



Laboratorios SDR con GNU Radio

Creación Colectiva

Bryan Estiven Garcia Zamora, Juan David Cristancho Gaona, Carlos Fernando Bustos Quevedo, Carlos Mario Herrera Fernandez, Esther Alexandra Ramos Arias, Héctor Javier Vega Lozano, Ferney Genaro Vasquez Sanabria, Luis Mateo Cuervo Romero, Julian Fernando Barreto Coronado, Sergio Andres Dimas Gomez, Leonardo Nicolas Solorzano Cruz, Alex Isaac Urrea Mahecha, Anderson Trullo Arias, Pedro Yovan Munca

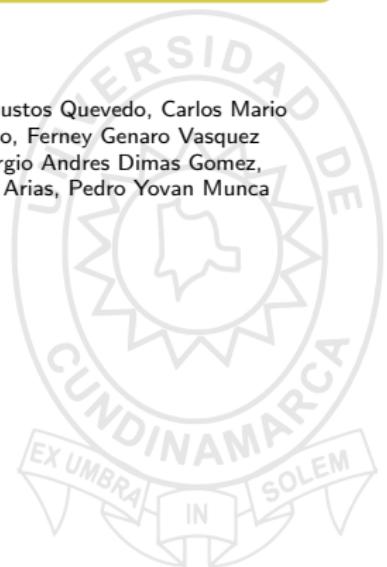
Cadena

Tutor

Rodríguez Mújica Leonardo

Facultad de Ingeniería
Universidad de Cundinamarca

Diciembre 2021





Contenidos generales

LABORATORIOS CON SOFTWARE

3





LABORATORIOS CON SOFTWARE



Parte I - Tabla de contenidos

Introducción a GNU Radio

5

Lab1: Primeros pasos

17

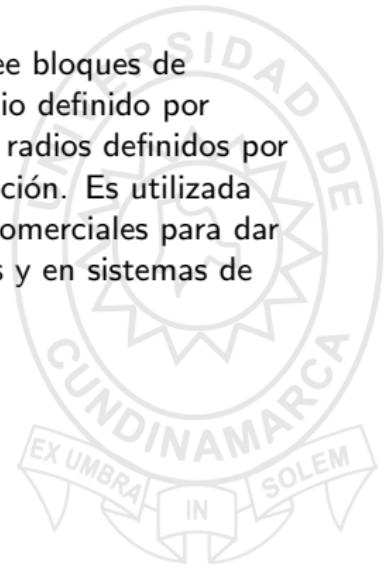


Introducción a GNU Radio



¿Qué es GNU Radio?

Es una herramienta de desarrollo libre y abierta que provee bloques de procesamiento de señal para implementar sistemas de radio definido por software. Puede utilizarse con hardware de RF para crear radios definidos por software o sin hardware para crear un ambiente de simulación. Es utilizada extensivamente en ambientes académicos, aficionados y comerciales para dar soporte a la investigación en comunicaciones inalámbricas y en sistemas de radio en el mundo.





Aplicaciones

RASTREO SATELITAL



REDES DIGITALES

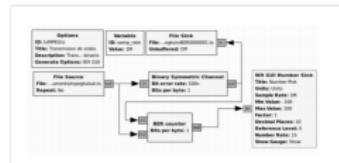


SISTEMAS DE RADAR

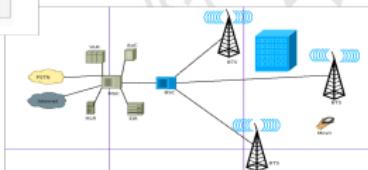


AUDIO PROCESAMIENTO

comunicaciones móviles



REDES GSM





Instalación de GNU Radio en Linux

Para instalar GNU Radio se deben seguir los siguientes pasos:

1. Ingresar a la ventana de órdenes (o terminal) del sistema de su equipo.
2. Estando conectado a internet, escriba dentro del terminal:

```
sudo apt-get install gnuradio
```

3. Si su dispositivo tiene contraseña, debe ingresarla, al ser solicitada y oprimir .
4. Luego se deben aceptar los términos de la instalación oprimiendo la letra  seguido de .
5. Una forma de verificar la correcta instalación es volviendo a ingresar la orden indicada en el punto 2, y si aparece un mensaje anunciando que GNU Radio ya está en su versión más reciente, su instalación fue correcta.



Paquetes

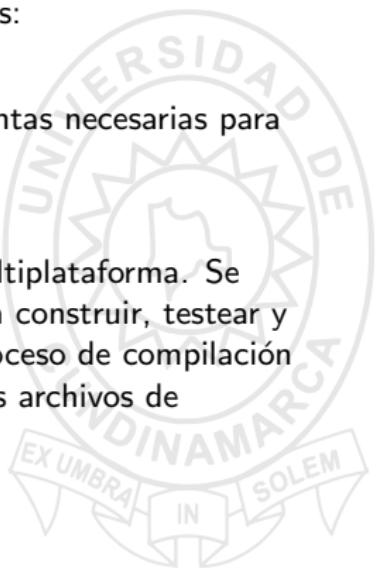
Con el objetivo de clonar el repositorio y obtener los ejemplos de GNU Radio en nuestro ordenador se deben instalar los siguientes paquetes:

- build-essential

Build essential es un paquete que contiene herramientas necesarias para la creación, compilación e instalación de programas.

- cmake

Es un sistema de construcción de código abierto multiplataforma. Se trata de un conjunto de herramientas diseñadas para construir, testear y empaquetar software. Se utiliza para controlar el proceso de compilación de software utilizando una plataforma sencilla y unos archivos de configuración independientes del compilador.





Paquetes

- git

Este paquete contiene un sistema de control de versiones distribuidas de código abierto desarrollado originalmente por Linux Torvalds para apoyar el desarrollo del kernel de Linux.

El control de versiones es un sistema que registra los cambios realizados sobre un archivo o conjunto de archivos a lo largo del tiempo, de modo que se puedan recuperar versiones específicas más adelante.

- libboost-all-dev

Boost es un conjunto de bibliotecas para el lenguaje de programación C++ que suministra un apoyo para tareas y estructuras como álgebra lineal, generación de números pseudoaleatorios, procesamiento de imágenes, expresiones regulares y pruebas unitarias. En el momento contiene 162 bibliotecas individuales.



Paquetes

- `libcppunit-dev`
 - Biblioteca de pruebas unitarias para C++.
 - Una prueba unitaria es una forma de comprobar el correcto funcionamiento de una unidad de código. Por ejemplo, en diseño estructurado o en diseño funcional, una función o un procedimiento, en diseño orientado a objetos una clase. Esto sirve para asegurar que cada unidad funcione correcta y eficientemente por separado.
- `doxygen` Es una herramienta para generar documentación a partir de código fuente. Es un sistema de documentación para C++, C, Java, Python. Es necesario solo si se desea generar referencias a documentación externa de la que no tiene las fuentes.



Instalación de paquetes

1. La instalación de cada uno de los paquetes anteriormente mencionados se realiza colocando en la ventana de terminal, las siguientes órdenes:

```
sudo apt-get install build-essential
```

```
sudo apt-get install cmake
```

```
sudo apt-get install git
```

```
sudo apt-get install libboost-all-dev
```

```
sudo apt-get install libcppunit-dev
```

```
sudo apt-get install doxygen
```





Clonar repositorio

El código fuente de los ejemplos está almacenado en github, por lo tanto para clonar el repositorio se debe realizar lo siguiente:

- Abrir la ventana de órdenes o terminal.
- Después se debe ingresar la siguiente orden para clonar el directorio git:

```
git clone https://github.com/gnuradio/gr-tutorial
```

- Una vez clonado el directorio, gr-tutorial, en el PC empleado se deben ver exactamente los mismos archivos y carpetas que los del repositorio github.



Instalación de módulos

- Luego de haber clonado el repositorio, debemos buscar la carpeta “**gr-tutorial**” e ingresar a ella desde el terminal, para ello se digitán las siguientes órdenes:

```
ls
```

```
cd gr-tutorial
```

Es importante mencionar que al escribir la primera orden se podrán observar la diferentes carpetas que se encuentran en el dispositivo, por lo tanto “**gr-tutorial**” debe aparecer entre las opciones para poder cambiar de directorio.



Instalación de módulos

- Estando dentro de la carpeta, desde la terminal, se deben escribir las siguientes órdenes, con la finalidad de instalar las soluciones o módulos:

```
mkdir build
```

```
cd build
```

```
cmake ..
```

```
make -j8
```

```
sudo make install
```

```
sudo ldconfig
```





Resultado

En el directorio examples encontrará los tutoriales:

tutorial1

tutorial2

...

tutorial7

Puede explorar los ejemplos de los desarrolladores de GNU Radio





Lab1

Primeros pasos

Primeros pasos

The screenshot shows the GNU Radio Companion (GRC) application window. The main area displays a flow graph for an FM stereo receiver. A 'Signal Source' block (Sample Rate: 32k, Waveform: Cosine, Frequency: 3k, Amplitude: 1, Offset: 0) is connected to a 'Throttle' block (Sample Rate: 32k). The 'Throttle' block is then connected to a 'QT GUI Time Sink' block (Number of Points: 1,024k, Sample Rate: 32k, Autoscale: Yes). On the left side, there are three variable blocks: 'Options' (ID: 'audio_src', Generate Options: QT Gui), 'Variable' (ID: 'samp_rate', Value: 32k), and 'Variable' (ID: 'my_var', Value: 0). A tooltip box says: "Arrastre los bloques de la lista hacia el lienzo". Another tooltip box says: "Lista de bloques: presione CTRL + F para buscar por nombre." A third tooltip box says: "Lienzo (área de construcción del diagrama de flujo)". A large callout arrow points from the right margin towards the central workspace.

Qt GUI Range
ID: fmrc
Label: freq
Default Values: 3k
Start: 0
Stop: 1k
Step: 1k

Variable
ID: samp_rate
Value: 32k

Variable
ID: my_var
Value: 0

Signal Source
Sample Rate: 32k
Waveform: Cosine
Frequency: 3k
Amplitude: 1
Offset: 0

Throttle
Sample Rate: 32k

QT GUI Time Sink
Number of Points: 1,024k
Sample Rate: 32k
Autoscale: Yes

Ventana de registros: terminal del programa

```
WARNING: No se ha encontrado una tarjeta de sonido compatible con la interfaz de audio.
Using HackRF One with firmware 2015.07.2
UUUlen(interp_taps)=14
```

```
(top_block.py:4028): Gtk-WARNING **: 10:30:22.222: El efecto 'fader' no es compatible con la tarjeta de sonido actual. Se ha establecido el efecto 'volume' en su lugar.
(top_block.py:4028): Gtk-WARNING **: 10:30:22.222: El efecto 'fader' no es compatible con la tarjeta de sonido actual. Se ha establecido el efecto 'volume' en su lugar.
>>> Done
```

Id	Value
Imports	
Variables	
freq	<Open Properties>
my_var	0

File Edit View Run Tools Help

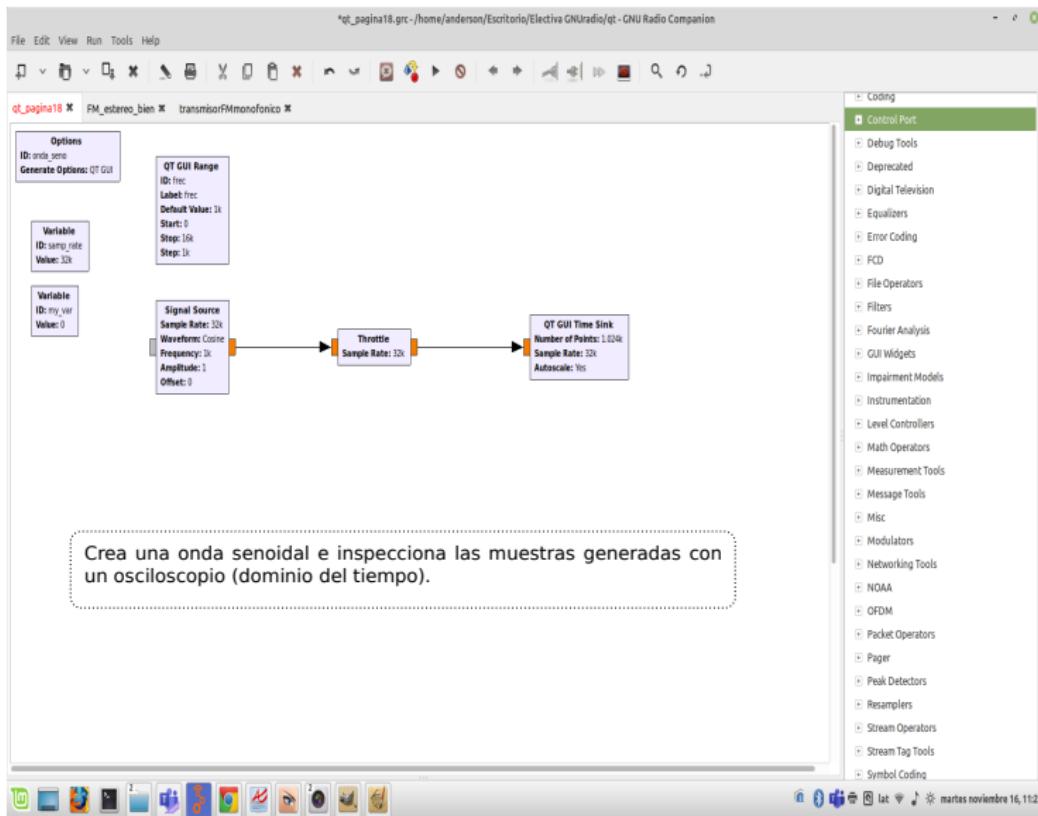
qt_pagina18.grc - /home/anderson/Escritorio/Electiva GNUradio/qt - GNU Radio Companion

Control Port Debug Tools Deprecated Digital Television Equalizers Error Coding FCD File Operators Filters Fourier Analysis GUI Widgets Impairment Models Instrumentation Level Controllers Math Operators Measurement Tools Message Tools Mic Modulators Networking Tools NOAA OFDM Packet Operators Pager Peak Detectors Resamplers Stream Operators Stream Tag Tools Symbol Coding

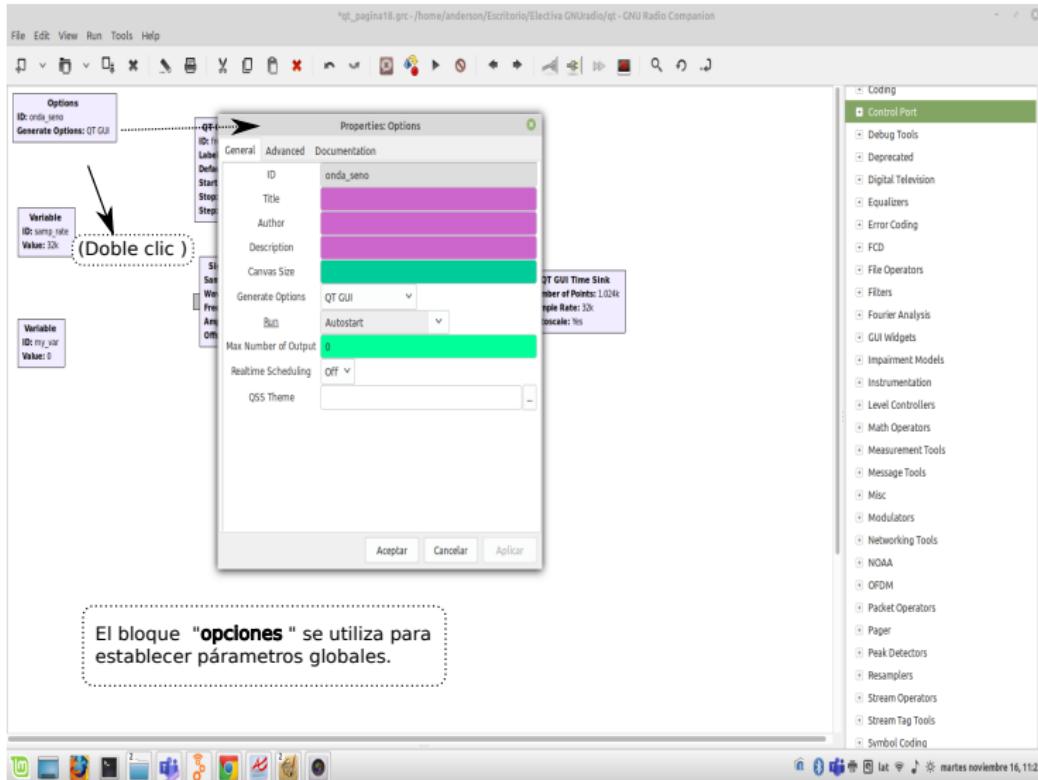
martes noviembre 16, 11:14



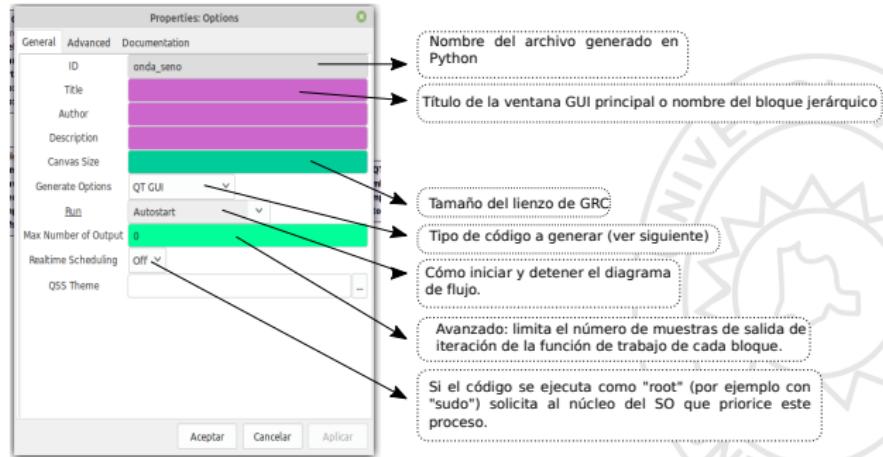
Primeros pasos



Primeros pasos



Primeros pasos





Primeros pasos

The screenshot shows the 'Properties' dialog box with the 'General' tab selected. The fields and their current values are:

Field	Value
ID	top_block
Title	(Redacted)
Author	(Redacted)
Description	(Redacted)
Window Size	1280, 1024
Generate Options	WX GUI
Run	Run
Max Number of Output	Hier Block
Realtime Scheduling	Off

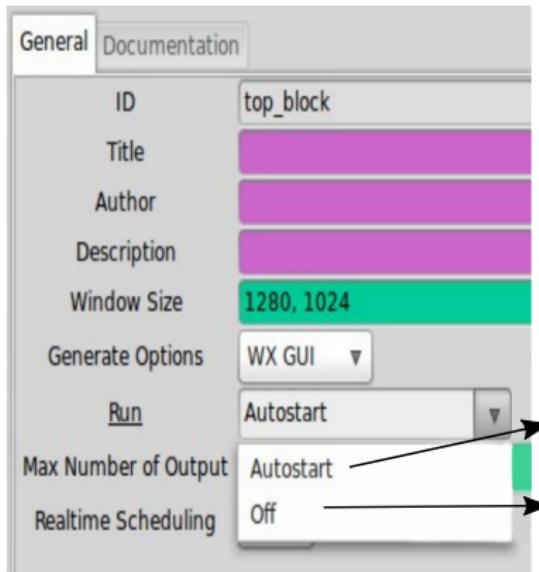
Annotations explain the settings:

- ID:** Aplicación de la GUI que utiliza el kit de herramienta WX (utiliza bloques GUI WX).
- Window Size:** Aplicación GUI que utiliza el kit de herramientas QT (utiliza bloques GUI Qt).
- Run:** Aplicación de la línea de órdenes sin GUI (basada en texto, ejecutada en una consola).
- Max Number of Output:** Cree un bloque jerárquico que aparezca en la lista de bloques (un componente reutilizable, no una aplicación: use los bloques Pad Source/Sink para exponer puertos y los bloques paramétricos para exponer variables de configuración).

Buttons at the bottom: Cancelar, Aceptar.



Primeros pasos





Primeros pasos

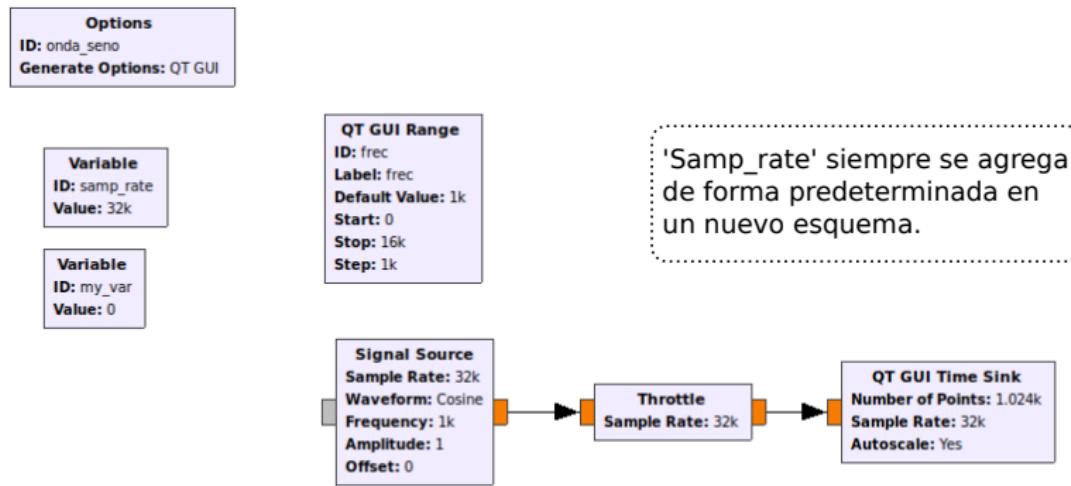
General Documentation

ID	top_block
Title	
Author	
Description	
Window Size	1280, 1024
Generate Options	No GUI ▾
Run Options	Run to Completion
Max Number of Output	Prompt for Exit
Realtime Scheduling	Off ▾

Saldrá automáticamente cuando termine

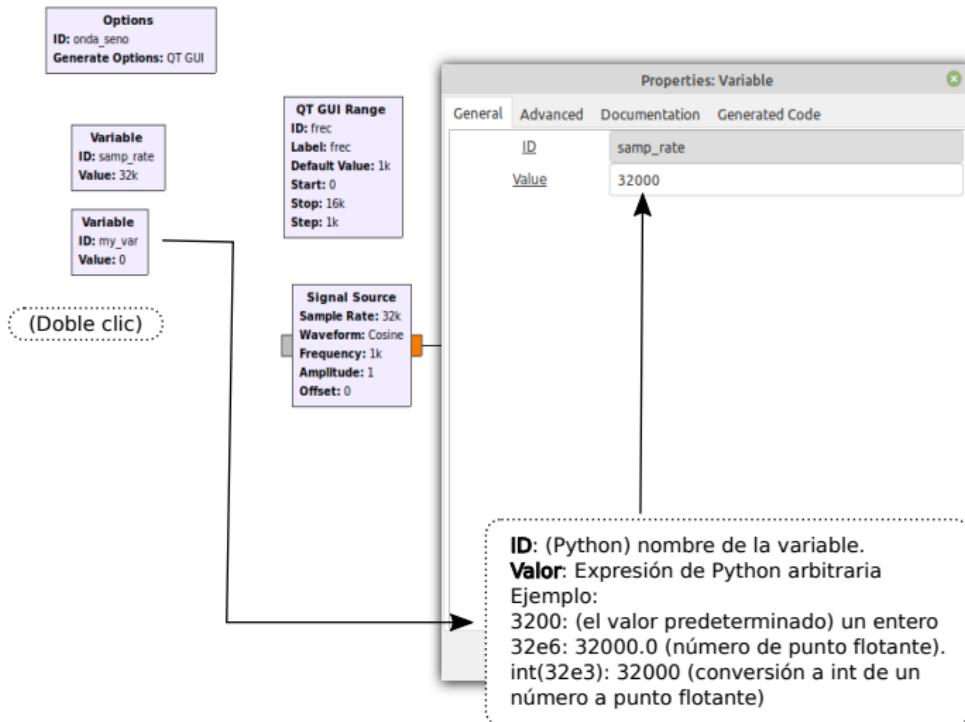
Presionando ENTER saldrá

Primeros pasos

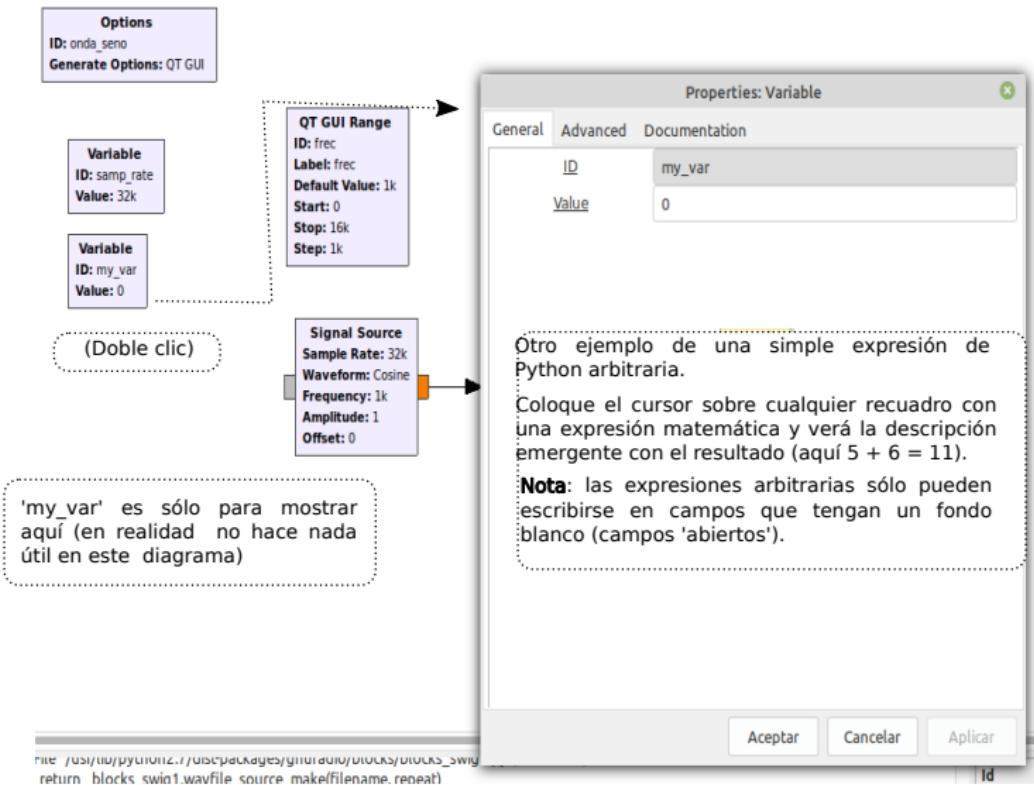


Variable: un bloque que contiene una expresión de Python arbitraria.

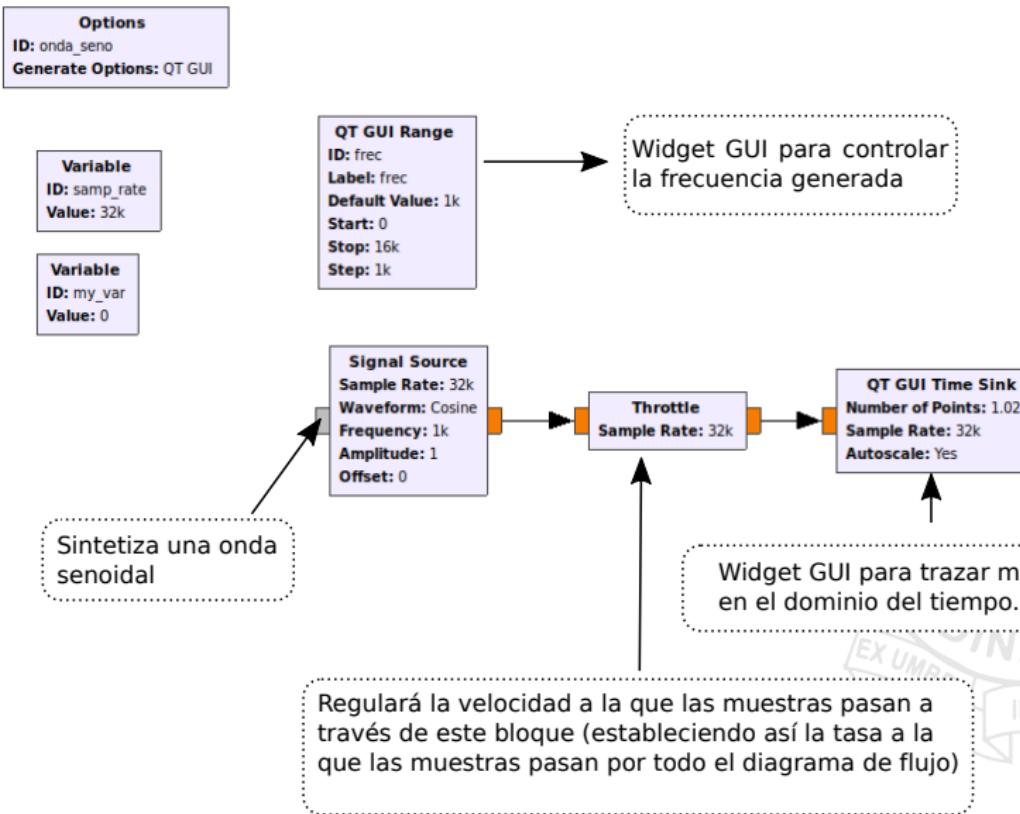
Primeros pasos



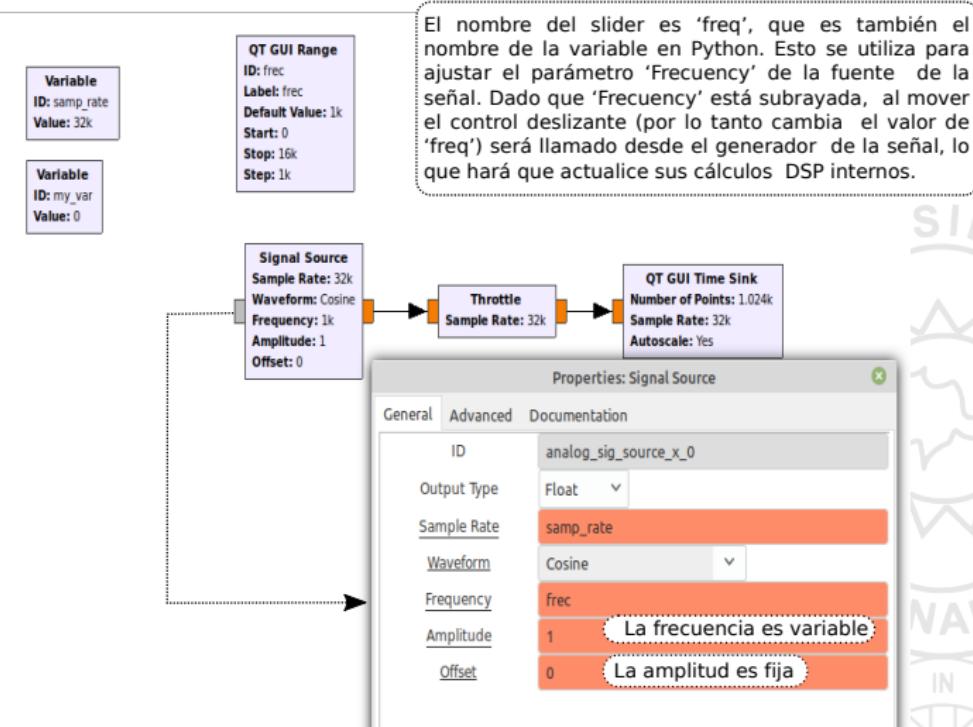
Primeros pasos



Primeros pasos



Primeros pasos



Primeros pasos

The screenshot shows a software interface for configuring variables and components. On the left, two variable definitions are listed:

- Variable ID: samp_rate Value: 32k
- Variable ID: my_var Value: 0

In the center, a **QT GUI Range** component is defined with the following parameters:

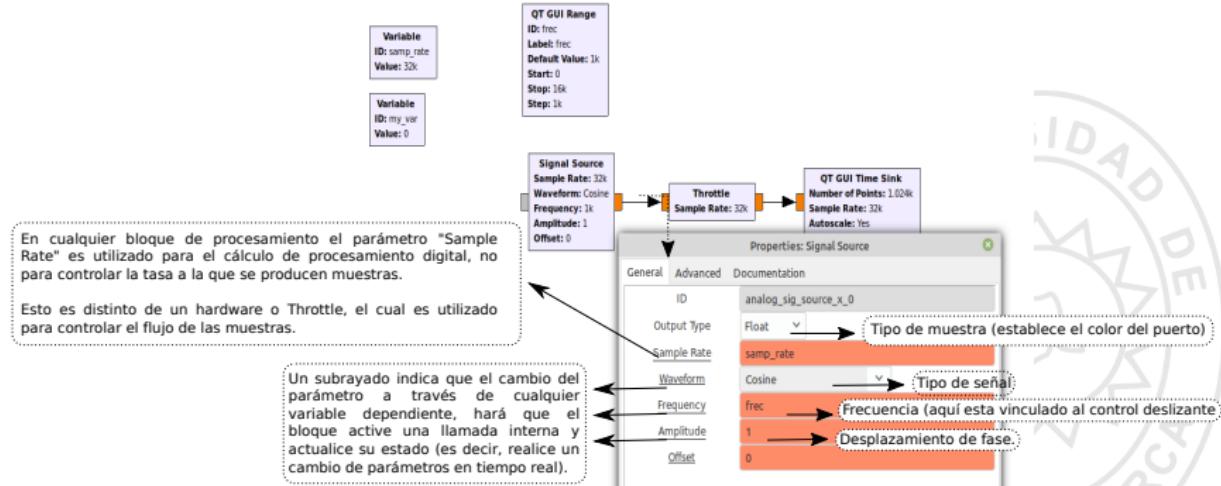
- ID: freq
- Label: freq
- Default Value: 1k
- Start: 0
- Stop: 16k
- Step: 1k

A callout box points to the **Default Value** field in the properties dialog, which contains the value **1e3**. A note next to it states: "Etiqueta junto al widget en la GUI: 1000.0 en notación científica".

The properties dialog for the **QT GUI Range** component has the following settings:

General	Advanced	Documentation
ID: freq		
Label: freq		Etiqueta junto al widget en la GUI: 1000.0 en notación científica
Type: Float		1000.0 en notación científica
Default Value: 1e3		Frecuencia mínima del slider
Start: 0		Frecuencia máxima del slider
Stop: 16e3		
Step: 1000		
Widget: Counter + Slider		El parámetro 'freq' debería ser un número: de punto flotante o un número entero.
Minimum Length: 200		
GUI Hint:		

Primeros pasos



Primeros pasos

El color de un puerto indica el tipo de muestras que fluyen a través de él. Los colores también se aplican a los parámetro de los bloques.

Valores de punto flotante de precisión simple

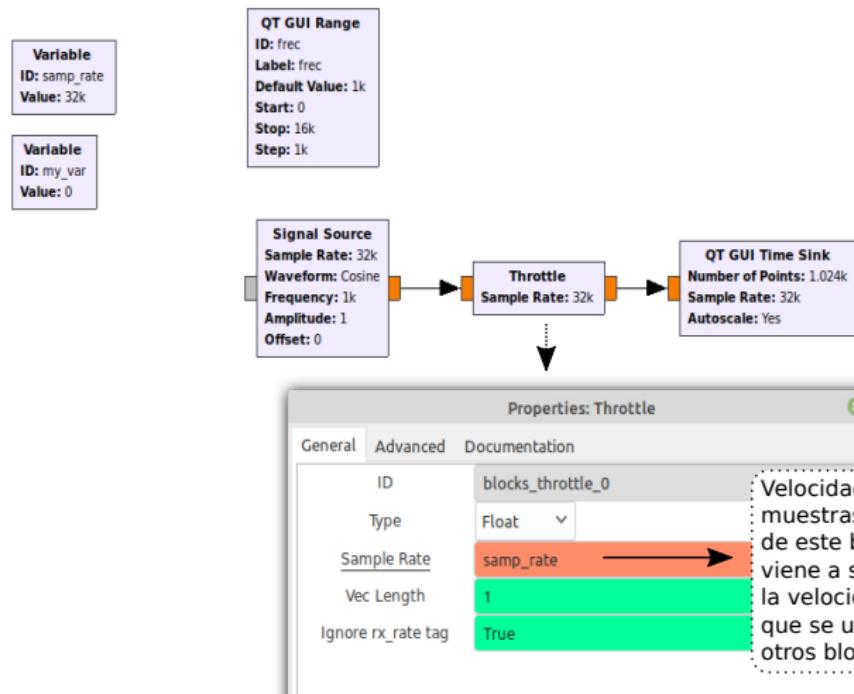
Consejo:
Después de hacer clic en el bloque, presione las teclas de flecha arriba / abajo para cambiar el tipo de muestras.
(En las opciones de bloque se puede configurar manualmente, en el primer parámetro)

Color Mapping

Types
Complex Float 64
Complex Float 32
Complex Integer 64
Complex Integer 32
Complex Integer 16
Complex Integer 8
Float 64
Float 32
Integer 64
Integer 32
Integer 16
Integer 8
Message Queue
Async Message
Bus Connection
Wildcard



Primeros pasos



Primeros pasos

Properties: QT GUI Time Sink

General Trigger Config Advanced Documentation

ID: qtgui_time_sink_x_0

Type: Float

Name: **

Y Axis Label: Amplitude

Y Axis Unit: 1

Number of Points: 1024

Sample Rate: samp_rate

Grid: No

Autoscale: Yes

Y min: -1

Y max: 1

Number of Inputs: 1

Update Period: 0.10

Disp. Tags: Yes

GUI Hint:

Aceptar Cancelar Aplicar

(*) Trazar dos flujos de tipo flotante en sincronización cambiando el tipo de ámbito a Complejo y utilizar el bloque Flotante en complejo antes.

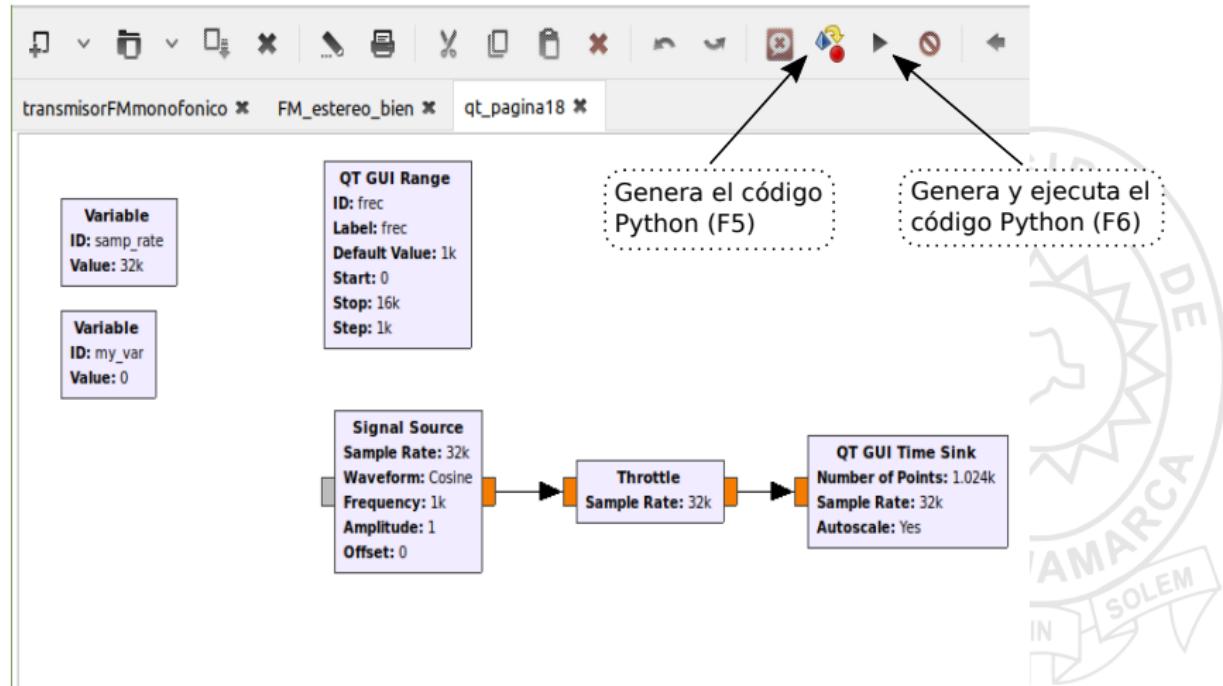
Este es solamente para generar los pasos correctos en el eje X!

QT GUI Time Sink
Number of Points: 1.024k
Sample Rate: 32k
Autoscale: Yes

permite que el gráfico se auto-escala a la señal entrante. Al ingresar cualquier otro valor, se mostrará en una escala/ desplazamiento fijo en esa dimensión.



Primeros pasos

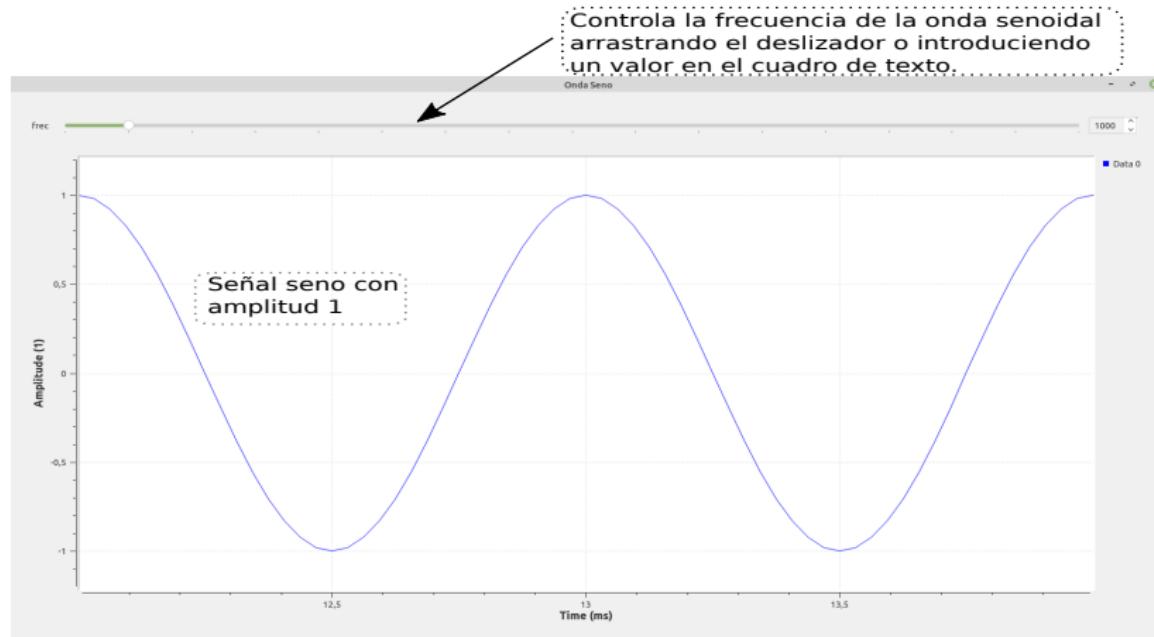


Primeros pasos

Programa en Python generado por GRC

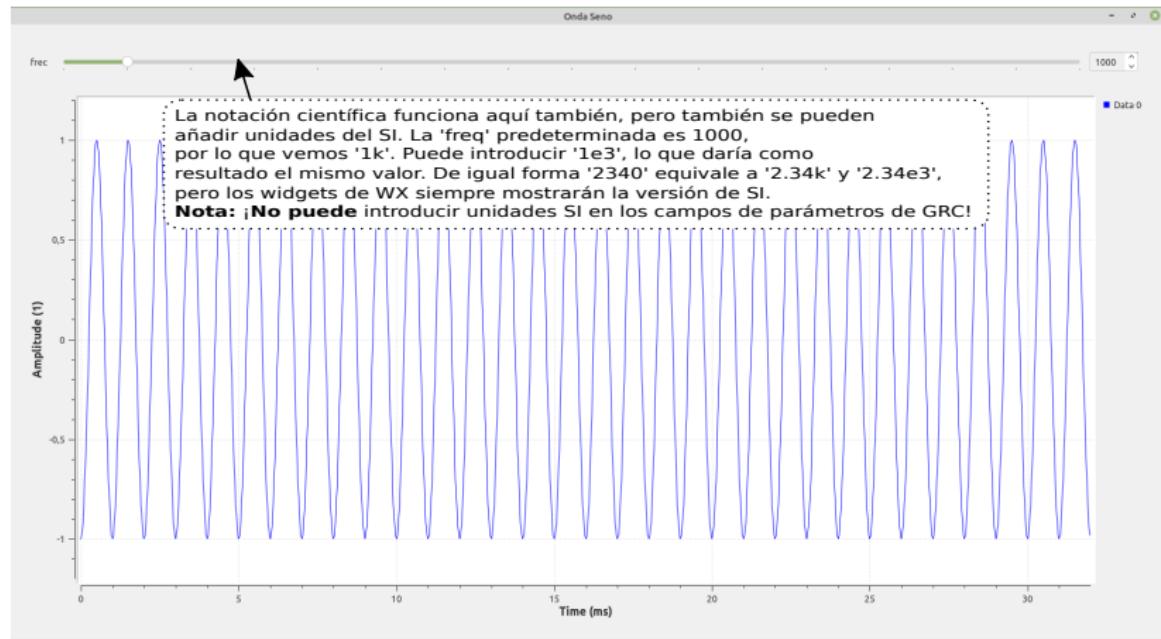


Primeros pasos





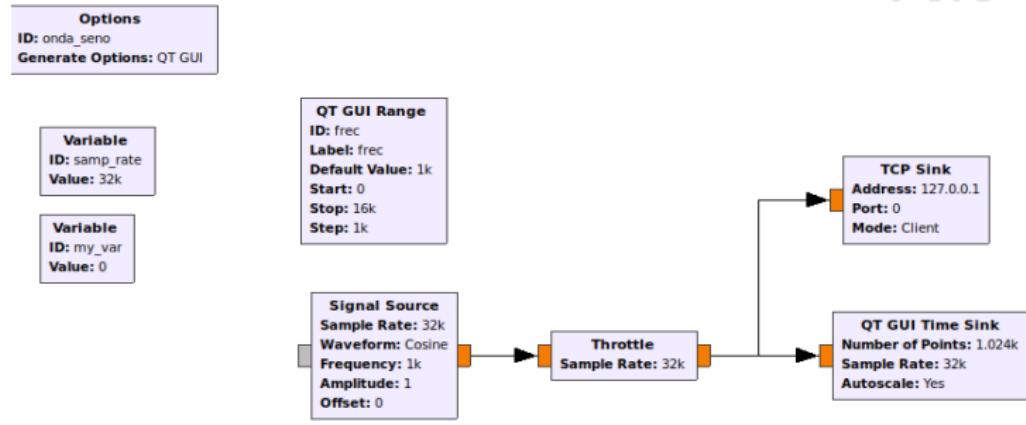
Primeros pasos



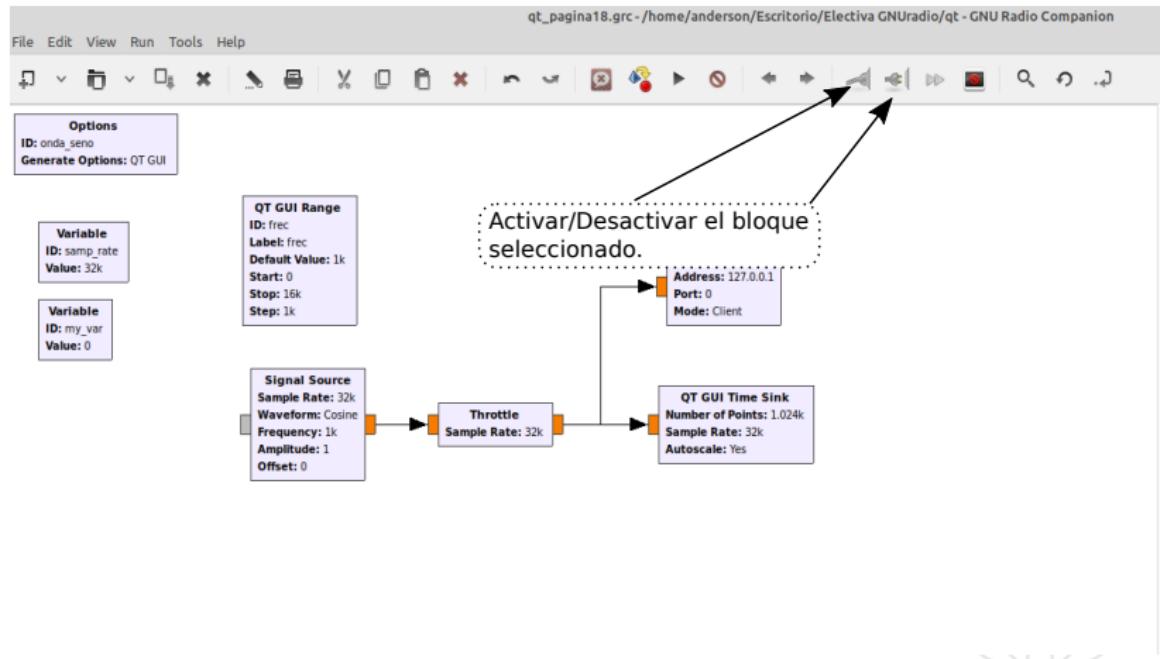


Primeros pasos

Crear una onda senoidal y transmitir las muestras generadas a través de una conexión TCP.

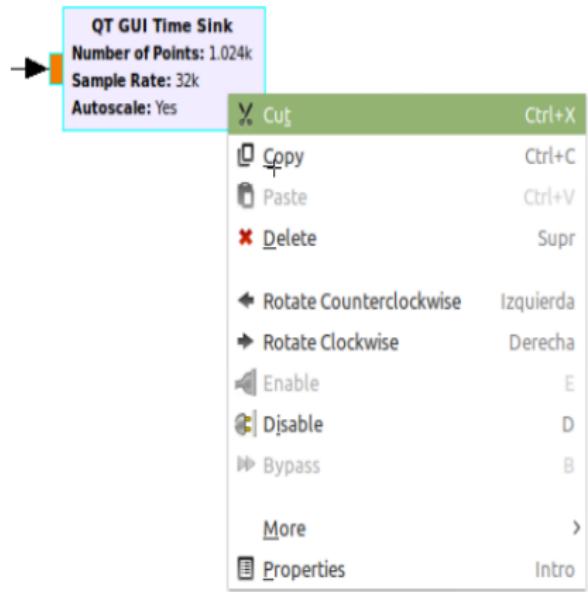


Primeros pasos





Primeros pasos



Consejo:

Tome nota de los atajos de teclado en el menú contextual del bloque.



Primeros pasos

The screenshot shows the "Properties: TCP Sink" dialog box. It has tabs for General, Advanced, and Documentation, with General selected. The fields are as follows:

ID	blks2_tcp_sink_0
Input Type	Float
Address	127.0.0.1
Port	0
Mode	Client
Vec Length	1

Annotations explain the settings:

- TCP Sink**:
Address: 127.0.0.1
Port: 0
Mode: Client
→ Se debe introducir la dirección Ip del servidor.
- QT GUI Time Sink**:
Number of Points: 1.024k
Sample Rate: 32k
Autoscale: Yes

Consejo:
El diagrama de flujo no arrancará a menos que se establezca una conexión TCP. Si falla la conexión TCP, se produce una excepción de Python y el programa no arranca.

* La implementación de cliente / servidor TCP actual no funciona en Windows.



Primeros pasos

Options
ID: tcp_server
Generate Options: QT GUI

Variable
ID: samp_rate
Value: 32k

TCP Source
Address: 0.0.0.0
Port: 12.345k
Mode: Server

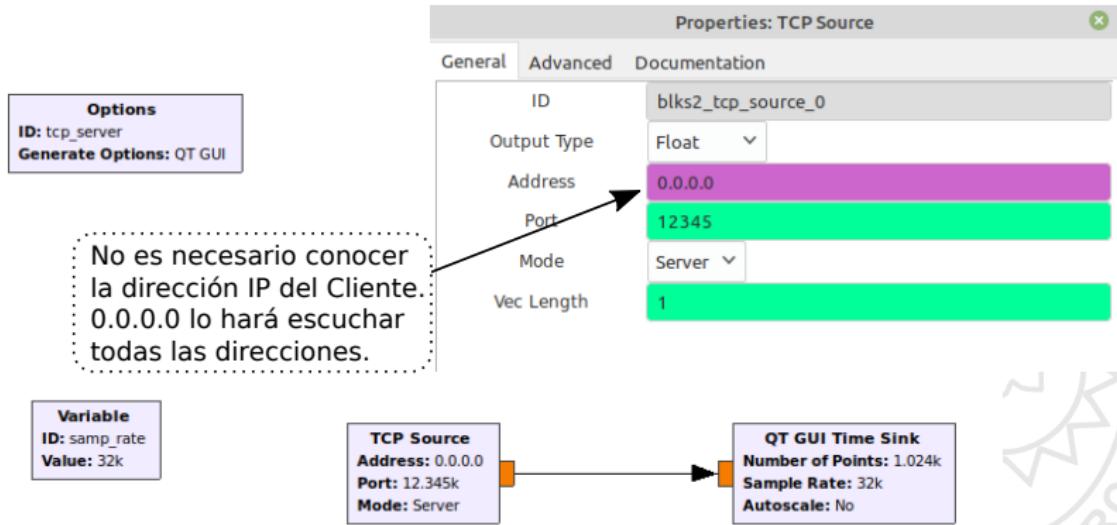
QT GUI Time Sink
Number of Points: 1.024k
Sample Rate: 32k
Autoscale: No

Recibe muestras de una conexión TCP entrante y las dibuja en un osciloscopio.

* Se debe crear un nuevo archivo



Primeros pasos

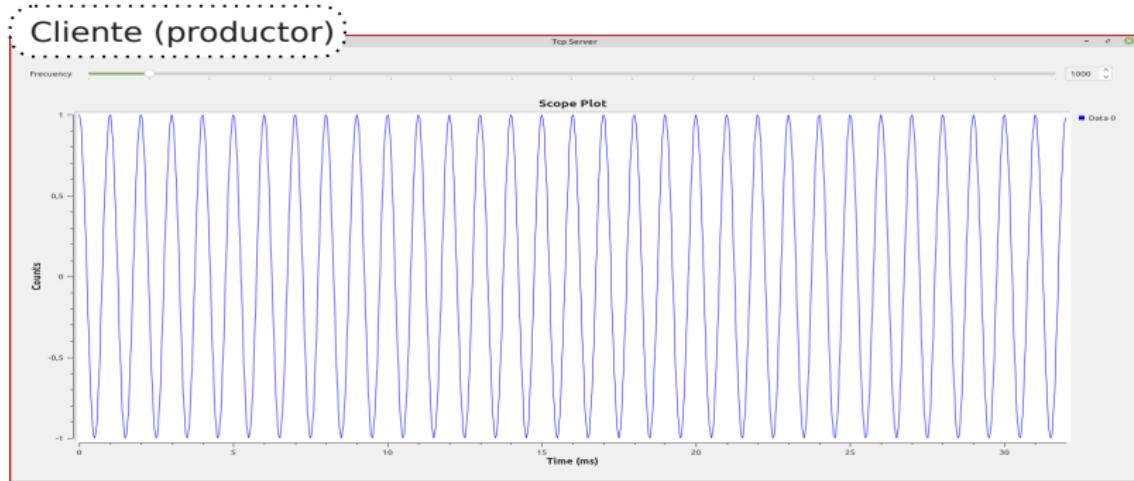


Consejo:

El diagrama de flujo no se iniciará hasta que se acepte una conexión TCP. En este caso, la GUI no aparecerá hasta que el cliente se haya conectado.



Primeros pasos



Servidor (consumidor)

Puede ejecutar cada aplicación por separado en dos máquinas conectadas en red. Simplemente cambie la dirección IP de destino del cliente a la máquina en la que se está ejecutando el servidor. Recuerde ejecutar primero el archivo del servidor y luego el del cliente.



Actividades



Transmisión de señales por multiplexación

En esta actividad usted debe transmitir 3 señales periódicas por medio del TCP (Protocolo de Control de Transmisión) desde el cliente, mediante el proceso de multiplexación de señales, al servidor, que se encargará de demultiplexar la señal recibida y mostrar las 3 que fueron transmitidas en el Scope Sink.

Es importante saber que:

1. La multiplexación es el proceso mediante el cual diferentes mensajes de información (Señales) se combinan en una única señal con el fin de trasmisitirla.
2. La demultiplexación es el proceso mediante el cual la señal multiplexada recibida se divide en cada una de las señales que la generaron.



Pistas para la actividad

Las pistas son:

- ① Modo cliente: Bloques y conexiones de los primeros pasos. (WX GUI Slider, Signal Source, TCP Sink, Scope Sink, Throttle, Stream Mux, Variable)
- ② Modo servidor: Bloques y conexiones de los primeros pasos. (TCP Source, Scope Sink, Stream to Streams)
- ③ Multiplexación: Bloque Stream Mux. En el parámetro Lengths se debe agregar el número de ítems de cada señal, en forma de lista. Para la actividad las señales cuentan con un solo ítem. Como ejemplo de lo anterior tenemos que:
 - 2 señales = 1,1
 - 3 señales = 1,1,1
 - 4 señales = 1,1,1,1
- ④ Demultiplexación: Bloque Stream to Streams.
- ⑤ El tipo de dato en todos los bloques debe ser el mismo. (float)
- ⑥ Habilitar las entradas o salidas suficientes para los bloques.



Modificación de las variables de una señal periódica

Aprender a variar los componentes básicos de una señal periódica (amplitud, frecuencia, fase y nivel DC) en GNU Radio.

Es importante saber que:

1. La forma más simple de representar una señal periódica es una senoidal como se presenta matemáticamente a continuación

$$x(t) = A \sin(2\pi f t + \phi) + K$$

En donde:

2. Amplitud (A): Es el valor máximo que toma la señal, es decir, la distancia entre el punto máximo de la señal y cero, este punto puede ser tanto positivo como negativo.
3. Frecuencia (f): Es el número de ciclos que realiza la señal por unidad de tiempo, esta medida está dada en Hertz (Hz), que equivale a un ciclo por segundo, lo que significa que 80 Hz son 80 ciclos por cada segundo que transcurre.



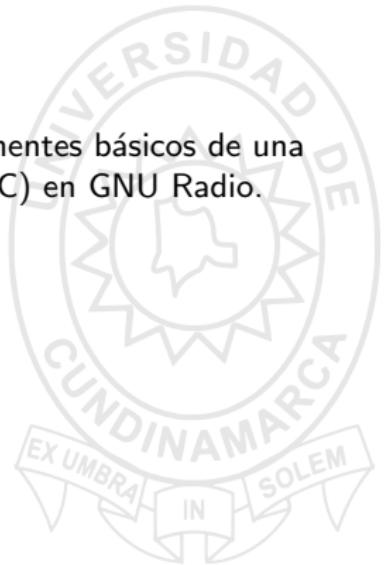
Modificación de las variables de una señal periódica

1. La fase (ϕ): Indica la magnitud de una de variación cíclica, siendo la fracción del período que transcurre desde el instante tomado al estado hasta la referencia, es decir el desplazamiento que tiene una señal en grados con respecto a su referencia.
2. Nivel DC (K): Es el valor medio de la señal, lo que quiere decir que es un voltaje en DC que se le suma a la señal AC, para obtener un desplazamiento en la amplitud de la señal, puede ser tanto positivo como negativo.



Actividad

1. Realizar un programa que permita variar los componentes básicos de una señal periódica (amplitud, frecuencia, fase y nivel DC) en GNU Radio.





Pistas para la actividad

Las pistas son:

1. Repasar los bloques utilizados en la primera guía "Primeros pasos"
2. Debe existir un WX GUI Slider para cada variable que se desee modificar en el osciloscopio.
3. Para una mejor apreciación de los cambios en las variables de la señal se sugiere añadir otra señal que sirva como referencia
4. Coherencia entre los tipos de dato entre bloques
5. Para variar el nivel DC de la señal, el osciloscopio debe tener la opción "Coupling" en DC
6. Para poder configurar correctamente la fase se debe dejar una frecuencia fija, se agrega un bloque "Delay" y en este bloque se representa la formula de la fase de la siguiente forma:
$$(\text{Fase} * \text{Samp_rate})/360/\text{Frecuencia}$$