MIGUEL ÁNGEL CALIFA URQUIZA - 1160950

CARLOS MAURICIO PALLARES – 1161079

MARIA INES MEJIA CUELLAR – 1160373

JUAN MIGUEL GALEZO GOMEZ – 1160300

DUVAN VARELA - 1161498  
***Universidad Francisco de Paula Santander***

CONSTRUCCION DE UN ANALIZADOR DE ESPECTRO EMPLEANDO ARDUINO Y MATLAB

***Resumen—*El analizador de espectro es un instrumento de laboratorio empleado para observar las ondas electromagnéticas en el espectro de la frecuencia y poder interpretar mediante ello su comportamiento en diferentes medios, la implementación requiere de dispositivos electrónicos de alta velocidad e interfaces que permitan el filtrado y análisis rápido de estas ondas.**

***Palabras claves—* Analizador de espectro, Onda electromagnética, Frecuencia, Filtrado,** **Estudio de ondas.**

# INTRODUCCIÓN

Actualmente se emplean simulaciones para las prácticas de laboratorio, esto se debe a que las ondas electromagnéticas son invisibles y por ende su estudio se vuelve netamente teórico. En la búsqueda de la innovación de las prácticas de laboratorio se ha definido como proyecto final realizar un analizador de espectro empleando las herramientas disponibles en el mercado local y nacional.

El ingeniero electrónico en su perfil profesional tiene como visión innovar y esto siempre va de la mano con la capacidad de minimizar los tamaños y maximizar la funcionabilidad, datos que pueden ser corroborados con la evolución de la telefonía celular. Medios de transmisión es el espacio en el que el ingeniero forma bases conceptuales para el entendimiento del fenómeno electromagnético y empieza a ver las ventajas al implementar sistemas electrónicos empleando sistemas embebidos y lógica programable.

Un analizador de espectro es un dispositivo capaz de mostrar las características de una onda a una o a un rango de frecuencias establecido, este dispositivo será ensamblado sobre la tarjeta Arduino debido a que es uno de los sistemas embebidos más empleado en el ámbito investigativo.

Teniendo en cuenta los materiales disponibles, el tiempo de distribución y la documentacion proporcionada por el fabricante se puede concluir que la mejor combinación de elementos para hacer este proyecto es la integración del Arduino uno y el modulo Si 4432 de Silicon Labs, debido a que este modulo permite mediante el IDE de Arduino programar la frecuencia de recepción y calcular matemáticamente la intensidad de la señal recibida a dicha frecuencia.

# Objetivos

## Objetivo General

## Diseñar e implementar un analizador de espectro.

## Objetivos Específicos

* Recopilar información, estudiar el estado actual del arte, conocer los principales conceptos que rodean la elaboración de este dispositivo.
* Diseñar en un esquema, un bosquejo general del sistema.
* Elaborar un diseño electrónico de los componentes de hardware que realizaran cada una de las tareas identificadas en el esquema general del sistema.
* Realizar la implementación y la programación de los módulos para realizar la visualización de los niveles de señal en el espectro de la frecuencia.
* Implementar la interfaz grafica en Matlab y acoplar los sistemas empleando el puerto serie virtual disponible en los sistemas operativos (Windows).

# Marco teórico

Analizador de espectro

Un analizador de espectro es un dispositivo electrónico capaz de mostrar en una pantalla las características de una señal en un espectro de frecuencias de cualquier señal que se encuentre en la entrada, es un dispositivo bastante útil que a pesar de no poder medir los valores de campo eléctrico ni magnético puede proveer la una gran parte de la información necesaria para hallarlos, tiene aplicaciones en el área de las telecomunicaciones, sobre todo para el análisis de señales en antenas mediante su factor K [1].

Módulo Si-4432

Los dispositivos Si4430/31/32 son transceptores inalámbricos los cuales vienen encapsulados y adaptados en un solo circuito integrado, lo que simplifica no solo el diseño de dispositivos de radiofrecuencia, sino que reduce los costes de estas. Estos dispositivos permiten tener cobertura desde 240 MHz hasta los 960 MHz, permitiendo un control preciso de sintonía con sus pasos de 156 Hz o 312 Hz. También tiene integrado un ADC y un modem basado en DSP que realiza demodulación, filtrado y manejo de paquetes [2].

Arduino

Una placa Arduino es una PCB basada en un microcontrolador y que incorpora pines de entrada y salida que facilita la implementación de sistemas de control al permitir de manera muy sencilla conectar sensores y actuadores. Las placas Arduino se dividen en varios modelos que varían en sus capacidades y tamaños por lo que se debe escoger correctamente el modelo que mejor se adapta a las necesidades del proyecto a realizar, dos de estos modelos más utilizados son el Arduino UNO, el cual por su sencillez y bajo coste es muy comúnmente utilizado para diversas aplicaciones y para aprendizaje del lenguaje, otro muy común es el Arduino PRO MINI, el cual por su tamaño y capacidad, además de su bajo coste permite gran versatilidad en sus aplicaciones [3].

Pantalla Nokia 5110

Esta pantalla utiliza el chip PCD8544 de Philips y fue utilizado en los celulares Nokia 5110 y 3310, funciona a 3.3 V y hoy en día se ha estado haciendo muy popular por ser un LCD económico pero que permite mostrar gráficos y visualizar aplicaciones como el popular juego de SNAKE que se podía disfrutar en los antiguos teléfonos Nokia [4].

# Procedimiento y Desarrollo de Operaciones

En primera instancia se requiere de una investigación en las bases de datos proporcionadas por la universidad, como science direct y en fuentes externas como Google Scholar,`donde se evidencio la falta de investigación hacia estos dispositivos.

Por lo tanto investigando en blogs y paginas de terceros se ha encontrado un articulo que se titula “Arduino 240~930 MHz Spectrum Analyser” en el cual el autor 吳勻昌 afirma haber ensamblado un analizador de espectro empleando un Arduino UNO, el modulo SI-4432 y una pantalla Nokia 5110 [5].

Arduino también posee un foro para desarrolladores e investigadores, en el cual una gran comunidad aporta iniciativas y resultados de experimentos que en algunas ocasiones no llegan a ser artículos científicos pero contienen resultados que pueden ser empleados como base inicial, este es el caso del articulo denominado “Arduino Spectrum Analyzer” [6] en el cual se expone la misma temática abordada en la presente investigación.

En cada uno de los referentes anteriormente mencionados se encuentran útiles consejos, herramientas y bases conceptuales para el proyecto a ensamblar.

Como primer paso se va a realizar un bosquejo general del sistema a implementar, empleando graficos en línea para realizar cada uno de los bosquejos. (www.lucidchart.com/).

## Bosquejo general.

En primera estancia debido a que el objetivo principal es hacer el analizador de espectro se conoce que el comienzo del proceso son las señales electromagnéticas captadas por un sensor, y el final del proceso es la comunicaicon con el computador que tiene Matlab preinstalado de la siguiente manera:

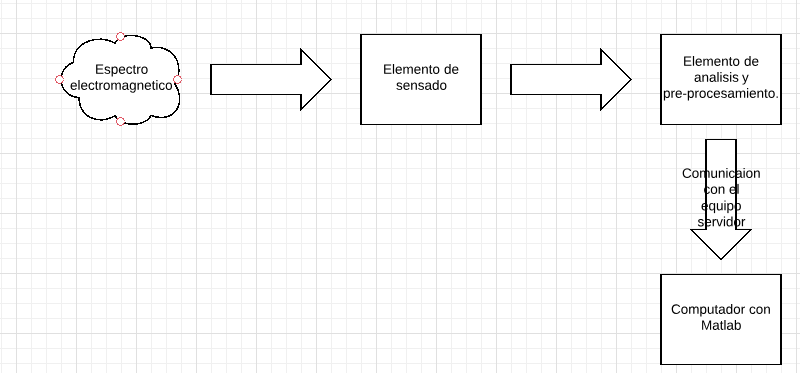


Figura 1. Bosquejo general del sistema

Ahora cada elemento debe ser expuesto de una manera mas explicita teniendo en cuenta la función que va a realizar.

El primer elemento y mas importante de la cadena es el elemento de sensado, debido a que debe ser capaz de obtener la onda electromagnética, filtrarla según la frecuencia seleccionada en la computadora y determinar la potencia de la señal recibida, por ello hemos optado por emplear el modulo Si-4432 que posee todo esto integrado y se comunica mediante una interfaz SPI simple.

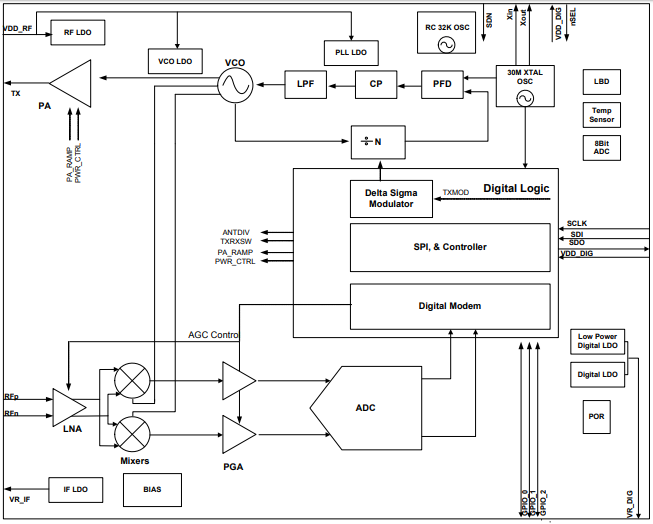


Figura 2. Diagrama interno del modulo Si-4432 [2]

Como se observa el modulo cuenta previamente con los elementos de control necesarios para realizar el filtrado y análisis de la señal, dependiendo de los parámetros que se indiquen por medio de señales electrónicas atravez del bus que usa protocolo SPI para comunicarse.

A este sensor debe acoplarse un elemento intermedio que realice la comunicación entre el computador y el dispositivo. Este rol será asumido por la tarjeta Arduino uno, la cual posee conexión serial via usb para programación que puede ser empleada como canal de comunicación entre el computador y el sensor, además de ello posee el bus de comunicación SPI que permite acoplarse al sensor y realizar el sensado de manera optima.

Como ultimo elemento se empleara un computador de laboratorio de la universidad Francisco de Paula Santander que posee procesador core i7 7700, 16 GB de memoria RAM y disco duro de 1TB que cuenta con Matlab versión estudiantil en el cual se desarrollara las interfaces del aplicativo Windows.

En resumen el bosquejo general se concluye de la siguiente manera:

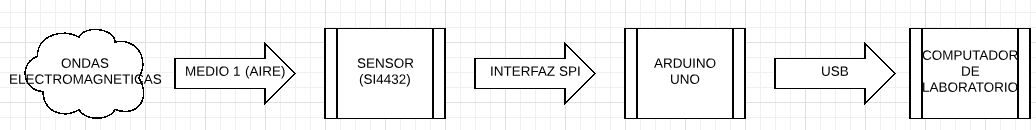


Figura 3. Bosquejo general (vista hardware)

## Diseño del software

El modulo Si-4432 desempeña un rol fundamental en el desarrollo del proyecto, sin embargo no realiza ninguna tarea sino es previamente programada por el bus SPI, por ello el fabricante del modulo proporciona atravez del datasheet toda la información pertinente para realizar la configuración y el uso básico y avanzado del modulo.

Al modulo Si-4432 se conecta la tarjeta Arduino uno por medio del bus SPI, por ello se asume que el encargado de precargar estas configuraciones será el Arduino uno, para ello se requiere de programación en Arduino IDE capaz de obedecer a las especificaciones del fabricante.

Para esta tarea se empleara la librería RF22 bastante extendida para el manejo de estos modulos, la cual posee el método Begin(), con el cual incializa la comunicación y carga los parámetros básicos. El método setFrequency() con el cual fija la frecuencia de recepción y el método calculateRssi() el cual calcula la potencia de la señal de llegada.

Empleando estos métodos se puede obtener los parámetros de la onda, acción seguida empleando la librería Serial pre-existente en el IDE de Arduino, se realiza la transmision via puerto serie de los parámetros de la onda.

## Implementación del hardware.

Las tarjetas Arduino usualmente se alimentan a 5V VCC, sin embargo el modulo Si-4432 se alimenta a 3V VCC, por lo que cualquier voltaje por encima de 3V puede causar mal funcionamiento del modulo o inclusive un daño irreparable.

Para solucionar esta problemática la tarjeta Arduino Uno será reemplazada por la tarjeta Arduino Pro Mini la cual puede ser alimentada a 5V o a 3v3, con lo cual el voltaje de entrada, lógico y de comunicación estará por los niveles que el sensor tolera.

En el IDE de Arduino basta con declarar la nueva tarjeta y el compilador realiza automáticamente los ajustes necesarios para transmitir un código que realiza exactamente las mismas tareas en la tarjeta Arduino Pro Mini.

Finalmente, la implementación prevista es de la siguiente manera:

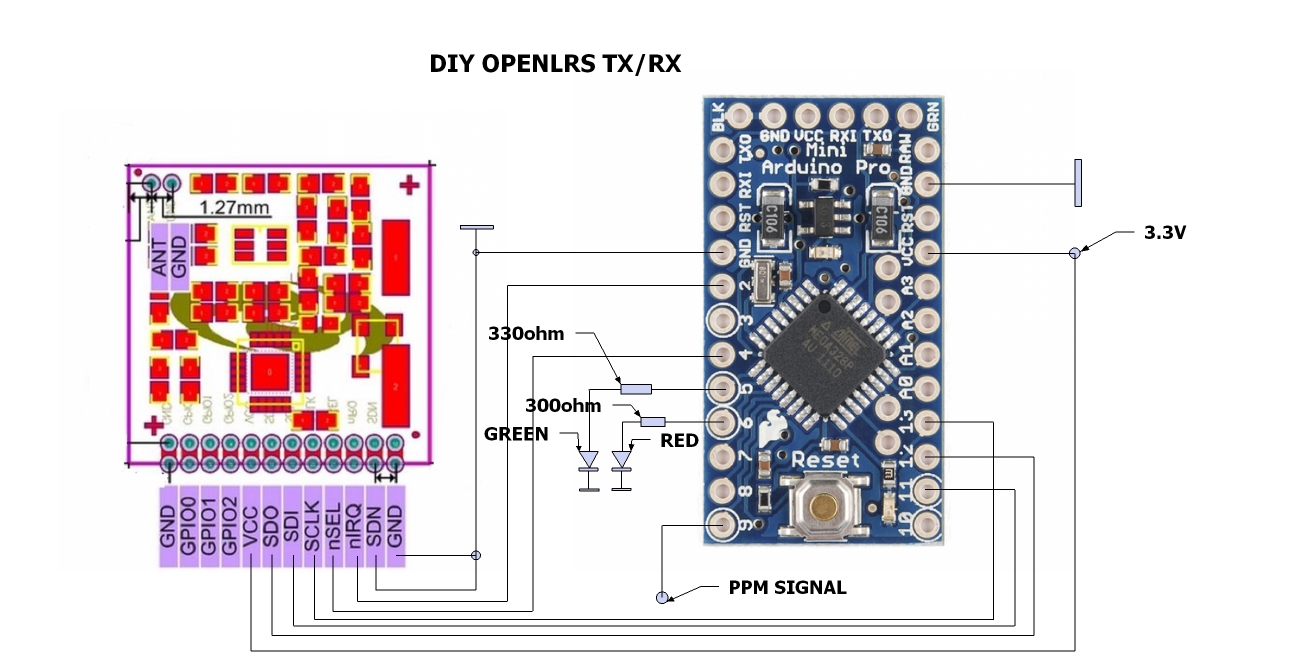


Figura 4. Implementacion del modulo Si-4432 con el arduino Pro Mini [7]

## Implementación de la interfaz grafica de usuario (GUIDE) en MATLAB.

En Matlab es posible crear una interfaz grafica de usuario gracias al editor GUIDE que trae incorporado Matlab.

Luego de incorporar en el espacio de trabajo los elementos básicos como:

* Lista desplegable para seleccionar el puerto de comunicación.
* Boton de conexión.
* Cajas para visualización de mensajes.
* Eje para grafica principal.
* Eje para graficos auxiliares (Logo universidad)
* Slider (Botones para seleccionar la sección de espectro a visualizar).

Inmediatamente se colocan estos botones sobre el lienzo de trabajo, la interfaz toma el siguiente aspecto:

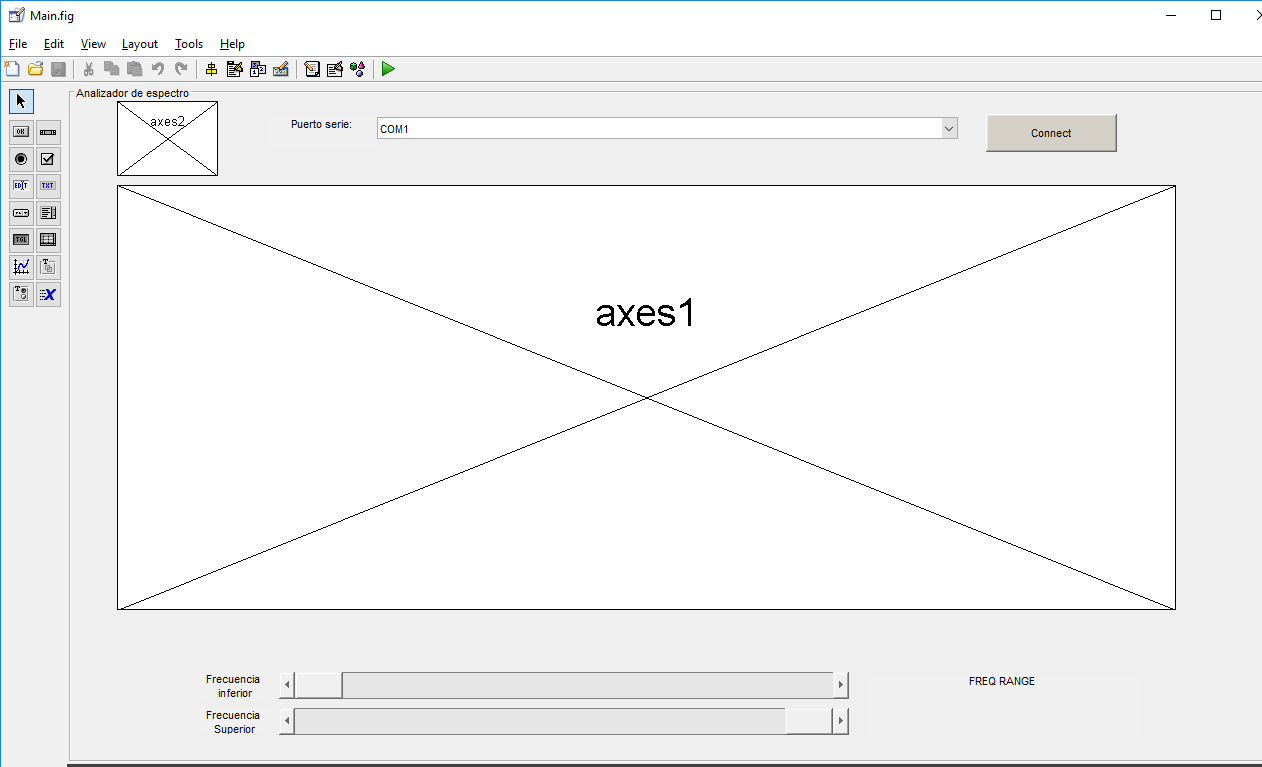


Figura 5. Interfaz grafica v1.00

Luego de ello, automáticamente Matlab crea los eventos y las configuraciones en el archivo de extensión .m que indica que es código de Matlab.

En el se programa cada uno de los eventos que se ejecutaran tras la ejecución de cada uno de los botones.

Hasta el momento se ha realizado la programación del evento que inicia la comunicación del puerto serie y la grafica punto a punto.

El código se encuentra en el anexo 2 y el código completo de Arduino y de Matlab se encuentra en el repositorio de Github [8].

Finalmente y luego de ensamblado todo el sistema, la señal de salida se observa tal cual el anexo numero 3 del documento.

# Conclusiones

El estudio practico de las ondas electromagnéticas requiere del uso de electrónica de estado sólido con velocidades de reloj altas para poder observar el comportamiento de la onda en el espacio libre y determinar los factores que introducen ruido, error y perturbación mientras la señal se desplaza en el medio.

El voltaje de alimentación en equipos que realizan mediciones desempeña un rol fundamental debido a que la mas mínima variación de voltaje en la fuente introduce un error considerable al sistema.

Los módulos que actualmente se encuentran disponibles en el mercado para la transmisión y recepción de señales via inalámbrica por medios no guiados no poseen la opción para graficar la señal análoga proveniente del medio ambiente, debido a la frecuencia tan elevada a la que transmite el modulo.

Los niveles de ruido existentes en el medio ambiente impiden que el modulo tome una referencia de 0 en cuanto a RSSi, por lo que se puede considerar que una gran parte del espectro electromagnético libre se encuentra perturbado en todo momento.

Al realizar pruebas en espacios con alto contenido de estatica (cuando existe aire acondicionado), causa una notable elevación del nivel de error percibido por el sensor.

La interfaz serial no es la más adecuada al momento de realizar transmisión de un vector de datos grande (Largo de la palabra mayor a 512Kbytes), esta experiencia se evidencio durante el desarrollo del GUIDE de Matlab en el cual se realizo un cambio en el parámetro de entrada del puerto serie para permitir la recepción de todos los datos en una sola línea.

# Referencias Bibliográficas

|  |  |
| --- | --- |
| [1] | IngenieriaElectronica.org, «IngenieriaElectronica,» 13 07 16. [En línea]. Available: https://ingenieriaelectronica.org/analizador-de-espectros-definicion-tipos-y-caracteristicas/. [Último acceso: 01 05 18]. |
| [2] | Silicon Labs, «flytron,» 01 2009. [En línea]. Available: http://www.flytron.com/pdf/si4432.pdf. [Último acceso: 06 04 2018]. |
| [3] | Arduino, «Whats arduino,» [En línea]. Available: IngenieriaElectronica. |
| [4] | hispavila, 28 06 2015. [En línea]. Available: https://www.hispavila.com/nokia-5110/. [Último acceso: 27 04 2018]. |
| [5] | 吳勻昌, «Yulie.wu,» 04 2017. [En línea]. Available: http://yulie-wu.blogspot.com.co/2017/04/arduino-240930-mhz-spectrum-analyser.html. |
| [6] | F. Arduino, «Forum Arduino,» Arduino, 22 02 2015. [En línea]. Available: http://forum.arduino.cc/index.php?topic=302897.0. [Último acceso: 10 03 2018]. |
| [7] | e\_lm\_70, rcgroups, 01 03 2014. [En línea]. Available: https://www.rcgroups.com/forums/showthread.php?2114859-Cheap-Easy-DIY-openLRS-openLRSng-RX-TX-modules. [Último acceso: 28 03 2018]. |
| [8] | M. A. C. Urquiza, «Github,» 17 03 2018. [En línea]. Available: https://github.com/miguel5612/Espectrum-Analizer-Arduino. [Último acceso: 01 05 2018]. |

# ANEXOS

ANEXO 1. Codigo fuente (Tarjeta Arduino pro mini)

#include <RF22.h>

#include <SPI.h>

#include<stdlib.h>

char temp[51];

uint8\_t rssi;

RF22 rf22;

void setup()

{

Serial.begin(115200);

while (!rf22.init()){

Serial.println("RF22 init failed");

delay(500);

}

Serial.print("RF22 OK");

rf22.setModemConfig(RF22::GFSK\_Rb2Fd5);

rf22.setModeRx();

}

float Start = 430;

float End = 460;

float Step = 0.1;

float freq;

void loop()

{

ProcessRx();

for (freq = Start; freq < End; freq += Step)

{

rf22.setFrequency(freq);

delayMicroseconds(4000); // Let the freq settle

rssi = rf22.rssiRead();

if(rssi>254){setup();}

Serial.print(rssi, DEC);

Serial.print(",");

//Serial.println(dtostrf(freq,0,3,temp));

}

Serial.println();

//Serial.println("");

}

void ProcessRx() {

byte n = 0;

byte m = 0;

char\* sptr1;

char\* tempstr;

if (Serial.available()) {

temp[n] = Serial.read();

while ((temp[n] != '\n')&&(n<50)) {

if (Serial.available()) {

n++;

temp[n] = Serial.read();

}

}

//Serial.println(temp);

tempstr = strtok\_r(temp,",\n",&sptr1);

do

{

switch (m) {

case 0: // Wheel 1

Start = atof(tempstr);

break;

case 1: // Wheel 2

End = atof(tempstr);

break;

case 2: // Wheel 3

Step = atof(tempstr);

break;

}

m++;

}

while (tempstr = strtok\_r(NULL, ",\n",&sptr1));

}

}

Anexo 2. Codigo base V1.00 (Sin interfaz grafica).

clc

clear all

delete(instrfindall);

s = serial('COM4','BAUD',115200);

s.InputBufferSize = 2048;

fopen(s);

data1 = fgetl(s)

splittedPoints1 = strsplit(data1,',')

numbers1 = str2double(splittedPoints1)

[m,n] = size(splittedPoints1)

mychar = splittedPoints1(n)

StrEnd = mychar{1}

if strfind(StrEnd,'END')

h=plot(1);

numbers3 = [numbers1(4:n-1)]

f = numbers1(2):numbers1(4):numbers1(3)

set(h,'XData',f,'YData',numbers3)

title('Signal Analysis')

ylabel('Power received RSSi (dBm)')

xlabel('Frecuency (MHz)')

legend('RSSi')

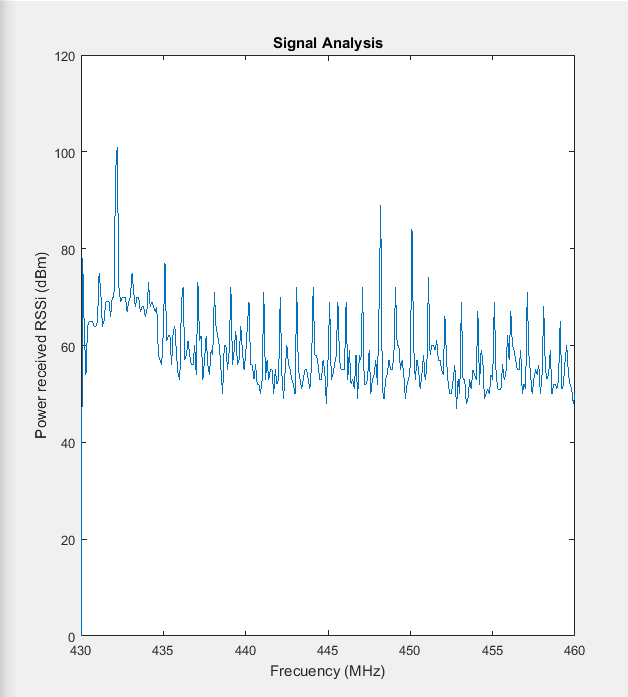
end

fclose(s)

delete(s)

clear s

Anexo 3. Visualizacion de las señales empleando el código base desarrollado en MATLAB.



Anexo 4. Visualizacion de las señales empleando la interfaz grafica desarrollada en MATLAB:

