

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-27

Implementación de una Estación Celular BTS con *hardware* SDR basado en Sistema Operativo Linux que permita la convergencia con una Planta Telefónica Panasonic TDE-100

Juan Felipe Castaño Ocampo

Andres Felipe Ramírez Zapata

Ingeniería en Telecomunicaciones

Asesor. Germán David Góez Sánchez

Ingeniero de Telecomunicaciones

Maestría en automatización y Control Industrial

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

INSTITUTO TECNOLÓGICO METROPOLITANO

MEDELLÍN, 2018

RESUMEN

El diseño e implementación de laboratorios de comunicaciones móviles con fines educativos basados en estaciones bases celulares o BTs, son complejos de implementar debido a su alto costo, equipos con tecnología cerrada y dedicada a operadores exclusivos. En este sentido, el desarrollo de sistemas de comunicaciones móviles de bajo costo y tecnología abierta son una buena opción para implementar laboratorios con fines académicos.

En este proyecto se propuso la implementación de un prototipo de estación celular basado en el proyecto OpenBTS, el cual es un proyecto (open source software) con parámetros programables, usando el software GNU-Radio, integrado a una planta telefónica. Esto permite emular una red celular integrada a la telefonía pública conmutada. Finalmente, se entrega una guía básica de instalación y configuración de la estación base propuesta, con el fin de ser usada en las asignaturas relacionadas con las comunicaciones móviles.

Palabras clave: Linux, Celular, software, TDE-100.

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

ACRÓNIMOS

FSF (Free Software Foundation)

SDR (Software-Defined Radio, Radio Definido por Software).

OpenBTS (Open Base Transceiver Station)

VoIP (protocolo de voz sobre IP)

USRP (Universal Software Radio Peripheral, El radio transceptor)

BTS (Base Tranceiver Station)

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

TABLA DE CONTENIDO

RESUMEN	2
ACRÓNIMOS	3
TABLA DE CONTENIDO	4
LISTA DE FIGURAS	6
1. INTRODUCCIÓN	7
1.1. PLANTEAMIENTO DEL PROBLEMA.....	8
1.2. OBJETIVOS.....	9
1.2.1. General.....	9
1.2.2. Específicos.....	9
2. MARCO TEÓRICO	10
2.1. Sistema operativo Linux, software SDR y URSP	10
2.1.1. Sistema operativo Linux.....	10
2.2. GNU Radio.....	10
2.2.1. OpenBTS.....	11
2.2.2. Planta telefónica TDE 100	11
2.2.3. USRP.....	11
3. METODOLOGÍA.....	12
3.1. Objetivo 1: Identificación de los diferentes avances científicos y tendencias tecnológicas acerca de las nuevas tecnologías de radio comunicación celular	12
3.2. Objetivo 2: Integración del software GNU-Radio, OpenBTS y Astersik con la Planta telefónica TDE-100.....	12
3.3. Objetivo 3: Implementación un caso de estudio con el fin de realizar pruebas de funcionamiento del prototipo implementado, además, documentar	13
4. RESULTADOS Y DISCUSIÓN.....	14
4.1. Objetivo 1: Identificación de los diferentes avances científicos y tendencias tecnológicas sobre tecnologías de radio comunicación celular.....	14
4.2. Objetivo 2: Integración del software GNU-Radio, OpenBTS y Astersik con la Planta telefónica TDE-100.....	14
4.2.1. Implementar, OpenBTS	14

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

4.2.2.	Instalación OpenBTS	17
4.2.3.	Comandos GSM	18
4.2.4.	Acceso de Usuarios	19
4.2.5.	Llamadas y mensaje de prueba	20
4.2.6.	Configuración y Asignación de IP fija en Servidor Ubuntu	20
4.3.	Guía para Instalación de Planta BTS	28
4.3.1.	Configuración de la troncal SIP en la planta Panasonic KX TDE 100:	32
4.3.2.	Configurar salida de llamadas a planta telefónica Asterisk.....	35
5.	CONCLUSIONES, RECOMENDACIONES Y TRABAJO FUTURO	39
6.1.	Conclusiones	39
6.2.	Recomendaciones	40
6.3.	Trabajo Futuro	40
	REFERENCIAS	41

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

LISTA DE FIGURAS

Figura 2. D-Link DES-3010F.....	29
Figura 3. NI USRP-2900	29
Figura 4. Planta Panasonic KX TDE 100	30
Figura 5. PBX Unified Maintenance Console	30
Figura 6. Panasonic Pbx Unified Maintenance Console	31
Figura 7. Conectar(C)	31
Figura 8. Descripción Y Valor	32
Figura 9. Clave De Activación	33
Figura 10. IPCMPR Ranura Virtual 8.....	33
Figura 11. Pestaña “Principal”	34
Figura 12. Pestaña “Cuenta”	34
Figura 13. Pestaña “Registrar”	34
Figura 14. Clic en INS.....	35
Figura 15. Columna “Marcación Inicial”	36
Figura 16. Pestaña “Cod. Acceso Otra PBX”	36
Figura 17. Pestaña “Prioridad 1 - 2”	37
Figura 18. “Número de grupo de L.N”	37
Figura 19. Pestaña “DIL”	38

	<p style="text-align: center;">INFORME FINAL DE TRABAJO DE GRADO</p>	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

1. INTRODUCCIÓN

El hombre ha sabido aprovechar las ventajas de las ondas electromagnéticas, desde las primeras investigaciones y descubrimientos en esta área se han podido desarrollar sistemas de comunicación que hace algunos siglos atrás no se creía posible, con el nacimiento de los dispositivos de hardware como circuitos de radio y antenas la humanidad ha podido dar un gran salto en la ciencia, el hombre se ha podido comunicar por medios inalámbricos alrededor de todo el mundo y casi en cualquier lugar, a medida que la tecnología avanza y surgen nuevos medios y dispositivos de comunicación como la fibra óptica, la comunicación por radio sigue siendo vital en nuestro día a día, ya será para el entretenimiento como las emisoras de música como también en la seguridad con las radio patrullas de la policía, bomberos o ambulancias, en el sistema de transporte aéreo en los aviones y en la navegación marítima, redes de telefonía celular entre muchas más, las ondas electromagnéticas y su uso con sistemas de radio siguen vigentes en nuestros días pero a medida de que se desarrolla la ciencia nacen nuevos sistemas, protocolos y herramientas que permiten optimizar esta tecnología y adaptarla al rápido avance de nuestra era.

En un principio, todos los sistemas de radio eran completamente análogos, hoy en día se ha logrado fusionar con sistema digitales y software que permiten programar las funciones de estos radios de forma rápida ya sea con un código de programación o con un entorno grafico ya compilado, una de las principales desventajas de los sistemas de radio actuales es que para llevar a cabo investigaciones.

Para la realización de este proyecto se usó un sistema de radio definida por software conocida por sus siglas en ingles SDR (Software Defined Radio) ya que es un sistema de radio comunicación que permite implementar por medio de software muchos de los componentes generalmente implementados en hardware como lo son mezcladores, filtros, moduladores, demoduladores, entre otros, permitiendo así desde un computador y el respectivo software poder programar y controlar los parámetros requeridos para controlar

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

las funciones de la radio comunicación; con lo cual estos sistemas se adaptan a radios físicos o de hardware con su respectiva antena, brindando una completa arquitectura de comunicaciones más rápida de implementar y con un nivel de flexibilidad mayor facilitando la integración con redes e interfaces en protocolos diferentes.

Para llevar a cabo la implementación del sistema se utilizó con un radio transceptor que es un dispositivo de hardware con tecnología USRP (Universal Software Radio Peripheral) que permiten una sinergia entre hardware y software haciendo más fácil la implementación de un sistema de radio.

1.1. PLANTEAMIENTO DEL PROBLEMA

El diseño e implementación de laboratorios de comunicaciones móviles con fines educativos basados en estaciones bases celulares o BTS, son complejos de implementar debido a su alto costo, equipos con tecnología cerrada y dedicada a operadores exclusivos. En este sentido, el desarrollo de sistemas de comunicaciones móviles de bajo costo y tecnología abierta son una buena opción para implementar laboratorios con fines académicos.

En este trabajo, se propone implementar un prototipo de una central celular BTS programable por medio de software que permite integrar una pieza de software SDR con un dispositivo de hardware USRP con su respectiva antena para poder aprovechar los recursos físicos y lógicos de un computador y las diferentes herramientas con las que cuenta el software y así poder emular sistemas de comunicaciones móviles integrados a la telefonía pública conmutada, que puedan ser usados por la comunidad académica para realizar trabajos relacionados con las comunicaciones móviles en un entorno real reduciendo el tiempo de desarrollo, costo en equipos.

	<p style="text-align: center;">INFORME FINAL DE TRABAJO DE GRADO</p>	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

1.2. OBJETIVOS

1.2.1. General

Diseñar, implementar y documentar un prototipo de una estación celular BTS con parámetros programables y ajustables, usando el software GNU-Radio, integrada a una planta telefónica TDE 100, con el fin de emular una red celular integrada a la telefonía pública conmutada.

1.2.2. Específicos

- Identificar los diferentes avances científicos y tendencias tecnológicas acerca de las nuevas tecnologías de radio comunicación celular, analizar la influencia de la radio definida por software en la integración y desarrollo de las comunicaciones móviles.
- Integrar el software GNU-Radio, OpenBTS y Astersik con la Planta telefónica TDE-100.
- Implementar el caso de estudio con el fin de realizar pruebas de funcionamiento del prototipo; además, documentar y construir una guía de uso para ser entregada a la comunicada académica.

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

2. MARCO TEÓRICO

2.1. Sistema operativo Linux, software SDR y URSP

2.1.1. Sistema operativo Linux

Linux es un sistema operativo libre basado en el desarrollo del sistema operativo GNU diseñado por la FSF (Free Software Foundation) con un núcleo (kernel) adaptativo. En la actualidad existen muchas versiones de Linux, llamadas distribuciones que son desarrolladas por empresas, instituciones o fundaciones diferentes, cada una de ella cuenta con un kernel Unix, pero con sus propias variaciones en el código fuente del sistema operativo, cada entidad se encarga de desarrollar y mantener actualizadas las versiones de sus distribuciones, entre las distribuciones más comunes se encuentran Ubuntu, Debian, Open Suse, Arch Linux, Fedora, Kali Linux, en este proyecto se escoge Ubuntu por la abundancia de sus librerías (Valeiro, 2008).

Las distribuciones de Linux están basadas en la licencia GPL (Licencia Publica General de GNU) y otras licencias libres lo que permite descargar, copiar, utilizar, modificar y re-distribuir; los sistemas operativos Linux son gratuitos y se pueden usar libremente para usos comerciales, las entidades se financian a través de donaciones o cobrando solamente por el soporte técnico además de que cuentan con una gran comunidad a nivel mundial lo que hace de Linux una herramienta poderosa, gratuita y efectiva para el desarrollo de cualquier proyecto basado en TIC's (Guided Tutorial GNU Radio in Python, 2017).

2.2. GNU Radio

GNU Radio es un software de código abierto distribuido bajo la licencia GPLv3 (General Public License Version 3), que permite la instalación de un conjunto de paquetes y librerías para crear y desarrollar sistemas de radio definido por software SDR (Software-Defined Radio, Radio Definido por Software). El software cuenta además con soporte disponible a través de foros de internet, el software permite crear muchas aplicaciones que se pueden realizar de forma simplificada ya que cuenta con un framework con muchos bloques que incluyen filtros, modulares, demoduladores y otros elementos de manipulación de señales que generalmente tienen los equipos de radio físicos o hardware permitiendo la simple implementación de un procesamiento digital de señales por medio de grafos (GNU-Radio Official, 2012).

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

2.2.1. OpenBTS

OpenBTS (Open Base TransceiverStation) es una pieza de software para sistemas operativos de kernel (nucelo) Unix que usa un sistema de radio para generar una interfaz y punto de acceso GSMa partir del mismo software que permite operar con cualquier teléfono celular GSM estándar, esta herramienta es la primera implementación de software libre del protocolo de stock del estándar industrial GSM. OpenBTS permite que los celulares estándar se conecten a una red sin necesidad de interactuar con las redes de telecomunicaciones comerciales establecidas, permite que los celulares vean una completa red GSM a través de su interfaz permitiendo recibir y realidad llamadas sin usar el operador comercial, El proyecto OpenBTS forma la base de un nuevo tipo de red celular que puede ser desarrollada y operada a un costo más bajo que las tecnologías existentes incluyendo zonas rurales y redes privadas de celular,el proyecto OpenBTS fue desarrollado por la entidad Range Networks (Ni, Collings, Lipman, & Wang, 2015).

2.2.2. Planta telefónica TDE 100

La planta telefónica Panasonic TDE 100 es una plataforma de comunicación avanzada que contribuye a aprovechar al máximo las redes IP de las empresas. Esta soporta las ultimas redes análogas, digitales y protocolo de voz sobre IP (VoIP), tecnologías de protocolo SIP para manejar de manera efectiva y rentable las comunicaciones dentro de las empresas, entre sucursales y larga distancia, montado sobre redes comunicadas; tanto por voz, como por sistema de datos. Se pueden conectar en red y operar de manera centralizada hasta ocho plantas del tipo KX-TDE, sin necesidad de un servidor.

2.2.3. USRP

El radio transeptor USRP (Universal Software Radio Peripheral) es un dispositivo de hardware desarrollado por la compañía EttusReserach que se puede conectar a un computador estándar y ser configurado como un sistema de generación de prototipos inalámbricos lo que comprende una plataforma flexible para SDR. En su núcleo implementa las secciones de procesamiento digital en banda base y frecuencia intermedia. Su diseño básico consiste en la transformación de la forma de onda, tal como la modulación y demodulación (Ettus Research, 2017).

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

3. METODOLOGÍA

Las herramientas empleadas en este proyecto fueron: el sistema operativo Linux, el radio USRP con sus respectivas antenas, el software OpenBTS, GNU-Radio Y Asterisk, así como la Planta telefónica TDE-100 para diseñar e implementar un prototipo que permitieron emular el funcionamiento de una central celular BTS, en un ambiente controlado, realizando las respectivas pruebas de funcionamiento y efectividad, documentando los resultados para que sirvan como referencia en futuros estudios y realización de pruebas en centrales celulares para comprobación de los cálculos matemáticos.

3.1. Objetivo 1: Identificación de los diferentes avances científicos y tendencias tecnológicas acerca de las nuevas tecnologías de radio comunicación celular

Para el desarrollo de este objetivo se realizó una búsqueda en revistas científicas relacionadas con las comunicaciones móviles, radio definida por software USRP e integración de estos sistemas a redes de comunicación comerciales. También se investigó las ventajas del software, así como su efectividad y de los últimos avances tecnológicos en la integración de hardware y software para las comunicaciones inalámbricas.

Se consultaron bases de datos científicas y los manuales de fabricantes y desarrolladores: Ubuntu, GNU-Radio, Ettus Research®, Panasonic®.

3.2. Objetivo 2: Integración del software GNU-Radio, OpenBTS y Astersik con la Planta telefónica TDE-100.

Durante esta fase del proyecto se realizaron actividades técnicas para el diseño e implementación del sistema de comunicación por medio de software y hardware. Primero se instaló la distribución de Linux aprovechando sus librerías, una vez instalado se procedió a su respectiva configuración y actualización de paquetes y *drivers*. Ya preparado el sistema,

	<p style="text-align: center;">INFORME FINAL DE TRABAJO DE GRADO</p>	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

se instaló los aplicativos GNU-Radio, Asterisk, OpenBTS, con los paquetes necesarios para el uso de las antenas y el radio transceptor USRP (Hamza F. A., 2008). Concretamente:

Linus Ubuntu. Para este proyecto se usó la distribución Ubuntu debido a que es una de las distribuciones más completas y estables, para realizar la instalación primero se descargó la Versión 16.04.3 LTS, el sistema operativo se descargará como imagen de disco ISO, se configuró de forma *bootable* por medio de memoria USB (Montero-Hidalgo J. P., 2014).

GNU Radio. Para realizar la instalación del aplicativo GNU-Radio se usaron los repositorios con los que cuenta el sistema operativo.

Asterisk. La configuración del aplicativo Asterisk fue desde página web oficial <http://www.asterisk.org/downloads>, en la página se realiza la descarga del código fuente en un archivo comprimido de extensión *tar.gz*. Una vez descargado el archivo se ingresa la terminal de Linux y se compila.

OpenBTS. Esta instalación fue análoga a la de Asterisk.

Una vez instalado los aplicativos se procede con la configuración de cada uno de ellos para sincronizarlos con el radio USRP. Y acto seguido, se integró con la planta TDE 100 para su buen funcionamiento y estabilidad en la comunicación.

3.3. Objetivo 3: Implementación un caso de estudio con el fin de realizar pruebas de funcionamiento del prototipo implementado, además, documentar

En esta fase se realizó un diagrama de red y sus respectivos protocolos. Luego, se realizarán todas las conexiones físicas e inalámbricas para comprobar el funcionamiento de la red y se llevarán a cabo las pruebas de funcionalidad y estabilidad de la comunicación a través del sistema.

Finalmente, todas la información adquirida y producida durante las anteriores actividades se recopiló y documentó.

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

4. RESULTADOS Y DISCUSIÓN

4.1. Objetivo 1: Identificación de los diferentes avances científicos y tendencias tecnológicas sobre tecnologías de radio comunicación celular

Las comunicaciones móviles se han desarrollado de forma acelerada en las últimas tres décadas, su desarrollo está ligado directamente a la demanda y la necesidad de proveer nuevas tecnologías que soporten a más usuarios con un mejor ancho de banda. En este orden de ideas, las nuevas propuestas en relación a las comunicaciones móviles se basan en la conectividad tanto de personas como de objetos que de algún modo necesitan estar en red. Un ejemplo de esto se da en el nuevo concepto de IoT o internet de las cosas (Qhgg, Lqyhvwphqw, Wkh, & Lqiudvwuxfwxuh, 2017).

Pero si la necesidad es de comunicación entre personas, la implementación de tecnologías 4G y en las que están en desarrollo como lo es la 5G, permiten observar un futuro prometedor en términos de conectividad y ancho de banda. Otras propuestas se basan en mejorar los sistemas electrónicos que intervienen en los sistemas de comunicaciones, los autores de Makino, Mizutani, Matsumura, & Harada, (2018) proponen la evaluación del rendimiento de del VER en VHF para sistemas de banda ancha basados en el estándar IEEE 802.16. Los autores usan como base la modulación OFDM y evalúan el rendimiento de BER en sistemas con multi trayecto.

4.2. Objetivo 2: Integración del software GNU-Radio, OpenBTS y Astersik con la Planta telefónica TDE-100.

4.2.1. Implementar, OpenBTS

Instalar Ubuntu 16.04.3 LTS Desktop

Puede descargarse de la siguiente dirección: <http://releases.ubuntu.com/16.04/>

Actualizar repositorios y dependencias

```
sudo apt-get update
```

Instalar GNU-Radio

```
apt-get install gnuradio
```

Instalación de dependencias Python y SQL

En caso de que alguna falle es porque está actualizada, en tal caso simplemente no se digita:

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

```

sudo apt-get -y install git-core autoconf automake libtool g++
python-dev swig pkg-con_g lib_tw3-dev libcppunit-dev libgsl0-dev
vim ssh libusb-dev sdcc libstd1.2-dev python-wxgtk2.8 python-numpy
python-cheetah python-lxml doxygen python-qt4 python-qwt5-qt4
libxi-dev libqt4-opengl-dev libqwt5-qt4-dev libfontcon_g1-dev
libxrender-dev libusb-dev fort77 qt4-dev-tools libqwtplot3d-qt4-dev
pyqt4-dev-tools libortp-dev libxml2-dev gawk libdbd-mysql

```

Instalación de dependencias adicionales

```

sudo apt-get install libdbi-dev
sudo apt-get install libdbd-pgsql
sudo apt-get install bind9
sudo apt-get install ntp

```

Ahora, hay que descargar, descomprimir e instalar el paquete libosip.

```

wget http://ftp.gnu.org/gnu/osip/libosip2-3.5.0.tar.gz
gunzip libosip2-3.5.0.tar.gz
tar -xvf libosip2-3.5.0.tar
cd libosip2-3.5.0
sudo ./configure
sudo make
sudo make install
sudo apt-get install cmake
sudo apt-get install libusb-1.0-0-dev
sudo apt-get install sdcc
sudo apt-get install libaudio-dev
sudo apt-get install libuhd-dev libuhd003 uhd-host
sudo apt-get install libboost-all-dev python-mako doxygen
sudo apt-get install python-docutils build-essential

```

Descarga de drivers UHD

Estos drivers sirven para acceder al radio por cable USB, se descargan desde:

https://files.ettus.com/binaries/uhd/uhd_003.009.003-release/uhd-3.9.3.tar.gz

Una vez descargados los drivers, se accede al directorio **Descargas** para luego instalarlos:

```

cd /home/Descargas      (ruta similar)
tar -xvf uhd-3.9.3.tar.gz

```

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

```
cd uhd-3.9.3
sudo mkdir build
cd build
sudo cmake ../
sudo make
sudo make install
sudo /usr/local/lib/uhd/utils/uhd_images_downloader.py
```

Verificación de carpeta Images

Se debe revisar que la carpeta **Images** se encuentre en las siguientes dos rutas, en caso de que falte en alguna ruta copiarla usando el comando **nautilus**.

```
/usr/local/share/uhd
/usr/local/lib/uhd
```

Copia de carpeta uhd

Para finalizar la instalación de los drivers, se debe copiar la carpeta **uhd** desde alguna de las rutas anteriores a la ruta:

```
/usr/share
```

Librerías de servicio de mensajería SMQueue para SMS

```
sudo apt-get install g++ erlang libreadline6-dev
sudo apt-get install bind9 ntp autoconf libboost-all-dev ntpdate
sudo apt-get install software-properties-common python-software-properties
sudo add-apt-repository ppa:chris-lea/zeromq
sudo apt-get update
```

Obtención de código fuente de OpenBTS

Primero, crear una carpeta para el servicio OpenBTS:

```
cd /home/ubuntu/Documents (ruta similar)
mkdir OpenBTS
cd OpenBTS
```

Estando dentro de esta carpeta:

```
git clone https://github.com/RangeNetworks/dev.git
```

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

```
cd dev
./clone.sh
./switchto.sh master
cd liba53
sudo make install
sudo ldconfig
cd ..
```

El siguiente comando es crítico, pues especifica la referencia del radio USRP a utilizar (B200):

```
./build.sh B200
```

4.2.2. Instalación OpenBTS

Acceder a la carpeta **builds** dentro de la carpeta OpenBTS (creada en el punto 11).

```
cd /home/gnuradio/Documentos/OpenBTS/dev/BUILDS
ls
```

Debe haber una carpeta cuyo nombre es la fecha de compilación, acceder a ella.

```
cd "nombre de carpeta"
sudo dpkg -i *.deb
```

Inicialización de Servicios SYSTEMD

En primer lugar, clonamos el repositorio para el comando **systemd** dentro de la carpeta creada de OpenBTS (creada en el punto 11):

```
git clone https://github.com/nadiia-kotelnikova/openbts_systemd_scripts.git
```

Entrar a la carpeta recién descargada y acceder a la carpeta **systemd**:

```
cd openbts_systemd_scripts/systemd
```

Copiar todos los archivos

```
sudo cp * /etc/systemd/system/
cd ..
```

	<p style="text-align: center;">INFORME FINAL DE TRABAJO DE GRADO</p>	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

Recordar reiniciar el sistema en este punto para que todas las configuraciones queden correctamente agregadas. A continuación, ingresar nuevamente a la carpeta **OpenBTS**, entrar a la carpeta **openbts_systemd_scripts** ejecutar el script:

```
cd /home/gnuradio/Documentos/OpenBTS/openbts_systemd_scripts/  
./openbts-start.sh
```

Ejecutar OpenBTS

Ubicarse en la ruta

```
cd /home/gnuradio/Documentos/OpenBTS/openbts_systemd_scripts/
```

Y ejecutar para abrir la CLI:

```
./openbts-cli.sh
```

4.2.3. Comandos GSM

Ya estando en línea de comando OpenBTS,

```
OpenBTS> help          (para familiarizarse con los comandos)}
```

Se procede a configurar la red GSM, leer detenidamente cada uno de los parámetros que mostró el comando.

```
OpenBTS> config GSM.Radio
```

Queremos que la frecuencia 1800 MHz y el CO 650:

```
OpenBTS> config GSM.Radio.Band 850
```

```
OpenBTS> config GSM.Radio.CO 166
```

Se recomienda una ganancia de 10 dB o más baja

```
OpenBTS> devconfig GSM.Radio.RxGain 10
```

Ejecutar los siguientes comandos para reducción de ruido

```
OpenBTS> power 20
```

```
OpenBTS> noise
```

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

Finalmente, en un celular Android acceder a “*Redes móviles*> *Redes disponibles* > *Buscar redes*” y conectarse a la red USRP, que deberá llamarse **00101** o **RANGE**.

4.2.4. Acceso de Usuarios

Ubicarse en la ruta

```
cd /home/gnuradio/Documentos/OpenBTS/dev/openbts/NodeManager
print "command examples:"
```

```
print ".nmcli.py target version (query the component version)"
print ".nmcli.py target config read (show all configuration values)"
print ".nmcli.py target config read Log.Level (show a specific configuration value)"
print ".nmcli.py target config update Log.Level INFO (update a specific configuration value)"
print ".nmcli.py openbts monitor (query some live radio data)"
print ".nmcli.py sipauthserve subscribers read (show all subscribers)"
print ".nmcli.py sipauthserve subscribers create name imsi msisdn (create a subscriber which uses cache auth)"
print ".nmcli.py sipauthserve subscribers create name imsi msisdn ki (create a subscriber which uses full auth)"
print ".nmcli.py sipauthserve subscribers delete imsi the-imsi (delete a subscriber by imsi)"
print ".nmcli.py sipauthserve subscribers delete msisdn the-msisdn (delete a subscriber by imsi)"
print "response codes:"
```

```
print " 200 : action ok with response data"
print " 204 : action ok with no response data"
print " 204 : action ok with no change"
print " 404 : unknown key or action"
print " 406 : request is invalid"
print " 409 : conflicting value"
print " 500 : storing new value failed"
print "usage:"
print ".nmcgi.py target command (action (key (value)))"
```

	<p style="text-align: center;">INFORME FINAL DE TRABAJO DE GRADO</p>	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

Para visualizar el nombre de la red

```
config GSM.Identity.ShortName
```

Cambiar nombre de la red

```
config GSM.Identity.ShortName "nombre"
```

Nota: tener en cuenta que el nombre debe estar entre comillas. Ejecutar el siguiente comando para cada usuario que se quiera registrar:

```
./nmcli.py sipauthserve subscribers create "Rene" IMSI732111140142310 \3324006
```

Leer usuarios inscritos

```
./nmcli.py sipauthserve subscribers read
```

Borrar usuarios inscritos

```
./nmcli.py sipauthserve subscribers delete imsi IMSI732101511208288
```

4.2.5. Llamadas y mensaje de prueba

Desde un celular conectado enviar mensaje a número **411 y al 101, este último con el número telefónico del dispositivo**

Llamar a **2600 y 2602**

Mensaje de texto SMS

Ingresar a la CLI de OpenBTS y usar el comando **sendsms**

```
OpenBTS> sendsms IMSI Numero-Emisor Mensaje
message submitted for delivery
```

4.2.6. Configuración y Asignación de IP fija en Servidor Ubuntu

```
Modificar el archivo "/etc/network/interfaces"
```

Ruta

	<p style="text-align: center;">INFORME FINAL DE TRABAJO DE GRADO</p>	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

```
cd /etc/network/
sudo nano /etc/network/interfaces
```

Asignar los datos de la red

En nuestro caso el adaptador de red se llama **“enp4s8”**

```
auto eth0 (enp4s8)
iface eth0 (enp4s8) inet static
address 10.2.10.200
netmask 255.255.254.0
gateway 10.2.10.1
dns-nameservers 8.8.8.8 8.8.4.4
```

Reiniciar el servicio networking

```
sudo ifdown eth0 (enp4s8)
sudo ifup eth0 (enp4s8)
sudo service networking restart
```

Tener en cuenta que las direcciones IP de los computadores, Planta Panasonic TDE-100 y Planta Asterisk deben estar en el mismo segmento de red.

Configurar Asterisk para llamadas

En primer lugar, acceder a la ruta:

```
cd /etc/asterisk
```

Abrir el archivo sip.conf

```
openvt gedit sip.conf
```

En el archivo **sip.conf** debe ir lo siguiente:

```
[general]
;; Cadena de registro SIP
;; register => user[:secret[:authuser]]@host[:port][/extension]
register=>501:passwd@10.2.10.202:5060/501
```

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

```
[501]
type=friend
secret=passwd
qualify=yes
trunk=yes
context=saliente
host=dynamic
disallow=all
allow=ulaw
allow=alaw
allow=gsm
```

```
bindport=5060 ; asterisk 1.6
; UDP Port to bind to (SIP standard port for unencrypted UDP
; and TCP sessions is 5060)
; bindport is the local UDP port that Asterisk will listen on
bindaddr=0.0.0.0 ; asterisk 1.6
srvlookup=no
```

```
[phones](!)
qualify=yes ; IP address to bind UDP listen socket to (0.0.0.0 binds to all)
; You can specify port here too, like 123.123.123.123:5080
udpbindaddr=0.0.0.0 ; asterisk 1.8
; IP address to bind UDP listen socket to (0.0.0.0 binds to all)
; Optionally add a port number, 192.168.1.1:5062 (default is port
5060)
```

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

```

tos_sip=cs3           ; Sets TOS for SIP packets.
tos_audio=ef         ; Sets TOS for RTP audio packets.
tos_video=af41       ; Sets TOS for RTP video packets.
tos_text=af41        ; Sets TOS for RTP text packets.

```

```

cos_sip=3            ; Sets 802.1p priority for SIP packets.
cos_audio=5          ; Sets 802.1p priority for RTP audio packets.
cos_video=4          ; Sets 802.1p priority for RTP video packets.
cos_text=3           ; Sets 802.1p priority for RTP text packets.

```

```

maxexpiry=3600      ; Maximum allowed time of incoming registrations
                    ; and subscriptions (seconds)
minexpiry=60         ; Minimum length of registrations/subscriptions (default 60)
defaultexpiry=3600  ; Default length of incoming/outgoing registration
dynamic_exclude_static=yes ; Disallow all dynamic hosts from registering
                    ; as any IP address used for statically defined
                    ; hosts. This helps avoid the configuration
                    ; error of allowing your users to register at
                    ; the same address as a SIP provider.
use_q850_reason=yes ; Set to yes add Reason header and use Reason header if it is
                    ; available.

```

```

t1min=100           ; Minimum roundtrip time for messages to monitored hosts
                    ; Defaults to 100 ms
timert1=500         ; Default T1 timer
                    ; Defaults to 500 ms or the measured round-trip
                    ; time to a peer (qualify=yes).
timerb=32000        ; Call setup timer. If a provisional response is not received

```

```

; in this amount of time, the call will autocongest
; Defaults to 64*timert1
rtptimeout=60 ; Terminate call if 60 seconds of no RTP or RTCP activity
; on the audio channel
; when we're not on hold. This is to be able to hangup
; a call in the case of a phone disappearing from the net,
; like a powerloss or grandma tripping over a cable.
rtpholdtimeout=300 ; Terminate call if 300 seconds of no RTP or RTCP activity
; on the audio channel
; when we're on hold (must be > rtptimeout)

```

```

allowguest=no ; Allow or reject guest calls (default is yes)
autocreatepeer=no ; The Autocreatepeer option allows,
; if set to Yes, any SIP ua to register with your Asterisk PBX as a peer.
; This peer's settings will be based on global options.
; The peer's name will be based on the user part of the Contact:
header field's URL.

```

```

allowoverlap=yes ; Disable overlap dialing support. (Default is yes)

```

```

canreinvite=no ; no reinvites from Asterisk

```

```

directmedia=no ; Asterisk by default tries to redirect the
; RTP media stream to go directly from
; the caller to the callee. Some devices do not
; support this (especially if one of them is behind a NAT).
; The default setting is YES. If you have all clients
; behind a NAT, or for some other reason want Asterisk to

```

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

; stay in the audio path, you may want to turn this off.

; This setting also affect direct RTP
; at call setup (a new feature in 1.4 - setting up the
; call directly between the endpoints instead of sending
; a re-INVITE).

#include sip-custom-register.conf

```
[CodecBTS](!)
;disallow=all      ; need to disallow=all before we can use allow
;allow=gsm         ; GSM
;allow=ulaw        ; ISDN US
;allow=alaw        ; ISDN EU
```

```
[optionsBTS](!)
;type=peer
;context=from-openBTS
;dtmfmode=rfc2833
;canreinvite=no
,qualify=no        ; openbts do not support OPTION
;insecure=port,invite
```

;If you need to make any changes please add them to sip-custom-contexts.conf
#include sip-custom-contexts.conf

```
[IMS1732111140142310](phones)
callerid=103 ;Rene
;username=103
```

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

```

type=friend
context=sip-sim
;allowssubscribe=yes
disallow=all
;allow=ulaw
;allow=alaw
allow=gsm
host=dynamic
;port=5063
nat=no
dtmfmode=info

```

```

[IMSI732101511208288](phones)
callerid=101 ;Juan
;username=101
;canreinvite=no
type=friend
context=sip-sim
;allowssubscribe=yes
disallow=all
;allow=ulaw
;allow=alaw
allow=gsm
host=dynamic
;port=5064
nat=no
dtmfmode=info

```

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

```
[IMS1732123907813398](phones)
```

```
callerid=102 ;Andres
;username=102
canreinvite=no
type=friend
context=sip-sim
;allowssubscribe=yes
disallow=all
;allow=ulaw
;allow=alaw
allow=gsm
host=dynamic
;port=5065
nat=no
dtmfmode=info
```

Abrir el archivo extensions.conf

```
openvt gedit extensions.conf
```

En el archivo **extensions.conf** debe ir lo siguiente:

```
#include extensions-range.conf ;System dialplan
```

```
[globals](+)
```

```
[default](+);Here you can add any
dialplan the phones must be able to call internaly
```

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

```
#include extensions-range-test.conf ;Test numbers,
should be removed in a production system
```

```
[macro-dialGSM]
```

```
[sip-sim]
include=saliente
exten => 103,1,Dial(SIP/IMSI732111140142310@127.0.0.1:5062)
exten => 101,1,Dial(SIP/IMSI732101511208288@127.0.0.1:5062)
exten => 102,1,Dial(SIP/IMSI732123907813398@127.0.0.1:5062)
```

```
[saliente]
include=sip-sim
exten=>2101,1,Dial(SIP/501/${EXTEN})
exten=>2101,n,Hangup()
```

```
echo test:
exten => 1000,1,Answer()
exten => 1000,2,Echo
exten => 1000,3,Hangup
```

4.3. Guía para Instalación de Planta BTS

El procedimiento para restablecer valores predeterminados de la planta es el siguiente: Ubicar en la planta Panasonic la tarjeta *IPCMPR* el *switch* ubicado bajo el botón reset, se debe accionar hacia arriba (*System Initialize*). Se enciende la planta desde el botón Power. Esperar 20 segundos, y devolver el *switch* hacia la posición inicial (*Normal*) de la tarjeta *IPCMPR*.

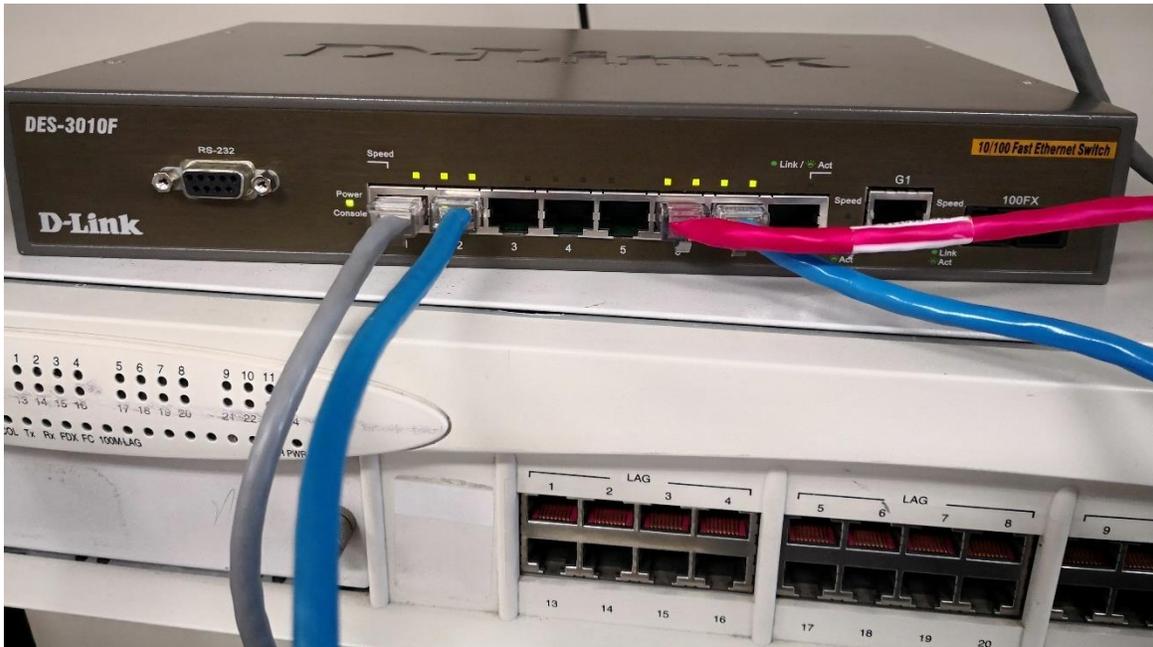


Figura 1. D-Link DES-3010F
 Fuente: Autores



Figura 2. NI USRP-2900
 Fuente: Autores

	<p style="text-align: center;">INFORME FINAL DE TRABAJO DE GRADO</p>	Código	FDE 089
		Versión	03
		Fecha	2015-01-22



Figura 3. Planta Panasonic KX TDE 100
Fuente: Autores

Conectar un cable Ethernet desde el puerto LAN de la planta, hacia el PC donde se gestionará la configuración. Cambiar la configuración IP en el adaptador de red y que esté dentro del rango IP por defecto para acceder a la planta, el cual es 192.168.0.101. Ingresar al programa PBX Unified Maintenance Console. Ingresar como INSTALLER



Figura 4. PBX Unified Maintenance Console
Fuente: Autores

Posteriormente, presionamos clic en “Conectar”

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

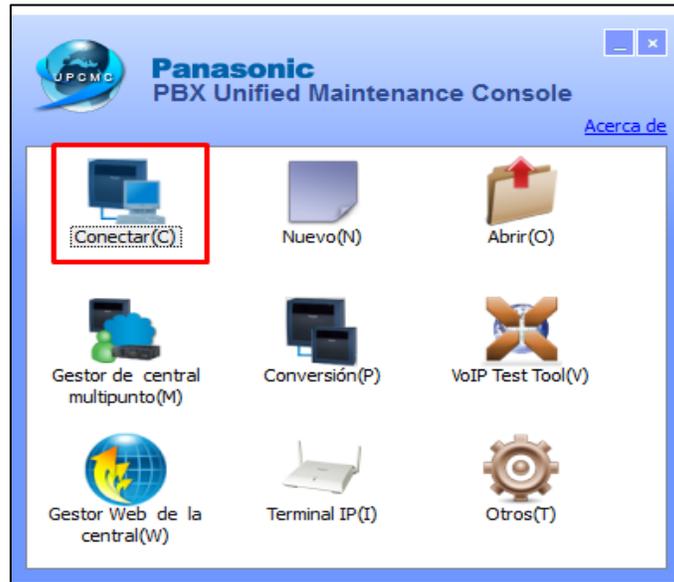


Figura 5. Panasonic Pbx Unified Maintenance Console
Fuente: Autores

Seleccionar la pestaña “Red LAN”. Digitar la IP mencionada anteriormente (192.168.0.101), especificar el puerto 35300 y la contraseña por defecto 1234

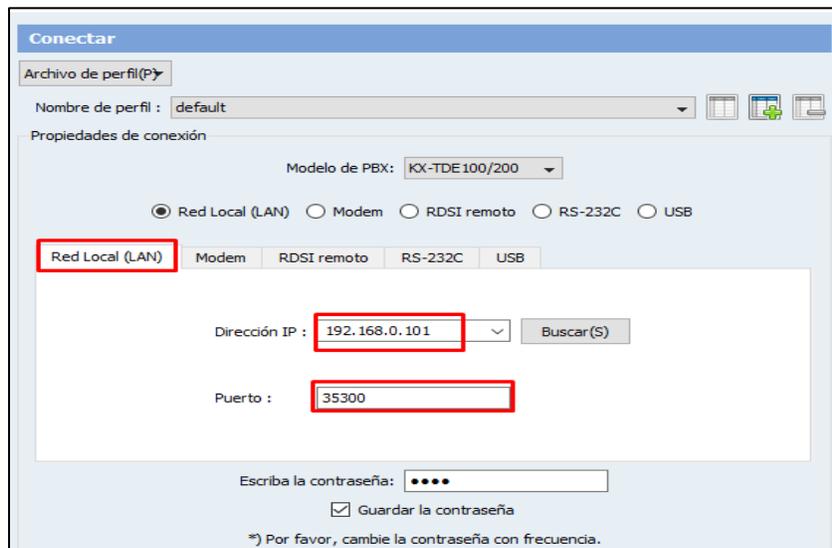


Figura 6. Conectar(C)
Fuente: Autores

A las ventanas que nos aparezcan a continuación, presionamos clic en OK o Aceptar, hasta llegar a la configuración IP.

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

Configuramos la dirección IP del adaptador de red de la planta y de la tarjeta VoIP, así como la máscara de subred y el gateway. Tener en cuenta que deben estar en el segmento de red del laboratorio para poder realizar las configuraciones en el desarrollo del proyecto.

Descripción	Valor
Cliente DHCP	Desactivar
Dirección MAC para la tarjeta IPCMPR	00:80:F0:3C:C2:D6
Dirección IP para la tarjeta IPCMPR	10.2.10.202
Dirección MAC para VoIP-DSP	00:80:F0:3C:C2:D7
Dirección IP para VoIP-DSP	10.2.10.203
Máscara de Subred	255.255.255.0
Gateway por defecto	10.2.10.1
Velocidad / Conexión bidireccional para el puerto principal	Auto negociación
Velocidad / Conexión bidireccional para el puerto de mantenimiento	Auto negociación
Tiempo para desconexión de la LAN	5 s
Número de puerto DHCP	67
DHCP Time Out	10 s

Figura 7. Descripción Y Valor

Fuente: Autores

Para que la planta asuma los cambios que se realizaron, es necesario reiniciarla desde el mismo software, ubicando en la parte superior el menú “Utilidades” presionar clic al final en “Reinicio de Sistema” y luego “Reinicio por comando”

Esperamos a que los leds en la planta telefónica se establezcan en color verde al mismo tiempo que vamos modificando nuevamente la IP del adaptador de red del PC para ingresarlo al segmento de red del laboratorio. Conectar la planta y el PC al *switch*.

4.3.1. Configuración de la troncal SIP en la planta Panasonic KX TDE 100:

Para realizar la configuración de la troncal, se deben seguir los siguientes pasos ubicados en el Menú del sistema:

Clic en “1. Configuración” y seleccionamos “1. Ranura”

Clic en “Clave de act” y verificar que “Número de IP-GW activados” y “Número de IP Softphone activados” esté en 0 (cero), clic en “aplicar” y en “ok”

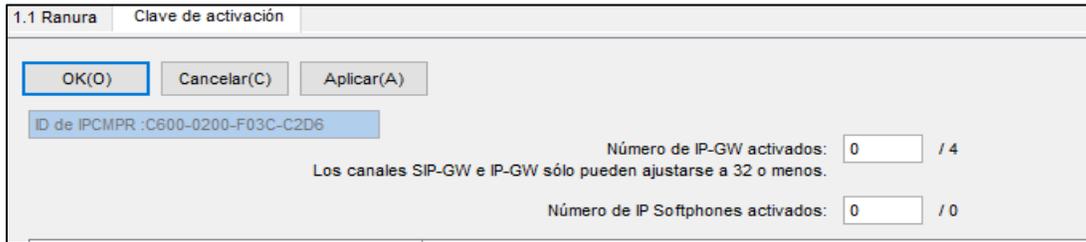


Figura 8. Clave De Activación
Fuente: Autores

En el menú de tarjetas virtuales ubicado en el recuadro blanco a la derecha, seleccionamos la tarjeta **“V-SIPGW16”** y la llevamos hacia el Slot de tarjetas virtuales (azul claro) para troncales (Trunk). Apagar la tarjeta (OUS) y luego ingresamos a propiedades del puerto (Prop. puerto)

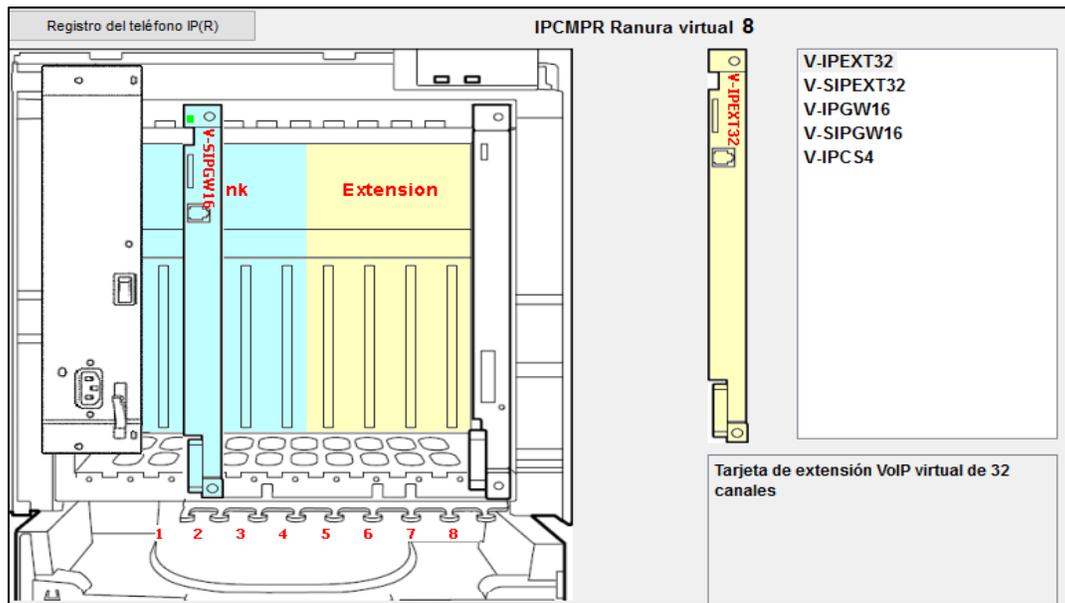


Figura 9. IPCMPR Ranura Virtual 8
Fuente: Autores

En la pestaña **“Principal”** ubicar la columna **“Atributos de canal”** y seleccionar para el 1 **“Basic Channel”**, para el 2, 3 y 4, **“Additional channel for Ch 1”**

En las columnas, **“Nombre de proveedor”**, **“Nombre”** y **“Número de abonado”** poner el nombre de la troncal, para nuestro caso fue el **501**. En la columna **“Dirección IP”**,

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

especificamos la IP de la planta telefónica **Asterisk**. Verificar que el puerto en la columna **“Nombre del servidor SIP”** sea el **5060** (por defecto)

Principal	Cuenta	Registrar	NAT	Opción	Interlocutor llamante	Interlocutor llamado	Voz / FAX	RTP/RTCP	T.38	Opción
-	Ubicación	Ranura	Puerto	Atributos de canal		Nombre de proveedor (20 caracteres)	Ubicación del servidor SIP			
							Nombre (20 caracteres)	Dirección IP		
	1	2	1		Basic channel	501	501	10.2.10.200		
	2	2	2		Additional channel for Ch1					
	3	2	3		Additional channel for Ch1					
	4	2	4		Additional channel for Ch1					

Figura 10. Pestaña “Principal”
Fuente: Autores

Ubicar la pestaña **“Cuenta”** y en las columnas **“Nombre y “ID de autenticación”** especificar el nombre de la troncal (501). En la columna **“Contraseña de autenticación”** la contraseña para autenticarse con la troncal, para nuestro caso es **“passwd”**

Principal	Cuenta	Registrar	NAT	Opción	Interlocutor llamante	Interlocutor llamado	Voz / FAX	RTP/RTCP	T.38	Opción
-	Ubicación	Ranura	Puerto	Conexión	Nombre (64 caracteres)	ID de autenticación (64 caracteres)	Contraseña de autenticación (32 caracteres)			
	1	2	1	OUS	501	501	passwd			
	2	2	2	OUS						

Figura 11. Pestaña “Cuenta”
Fuente: Autores

En la pestaña **“Registrar”** la columna **“Nombre”** debe tener el nombre de la troncal (501) y en la **“Dirección IP”**, especificamos la IP de la planta telefónica **Asterisk**. Clic en **“OK”**

Principal	Cuenta	Registrar	NAT	Opción	Interlocutor llamante	Interlocutor llamado	Voz / FAX	RTP/RTCP	T.38	Opción
-	Ubicación	Ranura	Puerto	Registrar capacidad	Registrar intervalo de envío (s)	Dar de baja la capacidad cuando el puerto INS	Servidor Registrar			Núm Serv
							Nombre (20 caracteres)	Dirección IP		
	1	2	1	ivar	3600	Activar	501	10.2.10.200		5060

Figura 12. Pestaña “Registrar”
Fuente: Autores

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

Finalmente, debemos encender la tarjeta **“V-SIPGW16”**, ubicándose en la misma y presionando clic en **INS**. Luego se debe verificar que los puertos hayan quedado **“INS”** (en servicio), ingresando a propiedades del puerto (Prop. puerto)

-	Ubicación	Ranura	Puerto	Conexión	Atributos de canal	Nombre de proveedor (20 caracteres)
	1	2	1	INS	Basic channel	501
	2	2	2	INS	Additional channel for Ch1	
	3	2	3	INS	Additional channel for Ch1	
	4	2	4	INS	Additional channel for Ch1	

Figura 13. Clic en INS
Fuente: Autores

4.3.2. Configurar salida de llamadas a planta telefónica Asterisk

Clic en **“2. Sistema”** y seleccionamos **“6 Plan de numeración”** y **“1. Principal”**. Estando allí, ubicamos la pestaña **“Extensión”** y en la columna **“Marcación inicial”** especificar los primeros dos dígitos de las extensiones internas, para nuestro caso iniciamos con la **21** en secuencia hasta la **26**. Tener en cuenta que las **XX** que hay en la columna **“Número de dígitos adicionales”**, serán la posición física del teléfono en el **Patch Panel**, en nuestro caso, lo ubicamos en la posición 1, por lo tanto, la extensión de nuestro teléfono análogo será la **2101**

Extensión	Función	Cód. Acceso Otra PBX	KX-T7710	
-	Ubicación	Función	Marcación inicial (3 dígitos)	Nº de dígitos adicionales
1		Bloque de extensión 01	21	XX
2		Bloque de extensión 02	22	XX
3		Bloque de extensión 03	23	XX
4		Bloque de extensión 04	24	XX
5		Bloque de extensión 05		XX
6		Bloque de extensión 06		XX
7		Bloque de extensión 07		XX
8		Bloque de extensión 08		XX
9		Bloque de extensión 09		XX
10		Bloque de extensión 10		XX
11		Bloque de extensión 11		XX
12		Bloque de extensión 12		XX
13		Bloque de extensión 13		XX
14		Bloque de extensión 14		XX
15		Bloque de extensión 15		XX
16		Bloque de extensión 16		XX
17		Bloque de extensión 17		XX
18		Bloque de extensión 18		XX
19		Bloque de extensión 19		XX
20		Bloque de extensión 20		XX
21		Bloque de extensión 21	25	XX
22		Bloque de extensión 22	26	XX

Figura 14. Columna “Marcación Inicial”

Fuente: Autores

Clic en la pestaña “Cod. Acceso Otra PBX”. En la columna “Marcar” especificar el primer dígito de marcación de los números celulares, para nuestro caso, sería el 1 (uno). Clic en “Aplicar” y “OK”

Extensión	Función	Cód. Acceso Otra PBX	KX-T7710
Ubicación	Función		Marcar (3 dígitos)
1	Númeración de otra PBX (Red Privada) 01		1
2	Númeración de otra PBX (Red Privada) 02		

Figura 15. Pestaña “Cod. Acceso Otra PBX”

Fuente: Autores

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

Clic en **“9. Red Privada”**, ubicar la pestaña **“Prioridad 1 - 2”** y en la columna **“Grupo de líneas externas”** especificar el canal de acceso a la troncal, en este caso es el 1 (uno).

Prioridad 1 - 2		Prioridad 3 - 4	Prioridad 5 - 6	Prioridad 7 - 8	CV Centralizado			
-	Ubicación	Núm. Inicio de marcación (3 dígitos)	Prioridad 1			Prioridad 2		
			Núm. de dígitos eliminados	Núm. a añadir (32 dígitos)	Grupo de líneas externas	Núm. de dígitos eliminados	Núm. a añadir (32 dígitos)	Grupo de líneas externas
	1	1	0		1	0		Ninguno

Figura 16. Pestaña “Prioridad 1 - 2”
Fuente: Autores

Clic en **“10. LN y Entrada de llamadas”** y luego **“1. LN”**. Se debe garantizar que la tarjeta virtual **“V-SIPGW16”** esté en el **“Número de grupo de L.N”** 1 (uno)

Nº de LN	Armario	Ranura	Puerto	Tipo de tarjeta	Nombre LN (20 caracteres)	Número de grupo de LN
1	Físico	2	1	LCOT8		1
2	Físico	2	2	LCOT8		1
3	Físico	2	3	LCOT8		1
4	Físico	2	4	LCOT8		1
5	Físico	2	5	LCOT8		1
6	Físico	2	6	LCOT8		1
7	Físico	2	7	LCOT8		1
8	Físico	2	8	LCOT8		1
9	Virtual	2	1	V-SIPGW16		1
10	Virtual	2	2	V-SIPGW16		1
11	Virtual	2	3	V-SIPGW16		1
12	Virtual	2	4	V-SIPGW16		1

Figura 17. “Número de grupo de L.N”
Fuente: Autores

Dentro del menú **“10. LN y Entrada de llamadas”**, clic en **“2 Prog. DIL/DDI/MSN/TIE”**. En la pestaña **“DIL”** Especificamos el **“Método de distribución”** DIL para los primeros 4 canales virtuales de la tarjeta **“V-SIPGW16”**. Clic en **“Aplicar”** y **“OK”**

DIL								
CLI para DIL		DDI / DID / TIE / MSN						
Ajuste de Destino(S)								
-	Ubicación	Armario	Ranura	Puerto	Tipo de tarjeta	Propiedad de línea externa	Método de distribución	Día
	1	Físico	2	1	LCOT8		DIL	
	2	Físico	2	2	LCOT8		DIL	
	3	Físico	2	3	LCOT8		DIL	
	4	Físico	2	4	LCOT8		DIL	
	5	Físico	2	5	LCOT8		DIL	
	6	Físico	2	6	LCOT8		DIL	
	7	Físico	2	7	LCOT8		DIL	
	8	Físico	2	8	LCOT8		DIL	
	9	Virtual	2	1	V-SIPGW16	Público	DIL	
	10	Virtual	2	2	V-SIPGW16	Público	DIL	
	11	Virtual	2	3	V-SIPGW16	Público	DIL	
	12	Virtual	2	4	V-SIPGW16	Público	DIL	

Figura 18. Pestaña "DIL"
Fuente: Autores

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

5. CONCLUSIONES, RECOMENDACIONES Y TRABAJO FUTURO

6.1. Conclusiones

Se realizó la descripción de los conceptos utilizados en la implementación de la solución, para ello se tomó en cuenta un prototipo de una estación celular BTS con parámetros programables y ajustables, y se concluye que esta síntesis permite emular una red celular integrada a la telefonía pública conmutada que la información se comunique en forma vertiginosa y sencilla siendo esta fundamental para los dispositivos móviles.

Con la identificación en desarrollos científicos y de tendencia en tecnología se acerca nuevos avances en la radio comunicación celular para así analizar la influencia de la radio definida por software en la unión y desarrollo de las comunicaciones móviles.

Luego de explicar la situación actual se analizó los requerimientos del sistema y a partir de ellos se logró la Integración del software GNU-Radio, OpenBTS y Planta telefónica TDE 100 usando un sistema operativo Linux.

Se diseñó la solución y a partir de ello se concluye que la solución ofrece que permita emular un prototipo de red celular BTS con los componentes físicos de un equipo de hardware como los filtros, barridos de frecuencias, receptor y analizador de espectro, desarrollando líneas de código en el lenguaje de programación Python.

Se implementó con éxito el sistema en el equipo servidor y el terminal móvil, además de ello se hizo un caso de estudio con el fin de realizar pruebas de funcionamiento del prototipo implementado, además, documentar y construir una guía de uso para ser entregada a la comunicada académica.

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

6.2. Recomendaciones

Se sugieren, en lo subsiguiente, algunas recomendaciones a tener en cuenta sobre el presente proyecto. Se encomienda que el equipo sobre el cual se encuentre montado el servidor de aplicaciones y el servidor de base de datos sea de alto cometido puesto que tal se sabe la representación de las conexiones realizadas es proporcional a la cantidad de memoria RAM utilizable para este fin.

6.3. Trabajo Futuro

Tomando como base este proyecto se pueden derivar distintos trabajos, entre ellos se tiene: Implementar un servicio más avanzado usando redes celulares de tercera y cuarta generación. Asimismo, añadir funcionalidades adicionales a la aplicación como por ejemplo video llamadas.

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

REFERENCIAS

- Angulo, H. &. (Nov. de 2012). Obtenido de
<http://journal.espe.edu.ec/index.php/maskay/article/download/150/144>
- Angulo, H., & Paredes, C. M. (2012). SDR for Ecuadorian Army, through the use of a USRP and Simulink tool of MATLAB. *ESPE*, 68. Obtenido de
<http://journal.espe.edu.ec/index.php/maskay/article/download/150/144>
- Ettus Research. (2012). *USRP Series*. Obtenido de
https://www.ettus.com/content/files/06983_Ettus_USRP1_DS_Flyer_HR.pdf
- Ettus Research. (2017). *USRP Hardware Driver and USRP Manual*. Recuperado el Mayo de 2017, de
http://files.ettus.com/manual/page_devices.html
- GNU-Radio Official. (2012). GNU Radio. Obtenido de
<http://gnuradio.org/redmine/projects/gnuradio/wiki>
- GNU-Radio Official. (2012). *GNU Radio*. Obtenido de
<http://gnuradio.org/redmine/projects/gnuradio/wiki>
- Guided Tutorial GNU Radio in Python. (2017). GNU Radio. Obtenido de
https://wiki.gnuradio.org/index.php/Guided_Tutorial_GNU_Radio_in_Python
- Guided Tutorial GNU Radio in Python. (2017). *Guided Tutorial GNU Radio in Python*. Obtenido de
https://wiki.gnuradio.org/index.php/Guided_Tutorial_GNU_Radio_in_Python
- Hamza, F. A. (11 de Jun de 2008). *Scribd*. Obtenido de
<https://es.scribd.com/document/9688095/USRP-Documentation>
- Hamza, F. A. (2008). *The USRP Under 1.5X Magnifying Lens*. Boston, MA: GNURADIO Community.
 Obtenido de <https://es.scribd.com/document/9688095/USRP-Documentation>
- Montero-Hidalgo, J. P. (s.f.). Obtenido de
<http://arantxa.ii.uam.es/~jms/pfcsteleco/lecturas/20140130JuanPabloMonteroHidalgo.pdf>
- Montero-Hidalgo, J. P. (2014). *Implementación de un sistema de comunicaciones basado en Software Radio*. Madrid: Universidad Autónoma de Madrid. Obtenido de
<http://arantxa.ii.uam.es/~jms/pfcsteleco/lecturas/20140130JuanPabloMonteroHidalgo.pdf>

	<p style="text-align: center;">INFORME FINAL DE TRABAJO DE GRADO</p>	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

- Müller, A. (2008). Obtenido de
<https://pdfs.semanticscholar.org/185d/3473451a01a328c106c7234bb7326fe5bcf7.pdf>
- Müller, A. (2008). *Software Receiver Implementation*. Zürich: Swiss Federal Institute Of Technology Zurich. Obtenido de
<https://pdfs.semanticscholar.org/185d/3473451a01a328c106c7234bb7326fe5bcf7.pdf>
- Ni, W. C. (6 de Marzo de 2015). Software-defined online small cell management. *IEEE Wireless Communications*, 22(1), 52-60. *Graph theory and its applications to future network planning*, <http://ieeexplore.ieee.org>.
- Ni, W., Collings, I. B., Lipman, J., & Wang, X. (6 de Marzo de 2015). Graph theory and its applications to future network planning: software-defined online small cell management. *IEEE Wireless Communications*, 22(1), 52-60. Obtenido de
<http://ieeexplore.ieee.org/abstract/document/7054719/>
- Research, E. (2012). Ettus. Obtenido de
https://www.ettus.com/content/files/06983_Ettus_USRP1_DS_Flyer_HR.pdf
- Valeiro, D. (2008). Open Source Software-Defined Radio: A survey on GNU-radio and its applications. *Forschungszentrum Telekommunikation Wien, Technical Report*. Obtenido de
<https://pdfs.semanticscholar.org/90cd/fd630dabf4ea75aea53bbc9c22ae2367e737.pdf>
- Valerio, D. (2008). Open Source Software-Defined Radio. Obtenido de
<https://pdfs.semanticscholar.org/90cd/fd630dabf4ea75aea53bbc9c22ae2367e737.pdf>

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

FIRMA ESTUDIANTES _____

FIRMA ASESOR _____

FECHA ENTREGA: _____

FIRMA COMITÉ TRABAJO DE GRADO DE LA FACULTAD _____

RECHAZADO__ ACEPTADO__ ACEPTADO CON MODIFICACIONES_____

ACTA NO. _____

FECHA ENTREGA: _____

FIRMA CONSEJO DE FACULTAD _____

ACTA NO. _____

FECHA ENTREGA: _____