

GRADO EN INGENIERÍA INFORMÁTICA DE GESTIÓN
Y SISTEMAS DE INFORMACIÓN

TRABAJO FIN DE GRADO

***ESTUDIO DE CARACTERÍSTICAS DE
LAS TECNOLOGÍAS INALÁMBRICAS
PARA LA TRANSMISIÓN DE
PEQUEÑOS VOLÚMENES DE DATOS***

DOCUMENTO ADJUNTO – MANUAL DE USUARIO

Alumno/Alumna: Jose Miguel Vega Díaz

Director/Directora (1): Javier Jesús Bilbao Landache

Curso: 2018-2019

Fecha: Bilbao, 17, 07, 2019

Contenido

INTRODUCCION	4
HERRAMIENTAS	5
Arduino IDE.....	5
XCTU.....	5
PyCharm	5
PERIFÉRICOS.....	6
General	6
Arduino UNO.....	6
KIT ARDUINO	6
Cable alimentación Arduino.....	7
Cable conexión Arduino	8
Wi-fi.....	8
ESP8266 – 001	9
ESP8266 USB ADAPTER	9
Bluetooth.....	9
BLE HC-08.....	10
RF.....	10
NRF24L01 + PA +LNA.....	10
NRF24L01 Adapter	11
ZigBee	11
XBEE XB24CZ7WIT – 004	11
XBee Explorer USB	12
XBee Explorer Regulated.....	12
Cable alimentación XBee Explorer USB.....	13
WI-FI	14
Materiales requeridos.....	14
Configuración	14
Conexión	14
Software.....	15
Proceso de prueba.....	18
BLUETOOTH	23
Materiales requeridos.....	23
Configuración	23
Conexión	23
Software.....	23

Proceso de prueba.....	24
ZIGBEE.....	28
Materiales requeridos.....	28
Configuración	28
Conexión	28
Software.....	28
Proceso de prueba.....	28
Proceso de análisis de espectro	36
RF.....	38
Materiales requeridos.....	38
Configuración	38
Conexión	38
Software.....	38
Proceso de prueba.....	39

INTRODUCCION

Este documento tiene como objetivo complementar a la memoria dada, para que cualquier usuario pueda seguir los pasos del estudio que le interesen de manera simple y más guiada en su detalle.

Contendrá una guía de materiales y herramientas donde podremos ver los periféricos y software necesarios que se han utilizado en este trabajo además de enlaces de compra para facilitar la obtención de los mismos.

Está dividido por secciones tecnológicas ya que la realización de pruebas de cada es completamente independiente de las demás. Cada sección mencionara los materiales que se procederán a usar, la configuración realizada tanto de conexión como de software y un ejemplo de evidencias de la realización de una prueba.

Todas las imágenes que podemos encontrar en este manual están extraídas del documento principal de estudio donde se encuentran referenciadas o son extraídas de los enlaces de compra que se muestran en el documento.

Todos los códigos utilizados se encuentran en los anexos de la documentación del estudio y serán referenciados. De forma adicional se encontraran también en una carpeta entregada con este trabajo de fin de grado.

HERRAMIENTAS

Arduino IDE

Para poder programar la gran mayoría de los módulos que se utilizarán en el proceso de realización de pruebas se utilizará el software propio de Arduino que descargaremos de su página oficial de Arduino -> www.arduino.cc . Este software funciona tanto en Mac como Linux y Windows, es de código abierto y está escrito en java.

XCTU

Para poder configurar nuestros módulos XBee, realizar pruebas y hacer lecturas de espectro utilizaremos la app gratuita XCTU. Esta app es multiplataforma y la podremos encontrar en la página oficial de Digi -> www.digi.com .

PyCharm

Para poder realizar el script de prueba utilizado para probar el módulo Wi-fi necesitaremos una plataforma para desarrollar en python. Nosotros hemos escogido PyCharm ya que es con la que más estamos familiarizados en el grado de ingeniería informática. Esta herramienta la podremos encontrar en la página oficial de JetBrains -> www.jetbrains.com. Para poder utilizarla necesitaremos una licencia que deberemos comprar o por el contrario en caso de ser estudiantes universitarios utilizar la que nos proporciona la universidad.

Serial Bluetooth Terminal

Serial Bluetooth Terminal es una app pública para Android de Kai Morich y se puede encontrar en el siguiente [enlace](#). Esta app nos permitirá realizar la conexión desde nuestro dispositivo al módulo Bluetooth y realizar los envíos necesarios. En ella veremos los tiempos de envío y de conexión además de los mensajes recibidos.

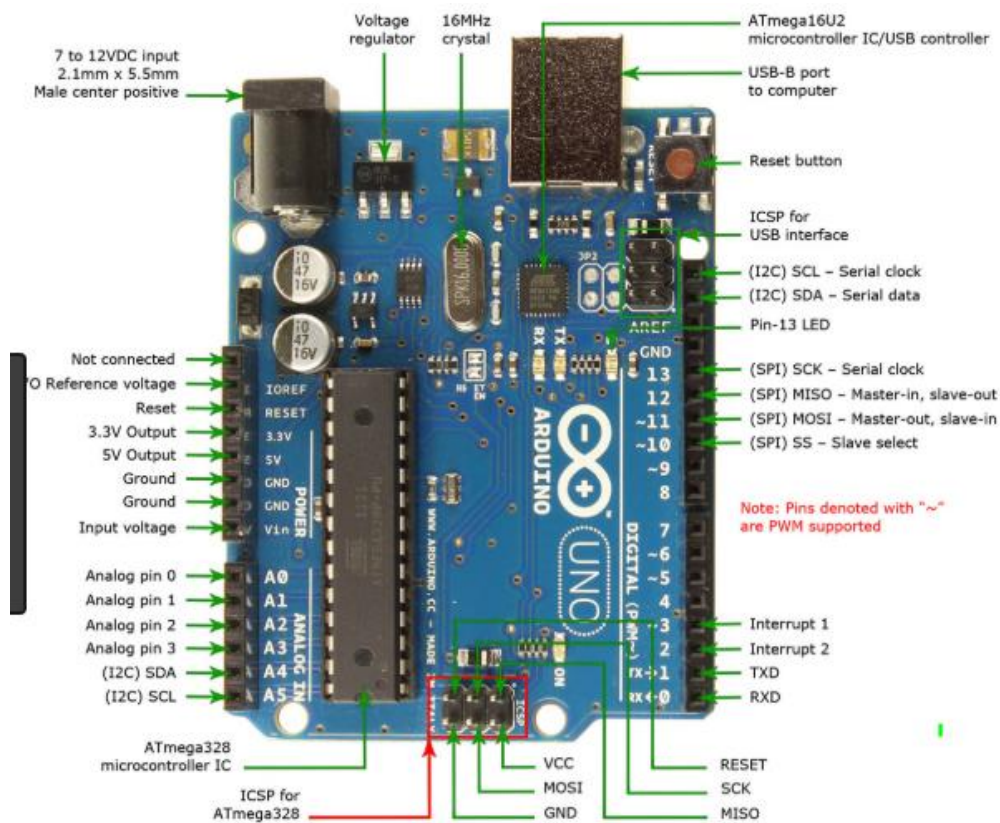
PERIFÉRICOS

General

Aquí se muestran los materiales que se utilizan para la mayoría de las tecnologías probadas si no es para todas. Veremos una imagen de cada uno a parte de comentar algunas especificaciones y se mencionara alguna tienda donde se puedan adquirir.

Arduino UNO

Utilizaremos dos placas microcontroladoras para la programación de la mayoría de los dispositivos. Más en concreto usaremos dos Arduino Uno Rev 3 que poseen las siguientes características.

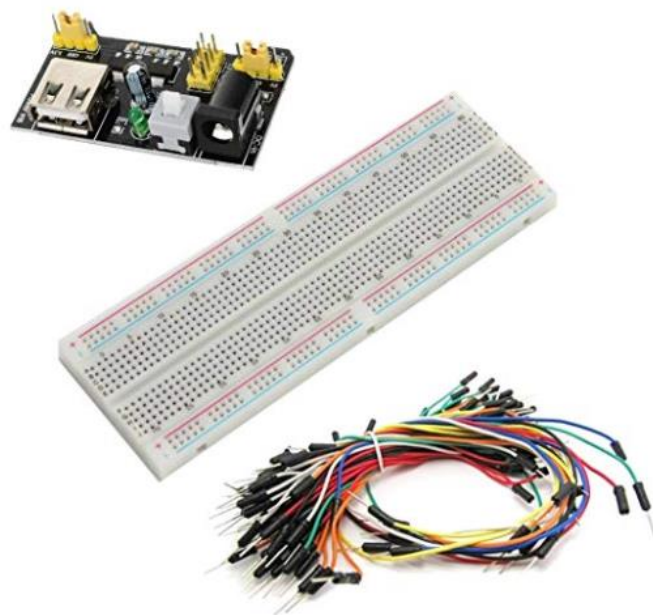


Las especificaciones concretas de este dispositivo las encontramos más a fondo en las datasheet adjuntadas a este estudio. Esta placa microcontroladora la podremos obtener en <https://tienda.bricogeek.com/arduino/305-arduino-uno.html>.

KIT ARDUINO

Para realizar las conexiones y alimentar algunos de los dispositivos es interesante tener o bien los cables y fuentes de alimentación por separado u obtener un kit básico de Arduino que incluya estos materiales:

- Una fuente de alimentación externa que nos proporcione 3,3V o bien 5V.
- Una protoboard para realizar las conexiones de manera más cómoda. (Opcional)
- Un conjunto de cables para realizar las conexiones que se requieran,



Cualquier Kit que contenga estos elementos es recomendable para la realización de este trabajo. Se pueden encontrar en la tienda online de Amazon de todas las categorías. ([Enlace de compra](#)).

Cable alimentación Arduino

Este cable es opcional ya que se puede alimentar perfectamente las placas Arduino mediante el cable de conexión USB pero es recomendable tener al menos uno.



Este cable debería proporcionar entre 7 y 12V de corriente DC ya que es la corriente que soporta de entrada las placas Arduino. Se puede encontrar en la tienda oficial de Amazon ([Enlace de compra](#)).

Cable conexión Arduino

Utilizaremos dos cables de conexión entre el Arduino y los ordenadores de realización de pruebas.



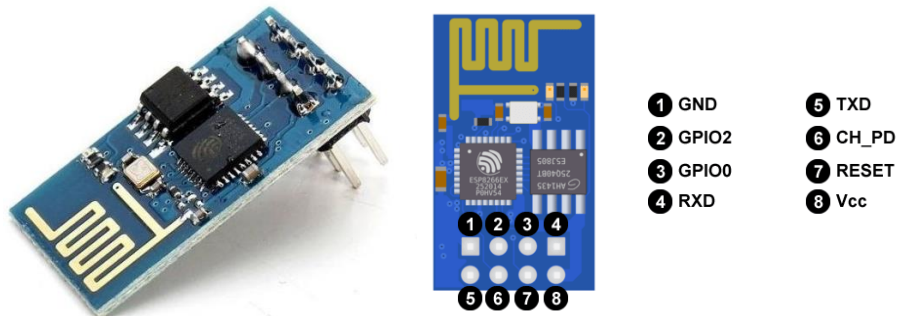
Este cable nos permitirá poder programar las placas de Arduino utilizando el software Arduino IDE y poder ver las recepciones de los dispositivos en el ordenador. Se puede encontrar en la tienda online de Amazon ([Enlace de compra](#)).

Wi-fi

En este apartado veremos los materiales necesarios para la realización de pruebas sobre la tecnología Wi-fi.

ESP8266 – 001

Utilizaremos un módulo ESP8266 – 001 para realizar las pruebas, este hará de punto de acceso al que realizaremos las peticiones de información. Como se refleja en la documentación, este dispositivo posee dos modos: modo UART y modo FLASH. En modo UART cargaremos la información en la memoria del dispositivo y en modo FLASH el dispositivo ejecutara las instrucciones. Es un módulo muy asequible que tiene una amplia gama de funcionalidades. Se puede obtener en la tienda online de Bricogeek por 4,50€ ([Enlace de compra](#)).



ESP8266 USB ADAPTER

Para realizar la programación del módulo ESP8266 es recomendable adquirir un adaptador USB serial. Se puede realizar la conexión a través del Arduino bloqueando el procesador del mismo pero resulta más sencillo de la manera que proponemos.

Estos adaptadores se pueden obtener por muy bajo precio en la tienda online de Amazon ([Enlace de compra](#)).

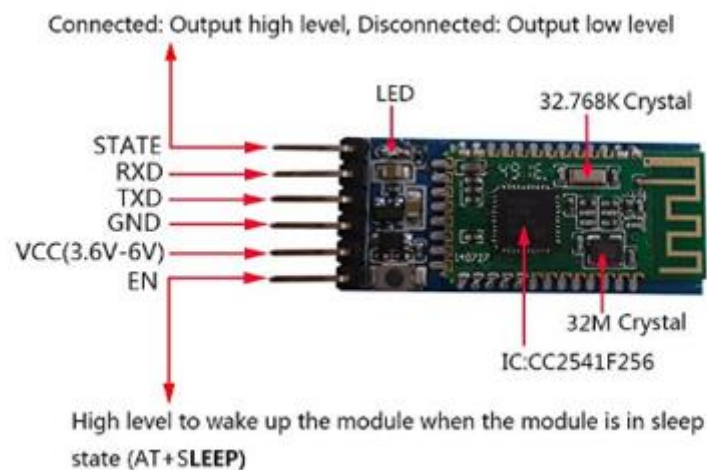


Bluetooth

En este apartado veremos los materiales necesarios para la realización de pruebas sobre la tecnología Bluetooth.

BLE HC-08

Utilizaremos un módulo Bluetooth BLE HC-08. Este módulo utiliza tecnología Bluetooth 4.0 de bajo consumo y se asemeja a los modelos que actualmente podemos tener en nuestros dispositivos del día a día. Lo podremos obtener en la tienda oficial de Amazon por un precio bastante asequible: ([Enlace de compra](#)).



RF

En este apartado veremos los materiales necesarios para la realización de pruebas sobre la tecnología RF.

NRF24L01 + PA +LNA

Para la tecnología RF son necesarios dos dispositivos NRF24L01 con o sin antena. En este estudio se han escogido y se mostrara en ejemplo los dispositivos con antena para poder alcanzar un mayor rango.



Estos dispositivos se pueden encontrar en la tienda online de Amazon por menos de 10€ .

NRF24L01 Adapter

Adicionalmente es recomendable obtener unos adaptadores para asegurarse de que el voltaje que le llega a cada uno de los pines de los módulos es el correcto y no se dañan por error. Estos adaptadores se pueden, al igual que los modulos, obtener en la tienda online de Amazon ([Enlace de compra](#)).



ZigBee

En este apartado veremos los materiales necesarios para la realización de pruebas sobre la tecnología ZigBee.

XBEE XB24CZ7WIT – 004

Para probar la tecnología ZigBee será necesario que obtengamos dos dispositivos XBee. Para este estudio hemos seleccionado dos dispositivos de la versión 2.5. Se pueden obtener en la tienda online de Bricogeek ([Enlace de compra](#)).



XBee Explorer USB

Para poder establecer la conexión con el ordenador y transmitir los datos del software al módulo necesitaremos un adaptador USB. Este adaptador lo podremos encontrar en la tienda online de Bricogeek ([enlace de compra](#)).



XBee Explorer Regulated

Adicionalmente al adaptador XBee Explorer USB necesitaremos otro adaptador. Puede ser otro Explorer USB como el mencionado anteriormente o un XBee Explorer regulada que es más económico. Este adaptador se puede obtener en la tienda online de Bricogeek ([Enlace de compra](#)).



Cable alimentación XBee Explorer USB

Para poder establecer la conexión entre el adaptador y el ordenador además de transmitir la información del software de configuración al módulo necesitaremos un cable de conexión para el adaptador XBee Explorer USB. Este cable lo podemos encontrar en la tienda oficial de Amazon ([Enlace de compra](#)).



WI-FI

Materiales requeridos

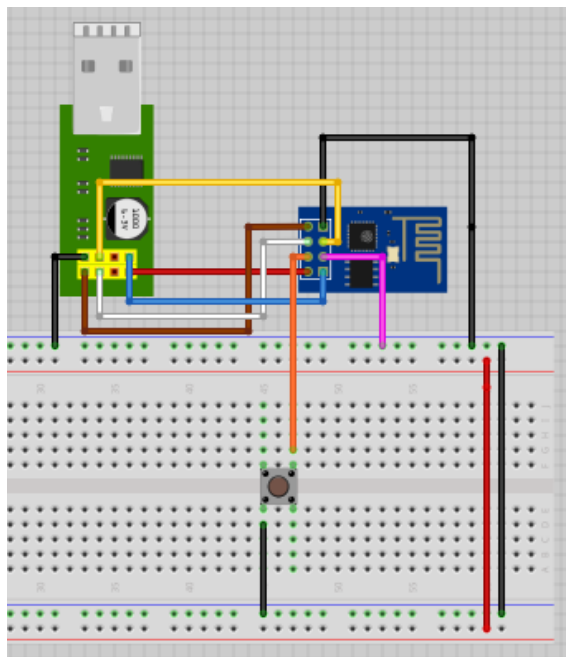
- 1x ESP8266 – 001
- 1x ESP8266 ADAPTER
- 1x KIT ARDUINO
 - Cables de conexiones
 - Protoboard

Configuración

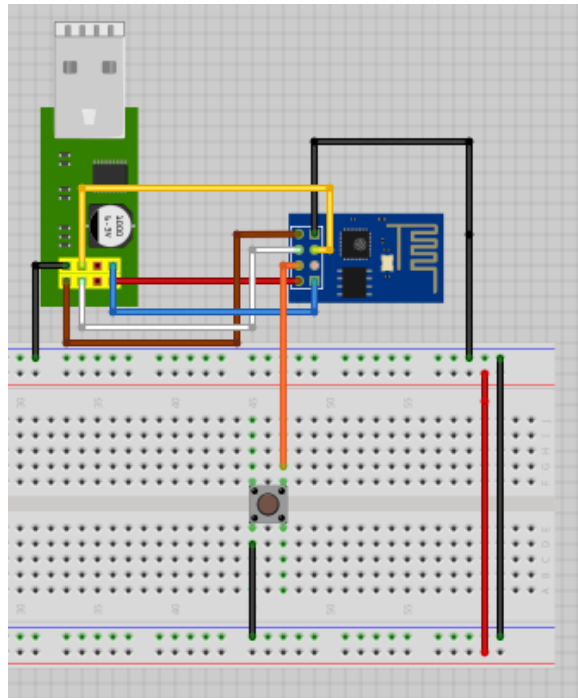
Conexión

Seguiremos los siguientes esquemas de conexiones para poder establecer la configuración desde el software.

Por una parte a la hora de cargar el programa utilizaremos la siguiente configuración que nos establece el modulo en modo UART.



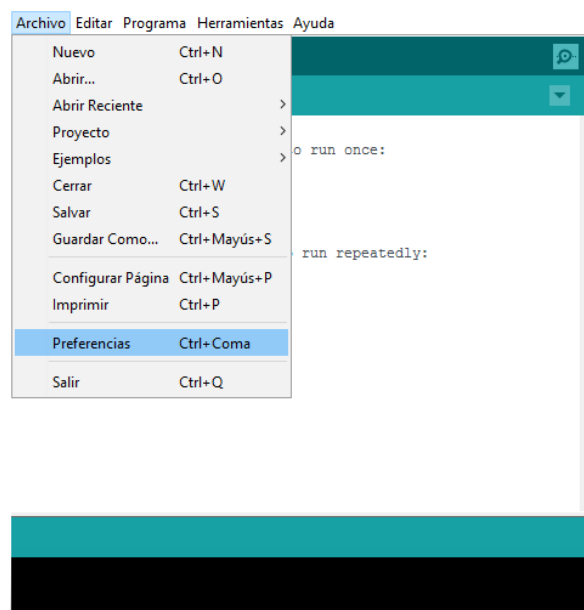
Para cambiar de modo UART a modo FLASH utilizaremos la siguiente configuración. Como veremos en el esquema la única diferencia que existe es que en el modo UART el pin GPIO0 está conectado a GND mientras que en modo FLASH este se encuentra sin conexión alguna. La conexión de este pin se encuentra representada por el cable rosa de los esquemas.



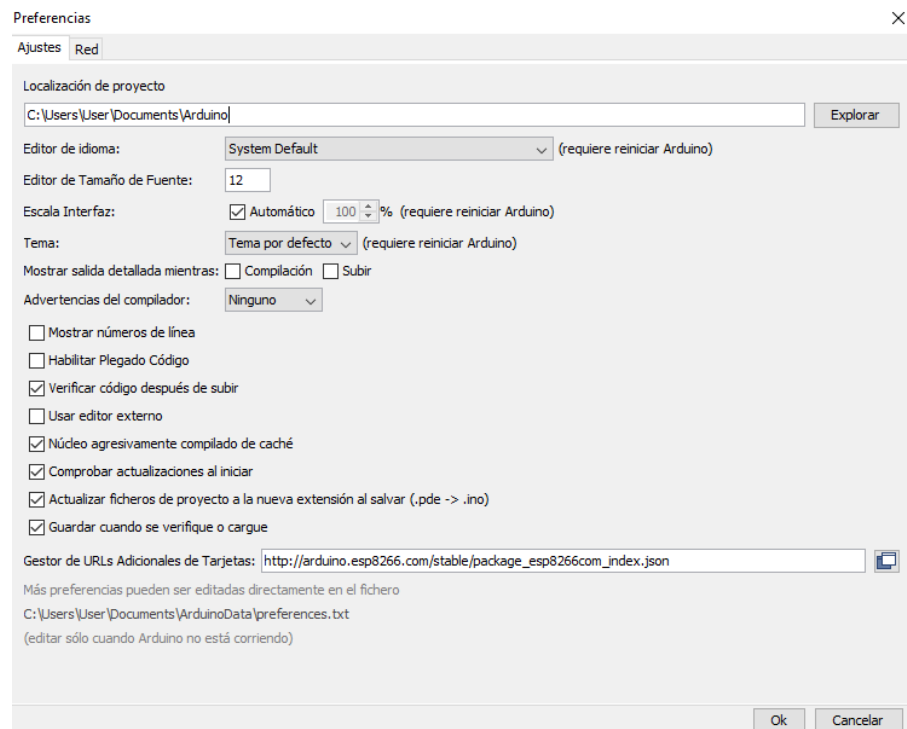
Software

En cuanto al software que utilizaremos será el de Arduino IDE para la configuración y PyCharm para las pruebas. En cuanto a la configuración de PyCharm lo mostraremos junto con el proceso de prueba.

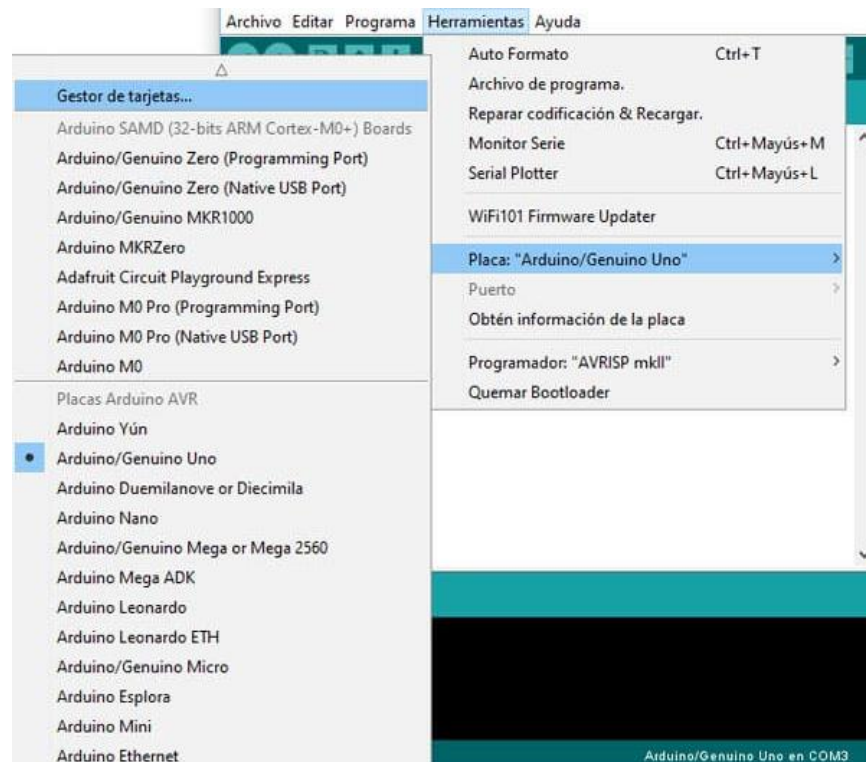
Para tener Arduino IDE configurado necesitaremos instalar una biblioteca específica que nos permita manejar este módulo. Primero clicaremos sobre Archivo y seleccionaremos la opción **Preferencias**.



Se nos abrirá una nueva ventana donde tendremos que ingresar la siguiente dirección URL : http://arduino.esp8266.com/stable/package_esp8266com_index.json



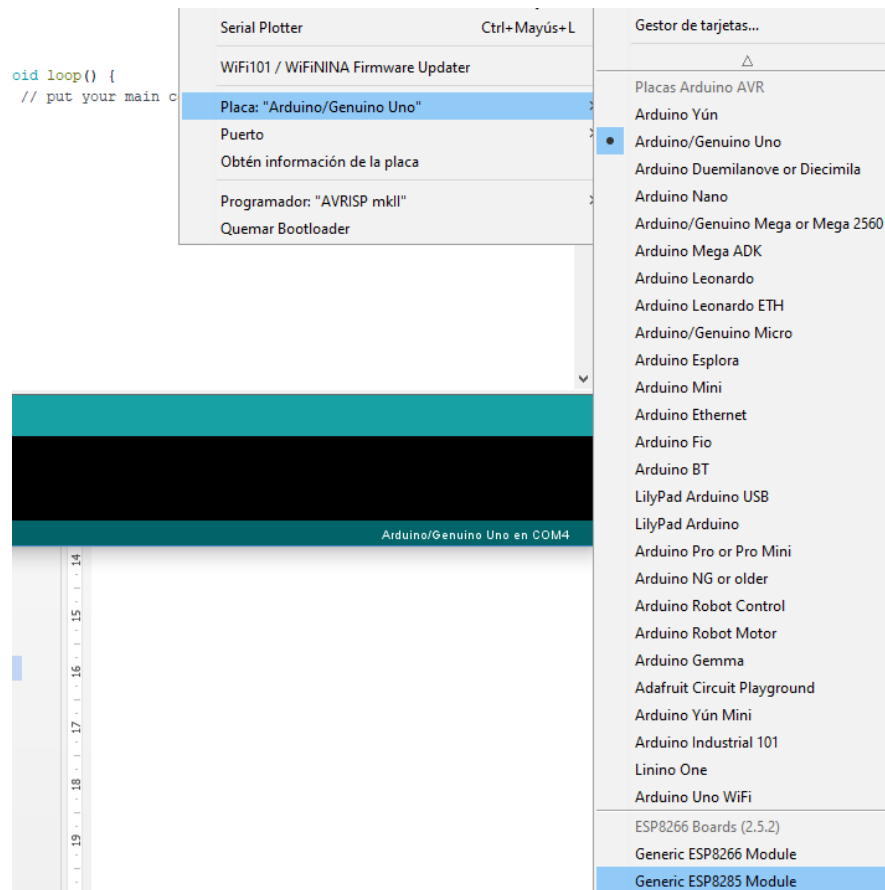
Tras esto clicaremos sobre el tab de herramientas y seleccionaremos dentro de las opciones de **Placa** la opción de **Gestor de tarjetas**.



Tras realizar esto se nos abrirá una nueva ventana donde tendremos que buscar el nombre de nuestro modulo 'esp8266' y posteriormente escogeremos la biblioteca ESP8266 Community y la instalamos.



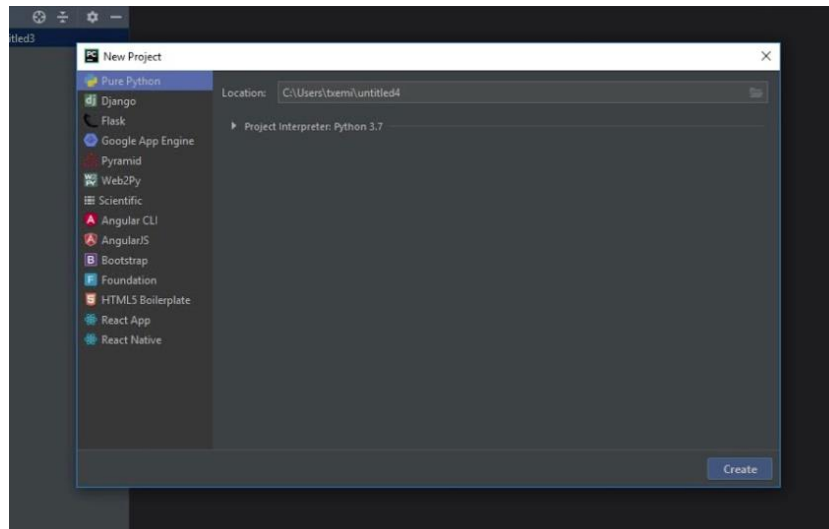
Una vez tengamos instalado este gestor seleccionaremos en la pestaña de herramientas como placa la siguiente:



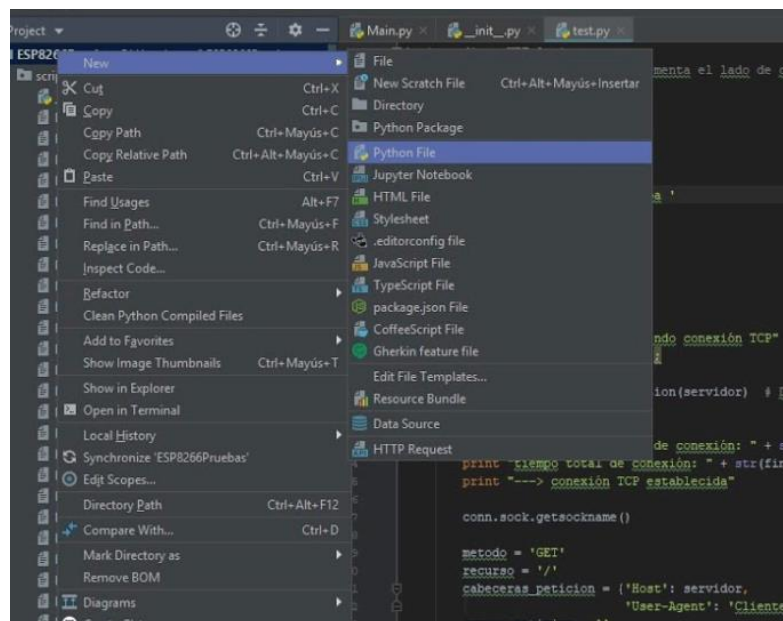
Llegados a este punto ya tenemos nuestro Arduino IDE configurado para poder trabajar con el módulo ESP8266.

Proceso de prueba

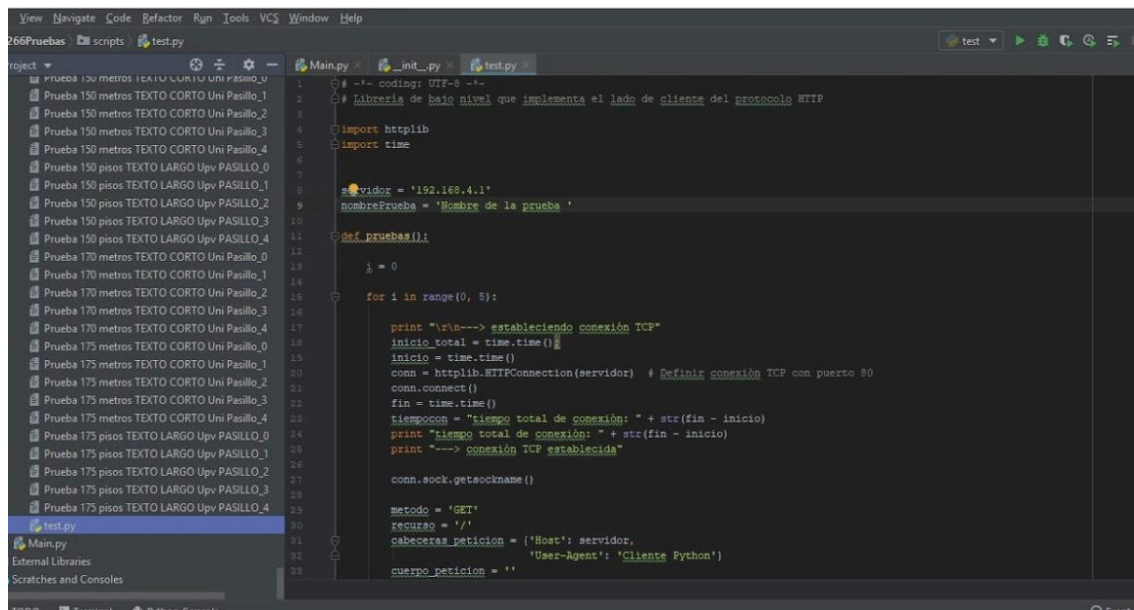
Para realizar las pruebas comenzaremos configurando nuestra herramienta que nos permita realizar el programa de prueba. En este caso utilizaremos PyCharm. Comenzaremos creando un nuevo proyecto de **Pure Python** clicando sobre el Tab **File** de nuestro entorno.



En este nuevo proyecto clicaremos con el botón derecho sobre la carpeta de proyecto, seleccionaremos **New** y posteriormente **Python File** para crear un nuevo archivo.



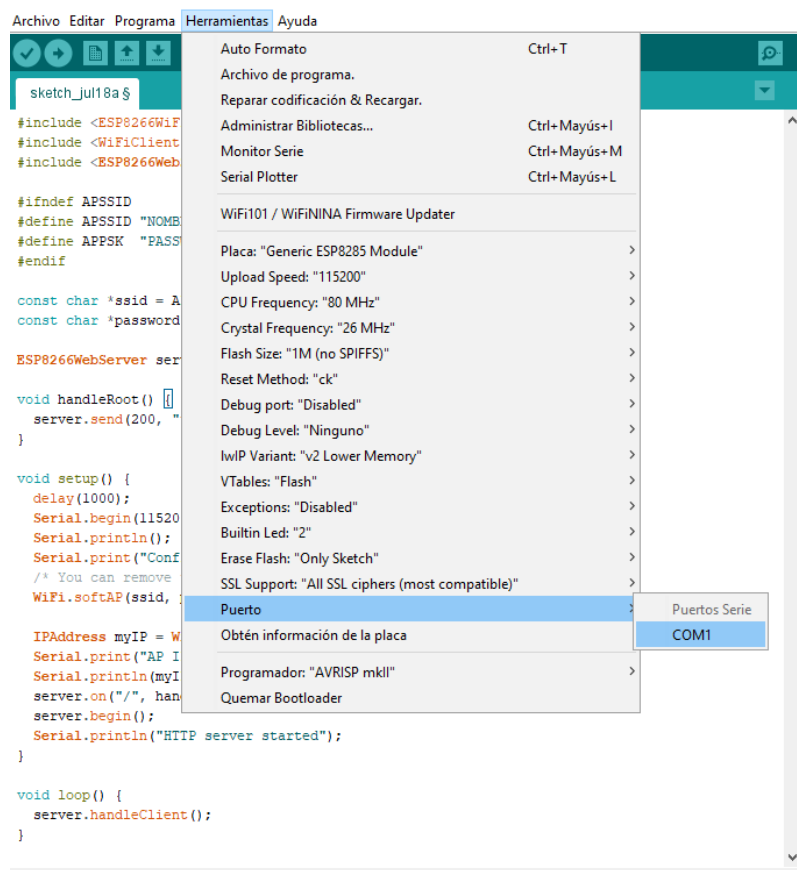
Introduciremos el código que podemos encontrar en el anexo 7 de la memoria de este estudio. Llamaremos a nuestro *Test.py*.



Una vez ya tenemos el código en nuestro archivo Python solo necesitaremos darle al icono de flecha verde situado arriba a la derecha para que se ejecute. Otra opción es clicar con el botón derecho sobre el archivo y seleccionar la opción **Run**.

Teniendo ya el entorno preparado para realizar las pruebas lo único que nos hará falta será configurar el modulo y ponerlo en modo ejecución para que ejecute el programa que hemos cargado.

1. Abrimos Arduino IDE y nos aseguramos que la placa que está seleccionada es la que corresponde con nuestro modulo como se menciona en la configuración inicial.
2. Copiamos en el Sketch el código que encontramos en el anexo número 6 de nuestro estudio y lo guardamos. Dependiendo del texto que queramos cargar en nuestro punto de acceso Wi-fi, deberemos sustituir el texto que se encuentra en la línea 16 del código entre etiquetas <h1>INTRODUCE TU TEXTO</h1> por nuestro texto.
3. Conectamos a nuestro ordenador el adaptador del módulo con las conexiones realizadas como se ven anteriormente en el apartado de conexiones. Utilizaremos el módulo en modo UART.
4. En la pestaña de **Herramientas** indicamos el puerto en el que se encuentra conectado nuestro adaptador junto con el módulo. En este punto deberíamos tener el Arduino de la siguiente manera. En caso de que la configuración no se parezca a la mostrada en la siguiente imagen habrá que comprobar que hayamos realizado de forma correcta lo descrito en el apartado Software de configuración y avanzar paso a paso hasta este punto.



5. Una vez tengamos el Arduino IDE configurado de manera correcta y el adaptador conectado a nuestro ordenador podremos compilar y subir el código al módulo. Para realizar esto último bastaría con clicar en la flecha con dirección hacia la derecha que se encuentra en la esquina superior izquierda.
6. Cuando el proceso de subida del Sketch al módulo haya finalizado, lo desconectaremos del ordenador.
7. Cambiaremos las conexiones de modo UART a modo FLASH para que el módulo ejecute el Sketch y lo volveremos a conectar.
8. En este momento ya tenemos el módulo emitiendo una señal Wi-fi y en modo punto de acceso.
9. Nos conectaremos a esta señal Wi-fi desde el ordenador donde procederemos a ejecutar nuestro cliente Python.
10. Por ultimo ejecutaremos el código Python que hemos preparado antes y observaremos como nos crea 5 archivos por ejecución, en la carpeta donde tenemos el código ejecutado, con el nombre que hayamos establecido en la línea 9 del código. Dentro de estos archivos encontraremos los resultados de las pruebas (tal y como podemos ver en los anexos 9, 10 y 11 con el STATUS, el tiempo de conexión y el tiempo de request además del mensaje)

```

1 contenido: <h1>l</h1>
2 Status: 200
3 tiempo total de conexión: 0.0019998550415
4 tiempo total de request: 0.000999927520752

```

Esta imagen anterior es un ejemplo de uno de los archivos generados por nuestro cliente Python. Ya con los resultados obtenidos lo siguiente es realizar los cálculos que se necesiten en función de los criterios que se quieran evaluar.

BLUETOOTH

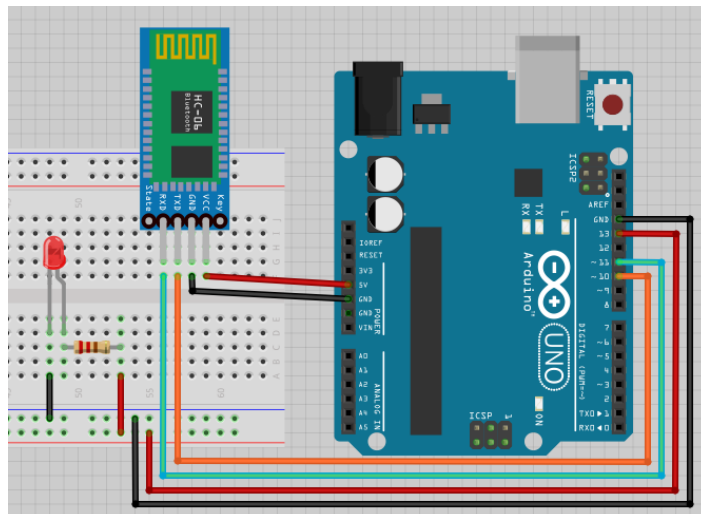
Materiales requeridos

- 1x BLE HC - 08
- 1x Arduino UNO
- 1x KIT ARDUINO
 - Cables de conexiones
 - Protoboard (Opcional)
 - Led (Opcional)
- 1x Cable conexión Arduino - PC

Configuración

Conexión

Para todas las pruebas utilizaremos la conexión mostrada a continuación. El apartado del led no es necesario para las pruebas pero es útil para ver si recibe de manera correcta el dato en las pruebas de control. Para que resulte más fácil y genérico para los tres tipos de pruebas se explicara sin la opción del led.

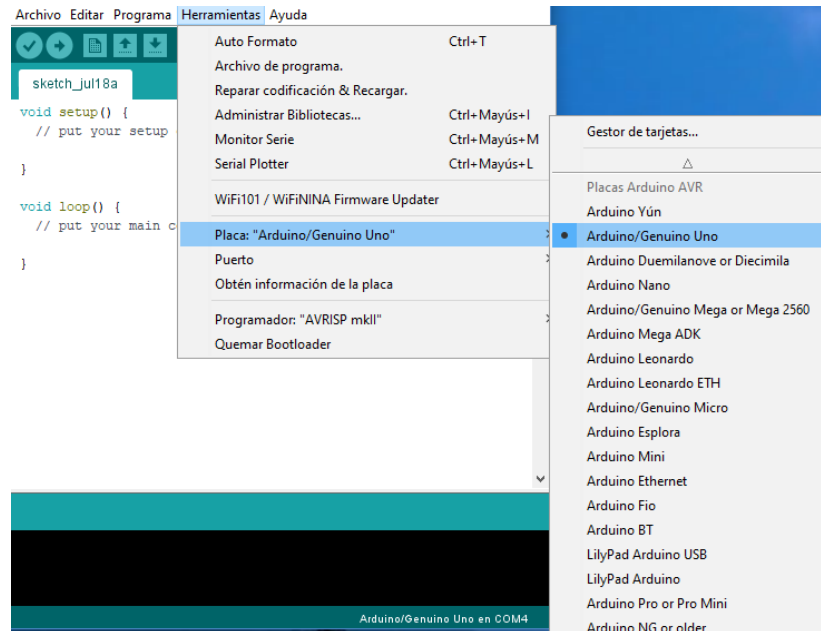


El pin TDX del módulo irá conectado al pin 10 del Arduino y el RDX del módulo al pin 11. Respecto a la alimentación, conectaremos el modulo al pin de 5V del Arduino.

Software

Para realizar estas pruebas utilizaremos Arduino IDE y una aplicación móvil disponible en el Google Play de Android. Esta aplicación se llama *Serial Bluetooth Terminal* de Kai Morich y se puede encontrar en el siguiente [enlace](#).

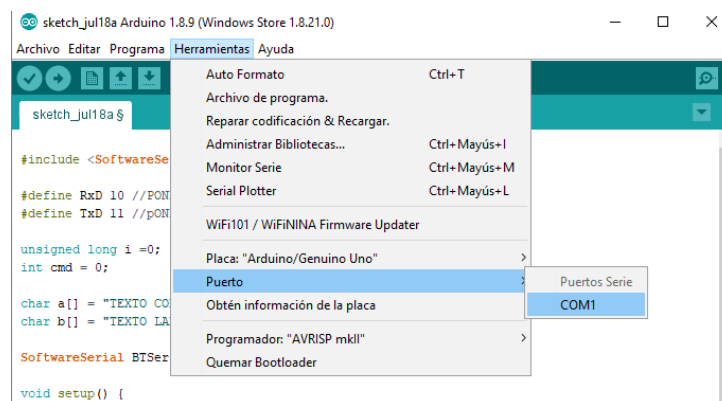
Respecto a la configuración del Arduino IDE lo único necesario es en la pestaña de **Herramientas** seleccionar la placa **Arduino/Genuino Uno**.



Antes de realizar ninguna prueba, de manera opcional se podrá utilizar el código del anexo 4 de la documentación cargado en el Arduino para configurar nuestro modulo Bluetooth mediante comandos AT, si se le quiere dar un nombre específico o tocar alguna que otra característica.

Proceso de prueba

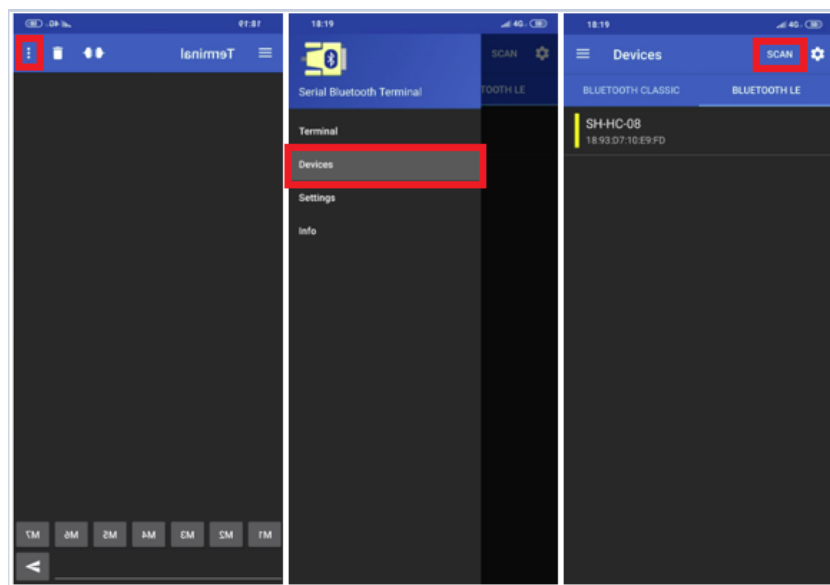
Una vez tenemos las conexiones realizadas y el Arduino como se ha mencionado en el apartado anterior iniciaremos Arduino IDE y crearemos un nuevo archivo en el que pegaremos el código que encontramos en el anexo 5 de la documentación de este proyecto. Una vez copiado conectamos nuestro Arduino al ordenador y seleccionamos el puerto en el que se encuentra.



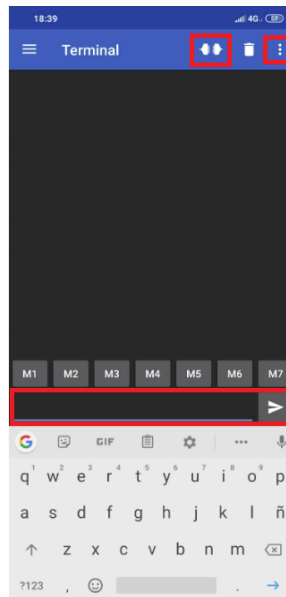
Procederemos a compilar y subir el código a nuestro Arduino clicando en la flecha con dirección a la derecha que se encuentra en la parte superior izquierda.

Si todo ha ido correctamente tendremos una luz roja en el módulo en estado de parpadeo constante. Una vez conectado nuestro dispositivo de prueba esta luz se volverá fija.

Procedemos a iniciar en nuestro dispositivo Android en el que realizaremos las pruebas la app previamente instalada *Serial Bluetooth Terminal*. Para poder conectarnos al módulo habilitaremos las conexiones Bluetooth de nuestro dispositivo. En la interfaz de la app haremos click sobre el menú, seleccionaremos **Devices** y escanaremos el entorno hasta que veamos nuestro dispositivo. Por defecto viene con el nombre que veremos en el flujo de imágenes.

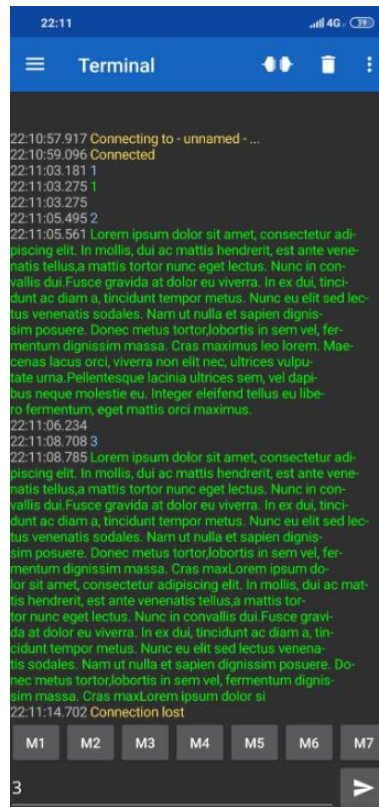


Una vez encontrado nuestro dispositivo vamos al menú de nuevo y seleccionamos **Terminal**. En esta pantalla tendremos la barra para escribir nuestro mensaje, el botón arriba a la derecha para establecer la conexión con nuestro dispositivo y el botón de la esquina superior derecha para poder guardar los datos obtenidos o configurar la app.



En este momento nos encontramos con el modulo configurado y conectado a la espera de establecer una conexión y con el dispositivo con el que vamos a realizar las pruebas con la app iniciada y configurada. Lo único que nos queda es establecer la conexión dándole al botón situado a la izquierda de la papelera en la esquina superior derecha.

Una vez establecida la conexión, tal y como está programado el código, lo único que tenemos que hacer es desde la app enviar un mensaje para solicitar una transferencia ya sea de control, texto corto o texto largo. Para que el modulo nos responda con un mensaje de control enviaremos desde el dispositivo un '1', para solicitar el texto corto enviaremos un '2' y para solicitar un texto equivalente en bytes al largo enviaremos un '3'. En el dispositivo recibiremos los mensajes de la siguiente manera. Podremos ver los tiempos de envío y los datos recibidos para evaluar la transmisión.



ZIGBEE

Materiales requeridos

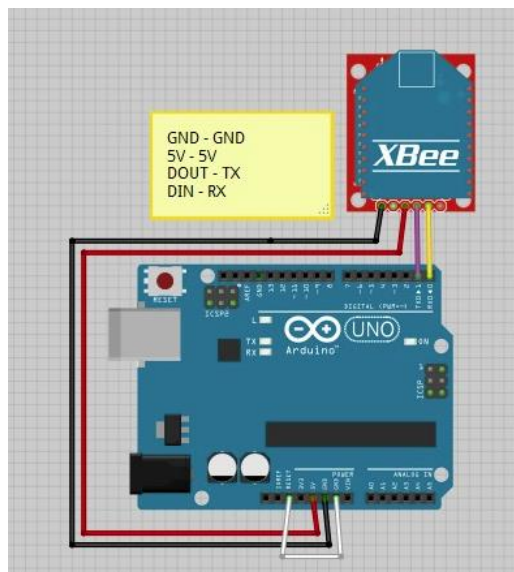
- 2x XBEE XB24CZ7WIT – 004
- 1x XBEE Explorer USB
- 1x XBEE Explorer Regulada
- 1x Arduino UNO o Fuente de alimentación externa
- 1x Cable de conexión USB XBee Explorer USB

Configuración

Conexión

Para realizar las configuraciones de los módulos utilizaremos el adaptador XBee Explorer USB y bastara con conectar encima el modulo y conectar la salida usb al ordenador con el cable de conexión.

Para conectar el modulo a una fuente de alimentación externa podemos utilizar el propio Arduino ya que nos da la potencia suficiente. Para ello conectaremos el modulo XBee a la placa XBee Explorer reuglada y esta al Arduino siguiendo el siguiente esquema.



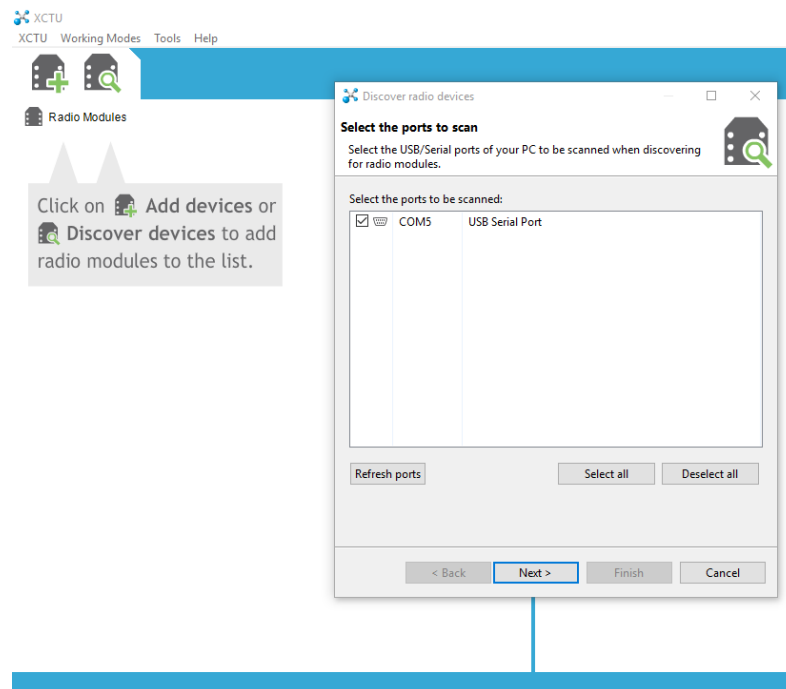
Software

Como software para las pruebas utilizaremos el XCTU y no habrá que realizar ninguna configuración específica de este.

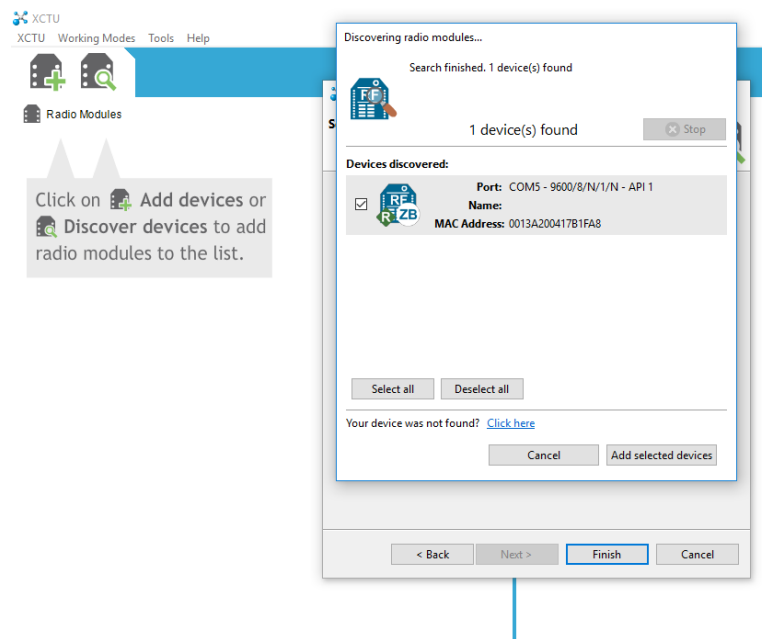
Proceso de prueba

Para realizar las pruebas estableceremos una conexión entre dispositivos punto a punto de manera que tendremos un módulo que actuará como router y otro que hará la función de coordinador. Realizaremos la configuración de estos módulos de la siguiente manera.

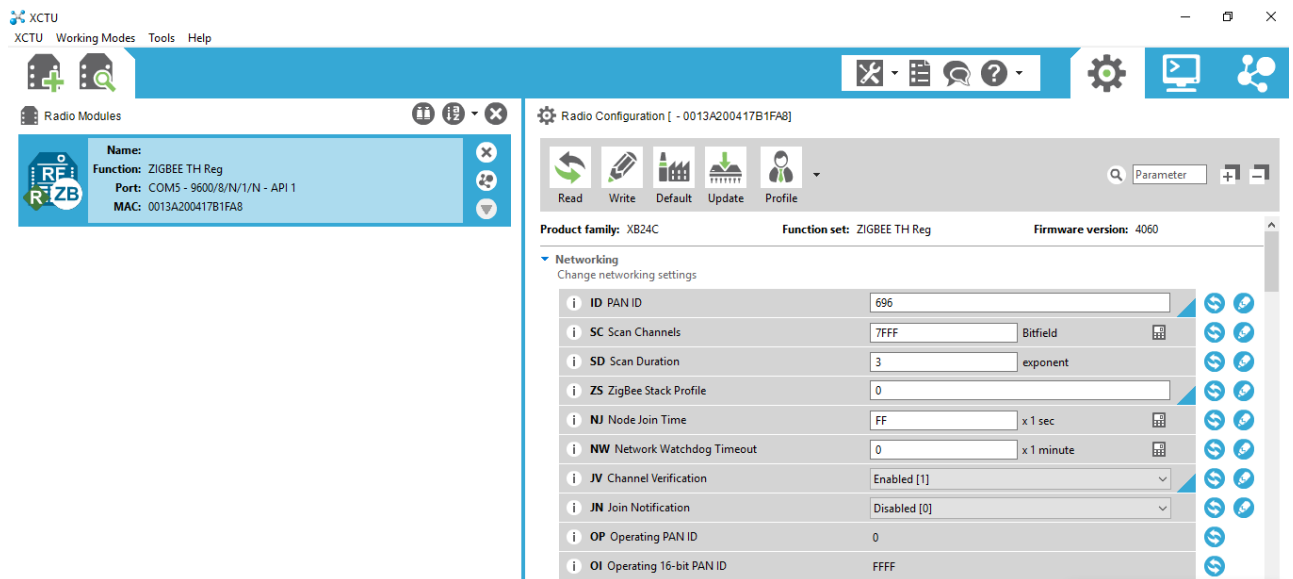
1. Iniciamos el programa.
2. Hacemos click en el icono de **Discover devices** como nos indican las instrucciones del mismo programa.



3. Seguimos el flujo marcado hasta que vemos nuestro dispositivo encontrado y clicaremos sobre **Add selected devices**.



4. En este momento se vinculará el dispositivo y se cargarán sus opciones como veremos en la siguiente imagen.



5. Hasta este paso se ha de realizar con los dos dispositivos de la misma manera. Es en este instante cuando configuraremos cada uno de forma distinta. Buscaremos los parámetros indicados y estableceremos los siguientes valores.

a. XBee-1 Coordinador

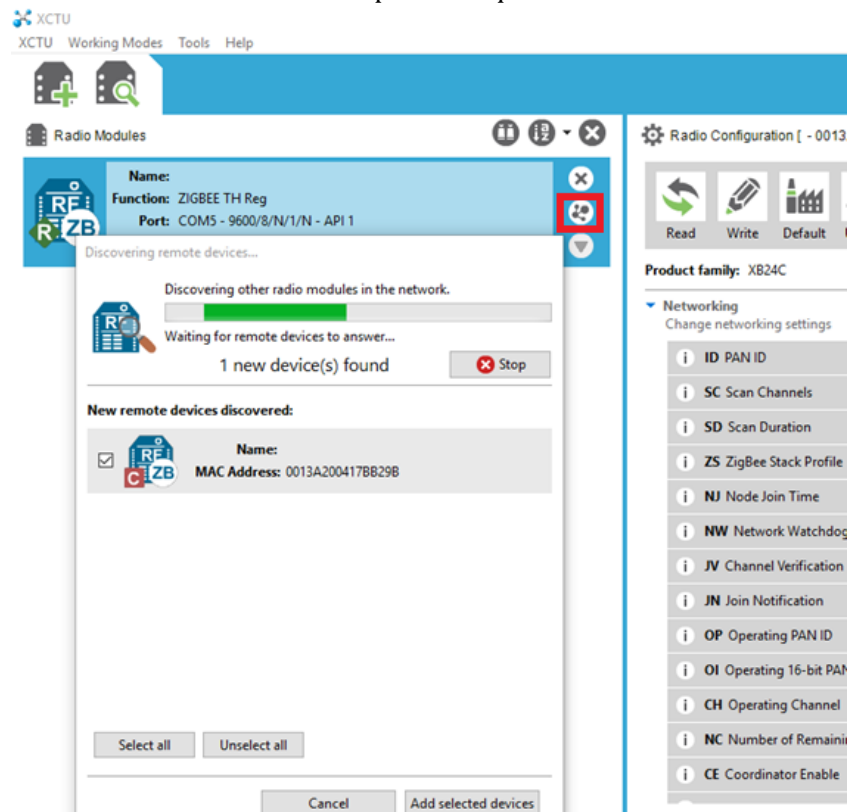
ID PAN ID	696
AP API Enable	API enabled [1]
CE Coordinator Enable	Enabled [1]

b. XBee-2 Router

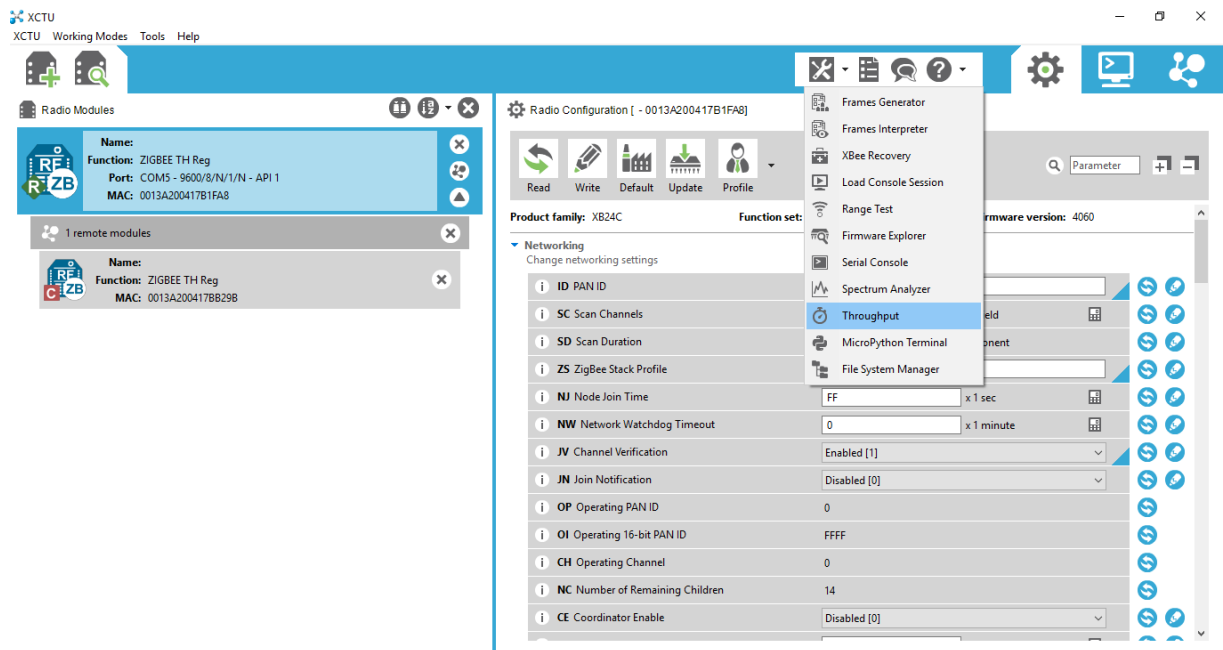
ID PAN ID	696
JV Channel Verification	Enabled [1]
DH Destination Address High	0
DL Destination Address Low	0

6. Una vez tenemos configuradas las opciones guardamos los cambios (daremos al botón con el icono de un lápiz que pone write) y conectamos el dispositivo que actúa de router al ordenador.

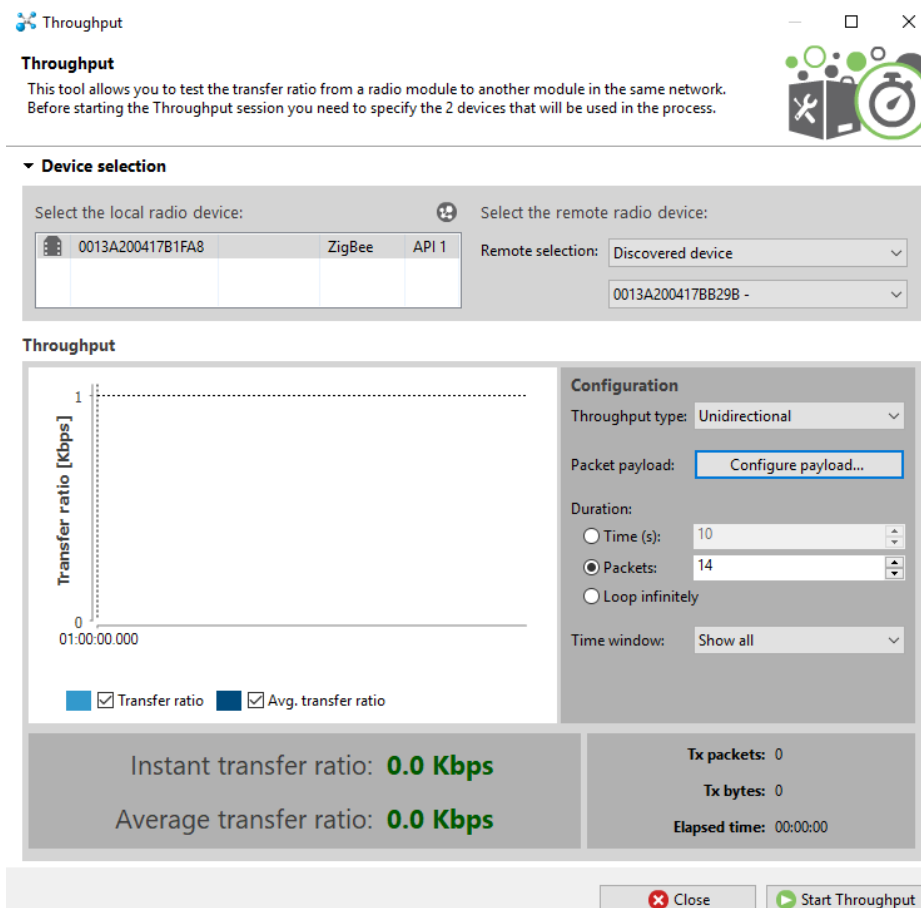
7. Ahora con el router y dando al botón indicado en la imagen a continuación, haremos una lectura de los dispositivos que se encuentran en la misma red.



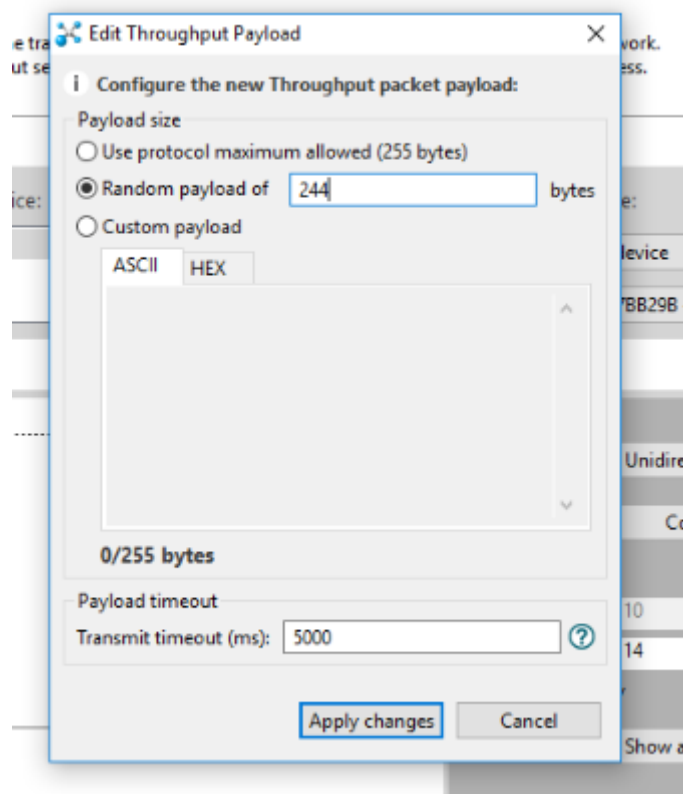
8. Como podemos observar, esto escaneará la red que hemos establecido en configuración en el atributo de PAN ID y nos conectara con los dispositivos que tengan el mismo valor en ese campo. En nuestro caso veremos el coordinador que hemos configurado y clicaremos sobre el botón de **Add selected devices**.
9. Una vez hayamos tengamos en pantalla el dispositivo coordinador asignado al router, utilizaremos una herramienta que nos proporciona el mismo software XCTU para realizar mediciones de velocidad de transmisión y calidad de transmisión. Para ello haremos clic sobre el botón de herramientas situado en la mitad derecha del tab superior. Se nos desplegarán una serie de herramientas. La que vamos a usar para realizar las pruebas es el Throughput, así que seleccionaremos esta opción. (Más adelante en el apartado Proceso de análisis de espectro explicaremos como utilizar la herramienta Spectrum Analyzer.



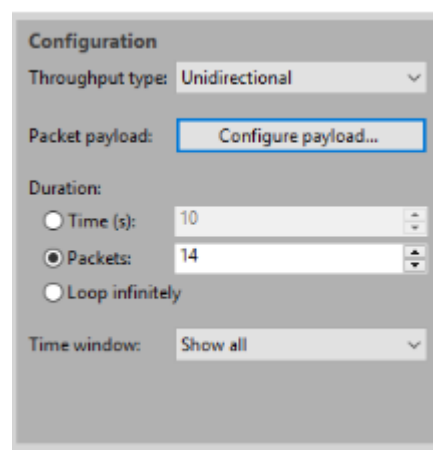
10. Sobre la pantalla se desplegará una nueva ventana para configurar e iniciar la prueba. En esta ventana estableceremos la siguiente configuración. Seleccionaremos el dispositivo que nos sale en la lista como dispositivo local y como dispositivo remoto clicaremos sobre el botón de picklist que tenemos en la derecha y seleccionaremos nuestro coordinador.



11. Una vez con los dispositivos seleccionados clicaremos sobre el botón **configure payload**. Este botón nos abrirá la siguiente ventana donde configuraremos la cantidad de bytes a transmitir y el tiempo de espera de respuesta que queremos.

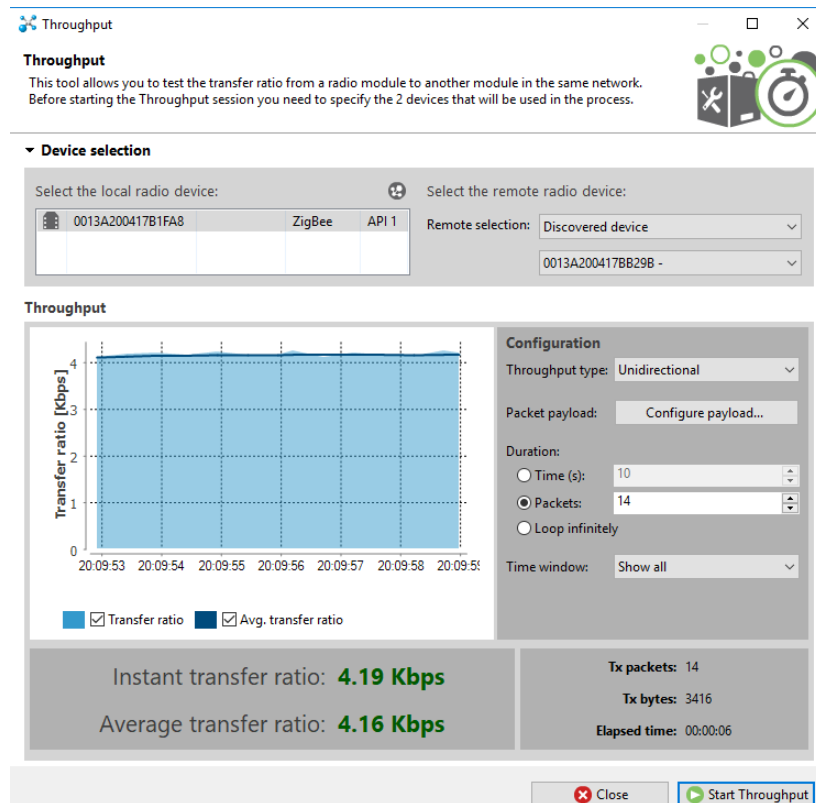


12. Como vamos a transmitir un volumen de 3416 Bytes, que sería lo correspondiente al texto largo, establecemos paquetes de 244 Bytes ya que el máximo es 255. Cuando apliquemos los cambios estableceremos en el recuadro de la configuración de la prueba una duración por paquetes de 14. Esto quiere decir que se transmitirán 14 paquetes de 244 Bytes cada uno, es decir un total de 3416 Bytes que es lo que estamos buscando.



13. El último paso es clicar sobre el botón de la esquina inferior derecha que lanzara la ejecución de esta prueba. Observaremos los resultados y realizaremos los cálculos que sean necesarios para evaluar los criterios que se deseen. El resultado de esta prueba de ejemplo se vería de la siguiente forma. Por una parte tenemos la gráfica donde podremos ver reflejados el estado de la transmisión durante la prueba. Más abajo tendremos la

velocidad de transferencia actual y la media, y por ultimo justo a la derecha podemos ver los paquetes lo bytes enviados correctamente y el tiempo.

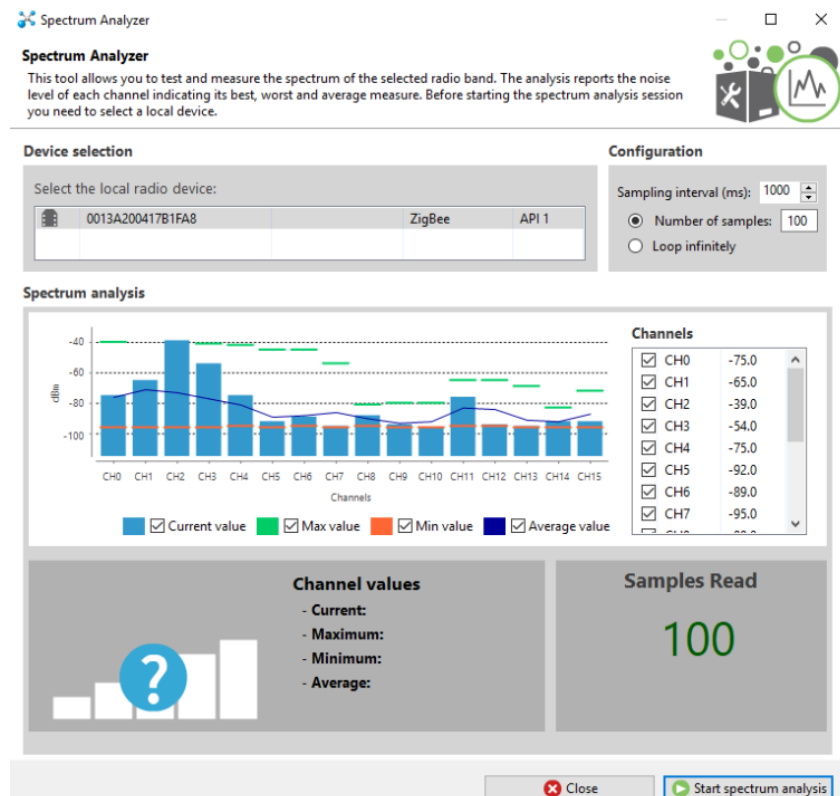


Proceso de análisis de espectro

Partiendo del paso 9 del anterior apartado clicaremos sobre la herramienta **Spectrum Analyzer**. Esta herramienta nos permitirá realizar un análisis del entorno donde podremos ver como de saturados se encuentran los canales de la banda de 2,4 GHz.

Esta herramienta es bastante interesante y muy sencilla de utilizar. Basta con seleccionar el dispositivo ZigBee local, el intervalo entre cada prueba y el número de pruebas que se quieren realizar.

Una vez configurado a nuestro gusto pincharíamos sobre el botón de la esquina inferior derecha para empezar y tras un rato obtendríamos unos resultados como los siguientes.



Aquí veremos representados en una gráfica de barras los canales con sus valores mínimos, máximos y medios de ruido.

RF

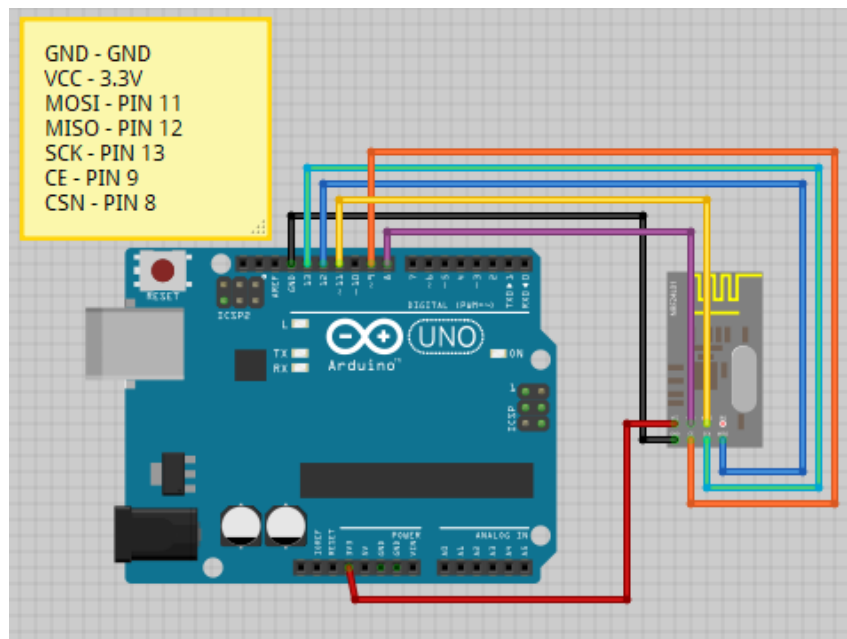
Materiales requeridos

- 2x NRF24L01
- 2x Arduino UNO
- 1x KIT ARDUINO
 - Cables de conexiones
 - Fuente de alimentación (Opcional).
- 2x Cable conexión Arduino - PC

Configuración

Conexión

Para realizar las pruebas de RF utilizaremos el siguiente esquema de conexiones tanto para el emisor como para el receptor. Adicionalmente si se desea se puede alimentar el modulo utilizando una fuente de alimentación externa.

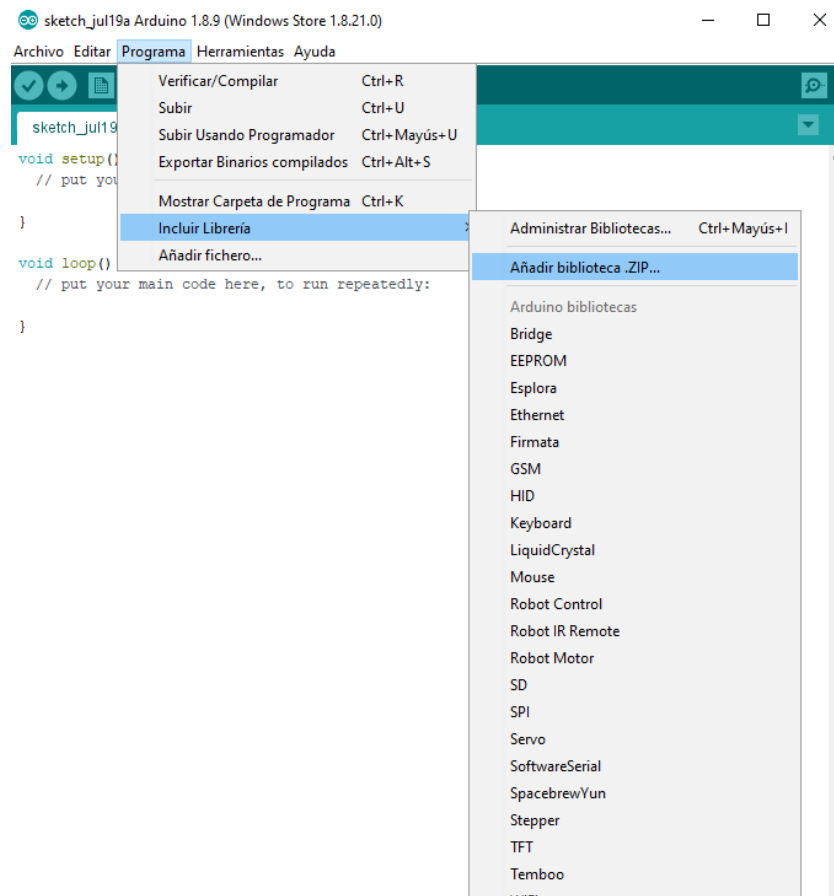


Para aplicar la fuente de alimentación externa sobre este esquema bastaría con conectar las tierras del Arduino, modulo y fuente de alimentación entre si y alimentar al módulo directamente con el voltaje externo.

Software

Para realizar estas pruebas utilizaremos Arduino IDE. Por una parte será necesario que en la pestaña de **Herramientas** seleccionemos la placa

Arduino/Genuino Uno. Además de esto deberemos instalar la última versión de la librería RF24 que podremos encontrar [aquí](#). El proceso a seguir es; descargar la librería, ir a nuestro Arduino IDE e ir a la pestaña de **Programa > Incluir Librería > Añadir biblioteca .ZIP**.

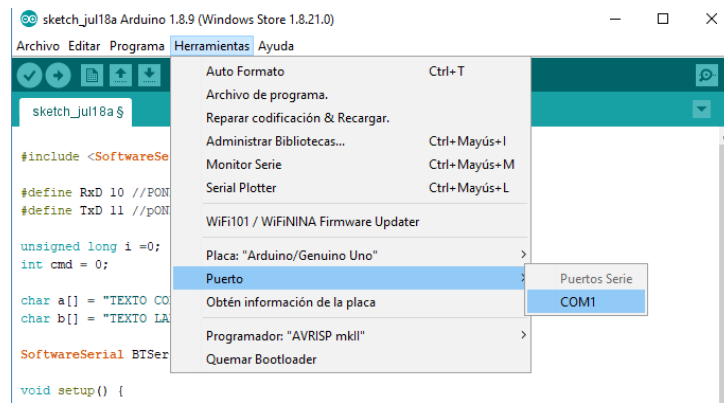


Seleccionamos esta última opción, buscamos nuestro archivo comprimido descargado de RF24 y lo seleccionamos.

Proceso de prueba

Una vez configurado todo realizamos las conexiones indicadas para ambos módulos cada uno con un Arduino diferente. Utilizando dos ordenadores con Arduino IDE instalado completaremos los programas tanto para emisor como para receptor con los códigos de los anexos 2 y 3 de la documentación. En caso de que queramos comprobar que los módulos están bien conectados utilizaremos el código del anexo 1 que nos devolverá los datos de cada módulo.

Una vez tenemos los programas completos procederemos a conectar los Arduinos a nuestros ordenadores, seleccionar los puertos correspondientes y compilar y subir los programas.



Tras cargar los programas abriremos el monitor serial en cada uno de los dispositivos clicando en la lupa de la parte superior derecha como se indica en la siguiente imagen.



Nuestro programa está preparado para que realice en una misma ejecución 5 envíos de control, 1 envío de texto corto y 1 envío de bytes correspondientes al texto largo definido en la documentación de este trabajo. Este trabajo lo realizara 5 veces ya que se encuentra dentro de un bucle. En el monitor serial podremos ver los tiempos en los que se envía y se reciben los mensajes, el número de paquetes y el contenido del mensaje recibido. Dado que el protocolo que utilizan estos módulos tiene como capacidad máxima 32 bytes los mensajes son divididos y enviados en paquetes de esta dimensión.

Tras realizar la ejecución tendremos en un ordenador lo enviado y en otro lo recibido por el modulo receptor.

Deberíamos ver lo siguiente en los monitores tanto por parte del emisor como por parte del receptor.

campo abierto ciudad emisor 100.txt		campo abierto ciudad 100 m.txt	
1	21:12:55.413 -> 1	1	21:12:55.579 -> 1148
2	21:12:55.413 -> 1	2	21:12:55.579 -> 1
3	21:12:55.413 -> 1	3	21:12:55.641 -> 1149
4	21:12:55.413 -> 1	4	21:12:55.641 -> 1
5	21:12:55.413 -> 1	5	21:12:55.641 -> 1150
6	21:12:55.413 -> Envio de paquetes: 5	6	21:12:55.641 -> 1
7	21:12:55.449 -> Lorem ipsum dolor sit amet, consl	7	21:12:55.641 -> 1151
8	21:12:55.499 -> ctetur adipiscing elit. In mollil	8	21:12:55.641 -> 1
9	21:12:55.532 -> , dui ac mattis hendrerit, est al	9	21:12:55.641 -> 1152
10	21:12:55.565 -> te venenatis tellus,a mattis torl	10	21:12:55.641 -> 1
11	21:12:55.616 -> or nunc eget lectus. Nunc in conl	11	21:12:55.641 -> 1153
12	21:12:55.616 -> allis dui. Fusce gravida at dolol	12	21:12:55.641 -> Lorem ipsum dolor sit amet, cons
13	21:12:55.666 -> eu viverra. In ex dui, tincidunt	13	21:12:55.675 -> 1154
14	21:12:55.716 -> ac diam a, tincidunt tempor metl	14	21:12:55.675 -> ctetur adipiscing elit. In molli
15	21:12:55.765 -> s. Nunc eu elit sed lectus venenl	15	21:12:55.712 -> 1155
16	21:12:55.765 -> tis sodales.Nam ut nulla et sapil	16	21:12:55.712 -> , dui ac mattis hendrerit, est a
17	21:12:55.815 -> n dignissim posuere. Donec metusl	17	21:12:55.779 -> 1156
18	21:12:55.848 -> tortor, lobortis in sem vel, ferl	18	21:12:55.779 -> te venenatis tellus,a mattis tor
19	21:12:55.882 -> entum dignissim massa. Cras maxil	19	21:12:55.813 -> 1157
20	21:12:55.915 -> us leo lorem.Maecenas lacus orcil	20	21:12:55.813 -> or nunc eget lectus. Nunc in con
21	21:12:55.966 -> viverra non elit nec, ultrices 1	21	21:12:55.863 -> 1158
22	21:12:56.015 -> ulputate urna.Pellentesque lacinl	22	21:12:55.863 -> allis dui. Fusce gravida at dolo
23	21:12:56.015 -> a ultrices sem, vel dapibus nequ	23	21:12:55.895 -> 1159
24	21:12:56.065 -> molestie. Integer eleifend telll	24	21:12:55.895 -> eu viverra. In ex dui, tincidunt
25	21:12:56.115 -> s eu libero fermentum, eget mattl	25	21:12:55.929 -> 1160
26	21:12:56.148 -> s orci maximus.	26	21:12:55.929 -> ac diam a, tincidunt tempor met
27	21:12:56.148 -> Envio de paquetes: 20	27	21:12:55.982 -> 1161
28	21:12:56.182 -> Lorem ipsum dolor sit amet, consl	28	21:12:55.982 -> s. Nunc eu elit sed lectus venen
29	21:12:56.231 -> ctetur adipiscing elit. Etiam pol	29	21:12:56.042 -> 1162
30	21:12:56.265 -> uere orci et eros varius, sit aml	30	21:12:56.042 -> tis sodales.Nam ut nulla et sapi
31	21:12:56.298 -> t imperdiet sem mollis.	31	21:12:56.042 -> 1163
32	21:12:56.298 -> Lorem ipsum dolor sit amet, consl	32	21:12:56.079 -> n dignissim posuere. Donec metus
33	21:12:56.349 -> ctetur adipiscing elit. Etiam pol	33	21:12:56.079 -> 1164
34	21:12:56.400 -> uere orci et eros varius, sit aml	34	21:12:56.142 -> tortor, lobortis in sem vel, fer
35	21:12:56.449 -> t imperdiet sem mollis.	35	21:12:56.142 -> 1165
36	21:12:56.449 -> Lorem ipsum dolor sit amet, consl	36	21:12:56.142 -> entum dignissim massa. Cras maxi
37	21:12:56.498 -> ctetur adipiscing elit. Etiam pol	37	21:12:56.178 -> 1166
38	21:12:56.547 -> uere orci et eros varius, sit aml	38	21:12:56.178 -> us leo lorem.Maecenas lacus orci
39	21:12:56.547 -> t imperdiet sem mollis.	39	21:12:56.212 -> 1167
40	21:12:56.581 -> Lorem ipsum dolor sit amet, consl	40	21:12:56.212 -> viverra non elit nec, ultrices
41	21:12:56.615 -> ctetur adipiscing elit. Etiam pol	41	21:12:56.269 -> 1168
42	21:12:56.648 -> uere orci et eros varius, sit aml	42	21:12:56.269 -> ulputate urna.Pellentesque lacin
43	21:12:56.682 -> t imperdiet sem mollis.	43	21:12:56.313 -> 1169
44	21:12:56.732 -> Lorem ipsum dolor sit amet, consl	44	21:12:56.313 -> a ultrices sem, vel dapibus nequ
		45	21:12:56.362 -> 1170
		46	21:12:56.362 -> molestie. Integer eleifend tell
		47	21:12:56.412 -> 1171
		48	21:12:56.412 -> s eu libero fermentum, eget matt
		49	21:12:56.412 -> 1172