

UNIVERSIDAD INDUSTRIAL DE SANTANDER
PROGRAMA DE ELECTRONICA
LABORATORIO 1 – PROGRAMACIÓN DE SISTEMAS DE TIEMPO REAL

1. IDENTIFICACIÓN

FACULTAD E3T	PROGRAMA: INGENIERIA ELECTRONICA
ASIGNATURA	COMUNICACIONES II
UNIDAD TEMÁTICA	GNU RADIO
TEMA	PROGRAMACIÓN DE SISTEMAS DE TIEMPO REAL
DOCENTE	JOHAN LEANDRO TÉLLEZ GARZÓN
ALUMNOS	CRISTIAN MANUEL SIERRA JEREZ – 2192308 JHONATAN OSWALDO FRANÇO GUERRON – 2194673 JEISSON ESTEBAN BRAVO LÓPEZ – 2195585
FECHA	8/09/2023

2. DESCRIPCIÓN Y OBJETIVOS

Mediante esta guía de enseñanza se hace un acercamiento inicial a la programación de bloques en GNU radio mediante Python y se analizan los procesos que deben ser realizados para tratar un flujo de información en formato stream.

- Practicar la programación de sistemas de tiempo real y los promedios de tiempo

3. REVISIÓN PRELIMINAR

Conozca el [Manual de Manuales](#) para que aprenda a encontrar las ayudas disponibles en temas de SDR en la UIS. Familiarícese con las variables usadas en los flujogramas. [El libro de la asignatura](#) contiene además cosas como:

- En el capítulo 6, del [libro Vol I](#), se tiene una descripción de las variables y siglas que se usan en los flujogramas para cualquier práctica de la asignatura.
- Enlaces a flujogramas usados en el libro. Debajo de cada gráfica con flujogramas hay una nota que dice: "Flujograma usado". Esos flujogramas usados en el libro están en la página del libro: <https://sites.google.com/saber.uis.edu.co/comdig/sw> o directamente en github: https://github.com/hortegab/comdig_su_software_libro3.8.git

1. Realización de tutoriales prácticos de la página oficial de GNU radio haciendo énfasis en la programación del bloque denominado Python block y en el tratamiento de estructuras de datos tipo stream:

<https://wiki.gnuradio.org/index.php?title=Tutorials>

Defina que es una variable tipo stream, cuáles son sus ventajas y desventajas e identifique las características esenciales del bloque Python block. Este punto es opcional.

2. Investigue e identifique las operaciones que deben ser realizadas para encontrar la media cuadrática, el valor RMS, la potencia promedio y la desviación estándar de una señal. Use el libro de la E3T.

4. ACTIVIDADES Y PROCEDIMIENTOS

3. Genere tres señales diferentes (seno, ruido, vector repetitivo) y visualícelas en el dominio del tiempo. Utilice los parámetros que crea convenientes.
4. Cree un bloque con programación Python, usando un Python Block que permita entregar al mismo tiempo varios promedios de tiempo como: la media, la media cuadrática, el valor RMS, la potencia promedio, la desviación estándar. Los siguientes son requerimientos:
El bloque tiene una entrada y 5 salidas, una para cada parámetro: la media, la media cuadrática, el valor RMS, la potencia promedio, la desviación estándar
Todo el código se implementa dentro de un solo Python Block
Ten en cuenta que si hay 5 salidas, la variable `output_items`, puede verse como:

```
y0=output_items[0]
y1=output_items[1]
y2=output_items[2]
y3=output_items[3]
y4=output_items[4]
```
5. Demuestra el funcionamiento del bloque, mediante un flujograma que calcule los diferentes parámetros para las tres señales implementadas en el punto 2. Analizar los parámetros de la señal y visualizarlos en tiempo real.
6. Los valores obtenidos son los esperados?.
7. Conectar el USRP, la antena y el cable Ethernet al computador con la finalidad de adquirir una señal en la banda de operación de la antena, seleccione un canal o frecuencia de su elección y realice mediciones de la potencia normalizada, valor RMS y desviación estándar con el bloque implementado en Python considerando tres valores distintos de frecuencia de muestreo. Analizar los resultados obtenidos.
8. Elija un valor fijo de frecuencia de muestreo que permita escuchar la señal de referencia en su totalidad y cambie los valores del amplificador de RX (LNA) del USRP en cuatro valores distintos, documente los resultados obtenidos en términos de potencia normalizada, valor RMS y desviación estándar. Analizar los resultados obtenidos.

5. RECURSOS E INFORMACIÓN COMPLEMENTARIA

Los recursos e informaciones relacionadas al desarrollo de este laboratorio son los siguientes:

- Computador con mínimo 4 GB de RAM, 2GB de espacio en disco y procesador Core i3 o superior.
- Documentación oficial del GNU RADIO.
- Libro E3T

6. EVIDENCIA, RESULTADOS Y CONCLUSIONES DEL LABORATORIO

Para poder realizar la practica de la manera correcta fue necesario basarnos y ayudarnos del libro guía y tutorial brindado por el profesor en la guía. Con esto claro se procedió a crear un bloque llamado “Promedios_de_tiempos” el cual fue creado con programación directamente en Python y constaba de una entrada y 5 salidas, este bloque lo que realiza es el cálculo de características propias del sistema tales como la media, valor RMS, la media cuadrática, potencia y desviación estándar. Posterior a esto se colocan los bloques necesarios para la visualización de la señal en el dominio del tiempo y la visualización de las diferentes variables calculadas por el bloque “Promedios_de_tiempos”. (Figura 1)

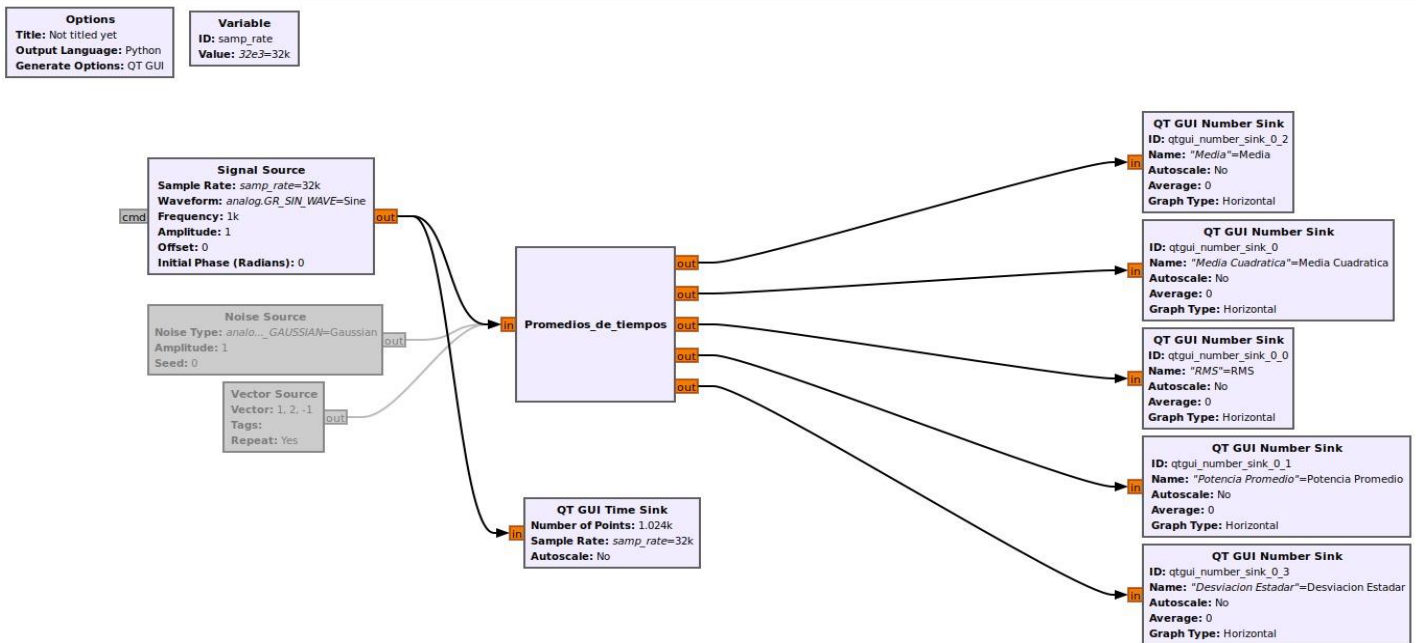


figura 1. Arreglo de bloques

Para realizar el siguiente paso se crearon tres situaciones donde a cada una se calculaban los mismos parámetros, pero se usaban señales diferentes. En primer caso se realizó el análisis con una señal seno y se realizó la obtención de los valores mediante el GNU donde se podía ver cada valor debajo de la señal en el dominio del tiempo. (Figura 2)

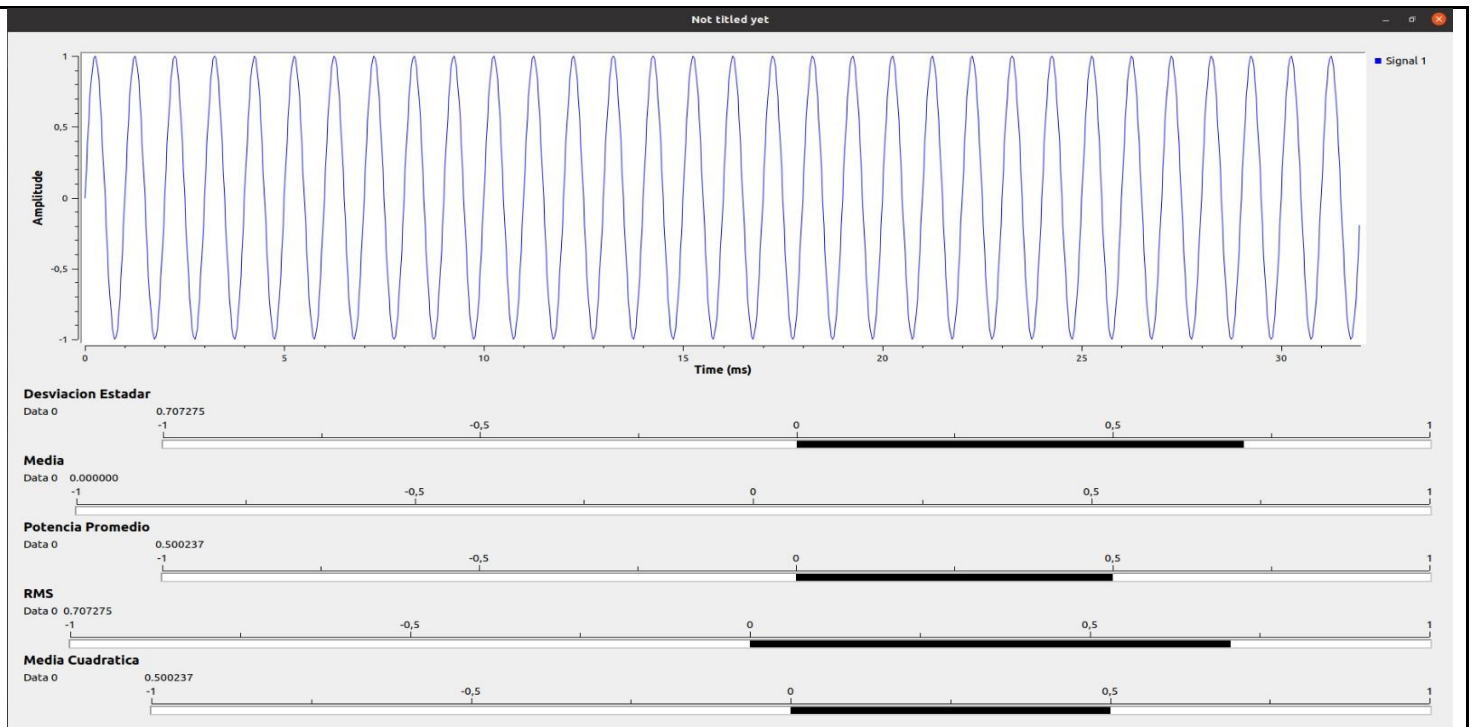


figura 2. Señal seno

Seguidamente se cambio la señal por una señal de ruido, donde se pudo visualizar dicho ruido en dominio tiempo y también cada valor bajo la grafica de la señal, este proceso se realiza cambiando únicamente en el bloque de "Signal Source" el tipo de señal. (Figura 3)

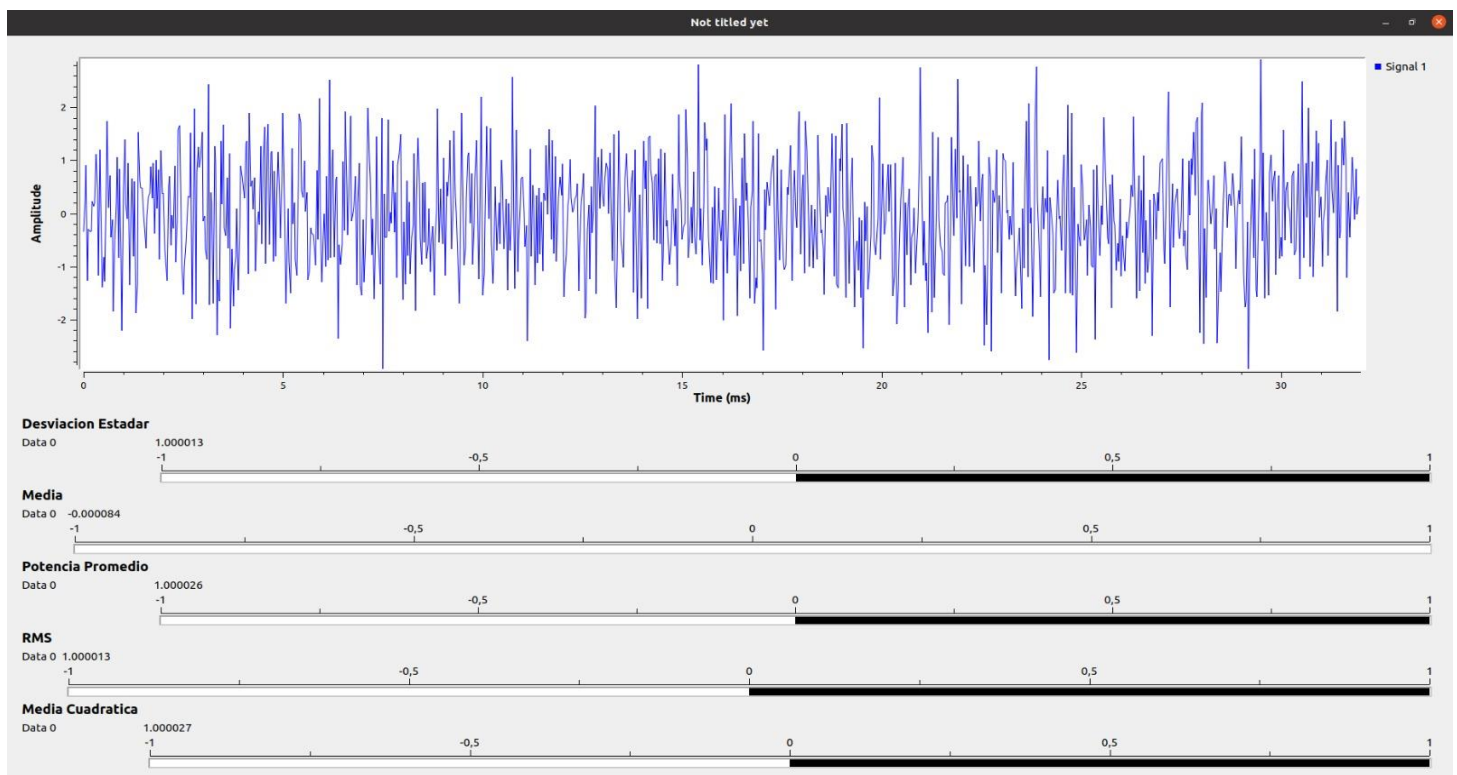


figura 3. Señal ruido

Por ultimo se creo una señal vector donde se pudo visualizar también cada parámetro con diferencias entre ellos dado que cada señal tiene amplitudes y características diferentes entre ellas. Esta señal se pudo visualizar también en dominio tiempo y se obtuvieron los cálculos como en las anteriores señales. (Figura 4)

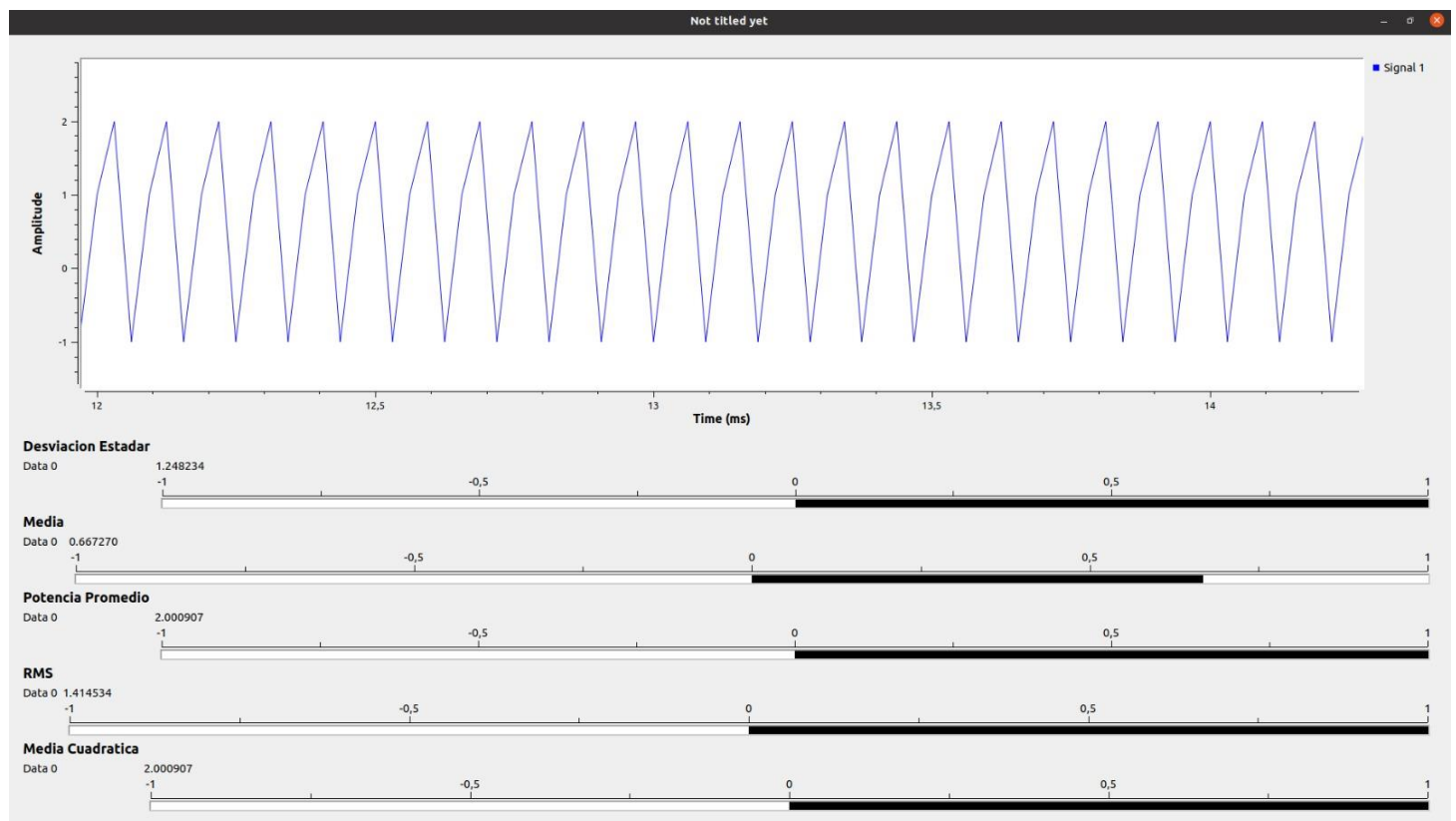


figura 4. Señal vector

Como resultado de cada caso se obtuvieron entonces:

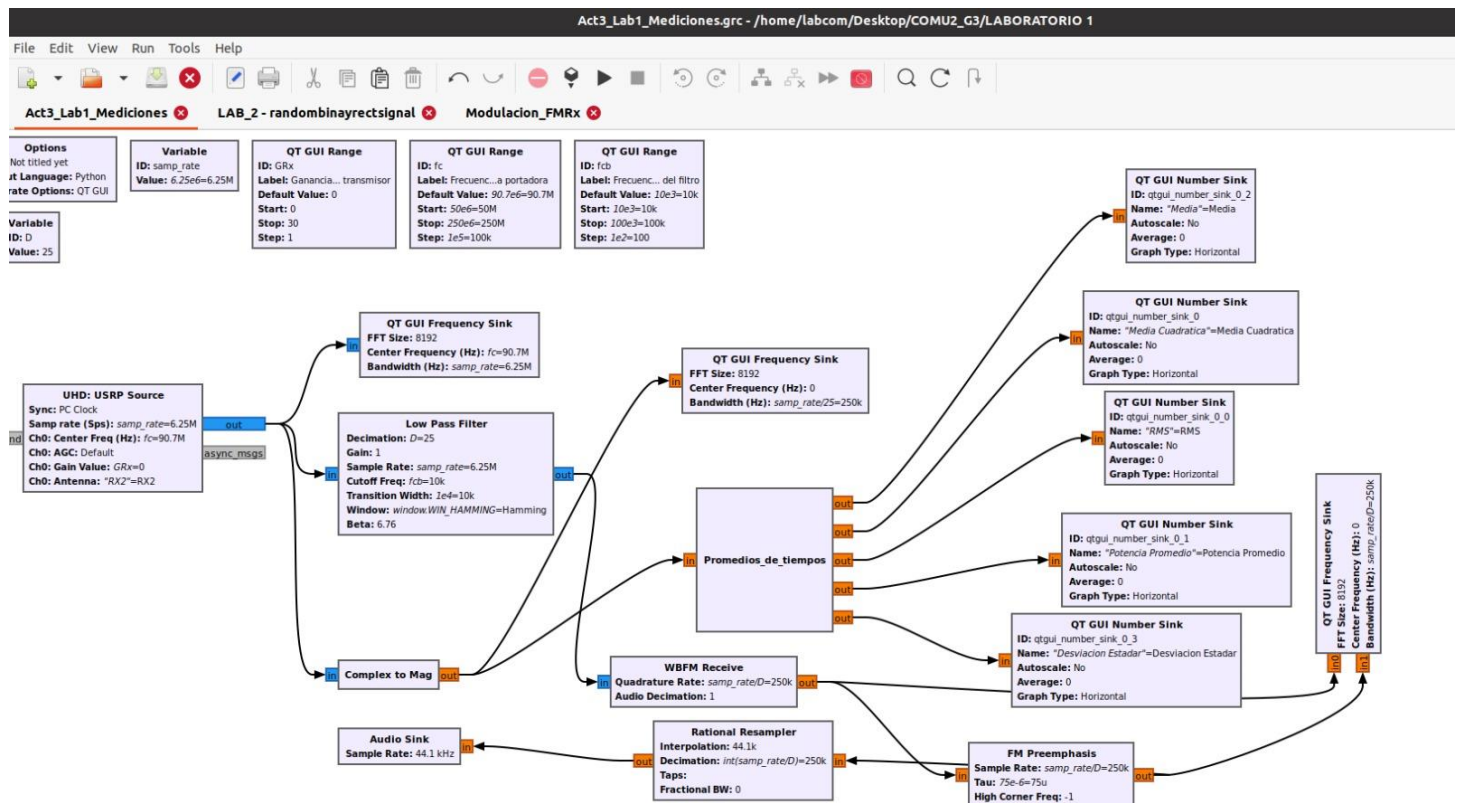
Señal	Desviación estándar	Media	Potencia promedio	RMS	Media Cuadrática
Seno	0.7073	0	0.5	0.71	0.5
Ruido	1.000013	-0.000084	1.000026	1.000013	1.000027
Vector	1.248234	0.66727	2.000907	1.414534	2.000907

Tabla 1. Valores para cada señal

Estos valores son valores esperados dado que por ejemplo tiene sentido que una señal senoidal tenga media cero dado que su amplitud es igual en valores positivos como en valores negativos lo que hace que se tenga una media de cero. Calcular cada parámetro matemáticamente nos dio una idea de que el bloque realiza el cálculo de manera efectiva.

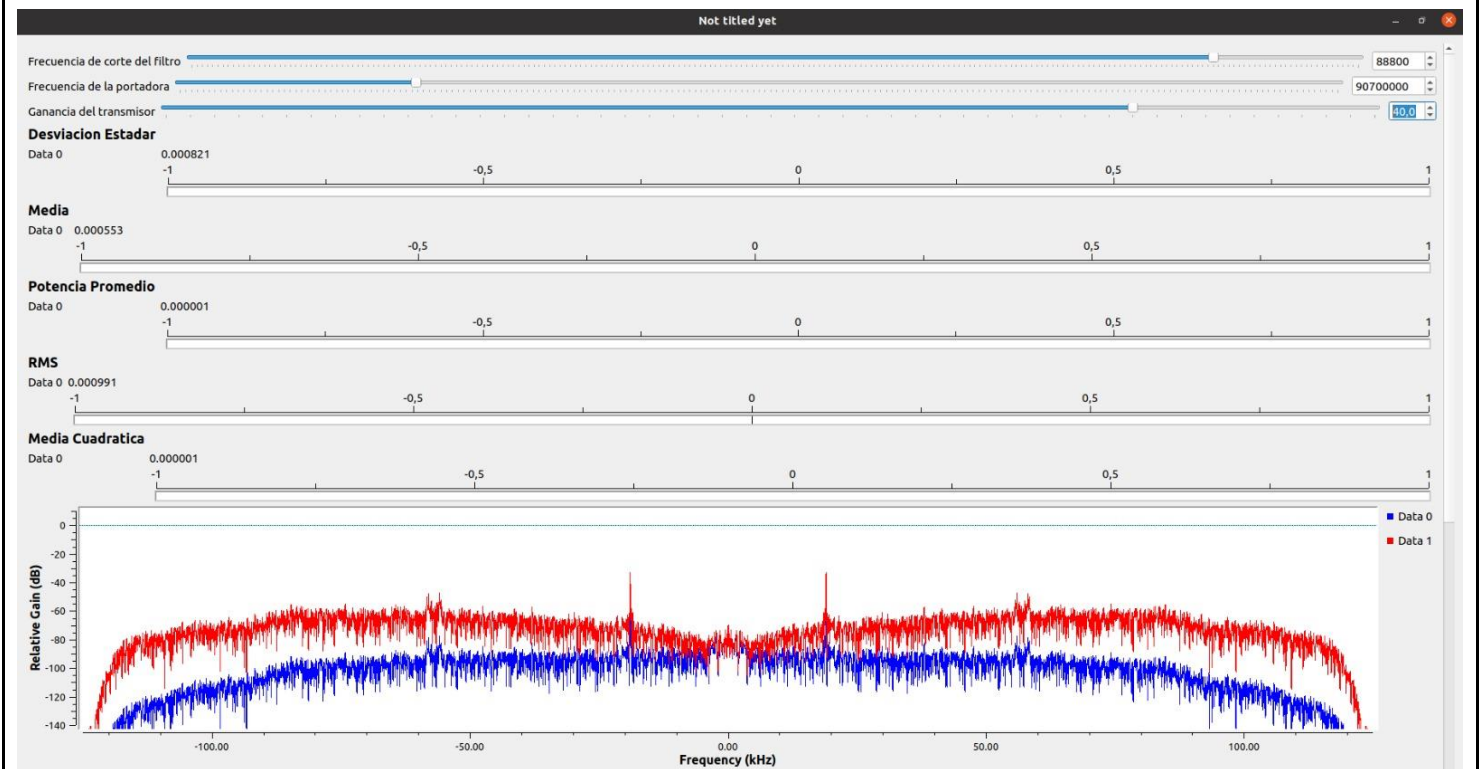
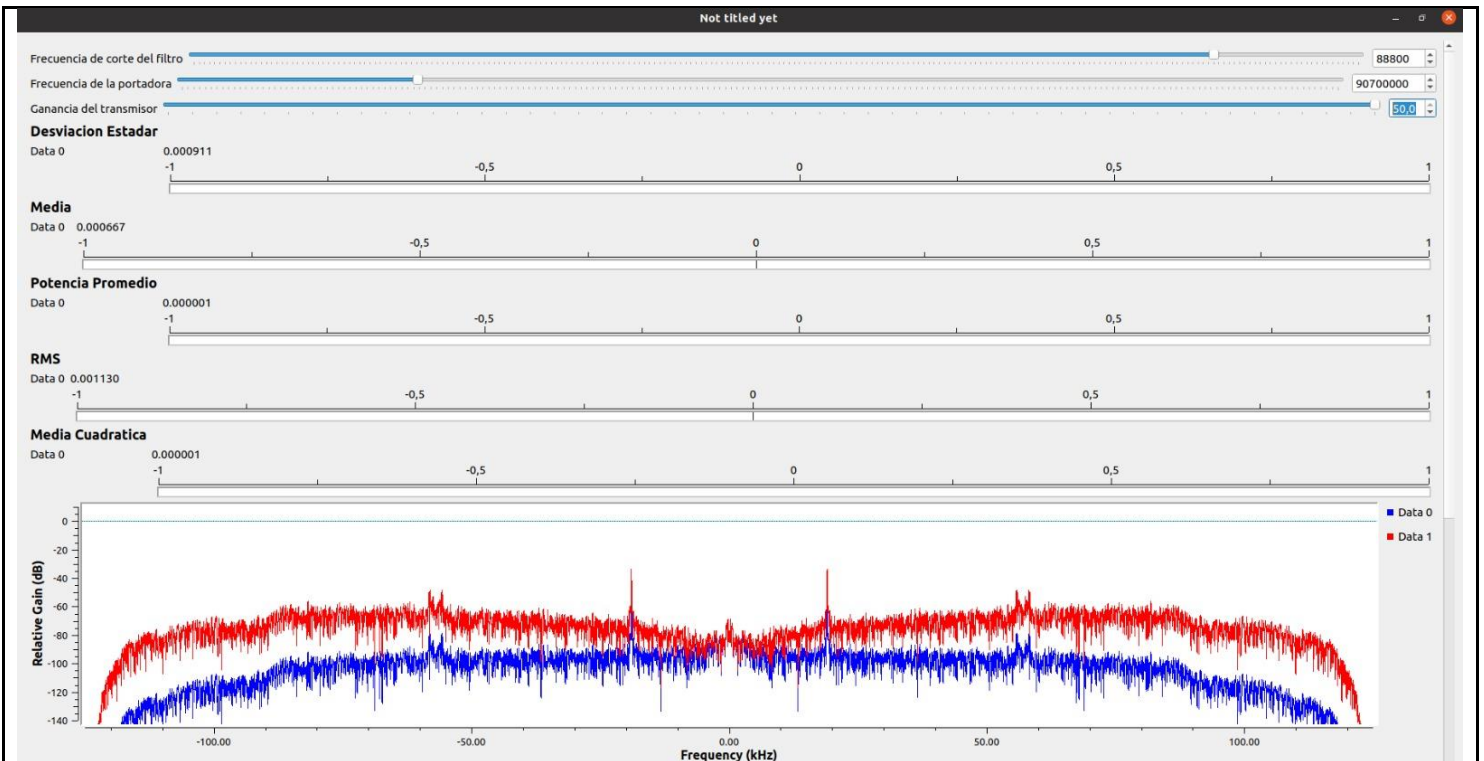
Como siguiente punto se propuso realizar estos cálculos, pero sobre una señal capturada en tiempo real, lo que se puede entender como una emisora o frecuencia en particular mediante el USRP y la antena al computador, esto nos permite entonces visualizar el trabajo que realiza el bloque creado por programación en situaciones o entradas reales como lo es una frecuencia específica.

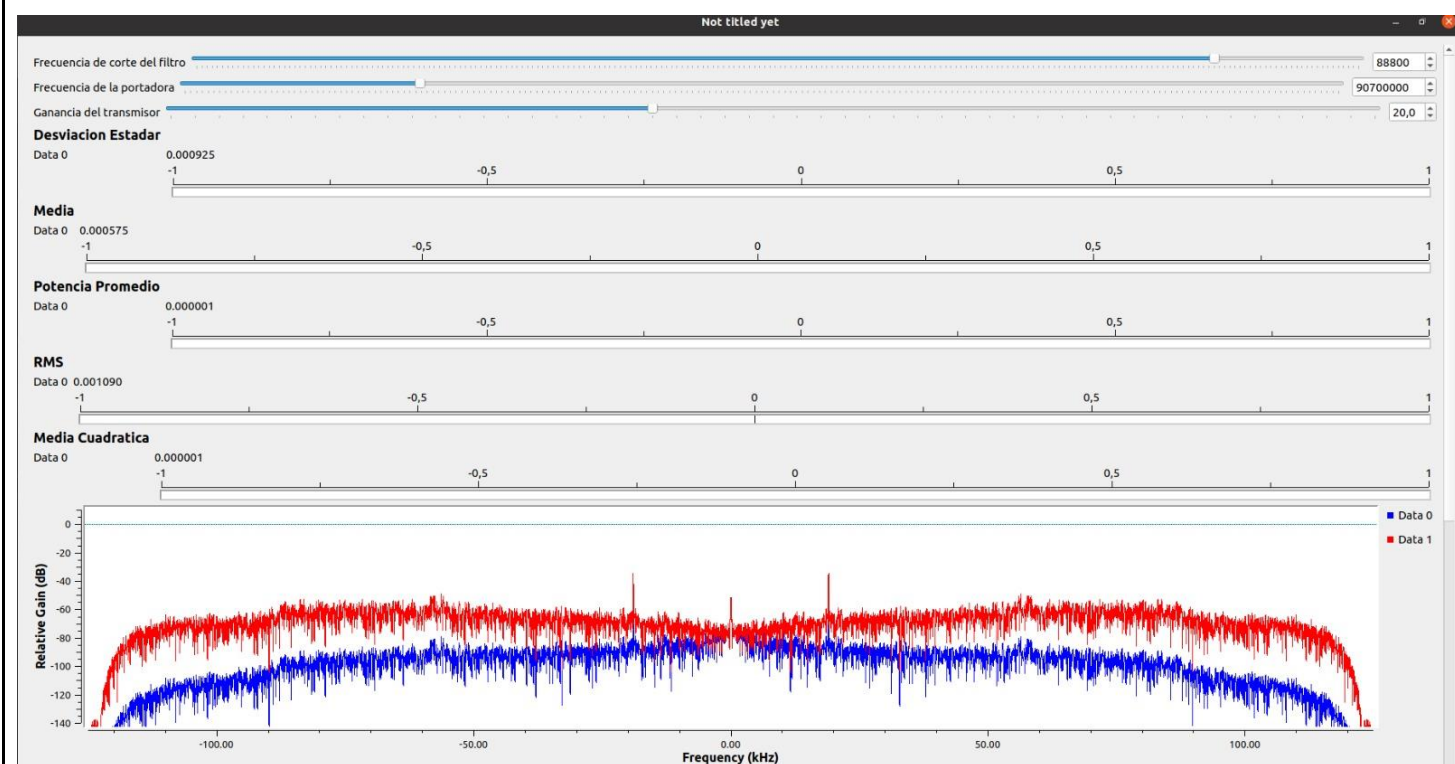
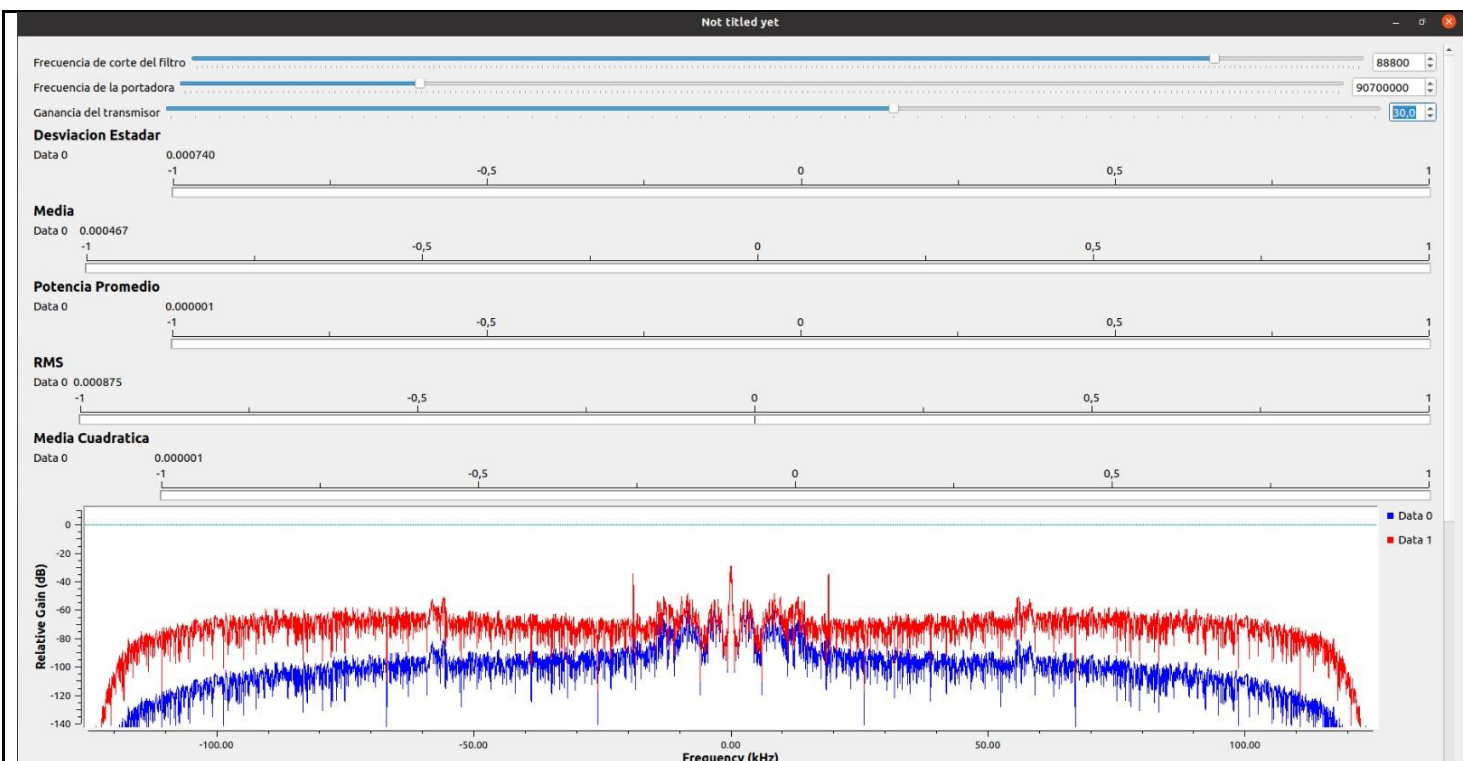
Para lograr una sintonización de una señal real como lo es una emisora fue necesario agregar bloques que cumplieran la función de demodular la información que provenia de dicha transmisión FM, para esto el flujograma se vio alterado y modificado de la siguiente manera:



Con este arreglo de bloques se pudo sintonizar y trabajar con una emisora real donde variando parámetros de filtrado y ganancia pudimos realizar la medición de las características como la potencia y valor RMS de una señal variando la calidad de sintonización, donde cada que se variaba la ganancia se notaba una distorsión o mejora del sonido que se estaba capturando, es así como procedemos a analizar los resultados que nos brinda el bloque creado en programación y con esto comparar el efecto de diferentes valores de ganancia a los resultados que este arroja.

Para cada analizar de manera mas directa el funcionamiento se fue modificando el valor de ganancia con la cual se estaba visualizando la señal. Con esto se podrán ver variaciones en los valores reflejados de cada variable. Como aclaración se siguieron usando los mismos cálculos que la puesta en práctica anterior.





Figuras 5,6,7,8. Señal de frecuencia 90.7Mhz con diferentes ganancias

Como resultado se pudo apreciar que la modificación de la ganancia no alteraba significativamente los valores obtenidos, lo que nos dice que el bloque actúa correctamente y nos muestra un calculo constante de la señal en tiempo real.

Por último dado que el funcionamiento del bloque fue lo esperado se deja en evidencia la programación del mismo y la utilización de la guía encontrada tanto en el libro como en las demás especificaciones dadas por el docente en la práctica. (Figura 10)

```
9 import numpy as np
10 from gnuradio import gr
11
12 class blk (gr.sync_block ):
13
14     def __init__ ( self ) : # only default arguments here
15         gr.sync_block . __init__ (
16             self ,
17             name = 'Promedios_de_tiempos', # will show up in GRC
18             in_sig =[ np.float32 ],
19             out_sig =[ np.float32 ,np.float32 ,np.float32 ,np.float32 ,np.float32 ]
20         )
21         self.acum_anterior = 0
22         self.Ntotales = 0
23         self.acum_anterior1 = 0
24         self.acum_anterior2 = 0
25
26     def work (self, input_items , output_items ):
27         x = input_items [0] # Senial de entrada .
28         y0 = output_items [0] # Promedio de la senial
29         y1 = output_items [1] # Media cuadratica de la senial
30         y2 = output_items [2] # RMS de la senial
31         y3 = output_items [3] # Potencia promedio de la senial
32         y4 = output_items [4] # Desviacion estandar de la senial
33
34         # Calculo del promedio
35         N = len(x)
36         self.Ntotales = self.Ntotales + N
37         acumulado = self.acum_anterior + np.cumsum(x)
38         self.acum_anterior = acumulado[N -1]
39         y0[:]= acumulado/self.Ntotales
40
41         # Calculo de la media cuadratica
42         x2=np.multiply(x,x)
43         acumulado1 = self.acum_anterior1 + np.cumsum(x2)
44         self.acum_anterior1 = acumulado1[N -1]
45         y1[:] = acumulado1/self.Ntotales
46
47         # Calculo de la RMS
48         y2[:] = np.sqrt(y1)
49
50         # Calculo de la potencia promedio
51         y3[:] = np.multiply(y2 ,y2)
52
53         # Calculo de la desviacion estandar
54         x3 = np.multiply(x-y0 ,x-y0)
55         acumulado2 = self.acum_anterior2 + np.cumsum(x3)
56         self.acum_anterior2 = acumulado2[N -1]
57         y4[:] = np.sqrt(acumulado2/self.Ntotales)
58
59         return len (x)
60
```

figura 10. Código del bloque “Promedios_de_tiempos”

Conclusiones:

- El uso de GNU radio facilita el análisis de señales y ofrece herramientas con las cuales se puede realizar gran variedad de operaciones, en el desarrollo de la práctica se pudo verificar el uso de varios bloques haciendo énfasis en el bloque “Python block”, mediante el cual se puede programar, con este se pudo realizar el cálculo de varios parámetros de una señal solicitados en la descripción de la guía de laboratorio.
- Para verificar de manera práctica el bloque de estudio se demoduló señales de radio usando varios bloques, entre los cuales estaba un filtro pasa bajas del cual se pudo observar una relación entre la frecuencia de corte del filtro y la calidad de recepción de las señales de audio (a mayor frecuencia de corte mejor calidad), por otro lado la ganancia del transmisor también genera una mejor recepción conforme se aumenta su valor.
- Mediante el uso de GNU Radio también se logró verificar valores de medición debido al uso del valor RMS, desviación estándar, valor medio, potencia de la señal y poder compararlos las mediciones prácticas con las experimentales.

7. REJILLA DE EVALUACION

Método de calificación por lista de cotejo

N°	Criterios	EXCELENTE (5)	BUENO (4)	REGULAR (3)
1	Los Procedimientos son completos y permiten cumplir el objetivo general y los objetivos específicos. Caso se solicite responder preguntas al final, estas son respondidas de forma adecuada y coherente.			
2	Los Resultados cumplen los siguientes criterios: <ul style="list-style-type: none">• Coherencia con los objetivos• Tienen Comentados de análisis pertinentes• Están en su totalidad (tiempo, frecuencia u otros solicitados)			
3	Calidad del informe: <ul style="list-style-type: none">• Es organizado de inicio a fin• Etiqueta imágenes y las cita en el texto• Tiene ortografía• La escrita es clara y concisa• No repite informaciones			

4	Tiene al menos una conclusión que sea resultado directo de la ejecución del laboratorio				
Total					

