

# ESTACIÓN DE COMUNICACIÓN MÓVIL GSM BASADA EN SDR



**Diego Fernando Galvis Calambás  
José Luis Sandoval Jácome**

**Universidad del Cauca  
Facultad de Ingeniería Electrónica y Telecomunicaciones  
Departamento de Telecomunicaciones  
Grupo de I+D GNTT  
MAESTRÍA EN TELECOMUNICACIONES  
Popayán, Cauca  
2022**

# ESTACIÓN DE COMUNICACIÓN MÓVIL GSM BASADA EN SDR



Trabajo de grado presentado como requisito para obtener el título de Magister en  
Telecomunicaciones

**Diego Fernando Galvis Calambás**  
**José Luis Sandoval Jácome**

ANEXO 1  
INSTALACIÓN DE ESTACIÓN DE COMUNICACIÓN MÓVIL GSM BASADA EN  
SDR

**Universidad del Cauca**  
**Facultad de Ingeniería Electrónica y Telecomunicaciones**  
**Departamento de Telecomunicaciones**  
**Grupo de I+D GNTT**  
**MAESTRÍA EN TELECOMUNICACIONES**  
**Popayán, Cauca**  
**2022**

## Tabla de contenido

<b>1</b>	<b>INSTALACIÓN INFRAESTRUCTURA DE RED.....</b>	<b>1</b>
1.1	<i>Instalación de sistema Operativo .....</i>	3
1.1.1	Actualización del Sistema Operativo Ubuntu .....	3
<b>2</b>	<b>CONFIGURACIÓN ENTORNO DE DESARROLLO OPENBTS .....</b>	<b>6</b>
2.1	<i>Actualización e Instalación de Paquetes Adicionales.....</i>	14
2.1.1	Instalación y Compilación del Software DEB .....	15
2.1.2	Activación Reenvío de Datos y Configuración de Tablas IP .....	17
2.1.3	Carga de Base de Datos de Componentes.....	19
2.2	<i>Configuración Asterisk.....</i>	20

## Lista de ilustraciones

Ilustración 1. Conexión USRP a servidor físico .....	1
Ilustración 2. Validación USRP B210 disponible en PC Windows.....	2
Ilustración 3. Especificaciones Máquina virtual .....	3
Ilustración 4. Actualización sistema operativo Ubuntu 1/5 .....	3
Ilustración 5. Actualización sistema operativo Ubuntu 2/5 .....	4
Ilustración 6. Actualización sistema operativo Ubuntu 3/5 .....	4
Ilustración 7. Actualización sistema operativo Ubuntu 4/5 .....	4
Ilustración 8. Actualización sistema operativo Ubuntu 5/5 .....	5
Ilustración 9. Servidor Ubuntu visualización USRP sin controladores.....	5
Ilustración 10. OpenBTS creando carpeta SDR .....	6
Ilustración 11. Configuración OpenBTS clonando Asterisk .....	6
Ilustración 12. Configuración OpenBTS clonando NodeManager – OpenBTS.....	7
Ilustración 13. Configuración OpenBTS clonando Smqueue .....	7
Ilustración 14. OpenBTS clonando subscriberRegistry – System Config .....	8
Ilustración 15. Configuración OpenBTS clonando CommonLibs.....	8
Ilustración 16. Configuración OpenBTS conmutando a master.....	9
Ilustración 17. Configuración OpenBTS actualización componentes con master ...	9
Ilustración 18. Instalación USRP B210.....	10
Ilustración 19. Descarga manual firmware USRP .....	10
Ilustración 20. Validación instalación USRP.....	11
Ilustración 21. USRP como dispositivo genérico en Ubuntu 16.04.....	11
Ilustración 22. Selección del dispositivo USRP en Ubuntu.....	11
Ilustración 23. USRP vista por servidor Ubuntu .....	12
Ilustración 24. USRP seleccionada para el servidor Ubuntu .....	12
Ilustración 25. Prueba funcional de la USRP línea de comandos 1/2 .....	12
Ilustración 26. Prueba funcional de la USRP línea de comandos 2/2 .....	13
Ilustración 27. Paquetes complementarios OpenBTS.....	13
Ilustración 28. Actualización e instalación paquetes adicionales 1/3.....	14
Ilustración 29. Actualización e instalación paquetes adicionales 2/3.....	14
Ilustración 30. Actualización e instalación paquetes adicionales 3/3.....	14
Ilustración 31. Instalación y compilación del software DEB 1/7 .....	15
Ilustración 32. Instalación y compilación del software DEB 2/7 .....	16
Ilustración 33. Instalación y compilación del software DEB 3/7 .....	16
Ilustración 34. Instalación y compilación del software DEB 4/7 .....	16
Ilustración 35. Instalación y compilación del software DEB 5/7 .....	17
Ilustración 36. Instalación y compilación del software DEB 6/7 .....	17
Ilustración 37. Instalación y compilación del software DEB 7/7 .....	17
Ilustración 38. Activación reenvío datos, configuración tablas IP en Ubuntu .....	18
Ilustración 39. Salida de configuración de iptables.....	18
Ilustración 40. Descripción tarjeta de red servidor Ubuntu .....	18
Ilustración 41. Salida restablecimiento tablas IP .....	19
Ilustración 42. Carga BD OpenBTS.....	19
Ilustración 43. Carga BD subscriberRegistry.....	20
Ilustración 44. Carga BD smqueue.....	20

Ilustración 45. Archivos de configuración Asterisk .....	20
Ilustración 46. Modificación archivo configuración asterisk sip.conf.....	21
Ilustración 47. Modificación archivo configuración asterisk extensions.conf .....	22
Ilustración 48. Relación sip.conf - extensions.conf .....	23

## Lista de tablas

Tabla 1. IMSI teléfonos móviles .....	20
Tabla 2. Resumen MS prueba.....	23



## 1 INSTALACIÓN INFRAESTRUCTURA DE RED

En la instalación de la infraestructura de red se presenta el paso a paso de cada uno de los componentes requeridos para la implementación de la estación de comunicación, iniciando por la compatibilidad hardware software entre los equipos y los componentes software.

Dentro de los prerrequisitos de la instalación de la infraestructura de red se debe tener en cuenta que, la plataforma hardware USRP B210, debe ser reconocida inicialmente por el PC (en este caso Windows 10) que aloja el servidor virtualizado Ubuntu 16.04.6 LTS, para posteriormente sea vista por este servidor y de esta manera pueda interactuar con el paquete software OpenBTS, para ello es necesario que se encuentre instalado el firmware de la USRP B210 en el dispositivo hardware, para que este a su vez pueda enseñarlo a la máquina virtual en donde estará alojado el paquete software OpenBTS; en otras palabras el firmware proporcionará compatibilidad con el controlador software de la plataforma hardware.

Los controladores se pueden descargar desde la página del *Ettus Research* en el siguiente enlace [https://files.ettus.com/manual/page\\_install.html](https://files.ettus.com/manual/page_install.html).

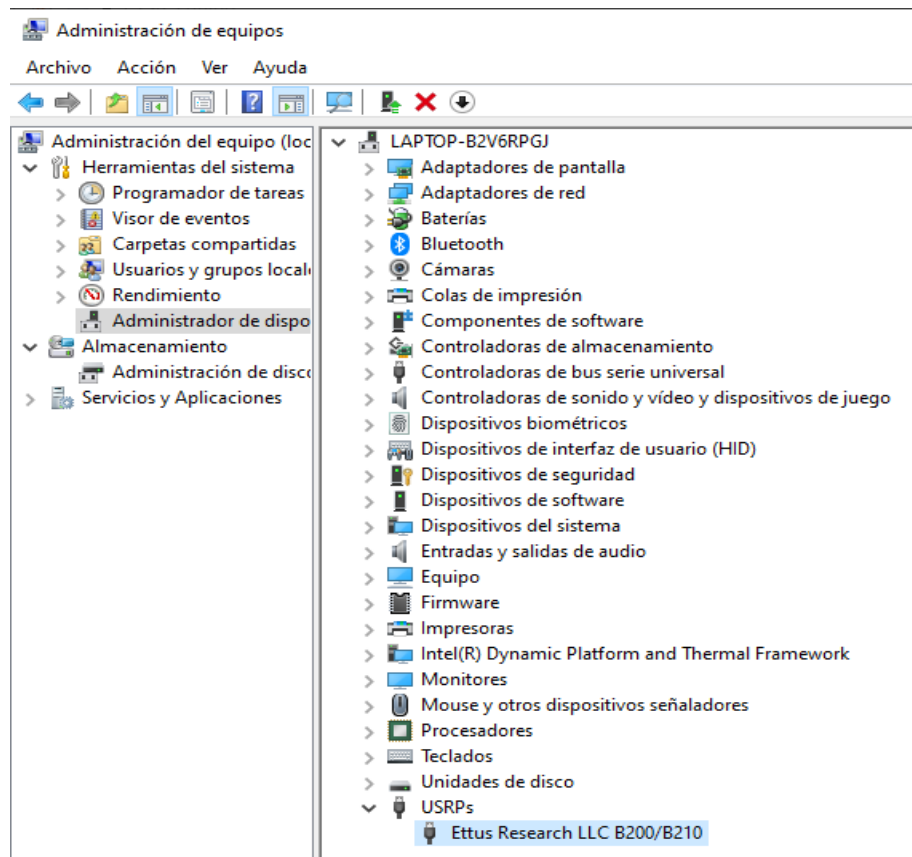
Para que el dispositivo USRP B210 sea reconocido debe estar conectado al puerto USB 3.0 del PC, tal como se puede identificar en la Ilustración 1, teniendo en cuenta que previo a su conexión se deben cumplir unas condiciones básicas para salvaguardar la integridad del dispositivo, una de ellas es que en el puerto mixto de transmisión-recepción debe estar conectada la antena respectiva, en este caso particular Vert900, también el dispositivo posee una fuente de poder la cual no es necesario su uso a menos que se pretenda trabajar con niveles de potencia elevados, en su defecto solo es necesaria su conexión mediante el puerto USB 3.0.



**Ilustración 1. Conexión USRP a servidor físico**



Una vez el dispositivo USRP B210 sea reconocido por el sistema operativo Windows se puede validar su instalación ingresando al administrador de dispositivos, en donde aparece con su nombre, tal como se puede identificar en la Ilustración 2.

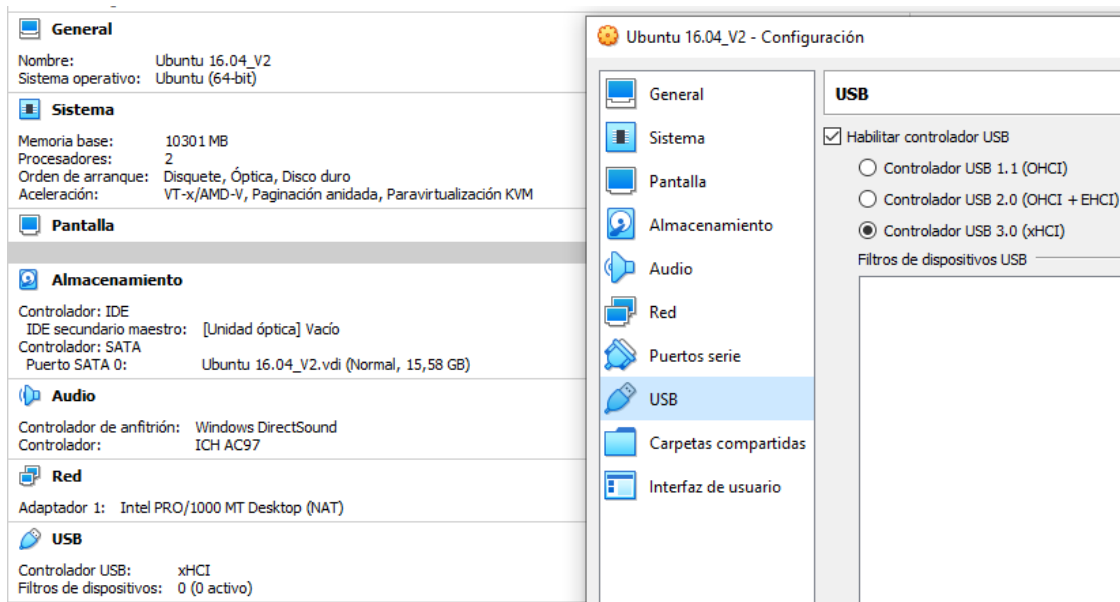


**Ilustración 2. Validación USRP B210 disponible en PC Windows**

El sistema operativo Ubuntu se instaló en una máquina virtualizada con *Oracle VM Virtual Box 6.1.2 r135662* software que puede descargarse en el enlace [https://www.virtualbox.org/wiki/Download\\_Old\\_Builds\\_6\\_1](https://www.virtualbox.org/wiki/Download_Old_Builds_6_1).

En la Ilustración 3 se pueden identificar las especificaciones configuradas en el virtual box; aunque es posible realizar la instalación del sistema operativo con una máquina con menores especificaciones no se recomienda, dado que puede afectar el rendimiento de los servicios prestados por la estación de comunicación móvil.





**Ilustración 3. Especificaciones Máquina virtual**

## 1.1 Instalación de sistema Operativo

En la instalación del sistema operativo se define idioma, nombre de usuario administrador y una recomendación importante es activar la casilla que permite al sistema operativo actualizarse posterior a dar inicio, pues ello previene que las actualizaciones afecten el funcionamiento de los paquetes software de la estación base transceptora.

### 1.1.1 Actualización del Sistema Operativo Ubuntu

La primera actividad que se debe realizar es validar si la máquina virtual configurada es capaz de navegar, para actualizar el sistema operativo Ubuntu 16.04.06, lo cual se realiza ejecutando en una ventana de línea de comandos las siguientes sentencias.

```
| sudo apt-get install software-properties-common python-software-properties  
| sudo add-apt-repository ppa:git-core/ppa  
| sudo apt-get update  
| sudo apt-get install git
```

Las Ilustraciones 4 a 8 muestran pantallazos de la ejecución de la actualización.

```
diegofgalvis@diegofgalvis-VirtualBox:~$ sudo apt-get install software-properti  
es-common python-software-properties  
Leyendo lista de paquetes... Hecho  
Creando árbol de dependencias  
Leyendo la información de estado... Hecho  
software-properties-common ya está en su versión más reciente (0.96.20.10).  
python-software-properties ya está en su versión más reciente (0.96.20.10).  
0 actualizados, 0 nuevos se instalarán, 0 para eliminar y 185 no actualizados.
```

**Ilustración 4. Actualización sistema operativo Ubuntu 1/5**



```
diegofgalvisc@diegofgalvisc-VirtualBox:~$ sudo add-apt-repository ppa:git-core/p
pa
The most current stable version of Git for Ubuntu.

For release candidates, go to https://launchpad.net/~git-core/+archive/candidate
.
Más información: https://launchpad.net/~git-core/+archive/ubuntu/ppa
Pulse [Intro] para continuar o ctrl-c para cancelar

gpg: anillo «/tmp/tmp8jk7kiye/secring.gpg» creado
gpg: anillo «/tmp/tmp8jk7kiye/pubring.gpg» creado
gpg: solicitando clave E1DF1F24 de hkp servidor keyserver.ubuntu.com
gpg: /tmp/tmp8jk7kiye/trustdb.gpg: se ha creado base de datos de confianza
gpg: clave E1DF1F24: clave pública "Launchpad PPA for Ubuntu Git Maintainers" im
portada
gpg: Cantidad total procesada: 1
gpg:          importadas: 1 (RSA: 1)
OK
```

**Ilustración 5. Actualización sistema operativo Ubuntu 2/5**

```
diegofgalvisc@diegofgalvisc-VirtualBox: ~
diegofgalvisc@diegofgalvisc-VirtualBox:~$ sudo apt-get update
[sudo] password for diegofgalvisc:
Obj:1 http://co.archive.ubuntu.com/ubuntu xenial InRelease
Obj:2 http://security.ubuntu.com/ubuntu xenial-security InRelease
Obj:3 http://co.archive.ubuntu.com/ubuntu xenial-updates InRelease
Des:4 http://ppa.launchpad.net/git-core/ppa/ubuntu xenial InRelease [20,8 kB]
Obj:5 http://co.archive.ubuntu.com/ubuntu xenial-backports InRelease
Des:6 http://ppa.launchpad.net/git-core/ppa/ubuntu xenial/main amd64 Packages [3
.336 B]
Des:7 http://ppa.launchpad.net/git-core/ppa/ubuntu xenial/main i386 Packages [3.
332 B]
Des:8 http://ppa.launchpad.net/git-core/ppa/ubuntu xenial/main Translation-en [2
.432 B]
Descargados 29,9 kB en 2s (11,3 kB/s)
Leyendo lista de paquetes... Hecho
```

**Ilustración 6. Actualización sistema operativo Ubuntu 3/5**

```
diegofgalvisc@diegofgalvisc-VirtualBox:~$ sudo apt-get install git
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
  git-man liberror-perl libpcre2-8-0
Paquetes sugeridos:
  git-daemon-run | git-daemon-sysvinit git-doc git-email git-gui gitk gitweb
  git-cvs git-mediawiki git-svn
Se instalarán los siguientes paquetes NUEVOS:
  git git-man liberror-perl libpcre2-8-0
0 actualizados, 4 nuevos se instalarán, 0 para eliminar y 185 no actualizados.
Se necesita descargar 8.995 kB de archivos.
Se utilizarán 39,9 MB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n]
```

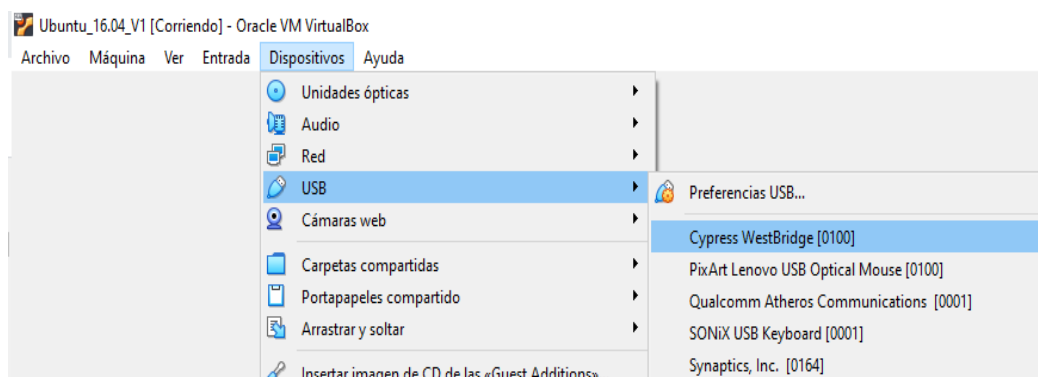
**Ilustración 7. Actualización sistema operativo Ubuntu 4/5**



```
64 10.21-1 [165 kB]
Des:2 http://ppa.launchpad.net/git-core/ppa/ubuntu xenial/main amd64 git-man all
1:2.35.1-0ppa1~ubuntu16.04.1 [1.904 kB]
Des:3 http://co.archive.ubuntu.com/ubuntu xenial/main amd64 liberror-perl all 0.
17-1.2 [19,6 kB]
Des:4 http://ppa.launchpad.net/git-core/ppa/ubuntu xenial/main amd64 git amd64 1
:2.35.1-0ppa1~ubuntu16.04.1 [6.906 kB]
Descargados 8.995 kB en 26s (344 kB/s)
Seleccionando el paquete libpcre2-8-0:amd64 previamente no seleccionado.
(Leyendo la base de datos ... 179480 ficheros o directorios instalados actualmen
te.)
Preparando para desempaquetar .../libpcre2-8-0_10.21-1_amd64.deb ...
Desempaquetando libpcre2-8-0:amd64 (10.21-1) ...
Seleccionando el paquete liberror-perl previamente no seleccionado.
Preparando para desempaquetar .../liberror-perl_0.17-1.2_all.deb ...
Desempaquetando liberror-perl (0.17-1.2) ...
Seleccionando el paquete git-man previamente no seleccionado.
Preparando para desempaquetar .../git-man_1%3a2.35.1-0ppa1~ubuntu16.04.1_all.deb
...
Desempaquetando git-man (1:2.35.1-0ppa1~ubuntu16.04.1) ...
Seleccionando el paquete git previamente no seleccionado.
Preparando para desempaquetar .../git_1%3a2.35.1-0ppa1~ubuntu16.04.1_amd64.deb .
..
Desempaquetando git (1:2.35.1-0ppa1~ubuntu16.04.1) ...
Procesando disparadores para libc-bin (2.23-0ubuntu11.2) ...
Procesando disparadores para man-db (2.7.5-1) ...
Configurando libpcre2-8-0:amd64 (10.21-1) ...
Configurando liberror-perl (0.17-1.2) ...
Configurando git-man (1:2.35.1-0ppa1~ubuntu16.04.1) ...
Configurando git (1:2.35.1-0ppa1~ubuntu16.04.1) ...
Procesando disparadores para libc-bin (2.23-0ubuntu11.2) ...
```

**Ilustración 8. Actualización sistema operativo Ubuntu 5/5**

Cuando el servidor virtual se encuentra actualizado, se puede asignar la USRP como un recurso que pueda administrar el servidor, el cual tiene asignado el servidor físico; este proceso se realiza asignando el recurso mediante el puerto USB 3.0, previamente configurado en las características del servidor virtual, es decir mediante el VirtualBox, la máquina Ubuntu lo reconocerá como un dispositivo genérico tal como se presenta en la Ilustración 9.



**Ilustración 9. Servidor Ubuntu visualización USRP sin controladores**



## 2 CONFIGURACIÓN ENTORNO DE DESARROLLO OPENBTS

La configuración del entorno de desarrollo OpenBTS parte de la creación de una carpeta en el servidor virtualizado, para posteriormente clonar y descargar el paquete de instalación, después se debe compilar el código fuente descargado, debido que a este código se accede mediante la descarga desde un servidor github, se debe garantizar que el servidor virtualizado pueda navegar. Las sentencias a ejecutar desde una ventana de línea de comandos son:

```
| mkdir sdr // Crea una nueva carpeta sdr  
| cd sdr // Ingresa a la carpeta sdr  
| git clone https://github.com/RangeNetworks/dev.git  
| cd dev  
| ./clone.sh //Clona el código de Github  
| ./switchto.sh master // Cambia a maestro  
| ./build.sh B210
```

En las ilustraciones 10 a 17 se puede identificar gráficamente el proceso ejecutado y los resultados en la línea de comandos.

```
diegofgalvis@diegofgalvis-VirtualBox:~$ mkdir sdr  
diegofgalvis@diegofgalvis-VirtualBox:~$ cd sdr  
diegofgalvis@diegofgalvis-VirtualBox:~/sdr$ git clone https://github.com/Range  
Networks/dev.git  
Clonando en 'dev'...  
remote: Enumerating objects: 169, done.  
remote: Counting objects: 100% (11/11), done.  
remote: Compressing objects: 100% (11/11), done.  
remote: Total 169 (delta 6), reused 2 (delta 0), pack-reused 158  
Recibiendo objetos: 100% (169/169), 50.17 KiB | 338.00 KiB/s, listo.  
Resolviendo deltas: 100% (105/105), listo.
```

**Ilustración 10. OpenBTS creando carpeta SDR**

```
diegofgalvis@diegofgalvis-VirtualBox:~/sdr/dev$ ./clone.sh  
# cloning asterisk  
git clone https://github.com/RangeNetworks/asterisk.git  
Clonando en 'asterisk'...  
remote: Enumerating objects: 174, done.  
remote: Total 174 (delta 0), reused 0 (delta 0), pack-reused 174  
Recibiendo objetos: 100% (174/174), 36.63 KiB | 426.00 KiB/s, listo.  
Resolviendo deltas: 100% (72/72), listo.  
git checkout --track origin/3.1  
branch '3.1' set up to track 'origin/3.1'.  
Cambiado a nueva rama '3.1'  
git checkout --track origin/4.0  
branch '4.0' set up to track 'origin/4.0'.  
Cambiado a nueva rama '4.0'  
git checkout --track origin/5.0  
branch '5.0' set up to track 'origin/5.0'.  
Cambiado a nueva rama '5.0'  
git checkout master  
Cambiado a rama 'master'  
Tu rama está actualizada con 'origin/master'.  
git submodule update --init --recursive --remote  
  
# cloning asterisk-config  
git clone https://github.com/RangeNetworks/asterisk-config.git  
Clonando en 'asterisk-config'...  
remote: Enumerating objects: 713, done.  
remote: Total 713 (delta 0), reused 0 (delta 0), pack-reused 713  
Recibiendo objetos: 100% (713/713), 417.15 KiB | 1.22 MiB/s, listo.  
Resolviendo deltas: 100% (431/431), listo.  
git checkout --track origin/3.1  
branch '3.1' set up to track 'origin/3.1'.  
Cambiado a nueva rama '3.1'
```

**Ilustración 11. Configuración OpenBTS clonando Asterisk**



```
# cloning NodeManager
git clone https://github.com/RangeNetworks/NodeManager.git
Clonando en 'NodeManager'...
remote: Enumerating objects: 119, done.
remote: Total 119 (delta 0), reused 0 (delta 0), pack-reused 119
Recibiendo objetos: 100% (119/119), 79.35 KiB | 439.00 KiB/s, listo.
Resolviendo deltas: 100% (44/44), listo.
git checkout --track origin/4.0
branch '4.0' set up to track 'origin/4.0'.
Cambiado a nueva rama '4.0'
git checkout --track origin/5.0
branch '5.0' set up to track 'origin/5.0'.
Cambiado a nueva rama '5.0'
git checkout master
Cambiado a rama 'master'
Tu rama está actualizada con 'origin/master'.
git submodule update --init --recursive --remote

# cloning openbts
git clone https://github.com/RangeNetworks/openbts.git
Clonando en 'openbts'...
remote: Enumerating objects: 3527, done.
remote: Counting objects: 100% (11/11), done.
remote: Compressing objects: 100% (11/11), done.
remote: Total 3527 (delta 0), reused 2 (delta 0), pack-reused 3516
Recibiendo objetos: 100% (3527/3527), 5.68 MiB | 3.12 MiB/s, listo.
Resolviendo deltas: 100% (2379/2379), listo.
git checkout --track origin/3.1
branch '3.1' set up to track 'origin/3.1'.
Cambiado a nueva rama '3.1'
git checkout --track origin/4.0
```

**Ilustración 12. Configuración OpenBTS clonando NodeManager – OpenBTS**

```
# cloning smqueue
git clone https://github.com/RangeNetworks/smqueue.git
Clonando en 'smqueue'...
remote: Enumerating objects: 975, done.
remote: Total 975 (delta 0), reused 0 (delta 0), pack-reused 975
Recibiendo objetos: 100% (975/975), 2.54 MiB | 3.51 MiB/s, listo.
Resolviendo deltas: 100% (616/616), listo.
git checkout --track origin/3.1
branch '3.1' set up to track 'origin/3.1'.
Cambiado a nueva rama '3.1'
git checkout --track origin/4.0
branch '4.0' set up to track 'origin/4.0'.
Cambiado a nueva rama '4.0'
git checkout --track origin/5.0
branch '5.0' set up to track 'origin/5.0'.
Cambiado a nueva rama '5.0'
git checkout master
Cambiado a rama 'master'
Tu rama está actualizada con 'origin/master'.
git submodule update --init --recursive --remote
Submódulo 'CommonLibs' (https://github.com/RangeNetworks/CommonLibs.git) registr
ado para ruta 'CommonLibs'
Submódulo 'NodeManager' (https://github.com/RangeNetworks/NodeManager.git) regis
trado para ruta 'NodeManager'
Submódulo 'SR' (https://github.com/RangeNetworks/subscriberRegistry.git) registr
ado para ruta 'SR'
Clonando en '/home/diegofgalvis/sdr/dev/smqueue/CommonLibs'...
Clonando en '/home/diegofgalvis/sdr/dev/smqueue/NodeManager'...
Clonando en '/home/diegofgalvis/sdr/dev/smqueue/SR'...
Ruta de submódulo 'CommonLibs': check out realizado a '76b71d509bb5e6f146d7305d0
1b4c1cfec21302a'
Ruta de submódulo 'NodeManager': check out realizado a '8c8a6e8f7a93543b60b85cea
```

**Ilustración 13. Configuración OpenBTS clonando Smqueue**



```
Clonando en 'subscriberRegistry'...
remote: Enumerating objects: 486, done.
remote: Total 486 (delta 0), reused 0 (delta 0), pack-reused 486
Recibiendo objetos: 100% (486/486), 162.18 KiB | 691.00 KiB/s, listo.
Resolviendo deltas: 100% (286/286), listo.
git checkout --track origin/3.1
branch '3.1' set up to track 'origin/3.1'.
Cambiado a nueva rama '3.1'
git checkout --track origin/4.0
branch '4.0' set up to track 'origin/4.0'.
Cambiado a nueva rama '4.0'
git checkout --track origin/5.0
branch '5.0' set up to track 'origin/5.0'.
Cambiado a nueva rama '5.0'
git checkout master
Cambiado a rama 'master'
Tu rama está actualizada con 'origin/master'.
git submodule update --init --recursive --remote
Submódulo 'CommonLibs' (https://github.com/RangeNetworks/CommonLibs.git) registrado para ruta 'CommonLibs'
Submódulo 'NodeManager' (https://github.com/RangeNetworks/NodeManager.git) registrado para ruta 'NodeManager'
Clonando en '/home/diegofgalvis/sdr/dev/subscriberRegistry/CommonLibs'...
Clonando en '/home/diegofgalvis/sdr/dev/subscriberRegistry/NodeManager'...
Ruta de submódulo 'CommonLibs': check out realizado a '76b71d509bb5e6f146d7305d01b4c1cfec21302a'
Ruta de submódulo 'NodeManager': check out realizado a '8c8a6e8f7a93543b60b85cea9fb2b3479cd57584'

# cloning system-config
git clone https://github.com/RangeNetworks/system-config.git
Clonando en 'system-config'...
```

**Ilustración 14. OpenBTS clonando subscriberRegistry – System Config**

```
ado para ruta 'CommonLibs'
Submódulo 'NodeManager' (https://github.com/RangeNetworks/NodeManager.git) registrado para ruta 'NodeManager'
Clonando en '/home/diegofgalvis/sdr/dev/subscriberRegistry/CommonLibs'...
Clonando en '/home/diegofgalvis/sdr/dev/subscriberRegistry/NodeManager'...
Ruta de submódulo 'CommonLibs': check out realizado a '76b71d509bb5e6f146d7305d01b4c1cfec21302a'
Ruta de submódulo 'NodeManager': check out realizado a '8c8a6e8f7a93543b60b85cea9fb2b3479cd57584'

# cloning system-config
git clone https://github.com/RangeNetworks/system-config.git
Clonando en 'system-config'...
remote: Enumerating objects: 317, done.
remote: Total 317 (delta 0), reused 0 (delta 0), pack-reused 317
Recibiendo objetos: 100% (317/317), 41.37 KiB | 258.00 KiB/s, listo.
Resolviendo deltas: 100% (193/193), listo.
git checkout --track origin/3.1
branch '3.1' set up to track 'origin/3.1'.
Cambiado a nueva rama '3.1'
git checkout --track origin/4.0
branch '4.0' set up to track 'origin/4.0'.
Cambiado a nueva rama '4.0'
git checkout --track origin/5.0
branch '5.0' set up to track 'origin/5.0'.
Cambiado a nueva rama '5.0'
git checkout master
Cambiado a rama 'master'
Tu rama está actualizada con 'origin/master'.
git submodule update --init --recursive --remote
```

**Ilustración 15. Configuración OpenBTS clonando CommonLibs**



```
diegofgalvisc@diegofgalvisc-VirtualBox:~/sdr/dev$ ./switchto.sh master
#####
# asterisk
Ya en 'master'
Tu rama está actualizada con 'origin/master'.

#####
# asterisk-config
Ya en 'master'
Tu rama está actualizada con 'origin/master'.

#####
# CommonLibs
Ya en 'master'
Tu rama está actualizada con 'origin/master'.

#####
# liba53
Ya en 'master'
Tu rama está actualizada con 'origin/master'.

#####
# libcoredumper
Ya en 'master'
Tu rama está actualizada con 'origin/master'.

#####
# NodeManager
Ya en 'master'
```

**Ilustración 16. Configuración OpenBTS conmutando a master**

```
Ya en 'master'
Tu rama está actualizada con 'origin/master'.

#####
# smqueue
M      NodeManager
M      SR
Ya en 'master'
Tu rama está actualizada con 'origin/master'.
Ruta de submódulo 'NodeManager': check out realizado a '4b60fc9772ab445c46a3d741
d3d34d4bf1fac936'
Ruta de submódulo 'SR': check out realizado a '3cae0655cb7da9b83b1444e872d939d67
6f986fd'
Ruta de submódulo 'SR/CommonLibs': check out realizado a '11d8baa82662c9c862aa0f
e8a5c585c2eb9cfb89'
Ruta de submódulo 'SR/NodeManager': check out realizado a '4b60fc9772ab445c46a3d
741d3d34d4bf1fac936'

#####
# subscriberRegistry
M      NodeManager
Ya en 'master'
Tu rama está actualizada con 'origin/master'.
Ruta de submódulo 'NodeManager': check out realizado a '4b60fc9772ab445c46a3d741
d3d34d4bf1fac936'

#####
# system-config
Ya en 'master'
Tu rama está actualizada con 'origin/master'.
```

**Ilustración 17. Configuración OpenBTS actualización componentes con master**



Con estas sentencias se inicia el proceso de descarga y compilación, el cual puede tardar entre 30 y 45 minutos según la velocidad de descarga de la red y el rendimiento de la máquina al compilar; al finalizar en el sistema operativo Ubuntu 16.04.6 estará instalado automáticamente los entornos GnuRadio y el controlador UHD de la USRP. Tal como lo muestra la Ilustración 18, se han descargado los controladores correspondientes a la USRP B210.

```
dgalvis@Maestria:~/sdr/dev$ ./build.sh B210
# checking for a supported radio type
# - found
# checking for a compatible build host
# - fully supported host detected: Ubuntu 16.04
#
# adding additional repo tools
# - done

# checking build dependencies
dpkg-query: el paquete `autoconf' no está instalado y no hay ninguna información
disponible.
Utilice dpkg --info (= dpkg-deb --info) para examinar archivos,
y dpkg --contents (= dpkg-deb --contents) para listar su contenido.
# - missing autoconf, installing dependency
[sudo] password for dgalvis: █
```

**Ilustración 18. Instalación USRP B210**

Para controlar adecuadamente la USRP debe descargarse de forma adicional el firmware, como lo indica la siguiente sentencia, la cual se puede identificar en la Ilustración 19.

```
| $sudo python /usr/lib/uhd/uhd_images_downloader.py
```

```
diegofgalvis@diegofgalvis:~/sdr/dev$ sudo python /usr/lib/uhd/uhd_images
_downloader.py
Images destination: /usr/share/uhd/images
Downloading images from: http://files.ettus.com/binaries/images/uhd-images_003.0
09.002-release.zip
Downloading images to: /tmp/tmp5m3xu/uhd-images_003.009.002-release.zip
26296 kB / 26296 kB (100%)

Images successfully installed to: /usr/share/uhd/images
```

**Ilustración 19. Descarga manual firmware USRP**

Una vez se haya instalado esta imagen será posible controlar la USRP desde el servidor ubuntu, como prueba de ello se puede mover a la ruta `/sdr/dev/openbts/Transceiver52M/`, tal como se muestra en la Ilustración 20 y ejecutar la sentencia:

```
| Uhd_usrp_probe
```

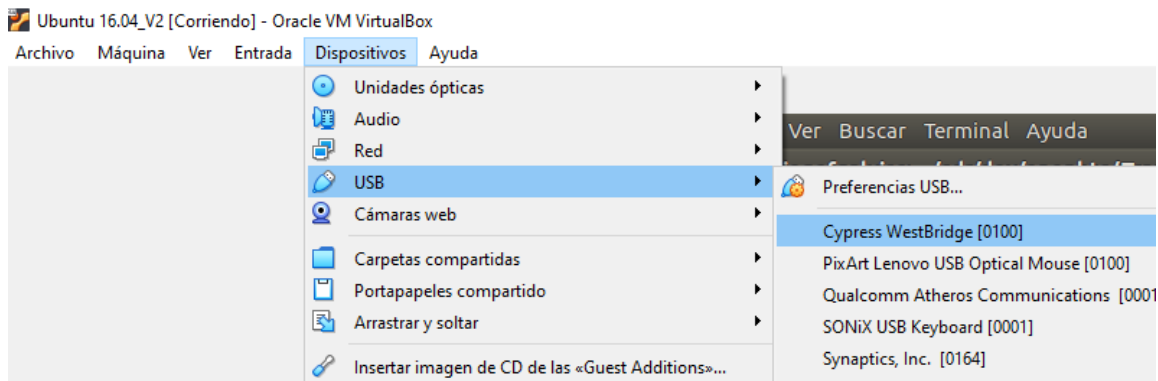




```
diegofgalvis@diegofgalvis:~/sdr/dev/openbts/Transceiver52M$ uhd_usrp_probe  
linux; GNU C++ version 5.3.1 20151219; Boost_105800; UHD_003.009.002-0-unknown  
  
Error: LookupError: KeyError: No devices found for ----->  
Empty Device Address
```

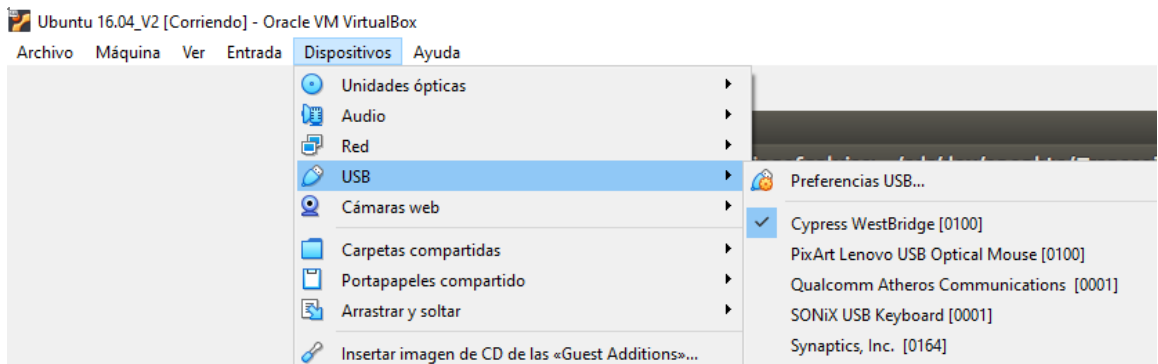
**Ilustración 20. Validación instalación USRP**

En este caso se puede evidenciar que el servidor Ubuntu 16.04 aun no puede encontrar el dispositivo USRP, pues lo está viendo como un dispositivo genérico, tal como se muestra en la Ilustración 21.



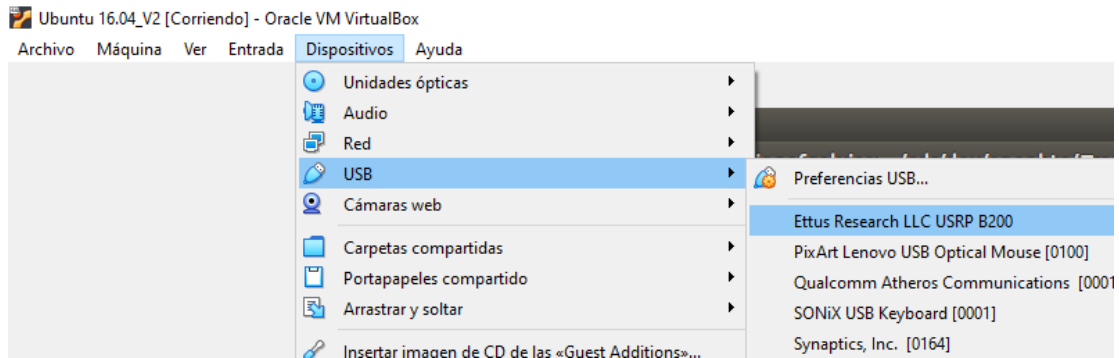
**Ilustración 21. USRP como dispositivo genérico en Ubuntu 16.04**

Como el servidor Ubuntu ya tiene el firmware para controlarlo, se debe proceder a seleccionar el dispositivo genérico asignándolo al puerto USB, y a continuación realizar nuevamente la validación, con la sentencia `uhd_usrp_probe` así como se aprecia en la Ilustración 22.

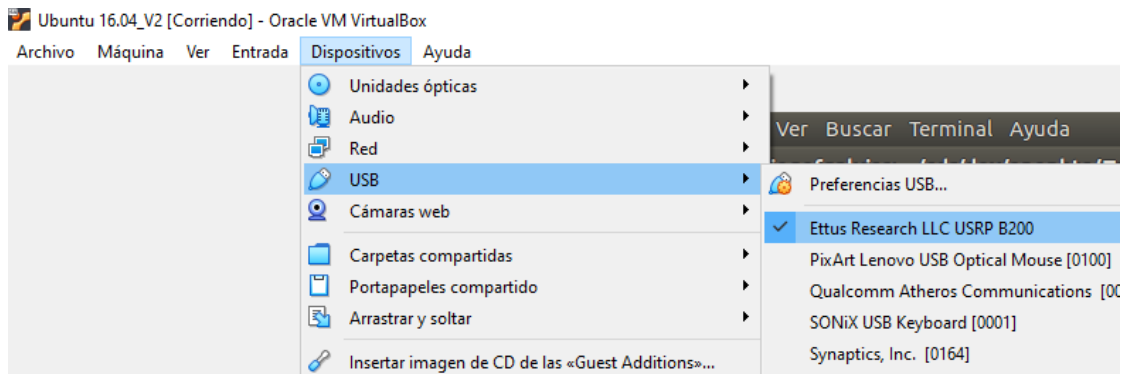


**Ilustración 22. Selección del dispositivo USRP en Ubuntu**

Posterior a asignar el dispositivo genérico se debe ir nuevamente a los dispositivos USB en donde debe aparecer la USRP, tal como aparece en la Ilustración 23, y posteriormente se debe proceder a seleccionar como parece en la Ilustración 24.



**Ilustración 23. USRP vista por servidor Ubuntu**



**Ilustración 24. USRP seleccionada para el servidor Ubuntu**

A continuación nuevamente se debe ejecutar la sentencia para probar la conexión de la USRP, sentencia que permitirá validar que el servidor Ubuntu puede ver la USRP y la puede controlar, esto cuando se tenga una imagen similar en la línea de comandos la que se puede visualizar en la Ilustración 25 y 26.

```
diegofgalvisc@diegofgalvisc:~/sdr/dev/openbts/Transceiver52M$ uhd_usrp_probe
Linux; GNU C++ version 5.3.1 20151219; Boost_105800; UHD_003.009.002-0-unknown
-- Detected Device: B210
-- Loading FPGA image: /usr/share/uhd/images/usrp_b210_fpga.bin... 98%
```

**Ilustración 25. Prueba funcional de la USRP línea de comandos 1/2**



```
diegofgalvisc@diegofgalvisc:~/sdr/dev/openbts/Transceiver52M$ uhd_usrp_probe
linux; GNU C++ version 5.3.1 20151219; Boost_105800; UHD_003.009.002-0-unknown

-- Detected Device: B210
-- Loading FPGA image: /usr/share/uhd/images/usrp_b210_fpga.bin... done
-- Operating over USB 2.
-- Detecting internal GPSDO... No GPSDO found
-- Initialize CODEC control...
-- Initialize Radio control...
-- Performing register loopback test... pass
-- Performing register loopback test... pass
-- Performing CODEC loopback test... pass
-- Performing CODEC loopback test... pass
-- Asking for clock rate 16.000000 MHz...
-- Actually got clock rate 16.000000 MHz.
-- Performing timer loopback test... pass
-- Performing timer loopback test... pass
-- Setting master clock rate selection to 'automatic'.

-----
Device: B-Series Device
-----
Mboard: B210
revision: 4
product: 2
serial: 30EAE45
name: MyB210
FW Version: 8.0
FPGA Version: 13.0

Time sources: none, internal, external, gpsdo
```

**Ilustración 26. Prueba funcional de la USRP línea de comandos 2/2**

Una vez terminada etapa de compilación del sistema, en el directorio `/sdr/dev/BUILDS/AAAA-MM-DD-hh-mm-ss/` en donde los datos `AAAA-MM-DD-hh-mm-ss` de la última carpeta corresponde a la fecha y hora de compilación del sistema. Al listar los archivos de la carpeta se encuentran archivos compilables para los paquetes complementarios al paquete OpenBTS, tal como se puede identificar en la Ilustración 27.

```
diegofgalvisc@diegofgalvisc:~/sdr/dev/BUILDS/2022-04-11--10-32-46$ ls
liba53_0.1_amd64.changes      range-asterisk-config_5.0_all.deb
liba53_0.1_amd64.deb         range-asterisk-config_5.0_amd64.changes
liba53_0.1.dsc               range-asterisk-config_5.0.dsc
liba53_0.1.tar.gz           range-asterisk-config_5.0.tar.gz
libcoredumper1_1.2.1-1_amd64.deb  range-configs_5.1-master_all.deb
libcoredumper_1.2.1-1_amd64.changes  range-configs_5.1-master_amd64.changes
libcoredumper_1.2.1-1.dsc      range-configs_5.1-master.dsc
libcoredumper_1.2.1-1.tar.gz  range-configs_5.1-master.tar.gz
libcoredumper-dev_1.2.1-1_amd64.deb  sipauthserve_5.0_amd64.changes
openbts_5.0_amd64.changes     sipauthserve_5.0_amd64.deb
openbts_5.0_amd64.deb        sipauthserve_5.0.dsc
openbts_5.0.dsc              sipauthserve_5.0.tar.gz
openbts_5.0.tar.gz           smqueue_5.0_amd64.changes
range-asterisk_11.7.0.5_amd64.changes  smqueue_5.0_amd64.deb
range-asterisk_11.7.0.5_amd64.deb  smqueue_5.0.dsc
range-asterisk_11.7.0.5.dsc      smqueue_5.0.tar.gz
range-asterisk_11.7.0.5.tar.gz
```

**Ilustración 27. Paquetes complementarios OpenBTS**



## 2.1 Actualización e Instalación de Paquetes Adicionales

Se deben realizar la actualización de paquetes adicionales mediante la ejecución de las siguientes sentencias, cuyo resultado de la ejecución se puede validar en las Ilustraciones 28, 29, 30.

```
| sudo apt-get install software-properties-common python-software-properties  
| sudo add-apt-repository ppa:chris-lea/zeromq  
| sudo apt-get update
```

```
diegofgalvisc@diegofgalvisc:~/sdr/dev$ sudo apt-get install software-properties-  
common python-software-properties  
[sudo] password for diegofgalvisc:  
Leyendo lista de paquetes... Hecho  
Creando árbol de dependencias  
Leyendo la información de estado... Hecho  
software-properties-common ya está en su versión más reciente (0.96.20.10).  
python-software-properties ya está en su versión más reciente (0.96.20.10).  
0 actualizados, 0 nuevos se instalarán, 0 para eliminar y 176 no actualizados.
```

**Ilustración 28. Actualización e instalación paquetes adicionales 1/3**

```
diegofgalvisc@diegofgalvisc:~/sdr/dev$ sudo add-apt-repository ppa:chris-lea/zer  
omq  
OMQ is a software library that lets you quickly design and implement a fast mes  
sage-based application.  
Más información: https://launchpad.net/~chris-lea/+archive/ubuntu/zeromq  
Pulse [Intro] para continuar o ctrl-c para cancelar  
  
gpg: anillo «/tmp/tmp0m9ci_ta/secring.gpg» creado  
gpg: anillo «/tmp/tmp0m9ci_ta/pubring.gpg» creado  
gpg: solicitando clave C7917B12 de hkp servidor keyserver.ubuntu.com  
gpg: /tmp/tmp0m9ci_ta/trustdb.gpg: se ha creado base de datos de confianza  
gpg: clave C7917B12: clave pública "Launchpad chrislea" importada  
gpg: Cantidad total procesada: 1  
gpg: importadas: 1 (RSA: 1)  
OK
```

**Ilustración 29. Actualización e instalación paquetes adicionales 2/3**

```
diegofgalvisc@diegofgalvisc:~/sdr/dev$ sudo apt-get update  
Obj:1 http://co.archive.ubuntu.com/ubuntu xenial InRelease  
Des:2 http://co.archive.ubuntu.com/ubuntu xenial-updates InRelease [99,8 kB]  
Ign:3 http://ppa.launchpad.net/chris-lea/zeromq/ubuntu xenial InRelease  
Des:4 http://security.ubuntu.com/ubuntu xenial-security InRelease [99,8 kB]  
Obj:5 http://ppa.launchpad.net/git-core/ppa/ubuntu xenial InRelease  
Des:6 http://co.archive.ubuntu.com/ubuntu xenial-backports InRelease [97,4 kB]  
Des:7 http://co.archive.ubuntu.com/ubuntu xenial-updates/main amd64 DEP-11 Metada  
ta [326 kB]  
Ign:8 http://ppa.launchpad.net/chris-lea/zeromq/ubuntu xenial Release  
Des:9 http://security.ubuntu.com/ubuntu xenial-security/main amd64 DEP-11 Metada  
ta [93,6 kB]  
Ign:10 http://ppa.launchpad.net/chris-lea/zeromq/ubuntu xenial/main amd64 Packag  
es  
Des:11 http://co.archive.ubuntu.com/ubuntu xenial-updates/universe amd64 DEP-11  
Metadata [281 kB]  
Ign:12 http://ppa.launchpad.net/chris-lea/zeromq/ubuntu xenial/main i386 Package  
s  
Des:13 http://security.ubuntu.com/ubuntu xenial-security/universe amd64 DEP-11 M  
etadata [130 kB]  
Des:14 http://co.archive.ubuntu.com/ubuntu xenial-updates/multiverse amd64 DEP-1  
1 Metadata [5.980 B]  
Des:15 http://co.archive.ubuntu.com/ubuntu xenial-backports/main amd64 DEP-11 Me  
tadata [3.328 B]  
Des:16 http://co.archive.ubuntu.com/ubuntu xenial-backports/universe amd64 DEP-1  
1 Metadata [6.612 B]  
Ign:17 http://ppa.launchpad.net/chris-lea/zeromq/ubuntu xenial/main all Packages  
Des:18 http://security.ubuntu.com/ubuntu xenial-security/multiverse amd64 DEP-11  
Metadata [2.464 B]  
Ign:19 http://ppa.launchpad.net/chris-lea/zeromq/ubuntu xenial/main Translation-
```

**Ilustración 30. Actualización e instalación paquetes adicionales 3/3**



### 2.1.1 Instalación y Compilación del Software DEB

En el directorio BUILDS se generó una carpeta con la fecha de creación, en donde se encuentra el paquete DEB en el caso de poseer un sistema operativo es de 32 bits se generará archivos *i386.deb*; o como en este caso si el sistema operativo es de 64 bits se generarán archivos *amd64.deb*.

A continuación se deben correr las siguientes sentencias sobre la carpeta */sdr/dev/BUILDS/AAAA-MM-DD-hh-mm-ss/*, en este caso particular */sdr/dev/BUILDS/2022-04-11-10-32-46/*, las Ilustraciones 31 a 37 corresponden a las salidas de la ejecución de dichas sentencias:

```
| sudo dpkg -i libcoredumper1_1.2.1-1_amd64.deb libcoredumper-dev_1.2.1-1_amd64.deb  
| sudo dpkg -i liba53_0.1_amd64.deb  
| sudo dpkg -i range-configs_5.0_all.deb  
| sudo dpkg -i range-asterisk*.deb  
| sudo dpkg -i sipauthserve_5.0_amd64.deb  
| sudo dpkg -i smqueue_5.0_amd64.deb  
| sudo dpkg -i openbts_5.0_amd64.deb
```

En la ejecución de las anteriores sentencias se debe validar que se ejecuten sin errores, en caso de presentar alguno, validar los nombres de los archivos a compilar los cuales se encuentran sobre la carpeta */sdr/dev/BUILDS/AAAA-MM-DD-hh-mm-ss/*, en el desarrollo de este trabajo de grado se muestra en la Ilustración 27. En caso de que el nombre del archivo de encuentre bien escrito y persista el error, ejecutar la siguiente sentencia:

```
| sudo apt-get install -f
```

```
diegofgalvisc@diegofgalvisc:~/sdr/dev/BUILDS/2022-04-11--10-32-46$ sudo dpkg -i  
libcoredumper1_1.2.1-1_amd64.deb libcoredumper-dev_1.2.1-1_amd64.deb  
(Leyendo la base de datos ... 212546 ficheros o directorios instalados actualmen  
te.)  
Preparando para desempaquetar libcoredumper1_1.2.1-1_amd64.deb ...  
Desempaquetando libcoredumper1 (1.2.1-1) sobre (1.2.1-1) ...  
Preparando para desempaquetar libcoredumper-dev_1.2.1-1_amd64.deb ...  
Desempaquetando libcoredumper-dev (1.2.1-1) sobre (1.2.1-1) ...  
Configurando libcoredumper1 (1.2.1-1) ...  
Configurando libcoredumper-dev (1.2.1-1) ...  
Procesando disparadores para libc-bin (2.23-0ubuntu11.2) ...  
Procesando disparadores para man-db (2.7.5-1) ...
```

**Ilustración 31. Instalación y compilación del software DEB 1/7**



```
diegofgalvis@diegofgalvis:~/sdr/dev/BUILDS/2022-04-11--10-32-46$ sudo dpkg -i  
liba53_0.1_amd64.deb  
(Leyendo la base de datos ... 212546 ficheros o directorios instalados actualmen  
te.)  
Preparando para desempaquetar liba53_0.1_amd64.deb ...  
Desempaquetando liba53 (0.1) sobre (0.1) ...  
Configurando liba53 (0.1) ...  
Procesando disparadores para libc-bin (2.23-0ubuntu11.2) ...
```

**Ilustración 32. Instalación y compilación del software DEB 2/7**

```
diegofgalvis@diegofgalvis:~/sdr/dev/BUILDS/2022-04-11--10-32-46$ sudo dpkg -i  
range-configs_5.1-master_all.deb  
(Leyendo la base de datos ... 212741 ficheros o directorios instalados actualmen  
te.)  
Preparando para desempaquetar range-configs_5.1-master_all.deb ...  
Desempaquetando range-configs (5.1-master) sobre (5.1-master) ...  
Configurando range-configs (5.1-master) ...  
nothing to configure
```

**Ilustración 33. Instalación y compilación del software DEB 3/7**

```
diegofgalvis@diegofgalvis:~/sdr/dev/BUILDS/2022-04-11--10-32-46$ sudo dpkg -i  
range-asterisk*.deb  
Seleccionando el paquete range-asterisk previamente no seleccionado.  
(Leyendo la base de datos ... 212741 ficheros o directorios instalados actualmen  
te.)  
Preparando para desempaquetar range-asterisk_11.7.0.5_amd64.deb ...  
Desempaquetando range-asterisk (11.7.0.5) ...  
Seleccionando el paquete range-asterisk-config previamente no seleccionado.  
Preparando para desempaquetar range-asterisk-config_5.0_all.deb ...  
Desempaquetando range-asterisk-config (5.0) ...  
Reemplazando ficheros del paquete antiguo odbcinet (2.3.1-4.1) ...  
Configurando range-asterisk (11.7.0.5) ...  
Adding system user for Asterisk  
Añadiendo al usuario `asterisk' al grupo `dialout' ...  
Añadiendo al usuario asterisk al grupo dialout  
Hecho.  
Añadiendo al usuario `asterisk' al grupo `audio' ...  
Añadiendo al usuario asterisk al grupo audio  
Hecho.  
Añadiendo al usuario `asterisk' al grupo `www-data' ...  
Añadiendo al usuario asterisk al grupo www-data  
Hecho.  
Configurando range-asterisk-config (5.0) ...  
Instalando una nueva versión del fichero de configuración /etc/odbc.ini ...  
Fichero de configuración `/etc/odbcinst.ini'  
==> Fichero en el sistema creado por usted o por algún script.  
==> Fichero también en el paquete.  
¿Qué quisiera hacer al respecto? Sus opciones son:  
Y o I : instalar la versión del desarrollador del paquete  
N o O : conservar la versión que tiene instalada actualmente
```

**Ilustración 34. Instalación y compilación del software DEB 4/7**



```
diegofgalvisc@diegofgalvisc:~/sdr/dev/BUILDS/2022-04-11--10-32-46$ sudo dpkg -i sipauthserve_5.0_amd64.deb
[sudo] password for diegofgalvisc:
Seleccionando el paquete sipauthserve previamente no seleccionado.
(Leyendo la base de datos ... 213130 ficheros o directorios instalados actualmente.)
Preparando para desempaquetar sipauthserve_5.0_amd64.deb ...
Desempaquetando sipauthserve (5.0) ...
Configurando sipauthserve (5.0) ...
Procesando disparadores para ureadahead (0.100.0-19.1) ...
```

**Ilustración 35. Instalación y compilación del software DEB 5/7**

```
diegofgalvisc@diegofgalvisc:~/sdr/dev/BUILDS/2022-04-11--10-32-46$ sudo dpkg -i smqueue_5.0_amd64.deb
Seleccionando el paquete smqueue previamente no seleccionado.
(Leyendo la base de datos ... 213144 ficheros o directorios instalados actualmente.)
Preparando para desempaquetar smqueue_5.0_amd64.deb ...
Desempaquetando smqueue (5.0) ...
Configurando smqueue (5.0) ...
Procesando disparadores para ureadahead (0.100.0-19.1) ...
```

**Ilustración 36. Instalación y compilación del software DEB 6/7**

```
diegofgalvisc@diegofgalvisc:~/sdr/dev/BUILDS/2022-04-11--10-32-46$ sudo dpkg -i openbts_5.0_amd64.deb
Seleccionando el paquete openbts previamente no seleccionado.
(Leyendo la base de datos ... 213148 ficheros o directorios instalados actualmente.)
Preparando para desempaquetar openbts_5.0_amd64.deb ...
Desempaquetando openbts (5.0) ...
Configurando openbts (5.0) ...
Adding system user for OpenBTS
Añadiendo al usuario `openbts' al grupo `sudo' ...
Añadiendo al usuario openbts al grupo sudo
Hecho.
Añadiendo al usuario `openbts' al grupo `asterisk' ...
Añadiendo al usuario openbts al grupo asterisk
Hecho.
Añadiendo al usuario `openbts' al grupo `www-data' ...
Añadiendo al usuario openbts al grupo www-data
Hecho.
Procesando disparadores para ureadahead (0.100.0-19.1) ...
```

**Ilustración 37. Instalación y compilación del software DEB 7/7**

### 2.1.2 Activación Reenvío de Datos y Configuración de Tablas IP

El tráfico de la red GSM se genera desde la estación de la estación base OpenBTS la cual a su vez está implementada sobre un sistema operativo virtualizado, por lo cual se hace necesario habilitar el reenvío de los paquetes de datos y configurar las reglas del firewall para establecer la comunicación GSM.

El reenvío de datos en el servidor Ubuntu debe habilitarse con el usuario root, para ello ejecutar la sentencia en línea de comandos:

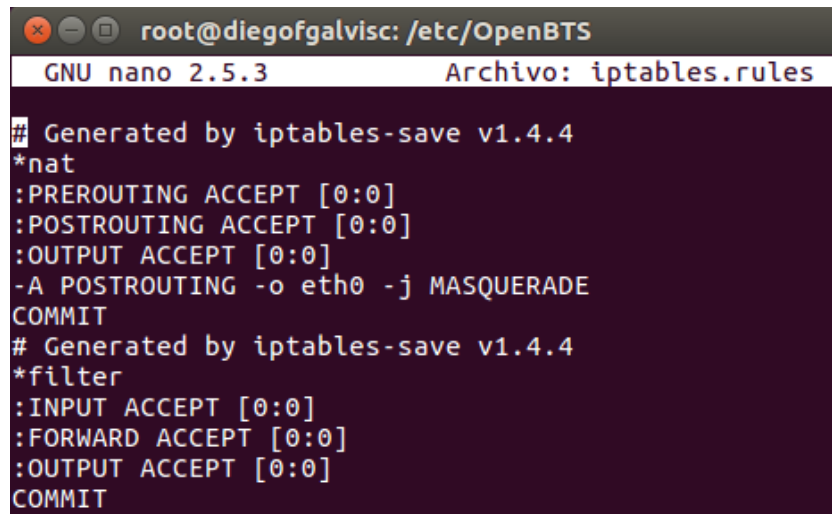


```
| sudo su  
| echo 1 >> /proc/sys/net/ipv4/ip_forward
```

La ejecución de esta sentencia no tiene salida, tal como se puede identificar en la Ilustración 38, pero se pueden validar la reglas de configuración en la ruta `/etc/OpenBTS/iptables.rules`, tal como se ve en la Ilustración 39.

```
diegofgalvisc@diegofgalvisc:~$ sudo su  
[sudo] password for diegofgalvisc:  
root@diegofgalvisc:/home/diegofgalvisc# echo 1 >> /proc/sys/net/ipv4/ip_forward
```

**Ilustración 38. Activación reenvío datos, configuración tablas IP en Ubuntu**



```
root@diegofgalvisc: /etc/OpenBTS  
GNU nano 2.5.3 Archivo: iptables.rules  
# Generated by iptables-save v1.4.4  
*nat  
:PREROUTING ACCEPT [0:0]  
:POSTROUTING ACCEPT [0:0]  
:OUTPUT ACCEPT [0:0]  
-A POSTROUTING -o eth0 -j MASQUERADE  
COMMIT  
# Generated by iptables-save v1.4.4  
*filter  
:INPUT ACCEPT [0:0]  
:FORWARD ACCEPT [0:0]  
:OUTPUT ACCEPT [0:0]  
COMMIT
```

**Ilustración 39. Salida de configuración de iptables**

En el archivo `iptables.rules`, la tarjeta de red está configurada con el nombre de `eth0`, por lo tanto se debe garantizar que la tarjeta de red del servidor Ubuntu posea dicho nombre, tal como se puede identificar en la Ilustración 40, es común que esté con otro nombre, en ese caso se deben unificar.

```
diegofgalvisc@diegofgalvisc:~$ ifconfig  
eth0      Link encap:Ethernet direcciónHW 08:00:27:2f:a1:f9  
          Direc. inet:10.0.2.15 Difus.:10.0.2.255 Másc:255.255.255.0  
          Dirección inet6: fe80::a00:27ff:fe2f:a1f9/64 Alcance:Enlace  
          ACTIVO DIFUSIÓN FUNCIONANDO MULTICAST MTU:1500 Métrica:1  
          Paquetes RX:2956 errores:0 perdidos:0 overruns:0 frame:0  
          Paquetes TX:1951 errores:0 perdidos:0 overruns:0 carrier:0  
          colisiones:0 long.colaTX:1000  
          Bytes RX:2546707 (2.5 MB) TX bytes:479284 (479.2 KB)  
  
lo        Link encap:Bucle local  
          Direc. inet:127.0.0.1 Másc:255.0.0.0  
          Dirección inet6: ::1/128 Alcance:Anfitrión  
          ACTIVO BUCLE FUNCIONANDO MTU:65536 Métrica:1  
          Paquetes RX:16 errores:0 perdidos:0 overruns:0 frame:0  
          Paquetes TX:16 errores:0 perdidos:0 overruns:0 carrier:0  
          colisiones:0 long.colaTX:1000  
          Bytes RX:774 (774.0 B) TX bytes:774 (774.0 B)
```

**Ilustración 40. Descripción tarjeta de red servidor Ubuntu**





Para restablecer las configuraciones de las tablas IP se pueden usar las sentencias:

```
| sudo iptables-restore < /etc/OpenBTS/iptables.rules  
| sudo iptables -t nat -L -n -v
```

Cuya salida se muestra en la Ilustración 41.

```
diegofgalvisc@diegofgalvisc:/$ sudo iptables -t nat -L -n -v  
Chain PREROUTING (policy ACCEPT 0 packets, 0 bytes)  
pkts bytes target      prot opt in     out    source destination  
  
Chain INPUT (policy ACCEPT 0 packets, 0 bytes)  
pkts bytes target      prot opt in     out    source destination  
  
Chain OUTPUT (policy ACCEPT 3 packets, 228 bytes)  
pkts bytes target      prot opt in     out    source destination  
  
Chain POSTROUTING (policy ACCEPT 0 packets, 0 bytes)  
pkts bytes target      prot opt in     out    source destination  
3    228 MASQUERADE all  --  *      eth0   0.0.0.0/0 0.0.0.0/0
```

**Ilustración 41. Salida restablecimiento tablas IP**

### 2.1.3 Carga de Base de Datos de Componentes

Existen tres bases de datos de componentes, ellas son, OpenBTS, responsable de implementar la interfaz aérea GSM software, SubscriberRegistry que se encarga del registro de suscriptores y Smqueue que corresponde a la aplicación de mensajería SIP las cuales se deben cargar inicialmente, por una única vez; para esta tarea se deben ejecutar las siguientes sentencias, ubicados en las carpetas respectivas:

```
| cd sdr/dev/openbts/apps  
| sudo sqlite3 -init OpenBTS.example.sql /etc/OpenBTS/OpenBTS.db ".quit"
```

La salida de la sentencia se puede visualizar en la Ilustración 42.

```
diegofgalvisc@diegofgalvisc:~$ cd sdr/dev/openbts/apps/  
diegofgalvisc@diegofgalvisc:~/sdr/dev/openbts/apps$ sudo sqlite3 -init OpenBTS.e  
xample.sql /etc/OpenBTS/OpenBTS.db ".quit"  
[sudo] password for diegofgalvisc:  
-- Loading resources from OpenBTS.example.sql
```

**Ilustración 42. Carga BD OpenBTS**

```
| cd sdr/dev/subscriberRegistry/apps  
| sudo sqlite3 -init sipauthserve.example.sql /etc/OpenBTS/sipauthserve.db  
".quit"
```



La salida de la sentencia se puede visualizar en la Ilustración 43.

```
diegofgalvisc@diegofgalvisc:~/sdr/dev/subscriberRegistry/apps$ sudo sqlite3 -init sipauthserve.example.sql /etc/sipauthserve/OpenBTS.db ".quit"
-- Loading resources from sipauthserve.example.sql
```

**Ilustración 43. Carga BD subscriberRegistry**

```
| cd sdr/dev/smqueue/smqueue
| sudo sqlite3 -init smqueue.example.sql /etc/OpenBTS/smqueue.db ".quit"
```

La salida de la sentencia se puede visualizar en la Ilustración 44.

```
diegofgalvisc@diegofgalvisc:~/sdr/dev/smqueue/smqueue$ sudo sqlite3 -init smqueue.example.sql /etc/smqueue/OpenBTS.db ".quit"
-- Loading resources from smqueue.example.sql
```

**Ilustración 44. Carga BD smqueue**

## 2.2 Configuración Asterisk

Asterisk es una central telefónica software que permite las funciones de control de llamadas y las tareas de gestión de movilidad, presentando a cada MS conectado en la red GSM como un cliente SIP.

Para realizar la configuración de Asterisk es necesario modificar los archivos de configuración *sip.conf* y *extensions.conf*, los cuales están ubicados en el directorio */etc/asterisk*, tal como lo muestra la Ilustración 45.

```
diegofgalvisc@diegofgalvisc:/etc/asterisk$ ls
asterisk.conf          extensions-range-test.conf  res_odbc.conf
cdr.conf              func_odbc.conf            rtp.conf
cdr_custom.conf       iax.conf                 sip.conf
extconfig.conf        indications.conf         sip-custom-contexts.conf
extensions.conf        logger.conf              sip-custom-register.conf
extensions-custom.conf modules.conf              voicemail.conf
extensions-range.conf musiconhold.conf         voicemail-customer.conf
```

**Ilustración 45. Archivos de configuración Asterisk**

En los archivos de configuración de asterisk se debe registrar el código de identificación de usuario internacional IMSI y el número asignado al teléfono móvil, se utilizará para los dispositivos los códigos mostrados en la Tabla 1.

**Tabla 1. IMSI teléfonos móviles**

Móvil	IMSI														
	MCC			MNC			MSIN								
Alcatel One Touch	7	3	2	1	1	1	1	4	0	8	0	3	5	7	0



<b>Samsung Galaxy J6+</b>	7	3	2	1	2	3	6	0	3	6	2	6	9	6	5
<b>Samsung Galaxy Mote 9 Plus</b>	7	3	2	1	0	1	1	7	6	9	6	1	5	2	1
<b>Sony Ericsson Xperia Neo V</b>	7	3	2	1	0	1	6	3	8	0	6	6	9	8	9
<b>Huawei Mate 8</b>	7	3	2	1	0	1	5	7	8	6	2	8	9	4	2
<b>Huawei P8 Lite</b>	7	3	2	1	0	1	6	3	8	0	3	9	9	9	2
<b>Xiaomi Poco X3 NFC</b>	7	3	2	1	2	3	6	1	0	5	5	8	3	0	0
<b>Samsung Krono K32</b>	7	3	2	1	0	1	6	3	8	0	6	6	9	8	7

Haciendo uso de un editor de texto se modifican los archivos de configuración indicados, para ello se requiere permisos de administrador.

Al archivo de configuración original *sip.conf* adicionar al final las líneas de comando indicadas en la Ilustración 46, en donde el código IMSI que se encuentra en la SIM de los dispositivos móviles GSM para obtener este código se puede hacer uso alguna de las alternativas:

- Mediante el uso de un lector USB de tarjetas de memoria con capacidad de chip SIM, el cual se puede encontrar en el comercio.
- La segunda opción es que una vez OpenBTS se encuentre funcionando aunque asterisk no esté configurado, guardará un archivo temporal *TMSIS* el cual contiene el código IMSI de cualquier terminal GSM que busque y detecte de forma manual la celda activa de OpenBTS en su lista de operadores disponibles y solicite conectarse a la red.

```
diegofgalvisc@diegofgalvisc: /etc/asterisk
GNU nano 2.5.3 Archivo: sip.conf

insecure=port,invite

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

openbts
[IMSI73211140803570] ;MCC 732=Colombia MNC 111
callerid=1111 ;Número asignado al dispositivo móvil
canreinvite=no ;No reenvío de instrucciones después de conectado
type=friend
context=sip-external ;Llamadas anonimas #s externos pueden llamar
allow=gsm ;Permite gsm
host=dynamic
tmfmode=info

[IMSI732123603626965]
callerid=2222
canreinvite=no
type=friend
context=sip-external
allow=gsm
host=dynamic
tmfmode=info
```

**Ilustración 46. Modificación archivo configuración asterisk sip.conf**



Al archivo de configuración original `extensions.conf` adicionar al final las líneas de comando indicadas en la Ilustración 47, las cuales corresponden a la información del número asignado al móvil y su respectiva IMSI. Es decir que en este archivo de configuración se definen los usuarios que pueden acceder a la central telefónica y a su vez a la estación de comunicación móvil basada en SDR. Entre otras características se están definiendo los códecs que va a utilizar OpenBTS y el tipo de usuario que le corresponderá el teléfono móvil.

Una vez configurados los usuarios SIP, en el archivo `sip.conf`, se debe configurar el plan de marcación. El plan de llamadas está organizado en contextos. Cada usuario dispone de un contexto, y es dicho contexto el que determina qué hacer cuando el usuario realiza una llamada.

```
diegofgalvis@diegofgalvis: /etc/asterisk
GNU nano 2.5.3 Archivo: extensions.conf

([from-pstn](+))
exten => _011!, 1,Set(CDR(B-Number)="+${EXTEN:3})
exten => _00!, 1,Set(CDR(B-Number)="+${EXTEN:2})
exten => _1NXXNXXXXXX, 1,Set(CDR(B-Number)="+${EXTEN})
exten => _NXXNXXXXXX, 1,Set(CDR(B-Number)="+1${EXTEN})
;
(
)
([default](+))
exten => 111, 1,Goto(VoicemailMain,${CDR(A-Number)},1)

##include extensions-range-test.conf

([sip-external])
exten =>100,1,Dial(SIP/IMSI732111140803870)
exten =>101,1,Dial(SIP/IMSI732123603626965)
;
exten =>200,1,Answer()
same => n,Playback(hola-mundo)
same => n,Hangup()
(
)
exten =>201,1,Goto(TestMenu,start,1)

exten =>1111,1,Dial(SIP/IMSI732111140803570@127.0.0.1:5062)
exten =>2222,1,Dial(SIP/IMSI732123603626965@127.0.0.1:5062)
```

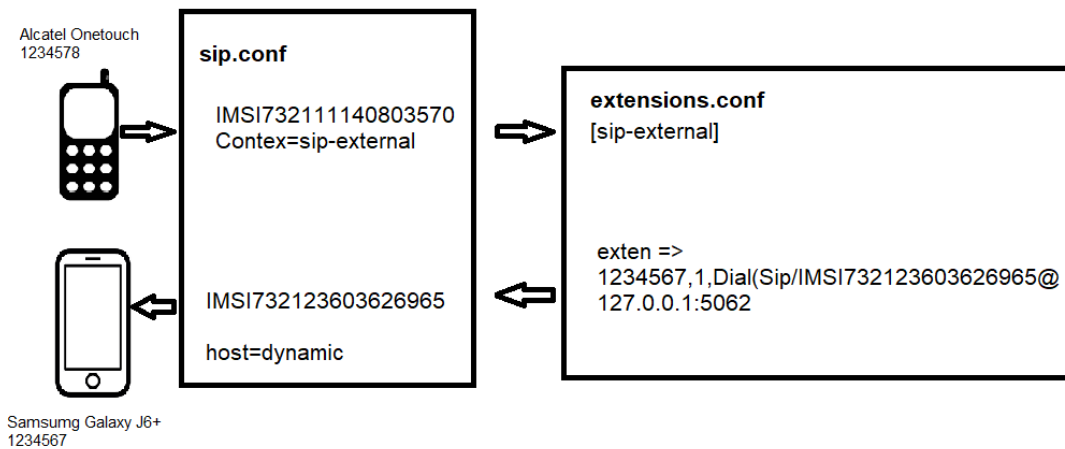
**Ilustración 47. Modificación archivo configuración asterisk `extensions.conf`**

En caso de ser requerida la modificación de los archivos de configuración `sip.conf` y/o `extensión.conf` se recomienda recargar ambos archivos en la consola del asterisk ejecutando las sentencias:



- | Sip reload
- | Dialplan reload

La configuración de sip.conf y extensions.conf es muy importante para establecer la comunicación en asterisk, se identificará la relación que mantienen entre ambos archivos en la Ilustración 48 . Para ello se toma como ejemplo la llamada realizada desde el MS Alcatel Onetouch con Número 1234578 al MS Samsung Galaxy J6+ con número 1234567.



**Ilustración 48. Relación sip.conf - extensions.conf**

En la Tabla 2 se presenta el resumen de datos de las MS usadas para la ejecución de pruebas , después de realizar las respectivas configuraciones.

**Tabla 2. Resumen MS prueba**

Marca/Modelo	IMEI	IMSI	Número Asignado	Operador Simcard
Alcatel One Touch	014304007458500	732111140803570	1234578	Tigo
Samsung Galaxy J6 +	351756100380790	732123603626965	1234567	Movistar
Samsung Galaxy Note 9 Plus	359426091026750	732101176961521	1234579	Claro
Sony Ericsson Xperia Neo V	358518040786130	732101638066989	1234560	Claro
Huawei Mate 8	868662021491770	732101578628942	1234589	Claro
Huawei P8 Lite	860715036177400	732101638039992	1234509	Claro
Xiaomi Poco X3 NFC	869819058525940	732123610558300	1234568	Movistar
Samsung Krono K32	358812081876090	732101638066987	1234590	Claro



Estación de comunicación móvil GSM basado en SDR  
Diego Fernando Galvis Calambás – José Luis Sandoval Jácome  
Anexo 1