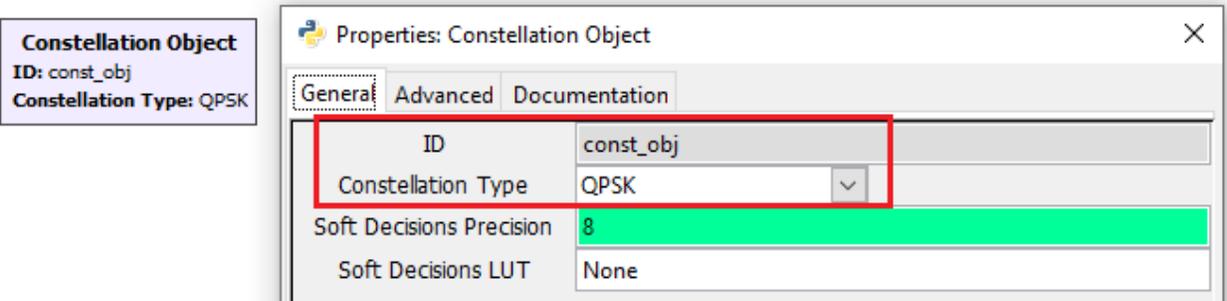


Fuente: Elaborado por el autor

El siguiente paso consiste en agregar en el diagrama de flujo un bloque llamado “*Constellation Object*” (*Objeto de Constelación*) este bloque es requerido por el modulador de constelación para detallar el tipo de modulación y el mapeo de símbolos a puntos de la constelación. En la figura 89 se puede observar las modificaciones de los parámetros respecto a: *ID: const_obj* y *Constellation Type (Tipo de Constelación): QPSK*.

Figura 89

Bloque objeto de constelación



Fuente: Elaborado por el autor

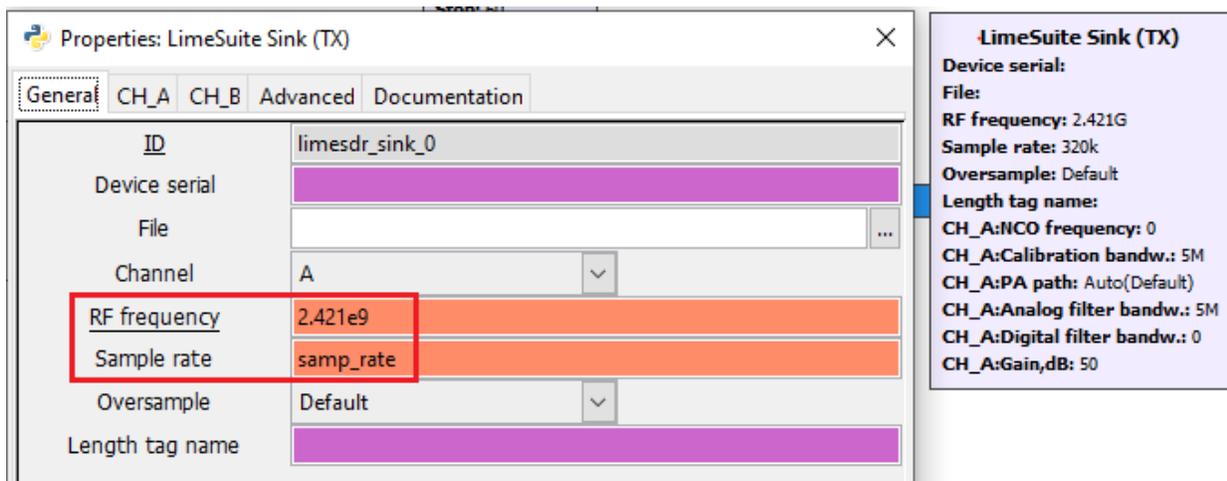
Ahora el siguiente paso, consiste en insertar un bloque denominado “*LimeSuite Sink*” ubicado después del bloque modulador de constelación con el fin de generar y transmitir la

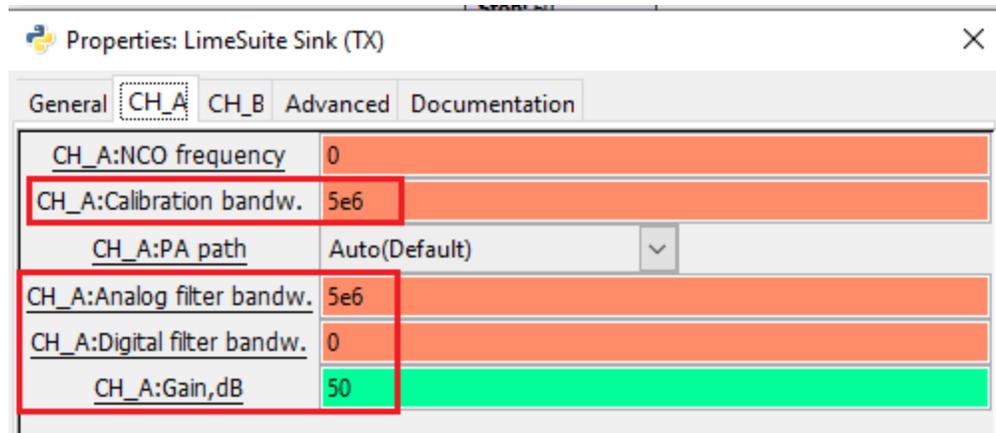
forma de onda QPSK modulada. Por lo tanto, en la figura 90 se puede observar los cambios realizados referente a los parámetros como se detalla a continuación:

- *Generate tab (pestaña generar)*
 - RF frequency (Frecuencia RF): $2.421e9$ (2.421 GHz)
 - Sample rate (Frecuencia de muestreo): *samp_rate*
- *CH_A tab (pestaña CH_A)*
 - Calibration bandw (Ancho de banda de calibración): $5e6$
 - Analog filter bandw (Ancho de banda del filtro analógico): $5e6$
 - Digital filter bandw (ancho de banda del filtro digital): 0
 - Gain, dB (Ganancia): *gain*

Figura 90

Bloque LimeSuite Sink





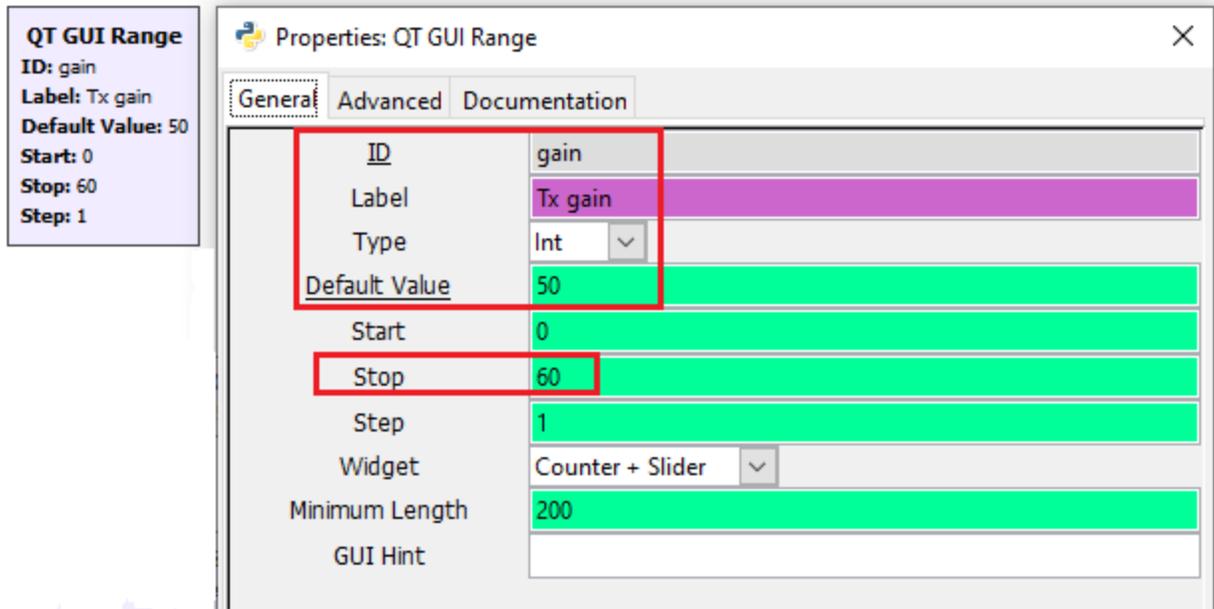
Fuente: Elaborado por el autor

El siguiente paso a ejecutar es añadir un bloque “*QT GUI Range*” que sirve para la creación de una variable con un control deslizante para poder realizar un ajuste de la ganancia del transmisor. De igual manera en la figura 91 se ilustran las modificaciones de los parámetros del bloque, los cuales se detallan a continuación:

- ID: *gain* (ganancia)
- Label (Etiqueta): *Tx Gain*
- Type (Tipo): *int*
- Default value (valor predeterminado): *50*
- Stop (Detener): *60*

Figura 91

Bloque QT GUI Range

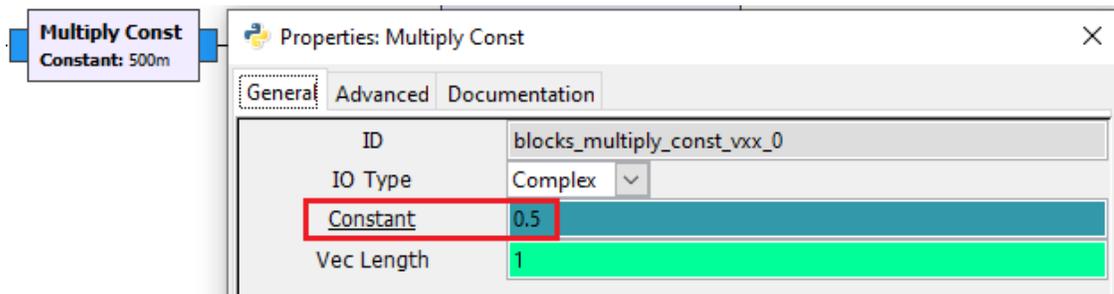


Fuente: Elaborado por el autor

Ahora es necesario agregar el bloque “*Multiply Const*” antes del bloque LimeSuite Sink con el objetivo de multiplicar la salida del bloque Modulador de constelación por 0.5. Es decir, el bloque LimeSuite Sink disminuirá los valores de entrada mayores a 1. En la figura 92 se puede observar el cambio realizado del parámetro del multiplicador. Finalmente, se debe conectar la salida del bloque Multiply const al bloque LimeSuite Sink.

Figura 92

Bloque Multiply const

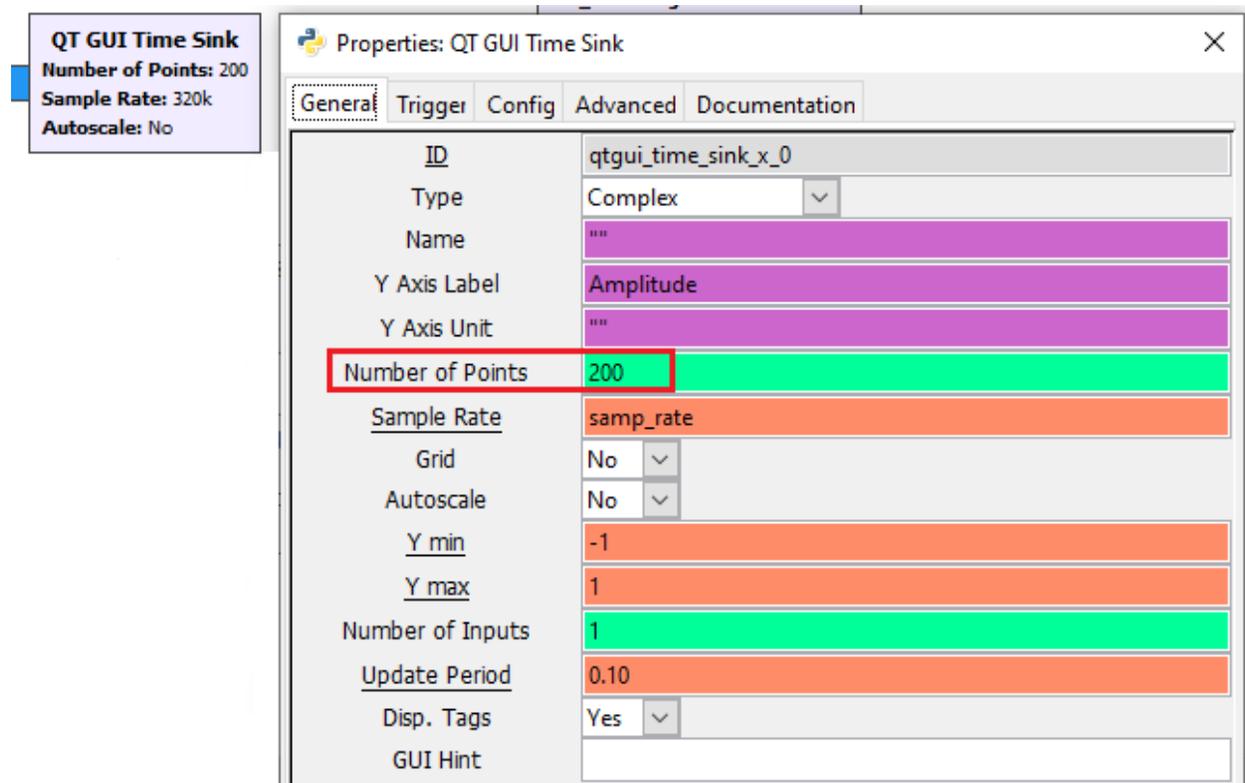


Fuente: Elaborado por el autor

El siguiente paso, consiste en agregar el bloque “*QT GUI Time Sink*” (*Disipador de tiempo de GUI de QT*) para mostrar la forma de onda en el dominio del tiempo de la señal modulada. En la figura 93 se ilustra al bloque y la modificación respectiva en el parámetro “*Number of Points*” (*número de puntos*) con un valor de 200.

Figura 93

Bloque QT GUI Time Sink

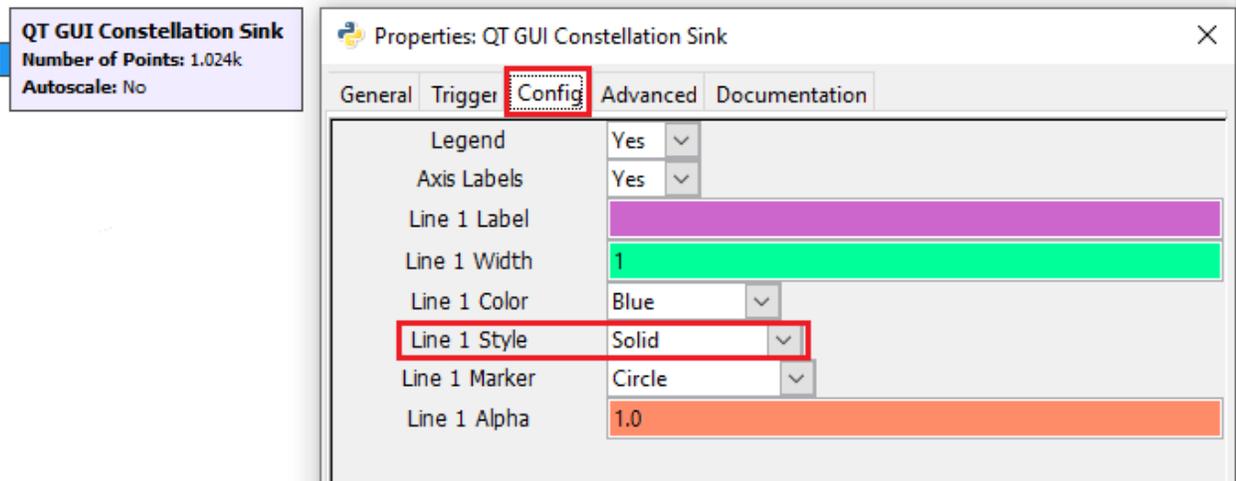


Fuente: Elaborado por el autor

Así mismo, se continúa con el siguiente paso y se agrega el bloque “*QT GUI Constellation Sink*” (*Disipador de constelación de GUI de QT*) para mostrar el diagrama de constelación de la señal modulada. En la figura 94 se muestra el bloque y la modificación respectiva en base a la pestaña “*Config*” en el cual se cambia el “*Line 1 Style*” a “*Solid*”.

Figura 94

Bloque QT GUI Constellation Sink

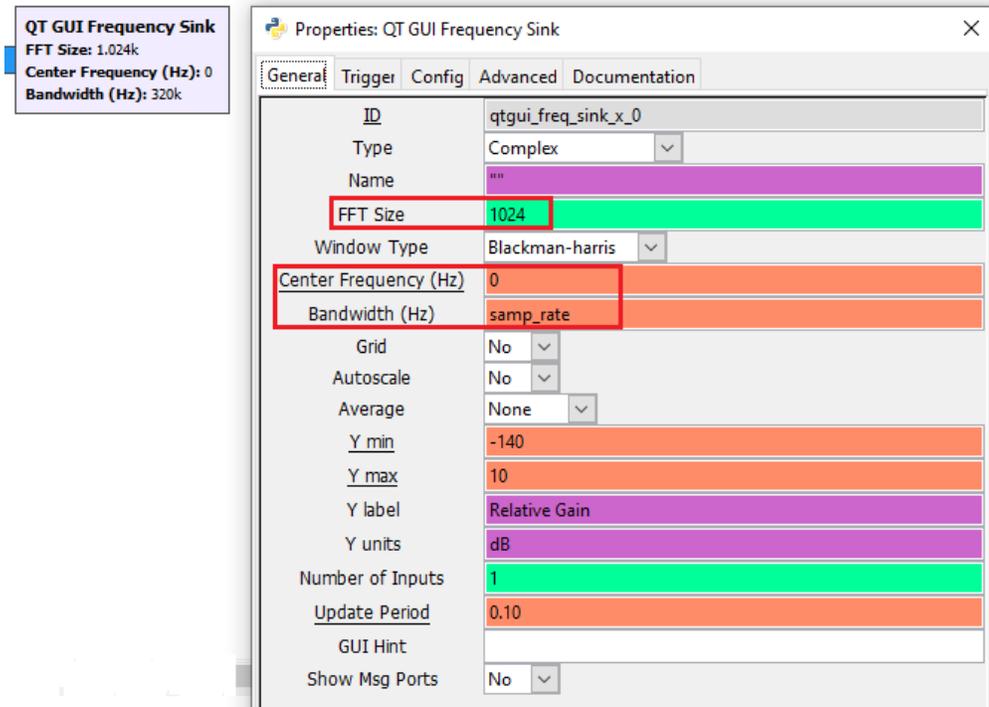


Fuente: Elaborado por el autor

Finalmente, se agrega el bloque “*QT GUI Frequency Sink*” (*Disipador de frecuencia de GUI de QT*) para mostrar el espectro de frecuencia de la señal modulada. En la figura 95 se muestra el bloque marcando los parámetros más importantes del mismo.

Figura 95

Bloque QT GUI Frequency Sink

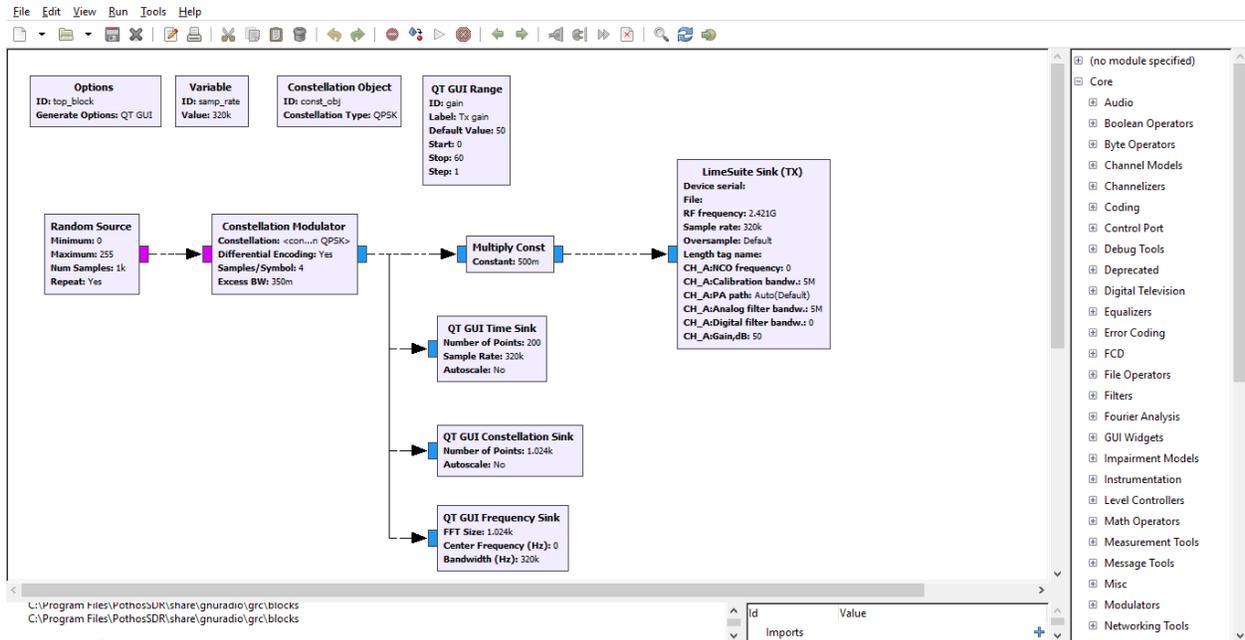


Fuente: Elaborado por el autor

Una vez agregados los diferentes bloques, se muestra el diagrama de flujo completo para el transmisor QPSK, este diagrama se puede apreciar en la figura 96.

Figura 96

Diagrama de flujo completo transmisor QPSK final



Fuente: Elaborado por el autor

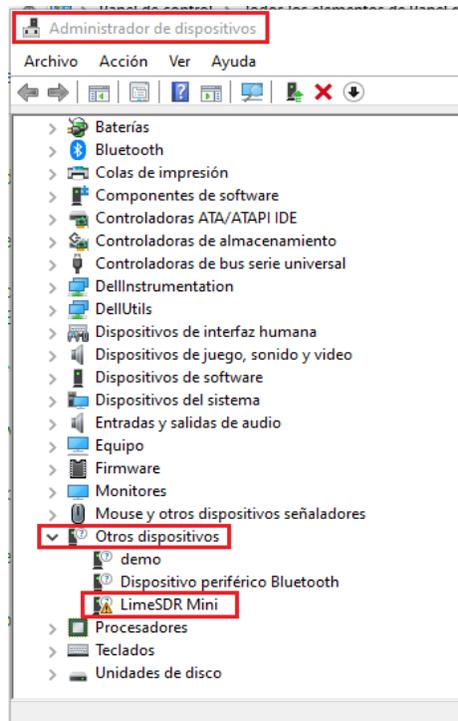
5.6.4. Conexión De LimeSDR Mini a PC

Al finalizar el diagrama de flujo para el transmisor QPSK se procede a conectar el LimeSDR Mini al puerto USB de la PC, por lo que se usa un cable USB macho a USB hembra.

Al realizar la conexión al ingresar al “Administrador de dispositivos” de la PC se determina, como se muestra en la figura 97 que el controlador del LimeSDR Mini presenta problemas para su correcto funcionamiento.

Figura 97

Ventana administrador de dispositivos LimeSDR Mini no disponible



Fuente: Elaborado por el autor

Para solucionar el problema, es necesario instalar un controlador nuevo, para ello se dirige al siguiente enlace <https://ftdichip.com/drivers/d3xx-drivers/> seleccionado el sistema operativo adecuado a la PC para la descarga.

Al realizar la instalación del controlador con el asistente del Administrador de dispositivos, en la figura 98, se muestra al controlador FTDI FT601 USB 3.0 Bridge Device, correctamente instalado.

Figura 98

Instalación correcta del controlador para LimeSDR Mini

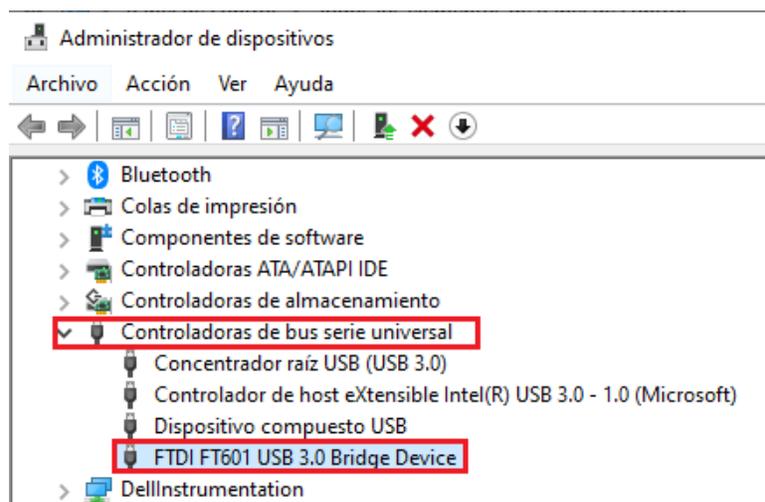


Fuente: Elaborado por el autor

Para verificar que el controlador para el LimeSDR Mini está funcionando correctamente se ingresa al Administrador de dispositivos, en la sección de *Controladoras de bus serie universal*, se puede comprobar tal como se observa en la figura 99 que el controlador se ha agregado y está listo para su funcionamiento.

Figura 99

Verificación de funcionamiento de controlador para LimeSDR Mini



Fuente: Elaborado por el autor

5.6.5. *Generación y Ejecución De Señal De Banda Base Modulada QPSK*

Una vez instalado el controlador para el funcionamiento del LimeSDR Mini, se procede a generar y ejecutar el diagrama de flujo de GNU Radio, con el fin de visualizar cómo los bloques funcionales interactúan para el transmisor QPSK, en la figura 100 se observa la barra de herramientas con las opciones para generar y ejecutar respectivamente el diagrama de flujo.

Figura 100

Barra de herramientas GNU para, generar y ejecutar diagrama de flujo



Fuente: Elaborado por el autor

Al generar el diagrama de flujo, en el terminal de GNU Radio, se presentan mensajes de estado generados por los bloques configurados, los cuales pueden indicar su estado interno, como las configuraciones actuales, tasas de muestreo o frecuencias. En la figura 101 se muestra los mensajes en el terminal al generar el diagrama de flujo.

Figura 101

Mensajes presentados en el terminal al generar el diagrama de flujo

```

LimeSuite Sink (TX) info
#####
Connecting to device
#####
Devices found: 1
#####
Device list:
Nr.:|0|device:|LimeSDR Mini, media=USB 3, module=FT601, serial=1D4C367CED6586, index=0
#####
INFO: device_handler::open_device(): no serial number. Using first device in the list. Use "LimeUtil --find" in terminal to find preferred device serial.
#####
Device number 0 from the list is used.
Reference clock 40.00 MHz
#####

INFO: device_handler::enable_channels(): SISO CH0 set for device number 0.
INFO: device_handler::set_samp_rate(): set sampling rate: 0.32 MS/s.
Selected TX path: Band 1
INFO: device_handler::set_rf_freq(): RF frequency set [TX]: 2421 MHz.
Filter calibrated. Filter order-4th, filter bandwidth set to 5 MHz.Real pole 1st order filter set to 2.5 MHz. Preemphasis filter not active
TX LPF configured
INFO: device_handler::set_analog_filter(): INFO: device_handler::set_gain(): set gain [TX] CH0: 50 dB.
Selected TX path: Band 1
INFO: device_handler::set_antenna(): CH0 antenna set [TX]: BAND1.
Tx calibration finished
C:\Users\Dell\AppData\Roaming\gnuradio\prefs\vmcircularbuf_default_factory: No such file or directory
gr::pagesize: no info; setting pagesize = 4096
INFO: device_handler::calibrate(): INFO: sink_impl::init_stream(): sink channel 0 (device nr. 0) stream setup done.

```

Id	Value
Imports	
Variables	
const_obj	<Open Properties>
gain	<Open Properties>
samp_rate	320000

Fuente: Elaborado por el autor

De igual manera al ejecutar el diagrama de flujo, en el terminal de GNU Radio se presentan mensajes de estado. En la figura 102 se observan los mensajes en el terminal al ejecutar el diagrama de flujo.

Figura 102

Mensajes presentados en el terminal al ejecutar el diagrama de flujo

```

Executing: C:\Python27\python.exe -u C:\Users\Dell\Documents\top_block.py
-----
LimeSuite Sink (TX) info
#####
Connecting to device
#####
Devices found: 0
#####
>>> Done

```

Id	Value
Imports	
Variables	
const_obj	<Open Properties>
gain	<Open Properties>
samp_rate	320000

Fuente: Elaborado por el autor

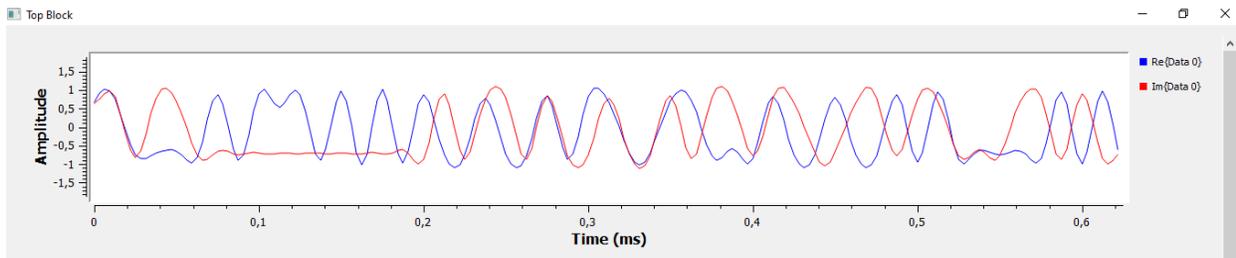
Finalmente, al generar y ejecutar correctamente el diagrama de flujo, se abre una nueva ventana con las gráficas de la señal de banda base modulada QPSK.

El bloque “*LimeSuite Sink (TX)*” es el encargado de generar y transmitir la forma de onda QPSK modulada. Debido a que brinda la capacidad de enviar señales moduladas a través del dispositivo LimeSDR con el uso de GNU Radio.

El bloque “*QT GUI Time Sink*” (*Disipador de tiempo de GUI de QT*) configurado previamente, tiene la función de mostrar la forma de onda en el dominio del tiempo, tal como se puede apreciar en la figura 103 que se muestra a continuación.

Figura 103

Gráfica de la forma de onda en el dominio del tiempo

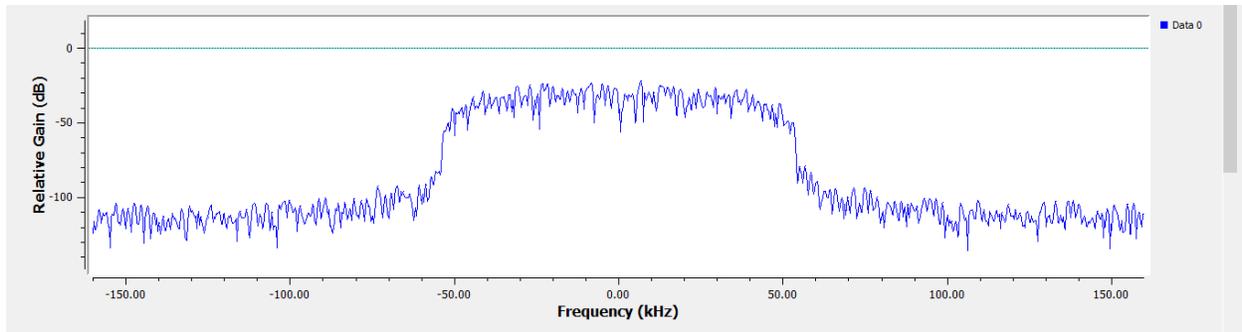


Fuente: Elaborado por el autor

Del mismo modo, el bloque “*QT GUI Frequency Sink*” (*Disipador de frecuencia de GUI de QT*) tiene la función de presentar en tiempo real el espectro de frecuencia de la señal modulada. En consecuencia, en la figura 104 se puede observar la representación gráfica de la señal en el dominio de la frecuencia.

Figura 104

Gráfica del espectro de frecuencia en tiempo real

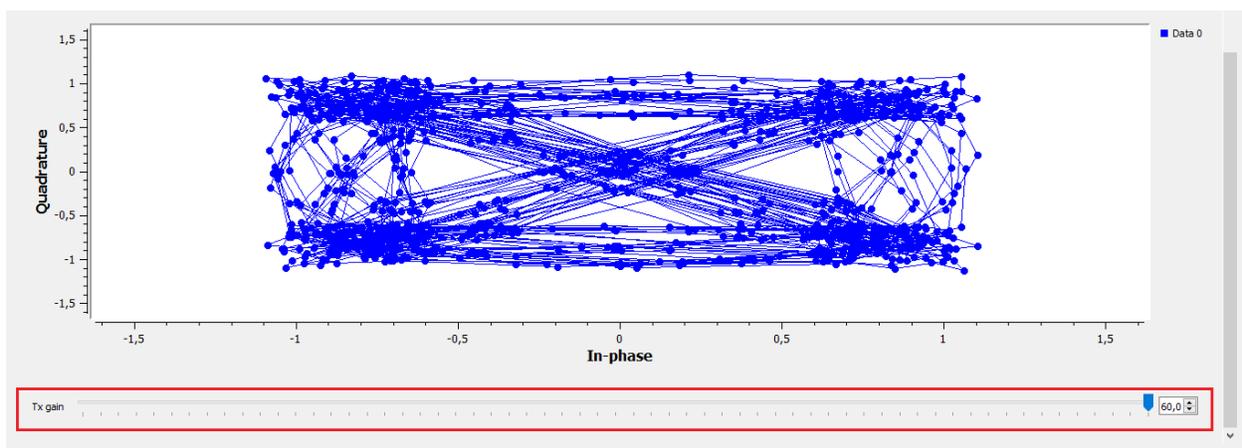


Fuente: Elaborado por el autor

Por último, los bloques “*QT GUI Constellation Sink*” (*Disipador de constelación de GUI de QT*) el cual muestra el diagrama de constelación de la señal modulada. Y el bloque “*QT GUI Range*” que facilita la creación de una variable mediante un control deslizante para ajustar la ganancia del transmisor. En la figura 105 presentada a continuación, se aprecia que cada punto en la constelación representa un símbolo transmitido y su ubicación se determina por la amplitud y la fase.

Figura 105

Gráfica diagrama de constelación de la señal modulada



Fuente: Elaborado por el autor

5.7. Guía 5: Analizador De Espectro De Radio Definido Por Software

El propósito de esta guía tiene como finalidad proporcionar instrucciones para crear un diagrama de flujo de un analizador de espectro de Radio Definido por Software. Esto mediante el adecuado funcionamiento de GNU Radio, previamente instalado, en combinación con el dispositivo LimeSDR Mini.

UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS
CARRERA DE INGENIERÍA EN ELECTRÓNICA Y REDES DE
COMUNICACIÓN



TEMA: Analizador de espectro de Radio Definido por Software

AUTOR: ROMEL RICARDO ROSERO

Esta guía se enfoca en el uso de GNU Radio como implementación de aplicaciones de Radio Definido por Software para la creación de un analizador de espectro, esto en conjunto con el uso del dispositivo LimeSDR Mini

Objetivos

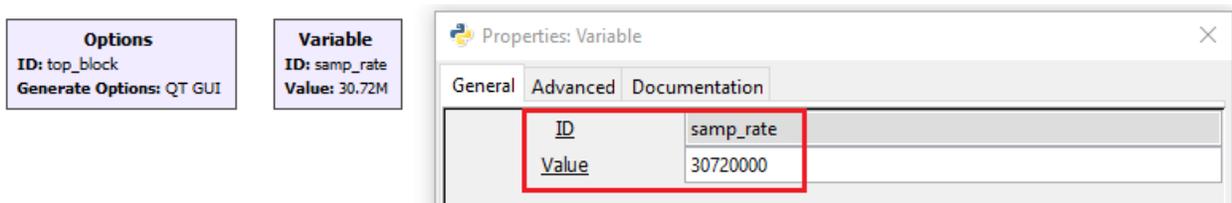
- Identificar los bloques correctos para elaborar el diagrama de flujo
- Crear un analizador de espectro de Radio Definido por Software con el uso de GNU Radio en combinación con LimeSDR Mini

5.7.1. Creación De Diagrama De Flujo Para Analizador De Espectro SDR

El primero paso consiste en crear un nuevo diagrama de flujo, de igual manera se presentan de forma predeterminada dos bloques: el bloque “Options” y el bloque “Variable”. En este último bloque es necesario ajustar el parámetro *samp_rate* en 30,72 MHz en el bloque Variable al hacer doble clic se abre la ventana de propiedades de la variable, en la cual se ajusta el valor del parámetro. La figura 106 muestra al bloque Options” y “Variable”.

Figura 106

Bloques Options, Variable y configuración de tasa de muestreo



Fuente: Elaborado por el autor

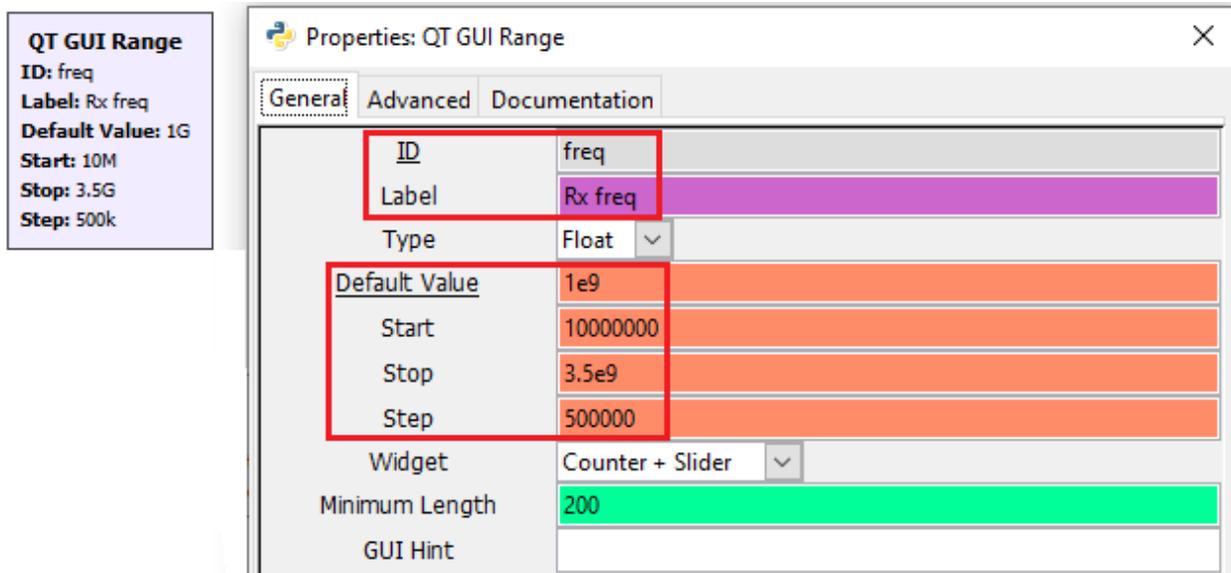
El siguiente paso consiste en añadir dos bloques “QT GUI Range” para la creación de dos variables ajustables con un control deslizante para realizar un ajuste para la ganancia y frecuencia. La figura 107 se observan las modificaciones de los parámetros del bloque QT GUI Range 1 (frecuencia), los cuales se detallan a continuación:

- *QT GUI Range 1 (frecuencia):*
 - ID: *freq (frecuencia)*
 - Label (Etiqueta): *Rx Freq*
 - Default value (valor predeterminado): *1e9*
 - Start (Iniciar): *10e6*
 - Stop (Detener): *3.5e9*

- Step (paso – cantidad en la que el valor del parámetro se incrementa o disminuye): $0.5e6$

Figura 107

Bloque QT GUI Range 1 (frecuencia)



Fuente: Elaborado por el autor

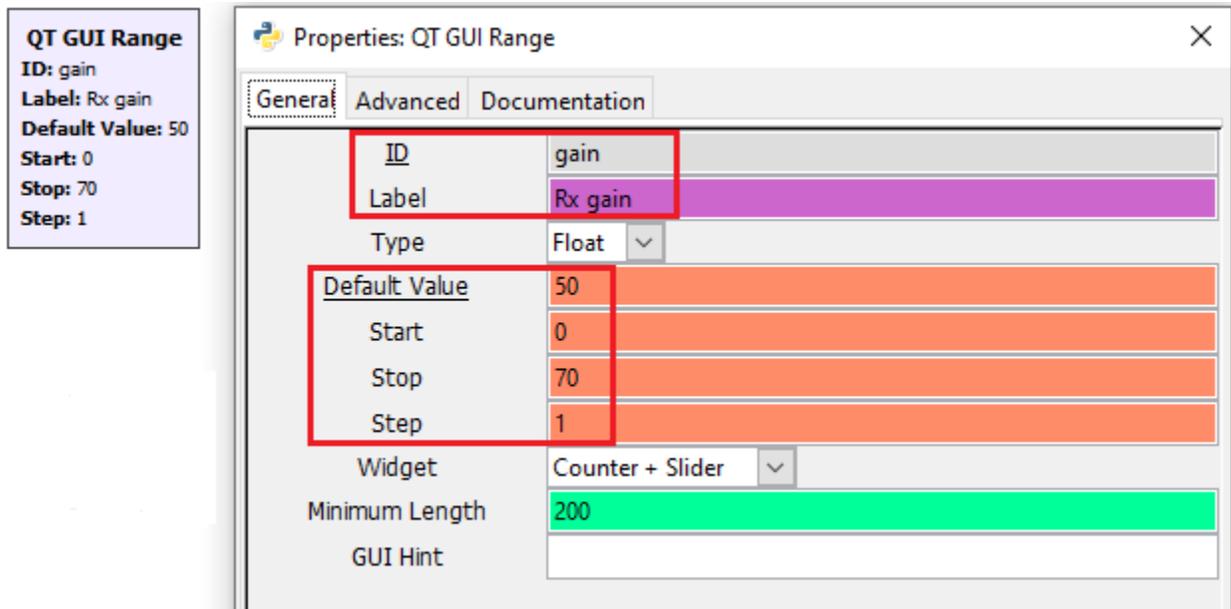
De la misma manera, en la figura 108 se puede apreciar las modificaciones de los parámetros del bloque QT GUI Range 2 (ganancia), detallados a continuación:

- *QT GUI Range 2 (ganancia):*
 - ID: *gain (ganancia)*
 - Label (Etiqueta): *Rx Gain*
 - Default value (valor predeterminado): *50*
 - Start (Iniciar): *0*
 - Stop (Detener): *70*

- Step (paso – cantidad en la que el valor del parámetro se incrementa o disminuye): 1

Figura 108

Bloque QT GUI Range 2 (ganancia)



Fuente: Elaborado por el autor

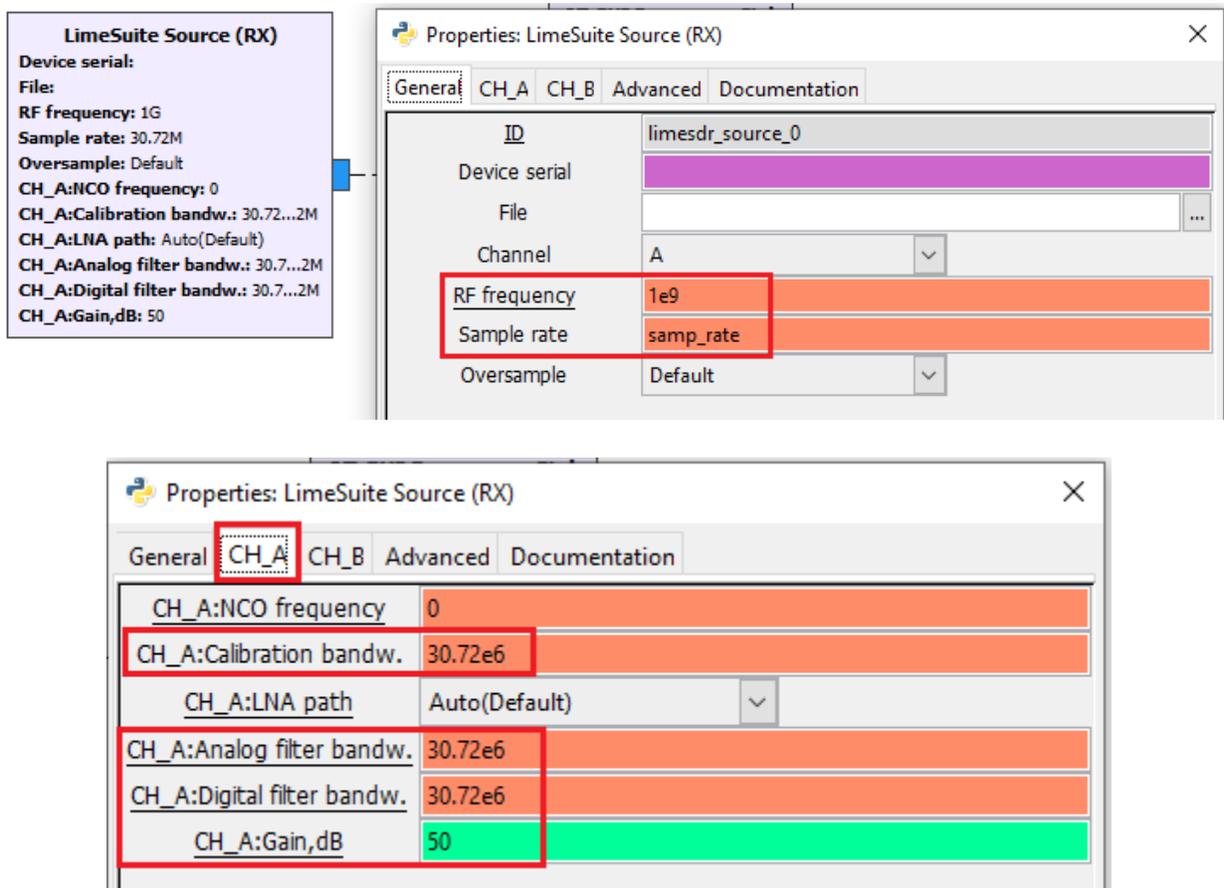
El siguiente paso, consiste en insertar el bloque “*LimeSuite Source (Rx)*” que sirve para recibir señales de radio desde el dispositivo LimeSDR. en el que se deben usar los ID de los bloques QT GUI Range. Por lo tanto, en la figura 109 se observa las configuraciones realizadas referente a los parámetros detallados a continuación:

- *Generate tab (pestaña General)*
 - RF frequency (Frecuencia RF): *freq*
 - Sample rate (Frecuencia de muestreo): *samp_rate*
- *CH_A tab (pestaña CH_A)*

- Calibration bandw (Ancho de banda de calibración): *samp_rate*
- Analog filter bandw (Ancho de banda del filtro analógico): *samp_rate*
- Digital filter bandw (ancho de banda del filtro digital): *samp_rate*
- Gain, dB (Ganancia): *gain*

Figura 109

Bloque LimeSuite Source



Fuente: Elaborado por el autor

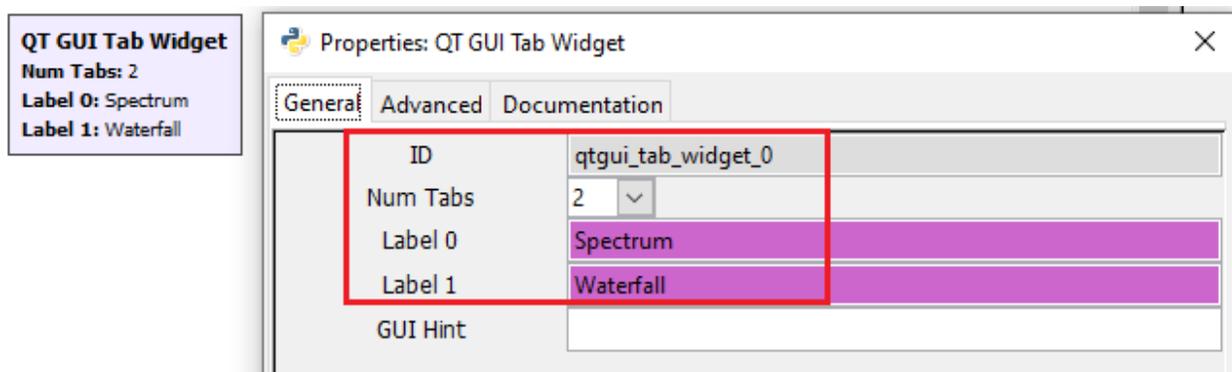
Continuado con el desarrollo, el siguiente paso ahora corresponde agregar el bloque “*QT GUI Tab Widget*” que sirve para crear los widgets de pestañas. En la figura 110 se puede

observar los parámetros a configurar para las pestañas. Estos parámetros se detallan a continuación.

- ID: *tab*
- Num Tabs (número de pestañas): 2
- Label 0 (Etiqueta 0): *Spectrum*
- Label 1 (Etiqueta 1): *Waterfall*

Figura 110

Bloque QT GUI Tab Widget



Fuente: Elaborado por el autor

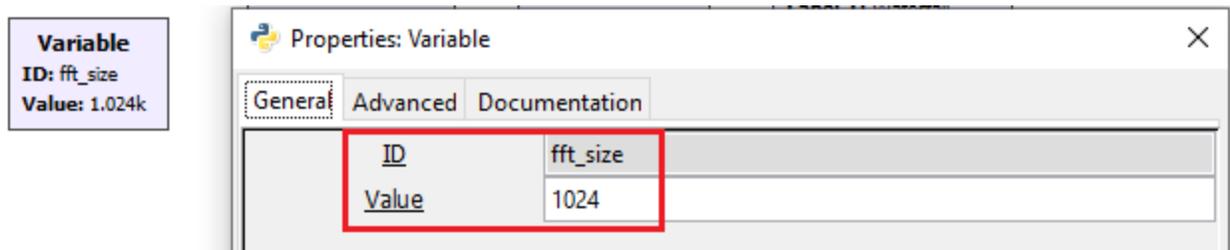
A continuación, se procede a añadir otro bloque “*Variable*” para referirse al tamaño de la transformada de Fourier de la señal en el dominio del tiempo. Es decir, se usa para convertir una señal en el dominio del tiempo a su representación en el dominio de la frecuencia. En la figura 111 se muestra la configuración del parámetro *fft_size* para determinar la cantidad de puntos de datos que se usan en la transformada. La configuración para estos parámetros se detalla a continuación.

- ID: *fft_size*

- Value (valor): 1024

Figura 111

Bloque Variable para tamaño de la transformada de Fourier



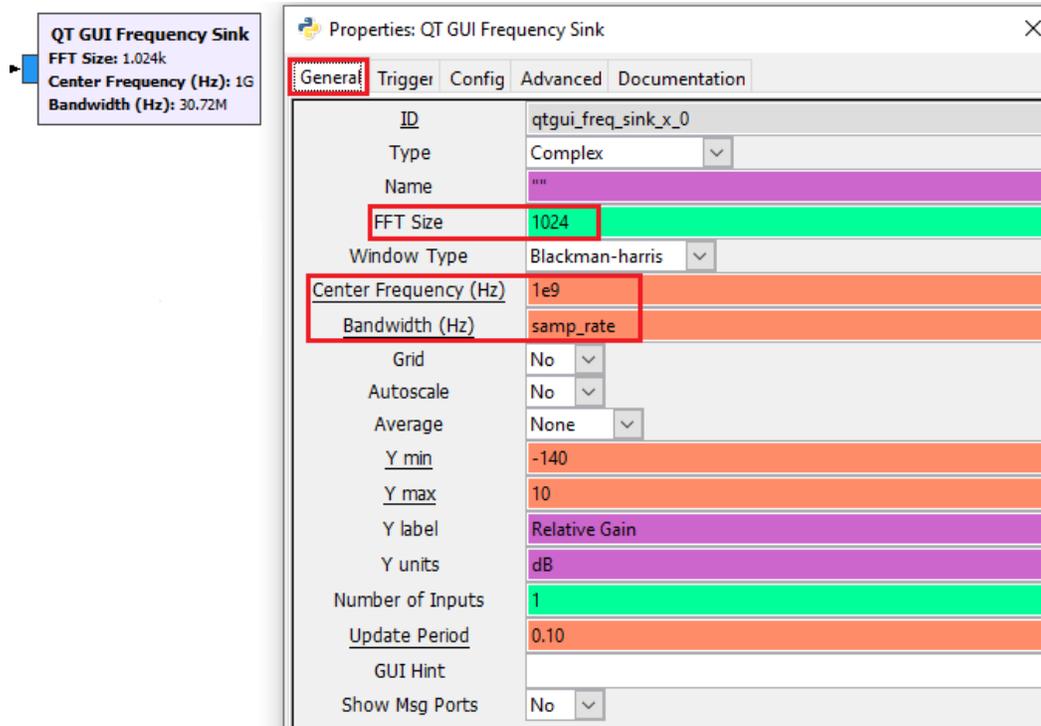
Fuente: Elaborado por el autor

De inmediato, se agrega también el bloque “*QT GUI Frequency Sink*” (*Disipador de frecuencia de GUI de QT*) para mostrar el espectro de frecuencia de la señal modulada. Por lo tanto, la salida del bloque LimeSuite Source (RX) debe ir conectado a la entrada del bloque QT GUI Frequency Sink. En la figura 112 se muestra el bloque con los parámetros a configurar, los cuales son detallados.

- *General tab (pestaña General)*
 - FFT size (Tamaño de FFT): *fft_size*
 - Center frequency (Frecuencia central) Hz: *freq*
 - Bandwidth (Ancho de banda) Hz: *samp_rate*
 - GUI hint (sugerencia de GUI): *tab@0*
- *Config tab (pestaña de configuración)*
 - Control panel (panel de control): *yes*

Figura 112

Bloque QT GUI Frequency Sink



Fuente: Elaborado por el autor

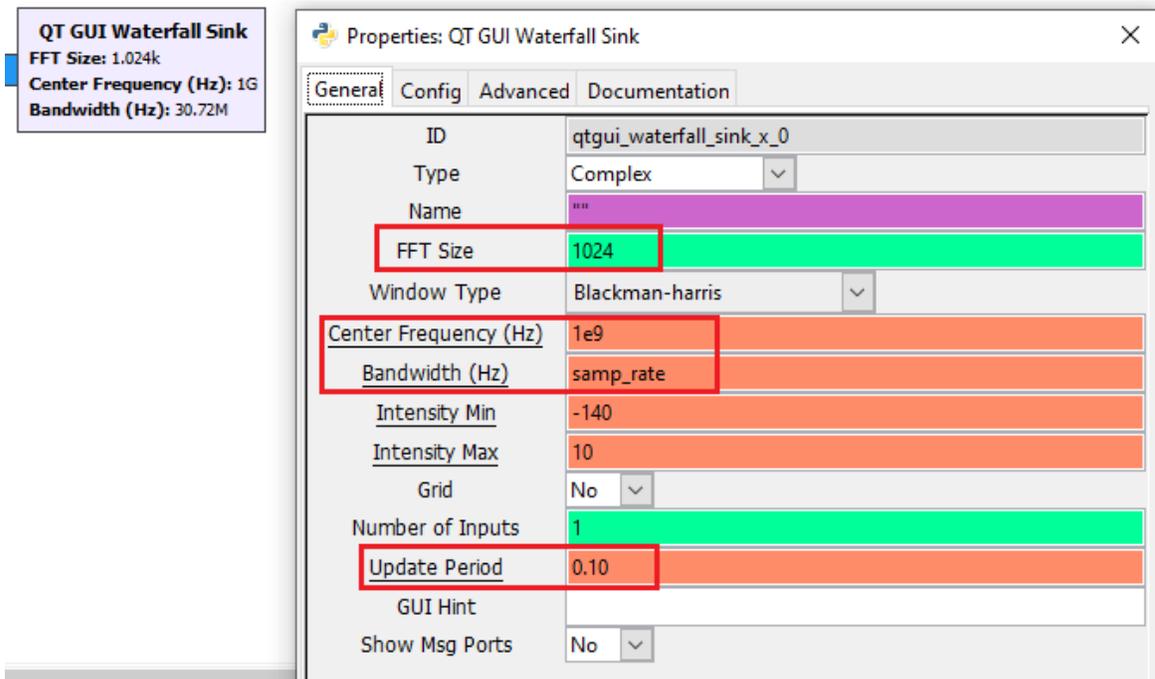
Para concluir, se agrega también el bloque “*QT GUI Waterfall Sink*” (*Desplazamiento de cascada GUI de QT*) el cual se usa para obtener una visualización de tipo cascada de la señal en el dominio de la frecuencia a lo largo del tiempo. En este sentido, la salida del bloque LimeSuite Source (RX) debe ir conectado a la entrada del bloque QT GUI Waterfall Sink. La figura 113 da a conocer el bloque con los parámetros a configurar, los cuales son detallados de la misma manera.

- *General tab (pestaña General)*
 - FFT size (Tamaño de FFT): *fft_size*
 - Center frequency (Frecuencia central) Hz: *freq*
 - Bandwidth (Ancho de banda) Hz: *samp_rate*
 - Update period (Período de actualización): *0.0005*

- GUI hint (sugerencia de GUI): *tab@1*

Figura 113

Bloque QT GUI Waterfall Sink

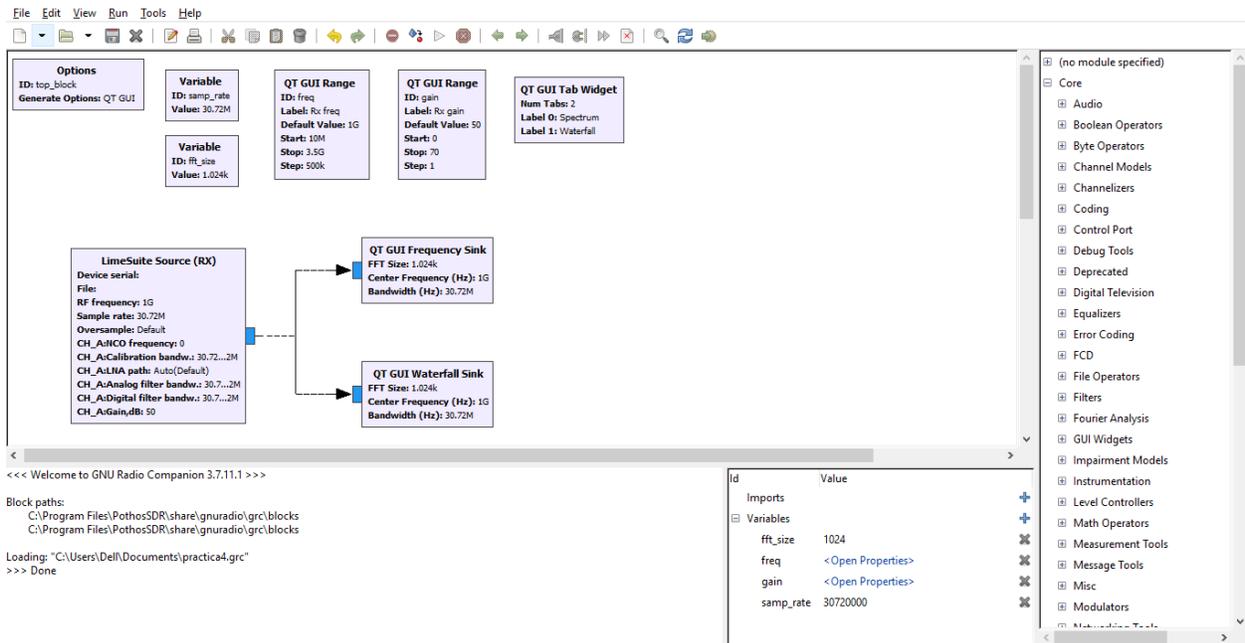


Fuente: Elaborado por el autor

Una vez agregados todos los diferentes bloques con la configuración de sus parámetros respectivos, se muestra el diagrama de flujo completo para el analizador de espectro de Radio Definido por Software, este diagrama se puede apreciar en la figura 114.

Figura 114

Diagrama de flujo completo analizador de espectro de SDR



Fuente: Elaborado por el autor

5.7.2. Generación y Ejecución Del Analizador De Espectro SDR

Al finalizar el diagrama de flujo para el analizador de espectro SDR se conecta de igual manera el LimeSDR Mini al puerto USB de la PC, como en la Guía 4 conectado el cable USB macho a USB hembra.

Al estar en correcto funcionamiento el controlador para el LimeSDR Mini, se procede a generar y ejecutar el diagrama de flujo de GNU Radio, con el objetivo de visualizar a los bloques funcionales interactuando para el analizador de espectro de SDR.

Ahora se procede a generar el diagrama de flujo y en el terminal de GNU Radio, se presentan mensajes de estado presentados por los bloques configurados, mostrando tanto su estado interno, como configuraciones actuales, tasas de muestreo o frecuencias. En la figura 115 se observan los mensajes en el terminal al generar el diagrama de flujo.

Figura 115

Mensajes mostrados en el terminal al generar el diagrama de flujo

```
<<< Welcome to GNU Radio Companion 3.7.11.1 >>>

Block paths:
  C:\Program Files\PothosSDR\share\gnuradio\grc\blocks
  C:\Program Files\PothosSDR\share\gnuradio\grc\blocks

Loading: "C:\Users\Dell\Documents\practica4_2.grc"
>>> Done

Generating: 'C:\Users\Dell\Documents\top_block.py'
```

Id	Value
Imports	
Variables	
fft_size	1024
freq	<Open Properties>
gain	<Open Properties>
samp_rate	30720000

Fuente: Elaborado por el autor

Además, al ejecutar el diagrama de flujo, se presentan mensajes de estado en el terminal de GNU Radio. En la figura 116 exhibe los mensajes que aparecen en el terminal al ejecutar el diagrama de flujo.

Figura 116

Mensajes mostrados en el terminal al ejecutar el diagrama de flujo

```
Generating: 'C:\Users\Dell\Documents\top_block.py'
Executing: C:\Python27\python.exe -u C:\Users\Dell\Documents\top_block.py

-----
LimeSuite Source (RX) info
#####
Connecting to device
#####
Devices found: 1
#####
Device list:
Nr.:0|device:|LimeSDR Mini, media=USB 3, module=FT601, serial=1D4C367CED6586, index=0
#####
INFO: device_handler::open_device(): no serial number. Using first device in the list. Use "LimeUtil --find" in terminal to find preferred device serial.
#####
Device number 0 from the list is used.
Reference clock 40.00 MHz
#####

INFO: device_handler::enable_channels(): SISO CH0 set for device number 0.
INFO: device_handler::set_samp_rate(): set sampling rate: 30.72 MS/s.
Selected RX path: LNAW
INFO: device_handler::set_rf_freq(): RF frequency set [RX]: 1000 MHz.
RX LPF configured
GFIR LPF cannot be set to the requested bandwidth
INFO: device_handler::set_analog_filter(): INFO: device_handler::set_digital_filter(): digital filter CH0 [RX]: 30.72 MHz.
INFO: device_handler::set_gain(): set gain [RX] CH0: 50 dB.
Selected RX path: LNAW
INFO: device_handler::set_antenna(): CH0 antenna set [RX]: LNAW.
Rx calibration finished
gr::pagesize: no info; setting pagesize = 4096
INFO: device_handler::calibrate(): INFO: source_impl::init_stream(): source channel 0 (device nr. 0) stream setup done.
```

Id	Value
Imports	
Variables	
fft_size	1024
freq	<Open Properties>
gain	<Open Properties>
samp_rate	30720000

Fuente: Elaborado por el autor

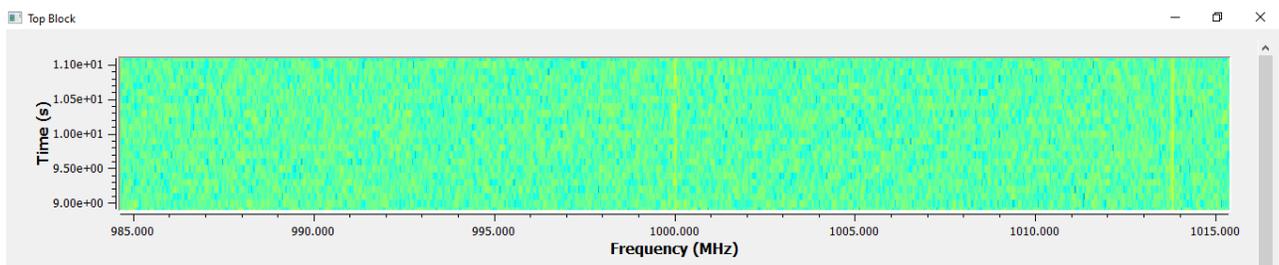
Al generar y ejecutar de manera correcta el diagrama de flujo, se abre una nueva ventana con las gráficas del espectro.

En este sentido, el bloque “*LimeSuite Source (RX)*” es el encargado de recibir la señal de radio. Brinda la capacidad de capturar señales a través del dispositivo LimeSDR con el uso de GNU Radio.

En la figura 117 se muestra la representación gráfica de la señal de radio recibida, tomando en cuenta los parámetros configurados anteriormente, en base a la frecuencia central de recepción, la tasa de muestreo, el ancho de banda de calibración, el ancho de banda del filtro analógico para eliminar componentes no deseados de la señal, el ancho de banda del filtro digital y el nivel de ganancia aplicada a la señal recibida. Estos ajustes son indispensables para optimizar la calidad de la recepción y que el dispositivo LimeSDR capture de manera adecuada la señal.

Figura 117

Gráfica de la señal de radio capturada



Fuente: Elaborado por el autor

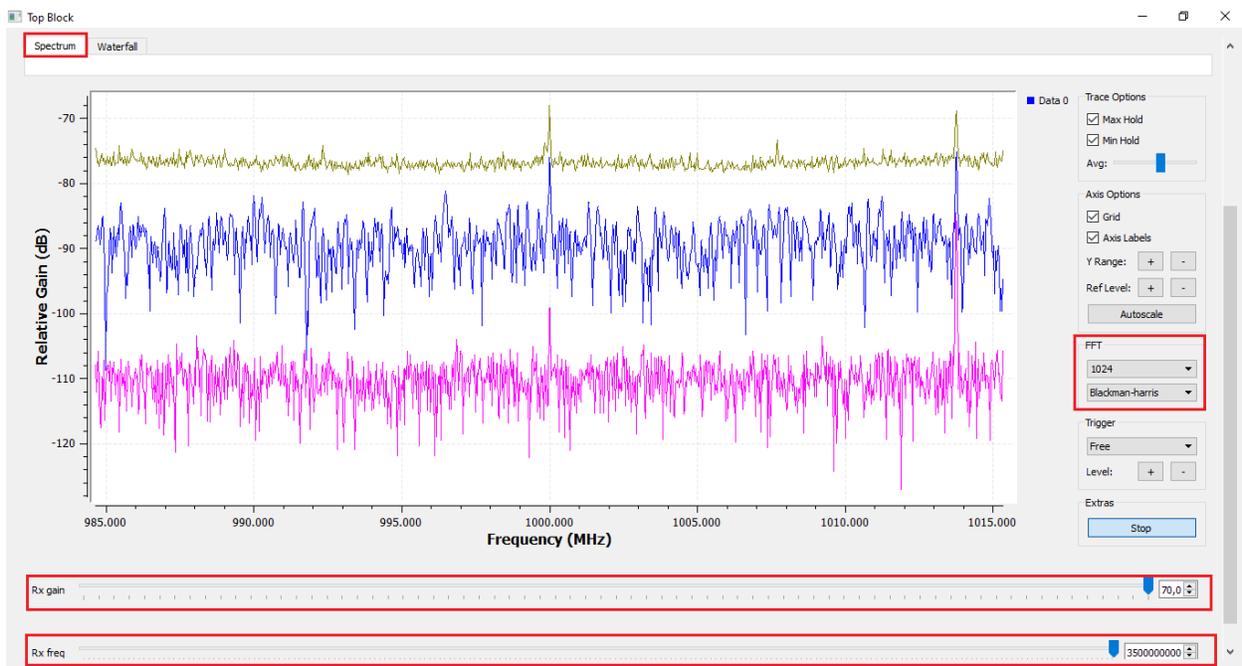
El bloque “*QT GUI Frequency Sink*” (*Disipador de frecuencia de GUI de QT*) presenta el espectro de frecuencia en tiempo real. Este a su vez y debido al bloque “*QT GUI Tab Widget*” que sirve para crear widgets de pestañas, se le denomina como Spectrum (Espectro).

En consecuencia, en la figura 118 se puede observar la representación gráfica del espectro de frecuencia, considerando los parámetros configurados previamente referente al tamaño de la transformada de Fourier, la frecuencia central y el ancho de banda, ajustados a los valores especificados.

Además, gracias a los bloques “*QT GUI Range 1 (frecuencia)*” y “*QT GUI Range 2 (ganancia)*” se muestra un panel de control deslizante que permite al usuario realizar ajustes en tiempo real para la visualización del espectro.

Figura 118

Gráfica del espectro de frecuencia



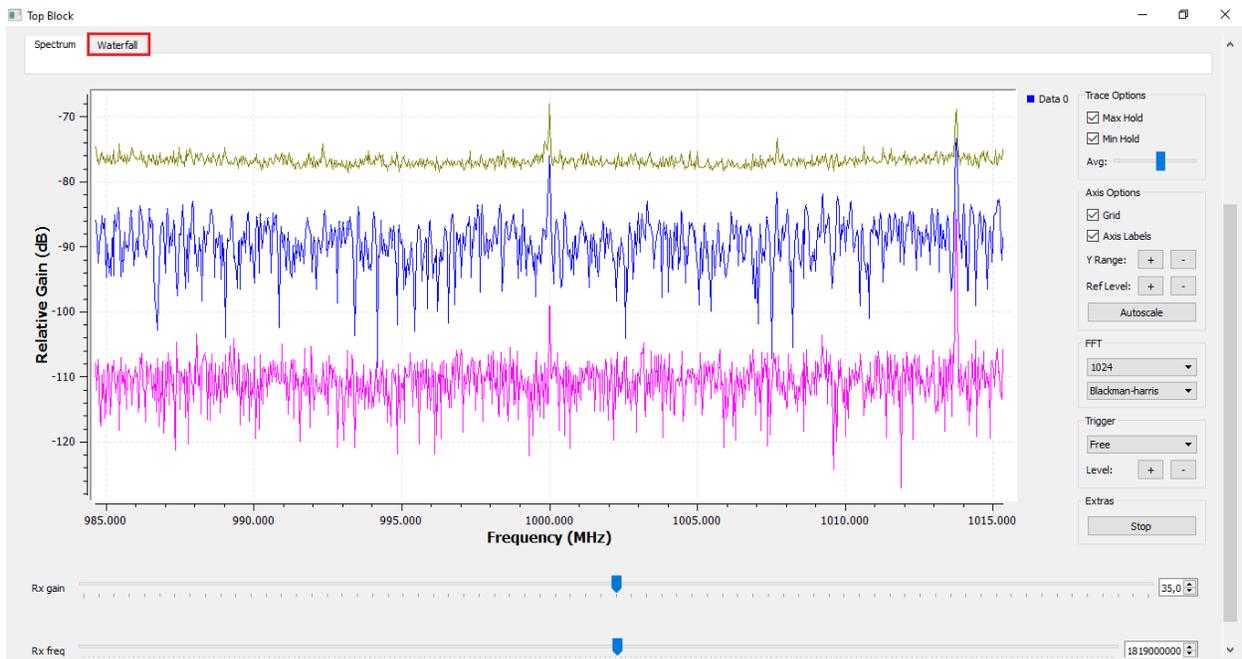
Fuente: Elaborado por el autor

Finalmente, el bloque “*QT GUI Waterfall Sink*” (*Receptor de cascada de GUI de QT*) presenta una visualización en cascada de espectrogramas de la señal en tiempo real. De igual manera, debido al bloque “*QT GUI Tab Widget*”, se denomina como Waterfall (Espectrograma).

Por lo tanto, en la figura 119 se exhibe la representación gráfica del espectrograma, con los parámetros configurados anteriormente referente al tamaño de la transformada de Fourier, la frecuencia central, el ancho de banda y el período de actualización.

Figura 119

Gráfica del espectrograma (Waterfall)



Fuente: Elaborado por el autor

Conclusiones

El sistema desplegado en este proyecto se ha integrado correctamente en el entorno del laboratorio de la asignatura de Radio Definido por Software (SDR) sin afectar a los equipos de LTE ya existentes, para un entorno didáctico, es decir que se pueda experimentar, observar y modificar sus parámetros de configuración y como interactúan sus componentes.

En este proyecto, se ha llevado a cabo el despliegue de un sistema de recepción de señales LTE 4G, utilizando la plataforma de código abierto OpenAirInterface (OAI), la cual proporciona dos módulos de software esenciales para la red de acceso radio (E-UTRAN), también conocida como una red distribuida de estaciones base (eNodeB). Es decir, el módulo openairinterface5g es el encargado de proporcionar el software para el despliegue del eNodeB, así como la emulación de un equipo de usuario (UE).

En lo que respecta a la implementación del núcleo de red (EPC), de manera similar, con el uso de la plataforma OAI, específicamente a través del módulo openair-cn que es el encargado del EPC, para la gestión del tráfico de red. Este módulo ofrece el software para la implementación de las funcionalidades del HSS, MME y S-PGW. Esto en base a la arquitectura de red LTE.

Para la realización de las guías referente al osciloscopio y mediciones de red, respectivamente, es posible modificar el parámetro de la configuración del eNodeB específicamente del PRB (Bloque de Recursos Físicos) establecido como la unidad mínima de recursos que el eNodeB asigna al UE, con el fin de analizar el comportamiento de la red.

En lo que respecta al rendimiento de la red, se observa que las velocidades de datos medidas son significativamente más bajas que las velocidades estimadas, determinando que la

diferencia aumenta cuando el ancho de banda es mayor, esta variación puede deberse a cambios en el índice MCS (Esquema de Modulación y Codificación) o a las condiciones de la red en el campus universitario durante la medición.

En cuanto a la realización de la guía de análisis del protocolo LTE se ha logrado capturar el tráfico del plano de control como del plano de usuario, así como también en la interfaz de radio e interfaz S1. Con el fin de observar y analizar los procedimientos de la red LTE, con los protocolos MAC, RLC, RRC y NAS. Además de analizar el tráfico en el plano de usuario con el fin de evaluar el rendimiento de la red.

Mediante la familiarización y uso de GNU Radio Companion para la implementación de aplicaciones de Radio Definido por Software, se ha logrado adquirir una comprensión sólida de los fundamentos. A través de la práctica y la exploración de los bloques existentes en GNU Radio, con el fin de construir tanto el transmisor QSPK como también con el objetivo de capturar y analizar el espectro de LTE, con el funcionamiento en conjunto de GNU Radio y LimeSDR Mini. Estos logros reflejan la capacidad para aplicar de manera efectiva la herramienta en este campo de la tecnología.

Recomendaciones

La tecnología SDR, ofrece la posibilidad de desarrollar aplicaciones para implementar prototipos de redes 4G LTE. En este contexto, OpenAirInterface se destaca como una solución de código abierto que ofrece documentación, soporte y actualizaciones. Esta elección proporciona una base sólida y respaldada para la implementación de prototipos de redes LTE, lo que facilita el desarrollo y la evolución de soluciones en este campo.

Además, OpenAirInterface se destaca como una implementación altamente adecuada para entornos docentes y de investigación debido a que incluye características para su estudio en laboratorios, así como para la aplicación práctica. Garantiza la inclusión de nuevas características y la adaptación a los distintos Release (versiones) de las especificaciones LTE.

Es importante tener en cuenta las modificaciones necesarias para configurar los parámetros tanto del eNodeB como del EPC, respectivamente de cada componente, estas modificaciones se realizan a través de los ficheros de texto (scripts) proporcionados por la plataforma OAI.

En este contexto, es esencial asegurarse de ajustar adecuadamente los parámetros para lograr un funcionamiento óptimo y personalizado de los componentes de la red LTE 4G basada en la plataforma. Esto garantizará que el despliegue se adapte de manera efectiva a las necesidades específicas y maximice su eficiencia.

En lo que respecta a la configuración del Dongle USB LTE, es fundamental destacar que se debe evitar actualizar el firmware del Dongle a su versión más reciente, ya que esto provocaría su inoperatividad. Las versiones de firmware más recientes carecen de la Interfaz de Programación de Aplicaciones (API) necesaria para gestionar el módem.

Con la ayuda de GNU Radio y el entorno de desarrollo Pothos SDR, se ofrece a los usuarios, en este caso de Windows la posibilidad de explorar y desarrollar con hardware y software SDR convencionales. Esto elimina la necesidad de descargar y configurar paquetes de software, lo que simplifica en gran medida el proceso de trabajar con tecnologías SDR en sistemas Windows.

El uso de GNU Radio es altamente recomendado, ya que esta herramienta permite a los usuarios diseñar, simular e implementar Radio Definido por Software de alta capacidad en el mundo real. Proporciona a los usuarios la posibilidad de crear archivos de Python de manera gráfica en lugar de crearlos en código. Al poseer una biblioteca completa de bloques de procesamiento, se pueden combinar para crear aplicaciones de procesamiento de señales.

6. Bibliografía

- Tafur Arenazas, C. A. (2017). *Diseño de una estación base 4G LTE para la mejora de los servicios multimedia del distrito de Huanza provincia de Huarochirí.*
- 3GPP TS 36.213. (2017). *LTE; Evolved Universal Terrestrial Radio Access (E-UTRA); Physical layer procedures.* European Telecommunications Standards Institute.
- Agusti Comes, R., Bernardo Álvarez, F., Casadevall Palacio, F., Ferrús Ferre, R., Pérez Romero, J., y Sallent Roig, O. (2010). *LTE: Nuevas Tendencias en Comunicaciones Móviles.* Fundación Vodafone España.
- Airbrake. (15 de December de 2016). *Iterative Model: What Is It And When Should You Use It?*
<https://blog.airbrake.io/blog/sdlc/iterative-model>
- Alvarez Paredes, D. C. (2014). “*DISEÑO DE UNA RED 4G (LONG TERM EVOLUTION) EN EL CLÚSTER 2 DE LA CIUDAD DE QUITO PARA EL OPERADOR CNT EP.*”.
- Cardona, N., Olmos, J. J., García, M., y Monserrat, J. (2011). *3GPP LTE: Hacia la 4G móvil.* Marcombo.
- Chica Pedraza, G. A., Díaz Sánchez, Á. F., Espinosa Buitriago, M., y Angulo Esguerra, D. N. (2020). *Implementación de estación base GSM recepción de señales LTE aplicando radio definido por software.* Bucaramanga: Revista ITECKNE.
- Codon, M. N. (2019). *Implementación de red LTE con Software-Defined Radio.* Universidad Nacional de San Martín.

- Damini , R., y Abhishek , D. (2017). *LTE PRINCIPLES AND OPTIMIZATION (A 4G WIRELESS TECHNOLOGY)*. International Journal of Engineering Sciences & Research Technology.
- Demel, J. (2012). *Empfang von LTE-Signalen in GNU Radio*. KARLSRUHE INSTITUTE OF TECHNOLOGY, Communications Engineering Lab.
- DreamCatcher. (2020). *ME1130 Essentials of LTE and LTE - A courseware. Quick Start Guide*.
- DreamCatcher. (2020). *ME1130 Essentials of LTE and LTE-Advanced*.
- DreamCatcher Courseware . (2020). *University Courseware Teaching Lab Exercises about LTE Technologies*.
- Escobar Chamba, A. N. (2021). *Diseño de una estación base para su integración en una red celular con tecnología UMTS y LTE* .
- GNU Radio. (2023). *GNU Radio*. <https://www.gnuradio.org/about/>
- Gutierrez, D., Gimenez, F., Zerbini, C., y Riva, G. (2020). *Measurement of 4G LTE Cells with SDR*. IEEE LATIN AMERICA TRANSACTIONS.
- Holma, H., y Toskala, A. (2009). *LTE for UMTS – OFDMA and SC-FDMA Based Radio Access*. John Wiley & Sons Ltd.
- Huawei Device Co., Ltd. (s.f.). *HUAWEI 4G Dongle E3372*.
<https://consumer.huawei.com/en/support/routers/e3372/>
- Huawei Technologies Co., Ltd. (2010). *LTE Air Interface Training Manual*.

- ITU. (2008). *ITU-T E.212 International Mobile, shared codes*. https://www.itu.int/net/ITU-T/inrdb/e212_901.aspx
- Martínez García, D. (2018). *Despliegue de un prototipo de red 4G-LTE con OperAirInterface para entorno didáctico. Proyecto Fin de Carrera / Trabajo Fin de Grado, E.T.S.I. y Sistemas de Telecomunicación (UPM)*. Madrid.
- MyriadRF. (16 de February de 2023). *LimeSDR-Mini*. <https://wiki.myriadrf.org/LimeSDR-Mini>
- MyriadRF. (16 de February de 2023). *LimeSDR-USB*. <https://wiki.myriadrf.org/LimeSDR-USB>
- OpenAirInterface. (2023). *OpenAirInterface - 5G software alliance for democratising wireless innovation*. <https://openairinterface.org/>
- OpenAirInterface5G. (2016). *IttiAnalyzer WireShark Interface - OpenAirInterface Wiki - GitLab*. <https://gitlab.eurecom.fr/oai/openairinterface5g/-/wikis/IttiAnalyzer>
- Penttinen, J. (2015). *The Telecommunications Handbook: Engineering Guidelines for Fixed, Mobile and Satellite Systems*. John Wiley & Sons, Ltd.
- Rumney, M. (2013). *LTE and the Evolution to 4G Wireless Design and Measurement Challenges, Second Edition*. John Wiley & Sons, Ltd.
- SDR Forum. (08 de Noviembre de 2007). *SDRF Cognitive Radio Definitions*. http://www.sdrforum.org/pages/documentLibrary/documents/SDRF-06-R-0011-V1_0_0.pdf
- Sesia, S., Toufik, I., y Baker, M. (2009). *LTE – The UMTS Long Term Evolution From Theory to Practice*. John Wiley & Sons Ltd.

W3schools® of Technology. (2009). *SDLC Iterative Model*.

<https://www.w3schools.in/sdlc/iterative-model>

Wipro Technologies. (Agosto de 2002). *White Paper: Software-Defined Radio - A Technology*

Overview. https://www.ab4oj.com/dl/sdr_wipro.pdf

Xiang, X. (2018). *Estudio de plataformas SDR para LTE-5G*.

7. Anexos

7.1. Anexo 1. Encuesta de Stakeholders

Encuesta para la obtención de Requerimientos de Stakeholders

El objetivo de la presente encuesta es para recopilar datos e información relevante para la determinación de los requerimientos de Stakeholders, para el desarrollo de mi tema de tesis enfocado en Radio Definido por Software (SDR) con el fin del uso en prácticas de laboratorio de la misma.

Hola, ROMEL RICARDO. Cuando envíe este formulario, el propietario verá su nombre y dirección de correo electrónico.

* Obligatorio

1. ¿Cree Ud. que hace falta más herramientas, prototipos o aplicaciones educativas y didácticas para la materia de Radio Definido por Software (SDR) en prácticas de laboratorio y a la vez que tenga la disponibilidad de funcionamiento continuo para el desarrollo de las mismas? *

SI

NO

2. ¿Considera que el sistema con el uso de SDR como interfaz de RF (Radio Frecuencia) de LTE, sería útil para el fortalecimiento de conocimientos y de la investigación como las principales características de LTE y la arquitectura general del sistema? *

SI

NO

3. ¿Estaría de acuerdo en que el hardware de LTE se pueda utilizar para simulación o como implementación en tiempo real de la red LTE para que los estudiantes conozcan como configurar y conectar un equipo de usuario a la red LTE? *

SI

NO

4. ¿Cree que el sistema LTE pueda reforzar los conocimientos enfocados en temas de la materia como el análisis de señales en dominios de tiempo y frecuencia? * 

SI

NO

5. Mediante el uso del sistema junto con aplicaciones para el análisis del espectro de RF para dispositivos SDR de software libre, como por ejemplo, GNU Radio u OpenAirInterface, ¿cree usted que sea considerable desarrollar y comprender el procesamiento de señales con el fin de proporcionar capacidades de análisis de espectro LTE? * 

SI

NO

6. ¿Considera Ud. que el sistema sea didáctico, de fácil entendimiento y brinde al usuario una comprensión práctica en la tecnología LTE? * 

SI

NO

7. Cómo calificaría el nivel de aprendizaje que proporcionaría el sistema, con el que los usuarios exploren los protocolos y a su vez aprendan a construir una red LTE de manera independiente. * 

Malo

Regular

Bueno

Excelente

8. Cómo calificaría el nivel de aprendizaje que proporcionaría el sistema, con el que los usuarios aprenderán a configurar y conectar un Equipo de Usuario a la red LTE para obtener información sobre el procesamiento de señales en tiempo real y los protocolos de LTE. * 

- Malo
- Regular
- Bueno
- Excelente

9. Detalle las dificultades que usted considera sean fundamentales en cuanto al uso que puedan encontrarse en el sistema y señale de ser el caso las recomendaciones que se deberían implementar para mejorar el mismo * 

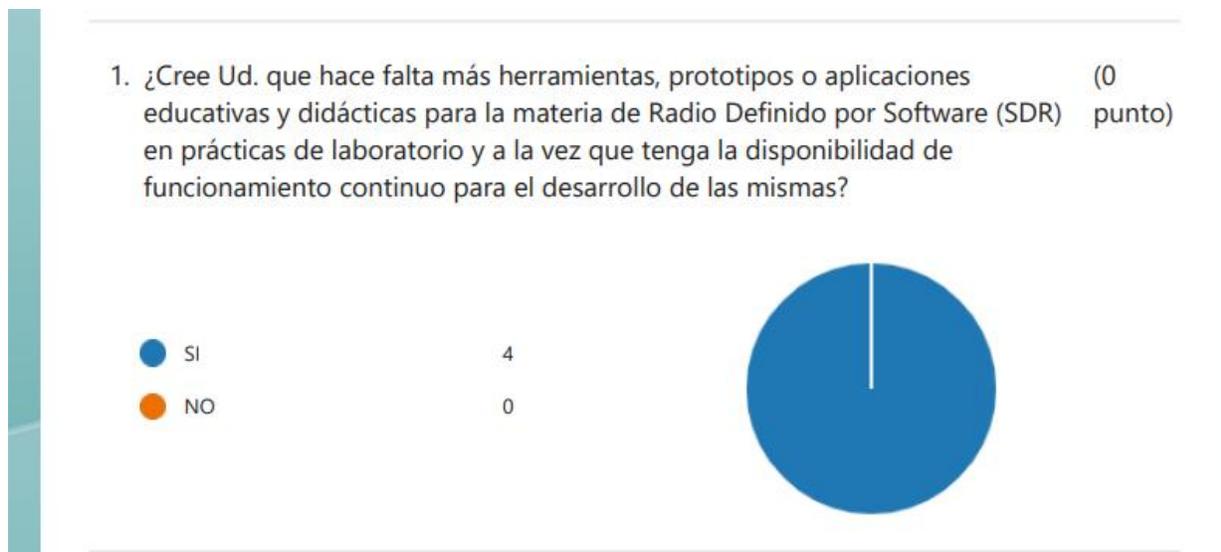
Escriba su respuesta

7.2. Anexo 2. Tabulación de Encuesta

En la primera pregunta se solicitó a los encuestados si consideran que hace más falta herramientas, prototipos o aplicaciones educativas y didácticas para la materia de Radio Definido por Software (SDR) en prácticas de laboratorio y a la vez que tenga la disponibilidad de funcionamiento continuo, en este caso el 100% especificó que si se requiere ya sea de herramientas, prototipos o aplicaciones para la asignatura de SDR. Estos resultados se muestran en el gráfico.

Figura 120

Gráfico de tabulación de pregunta 1

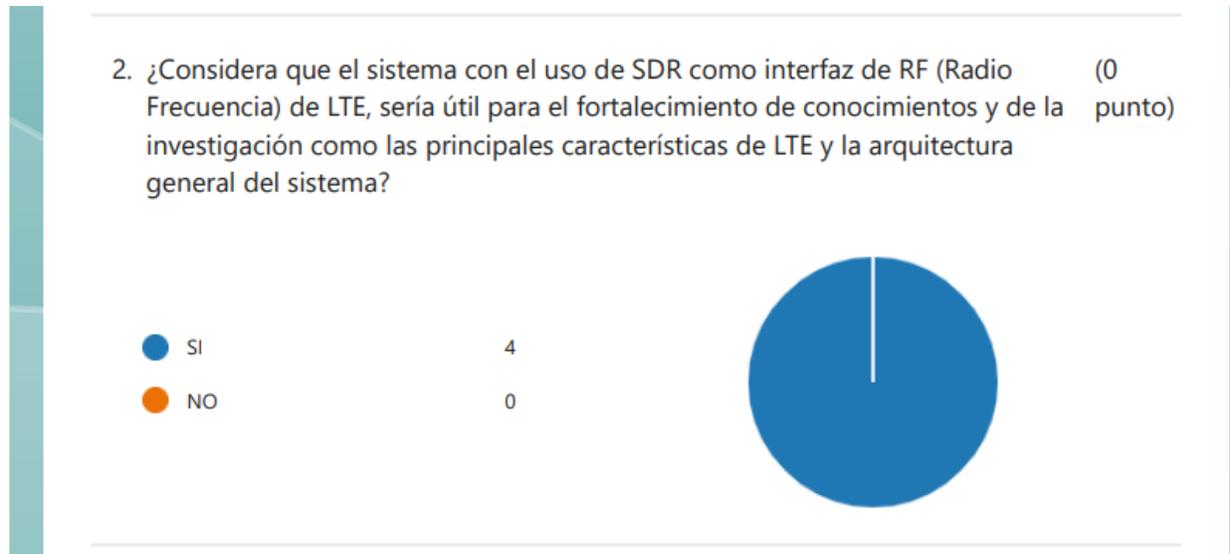


Fuente: elaborado por el autor

En la siguiente pregunta realizada, el 100% de la muestra concuerda que el sistema con el uso de SDR como interfaz de Radio Frecuencia de LTE, si será útil para el fortalecimiento de conocimientos y a su vez de la investigación en base a las características de LTE y la arquitectura del sistema. Estos resultados se muestran en el gráfico.

Figura 121

Gráfico de tabulación de pregunta 2



Fuente: elaborado por el autor

En el proceso de la encuesta, los participantes al 100% de la muestra, está de acuerdo en que el hardware de LTE se pueda utilizar para la simulación o implementación en tiempo real de la red LTE, teniendo como objetivo que los estudiantes conozcan como configurar y conectar un equipo de usuario a la red LTE. A continuación, se muestra el gráfico de los resultados.

Figura 122

Gráfico de tabulación de pregunta 3

3. ¿Estaría de acuerdo en que el hardware de LTE se pueda utilizar para simulación o como implementación en tiempo real de la red LTE para que los estudiantes conozcan como configurar y conectar un equipo de usuario a la red LTE? (0 punto)



Fuente: elaborado por el autor

La siguiente pregunta realizada fue enfocada a los conocimientos que se puedan aportar en la asignatura, en el cual el 100% de la muestra considera que el sistema LTE ayudará a reforzar los conocimientos enfocados en temas de la asignatura de SDR, en base al análisis de señales en dominio de tiempo y frecuencia. Esto se puede apreciar en el gráfico mostrado a continuación.

Figura 123

Gráfico de tabulación de pregunta 4

4. ¿Cree que el sistema LTE pueda reforzar los conocimientos enfocados en temas de la materia como el análisis de señales en dominios de tiempo y frecuencia? (0 punto)



Fuente: elaborado por el autor

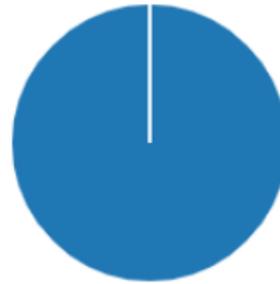
Para comprender el procesamiento de señales con el objetivo de proporcionar capacidades de análisis de espectro LTE, el 100% de los participantes, considera positivo que gracias al uso del sistema con aplicaciones o plataformas para el análisis del espectro de RF para dispositivos SDR de software libre como OpenAirInterface o GNU Radio, aportará para el entendimiento del procesamiento de señales y estudio del análisis de espectro LTE. Esto se muestra en el siguiente gráfico.

Figura 124

Gráfico de tabulación de pregunta 5

5. Mediante el uso del sistema junto con aplicaciones para el análisis del espectro de RF para dispositivos SDR de software libre, como por ejemplo, GNU Radio u OpenAirInterface, ¿cree usted que sea considerable desarrollar y comprender el procesamiento de señales con el fin de proporcionar capacidades de análisis de espectro LTE? (0 punto)

● SI 4
● NO 0



Fuente: elaborado por el autor

La siguiente pregunta, enfocada en la comprensión del sistema, de los participantes el 100% está de acuerdo y considera que el sistema será didáctico a su vez de fácil entendimiento y podrá brindar al usuario una experiencia práctica en la tecnología LTE. Dicho resultado se muestra en la siguiente gráfica.

Figura 125

Gráfico de tabulación de pregunta 6

6. ¿Considera Ud. que el sistema sea didáctico, de fácil entendimiento y brinde al usuario una comprensión práctica en la tecnología LTE? (0 punto)



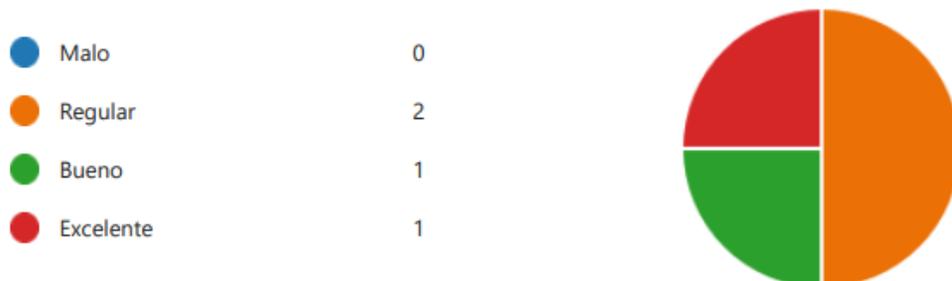
Fuente: elaborado por el autor

En cuanto al nivel de aprendizaje que proporcionaría el sistema, con el que los usuarios exploren los protocolos, así como también puedan aprender a construir una red LTE de manera independiente, el 50% establece una calificación de Regular, mientras que el 25% lo considera como Bueno y el 25% restante lo establece como Excelente. Esto se puede observar en el gráfico mostrado a continuación.

Figura 126

Gráfico de tabulación de pregunta 7

7. Cómo calificaría el nivel de aprendizaje que proporcionaría el sistema, con el que los usuarios exploren los protocolos y a su vez aprendan a construir una red LTE de manera independiente. (0 punto)



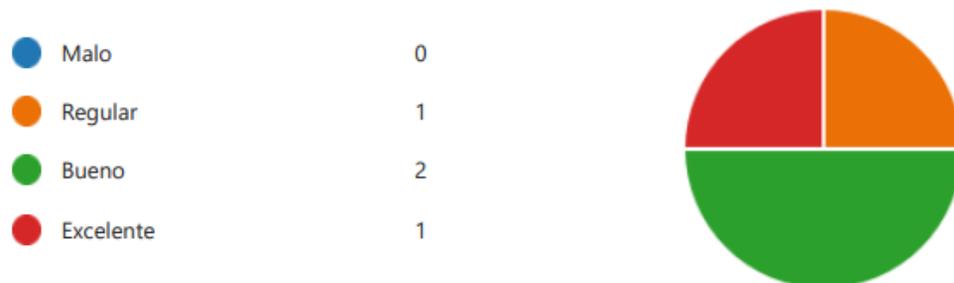
Fuente: elaborado por el autor

Continuando con el desarrollo de la encuesta, en este caso en cuanto a la calificación del nivel de aprendizaje que podría proporcionar el sistema, en que los usuarios conozcan como configurar y conectar un Equipo de Usuario a la red LTE para así obtener información sobre el procesamiento de señales en tiempo real y los protocolos LTE, el 25% establece una calificación de Regular, el 50% considera como Bueno y el 25% lo define como Excelente. Esto podemos ver en el siguiente gráfico estadístico.

Figura 127

Gráfico de tabulación de pregunta 8

8. Cómo calificaría el nivel de aprendizaje que proporcionaría el sistema, con el (0 que los usuarios aprenderán a configurar y conectar un Equipo de Usuario a la punto) red LTE para obtener información sobre el procesamiento de señales en tiempo real y los protocolos de LTE.



Fuente: elaborado por el autor

Finalmente, se preguntó que se detalle las dificultades que consideran sean fundamentales en cuanto al uso que puedan encontrarse en el sistema y señale las recomendaciones que se deberían implementar para mejorar el mismo obteniendo las siguientes respuestas. De igual forma se presenta las respuestas proporcionadas en el gráfico en una vista previa.

- Una de las dificultades que se encuentran es el utilizar equipos o software cerrado que no permita hacer análisis o modificaciones al funcionamiento de los escenarios de prácticas, para mejorar esto se pueden construir o programar módulos de hardware y software que el estudiante pueda utilizar para mejorar el aprendizaje de estas tecnologías inalámbricas, realizando pruebas considerando situaciones reales.
- Parámetros para las diferentes mediciones y configuraciones.
- Generar manuales didácticos detallados.
- No se cuenta con los suficientes kits para desarrollo de prácticas de laboratorio en el ámbito de las redes móviles. Se debería generar propuestas de prácticas donde

participen un grupo de estudiantes con la finalidad de obtener resultados y análisis sobre LTE.

Figura 128

Gráfico de respuestas de pregunta 9

9. Detalle las dificultades que usted considera sean fundamentales en cuanto al uso que puedan encontrarse en el sistema y señale de ser el caso las recomendaciones que se deberían implementar para mejorar el mismo (0 punto)

4
Respuestas

Respuestas más recientes

"No se cuenta con los suficientes kits para desarrollo de práct..."

"Generar manuales didácticos detallados."

"Parámetros para las diferentes mediciones y configuraciones."

Fuente: elaborado por el autor

7.3. Anexo 3. Fichero de configuración del eNodeB

Fichero: enb.band7.tm1.25PRB.lmssdr

```
Active_eNBs = ("eNB_LTEBox");
# Asn1_verbosity, choice in: none, info, annoying
Asn1_verbosity = "none";

eNBs =
(
{
////////// Identification parameters:
eNB_ID = 0xe00;

cell_type = "CELL_MACRO_ENB";

eNB_name = "eNB_LTEBox";

// Tracking area code, 0x0000 and 0xfffe are reserved values
tracking_area_code = "1";

mobile_country_code = "901";

mobile_network_code = "70";

////////// Physical parameters:

component_carriers = (
{
node_function = "eNodeB_3GPP";
node_timing = "synch_to_ext_device";
node_synch_ref = 0;
```

```

frame_type                = "FDD";
tdd_config                = 3;
tdd_config_s              = 0;
prefix_type               = "NORMAL";
eutra_band                = 7;
downlink_frequency        = 2680000000L;
uplink_frequency_offset   = -120000000;
Nid_cell                  = 0;
N_RB_DL                   = 25;
Nid_cell_mbsfn            = 0;
nb_antenna_ports          = 1;
nb_antennas_tx            = 1;
nb_antennas_rx            = 1;
tx_gain                   = 70;
rx_gain                   = 103;
prach_root                = 0;
prach_config_index        = 0;
prach_high_speed          = "DISABLE";
prach_zero_correlation    = 1;
prach_freq_offset         = 2;
pucch_delta_shift         = 1;
pucch_nRB_CQI             = 1;
pucch_nCS_AN              = 0;
pucch_n1_AN               = 32;
pdsch_referenceSignalPower = -48;
pdsch_p_b                 = 0;
pusch_n_SB                = 1;
pusch_enable64QAM         = "DISABLE";
pusch_hoppingMode         = "interSubFrame";
pusch_hoppingOffset       = 0;
pusch_groupHoppingEnabled = "ENABLE";

```

```

pusch_groupAssignment                = 0;
pusch_sequenceHoppingEnabled         = "DISABLE";
pusch_nDMRS1                         = 1;
phich_duration                       = "NORMAL";
phich_resource                       = "ONESIXTH";
srs_enable                           = "DISABLE";
/* srs_BandwidthConfig                =;
srs_SubframeConfig                  =;
srs_ackNackST                       =;
srs_MaxUpPts                        =;*/

pusch_p0_Nominal                     = -85;
pusch_alpha                          = "AL1";
pucch_p0_Nominal                     = -96;
msg3_delta_Preamble                  = 6;
pucch_deltaF_Format1                 = "deltaF2";
pucch_deltaF_Format1b                = "deltaF3";
pucch_deltaF_Format2                 = "deltaF0";
pucch_deltaF_Format2a                = "deltaF0";
pucch_deltaF_Format2b                = "deltaF0";

rach_numberOfRA_Preambles            = 64;
rach_preamblesGroupAConfig           = "DISABLE";
/*
rach_sizeOfRA_PreamblesGroupA        = ;
rach_messageSizeGroupA               = ;
rach_messagePowerOffsetGroupB        = ;
*/
rach_powerRampingStep                = 4;
rach_preambleInitialReceivedTargetPower = -108;
rach_preambleTransMax                 = 10;

```

```

rach_raResponseWindowSize          = 10;
rach_macContentionResolutionTimer  = 48;
rach_maxHARQ_Msg3Tx               = 4;

pcch_default_PagingCycle          = 128;
pcch_nB                           = "oneT";
bcch_modificationPeriodCoeff      = 2;
ue_TimersAndConstants_t300        = 1000;
ue_TimersAndConstants_t301        = 1000;
ue_TimersAndConstants_t310        = 1000;
ue_TimersAndConstants_t311        = 10000;
ue_TimersAndConstants_n310        = 20;
ue_TimersAndConstants_n311        = 1;

    ue_TransmissionMode            = 1;
}
);

srb1_parameters :
{
    # timer_poll_retransmit = (ms) [5, 10, 15, 20,... 250, 300, 350, ... 500]
    timer_poll_retransmit    = 80;

    # timer_reordering = (ms) [0,5, ... 100, 110, 120, ... ,200]
    timer_reordering        = 35;

    # timer_reordering = (ms) [0,5, ... 250, 300, 350, ... ,500]
    timer_status_prohibit   = 0;

    # poll_pdu = [4, 8, 16, 32 , 64, 128, 256, infinity(>10000)]
    poll_pdu                = 4;

```

```

# poll_byte = (kB) [25,50,75,100,125,250,375,500,750,1000,1250,1500,2000,3000,infinity(>10000)]
poll_byte          = 99999;

# max_retx_threshold = [1, 2, 3, 4 , 6, 8, 16, 32]
max_retx_threshold = 4;
}

# ----- SCTP definitions
SCTP :
{
# Number of streams to use in input/output
SCTP_INSTREAMS = 2;
SCTP_OUTSTREAMS = 2;
};

////////// MME parameters:
mme_ip_address  = ( { ipv4    = "127.0.0.20";
                    ipv6    = "192:168:30::17";
                    active  = "yes";
                    preference = "ipv4";
                    }
);

NETWORK_INTERFACES :
{
ENB_INTERFACE_NAME_FOR_S1_MME      = "lo";
ENB_IPV4_ADDRESS_FOR_S1_MME        = "127.0.0.10/8";

ENB_INTERFACE_NAME_FOR_S1U         = "lo";
ENB_IPV4_ADDRESS_FOR_S1U           = "127.0.0.10/8";
ENB_PORT_FOR_S1U                   = 2152; # Spec 2152
}

```

```
};

log_config :
{
    global_log_level          = "info";
    global_log_verbosity      = "medium";
    hw_log_level              = "info";
    hw_log_verbosity          = "medium";
    phy_log_level             = "info";
    phy_log_verbosity         = "medium";
    mac_log_level             = "info";
    mac_log_verbosity         = "high";
    rlc_log_level             = "info";
    rlc_log_verbosity         = "medium";
    pdcp_log_level           = "info";
    pdcp_log_verbosity        = "medium";
    rrc_log_level             = "info";
    rrc_log_verbosity         = "medium";
};
}
);
```

7.4. Anexo 4. Fichero de configuración del HSS

```
#####  
# Licensed to the OpenAirInterface (OAI) Software Alliance under one or more  
# contributor license agreements. See the NOTICE file distributed with  
# this work for additional information regarding copyright ownership.  
# The OpenAirInterface Software Alliance licenses this file to You under  
# the Apache License, Version 2.0 (the "License"); you may not use this file  
# except in compliance with the License.  
# You may obtain a copy of the License at  
#  
# http://www.apache.org/licenses/LICENSE-2.0  
#  
# Unless required by applicable law or agreed to in writing, software  
# distributed under the License is distributed on an "AS IS" BASIS,  
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
# See the License for the specific language governing permissions and  
# limitations under the License.  
#-----  
# For more information about the OpenAirInterface (OAI) Software Alliance:  
# contact@openairinterface.org  
#####  
HSS :  
{  
## MySQL mandatory options  
MYSQL_server = "127.0.0.1"; # HSS S6a bind address  
MYSQL_user = "root"; # Database server login  
MYSQL_pass = "lte"; # Database server password  
MYSQL_db = "oai_db"; # Your database name  
  
## HSS options  
OPERATOR_key = "";  
#OPERATOR_key = "1006020f0a478bf6b699f15c062e42b3"; # OP key matching your database
```

```
#OPERATOR_key = "11111111111111111111111111111111"; # OP key matching your database
```

```
RANDOM = "true"; # True random or only pseudo random (for subscriber vector  
generation)
```

```
## Freediameter options
```

```
FD_conf = "/usr/local/etc/oai/freeDiameter/hss_fd.conf";
```

```
};
```

7.5. Anexo 5. Fichero de configuración del MME

```
#####  
# Licensed to the OpenAirInterface (OAI) Software Alliance under one or more  
# contributor license agreements. See the NOTICE file distributed with  
# this work for additional information regarding copyright ownership.  
# The OpenAirInterface Software Alliance licenses this file to You under  
# the Apache License, Version 2.0 (the "License"); you may not use this file  
# except in compliance with the License.  
# You may obtain a copy of the License at  
#  
# http://www.apache.org/licenses/LICENSE-2.0  
#  
# Unless required by applicable law or agreed to in writing, software  
# distributed under the License is distributed on an "AS IS" BASIS,  
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
# See the License for the specific language governing permissions and  
# limitations under the License.  
#-----  
# For more information about the OpenAirInterface (OAI) Software Alliance:  
# contact@openairinterface.org  
#####  
  
MME :  
{  
    REALM                = "OpenAir5G.Alliance";          # YOUR REALM HERE  
    PID_DIRECTORY        = "/var/run";  
    # Define the limits of the system in terms of served eNB and served UE.  
    # When the limits will be reached, overload procedure will take place.  
    MAXENB                = 2;                            # power of 2  
    MAXUE                 = 16;                           # power of 2  
    RELATIVE_CAPACITY     = 10;
```

```

EMERGENCY_ATTACH_SUPPORTED          = "no";
UNAUTHENTICATED_IMSI_SUPPORTED      = "no";

# EPS network feature support
EPS_NETWORK_FEATURE_SUPPORT_IMS_VOICE_OVER_PS_SESSION_IN_S1 = "no"; #
DO NOT CHANGE
EPS_NETWORK_FEATURE_SUPPORT_EMERGENCY_BEARER_SERVICES_IN_S1_MODE =
"no"; # DO NOT CHANGE
EPS_NETWORK_FEATURE_SUPPORT_LOCATION_SERVICES_VIA_EPC          = "no"; # DO
NOT CHANGE
EPS_NETWORK_FEATURE_SUPPORT_EXTENDED_SERVICE_REQUEST          = "no"; # DO
NOT CHANGE

# Display statistics about whole system (expressed in seconds)
MME_STATISTIC_TIMER          = 10;

IP_CAPABILITY = "IPV4V6";          # UNUSED, TODO

INTERTASK_INTERFACE :
{
    # max queue size per task
    ITTI_QUEUE_SIZE          = 2000000;
};

S6A :
{
    S6A_CONF          = "/usr/local/etc/oai/freeDiameter/mme_fd.conf"; # YOUR MME freeDiameter
config file path
    HSS_HOSTNAME          = "hss";          # THE HSS HOSTNAME
};

# ----- SCTP definitions

```

```

SCTP :
{
    # Number of streams to use in input/output
    SCTP_INSTREAMS = 8;
    SCTP_OUTSTREAMS = 8;
};

# ----- S1AP definitions
S1AP :
{
    # outcome drop timer value (seconds)
    S1AP_OUTCOME_TIMER = 10;
};

# ----- MME served GUMMEIs
# MME code DEFAULT size = 8 bits
# MME GROUP ID size = 16 bits
GUMMEI_LIST = (
    {MCC="901" ; MNC="70"; MME_GID="4" ; MME_CODE="1"; }           # YOUR GUMMEI
CONFIG HERE
);

# ----- MME served TAIs
# TA (mcc.mnc:tracking area code) DEFAULT = 208.34:1
# max values = 999.999:65535
# maximum of 16 TAIs, comma separated
# !!! Actually use only one PLMN
TAI_LIST = (
    {MCC="901" ; MNC="70"; TAC = "1"; }                           # YOUR TAI CONFIG HERE
);

```

NAS :

{

3GPP TS 33.401 section 7.2.4.3 Procedures for NAS algorithm selection

decreasing preference goes from left to right

ORDERED_SUPPORTED_INTEGRITY_ALGORITHM_LIST = ["EIA2" , "EIA1" , "EIA0"];

ORDERED_SUPPORTED_CIPHERING_ALGORITHM_LIST = ["EEA0" , "EEA1" , "EEA2"];

EMM TIMERS

T3402 start:

At attach failure and the attempt counter is equal to 5.

At tracking area updating failure and the attempt counter is equal to 5.

T3402 stop:

ATTACH REQUEST sent, TRACKING AREA REQUEST sent.

On expiry:

Initiation of the attach procedure, if still required or TAU procedure

attached for emergency bearer services.

T3402 = 1 # in minutes (default is 12 minutes)

T3412 start:

In EMM-REGISTERED, when EMM-CONNECTED mode is left.

T3412 stop:

When entering state EMM-DEREGISTERED or when entering EMM-CONNECTED mode.

On expiry:

Initiation of the periodic TAU procedure if the UE is not attached for

emergency bearer services. Implicit detach from network if the UE is

attached for emergency bearer services.

T3412 = 54 # in minutes (default is 54 minutes, network dependent)

T3422 start: DETACH REQUEST sent

T3422 stop: DETACH ACCEPT received

ON THE 1st, 2nd, 3rd, 4th EXPIRY: Retransmission of DETACH REQUEST

T3422 = 6 # in seconds (default is 6s)

T3450 start:

ATTACH ACCEPT sent, TRACKING AREA UPDATE ACCEPT sent with GUTI, TRACKING AREA UPDATE ACCEPT sent with TMSI,

GUTI REALLOCATION COMMAND sent

T3450 stop:

ATTACH COMPLETE received, TRACKING AREA UPDATE COMPLETE received, GUTI REALLOCATION COMPLETE received

ON THE 1st, 2nd, 3rd, 4th EXPIRY: Retransmission of the same message type

T3450 = 6 # in seconds (default is 6s)

T3460 start: AUTHENTICATION REQUEST sent, SECURITY MODE COMMAND sent

T3460 stop:

AUTHENTICATION RESPONSE received, AUTHENTICATION FAILURE received,

SECURITY MODE COMPLETE received, SECURITY MODE REJECT received

ON THE 1st, 2nd, 3rd, 4th EXPIRY: Retransmission of the same message type

T3460 = 6 # in seconds (default is 6s)

T3470 start: IDENTITY REQUEST sent

T3470 stop: IDENTITY RESPONSE received

ON THE 1st, 2nd, 3rd, 4th EXPIRY: Retransmission of IDENTITY REQUEST

T3470 = 6 # in seconds (default is 6s)

ESM TIMERS

T3485 = 8 # UNUSED in seconds (default is 8s)

T3486 = 8 # UNUSED in seconds (default is 8s)

T3489 = 4 # UNUSED in seconds (default is 4s)

T3495 = 8 # UNUSED in seconds (default is 8s)

};

NETWORK_INTERFACES :

{

```
# MME binded interface for S1-C or S1-MME communication (S1AP), can be ethernet interface,
virtual ethernet interface, we don't advise wireless interfaces
```

```
MME_INTERFACE_NAME_FOR_S1_MME    = "lo";                # YOUR NETWORK
CONFIG HERE
```

```
MME_IPV4_ADDRESS_FOR_S1_MME      = "127.0.0.20/8";    # YOUR NETWORK
CONFIG HERE
```

```
# MME binded interface for S11 communication (GTPV2-C)
```

```
MME_INTERFACE_NAME_FOR_S11_MME    = "lo";                # YOUR NETWORK
CONFIG HERE
```

```
MME_IPV4_ADDRESS_FOR_S11_MME      = "127.0.0.20/8";    # YOUR NETWORK
CONFIG HERE
```

```
MME_PORT_FOR_S11_MME              = 2123;                # YOUR NETWORK CONFIG
HERE
```

```
};
```

```
LOGGING :
```

```
{
```

```
# OUTPUT choice in { "CONSOLE", "SYSLOG", `path to file`, "IPv4@:`TCP port num`" }
```

```
# `path to file` must start with '.' or '/'
```

```
# if TCP stream choice, then you can easily dump the traffic on the remote or local host: nc -l `TCP
port num` > received.txt
```

```
OUTPUT        = "CONSOLE";
```

```
#OUTPUT       = "SYSLOG";
```

```
#OUTPUT       = "/tmp/mme.log";
```

```
#OUTPUT       = "127.0.0.1:5656";
```

```
# THREAD_SAFE choice in { "yes", "no" } means use of thread safe intermediate buffer then a single
thread pick each message log one
```

```
# by one to flush it to the chosen output
```

```
THREAD_SAFE   = "yes";
```

```
# COLOR choice in { "yes", "no" } means use of ANSI styling codes or no
```

```
COLOR         = "yes";
```

```

# Log level choice in { "EMERGENCY", "ALERT", "CRITICAL", "ERROR", "WARNING",
"NOTICE", "INFO", "DEBUG", "TRACE" }

SCTP_LOG_LEVEL = "TRACE";
S11_LOG_LEVEL = "TRACE";
GTPV2C_LOG_LEVEL = "TRACE";
UDP_LOG_LEVEL = "TRACE";
S1AP_LOG_LEVEL = "TRACE";
NAS_LOG_LEVEL = "TRACE";
MME_APP_LOG_LEVEL = "TRACE";
S6A_LOG_LEVEL = "TRACE";
UTIL_LOG_LEVEL = "TRACE";
MSC_LOG_LEVEL = "ERROR";
ITTI_LOG_LEVEL = "ERROR";
MME_SCENARIO_PLAYER_LOG_LEVEL = "TRACE";

# ASN1 VERBOSITY: none, info, annoying
# for S1AP protocol
ASN1_VERBOSITY = "none";
};

TESTING :
{
# file should be copied here from source tree by following command: run_mme --install-mme-files ...
SCENARIO_FILE = "/usr/local/share/oai/test/mme/no_regression.xml";
};
};

S-GW :
{
# S-GW binded interface for S11 communication (GTPV2-C), if none selected the ITTI message interface
is used

SGW_IPV4_ADDRESS_FOR_S11 = "127.0.0.30/8"; # YOUR NETWORK
CONFIG HERE
};

```

7.6. Anexo 6. Fichero de configuración del S-PGW

```
#####  
# Licensed to the OpenAirInterface (OAI) Software Alliance under one or more  
# contributor license agreements. See the NOTICE file distributed with  
# this work for additional information regarding copyright ownership.  
# The OpenAirInterface Software Alliance licenses this file to You under  
# the Apache License, Version 2.0 (the "License"); you may not use this file  
# except in compliance with the License.  
# You may obtain a copy of the License at  
#  
# http://www.apache.org/licenses/LICENSE-2.0  
#  
# Unless required by applicable law or agreed to in writing, software  
# distributed under the License is distributed on an "AS IS" BASIS,  
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
# See the License for the specific language governing permissions and  
# limitations under the License.  
#-----  
# For more information about the OpenAirInterface (OAI) Software Alliance:  
# contact@openairinterface.org  
#####  
S-GW :  
{  
  NETWORK_INTERFACES :  
  {  
    # S-GW binded interface for S11 communication (GTPV2-C), if none selected the ITTI message  
    interface is used  
    SGW_INTERFACE_NAME_FOR_S11      = "lo";          # STRING, interface name,  
    YOUR NETWORK CONFIG HERE  
    SGW_IPV4_ADDRESS_FOR_S11        = "127.0.0.30/8"; # STRING, CIDR, YOUR  
    NETWORK CONFIG HERE
```

```
# S-GW binded interface for S1-U communication (GTPV1-U) can be ethernet interface, virtual ethernet interface, we don't advise wireless interfaces
```

```
SGW_INTERFACE_NAME_FOR_S1U_S12_S4_UP = "lo";           # STRING, interface name, YOUR NETWORK CONFIG HERE, USE "lo" if S-GW run on eNB host
```

```
SGW_IPV4_ADDRESS_FOR_S1U_S12_S4_UP   = "127.0.0.30/8"; # STRING, CIDR, YOUR NETWORK CONFIG HERE
```

```
SGW_IPV4_PORT_FOR_S1U_S12_S4_UP      = 2152;          # INTEGER, port number, PREFER NOT CHANGE UNLESS YOU KNOW WHAT YOU ARE DOING
```

```
# S-GW binded interface for S5 or S8 communication, not implemented, so leave it to none
```

```
SGW_INTERFACE_NAME_FOR_S5_S8_UP      = "none";       # STRING, interface name, DO NOT CHANGE (NOT IMPLEMENTED YET)
```

```
SGW_IPV4_ADDRESS_FOR_S5_S8_UP        = "0.0.0.0/24";   # STRING, CIDR, DO NOT CHANGE (NOT IMPLEMENTED YET)
```

```
};
```

```
INTERTASK_INTERFACE :
```

```
{
```

```
# max queue size per task
```

```
ITTI_QUEUE_SIZE      = 2000000;          # INTEGER
```

```
};
```

```
LOGGING :
```

```
{
```

```
# OUTPUT choice in { "CONSOLE", "SYSLOG", `path to file`, "IPv4@:`TCP port num`" }
```

```
# `path to file` must start with '.' or '/'
```

```
# if TCP stream choice, then you can easily dump the traffic on the remote or local host: nc -l `TCP port num` > received.txt
```

```
OUTPUT      = "CONSOLE";                 # see 3 lines above
```

```
#OUTPUT     = "SYSLOG";                 # see 4 lines above
```

```
#OUTPUT     = "/tmp/spgw.log";         # see 5 lines above
```

```
#OUTPUT     = "127.0.0.1:5656";       # see 6 lines above
```

```
# THREAD_SAFE choice in { "yes", "no" } means use of thread safe intermediate buffer then a single thread pick each message log one
```

```

# by one to flush it to the chosen output
THREAD_SAFE    = "no";

# COLOR choice in { "yes", "no" } means use of ANSI styling codes or no
COLOR          = "yes";

# Log level choice in { "EMERGENCY", "ALERT", "CRITICAL", "ERROR", "WARNING",
"NOTICE", "INFO", "DEBUG", "TRACE" }
UDP_LOG_LEVEL  = "TRACE";
GTPV1U_LOG_LEVEL = "TRACE";
GTPV2C_LOG_LEVEL = "TRACE";
SPGW_APP_LOG_LEVEL = "TRACE";
S11_LOG_LEVEL  = "TRACE";
};
};

P-GW =
{
  NETWORK_INTERFACES :
  {
    # P-GW binded interface for S5 or S8 communication, not implemented, so leave it to none
    PGW_INTERFACE_NAME_FOR_S5_S8    = "none";           # STRING, interface name,
DO NOT CHANGE (NOT IMPLEMENTED YET)

    # P-GW binded interface for SGI (egress/ingress internet traffic)
    PGW_INTERFACE_NAME_FOR_SGI      = "eno1";           # STRING, YOUR
NETWORK CONFIG HERE

    PGW_MASQUERADE_SGI              = "yes";           # STRING, {"yes", "no"}. YOUR
NETWORK CONFIG HERE, will do NAT for you if you put "yes".

    UE_TCP_MSS_CLAMPING             = "no";           # STRING, {"yes", "no"}.
};
};

```

```

# Pool of UE assigned IP addresses
# Do not make IP pools overlap
# first IPv4 address X.Y.Z.1 is reserved for GTP network device on SPGW
# Normally no more than 16 pools allowed, but since recent GTP kernel module use, only one pool
allowed (TODO).
IP_ADDRESS_POOL :
{
    IPV4_LIST = (
        "172.16.0.0/12"                # STRING, CIDR, YOUR NETWORK CONFIG
HERE.
    );
};

# DNS address communicated to UEs
DEFAULT_DNS_IPV4_ADDRESS = "8.8.8.8";                # YOUR NETWORK CONFIG
HERE
DEFAULT_DNS_SEC_IPV4_ADDRESS = "8.8.4.4";            # YOUR NETWORK
CONFIG HERE

# Non standard feature, normally should be set to "no", but you may need to set to yes for UE that do not
explicitly request a PDN address through NAS signalling
FORCE_PUSH_PROTOCOL_CONFIGURATION_OPTIONS = "no";    # STRING,
{"yes", "no"}.
UE_MTU = 1500 # INTEGER
};

```