



ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

Facultad de Ingeniería en Electricidad y Computación

“RED DE COMUNICACIÓN XBEE ENTRE MINICOMPUTADORA  
RASPBERRY PI Y PC CON CAPACIDAD DE COMUNICACIÓN WIFI PARA  
EL ALMACENAMIENTO DE INFORMACIÓN EN BASE DE DATOS  
REMOTA”

TESINA DE SEMINARIO

Previo la obtención del Título de:

INGENIERO EN ELECTRÓNICA Y

TELECOMUNICACIONES

Presentado por:

Cinthia Leonor Espinoza Moran

Christian Fernando Cando Cevallos

GUAYAQUIL – ECUADOR

2013

## **AGRADECIMIENTO**

Agradezco a Dios por tener la oportunidad de culminar con este seminario mi carrera universitaria.

Gracias a mi madre quien me dio todo el apoyo necesario día tras día a lo largo de todos estos años en mi formación universitaria, agradezco a mis hermanos, a mis tíos y mis amigos por darme el apoyo cuando lo necesitaba.

Gracias a mis tres ángeles que me guían y me cuidan desde el cielo ya que a ellas le agradezco mis estudios y lo que soy.

Christian Fernando Cando Cevallos.

Agradezco en primer lugar a Dios por permitirme terminar esta etapa de mi vida.

Gracias a mis padres y hermanos que fueron mi apoyo incondicional.

Gracias a esa persona especial que me motivo diariamente para cumplir con este objetivo.

Agradezco a mis compañeros, y profesores que me acompañaron durante esta etapa.

Cinthia Leonor Espinoza Moran.

## DEDICATORIA

El trabajo y lo que representa la elaboración de este documento no sería posible sin el soporte y respaldo de mis Padres, sin el apoyo incondicional de mis hermanos y amigos y sin la oportunidad que me dio la Escuela. Mi dedicatoria va para todos ellos.

Christian Fernando Cando Cevallos

El presente Trabajo se lo Dedico a Dios que me dio sabiduría para alcanzar mi objetivo, y a mis padres por su apoyo incondicional, a mi amigo y compañero de tesis y a todos las personas que aportaron para cumplir con esta meta.

Cinthia Leonor Espinoza Moran

## TRIBUNAL DE SUSTENTACIÓN

---

**Ing. Carlos Valdivieso A.**

Profesor del Seminario de Graduación

---

**Ing. Hugo Villavicencio.**

Profesor Delegado del Decano

## **DECLARACIÓN EXPRESA**

“La responsabilidad del contenido de esta Tesina, nos corresponde exclusivamente; y el patrimonio intelectual de la misma a la Escuela Superior Politécnica del Litoral”.

(Reglamento de Graduación de la ESPOL)

---

**Christian Cando Cevallos**

---

**Cinthia Espinoza Morán**

## ÍNDICE GENERAL

<b>AGRADECIMIENTO .....</b>	<b>II</b>
<b>DEDICATORIA .....</b>	<b>IV</b>
<b>TRIBUNAL DE SUSTENTACIÓN .....</b>	<b>V</b>
<b>DECLARACIÓN EXPRESA.....</b>	<b>VI</b>
<b>ÍNDICE GENERAL .....</b>	<b>VII</b>
<b>ÍNDICE DE FIGURAS.....</b>	<b>X</b>
<b>ABREVIATURAS.....</b>	<b>XIII</b>
<b>RESUMEN.....</b>	<b>XIV</b>
<b>INTRODUCCIÓN .....</b>	<b>XV</b>
<b>CAPÍTULO 1.....</b>	<b>1</b>
<b>1.1 Objetivos.....</b>	<b>1</b>
<b>1.1.1 Objetivos Generales.....</b>	<b>1</b>
<b>Objetivos Específicos .....</b>	<b>2</b>
<b>Antecedentes.....</b>	<b>2</b>
<b>Identificación del Problema.....</b>	<b>3</b>
<b>Descripción del Proyecto .....</b>	<b>4</b>

<b>Limitaciones .....</b>	<b>5</b>
<b>CAPÍTULO 2.....</b>	<b>6</b>
<b>FUNDAMENTO TEORICO.....</b>	<b>6</b>
<b>2.1.- Comunicación .....</b>	<b>6</b>
<b>2.1.1. Sistema de Comunicación .....</b>	<b>6</b>
<b>2.1.2 Medios de Comunicación .....</b>	<b>7</b>
<b>2.1.2.1 Comunicación Alámbrica.....</b>	<b>7</b>
<b>2.2 Herramientas de Software Para el Desarrollo del .....</b>	<b>12</b>
<b>Proyecto.....</b>	<b>12</b>
<b>2.2.1 PYTHON .....</b>	<b>12</b>
<b>2.3.1 RASPBERRY PI.....</b>	<b>17</b>
<b>2.3.2 XBee .....</b>	<b>17</b>
<b>2.3.2.1 Aplicaciones .....</b>	<b>20</b>
<b>2.3.3 Wifi TP-LINK .....</b>	<b>20</b>
<b>2.4.3 Conexión Remota.....</b>	<b>22</b>
<b>2.4.3.1 VPN.....</b>	<b>22</b>
<b>3.1 Actualización del Software .....</b>	<b>26</b>



<b>3.2 Red Xbee.....</b>	<b>29</b>
<b>3.2.1 Configuraciones XBee's .....</b>	<b>29</b>
<b>3.3 Red LAN.....</b>	<b>32</b>
<b>CAPÍTULO 4.....</b>	<b>37</b>
<b>4.1 Simulación del proyecto desde PC A PC.....</b>	<b>37</b>
<b>4.2 Simulación de la Potencia de datos enviados.....</b>	<b>39</b>
<b>4.3 Simulación desde PC – RASPBERRY PI.....</b>	<b>40</b>
<b>4.3 Simulación paso a paso de Conexión Remota a la RASPBERRY PI .....</b>	<b>41</b>
<b>4.4 Simulación de envío y almacenamiento de datos.....</b>	<b>45</b>
<b>CONCLUSIONES .....</b>	<b>47</b>
<b>RECOMENDACIONES .....</b>	<b>49</b>
<b>REFERENCIAS BIBLIOGRAFICAS.....</b>	<b>51</b>

## ÍNDICE DE FIGURAS

<b>Figura1-1 Imagen de la RASPBERRY PI [1].....</b>	<b>3</b>
<b>Figura 1-2 Imagen de los Xbee [2].....</b>	<b>3</b>
<b>Figura 2-1 Elemento de la comunicación [3].....</b>	<b>7</b>
<b>Figura 2-2 Comunicación Alámbrica [6].....</b>	<b>8</b>
<b>Figura 2-3 Cable Coaxial [8].....</b>	<b>9</b>
<b>Figura 2-4 Cable Fibra [10].....</b>	<b>10</b>
<b>Figura 2-5 Comunicación Inalámbrica [12].....</b>	<b>11</b>
<b>Figura 2-6 Entorno de desarrollo de PYTHON [16].....</b>	<b>14</b>
<b>Figura 2-7 Instalación del PYTHON – Gpio en el Lx Terminal [18]. .....</b>	<b>15</b>
<b>Figura 2-8 X-CTU [20].....</b>	<b>16</b>
<b>Figura 2-9 RASPBERRY Pi [22].....</b>	<b>17</b>
<b>Figura 2-10 XBee [24].....</b>	<b>19</b>
<b>Figura 2-11 TP-LINK - WN725N [26].....</b>	<b>21</b>
<b>Figura 3-1 Diagrama de Bloques Simplificado.....</b>	<b>25</b>
<b>Figura 3-2 Diagrama de Bloques General.....</b>	<b>25</b>

<b>Figura 3-3 Configuración de nuestro puerto de acceso remoto con la ip de nuestra RASPBERRY PI.....</b>	<b>28</b>
<b>Figura 3-4 Encriptación WPA en un equipo wifi Tellion .....</b>	<b>28</b>
<b>Figura 3-5 Configuración de los módulos XBee S2 [29].....</b>	<b>29</b>
<b>Figura 3-6 Configuración del XBee coordinator.....</b>	<b>30</b>
<b>Figura 3-7 Configuración del XBee end device .....</b>	<b>31</b>
<b>Figura 3-8 Consola LXTerminal muestra datos recibidos.[30].....</b>	<b>32</b>
<b>Figura 3-9 Datos mostrados en la Consola LXTerminal.[30].....</b>	<b>32</b>
<b>Figura 3-10 Configuración de la red wifi mediante DHCP .....</b>	<b>33</b>
<b>Figura 3-11 Conexión a una red inalámbrica desde la RASPBERRY PI</b>	<b>34</b>
<b>Figura 3-12 Colocar la letra “N” para tener privilegios.[31] .....</b>	<b>34</b>
<b>Figura 3-13 Asignación de la IP de la Raspberry Pi.[32] .....</b>	<b>35</b>
<b>Figura 3-14 Colocación de la clave de la Raspberry Pi.[32].....</b>	<b>35</b>
<b>Figura 3-15 Instalación del acceso remoto xrdp [33].....</b>	<b>36</b>
<b>Figura 3-16 Usuario y contraseña en el acceso remoto [33].....</b>	<b>36</b>
<b>Figura 4-1 Simulación del emisor ( coordinator).....</b>	<b>38</b>
<b>Figura 4-2 Simulación del receptor ( end device) .....</b>	<b>38</b>

<b>Figura 4-3</b>	<b> simulación de la Potencia en los datos enviados.....</b>	<b> 39</b>
<b>Figura 4-4</b>	<b> Envío y recepción de datos .....</b>	<b> 40</b>
<b>Figura 4-5</b>	<b> Simulación de la PC envío y recepción de datos.....</b>	<b> 41</b>
<b>Figura 4-6</b>	<b> Ejecutando comando ifconfig.....</b>	<b> 42</b>
<b>Figura 4-7</b>	<b> Asignación de puerto 3350 .....</b>	<b> 42</b>
<b>Figura 4-8</b>	<b> Configuración conexión remota .....</b>	<b> 43</b>
<b>Figura 4-9</b>	<b> RASPBERRY PI remotamente.....</b>	<b> 44</b>
<b>Figura 4-10</b>	<b> Conexión a Internet mediante Wifi .....</b>	<b> 44</b>
<b>Figura 4-11</b>	<b> Conexión a <a href="http://www.mail.espol.edu.ec">www.mail.espol.edu.ec</a> .....</b>	<b> 45</b>
<b>Figura 4-12</b>	<b> Recepción de datos exitosa.....</b>	<b> 46</b>
<b>Figura 4-13</b>	<b> Almacenamiento de datos .....</b>	<b> 46</b>

## ABREVIATURAS

DVA	Digital Voice Announcer
GPIO	General Purpose Input/Output
USB	Universal Serial Bus
HDMI	High-Definition Multimedia Interface
CPU	Central Processing Unit
SD	Secure Digital
WEP	Wired Equivalent Privacy
WAP	Wi-Fi Protected Access
PSK	Pre Shared Key
TKIP	Temporal Key Integrity Protocol
VPN	Virtual Private Network
SH	Serial High
SL	Serial Low
DH	Data High
DL	Data Low
IP	Internet Protocol

## RESUMEN

El objetivo de este proyecto es de entregar una guía práctica, y un material indispensable para todo estudiante que desee conocer sobre la Tecnología RASPBERRY PI.

Este proyecto se basa en una red de comunicación XBee con una minicomputadora RASPBERRY PI, implementando una comunicación wifi para el almacenamiento de información en una base de datos remota.

Aquí vamos a tratar puntos importantes de comunicación de la RASPBERRY PI que son: wifi, VPN y procesamiento de datos mediante los módulos XBee.

Los problemas implementados se realizaron mediante el programa PYTHON, en lenguaje C para la RASPBERRY PI y X-CTU para los XBee, y su implementación física se presenta mediante placas electrónicas para hacerlo muy didáctico.

## INTRODUCCIÓN

La tecnología ha ido evolucionando y con ella se han dado muchas mejoras en los diferentes campos de la misma. En este trabajo se implementara una red de comunicación XBee con la mini computadora RASPBERRY PI y se manipulara procesamiento y almacenamiento de datos, esperando que esta guía sea una herramienta para el desarrollo de la tecnología.

En este proyecto se concentra en el tema de las comunicaciones, desarrollando tres aplicaciones, de las cuales una es desarrollada en lenguaje C, los XBee en procesos de Test/Query y el wifi en líneas de comandos del lx terminal de la RASPBERRY PI.

Este trabajo se divide en una gama de cuatro capítulos los cuales serán descritos en base a la teoría y aplicación en el tema de las comunicaciones, considerando las especificaciones y aplicaciones de la RASPBERRY PI.

# **CAPÍTULO 1**

## **GENERALIDADES**

En este capítulo se realizará el planteamiento del proyecto. Se expone las ideas y se explica brevemente cómo implementar la solución a dicho problema.

### **1.1 Objetivos**

#### **1.1.1 Objetivos Generales**

- Los objetivos generales para realizar este proyecto son los siguientes:
- Mencionar los conceptos básicos acerca de la comunicación y sus



medios.

- Aprender acerca de la minicomputadora RASPBERRY PI su funcionamiento y sus aplicaciones.
- Estudiar módulos XBee, para su configuración.
- Desarrollo de un código en lenguaje C, para el almacenamiento de información, mediante la elaboración de una base de datos.

## **Objetivos Específicos**

El objetivo principal de este proyecto es generar una comunicación remota para almacenar información mediante una base de datos.

## **Antecedentes**

Debido al desarrollo de la tecnología y de las múltiples necesidades como de adquirir un ordenador a bajos costos la fabricación de esta minicomputadora se ajusta a las necesidades de esta época y nos ayuda a solucionar nuestro proyecto mediante sus aplicaciones.

En el año 2009 como una Fundación de caridad se registró “Raspberry Pi Foundation” que impulsó el estudio a las ciencias computacionales, su logotipo está basado en una buckyball que fue seleccionado por varios miembros de la comunidad [1].

## Identificación del Problema



Figura1-1 Imagen de la RASPBerry PI [1]



Figura 1-2 Imagen de los Xbee [2]

El propósito de este trabajo es mostrar algunas aplicaciones mediante el uso de la RASPBerry PI como se muestra en la Figura 1-1. A esto se suma los módulos XBee como se muestra en la Figura 1-2 que transmiten y reciben la información.

## Descripción del Proyecto

Este proyecto se basa en una red de comunicación XBee con una minicomputadora RASPBERRY PI.

Los XBee transmitirán datos inalámbricamente, la cual uno será emisor y el otro receptor. El XBee receptor interactuará con la minicomputadora RASPBERRY PI mediante el puerto USB de la tarjeta, la cual se instalará y configurará dicho puerto.

La configuración de la tarjeta RASPBERRY PI se la hará mediante dos software, llamadas lx terminal de sistema operativo de la tarjeta y PYTHON, que es un programa que se lo instalará mediante líneas de comandos en el lx terminal de la tarjeta, la cual nos permitirá trabajar los puertos GPIO de la minicomputadora RASPBERRY PI.

En el lx terminal se ejecutará los comandos deseados para instalar nuestra antena wifi, el acceso remoto y el puerto USB para el módulo XBee. Las líneas de comandos están basadas a la programación del Sistema Operativo Linux.

La antena wifi transmitirá datos de la minicomputadora a una computadora que esté conectado a la misma red. Los datos transmitidos de la RASPBERRY PI a un computador serán mediante acceso remoto. Dicho datos serán almacenados en la base de datos del computador, modificados, y guardados si el usuario lo desea.

## Limitaciones

Como todo producto, y elementos tiene sus limitaciones; vamos a mencionar dichas limitaciones de la minicomputadora RASPBERRY PI y el módulo XBee.

Unas de las principales limitaciones de la RASPBERRY PI es que comprende de dos puertos USB, la cual nos permite solo conectar dos dispositivos de entrada o salida a la tarjeta, pero cada puerto USB está disponible para el mouse y el teclado. Si se requiere conectar otro dispositivo a la tarjeta tenemos que utilizar el HUB USB para conectar otros elementos.

La tarjeta no incluye disco duro y posee una memoria RAM de 256MB, esto indica que no es un buen procesador para poder ejecutar y trabajar varios programas a la vez.

No posee puerto DVA o DVI para conectar un monitor directamente a la tarjeta, la cual se necesita un conector HDMI-DVI para conectar el monitor.

No puede instalarse ningún otro sistema operativo, este ordenador posee su propio sistema operativo linux. Si alguien de los usuarios puede portar una versión de Android para funcionar con 256MB o 512MB de RAM, entonces correrá en la RASPBERRY PI.

XBee.

- Tiene alcance hasta 100metros de radio.
- Trabajan a la misma velocidad en baudios (emisor y receptor).
- Trabajan en todas las radio frecuencia.

## **CAPÍTULO 2**

### **FUNDAMENTO TEORICO**

Se describe los fundamentos teóricos básicos de las herramientas de software y hardware utilizados para la implementación de este proyecto.

#### **2.1.- Comunicación**

##### **2.1.1. Sistema de Comunicación**

Al empezar hablar de comunicación wireless, es necesario mencionar conceptos básicos que hemos aprendido en cursos anteriores.

Un sistema de comunicación está formado de emisor, receptor y canal:

- Emisor es el que se encarga de transmitir la información.

- Receptor se encarga de recibir la información.
- Canal es el medio por el cual viaja la información entre el emisor y el receptor.

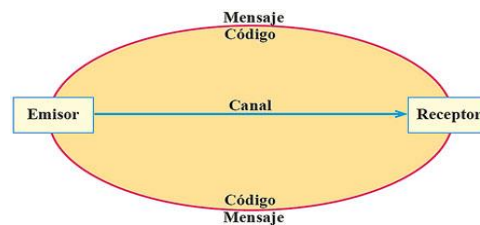


Figura 2-1 Elemento de la comunicación [3]

La información que viaja entre el emisor y el receptor debe adaptarse al canal de transmisión, esto implica la necesidad de disponer de un soporte adecuado por el cual pueda transmitir la información. Actualmente los sistemas de comunicación utilizan dos tipos de soporte que permite hablar de dos medios diferentes de comunicación [4]:

- Comunicación Alámbrica
- Comunicación Inalámbrica

## 2.1.2 Medios de Comunicación

### 2.1.2.1 Comunicación Alámbrica

Es aquella que se transmite por medio de cables que une al emisor y al receptor, o quizás dispositivos electrónicos que se conectan entre sí.

Si este medio consta solo de cable, el medio de comunicación es llamado pasivo, caso contrario si depende de algún dispositivo que amplifique o module la señal el medio se llama activo. [5]

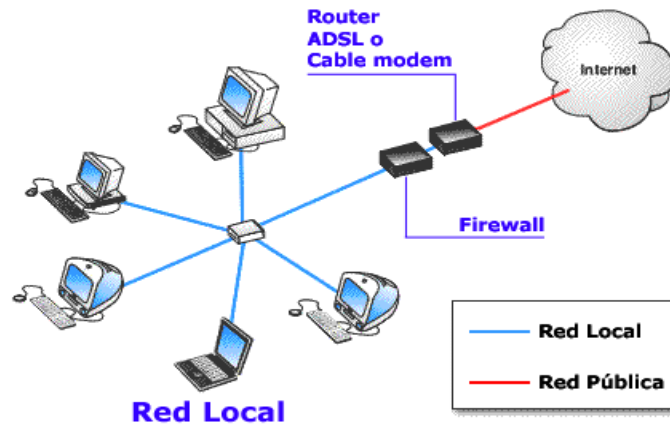


Figura 2-2 Comunicación Alámbrica [6]

## Tipos de Cable

Los tipos de cable más comunes en la transmisión de dato son: cable de cobre o cable coaxial, cable de fibra óptica.

### Cable Coaxial

Es un cable utilizado para transportar señales eléctricas de alta frecuencia, que posee dos conductores concéntricos, uno central llamado vivo que lleva la información y otro exterior de aspecto tubular llamado malla que tiene como referencia de tierra y retorno de las corrientes. [7]

Características:

- RG-58/U: Núcleo de cobre sólido.
- RG-58 A/U: Núcleo de hilos trenzados.

- RG-59: Transmisión en banda ancha (TV).
- RG-6: Mayor diámetro que el RG-59 y considerado para frecuencias más altas que este, pero también utilizado para transmisiones de banda ancha.
- RG-62: Redes ARCnet.

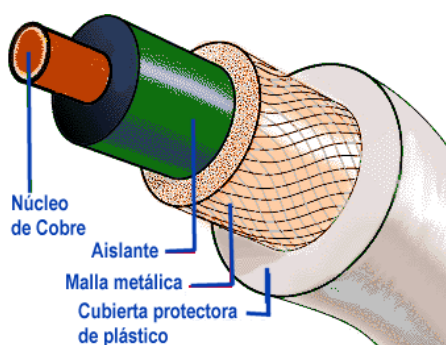


Figura 2-3 Cable Coaxial [8]

## Fibra Óptica

Es un medio de transmisión utilizado en redes de datos; Un hilo muy fino de material transparente, vidrio o material plástico por el cual se envía haz de luz que son representado en datos a transmitir. Actualmente es muy usado en el campo de las telecomunicaciones, ya que permite enviar gran cantidad de datos con gran velocidad que superan al cable convencional. También se utilizan para redes locales donde necesita aprovechar las ventajas de la fibra óptica sobre otros medios de transmisión. [9]

Características:

- Cobertura más resistente: La cubierta contiene un 25% más material que las cubiertas convencionales. [9]



- Uso dual (interior y exterior): La resistencia al agua y emisiones ultravioleta, la cubierta resistente y el funcionamiento ambiental extendido de la fibra óptica contribuyen a una mayor confiabilidad durante el tiempo de vida de la fibra. [9]
- Mayor protección en lugares húmedos: Es resistente a la humedad en el interior de la fibra con múltiples capas de protección alrededor de ésta, lo que proporciona a la fibra, una mayor vida útil y confiabilidad en lugares húmedos. [9]
- Empaquetado de alta densidad: Con el máximo número de fibras en el menor diámetro posible se consigue una rápida y fácil instalación, donde el cable debe enfrentar pliegues agudos y espacios estrechos. Se ha llegado a conseguir un cable con 72 fibras de construcción súper densa cuyo diámetro es un 50% menor al de los cables convencionales. [9]

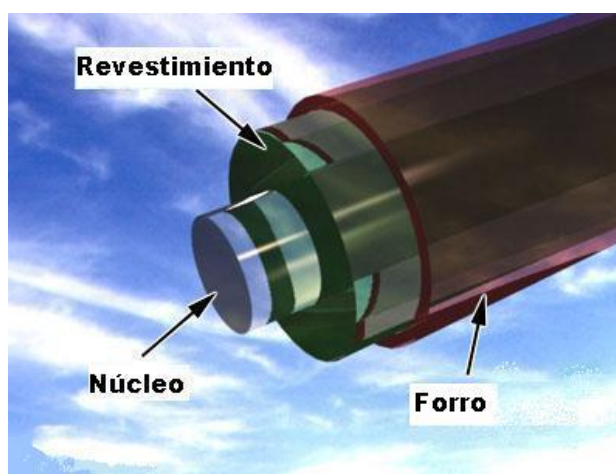
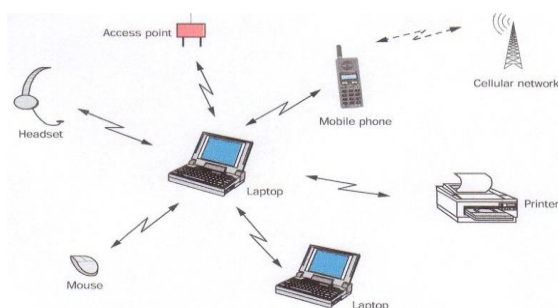


Figura 2-4 Cable Fibra [10]

### 2.1.2.2 Comunicación Inalámbrica

La comunicación inalámbrica es la que se transmite por medio de ondas de radio, no se encuentran unidos por medio de propagación físico. Los dispositivos físicos solo están presente en los emisores y receptores de la señal, entre ellos están: antenas, computadoras portátiles, PDA, teléfonos móviles. [11]

El rol de la tecnología microonda es obtener en un futuro sistemas de comunicación inalámbrica. [12]



**Figura 2-5 Comunicación Inalámbrica [12]**

Algunos aspectos en dicha tecnología utilizan ondas de radio frecuencia de baja potencia y una banda específica de uso libre o privada para transmitir entre dispositivos. [13]

La utilidad de las redes inalámbricas sin necesidad de licencia para conectar ha propiciado que el número de equipos especialmente las portátiles haya crecido notablemente.

Básicamente, se necesitan dos ítems en una conexión inalámbrica: un hotspot y un dispositivo con capacidad de comunicación wireless. Hotspot es el punto de acceso por el cual se transmite la señal. Actualmente, todas las notebooks vienen con un sistema de comunicación wireless

integrado. Además, muchas casas, oficinas, establecimientos comerciales, restaurantes, librerías, etc., poseen sus propias redes wireless, posibilitando así la conexión a internet sin ningún tipo de cableado, lo que significa un gran adelanto en términos de versatilidad y eficiencia. [14]

La tecnología wireless se presenta como una alternativa a las redes convencionales, ya que posibilita las mismas funcionalidades pero de una forma flexible, de fácil configuración y con buena conectividad. [14]

## **2.2 Herramientas de Software Para el Desarrollo del**

### **Proyecto**

Para el desarrollo del Proyecto se utilizara las siguientes herramientas de Software:

**PYTHON**, Entorno de programación para minicomputadora RASPBERRY PI.

**X-CTU**, Software para programar módulos XBee.

#### **2.2.1 PYTHON**

Para obtener el máximo rendimiento de una computadora y para obtener la máxima flexibilidad, hay muchos lenguajes de programación informática, técnicas para la RASPBERRY PI. Para nuestra aplicación nos basamos en programa PYTHON. Este programa es gratuito porque posee una licencia de código abierto, denominada PYTHON Software Foundation License, es compatible con la licencia pública general de GNU se lo puede acceder mediante el website [www.python.org](http://www.python.org) o mediante líneas de comandos en el lx terminal “sudo apt-get install python-dev. Utilizamos este programa porque

nos interesa trabajar con el puerto GPIO de nuestra tarjeta. Posee fácil manejo comparando con programas vistos en cursos anteriores.

PYTHON apoya a diseñar, desarrollar, depurar y comprobar el proceso de nuestro problema planteado. [15]

### **2.2.1.1 Descripción general de PYTHON**

Se trata de un lenguaje de programación multiparadigma, que soporta orientación a objetos, programación imperativa, y funcional. Es un lenguaje interpretado, usa tipado dinámico y es multiplataforma. [15]

PYTHON proporciona herramientas para la administración de proyectos, edición de archivo fuente, simulación del chip e interfaz para emulación In-circuit para la poderosa familia Arduino y RASPBERRY PI. [15]

PYTHON consiste de muchas ventanas y sub-módulos. Cada ventana apoya a las partes del trabajo que se intenta emprender. En la Figura 2.6 se puede apreciar las ventanas principales. [15]

Los usuarios de PYTHON hacen una analogía a la filosofía de Unix, también incluye un modo interactivo en la cual se escribe las instrucciones en una especie de intérprete de comandos. [15]

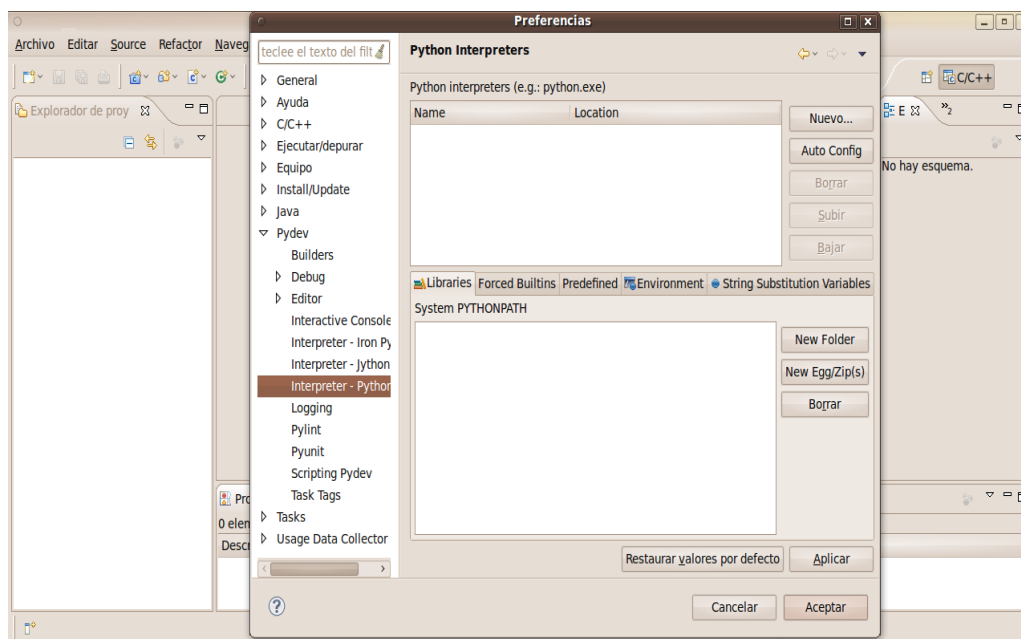
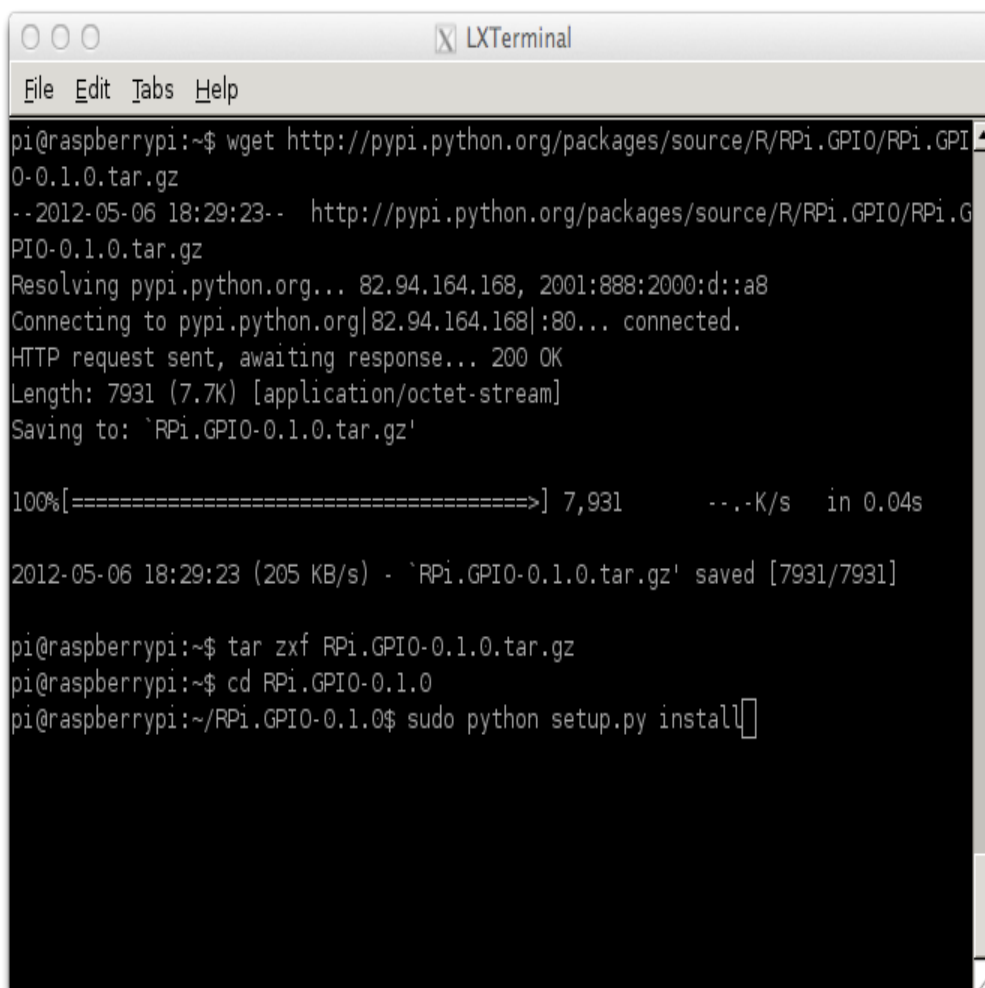


Figura 2-6 Entorno de desarrollo de PYTHON [16]

### 2.2.1.2 Simulación de PYTHON.

PYTHON es un simulador para la arquitectura y dispositivos de Arduino y RASPBERRY PI. Este simula la CPU, incluyendo todas las instrucciones, y la mayoría de los módulos de I/O del chip dependiendo de la tarjeta a utilizar. En la figura 2.7 se muestra la instalación de PYTHON GPIO. [17]



```
pi@raspberrypi:~$ wget http://pypi.python.org/packages/source/R/RPi.GPIO/RPi.GPIO-0.1.0.tar.gz
--2012-05-06 18:29:23-- http://pypi.python.org/packages/source/R/RPi.GPIO/RPi.GPIO-0.1.0.tar.gz
Resolving pypi.python.org... 82.94.164.168, 2001:888:2000:d::a8
Connecting to pypi.python.org[82.94.164.168]:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 7931 (7.7K) [application/octet-stream]
Saving to: `RPi.GPIO-0.1.0.tar.gz'

100%[=====>] 7,931      --.-K/s   in 0.04s

2012-05-06 18:29:23 (205 KB/s) - `RPi.GPIO-0.1.0.tar.gz' saved [7931/7931]

pi@raspberrypi:~$ tar xzf RPi.GPIO-0.1.0.tar.gz
pi@raspberrypi:~$ cd RPi.GPIO-0.1.0
pi@raspberrypi:~/RPi.GPIO-0.1.0$ sudo python setup.py install
```

Figura 2-7 Instalación del PYTHON – Gpio en el Lx Terminal [18].

X-CTU este software sirve para configurar módulos XBee para que se comuniquen entre ellos (envían y reciben información). [19]

En la figura 2.8 se muestra la pantalla para configurar un módulo XBee.

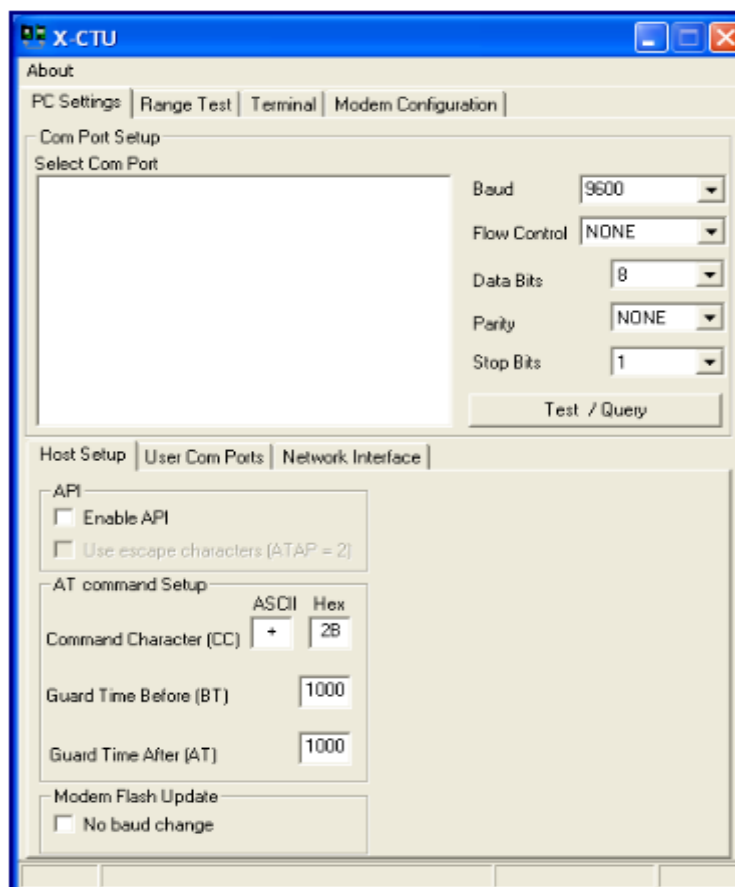


Figura 2-8 X-CTU [20]

## 2.3 Herramientas de Hardware para la Implementación del Proyecto

RASPBERRY PI y sus accesorios a utilizar, es el hardware para el desarrollo del proyecto, que describiremos posteriormente.

Xbee, módulos de transmisión y recepción de datos.

TP-LINK adaptador inalámbrico de wifi.

### 2.3.1 RASPBERRY PI

RASPBERRY PI es una minicomputadora de bajo costo desarrollada en Reino Unido con el objetivo de enseñar ciencias informáticas. Una de las características importantes es que no incluye un disco duro de estado sólido, ya que usa una tarjeta SD para el almacenamiento permanente. [21]

Uno de los beneficios es que no incluye fuente de alimentación o carcasa; sus costos son bajos en el mercado. Usualmente los programas que utilizamos con esta tarjeta poseen licencia gratuita debido al que sistema operativo es Linux. [21]

RASPBERRY PI se originó gracias al avance del Microcontrolador Atmel ATmega644. En la figura 2.1 se muestra el hardware de dicha tarjeta.[21].



Figura 2-9 RASPBERRY Pi [22]

El lanzamiento de la tarjeta fue en el 2009 con el modelo A, Luego de algunos años se trabajó en una versión mejorada lanzando el modelo B.

### 2.3.2 XBee

XBee es también llamado comercialmente Digi International, son módulos de comunicación. Los XBee primero fueron introducidos bajo la



marca MaxStream en el 2005 y se basaron en el estándar 802.15.4 del 2003, diseñado para aplicaciones de comunicación punto a punto y punto a multipunto. [23].

**Comunicación punto a punto:** realizar esta comunicación es una tarea sencilla, porque los módulos ya vienen con la configuración predeterminada.

**Comunicación punto a multipunto:** esta comunicación se genera con un coordinador que opera como central en toda red, tiene la capacidad de almacenar varios mensajes en espera que los módulos correspondientes los entreguen [23].

El módulo XBee con antena externa es más potente, y cuenta con un receptor más sensible con lo que logra distancias superiores [23]

#### Especificaciones Técnicas

- Led indicador de encendido (3V3), RSS, TX y RX
- Bus de 4 pines para alimentación y comunicaciones serial.
- Transferencia de datos 300 Baud hasta 3MBaud
- Soporta 7 a 8 bits datos, 1 o 2 bits stop, y odd/even/mark/space/no parity

Unos de los beneficios es que tiene un alto rendimiento que requiere baja latencia y sincronización de la comunicación, también es útil en potencias bajas, posee un bajo costo. En el siguiente contexto podemos visualizar la figura de un XBee. [23].



Figura 2-10 XBee [24]

## Pines del XBee

Pin #	Name	Direction	Description
1	VCC	-	Power supply
2	DOUT	Output	UART Data Out
3	DIN / CONFIG	Input	UART Data In
4	DO8*	Output	Digital Output 8
5	RESET	Input	Module Reset (reset pulse must be at least 200 ns)
6	PWM0 / RSSI	Output	PWM Output 0 / RX Signal Strength Indicator
7	PWM1	Output	PWM Output 1
8	[reserved]	-	Do not connect
9	DTR / SLEEP_RQ / DI8	Input	Pin Sleep Control Line or Digital Input 8
10	GND	-	Ground
11	AD4 / DIO4	Either	Analog Input 4 or Digital I/O 4
12	CTS / DIO7	Either	Clear-to-Send Flow Control or Digital I/O 7
13	ON / SLEEP	Output	Module Status Indicator
14	VREF	Input	Voltage Reference for A/D Inputs
15	Associate / AD5 / DIO5	Either	Associated Indicator, Analog Input 5 or Digital I/O 5
16	RTS / AD6 / DIO6	Either	Request-to-Send Flow Control, Analog Input 6 or Digital I/O 6
17	AD3 / DIO3	Either	Analog Input 3 or Digital I/O 3
18	AD2 / DIO2	Either	Analog Input 2 or Digital I/O 2
19	AD1 / DIO1	Either	Analog Input 1 or Digital I/O 1
20	AD0 / DIO0	Either	Analog Input 0 or Digital I/O 0

## NOTAS DE DISEÑO.

- Conexiones mínimas: VCC, GND, DOUT y DIN

- Conexiones mínimas para la actualización de firmware: VCC, GND, DIN, DOUT, RTS y DTR
- Dirección de señal, se especifica con respecto al módulo
- El módulo incluye una resistencia pull-up conectada a RESET
- Varias de las entradas de pull-ups se pueden configurar mediante el comando PR
- Los pines no utilizados deben dejarse desconectado

### **2.3.2.1 Aplicaciones**

- Interface de comunicación serial para niveles TTL con módulo XBee.
- Adquisición y envío de datos seriales mediante módulo XBee

### **2.3.3 Wifi TP-LINK**

El adaptador TL-WN725N permite que los usuarios conecten una computadora de escritorio o portátil a una red inalámbrica a 150Mbps. Este adaptador en miniatura está diseñado para que sea lo más convenientemente posible. [25].

Una de las características de este adaptador es que trabajan sin inconveniente con sistema operativo Linux. [25].

Especificaciones Técnicas

- Diseño liso y brillante en miniatura tan pequeño que una vez conectado, puede dejarlo en un puerto USB de la Laptop
- Rápida transmisión de velocidad de hasta 150Mbps ideal para el streaming de video o llamadas por internet
- Seguridad avanzada: Soporta 64/128 WEP, WPA, PA2/WPA-PSK/WPA2-PSK(TKIP/AES)
- Conéctese de inmediato con la fácil utilidad de instalación en 14 idiomas
- Dimensiones (An x P) 0.73x0.59x0.28in. (18.6x15x7.1mm)
- Peso 0,07 oz / 2,1 gramos (sin embalaje)
- Antena interna
- Funciones inalámbricas
- Wireless de normas IEEE 802.11b, IEEE 802.11g, IEEE 802.11n
- Frecuencia 2.400-2.4835GHz
- Tasa de señal 11b: Hasta 11Mbps (dinámico)
- 11g: Hasta 54Mbps (dinámico)

En la siguiente figura podemos observar físicamente el modelo de un adaptador wifi de marca Tp-link modelo TL-WN725N



Figura 2-11 TP-LINK - WN725N [26]

## Otros

<b>Certificación</b>	CE, FCC, IC, RoHS
<b>Contenido del Paquete</b>	Adaptador USB Nano Inalámbrico N de 150Mbps, TL-WN725N CD de Recursos Guía de Rápida Instalación
<b>Requisitos del Sistema</b>	Windows 7(32/64bits), Windows Vista(32/64bits), Windows XP(32/64bits)
<b>Ambiente</b>	Temperatura de Operación: 0°C~40°C (32°F~104°F) Temperatura de Almacenamiento: -40°C~70°C (-40°F~158°F) Humedad de Operación: 10%~90% sin condensación Humedad de Almacenamiento: 5%~90% sin condensación

### 2.4.3 Conexión Remota

La conexión remota es una operación que realiza en un ordenador a través de una red de ordenadores, como si se tratara de una conexión local o externa. [27].

En el acceso remoto se ven implicados protocolos para la comunicación entre maquinas, en ambos ordenadores se permite recibir y enviar datos. Además se debe de contar con un fuerte sistema de seguridad.

Existen múltiples programas como el VPN que permite controlar una computadora remotamente. [27].

#### 2.4.3.1 VPN

El VPN es una tecnología de red que permite una extensión segura de la red local sobre una red pública o no controlada como Internet. Permite que la computadora en la red envíe y reciba datos sobre redes compartidas o

públicas como si fuera una red privada con toda la funcionalidad, seguridad y políticas de gestión de una red privada. Esto se realiza estableciendo una conexión virtual punto a punto mediante el uso de conexiones dedicadas, encriptación o la combinación de ambos métodos. [28].

Especificaciones Técnicas.

- Autenticación y autorización: ¿Quién está del otro lado? Usuario/Equipo y qué nivel de acceso debe tener.
- Integridad: de que los datos enviados no han sido alterados. Para ello se utiliza funciones de Hash. Los algoritmos de hash más comunes son los Message Digest (MD2 y MD5) y el Secure Hash Algorithm (SHA).
- Confidencialidad: Dado que sólo puede ser interpretada por los destinatarios de la misma. Se hace uso de algoritmos de cifrado como Data Encryption Standard (DES), Triple DES (3DES) y Advanced Encryption Standard (AES).
- No repudio: es decir, un mensaje tiene que ir firmado, y quien lo firma no puede negar que envió el mensaje.

## **CAPÍTULO 3**

### **DISEÑO E IMPLEMENTACIÓN**

En este capítulo se procederá con el diseño e implementación del proyecto RED DE COMUNICACIÓN XBEE ENTRE MINICOMPUTADORA RASPBERRY PI E IMPLEMENTACION CON CAPACIDAD DE COMUNICACIÓN WIFI PARA EL ALMACENAMIENTO DE INFORMACION EN BASE DE DATOS REMOTA. Para la explicación se ha elaborado un diagrama de bloques así como también su diagrama de flujo. En la figura 3-1 podemos apreciar el diagrama de bloques simplificado, el cual resume el funcionamiento básico del proyecto.

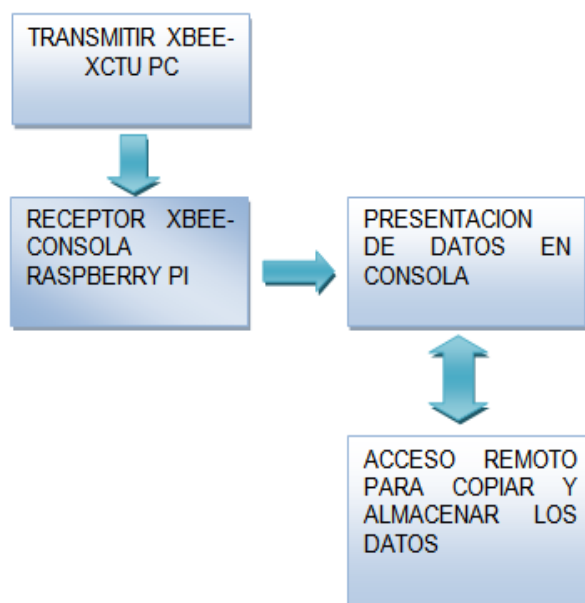


Figura 3-1 Diagrama de Bloques Simplificado

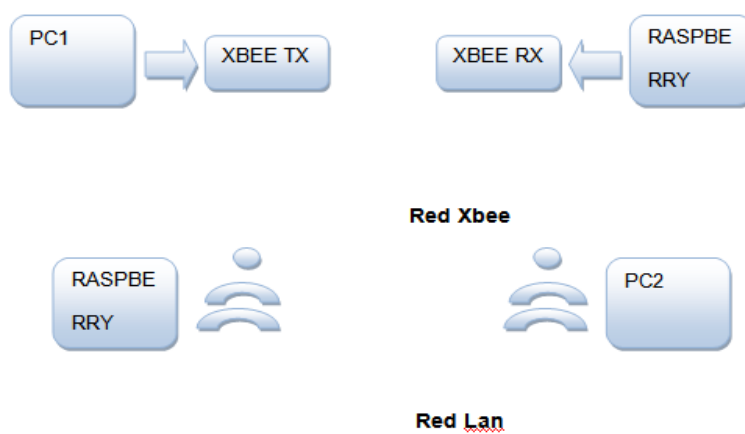


Figura 3-2 Diagrama de Bloques General



La figura 3-2 nos muestra con mayor detalle el diseño del proyecto cuya finalidad es tomar los datos recibidos que transmite el coordinador (emisor) pc1 al end device (receptor) que está conectado a la RASPBERRY PI. Por tal motivo se configura los XBee S2 y se ejecuta algunos comandos en el LXTerminal de la Raspberry Pi la cual los muestra en la consola de la minicomputadora. Otra manera de hacer las pruebas es conectar una PC los XBee y ejecutando el programa XCT-U la cual también nos muestra los datos transmitidos y recibidos. La PC2 se conecta mediante acceso remoto a la minicomputadora RASPBERRY PI para poder tomar los datos que se muestran y guardarla en una Base de Datos.

### 3.1 Actualización del Software

Se procede a configurar la RASPBERRY PI colocando una ip. Ejecutamos el siguiente comando:

```
>> sudo chmod 666/etc/network/interfaces
```

Con la línea de comando mencionada se procede a dar permiso para poder modificar el documento interfaz de red, estableciendo una ip estática.

Buscamos el directorio `/etc/network/` y modificamos el archivo `interfaces` escribiendo esto:

```
Auto eth0
iface lo inet loopback
iface eth0 inet static
address "colocamos aqui nuestra direccion ip"
netmask "colocamos aquí nuestra mascara de red"
Gateway "colocamos aquí nuestra puerta de enlace"
```

Guardamos el archivo y cerramos los permisos ejecutando el comando:

```
>> sudo chmod 644 /etc/network/interfaces
```

Definimos el servidor DNS ejecutamos los comandos:

```
>> sudo chmod 666 /etc/resolv.conf
```

```
>> sudo echo "nameserver 8.8.8.8" > /etc/resolv.conf
```

```
>> sudo chmod 644 /etc/resolv.conf
```

Y procedemos a reiniciar el servidor de red

```
>> sudo /etc/init.d/networking restart
```

Luego de tener acceso a internet procedemos a actualizar los softwares de la RASPBERRY PI con los comandos:

```
>> sudo apt-get update
```

```
>> sudo apt-get upgrade
```

Y procedemos a reiniciar la minicomputadora

```
>> sudo reboot.
```

Después de asignar una ip a la RASPBERRY PI accedemos al router de nuestra red para poder habilitar un puerto que nos permita hacer acceso remoto a nuestra RASPBERRY PI.

Source IP	<input type="text"/>
Source Port	<input type="text"/> ~ <input type="text"/>
External Port	<input type="text"/> ~ <input type="text"/>
Internal IP	<input type="text"/>
Internal Port	<input type="text"/> ~ <input type="text"/>
Protocol	TCP
Description	<input type="text"/>

**Add**

Check	Source IP	Source Port	Ext. Port	Internal IP	Internal Port	Protocol	Description
<input type="checkbox"/>			3350	172.30.1.7	3350	TCP	dem
<input type="checkbox"/>			3350	172.30.1.2	3350	TCP	

Figura 3-3 Configuración de nuestro puerto de acceso remoto con la ip de nuestra RASPBERRY PI

La RASPBERRY PI no se acopla a la encriptación o seguridad de los equipo wifi. En un router de marca Tellion acepta solo encriptación WPA, pero en un router trendnet solo acepta encriptación WEP.

Wireless
Wireless Common Settings
**DATA1 Settings**
Active Wireless Client Table

WiFi Protected Setup
DATA2 Settings
Authentication/Account Server

WMM Settings
VoIP Settings

WLAN MAC Filtering settings
WDS Settings

**DATA1 Settings**

**Active**  Enable  Disable

**SSID** Netlife-WEspinoza Hide SSID

**Encryption**  None  802.1x  WEP  WPA-PSK

**WPA Mode**  WPA  WPA2  WPA2/WPA2

**Encryption Type**  TKIP  AES  TKIP&AES

**Pre-Shared Key Format**  Passphrase  Hex

**Pre-Shared Key** 0907911820

**Broadcast Key Update**  Enable  Disable 3600 Sec

**WMM**  Enable  Disable

**Apply**

Figura 3-4 Encriptación WPA en un equipo wifi Tellion

## 3.2 Red Xbee

La red XBee se genera mediante una comunicación punto-multipunto con coordinator, se procede a instalar el X-CTU al PC1 y conectar el modulo usb XBee y configurarlo como XBee coordinator (emisor). En otra PC se configura el XBee end device (receptor) para que puede coordinarse con el receptor, se conectar a la RASPBERRY PI para que luego proceda hacer la comunicación. El emisor trabaja como XBee o router principal, es decir que si se quiere agregar otro XBee se lo tiene que configurar como “end device” para que se pueda acoplar a nuestra red XBee.[29]

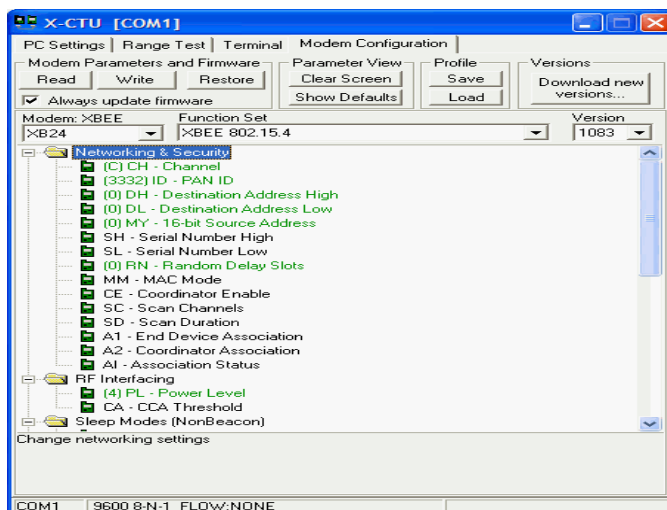


Figura 3-5 Configuración de los módulos XBee S2 [29]

### 3.2.1 Configuraciones XBee's

#### XBee COORDINATOR y AND DEVICE

En el X-CTU se procede a configurar el XBee emisor, también llamado XBee “coordinator”. Se elige la función ZIGBEE COORDINATOR AT y procedemos a configurar sus parámetros. Se coloca un ID y un canal la cual nos genera un

datos en el SH y SL (Números Seriales). Los receptores XBee tienen que llevar el mismo ID y el mismo canal del emisor para que conmuten entre ellos.

Se procede a abrir otro XCT-U para proceder a configurar el receptor, también llamado “end device”. Se elige la función ZIGBEE END DEVICE AT y procedemos a configurar sus parámetros. Colocamos el ID y el canal del XBee receptor y nos genera unos datos en SH y SL.

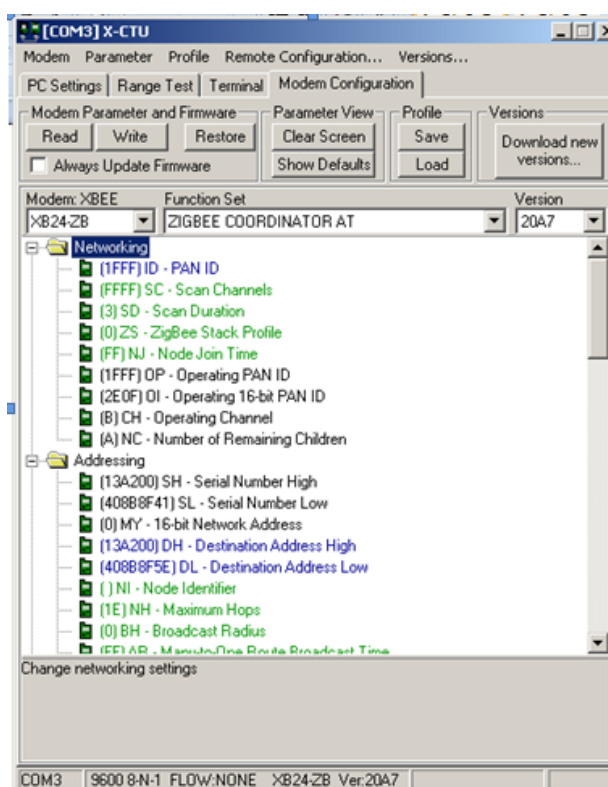


Figura 3-6 Configuración del XBee coordinator

En el DH del XBee “end device” se procede a colocar el dato de SH del XBee “coordinador”, el DL del XBee “end device” se coloca el dato de SL de XBee “coordinador”, y el mismo caso se aplica para el XBee coordinador. Después de configurar los dos XBee’s se procede a escribir (write), para que queden configurados.

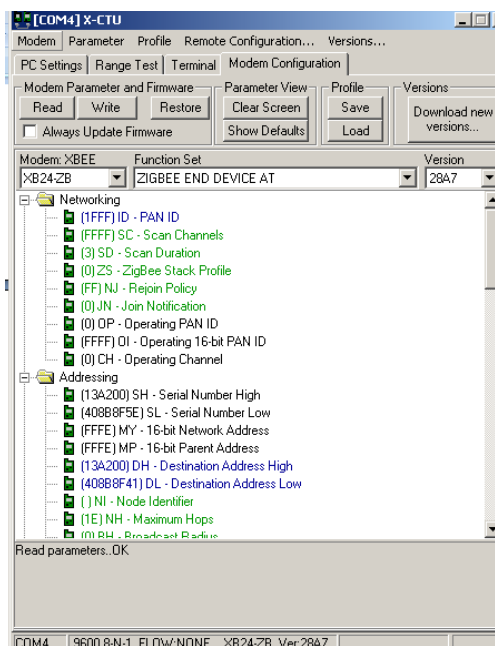


Figura 3-7 Configuración del XBee end device

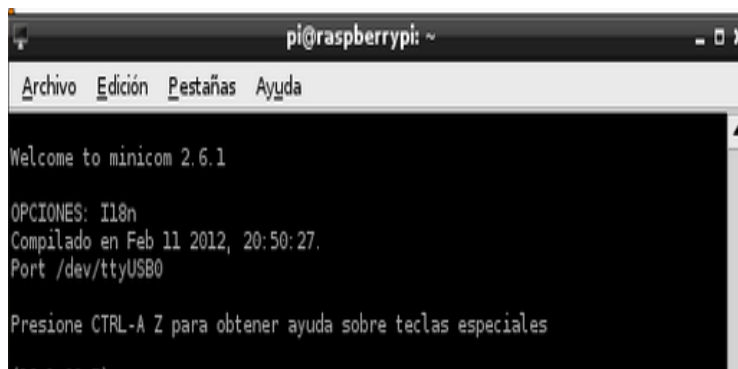
En la minicomputadora RASPBERRY PI se procede a instalar el puerto serial minicom para que puede reconocer los módulos XBee. [30]

```
>> sudo apt-get install minicom
```

Con este comando reconoce los dispositivos USB que se conecten a la RASPBERRY PI.

Después instalado el minicom se procede a conectar el USB del XBee end device y luego ejecutamos esta línea de comando:[30]

```
>> minicom -b 9600 -o -D /dev/ttyUSB0
```



```
pi@raspberrypi: ~
Archivo Edición Pestañas Ayuda

Welcome to minicom 2.6.1

OPCIONES: I18n
Compilado en Feb 11 2012, 20:50:27.
Port /dev/ttyUSB0

Presione CTRL-A Z para obtener ayuda sobre teclas especiales
```

Figura 3-8 1Consola LXTerminal muestra datos recibidos.[30]

Se recibe datos sin perdidas de paquetes en nuestra red XBee. Los datos son mostrados.



```
pi@raspberrypi: ~
Archivo Edición Pestañas Ayuda

Welcome to minicom 2.6.1

OPCIONES: I18n
Compilado en Feb 11 2012, 20:50:27.
Port /dev/ttyUSB0

Presione CTRL-A Z para obtener ayuda sobre teclas especiales

{18.1,39.7}
{18.1,39.3}
{18.1,39.2}
{18.1,39.0}
```

Figura 3-9 Datos mostrados en la Consola LXTerminal.[30]

### 3.3 Red LAN

En nuestra Red LAN, conectamos a la RASPBERRY PI una antena USB wifi para que le asigne una dirección ip y podamos realizar la transmisión de datos.

Después de haber actualizado el sistema operativo de nuestra RASPBERRY PI se procede ahora a modificar nuestro directorio `/etc/network/` en la carpeta “interface”, se ejecuta el siguiente comando para poder abrir el permiso de modificar este archivo:

```
pi@raspberrypi~$ sudo chmod 666 /etc/network/interfaces
```

Se procede a escribir lo que se muestra en la siguiente imagen.



```
<interfaces>
File Edit Search Options Help
auto lo

iface lo inet loopback
iface eth0 inet dhcp

allow-hotplug wlan0
iface wlan0 inet manual
wpa-roam /etc/wpa_supplicant/wpa_supplicant.conf
iface default inet dhcp
```

Figura 3-10 Configuración de la red wifi mediante DHCP

Si queremos asignar un ip estática a nuestro wifi de la RASPBERRY PI se modifica la última línea “dhcp” por “static” seguido por:

address “dirección ip deseada”

netmask “ mascara de la red”

gateway “puerta de enlace”

En la RASPBERRY PI incluye un programa llamada WPA\_GUI, la cual permite conectarse a una red inalámbrica.



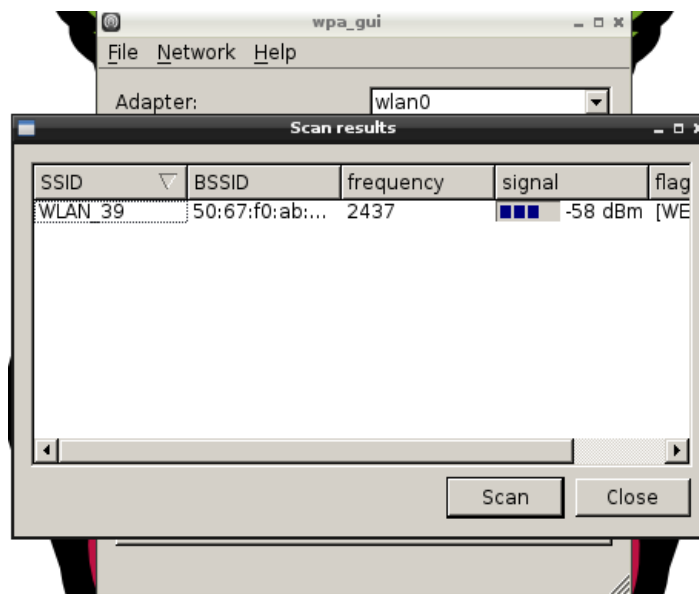


Figura 3-11 1 Conexión a una red inalámbrica desde la RASPBERRY PI

En el terminal se procede a instalar una VNC para que una PC se pueda conectar remotamente a la RASPBERRY PI.

Se ejecuta el siguiente comando:

```
>> sudo apt-get install tightvncserver
```

Luego ejecutamos el comando:

```
>> tightvncserver
```

Este comando nos sirve para poder dar privilegios de acceder a la raspberry pi mediante una contraseña que nosotros establezcamos.



Figura 3-12 Colocar la letra "N" para tener privilegios.[31]

Se instala el VNC para que podamos realizar el acceso remoto a la RASPBerry PI y podamos coger los datos que se presentan en el Ixterminal y poderlo enviar a una Base de Datos.[20]

En el VNC se escribe la dirección ip que le asigna el USB wifi seguido de “:1” y colocar la clave que se le asigno en la RASPBerry PI.[32]

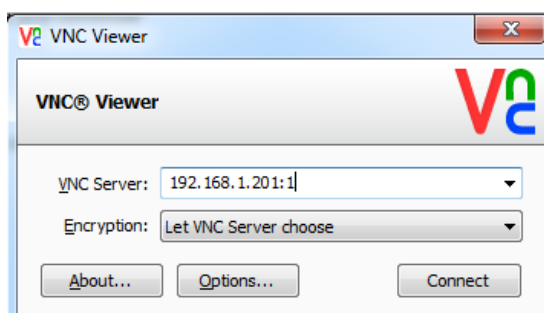


Figura 3-13 Asignación de la IP de la Raspberry Pi.[32]

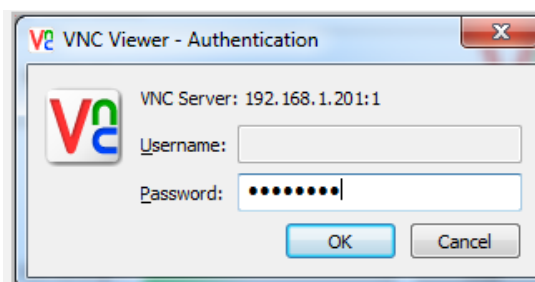
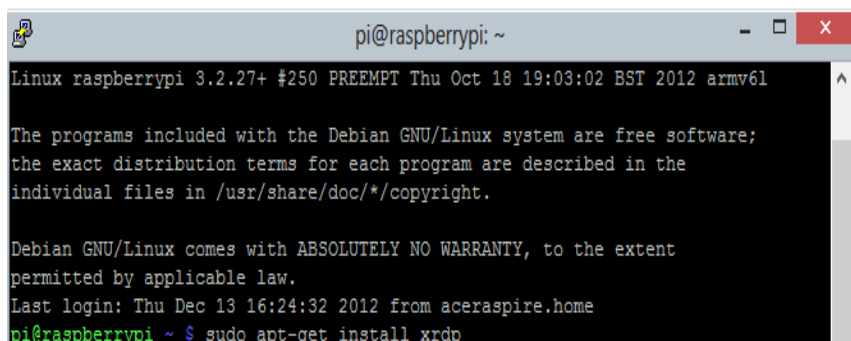


Figura 3-14 Colocación de la clave de la Raspberry Pi.[32]

El VCN no es 100% confiable para realizar acceso remoto a una máquina que está conectada mediante wifi. Nosotros procedimos a utilizar otro acceso remoto llamado “Acceso Remoto de Windows” que es confiable siempre y cuando habilitemos el puerto 3350 asignado a la ip de nuestra RASPBerry PI.

Procedemos a instalar el acceso remoto xrdp a la RASPBerry PI ejecutamos el siguiente comando para realizar la instalación: [33]

>> sudo apt-get install xrdp



```
pi@raspberrypi: ~  
Linux raspberrypi 3.2.27+ #250 PREEMPT Thu Oct 18 19:03:02 BST 2012 armv6l  
The programs included with the Debian GNU/Linux system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*/copyright.  
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent  
permitted by applicable law.  
Last login: Thu Dec 13 16:24:32 2012 from aceraspire.home  
pi@raspberrypi ~ $ sudo apt-get install xrdp
```

Figura 3-15 Instalación del acceso remoto xrdp [33]

En nuestra PC abrimos el acceso remoto de Windows y colocamos la ip asignada a nuestra RASPBERRY PI, colocamos el usuario y contraseña que le pusimos a nuestra minicomputadora.

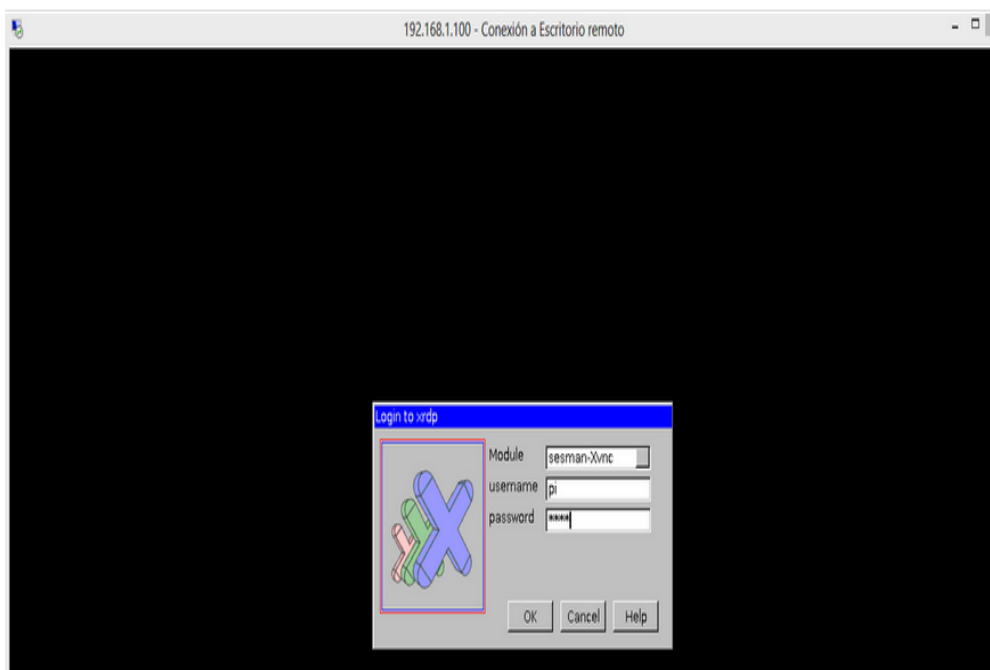


Figura 3-16 Usuario y contraseña en el acceso remoto [33]

## **CAPÍTULO 4**

### **4. Simulación y Resultados**

En este capítulo se muestra las pruebas realizadas, incluyendo la presentación por pantalla mediante una serie de pasos para transmitir datos.

#### **4.1 Simulación del proyecto desde PC A PC**

A continuación se mostrará las pantallas capturadas de las simulaciones realizadas desde PC a PC mediante el uso de módulos XBee utilizando el programa X-CTU mediante los puertos minicom generando una comunicación entre un emisor llamado coordinator y un receptor llamado end device como se muestra en la Figura 4.1 y la Figura 4.2.

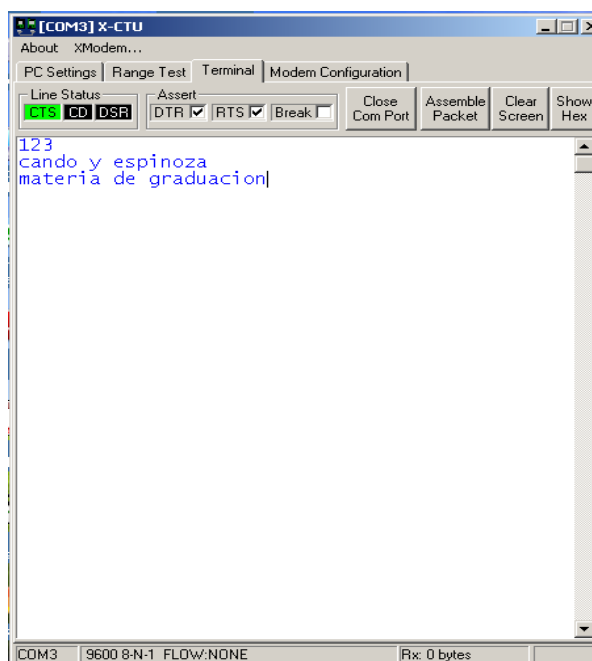


Figura 4-1 Simulación del emisor ( coordinator)

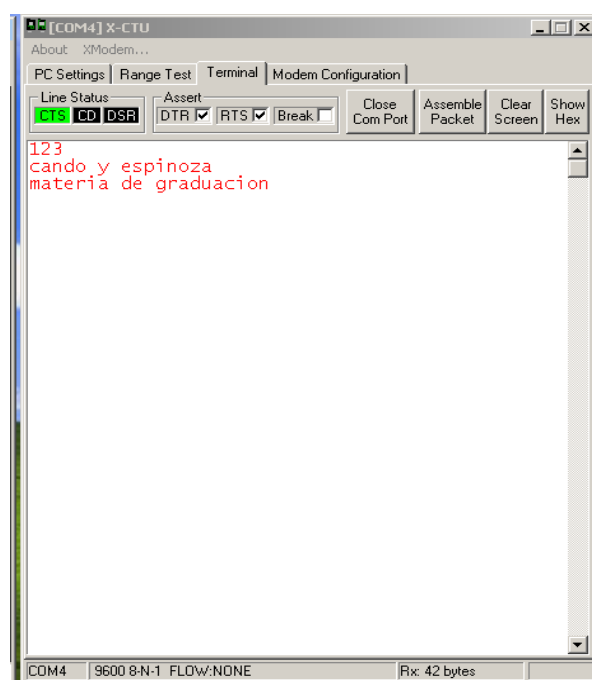


Figura 4-2 Simulación del receptor ( end device)

## 4.2 Simulación de la Potencia de datos enviados

A continuación en la Figura 4.3 se muestra la potencia de los datos que se envía, el tiempo de retardo en llegar el dato.

Hay que tomar en cuenta que en esta simulación se usa una sola PC y conectados a diferentes puertos los módulos XBee.

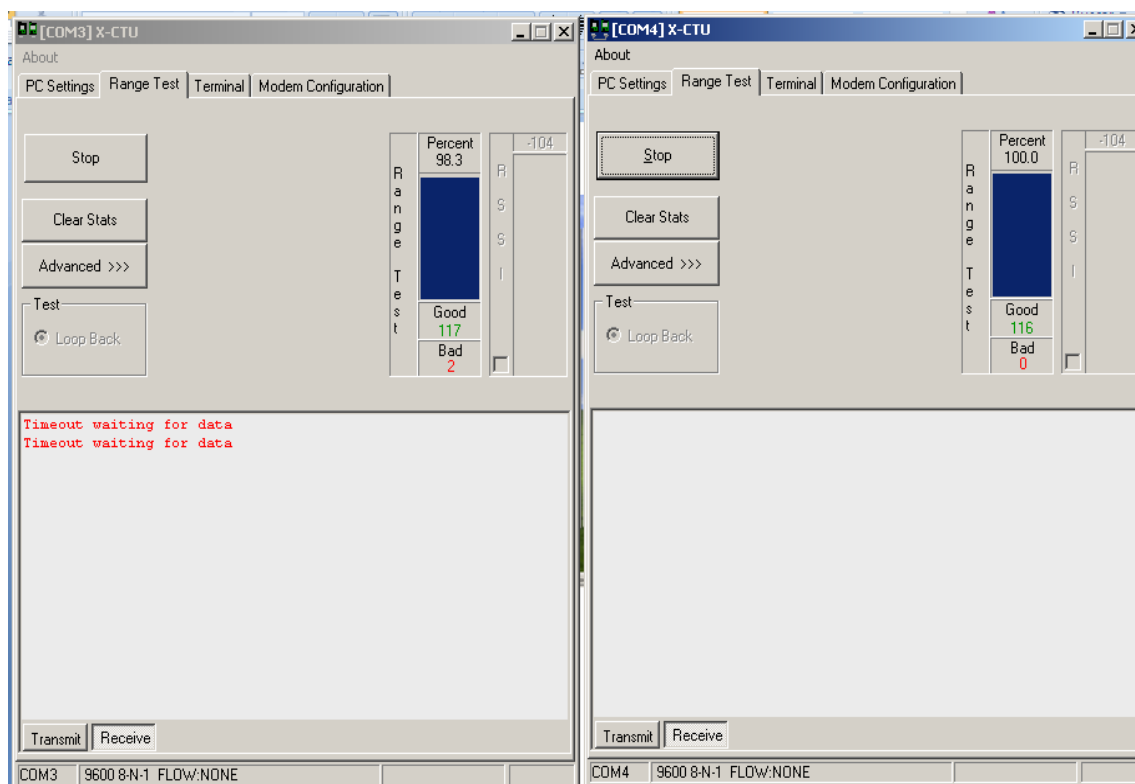


Figura 4-3 simulación de la Potencia en los datos enviados

### 4.3 Simulación desde PC – RASPBERRY PI

En la pantalla siguiente se encuentra la simulación entre una PC – RASPBERRY PI generando una comunicación mediante los puertos minicom de módulos XBee generan una comunicación, cabe recalcar que tanto como la PC y la RASPBERRY PI pueden enviar y recibir información como se muestra en la Figura 4.4 y la Figura 4.5.

```
Welcome to minicom 2.6.1
OPTIONS: I18n
Compiled on Apr 28 2012, 19:24:31.
Port /dev/ttyUSB0

Press CTRL-A Z for help on special keys
proyecto de graduacion cando espinoza
```

Figura 4-4 Envío y recepción de datos

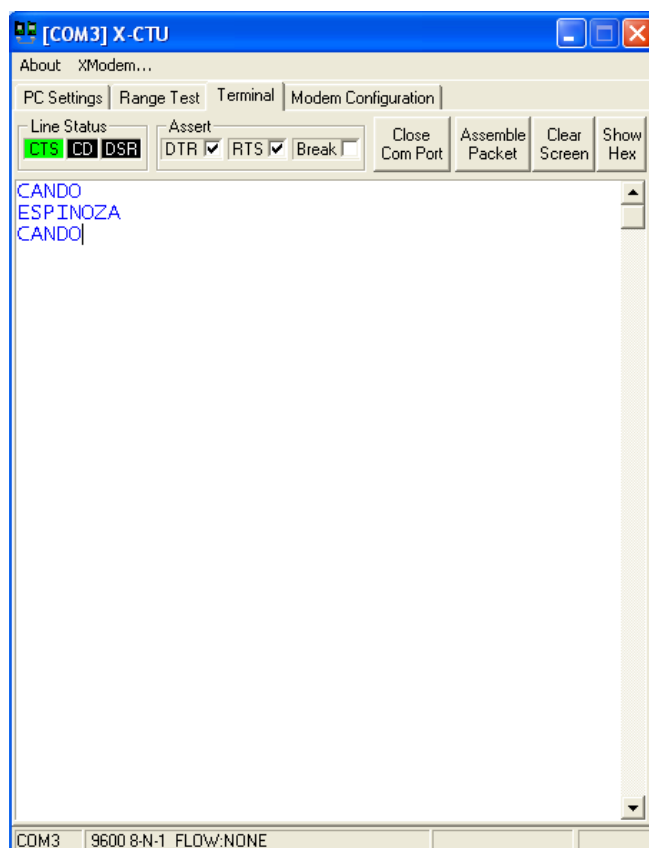


Figura 4-5 Simulación de la PC envío y recepción de datos

### 4.3 Simulación paso a paso de Conexión Remota a la RASPBERRY PI

Paso 1: En la Figura 4-6 que se muestra a continuación se observa la ejecución del comando ifconfig que nos da información de la red como dirección ip y máscara que tiene la RASPBERRY PI que se usará para la conexión remota.



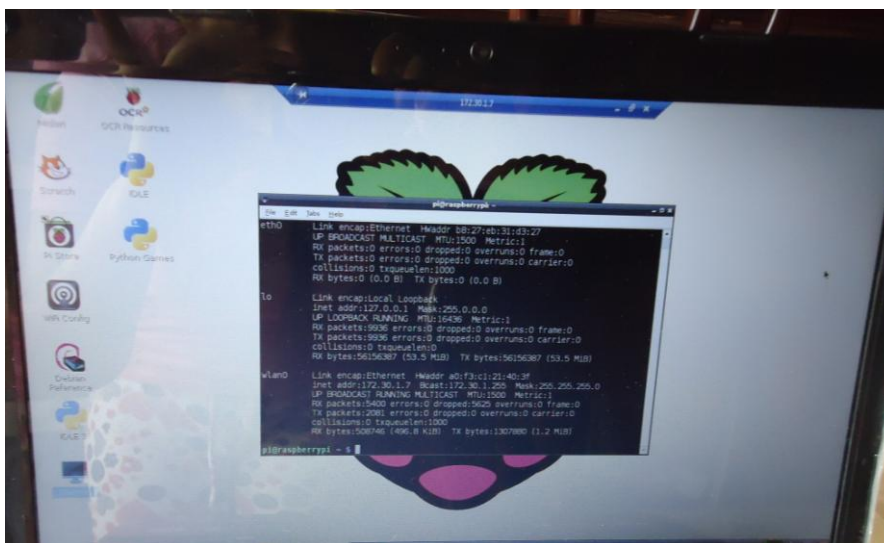


Figura 4-6 Ejecutando comando ifconfig

Paso 2: Como se muestra la Figura 4-7 se observa la asignación y habilitación de puerto 3350 por el que se hará la conexión.



Figura 4-7 Asignación de puerto 3350

Paso 3: A continuación se observa la configuración para conexión remota colocando la dirección ip por medio del puerto habilitado.

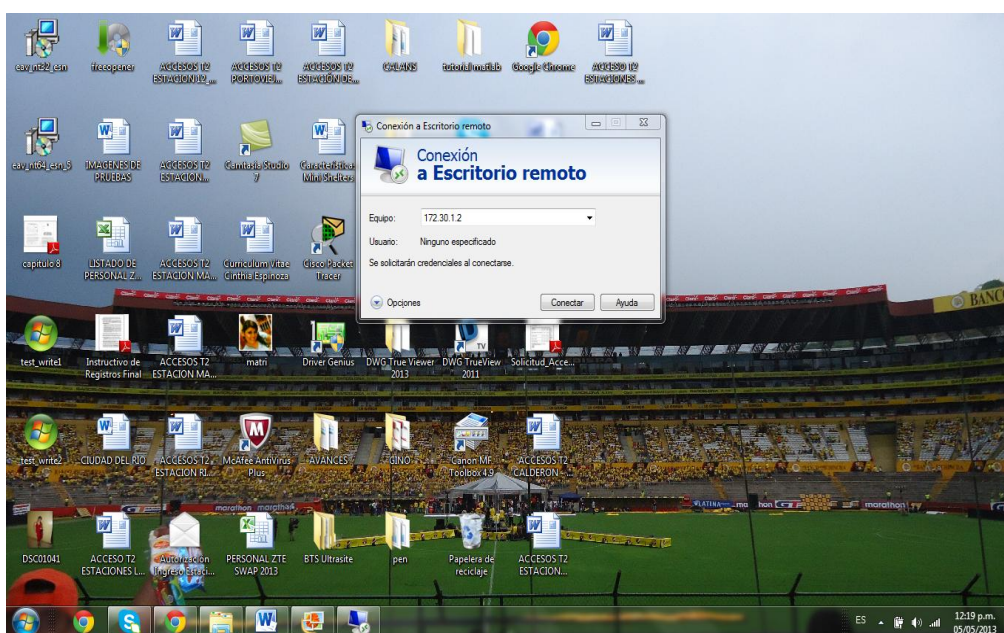


Figura 4-8 Configuración conexión remota

Paso 4: Acceso remoto a la RASPBERRY PI y conexión a internet por medio de una red wifi como se muestra a continuación.

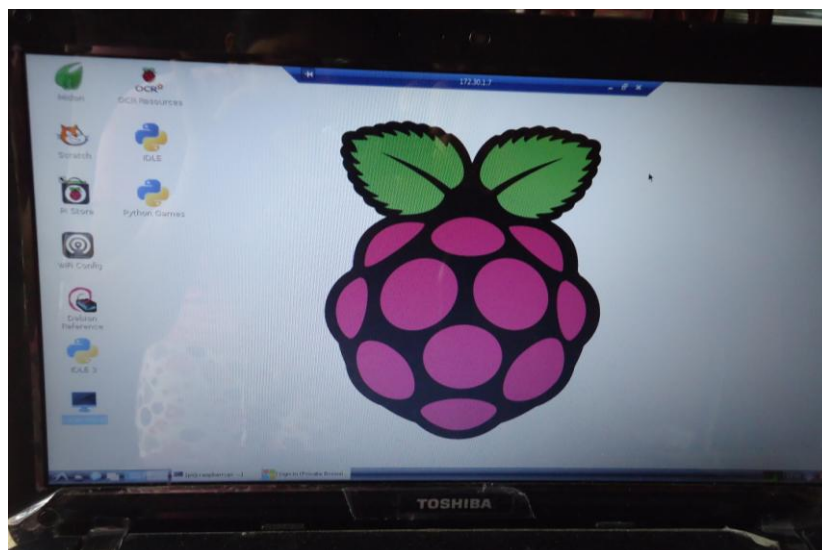


Figura 4-9 RASPBERRY PI remotamente

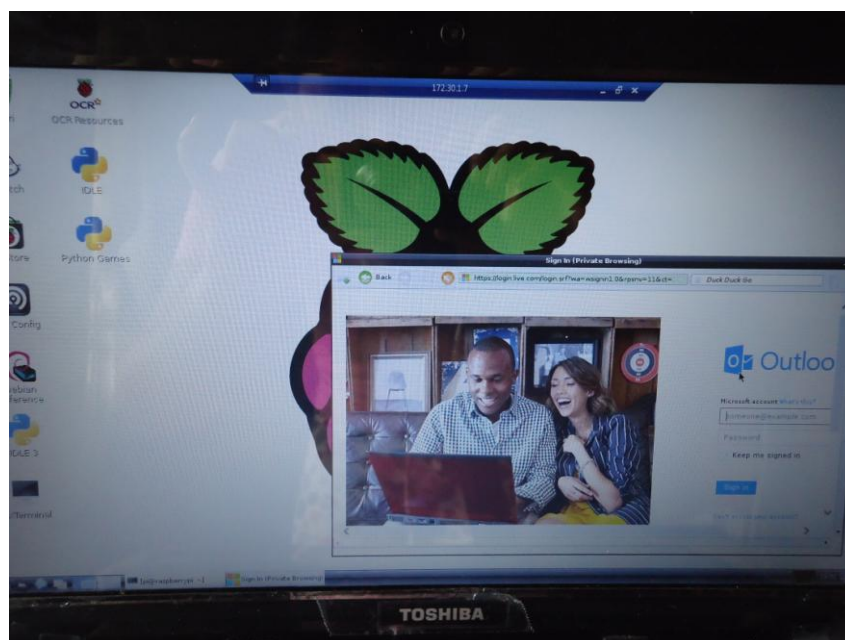


Figura 4-10 Conexión a Internet mediante Wifi

## 4.4 Simulación de envío y almacenamiento de datos

En este punto se mostrará el envío de un documento txt que fue guardado en el escritorio luego que se generó la comunicación entre los módulos XBee como se mostró en figuras anteriores, el archivo guardado será enviado por medio del correo ESPOL que tiene como destinatario a `criscand_pol88@hotmail.com`, a continuación se observan las figuras mencionadas.

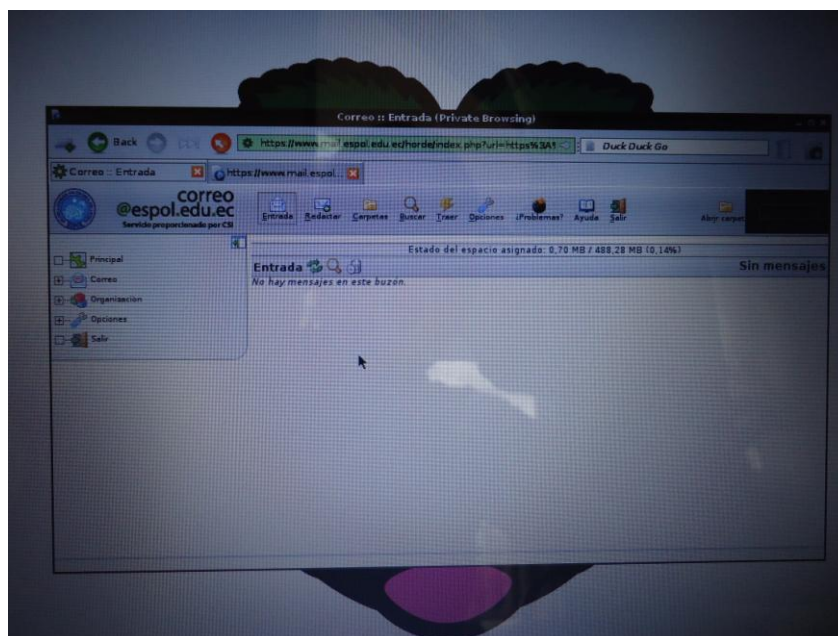


Figura 4-11 Conexión a `www.mail.espol.edu.ec`

A continuación se muestra que el documento enviado fue recibido exitosamente

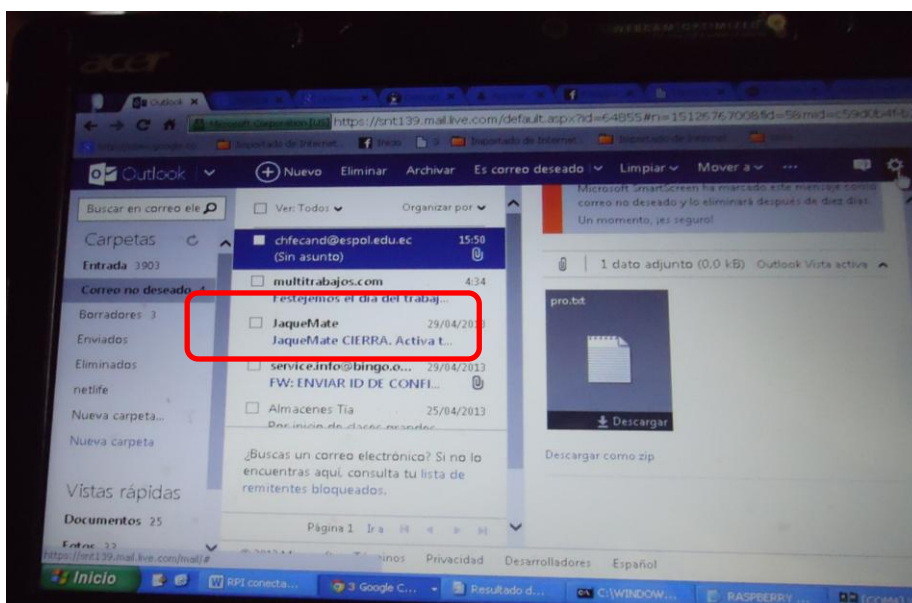


Figura 4-12 Recepción de datos exitosa

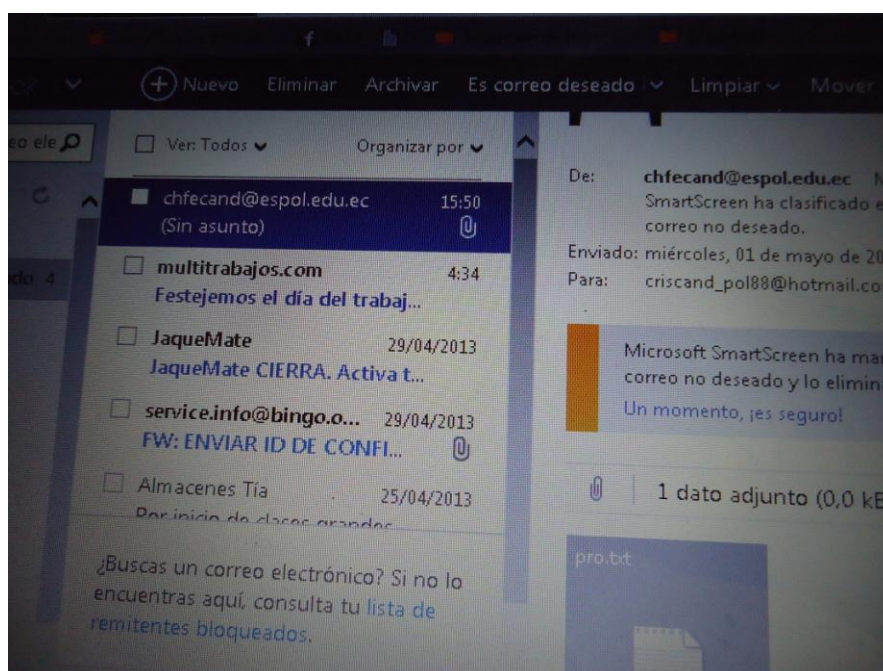


Figura 4-13 Almacenamiento de datos

## **CONCLUSIONES**

1. De la programación utilizada en la RASPBERRY PI y módulos XBee, se logró realizar y configurar una red XBee con la minicomputadora y la PC, obteniendo en la simulación de la misma resultados esperados de la transmisión de datos, tanto en la simulación como en la implementación.
2. El presente proyecto puede ser utilizado en el campo de las redes alámbricas, inalámbricas, la electrónica o cualquier otra aplicación en donde sea necesaria la visualización, de los elementos electrónicos XBee y de transmisión de datos, teniendo la ventaja de manejar e integrar una minicomputadora RASPBERRY PI puede ser adaptada

fácilmente a las condiciones que amerite el área en que se aplique el mismo.

3. Se hizo la implementación en un edificio con la finalidad de obtener información a distancia máxima de 100 metros, mediante la red XBee y poder enviar esa información remotamente a una base de datos compuesta por una red local comprendida con la minicomputadora RASPBERRY PI.
4. Para efecto de visualización, las informaciones recopiladas son enviados mediante wifi a la ventana de comando de la minicomputadora RASPBERRY PI. Estas a su vez procesadas con el programa con la ayuda de un programa de acceso remoto de Windows llamado XRDP para obtener la información en una PC por lo que concluyo que esta es una forma muy sencilla de lograr una comunicación entre una RASPBERRY PI, XBee y una Computadora para cualquier tipo de proyecto.
5. Este proyecto nos ayudó al crecimiento profesional ya que la RASPBERRY PI es usado en muchos países desarrollados por su bajo costo en comparación a una Laptop, notebook, u ordenador de escritorio. Esta placa posee múltiples aplicaciones que incentivan al estudiante a aprender ciencias computacionales mediante el uso de software libre.

## RECOMENDACIONES

1. Para una Red LAN pequeña se procede a utilizar el acceso remoto VNC en caso contrario se procede a utilizar el acceso remoto de xrdp que es utilizable para Redes LAN más grandes, siempre y cuando se habilite el puerto 3350 con la ip asignada en la RASPBERRY PI.
2. En algunos router la seguridad de encriptación es importante. Es recomendable utilizar seguridad WEP que está abierto para sistemas operativo LINUX. En el caso de los router TELLION solo se conectaba con encriptación WPA y tener solo habilitado una red wifi, ya que los router TELLION tienen dos redes wifi's integradas.
3. Cuando se trate de implementación y almacenamiento de datos es recomendable utilizar una RASPBERRY PI con una capacidad mayor a 256MB de RAM para que su procesamiento sea mucho más rápido en el momento de ejecutar varios programas.



4. Les recomiendo que en el momento de implementar o desarrollar un proyecto de bajo costo mediante una RASPBERRY PI se debe considerar la cantidad de accesorios a utilizar ya que los puertos USB son limitados y con ayuda de un hub con fuente, tenemos la facilidad de conectar varios elementos a nuestro proyecto y a su vez no exceder la carga eléctrica que entrega la RASPBERRY PI.

## BIBLIOGRAFÍA

- [1.] Beech Paul, Historia de la Raspberry pi, <http://raspberrypi.cl/historia-del-raspberry-pi/>, fecha de consulta: 18 de junio del 2013.
- [2.] 5HertzElectronica, XBee, [http://www.5hertz.com/xbee\\_unbrick.html](http://www.5hertz.com/xbee_unbrick.html), fecha de consulta: 10 de marzo del 2013.
- [3.] Abc, Elemento de la Comunicación, <http://archivo.abc.com.py/fotos/2008/04/21/080421164739763.jpg>, fecha de consulta: 10 de marzo del 2013
- [4.] Grupo Prisa, Medios de Comunicación, [http://www.kalipedia.com/tecnologia/tema/comunicaciones/comunicacion-alambrica-inalambrica.html?x=20070821kpinginf\\_50.Kes&ap=1](http://www.kalipedia.com/tecnologia/tema/comunicaciones/comunicacion-alambrica-inalambrica.html?x=20070821kpinginf_50.Kes&ap=1), fecha de consulta: 10 de marzo del 2013.
- [5.] Polanco Anyer, Comunicación Alámbrica, <http://es.slideshare.net/anyerp/medios-de-comunicacin-8016844>, fecha de consulta: 10 de marzo del 2013.
- [6.] Tecnología de la comunicación, Comunicación Alámbrica, <https://sites.google.com/site/tecnocomunicativa/home/comunicacion-alambrica-e-inalambrica>, fecha de consulta: 10 de marzo del 2013.
- [7.] Commons Atribucion, Cable Coaxial, [http://es.wikipedia.org/wiki/Cable\\_coaxial](http://es.wikipedia.org/wiki/Cable_coaxial), fecha de consulta: 10 de marzo del 2013
- [8.] Electricidad De Bricopage, Imagen de Cable Coaxial, <http://www.bricopage.com/electricidad/cablecoaxial.htm>, fecha de consulta: 10 de marzo del 2013.
- [9.] Optical Switching and Networking Handbook., Fibra Óptica, [http://es.wikipedia.org/wiki/Fibra\\_%C3%B3ptica](http://es.wikipedia.org/wiki/Fibra_%C3%B3ptica), fecha de consulta: 10 e marzo del 2013

- [10.] Tec de Monterrey, Imagen Cable de Fibra, [http://homepages.mty.itesm.mx/al789508/tema\\_de\\_interes.htm](http://homepages.mty.itesm.mx/al789508/tema_de_interes.htm), fecha de consulta: 10 de marzo del 2013.
- [11.] Preston Gralla, Comunicación Inalámbrica 1, [http://es.wikipedia.org/wiki/Comunicaci%C3%B3n\\_inal%C3%A1mbrica](http://es.wikipedia.org/wiki/Comunicaci%C3%B3n_inal%C3%A1mbrica), fecha de consulta: 10 de marzo del 2013.
- [12.] Gonzalez Andres Mauricio, Comunicación Inalámbrica 2, <http://www.monografias.com/trabajos16/comunicacion-inalambrica/comunicacion-inalambrica.shtml>, fecha de consulta: 10 de marzo del 2013.
- [13.] Polanco Anyer, Medios de Comunicación, <http://es.slideshare.net/anyerp/medios-de-comunicacin-8016844>, fecha de consulta: 10 de marzo del 2013.
- [14.] Informatica Hoy, Tecnologia wireless, <http://www.informatica-hoy.com.ar/redes-inalambricas-wifi/Que-es-la-tecnologia-Wireless.php>, fecha de consulta: 10 de marzo del 2013
- [15.] Knowlton Jim, Python, <http://es.wikipedia.org/wiki/Python>, fecha de consulta: 10 de marzo del 2013
- [16.] NosinmiUbuntu, Eclipse – Python, <http://www.nosinmiubuntu.com/2010/10/eclipse-python.html>, fecha de consulta:10 de marzo del 2013.
- [17.] Ciseco Plc, Gpio, <http://openmicros.org/index.php/articles/94-ciseco-product-documentation/raspberry-pi/217-getting-started-with-raspberry-pi-gpio-and-python>, fecha de consulta: 10 de marzo del 2013.
- [18.] JHosting, Instalacion de Python, <http://lwk.mjhosting.co.uk/wp-content/uploads/2012/05/RPi.GPIO-setup.png>, fecha de consulta: 10 de marzo del 2013

- [19.] TodoPic, X-CTU, <http://www.todopic.com.ar/foros/index.php?topic=36314.0>, fecha de consulta: 10 de marzo del 2013
- [20.] Faludi, X-CTU, [http://www.faludi.com/itp\\_coursework/meshnetworking/XBee/XBee\\_firmware\\_upgrade.html](http://www.faludi.com/itp_coursework/meshnetworking/XBee/XBee_firmware_upgrade.html) 13/03/2013, fecha de consulta: 10 de marzo del 2013.
- [21.] BBB-micro, Raspberry PI, [http://es.wikipedia.org/wiki/Raspberry\\_Pi](http://es.wikipedia.org/wiki/Raspberry_Pi), fecha de consulta: 10 de marzo del 2013
- [22.] Lee Alex, Raspberry PI, [http://www.cl.cam.ac.uk/projects/raspberrypi/tutorials/robot/components/raspberry\\_pi/](http://www.cl.cam.ac.uk/projects/raspberrypi/tutorials/robot/components/raspberry_pi/), fecha de consulta: 10 de marzo del 2013
- [23.] Benchimol Daniel, Microcontroladores, [http://books.google.com.ec/books?id=V1wLsfO1114C&dq=xbee&hl=es&source=gbs\\_navlinks\\_s](http://books.google.com.ec/books?id=V1wLsfO1114C&dq=xbee&hl=es&source=gbs_navlinks_s), UserShop, fecha de consulta 9 de junio del 2013.
- [24.] Hertz Electronica, XBee, [http://www.5hertz.com/xbee\\_unbrick.html](http://www.5hertz.com/xbee_unbrick.html), fecha de consulta: 10 de marzo del 2013
- [25.] Tp-link, Tp-link, <http://www.tp-link.com/ar/products/#spec>, fecha de consulta, 10 de marzo del 2013
- [26.] Alinformatica, Tp-Link, <http://www.appinformatica.com/redes-inalambrica-tp-link-wn725n-adaptador-nano-150mbps-n.php>, fecha de consulta: 10 de marzo del 2013
- [27.] Alegsa, Acceso Remoto, <http://www.alegsa.com.ar/Dic/acceso%20remoto.php>, fecha de consulta: 10 de marzo del 2013
- [28.] Microsoft tech, Red privada virtual, [http://es.wikipedia.org/wiki/Red\\_privada\\_virtual](http://es.wikipedia.org/wiki/Red_privada_virtual), fecha de consulta: 13 de abril del 2013

- [29.] Faludi, X-CTU,  
[http://www.faludi.com/itp\\_coursework/meshnetworking/XBee/XBee\\_firmware\\_upgrade.html](http://www.faludi.com/itp_coursework/meshnetworking/XBee/XBee_firmware_upgrade.html) 13/03/2013 fecha de consulta: 13 de abril del 2013
- [30.] Jpereda, Arduino, <http://jperedadnr.blogspot.com/2012/12/arduinox-iaavafx-gui-for-home.html> fecha de consulta: 14 de abril del 2013
- [31.] Alex-electronico, VCN,  
<http://electrolitoblog.wordpress.com/2013/02/01/acceder-via-vnc-a-raspberry-pi/>,  
fecha de consulta: 13 de abril del 2013
- [32.] Polanko Cia., Raspberry-Acceso Remoto,  
<http://polankoycia.com/foro/index.php?topic=9800.0>, fecha de consulta: 13 de abril del 2013.
- [33.] Ruiz Toni, Xrpd, <http://www.electroensaimada.com/escritorio-remoto.html>,  
fecha de consulta: 13 de abril del 2013

Anexos

## ANEXO A

# MÓDULO XBEE-USB I&T



**IDEAS &  
Tecnología**  
SOLUCIONES ELECTRONICAS  
[www.ideastechnology.com](http://www.ideastechnology.com)



## Módulo XBEE-USB I&T

XBEE - USB I&T es un módulo de comunicación serial por puerto USB para la interface entre cualquier módulo XBEE y un computador o PC.



## Especificaciones:

- Cable mini -USB
- Led indicador de encendido (3V3), RSS, TX y RX
- Transferencia de datos 300 Baud hasta 3MBAud
- Soporta 7 a 8 bits datos, 1 o 2 bits stop, y odd/even/mark/space/no parity

## Aplicaciones:

- Interface de comunicación serial con el computador y módulo XBEE.
- Adquisición y envío de datos hacia el computador.
- Configuración de Módulos XBEE





## Características

### Fuente de Alimentación

- ✓ Alimentación por interfaz USB desde la PC.

### Señales de Control

- ✓ 3V3: Indicador de voltaje XBEE
- ✓ TX: Indicador transmisión de Datos
- ✓ RX: Indicador recepción de Datos
- ✓ RSSI: Indicador de actividad XBEE (RSSI)



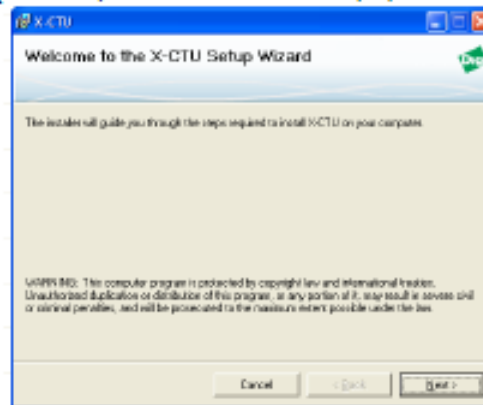
## ANEXO B

Antes de empezar con la configuración del módulo XBee tenemos que instalar el programa **X-CTU**. Aunque existen más programas en el mercado para la configuración del XBe , hemos escogido este porque es el que menos problemas presenta.

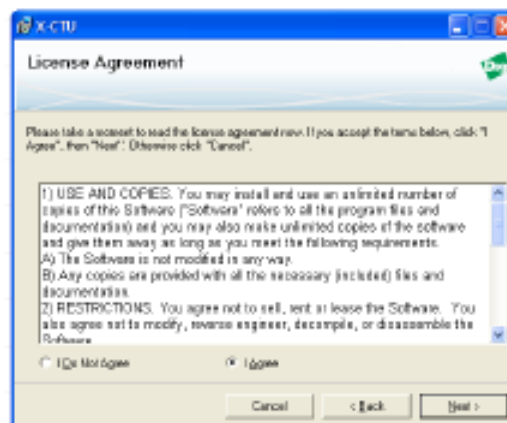
### Instalación de X-CTU.

Esta instalación se puede realizar tanto en linux (a través del programa wine) como en windows. Primero debemos descargar el archivo instalable, que lo podemos encontrar en el siguiente enlace [http://ftp1.digi.com/support/utilities/40002637\\_c.exe](http://ftp1.digi.com/support/utilities/40002637_c.exe), una vez descargado el archivo procedemos a la instalación del programa haciendo doble click en dicho archivo.

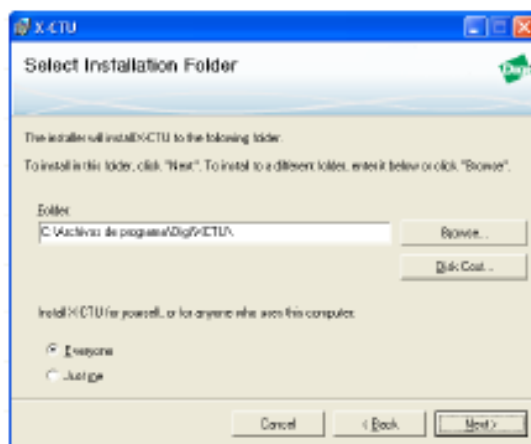
A continuación nos aparecerá la pantalla de bienvenida en la que pulsaremos el botón "Next".



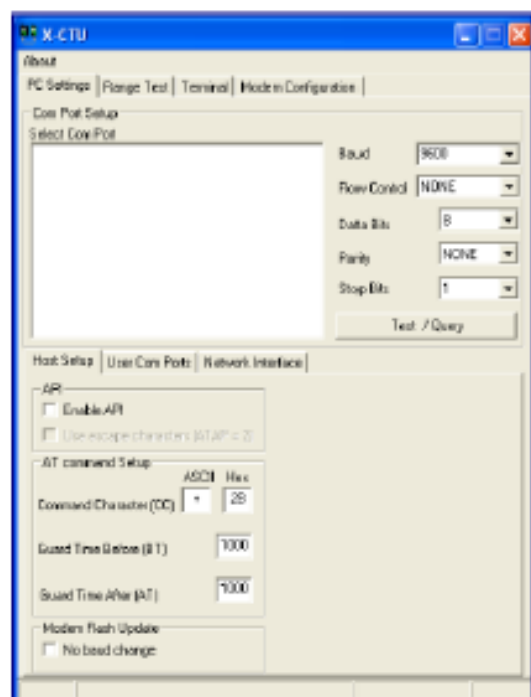
Nos aparecerá la siguiente pantalla de acuerdo de licencia, en la cual tendremos que seleccionar la opción "I Agree" y pulsar "Next".

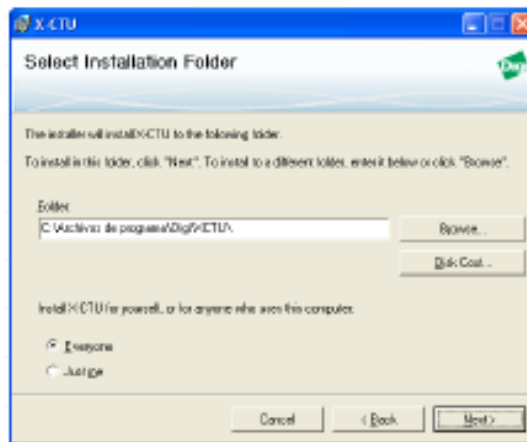


La siguiente pantalla es la de selección de la carpeta de instalación, si queremos cambiar la carpeta en la que queremos que se instale el programa pinchamos en "Browser", sino dejamos la carpeta por defecto y pulsamos "Next".

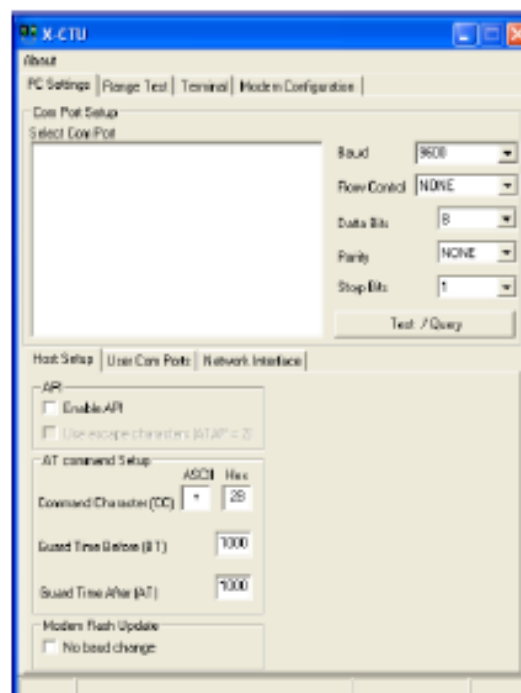


Nos aparecerá una pantalla de confirmación de instalación a la que le daremos al botón "Next" y justo después nos aparece una pantalla de actualización a la que pulsaremos el botón "Si". Una vez terminada la actualización, nos aparecerá una pantalla con la que nos informa que la instalación ha sido completada, pulsando en "Close" para terminar la instalación del programa. Cuando hayamos finalizado la instalación podemos ejecutar el programa a través de un acceso directo ubicado en el escritorio, llamado X-CTU y nos aparecerá una pantalla como la siguiente.



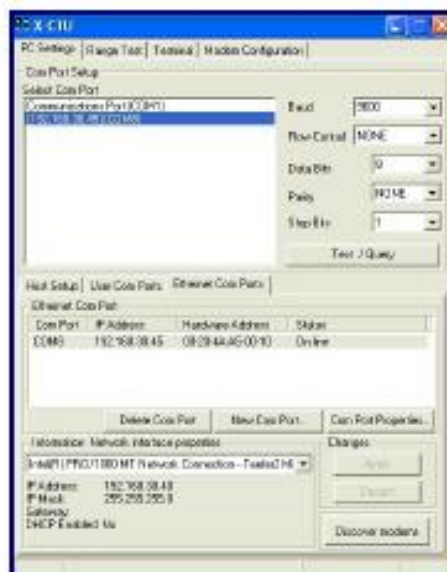


Nos aparecerá una pantalla de confirmación de instalación a la que le daremos al botón "Next" y justo después nos aparece una pantalla de actualización a la que pulsaremos el botón "SI". Una vez terminada la actualización, nos aparecerá una pantalla con la que nos informa que la instalación ha sido completada, pulsando en "Close" para terminar la instalación del programa. Cuando hayamos finalizado la instalación podemos ejecutar el programa a través de un acceso directo ubicado en el escritorio, llamado X-CTU y nos aparecerá una pantalla como la siguiente.

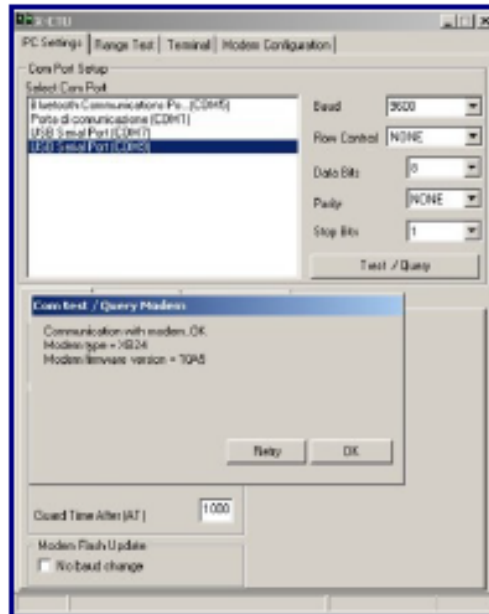




Después de poner en la casilla correspondiente a "Com Port Number" el nombre del puerto, pulsamos el botón "Add" y nos saldrá algo parecido a la siguiente imagen.



A continuación, después de añadir nuestro nuevo dispositivo, debemos señalar en la ventana "Select Com Port" dicho dispositivo, y seleccionamos el botón "Test/Query". Nos aparecerá una ventana como la que aparece en la siguiente imagen.



Si nos sale otro texto como "Unable to open com port" le cambiaremos el valor a la casilla "Baud" aunque el valor configurado de fábrica es "9600" pero tenemos que ir probando.

Después de efectuar todo esto ya tendremos añadido el usb para poder configurar el módulo XBee.

### Configuración del nodo pasarela.

A continuación vamos a explicar como se configura el nodo encargado de recibir la información de los nodos sensores y enviar dicha información al servidor.

Existen dos formas de realizar la configuración de nuestros nodos:

1. Terminal
2. Modem Configuration

Si utilizamos la configuración a través de "Terminal" debemos escribir +++ antes de escribir cualquier comando, tal y como se muestra en la imagen de abajo. Los comandos que vamos a utilizar para la configuración de nuestro nodo son:

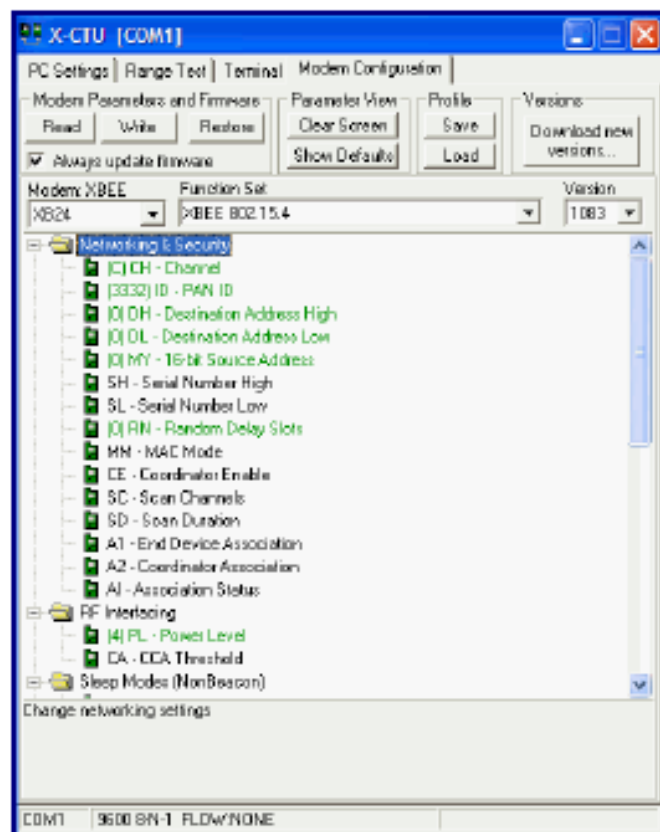
- ATRE: restaura los valores predeterminados de fábrica antes de realizar cualquier modificación.
- ATAP\*: configuración de la API de XBee. Colocar el número que tiene la API de XBee, en nuestro caso 2, por lo que el comando sería ATAP2.
- ATCE1: configuración del módulo XBee en modo Coordinador.

- ATMY\*: dirección del módulo XBee en modo Coordinador. El valor de \* en nuestro caso será 1234. (ATMY1234)
- ATID\*: ID de la conexión que vamos a crear entre nuestros módulos XBee. El valor de \* en nuestro caso es 1111. (ATID1111)
- ATCH\*: Canal por el cual los módulos XBee se van a conectar. El valor de \* en nuestro caso será 0C. (ATCH0C)
- ATWR: escribe una nueva configuración en la memoria no volátil. Si no se escribiese este comando, las modificaciones realizadas solo duraría hasta que el módulo se quede sin batería.
- ATFR: reinicia el módulo XBee.



Configuración por Terminal

Si realizamos la configuración a través de "Modem Configuration" buscamos los nombres, por ejemplo, si queremos configurar el canal por el cual los módulos XBee se van a conectar, tendríamos que buscar CH - Channel y poner el canal 1111 descrito anteriormente. Para escribir en la memoria no volátil tendríamos que pulsar el botón "Write".



Configuración a través de Modem Configuration

### Configuración de los nodos sensores.

Después de configurar el nodo pasarela, vamos a configurar los nodos sensores que son los encargados de enviar la información al nodo que hemos configurado anteriormente.

Al igual que en el apartado anterior, si utilizamos la configuración a través de "Terminal" debemos escribir +++ antes de escribir cualquier comando. Los comandos que vamos a utilizar para la configuración de nuestros nodos son:

**ATRE:** restaura los valores predeterminados de fábrica antes de realizar cualquier modificación.

**ATAP\*:** configuración de la API de XBee. Colocar el número que tiene la API de XBee, en nuestro caso 2, por lo que el comando sería ATAP2.

**ATCE0:** configuración del módulo XBee en modo Nodo Enviante.

**ATMY\*:** dirección del módulo XBee en modo Nodo Enviante. El valor de \* en nuestro caso será 5678. (ATMY5678)

**ATID\*:** ID de la conexión que vamos a crear entre nuestros módulos XBee. El valor de \* en nuestro caso es 1111. (ATID1111)



ATCH\*: Canal por el cual los módulos XBee se van a conectar. El valor de \* en nuestro caso será 0C. (ATCH0C)

ATWR: escribe una nueva configuración en la memoria no volátil. Si no se escribiese este comando, las modificaciones realizadas solo duraría hasta que el módulo se quede sin batería.

ATFR: reinicia el módulo XBee.

Si realizamos la configuración a través de "Modem Configuration" buscamos los nombres, por ejemplo, si queremos configurar el canal por el cual los módulos XBee se van a conectar, tendríamos que buscar CH - Channel y poner el canal 1111 descrito anteriormente. Para escribir en la memoria no volátil tendríamos que pulsar el botón "Write".

Ya tenemos configurado los módulos XBee, con lo cual solo tendremos que subir los Sketch al arduino.

## ANEXO C

### XBee™ Series 2 OEM RF Modules

---

XBee Series 2 OEM RF Modules  
ZigBee™ Networks  
RF Module Operation  
RF Module Configuration  
Appendices



#### Product Manual v1.x.1x - ZigBee Protocol

For OEM RF Module Part Numbers: XB24-Rx/T-00x

ZigBee OEM RF Modules by MaxStream, Inc. - a Digi International brand

Firmware Versions:

- 1.0xx - Coordinator, Transparent Operation
- 1.1xx - Coordinator, API Operation
- 1.2xx - Router, End Device, Transparent Operation
- 1.3xx - Router, End Device, API Operation



355 South 520 West, Suite 180  
London, UT 84042  
Phone: (801) 765-9885  
Fax: (801) 765-9885  
rf-ops@maxstream.net  
www.MaxStream.net (live chat support)

9000866\_A  
2007.06.01

# 1. XBee Series 2 OEM RF Modules

The XBee Series 2 OEM RF Modules were engineered to operate within the ZigBee protocol and support the unique needs of low-cost, low-power wireless sensor networks. The modules require minimal power and provide reliable delivery of data between remote devices.

The modules operate within the ISM 2.4 GHz frequency band.



## 1.1. Key Features

### High Performance, Low Cost

- + Indoor/Urban: up to 133' (40 m)
- + Outdoor line-of-sight: up to 400' (120 m)
- + Transmit Power: 2 mW (+3 dBm)
- + Receiver Sensitivity: -95 dBm

RF Data Rate: 250,000 bps

### Advanced Networking & Security

Retries and Acknowledgements

DSSS (Direct Sequence Spread Spectrum)

Each direct sequence channel has over 65,000 unique network addresses available

Point-to-point, point-to-multipoint and peer-to-peer topologies supported

Self-routing, self-healing and fault-tolerant mesh networking

### Low Power

XBee Series 2

- + TX Current: 40 mA (@3.3 V)
- + RX Current: 40 mA (@3.3 V)
- + Power-down Current: < 1  $\mu$ A @ 25°C

### Easy-to-Use

No configuration necessary for out-of box RF communications

AT and API Command Modes for configuring module parameters

Small form factor

Extensive command set

Free X-CTU Software (Testing and configuration software)

Free & Unlimited Technical Support

### 1.1.1. Worldwide Acceptance

FCC Approval (USA) Refer to Appendix A [p50] for FCC Requirements.

Systems that contain XBee Series 2 RF Modules Inherit MaxStream Certifications.

ISM (Industrial, Scientific & Medical) 2.4 GHz frequency band

Manufactured under ISO 9001:2000 registered standards

XBee Series 2 RF Modules are optimized for use in US, Canada, Australia, Israel and Europe (contact MaxStream for complete list of agency approvals).



## 1.2. Specifications

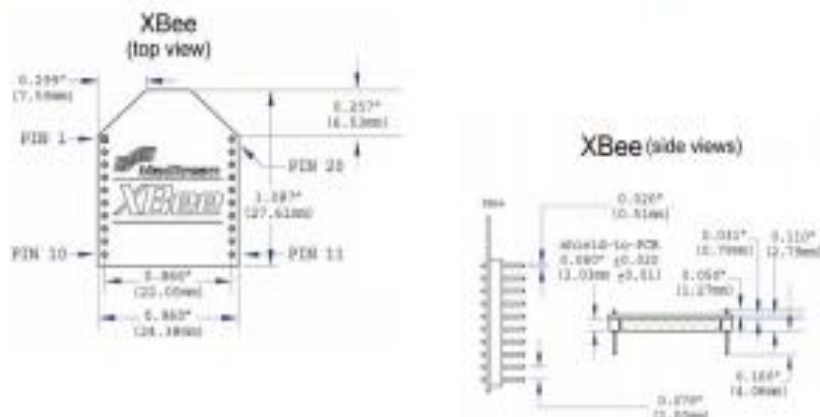
Table 1-01. Specifications of the Xtreme Series 2 OEM RF Module (PRELIMINARY)

Specification	Xtreme Series 2
<b>Performance</b>	
Indoor/Urban Range	up to 100 ft. (40 m)
Outdoor RF line-of-sight Range	up to 400 ft. (100 m)
Transmit Power Output (software selectable)	2mW (+3dBm)
RF Data Rate	250/500 kbps
Serial Interface Data Rate (software selectable)	1200 - 230400 bps (non-standard baud rates also supported)
Receiver Sensitivity	-95 dBm (1% packet error rate)
<b>Power Requirements</b>	
Supply Voltage	3.0 - 3.4 V
Operating Current (Transmit)	40mA (@ 3.3 V)
Operating Current (Receive)	40mA (@ 3.3 V)
Power-down Current	+ 1 uA (@ 25°C)
<b>General</b>	
Operating Frequency Band	ISM 2.4 GHz
Dimensions	0.980" x 1.087" (2.480cm x 2.761cm)
Operating Temperature	-40 to 85°C (industrial)
Antenna Options	Integrated Whip, Chip, RF SMA, or U.F.L. Connector
<b>Networking &amp; Security</b>	
Supported Network Topologies	Point-to-point, Point-to-multipoint, Peer-to-peer & Mesh
Number of Channels (software selectable)	16 Direct Sequence Channels
Addressing Options	MAC ID and Address, Cluster ID and Endpoints (optional)
<b>Agency Approvals</b>	
United States (FCC Part 15.247)	Pending
Industry Canada (IC)	Pending
Europe (CE)	Pending

Antenna Options: The ranges specified are typical when using the Integrated Whip (1.5 dBi) and Dipole (2.1 dBi) antennas. The Chip antenna option provides advantages in its form factor; however, it typically yields shorter range than the Whip and Dipole antenna options when transmitting outdoors. For more information, refer to the "Xtreme Series 2 Antenna" application note located on MaxStream's web site <http://www.maxstream.net/support/knowledgebase/article.php?kb=153>

### 1.3. Mechanical Drawings

Figure 1-81. Mechanical drawings of the Xbee Series 2 OEM RF Modules (antenna options not shown)



### 1.4. Mounting Considerations

The Xbee Series 2 RF Module (through-hole) was designed to mount into a receptacle (socket) and therefore does not require any soldering when mounting it to a board. The Xbee Series 2 Development Kits contain RS-232 and USB interface boards which use two 20-pin receptacles to receive modules.

Figure 1-82. Xbee Series 2 Module Mounting to an RS-232 Interface Board.



The receptacles used on MaxStream development boards are manufactured by Century Interconnect. Several other manufacturers provide comparable mounting solutions; however, MaxStream currently uses the following receptacles:

- Through-hole single-row receptacles -  
Samtec P/N: MMS-110-01-L-SV (or equivalent)
- Surface-mount double-row receptacles -  
Century Interconnect P/N: CPRHSL20-D-0-1 (or equivalent)
- Surface-mount single-row receptacles -  
Samtec P/N: SMM-110-02-SM-S

MaxStream also recommends printing an outline of the module on the board to indicate the orientation the module should be mounted.

## 1.5. Pin Signals

Figure 1-48. Xbee Series 2 RF Module Pin Number (top sides shown - shields on bottom)

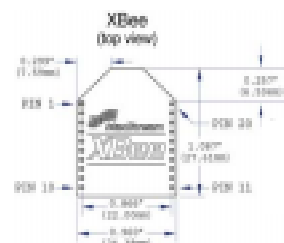


Table 1-41. Pin Assignments for the Xbee Series 2 Modules (Low-asserted signals are distinguished with a horizontal line above signal name.)

Pin #	Name	Direction	Description
1	VCC	-	Power supply
2	DCOUT	Output	UART Data Out
3	DIN / CONFIG	Input	UART Data In
4	DIO0	Either	Digital IO 0
5	RESET	Input	Module Reset (reset pulse must be at least 200 ns)
6	PWM / RSSI / DIO10	Output	PWM Output 0 / Rx Signal Strength Indicator / Digital IO
7	PWM / DIO11	Either	Digital IO 11
8	[reserved]	-	Do not connect
9	DTR / SLEEP / RC / DIN	Input	Pin Sleep Control Line or Digital Input 8
10	GND	-	Ground
11	DIO4	Either	Digital IO 4
12	CTS / DIO7	Either	Clear-to-Send Flow Control or Digital IO 7
13	DS / SLEEP	Output	Module Status Indicator
14	[reserved]	-	Do not connect
15	Associate / DIO5	Either	Associated Indicator, Digital IO 5
16	RTS / DIO8	Either	Request-to-Send Flow Control, Digital IO 8
17	AD3 / DIO3	Either	Analog Input 3 or Digital IO 3
18	AD2 / DIO2	Either	Analog Input 2 or Digital IO 2
19	AD1 / DIO1	Either	Analog Input 1 or Digital IO 1
20	AD0 / DIO0	Either	Analog Input 0 or Digital IO 0

### Design Notes:

- Minimum connections: VCC, GND, DCOUT & DIN
- Minimum connections to support firmware upgrades: VCC, GND, DIN, DCOUT, RTS & DTR
- Signal Direction is specified with respect to the module
- Module includes a 30k Ohm resistor attached to **RESET**
- Several of the input pull-ups can be configured using the PR command
- Unused pins should be left disconnected

## 1.6. Electrical Characteristics

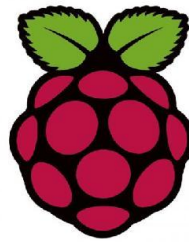
Table 1-03. DC Characteristics of the XBee Series 2 (VCC = 2.8 - 3.4 VDC)

Symbol	Parameter	Condition	Min	Typical	Max	Units
V <sub>IL</sub>	Input Low Voltage	All Digital Inputs	-	-	0.2 * VCC	V
V <sub>IH</sub>	Input High Voltage	All Digital Inputs	0.8 * VCC	-	0.18 * VCC	V
V <sub>OL</sub>	Output Low Voltage	I <sub>OL</sub> = 3 mA, VCC = 2.7 V	-	-	0.18 * VCC	V
V <sub>OH</sub>	Output High Voltage	I <sub>OH</sub> = -2 mA, VCC = 2.7 V	0.8 * VCC	-	-	V
I <sub>IN</sub>	Input Leakage Current	V <sub>IN</sub> = VCC or GND, all inputs, per pin	-	-	65uA	uA
I <sub>TX</sub>	Transmit Current	VCC = 3.3 V	-	45	-	mA
I <sub>RX</sub>	Receive Current	VCC = 3.3 V	-	50	-	mA
I <sub>WR-DWN</sub>	Power-down Current	SM parameter = 1	-	< 10	-	uA

ANEXO D

## Quick Start Guide

### The Raspberry Pi – Single Board Computer



Source: Raspberry Pi &

Wiki

Chapter 1: RPi Hardware Basic Setup



## Typical Hardware You Will Need

While the RPi can be used without any additional hardware (except perhaps a power supply of some kind), it won't be much use as a general computer. As with any normal PC, it is likely you will need some additional hardware.

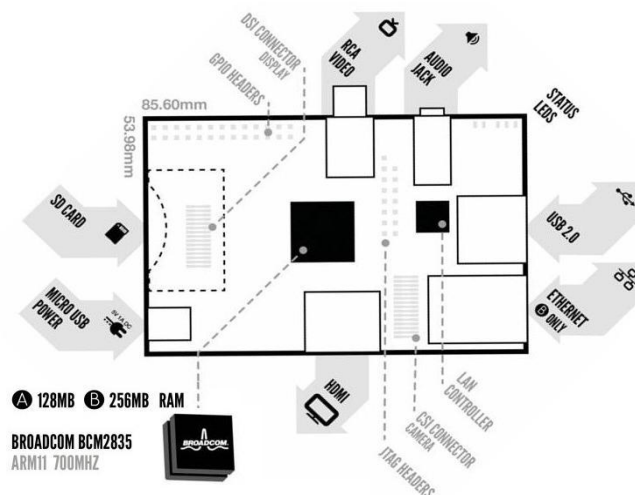
The following are more or less essential:

- Raspberry Pi board
- Prepared Operating System SD Card
- USB keyboard
- Display (with HDMI, DVI, Composite or SCART input)
- Power Supply
- Cables

Highly suggested extras include:

- USB mouse
- Internet connectivity - a USB WiFi adaptor (Model A/B) or a LAN cable (Model B)
- Powered USB Hub
- Case

## Connecting Together



You can use the diagram to connect everything together, or use the following instructions:

1. Plug the preloaded SD Card into the Pi.
2. Plug the USB keyboard and mouse into the Pi, perhaps via a USB Hub. Connect the Hub to power, if necessary.
3. Plug the video cable into the screen (TV) and into the Pi.
4. Plug your extras into the Pi (USB WiFi, Ethernet cable, hard drive etc.). This is where you may really need a USB Hub.
5. Ensure that your USB Hub (if any) and screen are working.

6. Plug the power source into the main socket.
7. With your screen on, plug the other end of the power source into the Pi.
8. The Pi should boot up and display messages on the screen.

It is always recommended to connect the MicroUSB Power to the unit last (while most connections can be made live, it is best practice to connect items such as displays/h/w pin connections with the power turned off).

The RPi may take a long time to boot when powered-on for the first time, so be patient!

## Prepared Operating System SD Card

---

As the RPi has no internal storage or built-in operating system it requires an SD-Card that is set up to boot the RPi.

- You can create your own preloaded card using any suitable SD card you have. Be sure to backup any existing data on the card.
- Preloaded SD cards will be available from the RPi Shop.

This guide will assume you have a preloaded SD card.

## Keyboard & Mouse

---

Most standard USB keyboards and mice will work with the RPi. Wireless keyboard/mice should also function, and only require a single USB port for an RF dongle. In order to use a Bluetooth keyboard or mouse you would need to use a Bluetooth dongle, which again uses a single port.

Remember that the Model A has a single USB port and the Model B only has two (typically a keyboard and mouse will use a USB port each).

## Display

---

There are two main connection options for the RPi display, *HDMI* (high definition) and *Composite* (low definition).

- HD TVs and most LCD Monitors can be connected using a full-size 'male' HDMI cable, and with an inexpensive adaptor if DVI is used. HDMI versions 1.3 and 1.4 are supported, and a version 1.4 cable is recommended. The RPi outputs audio and video via HDMI, but does not support HDMI input.
- Older TVs can be connected using Composite (a yellow-to-yellow cable) or via SCART (using a Composite to SCART adaptor). PAL and NTSC TVs are supported. When using composite video, audio is available from a 3.5mm (1/8 inch) socket, and can be sent to your TV, to headphones, or to an amplifier. To send audio your TV, you will need a cable which adapts from 3.5mm to double (red and white) RCA connectors.

**Note: There is no VGA output available, so older VGA monitors will require an expensive adaptor.**

Using an HDMI to DVI-D (digital) adaptor plus a DVI to VGA adaptor will not work. HDMI does not supply the DVI-A (analogue) needed to convert to VGA - converting an HDMI or DVI-D source to VGA (or component) needs an active converter. (It can work out cheaper to buy a new monitor.) The lack of VGA has been acknowledged as a priority issue.

## Power Supply

---

The unit uses a Micro USB connection to power itself (only the power pins are connected - so it will not transfer data over this connection). A standard modern phone charger with a micro-USB connector will do, but needs to produce at least 700mA at 5 volts. Check your power supply's ratings carefully. Suitable mains adaptors will be available from the RPi Shop and are recommended if you are unsure what to use.

You can use a range of other power sources (assuming they are able to provide enough current ~700mA):

- Computer USB Port or powered USB hub (will depend on power output)
- Special wall warts with USB ports
- Mobile Phone Backup Battery (will depend on power output) (in theory - needs confirmation)

To use the above, you'll need a USB A 'male' to USB micro 'male' cable - these are often shipped as data cables with MP3 players.

## Cables

---

You will probably need a number of cables in order to connect your RPi up.

1. Micro-B USB Power Cable
2. HDMI-A or Composite cable, plus DVI adaptor or SCART adaptor if required, to connect your RPi to the Display/Monitor/TV of your choice.
3. Audio cable, this is not needed if you use a HDMI TV/monitor.
4. Ethernet/LAN Cable

## Additional Peripherals

---

You may decide you want to use various other devices with your RPi, such as Flash Drives/Portable Hard Drives, Speakers etc.

### Internet Connectivity

This may be an Ethernet/LAN cable (standard RJ45 connector) or a USB WiFi adaptor. The RPi ethernet port is auto-sensing which means that it may be connected to a router or directly to another computer (without the need for a crossover cable).

## USB-Hub

In order to connect additional devices to the RPi, you may want to obtain a USB Hub, which will allow multiple devices to be used.

It is recommended that a **powered** hub is used - this will provide any additional power to the devices without affecting the RPi itself.

USB version 2.0 is recommended. USB version 1.1 is fine for keyboards and mice, but may not be fast enough for other accessories.

## Case

Since the RPi is supplied without a case, it will be important to ensure that you do not use it in places where it will come into contact with conductive metal or liquids, unless suitably protected.

## Expansion & Low Level Peripherals

If you plan on making use of the low level interfaces available on the RPi, then ensure you have suitable header pins for the GPIO (and if required JTAG) suitable for your needs.

Also if you have a particular low-level project in mind, then ensure you design in suitable protection circuits to keep your RPi safe.

# Chapter 2: RPi Advanced Setup

## Finding hardware and setting up

---

You'll need a preloaded SD card, USB keyboard, TV/Monitor (with HDMI/ DVI/ Composite/ SCART input), and power supply (USB charger or a USB port from a powered USB Hub or another computer).

You'll likely also want a USB mouse, a case, and a USB Hub (a necessity for Model A). A powered USB Hub will reduce the demand on the RPi. To connect to the Internet, you'll need either an Ethernet/LAN cable (Model B) or a USB WiFi adaptor (either model).

When setting up, it is advisable to connect the power after everything else is ready.

## Serial connection

---

The Serial Port is a simple and uncomplicated method to connect to the Raspberry Pi. The communication depends on byte wise data transmission, is easy to setup and is generally available even before boot time.

### First interaction with the board

---

Connect the serial cable to the COM port in the Raspberry Pi, and connect the other end to the COM port or USB Serial Adapter in the computer.

---

### Serial Parameters

The following parameters are needed to connect to the Raspberry. All parameters except **Port\_Name** and **Speed** are default values and may not need to be set.

**Port\_Name:** Linux automatically assigns different names for different types of serial connectors. Choose your option:

Standard Serial Port: ttyS0 ... ttySn

USB Serial Port Adapter: ttyUSB0 ...  
ttyUSBn **Speed:** 115200

Bits: 8

Parity:

None Stop

Bits: 1

Flow Control: None

The Serial Port is generally usable by the users in the group **dialout**. To add oneself to the group **dialout** the the following command needs to be executed with **root** privileges:

```
$useradd -G {dialout} your_name
```

### Super Easy Way Using GNU Screen

Enter the command below into a terminal window

```
screen Port_Name 115200
```

### Super Easy Way Using Minicom

Run minicom with the following parameters:

```
minicom -b 115200 -o -D Port_Name
```

### GUI method with GtkTerm

Start *GtkTerm*, select Configuration->Port and enter the values above in the labelled fields.

### Windows Users

Windows Users above Windows XP must download putty or a comparable terminal program. Users of XP and below can choose between using *putty* and *Hyperterminal*.

### First Dialog

If you get the prompt below, you are connected to the Raspberry Pi shell!

```
prompt> #
```

First command you might want try is "help":

```
prompt> # help
```

If you get some output, you are correctly connected to the Raspberry Pi! Congratulations!

## SD card setup

Now we want to use an SD card to install some GNU/Linux distro in it and get more space for our stuff. You can use either an SD or SDHC card. In the latter case of course take care that your PC card reader also supports SDHC. Be aware that you are not dealing with an x86 processor, but instead a completely different architecture called ARM, so don't forget to install the ARM port for the distro you are planning to use.

### Formatting the SD card via the mkcard.txt script

1. Download **mkcard.txt** .
2. `$ chmod +x mkcard.txt`
3. `$ ./mkcard.txt /dev/sdx`, where *x* is the letter of the card. You can find this by inserting your card and then running `dmesg | tail`. You should see the messages about the device being mounted in the log. Mine mounts as **sdC**.

Once run, your card should be formatted.

### Formatting the SD card via fdisk "Expert mode"

First, lets clear the partition table:

```
$ sudo fdisk /dev/sdb
```

```
Command (m for help): o
```

```
Building a new DOS disklabel. Changes will remain in memory only, until you decide to write them. After that, of course, the previous content won't be recoverable.
```

```
Warning: invalid flag 0x0000 of partition table 4 will be corrected by w(rite)
```

Print card info:

```
=====  
Command (m for help): p  
  
Disk /dev/sdb: 128 MB, 128450560 bytes  
.....  
=====
```

Note card size in bytes. Needed later below.

Then go into "Expert mode":

```
=====  
Command (m for help): x  
=====
```

Now we want to set the geometry to 255 heads, 63 sectors and calculate the number of cylinders required for the particular SD/MMC card:

```
=====  
Expert command (m for help): h  
Number of heads (1-256, default 4): 255  
  
Expert command (m for help): s  
Number of sectors (1-63, default 62): 63  
Warning: setting sector offset for DOS compatibility  
=====
```

NOTE: Be especially careful in the next step. First calculate the number of cylinders as follows:

B = Card size in bytes (you got it before, in the second step when you printed the info out)

C = Number of cylinders

```
C=B/255/63/512
```

When you get the number, you round it DOWN. Thus, if you got 108.8 you'll be using 108 cylinders.

```
=====  
Expert command (m for help): c  
Number of cylinders (1-1048576, default 1011): 15  
=====
```

In this case 128MB card is used (reported as 128450560 bytes by fdisk above), thus  $128450560 / 255 / 63 / 512 = 15.6$  rounded down to 15 cylinders. Numbers there are 255 heads, 63 sectors, 512 bytes per sector.

So far so good, now we want to create two partitions. One for the boot image, one for our distro. Create the FAT32 partition for booting and transferring files from Windows. Mark it as bootable.

```

Expert command (m for help): r
Command (m for help): n Command
action
  e   extended
  p   primary partition (1-4)
p
Partition number (1-4): 1
    
```

```

First cylinder (1-245, default 1): (press Enter) Using default
value 1
Last cylinder or +size or +sizeM or +sizeK (1-245, default 245): +50
    
```

```

Command (m for help): t
Selected partition 1
Hex code (type L to list codes): c
Changed system type of partition 1 to c (W95 FAT32 (LBA))
    
```

```

Command (m for help): a Partition
number (1-4): 1
    
```

Create the Linux partition for the root file system.

```

Command (m for help): n
Command action
  e   extended
  p   primary partition (1-4)
p
Partition number (1-4): 2
First cylinder (52-245, default 52): (press Enter) Using default
value 52
Last cylinder or +size or +sizeM or +sizeK (52-245, default 245):(press Enter)
Using default value 245
    
```

Print and save the new partition records.



```
Command (m for help): p
```

```
Disk /dev/sdc: 2021 MB, 2021654528 bytes
255 heads, 63 sectors/track, 245 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
```

Device	Boot	Start	End	Blocks	Id	System
/dev/sdc1	*	1	51	409626	c	W95 FAT32 (LBA)
/dev/sdc2		52	245	1558305	83	Linux

```
Command (m for help): w
```

```
The partition table has been altered!
```

```
Calling ioctl() to re-read partition table.
```

**WARNING: Re-reading the partition table failed with error 16: Device or resource busy. The kernel still uses the old table. The new table will be used at the next reboot.**

**WARNING: If you have created or modified any DOS 6.x partitions, please see the fdisk manual page for additional information.**

Syncing disks.

Now we've got both partitions, next step is formatting them.

**NOTE:** If the partitions (/dev/sdc1 and /dev/sdc2) does not exist, you should unplug the card and plug it back in. Linux will now be able to detect the new partitions.

```
$ sudo mkfs.msdos -F 32 /dev/sdc1 -n LABEL
mkfs.msdos 2.11 (12 Mar 2005)
```

```
$ sudo mkfs.ext3 /dev/sdc2 mke2fs
1.40-WIP (14-Nov-2006) Filesystem
label=
OS type: Linux
```

```
Block size=4096 (log=2) Fragment
size=4096 (log=2) 195072 inodes,
389576 blocks
19478 blocks (5.00%) reserved for the super user First data
block=0
```

```
Maximum filesystem blocks=402653184 12
block groups
```

```
32768 blocks per group, 32768 fragments per group
```

```
16256 inodes per group
```

```
Superblock backups stored on blocks:
```

```
32768, 98304, 163840, 229376, 294912
```

```
Writing inode tables: done
Creating journal (8192 blocks): done
Writing superblocks and filesystem accounting information:
```

All done!

*NOTE:* For convenience, you can add the `-L` option to the `mkfs.ext3` command to assign a volume label to the new `ext3` filesystem. If you do that, the new (automatic) mount point under `/media` when you insert that SD card into some Linux hosts will be based on that label. If there's no label, the new mount point will most likely be a long hex string, so assigning a label makes manual mounting on the host more convenient.

## Setting up the boot partition

The boot partition must contain:

- `bootcode.bin` : 2nd stage bootloader, starts with SDRAM disabled
- `loader.bin` : 3rd stage bootloader, starts with SDRAM enabled
- `start.elf`: The GPU binary firmware image, provided by the foundation.
- `kernel.img`: The OS kernel to load on the ARM processor. Normally this is Linux - see instructions for compiling a kernel.
- `cmdline.txt`: Parameters passed to the kernel on boot.

Optional files:

- `config.txt`: A configuration file read by the GPU. Use this to override set the video mode, alter system clock speeds, voltages, etc.
- `vlls` directory: Additional GPU code, e.g. extra codec's. Not present in the initial release.

## Additional files supplied by the foundation

These files are also present on the SD cards supplied by the foundation.

Additional kernels. Rename over `kernel.img` to use them (ensure you have a backup of the original `kernel.img` first!):

- `kernel_emergency.img` : kernel with busybox rootfs. You can use this to repair the main linux partition using `e2fsck` if the linux partition gets corrupted.

Additional GPU firmware images, rename over `start.elf` to use them:

- `arm128_start.elf` : 128M ARM, 128M GPU split (use this for heavy 3D work, possibly also required for some video decoding)
- `arm192_start.elf` : 192M ARM, 64M GPU split (this is the default)
- `arm224_start.elf` : 224M ARM, 32M GPU split (use this for Linux only with no 3D or video processing. It's enough for the 1080p frame buffer, but not much else)

## Writing the image into the SDcard and finally booting GNU/Linux

The easiest way to do this is to use PiCard. It even saves you from some hassles explained above. You will need your SD card + reader and a Linux pc to use PiCard. After that, just plug the card into your Rpi.

Setting up the boot args

## Wire up your Raspberry Pi and power it up

As explained in Chapter 1

### SD Card Cloning/Backup

*Note: Update these instructions if required once they've been tried*

From windows you can copy the full SD-Card by using Win32DiskImager. Alternatively, you can use the following instructions;

*Note:*

*Many built-in SD card readers do not work, so if you have problems use an external SD-USB adapter for this.*

## Required Software Setup

download a windows utility dd.exe from  
<http://www.chrysocome.net/dd> rename it windd.exe

*(This executable can write to your harddisk so exercise caution using it!)*

make a copy named dd-removable.exe

*(That executable refuses to write to your hard disk as it is named dd-removable. As long as you use dd-removable.exe you cannot lose your hard disk)*

Connect an SD card to the computer run "dd-removable – list"

## Should give something like this:

*rawwrite dd for windows version 0.6beta3. Written by John Newbiggin <jn@it.swin.edu.au>*

*This program is covered by terms of the GPL Version 2.*

```
NT      Block      Device      Objects
\\?\Device\Harddisk1\Partition0 link to
```

```
\\?\Device\Harddisk1\DR8
```

```
Removable media other than floppy. Block size = 512 size is  
4075290624 bytes
```

This "\\?\Device\Harddisk1\Partition0" is the part you need.

### Reading an image from the SD Card

**BEWARE: DO THIS WRONG AND YOU CAN LOSE YOUR HARDDISK!!!**

Obviously, you can NOT use 'dd-removable' to read an image as that executable refuses to write to your hard disk (so extra care is required here as you use 'windd').

To **read** an SD-card image from the SD-card use:

```
windd bs=1M if=\\?\Device\Harddisk1\Partition0 of=THE_IMAGE_READ -size  
Your disk name ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
```

### Copying an image to the SD Card

**BEWARE: DO THIS WRONG AND YOU CAN LOSE YOUR HARDDISK!!!**

To **copy** an image named "THEIMAGE" to the SD-card do this:

```
dd-removable bs=1M if=THEIMAGE of=\\?\Device\Harddisk1\Partition0  
Your disk name ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
```

### Software Development/Proving

A supported platform for the Raspberry is Qt , which is already being worked on. C/C++ is supported through a gcc cross-compiling tool chain.

After compiling, using QEMU and a Linux VM would be one way of testing your apps. This also works on Windows. Search the forum for the readymade ARM images. The choice of programming languages, IDEs and other tools ON the R-Pi is only determined by:

- The operating system compatibility ( at the moment the specific Linux distro used)
- The status of the respective ARM package repositories and their binary compatibility
- The possibility to build other software + its dependencies for the R-Pi from sources.

