

GRADO EN ELECTRÓNICA INDUSTRIAL Y AUTOMÁTICA

TRABAJO FIN DE GRADO

SISTEMA DE COMUNICACIÓN XBEE PARA UNA RED DE SENSORES

DOCUMENTO 1 - MEMORIA

Alumno/Alumna: Estévez, Santé, José Javier

Director/Directora: Oleagordia, Aguirre, Iñigo Javier

Curso: 2017-2018

Fecha: 23/07/2018

Índice

1	INTRODUCCIÓN.....	5
2	OBJETIVOS.....	6
3	TECNOLOGIA INALAMBRICA. ZIGBEE	7
4	HARDWARE DEL PROYECTO	11
4.1	MODULOS ZIGBIT 2,4GHz CON DOBLE CHIP ANTENA	12
4.1.1	Características.....	12
4.1.2	Diagrama de bloques.....	13
4.1.3	Evaluación de pines que se utilizan del módulo Zigbit.....	14
4.2	Xbee ZNet 2.5 RF Module.....	15
4.3	Placa Xbee Explorer USB.....	20
4.4	XBee SIP Adapter.....	21
4.4.1	Características.....	21
4.5	Bateria.....	21
4.6	Arduino uno.....	22
5	SOFTWARE DEL PROYECTO	23
5.1	IDE de Arduino.....	23
5.1.1	Programacion Leer un dato análogo de un XBee remoto en modo API.....	24
5.2	Programación de los módulos Xbee: X-CTU	25
5.3	PROTEUS.....	27
6	PLANOS	29
6.1	Coordinador o Router	29
6.2	Punto Final.....	30
7	LISTADO COMPLETO DE MATERIALES	32
8	PRECIOS UNITARIOS DE MATERIALES	33
9	PRESUPUESTO	34
10	MEJORAS PLANIFICADAS A FUTURO.....	35
11	BIBLIGRAFIA.....	36

Índice de figuras

Figura 1.1.1. Estructura de una red inalámbrica de sensores	5
Figura 2.1.1. Algunos módulos ZigBee	11
Figura 4.1.2. Módulo ZigBee.....	12
Figura 4.1.3. Diagrama de bloques.....	13
Figura 2.1.4. configuración de pines.....	14
Figura 4.2.1. Módulo Xbee ZNet 2.5 RF Series2 de Digi	16
Figura 4.2.2. Conexiones mínimas requeridas para el Xbee.	16
Figura 4.2.3. Ejemplo comando AT	17
Figura 4.2.4. Ejemplo de trama.....	18
Figura 4.2.5. Modos de funcionamiento del módulo Xbee.....	19
Figura 4.3.1. Xbee Explorer USB.....	20
Figura 4.3.2. Conexión del módulo Xbee al Explorer USB.....	20
Figura 4.4.1. USB XBee SIP Adapter.....	21
Figura 4.6.1. Disposición Placa Arduino Uno.....	22
Figura 5.1.1. IDE Arduino.....	24
Figura 5.2.1. Selección del dispositivo conectado	25
Figura 5.2.2. Panel configuración parámetros XBee.....	26
Figura 5.3. Esquema de los nodos sensores.....	28
Figura 6.1.1. Esquema Coordinador.....	29
Figura 6.1.2. Layout Coordinador.....	30
Figura 6.1.3. Esquema Punto Final.....	30
Figura 6.1.4. Layayout Punto Final.....	31

Índice de Tablas

Tabla 6.1. Listado de componentes empleados.....	32
Tabla 6.2. Listado de precios unitarios componentes	33
Tabla 6.2. Listado de precios unitarios componentes	34

1 INTRODUCCIÓN

Las redes inalámbricas de sensores se basan en pequeños dispositivos (nodos) que son capaces de obtener información del entorno, procesarla localmente, y enviarla de forma inalámbrica hasta un nodo central coordinador.

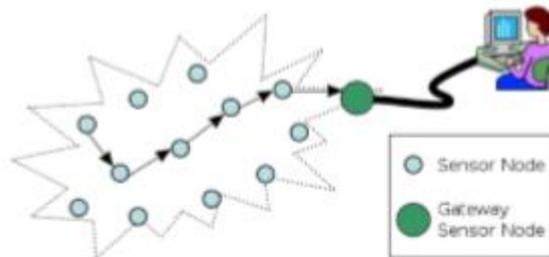


Figura 1.1.1. Estructura de una red inalámbrica de sensores

Estas redes prometen cambiar la forma de obtener información. Se prevé que estarán formadas por miles de sensores diminutos y que no estarán cerradas, es decir, no se centrarán en su aplicación, sino que estarán conectadas a internet. A parte del patrón de flujo de datos anteriormente descrito en el cual el sensor envía información a un punto coordinador, otra característica principal de las WSN son las restricciones energéticas. Uno de los grandes desafíos de este tipo de redes radica en la alimentación de los nodos. Estos requieren bajo consumo y ciclos de trabajo cortos, y será de estos factores de los que dependa en gran medida la fiabilidad de la red.

El proyecto consta de diferentes dispositivos de tipo hardware. Se empezó diseñando la red con módulos Zigbee de la casa ATMEL, pero dado que fueron descatálogos, excepto para la compra de grandes cantidades, el proyecto se reconvirtió a XBEE de la casa Digi.

2 OBJETIVOS

El objeto del proyecto fin de grado es el desarrollo e implementación de una red inalámbrica de sensores ZigBee mediante el módulo Xbee de la casa Digi. Con la comunicación inalámbrica de varios dispositivos, de bajo rango, algunas decenas de metros utilizado en aplicaciones como la domótica, automatización de edificios, control industrial, entre otros.

El proyecto consta de un sistema de sensorización desarrollado por mi compañero Ibon Fuente en el TFG denominado *Sistema de sensorización y visualización de variables físicas*. La estación es la encargada de la medición de las correspondientes variables físicas y la tarjeta de adquisición de datos que permite la comunicación con el entorno Software LabView.

La red inalámbrica hace la función de captar los datos de los sensores, mediante un módulo Xbee para su posterior envío al nodo coordinador mediante la correspondiente antena. Además, al ser una red inalámbrica, cada uno de los nodos sensores estará acompañado un sistema de alimentación independiente que podría ser mediante una placa solar o en su defecto una pequeña pila.

Gracias a la conjunción de ambos proyectos se obtiene una estación autónoma ubicable en cualquier zona sin la necesidad de emplear cables de conexión para el envío de datos. (Fig.2.1.)

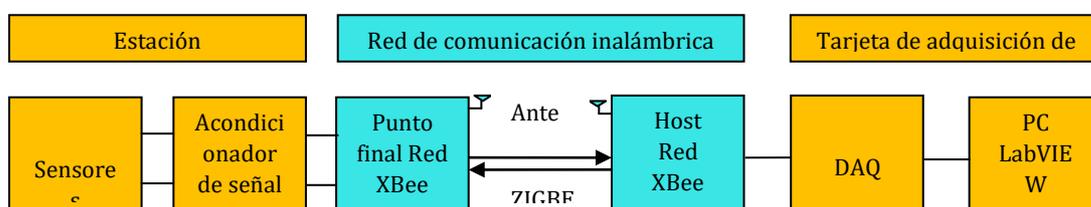


Figura 2.1. Representación del proyecto.

En base a estas características, este proyecto puede subdividirse en dos partes diferentes:

- Parte hardware: Centrado fundamentalmente en el diseño y montaje según su papel en la red:
 - Coordinador: Solo existe un nodo de este tipo por red y es el encargado de iniciar y controlar toda la red.
 - Router: Interconecta dispositivos separados en la topología de la red. Se asocia con el Coordinador de la red o con otro Router.

- Punto final: Posee la funcionalidad necesaria para comunicarse con su nodo padre (el Coordinador o Router).
- Parte Software: Programación de los módulos, según sus funciones para su correcto funcionamiento.

Se adjunta también documentación, planos de la primera versión en Zigbee que por un problema del modulo descatalogado se decidió adaptar la red a XBee.

3 TECNOLOGIA INALAMBRICA. ZIGBEE

ZigBee surge de la necesidad de desarrollar una tecnología inalámbrica de no muy alta transferencia de datos. Así, en 1998, un conjunto de empresas, conocidas como la ZigBee Alliance se juntaron para crear un estándar de comunicaciones que complementara a Wi-fi y Bluetooth. Surge así ZigBee, publicado por el IEEE en Mayo de 2003.

ZigBee se establece como principal objetivo el de comunicar aplicaciones que requieren una comunicación segura, con baja tasa de envío y bajo consumo. Se basa en dispositivos inalámbricos operando en la banda ISM para usos industriales, científicos y médicos (868 MHz, 915 MHz y 2.4 GHz) con una modulación en espectro ensanchado por secuencia directa (DSSS) también conocida como acceso múltiple por división de código en secuencia directa (DS-CDMA). En el rango de frecuencias de 2.4 GHz (banda más extendida) se definen 16 canales con un ancho de banda de 5 MHz.

IEEE 802.15.4 y ZigBee definen diferentes tipos de dispositivos:

- Tipos de dispositivos (según ZigBee):
 - Coordinador: Controla el ruteado y administra la red. Existe uno por red.
 - Router: Interconecta diferentes nodos mediante direccionamiento.
 - Dispositivo final: Elemento pasivo que responde ante peticiones de otros dispositivos. Se pasa la mayor parte del tiempo dormido.
- Tipos de dispositivos (según IEEE 802.15.4):
 - FFD (Full Function Device)
 - RFD (Reduced Function Device)

ZigBee está diseñado para ser una solución que permita crear redes e interconectar dispositivos remotos. De esta forma existen diversas topologías de red que se pueden formar con los dispositivos previamente mencionados:

- **Punto a punto**
 - Topología más sencilla.
 - Uno de los dispositivos es el coordinador y el otro puede ser un router o un dispositivo final.
- **Estrella**
 - Todos los dispositivos de la red se pueden comunicar con el coordinador pero no entre sí.
- **Malla**
 - Se trata de una comunicación punto a punto pero existiendo restricciones en la intercomunicación de dispositivos.
 - Cualquier dispositivo puede comunicarse con otro.
 - El coordinador se encarga de la gestión de caminos.
- **Árbol**
 - El coordinador establece la red inicial.
 - Los routers forman ramas y retransmiten los mensajes.
 - Los dispositivos finales son las hojas del árbol.

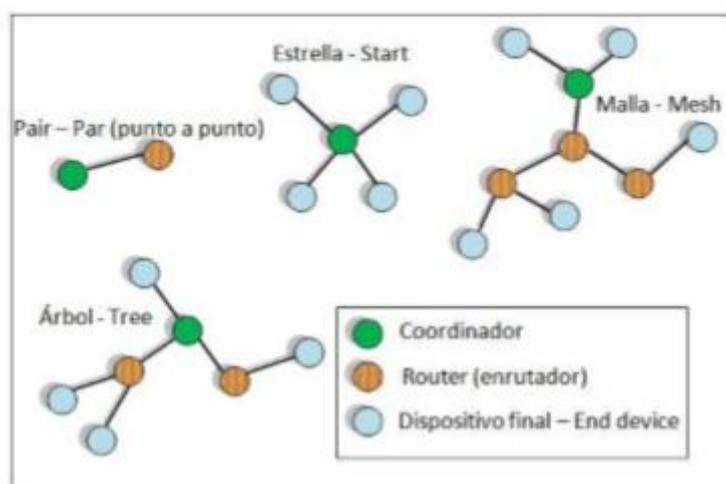


Figura 3.1. Topologías ZigBee..

En su concepción ZigBee se ideó para que tuviese una estructura por capas, lo que simplifica el problema y permite dividir el problema en otros más pequeños. Además una estructura por capas es más flexible ante cambios.

ZigBee sigue el modelo OSI (Interconexión de Sistemas Abiertos) pero reduce las 7 capas de la pila de protocolos a únicamente 4. El estándar 802.15.4 define las dos primeras capas, física y de enlace de datos, mientras que ZigBee define la capa de red y la capa de aplicación.

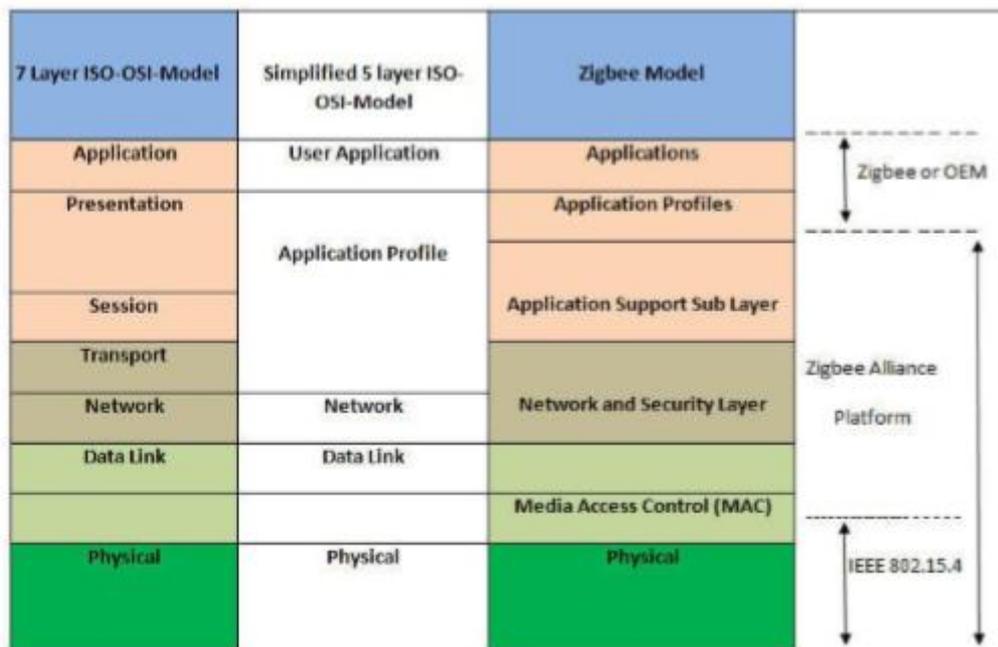


Figura 3.2. Pila de protocolos ZigBe

- **Capa Física:** Define las funciones con la capa de enlace de datos, los niveles de potencia del transmisor y la sensibilidad del receptor, transfiriendo los datos por medio de un entero de 8 bits. Sus funciones son: canales, numeración de canales, detección de energía, medición de portadora, indicador de calidad del enlace, evaluación de canal libre, cliente, servidor entre capas e interfaz/área de datos capa física, enlace de datos.
- **Capa de enlace de datos:** Provee la interfaz entre la capa física y la capa de red. Maneja los servicios y los datos. Sus funciones son: operación de la red usando balizas, espaciado entre tramas, CSMA-CA, nodo oculto y nodo expuesto, formato de la trama.

- **Capa de red ZigBee:** Otorga las funcionalidades para el armado y manejo de redes y una interfaz simple para las aplicaciones de usuarios. Se encarga de: establece la red, configuración de dispositivos, ruteado, y seguridad.
- **Capa de aplicación:** Nivel más alto de la especificación. Es la interfaz efectiva entre el nodo y el usuario. En él se ubican la mayor parte de los componentes definidos por la especificación como la funcionalidad requerida para los dispositivos. El estándar ZigBee ofrece la opción de emplear perfiles en el desarrollo de aplicaciones. Un perfil de aplicación permite la interoperabilidad entre productos de diferentes fabricantes como si fuesen del mismo. La ZigBee Alliance define varios identificadores de perfil, un número de 16 bits de la capa de aplicación que define el perfil.

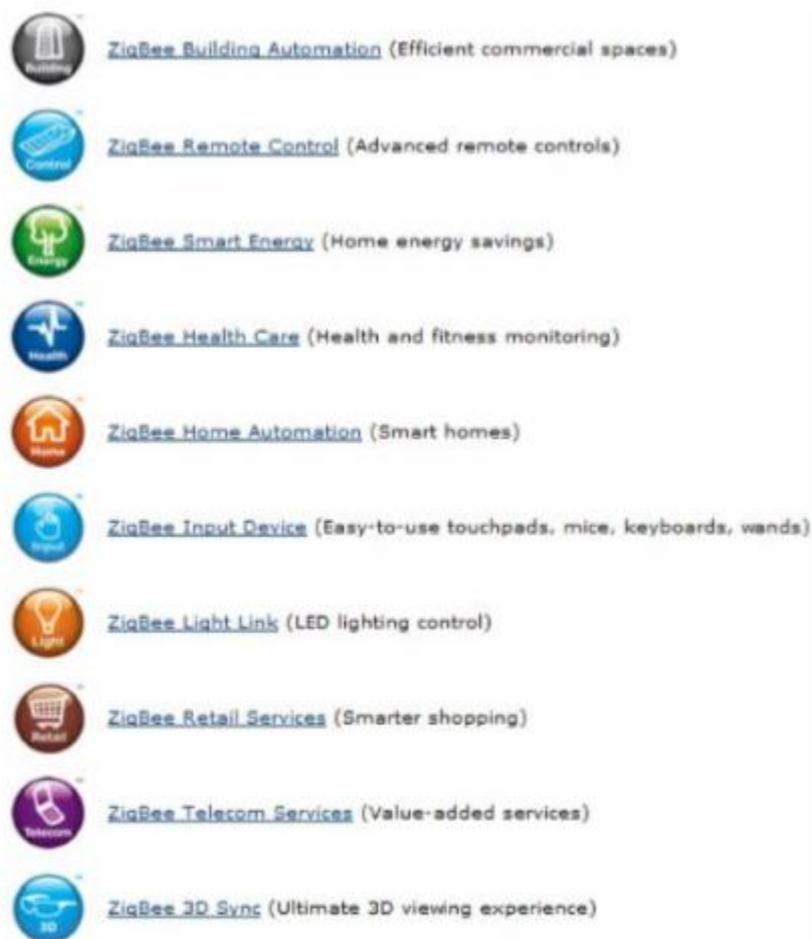


Figura 3.3. Perfiles definidos por la ZigBee Alliance

4 HARDWARE DEL PROYECTO

La elección del módulo ZIGBEE debe tener en cuenta aspectos tales como el precio, la sencillez de la programación o la extensión en el mercado.

Existe una amplia variedad de dispositivos ZigBee (transceptor + micro) en el mercado que proporcionan un listado de fabricantes. A continuación se resumen los utilizados y sus características.

En la figura 2.1.1, se muestran algunos módulos ZigBee existentes:

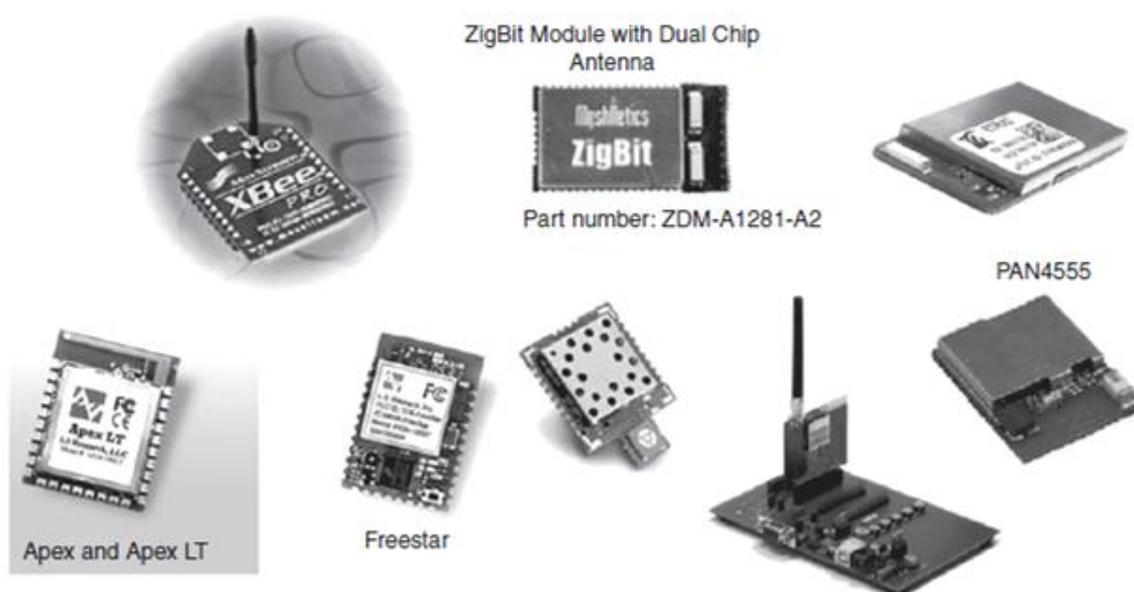


Figura 2.1.1. Algunos módulos ZigBee

Los módulos ZigBee se consideran la mejor solución para el proyecto, ya que entre otras cosas:

- manejo es muy sencillo.
- Reducen el tiempo de diseño.

Funcionan con comandos AT, lo cual es muy interesante, ya que permiten realizar prototipos de aplicaciones sencillas.

Como se puede observar existe una gran variedad de estos módulos ZigBee en el mercado.

4.1 MODULOS ZIGBIT 2,4GHz CON DOBLE CHIP ANTENA

El módulo ZigBit ATZB-24-A2, es un módulo de bajo consumo y alta sensibilidad que utiliza el IEEE 802.15.4/ZigBee.

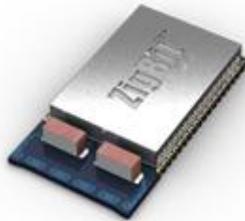


Figura 4.1.2. Módulo ZigBee

4.1.1 Características

Sus características más importantes son las siguientes:

- Tamaño ultra compacto.
- Hasta 3 dBm de salida.
- Muy bajo consumo: <math><6\mu\text{A}</math> en modo sleep.
19mA en modo Rx.
18mA en modo Tx.
- Amplios recursos de memoria: (128kBytes of flash memory, 8kBytes de RAM y 4kBytes de ROM). Por lo tanto, dispone de amplia memoria para aplicación de usuario.
- Trabaja en la banda ISM de 2,4GHz.
- Capacidad de red en malla.
- Software libre.
- Amplia gama de interfaces (I2C, SPI, USART, etc.).

Estas características, unidas a la sencillez de integración en los sistemas, proporcionan a los desarrolladores una reducción considerable de los tiempos de lanzamiento y un ahorro importante en costes.

Si se desean observar con más detalle las características eléctricas del módulo ZigBit, se puede encontrar su datasheet en la parte de documentación del proyecto.

4.1.2 Diagrama de bloques

En la figura 4.1.3, se puede observar su diagrama de bloques.

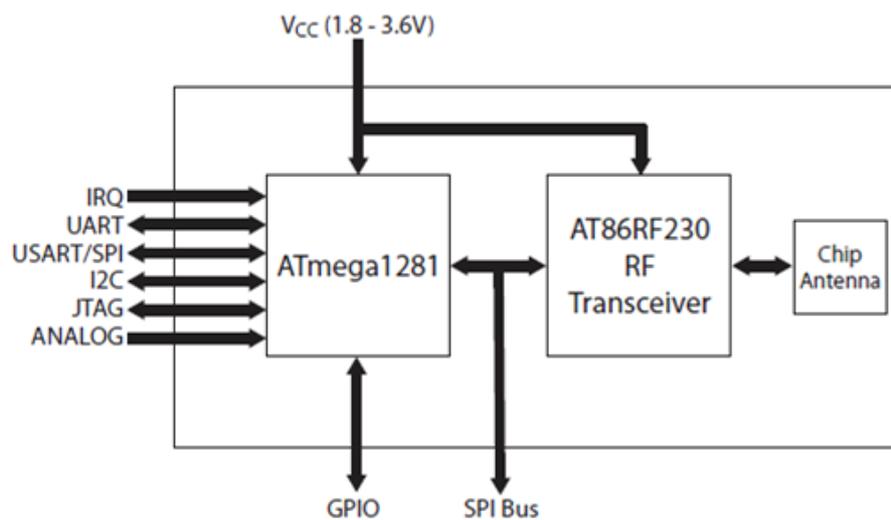


Figura 4.1.3. Diagrama de bloques

Como se puede observar, el módulo está compuesto por:

- ATmega1281

Es un microcontrolador de Atmel, cuyos parámetros más importantes son los siguientes:

- 128KBytes de memoria Flash
- 4096Bytes de memoria EEPROM
- 8192Bytes de memoria SRAM
- Max. Pin I/O=54
- Frecuencia máxima=16MHz
- Vcc=1,8V-5,5V

- AT86RF230 RF Transceiver

Es un transceiver de 2,4Ghz de baja potencia especialmente diseñado para aplicaciones IEEE 802.15.4, ZigBee y 6LoWLAN de bajo coste.

- Antena Chip

La antena viene incorporada en el propio módulo. Por lo tanto en el diseño no se tendrá que incorporar ninguna, por lo que se simplifica.

En el diagrama de bloques se observan también las salidas y entradas que ofrece el módulo.

4.1.3 Evaluación de pines que se utilizan del módulo Zigbit

En la figura 2.1.4 se muestra la configuración de los pines del módulo, la descripción de los mismos. Esta información se obtiene del datasheet del módulo (ATZB-24-A2/B0), el cual se encuentra en la carpeta Documentos del proyecto.

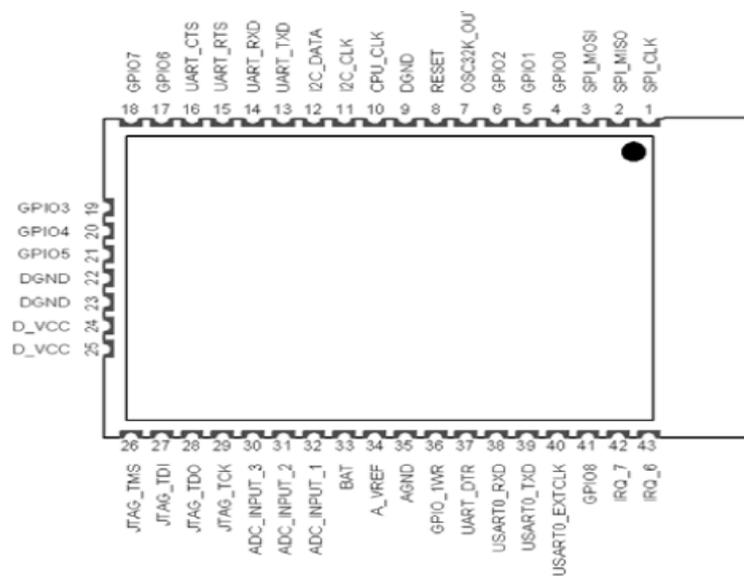


Figura 2.1.4. configuración de pines

Los pines que se van a utilizar son los siguientes:

- D-VCC: Es la tensión de alimentación, debe estar en el rango de 1,8V a 3,6V para que funcione. Existen dos pines de este tipo, el pin 24 y el 25. Ambos son iguales y por lo tanto deben ser conectados a la misma alimentación.
- DGND y AGND: Masa digital y masa analógica respectivamente. Existen tres pines DGND, pines 9, 22 y 23 y uno AGND, pin35. Todas ellas se tienen que unir a la misma masa exterior.
- RESET: Pin 8, como se puede observar en la tabla 5-1, es activo a nivel bajo. Este pin no se puede dejar al aire por que podría dar lugar a comportamientos no deseados,

por lo tanto, si se desea hacer un reset este pin se debe conectar a masa (a nivel bajo), y si no, como es el caso, debe estar conectado a D-VCC (a nivel alto).

-
- UART_TXD: Se trata del pin 13, recepción de la UART, es un pin de entrada.
- UART_RXD: Se trata del pin 14, transmisión de la UART, es un pin de salida.

En la descripción de los pines del módulo ZigBit en su datasheet (Documentación, xxx), al pin que es de transmisión (pin de salida) se le denomina RXD y al que es recepción (pin de entrada) se le denomina TXD. Lo cual resulta algo contradictorio, por lo que a partir de ahora en la redacción de los documentos, se le llamará TXD a la transmisión y RXD a la recepción, para evitar confusiones.

- GPIO: Son pines de entrada/salida. Existen nueve de este tipo, pines 4, 5, 6, 17, 18, 19, 20, 21 y 41.

Internamente se unirán los pines de alimentación D-VCC entre sí, y los pines DGND y AGND, para posteriormente llevar un cable al exterior de VCC y GND. Entre VCC y GND, se conecta un condensador de desacoplo de 10 μ F.

El pin de RESET se lleva internamente mediante una resistencia de 100k Ω a VCC. La función de la resistencia es la de permitir realizar un reset cuando sea necesario, sin tener que deshacer las conexiones a VCC, ya que si no se deshiciera esta conexión se provocaría un cortocircuito, lo cual no es lo deseado.

Se sacan las conexiones UART_TXD y UART_RXD mediante cables, para facilitar las conexiones al exterior.

4.2 Xbee ZNet 2.5 RF Module

- Bajo coste y bajo consumo típico de WSN.
- Fácil uso y entrega fiable de datos entre dispositivos.
- Pequeño tamaño. o No está configurado para ningún nivel de aplicación específico.
- Muy extendido entre el público.
- No dispone de microcontrolador.
- Posee un software de programación gratuito y se puede descargar de la propia web del fabricante. o Existe una versión superior que proporciona mayor potencia y cobertura a cambio de mayor tamaño y mayor consumo.



Figura 4.2.1. Módulo Xbee ZNet 2.5 RF Series2 de Digi

Existen dos tipos de módulos Xbee; los módulos Xbee Serie2 y los módulos Xbee Serie1. La principal diferencia existente entre ambos es que los de la Serie2 sí permiten hacer redes mesh. Además existen los mencionados Xbee Pro, que permiten mayor alcance y potencia de señal. En lo sucesivo describiremos el funcionamiento y configuración de todos estos módulos (pues es muy similar), sin embargo, nosotros emplearemos módulos Xbee Serie2.

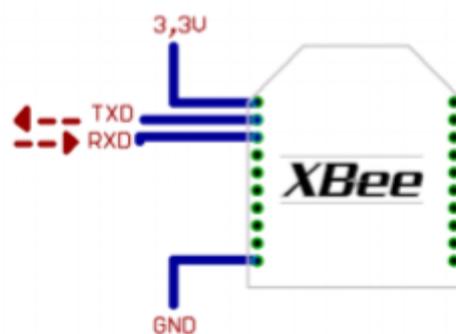


Figura 4.2.2. Conexiones mínimas requeridas para el Xbee.

El módulo requiere una alimentación de 2.8 a 3.4 V, la conexión a tierra y las líneas de transmisión de datos por medio del UART (TXD y RXD) para comunicarse con un microcontrolador o un puerto serial.

Los modos de operación del módulo Xbee son los siguientes:

- Modo recepción/transmisión.

Se encuentra en uno de estos modos cuando llega al módulo algún paquete RF por la antena (modo recepción) o cuando se manda información serial al buffer del pin 3 que luego será transmitida (modo transmisión).

- Modo de bajo consumo.

Este modo hace posible que el módulo se encuentre en un estado de bajo consumo cuando no está en uso. La configuración de los ciclos de sueño se realiza mediante comandos. Por defecto, los ciclos de sueño están deshabilitados (módulo en estado de reposos/recepción).

Mediante el pin de hibernación Sleep_RQ (pin 9) en estado alto, el módulo termina cualquier transmisión, recepción o procedimiento de asociación y entra en modo reposo y luego en modo sueño. En este estado no responde a comandos entrantes de ningún tipo. Sólo cuando se baje el estado lógico de Sleep_RQ el módulo saldrá del estado de sueño y podrá volver a enviar y recibir datos.

Existe otro modo de sueño a parte del de controlado por pin, este es el modo de sueño cíclico que se basa en despertar al módulo cada cierto período para que compruebe si existen datos para enviar.

- Modo de comando.

En este modo es posible configurar parámetros del módulo mediante comandos AT. Para ello se debe emplear un software como el Hyperterminal de Windows o el X-CTU de Digi.

Se entra en modo de comando escribiendo por el terminal el string “+++”. Tras la devolución de un OK, es posible ingresar comandos mediante la sintaxis siguiente.

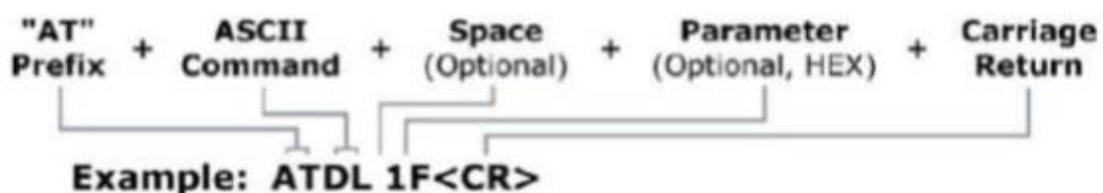


Figura 4.2.3. Ejemplo comando AT

- Modo transparente.

En modo transparente todo lo que ingresa por el pin 3 (DIN) se guarda en buffer de entrada y se transmite y aquello que ingresa como paquete RF se guarda en buffer de salida y se envía por el pin 2 (DOUT) inmediatamente o tras el paso de un tiempo

(parámetro RO). Este modo se destina a comunicación punto a punto, donde no es necesario ningún tipo de control.

- Modo de operación API.

Cuando se encuentra en este modo, el módulo empaqueta en tramas toda la información que entra y sale. Estos definen operaciones y eventos del módulo. Entre las opciones que permite la API, se definen:

- Transmitir información a múltiples destinatarios, sin entran al modo de comandos.
- Recibir estado de éxito/fallo de cada paquete RF transmitido.
- Identificar la dirección de origen de cada paquete recibido.

Estructura de una trama API

La estructura de la trama API se muestra en la figura.



Figura 4.2.4. Ejemplo de trama

En la trama el primer byte 0x7E indica el comienzo de la trama, esto muestra el tipo de funcionamiento en el que está configurado. Los siguientes dos bytes indican solamente la longitud de la trama que se especifica desde el byte más significativo (MSB = Most Significant Byte, LSB = Least Significant Byte) es decir toda la trama sin incluir el byte que indica el comienzo de la trama, la siguiente parte es la data frame o campo de longitud que tiene un valor de dos bytes que especifican el número de bytes que contendrá el campo de datos de la trama, pero sin incluir el checksum. El checksum es un byte que permite probar la integridad de los datos, para calcular no incluye los delimitadores de trama y la longitud, se agregan todos los bytes manteniendo solo 8 bits más bajos del resultado y se resta del =xFF, es la suma de todos los bytes contenidos en el paquete, si la suma no coincide el paquete queda descartado.

- Modo Idle.

Se dice que si el módulo no está en ninguno de los estados anteriormente mencionados se encuentra en estado IDLE.

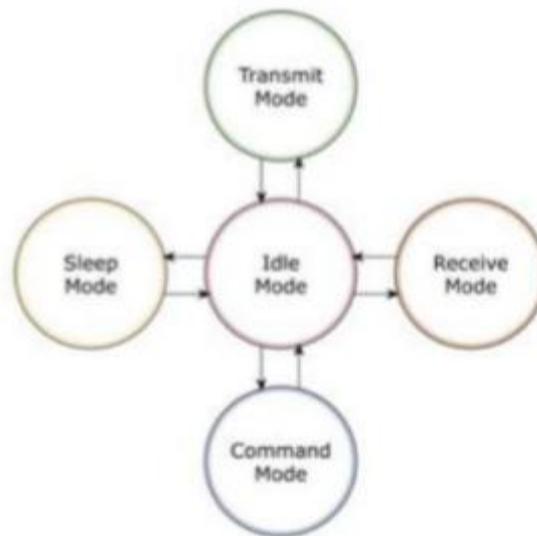


Figura 4.2.5. Modos de funcionamiento del módulo Xbee.

4.3 Placa Xbee Explorer USB

Esta pequeña placa impresa contiene los pines de conexión para un módulo Xbee así como un controlador que sirve para programarlos y tomar sus datos. Sirve para todo tipo de módulos Xbee, tanto serie1 como serie2 y también para la versión Pro.

Está equipada con una entrada mini USB permitiendo conectar la placa a un ordenador y leer y escribir datos a partir del puerto serie.



Figura 4.3.1. Xbee Explorer USB



Figura 4.3.2. Conexión del módulo Xbee al Explorer USB

En el caso que nos ocupa será el módulo Xbee que actúe como coordinador el que esté sobre el Xbee Explorer USB recibiendo los datos de los demás sensores inalámbricos y enviándoselos a un ordenador para que este pueda recogerlos, interpretarlos y mostrarlos por pantalla

4.4 XBee SIP Adapter

Viene completamente ensamblado y proporciona una solución de espacio reducido para conectar el microcontrolador a cualquier módulo XBee o XBee-Pro. Un regulador de 3.3 voltios y un buffer 74LVC244A a bordo proporcionan una conexión segura a un suministro de 5 voltios. El encabezado SIP doble 2 x 5 hace una conexión robusta a su placa de pruebas.



Figura 4.4.1. USB XBee SIP Adapter

4.4.1 Características

- Regulador 3.3 V incorporado
- 5 V a 3.3 V búfer de E / S comunes
- Seis LED indicadores de estado para alimentación, Tx, Rx, RSSI, asociado y modo (suspensión / encendido)
- Cabezal SIP dual de pequeña huella que brinda soporte y permite la interfaz fácil con DOUT (TX), DIN (RX), RTS, 5 V de suministro
- conexiones de cabezal hembra de 5 pines y tierra proporciona interconexión con otros pines XBee

4.5 Bateria

Se ha elegido una batería alcalina típica de 9V, pues ofrece una muy buena calidad precio. Las baterías alcalinas son las más comunes en hogares y son muy buenas para una gran

variedad de aplicaciones electrónicas, sobre todo porque las empresas de baterías han mejorado la química de estas.

Las pilas alcalinas son de bajo costo, ampliamente disponibles y son ideales para aplicaciones de baja corriente a temperatura ambiente, pero tienen dos defectos: son muy malas en condiciones frías y no tienden a funcionar correctamente en condiciones de alta corriente.

4.6 Arduino uno

Arduino Uno es una placa basada en el microcontrolador ATMEGA328. Posee 14 pines de entradas/salidas digitales (6 de los cuáles se pueden emplear como salidas PWM y dos de ellas se pueden emplear para la comunicación serie) y 6 entradas analógicas, un resonador de 16MHz, conexión USB, conector de alimentación, un puerto ICSP, y un botón de reset.

Arduino dispone de un entorno de desarrollo propio y está pensada para fomentar y facilitar el uso de la electrónica para el público en general, desde escuelas hasta aficionados a la electrónica.

Al tratarse de un microcontrolador con un lenguaje basado en C / C++, existen funciones de entrada/salida analógica y digital, funciones aritméticas, manejo de interrupciones, comunicaciones por el puerto serie, etc.

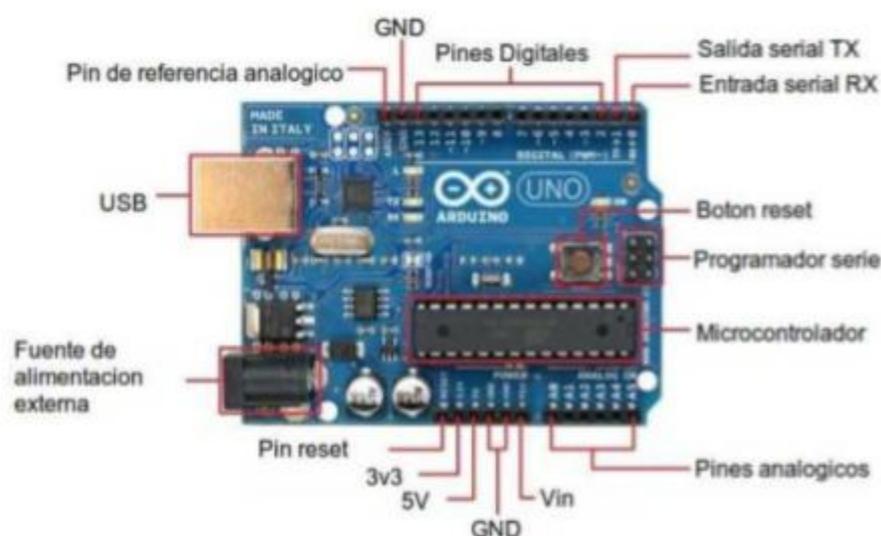


Figura 4.6.1. Disposición Placa Arduino Uno

5 SOFTWARE DEL PROYECTO

Para el desarrollo y realización de este proyecto se han utilizado diferentes herramientas software informáticas. Estos se han utilizado, por un lado, para la realización de las diseño de las placas de los módulos Zigbee y herramientas de programación para el funcionamiento de la red.

Los programas empleados se detallan a continuación.

5.1 IDE de Arduino

Arduino es una plataforma de hardware libre, su diseño como ya es sabido se basa en una placa con un microcontrolador Atmel AVR. En su arquitectura, como ya se comentó en el apartado anterior, dispone de varios puertos de entrada/salida, así como un entorno de desarrollo de fácil empleo para el desarrollo de proyectos.

El entorno de programación empleado por la plataforma Arduino para todos sus dispositivos es libre y de uso abierto. Es posible la descarga gratuita desde internet de las versiones del compilador y a fecha de redacción del presente documento, existe la posibilidad de descargarse desde la web <https://www.arduino.cc/en/Main/Software>.

Las plataformas que emplean este tipo de software, son beneficiosas para el desarrollo de aplicaciones y constan principalmente de las siguientes partes:

- Editor de código
- Un compilador
- Un depurador
- Constructor de interfaz gráfica

No se entrará a explicar el lenguaje de programación aquí (hay una buena referencia en la página web de Arduino) .

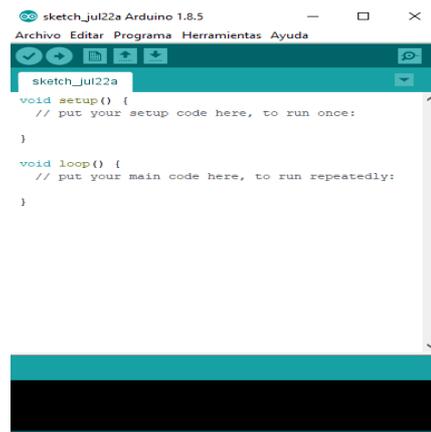


Figura 5.1.1. IDE Arduino.

5.1.1 Programacion: Leer un dato análogo de un XBee remoto en modo API

```
#include <SoftwareSerial.h>

SoftwareSerial Xbee(3,2) //RX TX

float temp;

void setup()
{
  Serial.begin(9600);
  Xbee.begin(9600);
}

void loop ()
{
  if (Xbee.available(>21)
  {
    for(int i=0; i<22;i++)
    {
      Serial.print(Xbee.read(),HEX);
```

```
Serial.print(",");
}

Serial.println("");
}
}
```

5.2 Programación de los módulos Xbee: X-CTU

Se puede emplear el Hyperterminal de Windows para configurar un módulo Xbee, o en su defecto instalarlo dado que la versión Win10 no lo trae de serie. Pero existe un programa llamado X-CTU de la casa Digi, el cual permite realizar estas operaciones de manera más natural, fácil y rápida. El cual se puede descargar desde la web de Digi: <https://www.digi.com/products/xbee-rf-solutions/xctu-software/xctu>.

Se va a proceder a explicar la configuración de los mudolos Xbee que se conectaran al PC mediante un cable USB.

- Para acceder a los parámetros de configuración del modulo Xbee desde X.CTU, se selecciona el icono de busca componentes conectados al PC.
- Se selecciona el puesto USB al que están conectado. En este caso COM3.
- Luego se seleccionan los parametod del puesto, como se muestra en la figura 5.2.1.

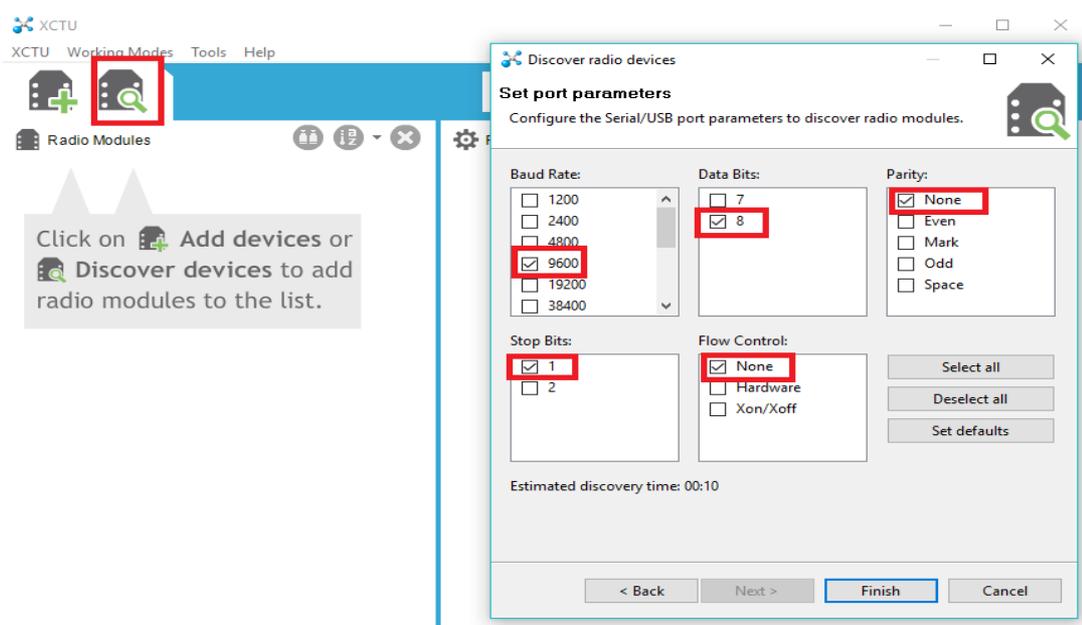


Figura 5.2.1. Selección del dispositivo conectado

- El programa busca y reconoce el módulo conectado mostrándolo en la parte izquierda, como se muestra en la figura 5.2.2. Para Acceder a los parámetros del XBee se selecciona el modulo y se muestra la información perteneciente al XBee en la parte de la derecha.

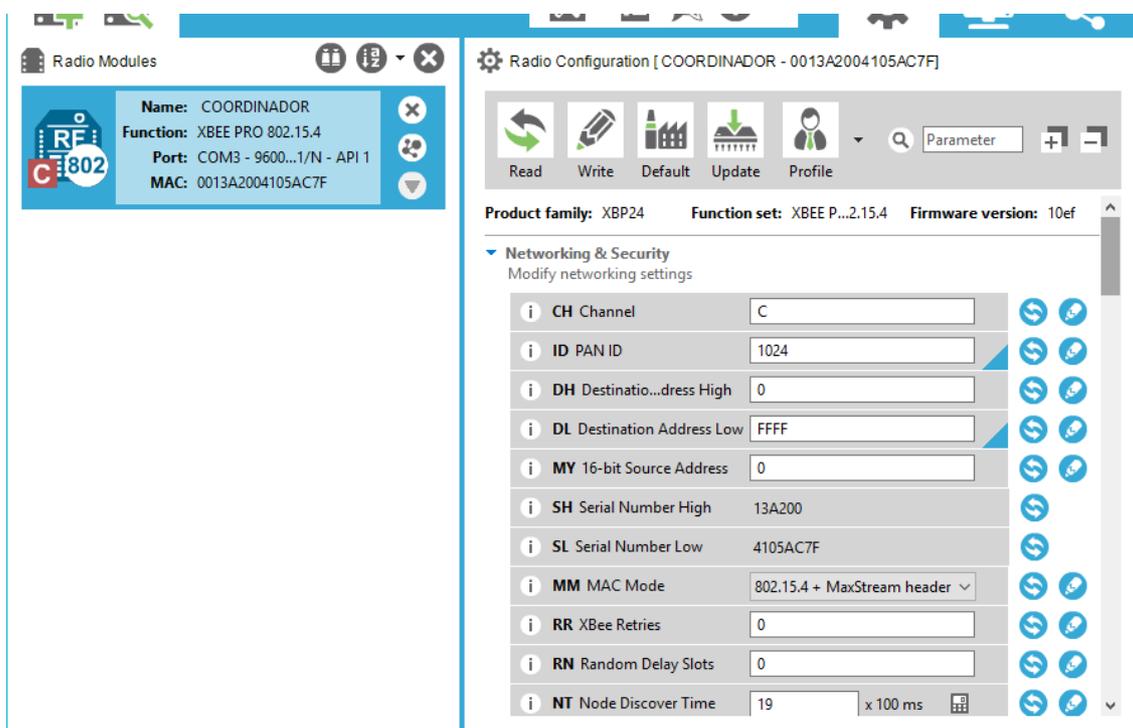


Figura 5.2.2. Panel configuración parámetros XBee

- Se procede a continuación a explicar los parámetros más importantes:
 - IDPAN ID: Identificador de red, por ejemplo 1024. Todos los dispositivos de la red tienen que tener el mismo.
 - CE Coordinador Enable: El coordinador tiene que tener este parámetro a 1.
 - 1 Coordinador
 - 0 Punto final
 - SH Serial Number High: Numero serial del dispositivo conectado hay 2 High y Low.
 - SL Serial Number Low: Numero serial del dispositivo conectado hay 2 High y Low.
 - DH Destination adress high: Dirección del dispositivo con el que queremos comunicarnos, por defecto es 0, lo que significa que solo se comunicara con el coordinador. En caso de querer comunicarte con otro dispositivo se colocara el SH Serial Number High y SL Serial Number Low del dispositivo en cuestión en el DH Destination adress low.

- DH Destination adress low: Se coloca FFFF, Así el mensaje se manda a todos los dispositiivos de la red.
- AP API Enable:
 - API disable, comunicación por comandos AT
 - Api enable 1, comunicación por medio de tramas
- SM Sleep Mode:
 - No Sleep 0, configurado como router
 - Con un valor diferente, configurado como un Dispositivo Final
- I/O Settings:
 - ADC: para recibir un dato analógico de un sensor
 - DI: Entrada digital
 - DO: salida digital en bajo o en alto

5.3 PROTEUS

Proteus es un entorno software de diseño electrónico. En él se han realizado la generación de las placas a implementar.

Para la realización de la red XBee se ha utilizado conexiones mediante una protoboard. Pero para la anterior red Zigbee se diseñaron las placas de Coordinador, Router y Punto final que se adjuntaran al proyecto.

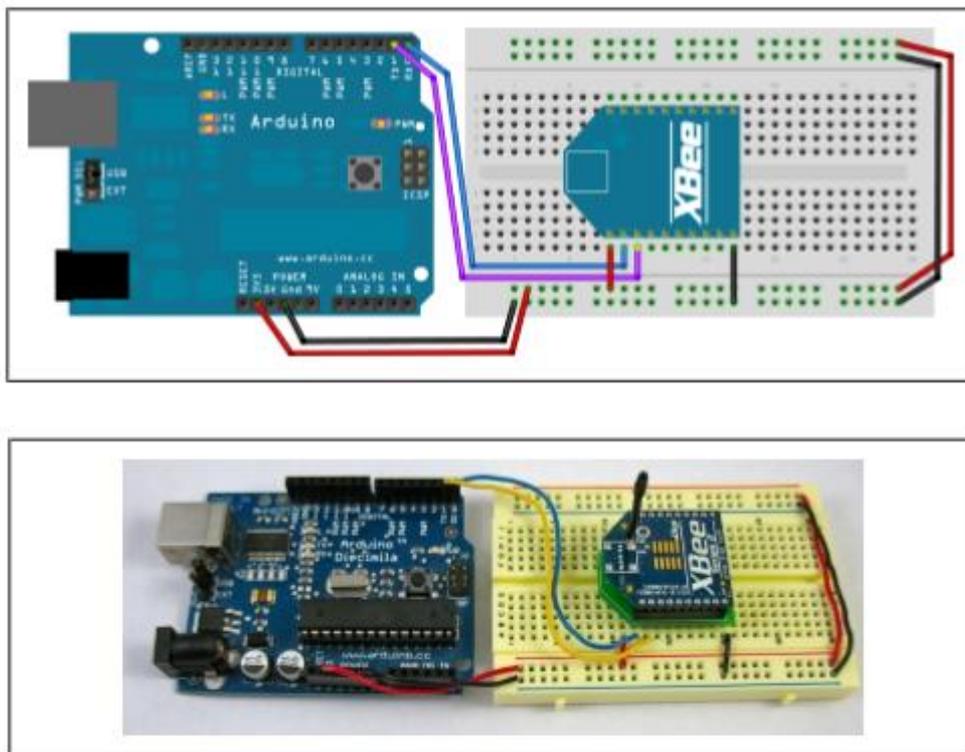


Figura 5.3. Esquema de los nodos sensores.

6 PLANOS

En este documento se pueden observar los esquemas realizados. El primero, es la conexión del módulo ZigBit de ATMEL con el PC, que también vale como router, mientras que el otro es el punto final.

Los planos se han realizado con herramienta de captura de esquemas ISIS de Labcenter Electronics.

6.1 Coordinador o Router

Este esquema representa las conexiones realizadas entre el módulo ZigBit y el ordenador. En el plano, se puede observar, la conversión de los niveles TTL de la UART del módulo ZigBit a niveles RS-232 para realizar la conexión con un adaptador comercial de USB a puerto serie. También se reflejan las conexiones internas realizadas en el módulo ZigBit.

Inicialmente todos los módulos deberán conectarse al ordenador, para que primero se les instale la aplicación SerialNet por encima del stack certificado “ZigBee PRO” de Atmel. Esta aplicación permite conectar el módulo ZigBit a otro host por medio de un interface serie; el control del módulo se realiza desde el host por medio de comandos AT normalizados. Todos los módulos ZigBit utilizados en el proyecto utilizan este sistema de conexión.

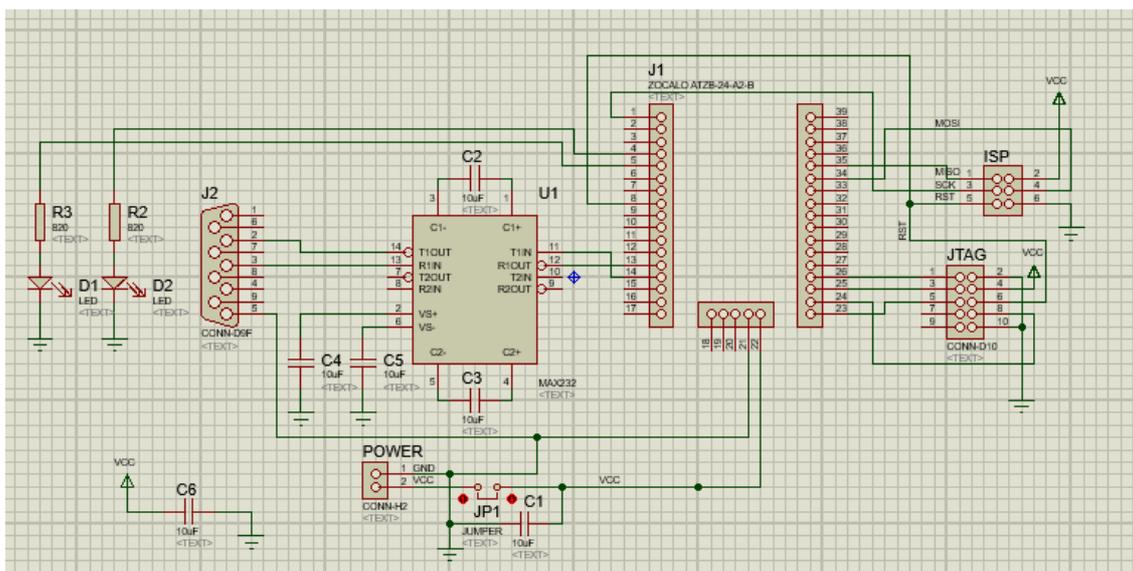


Figura 6.1.1. Esquema Coordinador.

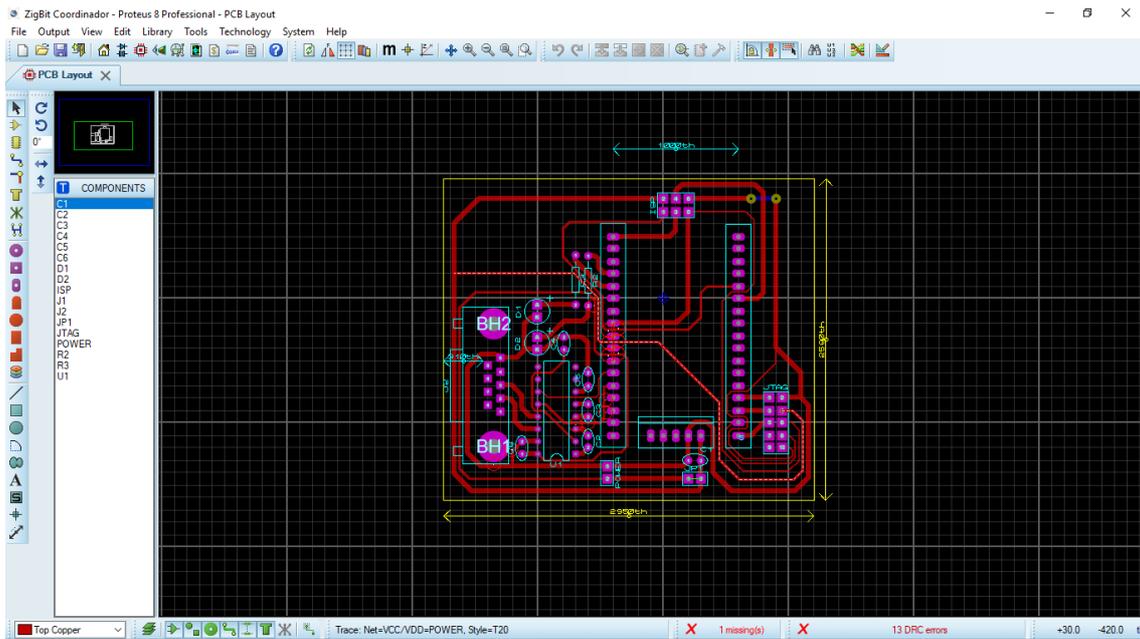


Figura 6.1.2. Layout Coordinador.

6.2 Punto Final

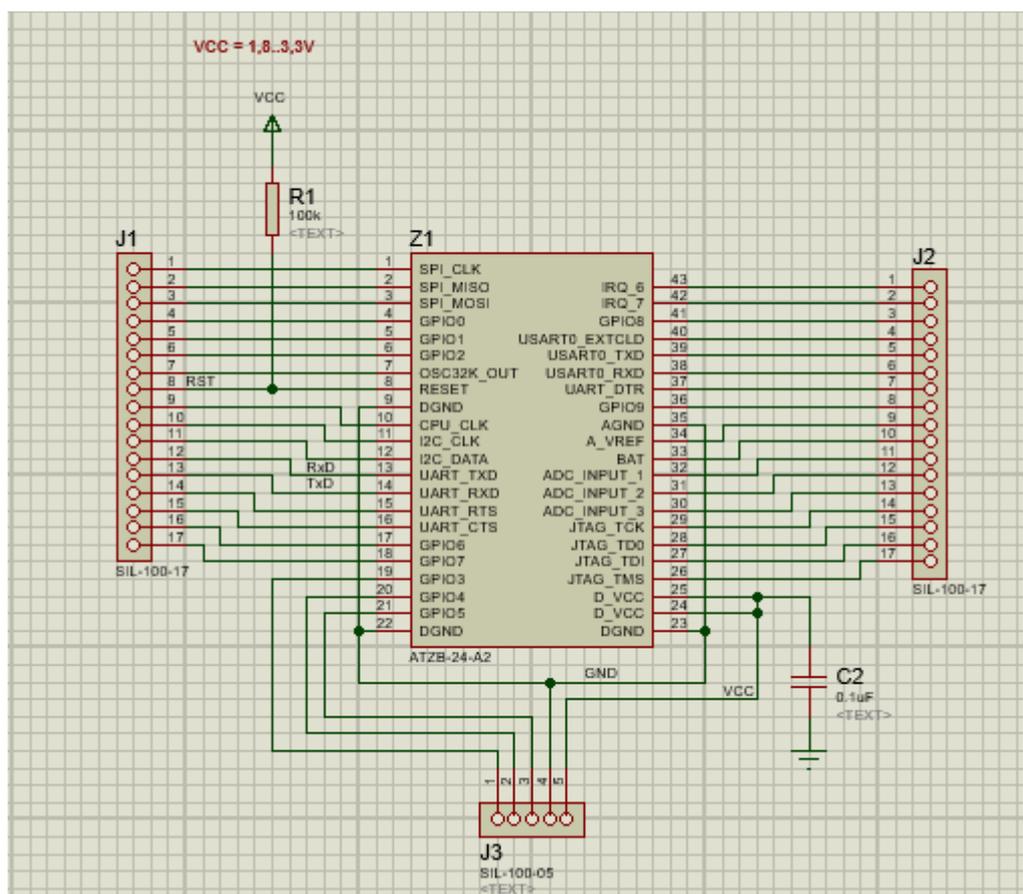


Figura 6.1.3. Esquema Punto Final.

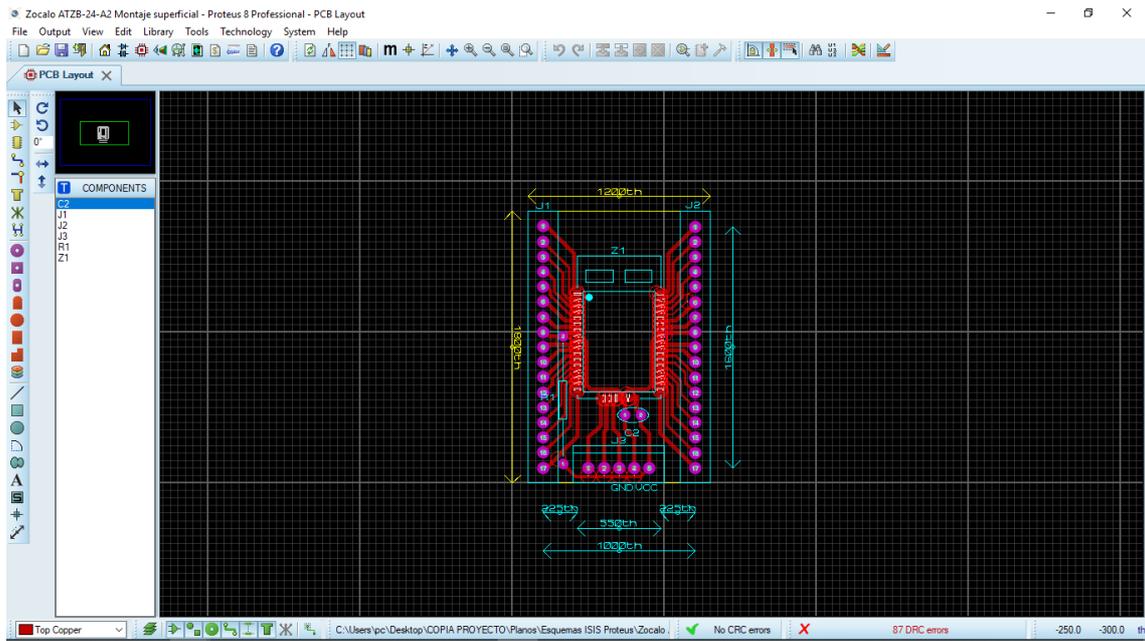


Figura 6.1.4. Layout Punto Final.

7 LISTADO COMPLETO DE MATERIALES

Los componentes empleados en el proyecto se han comprado por internet. Las páginas a través de las cuales se han adquirido son:

- Amazon (XBee para comunicación)
- RS-Amidata (Rs_Components)

Del mismo modo, se han adquirido componentes en la tienda física Radio Rihn ubicado en la calle Urkixo Zumarkalea, 32, 48010 Bilbo, Bizkaia. La lista completa de componentes Hardware empleados en el proyecto aparece desglosado en la siguiente tabla 6.1

Lista	Componente	Unidades
1	Módulo inalámbrico – Xbee ZNet 2.5 RF Series2	2
2	XBee SIP Adapter	2
3	Placa Xbee Explorer USB	2
4	Placa Protoboard	2
5	Estaño	1
6	Microprocesador Arduino UNO	1
7	Cable USB	1
8	Cable Conector Pila 9V	1
9	Pila 9V	1
10	Cable	1
11	Conectores PCB Macho	1

Tabla 6.1. Listado de componentes empleados

8 PRECIOS UNITARIOS DE MATERIALES

En el listado que aparece a continuación está el desglose del coste unitario por componente.

Lista	Componente	Precio unitario
1	Módulo inalámbrico – Xbee ZNet 2.5 RF Series2	55 €
2	XBee SIP Adapter	39,72 €
3	Placa Xbee Explorer USB	42 €
4	Placa Protoboard	7 €
5	Estaño	8 €
6	Microprocesador Arduino UNO	16,5 €
7	Cable USB	3,50 €
8	Cable Conector Pila 9V	0,5 €
9	Pila 9V	3 €
10	Cable	3,3 €
11	Conectores PCB Macho	0,5€

Tabla 6.2. Listado de precios unitarios componentes

9 PRESUPUESTO

En el listado que aparece a continuación esta el desglose del coste unitario por componente.

Lista	Componente	Unidades	Precio unitario
1	Módulo inalámbrico – Xbee ZNet 2.5 RF Series2	2	55 €
2	XBee SIP Adapter	2	39,72 €
3	Placa Xbee Explorer USB	2	42 €
4	Placa Protoboard	2	7 €
5	Estaño	1	8 €
6	Microprocesador Arduino UNO	1	16,5 €
7	Cable USB	1	3,50 €
8	Cable Conector Pila 9V	1	0,5 €
9	Pila 9V	1	3 €
10	Cable	1	3,3 €
11	Conectores PCB Macho	1	0,5€
TOTAL			322,74€

Tabla 6.2. Listado de precios unitarios componentes

10 MEJORAS PLANIFICADAS A FUTURO

Una de las principales mejoras a futuro y que sería interesante abordar es la generación de módulos inalámbricos para todos los sensores.

Por otra parte, la implementación de placas para cada módulo también es un punto que se debería tener en cuenta ya que en el presente proyecto se ha optado por la opción de prototipado rápido.

La alimentación empleando placas solares como recurso energético sostenible también sería un buen planteamiento. Como algún módulo va colocado en el exterior se puede aprovechar este recurso energético.

11 BIBLIGRAFIA

Para la realización del presente trabajo fin de grado se han consultado las siguientes referencias bibliográficas, que se citan a continuación:

- Essic. J. (2013). LabVIEW for Scientists and Engineers. New York. Oxford University Press.
- Manual de Telecomunicaciones, José M. Huidobro (Ra-Ma).
- Fernández Martínez, R. (2009). Redes Inalámbricas de sensores
- Guía usuario Xbee
- XBee-Quick-Reference-Guide.pdf
- Robert Faludi, Building Wireless sensor Network

Las páginas web de consulta son las siguientes:

- [1] <https://www.arduino.cc/>
- [2] <http://www.ni.com/es-es/support.html>
- [3] <https://forum.arduino.cc/index.php?board=32.0>
- [4] <https://www.forosdeelectronica.com/>
- [5] <https://www.digi.com/pdf/xbee-802-15-4-protocol-comparison>
- [6] https://www.sparkfun.com/pages/xbee_guide
- [7] <https://www.arduino.cc/en/Guide/ArduinoWirelessShieldS2>
- [8] <https://www.arduino.cc/en/Main/Software>
- [9] <https://www.digi.com/products/xbee-rf-solutions/xctu-software/xctu>