ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA

ESTUDIO COMPARATIVO ENTRE LAS TECNOLOGÍAS BLUETOOTH Y WI-FI EN AMBIENTES DE CORTO ALCANCE A TRAVÉS DE LA IMPLEMENTACIÓN DE DOS PROTOTIPOS Y DE SU SIMULACIÓN

PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN ELECTRÓNICA Y TELECOMUNICACIONES

ARIAS PILAQUINGA DIEGO BOLIVAR

MUELA VACA DIEGO FRANCISCO

DIRECTORA: MSc. SORAYA SINCHE

Quito, Diciembre 2007

DECLARACIÓN

Nosotros, Arias Pilaquinga Diego Bolivar, Muela Vaca Diego Francisco, declaramos bajo juramento que el trabajo aquí descrito es de nuestra autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que hemos consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedemos nuestros derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

Arias Pilaquinga Diego Bolivar

Muela Vaca Diego Francisco

CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por los señores Arias Pilaquinga Diego Bolivar y Muela Vaca Diego Francisco, bajo mi supervisión.

MSc. Soraya Sinche
DIRECTORA DEL PROYECTO

AGRADECIMIENTO

A Dios quien día a día me ha bendecido y me ha permitido seguir viviendo para alcanzar esta nueva meta.

A mis padres por haberme guiado por el sendero de la verdad y la justicia.

A mi abuelita quien me cuido cuando fui pequeño y me ha recordado siempre que debo cumplir con mi meta.

A mis hermanos Marco, Polo y Abdala, quienes me han apoyado y que con sus palabras de aliento me han hecho sentir una persona diferente.

A mis profesores forjadores de una juventud noble y justa que mañana harán del Ecuador una patria más libre y próspera.

A la MSc. Soraya Sinche por haber tenido paciencia en la dirección del presente Proyecto de Titulación.

A mis amigos y amigas por sus sabios consejos que han hecho que vaya por un camino de bien.

A la escuela Simón Bolívar de Angamarca, al Instituto Nacional Mejía y a la Escuela Politécnica Nacional cuyas aulas son testigas de sueños que hoy los veo realizados.

A mis novias quienes en su momento me brindaron su cariño y admiración.

Y a todas aquellas personas que de una u otra forma ayudaron y colaboraron para que se cristalice este nuevo sueño.

DIEGO ARIAS

AGRADECIMIENTO

A Dios por permitirme vivir día a día, y colmarme de bendiciones.

A mis padres David Muela y Marina Vaca, a mis hermanos Blanqui, Yoli, Gloria, Carlos, Pepe, Fernando, Anita y a mis cuñados y cuñadas por su apoyo incondicional que me han brindado. Gracias a ellos hoy puedo ver la culminación de una etapa más del camino de mi vida

A todos mis profesores de la Escuela Politécnica Nacional quienes me inculcaron todos los conocimientos para llegar a ser un profesional más del Ecuador.

A todos mis amigos por el apoyo incondicional en el desarrollo del Proyecto de Titulación

A la MSc. Soraya Sinche por haberme guiado y sobre todo haber tenido mucha paciencia en la dirección del presente Proyecto de Titulación.

Y a todas aquellas personas que de una u otra forma ayudaron y colaboraron para que se cristalice este nuevo sueño.

DIEGO MUELA

DEDICATORIA

Dedico este trabajo a mis padres, especialmente a mi madre quien deposito su confianza en mí desde que fui un niño.

También dedico este trabajo a todas las personas que no creyeron en mí y a mis enemigos ya que gracias a ellos fui adquiriendo fuerzas para seguir adelante.

DIEGO ARIAS

DEDICATORIA

Dedico este trabajo a mis padres y mi familia, por haber depositado toda su confianza en mí.

Dedico este trabajo a mi hermano Carlos y a mi cuñada Raquel por todo el apoyo brindado en el colegio y la universidad.

También dedico este trabajo a todas las personas que no creyeron en mí ya que gracias a ellos fui adquiriendo fuerzas para seguir adelante.

DIEGO MUELA

ÍNDICE GENERAL

ÍNDICE DE FIGURAS

CAPÍTULO 1

CAPÍTULO 2

CAPÍTULO 3

CAPÍTULO 4

ÍNDICE DE TABLAS

CAPÍTULO 1

CAPÍTULO 2

CAPÍTULO 4

RESUMEN

El presente Proyecto de Titulación se enfoca a la comparación de las tecnologías inalámbricas *Bluetooth* y *Wi-Fi* que se están utilizando en la actualidad, que hacen posible la implementación de redes inalámbricas personales, las cuales debido a su eficiencia y desempeño, son de gran utilidad para la prestación de servicios.

El Proyecto de titulación consta de las siguientes partes:

En el capítulo uno se analiza las tecnologías *Bluetooth* y *Wi-Fi*, se realiza un estudio de las características fundamentales que posee cada tecnología y sus posibles aplicaciones. Además se realiza un estudio de los factores de propagación inalámbrica y de los modelos para el cálculo del enlace.

En el capítulo dos se realiza el diseño de los prototipos inalámbricos con tecnología *Bluetooth* y *Wi-Fi*, se analizan las alternativas del acceso inalámbrico con estas tecnologías y se realizan las pruebas correspondientes, en las que se puede evaluar la correcta operación de los prototipos implementados, al igual que se determinan las limitaciones de los mismos.

En el capítulo tres se simula el funcionamiento de los prototipos, en la simulación se puede verificar cómo varía la velocidad, potencia, relación señal a ruido en función de la distancia.

Para la simulación de los prototipos de prueba se utiliza el ns-2, que permite simular eventos discretos, redes telemáticas e inalámbricas.

En el capítulo cuatro se realiza la comparación de resultados obtenidos en la implementación práctica y en la simulación de los prototipos. En base a estos resultados se determina cuál de las tecnologías se comporta de mejor manera en ambientes de corto alcance.

En el capítulo cinco se comentan las diferentes conclusiones y recomendaciones concernientes al desarrollo del proyecto de titulación.

En los anexos del proyecto de titulación se encuentra información utilizada en el desarrollo del mismo.

PRESENTACIÓN

La posibilidad de comunicarse en cualquier momento y desde cualquier lugar sin estar sujeto a un sitio específico, ha sido desde siempre uno de los principales objetivos de los sistemas de comunicación. En los últimos años se han ido desarrollando diversas aplicaciones para tecnologías inalámbricas las cuales facilitan a sus usuarios acceder a información y recursos en tiempo real sin necesidad de estar conectados a una red determinada.

Para el logro de los objetivos planteados en este proyecto se ha recopilado una cantidad valiosa de información de las tecnologías inalámbricas *Bluetooth* y *Wi-Fi*, que puede ser útil al momento de realizar consultas por los estudiantes.

El presente proyecto contiene las pautas necesarias para el diseño e implementación de prototipos inalámbricos con tecnología *Bluetooth* y *Wi-Fi*, en ambientes de corto alcance, además brinda al público en general la posibilidad de escoger qué tecnología utilizar de acuerdo a sus requerimientos.

Para la simulación de los prototipos *Bluetooth* y *Wi-Fi* se utilizó el simulador ns-2, conjuntamente con la librería *UCBT*. Este simulador puede convertirse en una excelente herramienta de aprendizaje para los estudiantes de ingeniería al momento de simular redes inalámbricas.

CAPÍTULO 1

1. ESTUDIO DE LAS TEGNOLOGÍAS BLUETOOTH Y WI-FI

En los últimos años las redes de área local inalámbricas *WLAN*, están ganando mucha aceptación, que va creciendo conforme aumentan sus prestaciones y se descubren nuevas aplicaciones para ellas. Las *WLAN* permiten a sus usuarios acceder a información y recursos en tiempo real sin necesidad de estar conectados a una red determinada.

Con las *WLAN* se elimina la necesidad de usar cables y establece nuevas aplicaciones añadiendo flexibilidad a la red. Un usuario dentro de una red *WLAN* puede transmitir y recibir voz, datos y vídeo dentro de edificios, entre edificios o campus e incluso sobre áreas metropolitanas a velocidades de 11 *Mbps* o superiores.

Las *WLAN* ofrecen una fácil incorporación de nuevos usuarios a la red, movilidad de los usuarios, bajo costo de implementación comparado con los sistemas cableados, velocidades de transmisión aceptables y áreas de cobertura que dependen de la potencia del interfaz utilizado, etc.

Es por esta razón que el objetivo principal del proyecto de titulación es la comparación de las tecnologías inalámbricas *Bluetooth* y *Wi-Fi*, en ambientes de corto alcance, es decir en distancias limitadas, ya que en la actualidad estas tecnologías son las utilizadas para este propósito.

1.1 BLUETOOTH [17]

Bluetooth es una tecnología estandarizada por la *IEEE 802.15.1*, define un estándar global de comunicaciones inalámbricas de corto alcance, que posibilita la transmisión de voz y datos tanto para estaciones fijas y móviles (*PCs*, teléfonos móviles, *PDA* (Asistentes Personales Digitales), etc.).

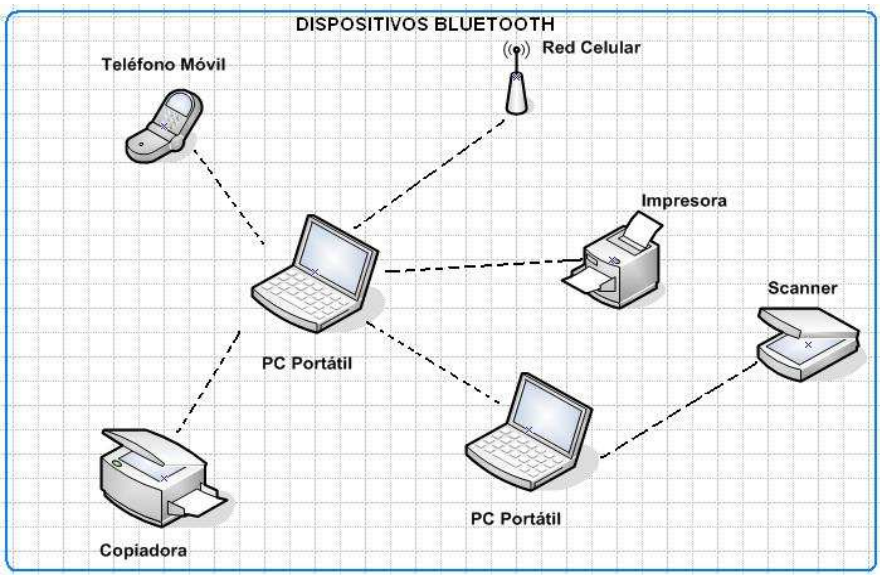


Figura 1.1 Equipos *Bluetooth*

Los objetivos que se persigue con este estándar son:

- Proporcionar comunicaciones entre equipos móviles y fijos
- Excluir cables y conectores entre éstos.
- Brindar la posibilidad de crear pequeñas redes inalámbricas y facilitar la sincronización de datos entre los equipos de trabajo.

1.1.1 EVOLUCIÓN [15] [17] [33]

En 1994 *Ericsson* investigó la viabilidad de utilizar un interfaz de radio, para la interconexión de teléfonos móviles y otros accesorios, con el fin de eliminar los cables existentes entre aparatos relativamente cercanos. El estudio partía de un largo proyecto que investigaba sobre multi-comunicadores conectados a una red celular, hasta que se llegó a un enlace de radio de corto alcance. Conforme avanzaba este proyecto, se estableció que este tipo de enlace podía ser utilizado en un gran número de aplicaciones, ya que tenia como principal virtud el que se basaba en un chip de radio de bajo costo y de corto alcance.

A principios de 1998 se creó el *SIG* (*Special Interest Group*, Grupo de Interés Especial) y estuvo integrado por 5 promotores que fueron: *Ericsson, Nokia, IBM, Toshiba e Intel*. La idea era lograr un conjunto adecuado de áreas de negocio.

Bluetooth se basó en el *SIG* y definió el estándar *IEEE 802.15.1* con el propósito principal, de establecer una interfaz aérea junto con su *software* de control, con el fin de asegurar la interoperabilidad de los equipos entre los diversos fabricantes.

"En la actualidad el *SIG* cuenta con miembros tales como *Motorola, 3Com*, *Lucent* y *Microsoft*, el respaldo de 1900 empresas de tecnología y 2000 empleados (delegados en el Congreso convocado por el *SIG*) de otras tantas empresas que investigan productos y servicios con aplicaciones *Bluetooth*." [13]

1.1.2 ARQUITECTURA BLUETOOTH [34]

La especificación *Bluetooth* utiliza una arquitectura de protocolos que divide las diversas funciones de red en un sistema de niveles. En conjunto, permiten el intercambio transparente de información entre aplicaciones diseñadas de acuerdo con dicha especificación y fomentan la interoperabilidad entre los productos de diferentes fabricantes.

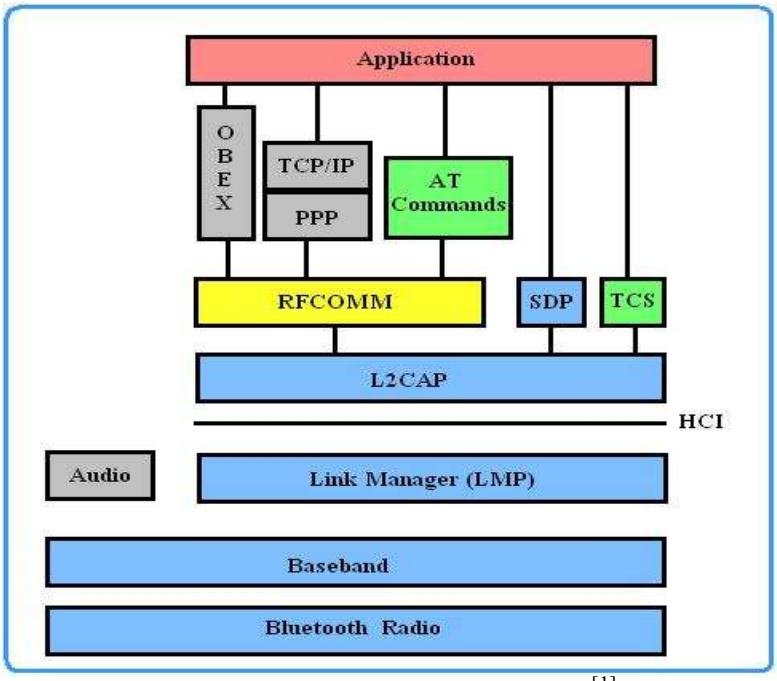


Figura 1.2 Arquitectura *Bluetooth* [1]

Cada aplicación puede operar bajo una estructura de protocolos definida por cada columna como se observa en la figura 1.2, o por un conjunto de ellas. Algunas columnas son usadas sólo como soporte de la aplicación principal, como lo son el

SDP (Protocolo de Descubrimiento de Servicio) y el *TCS* (Especificación de Control Telefónico).

La especificación es abierta, lo que permite el desarrollo de nuevos protocolos de aplicación en las capas superiores, lo cual se traduce en el desarrollo de una gran variedad de servicios por parte de los fabricantes.

Los protocolos pueden ser divididos de la siguiente forma:

- Protocolos Bluetooth Centrales (*BaseBand*, *LMP, L2CAP, SDP*).
- Protocolos de Reemplazo de Cable (*RFCOMM*).
- Protocolos de control de Telefonía (*TCS Binary, AT-Commands*).
- Protocolos Adaptados (*PPP, UDP/TCP/IP, OBEX, WAP, vCard, vCal, IrMC, WAE*).

El Grupo *Bluetooth SIG*, ha desarrollado los protocolos de la primera capa, los cuales son usados por la mayoría de los dispositivos *Bluetooth*. Por otra parte, el *RFCOMM* y el *TCS Binary* fueron desarrollados por el *SIG*, basándose en las especificaciones *ETSI-TS 07.10* y la *ITU-T Q.931*, respectivamente.

1.1.2.1 Capa Radio *Bluetooth*

En este nivel se especifica detalles del interfaz aire como: bandas de frecuencia, arreglos de canales, saltos de frecuencia, esquema de modulación y niveles de potencia.

1.1.2.1.1 Banda de Frecuencia utilizada por Bluetooth [2]

Para que *Bluetooth* opere globalmente, es indispensable que trabaje en una banda no licitada. La banda *ISM* (*Industrial, Scientific and Medical*, Industrial, Científica y Médica) de 2,45 *GHz* cumple con este requisito, con rangos que van de los 2.4 *GHz* a los 2.5 *GHz*, con algunas limitaciones en países como Francia, España y Japón. En la tabla 1.1 se muestra los rangos de frecuencia permitidos en diferentes regiones del mundo.

Ubicación Geográfica	Rango Regulatorio	Canales RF
USA, Europa	2.400 – 2.4835 GHz	F = 2402+ K*MHz , K = 0,....,78
España	2.445 – 2.475 GHz	F = 2449+ K*MHz , K = 0,....,22
Francia	2.4465 – 2.4835 GHz	F = 2454+ K*MHz , K = 0,....,22

Tabla 1.1 Bandas de frecuencia y canales de *RF Bluetooth* [2]

1.1.2.1.2 Espectro Ensanchado por Salto de Frecuencia [24]

Puesto que la banda *ISM* puede ser accedida sin necesidad de licencia, el sistema de radio *Bluetooth* deberá estar capacitado para evitar las múltiples interferencias que se pueden producir. Éstas pueden ser evitadas utilizando un sistema de Espectro Ensanchado por Salto de Frecuencia.

Este sistema divide la banda de frecuencia en 79 canales con un ancho de banda de 1 *MHz* cada canal, donde, los transceptores, durante la conexión van cambiando de uno a otro canal de salto de manera pseudo-aleatoria. Con lo que se consigue que el ancho de banda instantáneo sea muy pequeño y se tenga una propagación efectiva sobre el total del ancho de banda. En la figura 1.3 se observa el esquema de funcionamiento del sistema de (*FHSS)* Espectro Ensanchado por Salto de Frecuencia para *Bluetooth*.

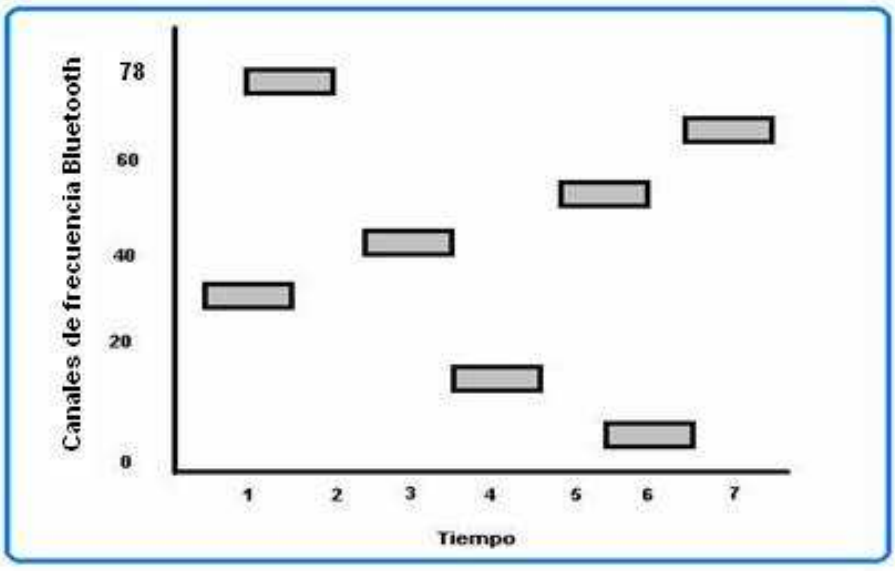


Figura 1.3 *FHSS* Salto de Frecuencia [26]

En conclusión, con el sistema *FHSS*, se pueden conseguir transceptores de banda estrecha con una gran inmunidad a las interferencias.

1.1.2.1.3 Modulación [3] [13]

En la banda de 2.4 *GHz* el ancho de banda para los sistemas *FH* está limitada en 1 *MHz*. El ancho de banda disponible es de 79 *MHz*, por lo que se dispone de 79 canales de salto en América.

Bluetooth utiliza una modulación *GFSK* (*Gaussian Frecuency Shift Keying*, Modulación por Desplazamiento de Frecuencia Gausiana) con un índice de modulación K=0.3. En donde un "1" binario representa una desviación de frecuencia positiva, y un "0" binario representa una desviación de frecuencia negativa. La desviación máxima de frecuencia está entre 140 *KHz* y 175 *KHz*.

La elección de este esquema radica en su robustez y simplicidad de implementación del mismo.

A continuación se detalla el proceso de transmisión entre maestro y esclavo, en una *piconet*.[1]

El canal está dividido en ranuras de tiempo, cada ranura corresponde a una frecuencia de salto y tiene una longitud de 625 *us*.

Cada secuencia de salto en una *piconet* está determinada por la dirección del maestro de la *piconet.* Todos los dispositivos conectados a la *piconet* están sincronizados con el canal en salto y tiempo.

En una transmisión, cada paquete debe estar alineado con el inicio de una ranura y puede tener una duración de 1, 3 o 5 ranuras de tiempo. Durante la transmisión de un paquete la frecuencia es fija. Para evitar fallas en la transmisión, el maestro

[1] Piconet: Dos o más dispositivos *Bluetooth* que comparten un mismo canal forman una *piconet*.

inicia enviando en las ranuras de tiempo pares y los esclavos en las ranuras de tiempo impares. En la figura 1.4 se puede observar este esquema de transmisión.

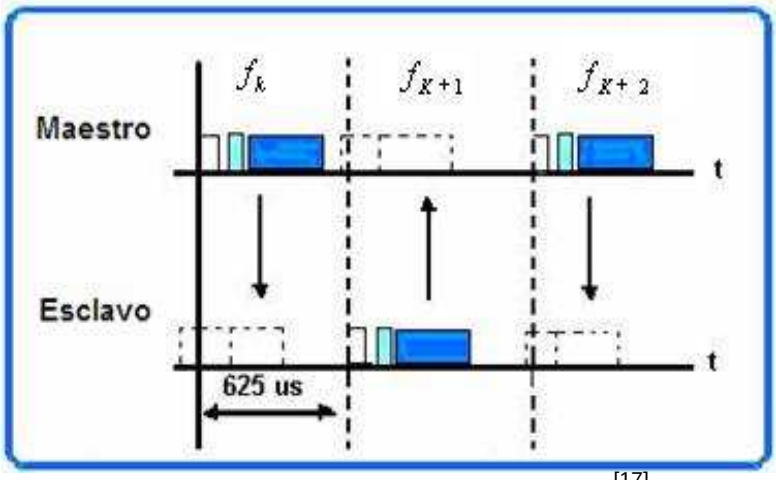


Figura 1.4 Transmisión en una *piconet* [17]

1.1.2.1.4 Potencia [16]

De acuerdo a las especificaciones *Bluetooth* los dispositivos de transmisión se dividen en tres grupos tal como se muestra en la tabla 1.2.

Clase de Transmisor	Potencia Máxima	Potencia Mínima	Alcance Máximo	Control de Potencia
Clase 1	100 mW	1 mW	100 m	Obligatorio
Clase 2	2.5 mW	0.25mW	10 m	Opcional
Clase 3	1mW	----------	10 cm.	----------

Tabla 1.2 Niveles de Emisión en *Bluetooth* [16]

“El equipo receptor debe poseer una sensibilidad de al menos -70 dBm y la tasa de error admisible debe ser menor o igual a 0.1 %” [16]

Los dispositivos de radio usados son de clase 2 que tienen una potencia de transmisión de 2.5 *mW*. La tecnología *Bluetooth* está diseñada para tener un consumo de potencia muy bajo. La tabla 1.3 representa la máxima potencia de salida permitida por regiones de acuerdo a la ubicación geográfica.

Máxima potencia de salida	Localización Geográfica	Documento de Complacencia
1000 mW	NORTE AMERICA	FCC 15.247
100 mW	EUROPA	ETS 300-328
10 mW/MHz	JAPÓN	MPT ordinance 79

Tabla 1.3 Niveles de Potencia de Transmisión para diferentes Regiones [2]

1.1.2.2 Capa Banda Base

En esta segunda capa se define el descubrimiento de dispositivos, establecimientos de conexión de una *piconet*, direccionamiento, formato de paquetes, temporización, control de potencia y comunicaciones asíncronas y síncronas entre pares.

1.1.2.2.1 Piconet [17]

Dos o más dispositivos *Bluetooth* que comparten un mismo canal forman una *piconet*. Para regular el tráfico en el canal cada *piconet* debe tener un maestro y puede tener hasta siete esclavos activos, además pueden haber muchos más esclavos en estado *parked*[2]. Los participantes podrían intercambiar los papeles si una unidad esclava quisiera asumir el papel de maestra. Sin embargo sólo puede haber un maestro en la *piconet* al mismo tiempo. En la figura 1.5 se puede observar una *piconet*.

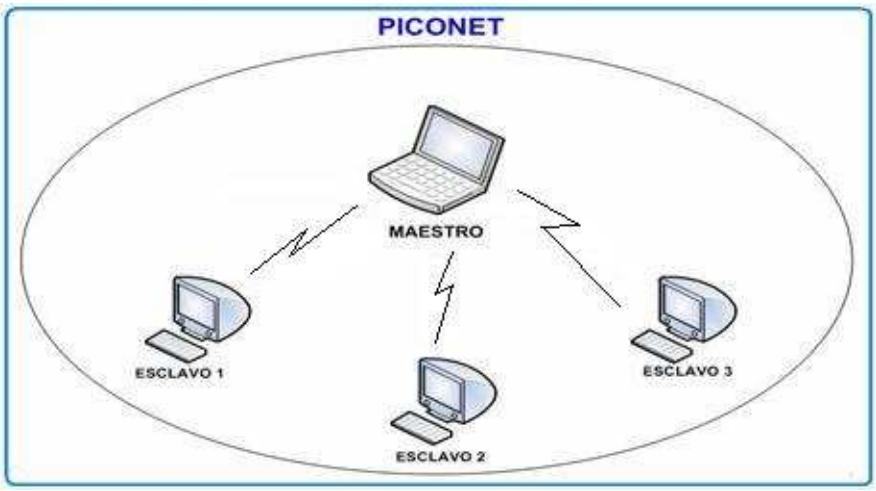


Figura 1.5 *Piconet*

[2] Parked: una unidad en una piconet se encuentra en este modo cuando está sincronizada pero no tiene una dirección MAC.

1.1.2.2.2 Scatternet [3] [17]

Dos o más p*iconets* que comparten una parte de su espacio físico de transmisión (*canal de transmisión*) forman una *scatternet*.

Las s*catternet* permiten aprovechar mejor el ancho de banda, y la velocidad efectiva individual de los usuarios es mucho mayor en la *scatternet* que si todos los usuarios estuviesen conectados a una misma *piconet.*

En la figura 1.6 se puede observar un ejemplo de una red *scatternet*

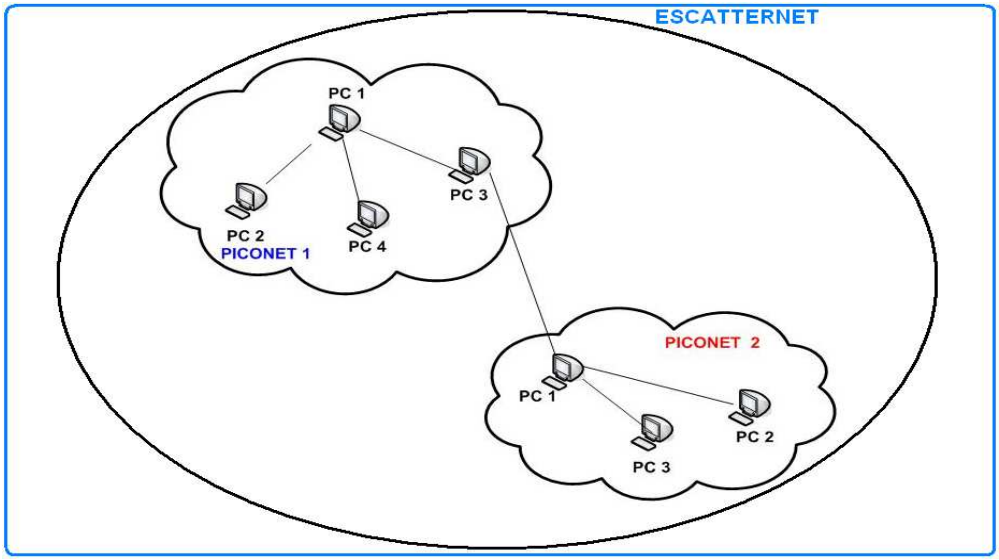


Figura 1.6 *Scatternet*

Las unidades que se encuentran en el mismo radio de cobertura pueden establecer potencialmente comunicaciones entre ellas. Sin embargo, sólo aquellas unidades que realmente quieran intercambiar información comparten un mismo canal creando la *piconet*. Este hecho permite que se creen varias *piconets* en áreas de cobertura superpuestas. A un grupo de *piconets* se le llama *scatternet.*

El rendimiento, en conjunto e individualmente de los usuarios de una *scatternet* es mayor que el que tiene cada usuario cuando participa en un mismo canal de 1 *MHz*.

Se debe tener en cuenta que cuantas más *piconets* se añaden a la *scatternet* su velocidad efectiva disminuye poco a poco, existiendo una reducción por término medio del 10%. Sin embargo el rendimiento que finalmente se obtiene de múltiples *piconets* supera al de una simple *piconet*.

Una estimación bastante simplificada de la velocidad efectiva normalizada es:

$$TH = \left(1 - \frac{1}{79}\right)^{N-1}$$ **Ecuación 1.1** Velocidad Efectiva Normalizada [3]

Donde: TH = Velocidad Efectiva
N = Número de *Piconets*

La información intercambiada por una *piconet* sólo es compartida por los miembros de la *piconet*, no por toda la *scatternet*. Una unidad puede participar en distintas *piconets* por medio de *TDD* (Duplexación por división de tiempo) pero esta unidad solo puede ser maestra en una sola *piconet*.

1.1.2.2.3 Enlace Físico [3] [17]

En la especificación *Bluetooth* se han definido dos tipos de enlace que permitan soportar incluso aplicaciones multimedia:

- Enlace sincrónico orientado a conexión (*SCO*)
- Enlace asíncrono no orientado a conexión (*ACL*)

a. Enlace Sincrónico Orientado a Conexión (*SCO*)

Los enlaces *SCO* soportan conexiones simétricas, punto a punto y conmutación de circuitos, estos enlaces se usan en conexiones de voz, estos enlaces se encuentran definidos en el canal de transmisión, reservándose dos ranuras consecutivas (envío y retorno) en intervalos fijos. La reserva de las ranuras la realiza el maestro cuando se establece la conexión entre el maestro y el esclavo. La conexión *SCO* debe establecerse explícitamente después de que se ha creado la *piconet*.

El maestro envía un mensaje de establecimiento a conexión al esclavo, usando el protocolo de gestión del enlace (*Link Management*). En una conexión *SCO* no se permiten paquetes multi-ranura es decir paquetes que ocupen 2 o más ranuras consecutivas. En una *piconet* el maestro puede contener hasta tres enlaces *SCO* con un solo esclavo o con esclavos diferentes y el esclavo puede mantener hasta dos enlaces *SCO* si los maestros con los que se comunica son diferentes.

b. Enlace Asíncrono no Orientado a Conexión (*ACL*)

Los enlaces *ACL* soportan conexiones simétricas o asimétricas, punto a multipunto y con conmutación de paquetes, típicamente usadas en la transmisión de datos. Por defecto cuando, se establece la *piconet* la unidad maestra establece una conexión *ACL* con las unidades esclavas.

Un enlace *ACL* no reserva ancho de banda, usa ranuras por demanda de 1, 3 y 5 ranuras consecutivas también usa control de errores para garantizar la entrega de los paquetes. La máxima velocidad de transmisión se obtiene enviando paquetes sin protección, de 5 ranuras, con capacidad de asignación asimétrica 721 *Kbps* en un sentido y 57,6 *Kbps* en el otro.

1.1.2.2.4 Formato del Paquete Bluetooth [13]

La figura 1.7, representa el formato de paquete general el cual consta de tres campos: código de acceso, cabecera y carga útil.

Figura 1.7 Paquete *Bluetooth* [17]

a. Código de acceso [14]

Usado para sincronización, identificación. Todos los paquetes que son enviados sobre el canal de transmisión en la *piconet* están antepuestos del mismo código de acceso. Existen tres tipos diferentes de código de acceso:

- **Código de Acceso al Canal (*CAC*):** identifica una *piconet* y es incluido en todos los paquetes intercambiados dentro de la *piconet*.

- **Código de Acceso de Dispositivo (*DAC*):** para procedimientos de señalización especiales, *paging* (utilizado para la localización y señalización de un dispositivo), entre otros.

- **Código de Acceso de Búsqueda (*IAC*): *(IAC) de tipo general*** se usa para descubrir otros dispositivos *Bluetooth* dentro de una *piconet*, o ***(IAC) dedicado*** cuando se quiere descubrir dispositivos *Bluetooth* específicos.

b. Cabecera de paquete [13]

La cabecera contiene información del control del enlace y está formada por seis campos:

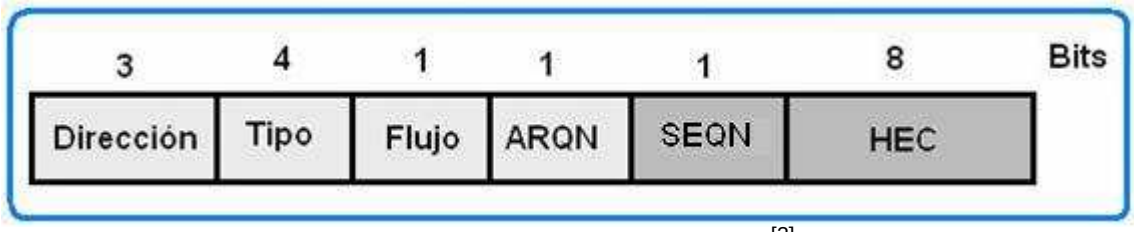


Figura 1.8 Cabecera del Paquete [3]

- **Dirección:** una dirección de dispositivo para distinguirlo de los demás dispositivos activos en la *piconet.* Se tienen 3 bits porque se pueden tener hasta 7 dispositivos activos en la *piconet*. El valor cero se tiene reservado por el maestro para enviar información a todos los esclavos en la *piconet*.

Dirección *M_ADDR*: permite identificar a los esclavos que están activos dentro de una *piconet*. Si la información está dirigida a todos los esclavos entonces los 3 bits son ceros.

- **Tipo:** define el tipo de paquete enviado. Éste dependerá del enlace asociado al paquete (*SCO* o *ACL*). El campo *Tipo* también indica el número de ranuras que ocupa el paquete actual que puede ser de 1, 3 o 5 ranuras consecutivas en caso de un enlace *ACL*.

- **Flujo:** usado para el control de flujo de los paquetes sobre el enlace *ACL*, para notificar al emisor cuando el *buffer* del receptor está lleno.

  Si el *buffer* del receptor para el enlace *ACL* está lleno, se devuelve una señal de parada para detener la transmisión de datos (Flujo = 0), esta señal de parada sólo se aplica a paquetes *ACL*, los paquetes con información de control de enlace y *SCO* se siguen recibiendo normalmente. Cuando se vacía el *buffer* del receptor, se devuelve una señal de continuar (Flujo = 1).

- ***ARQN*:** usado para informar si una transferencia es exitosa o no, *ARQN* puede ser un *ACK* (*ARQN*=1) si la recepción fue exitosa, o un *NAK* (**ARQN**=0) si la recepción fue errónea, en este caso el paquete se retransmite.

- ***SEQN*:** permite distinguir si el paquete enviado es nuevo o es un paquete retransmitido.

- ***HEC*:** permite verificar la integridad de la cabecera del paquete.

c. Carga útil [4]

La carga útil de un paquete puede ser dividida en dos campos:

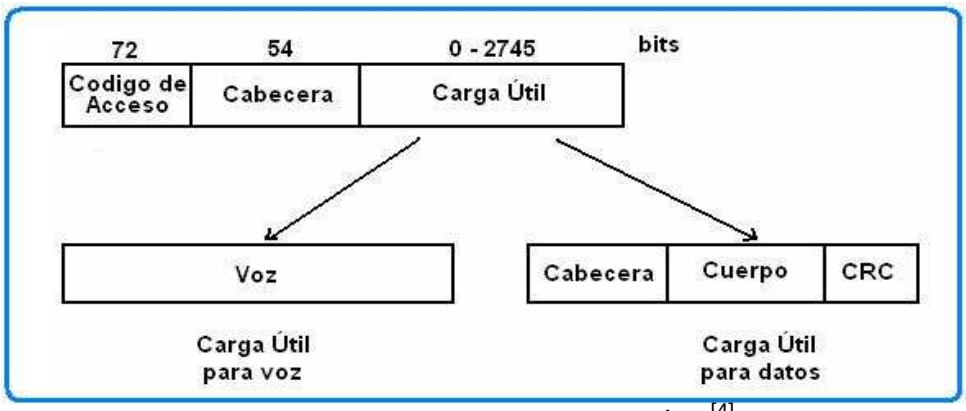


Figura 1.9 Tipos de Datos en la Carga Útil [4]

- **Campo de Voz:** consta de una carga útil, exclusiva para la transmisión de voz. Este campo no posee una cabecera, no realiza chequeo de errores.

- **Campo de Datos:** consta de tres partes, cabecera de carga útil, datos de carga útil, y código *CRC.*

c.1) División de la Carga Útil [4]

Como se puede observar en la figura 1.10, la carga útil destinada para el envío de datos se divide en tres campos que son cabecera, cuerpo y *CRC*.

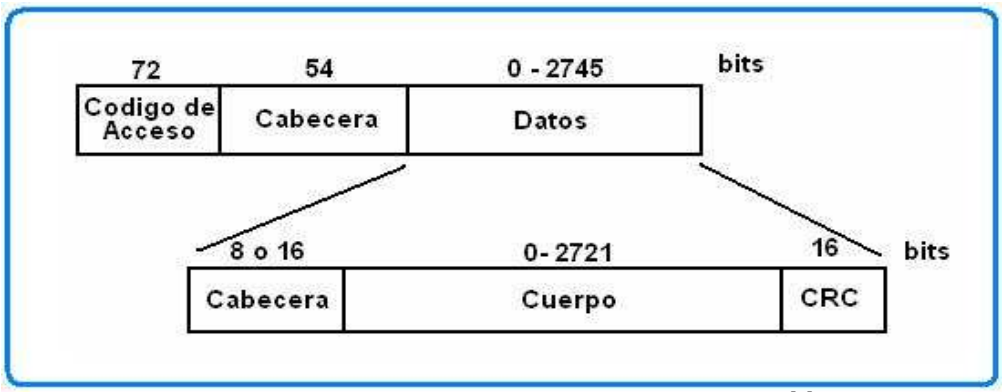


Figura 1.10 División de la Carga Útil para Datos [4]

- **Cabecera de la Carga Útil:** consta de 8 o 16 bits dependiendo si son paquetes de una sola ranura o paquetes multi-ranura.

Figura 1.11 Cabecera de la Carga Útil para Datos [4]

La cabecera de la carga útil consta de los siguientes campos:

- **L_CH:** consta de 2 bits y especifica el canal lógico.

- **Flujo:** campo de un 1 bit y permite realizar el control de flujo de los canales lógicos a nivel de *L2CAP*.

- **Longitud:** la longitud en bytes del cuerpo. Consta de 5 o 9 bits dependiendo si es un paquete que ocupa una sola ranura o es un paquete multi-ranura.

- **Cuerpo de la Carga Útil:** es un campo que va depender del número de bits que se van a transmitir y su longitud puede variar entre 0 y 2721 bits.

- **CRC:** es un campo de 16 bits para chequeo de errores.

1.1.2.2.5 Tipos de Paquetes [3]

Los tipos de paquetes se dividen en paquetes de control y de información.

Los paquetes de control son de cuatro tipos:

- **ID:** paquete de identificación. Consiste solo en el código de acceso.

- **NULL:** consiste en el código de acceso y la cabecera. Sirve para llevar información solo de control.

- **POLL:** similar al anterior; usado por el maestro para invitar a los esclavos.

- **FHS:** paquete de sincronización. Sirve para intercambiar información de identidad e información del reloj.

Los 12 códigos de paquetes restantes sirven para definir el tipo de servicio que se entrega (sincrónico o asincrónico) y el tamaño en ranuras del paquete. Los datos pueden o no ser protegidos con *FEC* (1/3 o 2/3).

Considerando la transmisión sin *FEC* se puede lograr una máxima tasa asimétrica de 723.2 Kbps con un enlace de retorno de 57.6 Kbps.

En la tabla 1.4 se presentan algunos paquetes para transmisión sincrónica y asincrónica con sus respectivas velocidades.

PAQUETES			
TIPO	**SIMÉTRICO**	**ASIMÉTRICO**	
		ENVÍO	**RETORNO**
DM1	108.8	108.8	108.8
DH1	172.8	172.8	172.8
DM3	256.0	384.0	54.4
DH3	384.0	576.0	86.4
DM5	286.7	477.8	36.3
DH5	432.8	721.0	57.6

Tabla 1.4 Paquetes para Transmisión Simétrico y Asimétrico [3]

Máximas tasas de transmisión promedio en Kbps.

- **DM**x: Paquetes de largo x slots, con *FEC*
- **DH**x: Paquetes de largo x slots, sin protección

1.1.2.2.6 Canales Lógicos [4]

Los canales lógicos definidos en *Bluetooth* son usados para actividades de control y para transporte de datos de usuario. Estos canales lógicos existen sobre los canales físicos *SCO* (Enlace Sincrónico Orientado a Conexión) o *ACL* (Enlace Asincrónico no Orientado a Conexión).

- **Canal de Control LC (*Link Control*):** implementado a través de la cabecera del paquete excepto en los paquetes de identificación (*ID*) que carecen de encabezado. Se encarga de transportar información de bajo nivel tal como:

  - ✓ Caracterización de la carga útil es decir el tipo de paquete que se envía.
  - ✓ Peticiones de repetición automática
  - ✓ Control de flujo.

- **Canal de Control LM (*Link Manager*):** transporta información de control para la administración del enlace entre el maestro y uno o más esclavos. Este canal lógico es transportado en la carga útil y puede estar presente en enlaces *SCO* o *ACL* soportando tráfico *LMP* (Protocolo de Administración del Enlace)

- **Canal de Usuario UA (*User Asynchronous*)**: transporta datos asíncronos de usuario. Generalmente es transportado en un enlace *ACL*.

- **Canal de usuario UI (*User Isochronous*):** se encarga de transportar datos isócronos de usuario. Estos datos se caracterizan porque la información de temporización está incluida en la cadena de datos. Los datos isócronos necesitan temporización de forma precisa como es el caso de enviar audio comprimido sobre un enlace *ACL*.

- **Canal de Usuario US (*User Sinchronous*):** transporta datos de usuario síncronos y están presentes sobre enlaces físicos *SCO*.

1.1.2.2.7 Establecimiento de la Conexión [4]

Para el establecimiento de una conexión en *Bluetooth* los dispositivos pueden estar en ciertos estados como son:

- **Inquiry:** el estado de *Inquiry* es un estado de búsqueda que es utilizado para descubrir otros dispositivos.

- **Scan:** cuando un dispositivo *Bluetooth* está en modo *STANDBY* (dormida), periódicamente escucha el canal, esperando a ser descubiertos por otros dispositivos.

- **Paging:** este estado es también llamado como un estado de paginación o localización. Es utilizado, generalmente, luego del estado *Inquiry* para establecer las conexiones.

a. Conexión entre dos dispositivos:

Figura 1.12 Conexión de Dispositivos

En una conexión de dispositivos *Bluetooth*, estos dispositivos pueden estar en diferentes estados como son: estado *Inquiry*, *Scan*, *Paging*. Dependiendo del estado de cada dispositivo la conexión se la puede realizar utilizando los siguientes procedimientos.

a.1. Inquiry [4]

Durante el procedimiento *Inquiry* el nodo fuente invita al nodo destino con un mensaje *inquiry*, luego de escuchar el mensaje de invitación el nodo destino responde la invitación aceptando la comunicación.

La información que envía en el mensaje el nodo fuente es un paquete de *ID* (paquete de identificación) con un código *IAC* (Código de Acceso de Búsqueda). El paquete *ID* es un paquete que no tiene cabecera ni carga útil y el código *IAC* es un código común para todos los dispositivos *Bluetooth*.

Si el destino recibe el mensaje *Inquiry* responde con un paquete *FHS* (Paquete de Sincronización) que contiene la dirección del dispositivo e información del reloj. Para evitar colisiones, los destinos difieren sus respuestas utilizando un temporizador de *backoff*.

La fuente captura información básica enviada por el destino y esta información es utilizada para hacer el *paging* con el dispositivo destino seleccionado.

A continuación en la figura 1.13 se muestra el proceso de *Inquiry*:

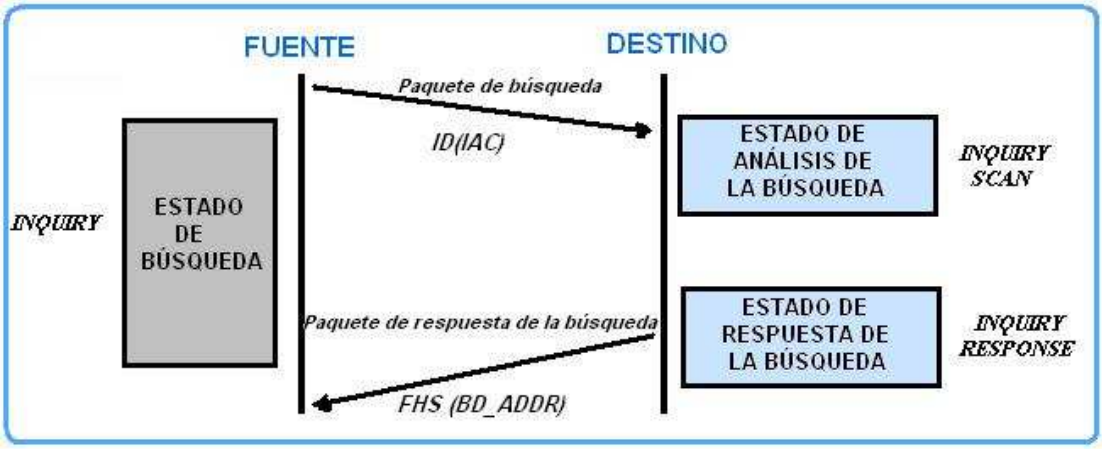


Figura 1.13 Proceso de *Inquiry* [4]

Pasos para el procedimiento del *Inquiry*:

- La fuente envía los paquetes de *inquiry*, que llevan la información de identificación y un código de acceso común.

- El destino que recibe los paquetes de *inquiry*, debe estar en el estado *Scan Inquiry*, en el que está atento a recibir los paquetes de *Inquiry*.

- El destino entra al estado de respuesta de la búsqueda o también llamado *Inquiry Response*, en el cual manda una contestación al mensaje de *Inquiry* enviado por la fuente.

- Una vez que el nodo destino responde a un *Inquiry*, se mueve al estado *Page Scan*

a.2. Paging [4]

La fuente envía un mensaje de *paging* que es único al destino. El destino contesta inmediatamente sin necesidad de esperar un periodo de *backoff*.

Para establecer una conexión es necesario conocer la dirección del dispositivo *Bluetooth* (*BD_ADDR*). Esta dirección es también utilizada para el cálculo de la secuencia de salto del *paging* (*page frequency- hopping sequence*), con el cual se contacta al dispositivo durante el *paging.*

El paquete que envía la fuente es un paquete *ID* (Paquete de Identificación) al cual se le ha añadido un Código de Acceso del Dispositivo (*DAC*) este código de acceso del dispositivo contiene una parte de la dirección del dispositivo (*BD-ADDR*).

Luego de recibir la respuesta al *paging*, la fuente se convierte en el maestro y el destino en esclavo de la nueva *piconet*.

A continuación se representa en la figura 1.14 el procedimiento del *paging*:

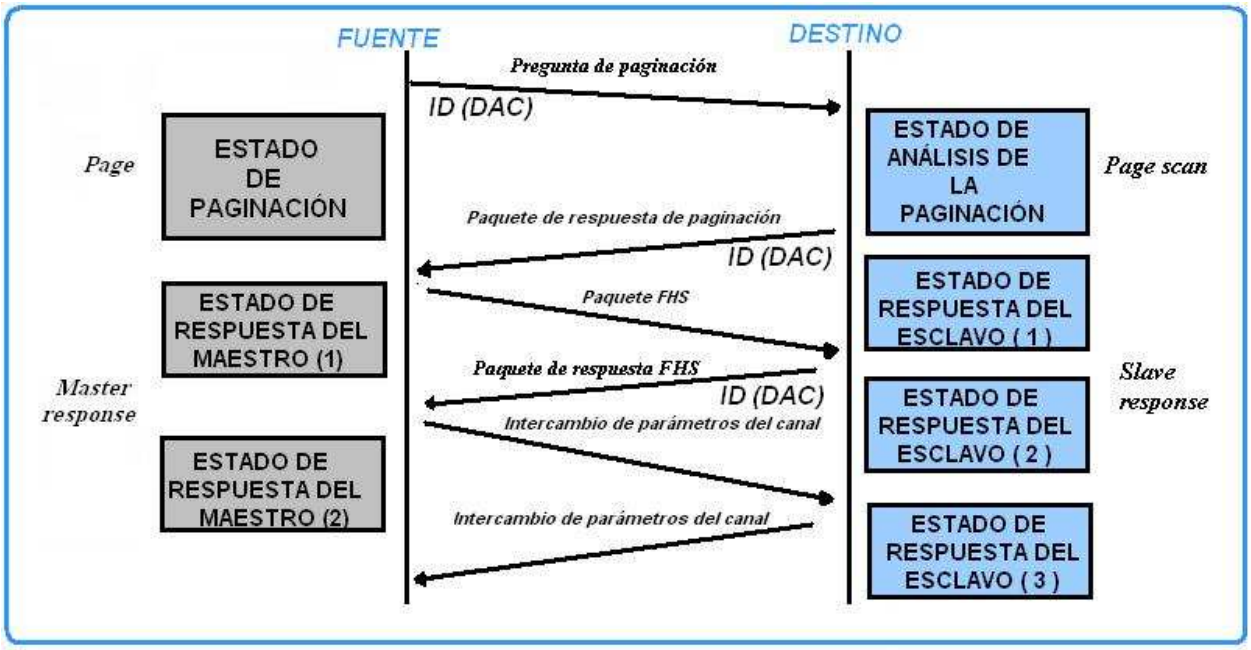


Figura 1.14 Proceso de *Paging* [4]

Procedimiento de *paging*:

- La fuente que se encuentra en el estado de *Paging* envía un *page* al destino, que contiene un paquete de Identificación (*ID*) pero esta vez con un Código de Acceso del Dispositivo.

- El destino que se encuentra en el estado de *Page Scan*[3] recibe el *page*.

- El destino manda una contestación a la fuente que se encuentra en el estado de *Master Response*. Esta contestación contienen un paquete idéntico al recibido (*ID*)

- La fuente que está en el estado de *Master Response* manda un paquete de sincronización (*FHS*) al destino. Este paquete de sincronización contiene la dirección del dispositivo *Bluetooth* fuente y el valor de su reloj de tiempo real *Bluetooth*.

- El destino que se encuentra en el estado *Slave Response* manda una segunda contestación a la fuente, confirmando la recepción del paquete de

[3] **Page Scan**: Subestado en el cual el esclavo escucha el código de acceso durante el tiempo que dura una ventana de scan.

Sincronización (*FHS*). Esta respuesta es el mismo paquete de identificación con el mismo Código de Acceso del dispositivo.

- La fuente y el destino que se encuentran en los estados *Master Response* y *Slave Response* respectivamente, intercambian los parámetros de sincronización del Canal. El destino empieza a utilizar la secuencia de salto definida por el maestro.

El maestro puede continuar realizando invitaciones a otros dispositivos.

a.3. Admisión de un nuevo esclavo [4]

Los pasos para la admisión de un nuevo esclavo son relativamente complejos. El maestro podría tomar una de las siguientes opciones:

- Empezar a descubrir nuevos nodos e invitarlos a unirse a la *piconet*.
- Esperar a ser descubierto por otros nodos, permaneciendo en un estado *Scan*.

En ambos casos, la comunicación en la *piconet* debe suspenderse durante el proceso de *Inquiry* y *Paging*. El retardo involucrado en la admisión de un nuevo nodo puede ser grande, especialmente si el maestro no pasa al estado de *Inquiry* o *Scan* frecuentemente; esto provoca una degradación en la capacidad de la *piconet*.

La operación de una *piconet* puede entenderse en base a “dos estados de operación” principales que se definen durante el establecimiento de una conexión.

- ***Standby:*** estado en el que se encuentra una unidad *Bluetooth* por defecto, es un estado de bajo consumo de potencia en el cual solo el reloj local está activo. Los dispositivos que se encuentran en este estado periódicamente entran al estado *Inquiry Scan.*

- ***Connection:*** en este estado una estación *Bluetooth* puede estar conectada a una *piconet* ya sea como esclavo o como maestro.

Todos los procedimientos se pueden combinar en un solo diagrama de estados, el mismo que se muestra en la figura 1.15.

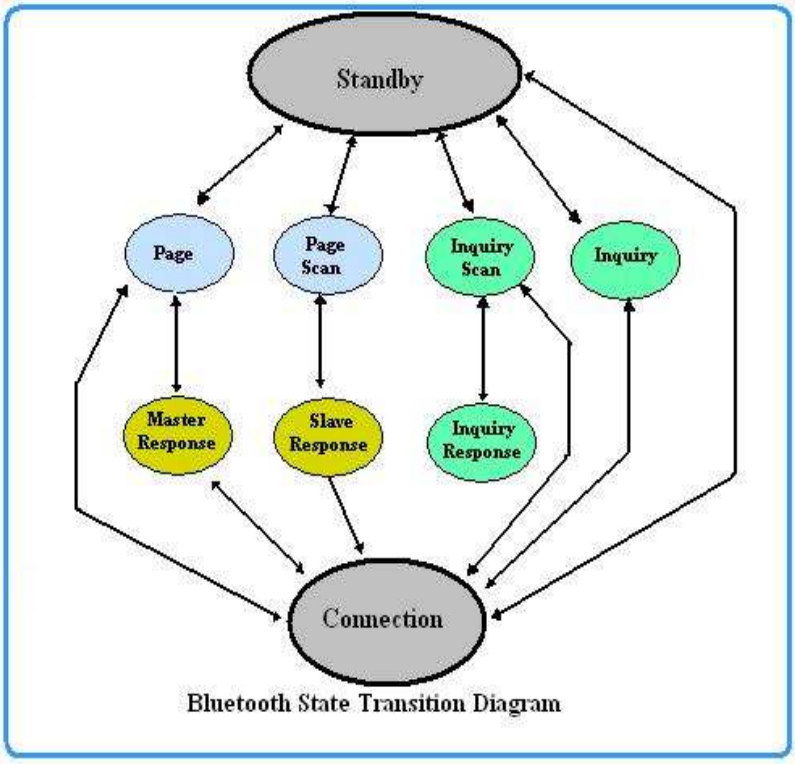


Figura 1.15 Diagrama de Estados de Transición *Bluetooth* [4]

1.1.2.2.8 Modos de Ahorro de Potencia [3] [14]

Bluetooth es un estándar que permite un ahorro considerable de energía no solo por la baja potencia que utiliza cada dispositivo sino también por la manera como éstos establecen una conexión.

Una unidad en modo *SCAN* se activa durante un pequeño periodo de tiempo de 10 [*ms*] para reducir el consumo de potencia mientras está en modo "*STANDBY*".

El modo *PAGE* necesita un consumo mayor de energía debido a que lleva el peso de la incerteza de frecuencia y tiempo. Se prefiere esta configuración debido a que el modo *PAGE* es más infrecuente que el modo *STANDBY*. También para

ahorrar energía, las unidades *Bluetooth* que detectan un paquete que no está dirigido hacia ellas siguen en estado *STANDBY*.

Además el esquema *FH* (Salto de Frecuencia) es robusto en términos de sincronización por lo que no es necesario que se estén constantemente enviando señales de temporización, lo que también reduce el consumo de energía. *Bluetooth* además define una serie de técnicas para ahorrar energía:

- **MODO *HOLD*:** el maestro puede ordenar al esclavo quedarse en modo *HOLD*. Durante este periodo no hay comunicación posible entre esclavo y maestro. Cuando el periodo expira el esclavo vuelve al canal y permanece sincronizado.

- **MODO *PARK*:** el esclavo también puede ser puesto en modo *PARK*. En este caso el esclavo entra a un ciclo de trabajo en donde los intervalos de escucha del maestro son más largos.

- **MODO *SNIFF*:** el esclavo no escucha todas las ranuras de tiempo, sino que solo escucha algunas. Para entrar al modo *SNIFF*, el esclavo y maestro deben acordar en qué ranuras el esclavo pondrá atención al canal.

1.1.2.3 Protocolo de Administración del Enlace *LMP* [14]

El protocolo *LMP* se usa para establecer y controlar un enlace. Las señales son interpretadas y filtradas por el protocolo *LMP* en el lado del receptor, y no se propagan a las capas superiores.

Los mensajes *LMP* son usados para el inicio, seguridad y control del enlace. Estos mensajes de administración del enlace tienen una prioridad mayor que los datos del usuario.

1.1.2.3.1 Emparejamiento con el Protocolo LMP

Si en el momento de iniciar el emparejamiento dos dispositivos *Bluetooth* no tienen una clave de enlace común, entonces se crea una clave de inicialización denominada *Kinit* basada en un número *PIN*, un número aleatorio y una dirección del dispositivo *Bluetooth* (*BD_ADDR*).

Para el procedimiento de emparejamiento existen cinco posibilidades:

- Contestador acepta el procedimiento “emparejamiento”
- Contestador tiene un número *PIN*.
- Contestador rechaza el procedimiento “emparejamiento”
- Creación de una clave de enlace.
- Intentos repetidos

1.1.2.3.2 Características Soportadas en el enlace

La radiocomunicación *Bluetooth* y el controlador de enlace pueden soportar solo una parte de los tipos de paquetes y características descritas en las especificaciones de Banda Base y Radio de *Bluetooth*. Las características soportadas pueden ser requeridas en cualquier momento, siguiendo un procedimiento exitoso de búsqueda en banda base. Cuando se hace un requerimiento, éste debe ser compatible con las características soportadas del otro dispositivo.

1.1.2.3.3 Requerimiento de Nombre del enlace LMP

El protocolo *LMP* soporta requerimiento de nombre a otro dispositivo *Bluetooth*, y es un nombre de usuario-amigo asociado al dispositivo.

1.1.2.3.4 Terminación del emparejamiento de LMP

La conexión entre dos dispositivos *Bluetooth* puede ser terminada en cualquier momento por el dispositivo maestro o por el dispositivo esclavo.

1.1.2.3.5 Establecimiento de Conexión de LMP

Después del procedimiento de búsqueda, el dispositivo maestro debe invitar al dispositivo esclavo.

1.1.2.3.6 Modos de Prueba de LMP

Este protocolo tiene *PDUs* para soportar diferentes modos de prueba *Bluetooth*, los cuales se usan para certificaciones y pruebas de cumplimiento de banda base y radio *Bluetooth*.

1.1.2.4 Interfaz del Controlador de Host (*HCI*) [13] [14]

Permite acceder al estado y los registros de control del aparato, además de proporcionar un método uniforme de acceder a todas las funciones de la banda base *Bluetooth*.

La sección *HCI* tiene dos funciones en la especificación *Bluetooth*:

- Definir las bases de una interfaz física para un módulo externo *Bluetooth*.
- Definir las funciones de control necesarias para todas las implementaciones *Bluetooth*.

El computador recibe notificaciones asincrónicas de eventos *HCI* independientemente de que capa de transporte se usa. Los eventos *HCI* son usados para notificar al computador cuando algo ocurre. Al descubrir éste, que ha ocurrido un evento, analizará un paquete recibido para determinar qué tipo de evento *HCI* se tiene.

1.1.2.5 Protocolo de Control y Adaptación de Enlace Lógico (*L2CAP*) [13]

L2CAP se encuentra sobre el protocolo de gestión de enlace (*LMP*) y reside en la capa de enlace de datos. *L2CAP* permite a protocolos de niveles superiores y a aplicaciones, la transmisión y recepción de paquetes de datos *L2CAP* de hasta 64

Kbytes, con capacidad de multiplexación de protocolo, segmentación y reensamble, y abstracción de grupos.

Para cumplir sus funciones, *L2CAP* espera que la banda base suministre paquetes de datos en los dos sentidos al mismo tiempo, que realice el chequeo de integridad de los datos y que reenvíe los datos hasta que hayan sido reconocidos satisfactoriamente. Las capas superiores que se comunican con *L2CAP* son por ejemplo el protocolo de descubrimiento de servicio (*SDP*), el *RFCOMM* y el control de telefonía (*TCS*).

1.1.2.5.1 Canales Lógicos de L2CAP

L2CAP está basado en el concepto de canales. Se asocia un identificador de canal, *CID*, a cada uno de los puntos finales de un canal *L2CAP*. Los *CIDs* están divididos en dos grupos, uno con identificadores reservados para funciones *L2CAP* y otro con identificadores libres para implementaciones particulares. Los canales de datos orientados a la conexión representan una conexión entre dos dispositivos, donde un *CID* identifica cada punto final del canal.

Los canales no orientados a la conexión limitan el flujo de datos a una sola dirección. La señalización de canal es un ejemplo de un canal reservado. Este canal es usado para crear y establecer canales de datos orientados a la conexión y para negociar cambios en las características de esos canales.

1.1.2.5.2 Multiplexación de Protocolo

L2CAP soporta Multiplexación de Protocolo, ya que el protocolo de banda base no soporta ningún campo "*TYPE*" que identifica al protocolo de capa superior como protocolos de descubriendo de servicio *SDP, RFCOMM* y control de telefonía.

1.1.2.5.3 Segmentación y Reensamblado

Los paquetes de datos definidos por el protocolo banda base están limitados en tamaño. Los paquetes grandes *L2CAP* deben ser segmentados en varios

paquetes más pequeños antes de transmitirse y luego deben ser enviados a la gestión de enlace.

En el receptor los paquetes pequeños recibidos de la banda base son reensamblados en paquetes *L2CAP* más grandes. Varios paquetes banda base recibidos pueden ser reensamblados en un solo paquete *L2CAP* seguido de un simple chequeo de integridad.

La segmentación y reensamblado de paquetes, es necesaria para soportar protocolos con paquetes grandes, que los soportados por la banda base, la figura 1.16 muestra la segmentación *L2CAP*.

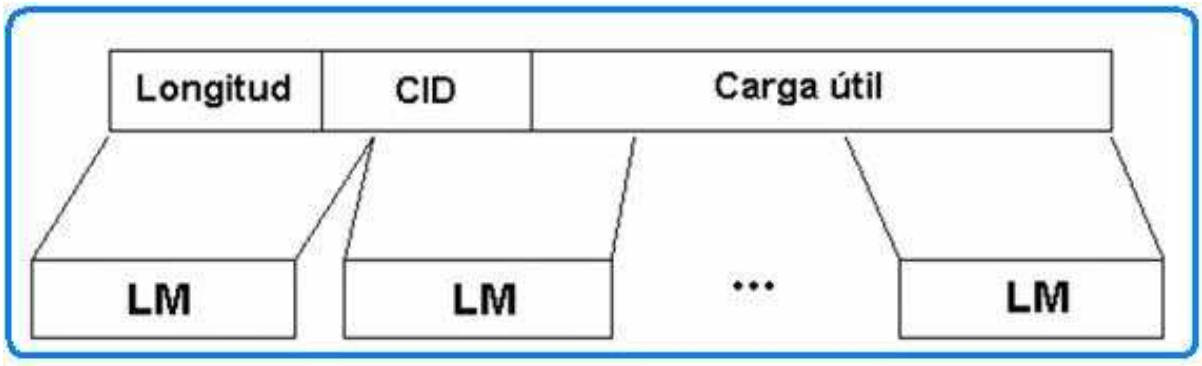


Figura 1.16 Segmentación L2CAP [13]

1.1.2.5.4 Eventos de L2CAP

Todos los mensajes que entran en la capa *L2CAP*, son llamados eventos. Los eventos se encuentran divididos en cinco categorías: indicaciones y confirmaciones de capas inferiores, peticiones de señal y respuestas de capas *L2CAP*, datos de capas *L2CAP*, peticiones y respuestas de capas superiores, y eventos causados por expiraciones de tiempo.

1.1.2.5.5 Formato del paquete de datos

L2CAP está basado en paquetes pero sigue un modelo de comunicación basado en canales. Como se puede observar en la figura 1.17, los paquetes de canal

orientado a la conexión están divididos en tres campos: longitud de la información, identificador de canal, e información.

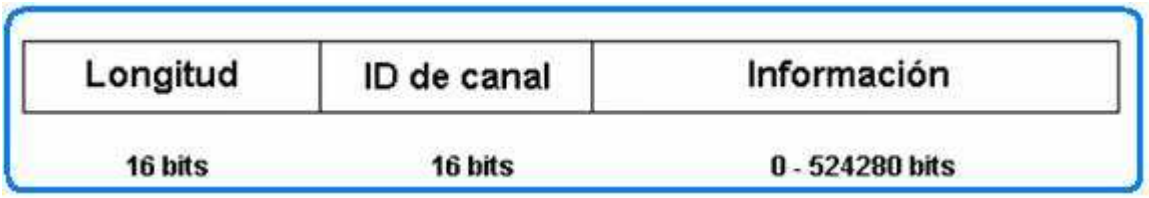


Figura 1.17 Paquete *L2CAP* [13]

Los paquetes de canal de datos no orientados a conexión son iguales a los paquetes orientados a conexión pero adicionalmente incluyen un campo con información multiplexada de protocolo.

1.1.2.6 Protocolo de Descubrimiento de Servicio SDP [4] [13]

El protocolo de descubrimiento de servicio, brinda a las aplicaciones recursos para descubrir qué servicios están disponibles y determinar las características de dichos servicios.

1.1.2.6.1 Descripción General

El *SDP* ofrece a los clientes la facilidad de averiguar sobre servicios que sean requeridos, basándose en la clase de servicio o propiedades específicas de estos servicios.

Los dispositivos *Bluetooth* que usan el *SDP* pueden ser vistos como un servidor y un cliente. El servidor posee los servicios y el cliente es quien desea acceder a ellos. En el *SDP* esto es posible ya que el cliente envía una petición al servidor y el servidor responde con un mensaje. El *SDP* solamente soporta el descubrimiento del servicio, no la llamada del servicio.

1.1.2.6.2 Registros de Servicio

Los registros de servicio contienen propiedades que describen un servicio determinado. Cada propiedad de un registro de servicio consta de dos partes, un identificador de propiedad y un valor de propiedad. El identificador de propiedad

es un número único de 16 bits que distingue cada propiedad de servicio de otro dentro de un registro. El valor de propiedad es un campo de longitud variable que contiene la información.

1.1.2.7 Capa *RFCOMM* [5] [13]

El protocolo *RFCOMM* brinda emulación de puertos seriales sobre el protocolo *L2CAP*, éste soporta hasta 60 puertos emulados simultáneamente. Dos unidades *Bluetooth* que usen *RFCOMM* en su comunicación pueden abrir varios puertos seriales emulados, los cuales son multiplexados entre sí, la figura 1.18 muestra el esquema de emulación para varios puertos seriales.

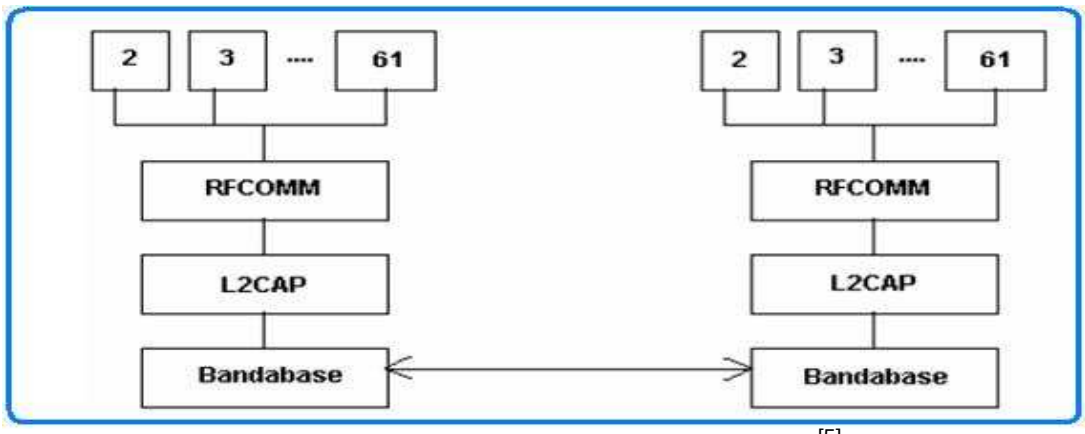


Figura 1.18 Puertos emulados por *RFCOMM* [5]

Muchas aplicaciones hacen uso de puertos seriales. El *RFCOMM* está orientado a hacer más flexibles estos dispositivos, soportando una fácil adaptación de comunicaciones seriales utilizando *Bluetooth*. Un ejemplo de una aplicación de comunicación serial es el protocolo punto-a-punto (*PPP*). El *RFCOMM* tiene construido un esquema para emular el cable que se utiliza en una transmisión serial de datos y usa a *L2CAP* para cumplir con el control de flujo requerido por alguna estación.

1.1.3 PERFILES BLUETOOTH [13]

Desde que se inició la especificación del estándar *Bluetooth*, una de las principales preocupaciones de *SIG* fue garantizar la interoperabilidad total entre

dispositivos de distintos fabricantes, siempre que éstos compartan iguales perfiles.

Los perfiles especifican cómo utilizar el conjunto de protocolos *Bluetooth* para implementar una solución que trabaje sin problemas con otras marcas. En cada uno se establecen opciones y parámetros, además de detallar cómo usar los distintos procedimientos de los diversos estándares que se encuentren implicados. En la figura 1.19 se observa la estructura de los perfiles *Bluetooth*.

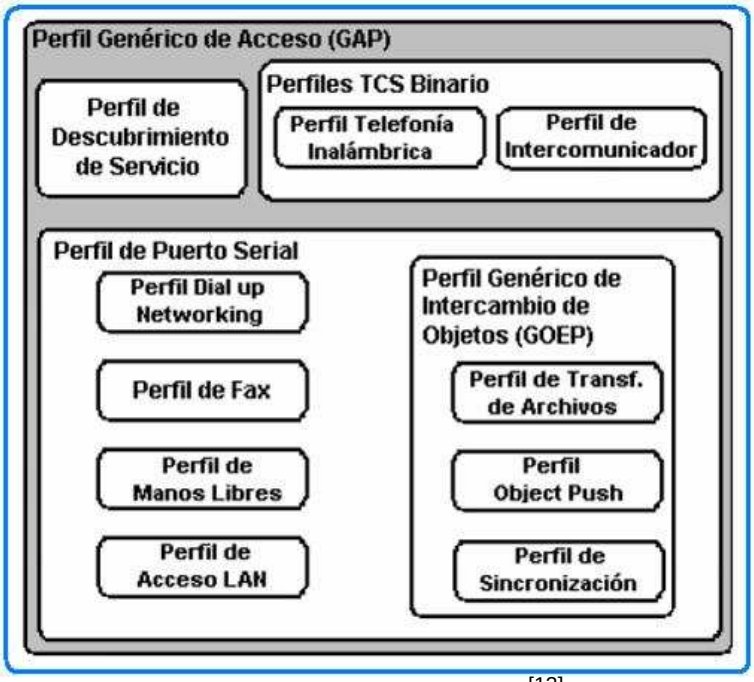


Figura 1.19 Perfiles *Bluetooth* [13]

A continuación se hace una breve descripción de algunos de los perfiles *Bluetooth*:

- **Perfil Genérico de Acceso**

  Define los procedimientos generales para el descubrimiento y establecimiento de conexión entre dispositivos *Bluetooth. A*segurando que cualquier par de dispositivos *Bluetooth* puedan intercambiar información para descubrir qué tipo de aplicaciones soportan las unidades.

- **Perfil Genérico de Intercambio de Objetos *GOEP***

  Este perfil puntualiza los protocolos y procedimientos usados por aplicaciones para ofrecer características de intercambio de objetos. Los dispositivos más comunes que usan este modelo son agendas electrónicas, *PDAs*, teléfonos celulares y teléfonos móviles. El *GOEP* es dependiente del perfil de puerto serial.

- **Transferencia de Archivos**

  Soporta la transferencia de directorios, archivos, documentos, imágenes, y formatos de *streaming*[4] Además soporta la exploración de directorios en el dispositivo remoto.

- **Perfil de Puerto Serial**

  Define los requerimientos necesarios para establecer una conexión de cable serial emulada usando *RFCOMM* entre dos dispositivos *Bluetooth* similares.

- **Perfil de Acceso a una *LAN***

  Permite a los dispositivos de una *piconet* conectarse a una *LAN* como que estuviera conectado a un cable. Usando el protocolo punto-a-punto, *PPP* sobre *RFCOMM*.

- **Perfil de Aplicación de Descubrimiento de Servicio**

  Define los protocolos y procedimientos para descubrir los servicios proporcionados por otra unidad *Bluetooth*. El Perfil de Aplicación de Descubrimiento de Servicio es dependiente del Perfil Genérico de Acceso.

[4] *Streaming:* este término se refiere a ver u oír un archivo directamente sin necedad de descargarlo antes al computador.

- **Perfil de Telefonía Inalámbrica**

  Define cómo un teléfono móvil puede ser usado para acceder a un servicio de telefonía de red fija a través de una estación base. El perfil incluye llamadas a través de una estación base, haciendo llamadas de intercomunicación directa entre dos terminales y accediendo adicionalmente a redes externas.

- **Perfil de Manos Libres**

  Este perfil precisa los requerimientos, para que los dispositivos *Bluetooth,* soporten el uso de manos libres. En este caso el dispositivo puede ser usado como una unidad de audio inalámbrico de entrada/salida.

- **Perfil *Dial-up Networking***

  Este perfil define los protocolos y procedimientos que deben ser usados por dispositivos que implementen el uso del modelo llamado Puente *Internet.* Este perfil es aplicado cuando un teléfono celular es usado como un *modem* inalámbrico.

- **Perfil de *Fax***

  Precisa los protocolos y procedimientos que deben ser usados por dispositivos que implementen el uso de *fax*. El *software* de *fax* opera directamente sobre *RFCOMM*.

- **Perfil de Intercomunicador**

  Permite a dos teléfonos móviles establecer enlaces de conversación directa. El enlace directo es establecido usando señalización de telefonía sobre *Bluetooth.*

- **Perfil *Object Push***

  Este perfil define protocolos y procedimientos usados en el modelo *object push.* En el modelo *object push* hay procedimientos para introducir en el *inbox*, sacar e intercambiar objetos con otro dispositivo *Bluetooth.*

- **Perfil de Sincronización**

  Define protocolos y procedimientos usados en el modelo de sincronización. Éste usa el *GOEP*, el modelo soporta intercambios de información, por ejemplo para sincronizar calendarios de diferentes dispositivos.

1.1.4 SEGURIDAD EN BLUETOOTH [17] [34]

Para asegurar la protección de la información se ha definido un nivel básico de encripción, incluido en el diseño del *chip* de radio para proveer seguridad en equipos que carezcan de capacidad de procesamiento, las principales medidas de seguridad son:

- Una rutina de pregunta-respuesta, para autentificación
- Una corriente cifrada de datos, para encripción
- Generación de una clave de sesión (que puede ser cambiada durante la conexión)

Tres entidades son utilizadas en los algoritmos de seguridad: la dirección de la unidad *Bluetooth*, que es una entidad pública; una clave de usuario privada, como una entidad secreta; y un número aleatorio, que es diferente en cada nueva transacción.

La dirección *Bluetooth* se puede obtener a través de un procedimiento de consulta. La clave privada se deriva durante la inicialización y no es revelada posteriormente. El número aleatorio se genera en un proceso pseudo-aleatorio en cada unidad *Bluetooth*.

La especificación *Bluetooth* detalla tres modos de seguridad bajo los que el protocolo puede operar.

- **Modo1: Sin seguridad.** Todos los mecanismos de seguridad (autenticación y cifrado) están deshabilitados. Además el dispositivo se sitúa en un modo en el cual, permite que todos los dispositivos *Bluetooth* se conecten a él.

- **Modo2: Seguridad a Nivel de Servicio**. Este modo permite un acceso más flexible. Este modo es el más apropiado para ejecutar varias aplicaciones en paralelo con diferentes requerimientos de seguridad. Este modo de seguridad es impuesto después del establecimiento del canal.

- **Modo3: Seguridad a Nivel de Enlace.** El dispositivo instala procedimientos de seguridad antes del establecimiento del canal.

Los dispositivos *Bluetooth* solo pueden estar en un solo modo de seguridad en un momento determinado. Un dispositivo que opere en modo 3 no podrá autenticarse ante otros dispositivos de forma selectiva, sino que tratará de autenticarse ante todos los dispositivos que intenten comunicarse con él.

1.1.4.1 Seguridad con el Emparejamiento de dispositivos

Al inicio de una comunicación entre dispositivos *Bluetooth* la comunicación entre éstos está protegida, motivo por el que todos los dispositivos pueden comunicarse. Un nodo *Bluetooth* puede solicitar autenticación para realizar un determinado servicio.

Para la autenticación es necesario un código *PIN*. El código *PIN* tiene una longitud de hasta 16 caracteres *ASCII*. Para que los dispositivos se comuniquen se debe ingresar en ambos lados el mismo código *PIN*. Una vez que el usuario ha introducido el *PIN* adecuado ambos dispositivos generan una clave de enlace. Una vez generada, la clave se puede almacenar en el propio dispositivo o en un dispositivo de almacenamiento externo. La siguiente vez que se comuniquen

ambos nodos se utilizará la misma clave.

El procedimiento descrito hasta este punto se denomina emparejamiento. Es importante recordar que si la clave de enlace se pierde en alguno de los dispositivos involucrados se debe volver a ejecutar el procedimiento de emparejamiento.

No existe ninguna limitación en los códigos *PIN* a excepción de su longitud.

1.1.4.2 Autenticación *Bluetooth*

Los dispositivos *Bluetooth* se autentican siguiendo el esquema "*challenge-response",* desafío-respuesta. A continuación en la figura 1.20 se puede observar el procedimiento de autenticación.

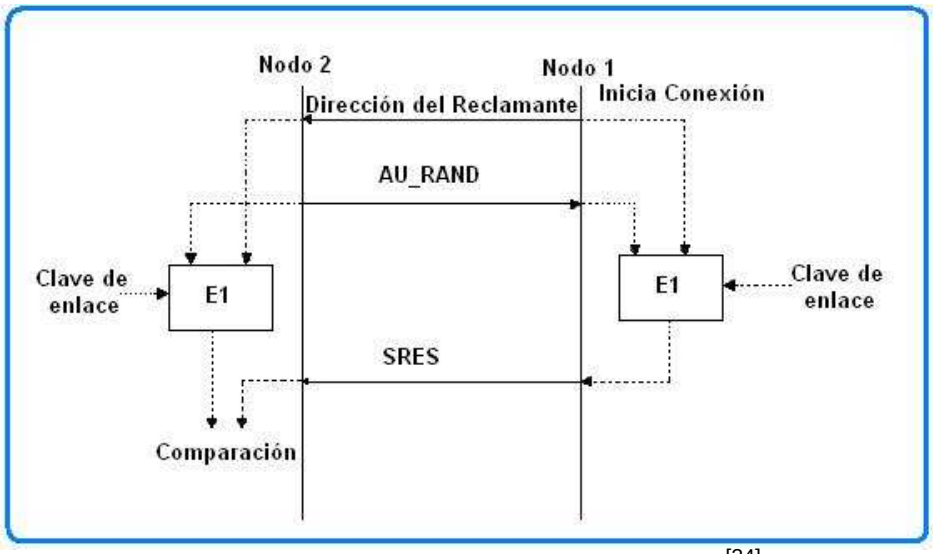


Figura 1.20 Proceso de autenticación *Bluetooth* [34]

Pasos para la autenticación *Bluetooth*:

- El nodo 1 inicia la conexión transfiere su dirección de 48 bits (*BD_ADDR*) al nodo 2, esta dirección es única e identifica al dispositivo.

- En respuesta el otro nodo 2 le transfiere un "desafío" aleatorio de 128 bits (*AU_RAND*) al nodo 1 que inició la conexión.

- El nodo 2 usa el algoritmo E1 para generar el “response” de autenticación, usando como parámetros la dirección del que inició la conexión, *BD_ADDR*, la clave de enlace, *Kab*, y el desafío. El nodo 1 realiza la misma operación.

- El nodo 1 que inició la conexión le devuelve el “*response*”, *SRES*, al otro nodo.

- El nodo 2 compara el *SRES* del demandante con el que él ha calculado.

- Si los valores de los 32 bits de los *SRES* son idénticos, el verificador establece la conexión.

1.1.5 APLICACIONES DE *BLUETOOTH*

En la actualidad se encuentra una gran cantidad de dispositivos *Bluetooth* que ofrecen aplicaciones muy variadas, permitiendo cambiar radicalmente la forma como los usuarios interactúan con los dispositivos que se encuentran relativamente cerca. Dentro de las aplicaciones se mencionan las siguientes:

- **Transferencia de archivos:** permite la transferencia de archivos sean estos: documentos en *Word*, imágenes, presentaciones en *Power Point*, etc. Además ofrece la posibilidad de ver el contenido de las carpetas existentes en otros dispositivos.

- **Conexión a Internet:** permite tener acceso inalámbrico a Internet mediante un teléfono móvil el cual actúa como si fuera una línea telefónica fija.

- **Escritorio Inalámbrico:** permite remplazar todos los dispositivos que utilizan cables permitiendo una comunicación vía radio. Utilizando desde un teclado inalámbrico hasta incluso utilizar un disco duro portátil que emplee esta tecnología para comunicarse.

- **Acceso inalámbrico a *LAN*:** a través de esta aplicación un grupo de dispositivos *Bluetooth* podrían conectarse a la red de Área Local a través de los llamados *LAP* (Puntos de Acceso *LAN*) y compartir los recursos

- **Sincronización Automática:** este servicio permite sincronizar automáticamente y de manera continua la Información de Administración Personal (*PIM*) con otros dispositivos *Bluetooth*; la información que se actualiza es la concerniente a calendario, lista de direcciones y teléfonos, mensajes y notas.

- **Dispositivo Manos Libres Inalámbrico***:* el dispositivo manos libres puede conectarse de manera inalámbrica al teléfono móvil, al ordenador portátil u otro móvil, con el fin de actuar como un dispositivo remoto con entrada y salida de audio.

1.2 IEEE 802.11

Los principales objetivos que se pretende conseguir con esta norma son la movilidad y la flexibilidad, los cuales vienen a ser fuertes argumentos a favor de la implementación de una Red Local Inalámbrica, *Wireless Fidelilty (Wi-Fi)*, en cualquier ambiente:

- Movilidad
- Simplicidad y rapidez en la instalación
- Flexibilidad en la instalación
- Costo de propiedad reducido
- Escalabilidad

1.2.1 EVOLUCIÓN [2] [27]

Las redes de área local inalámbricas funcionan desde hace más de quince años en entornos industriales y de investigación. Fueron implementadas por primera vez en el año 1979.

En marzo de 1985 la Comisión Federal de Comunicaciones, asignó a los sistemas *WLAN* las bandas de frecuencia 902-928 *MHz,* 2.4-2.48 *GHz* y 5.72-5.85 *GHz* también conocidas como *ISM.*

Desde 1985 hasta 1990 se siguió trabajando más en la fase de desarrollo hasta que en mayo de 1991 se publicaron varios trabajos referentes a *WLAN* operativas que superaban la velocidad de 1 *Mbps*, el mínimo establecido por el *IEEE 802* para que la red sea considerada realmente una *LAN*, con aplicación empresarial.

En junio del año 1997 el *IEEE* (Instituto de Ingenieros Electrónicos y Eléctricos) ratificó el estándar para *WLAN IEEE 802.11*, con velocidades de 1 y 2 *Mbps*, modulación de señal de espectro expandido por secuencia directa (*DSSS*), aunque también contempla la opción de espectro expandido por salto de frecuencia (*FHSS)* en la banda de 2,4 *GHz*, y se definió el funcionamiento y la interoperabilidad entre redes inalámbricas.

En el año 1999, se aprobó el estándar *802.11b*, una extensión del *IEEE 802.11* para *WLAN* empresariales, con una velocidad de 11 *Mbps* (otras velocidades normalizadas a nivel físico son: 5.5, 2 y 1 *Mbps*) y un alcance de 100 metros, que al igual que *Bluetooth* y *Home RF*, también emplea la banda de *ISM* de 2,4 GHz, pero en lugar de una simple modulación de radio digital y salto de frecuencia (*FH/* salto de frecuencia), utiliza técnicas de espectro expandido por secuencia directa (*DSSS*).

En julio de 1999 la *IEEE* publicó el suplemento del estándar en *802.11a*, que con modulación *OFDM* (*Orthogonal Frequency Division Multiplexing,* Modulación Ortogonal en División de Frecuencia) alcanza una velocidad de hasta 54 *Mbps* en la banda de 5 *GHz,* un alcance limitado a 30 metros dependiendo de la potencia de transmisión de los dispositivos utilizados.

“La banda de 5 *GHz* que utiliza se denomina *UNII* (Infraestructura de Información Nacional sin Licencia), que en los Estados Unidos está regulada por la *FCC*, el cual ha asignado un total de 300 *MHz*, cuatro veces más de lo que tiene la banda

ISM, para uso sin licencia, en tres bloques de 100 *MHz,* siendo en el primero la potencia máxima de 50 *mW,* en el segundo de 250 *mW*, y en el tercero puede llegar hasta 1 *W*, por lo que se reserva para aplicaciones en el exterior.” [2]

En el año 2003, se aprobó el estándar *IEEE 802.11g*, compatible con el *IEEE 802.11b*, capaz de alcanzar una velocidad de 54 *Mbps*, para competir con los otros estándares que prometen velocidades mucho más elevadas pero que son incompatibles con los equipos *802.11b* ya instalados, aunque pueden coexistir en el mismo entorno debido a que las bandas de frecuencias que emplean son distintas.

Estándar	Velocidad máxima	Interface de aire	Ancho de banda de canal	Frecuencia	Disponibilidad
802.11b	11 Mbps	DSSS	22 MHz	2.4 GHz	Ahora
802.11a	54 Mbps	OFDM	20 MHz	5.0 GHz	Ahora
802.11g	54 Mbps	OFDM/DSSS	22 MHz	2.4 GHz	Ahora

Tabla 1.5 Estandarización de *IEEE 802.11* [27]

1.2.2 ARQUITECTURA *Wi-Fi* [7]

La arquitectura básica *Wi-Fi* está definida por el estándar original *802.11.* Las especificaciones del estándar *802.11b* afectan únicamente a la capa física, añadiendo velocidades mayores y una conectividad más robusta.

El estándar *802.11* se centra en los dos niveles inferiores del modelo *OSI*, el físico y el de enlace de datos como se muestra en la figura 1.21

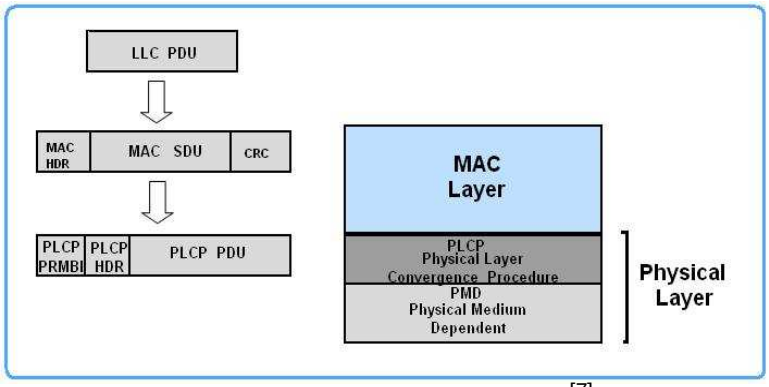


Figura 1.21 Arquitectura *Wi-Fi* [7]

Las tres capas físicas originalmente definidas en el *802.11* incluyen dos espectros de radio y una especificación de infrarrojos. Los estándares basados en radio operan dentro de la banda 2.4 *GHz*. Estas bandas de frecuencia son reconocidas por los reguladores internacionales, como *FCC* (USA), *ETSI* (Europa), y la *MKK* (Japón), como operaciones de radio sin licencia, para usos científicos, militares e industriales.

1.2.2.1 Capa Física [7]

En la Capa Física se define la modulación, señalización, características de la transmisión de datos, dos posibles topologías, tres tipos de medios inalámbricos que funcionan a cuatro posibles velocidades, potencia y banda de frecuencia.

1.2.2.1.1 Topologías que utiliza IEEE 802.11

La *IEEE 802.11* define la topología *Ad-hoc* y la topología Infraestructura.

- **Topología *Ad-Hoc***

Esta topología se caracteriza porque no existe punto de acceso (*AP*), las estaciones se comunican directamente entre si (punto a punto), de esta manera el área de cobertura está limitada por el alcance de cada estación individual.

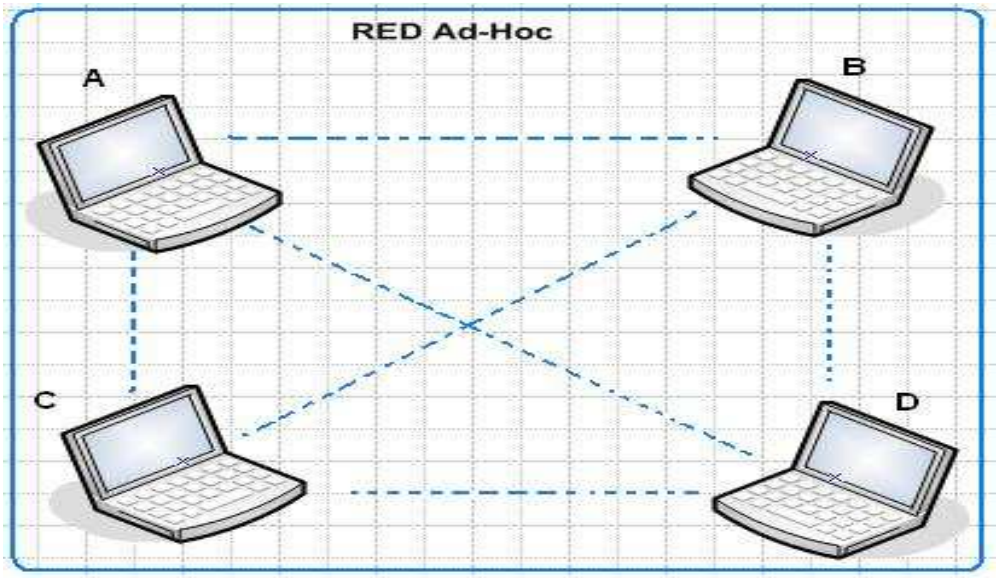


Figura 1.22 Red Ad-Hoc

- **Topología Infraestructura**

Esta topología se caracteriza por tener un punto de acceso (AP), el mismo que controla la comunicación entre estaciones, de esta manera el área de cobertura está limitada por el alcance del AP.

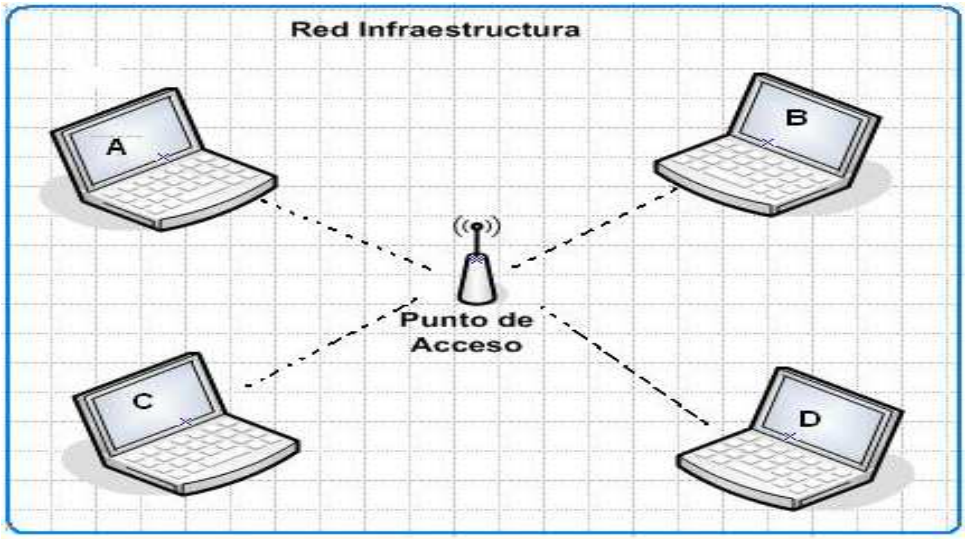


Figura 1.23 Red Infraestructura

a. Descripción general de componentes de las topologías [2]

- **Conjunto de servicios básicos (BSS):** es el bloque básico de construcción de una *LAN 802.11*. En el caso de tratarse de 2 estaciones se denomina *IBSS* (Independiente *BSS*), es lo que a menudo se denomina "red *Ad Hoc*".

- **Sistema de distribución (DS):** es la arquitectura que se utiliza para interconectar distintos *BSS*. El *AP* es el encargado de proveer acceso al *DS*, todos los datos que se mueven entre *BSS* y *DS* se hacen a través de estos *AP*.

- **Conjunto de servicios extendidos (ESS):** tanto *BSS* como *DS* permiten crear redes inalámbricas de tamaño arbitrario, este tipo de redes se denominan redes *ESS*.

La integración entre una red *802.11* y una no *802.11* se realiza

mediante un Portal. Es posible que un mismo dispositivo cumpla las funciones de *AP* y Portal.

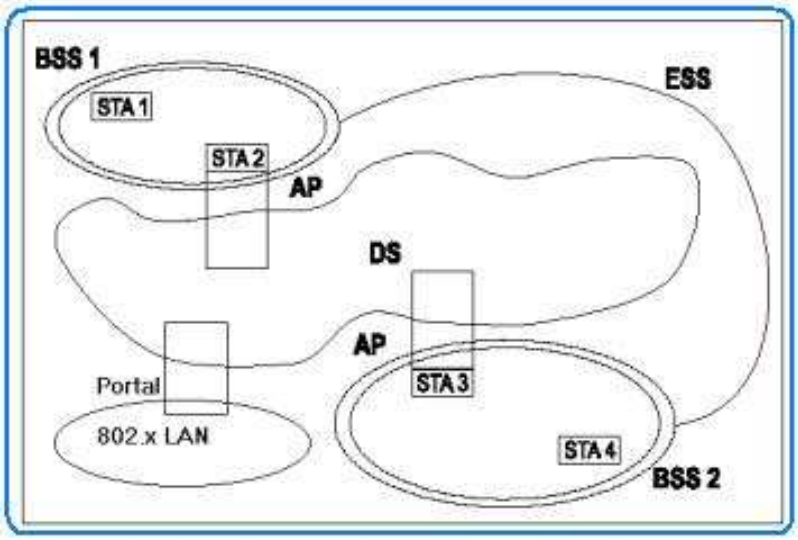


Figura 1.24 Componentes de la Arquitectura [2]

b. Servicios del Sistema de Distribución [2]

Tiene que ver con la administración de los miembros dentro de una celda y con la movilidad de las estaciones conforme entran y salen de la misma.

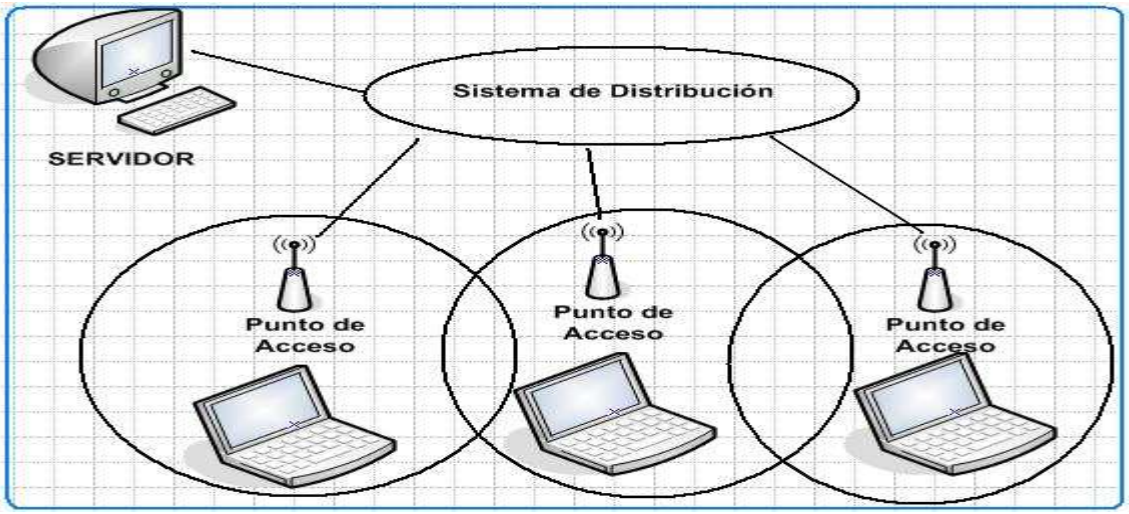


Figura 1.25 Direccionamiento en Modo Infraestructura

- **Distribución:** se encarga de llevar un paquete del punto de acceso de origen al de destino.

- **Integración:** se encarga de acoplar un sistema *IEEE 802.11* con otros sistemas *IEEE 802.x.*

  En concreto, define el componente portal que se encargará de aspectos necesarios como redireccionamiento.

- **Asociación:** servicio necesario para que una estación pueda adherirse al modo infraestructura y utilizar sus servicios.

- **Reasociación:** consiste en el campo de punto de acceso al que se asocia la estación para adherirse al modo infraestructura. También se utiliza para modificar las características de la asociación.

- **Autenticación y Desautenticación:** proceso necesario para que la estación se pueda conectar a la W*LAN* y consiste en la identificación de la estación.

- **Privacidad:** este servicio utilizará *WEP* para el encriptado de los datos en el medio.

- **Reparto de MSDUs entre STAs:** este es el servicio básico de intercambio.

c. Servicios de estación en el Sistema de Distribución [2]

Se relacionan con la actividad dentro de una sola celda:

- **Autenticación:** proceso necesario para que la estación pueda conectarse a la red *WLAN.*

- **Desautenticación:** cuando una estación previamente autenticada quiere abandonar la red.

- **Entrega de datos:** se realiza la entrega de datos entre las diferentes estaciones de trabajo.

1.2.2.1.2 Banda de Frecuencia utilizada por IEEE 802.11 [2]

Wi-Fi opera en la banda *ISM*, la cual es una banda de frecuencia abierta a cualquier sistema de radio independientemente del lugar del planeta donde se encuentre implementada. Sólo la banda *ISM* de 2,45 *GHz* cumple con este requisito, con rangos que van de los 2.4 *GHz* a los 2.5 *GHz*, y solo con algunas restricciones en países como Francia, España y Japón que se muestran en la tabla 1.6.

Localización Geográfica	**Rango Regulatorio**	**Canales RF**
USA, Europa	2.400 – 2.4835 GHz	F = 2402+ K*MHz , K = 0,....,76
España	2.445 – 2.475 GHz	F = 2449+ K*MHz , K = 0,....,22
Francia	2.4465 – 2.4835 GHz	F = 2454+ K*MHz , K = 0,....,22

Tabla 1.6 Banda de Frecuencia *Wi-Fi* [2]

1.2.2.1.3 Modulación [7]

El *IEEE 802.11* define tres tipos de modulación en la capa física para la transmisión y recepción de tramas.

- Espectro expandido por secuencia directa *DSSS*
- Espectro expandido por salto de frecuencias *FHSS*
- Luz Infrarroja en banda base sin modular

La explicación de cada modulación se encuentra en el *ANEXO H*.

1.2.2.1.4 Potencia [2]

Las antenas operan con un determinado nivel de potencia entregado por el transmisor. En el caso de *IEEE 802.11* se ajustan normalmente a 100 *mW* que es la potencia máxima permitida en Europa para la emisión de puntos de acceso o *NIC Wi-Fi*.

La limitación de potencia impuesta por las distintas autoridades influye evidentemente en la cobertura inalámbrica. A continuación se muestra los niveles de potencia permitidos en cada una de las regiones para la banda de 2.4 *Ghz*.

El nivel de potencia máximo permitido en este rango de frecuencias varía de un país a otro según sus normas regulatorias.

- Estados Unidos la *FCC* (*Federal Communication Commission*) a través de la norma Part. 15.247 limita la radiación de antena a 1*W* de potencia.
- En Japón el *MPT ordinance* 79, fija el nivel de potencia a 10 *mW* por 1 *MHz*
- En Europa el *ETS300-328 ETS96* (*European Telecommunications Standards Institute*) es el encargado de regular los límites de potencia de emisión, ésta limita la potencia hasta 100 *mW*.

Máxima potencia de salida	Localización Geográfica	Documento de Complacencia
1000 mW	EE.UU.	FCC 15.247
100 mW	EUROPA	ETS 300-328
10 mW/MHz	JAPÓN	MPT ordinance 79

Tabla 1.7 Niveles de Potencia de Transmisión para diferentes Regiones [2]

1.2.2.2 Capa Enlace de Datos [7]

La principal función de esta capa es el control de acceso al medio, realizando funciones de fragmentación, encriptación, gestión de alimentación eléctrica, sincronización y soporte de *roaming* entre múltiples *APs.*

La capa enlace de datos se divide en dos subcapas: Control de Enlace Lógico (*LLC*) y Control de Acceso al Medio (*MAC*). El estándar *802.11* utiliza el mismo *LLC* que el *802.2*, pero el nivel *MAC* es diferente.

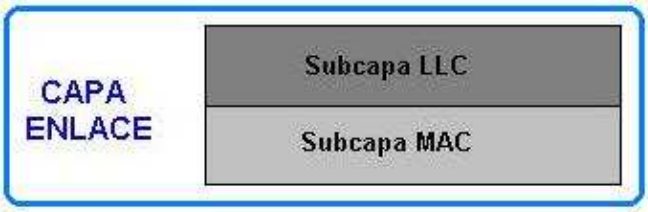


Figura 1.26 Capa Enlace de Datos [7]

1.2.2.2.1 Estructura de trama de la Capa de Enlace de Datos [2]

El estándar *802.11* define 3 clases diferentes de tramas: trama de datos, de control y de administración. Cada una de ellas tiene un encabezado con campos utilizados dentro de la subcapa *MAC* y algunos usados por la capa física.

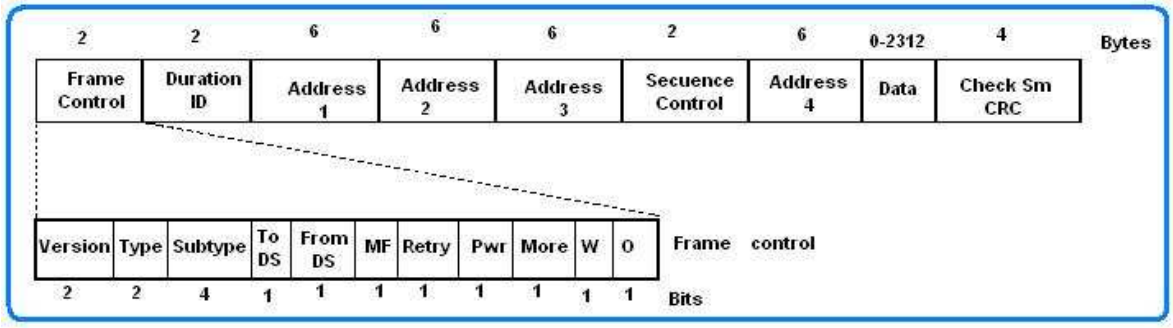


Figura 1.27 Trama Capa Enlace de Datos [2]

- **Frame Control:** tiene 11 subcampos:
  - ✓ **Versión del protocolo**: indica el tipo de protocolo de control con el que se trabaja
  - ✓ **Tipo:** tipo de datos, de Control o de Administración: ejemplo. *RTS* o *CTS.*
  - ✓ **To DS:** indica que la trama va al Sistema de Distribución.
  - ✓ **From DS:** indica que la trama viene del Sistema de Distribución.
  - ✓ **MF:** indica que siguen más fragmentos.
  - ✓ **Retransmisión:** indica que ésta es la retransmisión de una trama.

- ✓ **Pwr:** usado por la estación base para poner a una estación en estado de escucha o sacarla de este estado.
- ✓ **More:** indica que el emisor tiene tramas adicionales para el receptor.
- ✓ **W:** especifica que el cuerpo de la trama se ha empleado usando el algoritmo *WEP.*
- ✓ **O:** una secuencia de tramas que tenga este bit encendido debe procesarse en orden.

- **Duración:** indica cuanto tiempo ocuparán el canal, la trama y su *ACK*.
- **Direcciones:** 2 son para origen y destino y las otras 2 se usan para las estaciones base origen y destino, en el caso de tráfico entre celdas.
- **Secuencia:** permite numerar los fragmentos, 12 bits identifican la trama y 4 al fragmento.
- **Datos:** contiene la carga útil, hasta 2312 *bytes.*
- ***Checksum:*** contiene la suma de verificación.

1.2.2.2.2 Control de Acceso al Medio (MAC) [2] [7]

La principal función de esta capa es el control de acceso al medio, realizando funciones de fragmentación, encripción y sincronización.

- Acceso al canal
- Direccionamiento de las *PDU*
- Formato de las tramas
- Comprobación de errores
- Fragmentación y ensamblado de las *MSDU*
- Autenticación y privacidad para permitir servicios seguros
- Servicios de gestión *MAC* para permitir *Roaming* dentro de un *ESS* y para control de potencia de estaciones.

El estándar *IEEE* define dos modos de operación posibles:

- *DCF* (Función de Coordinación Distribuida)
- *PCF* (Función de Coordinación Puntual)

b. PCF Función de Coordinación Puntual [8]

Permite proporcionar diferenciación de servicios para soportar aplicaciones en tiempo real, pero resulta bastante ineficiente y compleja de implementar, *PCF* se puede implementar mediante un *AP* (Punto de Acceso), para permitir transmisión orientada a conexión de tramas *MAC* dentro de un intervalo de tiempo máximo.

PCF puede alternar entre:

- *CP* (*Contention Period*)
- *CFP* (*Contention - Free Period*): la utilización del medio está controlada por el *AP*, con el que se elimina la necesidad que las estaciones luchen por conseguir acceso al canal.

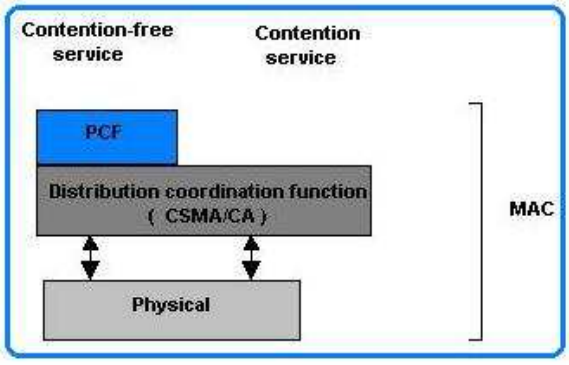


Figura 1.28 Estructura MAC [7]

c. DCF Función de Coordinación Distribuida [8]

Permite la transmisión de datos empleando el método del mejor esfuerzo, este método de contención se basa *CSMA/CA*. En redes *Ad-hoc* se hace uso exclusivo de *DCF*.

b.1 Modo de contención CSMA/CA (*Carrier Sense Multiple Access, Collision Avoidance*). [8]

Esta clase de métodos de acceso, denominados protocolos de acceso por contienda, son muy efectivos si la carga de uso del medio no es muy alta, ya que esto permitirá a las estaciones transmitir con un retardo mínimo. Hay que tener en cuenta además que pueden producirse colisiones debido a la posibilidad, que 2 estaciones “escuchen” el medio simultáneamente, detectando que esté libre e iniciando su transmisión al mismo tiempo.

El método que más se utiliza en redes inalámbricas es el *CSMA/CA*. Este protocolo evita colisiones en lugar de descubrir una colisión, como el algoritmo usado en la *IEEE 802.3*. En una red inalámbrica es difícil descubrir colisiones. Es por ello que se utiliza el *CSMA/CA* y no el *CSMA/CD* debido a que entre el final y el principio de una transmisión suelen provocarse colisiones en el medio. En *CSMA/CA*, cuando una estación identifica el fin de una transmisión espera un tiempo aleatorio antes de transmitir su información, disminuyendo así la posibilidad de colisiones.

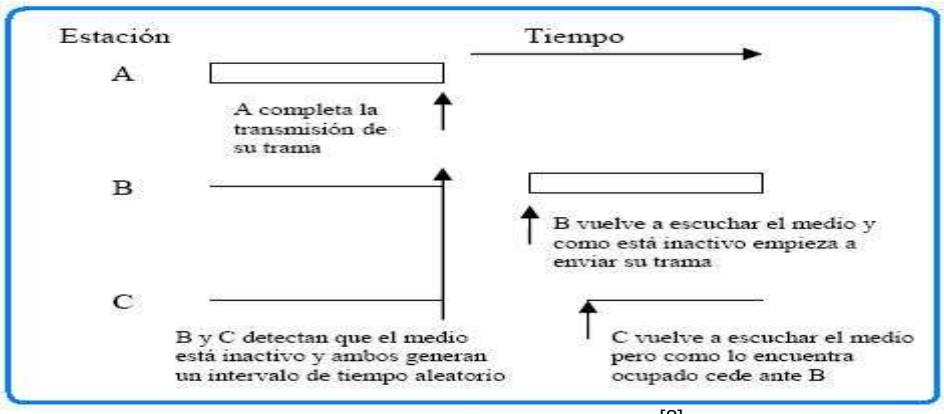


Figura 1.29 Método CSMA/CA [8]

b.1.1 Funcionamiento de CSMA/CA [8] [9]

Antes de transmitir información, la estación debe sensar el medio, o canal inalámbrico, para determinar su estado (libre/ocupado). Si el medio está desocupado la estación ejecuta una espera adicional llamada espaciado entre

tramas (*IFS*). Si durante este intervalo temporal, el medio continúa ocupado, la estación debe esperar hasta que finalice la transmisión antes de realizar cualquier otra acción.

Una vez finalizada esta espera, la estación ejecuta el algoritmo de *Backoff*, para determinar una espera adicional y aleatoria escogida uniformemente en un intervalo llamado ventana de contienda (*CW*). El algoritmo de *Backoff* da un número aleatorio y entero de ranuras temporales y su función es la de reducir la probabilidad de colisión, que es máxima cuando varias estaciones están esperando a que el medio quede libre para transmitir.

Mientras se ejecuta el algoritmo de *Backoff*, la estación continúa escuchando el medio hasta que el medio este libre. Si el medio continúa ocupado durante un tiempo igual o superior a *IFS*, el algoritmo de *Backoff* queda suspendido hasta que se cumpla esta condición. Cada retransmisión provocará que el valor de *CW*, que se encontrará entre *CWmin* y *CWmax* se duplique hasta llegar al valor máximo, el valor de las ranuras temporales para *IEEE 802.11b* es 20 *µseg.*

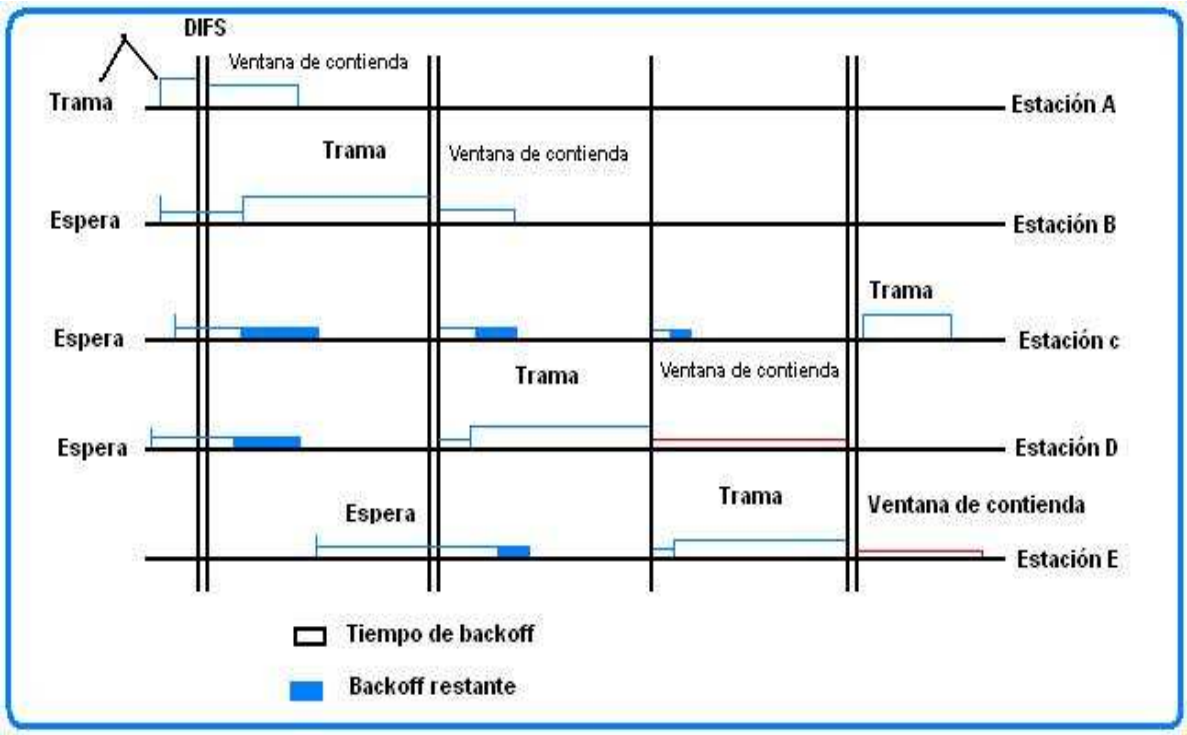


Figura 1.30 Acceso CSMA/CA [8]

Sin embargo, *CSMA/CA* en un entorno inalámbrico presenta una serie de problemas que se intentan resolver con alguna modificación. Los dos principales problemas que se pueden detectar son:

- "Nodos ocultos: una estación cree que el canal está libre, pero en realidad está ocupado por otro nodo que no oye." [9]

- "Nodos expuestos: una estación cree que el canal está ocupado, pero en realidad está libre pues el nodo al que oye no le interferiría para transmitir a otro destino, la solución que propone *802.11* es *MACA (Multiple Acces with Collision Avoidance)"*. [9]

La figura 1.31 representa u ejemplo de nodo escondido.

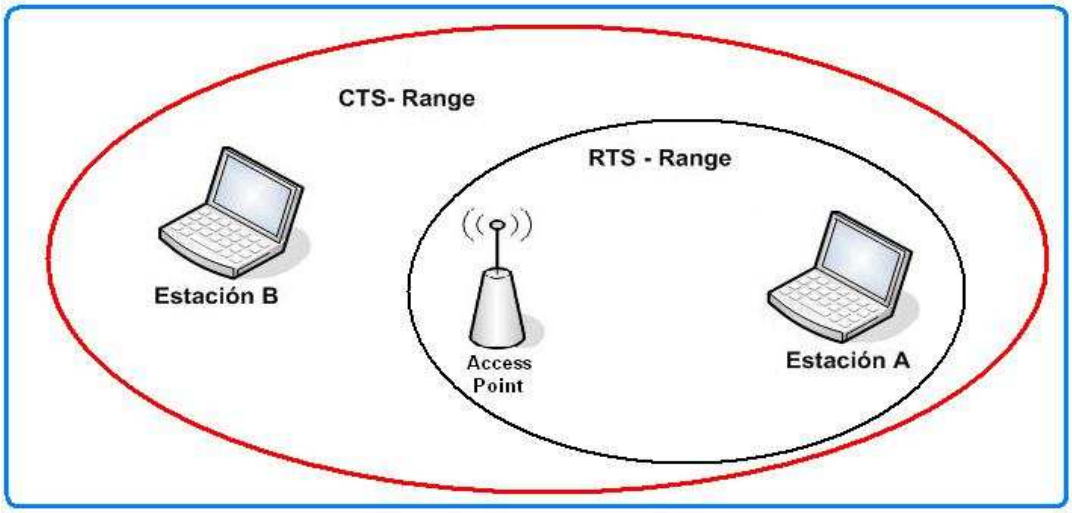


Figura 1.31 Ejemplo de nodo escondido

Un dispositivo inalámbrico puede transmitir con la potencia suficiente para que sea escuchado por un nodo receptor, pero no por otra estación que también desea transmitir y que por tanto no detecta la transmisión.

Para resolver este problema, la norma *IEEE 802.11* ha añadido al protocolo de acceso *CSMA/CA* un mecanismo de intercambio de mensajes con reconocimiento positivo, al que denomina *Reservation - Based Protocol*.

b.2 Modo de contención CSMA/CA con RTS/CTS [8]

Cuando una estación está lista para transmitir, primero envía una solicitud (*RTS*) al punto de acceso quien difunde el *NAV* (*Network Allocation Vector*) a todos los demás nodos para que queden informados de que se va a transmitir y cuál va a ser la duración de la transmisión. Si no encuentra problemas, responde con una autorización *(CTS)* que permite al solicitante enviar su trama (datos). Si no se recibe la trama *CTS,* se supone que ocurrió una colisión y los procesos *RTS* empiezan de nuevo.

Después de que se reciba la trama de los datos, se devuelve una trama de reconocimiento (*ACK*) notificando al transmisor que se ha recibido correctamente la información.

Aun así permanece el problema de que las tramas *RTS* sean enviadas por varias estaciones a la vez, sin embargo estas colisiones son menos dañinas ya que el tiempo de duración de estas tramas es relativamente corto.

Este protocolo también puede utilizarse si no existen dispositivos auxiliares en las redes *Ad-hoc*, en este caso no aparecería la trama *NAV*.

Se definen cuatro espaciados entre tramas (*IFS*) para dar prioridad de acceso al medio inalámbrico.

- **SIFS (*Short IFS*).** Es el tiempo de espera más corto, provoca transmisiones inmediatas. Usado para transmisiones de *ACKs, RTS* y *CTS*

- **DIFS (*DCF*).** Es el tiempo de espera habitual en las contiendas con mecanismo MACA.

La figura 1.32 representa el modo de contención CSMA/CA con RTS/CTS

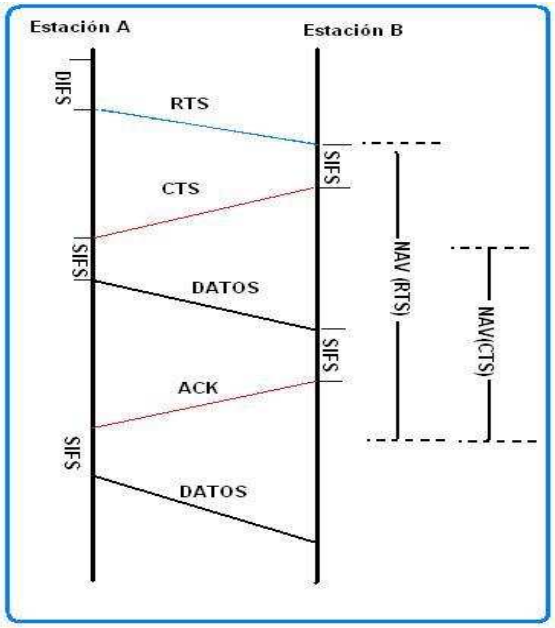


Figura 1.32 Modo de contención *CSMA/CA* con *RTS/CTS* [7]

b.3 Control de Acceso al Medio MACAW (*Multiple Acces with Collision Avoidance*) [8]

Con base en estudios de simulación de *MACA, Bharghavan* (1994) afinaron el *MACA* para mejorar su desempeño y llamaron *MACAW* a su nuevo protocolo. En *MACAW* las tramas perdidas no son retransmitidas en el nivel de enlace sino que se debe esperar a que la información llegue al nivel de transporte. Esto es resuelto obligando al receptor a enviar una trama *ACK* de datos correctamente enviados.

Además el algoritmo de *Backoff* se ejecuta para cada trama de datos fuente-destino para cada estación. Este cambio provoca la imparcialidad del protocolo. Finalmente, se añade un mecanismo para estaciones que intercambian información sobre la congestión y para hacer que el algoritmo de *backoff* no reaccione tan violentamente ante problemas temporales.

b.3.1. Funcionamiento de MACAW

MACAW funciona de manera similar a CSMA/CA, *MACAW* utiliza un intercambio de mensajes *RTS/CTS/DS* (*Data Send*) - DATA-ACK, además de implementar modificaciones al algoritmo de retransmisión o de *backoff.*

La utilización de un *ACK* en este nivel mejora los tiempos de respuesta, comparándolos con los que se obtendría si se dejara manejar la situación por el protocolo de nivel de transporte.

El nuevo *DS* permite distribuir la información de sincronización sobre los períodos de contienda, de forma que los nodos puedan “pelear" de igual forma por una ranura de tiempo para solicitar la transmisión.

La transmisión se lleva a cabo de la siguiente manera, el emisor envía un *RTS* al receptor, quien responderá con un *CTS*, una vez recibido el *CTS,* el emisor envía un *DS* seguido de los datos a transmitir. En caso de recibirse correctamente los datos el receptor devuelve un *ACK*, caso contrario no lo hace y se retransmite la información siguiendo el mismo esquema partiendo con el *RTS.*

En el caso de que el *ACK* se pierda, se enviará un nuevo *RTS* al cual se le responderá nuevamente con el mismo *ACK*.

1.2.3 Seguridad en Wi-Fi

1.2.3.1 WEP (*Wired Equivalent Protocol*) [25]

Es un sistema de encriptación estándar propuesto por el comité *802.11*, implementando a nivel de la capa *MAC* del modelo *OSI.* Dicho estándar comprime y cifra los datos que se envían a través de las ondas de radio.

WEP ofrece dos niveles de seguridad, encriptación a 64 o 128 bit. La encriptación usa un sistema de claves. La clave de la tarjeta de red del cliente debe coincidir con la clave del *AP*.

WEP utiliza una palabra clave que va a ser utilizada para autenticarse en redes cerradas y para cifrar los mensajes de la comunicación. Para generar la clave, en muchos *AP* se pide una frase y luego a partir de ella se generan 4 claves distintas para garantizar el máximo azar en la elección de la misma, pero en otros

simplemente se pide que se introduzca una clave con las restricciones de longitud que se configure y listo.

Para el cifrado de cada trama se añadirá una secuencia cambiante de bits, que se llama Vector de Inicialización *(IV),* para que no se utilice siempre la misma clave de cifrado y descifrado. Así, dos mensajes iguales no generarán el mismo resultado cifrado ya que la clave va cambiando. Cuando se tiene 4 claves, se debe marcar cual es la que se utiliza ya que *WEP* sólo utiliza 1 clave para todo. Si se ha seleccionado una opción de clave *WEP* de 64 bits, se tendrá 5 octetos (40 bits) son la clave y los 24 bits restantes serán el Vector de Inicialización *(IV)*. Es decir, en una comunicación normal se tendrá 2^{24} claves distintas de cifrado.

En el caso de *WEP* de 128 bits se tiene 13 octetos fijos (104 *bytes*) y 24 bits cambiantes, es decir, se tendrá el mismo número de claves pero de mayor longitud.

La figura 1.33 representa la creación de claves en seguridad con *WEP*

a. Creación de las claves en *WEP*

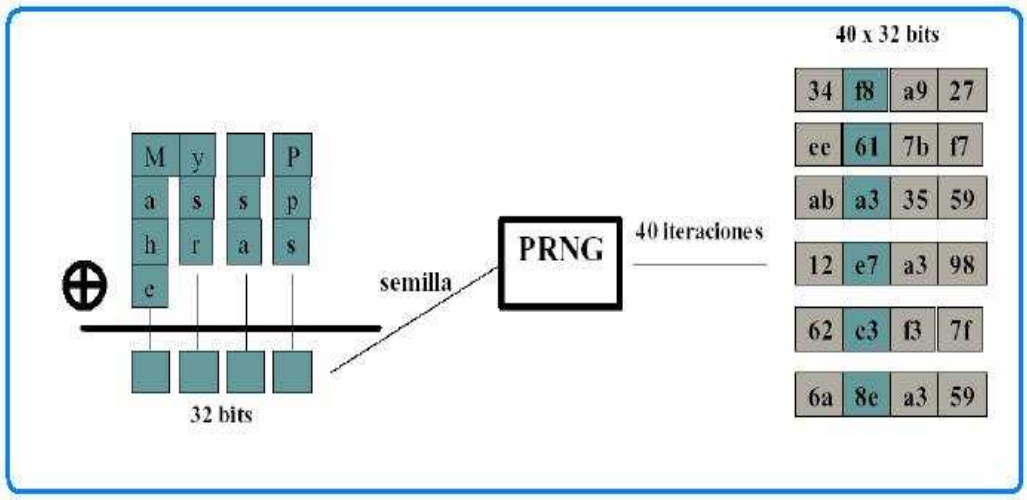


Figura 1.33 Creación de claves en WEP [25]

WEP utiliza el algoritmo *RC4* para la encriptación con claves de 64 bits, aunque existe también la posibilidad de utilizar claves de 128 bits. En realidad son 40 y 104 bits, ya que los otros 24 van en el paquete como Vector de Inicialización *(IV)*.

La clave de 40 o 104 bits, se genera a partir de una clave (*passphrase*) estática de forma automática, aunque existe *software* que permite introducir esta clave manualmente. La clave debe ser conocida por todos los clientes que quieran conectarse a la red inalámbrica que utiliza *WEP*, esto implica que muchas veces se utilice una clave fácil de recordar y que no se cambie de forma frecuente. A partir de la clave se generan 4 claves de 40 bits, sólo una de ellas se utilizará para la encriptación *WEP*.

Para generar las claves se hace una operación *XOR* con la cadena *ASCII* (*My Passphrase*) que queda transformada en una secuencia de 32 bits que utilizará el generador de números *pseudoaleatorios (PRNG)* para generar 40 cadenas de 32 bits cada una. Se toma un bit de cada una de las 40 cadenas generadas por el *PRNG* para construir una clave y se generan 4 claves de 40 bits.

1.2.3.2 WPA (*Wireless Application Protocol*) [25]

WPA emplea el cifrado de clave dinámico, lo que significa que la clave está cambiando constantemente y hacen que las incursiones en la red inalámbrica sean más difíciles que con *WEP*. *WPA* está considerado como uno de los más altos niveles de seguridad inalámbrica para su red, es el método recomendado si su dispositivo es compatible con este tipo de cifrado. Las claves se insertan como dígitos alfanuméricos, sin restricción de longitud, en la que se recomienda utilizar caracteres especiales, números, mayúsculas y minúsculas, y palabras difíciles de asociar entre ellas o con información personal.

Dentro de *WPA*, hay dos versiones de *WPA*, que utilizan distintos procesos de autenticación:

- **Para el uso personal doméstico:** el protocolo de integridad de claves temporales *(TKIP)* es un tipo de mecanismo empleado para crear el cifrado de clave dinámico y autenticación mutua. *TKIP* aporta las características de seguridad que corrige las limitaciones de *WEP*. Debido a que las claves están en constante cambio, ofrecen un alto nivel de seguridad para su red.

- **Para el uso empresarial de negocios:** el protocolo de autenticación extensible *(EAP)* se emplea para el intercambio de mensajes durante el proceso de autenticación. Emplea la tecnología de servidor *802.1x* para autenticar los usuarios a través de un servidor *RADIUS* (Servicio de usuario de marcado con autenticación remota). Esto aporta una seguridad de fuerza industrial para su red.

  Servidor *RADIUS* distribuye claves diferentes a cada usuario a través del protocolo *802.1x.*

  En la figura 1.34 se puede ver un ejemplo de autenticación *WPA*

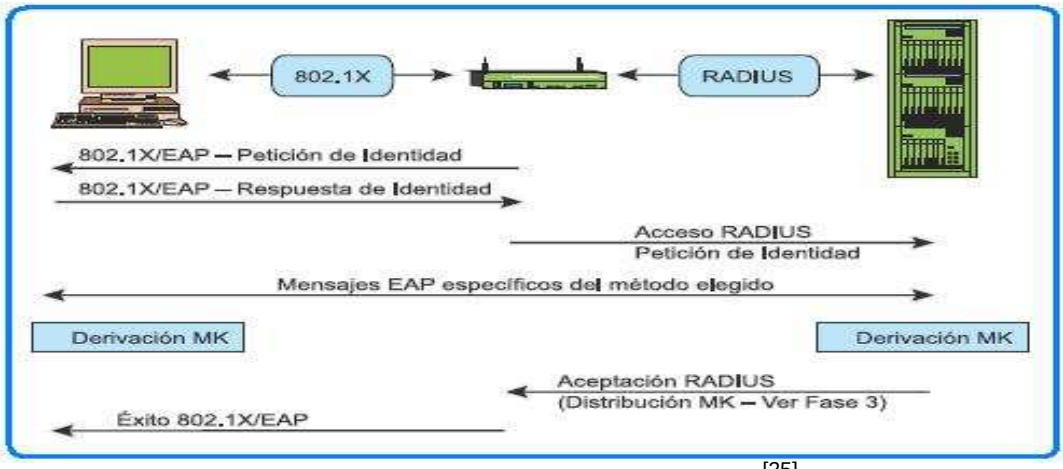


Figura 1.34 WPA (Protocolo de Autenticación) [25]

1. El cliente se asocia con el Punto de Acceso, que bloquea su tráfico
2. El cliente presenta credenciales que son autentificadas por *RADIUS*
3. El cliente autentifica al servidor *RADIUS* (*EAP-MD5* no válido)
4. Cliente y *RADIUS* derivan clave *WEP Unicast* (clave inicial *TKIP*)
5. Punto de acceso envía clave *WEP broadcast* cifrada con *WEP unicast*
6. Punto de acceso y cliente cifran sus comunicaciones

1.2.3.3 802.11i o WPA2 [25]

El *Task Group* de *IEEE 802.11i*, se conformó en el año 2001, con la intención de analizar una arquitectura de seguridad más robusta y escalable, debido a la inminente demanda del mercado en este tema y en julio de 2004 aprobó este

estándar. Por su parte la *Wi-Fi Alliance* lo lanzó al mercado en septiembre de ese año. En forma resumida, este nuevo estándar, propone a *802.1x* como protocolo de autenticación, pudiendo trabajar con su referencia *EAP*, éste último proporciona una gran flexibilidad en la metodología de autenticación.

Previo al estándar, *Cisco Systems* ofreció el primer tipo de autenticación que se denominó *LEAP* (*Lightweight EAP*), protocolo que inicialmente fue propietario de *Cisco*, pero en la actualidad lo emplean varios fabricantes.

Por su parte *Microsoft,* inicialmente junto con *Windows XP* lanzó al mercado su protocolo denominado *EAP/TLS* (*EAP with Transport Layer Security*), y fue aceptado por *IEEE*, se basa en certificados en lugar de contraseñas como credenciales de autenticación. Otros fabricantes han presentado *EAP/TTLS* (*EAP with Tunneling Transport Layer Security*), el cual realiza un túnel de nivel 2 entre el cliente y el *AP*, una vez establecido el túnel, *EAP/TTLS* opera sobre él, lo cual facilita el empleo de varios tipos de credenciales de autenticación que incluyen contraseñas y certificados, en realidad no deja de ser una variante de *EAP/TLS*.

La última variante es *PEAP* (*Protected Extensible Authentication Protocol*), inicialmente fue la versión “0” y ya está vigente la versión “1”, el cual aplica una metodología muy similar a *EAP/TTLS* en cuanto al empleo de túnel y sobre el una amplia variedad de credenciales de autenticación, este último ya está soportado por los más importantes fabricantes. En general, se considera que *PEAP* es el método más seguro del momento. Este protocolo fue desarrollado por *Microsoft, Cisco* y *RSA*.

802.1x: Este estándar no fue presentado para *Wi-Fi*, sino para el acceso seguro *PPP* (en tecnologías de cable). Una de las grandes características de *Wi-Fi* es emplear todas las herramientas que ya existen y pueden prestar utilidad al mismo. *802.1x* es uno de los mejores ejemplos de esto. La arquitectura *802.1x* está compuesta por tres partes:

- **Solicitante:** generalmente se trata del cliente *Wi-Fi*

- **Autenticador:** suele ser el *AP*, que actúa como traspaso de datos y como bloqueo hasta que se autoriza su acceso (importante esto último).

- **Servidor de autenticación:** suele ser un Servidor *RADIUS* o *Kerberos,* que intercambiará el nombre y credencial de cada usuario. El almacenamiento de las mismas puede ser local o remoto en otro servidor de *LDAP*, de base de datos o directorio activo.

Una de las grandes ventajas de emplear *802.1x* es que el servidor de autenticación, permite también generar claves de cifrado muy robustas.

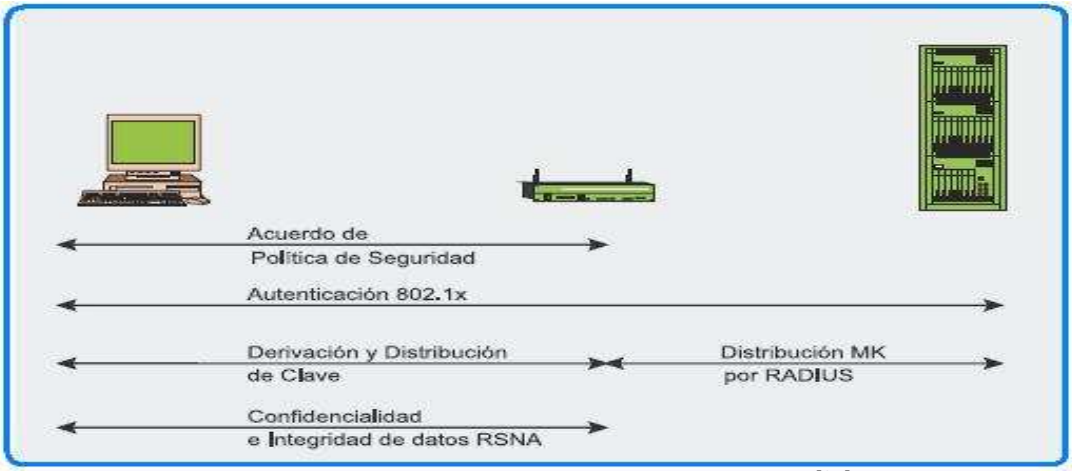


Figura 1.35 WPA2 (Protocolo de Autenticación) [25]

1.2.4 APLICACIONES

Hoy en día *Wi-Fi* amplia más sus aplicaciones gracias a que cada vez son más los dispositivos móviles que salen al mercado (ordenadores portátiles, agendas electrónicas, teléfonos). Las redes inalámbricas son la mejor solución para conectarlos a la red, aprovechando al máximo todas sus ventajas, sin perder la flexibilidad y movilidad que proporciona no necesitar cables.

- **Conexión a Internet:** La solución inalámbrica es idónea para la prestación de servicios de acceso a *Internet* o a *Intranets* en lugares donde no existe una infraestructura moderna de comunicaciones e incluso donde ya existe esta infraestructura para aportar más servicios y solucionar problemas antiguos.

Con la tecnología satélite con *Wi-fi* o *Wimax* con *Wi-Fi*, un punto de acceso es capaz de dar cobertura a varias decenas de ordenadores con un control total en el acceso como en el ancho de banda. La conexión a Internet es permanente y el usuario dispone de 11 *Mbps*.

- **Telefonía IP:** la integración de las comunicaciones de voz en las redes informáticas empresariales aporta enormes ventajas en cuanto a productividad y ahorro en comunicaciones. Soluciones de telefonía fija *IP* y de telefonía móvil *IP* basada en la norma *802.11b*. Las redes *Wi-Fi* en conjunto con la telefonía *IP* permiten que particulares o empresas de la misma población se comuniquen sin costo.

- **Redes privadas:** la ventaja que ofrece *Wi-Fi* de llegar a cualquier parte sin cables permite interconectar, a través de la red ya creada, las instalaciones o delegaciones de su empresa para la prestación de servicios de transmisión de datos a alta velocidad eliminando los costos mensuales por tráfico. La interconexión se puede realizar utilizando tantos enlaces punto a punto como sea necesario o utilizando nodos intermedios.

- **CCTV (Circuito Cerrado de Televisión):** las tecnologías inalámbricas ofrecen avanzados sistemas de seguridad. Con la llegada al mercado de las cámaras *IP* y la elevada capacidad de transmisión de la red *Wi-Fi* ha permitido desarrollar nuevos sistemas de video vigilancia remota a través de *Internet*.

  La posibilidad de obtener y enviar datos como video digital abre muchas posibilidades en el campo de la seguridad al permitir que unidades móviles dispongan a tiempo real de toda la información necesaria y de un completo control de las instalaciones.

- **Gestión de datos:** las soluciones móviles están aportando beneficios gracias a una mejor gestión de las empresas y mejora de la productividad. Además implementan nuevos servicios y reducen costos.

Las *PDAs* con conexión inalámbrica a *Internet* son utilizadas por empresas de logística para control de almacén, restaurantes, hoteles y cafeterías, para control de pedido.

1.3 FACTORES DE PROPAGACIÓN INALÁMBRICA

Son varios los factores a considerar a la hora de diseñar un sistema inalámbrico, algunos de los aspectos a tener en cuenta son los siguientes:

- Cobertura
- Rendimiento
- Interferencia

Entre los diferentes factores que afectan a la cobertura y rendimiento de un sistema inalámbrico se puede mencionar los siguientes:

1.3.1 ATENUACIÓN Y ABSORCIÓN DE ONDAS [10]

El espacio libre puede ser considerado como el vacío y no se consideran pérdidas. Cuando las ondas electromagnéticas se encuentran en el vacío, se llegan a dispersar y se reduce la intensidad de potencia a lo que es llamado atenuación. La atenuación se presenta tanto en el espacio libre como en la atmósfera terrestre. La atmósfera terrestre no se le considera vacío debido a que contiene partículas que pueden absorber la energía electromagnética y a este tipo de reducción de potencia se le llama pérdida por absorción la cual no se presenta cuando las ondas viajan afuera de la atmósfera terrestre

1.3.1.1 Atenuación [10]

La reducción de la intensidad de potencia con la distancia equivale a una pérdida de potencia y se suele llamar atenuación de la onda electromagnética. La atenuación de la onda se expresa en general en función del logaritmo de la relación de intensidades de potencia (pérdida en dB), la definición matemática de la atenuación es:

$$\gamma_a = 10\log\frac{P_1}{P_2}$$ **Ecuación 1.2** [10]

La reducción de la intensidad de potencia debida a la propagación en el espacio no libre se llama absorción.

1.3.1.2 Absorción [10]

Las ondas de radio que viajan por la atmósfera terrestre son atenuadas o debilitadas mediante la transferencia de energía a este medio. Entre los diferentes materiales que pueden absorber las ondas electromagnéticas se puede mencionar los siguientes: rocas, ladrillos, concreto, madera, árboles y otros materiales.

La absorción de onda por la atmósfera es análoga a una pérdida de potencia I^2R. Una vez absorbida la energía de onda ésta se pierde, y causa una atenuación en las intensidades de campo eléctrico, campo magnético, y una reducción en intensidad de potencia.

1.3.2 PÉRDIDAS EXISTENTES EN UN RADIO ENLACE

1.3.2.1 Pérdidas en la Trayectoria en el Espacio Libre (L_{patch}) [10]

Se define como la pérdida incurrida por una onda electromagnética al propagarse en línea recta a través del vacío, sin energías de absorción o reflexión debidas a objetos cercanos. Estas pérdidas dependen de la frecuencia, y aumentan con la distancia. La ecuación para determinar estas pérdidas es la siguiente:

$$L_{patch} = \left(\frac{4\pi \bullet R}{\lambda}\right)^2 = \left(\frac{4\pi \bullet f \bullet R}{c}\right)^2$$ **Ecuación 1.3** [10]

Donde:

L_{patch} = pérdidas en la trayectoria en espacio libre (adimensional)

R = distancia máxima entre dispositivos (metros)

f = frecuencia (hertz)

λ = longitud de onda (metros)

c = velocidad de la luz en el espacio libre ($3*10^8$ m/s)

Al pasar a dB se obtiene

$$L_{p(dB)} = 10 * \log\left(\frac{4\pi \bullet f \bullet R}{c}\right)^2$$

$$L_{p(dB)} = 20 * \log\left(\frac{4\pi \bullet f \bullet R}{c}\right)$$

$$L_{p(dB)} = 20 * \log\left(\frac{4\pi}{c}\right) + 20 * \log f + 20 * \log R \quad \textbf{Ecuación 1.4} \ [10]$$

1.3.2.2 Desvanecimiento por Múltiple Trayectoria (L_{fade}) [10]

En esencia el desvanecimiento por multitrayectoria es un factor ficticio que se incluye en la ecuación del cálculo de potencia de recepción (ecuación 1.6), para tener en cuenta las características no ideales y menos predecibles de la propagación de las ondas de radio. Estas reflexiones multitrayectoria de la onda transmitida se deben a obstáculos naturales o a objetos que actúan como dispersores, entre estos se puede citar: muebles, ventanas, paredes, puertas metálicas, etc. Pequeñas variaciones en la distancia entre emisor y receptor, del orden de una cuarta parte de la longitud de onda por ejemplo, pueden causar grandes cambios en la amplitud o en la fase de la señal.

Estas características de desvanecimientos son propias del área donde se quiera implementar la red inalámbrica, estos desvanecimientos alteran las pérdidas en la trayectoria en espacio libre, y por lo general, son perjudiciales para la eficiencia general del sistema.

Cuando la señal electromagnética se propaga por una estancia es afectada por múltiples fenómenos debido a los diferentes tipos de obstáculos que debe atravesar o en los cuales es reflejado Es por tanto imprescindible tener en cuenta estos fenómenos, que causan atenuaciones y desvanecimientos de la señal original, a la hora de diseñar un enlace inalámbrico en ambientes internos.

1.3.2.2.1 Reflexión [2]

Cuando una onda electromagnética que se propaga por el aire incide contra un objeto de grandes dimensiones en comparación con la longitud de onda se dice que ocurre reflexión. La consecuencia de este fenómeno es que la señal puede ser absorbida, reflejada o a su vez una combinación de ambas. Esta reacción depende principalmente de:

- Propiedades de la señal, como orientación, ángulo de incidencia y longitud de onda.
- Propiedades físicas del obstáculo, como su geometría, textura y composición.

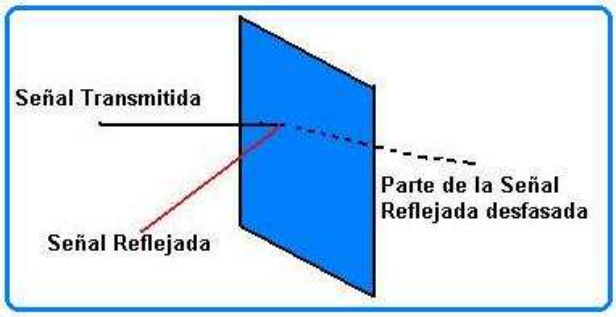


Figura 1.36 Reflexión de una señal [2]

1.3.2.2.2 Penetración [2]

La penetración es la capacidad de transmisión de una señal cuando ésta se encuentra en su camino con un obstáculo, cuando una señal penetra un obstáculo experimenta una pérdida, la cual será función del tipo de objeto.

En la tabla 1.8 se encuentran las pérdidas de penetración predecibles dependiendo del material.

Tipo de obstáculo	Pérdida (dB)
Espacio abierto	0
Ventana metálica (tintado no metálico)	3
Ventana metálica (tintado metálico)	5-8
Muros finos	5-8
Muros medios de madera	10
Muros gruesos	15-20
Muros muy gruesos	20-25
Suelo/Techo grueso	15-20
Suelo/Techo muy grueso	20-25

Tabla 1.8 Penetración a través de diferentes tipos de materiales [2]

1.3.2.2.3 Difracción [2]

La difracción ocurre cuando los obstáculos son impenetrables por las ondas de radio, el resultado de este fenómeno son ondas secundarias alrededor y detrás del obstáculo. La señal difractada depende de la geometría del objeto así como la amplitud, fase y polarización de la onda incidente en el punto de difracción.

Figura 1.37 Difracción de Señal [2]

1.3.2.2.4 Dispersión [2]

Cuando una señal transmitida se encuentra en el camino con objetos cuyas dimensiones son pequeñas con relación a la longitud de onda ocurre una dispersión. El resultado es que el frente de onda se rompe o se dispersa en múltiples direcciones. La dispersión en la práctica es provocada por:

- Señales de tráfico, focos.
- Conductos para los servicios eléctricos
- Estructuras de hierro forjado y tuberías.

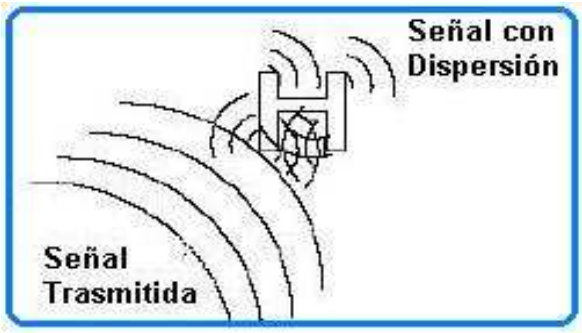


Figura 1.38 Dispersión de Señal [2]

1.3.2.2.5 Interferencia [2]

Las interferencias de radio frecuencia son uno de los asuntos más importantes a tener en cuenta para el éxito en el diseño, operación y mantenimiento de sistemas inalámbricos.

La señal de interferencia se caracteriza por ser una señal de naturaleza similar a la deseada, la misma que perturba a la señal que se desea transmitir. Se produce interferencia eléctrica cuando las señales de información de una fuente producen frecuencias que caen fuera de su ancho de banda asignado, e interfieren con otras señales de otra fuente.

“Las fuentes potenciales de interferencia de este tipo son numerosas: materiales metálicos, aislamientos, pinturas de plomo, etc. y pueden reducir la calidad de la señal radioeléctrica.” [2]

Existen otros dispositivos que utilizan la misma banda de frecuencia que también pueden ser fuente de interferencias como hornos microondas y ciertos teléfonos inalámbricos.

1.3.3 GANANCIA DE LA ANTENA [2]

La ganancia de la antena es la relación entre la intensidad de potencia radiada por la antena en una dirección específica y la intensidad de potencia radiada por una antena isotrópica alimentada con la misma potencia.

La mayoría de fabricantes especifica la ganancia de la antena en dBi (decibelios isotrópicos). Cada antena posee una ganancia diferente, la cual depende; de los materiales de elaboración, método de propagación (omnidereccionales o direccionales), es por eso que es necesario definir las características importantes de las antenas.

1.3.3.1 Características de las antenas

1.3.3.1.1 Diagrama de Radiación [2] [10]

El diagrama de radiación o lóbulo de radiación es la forma como se propaga la onda electromagnética. De acuerdo al diagrama de radiación existen dos tipos básicos de antenas que son; omnidireccionales y direccionales.

- **Antenas Omnidireccionales**

  Una antena omnidireccional es aquella diseñada para proveer un patrón de radiación de 360º. Propagan la señal de *RF* en todas las direcciones en el plano horizontal aunque tienen un rango limitado en el plano vertical. Son las más comunes en *WLAN* y se utilizan cuando se requiere dotar en un plano cobertura en todas las direcciones. Proporcionan la cobertura más amplia dentro de edificios, pudiendo formar celdas circulares mínimamente solapadas a lo largo del edificio.

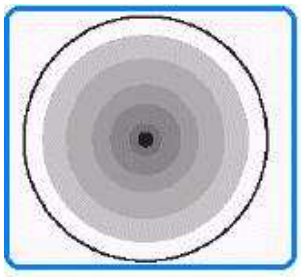


Figura 1.39 Diagrama de Radiación de una Antena Omnidireccional [2]

- **Antenas Direccionales**

  Las antenas direccionales son aquellas que han sido concebidas y construidas para que la mayor parte de la energía sea radiada en una dirección en concreto. Puede darse el caso en que se desee emitir en varias direcciones,

pero siempre que se este hablando de un número de direcciones determinado donde se encontrarán el lóbulo principal y los secundarios. Existen diferentes tipos de antena direccionales, cada una con una forma y estilo determinado, incluyendo *yagis*, antenas *patch* y parabólicas.

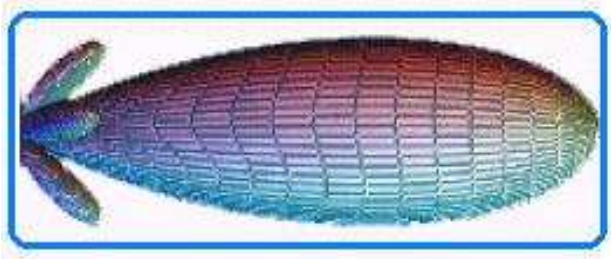


Figura 1.40 Diagrama de Radiación de una Antena Direccional [10]

1.3.3.1.2 Polarización de la Antena [10]

La polarización de una antena se refiere sólo a la orientación del campo eléctrico radiado desde ésta. Una antena puede polarizarse en forma lineal (en general, polarizada horizontal o vertical), en forma elíptica o circular.

Si una antena irradia una onda electromagnética polarizada verticalmente perpendicular al suelo, la antena se encuentra polarizada verticalmente; si la antena irradia una onda electromagnética polarizada horizontalmente paralela al suelo, se dice que la antena está polarizada horizontalmente.

1.3.3.1.3 Ancho de Banda [10]

El ancho de banda de la antena se define como el rango de frecuencias sobre las cuales la operación de la antena es "satisfactoria". Esto, por lo general, se toma entre los puntos de media potencia, pero a veces se refiere a las variaciones en la impedancia de entrada de la antena.

En *WLANs* las antenas tienen que estar sintonizadas para la banda de 2.4 *GHz* (*802.11b/g*) o 5 *GHz* (*802.11a*). Una antena funcionará de modo eficiente sólo si su ancho de banda está dentro de las frecuencias de radio utilizadas.

1.4 MODELOS PARA EL CÁLCULO DEL ENLACE

El cálculo del enlace consiste en calcular la potencia de recepción tomando en cuenta los diferentes obstáculos y dispositivos que pueden interferir con la señal transmitida y compararla con la sensibilidad del dispositivo, para determinar si estos cumplen con los requerimientos de cobertura del sistema inalámbrico.

1.4.1 MODELOS PARA EL CÁLCULO DEL ENLACE BLUETOOTH

Para el cálculo del enlace *Bluetooth* se consideró modelos para ambientes internos, ya que la implementación y las pruebas de los prototipos se las realizó en este tipo de entornos. A continuación se presentan los modelos analizados:

1.4.1.1 Modelo que considera las Pérdidas en la Trayectoria y Desvanecimientos Multitrayectoria [3] [35]

En este modelo la señal está sujeta a pérdidas en la trayectoria (L_{patch}) y desvanecimientos multitrayectoria (L_{fade}) debido a la presencia de obstáculos. Para el cálculo de la potencia que se espera recibir en el receptor se debe considerar además las ganancias de las antenas tanto transmisora como receptora, dando como resultado la siguiente expresión:

$$\mathrm{P_{RX}}[mW] = \frac{P_{TX}[mW] \bullet G_{TX} \bullet G_{RX}}{L_{patch} \bullet L_{fade}}$$ **Ecuación 1.5** [3]

Donde:

$\mathrm{P_{RX}}[mW]$ = Potencia de recepción

$P_{TX}[mW]$ = Potencia de transmisión

G_{TX} = Ganancia de la antena transmisora

G_{RX} = Ganancia de la antena receptora

L_{patch} = Pérdidas en la trayectoria

L_{fade} = Desvanecimientos por multitrayectoria

Si se transforma la ecuación 1.5 a dB se tiene:

$$P_{RX}[dBm] = P_{TX}[dBm] + G_{TX}[dB] + G_{TX}[dB] - L_{patch}[dB] - L_{fade}[dB] \quad \textbf{Ecuación 1.6} \ [3]$$

Las antenas utilizadas en la banda *ISM* no permiten antenas de alta directividad por lo que se usan antenas *omnidireccionales* con ganancia unitaria, con lo cual la ecuación 1.6 queda de la siguiente forma:

$$P_{RX}[dB] = P_{TX}[dBm] - L_{patch}[dB] - L_{fade}[dB] \quad \textbf{Ecuación 1.7} \ [3]$$

Las pérdidas por trayectoria se asumen a través de estimaciones de la siguiente forma:

$$L_{patch} = 20\log\left(\frac{4 \bullet \pi \bullet R}{\lambda}\right) \approx 40 + 20\log(R) \qquad R \leq 8.5[m] \quad \textbf{Ecuación 1.8} \ [3]$$

$$L_{patch} = 36\log\left(\frac{4 \bullet \pi \bullet R}{\lambda}\right) - 46.7[dB] \approx 25.3 + 36\log(R) \qquad R > 8.5[m] \quad \textbf{Ecuación 1.9} \ [3]$$

"Las pérdidas por desvanecimientos (L_{fade}) se asumen de 8[dB], se asume este valor ya que se estima una confiabilidad del sistema igual al 90%" [35]. Los desvanecimientos más profundos serán absorbidos por la diversidad de frecuencia del canal y causarán solo interrupciones ocasionales.

Entonces la ecuación para el cálculo del enlace para *Bluetooth* es la siguiente:

$$P_{RX}[dB] = P_{TX}[dBm] - L_{patch}[dB] - 8[dB] \quad \textbf{Ecuación 1.10} \ [3]$$

1.4.1.2 Modelo de Atenuación Lineal por Trayectoria

Este modelo es realmente sencillo en cuanto a su parte de aplicación teórica, pues la parte real de mediciones por pérdidas por trayectoria tienen un gran peso. Andelman lo propuso en el 2004 como un modelo a utilizarse cuando el transmisor y el receptor se encuentran en el mismo piso.

Este modelo de pérdidas por trayectoria lineal toma en cuenta trayectorias para interiores a partir de la potencia radiada, estas pérdidas están dadas por las pérdidas en el modelo del espacio libre, más el factor lineal que se obtiene experimentalmente.

La ecuación que describe las pérdidas en este modelo es la siguiente:

$$L = L_{FS} + a * d \quad \textbf{Ecuación 1.11} \ [18]$$

Donde:

L_{FS} = Pérdidas en el espacio libre

a = Coeficiente de atenuación lineal

d = distancia entre el transmisor y el receptor

"El coeficiente de atenuación lineal para un ambiente de oficinas es a=0.47 [dB/m]" [18]. Introduciendo la ecuación 1.4 en la ecuación 1.11 se tiene la siguiente ecuación:

$$L = 20 * \log\left(\frac{4\pi * f}{C}\right) + 20 * \log(d) + a * d \quad \textbf{Ecuación 1.12}$$

Para una frecuencia f=2.4 *GHz*, la velocidad de la luz C =3 * 10^8 m/s y el coeficiente de atenuación lineal a = 0.47 se tiene:

$$L = 40.1 + 20 * \log(d) + 0.47 * d \quad \textbf{Ecuación 1.13}$$

Tomando en cuenta que se usan antenas *omnidireccionales* con ganancia unitaria y las pérdidas totales la ecuación para el cálculo del enlace para *Bluetooth* es la siguiente:

$$P_{RX}[dBm] = P_{TX}[dBm] - 20 * \log(d)[dB] - 0.47 * d[dB] - 40.1[dB] \quad \textbf{Ecuación 1.14}$$

1.4.2 MODELOS PARA EL CÁLCULO DEL ENLACE Wi-Fi

Para el cálculo del enlace *Wi-Fi* al igual que en *Bluetooth* se utilizó modelos de propagación para ambientes internos, ya que las pruebas del prototipo se las realizó en este tipo de entornos. A continuación se analizarán los siguientes modelos:

1.4.2.1 Modelo de Pérdidas de Propagación de una Pendiente (1SM: *one-slope model)* [19] [36]

Este modelo (*one-slope model*), supone una dependencia lineal entre las pérdidas del trayecto en dB, y el logaritmo de la distancia.

$$L[dB] = L_0[dB] + 10 \bullet n \bullet \log(d)[dB]$$ **Ecuación 1.15** [19]

Donde:

L_0 = atenuación del trayecto para a una distancia de 1m.

n = exponente de pérdidas

d = distancia entre el transmisor y el receptor en (m)

Si se considera las pérdidas en el espacio libre:

$$L[dB] = 32.45 + 20 * \log(d) + 20 * \log(f)$$ **Ecuación 1.16** [36]

Para una distancia de un metro Lo:

$$Lo\ [dB] = 32.45 + 20 * \log(f)$$; *f* en *GHz*

Es el modelo más sencillo de utilizar pero necesita de una adecuada clasificación del tipo de edificio, para obtener un exponente de pérdidas para cada entorno que minimice la desviación típica. El principal problema de este modelo se da cuando se pretende utilizar en edificios de varias plantas, donde da lugar a grandes errores, por lo que se suele incluir un factor de pérdidas por penetración en el suelo.

La ecuación para el cálculo de las pérdidas para la banda de 2.4 *Ghz* es:

$$L[dB] = 40.1[dB] + 10 \bullet n \bullet \log(d)[dB] \quad \textbf{Ecuación 1.17} \ [19]$$

Tomando en cuenta que se usan antenas *omnidireccionales* con ganancia unitaria para la banda *ISM* la ecuación para el cálculo del enlace para *Wi-Fi* es la siguiente:

$$P_{RX}[dBm] = P_{TX}[dBm] - 10 \bullet n \bullet \log(d)[dB] - 40.1[dB] \quad \textbf{Ecuación 1.18}$$

Donde n toman los siguientes valores:

AMBIENTE	n
1 piso	4
atraviesa 2 paredes	5.2
Atraviesa más de 2 paredes	5.4
Abierto	1.9
Grande	2
Corredor	1.4

Tabla 1.9 Exponente de Pérdidas [19]

1.4.2.2 Modelo de Pérdidas con Factores de Atenuación por Suelo y Pared (MWM) [12] [20]

El modelo de *Keenan-Motley,* llamado también como modelo multi-pared (*MWM*), añade las pérdidas introducidas por las paredes y los suelos que atraviesa la onda directa entre el transmisor y el receptor. Su formulación más general viene dada por la siguiente expresión:

$$L = L_o + 10 * n * \log(d) + \sum_{i=1}^{I} K_{wi} * L_{wi} + \sum_{j=1}^{J} K_{fj} * L_{fj} \quad \textbf{Ecuación 1.19} \ [12]$$

L_o = pérdidas de referencia a 1 m en espacio libre.

n = pendiente de pérdidas con la distancia (n=2).

d = distancia entre el transmisor y el receptor.

I = número de categorías de paredes.

K_{wi} = número de paredes de la categoría *i*.

L_{wi} = pérdidas de la pared tipo *i*.

J = número de tipos de suelos.

K_{fj} = número de suelos de tipo *j*.

L_{fj} = pérdidas del suelo de tipo *j*.

La pérdida por suelos es una función no lineal del número de suelos atravesados. Esto se puede tener en cuenta introduciendo un factor empírico adicional. Es necesario indicar también que los factores de pérdidas de paredes y suelos no son pérdidas físicas reales, sino coeficientes del modelo, optimizados mediante procesos de medida.

Las normas *UMTS* de la *ETSI,* reconocen el modelo de *Keenan-Motley* modificado como una solución adecuada para el cálculo de la propagación en el interior de edificios, donde se incluye además del factor de pérdidas de penetración por suelos, un factor adicional de pérdidas de penetración por paredes u obstáculos. En la tabla 1.10 se puede observar algunas pérdidas que se deben considerar en el modelo de *Keenan-Motley.*

TIPO DE PÉRDIDA	DESCRIPCIÓN	FACTOR (dB)
L_f	**Suelos** • Baldosas • Revestimiento de hormigón • Espesor típico < 30 cm	18.3
L_{w1}	**Muros internos finos** • Yeso • Muros con ventanas	3.4
L_{w2}	**Muros internos** • Hormigón, ladrillos • Mínimo número de ventanas	6.9

Tabla 1.10 Factores de pérdidas según categoría [20]

La ecuación de *Keenan-Motley* para el caso de la banda de 2.4 GHz y considerando una pendiente de pérdidas n=2 es:

$$L = 40.1[dB] + 20 * \log(d) + \sum_{i=1}^{I} K_{wi} * L_{wi} + \sum_{j=1}^{J} K_{fj} * L_{fj}$$ **Ecuación 1.20** [12]

Tomando en cuenta que se usan antenas omnidireccionales con ganancia unitaria y las pérdidas de *Keenan-Motley,* la ecuación para el cálculo del enlace para Wi Fi es la siguiente:

$$P_{RX}[dBm] = P_{TX}[dBm] - \left(20 * \log(d) + \sum_{i=1}^{I} K_{wi} * L_{wi} + \sum_{j=1}^{J} K_{fj} * L_{fj} \right)[dB] - 40.1[dB]$$ **Ecuación 1.21** [12]

- **Tabla comparativa de las principales características de *Bluetooth* y *Wi-Fi***

En la tabla 1.11 se muestra la comparación de las principales características entre las tecnologías *Bluetooth* y *Wi-Fi.*

TEGNOLOGÍA	BLUETOOTH	WI-FI
Estándar	802.15.1	802.11 b
Velocidad	1 Mbps	11 Mbps
Medio	Inalámbrico	Inalámbrico
Topología	Ad-hoc e Infraestructura	Ad-hoc e Infraestructura
Modulación	GFSK	DBPSK (1 Mbps), DQPSK (2 Mbps), CCK (5.5 y 11 Mbps), DSSS
Potencia	2.5 mW	100 mW
Alcance	10 m	100 m
Banda de Frecuencia	ISM 2.4 GHz – 2.5 GHZ	ISM 2.4 GHz – 2.5 GHZ
Tipo de Enlace	Asincrónico y Sincrónico	Sincrónico
Ancho de Banda por canal	1 MHZ	22 MHz
Número de canales	79/23canales	77/23 canales
Consumo de energía	Baja (1/5 del consumo de Wi-Fi)	Alto
Seguridad	Obligatoria	Opcional

Tabla 1.11 Comparación teórica entre Bluetooth y Wi-Fi

CAPÍTULO 2

2. DISEÑO E IMPLEMENTACIÓN DE LOS PROTOTIPOS

Al momento de realizar el diseño de los prototipos inalámbricos *Bluetooth* y *Wi-Fi* se deben considerar los diferentes requerimientos para su correcto funcionamiento, entre éstos están: la cobertura, estándar, modo de operación y topología.

El área de cobertura en la cual se realizó el análisis de los prototipos es de 10 m, la frecuencia de trabajo es 2.4 *GHz ISM*, la topología utilizada es *Ad-Hoc*, el estándar para *Bluetooth* es *IEEE 802.15.1* y para *Wi-Fi IEEE 802.11 b.*

La implementación de los prototipos se la realizó en la *SUPTEL* (Superintendencia de Telecomunicaciones), debido a que en este lugar se cuenta con el equipo necesario para realizar medidas de potencia de transmisión de las interfaces utilizadas en los prototipos los cuales fueron facilitados para la realización del proyecto.

2.1 DISEÑO DE LOS PROTOTIPOS BLUETOOTH y WI-FI

Para el diseño del prototipo tanto *Bluetooth como Wi-Fi* se debe realizar un análisis del sitio en el cual se va a implementar, en base a un plano de construcción del lugar se definen las posibles ubicaciones de las estaciones dentro del área de cobertura deseada.

El principal objetivo de hacer un análisis del sitio es determinar las posibles causas y factores por las que se puede degradar el funcionamiento de los prototipos y de esta manera establecer si los interfaces que se van a utilizar están en la capacidad de satisfacer los requisitos de un enlace inalámbrico, esto es, alta capacidad, cobertura pequeña, conectividad de las estaciones, seguridad, facilidad de desplazamiento, etc.

El análisis de interferencias del sitio puede hacerse a través del *software* que viene incorporado en las NIC (Tarjeta de Interfaz de Red) inalámbricas, éstas permiten determinar el alcance y la calidad de un enlace inalámbrico, también se puede determinar la conectividad del enlace ejecutando un *ping*[5].

2.1.1 PLANO DE PLANTA ALTA DE LA *SUPTEL* (Superintendencia de Telecomunicaciones)

La figura 2.1 representa el plano arquitectónico de una de las instalaciones de la SUPTEL (Superintendencia de Telecomunicaciones), específicamente de la última planta, lugar donde se realizaron las pruebas de los prototipos.

[5] *Ping:* El comando ping es un programa básico que verifica conectividad en una red de computadoras enviando petición de envío y respuesta a través de datagramas de petición de eco.

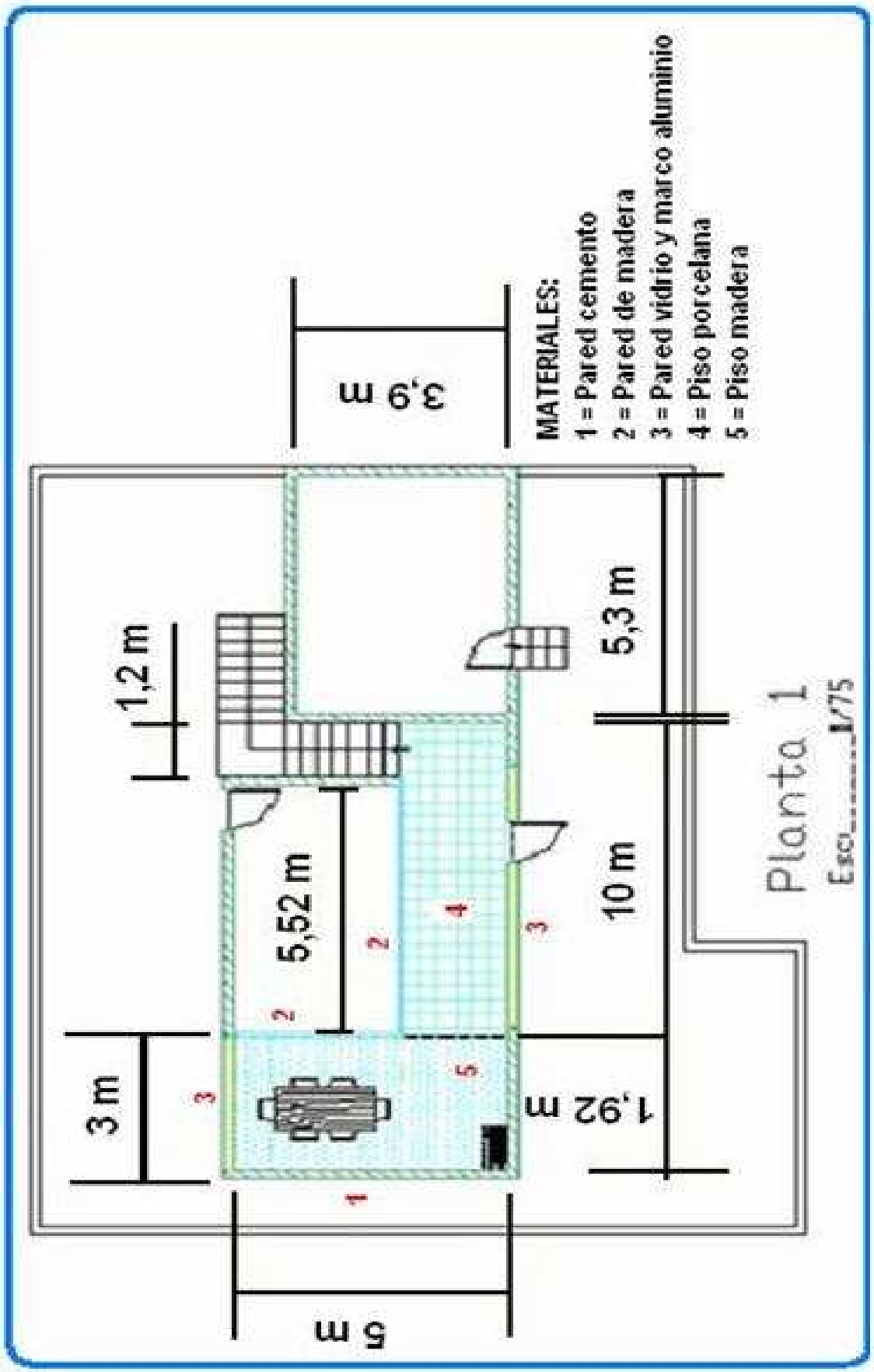


Figura 2.1 Plano de la planta alta de la SUPTEL (2D)

La figura 2.2 representa el plano arquitectónico del sitio en tres dimensiones, de lo que se puede apreciar en ésta y de la inspección que se hizo, se pudo determinar que existen paredes de distinto material los mismos que van a absorber o reflejar la señal en menor o mayor grado. También se observó que existen muebles y escritorios que van a degradar la señal así como un horno microondas que funciona en la banda de 2.4 *GHz* el mismo que cuando esté en funcionamiento podría causar interferencias.

Por último se observó que existen otras redes con tecnología *Wi-Fi* en el lugar, pertenecientes a Andinanet y el Banco del Pichincha que podrían interferir con los prototipos ya que funcionan en la misma banda de frecuencia de *2.4 GHz*.

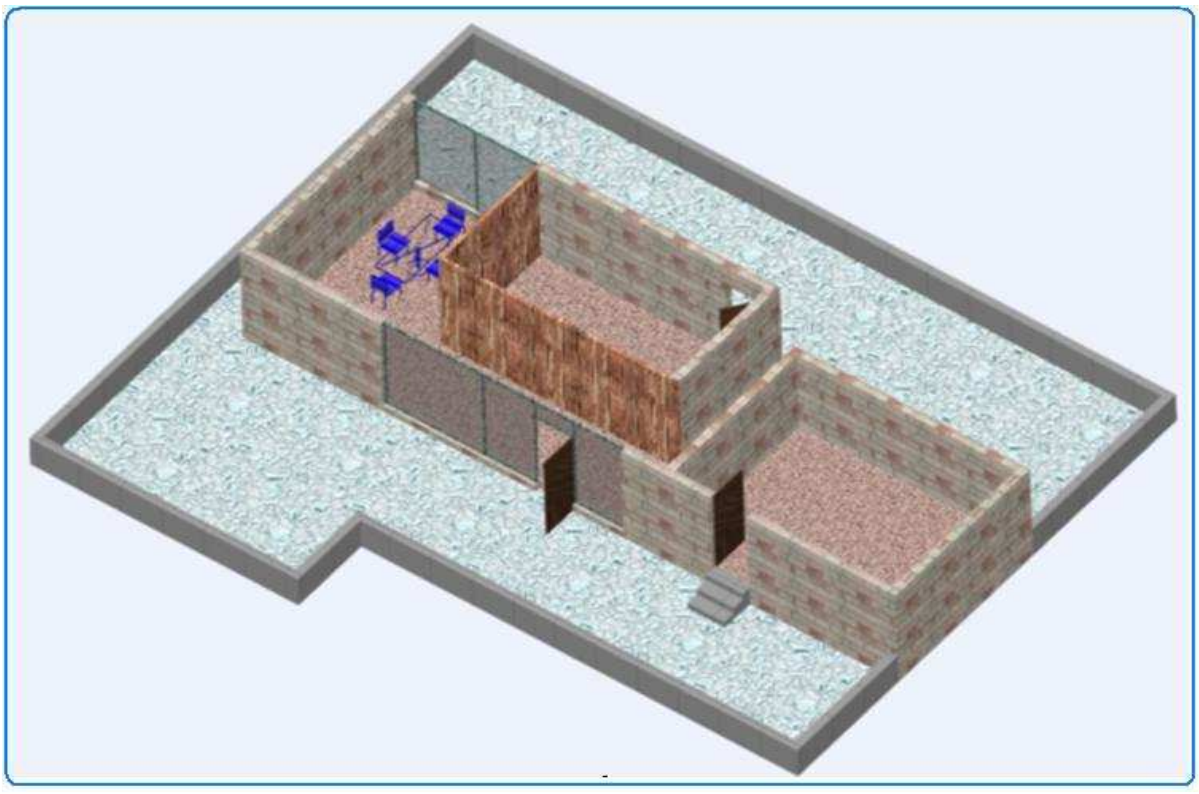


Figura 2.2 Plano Arquitectónico del lugar (3D)

2.1.2 IDENTIFICACIÓN DEL ÁREA DE COBERTURA

Como se puede apreciar en la figura 2.3 el área que se desea cubrir con los prototipos tanto *Bluetooth* como *Wi-Fi* es aproximadamente 10 metros.

Figura 2.3 Zona de Cobertura de los Prototipos

2.1.3 UBICACIÓN DE LAS ESTACIONES

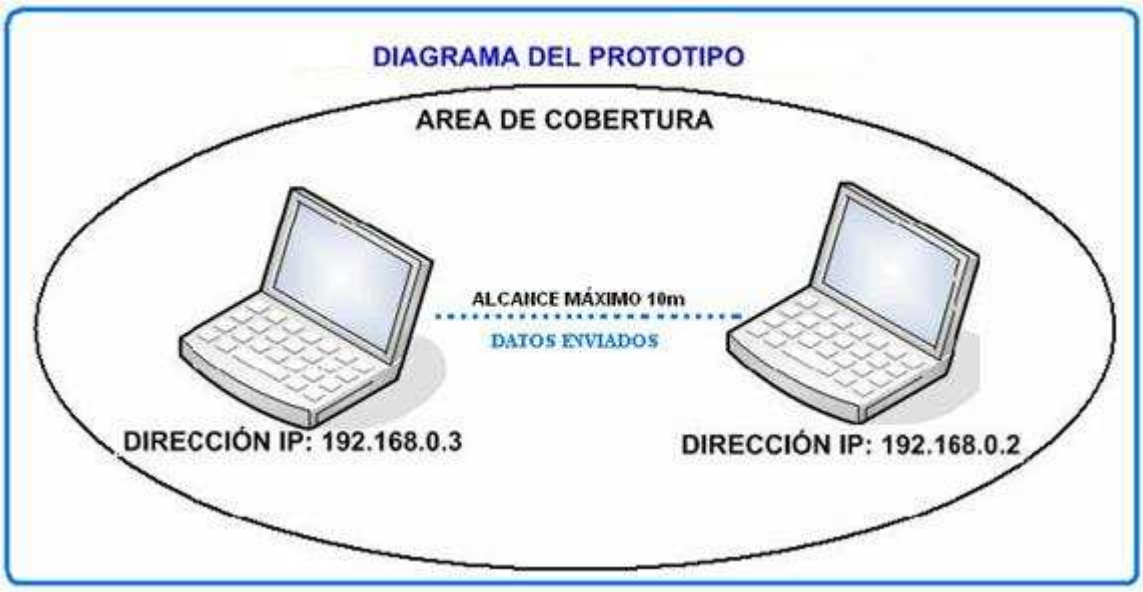


Figura 2.4 Diagrama de los Prototipos a Implementarse

La ubicación de las estaciones en una red *Ad-hoc* es esencial, ya que si existen obstáculos entre éstas la señal de transmisión podría degradarse e incluso podría perderse la comunicación entre las mismas.

Para determinar que los prototipos puedan cubrir un área determinada, se lo puede hacer en base a mediciones de campo, disponibilidad de la conexión entre los dispositivos de la red y mediante *software* que viene incorporado en las interfaces utilizadas para el prototipo *Bluetooth* y el prototipo *Wi-Fi*.

2.1.4 EQUIPOS A UTILIZARSE EN EL DISEÑO DE LOS PROTOTIPOS

Para la implementación de los prototipos inalámbricos, es necesario incorporar a las estaciones de trabajo dos adaptadores inalámbricos con tecnología *Bluetooth* para el prototipo *Bluetooth* y dos adaptadores inalámbricos con tecnología *Wi-Fi* para el prototipo *Wi-Fi*; estos dispositivos pueden ser de varios tipos y modelos, su elección dependerá de las características técnicas de cada uno, en la actualidad en el mercado existen gran variedad de adaptadores inalámbricos *PCMCIA, PCI* o *USB.*

2.1.4.1 Adaptadores Inalámbricos *Bluetooth y Wi-Fi*

En la tabla 2.1 se muestran los adaptadores con tecnología *Bluetooth* con sus principales características técnicas. Se eligió adaptadores *USB* debido a su fácil conexión.

	BELKIN. F8T001v	**D–Link** DBT–122	**ANYCOM** USB - 120
			
Especificaciones Técnicas			
Operación	Bluetooth 1.1	Bluetooth 1.2	Bluetooth 1.1
Almacenaje	Punto a Multipunto	Punto a Multipunto	Punto a Multipunto
Alcance de Cobertura			
Distancia	100 m	10 m	30 m
Potencia de Transmisión			
Potencia de transmisión	0 a 20 dBm	-6 a 4 dBm	0 a 13 dBm
Banda de Frecuencia			
Rango	2.4 – 2.483 GHz	2.4 – 2.483 GHz	2.4 – 2.483 GHz
Velocidad Máxima de transmisión de datos	56/723 Kbps asincrónico 400/420 Kbps sincrónico	723 Kbps asincrónico 433.9 Kbps sincrónico	1 Mbps

Tabla 2.1 Datos Técnicos de los Adaptadores *USB Bluetooth* [36] [37] [38]

2.1.4.2 Adaptadores Inalámbricos *Wi-Fi*

En la tabla 2.2 se puede observar las principales características técnicas de adaptadores inalámbricos *con tecnología Wi-Fi.*

	AIR802	DWL – G122	WUSB54GC
			

Especificaciones Técnicas

Operación	802.11 b/g 1.1	802.11 b/g 1.1	802.11 b 1.1
Almacenaje	Punto a Multipunto	Punto a Multipunto	Punto a Multipunto

Alcance de Cobertura

Distancia	300 m	100 m	100 m

Potencia de Transmisión

Clase 1	14 dBm ± 2dBm 802.11g 17 dBm ± 2dBm 802.11b	14 dBm ± 2dBm 802.11g	14 dBm ± 2dBm 802.11g

Banda de Frecuencia

Rango	2.4 a 2.497 GHz 2.4 a 2.4835 GHz	2.4 GHz	2.4 GHz
Velocidad Máxima de transmisión de datos	6, 9, 12, 18,24, 36, 48, 54 Mbps	54 Mbps	54 Mbps

Tabla 2.2 Datos Técnicos de los Adaptadores *USB Wi-Fi* [36] [39] [40]

2.1.4.3 Comparación de los Equipos

El análisis técnico comparativo se basa en los siguientes parámetros: potencia, velocidad, frecuencia de operación y cobertura de los dispositivos; estos parámetros permiten determinar qué dispositivo se debe utilizar de acuerdo a las necesidades del enlace inalámbrico.

2.1.4.3.1 Comparación de los Equipos Bluetooth

Como se puede observar en la tabla 2.1 todos los dispositivos trabajan en la banda de frecuencia de 2.4 *GHz,* la velocidad de los dispositivos *D-Link* y *BELKIN* es 723 kbps, *ANYCOM* presenta una velocidad de 1 *Mbps*.

El área de cobertura a cubrir con los dispositivos *Bluetooth* es 10 m, en la tabla 2.1 se puede apreciar que todos los fabricantes citados cumplen con éste requerimiento.

2.1.4.3.2 Comparación de los Equipos Wi-Fi

Como se puede observar en la tabla 2.2 todos los dispositivos *Wi-Fi* pueden cubrir el área de cobertura deseada, la banda de frecuencia de trabajo es 2.4 *GHz*, la velocidad máxima es 54 *Mbps* para estos tres fabricantes.

2.1.4.4 Costos Referenciales de los Equipos

La disponibilidad de los equipos *Bluetooth* y *Wi-FI* en el mercado no es la misma; existen marcas que se comercializan localmente en el país, y otras que no se comercializan localmente, los cuales deben ser adquiridos fuera del país.

El precio de los equipos varía de acuerdo a los fabricantes y si están disponibles a nivel local o no, a continuación se presenta en la tabla 2.3 y en la tabla 2.4 los precios de los equipos que se podrían utilizar para los prototipos *Bluetooth* y *Wi-Fi* respectivamente. Cabe mencionar que todos los costos incluyen IVA y para los equipos que no son comercializados localmente se incluye el costo de envío.

En la tabla 2.3 y 2.4 todos los precios indicados incluyen IVA.

FABRICANTE	MODELO	PRECIO UNITARIO (USD)	PRECIO DE ENVIO (USD)	COSTO POR UNIDAD (USD)
ANYCOM	USB - 120	50	0	50
BELKIN	F8T001v	50	50	100
D – LINK	DBT - 122	28.605	0	28.605

Tabla 2.3 Precios de los Equipos *Bluetooth* (fecha 25/06/2006) [36]

FABRICANTE	MODELO	PRECIO UNITARIO (USD)	PRECIO DE ENVIO (USD)	COSTO POR UNIDAD (USD)
D – LINK	DWL – G122	35.615	0	35.615
EVCOM	AIR802	38	50	88
LINKSYS	WUSB54GC	30	50	80

Tabla 2.4 Precios de los Equipos *Wi–Fi* (fecha 25/06/2006) [36]

2.1.4.5 Selección de los equipos a utilizarse en la implementación de los Prototipos

Técnicamente los equipos *Bluetooth* y *Wi-Fi* de los diferentes fabricantes tienen las mismas características aplicables al proyecto, por lo que todos ellos podrían ser de utilidad para realizar la implementación de los prototipos; la elección del equipo se basa principalmente en la disponibilidad en el mercado, facilidad de adquisición y costo.

Se ha seleccionado al equipo *DBT-122* para el prototipo *Bluetooth* y el equipo *DWL-G122 para el prototipo Wi-Fi* ambos modelos son del fabricante *D-Link,* ya que es una marca comercial en Ecuador, tiene gran disponibilidad de sus equipos y se los puede adquirir fácil e inmediatamente. Otra razón para la selección del fabricante es el costo, *D-Link* ofrece precios accesibles, y cumplen con los requerimientos de diseño.

2.1.5 CÁLCULO DEL ÁREA DE COBERTURA

Para el cálculo del área de cobertura para los prototipos *Bluetooth* y *Wi-Fi* se utilizaron dos modelos para cada prototipo, cada modelo utilizado se adapta de mejor manera para cada tecnología.

2.1.5.1 Cálculo del área de cobertura para Bluetooth

A continuación se procede a calcular el área de cobertura para el prototipo *Bluetooth.*

- **MODELO 1: Modelo que considera las Pérdidas en la Trayectoria y Desvanecimientos Multitrayectoria.**

Para el cálculo de la potencia de recepción se utiliza la ecuación 1.10

$$P_{RX}[dBm] = P_{TX}[dBm] - L_{patch}[dB] - 8[dB]$$

R = 10 m distancia de separación máxima entre estaciones
P_{TX} = - 6 dBm potencia especificada por la interfaz *DBT-122*

Cálculo de la potencia de recepción

Como se tiene una distancia mayor a 8.5 m se utiliza la ecuación 1.9 que permite calcular las pérdidas en la trayectoria L_{patch}

$$L_{patch} = 25.3 + 36\log(R)$$
$$L_{patch} = 25.3 + 36\log(10)$$
$$L_{patch} = 61.3\ [dB]$$
$$P_{RX}[dBm] = P_{TX}[dBm] - L_{patch}[dB] - 8[dB]$$
$$P_{RX} = -\ 6\ [dBm] - 61.3\ [dB] - 8\ [dB]$$
$$P_{RX} = -75.3\ [dBm]$$

- **MODELO 2: Modelo de Atenuación Lineal por Trayectoria**

Para el cálculo de la potencia de recepción se utiliza la ecuación 1.14

$$P_{RX}[dBm] = P_{TX}[dBm] - 20 * \log(d)[dB] - 0.47 * d[dB] - 40.1[dB]$$

R = 10 m distancia de separación máxima entre estaciones
P_{TX} = - 6 dBm potencia especificada por la interfaz *DBT-122*

Cálculo de la potencia de recepción

$$P_{RX} = -6[dBm] - 20 * \log(10)[dB] - 0.47 * 10[dB] - 40.1[dB]$$
$$P_{RX} = -6[dBm] - 20[dB] - 4.7[dB] - 40.1[dB]$$
$$P_{RX} = -70.8[dBm]$$

2.1.5.2 Cálculo el área de cobertura para Wi Fi

A continuación se procede a calcular el área de cobertura para el prototipo *Wi-Fi* con modelos que se adaptan de mejor manera a esta tecnología.

- **MODELO 1: Modelo de pérdida de propagación de una pendiente.**

Para el cálculo del enlace se considera un exponente de pérdidas n=4 que se establece en la tabla 1.9 y la ecuación 1.18.

$$P_{RX}[dBm] = P_{TX}[dBm] - 10 \bullet n \bullet \log(d)[dB] - 40.1[dB]$$

d = 10 m distancia de separación máxima entre estaciones
P_{TX} = 14 dBm potencia especificada por la interfaz *DWL-G122*

$$P_{RX} = 14[dBm] - 10 \bullet 4 \bullet \log(10)[dB] - 40.1[dB]$$
$$P_{RX} = -66.1\ [dBm]$$

- **MODELO 2: Pérdidas con factores de atenuación por suelo y pared**

Para el cálculo del enlace con este modelo, de la figura 2.2 correspondiente al plano arquitectónico del lugar se determina que en las peores condiciones la señal transmitida debe atravesar dos paredes, el índice de pérdidas por pared es $L_{wi} = 3.4$ que se encuentra en la tabla 1.10.

El cálculo se lo realiza en base a la ecuación 1.21.

$$P_{RX}[dBm] = P_{TX}[dBm] - \left(20 * \log(d) + \sum_{i=1}^{I} K_{wi} * L_{wi} + \sum_{j=1}^{J} K_{fj} * L_{fj} \right)[dB] - 40.1[dB]$$

d = 10 m distancia de separación máxima entre estaciones

P_{TX} = 14 dBm potencia especificada por la interfaz *DWL-G122*

$$P_{RX} = 14[dBm] - (20 * \log(10) + 2 \bullet 3.4 + 0)[dB] - 40.1[dB]$$

$$P_{RX} = -52.9[dBm]$$

2.1.5.3 Análisis de Resultados del Cálculo del Área de Cobertura

Una vez realizado el cálculo del enlace para *Bluetooth* y *Wi-Fi* en base a los dos modelos descritos para cada tecnología, se puede concluir que los dispositivos *DBT-122* elegidos para el prototipo *Bluetooth* y *DWL-G122* para el prototipo *Wi-Fi* cumplen con los requerimientos de cobertura deseada. Esto se puede observar en la tabla 2.5 y la tabla 2.6, la cual indica que la potencia de recepción calculada es mayor que la sensibilidad del receptor especificada en los dispositivos.

MODELOS BLUETOOTH	POTENCIA TX (dBm) DEL DBT -122	DISTANCIA MÁXIMA (m)	POTENCIA Rx Calculada (dBm)	Sensibilidad del DBT -122 (dBm)
MODELO 1	- 6	10	-75.3	-80
MODELO 2	- 6	10	-70.8	-80

Tabla 2.5 Cálculo del área de cobertura *Bluetooth*

MODELOS WI-FI	POTENCIA TX (dBm) DEL DWL-G122	DISTANCIA MÁXIMA (m)	POTENCIA Rx Calculada (dBm)	Sensibilidad del DWL -G122 (dBm)
MODELO 1	14	10	-66.1	-82
MODELO 2	14	10	-52.9	-82

Tabla 2.6 Cálculo del área de cobertura *Wi-Fi*

2.1.6 ESTÁNDAR DE LOS PROTOTIPOS

A pesar que los prototipos operen en la misma banda de frecuencia no significa que utilicen la misma tecnología y estándar, es por esta razón que se describe el estándar utilizado por cada prototipo.

2.1.6.1 Estándar del Prototipo Bluetooth

El estándar *IEEE 802.15.1* trabaja en la banda de frecuencia *ISM* de 2.4 *GHz*, en esta banda se puede implementar libremente cualquier red *WLAN*, la modulación utilizada por *IEEE 802.15.1* es *GFSK.*

La máxima velocidad que permite el estándar es de 723 *Kbps*, en un enlace asimétrico, el cual cumple con los requerimientos del diseño en cuanto a la transferencia de datos, lo cual es aceptable en una red inalámbrica *Ad-hoc*.

2.1.6.2 Estándar del Prototipo Wi-Fi

El estándar *IEEE 802.11b* trabaja en la banda de frecuencia *ISM* de 2.4 GHz, en la cual el prototipo *Wi-Fi* puede ser implementado libremente, la modulación utilizada por *IEEE 802.11b* es *DSSS*, la máxima velocidad teórica que permite el estándar es de 11 *Mbps.*

2.1.7 MODO DE OPERACIÓN Y TOPOLOGÍA

La topología utilizada en el diseño de los prototipos *Bluetooth* y *Wi-Fi* es la topología *Ad-Hoc*, la cual permite trabajar con sistemas inalámbricos punto a punto, estos sistemas se ajustan a los requerimientos de movilidad e independencia de las estaciones de trabajo ya que posee una arquitectura dinámica.

La ubicación de las estaciones en una red *Ad-hoc* es importante ya que al ser una red nómada estas pueden movilizarse de un lugar a otro dentro del área de

cobertura, motivo por el cual es indispensable ubicar las estaciones en sitios donde no existan obstáculos que impidan el correcto desempeño del prototipo.

- **Flexibilidad:** la flexibilidad en una red *Ad-hoc* es aceptable, ya que en este tipo de red las estaciones pueden cambiar fácilmente su topología física, es decir las estaciones no están sujetas a una posición fija.

- **Escalabilidad:** la escalabilidad en una red *Ad-hoc* para las tecnologías *Bluetooth y Wi-Fi* es aceptable, ya que permite incrementar el número de usuarios sin cambiar su topología, al incrementar el número de usuarios disminuye la velocidad de transmisión de datos, lo cual viene a ser una limitación de los prototipos.

2.1.8 SEGURIDAD

Bluetooth implementa un sistema de seguridad obligatorio, el cual se lo realiza al momento de emparejamiento de las estaciones. Mientras que *Wi-Fi* no utiliza un sistema de seguridad obligatorio motivo por el cual no se lo describe.

A continuación se explica el proceso de emparejamiento ofrecido por el interfaz *Bluetooth DBT-122* utilizado:

1. Identificar y visualizar todos los dispositivos *Bluetooth* que se encuentran en el área de cobertura del sistema, esto permite tener una lista de todos los usuarios que podrían tener acceso al sistema.

2. Para que un usuario acceda al sistema, es necesario ingresar un código *PIN* (número de identificación personal) de hasta 16 caracteres, el cual permite verificar la identidad de los usuarios que intervienen en la comunicación.

3. Autorización, aquí se permite o se niega el acceso del usuario a la red.

Luego de ejecutado este proceso se realiza la transmisión de datos. La figura 2.5 representa la solicitud del código *PIN* en la interfaz *DBT-122*

Una guía de este procedimiento se lo presenta en el ANEXO A

Figura 2.5 Ingreso del Código PIN

2.2 VERIFICACIÓN DEL FUNCIONAMIENTO DE LOS PROTOTIPOS

Para verificar el funcionamiento del los prototipos se realizó las siguientes pruebas: transmisión de datos y conectividad entre estaciones.

2.2.1 PRUEBAS DEL PROTOTIPO *BLUETOOTH*

Las pruebas que se realizaron consisten en transmitir paquetes a través del comando *ping* cada metro, desde una distancia mínima de un metro hasta una distancia máxima de 15 m. como se muestra en la figura 2.4

2.2.1.1 Conectividad de las estaciones

La verificación de conectividad entre estaciones en el prototipo *Bluetooth* se la realiza verificando los tiempos de respuesta a través del comando *ping*.

El comando *ping* entrega la siguiente información: tiempos de respuesta mínimo, máximo y promedio, y pérdida de paquetes. En la figura 2.6 se observa el funcionamiento del comando *ping* con el cual se verifica la conectividad existente entre las estaciones.

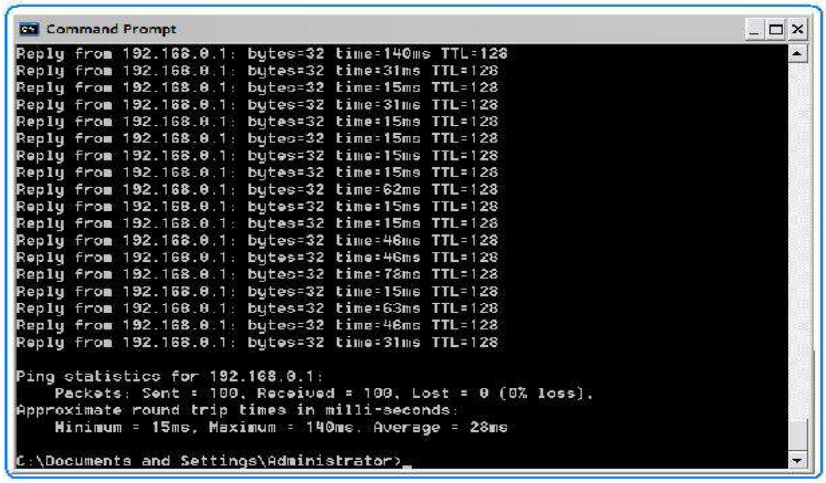


Figura 2.6 Ping entre las Estaciones del Prototipo *Bluetooth*

2.2.1.2 Estado de conexión Bluetooth

El estado de la conexión se verifica con el *software* incorporado en la interfaz *DBT-122* utilizada en el prototipo, mediante este se determina el nivel de señal recibido. En la figura 2.7 se observa que la señal recibida es buena.

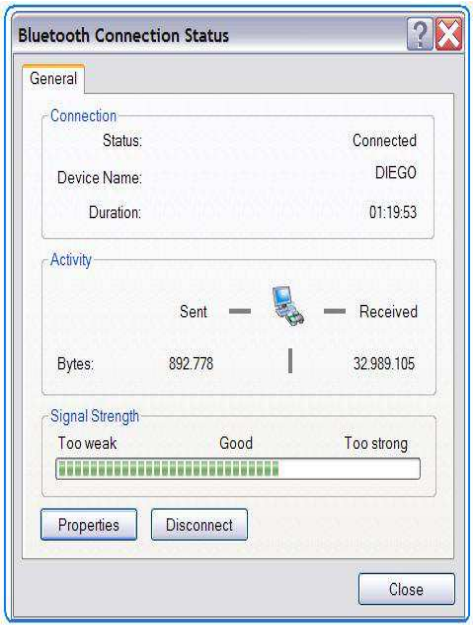


Figura 2.7 Estado de conexión

2.2.1.3 Transmisión de Datos Bluetooth

La prueba de transmisión de datos se la realizó mediante el comando *ping* entre las estaciones, el comando *ping* consiste en enviar 100 paquetes de 32 *bytes* y esperar una respuesta a esta petición.

Esta prueba se la realizó con una distancia de separación mínima entre estaciones de 1 m, hasta una distancia máxima de 15 m que supera el área cobertura de 10 m calculada en el diseño. Estas medidas se las realizaron ya que los dispositivos utilizados permitieron un alcance máximo de 15 m, esto también permitió observar la degradación del prototipo *bluetooth*.

En cada posición se realizó la toma de datos obtenidos con el comando *ping*, mediante éstos se puede comprobar si existe pérdidas de datos y los retardos entre estaciones.

Para visualizar de mejor manera los resultados de las pruebas del prototipo *Bluetooth* se las agrupó en la tabla 2.7.

Distancia (m)	Paquetes enviados	Paquetes recibidos	Paquetes perdidos (%)	Tiempo de respuesta mínimo (ms)	Tiempo de respuesta máximo (ms)	Tiempo de respuesta medio (ms)
1	100	100	0	19	120	37
2	100	100	0	19	120	39
3	100	100	0	19	130	40
4	100	100	0	19	160	38
5	100	100	0	19	120	39
6	100	100	0	19	140	39
7	100	100	0	12	140	38
8	100	100	0	19	140	40
9	100	100	0	20	140	40
10	100	100	0	19	160	38
12	100	100	0	19	130	40
15	100	100	0	19	562	55

Tabla 2.7 Resultados obtenidos en el Prototipo *Bluetooth*

El tiempo de respuesta es el tiempo que se tarda desde que la petición es enviada hasta recibir una respuesta a dicha petición, la cual se la realiza con el comando *ping*.

En la tabla 2.7 se puede observar que existen tiempos de respuesta altos, estos valores se deben a interferencias de otras redes y obstáculos existentes en el sitio donde se realizaron las pruebas, también se observa que en el escenario donde se realizaron las pruebas el prototipo *Bluetooth* no presenta pérdidas de datos

2.2.1.4 Representación Gráfica de los Resultados Obtenidos en el Prototipo Bluetooth

La figura 2.8 representa el tiempo mínimo de respuesta en función de la distancia. El tiempo mínimo de respuesta es el menor tiempo que tarda la estación en recibir una respuesta a su petición.

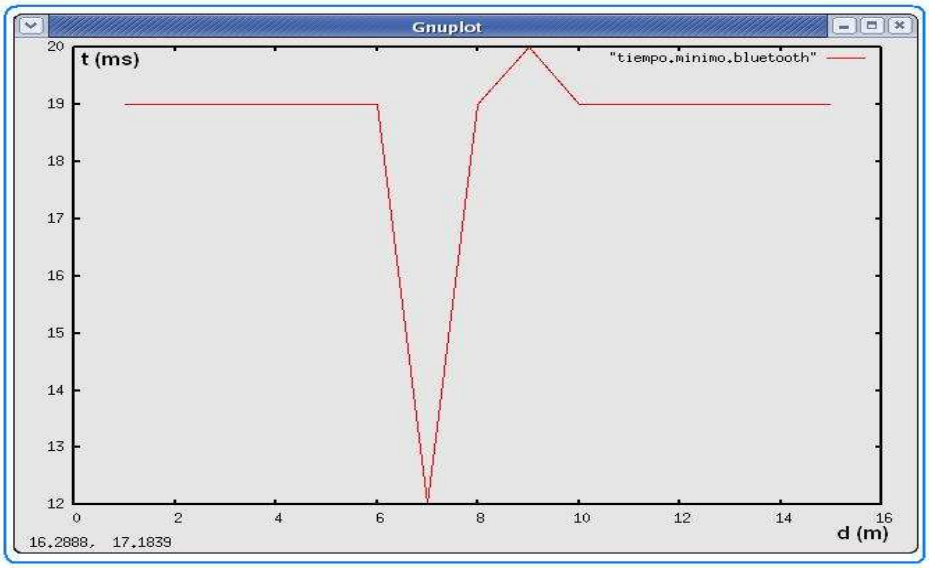


Figura 2.8 Tiempo de respuesta mínimo vs Distancia (*BLUETOOTH*)

La figura 2.9 representa la variación del tiempo de respuesta máximo en función de la distancia. El tiempo máximo de respuesta es el mayor tiempo que tarda la estación en recibir una respuesta a su petición con el comando *ping*.

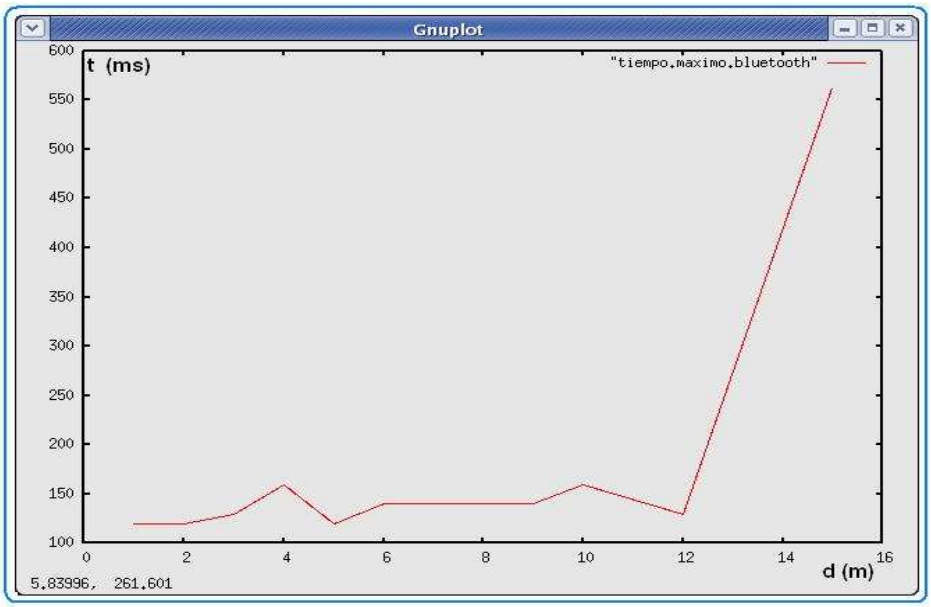


Figura 2.9 Tiempo de respuesta máximo vs Distancia (*BLUETOOTH*)

La figura 2.10 representa la variación del tiempo de respuesta medio en función de la distancia. El tiempo medio de respuesta es el promedio de todos los tiempos de respuesta que se tienen con el comando *ping*.

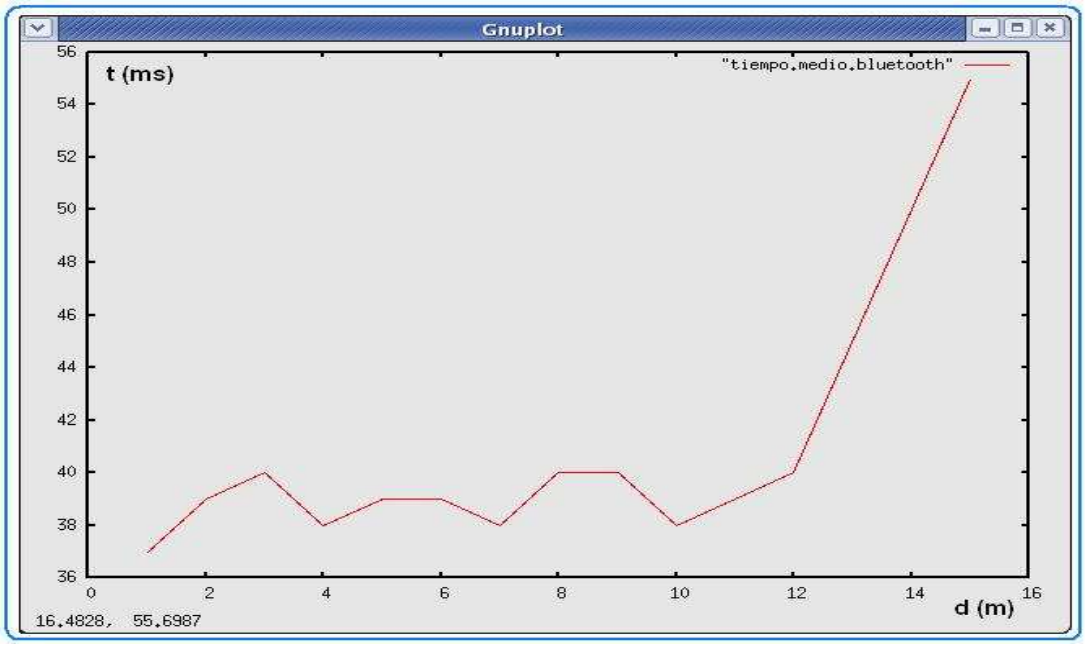


Figura 2.10 Tiempo de respuesta medio vs Distancia (*BLUETOOTH*)

De la figura 2.9 y figura 2.10 se concluye que, a medida que aumenta la distancia el tiempo de respuesta aumenta.

Mediante el *software* de la interfaz *DBT-122* se comprobó que el enlace está disponible en cada posición y también indica una velocidad de 700 *Kbps*.

2.2.2 PRUEBAS DEL PROTOTIPO WI-FI

Las pruebas que se realizaron con el prototipo *Wi-Fi* consisten en transmitir paquetes a través del comando ping cada metro, desde una distancia mínima de un metro hasta una distancia máxima de 15 m. como se muestra en la figura 2.4

2.2.2.1 Conectividad de las estaciones

La verificación de conectividad entre estaciones en el prototipo *Wi-Fi* se la realiza verificando los tiempos de respuesta a través del comando *ping.*

El comando *ping* entrega la siguiente información: tiempos de respuesta mínimo, máximo y promedio, y pérdida de paquetes. En la figura 2.11 se observa el funcionamiento del comando *ping* con el cual se verifica la conectividad existente entre las estaciones.

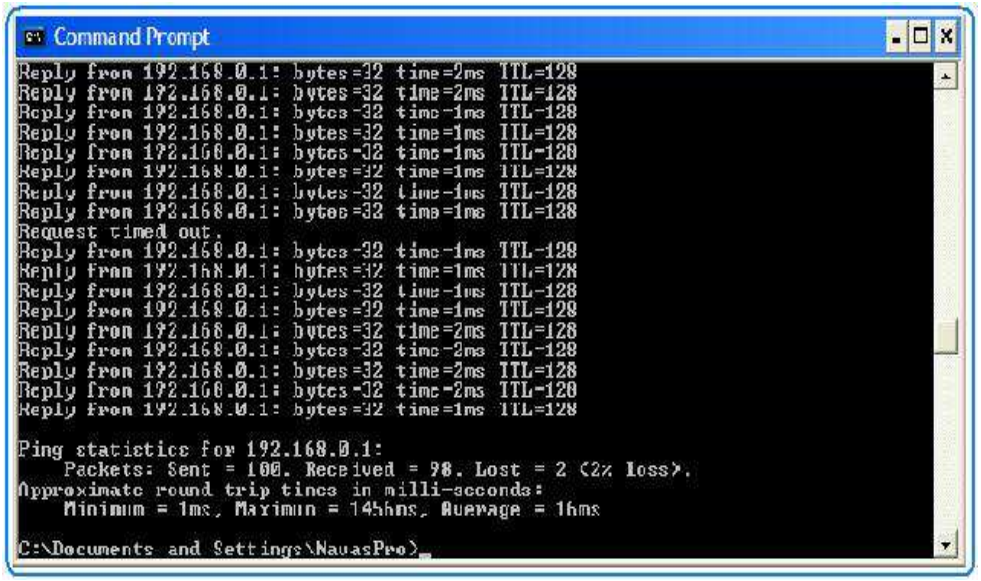


Figura 2.11 Ping entre las Estaciones del Prototipo Wi -Fi

2.2.2.2 Estado de conexión Wi-Fi

El estado de la conexión se lo verifica con el *software* incorporado en la interfaz *DWL-G122* utilizada en el prototipo, en la figura 2.12 se observa que el nivel de señal recibido es bueno.

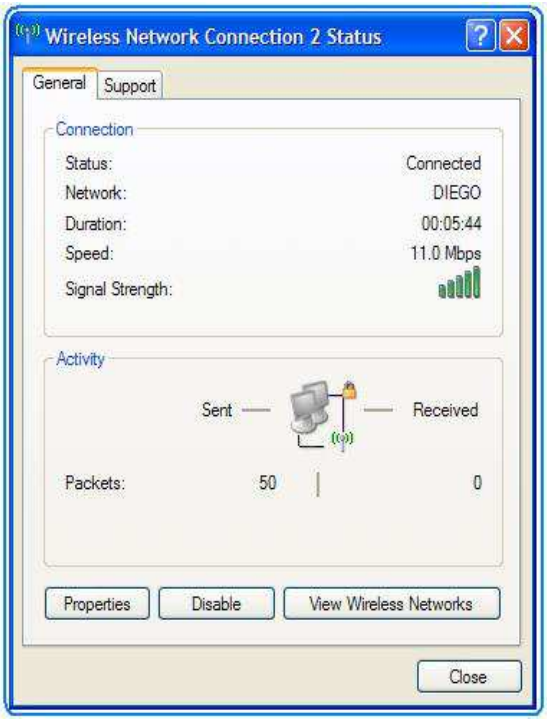


Figura 2.12 Estado de conexión *Wi-Fi*

2.2.2.3 Transmisión de Datos Wi - Fi

La prueba de transmisión de datos *Wi-Fi* se la realizó mediante el comando *ping* entre las estaciones, el comando *ping* consiste en enviar 100 paquetes de 32 *bytes* y esperar una respuesta a esta petición.

Esta prueba se la realizó con una distancia de separación mínima entre estaciones de 1 m, hasta una distancia máxima de 15 m que supera el área cobertura de 10 m calculada en el diseño. Estas medidas se las realizaron ya que los dispositivos utilizados permitieron un alcance mayor, esto también permitió observar la degradación del prototipo *Wi-Fi*.

En cada posición se realizo la toma de datos obtenidos con el comando *ping*, mediante éstos se puede comprobar si existe pérdidas de datos y los retardos entre estaciones.

Para visualizar de mejor manera los resultados de las pruebas del prototipo *Wi-Fi* se las agrupó en la tabla 2.8.

Distancia (m)	Paquetes enviados	Paquetes recibidos	Paquetes perdidos (%)	Tiempo mínimo (ms)	Tiempo máximo (ms)	Tiempo medio (ms)
1	100	99	1	1	2441	58
2	100	98	2	1	2189	23
3	100	99	1	1	2381	27
4	100	99	1	1	2650	58
5	100	97	3	1	2601	50
6	100	98	2	1	2610	54
7	100	99	1	1	2370	25
8	100	98	2	1	2359	61
9	100	96	4	1	2611	81
10	100	97	3	1	2631	53
12	100	81	19	1	3155	322
15	100	30	70	1	3878	894

Tabla 2.8 Resultados obtenidos en el Prototipo *Wi-Fi*

El tiempo de respuesta es el tiempo que se tarda desde que la petición es enviada hasta recibir una respuesta a dicha petición, la cual se la realiza con el comando *ping*.

En la tabla 2.8 se puede observar que existen tiempos de respuesta altos, estos valores se deben a interferencias de otras redes y obstáculos existentes en el sitio donde se realizaron las pruebas, también se observa que el prototipo *Wi-Fi* presenta mayor pérdida de datos a medida que las estaciones se alejan.

2.2.2.4 Representación Gráfica de los Resultados Obtenidos en el prototipo Wi-Fi

La figura 2.13 representa el tiempo mínimo de respuesta en función de la distancia. El tiempo mínimo de respuesta es el menor tiempo que tarda la estación en recibir una respuesta a su petición.

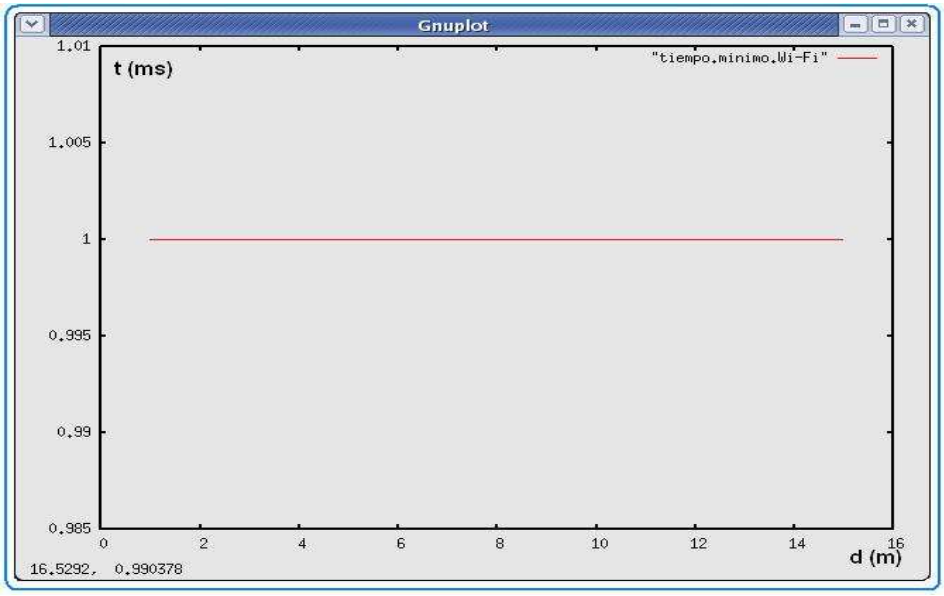


Figura 2.13 Tiempo de respuesta mínimo vs Distancia (*Wi-Fi*)

La figura 2.14 representa la variación del tiempo de respuesta máximo en función de la distancia. El tiempo máximo de respuesta es el mayor tiempo que tarda la estación en recibir una respuesta a su petición con el comando *ping*.

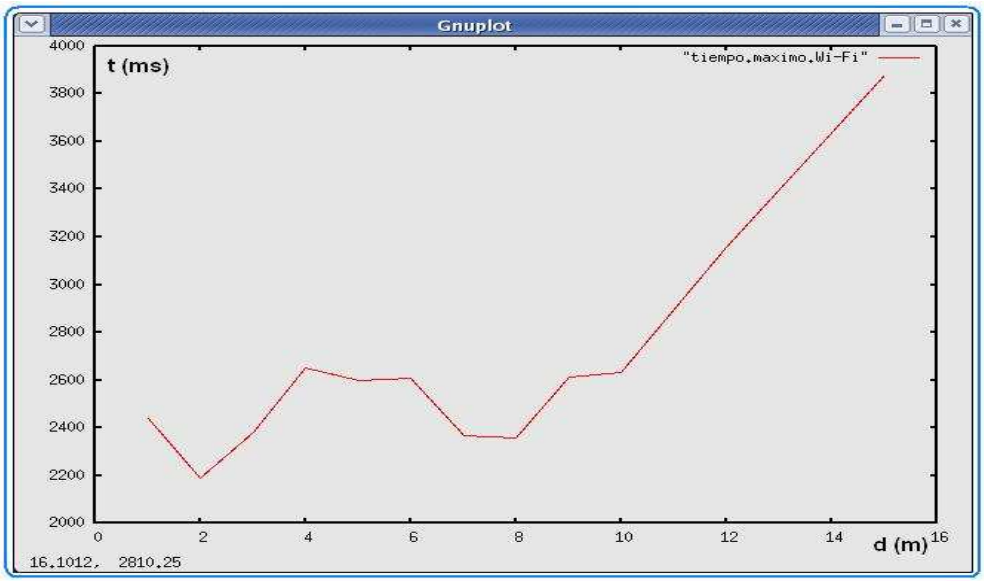


Figura 2.14 Tiempo de respuesta máximo vs Distancia (*Wi-Fi*)

La figura 2.15 representa la variación del tiempo de respuesta medio en función de la distancia. El tiempo medio de respuesta es el promedio de todos los tiempos de respuesta que se tienen con el comando *ping*.

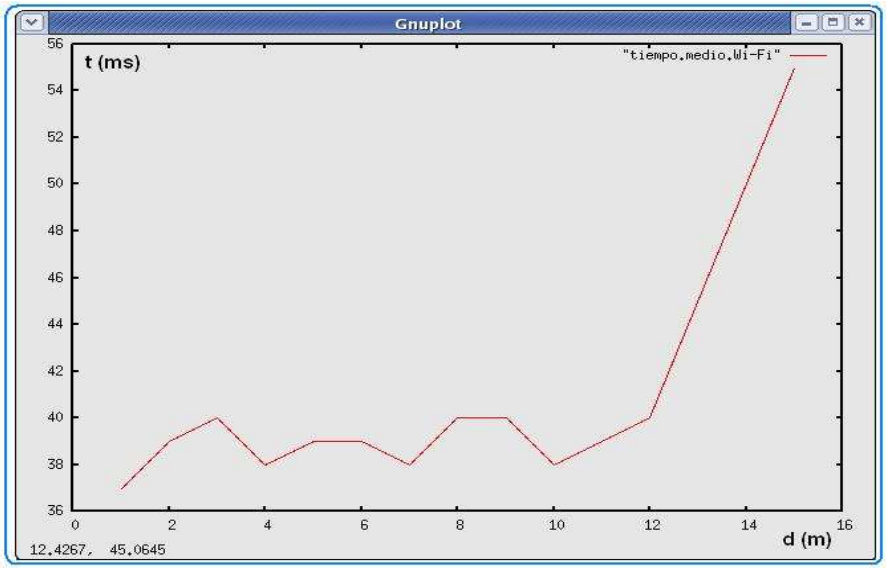


Figura 2.15 Tiempo de respuesta medio vs Distancia (*Wi-Fi*)

De la figura 2.14 y figura 2.15 se concluye que, a medida que aumenta la distancia el tiempo de respuesta aumenta.

La tabla 2.9 representa el estado de conexión del enlace *Wi-Fi* y la intensidad de potencia de recepción que indica el *software* de la interfaz *DWL-G122* utilizada en el prototipo. También este software indica una velocidad de transmisión de 11 *Mbps* para cada posición.

Distancia (m)	Estado conectado	Intensidad potencia (%)
1	SI	100
2	SI	100
3	SI	100
4	SI	100
5	SI	80
6	SI	80
7	SI	80
8	SI	60
9	SI	40
10	SI	40
12	SI	20
15	SI/NO	0-20

Tabla 2.9 Estado de conexión y velocidad de la interfaz

De las pruebas realizadas, se puede apreciar que el prototipo *Wi-Fi*, posee una mayor tasa de transferencia pero presenta una mayor pérdida de datos. Por otro lado *Bluetoot*h tiene menor velocidad de transferencia, pero para este caso el prototipo *Bluetooth* no presenta pérdida de datos.

2.3 COSTO REFERENCIAL DE LOS PROTOTIPOS

Una vez realizada la fase de diseño de los prototipos, es necesario hacer una investigación de costos referenciales para la implementación de los mismos, esta investigación servirá como un referente económico para la implementación de los prototipos.

En la fase de diseño de los prototipos se realizó previamente una comparación entre diferentes marcas de dispositivos *Bluetooth* y *Wi-Fi* disponibles en el mercado Nacional e Internacional para de esta manera conocer sus características técnicas y costos de cada uno de ellos.

2.3.1 COSTO REFERENCIAL DE EQUIPAMIENTO DEL PROTOTIPO CON TECNOLOGÍA *BLUETOOTH*

El interfaz seleccionado es el *DBT-122* (*Bluetooth* 1.2 Adaptador USB para PC) de marca *D-Link*, estos son equipos que se pueden encontrar con facilidad en el mercado nacional y en el exterior. Por lo tanto se puede obtener un amplio mercado de equipos de esta marca suficiente para tener buenas ofertas de diferentes proveedores. En este caso los precios son una medida media de los productos que existen en el mercado.

Se consideran la cantidad de clientes inalámbricos de acuerdo a lo mencionado en el diseño de los prototipos inalámbricos en el cual se consideran 2 tarjetas inalámbricas para el prototipo *Bluetooth*. A continuación se presenta en la tabla 2.10 los precios de estos productos, cabe destacar que estos precios incluyen IVA.

Equipo	Cantidad	Unidad (USD)	Precio(USD)
Tarjetas Inalámbricos USB Bluetooth	2	28.605	57.21
Subtotal			**57.21**

Tabla 2.10 Costos de equipamiento con tecnología *Bluetooth*

2.3.2 COSTO REFERENCIAL DE EQUIPAMIENTO DEL PROTOTIPO CON TECNOLOGÍA *WI-FI*

El interfaz seleccionado es el *DWL–G122* (Adaptador USB para PC) de marca *D-Link*, estos son equipos que se pueden encontrar con facilidad en el mercado nacional y en exterior. Por lo tanto se puede obtener un amplio mercado de equipos de esta marca suficiente para tener buenas ofertas de diferentes proveedores. En este caso los precios son una medida media de los productos que existen en el mercado.

Para la implementación del prototipo *Wi-Fi* es necesario utilizar dos interfaces con tecnología *Wi-Fi*, a continuación en la tabla 2.11 se presenta los precios de estos productos, cabe destacar que estos precios incluyen IVA

.

Equipo	Cantidad	Unidad (USD)	Precio(USD)
Tarjetas Inalámbricos USB Wi-Fi	2	35.615	71.23
Subtotal			**71.23**

Tabla 2.11 Costos de equipamiento con tecnología *Wi-Fi*

CAPÍTULO 3

3. SIMULACIÓN DE LOS PROTOTIPOS

Uno de los principales objetivos del proyecto, es la comparación de los resultados de la simulación de los prototipos *Bluetoot*h y *Wi-Fi* con la implementación de los mismos. Para la simulación de los prototipos se utilizó el simulador *ns-2, la versión que se utilizo es ns-2.29.3*

Antes de realizar la simulación es indispensable hacer una pequeña introducción del programa *ns-2*.

3.1 SIMULADOR ns-2 [21]

El *ns-2* es un simulador de eventos discretos creado para la investigación de redes telemáticas y esta disponible en múltiples plataformas. Probablemente *ns-2* es el simulador de redes gratuito más extendido tanto en investigación como para propósitos docentes.

Este simulador se empezó a desarrollar en 1989 como una variante del simulador *Real Network Simulator* y ha ido evolucionando substancialmente en los últimos años. En 1995, el desarrollo estaba bajo la supervisión del proyecto *VINT* (*Virtual InterNetwork Testbed*), finalmente su investigación acabó en manos de un grupo de investigadores y desarrolladores de la Universidad de California en *Berkeley*, el *LBL* (*Lawrence Berkeley Laboratory*), *XEROX Parc* y *USC/ISI* (*University of Southern California/ Information Sciences Institute*).

Entre los usos más habituales que posee *ns-2* se puede destacar los siguientes:

- Simular estructuras y protocolos de redes de todo tipo (satélite, *wireless*, cableadas, etc.)
- Desarrollar nuevos protocolos, algoritmos y comprobar su funcionamiento.

- Comparar distintos protocolos en cuanto a prestaciones.

- **UCBT - Extensión de Bluetooth para ns-2** [32]

UCBT (Extensión de Bluetooth para *ns-2* en la universidad de *Cincinnati*) es un módulo basado en *ns-2* para simular redes con tecnología *Bluetooth*. La mayoría de las especificaciones en la banda base y capas superiores como *LMP* (Protocolo de Administración del Enlace), *L2CAP* (Protocolo de Control y Adaptación de Enlace Lógico), *BNEP* (Protocolo de encapsulamiento de red bluetooth) se han simulado en *UCBT*, incluyendo el esquema de la utilización de frecuencia, descubrimiento del dispositivo, establecimiento de conexión, negociación del paquete de la multi-ranura, conexión de la voz utilizando *SCO* (Enlace sincrónico orientado a la conexión), etc.

UCBT no es el primer simulador utilizado por ns-2 para simular redes *Bluetooth.* Existen otros como *BlueHoc* creado por *IBM* y su extensión del *scatternet, Blueware*, ambos fueron desarrollados antes que el *UCBT*. Sin embargo, *UCBT* es el simulador más exacto, completo y actualizado de *Bluetooth*.

UCBT permite simular las versiones 1.1, 1.2 de *Bluetooth* cuya velocidad es menor a 1*Mbps*, también permite simular nuevas versiones como *bluetooth* 2.0 que incorpora la técnica "*Enhanced Data Rate*" (*EDR*) que permite mejorar la velocidad de transmisión hasta 2 o 3 Mbps. Una de las principales contribuciones del *UCBT* es que proporciona un marco flexible a la investigación de redes *scatternet*.

La única falencia que posee *UCBT*, es que no permite el posicionamiento ni la movilidad de los nodos, éste permite simular el funcionamiento de las capas de *Bluetooth,* esto es una desventaja con respecto a la librería *MAC/802.11* en el ns-2 ya que ésta, a más de simular la transmisión de datos permite dar movimiento a los nodos.

3.1.1 INSTALACIÓN DE LINUX

El *software ns-2* utilizado en la simulación se puede ejecutar en cualquier versión del sistema operativo *Linux*. Para el presente proyecto se utilizó la versión *Linux Centos-4i386*.

Es importante mencionar que para una correcta instalación del *ns-2* se requiere que *Linux* sea instalado con la opción *EVERYTHING* (total), de esta manera se copian las librerías necesarias para el correcto desempeño de *ns-2*, caso contrario, se necesitará un conocimiento avanzado acerca de la programación en *Linux*, para obtener dichas librerías y adjuntarlas en el entorno de trabajo.

Con la opción de instalación *EVERYTHING*, es conveniente disponer de características mínimas presentes en una PC, como por ejemplo 5 *Gbytes* de espacio libre en disco, memoria *RAM* de 256 *Mbytes* y un procesador *Pentium III*.

3.1.2 INSTALACIÓN DEL SIMULADOR Y LIBRERÍA UCBT

Para instalar el simulador *ns-2* conjuntamente con la librería *UCBT*, es necesario obtener el paquete "todo en uno del *ns-2*" que contiene todas las librerías, también es necesario obtener la librería *UCBT*. La versión del *ns-2* utilizada en el proyecto es ***ns-2.29.3***, para obtener este paquete hay que dirigirse a la siguiente página de *Internet* http://www.isi.edu/nsnam/. La versión del *UCBT* utilizada es ***UCBT 0.9.9.2*** para obtener esta librería hay que dirigirse a la siguiente página de *Internet* http://www.ececs.uc.edu/~cdmc/ucbt/.

Una vez obtenidos los paquetes *ns-2* y *UCBT*, lo primero que se debe hacer es descomprimir el *ns–2* en el directorio que se desee instalar, en este caso se escogió el directorio raíz dentro de la carpeta simulador de la siguiente forma:

```
#mkdir / simulador
#cd / simulador
# tar zxvf ns-allinone-2.29.3.tar.gz
```

Luego de descomprimir el paquete se crea el directorio *ns-allinone-2.29,* aquí se encuentra el directorio *ns-2.29*, dentro de éste se debe descomprimir la librería *UCBT* de la siguiente forma:

```
#cd / simulador
#ls
ns-allinone-2.29
#cd / simulador  / ns-allinone-2.29  / ns-2.29
# tar zxvf ucbt - 0.9.9.2 .gz
```

Una vez descomprimido el *UCBT* se debe ingresar a este directorio y ejecutar el siguiente comando:

```
#cd / simulador / ns-allinone-2.29 / ns-2.29 /ucbt – 0.9.9.2
./install-bt
```

Para utilizar las herramientas del simulador *ns-2* es necesario agregar un *PATH* permanente en el archivo */etc/profile* de la siguiente forma:

```
export PATH=”$PATH:/ simulador/ns-allinone-2.29/bin:/simulador/ns-allinone-2.29/tcl8.4.11/unix:/simulador/ns-allinone-2.29/tk8.4.11/unix
```

Además se debe realizar los siguientes enlaces simbólicos, a través de estos enlaces se crean puentes hacia los directorios de origen, permitiendo ejecutar estos programas desde cualquier parte:

```
cd /usr/bin
ln -s  /simulador/ns-allinone-2.29./ns-2.29/ns ns
ln –s  /simulador/ns-allinone-2.29/nam-1.11/nam nam
ln –s  /simulador/ns-allinone-2.29/xgraph-12.1/xgraph xgraph
```

Una vez instalado el simulador *ns-2* con la librería *UCBT*, éste debe ser validado de la siguiente forma:

```
./validate
```

Este proceso inicia una serie de pruebas propias de *ns-2* que toman varios minutos y que comprueban la correcta instalación del *software*. Ya validado el *ns-2*, se puede iniciar la implementación de las simulaciones.

3.2 SIMULACIÓN DE LOS PROTOTIPOS

Las simulaciones de los prototipos se basan en una red *Ad-hoc* punto a punto de corto alcance con tecnología inalámbrica. Los resultados se enfocan a determinar la velocidad efectiva, la relación señal a ruido y niveles de potencia en función de la distancia de cada uno de los prototipos.

La versión utilizada del ns-2 es *ns 2.29.3* se puede destacar que el uso de este simulador no ha sido sencillo debido a la dificultad de manejo, respecto a instalación, simulación de redes *wireless* e incompatibilidad de módulos y versiones.

Concretamente la simulación de redes *wireless* en *ns-2* es complicada debido a que existe poca documentación sobre el tema.

3.2.1 ESCENARIOS

Las simulaciones se han dividido en 2 escenarios diferentes:

- El primer escenario consta de dos nodos inalámbricos con tecnología *Bluetooth* y un enlace punto a punto unidireccional (figura. 3.1); en este escenario se realizarán diversas pruebas en relación a la capacidad del enlace como son: velocidad efectiva, niveles de potencia y la relación señal a ruido.

- El segundo escenario consta de dos nodos inalámbricos con tecnología *Wi-Fi* y un enlace punto a punto unidireccional (figura 3.8); en este escenario se realizarán diversas pruebas en relación a la capacidad del enlace como son: velocidad efectiva, niveles de potencia y la relación señal a ruido.

3.2.1.1 Escenario Bluetooth

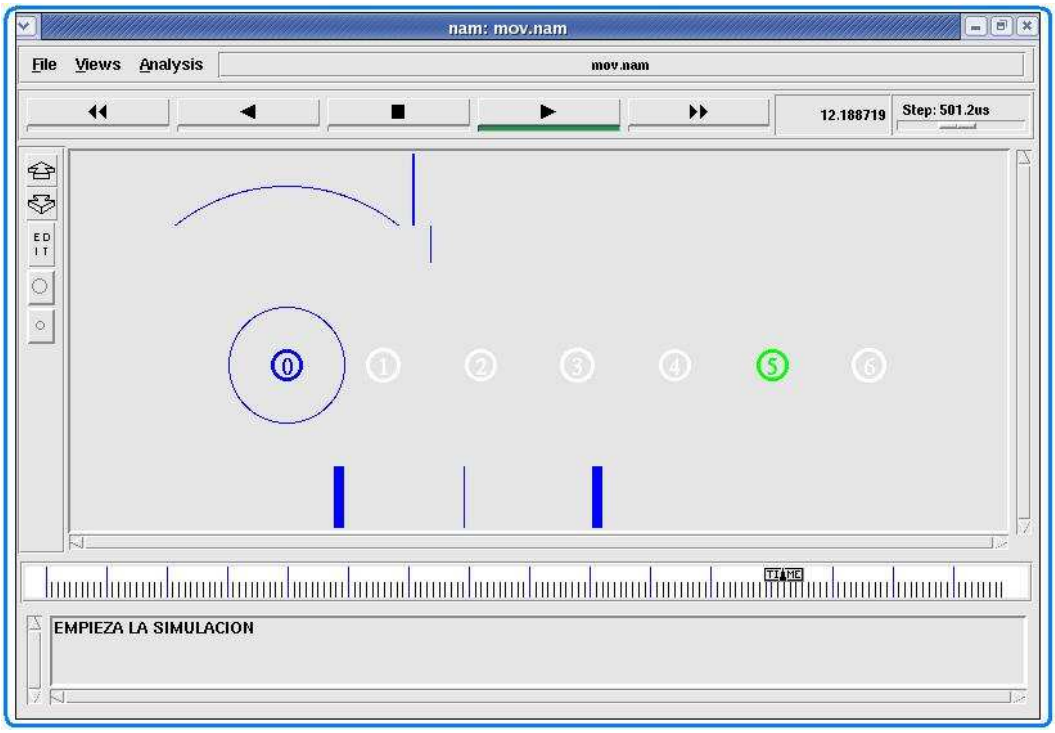


Figura 3.1 Escenario *Bluetooth*

3.2.1.1.1 Simulación Bluetooth

Para tomar las mediciones de campo cada dos metros en la simulación, se creó siete nodos inalámbricos con tecnología *Bluetooth,* la comunicación entre estos nodos es unidireccional (enlace *simplex*).

El motivo de crear siete nodos es emular el movimiento de estos, los nodos se van activando cada dos segundos, dando así un efecto de movilidad. Todo esto se realizó debido a que la librería *UCBT* no permite la movilidad y posicionamiento de los nodos. Para posicionar los nodos cada dos metros se modificó el archivo *wnode.cc.*

En la simulación se puede verificar la variación de la velocidad efectiva, la potencia y la relación señal a ruido en función de la distancia. El nodo *cero* recibirá paquetes *TCP* con tráfico *FTP* del nodo *uno*, luego del nodo *dos* y así respectivamente hasta el nodo *seis*.

El *script* que se utilizó en la simulación es fruto de varios *scripts* de prueba que se han ido utilizando a lo largo de todo el proyecto. Este código se ha modificado para adecuarlo a las necesidades de este proyecto.

A continuación se muestra el *script* utilizado para la simulación, en el cual se realiza un comentario de los aspectos más importantes.

- **SCRIPT BLUETOOTH**

Los siguientes comandos permiten almacenar en la variable *v*, la velocidad con la cuál se quiere realizar la simulación.

```
=========================================================================
#Las siguientes líneas permiten mostrar una ayuda al usuario en el caso
de que no sepa como utilizar el script
=========================================================================

if {$argc != 1} {
        puts ""
        puts "ERROR"
        puts ""
        puts "UTILICE EL SCRIPT DE LA SIGUIENTE FORMA"
        puts ""
        puts "ns bluetooth1.tcl velocidad "
        puts ""
        puts "VELOCIDAD:DH1,DH3,DH5,DM1,DM3,DM5"
        puts ""

        exit 1

} else {
        set v [lindex $argv 0]

}
```

Definición de las variables que se utilizan en la simulación como: la capa *MAC* que en este caso es *BNEP* para *Bluetooth* y el número de nodos que intervienen en la simulación.

```
=========================================================================
# Declaración de Variables a Utilizar para la presente Simulación
=========================================================================

set val(mac) Mac/BNEP    ;#Variable que identifica el tipo de nodo (BT)
set val(nn)    7         ;#Variable que especifica el número de nodos BT
```

Se crea una instancia a la clase simulador, para que se pueda realizar la simulación.

```
#=====================================================================
#Declaraciones Típicas de NS2 y NAM
#=====================================================================

set ns  [new Simulator]    ;#Definición de ns como nueva instancia de NS2
```

Se configura el nodo con el tipo de MAC definido anteriormente

```
$ns node-config -macType $val(mac) ;#Definición de nodos a usarse q sean
                                     BT
```

Se crean en modo escritura los ficheros que se van a utilizar:

- **mov.tr:** archivo de texto donde se almacenan las trazas generadas en la simulación. Se puede observar la evolución del envío de cada trama. Con este archivo y el archivo *efectiva.pl* se logra generar otro archivo que mediante el *xgraph* permite visualizar como varia la velocidad en función de la distancia.

- **mov.nam:** archivo de texto donde se almacenan las trazas *nam*, que permiten visualizar el escenario de simulación.

- **potencia.tr:** archivo de texto que se va a generar en la simulación y que mediante el *xgraph* permiten visualizar como varía la potencia en función de la distancia.

- **señalruido.tr:** archivo de texto que se va a generar en la simulación y que mediante el *xgraph* permite visualizar como varía la relación señal a ruido en función de la distancia.

```
set tracefile [open mov.tr w]       ;#Definición y creación de un archivo
                                    de trazas

set namfile   [open mov.nam w]      ;#Definición y creación de un archivo
                                    NAM (ambiente grafico)
set pot       [open ./potencia.tr w]
set SN        [open ./señalruido.tr w]
```

Estos comandos permiten generar las trazas adecuadas tanto en el archivo *mov.tr* y en el archivo *mov.nam* y habilitar su uso en el *nam*.

```
$ns trace-all $tracefile                    ;#Declaración del formato del
                                              archivo de trazas

$ns namtrace-all-wireless $namfile 7 7      ;#Declaración del tipo de
                                              ambiente y sus dimensiones

Simulator set MacTrace_ ON                  ;#Comando de habilitación para
                                              uso del NAM

Simulator set RouterTrace_ ON               ;#Comando de habilitación para
                                              el uso wireless en el NAM


$ns color 0 blue
```

Las siguientes líneas permiten crear los siete nodos, asignarles el protocolo de *routing AODV,* se utilizó este protocolo por ser uno de los más utilizados en este tipo de simulaciones; se activan los nodos a los cero segundos, se establece el tamaño de la cola en 30 paquetes para *L2CAP* y por último se asignan valores típicos de *INQUIRY* y *PAGE* a todos los nodos excepto al nodo 0.

El protocolo de enrutamiento *AODV* es un protocolo que no requiere que los nodos mantengan las rutas de los destinos que no tienen una conexión activa. Este protocolo usa números de secuencia de destino para cada ruta de entrada, el número de secuencia del destino es creado por el destino y este es enviado por cualquier ruta de información. Los números de secuencia nos permiten determinar cuál de las diversas rutas es la más accesible.

```
# ======================================================================
# Proceso de creación de Nodos Bluetooth
# ======================================================================

for {set i 0} {$i < $val(nn) } {incr i} {
    set node($i) [$ns node $i ]
  $node($i) rt AODV                 ;#se asigna el agente de ruteo AODV
  $ns at 0.0 "$node($i) on"         ;#se encienden todos los nodos al
                                      mismo tiempo en 0.0seg

  [$node($i) set l2cap_] set ifq_limit_ 30    ;#se establece el tamaño de
                                               la cola para la capa L2CAP

   if {$i > 0} {
      $node($i) inqscan 4096 2048    ;#se asignan los valores típicos de
                                      inquiry
```

```
      $node($i) pagescan 4096 2048  ;#se asignan los valores típicos de
                                      page

    }
}
```

Estas líneas asignan al nodo 0 el tipo de modelo inalámbrico adecuado para *Bluetooth,* la visualización de los paquetes *MAC* y se configura al Protocolo de Administración de Enlace para que el nodo cero realice una sola vez el *INQUIRY*

```
$node(0) LossMod BlueHoc         ;#se asigna el tipo de modelo
                                   Inalámbrico con o sin pérdidas

$node(0) trace-all-in-air on    ;#se asigna la visualización de tipos de
                                   paquetes MAC

[$node(0) set lmp_] set scan_after_inq_ 0    ;#se configura al LMP con
                                               comandos para q solo
                                               realice una vez inquiry
```

El procedimiento de configuración del enlace, tráfico y aplicaciones se lo hace para los nodos 1, 2, 3, 4, 5, 6 con el nodo cero. A continuación se explica el procedimiento de configuración del enlace entre el nodo “0” con el nodo “1”

Se crea un agente tipo *TCP* para el nodo 1, este nodo será el encargado de generar tráfico tcp y enviarlo al destino nodo 0.

```
=========================================================================
# Configuración de enlaces, trafico y aplicaciones                *
=========================================================================

set tcp0 [new Agent/TCP]      ;#Declaración de un agente de trafico TCP

$ns attach-agent $node(1) $tcp0    #Unión del agente con el nodo
                                     correspondiente (tx)
```

Comandos que permiten generar tráfico *FTP* sobre la conexión *TCP* que ya fue creada.

```
set ftp0 [new Application/FTP]  ;#Declaración de una aplicación soportada
                                   por el agente de trafico TCP

$ftp0 attach-agent $tcp0      ;#unión de la aplicación al agente de trafico
```

Se define la conducta del nodo destino y se le asigna a la fuente llamada *sink*. Este nodo destino es el encargado de generar *acks* (acuses de recibo) que garantizan el arribo de todos los paquetes al nodo 0.

```
set null0 [new Agent/TCPSink]   ;#Declaración del repositorio del agente
                                  de trafico TCP

$ns attach-agent $node(0) $null0  ;#Unión del repositorio con el nodo
                                   correspondiente (rx)
```

Se realiza la conexión entre el nodo 0 y el nodo 1.

```
$ns connect $tcp0 $null0     #unión del agente de trafico con el
                              repositorio
```

Estos comandos establecen un tamaño de 20 paquetes en el buffer esto indica que si el limite de paquetes es sobrepasado los paquetes serán descartados

```
set ifq [new Queue/DropTail]            ;#Declaración de la cola o Buffer
$ifq set limit_ 20                      ;#Límite de la cola (paquetes)
```

Variables creadas para iniciar la transferencia de tráfico *FTP* que son usadas más adelante para generar un efecto de movilidad

```
# Variables creadas para iniciar y terminar la transferencia de paquetes
de las aplicaciones

set nscmd0 "$ftp0 start"
set nscmd1 "$ftp1 start"
set nscmd2 "$ftp2 start"
set nscmd3 "$ftp3 start"
set nscmd4 "$ftp4 start"
set nscmd5 "$ftp5 start"
```

Variables creadas para finalizar la transferencia de tráfico *FTP* que son usadas más adelante para generar un efecto de movilidad.

```
set nscmd00 "$ftp0 stop"
set nscmd01 "$ftp1 stop"
set nscmd02 "$ftp2 stop"
set nscmd03 "$ftp3 stop"
set nscmd04 "$ftp4 stop"
```

Esta línea permite al tiempo 0.000001 visualizar en el *nam* el mensaje “EMPIEZA LA SIMULACIÓN”

```
#=========================================================================
# Organizador de Eventos                                                 *
#=========================================================================

$ns at 0.000001 "$ns trace-annotate \" EMPIEZA LA SIMULACION \""
```

Estas líneas permiten en un tiempo dado iniciar la conexión entre el nodo 0 y los otros nodos indicando el tipo de paquetes que envía y paquetes que recibe, estos paquetes serán ingresados por el usuario que pueden ser: *DH5*, *DH3*, *DH1*, *DM5*, *DM3* y *DM1*, las velocidades que se alcanzan cuando se transmiten estos paquetes se muestran en la tabla 1.4; también se establece el tamaño de la cola en el *buffer* que fue definido anteriormente. El tiempo en que se establece la conexión de todos los nodos depende del número de nodos que van a establecer una conexión, para este caso es de 4 segundos.

```
$ns at 0.1 "$node(0) make-bnep-connection $node(1) $v $v  noqos $ifq"
$ns at 0.2 "$node(0) make-bnep-connection $node(2) $v $v noqos $ifq1"
$ns at 0.3 "$node(0) make-bnep-connection $node(3) $v $v noqos $ifq2"
$ns at 0.4 "$node(0) make-bnep-connection $node(4) $v $v noqos $ifq3"
$ns at 0.5 "$node(0) make-bnep-connection $node(5) $v $v noqos $ifq4"
```

Esta línea de comando permite determinar el alcance máximo del simulador para bluetooth en el caso de que se la active no se podrá realizar la conexión entre los nodos y por ende no se inicia la transmisión *FTP*.

```
#$ns at 0.6 "$node(0) make-bnep-connection $node(6) DH5 DH5 noqos $ifq5"
```

Estas líneas permiten al simulador en un tiempo dado iniciar y terminar la transferencia de datos dando un efecto de movilidad entre el nodo cero y los demás.

```
$ns at 4.0 "$nscmd0"
$ns at 6.0 "$nscmd00"

$ns at 6.1 "$nscmd1"
$ns at 8.0 "$nscmd01"

$ns at 8.1 "$nscmd2"
$ns at 10.0 "$nscmd02"

$ns at 10.1 "$nscmd3"
$ns at 12.0 "$nscmd03"

$ns at 12.1 "$nscmd4"
$ns at 13.5 "$nscmd04"
```

Al tiempo t=4 segundos se llama a la función *record,* esta función es la encargada del cálculo del enlace.

```
=========================================================================
# Procedimiento para llamar a la función record la cual se encargara del
cálculo  de la potencia y de la relación señal a ruido
=========================================================================

#A los 4.0 segundos llamo a la función record
$ns at 4.0 "record"
```

Se define dos variables locales para la función *record* la una llamada *pot* y la otra llamada *SN*

```
proc record {} {
        global sink pot
        global sink SN
```

Indica la granularidad de 2 segundos y se almacena en la variable *time*

```
        set ns [Simulator instance]
        set time 2.0
```

Este comando permite determinar en que tiempo se encuentra la simulación

```
#Calculo de la distancia
        set now [$ns now]
```

Se calcula la distancia cada 2 segundos debido a que es el tiempo en el cual el nodo va a desplazarse de una posición a otra.

```
        set distancia [expr $now*1.0 - 2.0 ]
```

Se realiza el cálculo de las pérdidas en la trayectoria y el cálculo de la potencia para *Bluetooth* de acuerdo a las siguientes ecuaciones: ecuación 1.8 y ecuación 1.10. Para el cálculo de la potencia de transmisión se toma el valor de 4dBm que es la especificada para la interfaz *DBT -122.*

```
#Comparo la distancia de acuerdo a una referencia para el cálculo de la
potencia.
        if {$distancia <= 8.5} {

#Calculo de las pérdidas en dB para una distancia menor a 8.5 m

                set pérdidas [ expr 40.0 + 20.0*log10($distancia)]

#Cálculo de la potencia
           set potencia [expr 4-pérdidas-8.0]
```

Con la potencia calculada para una distancia menor a 8.5m se calcula la relación señal a ruido. El ruido se cálculo en base a las ecuaciones que se encuentran en el ANEXO F:

Para una temperatura de 27℃ y un ancho de banda de 1 *MHz* correspondiente a *Bluetooth* y transformando este valor a decibelios se obtuvo un ruido de -154.28 dB.

```
set sn [expr $potencia+154.28]
```

Se realiza el cálculo de las pérdidas en la trayectoria y el cálculo de la potencia para *Bluetooth* de acuerdo a las siguientes ecuaciones: ecuación 1.9 y ecuación 1.10. Para el cálculo de la potencia de transmisión se toma el valor de 4dBm que es la especificada para la interfaz *DBT -122.*

```
} else {
        set perdidas [ expr 25.3+36*log10($distancia)]
        set potencia [expr 4-$perdidas-8.0]
```

Con la potencia calculada para una distancia mayor a 8.5m se prosigue a calcular la relación señal a ruido como se hizo anteriormente.

```
        set sn [expr $potencia+154.28]
        }
```

Se imprime en el archivo *potencia.tr* la distancia y la potencia. En el archivo *señallruido.tr* se imprime la distancia y la relación señal a ruido.

```
puts $pot "$distancia  $potencia    "
puts $SN "$distancia  $sn "
```

Se llama a la función *record* cada dos segundos, para que realice el cálculo de los parámetros requeridos, esto se debe ya que cada dos segundos se activa el siguiente nodo que se encuentra en una posición diferente.

```
$ns at [expr $now+$time] "record"
}
```

Se indica el tiempo en el cual la simulación finaliza.

```
$ns at 15.9 "finish"
```

Se llama a la función *finish* que permite realizar todos los procesos para finalizar la simulación y permite visualizar con el *xgraph* los resultados obtenidos en la misma.

Para el cálculo de la velocidad efectiva se ejecuta el comando *perl* que conjuntamente con el archivo *efectiva.pl*, con el archivo *mov.tr*, indicando el nodo donde se necesita analizar el resultado y la granularidad generan el archivo *velo.tr* en el cual se almacena la velocidad efectiva de *Bluetooth* en función de la distancia. El archivo *efectiva.pl* se encuentra en el ANEXO I

```
#=========================================================================
# Procedimiento Final                                                    *
#=========================================================================

proc finish {} {
        global node
        $node(0) print-all-stat

        exec nam mov.nam &

        exec perl efectiva.pl mov.tr _0_ 0.1 > velo.tr & \

        exec xgraph velo.tr -t "VELOCIDAD BLUETOOTH vs DISTANCIA" -x
                                "DISTANCIA m" -y "VELOCIDAD bps" &
      exec xgraph potencia.tr -geometry "750x500" -P -t
" POTENCIA BLUETOOTH vs DISTANCIA" -x "DISTANCIA m" -y "POTENCIA dBm" &

        exec xgraph señalruido.tr -geometry "750x500" -P -t "S/N
BLUETOOTH vs DISTANCIA" -x "DISTANCIA m" -y "S/N dBm" &

        exit 0
}

$ns run
```

3.2.1.1.2 Ejemplo de Simulación del Prototipo Bluetooth

Para realizar la simulación el usuario debe ingresar al directorio en el cual se encuentran ubicados los s*cripts bluetooth.tcl* y *efectiva.pl* en este caso los *scripts* se encuentran en el directorio *Bluetooth*

Para acceder al directorio el usuario debe ingresar los siguientes comandos en el terminal de *Linux.*

cd /root/Desktop/tesis/Bluetooth/

Para obtener ayuda del uso del *script* ingresar el siguiente comando.

ns bluetooth.tcl

Después de ejecutar el comando *ns bluetooth.tcl* se despliega la siguiente pantalla, la cual indica la forma de utilizar el *script* para iniciar la simulación.

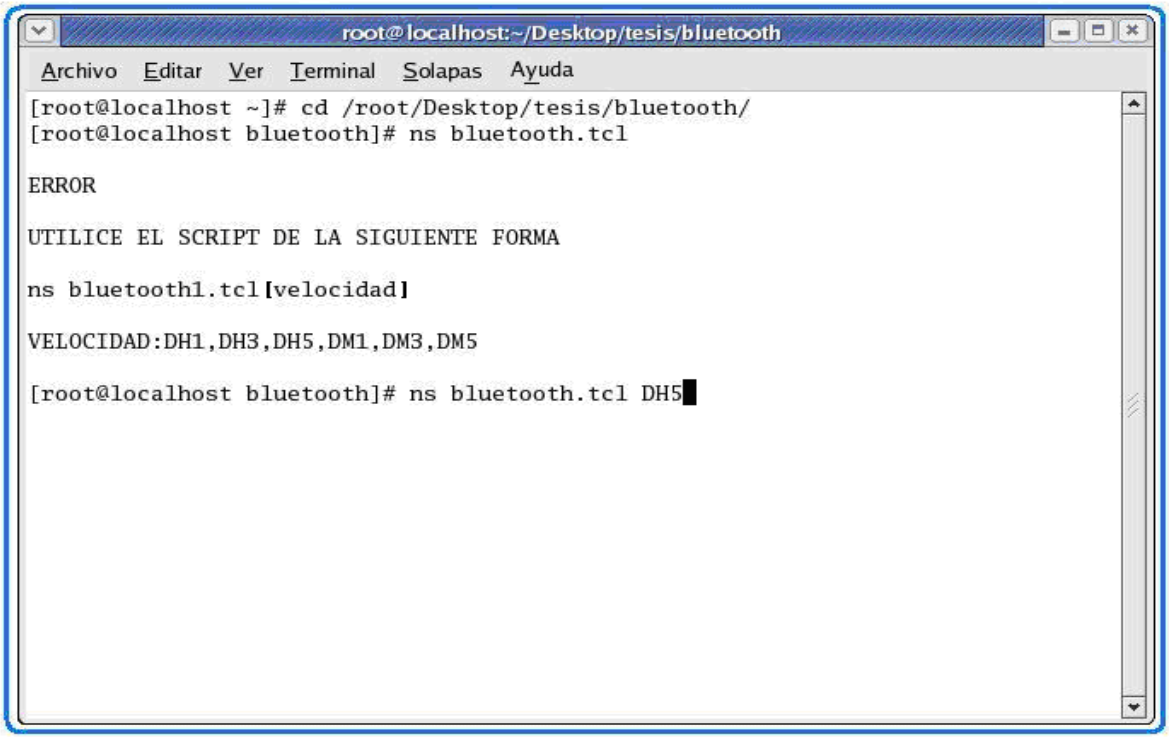


Figura 3.2 Ayuda para la simulación *Bluetooth*

Aquí el usuario puede escoger la velocidad para realizar la simulación. En la figura 3.2 la simulación se realizó con una velocidad *DH5*.

DH5 indica que se utiliza para la transmisión paquetes *DH5* que se transmiten a una velocidad de 721 Kbps, los paquetes y sus velocidades se los pueden visualizar en la tabla 1.4.

Una vez que se ejecute el programa se visualizará el escenario en el *nam* y los resultados que se obtienen de la simulación en el *xgraph* como son: potencia, relación señal a ruido y la velocidad efectiva.

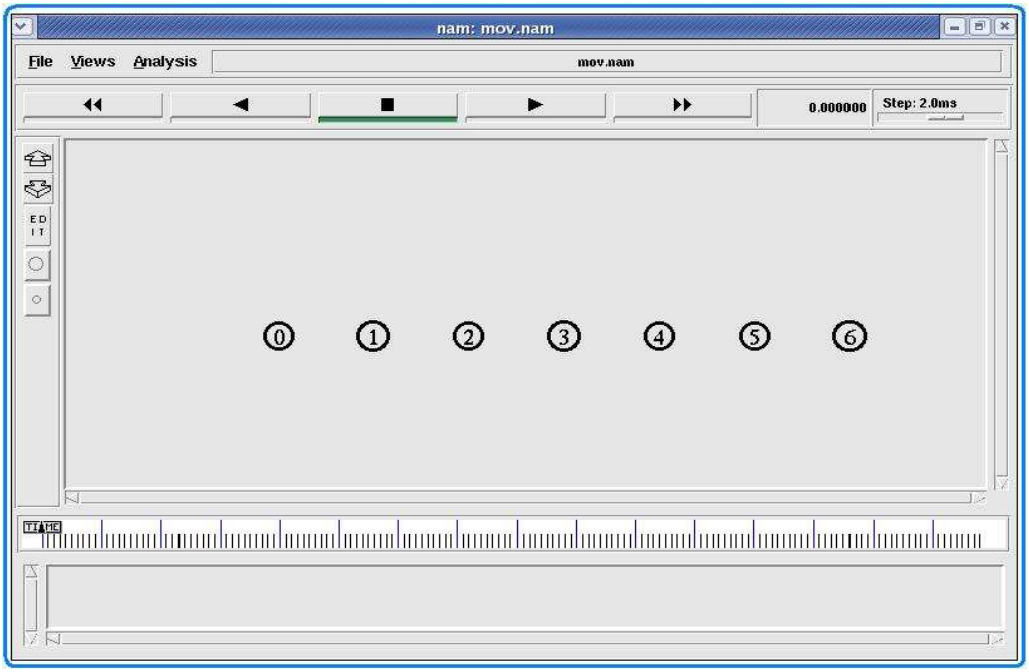


Figura 3.3 Pantalla inicial del nam *Bluetooth*

Luego de iniciar la simulación en el *nam* se visualiza como los paquetes son enviados.

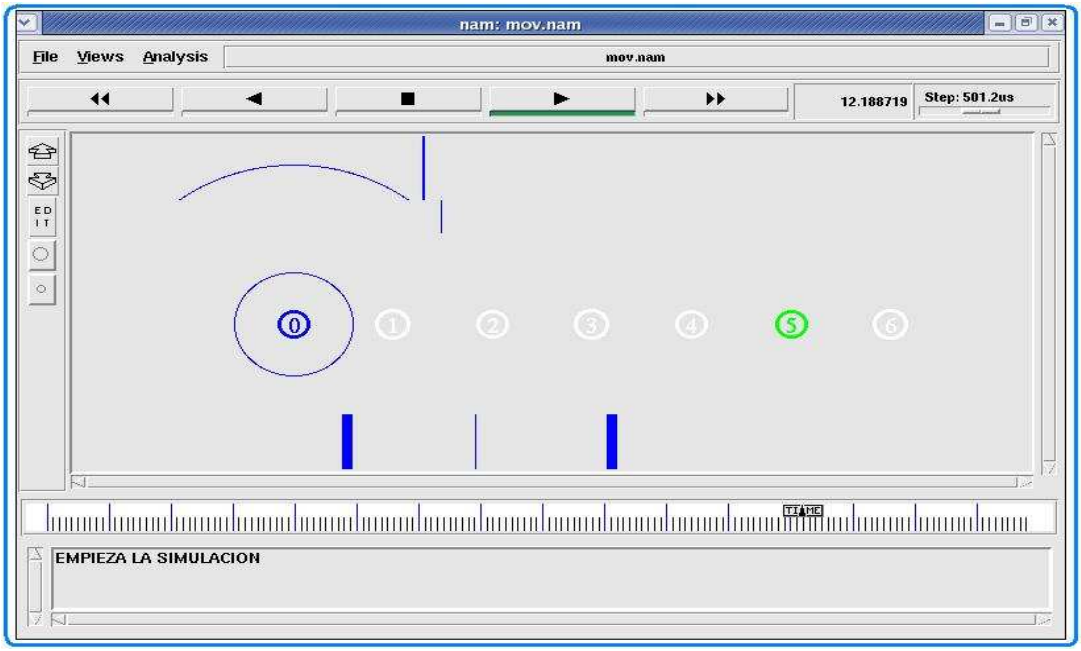


Figura 3.4 Simulación *Bluetooth* en el *nam*

Con la ayuda del *XGraph* y el archivo *potencia.tr* que se genera en la simulación, visualizamos como la potencia cambia en función de la distancia.

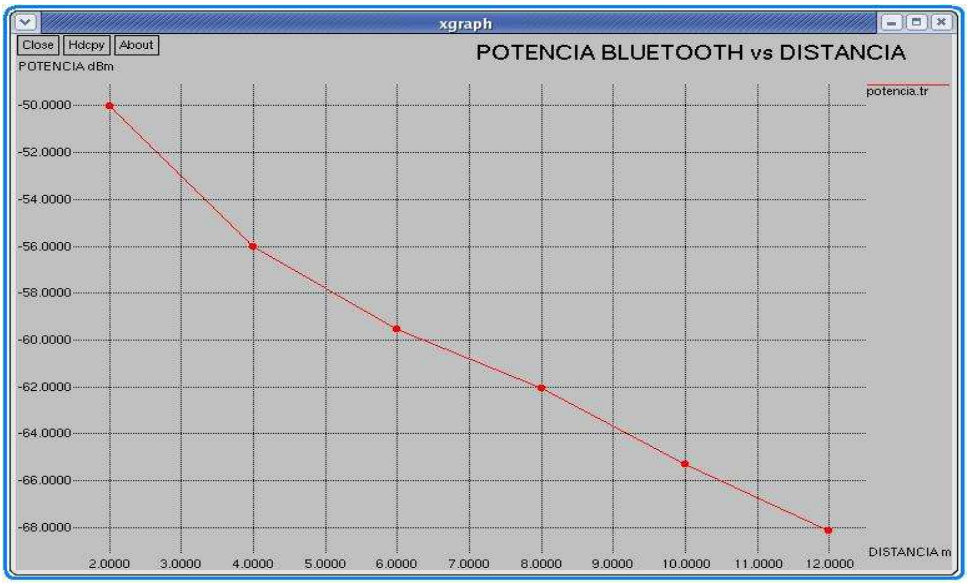


Figura 3.5 Potencia *Bluetooth* de la Simulación

Con la ayuda del *XGraph* y el archivo *señalruido.tr* generado en la simulación, se visualiza la relación señal a ruido en función de la distancia.

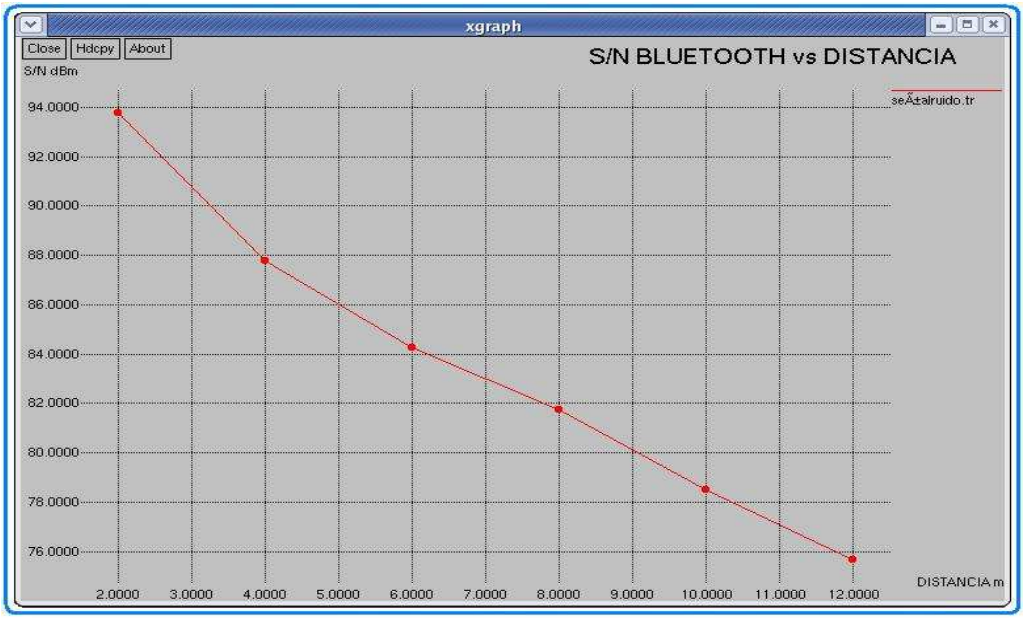


Figura 3.6 Señal a ruido *Bluetooth* de la Simulación

El archivo *efectiva.pl* es un programa que permite procesar el archivo *blue2.tr* generado en la simulación. Este programa permite crear un nuevo archivo con la velocidad efectiva en el nodo que recibe los datos.

Con la ayuda del xg*raph* y el nuevo archivo creado se genera la siguiente gráfica de la velocidad efectiva para *Bluetooth.*

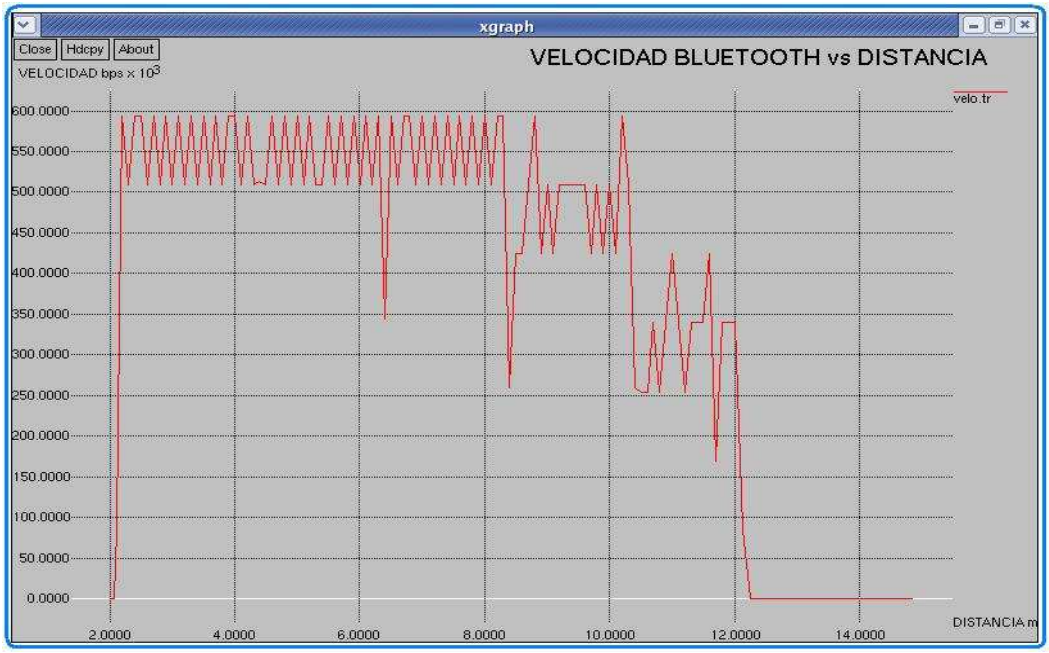


Figura 3.7 Velocidad *Bluetooth* de la Simulación

3.2.1.2 Escenario Wi-Fi

El segundo escenario consta de dos nodos inalámbricos con tecnología *Wi-Fi* (*802.11b*) y un enlace punto a punto unidireccional figura. 3.8. En este escenario se realizarán diversas pruebas en relación a la capacidad del enlace como son: velocidad efectiva, y niveles de potencia.

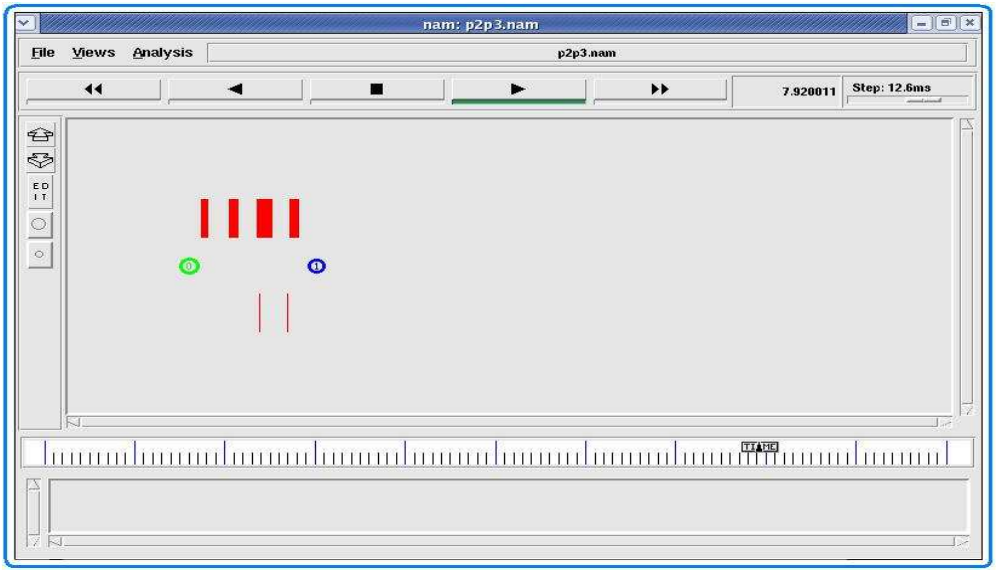


Figura 3.8 Escenario Wi-Fi

3.2.1.2.1 Simulación Wi-Fi

La simulación consiste en crear dos nodos inalámbricos *Wi-Fi* los mismos que se comunican con un enlace *simplex* e ir verificando como varia el *throughput*, la potencia, y la relación señal a ruido en función de la distancia. El nodo uno emitirá paquetes *TCP* con tráfico *FTP* al nodo dos mientras este se va alejando hasta una distancia máxima de 12m. Con la simulación se obtendrá un gráfico de todos los parámetros a analizar en función de la distancia.

El *script* que se ha utilizado en la simulación es fruto de varios *scripts* de prueba que se han ido utilizando a lo largo de todo el proyecto. El código base se ha reutilizado de los ejemplos que vienen con el *ns-2* y que a su vez también se utilizan en el tutorial de Marc Greis. Este código se ha modificado para adecuarlo a las necesidades de este proyecto.

A continuación se muestra el *script* utilizado para la simulación en el cual se comentaran los aspectos más importantes.

- **SCRIPT WI-FI**

Estas líneas permite almacenar en la variable *x*, la distancia y en la variable *n* el coeficiente de pérdidas para *Wi-Fi* con los cuales se quiere realizar la simulación

```
# ======================================================================
#Las siguientes líneas permiten mostrar una ayuda al usuario en el caso
de que no sepa como utilizar el script
# ======================================================================

if {$argc != 2} {

        puts ""
        puts "ERROR"
        puts ""
        puts "UTILICE EL SCRIPT DE LA SIGUIENTE FORMA"
        puts ""
        puts "ns wifi.tcl distancia n"
        puts ""
        puts "n: COEFICIENTE DE PERDIDAS "
        puts ""
        puts "n=1.4 Corredor,n=2 Grandes,n=1.9 Abierto,n=4 Oficinas,n=5.2
             Atraviesa una pared, n=5.4 Atraviesa mas de una pared"
        puts ""
```

```
        puts "DISTANCIAS: En metros"
        puts ""

        exit 1

} else {
  set x [lindex $argv 0]
  set n [lindex $argv 1]
}
```

Las siguientes líneas permiten imprimir en el *terminal* la distancia y el exponente de pérdidas.

```
puts "distancia= $x"
puts "n exponente de perdidas = $n"
```

Las siguientes líneas permiten realizar el cálculo de la potencia de recepción para la distancia ingresada e imprimir la misma en el *terminal*. Más adelante se indica como se realiza el cálculo de la potencia de recepción.

```
 set px [ expr 10*$n*log10($x)]

#Cálculo de las pérdidas totales

        set pT [expr 40.1+$px]

#Cálculo de la potencia
# La potencia para la  tarjeta DWL-G122 es de 14dbm +/- 2dB
        set P [expr 14.0 -$pT]

puts "POTENCIA = $P dbm"
```

Estas líneas permiten poner la velocidad con las que trabaja el interfaz *DWL-G122* en base a los niveles de sensibilidad especificados por la interfaz.

```
if {$P > -78} {
set velocidad 11M
puts "VELOCIDAD = 11 Mbps"

} else {

        if {$P > -82} {

        set velocidad 5.5M
        puts "VELOCIDAD = 5.5 Mbps"

        } else {

                if {$P > -85} {

                set velocidad 2M
                puts "VELOCIDAD = 2 Mbps"
```

```
            } else {

                    if {$P > -87} {

                    set velocidad 1M
                    puts "VELOCIDAD = 1 Mbps"

                    } else {
                            set velocidad 0M
                            puts " NO EXISTE CONEXION "
                            puts " NODOS FUERA DE COBERTURA "

                            }
                    }
            }

    }
```

Se define el canal inalámbrico, el modelo de propagación *Two Ray Ground*, el nivel físico y la *MAC* adecuados para trabajar con el *IEEE 802.11*. El *ns-2* no cambia la velocidad automáticamente a medida que los nodos se alejan es por ello que esto se trata de controlar con las primeras líneas de código.

El modelo de colas utilizado es el *DropTail*, que consiste en una cola simple *FIFO* en la que se descartan los paquetes que sobrepasen la capacidad del tamaño del *buffer* de la cola. El tamaño del *buffer* se ha delimitado a 41 paquetes. Además, al modelo *DropTail* se ha añadido la clase *PriQueue*, que significa que se está dando prioridad a los paquetes que se han enviado utilizando protocolos de enrutamiento.

El protocolo de enrutamiento utilizado es el *DSDV,* se utilizó este protocolo por ser uno de los más utilizados en este tipo de simulaciones. En este protocolo, los nodos vecinos van enviando mensajes de enrutamiento los unos a los otros. Se construye una tabla de enrutamiento en la que se actualizan los cambios. Si se da la situación de que llega un paquete del que no se conoce el destino, se envía a los nodos vecinos un mensaje de solicitud de ruta y se retiene el paquete hasta obtener una respuesta.

Otros parámetros que se definen son el tipo de antena (*omnidireccional*), el número de nodos inalámbricos de la simulación (2), las dimensiones del *nam* para poder posicionar a los nodos, el instante en el que finaliza la simulación (por

ejemplo, a los $x segundos). En este caso el tiempo de simulación coincide con la distancia ingresada.

```
# Define options
set val(chan)           Channel/WirelessChannel
set val(prop)           Propagation/TwoRayGround
set val(netif)          Phy/WirelessPhy
set val(mac)            Mac/802_11
set val(ifq)            Queue/DropTail/PriQueue
set val(ll)             LL
set val(ant)            Antenna/OmniAntenna
set val(ifqlen)         41
set val(nn)             2
set val(rp)             DSDV
set val(x)              50
set val(y)              10
set val(finish)         $x
```

La velocidad a la que se envían los datos por defecto en el *ns-2* es 2*Mbps*. Si se desea trabajar con las distintas velocidades del *IEEE 802.11b* es necesario modificar en el *script* la velocidad. A través de la variable *velocidad* se controla la velocidad a la que se envían los datos. Las líneas de código que empiezan por *Phy/WirelessPhy* o por *Mac/802_11* hacen referencia a variables del código fuente que están en los archivos *wireless-phy.cc* y *mac-802_11.cc.* Entonces, según el valor que se de a la variable velocidad, se acabará trabajando con una modulación o con otra.

Se ha cambiado el valor de la potencia de transmisión a 25 *mW* (14dBm) que es la potencia de transmisión del *DWL-G122*. El simulador utiliza por defecto 282 *mW* (24.5 dBm), que es un valor unos 10 dB superior a los valores especificados en las tarjetas más modernas y además está fuera del margen de niveles de potencia con los que se trabaja en Europa.

A pesar de que la mayoría de fabricantes de tarjetas tienen la funcionalidad de envío de tramas *RTS/CTS* desactivada, el *ns-2* por defecto la tiene activada. Esto ralentiza bastante el envío de información útil, sobretodo en escenarios sencillos como el que se ha planteado. Por este motivo se ha decidido prescindir del envío de este tipo de tramas fijando el parámetro *RTSThreshold_* a 3000 *bytes*. Esto significa que las tramas *RTS/CTS* se enviarán cuando se envíe una trama de

datos de más de 3000 *bytes*. Si por otro lado se fija la longitud de la trama a menos de 3000 *bytes,* este caso no se dará nunca.

Todos los paquetes se envían con un preámbulo, que es una cantidad de bits conocida que se envía al inicio de cada paquete. Esto permite que el receptor se sincronice y así esté listo para la recepción de los datos reales. El preámbulo se envía siempre a 1*Mbps* y puede ser de 144 bits o de 72 bits. El ns-2 utiliza por defecto el valor mayor, aunque en este escenario se ha fijado el valor del preámbulo al valor mínimo para así obtener mejores resultados.

```
Phy/WirelessPhy set bandwidth_ $velocidad
Phy/WirelessPhy set rate_ftp_ $velocidad
Phy/WirelessPhy set Pt_ 0.025

Mac/802_11 set dataRate_ $velocidad
Mac/802_11 set basicRate_ $velocidad
Mac/802_11 set RTSThreshold_ 3000
Mac/802_11 set PreambleLength_ 72
```

Se crea una instancia a la clase simulador, para que se pueda realizar la simulación.

```
set ns [new Simulator]
```

Se crean en modo escritura los ficheros que se utilizan:

- **p2p3.tr:** archivo de texto donde se almacenan las trazas generadas en la simulación. Se puede observar la evolución del envío de cada trama.

- **p2p3.nam:** archivo de texto donde se almacenan las trazas *nam* que nos van a permitir visualizar el escenario de simulación.

- **potenciaw.tr:** archivo de texto que se genera en la simulación y que mediante el *xgraph* permite visualizar como varía la potencia en función de la distancia.

- **señalruidow.tr:** archivo de texto que se genera en la simulación y que mediante el *xgraph* permite visualizar como varía la relación señal a ruido en función de la distancia.

```
set tracefd  [open p2p3.tr w]
set namtrace [open p2p3.nam w]
set pot      [open ./potenciaw.tr w]
set SN       [open ./señalruidow.tr w]
```

Tiene como parámetro el nombre de archivo donde las trazas deberían ir. Con este comando se trazan todos los eventos de acuerdo a un formato específico.

```
$ns trace-all $tracefd
$ns namtrace-all-wireless $namtrace $val(x) $val(y)
```

Se crea la topología en nam con los valores de ‘X’, ‘Y’ definidos anteriormente

```
#set up topology object
set topo [new Topography]
$topo load_flatgrid $val(x) $val(y)
create-god $val(nn)
# Create nn mobilenodes [$val(nn)] and attach them to the channel.
```

Se configuran los nodos con los valores de las variables definidos al inicio del *script*. Además, se activan todas las trazas salvo la *macTrace*. Estos datos se podrán visualizar en el archivo *p2p3.tr*

```
# configure the nodes

        $ns node-config -adhocRouting $val(rp) \
                        -llType $val(ll) \
                        -macType $val(mac) \
                        -ifqType $val(ifq) \
                        -ifqLen $val(ifqlen) \
                        -antType $val(ant) \
                        -propType $val(prop) \
                        -phyType $val(netif) \
                        -channelType $val(chan) \
                        -topoInstance $topo \
                        -agentTrace ON \
                        -routerTrace ON \
                        -macTrace OFF \
                        -movementTrace ON\
```

Se crean los nodos móviles y se les asigna a una variable.

```
for {set i 0} {$i < $val(nn)} {incr i} {
set node_($i) [$ns node]
}
```

Se posiciona el nodo cero en la posición (5, 5, 0) y el nodo 1 a un metro de separación del nodo 0 en la posición (6, 5, 0) en el *nam*.

```
# Provide initial location of mobilenodes
$node_(0) set X_ 5.0
$node_(0) set Y_ 5.0
$node_(0) set Z_ 0.0

$node_(1) set X_ 6.0
$node_(1) set Y_ 5.0
$node_(1) set Z_ 0.0
```

Comando que permite calcular la posición final en “x” del nodo 1

```
#Generation of movements
set posicion [expr 6 + $x]
```

Cuando transcurre 1 segundo, el nodo 1 empieza a desplazarse desde el punto donde está posicionado, que es a un metro de la ubicación del nodo 0 (posición 6.0, 5.0) a la posición x = $posicion, y = 5.0, a una velocidad de 1m/s.

```
$ns at 1.0 "$node_(1) setdest $posicion 5.0 1.0"
```

Se crea un agente tipo *TCP* para el nodo 0, este nodo será el encargado de generar tráfico *tcp* y enviarlo al destino. Se define la conducta del nodo destino y se le asigna a un puntero llamado *sink* este nodo destino es el encargado de generar *acks* que garantizan el arribo de todos los paquetes al nodo 1. Por último se realiza la conexión entre el nodo 0 y el nodo 1.

```
# Set a TCP connection between node_0 and node_(1)

set tcp [new Agent/TCP/Newreno]
$tcp set class_ 2
set sink [new Agent/TCPSink]
$ns attach-agent $node_(0) $tcp
$ns attach-agent $node_(1) $sink
$ns connect $tcp $sink
```

Las siguientes líneas generan tráfico *FTP* sobre una conexión *TCP ns-2* posee muchos parámetros por defecto que pueden ser cambiados. Como por ejemplo el tamaño del paquete *ftp* que en este caso es de 1024 bytes y la velocidad que ha sido cambiada al parámetro *velocidad*.

```
set ftp [new Application/FTP]
$ftp attach-agent $tcp
$ftp set packetSize 1024
```

```
$ftp set rate_ $velocidad
```

Al tiempo t = 2.0 segundos se empieza a generar tráfico *FTP*.

```
$ns at 2.0  "$ftp start"
```

Se define el tamaño del nodo en el *nam* en este caso de 1.0 para los dos nodos.

```
#Define node initial position in nam
for {set i 0} {$i < $val(nn)} {incr i} {
$ns initial_node_pos $node_($i) 1.0
}
```

A los dos segundos se llama a la función *record*

```
#Al 2.0 segundos se llama a la función record
$ns at 2.0  "record"
```

Se define tres variables locales para la función *record pot, SN* y *n.*

```
proc record {} {
        global sink pot
        global sink SN
        global sink n
```

Indica la granularidad de 0.25 segundos y se almacena en la variable *time*

```
    set ns_ [Simulator instance]
        set time 0.25
```

Indica el instante en el que se encuentra la simulación.

```
#Calculo de la distancia

        set now [$ns_ now]
```

Se calcula la distancia cada 0.25 segundos y se almacena en la variable distancia. Para el cálculo de la distancia se multiplica el tiempo por la velocidad de desplazamiento del nodo “1”

```
#Para el cálculo de la distancia multiplico el tiempo x la velocidad de
desplazamiento

        set distancia [expr $now*1.0 ]
```

Se realiza el cálculo de las pérdidas en la trayectoria y el cálculo de la potencia para *Wi-Fi* utilizando la ecuación 1.18. Para el cálculo de la potencia de transmisión se toma el valor de 14dBm que es la especificada para la interfaz *DWL-G122.*

El coeficiente de pérdidas *n* será ingresado por el usuario. Estos parámetros dependerán del lugar donde se realicen las pruebas.

```
#Calculo de las pérdidas en función de la distancia.

        set perdidas [ expr 10*$n*log10($distancia)]

#Calculo de las pérdidas totales

        set pérdidasT [expr 40.1+$pérdidas]

#Cálculo de la potencia
# La potencia para la  tarjeta DWL-G122 es de 14dbm +/- 2dB

        set potencia [expr 14.0 - $perdidasT]
```

La siguiente línea permite calcular la relación señal a ruido y almacenarla en la variable *sn.* El ruido se calculó en base a las ecuaciones que se encuentran en el ANEXO F.

Para una temperatura de 27℃ y un ancho de banda de 22 *MHz* correspondiente para *Wi-Fi* se obtuvo un ruido de -140.86 dB.

```
        set sn  [expr $potencia+140.86]
```

Se imprime en el archivo *potenciaw.tr* la distancia y la potencia. En el archivo *señallruidow.tr* se imprime la distancia y la relación señal a ruido.

```
#Imprimo en el archivo potecia.tr la distancia y la Prx en dBm

        puts $pot "$distancia    $potencia  "
        puts $SN  "$distancia    $sn "
```

Cada 0.25 segundos llamamos a la función *record* para que realice el cálculo de los parámetros requeridos.

```
        $ns_ at [expr $now+$time] "record"

}
```

Se comunica a los nodos cuando finaliza la simulación.

```
#telling nodes when the simulation ends

for {set i 0} { $i < $val(nn)} { incr i} {
```

```
        $ns at $val(finish) "$node_($i) reset"
}
```

Por último, se finaliza la simulación y se ejecuta el archivo *p2p3.nam*. El *script* acaba llamando a la instrucción *run*, que permite ejecutar todo el código.

```
#ending nam and the simulation
$ns at $val(finish) "$ns nam-end-wireless $val(finish)"
$ns at $val(finish) "finish"

set terminar [expr $x + 0.01]

$ns at $terminar   "puts \"end simulation\"; $ns halt"

proc finish {} {
        global node
        exec nam p2p3.nam &

}

$ns run
```

Las siguientes líneas son necesarias para visualizar automáticamente los resultados de la simulación en el *xgraph* como: potencia, velocidad efectiva, relación señal a ruido en función de la distancia.

Para el cálculo de la velocidad efectiva se ejecuta el comando *perl* que conjuntamente con el archivo *throughput.pl* y con el archivo *p2p3.tr*, indicando el nodo donde se necesita visualizar el tráfico y la granularidad, generan un archivo *thpwifi.tr* en el cual se almacena la velocidad efectiva de wi-fi en función de la distancia. El archivo *throughput.pl* se encuentra en el *ANEXO I*

```
#Comandos necesarios para visualizar resultados de la simulacion

  exec perl throughput1.pl p2p3.tr _1_ 0.1 > thpwifi.tr & \

  exec xgraph thpwifi.tr -t        "VELOCIDAD WI-FI VS DISTANCIA" -x
                                   "DISTANCIA m" -y "VELOCIDAD bps" &

  exec xgraph potenciaw.tr -t      "POTENCIA WI-FI VS DISTANCIA" -x
                                   "DISTANCIA m" -y "POTENCIA dBm" &


  exec xgraph señalruidow.tr -t    "S/N WI-FI VS DISTANCIA" -x "DISTANCIA
                                   m" -y "S/N dBm" &

exit 0
```

3.2.1.2.2 Ejemplo de Simulación del Prototipo Wi-Fi

Para realizar la simulación el usuario debe ingresar al directorio en el cual se encuentran los *scripts wifi.tcl* y *throughput1.pl* en este caso los *scripts* se encuentran en el directorio *wi-fi*

En este caso para acceder al directorio el usuario debe ingresar los siguientes comandos en el *terminal* de *Linux.*

cd/root/Desktop/BLUETOOTH1/definitivo/wi-fi/

Para obtener ayuda del uso del *script* ingresar el siguiente comando.

ns wifi.tcl

Después de ejecutar el comando ns *wifi.tcl* se despliega la siguiente pantalla, la cual indica la forma de utilizar el *script* para iniciar la simulación.

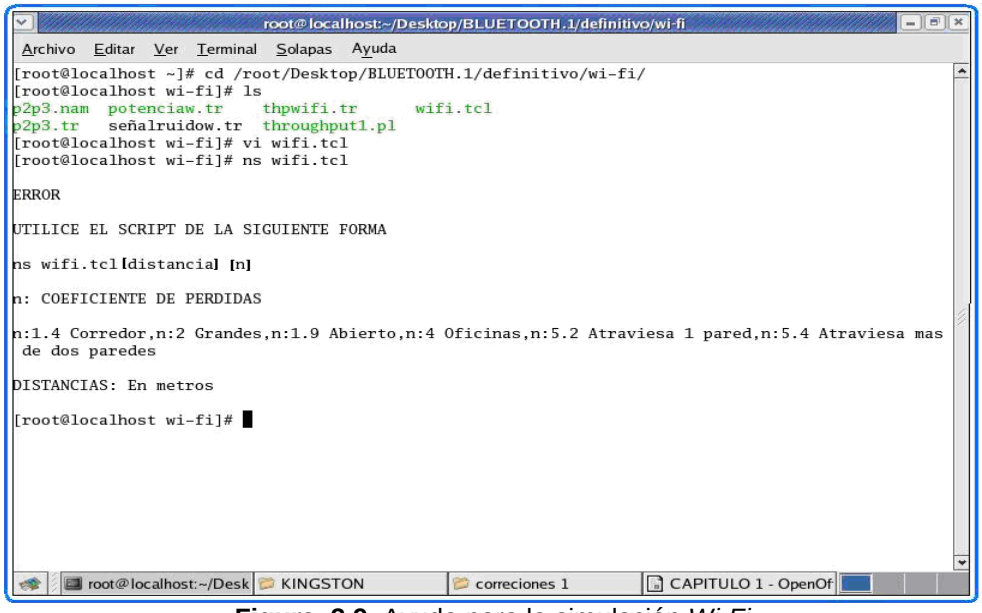


Figura 3.9 Ayuda para la simulación *Wi-Fi*

Aquí el usuario puede escoger la distancia y el coeficiente de pérdidas con el que se desea realizar la simulación. Para ejecutar el *script* ingresar el siguiente comando.

ns wifi.tcl 10 5.2

Una vez que se ejecute el programa se visualizará en el *terminal* de *Linux* la siguiente información. La simulación se realizó con una distancia igual 10 m y un coeficiente de pérdidas igual a 5.2

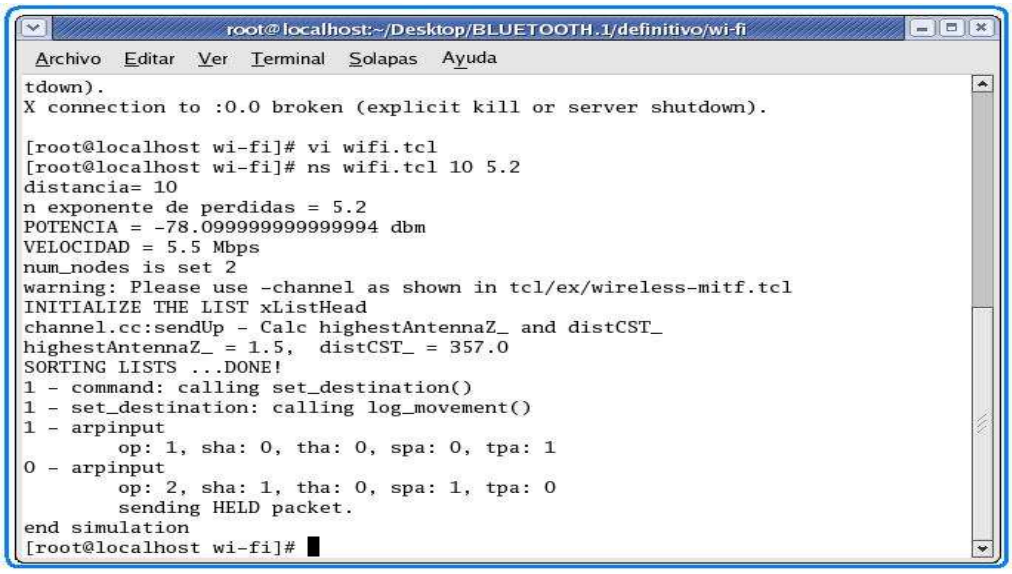


Figura 3.10 Información de la simulación *Wi-Fi*

También se visualizará de forma automática el escenario en el *nam* y los resultados que se obtienen de la simulación en el *xgraph* como son: potencia, relación señal a ruido y velocidad efectiva.

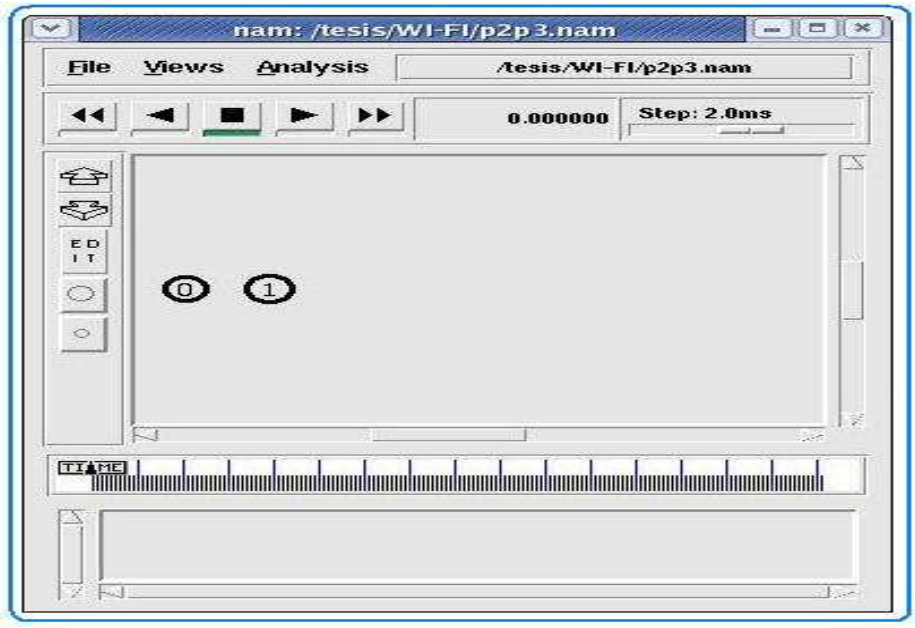


Figura 3.11 Pantalla inicial del nam W*i-Fi*

Luego de iniciar la simulación en el *nam* se visualiza como los paquetes son enviados.

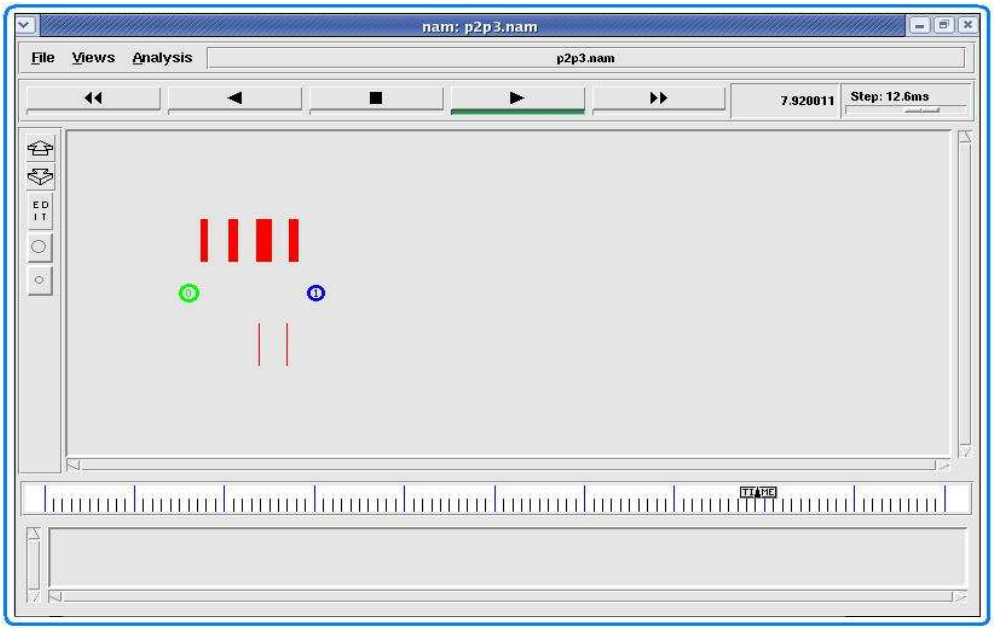


Figura 3.12 Simulación *Wi-Fi* en el *nam*

Con la ayuda del xg*raph* y el archivo *potenciaw.tr* que se genera en la simulación, visualizamos como la potencia cambia en función de la distancia.

Figura 3.13 Potencia *Wi-Fi* de la Simulación

Con la ayuda del *xgraph* y el archivo *señalruidow.tr* generado en la simulación, se visualiza la relación señal a ruido en función de la distancia.

Figura 3.14 Señal a ruido *Wi-Fi* de la Simulación

El archivo *througtput1*.pl que es un programa que permite procesar el archivo *p2p3.tr* generado en la simulación. Este programa permite crear un nuevo archivo con la velocidad efectiva en el nodo que recibe los datos.

Con la ayuda del *xgraph* y el nuevo archivo creado se genera la siguiente gráfica de la velocidad efectiva.

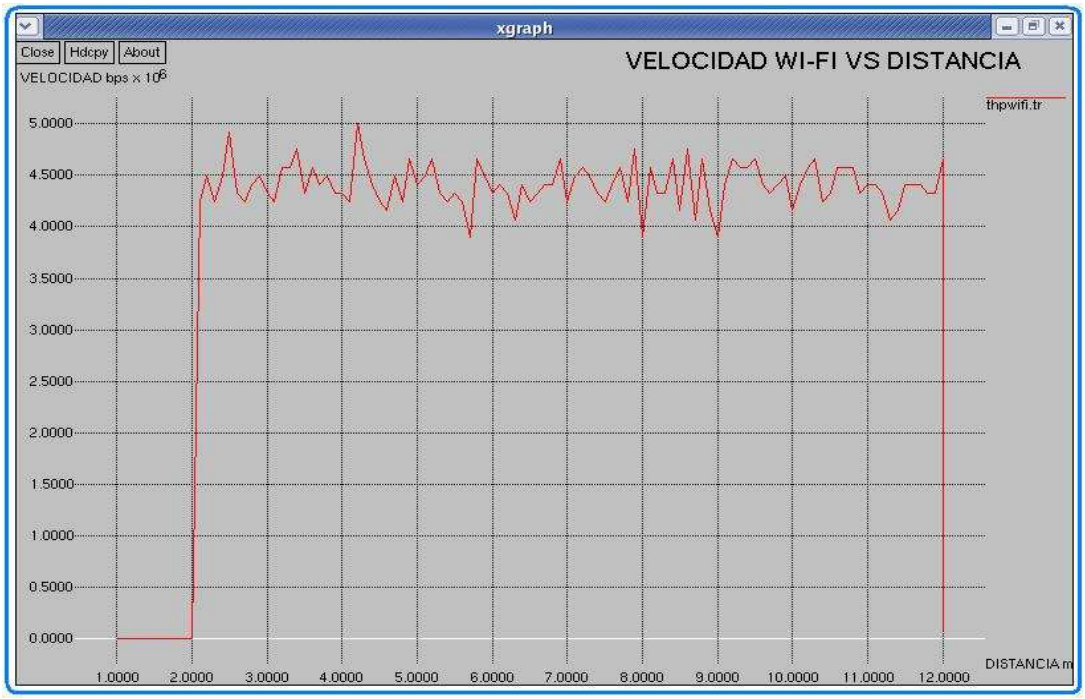


Figura 3.15 Velocidad Efectiva *Wi-Fi* de la Simulación

CAPÍTULO 4

PRUEBAS Y ANÁLISIS DE RESULTADOS

4.1 INTRODUCCIÓN

Uno de los objetivos principales del proyecto es la parte de pruebas, en este capítulo se realizarán las pruebas correspondientes que permitan evaluar el correcto funcionamiento del sistema implementado.

En base a estas pruebas se realizará la comparación de las tecnologías *Bluetooth* y *Wi-Fi*, para así determinar las ventajas y desventajas que implican el uso de estas. Las pruebas a realizarse se separan en dos partes:

- La primera se relaciona a todas las pruebas prácticas correspondientes a los prototipos inalámbricos con tecnología *Bluetooth* y *Wi-Fi*, entre las distintas pruebas a comprobarse están las siguientes: conectividad, nivel de potencia, velocidad, y pérdida de datos.

- La segunda corresponde a la simulación de cada prototipo en la cual se podrá ver como varía la velocidad efectiva, nivel de potencia de recepción y la relación señal a ruido. Para realizar esta prueba se utilizó el simulador *ns-2* versión (2.29.3).

4.2 PRUEBAS PRÁCTICAS

A continuación se detallan las pruebas prácticas correspondientes a los prototipos, las pruebas han sido realizadas iniciando con una distancia de separación mínima de un metro hasta una distancia de separación máxima de quince metros.

4.2.1 PRUEBAS PRÁCTICAS *BLUETOOTH*

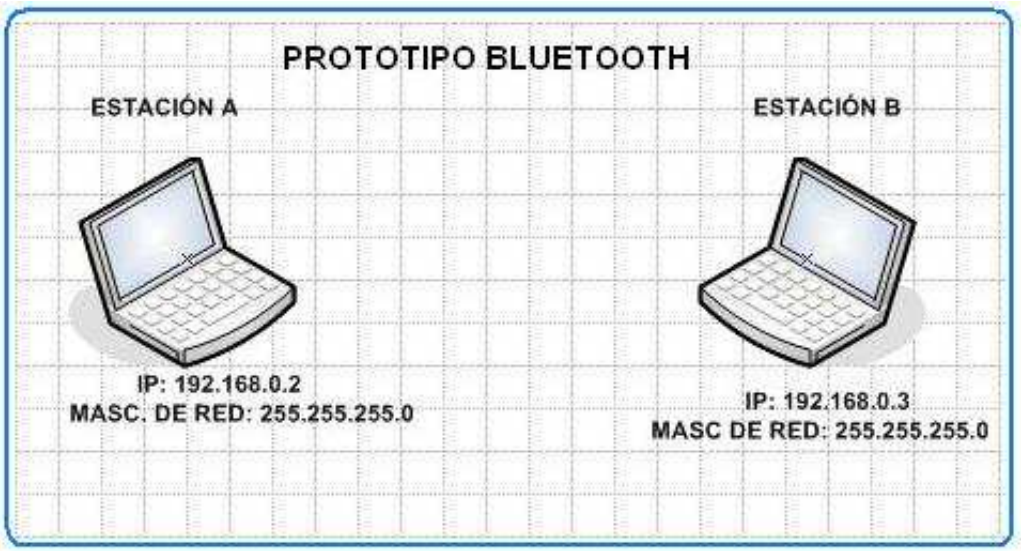


Figura 4.1 Prototipo *Bluetooth*

Se presenta el detalle de las medidas realizadas a la distancia de 1m, todo el conjunto de medidas hasta los 15m se encuentran en el ANEXO D

Medidas a 1 metro de separación

La figura 4.2 indica el nivel de señal detectada por el *softwar*e del interfaz *DBT-122*.

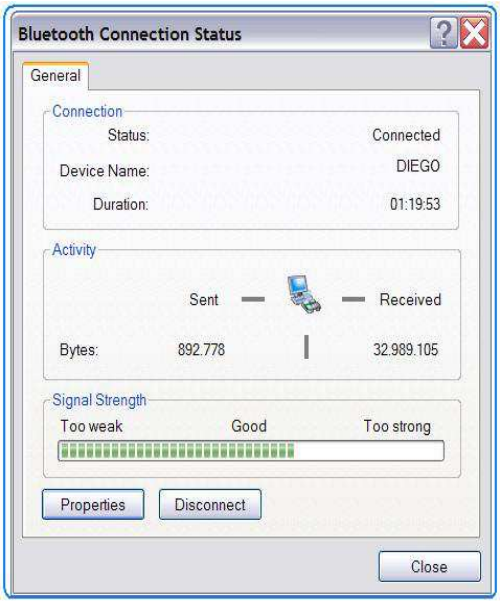


Figura 4.2 Estado de conexión 1m

La figura 4.3 representa el nivel de potencia de la señal medida con el analizador de espectros (d = 1 m, p = - 54.83 dBm, f = 2.4155 GHz)

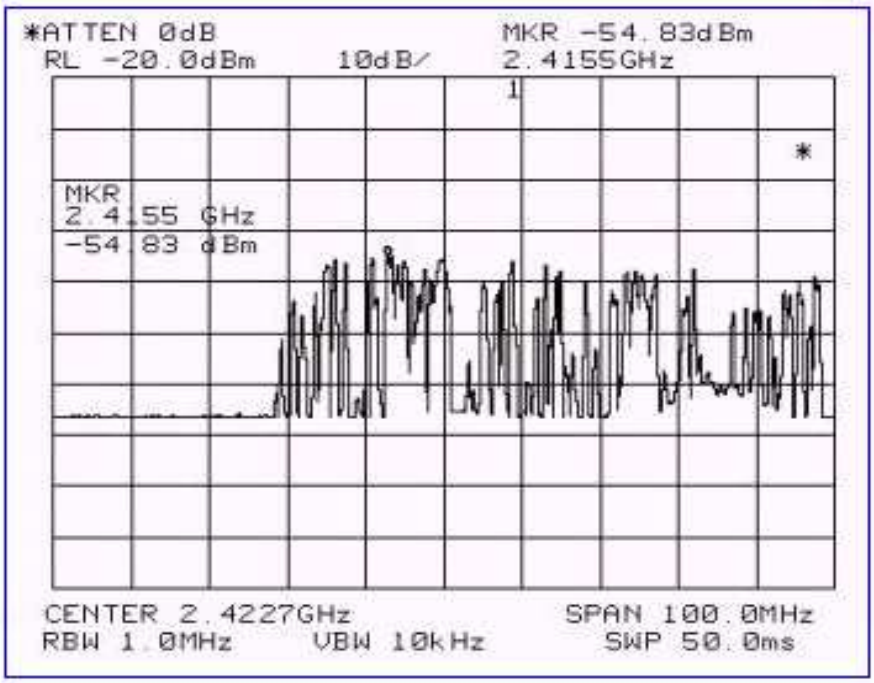


Figura 4.3 Nivel de potencia 1m

La figura 4.4 indica la respuesta entre las estaciones a través del comando *ping*.

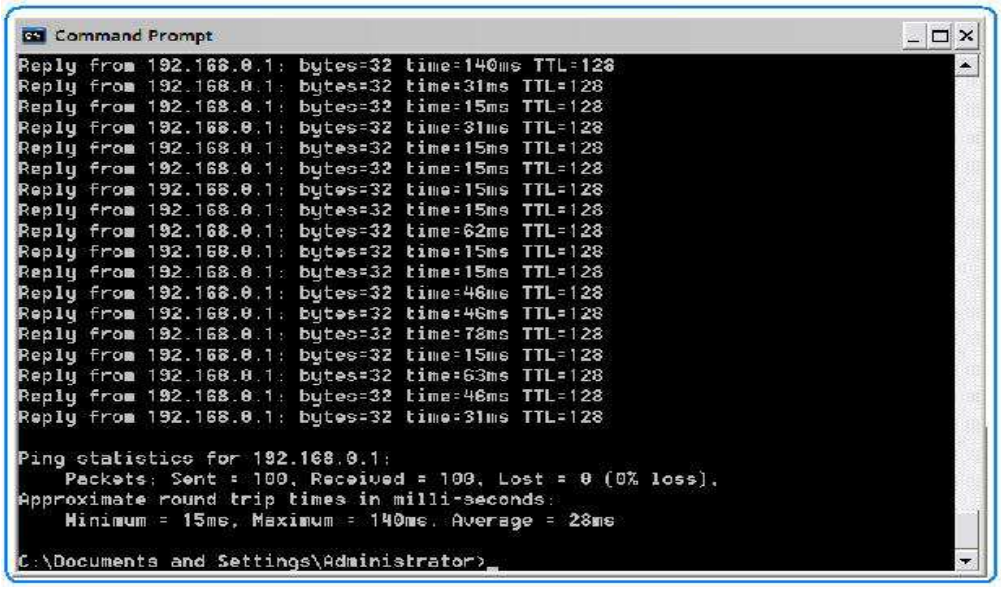


Figura 4.4 *Ping* entre las estaciones 1m

La figura 4.5 indica la velocidad que se obtiene al transferir un archivo en función del tiempo, utilizando el *software Smart FTP.* La velocidad esta expresada en *Kbytes/s.* la figura 4.6 indica la velocidad promedio y el tamaño del archivo transferido.

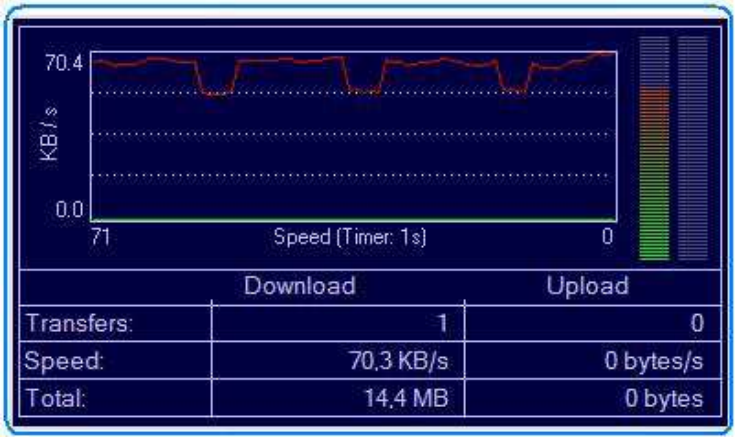


Figura 4.5 Velocidad a un metro de separación

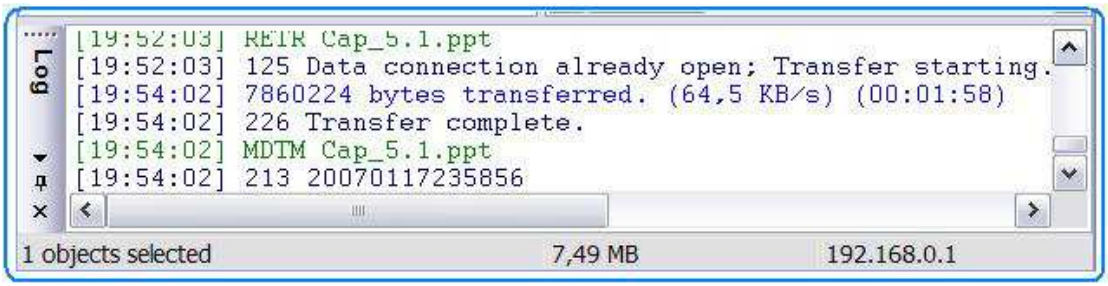


Figura 4.6 Velocidad Promedio a un metro de separación

4.2.2 PRUEBAS PRÁCTICAS *Wi-Fi*

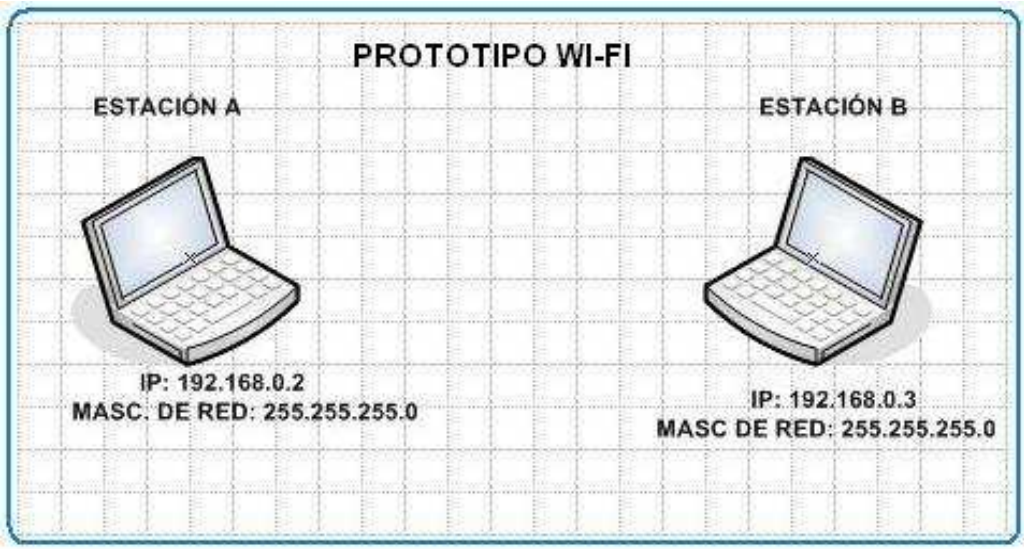


Figura 4.7 Prototipo *Wi-Fi*

Se presenta el detalle de las medidas realizadas a la distancia de 1m, todo el conjunto de medidas hasta los 15m se encuentran en el ANEXO E

Medidas a 1 metro de separación

La figura 4.8 indica el nivel de señal detectada por el *softwar*e del interfaz *DWL-G122*.

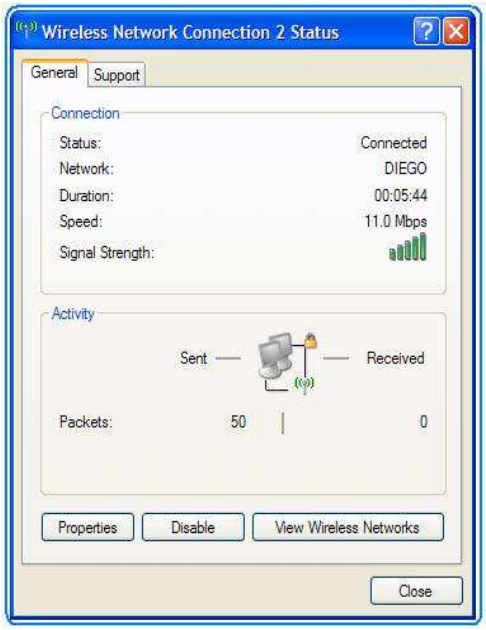


Figura 4.8 Estado de conexión 1m

La figura 4.9 representa el nivel de potencia de la señal medida con el analizador de espectros (d = 1 m, p = - 47.33 dBm, f = 2.4403 GHz)

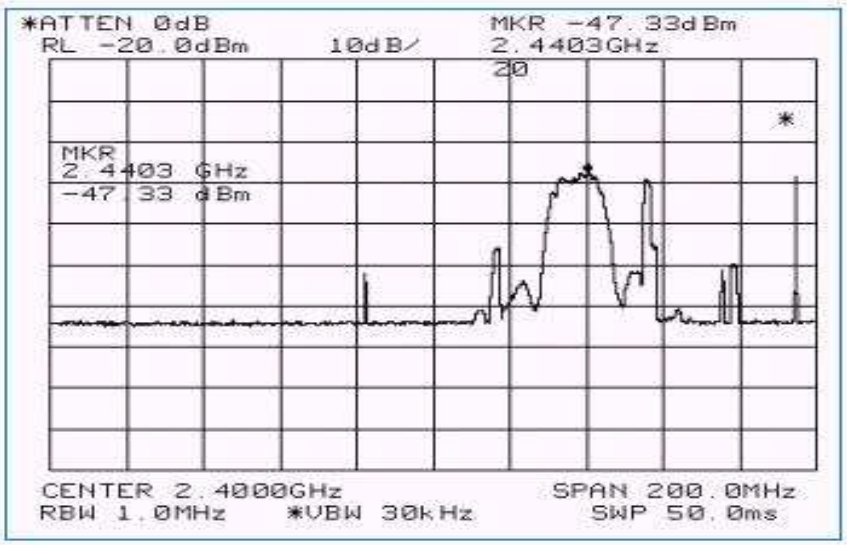


Figura 4.9 Nivel de potencia 1m

La figura 4.10 indica la respuesta entre las estaciones a través del comando *ping*

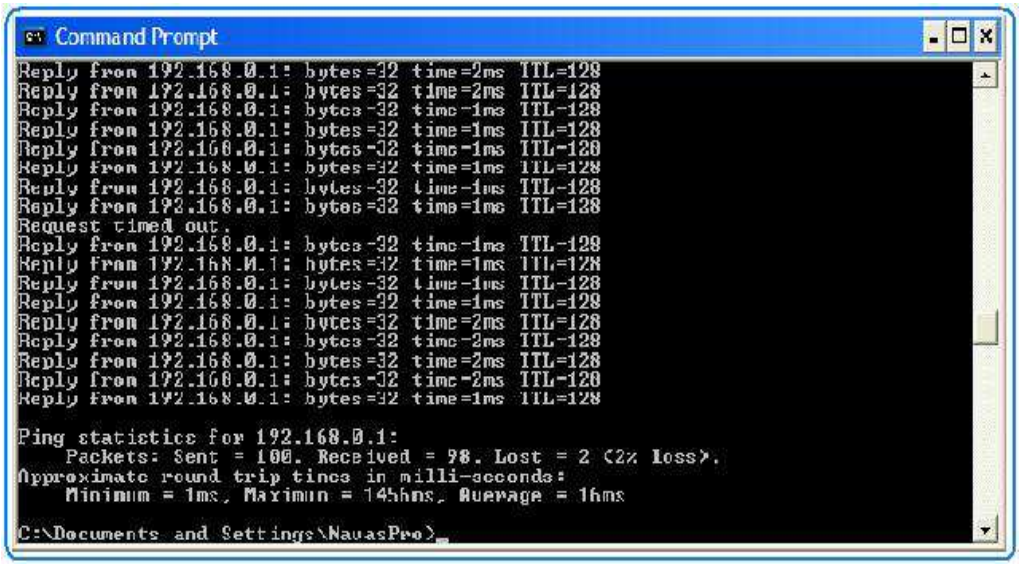


Figura 4.10 *Ping* entre las estaciones 1m

La figura 4.11 indica la velocidad que se obtiene al transferir un archivo en función del tiempo, utilizando el *software Smart FTP.* La velocidad esta expresada en *Kbytes/s.* la figura 4.12 indica la velocidad promedio y el tamaño del archivo transferido.

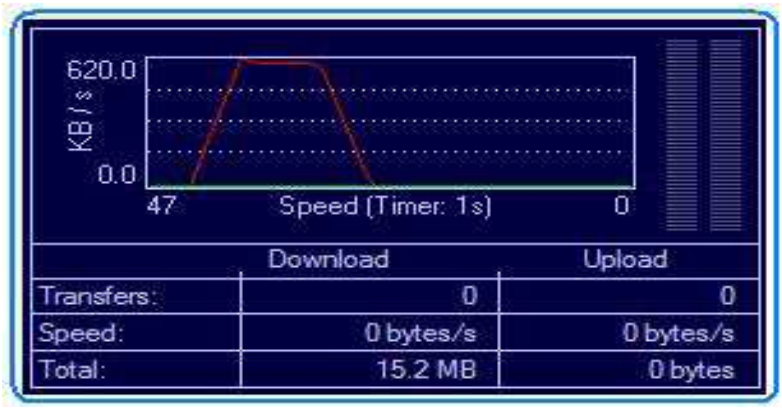


Figura 4.11 Velocidad 1m

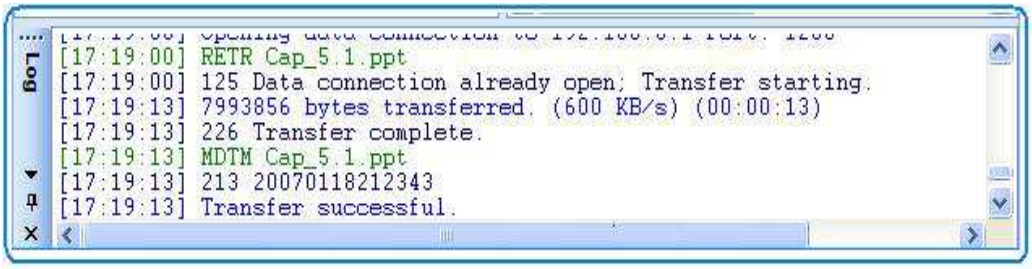


Figura 4.12 Velocidad Promedio 1m

4.2.2.1 Comparación y Análisis de Resultados de Pruebas Prácticas

Para realizar el análisis de los resultados obtenidos en las pruebas prácticas de *Bluetooth* y *Wi-Fi* de mejor manera, se los agrupó en la tabla 4.1 y tabla 4.2 respectivamente y además se realizó la representación gráfica de los datos obtenidos.

No. de prueba	Distancia (m)	Potencia (dBm)	Velocidad promedio (KB/s)	Velocidad promedio (Kbps)	Frecuencia (GHz)	Pérdida de datos %
1	1	-54.83	64.5	528.38	2.4155	0
2	2	-55.67	63.4	519.37	2.4747	0
3	3	-58.83	63.0	516.09	2.4740	0
4	4	-56.83	58.1	475.95	2.4537	0
5	5	-57.17	63.9	523.46	2.4557	0
6	6	-56.83	57.7	472.67	2.4420	0
7	7	-58.17	62.0	507.90	2.4287	0
8	8	-58.00	39.9	326.86	2.4277	0
9	9	-57.33	50.1	410.42	2.4177	0
10	10	-58.50	30.6	250.67	2.4307	0
11	12	-60.00	23.6	193.33	2.4397	0
12	15	-61.83	3.15	25.8	2.4207	0

Tabla 4.1 Resultados obtenidos en las pruebas de *Bluetooth*

No. de prueba	Distancia (m)	Potencia (dBm)	Velocidad promedio (KB/s))	Velocidad promedio (Mbps)	Frecuencia (GHz)	Pérdida de datos %
1	1	-47.33	600.0	4.91	2.4403	1
2	2	-59.00	613.0	5.02	2.4340	2
3	3	-60.83	610.0	4.99	2.4410	1
4	4	-56.83	557.0	4.56	2.4383	1
5	5	-58.33	589.0	4.82	2.4383	3
6	6	-62.50	587.0	4.80	2.4337	2
7	7	-58.50	594.0	4.86	2.4383	1
8	8	-66.83	604.0	4.94	2.4323	2
9	9	-66.50	607.0	4.97	2.4347	4
10	10	-68.83	610.0	4.99	2.4207	3
11	12	-69.39	627.2	5.13	2.4397	9
12	15	-70.50	561	4.59	2.4400	19

Tabla 4.2 Resultados obtenidos en las pruebas de *Wi-Fi*

4.2.2.2 Representación gráfica de resultados obtenidos en las pruebas prácticas

La figura 4.13 representa la pérdida de paquetes en función de la distancia para el prototipo *Bluetooth*, se puede observar que en este caso no existe pérdida de datos, por lo que se puede decir que este enlace es confiable.

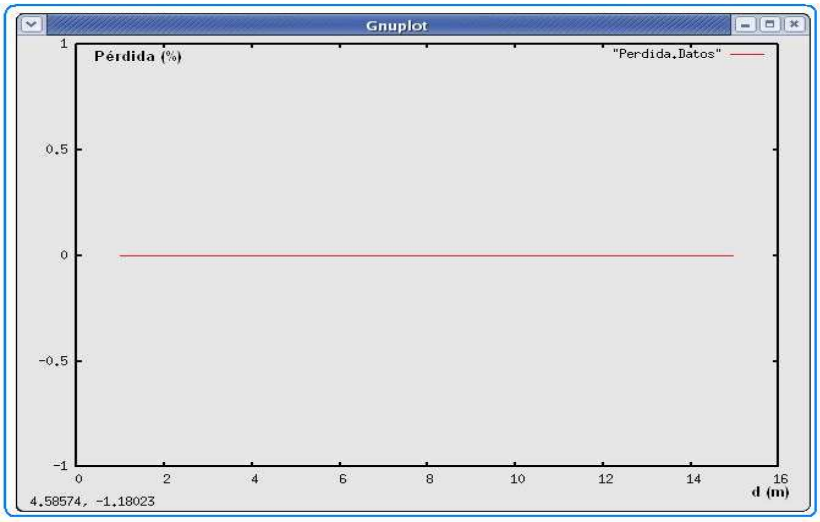


Figura 4.13 Pérdida de Datos vs Distancia de *Bluetooth*

La figura 4.14 representa la pérdida de paquetes en función de la distancia del prototipo *Wi-Fi*, aquí se observa que en distancias cortas las pérdidas son despreciables, pero a medida que aumenta la distancia estas pérdidas aumentan lo que ocasiona que el sistema sea desconfiable.

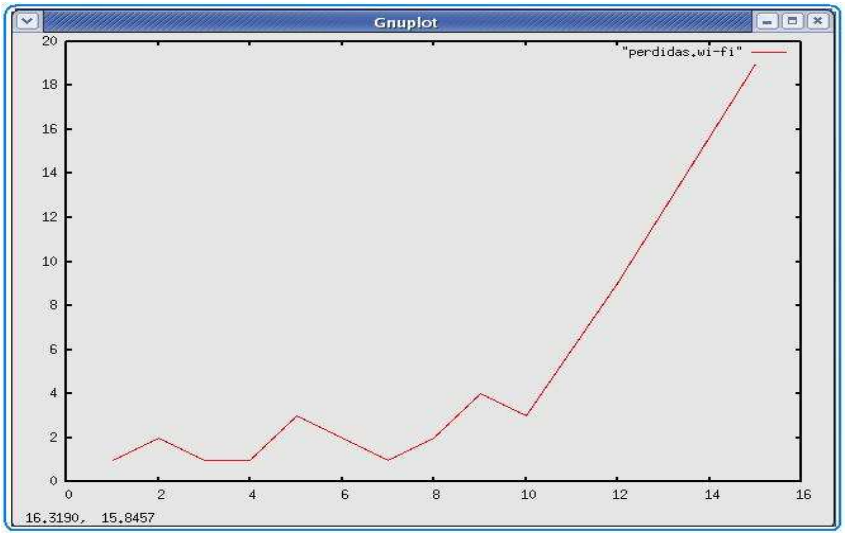


Figura 4.14 Pérdida de Datos vs Distancia de *Wi-Fi*

La figura 4.15 indica la variación del nivel de potencia en función de la distancia del prototipo *Bluetooth*. Se puede observar que a distancias menores a 10 metros los niveles de potencia no sufren atenuaciones considerables, pero a partir de la misma distancia el enlace sufre una atenuación considerable en la señal de potencia.

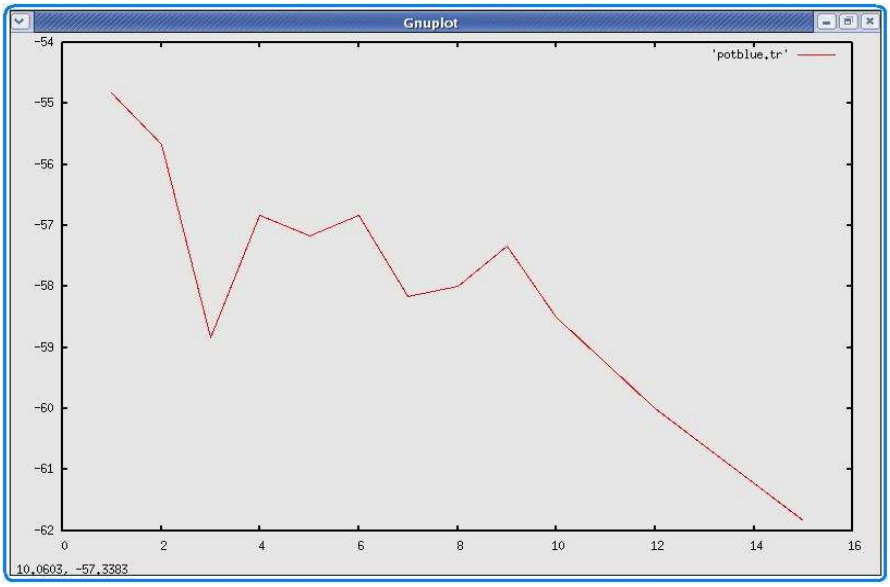


Figura 4.15 Potencia vs Distancia de *Bluetooth*

La figura 4.16 indica la variación del nivel de potencia del prototipo *Wi-Fi*, como se puede ver este sistema en distancias menores a 10 metros tiene atenuaciones considerables, pero a partir de la misma distancia el nivel de señal se mantiene estable.

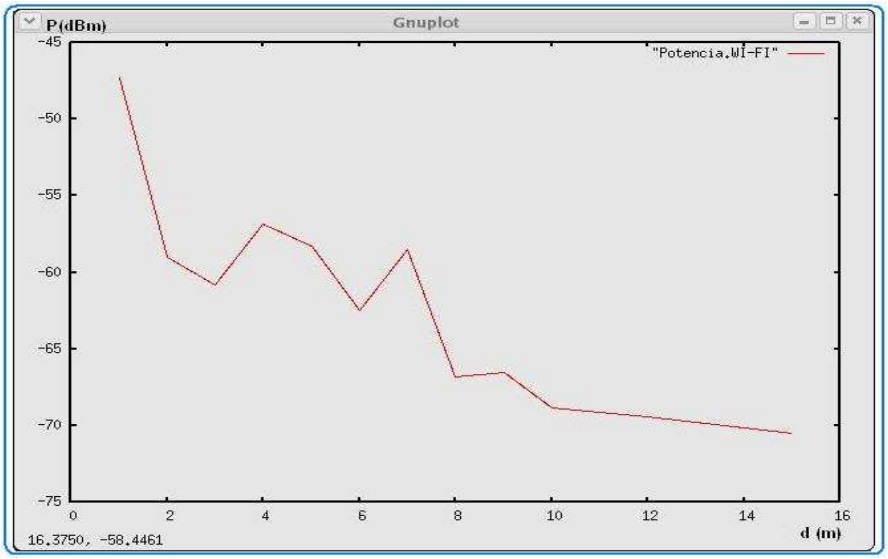


Figura 4.16 Potencia vs Distancia de *Wi-Fi*

La figura 4.17 indica la variación de la velocidad de transmisión en función de la distancia del prototipo *Bluetooth,* se puede observar que en distancias menores a los 6 metros la velocidad es estable, pero a distancias mayores a 7 metros esta velocidad disminuye notablemente.

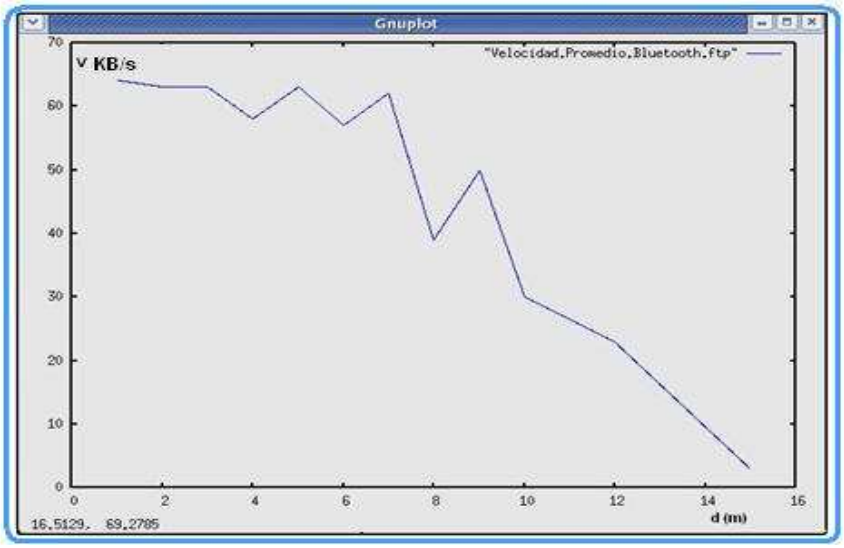


Figura 4.17 Velocidad Promedio vs Distancia de *Bluetooth*

La figura 4.18 indica la variación de la velocidad de transmisión de datos del prototipo *Wi-Fi,* aquí se puede observar que la velocidad esta dentro de los márgenes de velocidad aceptables. Cabe mencionar que para la distancia igual 4 m esta velocidad disminuye debido a obstáculos existentes donde se realizaron las pruebas que hacen que la señal se degrade.

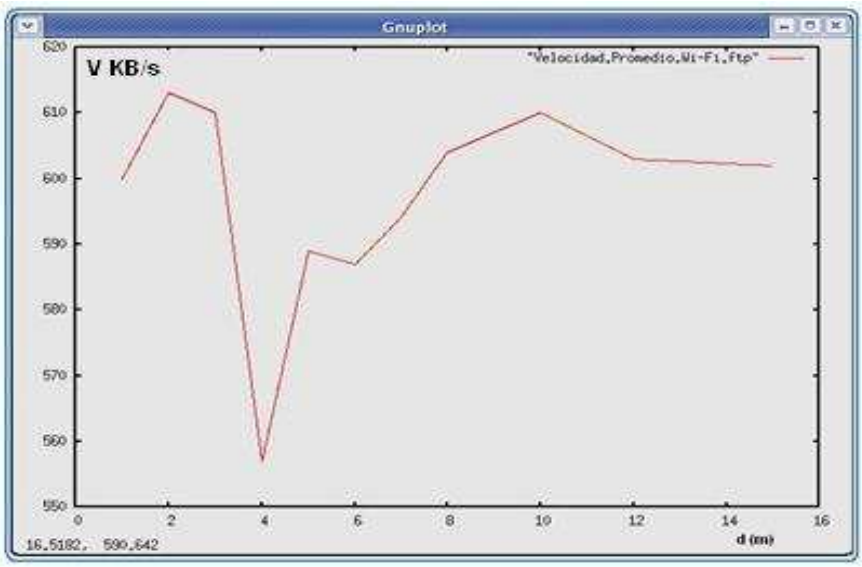


Figura 4.18 Velocidad Promedio vs Distancia de *Wi-Fi*

4.2.2.3 Comparación de pruebas prácticas

La figura 4.19 representa la comparación entre *Bluetooth* y *Wi-Fi* con respecto a pérdida de datos en función de la distancia, aquí se observa que *Bluetooth* no tiene pérdida de datos para este caso, ya que en el sitio donde se realizaron las pruebas no existían redes que utilicen este tipo de tecnología, mientras que en *Wi-Fi* la pérdida de datos ocurre a medida que aumenta la distancia. Como conclusión se puede decir que *Bluetooth* es más estable que *Wi-Fi* con respecto a pérdida de datos.

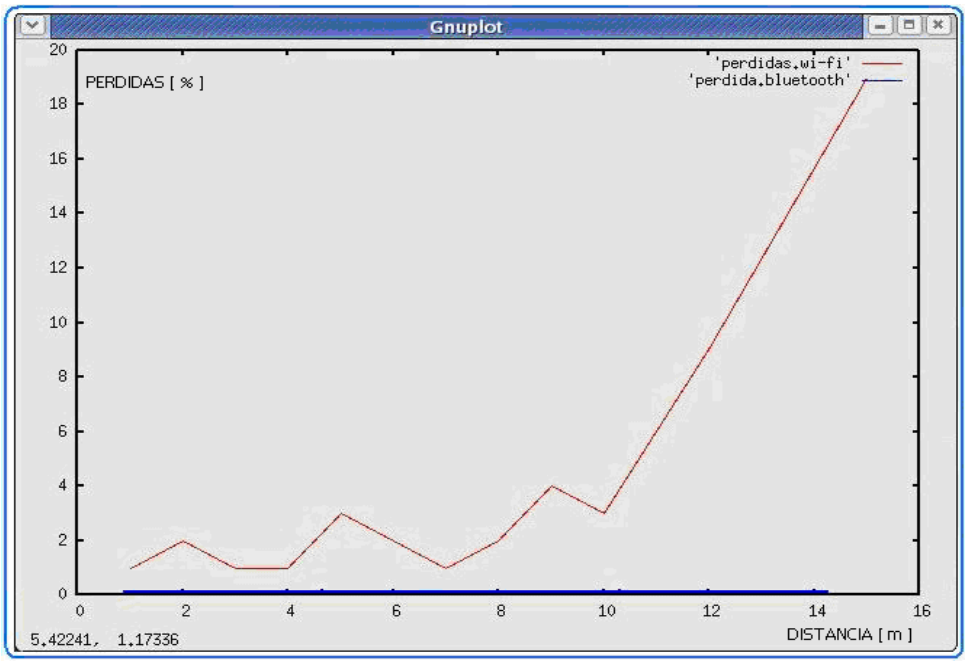


Figura 4.19 Pérdida de Datos *Bluetooth* y *Wi-Fi*

La figura 4.20 representa la comparación de los niveles de potencia para *Bluetooth* y *Wi-Fi* en función de la distancia, se observa que *Bluetooth* presenta mayor estabilidad que *Wi-Fi*. La curva que representa la potencia de *Wi-Fi* decrece más rápidamente que la de *Bluetooth*, porque los dispositivos *Wi-Fi* son más propensos a receptar mayores interferencias existentes en el medio.

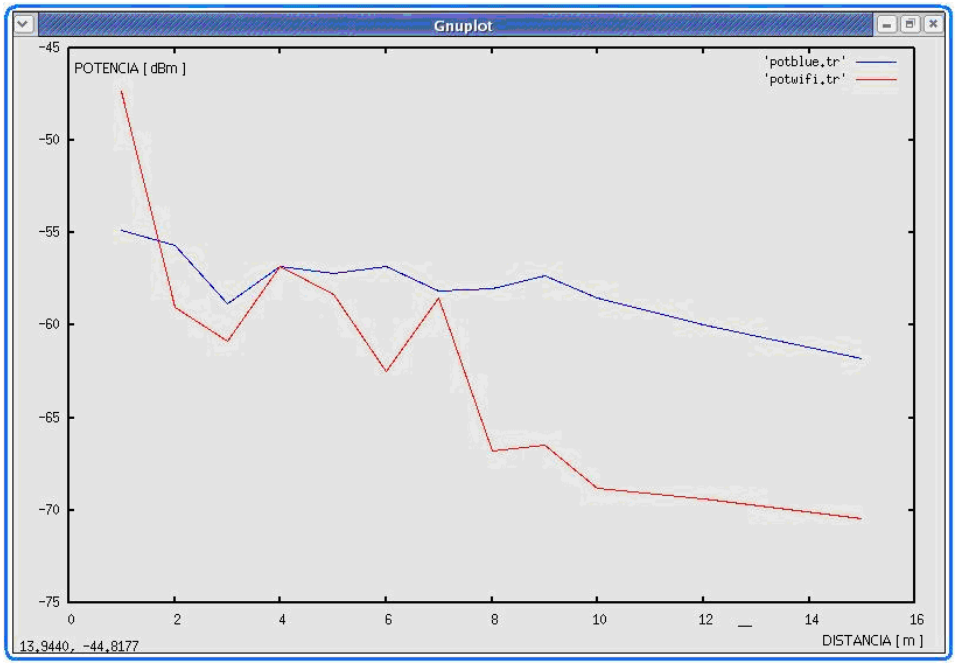


Figura 4.20 Potencia Práctica *Bluetooth* y *Wi-Fi*

En la figura 4.21 se puede observar que *Wi-Fi* tiene una mayor velocidad de transmisión que *Bluetooth*. Esto se debe ya que *Wi-Fi* transmite a una velocidad de 11 *Mbps* mientras que *Bluetooth* transmite a una velocidad máxima de 723 *Kbps.*

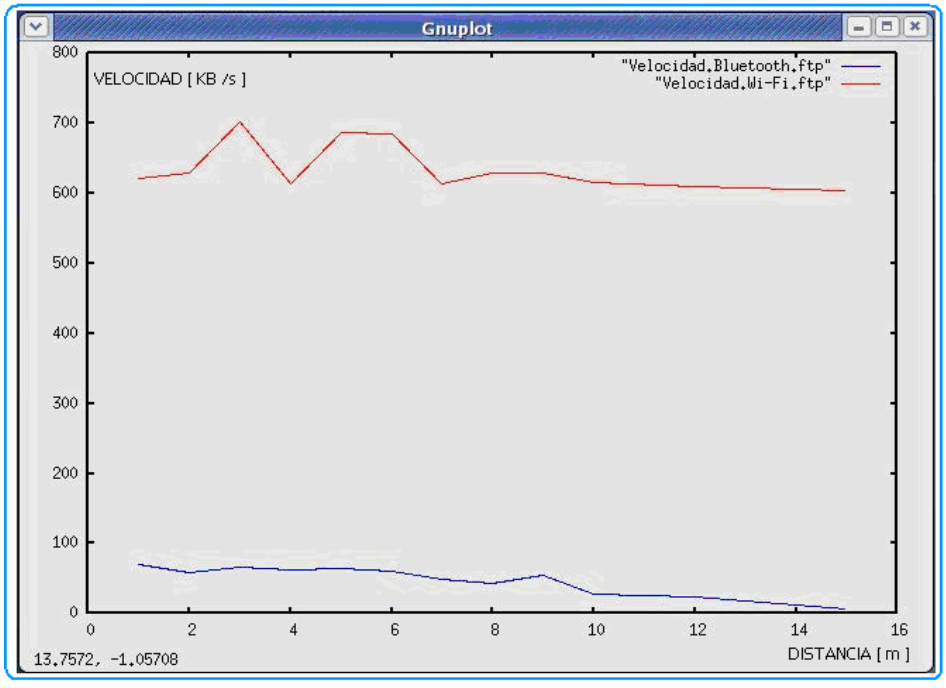


Figura 4.21 Velocidad Promedio *Bluetooth* y *Wi-Fi*

4.2.2.4 Análisis de resultados de las pruebas prácticas

Luego de realizar las pruebas y analizar los resultados obtenidos en la implementación de los prototipos se concluye que el prototipo *Bluetooth* es más estable que *Wi-Fi*, ya que la señal de potencia del prototipo *Bluetooth* sufre menos atenuaciones que *Wi-Fi.*

En el prototipo *Bluetooth* no existen pérdidas de datos lo que hace que este prototipo sea confiable al momento de transmitir datos, mientras que el prototipo *Wi-Fi* tiene pérdidas de datos considerables a partir de los 10 metros.

.

Con respecto a la velocidad de transmisión de datos, en el enlace con tecnología *Bluetooth* la velocidad varia a medida que aumenta la distancia, mientras que en el prototipo *Wi-Fi* la velocidad se mantiene estable.

4.3 PRUEBAS SIMULADAS

Mediante la simulación de los prototipos se trató de obtener las diferentes respuestas de potencia, señal a ruido, velocidad efectiva, lo más cercano a la implementación práctica a medida que el simulador permite. Debido a las limitaciones existentes para la simulación de *Bluetooth* la distancia varía desde los 2 m hasta los 12 m para ambos casos. A continuación se presenta los resultados de la simulación.

4.3.1 BLUETOOTH

La figura 4.22 indica la variación de la potencia en función de la distancia, aquí se observa que a medida que se alejan las estaciones la potencia disminuye tal como ocurre en la realidad.

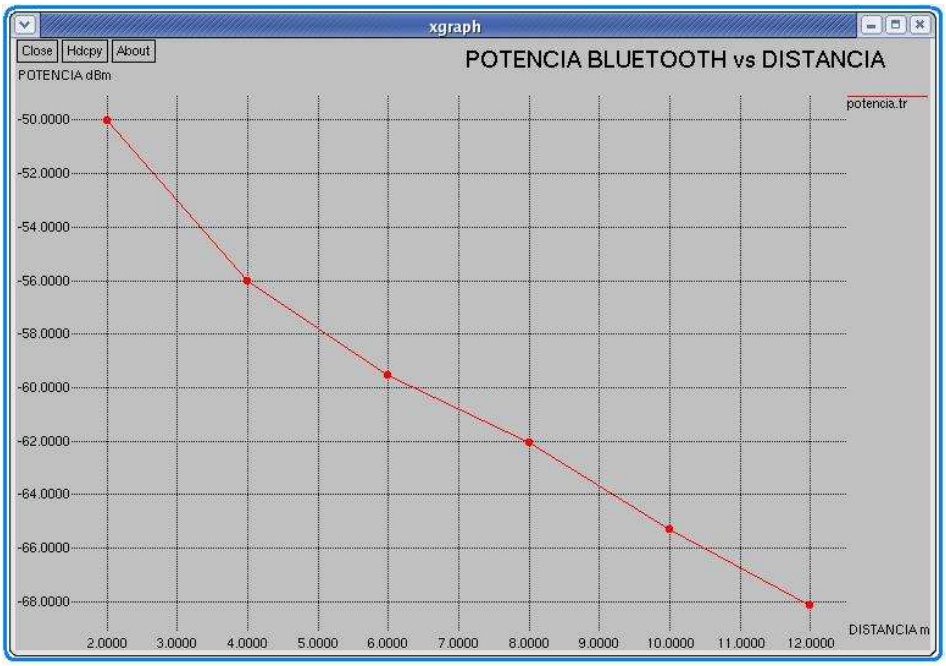


Figura 4.22 Potencia *Bluetooth* de la Simulación

La figura 4.23 representa la variación de la relación señal a ruido de la simulación en función de la distancia, a medida que se alejan las estaciones la relación señal a ruido decrece.

Figura 4.23 Señal a Ruido *Bluetooth* de la Simulación

La figura 4.24 representa la variación de la velocidad efectiva en función de la distancia, en este gráfico se puede observar valores máximos y mínimos de la velocidad esta variación depende de la cola del buffer, es decir si existe congestión en el enlace se emite un mensaje para detener el envió momentáneo de datos.

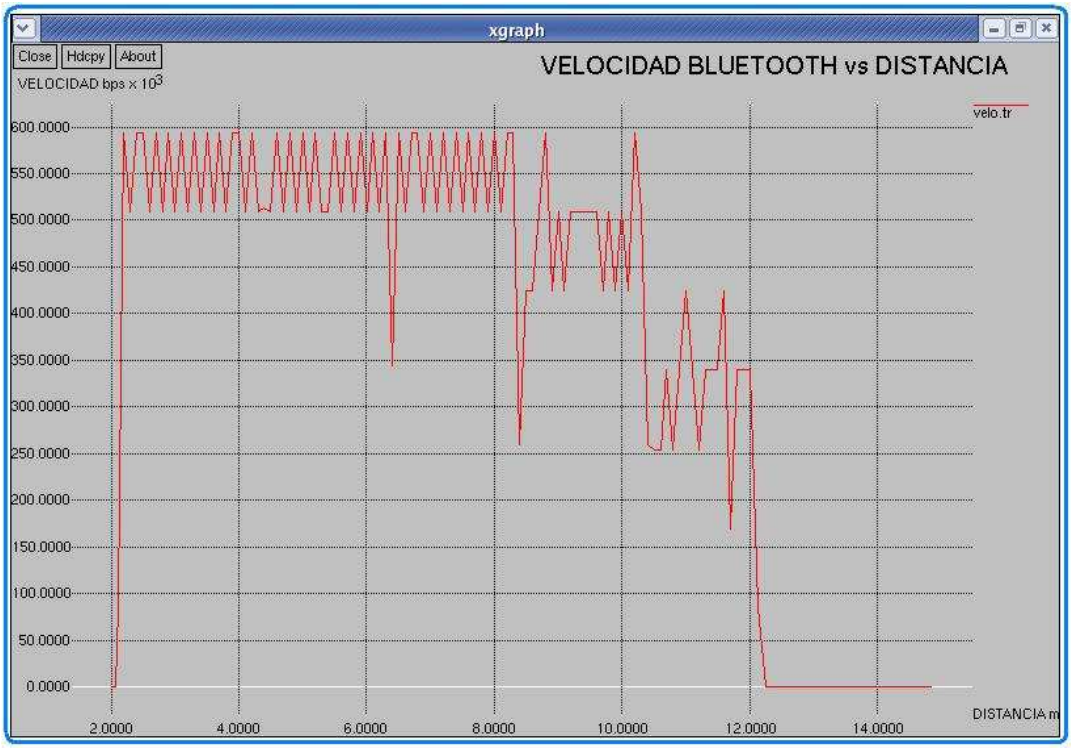


Figura 4.24 Velocidad *Bluetooth* de la Simulación

4.3.2 WI-FI

La figura 4.25 representa la variación de la potencia en función de la distancia, aquí se observa que a medida que se alejan las estaciones la potencia disminuye tal como ocurre en la realidad.

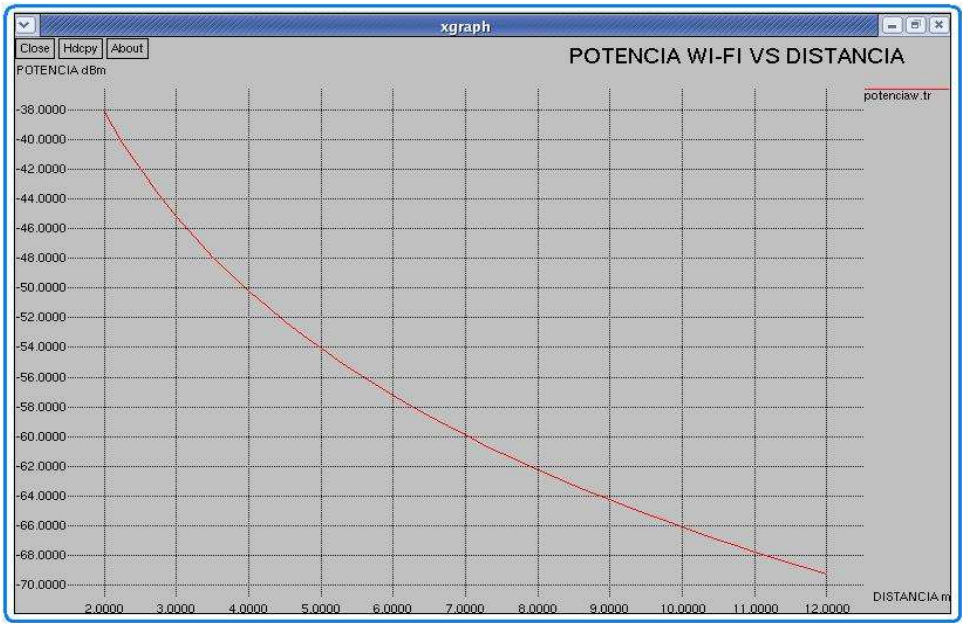


Figura 4.25 Potencia *Wi-Fi* de la Simulación

La figura 4.26 indica la variación de la relación señal a ruido de la simulación en función de la distancia para el prototipo *Wi-Fi,* este resultado depende de la distancia de separación de los dispositivos y del ruido existente

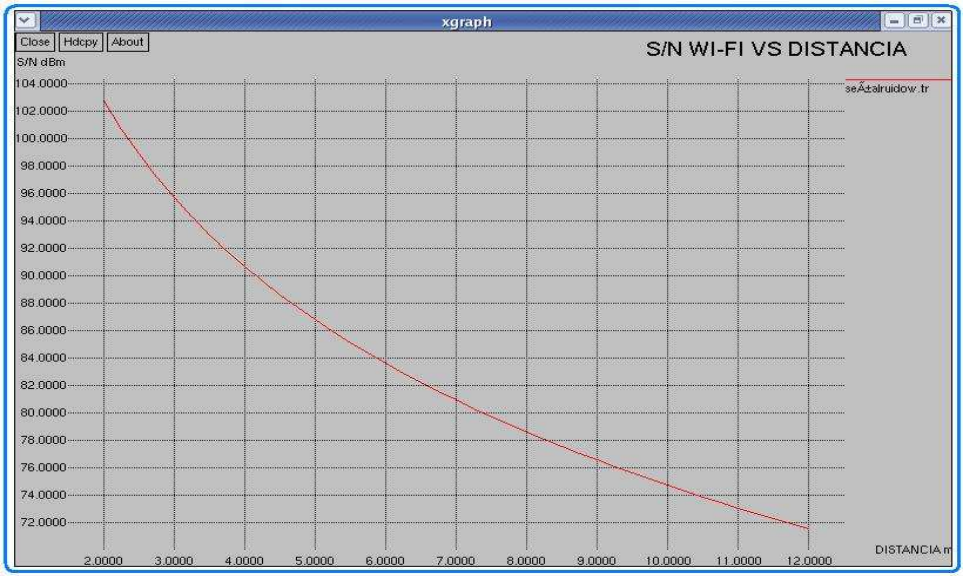


Figura 4.26 Señal a ruido *Wi-Fi* de la Simulación

En la figura 4.27 representa la variación de la velocidad efectiva de *Wi-Fi* en función de la distancia, los valores máximos y mínimos en la velocidad dependen de la cola del *buffer*, es decir si existe congestión en el enlace se emite un mensaje para detener el envío momentáneo de datos.

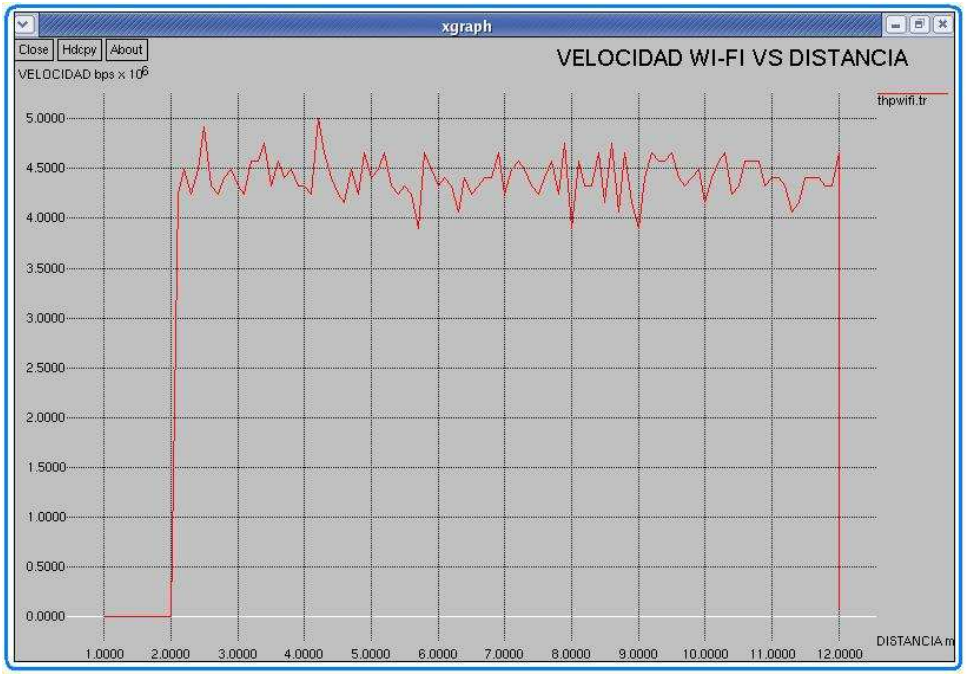


Figura 4.27 Velocidad Efectiva *Wi-Fi* de la Simulación

4.3.3 COMPARACIÓN GRÁFICA DE LAS SIMULACIONES

La figura 4.28 representa la comparación entre *Bluetooth* y *Wi-Fi* con respecto a la potencia, como se puede ver *Wi-Fi* posee niveles de potencia más altos que *Bluetooth* en distancias cortas, pero a medida que se alejan las estaciones *Bluetooth* se comporta de mejor manera.

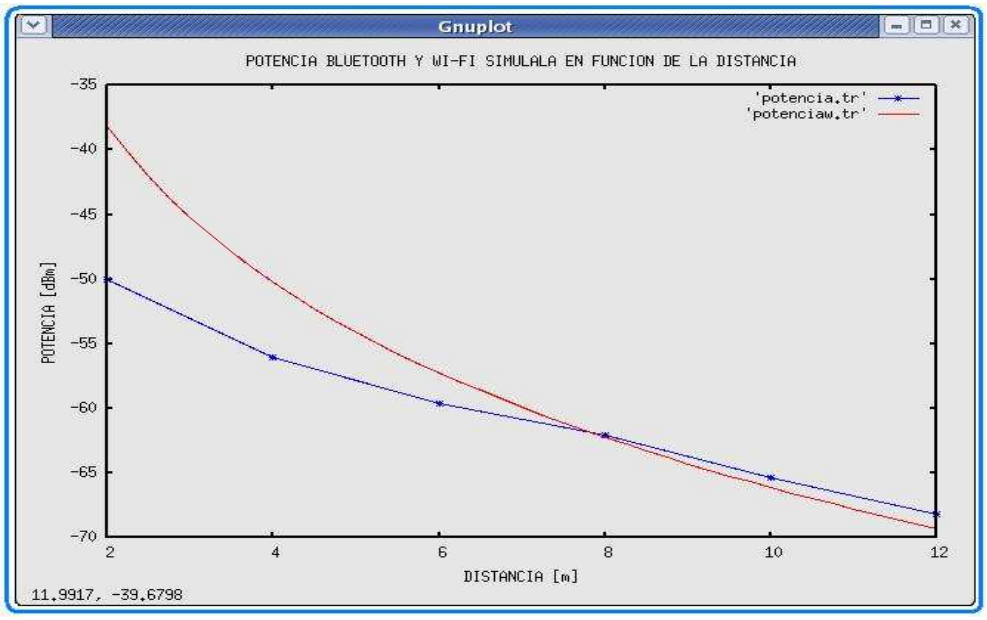


Figura 4.28 Potencia *Bluetooth* y *Wi-Fi* de la Simulación

La figura 4.29 representa la comparación de la relación señal a ruido entre *Bluetooth* y *Wi-Fi*, aquí se observa que *Bluetooth* tiene una mejor relación señal a ruido que *Wi-Fi.*

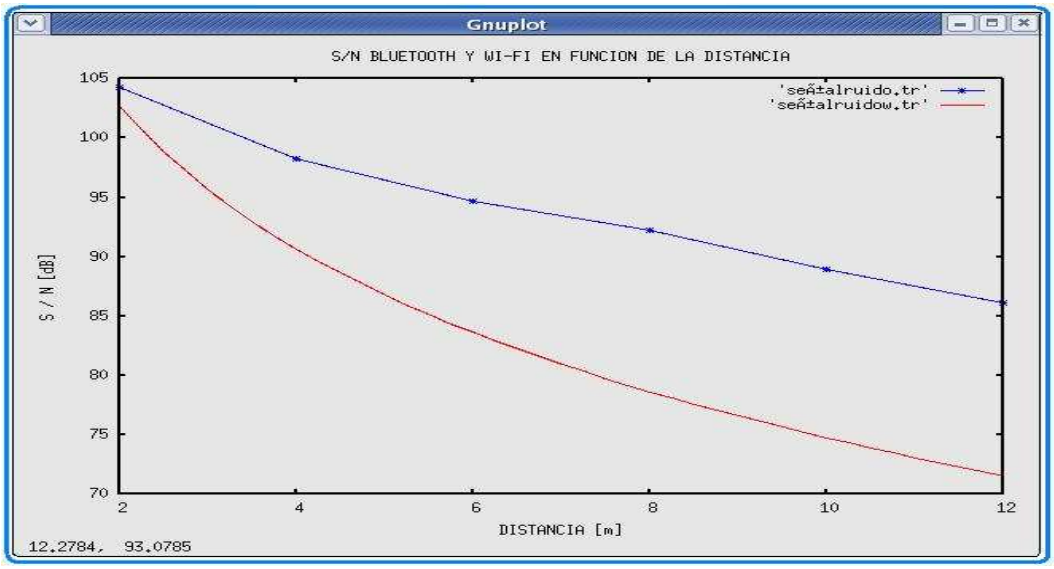


Figura 4.29 Señal a ruido *Bluetooth* y *Wi-Fi* simuladas

La figura 4.30 indica la comparación entre *Bluetooth* y *Wi-Fi* con respecto a la velocidad efectiva, aquí se puede ver claramente que *Wi-Fi* posee una mayor tasa de transferencia que *Bluetooth*, como ocurre en la práctica.

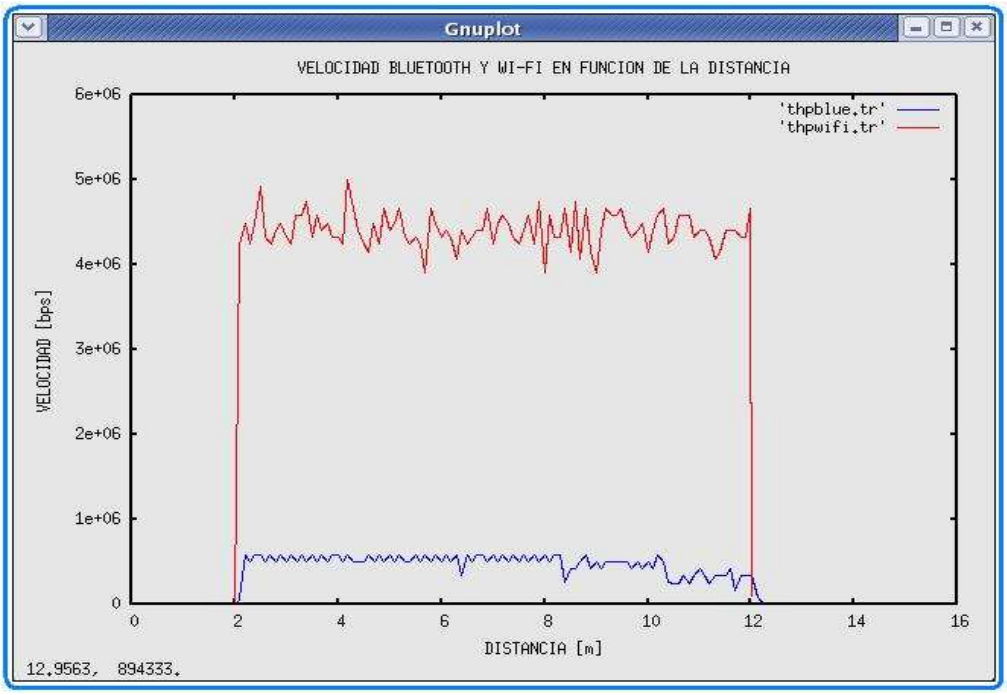


Figura 4.30 Velocidad *Bluetooth* y *Wi-Fi* simuladas

4.4 COMPARACIÓN PRUEBAS PRÁCTICAS CON SIMULADAS

A continuación se realiza una comparación gráfica entre las pruebas prácticas y simuladas en función de la distancia, desde los 2 metros hasta el alcance máximo de *Bluetooth* en el simulador que es de 12 m.

La figura 4.31 representa la potencia práctica y simulada para *Bluetooth,* como se puede apreciar la potencia simulada se asemeja a un decrecimiento logarítmico ya que por ser la simulación, ésta no va a ser inestable como en el caso práctico en el cual la señal está variando debido a que ésta depende del ambiente externo, como temperatura, objetos que se encuentran en el medio a más de otros ambientes inalámbricos.

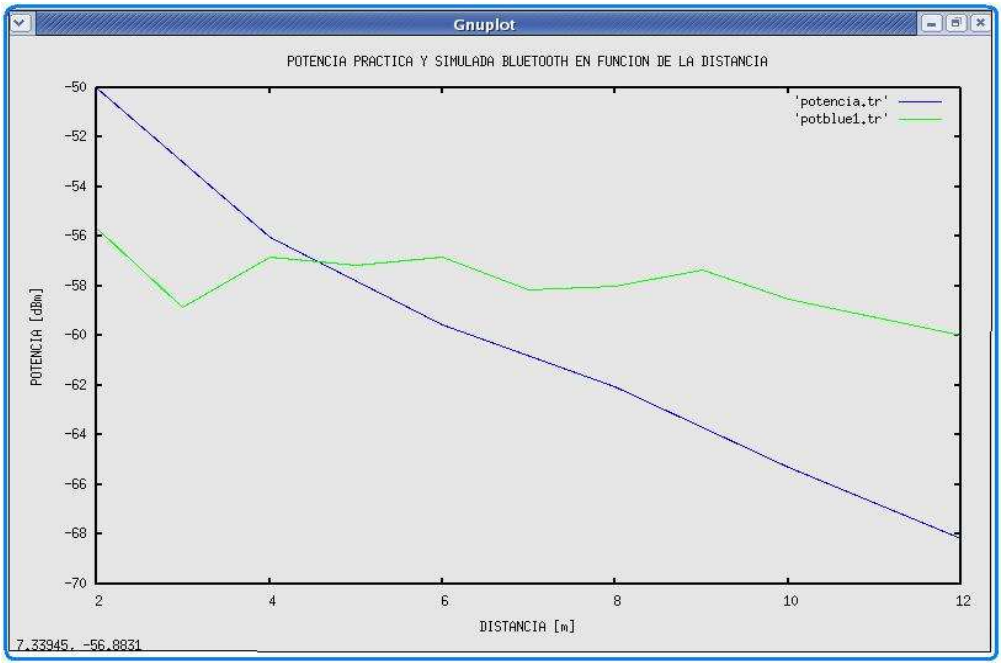


Figura 4.31 Potencia Práctica y Simulada *Bluetooth*

La figura 4.32 representa la velocidad práctica y simulada para *Bluetooth*, se puede observar claramente que tanto la velocidad práctica como la simulada varían con respecto a la distancia.

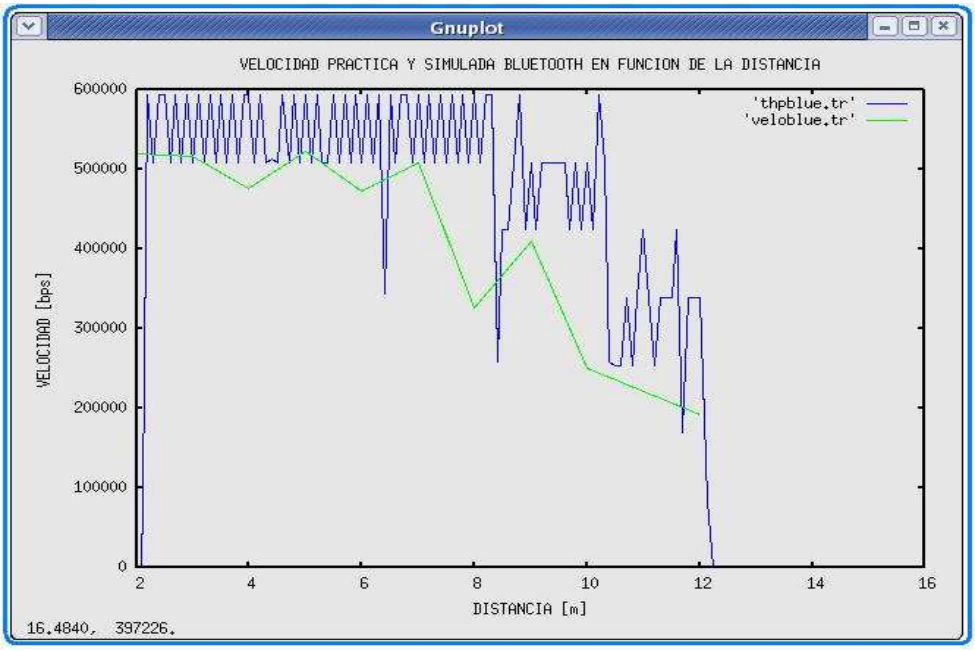


Figura 4.32 Velocidad Práctica y Simulada *Bluetooth*

La figura 4.33 representa la potencia práctica y simulada para *Wi-Fi,* como se puede apreciar la potencia simulada tiene un decrecimiento logarítmico ya que por ser la simulación esta no va a ser inestable como en el caso práctico en el cual la señal esta variando debido a que ésta depende el ambiente externo, como temperatura, objetos que se encuentran en el medio a más de otros ambientes inalámbricos.

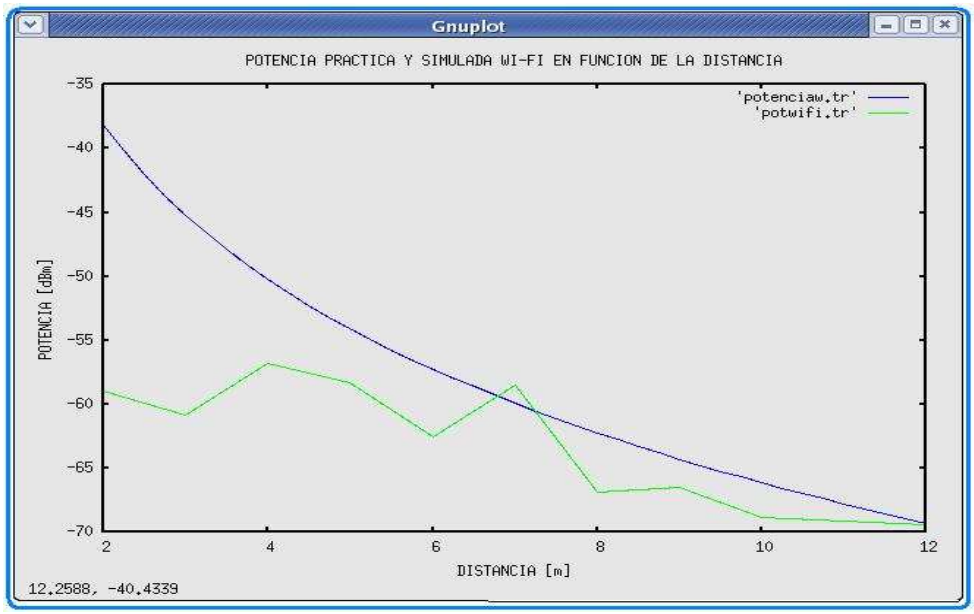


Figura 4.33 Potencia Práctica y Simulada *Wi-Fi*

La figura 4.34 representa las velocidades práctica y simulada *Wi-Fi*, se puede ver claramente que la velocidad simulada se mantiene en un rango aceptable, mientras que la velocidad práctica se encuentra variando en función de la distancia. También se observa que la velocidad simulada es similar a la práctica.

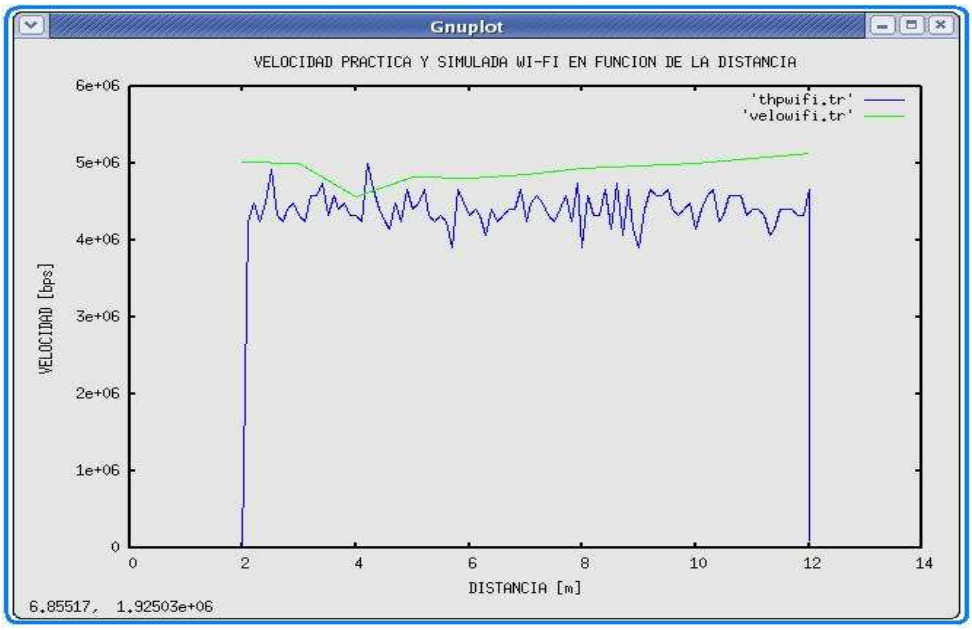


Figura 4.34 Velocidad Práctica y Simulada *Wi-Fi*

CAPÍTULO 5

CONCLUSIONES Y RECOMENDACIONES

5.1 CONCLUSIONES

- Hoy en día la tecnología *Bluetooth* está siendo adoptada por un buen número de fabricantes de *hardware*, ganando cada vez mayor aceptación y penetración en el mercado, que trae como consecuencia la disminución de sus costos de comercialización, lo cual beneficia al usuario final de esta tecnología.

- *Bluetooth* a más de prescindir de los tradicionales y molestos cables empleados para conectar dispositivos digitales entre sí (computadores de escritorio, *PC* portátiles, impresoras, teléfonos móviles, *PDAs*, etc.), permite la conformación de grupos cerrados de usuarios de manera dinámica, este tipo de usuarios operan en redes con infraestructura no fijas, y proporciona una interfaz universal que permite la interoperabilidad entre los diferentes dispositivos, gracias al carácter abierto de la especificación, por todas estas bondades mencionadas se puede decir que *Bluetooth* es un sistema revolucionario, comparado con los sistemas de comunicaciones inalámbricos existentes en la actualidad.

- Una de las principales desventajas que posee *Bluetooth,* frente a otras tecnologías inalámbricas es su alcance, velocidad de transmisión de datos, ya que esta tecnología fue creada para trabajar en ambientes de redes *WPAN.*

- La gran ventaja que posee *Bluetooth,* sobre otras tecnologías es que está diseñada para entregar servicio inalámbrico a dispositivos de gran movilidad, de reducido tamaño y bajo consumo de potencia que les proporcionen portabilidad e independencia de una fuente de energía fija; dispositivos de tipo periférico o que a su vez trabajen en ambientes de

corto alcance. Por esta razón se puede decir que las aplicaciones de esta tecnología es inmensa y que su límite se encuentra en la imaginación del que crea el sistema inalámbrico.

- Una de las principales limitaciones que actualmente posee *Wi-Fi*, es que ésta permite la interoperabilidad únicamente entre computadores de escritorio con interfaces inalámbricas *Wi-Fi 802.11 b,g* y *PC* portátiles que posean esta interfaz.

- *Wi-Fi* ofrece mayor cobertura y velocidades de transmisión relativamente altas. Por todo esto se puede decir que *Wi-Fi* lleva una ligera ventaja, frente a otras tecnologías inalámbricas, otra de las ventajas que posee *Wi-Fi* es que esta tecnología permite expandir una red cableada o a su vez reemplazarla.

- Luego de realizadas las pruebas prácticas de *Bluetooth* y *Wi-Fi* se puede concluir que en ambientes pequeños la señal de potencia de *Bluetooth* sufre menor atenuación que *Wi-Fi*, también se puede decir que en cuanto a estabilidad, pérdida de datos, *Bluetooth* tiene un mejor desempeño que *Wi-Fi*.

- En cuanto al consumo de potencia, las interfaces *Bluetooth* tiene menor consumo que las interfaces *Wi-Fi*, por este motivo se puede decir que las interfaces *Bluetooth* son aptos para ser usados en *PCs* Portátiles que necesitan accesorios externos como Mouse, permitiendo así una mayor duración de la batería y otorgando así mayor tiempo de duración de trabajo sin necesidad de alimentación externa.

- *Bluetooth* presenta excelente robustez frente al ruido, lo que hace que los sistemas que trabajen con esta tecnología, sean más estables y seguros que los demás, pero por su baja velocidad de transmisión de datos hace que en la actualidad *Bluetooth* no sea muy utilizado para redes inalámbricas de corto alcance.

- Se pudo comprobar prácticamente para *Bluetooth* y para *Wi-Fi* que los parámetros de campo como; velocidad efectiva, nivel de potencia y relación señal a ruido varían directamente en función de la distancia, es decir a medida que aumentaba la distancia de separación entre las estaciones estos parámetros decrecen.

- Como era de esperarse en el enlace *Wi-Fi* a medida que se alejan las estaciones éste presenta mejor velocidad efectiva que *Bluetooth* ya que el estándar *802.11b* se caracteriza por tener mayor alcance, velocidad, que el estándar *802.15.1*

- Se pudo comprobar que las tarjetas *Wi-Fi* utilizados en el proyecto, no cumplen con el alcance especificado en las mismas, motivo por el cual estas tarjetas son muy inestables, a partir de distancias mayores a 14 metros. Esto se debe a la baja sensibilidad de las tarjetas, pues en las pruebas de medición de potencia realizadas con el analizador de espectros se observa que la potencia transmitida por el interfaz llega al receptor con un valor aceptable mayor a la sensibilidad especificada por el interfaz.

- Para la simulación de los prototipos, se utilizó el programa *Network Simulator*, que además de ser un *software* muy utilizado en propósitos similares (redes satelitales, redes inalámbricas *Ad-Hoc* e Infraestructura, etc.) por diversos organismos en casi todo el mundo que se dedican a la investigación de redes, se puede afirmar que este simulador es muy confiable al momento de realizar las distintas simulaciones pero tiene la desventaja de ser complicado al momento de instalarlo y realizar las distintas simulaciones.

- En la simulación se pudo verificar que el enlace *Bluetooth* es más robusto que *Wi-Fi* frente al ruido. Esto se debe ya que *Bluetooth* utiliza un ancho de banda pequeño con respecto a *Wi-Fi* y es más inmune a la interferencia.

5.2 RECOMENDACIONES

- Una vez culminado el proyecto y realizadas las distintas pruebas que abarcaron el desarrollo del mismo, se recomienda implementar redes inalámbricas con tecnología *Bluetooth* en oficinas, lugares cerrados en general que tengan un diámetro no mayor a diez metros y en donde la velocidad de transmisión no sea importante, ya que Bluetooth en distancias menores o iguales a esta, tiene un buen desempeño en lo que corresponde a nivel de potencia, pérdida de datos.

- También se recomienda utilizar la tecnología *Bluetooth* en sistemas inalámbricos en los cuales la pérdida de datos deba ser mínima al momento de transmitir, ya que *Bluetooth* presenta una pérdida de datos mínima. Como es el caso de sensores inalámbricos.

- Una recomendación final con respecto a la implementación de sistemas inalámbricos con tecnología *Bluetooth*, es que *Bluetooth* en la actualidad puede interactuar con dispositivos móviles, celulares, impresoras, *mouse*, *PDAs*, etc. por lo que es recomendable implementar esta tecnología en lugares que posean este tipo de dispositivos.

- Se recomienda utilizar *Wi-Fi* en sistemas inalámbricos en los cuales no sea importante la pérdida de datos al momento de la transmisión de estos y la tasa de transferencia sea alta, ya que esta tecnología posee una pérdida de datos considerable a distancias mayores a los doce metros, pero a su vez posee alta velocidad de transmisión de datos.

- Se recomienda descargar el paquete completo del programa allinone del ns e instalarlo en su totalidad, para evitar futuros problemas en el momento de ejecutar sus librerías. Se puede obtener el programa simple e ir instalando las librerías una por una de acuerdo a las necesidades que se tenga.

- Se recomienda en un proyecto futuro se modifiquen las librerías del *UCBT* para lograr el posicionamiento y movimiento de los nodos ya que en el desarrollo de este trabajo se logró posicionar los nodos pero no dar movimiento a estos.

BIBLIOGRAFÍA

[1] Jorge Alfonso Briones García, “Un *middleware* para el desarrollo de aplicaciones en redes espontáneas de dispositivos móviles Bluetooth”, Tesis de Maestría, Centro de Investigación y de Estudios Avanzados del IPN Departamento de Ingeniería Eléctrica Sección de Computación.

[2] Néstor García Fernández, “Modelo de Cobertura en Redes Inalámbricas basado en Radiosidad por Refinamiento Progresivo”, Tesis Doctoral, Universidad de Oviedo, marzo 2006

[3] Msc. Soraya Sinche M. “Apuntes de Comunicaciones Inalámbricas”, Escuela Politécnica Nacional, Marzo-Agosto 2005.

[4] PhD. Iván Bernal,” Apuntes de Comunicaciones Inalámbricas”, Escuela Politécnica Nacional, Diciembre 2005

[5] BLUETOOTH SPECIAL INTEREST GROUP. “Bluetooth Core”, Specification of the Bluetooth System, Version 1.1, 22 de febrero de 2003

[6] Carlos García García, Doctorado PCSM “Protocolo de Comunicaciones para Sistemas Móviles”

[7] Ing. Pablo Hidalgo, “Apuntes de Telemática”, Escuela Politécnica Nacional Marzo 2005

[8] Mari Carmen Domingo, “Diferenciación de servicios y mejora de la supervivencia en redes Ad-hoc conectadas a redes fijas”, Tesis Doctoral, Universidad Politécnica de Cataluña, Año 2006

[9] Andrew S. Tanenbaum,”Redes de Computadoras”, Tercera edición, Prentice-Hall, páginas 263-265, 1997.

[10] Wayne Tomasi, “Sistemas de Comunicaciones Electrónicas”, Pearson Educación de México, Cuarta Edición 2003

[11] Msc. Maria Soledad Jiménez Jiménez,” Apuntes de Comunicación Digital”, Escuela Politécnica Nacional, 2004.

[12] A.J. Motley, J.M. Keeman, “Radio Coverage in Buildings”. British Telecom, Technology Journal, Vol. 8, Enero 1990.

[13] Oscar Darío Rodríguez Calvachi y Ricardo Andrés Maya Coral Implementación de una red Inalámbrica Bluetooth”, Tesis, Universidad del Valle, Santiago de Cali, 2003.

[14] Roberto Carlos Ortega y Wladimir Valdivieso, Diseño e Implementación de un acceso inalámbrico e interfaz al sistema de administración estudiantil (SAE) y biblioteca, basado en la tecnología Bluetooth, ubicado en el edifico antiguo de Ingeniería Eléctrica

[15] http://www.mastermagazine.info/articulo/3125.php

[16] http://www:bluetooth.com/Bluetooth/Learn/

[17] http://www.zonabluetooth.com/que_es_bluetooth2.htm

[18] http://www:catarina.udlap.mx/capitulo4.pdf

[19] http://www.senacitel.cl/downloads/senacitel2000/IDO33.pdf

[20] http://www.di.uniovi.es/investigacion/tesis/Nestor.pdf

[21] http://www.isi.edu/nsnsm/ns/.

[22] http://www.isi.edu/nsnam/ns/tutorial/index.html

[23] http://www.nile.wpi.edu/NS/

[24] http://www.bluetooth.com/Bluetooth/Learn/Security/

[25] http://www.haking9.org

[26] http://www.Wireless LAN Alliance

[27] http://www.eveliux.com

[28] http://www.isi.edu/nsnam/ns/ns-documentation.html

[29] http://www.ns-allinone-2.1b8a/ns-2.1b8a/tcl/lib

[30] http://www.expansys.es

[31] http://www.isi.edu/nsnam/

[32] http://www.ececs.uc.edu/~cdmc/ucbt/

[33] http://www.bluetooth.com/sig/membership/membership.asp

[34] http://bluehack.elhacker.net/proyectos/bluesec/bluesec.html

[35] http:// www.guw.cl/foros/

[36] INTCOMEX DEL ECUADOR S.A.

[37] http://www.dooyoo.es/targetas-de-red/anycom-blue-usb-120/precios/

[38] http://www.preciomania.com/search_attrib.php/vededorlds%5B%

[39] http://www.air802.com/home.php

[38] http://www.pixmania.com/es/es/597652/linksys/adaptador-usb-wifi-54-mb.html

GLOSARIO DE TÉRMINOS

GLOSARIO

1SM:	Modelo de una Sola Pendiente
ACL:	Enlace Asíncrono no Orientado a la Conexión
AP:	Punto de Acceso
AT- Commands:	Comandos de Telefonía
Backoff:	Espera aleatoria antes de transmitir un paquete
BD_ADDR:	Dirección del Dispositivo *Bluetooth*
BSS:	Conjunto de Servicios Básicos
CAC:	Código de Acceso al Canal
CCK	Modulación de Código Complementario
CFP:	Periodo Libre de Contención
CHIP:	Circuito Integrado
CID:	Identificador de Canal Lógico
CP:	Periodo de Contención
CRC:	Código de Redundancia Cíclica
CSMA/CA:	Acceso Múltiple con Escucha de Portadora Evitando Colisiones
CTS:	Limpieza de Envío
CW:	Ventana de Contienda
DAC:	Código de Acceso de Dispositivo
dBi:	Decibelios Isotrópicos
DCF:	Función de Coordinación Distribuida
DIFS:	Espaciado entre Tramas Distribuido
DS:	Sistema de Distribución
DS:	Envío de Datos
DSSS:	Espectro Expandido por Secuencia Directa
DBPSK	Modulación por Desplazamiento de Fase Bivalente Diferencial
DQPSK	Modulación por Desplazamiento de Fase Cuadrivalente Diferencial
EAP:	Protocolo de Autenticación Extensible
ESS:	Conjunto de Servicios Extendidos
ETSI:	Instituto de Estándares de Telecomunicaciones de Europa
FCC:	Comisión Federal de Comunicaciones
FEC:	Corrección Directa de Errores
FH:	Salto de Frecuencia

FHS:	Paquete de Sincronización
FHSS:	Espectro Expandido por Salto de Frecuencia
GFSK:	Modulación por Desplazamiento de Frecuencia Gausiana
GOEP:	Perfil Genérico de Intercambio de Objetos
HCI:	Interfaz de Controlador de Host
IAC:	Código de Acceso de Búsqueda
IBSS:	Independiente BSS
ID:	Paquete de Identificación
IEEE:	Instituto de Ingenieros Electrónicos y Eléctricos
IFS:	Espaciado entre Tramas
IP:	Protocolo de Internet
ISM:	Industrial, Científica y Médica
ITU:	Unión Internacional de Telecomunicaciones
IV:	Vector de Inicialización
IVA:	Impuesto de Valor Agregado
L2CAP:	Protocolo de Control y Adaptación de Enlace Lógico
LAN:	Red de Área Local
LAP:	Puntos de Acceso LAN
LC:	Control de Enlace
LEAP:	EAP Liviano
LLC:	Control de Enlace lógico
LM:	Enlace de Administración
LMP:	Protocolo de Administración del Enlace
M_ADDR:	Dirección que identifica a los esclavos activos en una *piconet*
MAC:	Control de Acceso al Medio
MACA:	Acceso Múltiple Evitando Colisiones
MSDU:	Unidad de Datos de Servicio MAC
NAV:	Vector de Localización de Red
NIC:	Tarjetas de Interfaz de Red
OBEX:	Protocolo para el Intercambio de Objetos
OFDM:	Multiplexación por División de Frecuencia Ortogonal
PC:	Computador Personal
PCF:	Función de Coordinación Puntual

PCs:	Computadora
PDA:	Asistentes Personales Digitales
PDU:	Unidades de Datos de Protocolo
PIM:	Información de Administración Personal
PIN:	Numero de identificación Personal
PPP:	Protocolo Punto a Punto
RADIUS:	Servicio de Usuario de Autenticación Remota
RF:	Radiofrecuencia
RFCOMM:	Protocolo para Emulación de Puertos Seriales sobre *L2CAP*
RTS:	Preguntar para Enviar
S/N:	Relación Señal a Ruido
SCO:	Enlace Sincrónico Orientado a Conexión
SDP:	Protocolo de Descubrimiento de Servicio
SIFS:	Espaciado entre Tramas Corto
SIG:	Grupo de Interés Especial
TCP:	Protocolo de Control de Transmisión
TCS:	Especificación de Control Telefónico
TDD:	Duplexación por División de Tiempo
TKIP:	Protocolo de Integridad de Claves Temporales
UA:	Datos Asíncronos de usuario
UDP:	Protocolo de Datagrama de Usuario
UI:	Datos Isócrono de Usuario
UMTS:	Sistema de Telecomunicaciones Móbiles Universal
UNII:	Infraestructura de Información Nacional sin Licencia
US:	Datos Síncrono de Usuario
vCard:	Visualizador de tarjetas virtuales
WAE:	Entorno Inalámbrico de Aplicación
WAP:	Protocolo de Aplicación Inalámbrica
WEP:	Privacidad Equivalente a la Cableada
Wi-Fi	Fidelidad Inalámbrica
WLAN:	Red de Área Local Inalámbrica
WPA:	Protocolo de Aplicación Inalámbrica

ANEXOS

ANEXO C

Módulos del ns-2 [21]

Módulos del ns-2

ns-2 dispone de varios módulos, a continuación se describirán los más importantes y aquellos que han sido utilizados en este proyecto.

- **Simulador ns**

Este módulo está desarrollado en C++, dispone de un núcleo principal, donde están definidos todos los protocolos. Por otra parte, los scripts donde se configuran los escenarios de la simulación se deben programar en el lenguaje OTcl, que al igual que C++ también es un lenguaje de programación orientado a objetos. Por lo tanto, para desarrollar las simulaciones en el ns-2 es necesario programar en los dos lenguajes.

C++ es más rápido al momento de ejecutar las simulaciones, pero tiene la desventaja que es más lento al momento de modificar que OTcl, entonces para ejecutar protocolos es mejor utilizar C++. Para la generación de los scripts es más fácil utilizar OTcl ya que este es más rápido y fácil de modificar.

El simulador se invoca introduciendo en la línea de comandos el siguiente comando.

./ns nom_fichero.tcl

Las simulaciones bajo ns-2 se basan en scripts programados por el propio usuario, se puede definir la topología de red y las características de los enlaces entre los nodos.

Los resultados obtenidos en la simulación se pueden visualizar utilizando las herramientas **nam** o **xgraph**.

- **Nam (*Network Animator*)**

Permite representar gráficamente la red diseñada, visualizar dinámicamente los resultados de la simulación realizada. Para ejecutar el nam se escribe en la línea de comandos el siguiente comando.

nam nombre_fichero.nam

En la figura 3.1 se puede ver la captura de un ejemplo de simulación realizada con el nam.

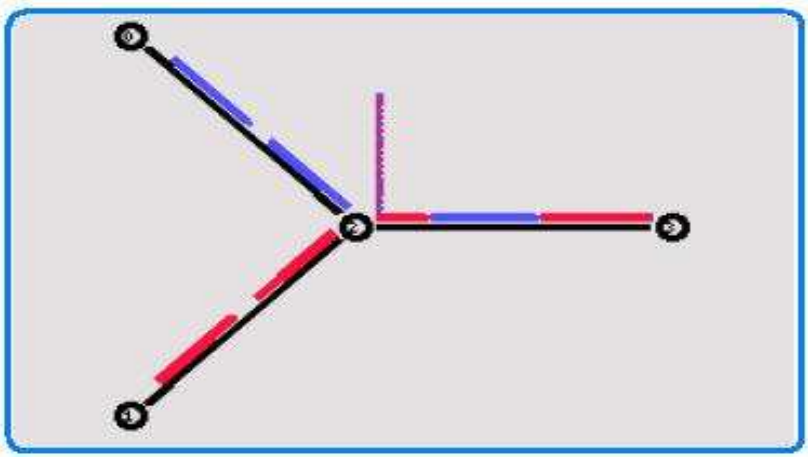


Figura 1 Simulación con nam

- **Xgraph**

Xgraph es una herramienta de graficación proporcionada por el ns-2. Además permite crear archivos postscript, Tgif, y otros. Puede ser invocado dentro de los comandos tcl para desplegar inmediatamente después de que la simulación haya concluido.

El comando xgraph espera como entradas uno o más archivos ASCII que contengan datos en forma de pares ordenados x-y en cada línea. Por ejemplo, xgraph f1 f2 imprimirá en la misma figura los archivos f1 y f2.
Algunas opciones que posee el xgraph son:

- Título: -t “título”

- Tamaño: -geometry xsize x ysize.
- Títulos de ejes: -x “xtitle” (para el título del eje x) y -y “ytitle” (para el título del eje y).
- Color de texto y grilla: con la bandera -v.

A continuación se muestra como se debe introducir el comando para realizar la graficación mediante el xgraf:

xgraph f1 f2 -geometry 800x400 -t “Loss rates” -x “time” -y “Lost packets”

Un ejemplo de uso del Xgraph, se indica en la figura 3.2, de acuerdo al siguiente código:

xgraph out.tr -geometry 800x400 -t "Ejemplo de uso del Xgraph" -x "tiempo" -y Paquetes perdidos

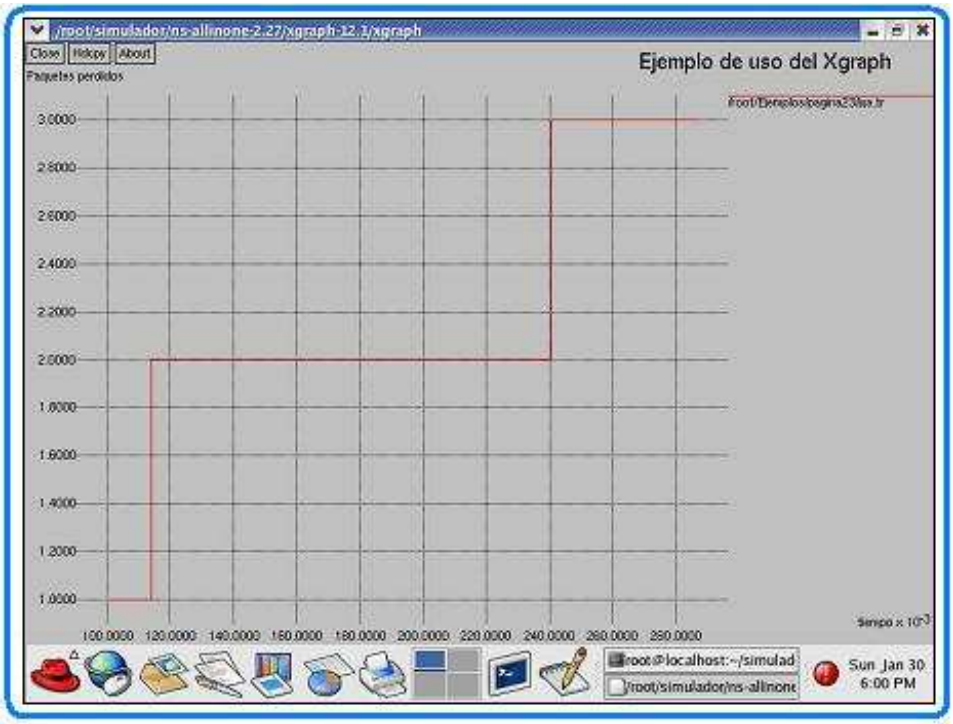


Figura 2 Ejemplo de uso del XGraph.

- **Gnuplot**

Gnuplot es un programa muy flexible creado para generar gráficas de funciones y datos. Este programa es compatible con muchos sistemas operativos como son

(*Linux, Unix, Windows,*). Gnuplot puede graficar sus resultados directamente en pantalla, así como en multitud de formatos de imagen, como PNG, EPS, SVG, JPEG, etc. Gnuplot se puede usar en forma interactiva o en modo por lotes usando scripts.

Para acceder al *gnuplot,* se ejecuta el siguiente comando.

Gnuplot
Plot nom_fichero.tr

La figura 3.3 representa en ejemplo de graficación utilizando gnuplot

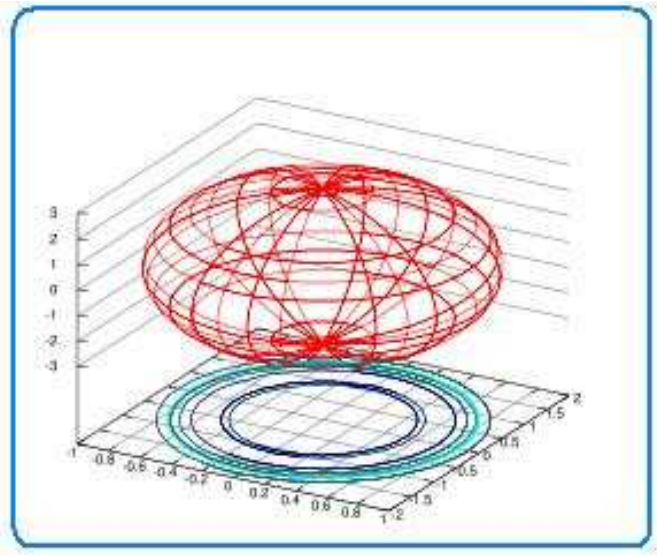


Figura 3 Ejemplo de gráfica utilizando gnuplot

ANEXO D

Pruebas Prácticas Bluetooth

Pruebas Prácticas de Bluetooth

Medidas a 2 metros de separación

Señal detectada

Figura 1 Estado de conexión 2m

Nivel de potencia (d = 2 m, p = - 55.67 dBm, f = 2.4747 GHz)

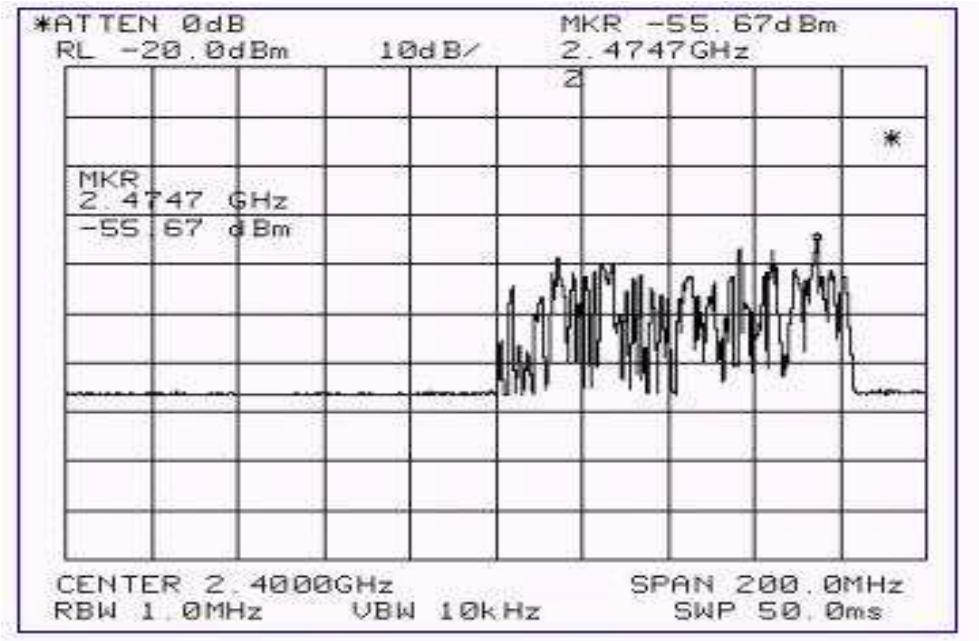


Figura 2 Nivel de potencia 2m

Respuesta entre las estaciones

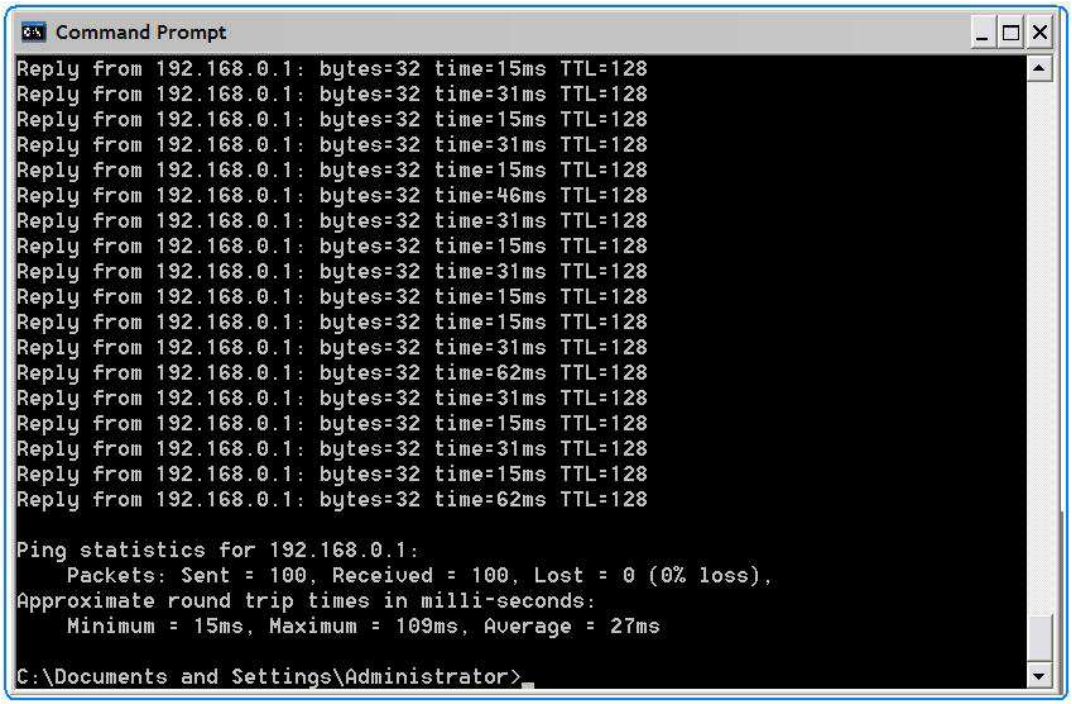


Figura 3 Ping entre las estaciones 2m

Velocidad

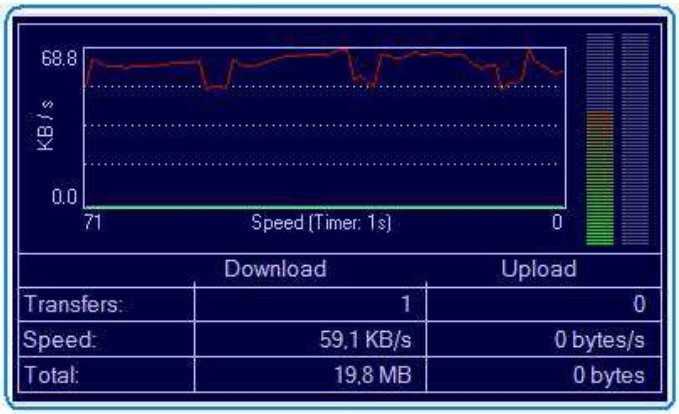


Figura 4 Velocidad 2m

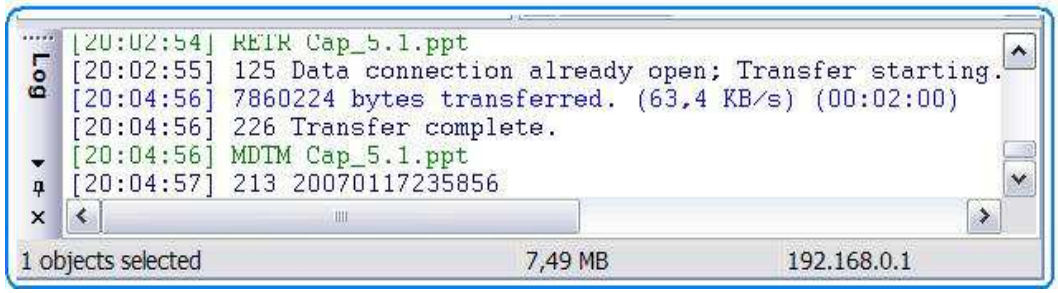


Figura 5 Velocidad Promedio 2m

Medidas a 3 metros de separación

Señal detectada

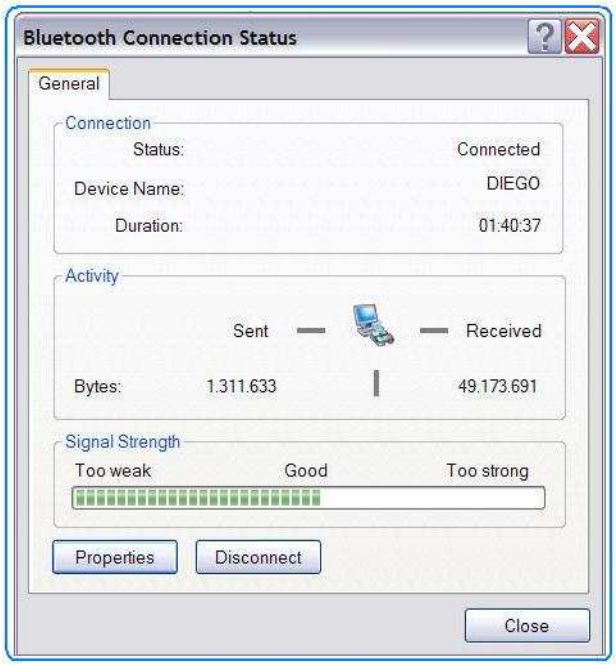


Figura 6 Estado de conexión 3m

Nivel de potencia (d = 3 m, p = - 58.83 dBm, f = 2.4740 GHz)

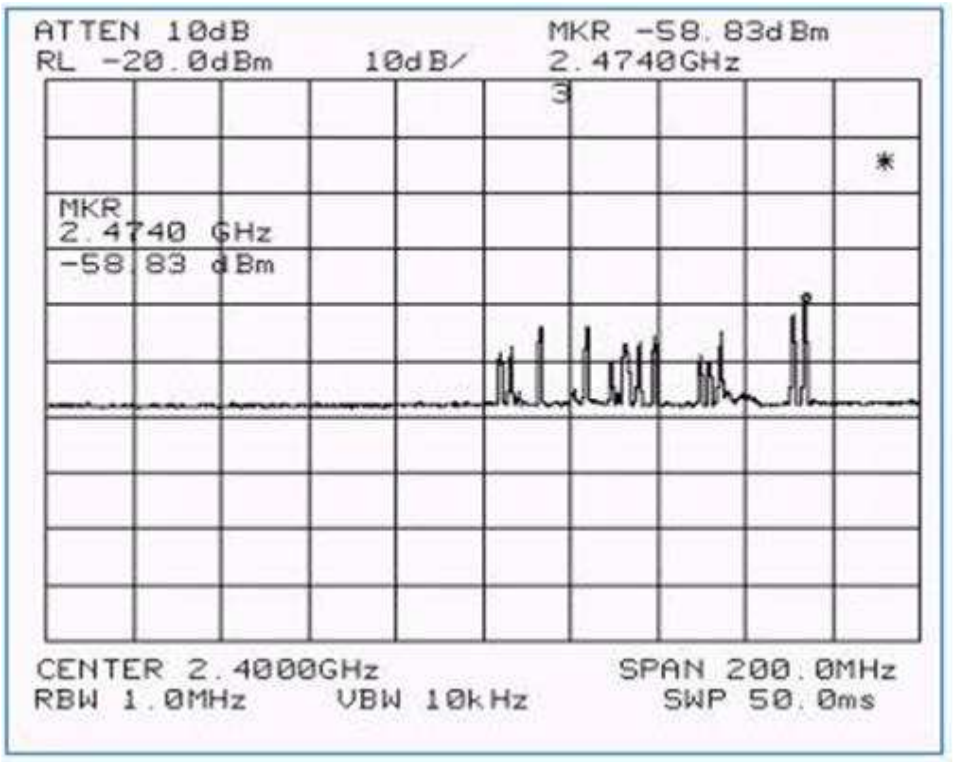


Figura 7 Nivel de potencia 3m

Respuesta entre las estaciones

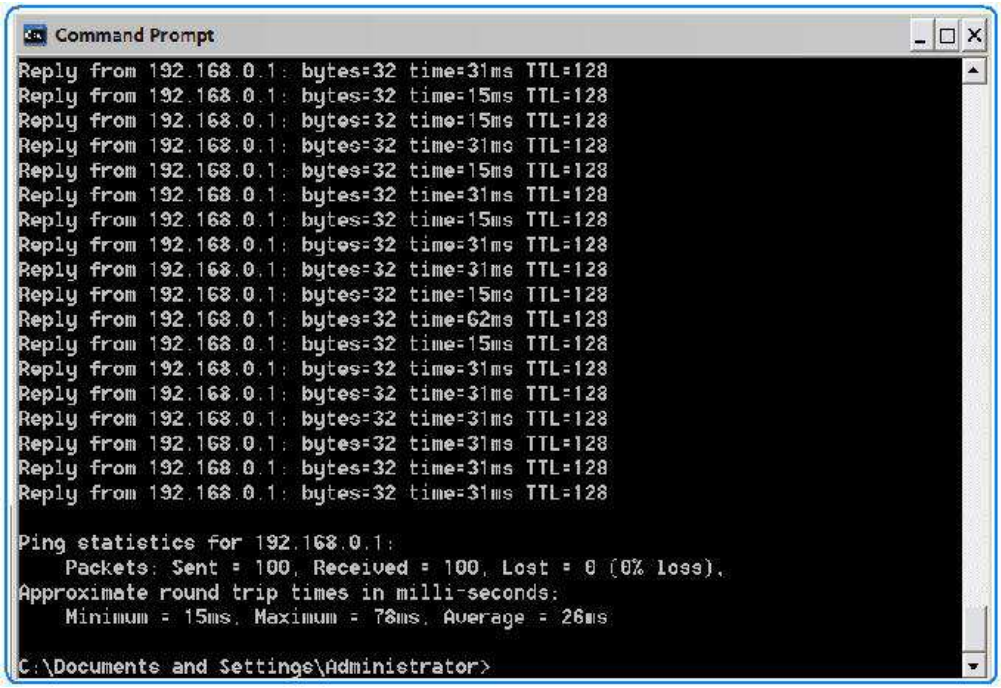


Figura 8 Ping entre las estaciones 3m

Velocidad

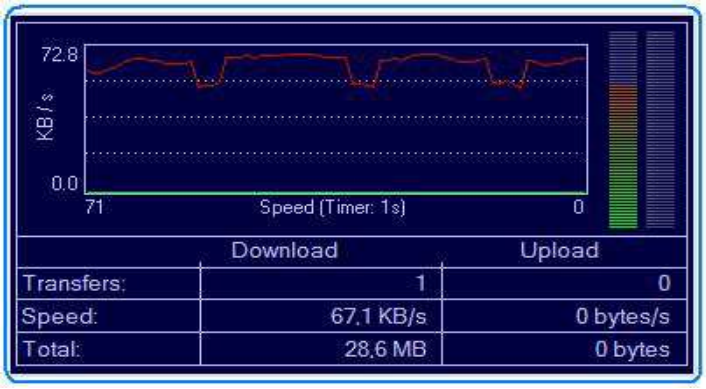


Figura 9 Velocidad 3m

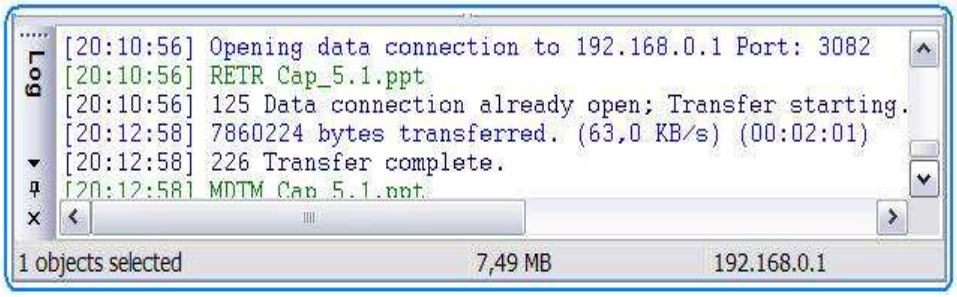


Figura 10 Velocidad Promedio 3m

Medidas a 4 metros de separación

Señal detectada

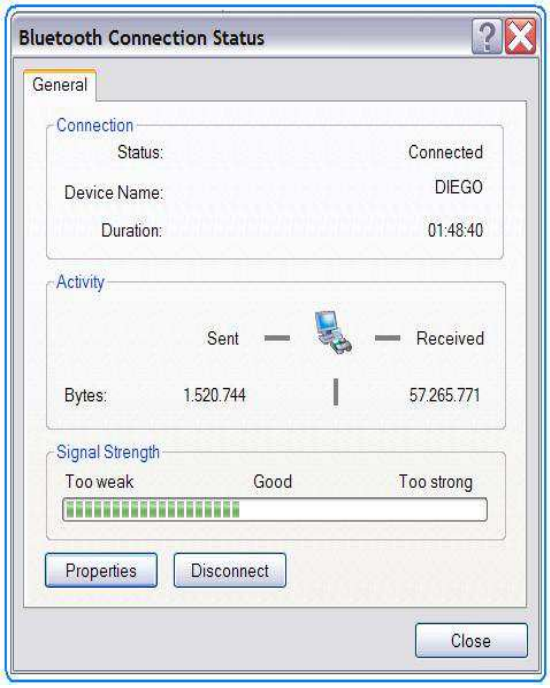


Figura 11 Estado de conexión 4m

Nivel de potencia (d = 4 m, p = - 56.83 dBm, f = 2.4537 GHz)

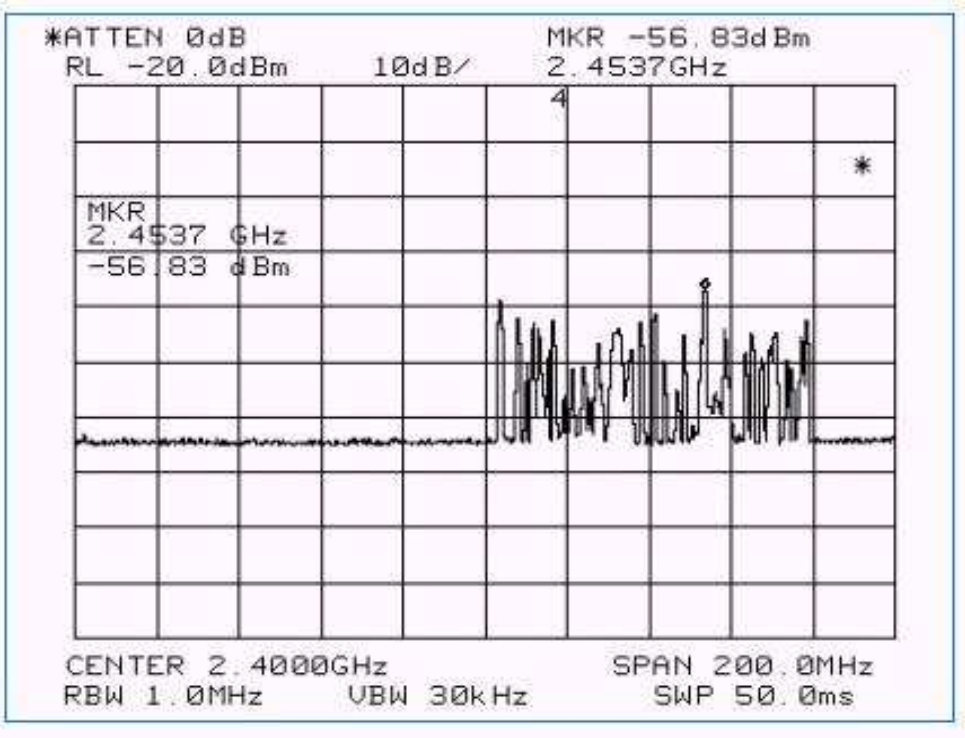


Figura 12 Nivel de potencia 4m

Respuesta entre las estaciones

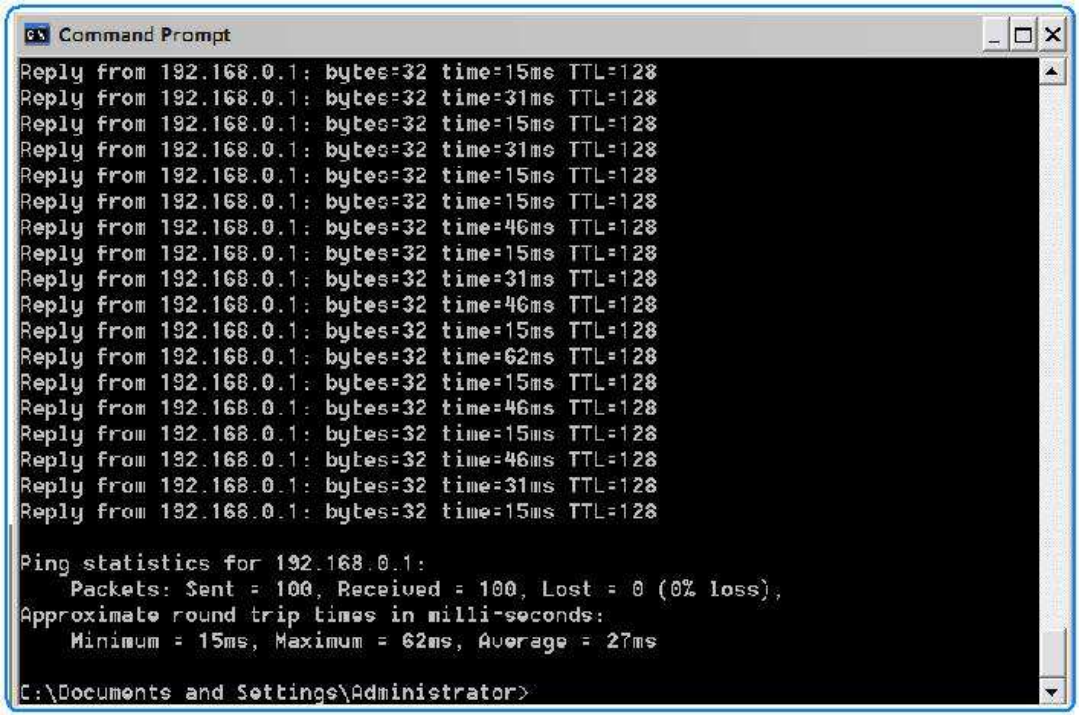


Figura 13 Ping entre las estaciones 4m

Velocidad

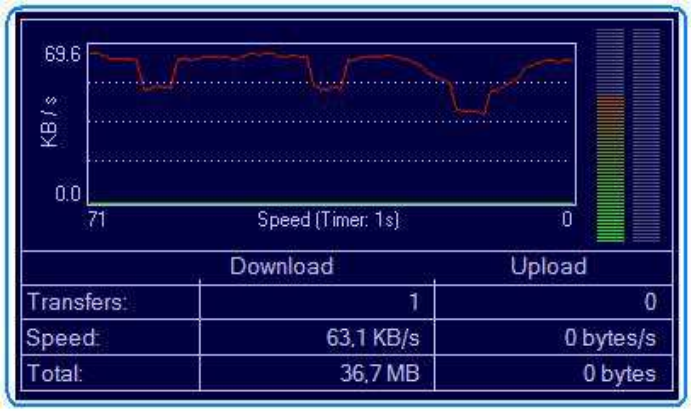


Figura 14 Velocidad 4m

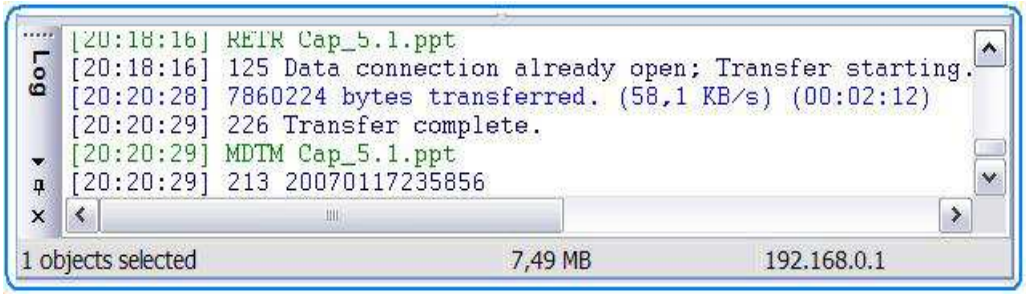


Figura 15 Velocidad 4m

Medidas a 5 metros de separación

Señal detectada

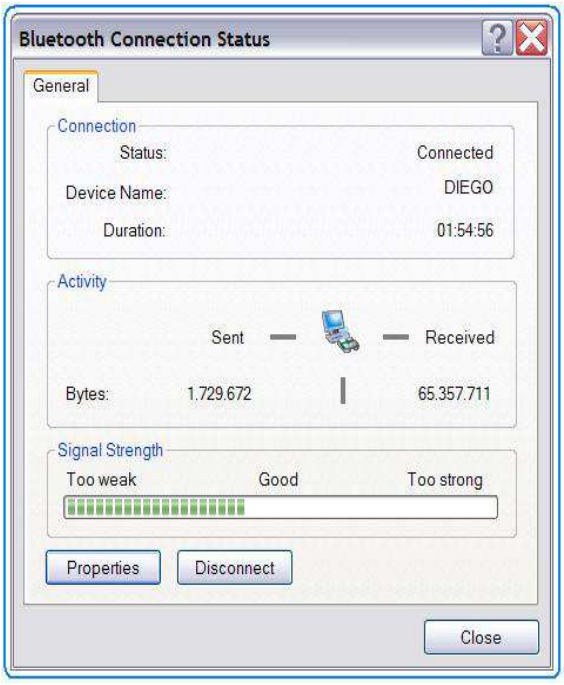


Figura 16 Estado de conexión 5m

Nivel de potencia (d = 5 m, p = - 57.17 dBm, f = 2.4557 GHz)

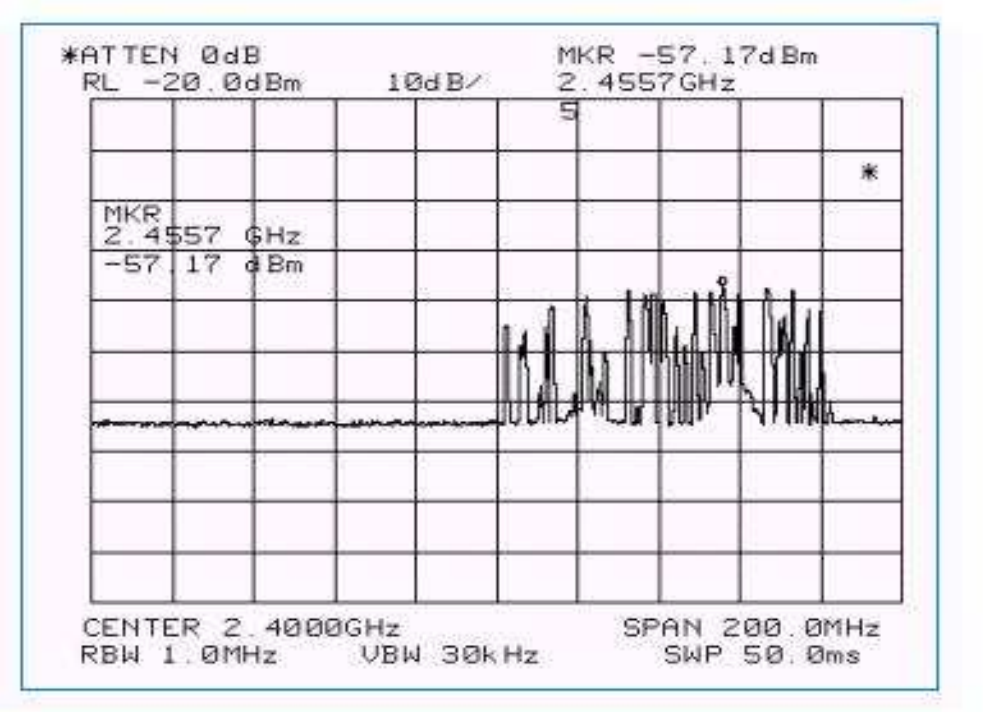


Figura 17 Nivel de potencia 5m

Respuesta entre las estaciones

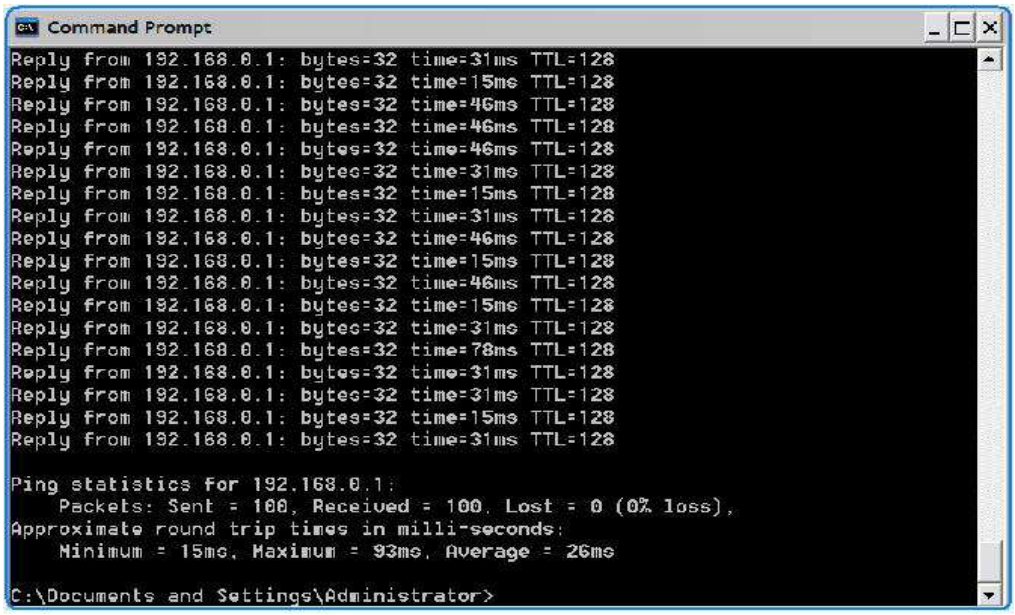


Figura 18 Ping entre las estaciones 5m

Velocidad

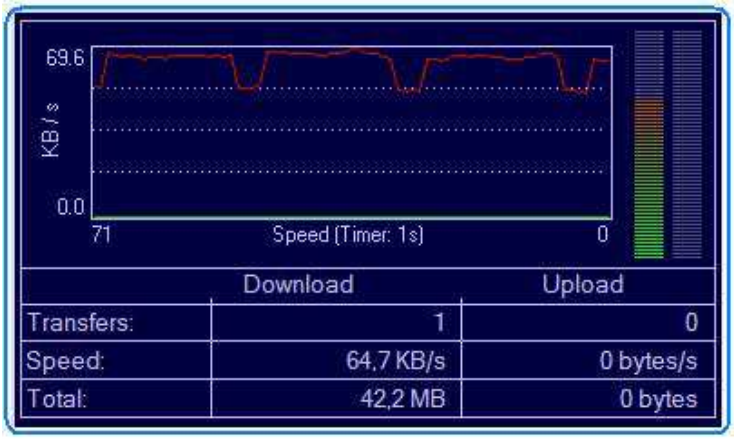


Figura 19 Velocidad 5m

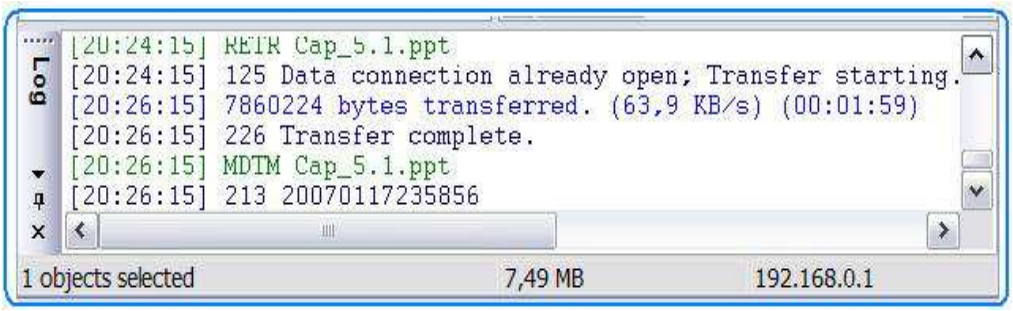


Figura 20 Velocidad Promedio 5m

Medidas a 6 metros de separación

Señal detectada

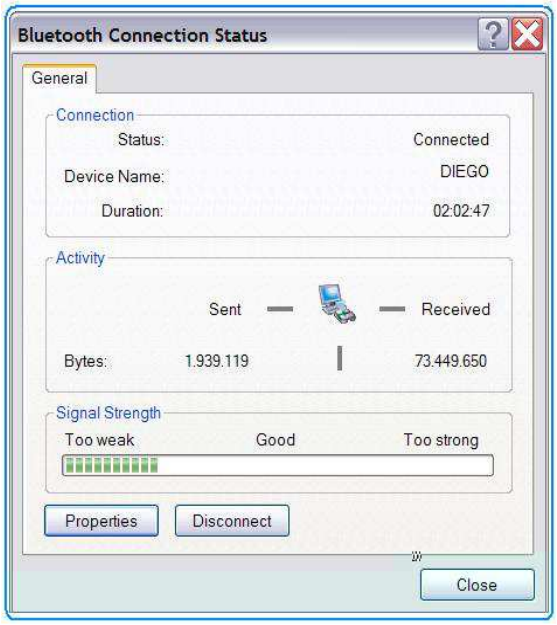


Figura 21 Estado de conexión 6m

Nivel de potencia (d = 6 m, p = - 56.83 dBm, f = 2.4420 GHz)

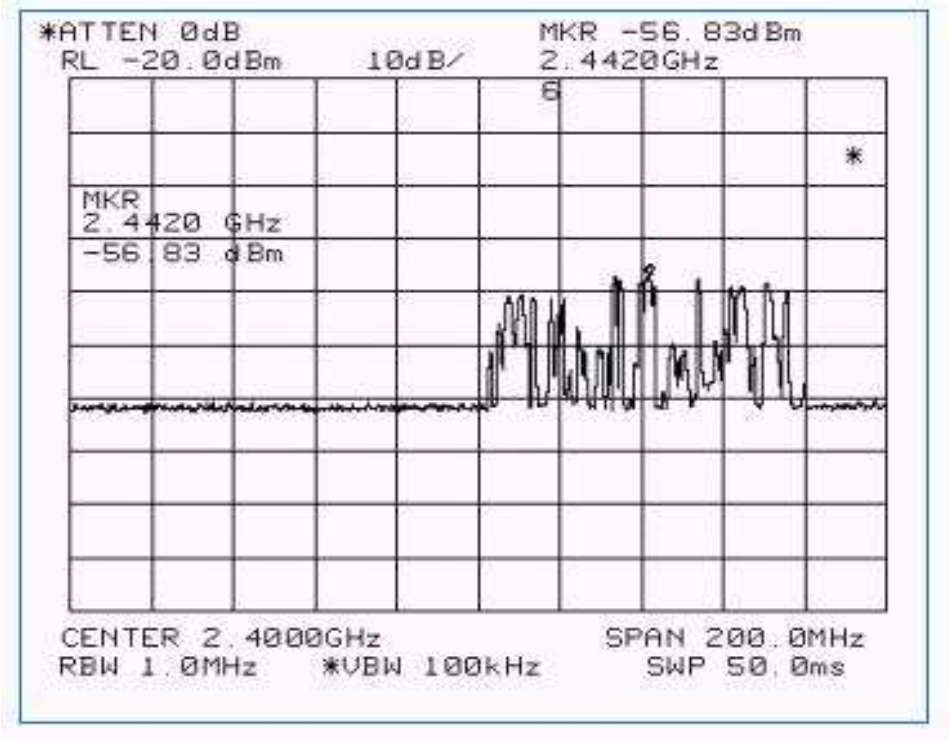


Figura 22 Nivel de potencia 6m

Respuesta entre las estaciones

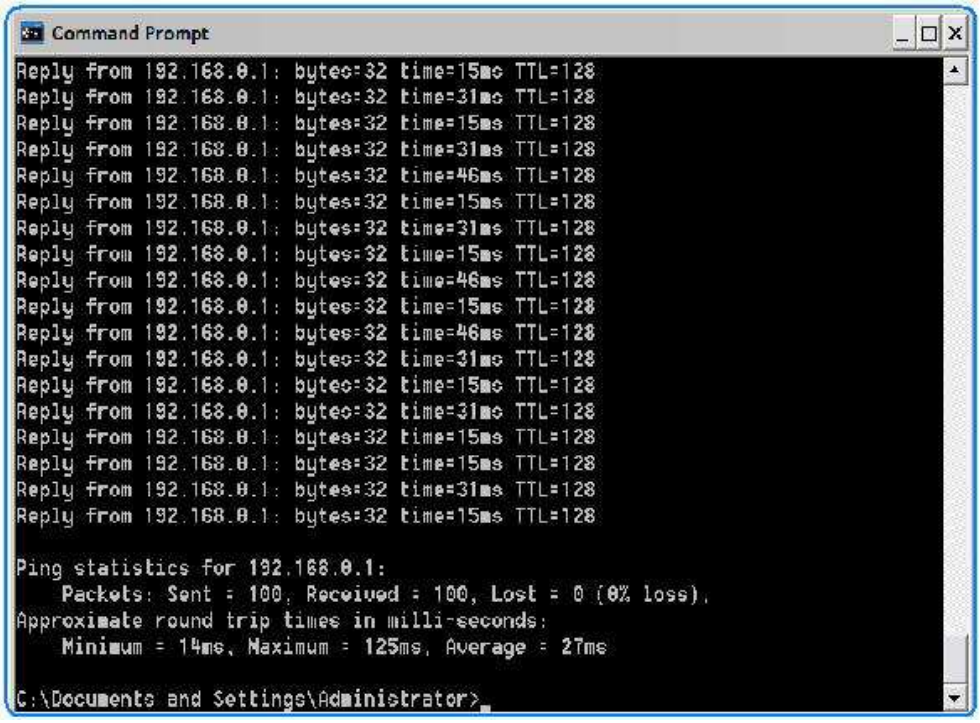


Figura 23 Ping entre las estaciones 6m

Velocidad

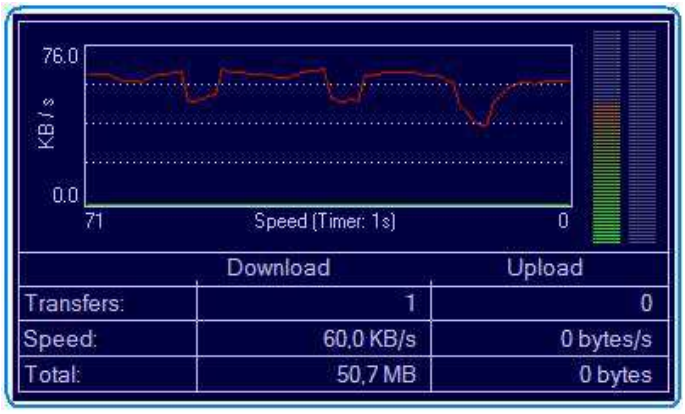


Figura 24 Velocidad 6m

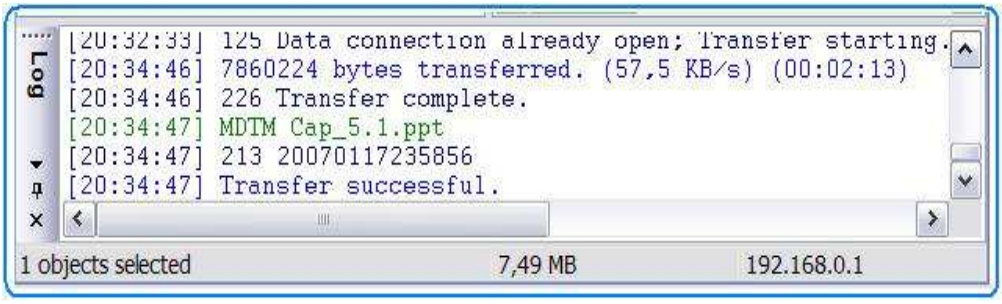


Figura 25 Velocidad Promedio 6m

Medidas a 7 metros de separación

Señal detectada

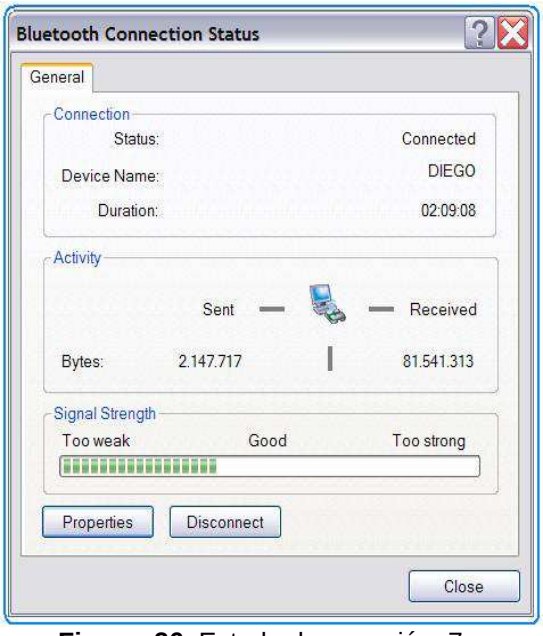


Figura 26 Estado de conexión 7m

Nivel de potencia (d = 7 m, p = - 58.17 dBm, f = 2.4287 GHz)

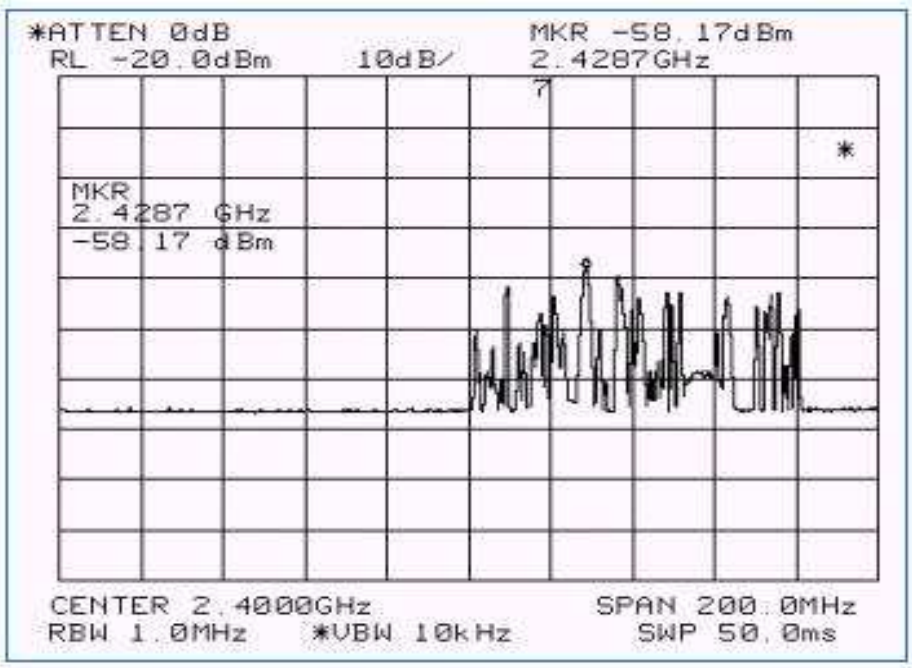


Figura 27 Nivel de potencia 7m

Respuesta entre las estaciones

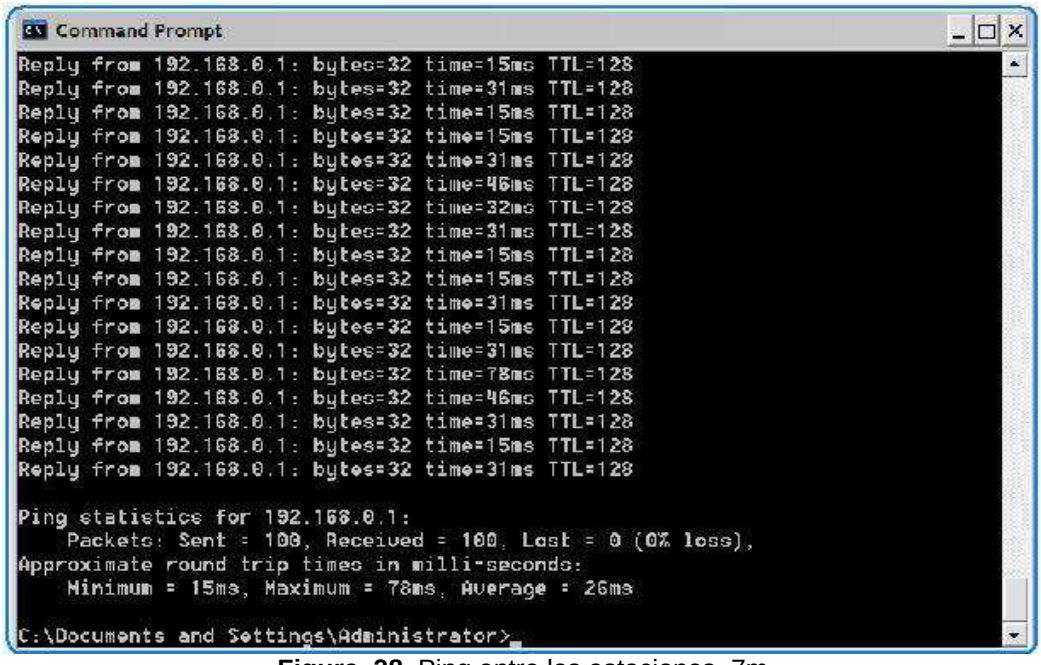


Figura 28 Ping entre las estaciones 7m

Velocidad

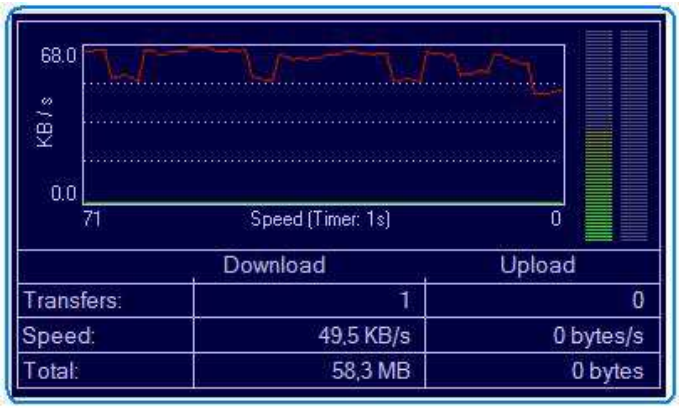


Figura 29 Velocidad 7m

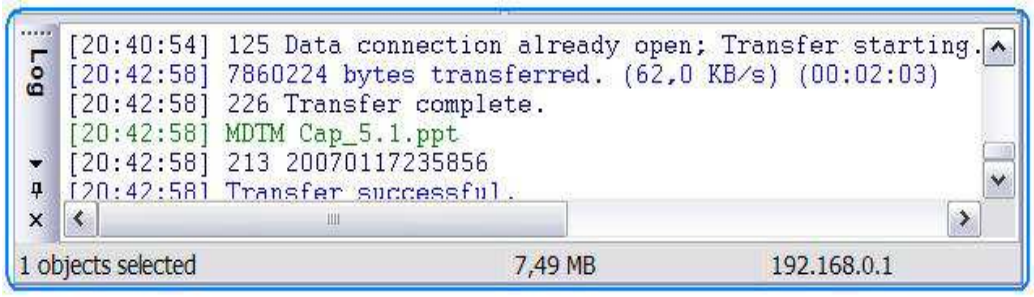


Figura 30 Velocidad Promedio 7m

Medidas a 8 metros de separación

Señal detectada

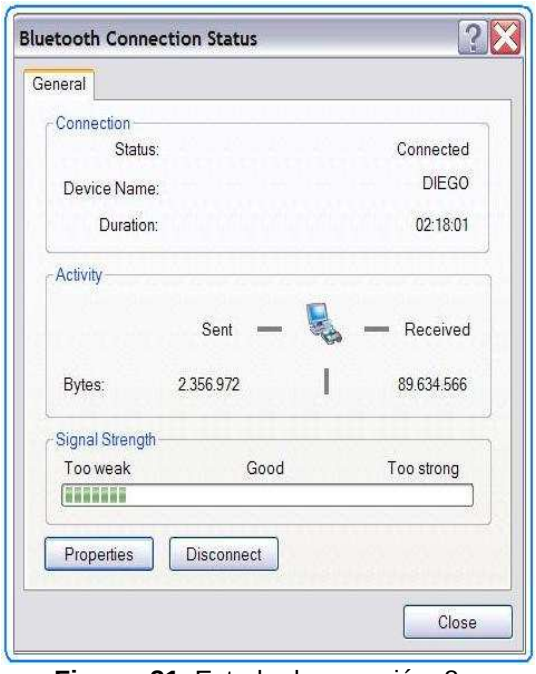


Figura 31 Estado de conexión 8m

Nivel de potencia (d = 8 m, p = - 58.00 dBm, f = 2.4277 GHz)

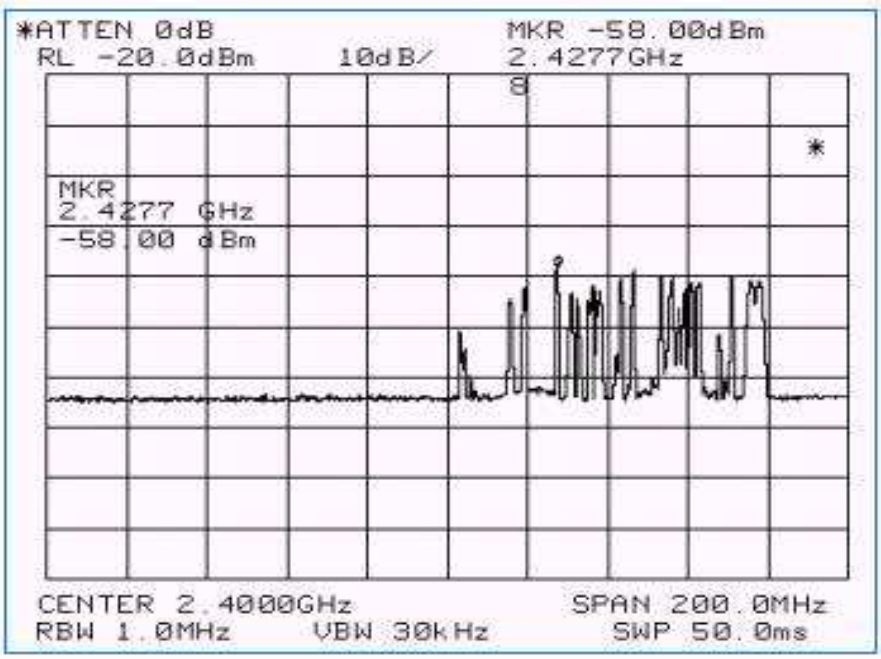


Figura 32 Nivel de potencia 8m

Respuesta entre las estaciones

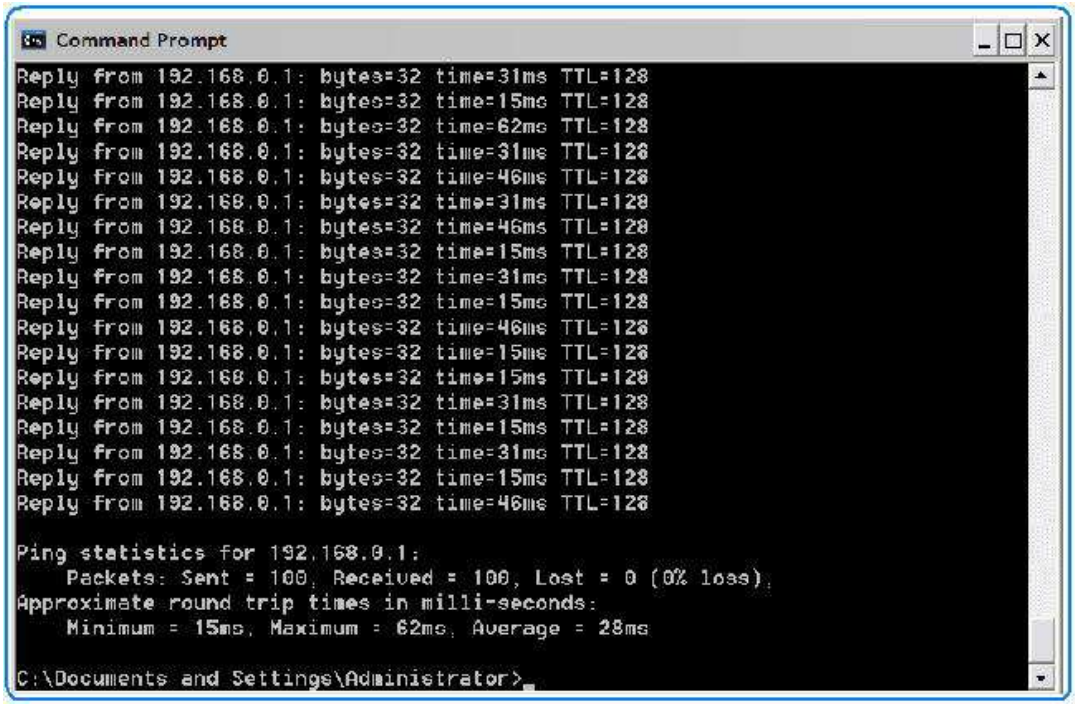


Figura 33 Ping entre las estaciones 8m

Velocidad

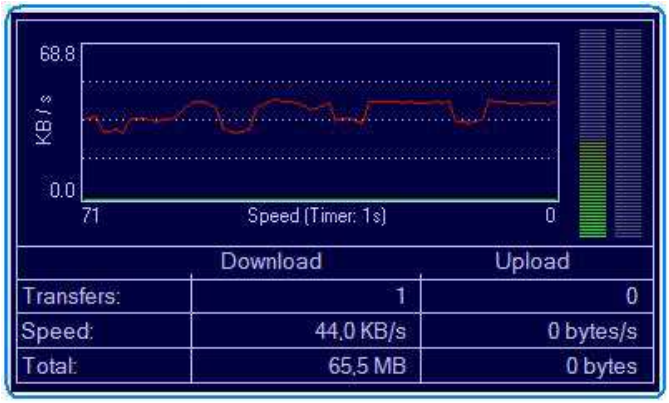


Figura 34 Velocidad 8m

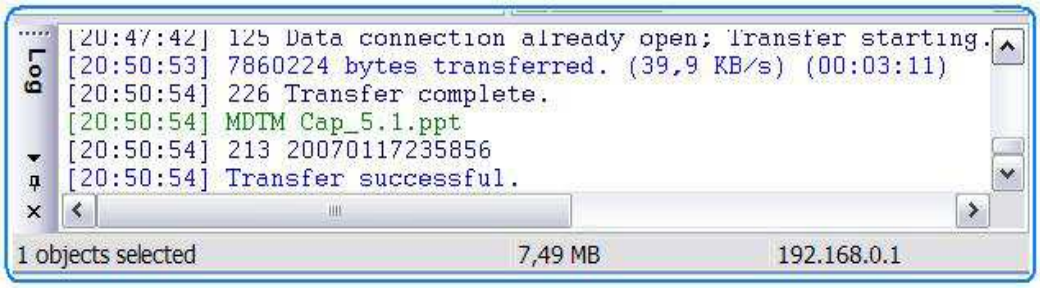


Figura 35 Velocidad Promedio 8m

Medidas a 9 metros de separación

Señal detectada

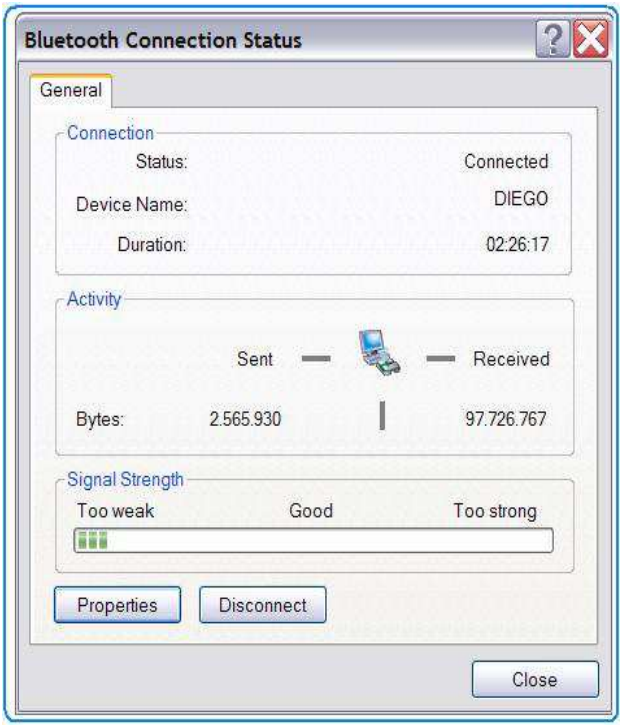


Figura 36 Estado de conexión 9m

Nivel de potencia (d = 9 m, p = - 57.33 dBm, f = 2.4177 GHz)

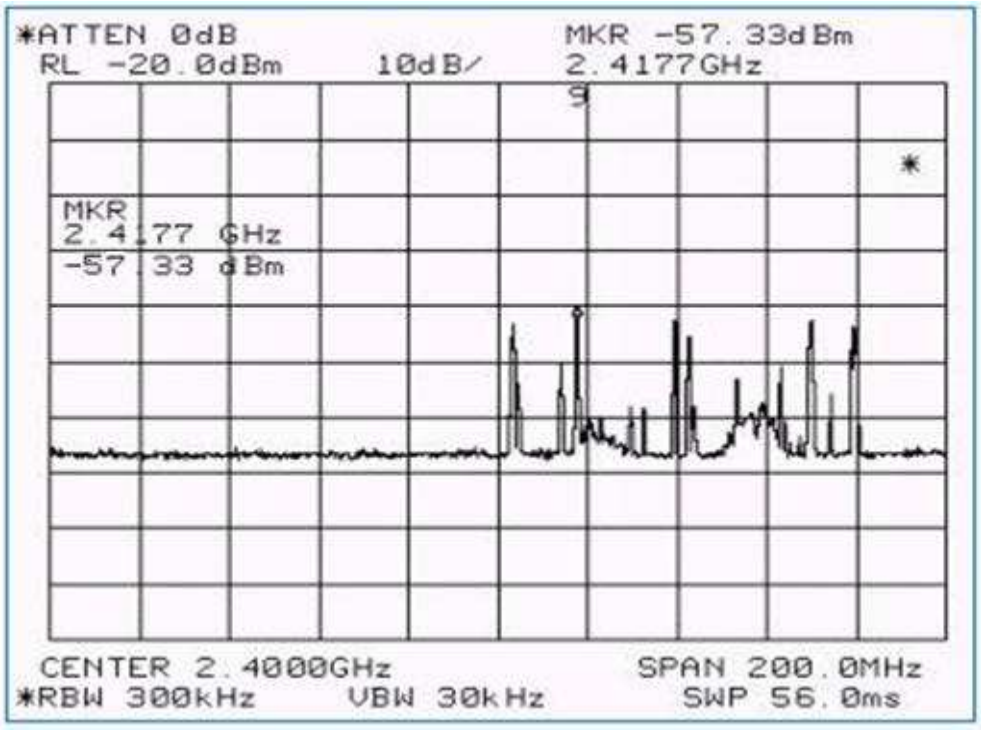


Figura 37 Nivel de potencia 9m

Respuesta entre las estaciones

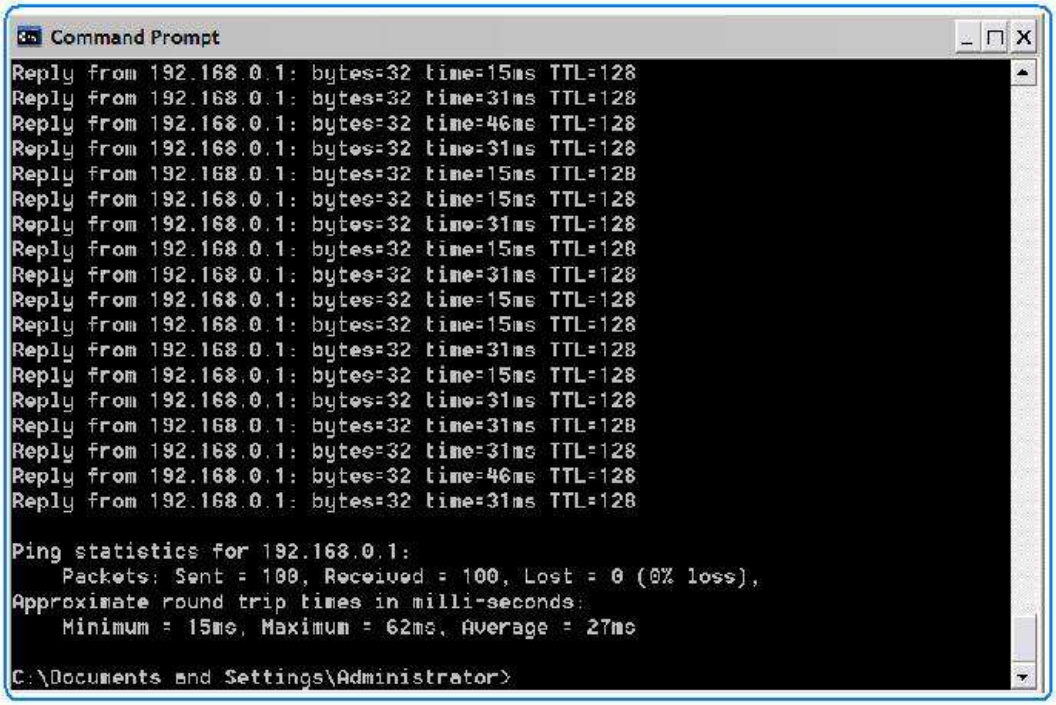


Figura 38 Ping entre las estaciones 9m

Velocidad

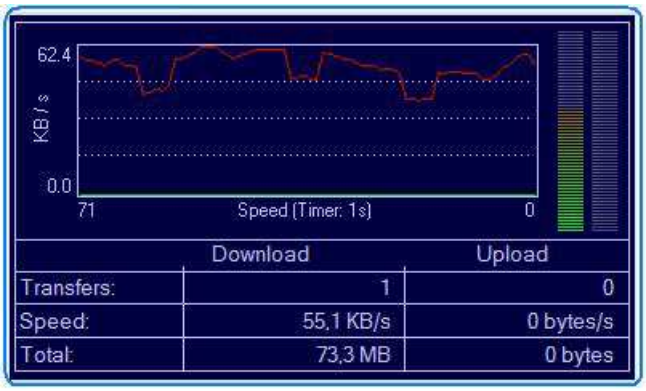


Figura 39 Velocidad 9m

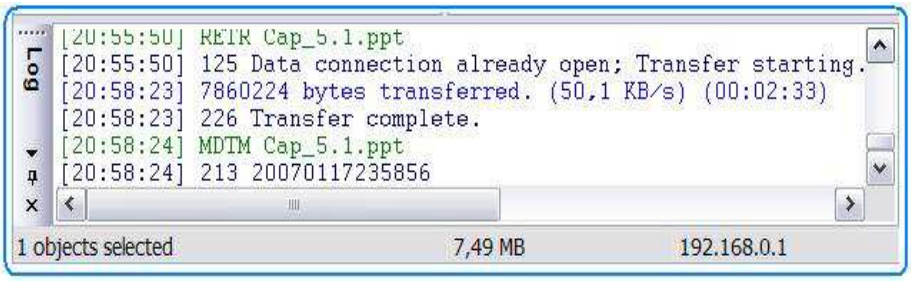


Figura 40 Velocidad Promedio 9m

Medidas a 10 metros de separación

Señal detectada

Figura 41 Estado de conexión 10m

Nivel de potencia (d = 10 m, p = - 58.50 dBm, f = 2.4307 GHz)

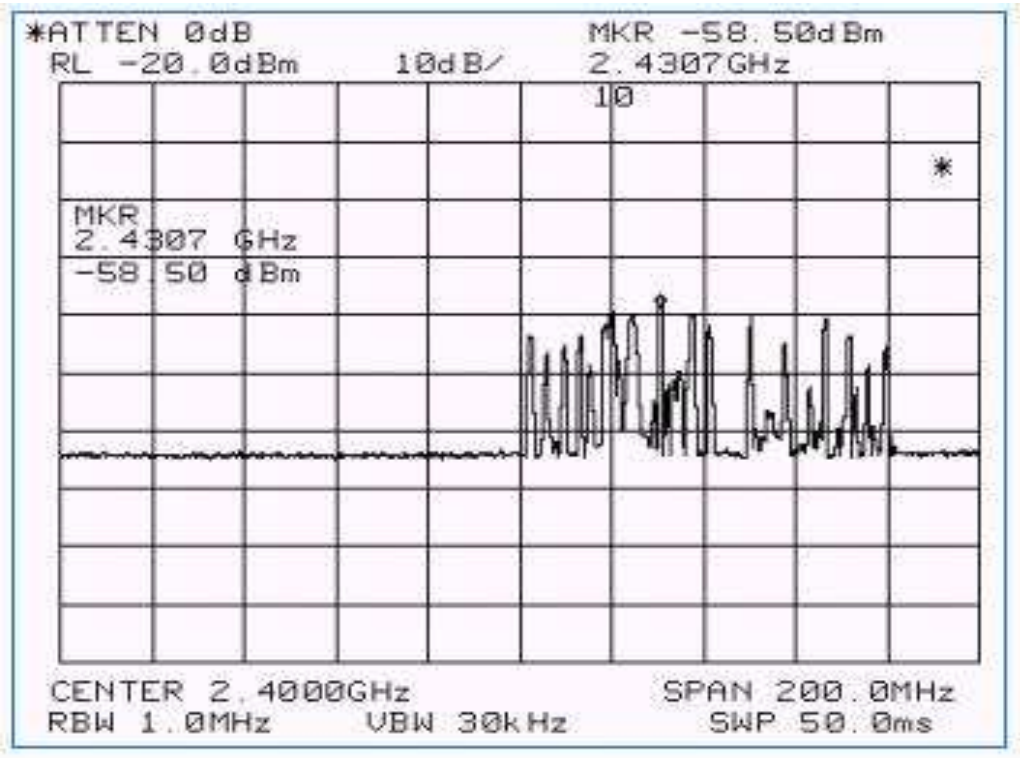


Figura 42 Nivel de potencia 10m

Respuesta entre las estaciones

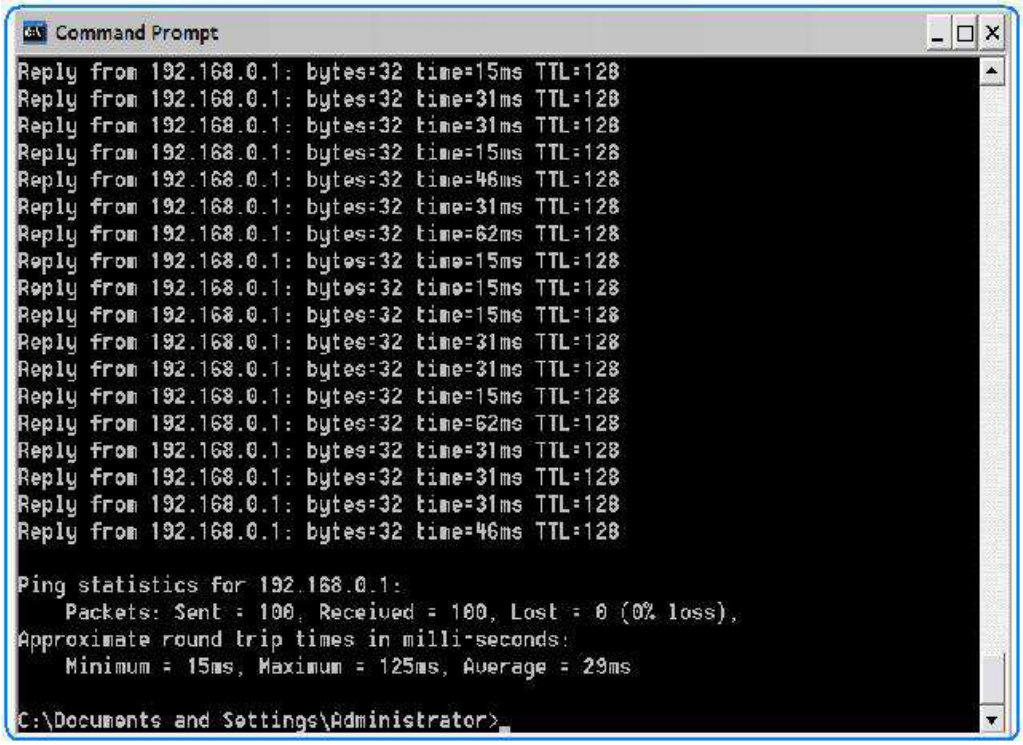


Figura 43 Ping entre las estaciones 10m

Velocidad

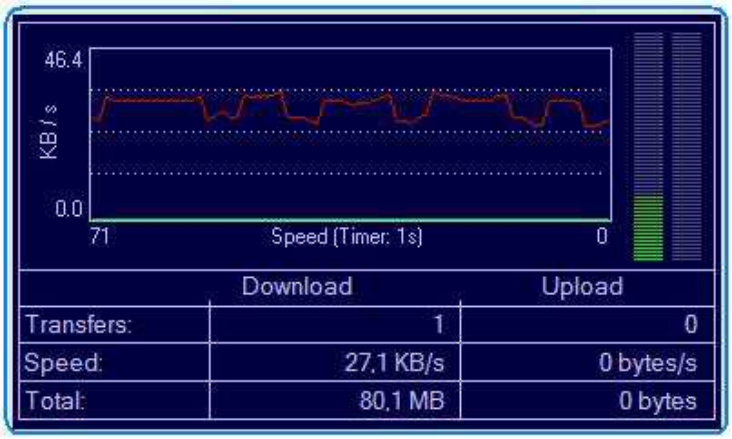


Figura 44 Velocidad 10m

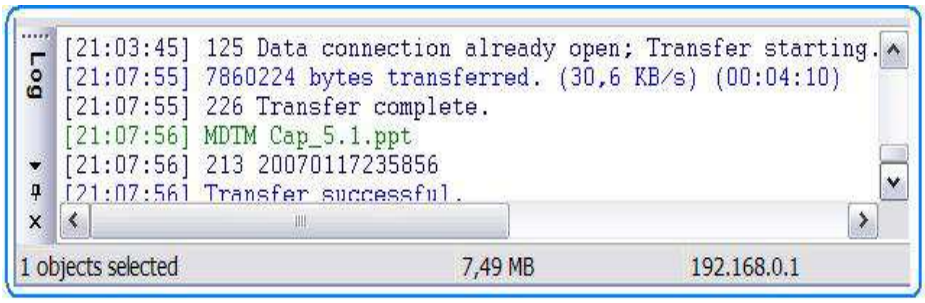


Figura 45 Velocidad Promedio 10m

Medidas a 12 metros de separación

Señal detectada

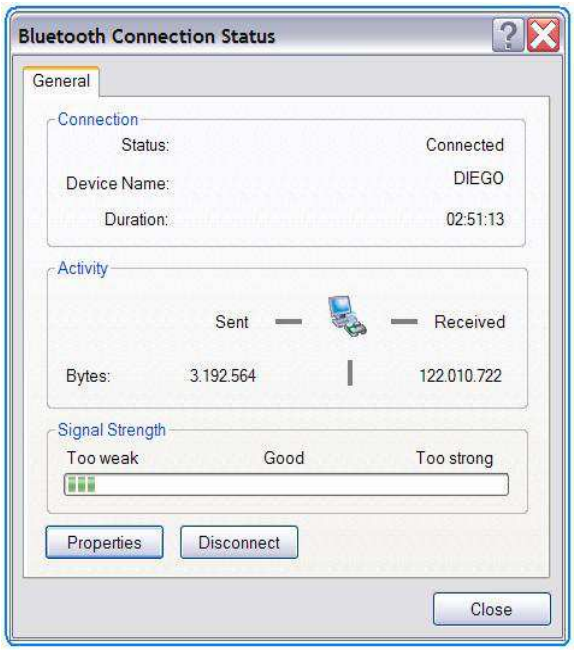


Figura 46 Estado de conexión 12m

Nivel de potencia (d = 12 m, p = - 60.00 dBm, f = 2.4397 GHz)

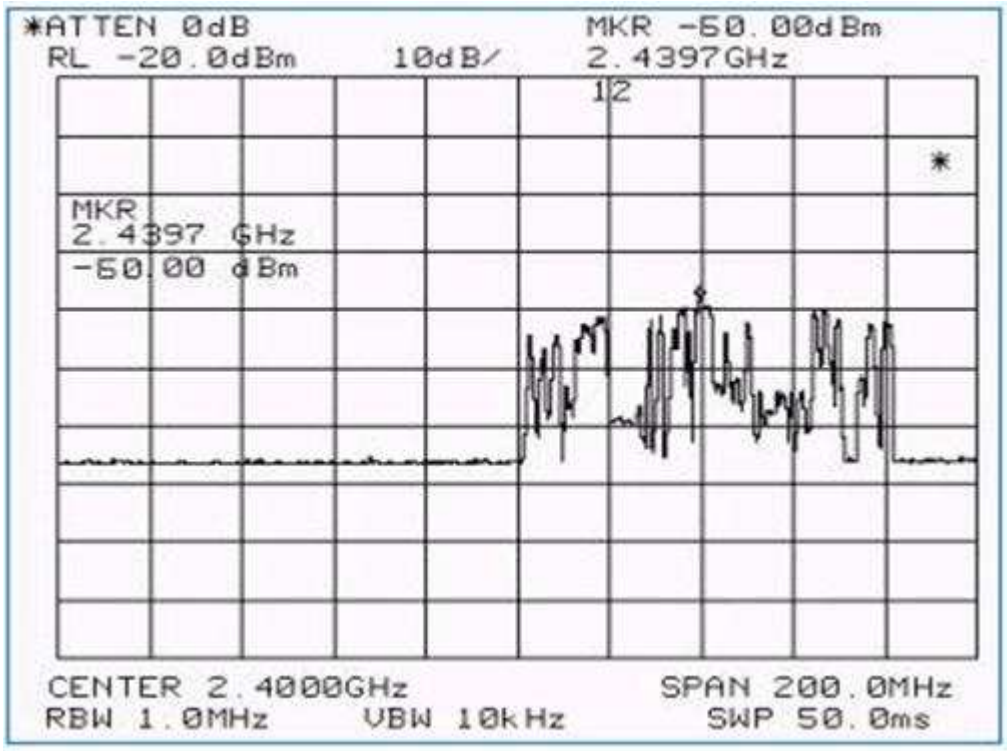


Figura 47 Nivel de potencia 12m

Respuesta entre las estaciones

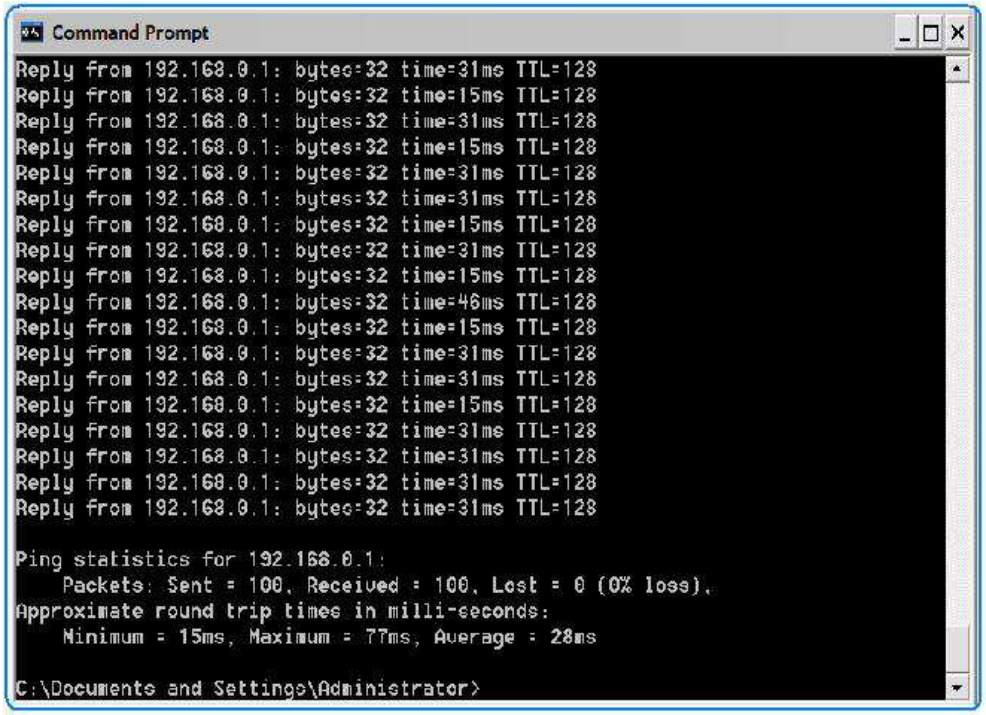


Figura 48 Ping entre las estaciones 12m

Velocidad

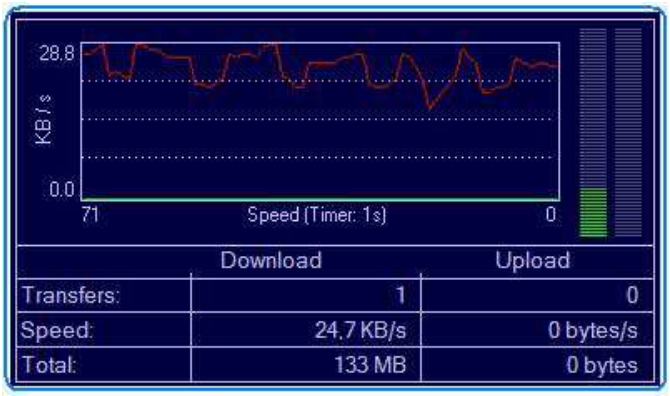


Figura 49 Velocidad 12m

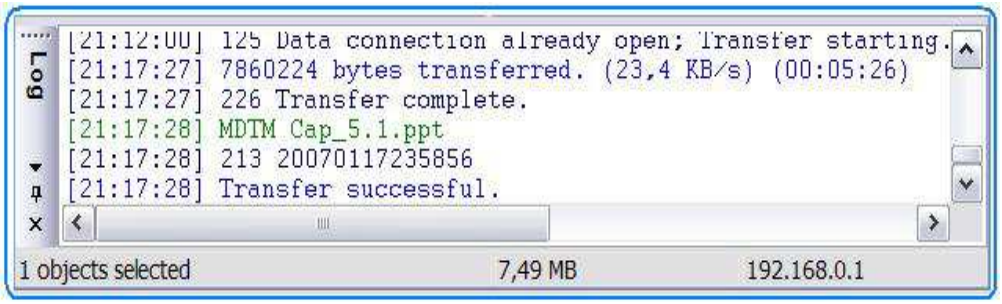


Figura 50 Velocidad Promedio 12m

Medidas a 15 metros de separación

Señal detectada

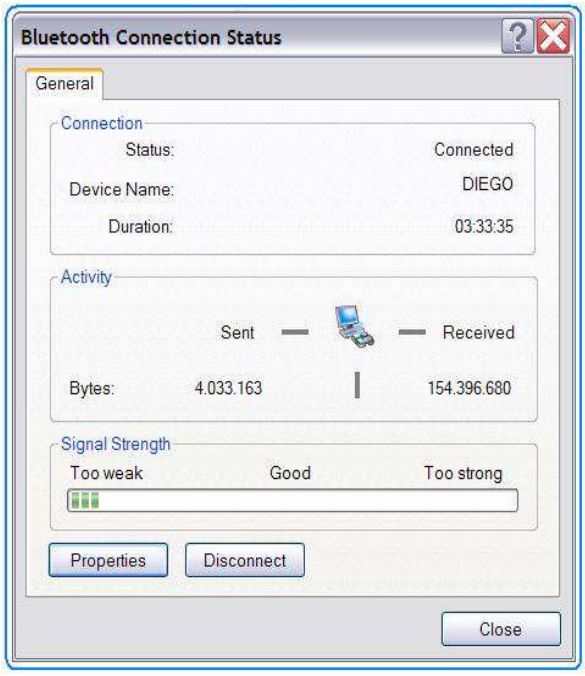


Figura 51 Estado de conexión 15m

Nivel de potencia (d = 15 m, p = - 61.83 dBm, f = 2.4207 GHz)

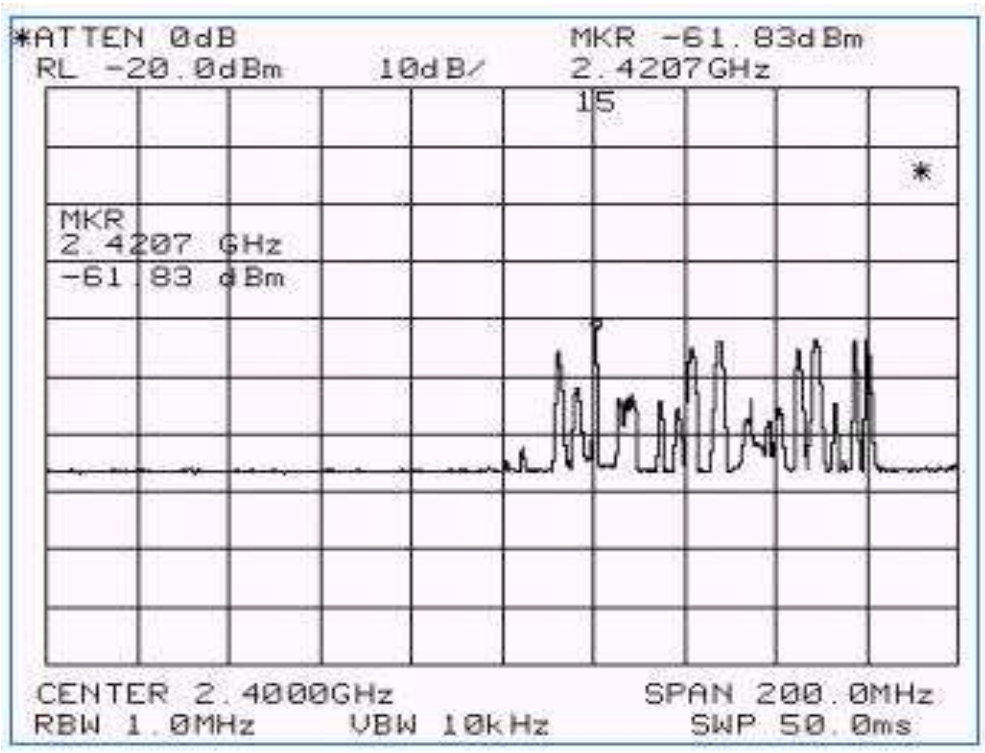


Figura 52 Nivel de potencia 15m

Respuesta entre las estaciones

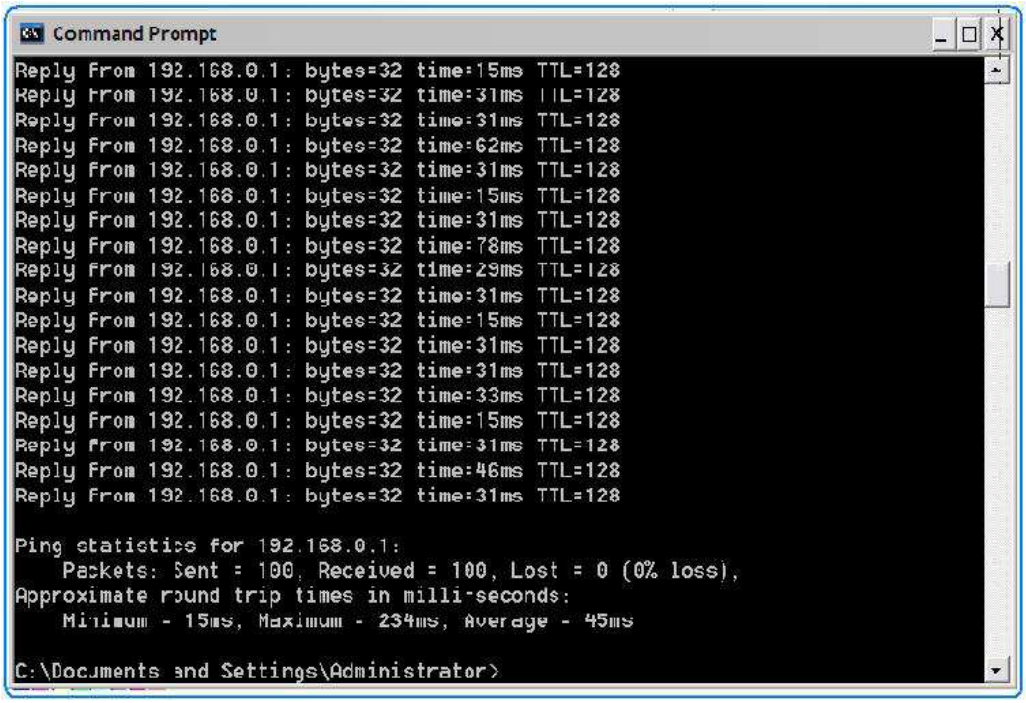


Figura 53 Ping entre las estaciones 15m

Velocidad

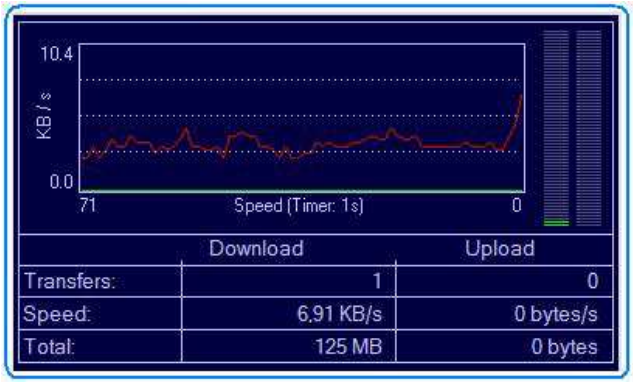


Figura 54 Velocidad 15m

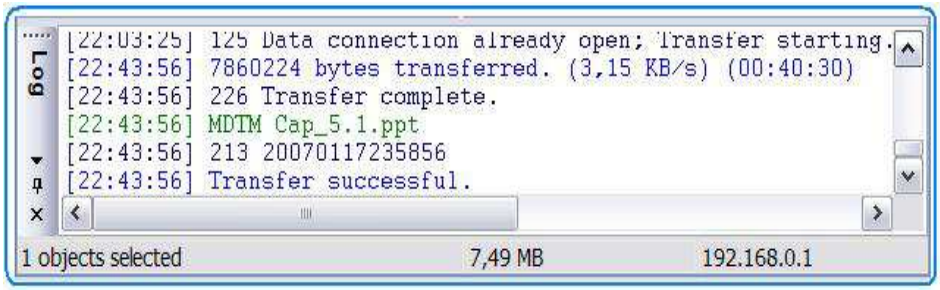


Figura 55 Velocidad 15m

ANEXO E

Pruebas Prácticas Wi-Fi

Pruebas Prácticas de Wi-Fi

Medidas a 2 metros de separación

Señal detectada

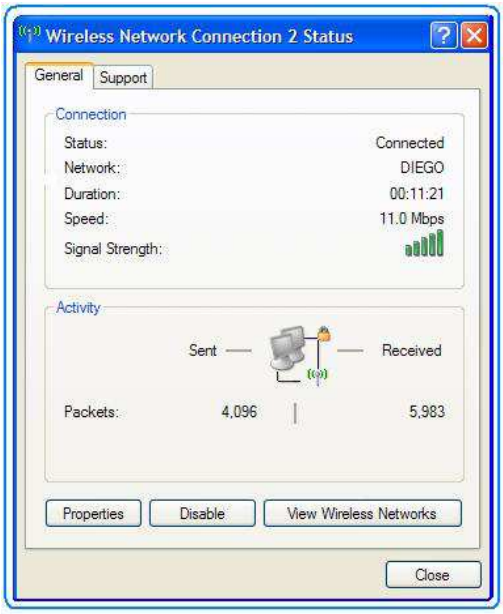


Figura 1 Estado de conexión 2m

Nivel de potencia (d = 2 m, p = - 59.00 dBm, f = 2.4340 GHz)

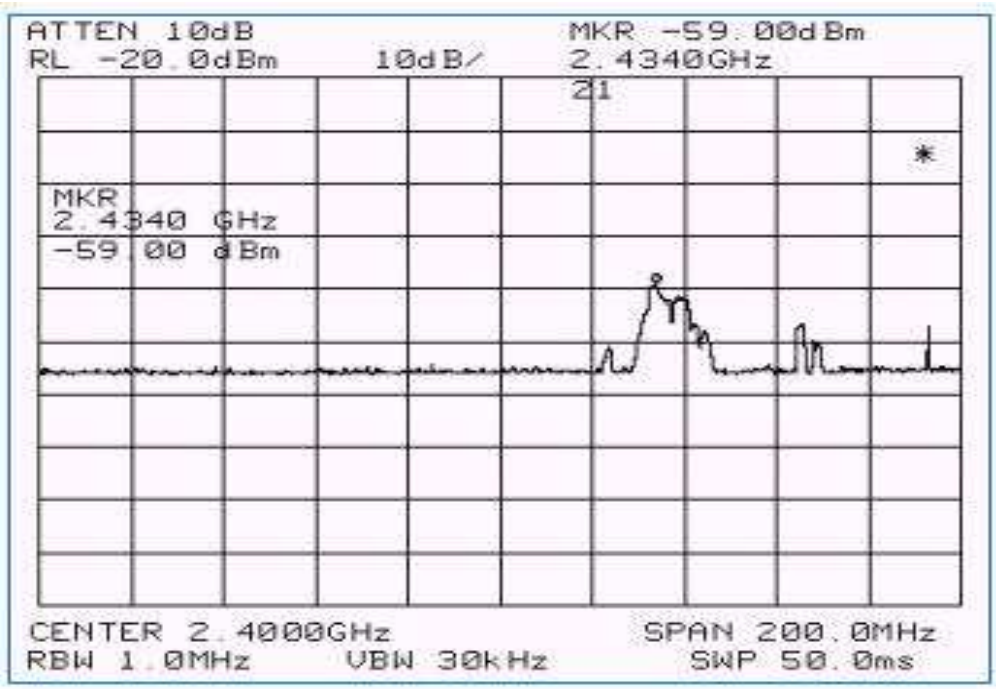


Figura 2 Nivel de potencia 2m

Respuesta entre las estaciones

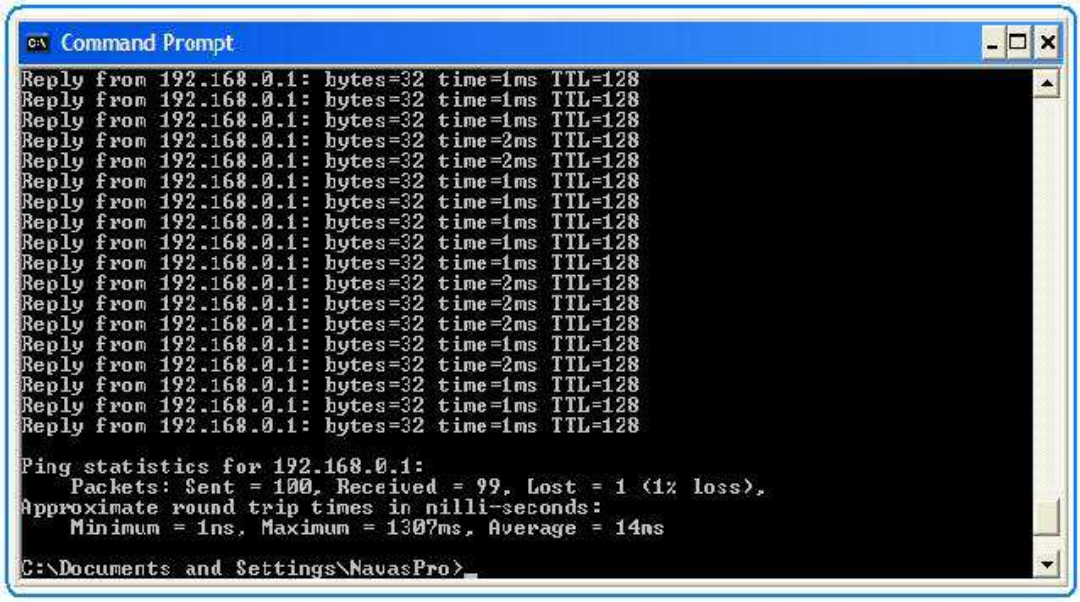


Figura 3 Ping entre las estaciones 2m

Velocidad

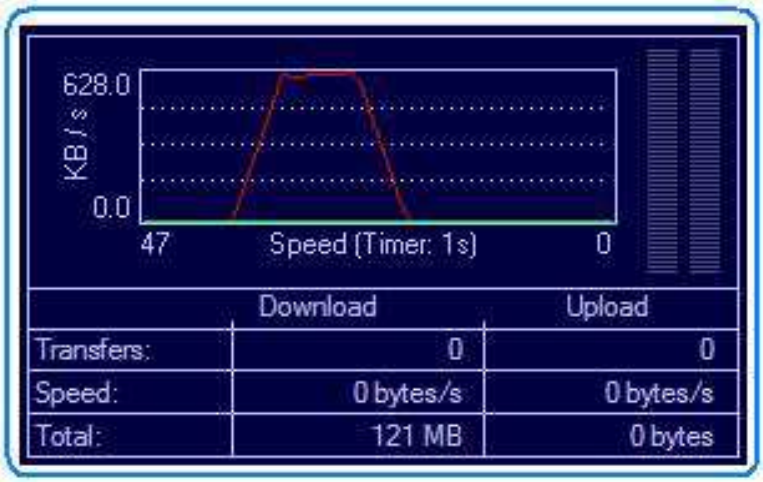


Figura 4 Velocidad 2m

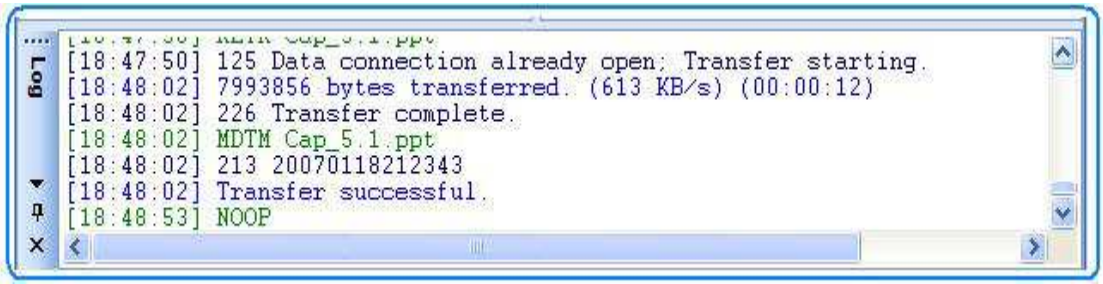


Figura 5 Velocidad Promedio 2m

Medidas a 3 metros de separación

Señal detectada

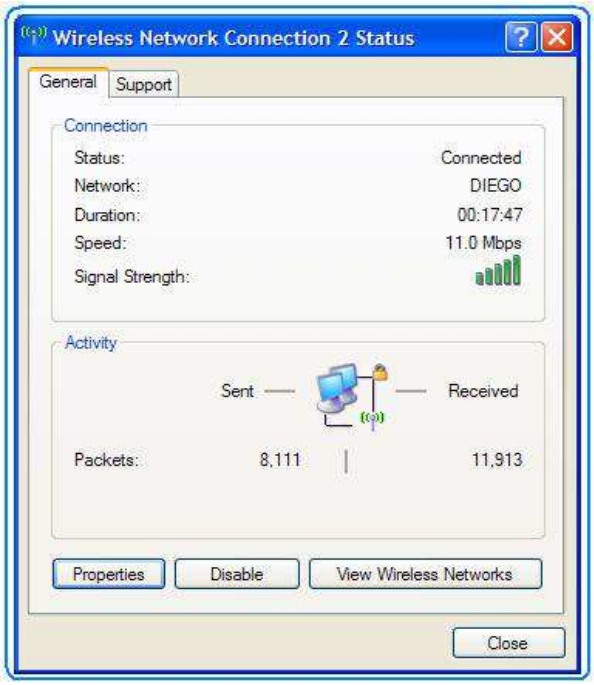


Figura 6 Estado de conexión 3m

Nivel de potencia (d = 3 m, p = - 60.83 dBm, f = 2.4410 GHz)

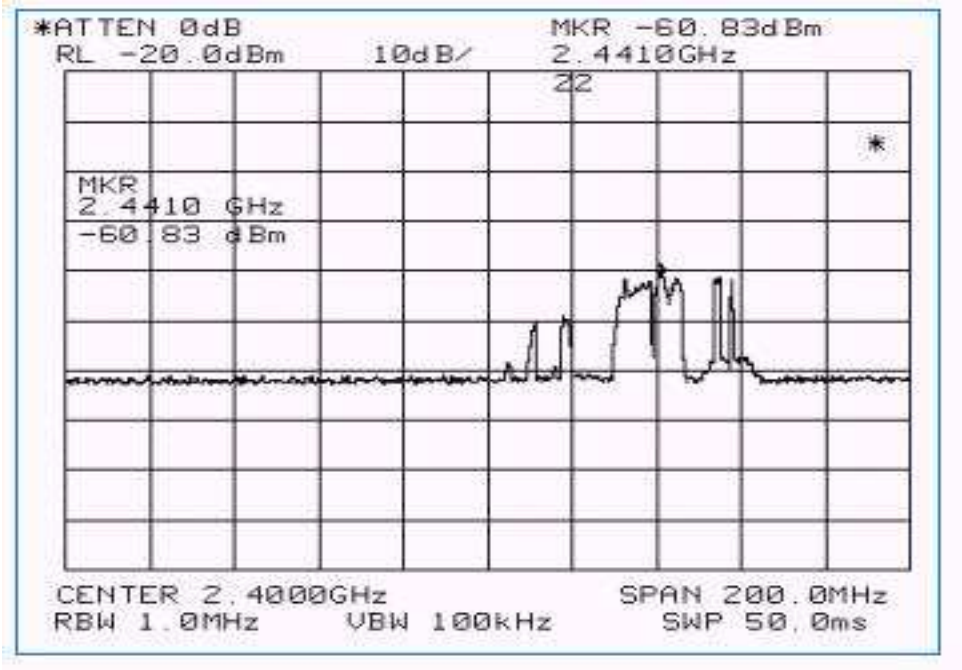


Figura 7 Nivel de potencia 3m

Respuesta entre las estaciones

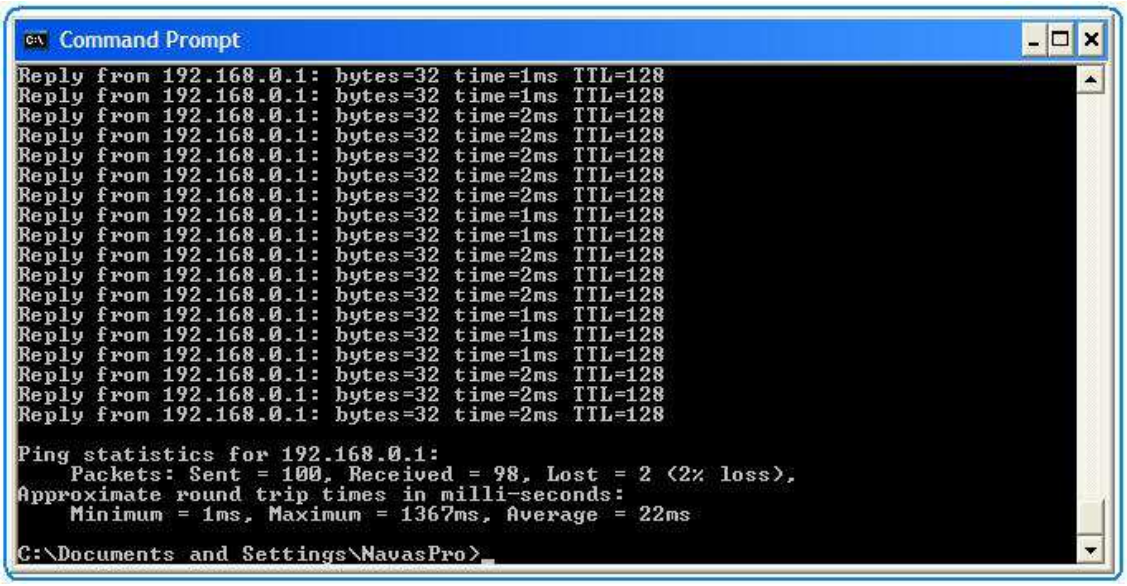


Figura 8 Ping entre las estaciones 3m

Velocidad

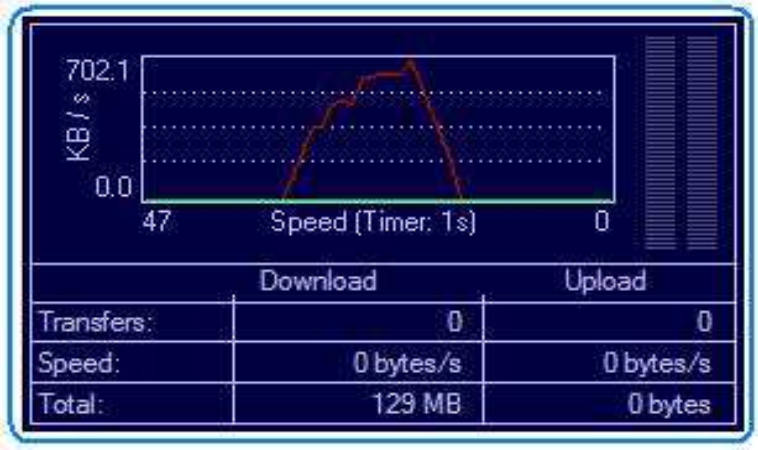


Figura 9 Velocidad 3m

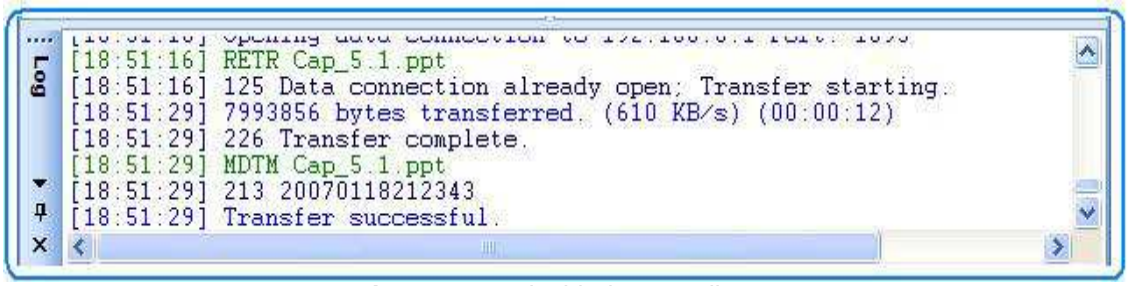


Figura 10 Velocidad Promedio 3m

Medidas a 4 metros de separación

Señal detectada

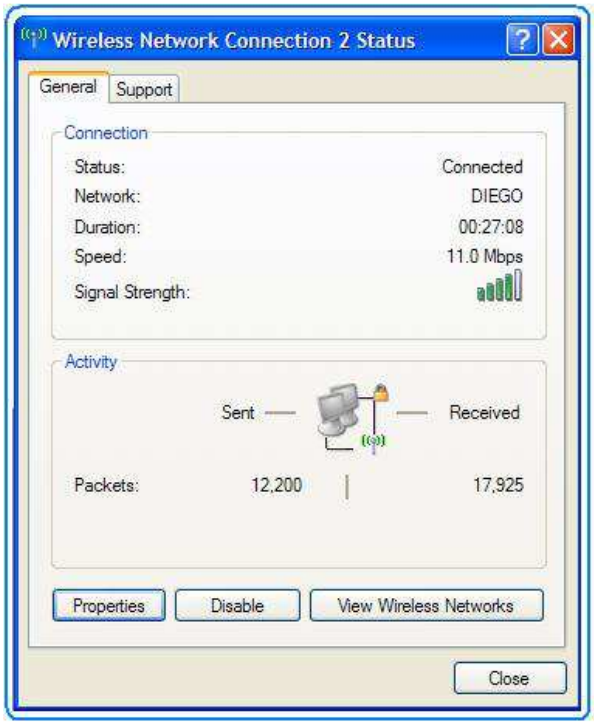


Figura 11 Estado de conexión 4m

Nivel de potencia (d = 4 m, p = - 56.83 dBm, f = 2.4383 GHz)

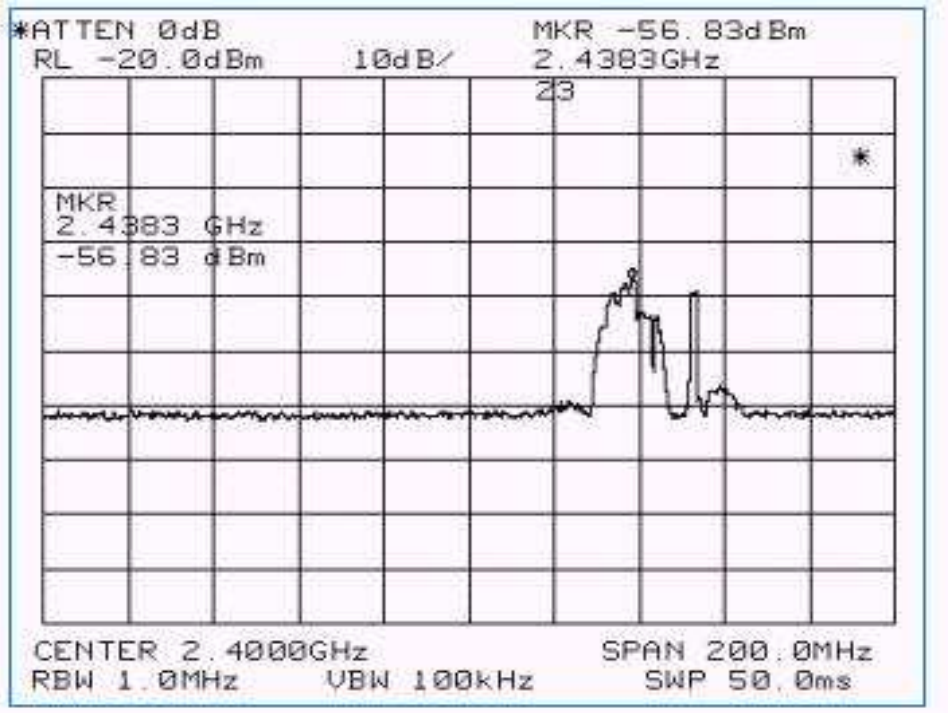


Figura 12 Nivel de potencia 4m

Respuesta entre las estaciones

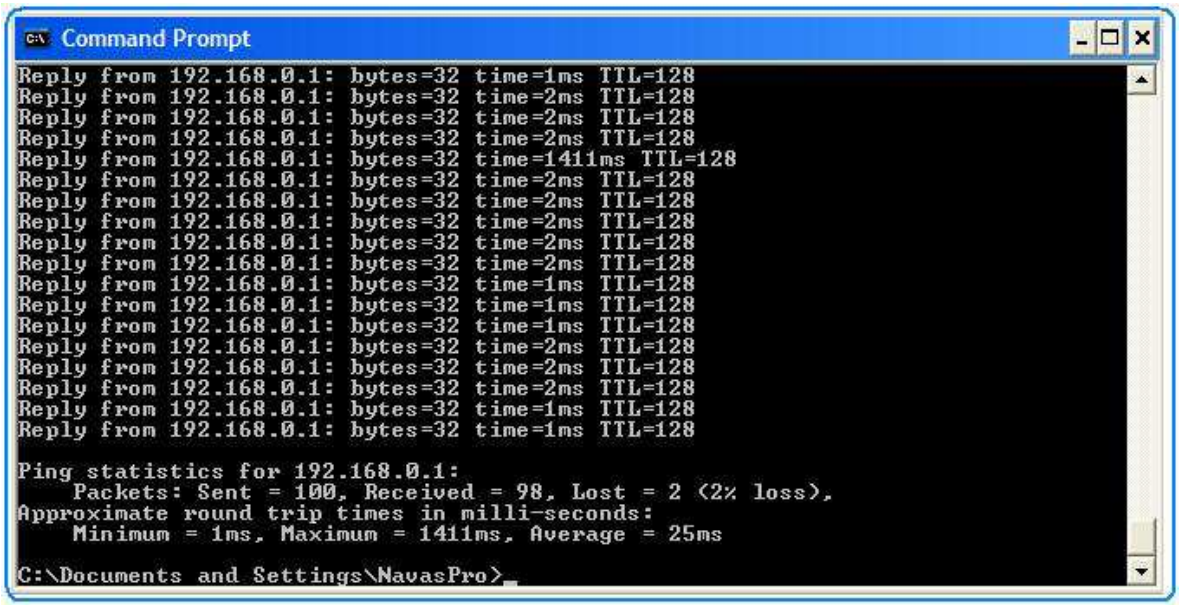


Figura 13 Ping entre las estaciones 4m

Velocidad

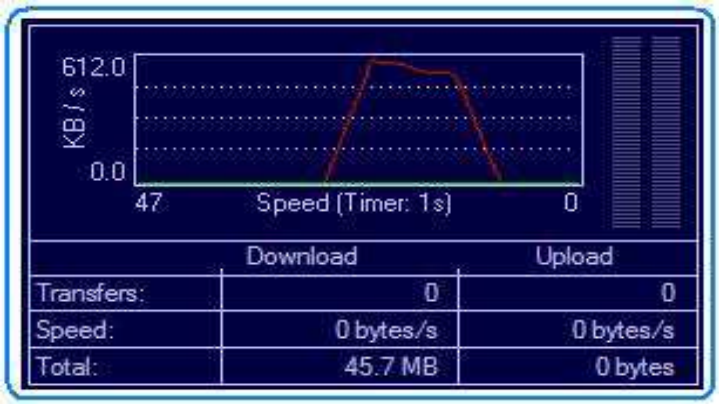


Figura 14 Velocidad 4m

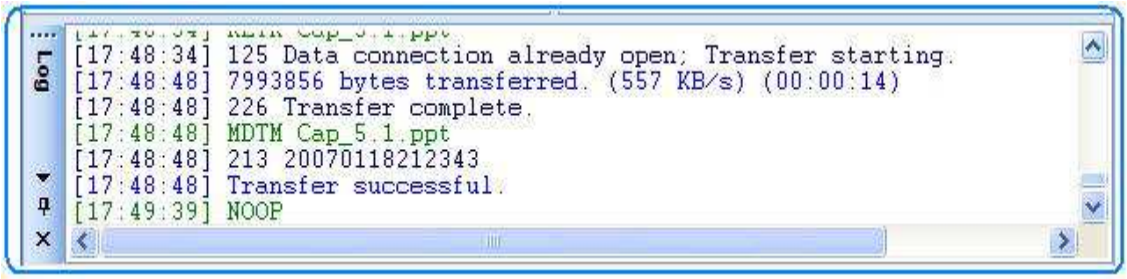


Figura 15 Velocidad Promedio 4m

Medidas a 5 metros de separación

Señal detectada

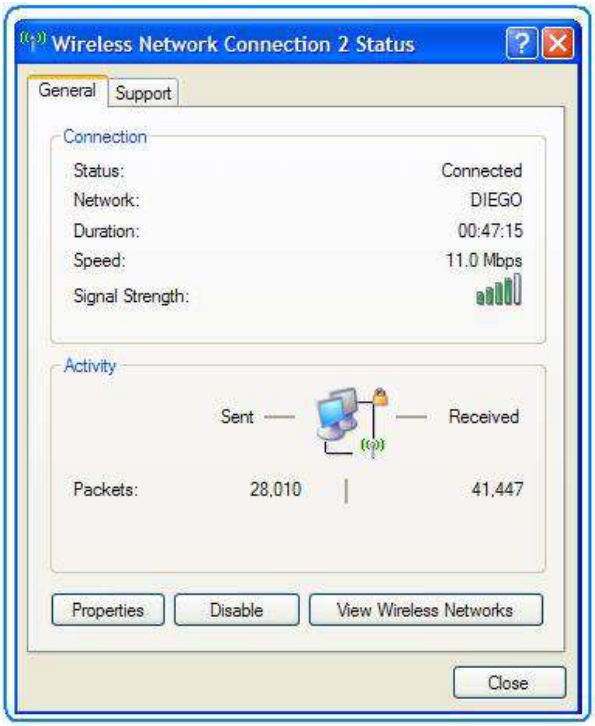


Figura 16 Estado de conexión 5m

Nivel de potencia (d = 5 m, p = - 58.33 dBm, f = 2.4383 GHz)

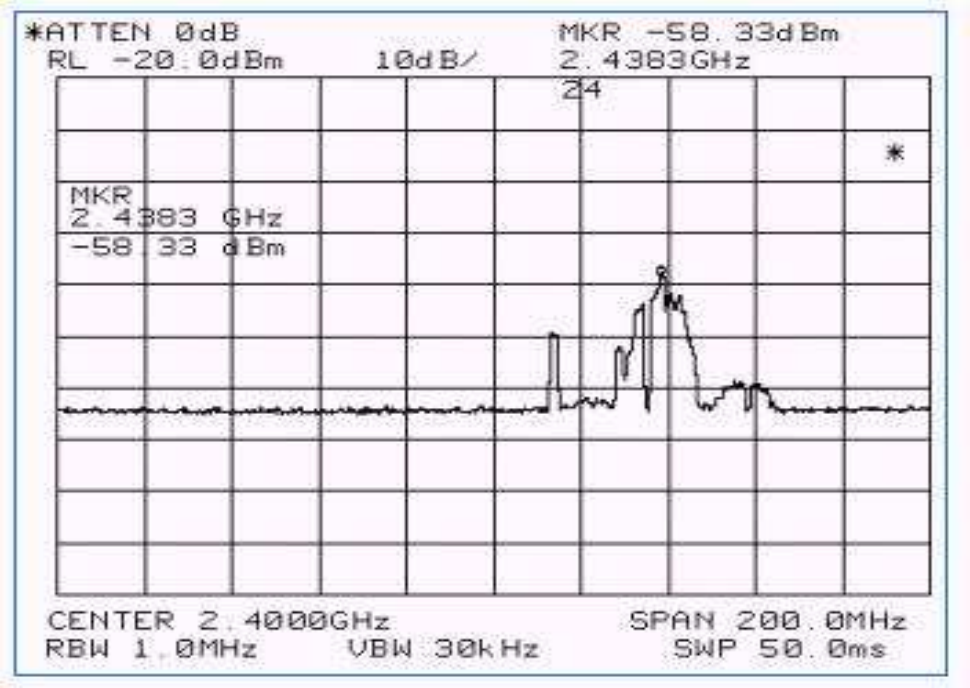


Figura 17 Nivel de potencia 5m

Respuesta entre las estaciones

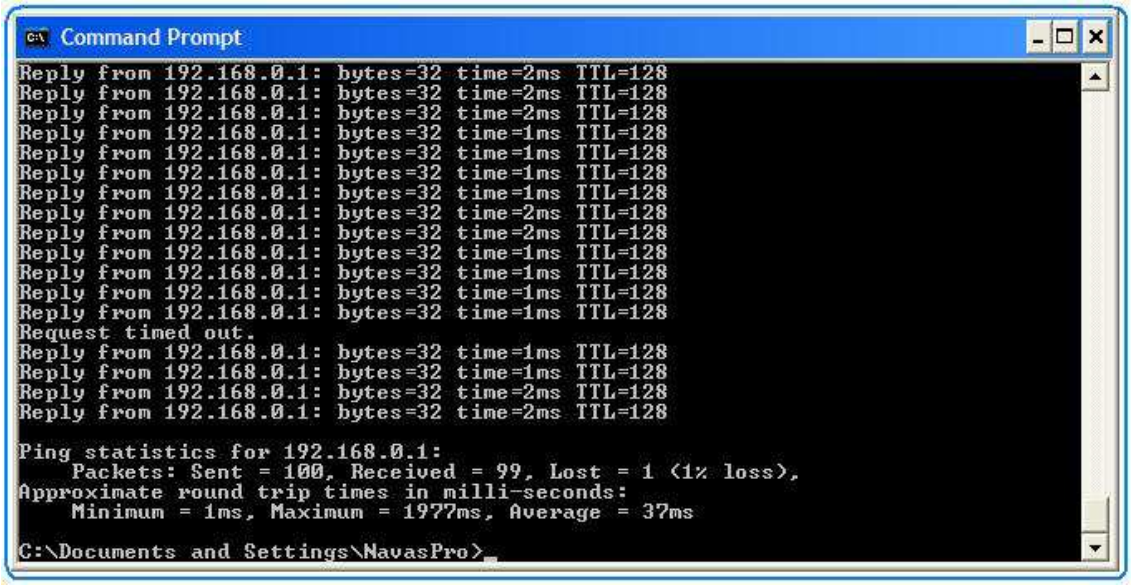


Figura 18 Ping entre las estaciones 5m

Velocidad

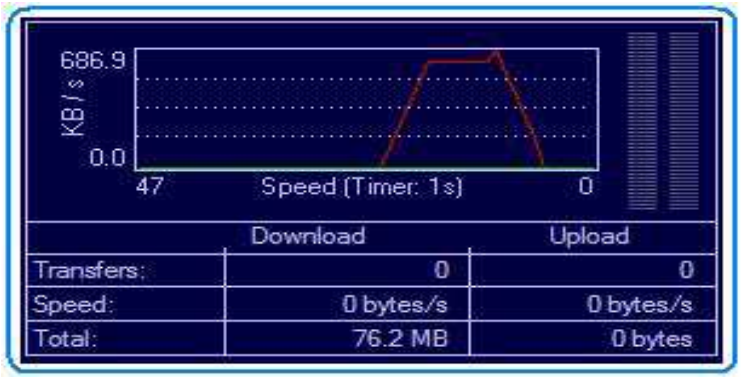


Figura 19 Velocidad 5m

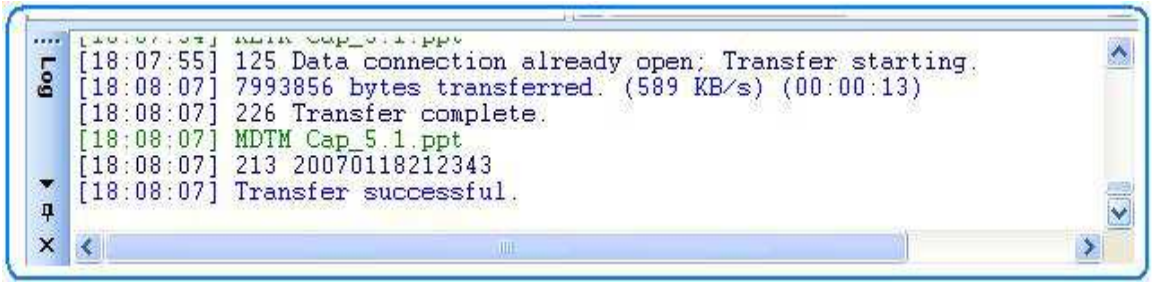


Figura 20 Velocidad Promedio 5m

Medidas a 6 metros de separación

Señal detectada

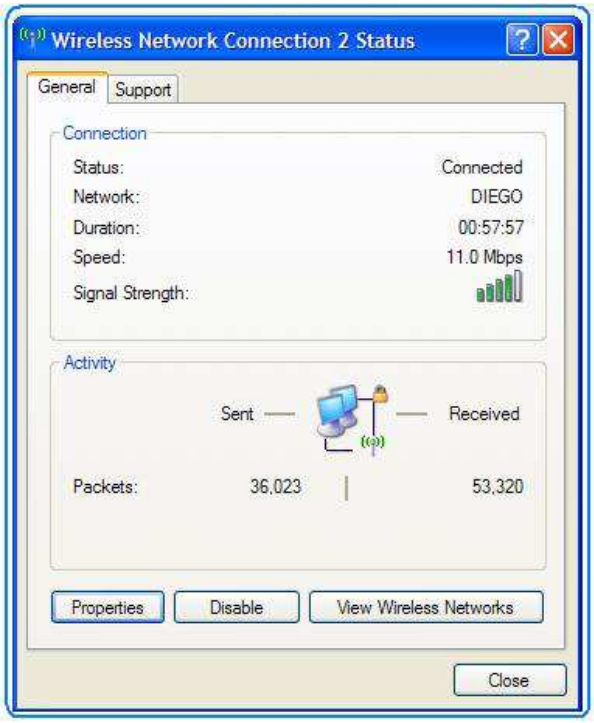


Figura 21 Estado de conexión 6m

Nivel de potencia (d = 6 m, p = - 62.50 dBm, f = 2.4337 GHz)

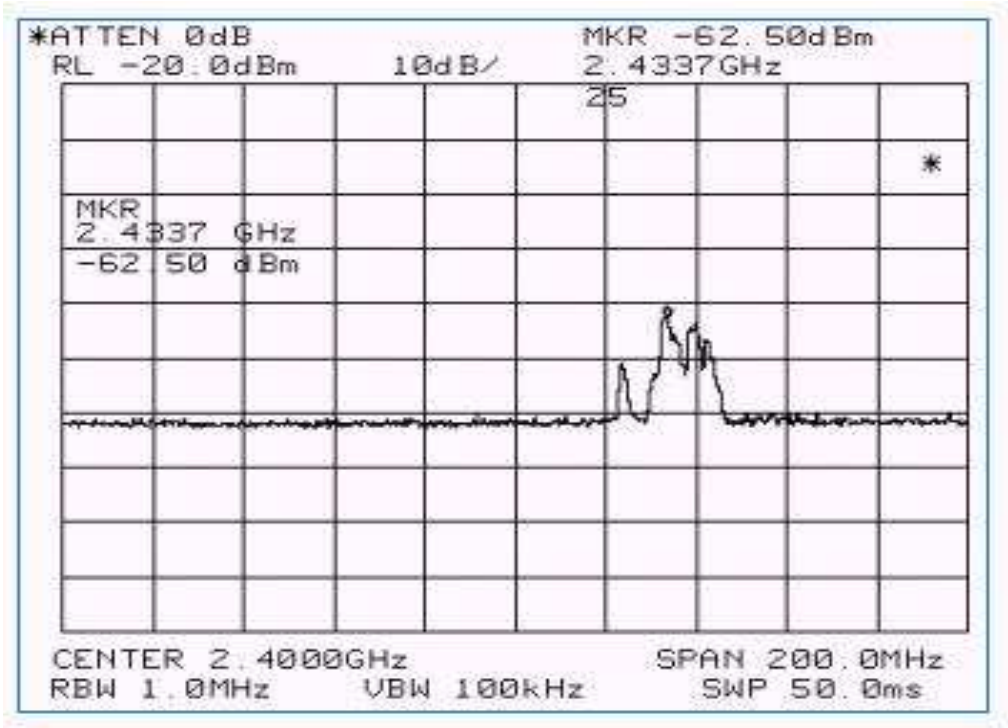


Figura 22 Nivel de potencia 6m

Respuesta entre las estaciones

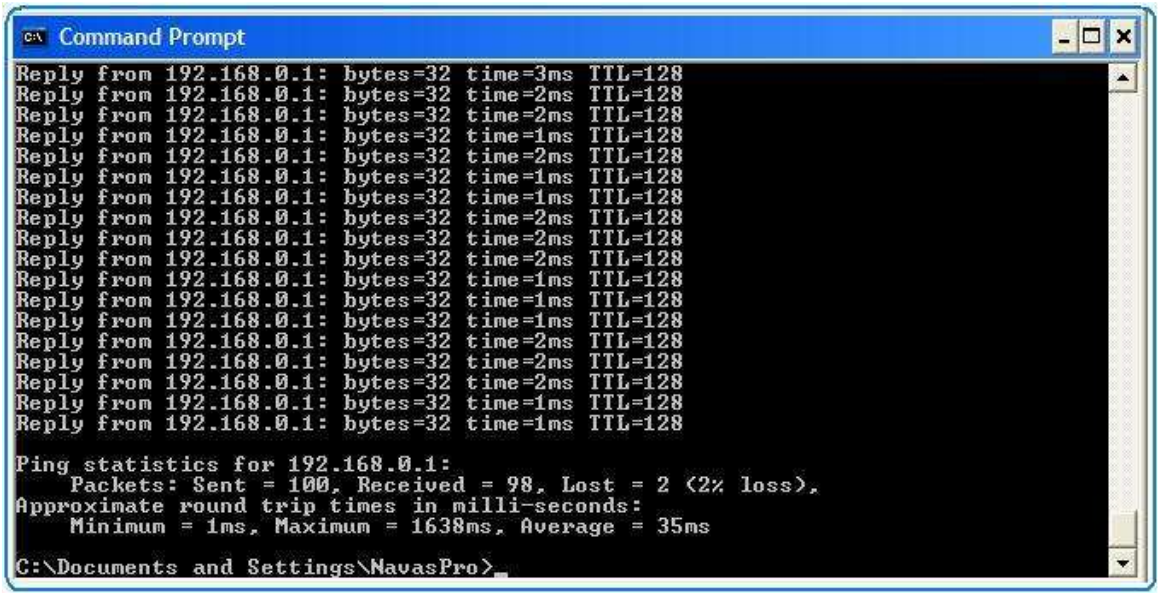


Figura 23 Ping entre las estaciones 6m

Velocidad

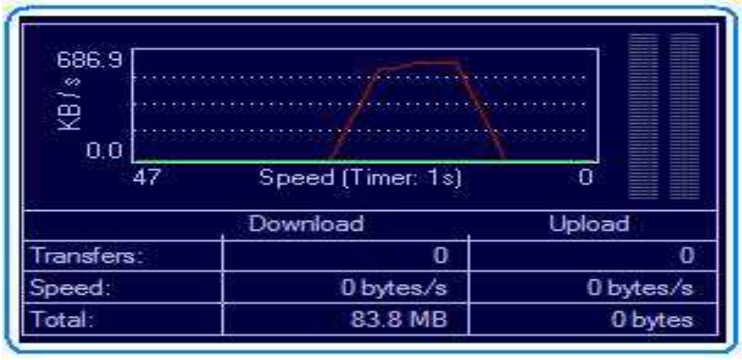


Figura 24 Velocidad 6m

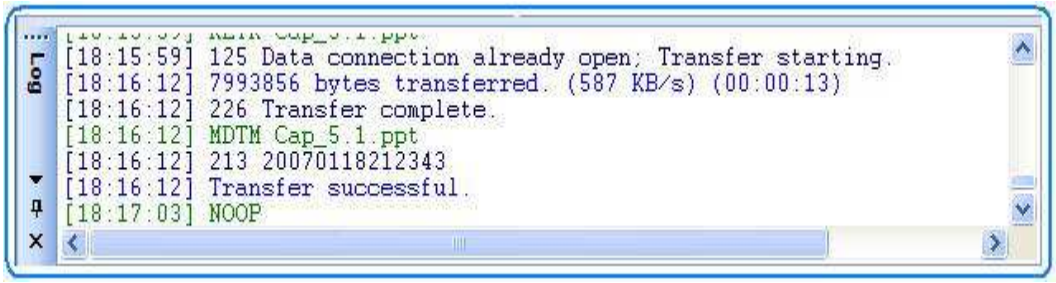


Figura 25 Velocidad Promedio 6m

Medidas a 7 metros de separación

Señal detectada

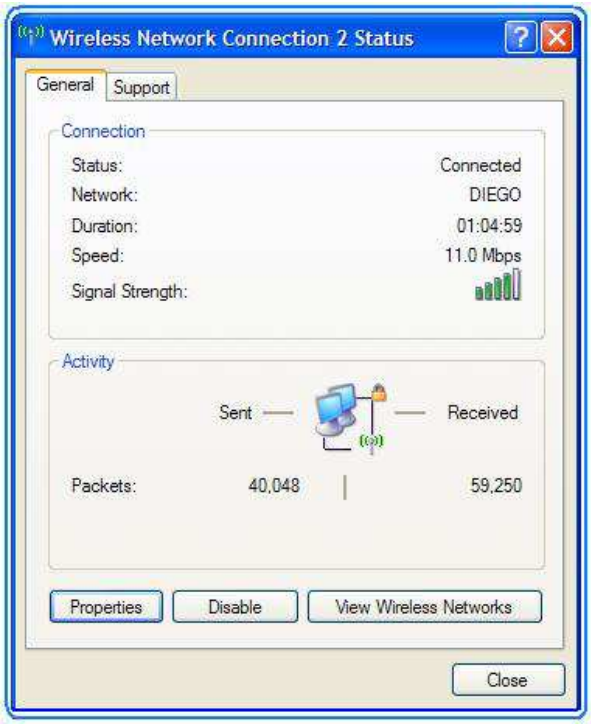


Figura 26 Estado de conexión 7m

Nivel de potencia (d = 7 m, p = - 58.50 dBm, f = 2.4383 GHz)

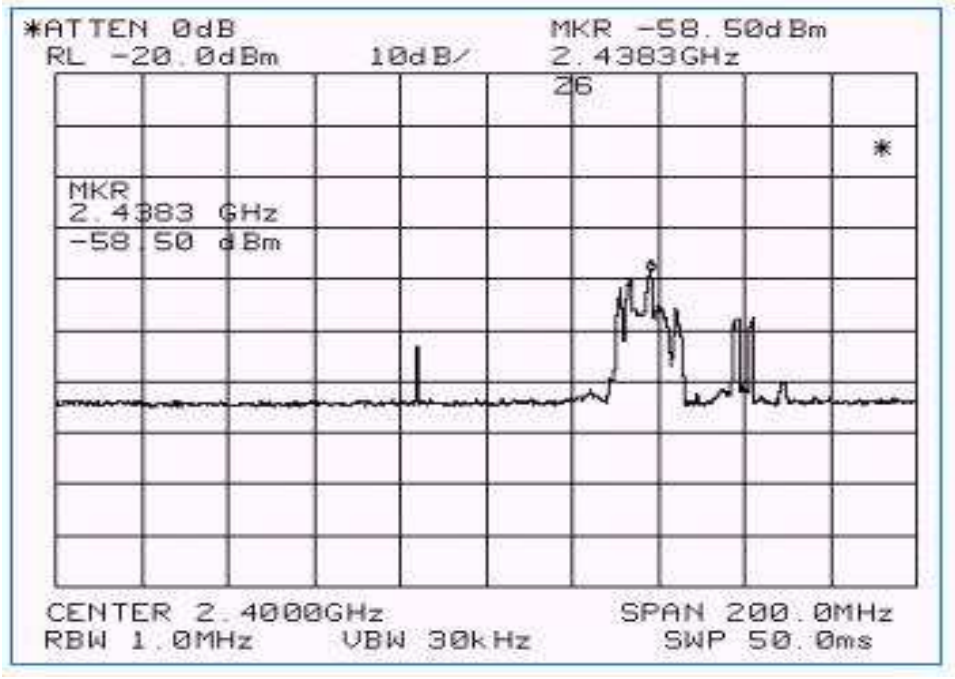


Figura 27 Nivel de potencia 7m

Respuesta entre las estaciones

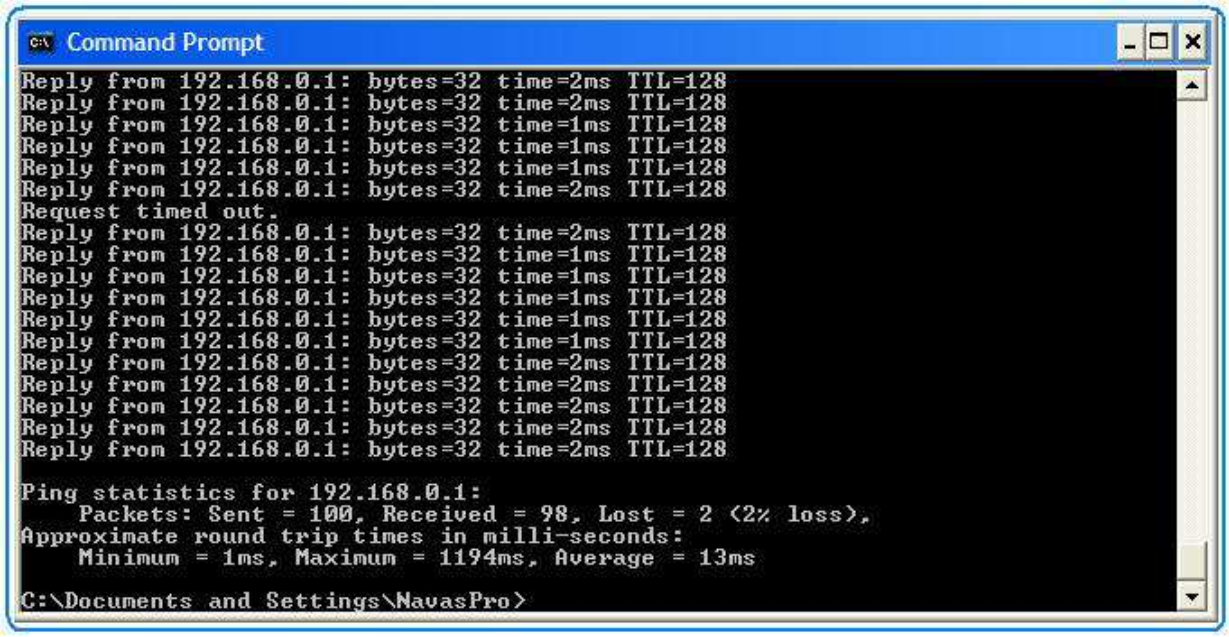


Figura 28 Ping entre las estaciones 7m

Velocidad

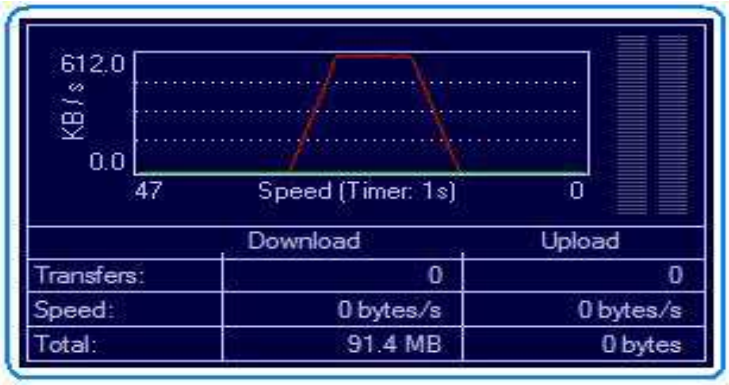


Figura 29 Velocidad 7m

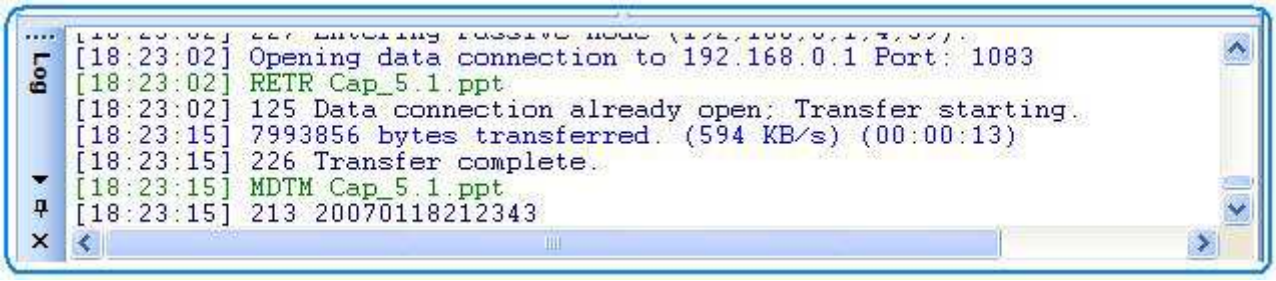


Figura 30 Velocidad Promedio 7m

Medidas a 8 metros de separación

Señal detectada

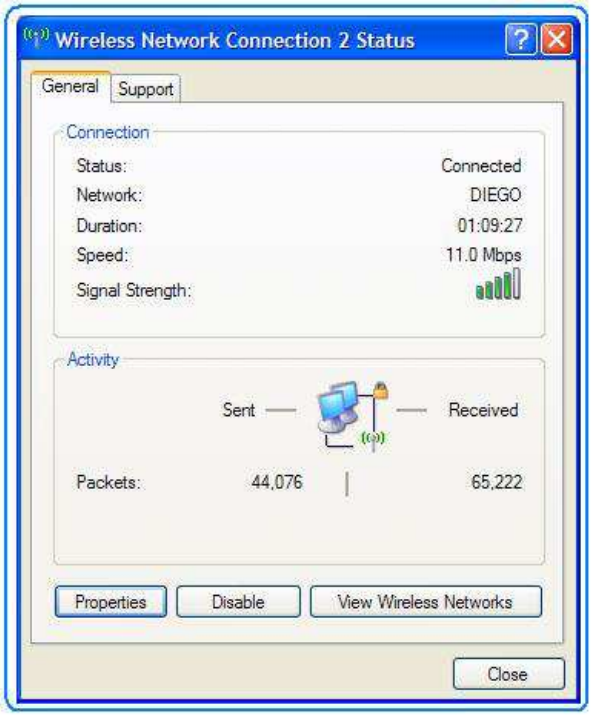


Figura 31 Estado de conexión 8m

Nivel de potencia (d = 8 m, p = - 66.83 dBm, f = 2.4323 GHz)

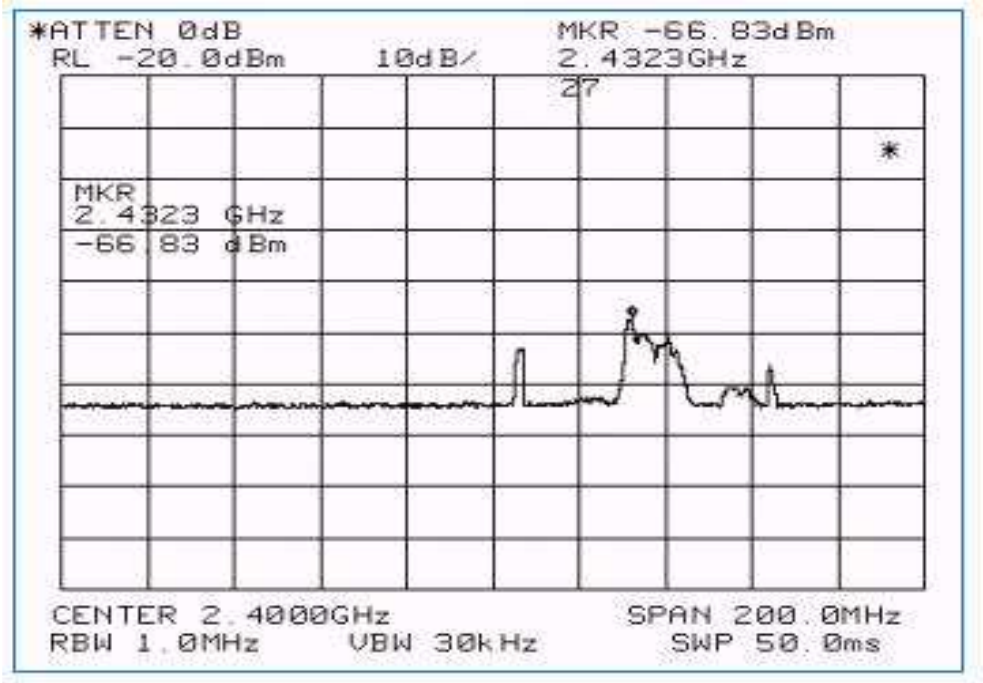


Figura 32 Nivel de potencia 8m

Respuesta entre las estaciones

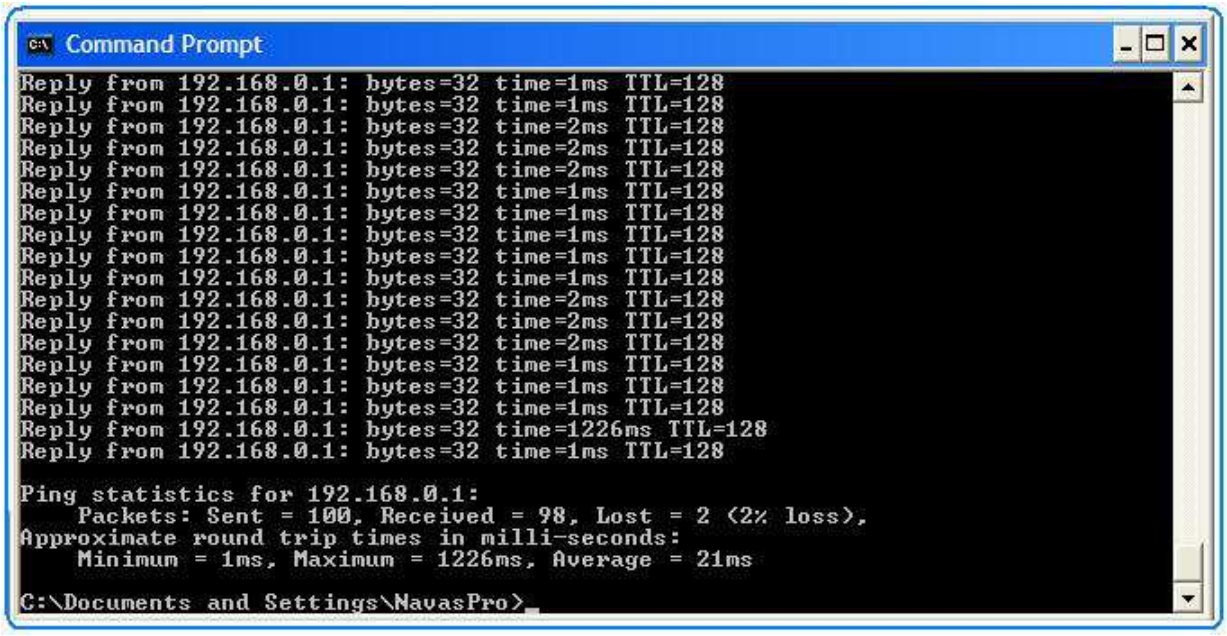


Figura 33 Ping entre las estaciones 8m

Velocidad

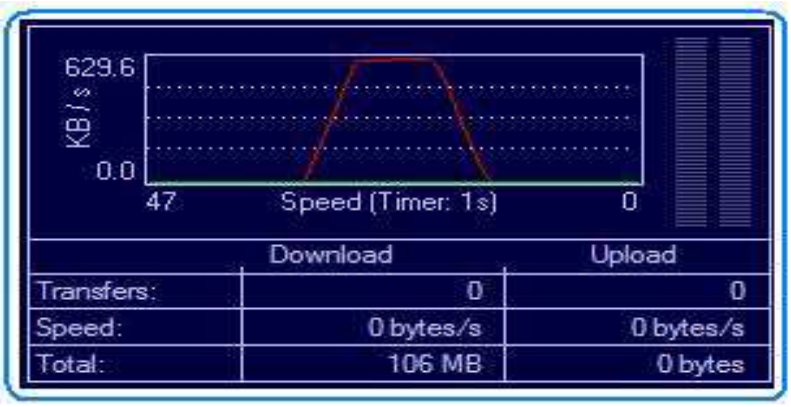


Figura 34 Velocidad 8m

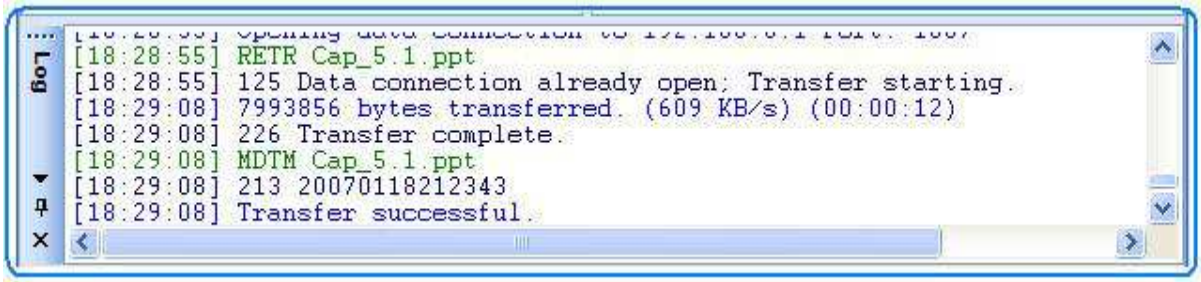


Figura 35 Velocidad 8m

Medidas a 9 metros de separación

Señal detectada

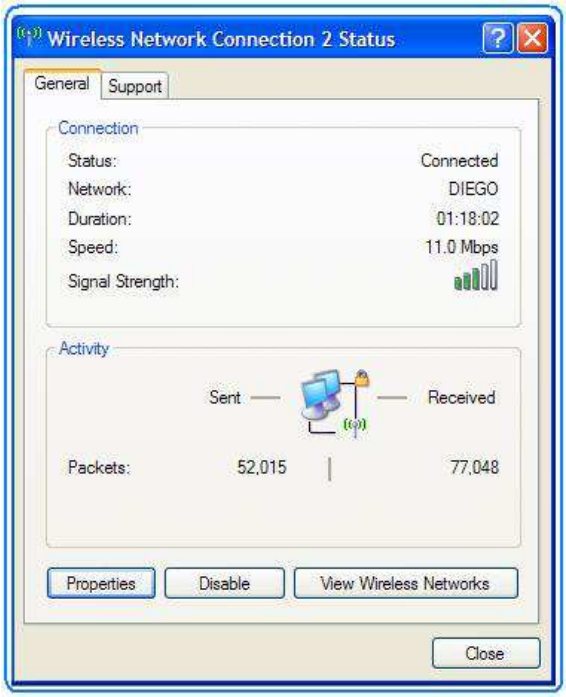


Figura 36 Estado de conexión 9m

Nivel de potencia (d = 9 m, p = - 66.50 dBm, f = 2.4347 GHz)

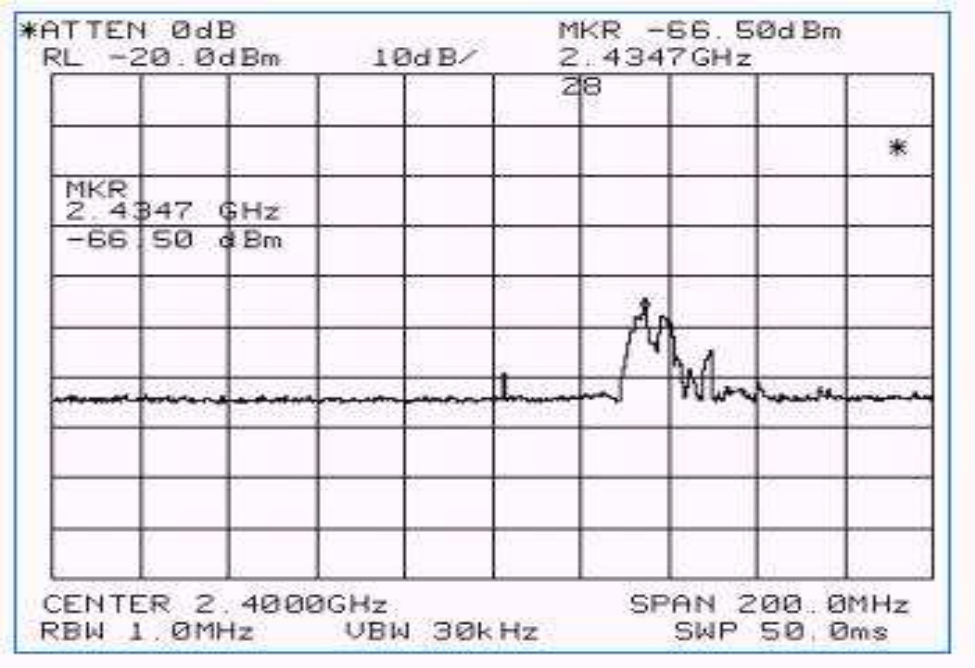


Figura 37 Nivel de potencia 9m

Respuesta entre las estaciones

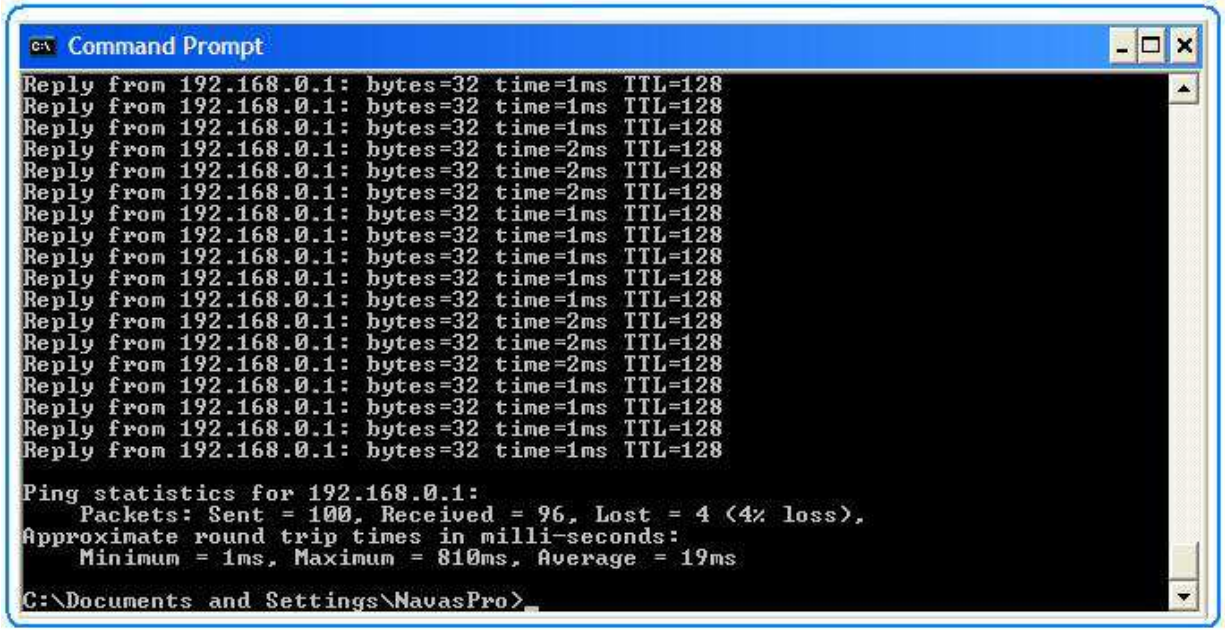


Figura 38 Ping entre las estaciones 9m

Velocidad

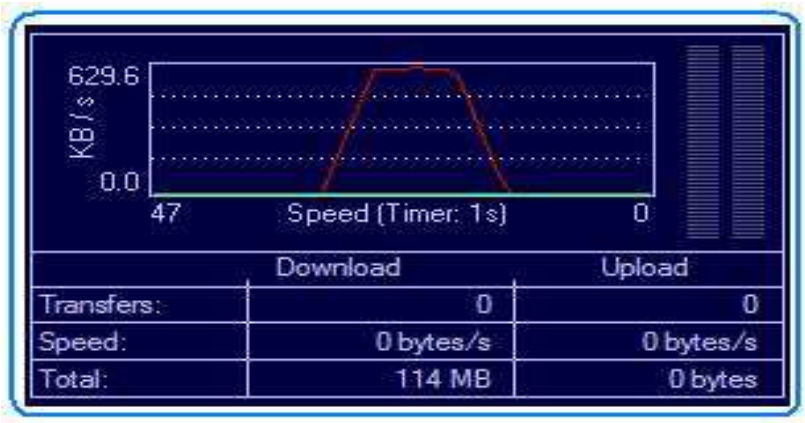


Figura 39 Velocidad 9m

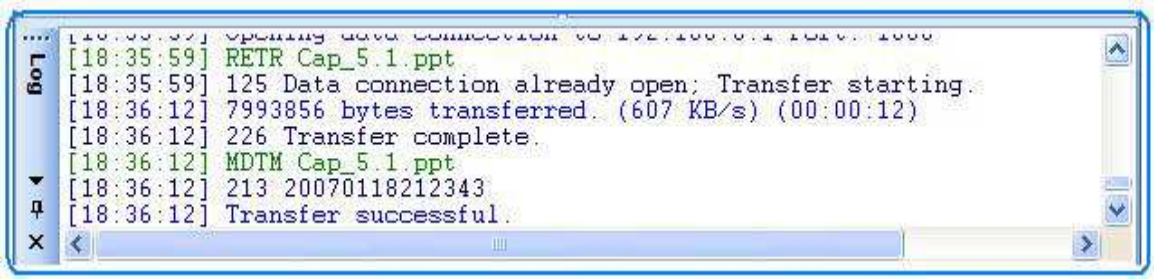


Figura 40 Velocidad Promedio 9m

Medidas a 10 metros de separación

Señal detectada

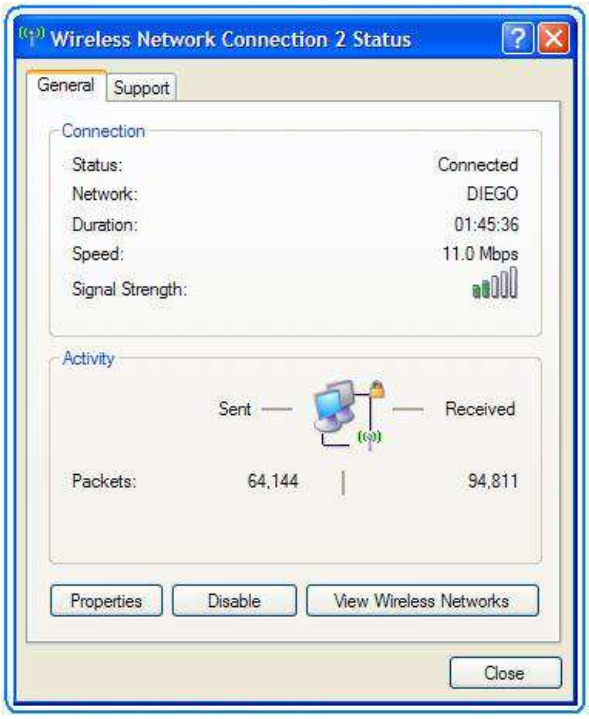


Figura 41 Estado de conexión 10m

Nivel de potencia (d = 10 m, p = - 68.83 dBm, f = 2.4207 GHz)

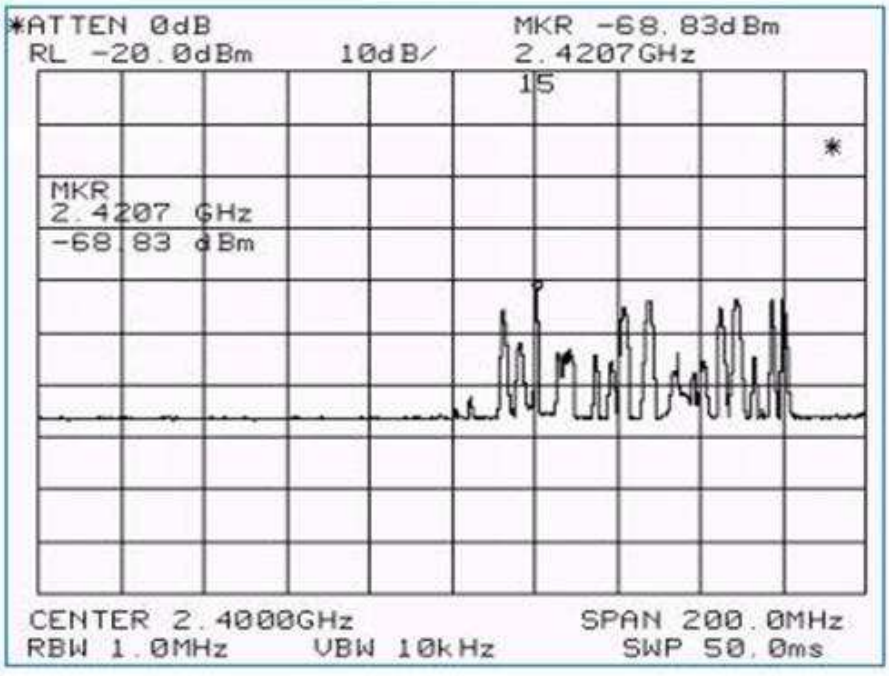


Figura 42 Nivel de potencia 10m

Respuesta entre las estaciones

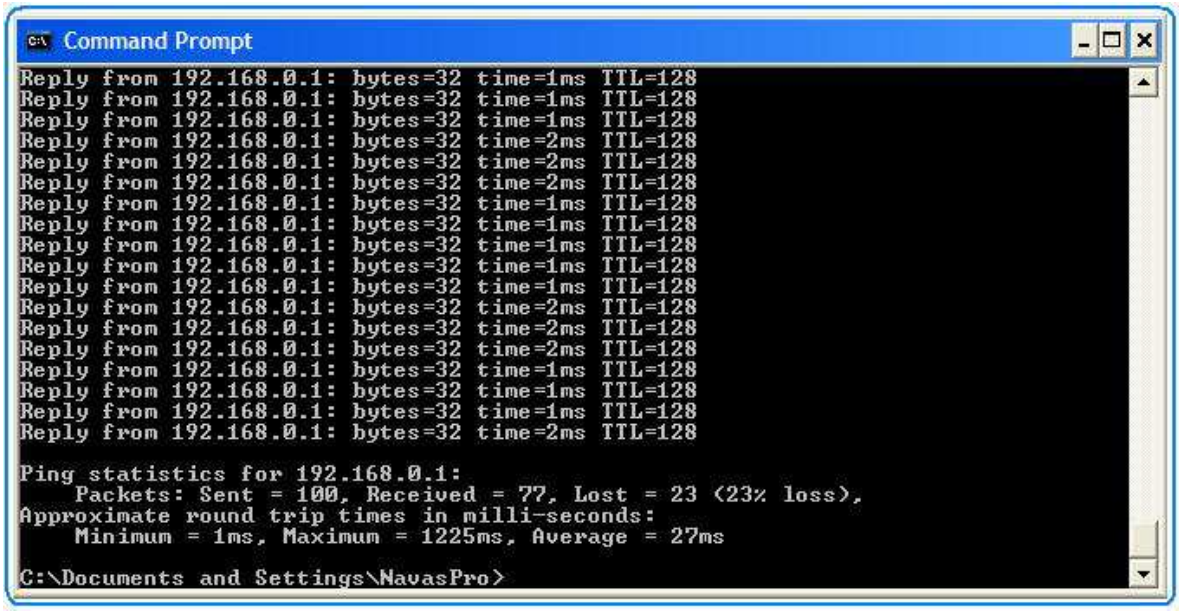


Figura 43 Ping entre las estaciones 10m

Velocidad

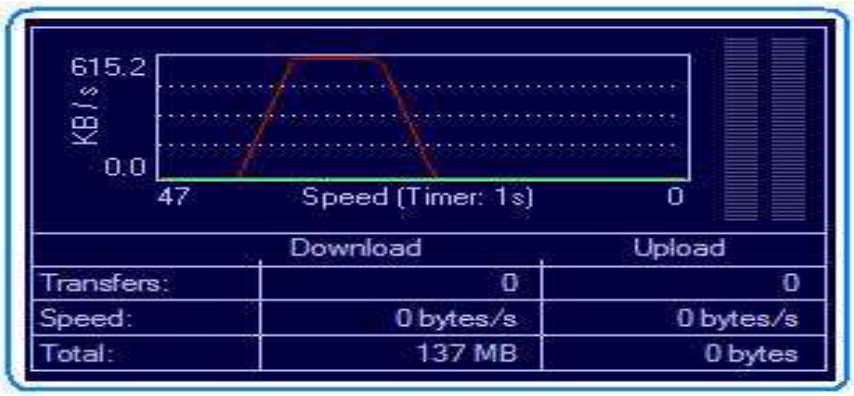


Figura 44 Velocidad 10m

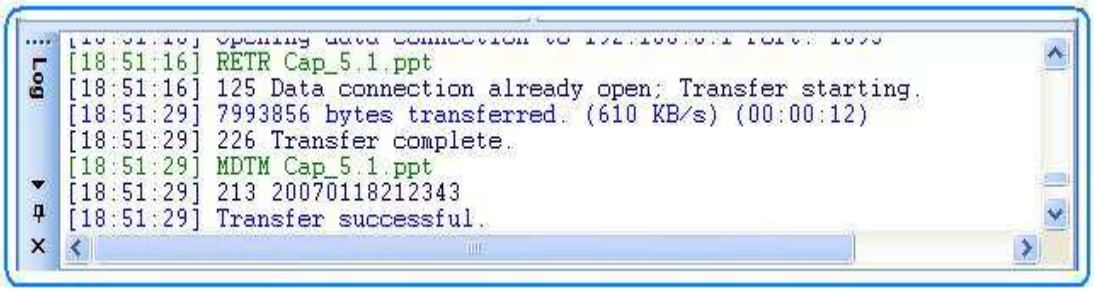


Figura 45 Velocidad 10m

Medidas a 12 metros de separación

Señal detectada

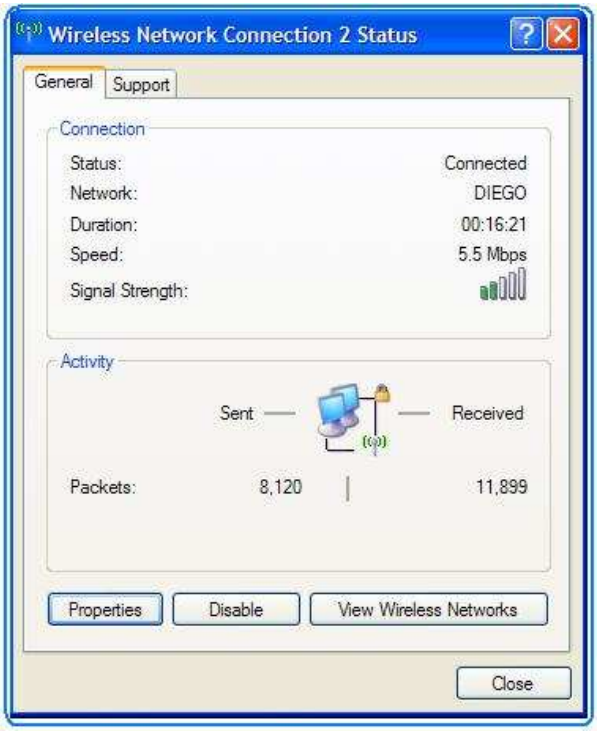


Figura 46 Estado de conexión 12m

Nivel de potencia (d = 12 m, p = - 69.39 dBm, f = 2.4397 GHz)

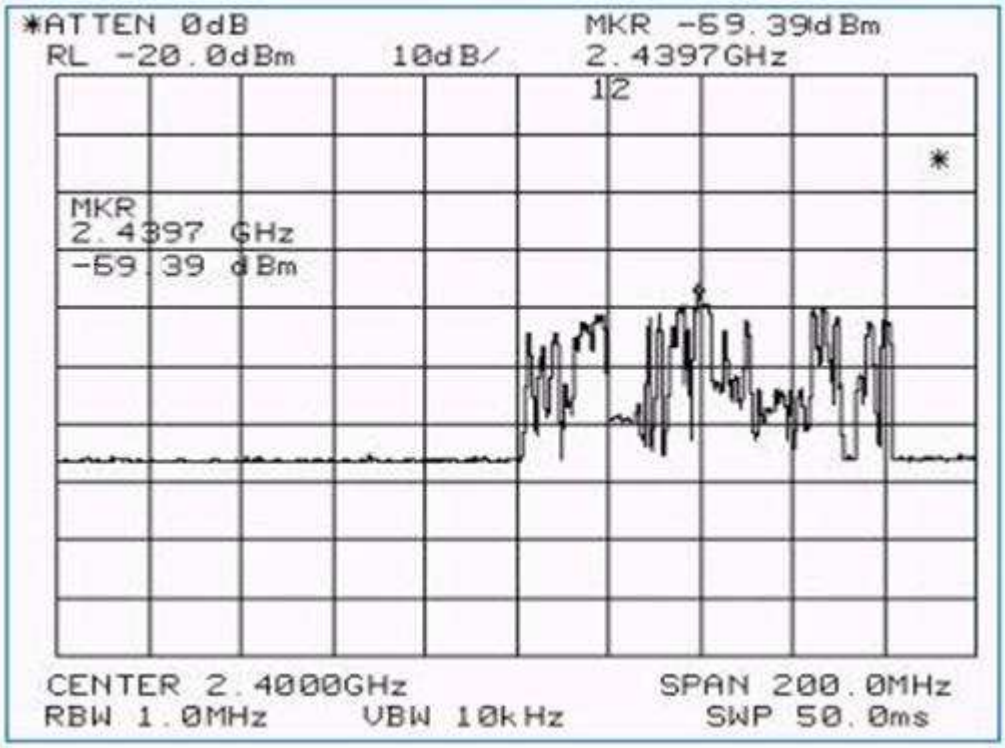


Figura 47 Nivel de potencia 12m

Respuesta entre las estaciones

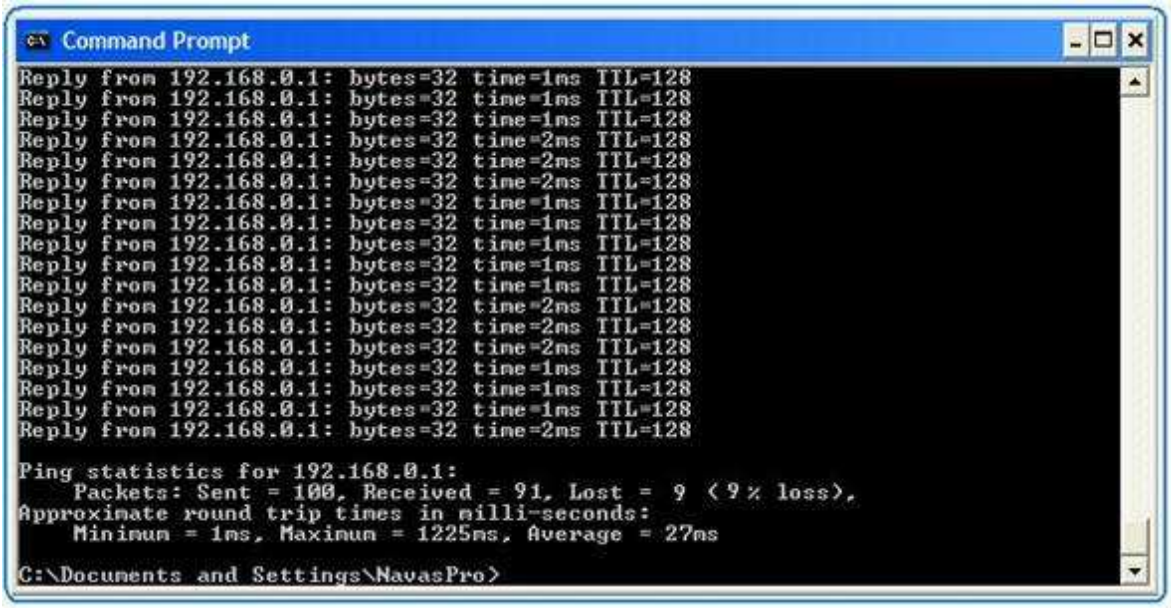


Figura 48 Ping entre las estaciones 12m

Velocidad

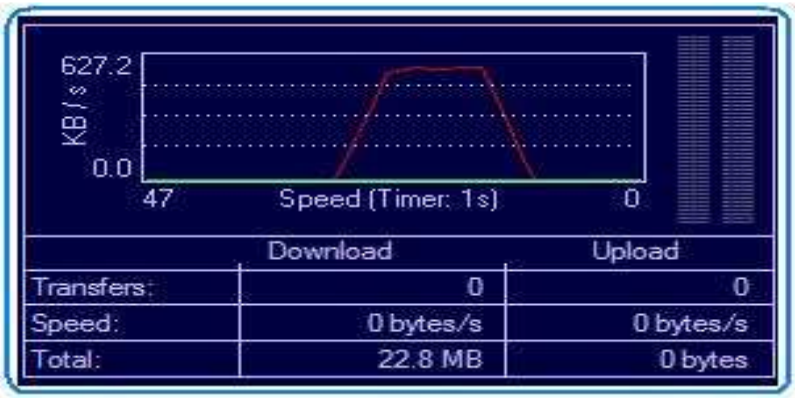


Figura 49 Velocidad 12m

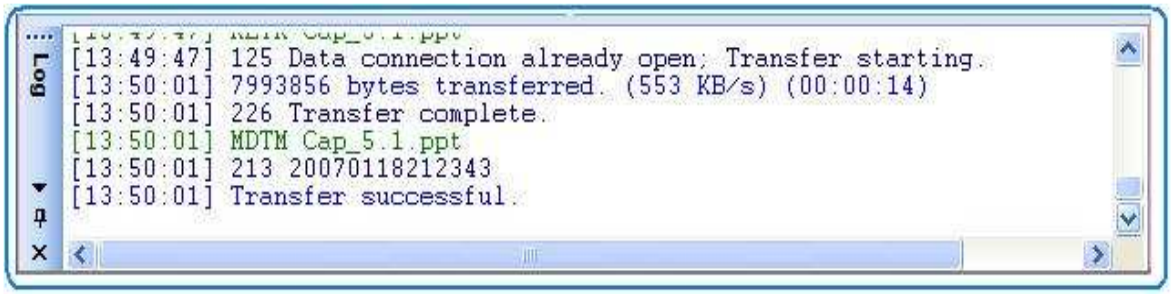


Figura 50 Velocidad Promedio 12m

Medidas a 15 metros de separación

Señal detectada

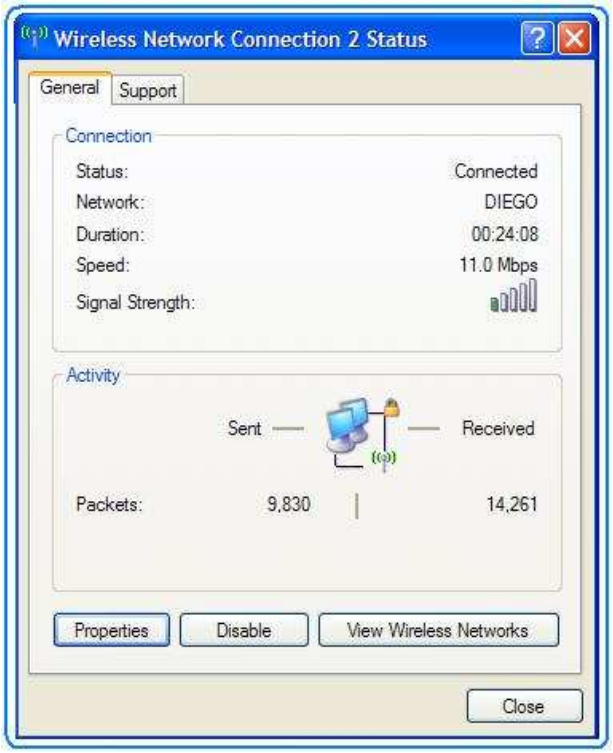


Figura 51 Estado de conexión 15m

Nivel de potencia (d = 15 m, p = - 70.50 dBm, f = 2.4400 GHz)

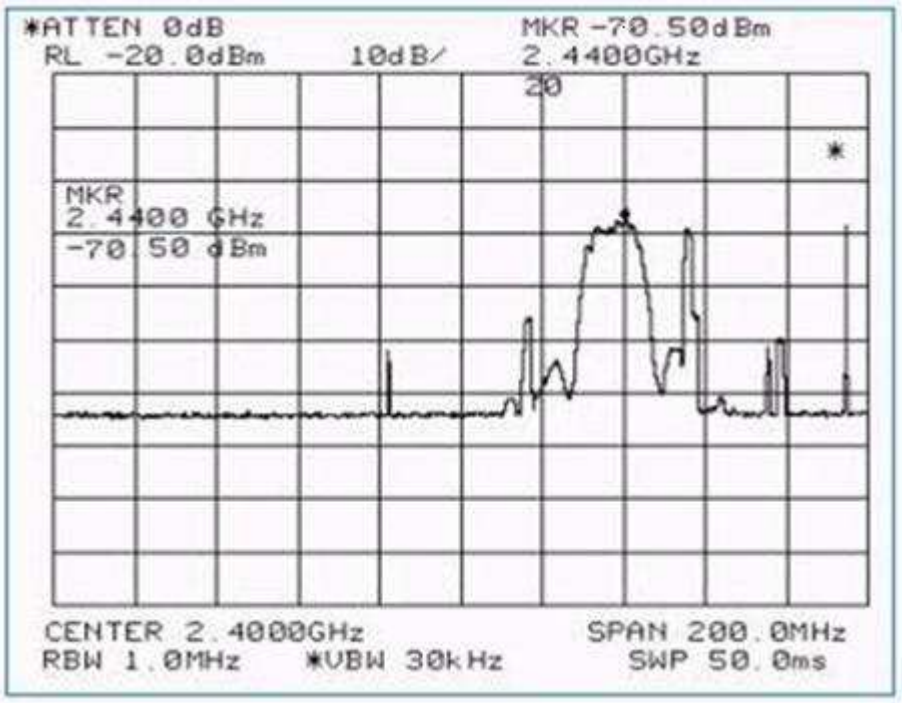


Figura 52 Nivel de potencia 15m

Respuesta entre las estaciones

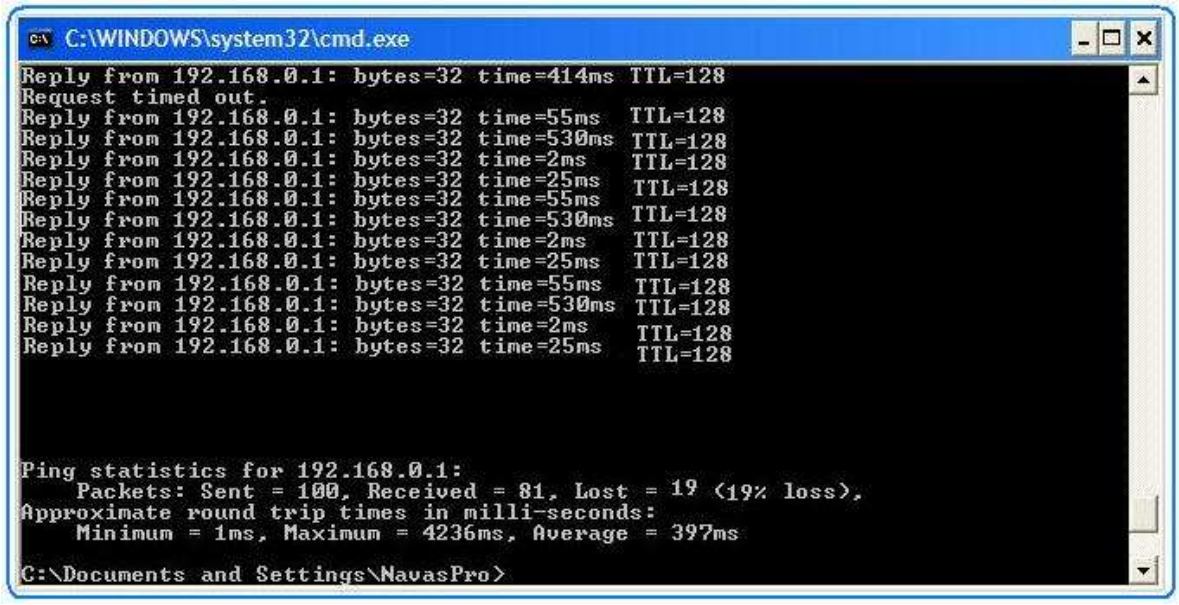


Figura 53 Ping entre las estaciones 15m

Velocidad

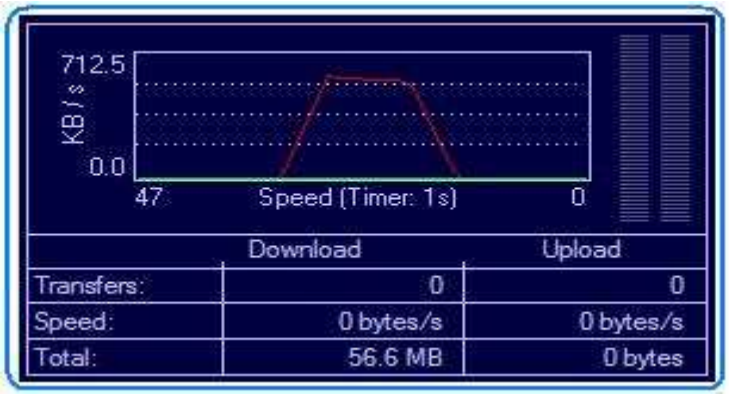


Figura 54 Velocidad 15m

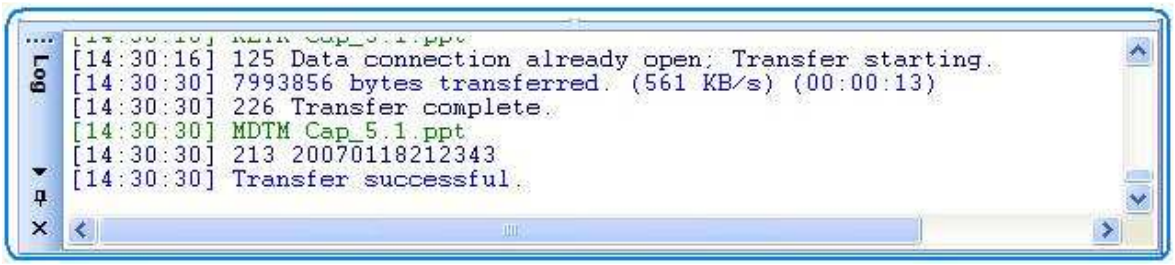


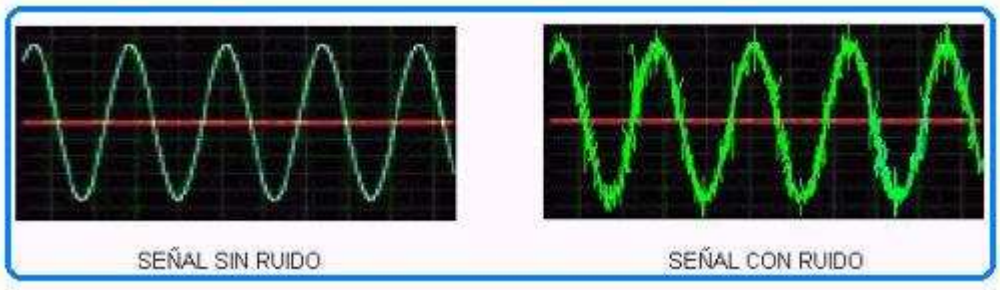
Figura 55 Velocidad Promedio 15m

ANEXO F

Ruido [11]

Ruido

El ruido es una señal de naturaleza aleatoria que contamina a la señal deseada.



Señal sin Ruido ***Señal con Ruido***

Figura. 1 Señal con ruido

El ruido reduce la capacidad del receptor para reconocer correctamente los símbolos, limitando de esta manera la velocidad de transmisión. Existe una gran cantidad de causas por las que el ruido se puede incrementar; a continuación se presenta una tabla resumida de las diferentes fuentes de ruido.

FUENTES DE RUIDO	**CAUSAS**
EXTERNAS	▪ Atmosféricas ▪ Galácticas ▪ Ruido generado por el hombre: Ignición, motores, otros.
INTERNAS	▪ Pérdidas disipativas ▪ Pérdidas de dispositivos
TÉRMICAS	▪ Por movimiento de electrones en los conductores ▪ Agitación térmica en todos los componentes

Tabla. 1 Fuentes de Ruido

A continuación solo definiremos el ruido térmico debido a que es el ruido con mayor importancia en un sistema electrónico de comunicaciones.

- **RUIDO TÉRMICO:** Este ruido se asocia con el movimiento rápido y aleatorio de los electrones dentro de un conductor, producido por la agitación térmica.

El ruido térmico es también llamado Ruido Blanco o de Jonson, esta presente en todas las componentes de frecuencia. Este ruido es predecible, aditivo, y esta presente en todos los dispositivos; es por ello que es el mas importante de todos los ruidos.

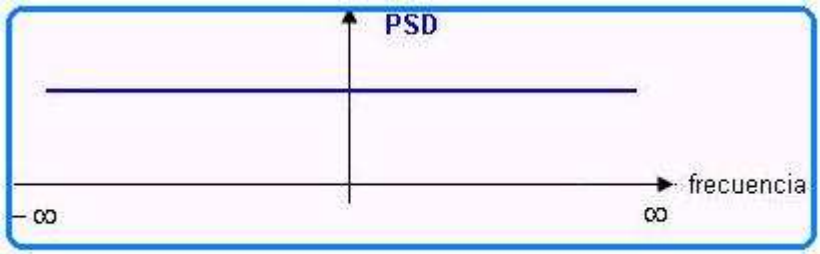


Figura. 2 Función Densidad Espectral de Potencia

$$PSD = N_0 = K \bullet T$$

T = Temperatura absoluta ($^0K = 273^0 + ^0C$)

K = Constante de Boltzman ($1.38 \times 10^{-}23$ J / ºK)

La potencia de ruido viene dado por la siguiente expresión:

$$N = N_0 \bullet AB$$

AB = ancho de banda (hertz)

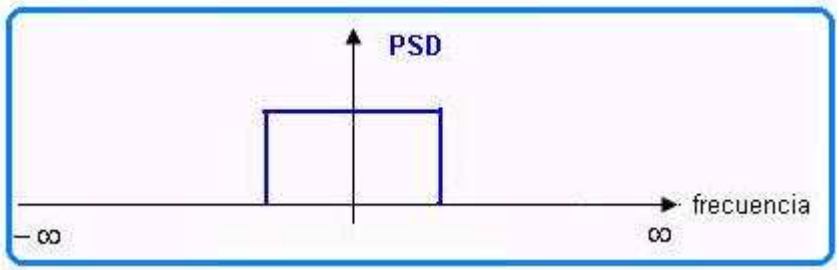


Figura. 3 Ruido Coloriado

ANEXO G

Modulación por división ortogonal de frecuencias (OFDM) [2]

Modulación por división ortogonal de frecuencias (OFDM)

Esta tecnología sólo está presente en 802.11a y en 802.11g como principal variación respecto a 802.11 y 802.11b. Se observa que la modulación pasa a ser OFDM (Orthogonal Frequency Division Multiplexing), en vez de la clásica y más fiable hasta entonces CCK (Complimentary Code Keying); aunque esta norma pueda coexistir en los puntos de acceso 802.11g, conservando a su vez la banda de los 2.4Ghz (precedido de un CCK RTS).

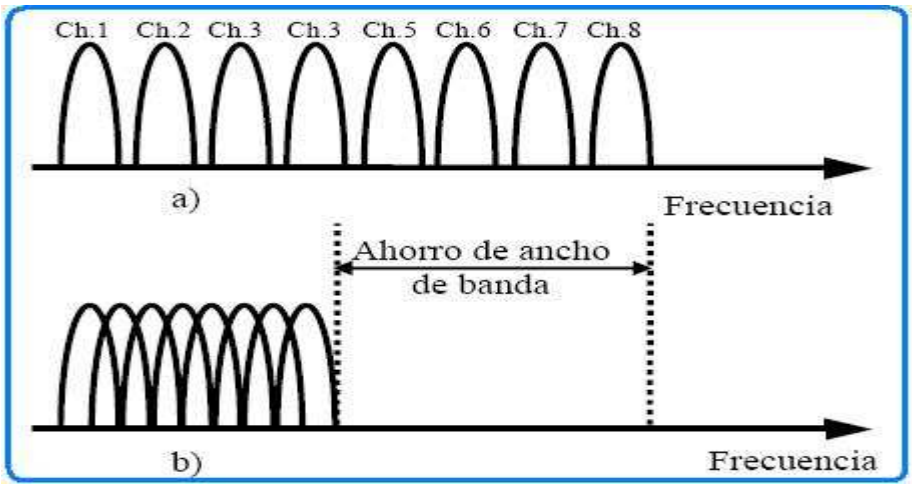


Figura 1 OFDM Orthogonal Frequency division Multiplexing

- Técnica multiportadora original.
- Modulación de portadoras ortogonales**.**

Durante los últimos años, se ha aceptado OFDM como tecnología de base para el 802.16a, que es un estándar de IEEE para redes de área metropolitana inalámbrica; y que puede proveer extensión inalámbrica para acceso de última milla de banda ancha en instalaciones de cable y DSL. El mismo cubre el rango de frecuencias de 2 a 11 GHz y alcanza hasta 50 kilómetros lineales, brindando conectividad de banda ancha inalámbrica sin necesidad de que exista una línea directa de visión a la estación de base. La velocidad de transmisión de datos puede llegar a 70 Mbps.

Una estación de base típica puede albergar hasta seis sectores. La calidad de servicio está integrada dentro del MAC, permitiendo la diferenciación de los niveles de servicio.

El origen del OFDM está en las décadas de los 50 y 60 en aplicaciones de uso militar, y trabaja dividiendo el espectro disponible en múltiples subportadoras. La transmisión sin línea de visión ocurre cuando entre el receptor y el transmisor existen reflexiones o absorciones de la señal, lo que resulta en una degradación de la señal recibida, que se manifiesta por medio de los siguientes efectos: atenuación plana, atenuación selectiva en frecuencia o interferencia inter-símbolo.

Estos efectos se mantienen bajo control con el W-OFDM, que es una tecnología propietaria de Wi-LAN, quién recibió en 1994 la patente 5.282.222 para comunicaciones inalámbricas de dos vías y banda ancha OFDM (WOFDM). Esta patente es la base para los estándares 802.11a, 802.11g, 802.11a R/A, 802.16a, estándares para HiperMAN.

Los sistemas W-OFDM incorporan además: estimación de canal, prefijos cíclicos y códigos Reed-Solomon de corrección de errores. Wi-LAN introdujo su línea de productos BWS 3000 basada en W-OFDM en octubre del 2001.

Actualmente ya ha introducido al mercado la tercera generación de equipos OFDM siendo el único proveedor mundial con una sólida experiencia en esta tecnología probada a través de la excelencia de sus productos.

Las tecnologías 802.11a y 802.11b definen una capa física diferente. Los emisores 802.11b transmiten a 2.4 GHz y envían datos a tasas tan altas como 11 Mbps usando modulación DSSS; mientras que los emisores 802.11a y 802.11g transmiten a 5 y 2,4 GHz respectivamente y envían datos a tasas de hasta 54 Mbps usando OFDM.

OFDM es una tecnología de modulación digital, una forma especial de modulación multi-portadora (multi-carrier) considerada la piedra angular de la próxima

generación de productos y servicios de radio frecuencia de alta velocidad, para uso tanto personal como corporativo. La técnica de espectro disperso de OFDM distribuye los datos en un gran número de portadoras (carriers) que están espaciados entre sí en distintas frecuencias precisas. Ese espaciado evita que los demoduladores vean frecuencias distintas a las suyas propias.

OFDM tiene una alta eficiencia de espectro y menor distorsión multi-ruta. Actualmente OFDM no sólo se usa en las redes inalámbricas LAN 802.11a y 802.11g, si no también en comunicaciones de alta velocidad por vía telefónica como las ADSL y en difusión de señales de televisión digital terrestre en Europa, Japón y Australia.

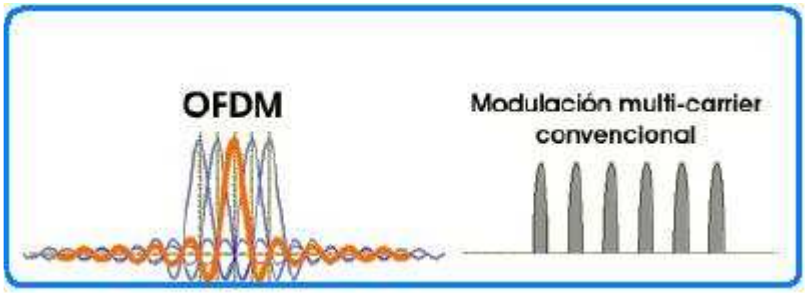


Figura 2 Espectro de OFDM solapado

ANEXO H

Modulación Wi-fi [2]

DSSS

FHSS

INFRARROJOS

- **Espectro Expandido por Salto de Frecuencia (FHSS)**

La tecnología de espectro expandido por salto en frecuencia (FHSS) consiste en transmitir una parte de la información en una determinada frecuencia durante un intervalo de tiempo llamado dwell time inferior a 400 ms. Pasado este tiempo se cambia la frecuencia de emisión y se sigue transmitiendo a otra frecuencia. De esta manera cada tramo de información se va transmitiendo en una frecuencia distinta durante un intervalo muy corto de tiempo.

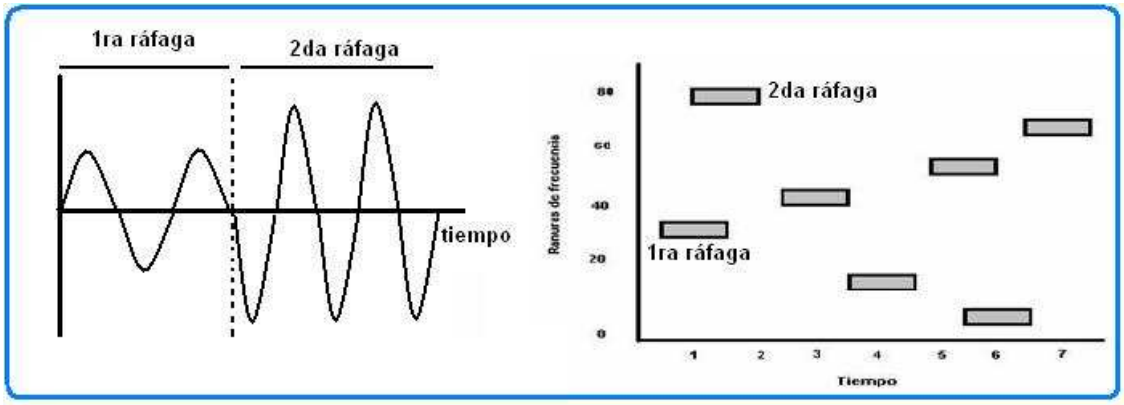


Figura 1 Codificación con Salto en Frecuencia

El orden en los saltos en frecuencia se determina según una secuencia pseudo aleatoria almacenada en unas tablas, que tanto el emisor y el receptor deben conocer.

Si se mantiene la sincronización en los saltos de frecuencias se consigue que, aunque en el tiempo se cambie de canal físico, a nivel lógico se mantiene un solo canal por el que se realiza la comunicación.

Esta técnica utiliza la zona de los 2.4GHz, la cual organiza en 79 canales con un ancho de banda de 1MHz cada uno. No obstante el número real de canales que son usados se regula por las autoridades competentes de cada país. El número de saltos por segundo está también regulado en cada país, así, por ejemplo, Estados Unidos fija una tasa mínima de saltos de 2,5 por segundo.
El estándar IEEE 802.11 define la modulación aplicable en este caso. Se utiliza la modulación en frecuencia FSK (Frequency Shift Keying), con una velocidad

de 1 Mbps ampliable a 2 Mbps. En la revisión 802.11b del estándar, la velocidad también ha aumentado a 11Mbps.

Formato de la trama FHSS

Figura 2 Trama FHSS

- **Preámbulo:** contiene dos subcampos separados: el campo de preámbulo de sincronización (SYNC) y el delimitador de comienzo de trama (Start Frame Delimiter).

- **Sincronismo:** contiene 80 bits con un patrón alternativo de unos-ceros, comenzando con cero y terminando con uno. Se usa para detectar una señal potencialmente válida, seleccionar una de las antenas si se usa un sistema de diversidad y sincronizarse temporalmente.

- **Delimitador de comienzo de trama (SFD):** contiene un patrón de 16 bits con patrón 0000 1100 1011 1101 que define el tempo de la trama.

- **Cabecera:** contiene 3 subcampos: Longitud de 12 bits, Señalización de 12 bits y Control de Errores de 16 bits.

- **Longitud:** indica la longitud del campo de datos que puede ser de hasta 4095 octetos.

- **Señalización:** campo de 4 bits que indica la velocidad de transmisión de los datos desde 1 Mbps a 4.5 Mbps en incrementos de 0.5 Mbps.

- **Control de errores (HEC):** campo de 16 bits para detección de errores que utiliza el polinomio generador CCITT CRC-16 $G(x) = X^{16} + X^{12} + X^{15} + 1$

El preámbulo y la cabecera son siempre transmitidos a 1 Mbps. El resto de la trama es transmitido a la velocidad indicada en el campo de señalización. Para minimizar el efecto de las reflexiones multitrayecto el FHSS tiene un salto de distancia mínima entre frecuencias. Esto es debido a que las reflexiones del salto anterior tienen un efecto mínimo sobre el siguiente salto debido a que, transcurrido el retardo producido por la reflexión hasta llegar al receptor, éste se encontrará entonces esperando por información en una frecuencia diferente.

❖ Espectro Expandido por Secuencia Directa (DSSS)

DSSS es el segundo tipo de modulación soportado por el IEEE 802.11 y el único especificado en el IEEE 802.11b, soportando velocidades de transmisión de 5.5 y 11Mbps.

En el caso de Estados Unidos y Europa la tecnología DSSS utiliza un rango de frecuencias que va desde los 2,4 GHz hasta los 2,4835 GHz, lo que permite tener un ancho de banda total de 83,5 MHz. Este ancho de banda se subdivide en canales de 5 MHz, lo que hace un total de 14 canales independientes. Cada país esta autorizado a utilizar un subconjunto de estos canales. En Europa existen 13 canales disponibles, excepto en Francia de los cuales solo 3 no están solapados.

Con arreglo a IEEE 802.11 debe existir una separación de 30 MHz entre las frecuencias centrales de los canales si las celdas se solapan y/o son adyacentes para no causar interferencias. En IEEE 802.11b la separación se reduce a 25 MHz. Esto significa que pueden existir 3 celdas con zonas solapadas y/o adyacentes sin causar interferencias entre ellas, tal y como se muestra en la Figura. 1.28

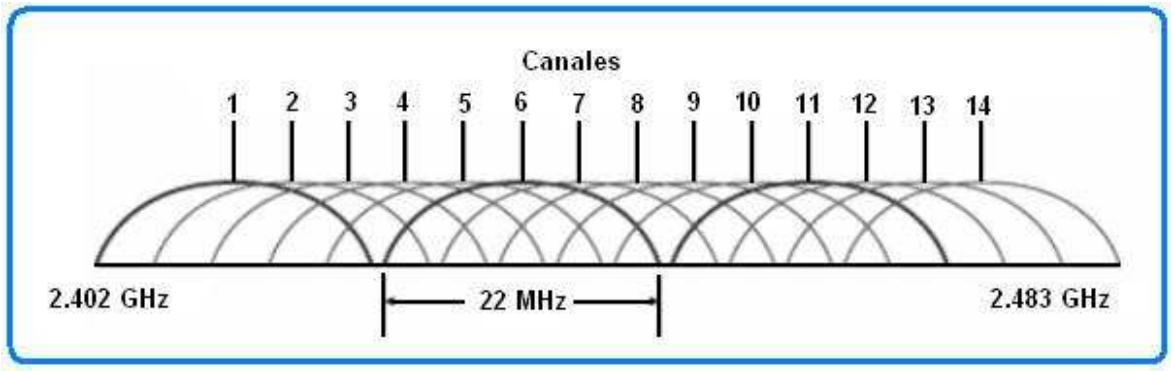


Figura 3 Canales DSSS

En configuraciones donde existan mas de una celda, éstas pueden operar simultáneamente y sin interferencias, siempre y cuando la diferencia entre las frecuencias centrales de las distintas celdas sea de al menos 30 MHz, lo que reduce a tres el número de canales independientes y funcionando simultáneamente en el ancho de banda total de 83,5 MHz.

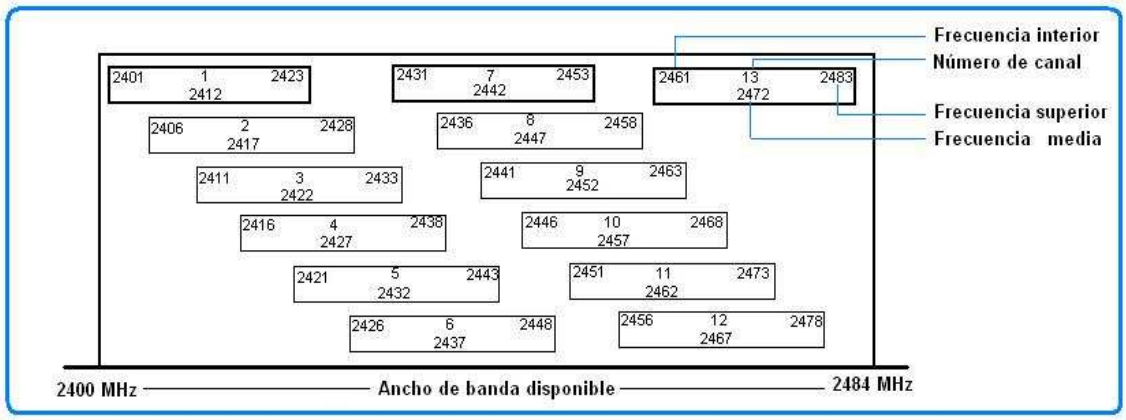


Figura 4 Tabla de Frecuencias DSSS

Formato de la trama DSSS

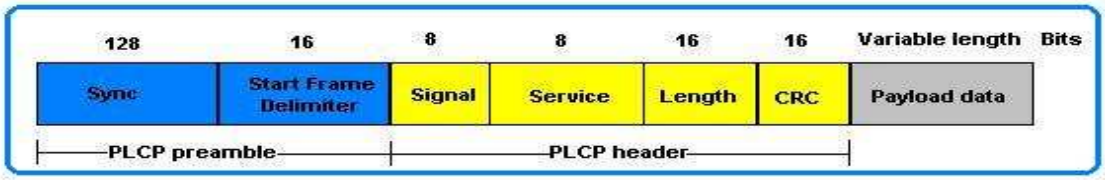


Figura 5 Trama DSSS

Al igual que en FHSS el preámbulo y la cabecera se transmiten siempre a 1 Mbps y el campo de señalización indica la velocidad de transmisión de los

datos. En el 802.11b este campo soporta velocidades mayores que el original de 1 y 2 Mbps (5,5 Mbps y 11 Mbps).

- **Sincronismo:** contiene una codificación de 128 bits que garantiza la sincronización previa del receptor.

- **Delimitador de comienzo de trama (SFD):** señaliza el comienzo de la trama real después del preámbulo.

- **Señal:** indica a la capa física que tipo de modulación se utilizara en la transmisión. La velocidad será igual al valor de este campo multiplicado por 1000Kbps.

- **Servicio:** reservado para usos futuros.

- **Longitud:** entero sin signo de 16 bits que indica el número de microsegundos requerido para transmitir los datos.

- **CRC:** los campos de cabecera están protegidos por una secuencia de verificación de trama CRC-16.

Proceso de modulación DSSS

En esta técnica se genera un patrón de bits redundante (señal de chip) para cada uno de los bits que componen la señal. Cuanto mayor sea esta señal, mayor será la resistencia de la señal a las interferencias. El estándar IEEE 802.11 recomienda un tamaño de 11 bits, pero el óptimo es de 100. En recepción es necesario realizar el proceso inverso para obtener la información original.

La secuencia de bits utilizada para modular los bits se conoce como secuencia de Barker (también llamado código de dispersión o Pseudo Noise). Es una secuencia rápida diseñada para que aparezca aproximadamente la misma

cantidad de 1 que de 0. Un ejemplo de esta secuencia es el siguiente:+1 -1 +1 +1 -1 +1 +1 +1 -1 -1 -1 -1.

Solo los receptores a los que el emisor haya enviado previamente la secuencia podrán recomponer la señal original. Además, al sustituir cada bit de datos a transmitir por una secuencia de 11 bits equivalente, aunque parte de la señal de transmisión se vea afectada por interferencias, el receptor aún puede reconstruir fácilmente la información a partir de la señal recibida.

A continuación podemos observar como se utiliza la secuencia de Barker para codificar la señal original a transmitir:

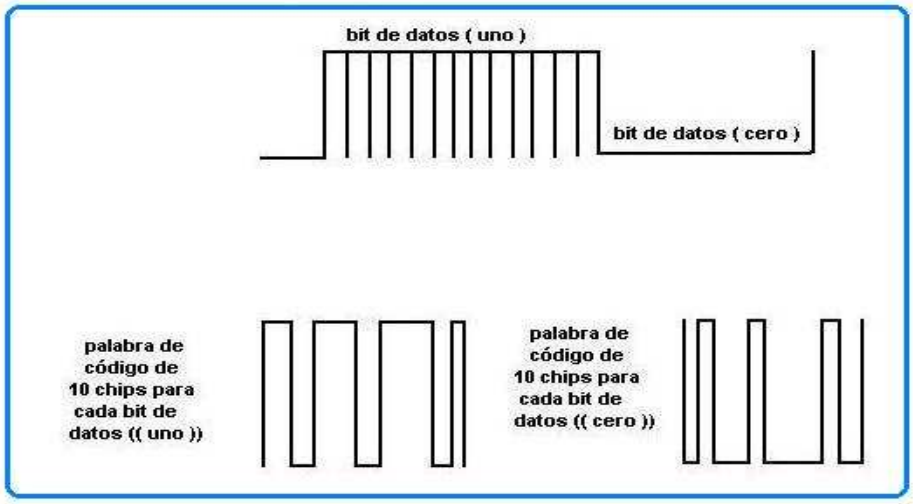


Figura 6 Secuencia Directa (DSSS)

Una vez aplicada la señal de chip, el estándar IEEE 802.11 ha definido dos tipos de modulación para la técnica de espectro ensanchado por secuencia directa (DSSS), la modulación DBPSK (Differential Binary Phase Shift Keying) y la modulación DQPSK (Differential Quadrate Phase Shift Keying), que proporcionan una velocidad de transferencia de 1 y 2 Mbps respectivamente.

- **Infrarrojo**

La verdad es que IEEE 802.11 no ha desarrollado todavía en profundidad esta área y solo menciona las características principales de la misma:

Entornos muy localizados, un aula concreta, un laboratorio, un edificio.

- Modulaciones de 16-PPM y 4-PPM que permiten 1 y 2 Mbps de transmisión.
- Longitudes de onda de 850 a 950 nanómetros de rango.
- Frecuencias de emisión entre 3,15*10E14 Hz y 3,52*10E14 Hz.

Las WLAN por infrarrojos son aquellas que usan el rango infrarrojo del espectro electromagnético para transmitir información mediante ondas por el espacio libre. Los sistemas de infrarrojos se sitúan en altas frecuencias, justo por debajo del rango de frecuencias de la luz visible. Las propiedades de los infrarrojos son, por tanto, similares a las que tiene la luz visible. De esta forma los infrarrojos son susceptibles de ser interrumpidos por cuerpos opacos pero se pueden reflejar en determinadas superficies.

Para describir esta capa física seguiremos las especificaciones del IrDA (Infrared Data Association) organismo que ha estado desarrollando estándares para conexiones basadas en infrarrojos.

Para la capa infrarroja tenemos las siguientes velocidades de transmisión:

- 1 y 2 Mbps Infrarrojos de modulación directa.
- 4 Mbps mediante Infrarrojos portadora modulada.
- 10 Mbps Infrarrojos con modulación de múltiples portadoras.

c.1 Clasificación de Infrarrojo

De acuerdo al ángulo de apertura con que se emite la información en el transmisor, los sistemas infrarrojos pueden clasificarse en sistemas de corta apertura, también llamados de rayo dirigido o de línea de vista (*line of sight*, LOS) y en sistemas de gran apertura, reflejados o difusos (*diffused*).

- **Los sistemas infrarrojos de corta apertura:** están constituidos por un cono de haz infrarrojo altamente direccional y funcionan de manera

similar a los controles remotos de las televisiones: el emisor debe orientarse hacia el receptor antes de empezar a transferir información, limitando por tanto su funcionalidad. Resulta muy complicado utilizar esta tecnología en dispositivos móviles, pues el emisor debe reorientarse constantemente. Este mecanismo solo es operativo en enlaces punto a punto exclusivamente. Por ello se considera que es un sistema inalámbrico pero no móvil, o sea que está más orientado a la portabilidad que a la movilidad.

- **Los sistemas de gran apertura:** permiten la información en ángulo mucho más amplio por lo que el transmisor no tiene que estar alineado con el receptor. Una topología muy común para redes locales inalámbricas basadas en esta tecnología, consiste en colocar en el techo de la oficina un nodo central llamado punto de acceso, hacia el cual dirigen los dispositivos inalámbricos su información, y desde el cual ésta es difundida hacia esos mismos dispositivos.

  La dispersión utilizada en los sistemas de gran apertura, hace que la señal transmitida rebote en techos y paredes, introduciendo un efecto de interferencia en el receptor, que limita la velocidad de transmisión (la trayectoria reflejada llega con un retraso al receptor). Esta es una de las dificultades que han retrasado el desarrollo del sistema infrarrojo en la norma 802.11.

ANEXO I

SCRIPTS

SCRIPT efectiva.pl

```
#PAGINA  35
# type: perl Throughput.pl <trace flie> <requerid node> <granularity> file

$infile=$ARGV[0];
$tonode=$ARGV[1];
$granularity=$ARGV[2];


#calculamos  cuantos bytes fueron transmitidos durante el intervalo de tiempo especificado
#Por el parametro granularity en segundos

$sum=0;
$clock=0;

        open (DATA,"<$infile")
          || die "Can't open $infile $!";

        while (<DATA>) {
                @x= split(' ');


#if ($x[1] >= 4.0)
#{


#columna 1 es el tiempo

if ($x[1]-$clock <= $granularity)
{

#chequeo si los eventos corresponden a recividos

if ($x[0] eq 'r')
{

#OJO AQUI
#chequeo si el destino corresponde al primer argumento
if ($x[2] eq $tonode)
{

#chequeo si el paquete es TCP
if ($x[6] eq 'tcp')
{

        $sum=$sum+$x[7];

}
```

```
}
}
}

else
{       $throughput=8.0*$sum/$granularity;


#       $dis=$x[1]-2.0;


        if ($x[1] >= 4.0)
{

       $dis=$x[1]-2.0;
        print STDOUT "$dis $throughput\n";
        $clock=$clock+$granularity;
        $sum=0;
}
}

}

        $throughput=8.0*$sum/$granularity;

#       $dis=$x[1]-2.0;

        print STDOUT "$x[1] $throughput\n";
        $clock=$clock+$granularity;
        $sum=0;


        close DATA;

#}

exit(0);
```

SCRIPT wi-fi.tcl

```
#PAGINA  35
# type: perl Throughput.pl <trace flie> <requerid node> <granularity> file

$infile=$ARGV[0];
$tonode=$ARGV[1];
$granularity=$ARGV[2];


#calculamos  cuantos bytes fueron transmitidos durante el intervalo de tiempo especificado
#Por el parametro granularity en segundos

$sum=0;
$clock=0;

      open (DATA,"<$infile")
        || die "Can't open $infile $!";

      while (<DATA>) {
            @x= split(' ');

#columna 1 es el tiempo

if ($x[1]-$clock <= $granularity)
{

#chequeo si los eventos corresponden a recividos

if ($x[0] eq 'r')
{

#OJO AQUI
#chequeo si el destino corresponde al primer argumento
if ($x[2] eq $tonode)
{
#chequeo si el paquete es TCP
if ($x[6] eq 'tcp')
{

      $sum=$sum+$x[7];

}
}
}
```

```
}

else
{       $throughput=8.0*$sum/$granularity;


        print STDOUT "$x[1] $throughput\n";
        $clock=$clock+$granularity;
        $sum=0;
}

}

        $throughput=8.0*$sum/$granularity;

        print STDOUT "$x[1] $throughput\n";
        $clock=$clock+$granularity;
        $sum=0;


        close DATA;

exit(0);
```

ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA

ESTUDIO COMPARATIVO ENTRE LAS TECNOLOGÍAS BLUETOOTH Y WI-FI EN AMBIENTES DE CORTO ALCANCE A TRAVÉS DE LA IMPLEMENTACIÓN DE DOS PROTOTIPOS Y DE SU SIMULACIÓN

PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN ELECTRÓNICA Y TELECOMUNICACIONES

ARIAS PILAQUINGA DIEGO BOLIVAR

MUELA VACA DIEGO FRANCISCO

DIRECTORA: MSc. SORAYA SINCHE

Quito, Diciembre 2007

DECLARACIÓN

Nosotros, Arias Pilaquinga Diego Bolivar, Muela Vaca Diego Francisco, declaramos bajo juramento que el trabajo aquí descrito es de nuestra autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que hemos consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedemos nuestros derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

Arias Pilaquinga Diego Bolivar

Muela Vaca Diego Francisco

CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por los señores Arias Pilaquinga Diego Bolivar y Muela Vaca Diego Francisco, bajo mi supervisión.

MSc. Soraya Sinche
DIRECTORA DEL PROYECTO

AGRADECIMIENTO

A Dios quien día a día me ha bendecido y me ha permitido seguir viviendo para alcanzar esta nueva meta.

A mis padres por haberme guiado por el sendero de la verdad y la justicia.

A mi abuelita quien me cuido cuando fui pequeño y me ha recordado siempre que debo cumplir con mi meta.

A mis hermanos Marco, Polo y Abdala, quienes me han apoyado y que con sus palabras de aliento me han hecho sentir una persona diferente.

A mis profesores forjadores de una juventud noble y justa que mañana harán del Ecuador una patria más libre y próspera.

A la MSc. Soraya Sinche por haber tenido paciencia en la dirección del presente Proyecto de Titulación.

A mis amigos y amigas por sus sabios consejos que han hecho que vaya por un camino de bien.

A la escuela Simón Bolívar de Angamarca, al Instituto Nacional Mejía y a la Escuela Politécnica Nacional cuyas aulas son testigas de sueños que hoy los veo realizados.

A mis novias quienes en su momento me brindaron su cariño y admiración.

Y a todas aquellas personas que de una u otra forma ayudaron y colaboraron para que se cristalice este nuevo sueño.

DIEGO ARIAS

AGRADECIMIENTO

A Dios por permitirme vivir día a día, y colmarme de bendiciones.

A mis padres David Muela y Marina Vaca, a mis hermanos Blanqui, Yoli, Gloria, Carlos, Pepe, Fernando, Anita y a mis cuñados y cuñadas por su apoyo incondicional que me han brindado. Gracias a ellos hoy puedo ver la culminación de una etapa más del camino de mi vida

A todos mis profesores de la Escuela Politécnica Nacional quienes me inculcaron todos los conocimientos para llegar a ser un profesional más del Ecuador.

A todos mis amigos por el apoyo incondicional en el desarrollo del Proyecto de Titulación

A la MSc. Soraya Sinche por haberme guiado y sobre todo haber tenido mucha paciencia en la dirección del presente Proyecto de Titulación.

Y a todas aquellas personas que de una u otra forma ayudaron y colaboraron para que se cristalice este nuevo sueño.

DIEGO MUELA

DEDICATORIA

Dedico este trabajo a mis padres, especialmente a mi madre quien deposito su confianza en mí desde que fui un niño.

También dedico este trabajo a todas las personas que no creyeron en mí y a mis enemigos ya que gracias a ellos fui adquiriendo fuerzas para seguir adelante.

DIEGO ARIAS

DEDICATORIA

Dedico este trabajo a mis padres y mi familia, por haber depositado toda su confianza en mí.

Dedico este trabajo a mi hermano Carlos y a mi cuñada Raquel por todo el apoyo brindado en el colegio y la universidad.

También dedico este trabajo a todas las personas que no creyeron en mí ya que gracias a ellos fui adquiriendo fuerzas para seguir adelante.

DIEGO MUELA

ÍNDICE GENERAL

ÍNDICE DE FIGURAS

CAPÍTULO 1

CAPÍTULO 2

CAPÍTULO 3

CAPÍTULO 4

ÍNDICE DE TABLAS

CAPÍTULO 1

CAPÍTULO 2

CAPÍTULO 4

RESUMEN

El presente Proyecto de Titulación se enfoca a la comparación de las tecnologías inalámbricas *Bluetooth* y *Wi-Fi* que se están utilizando en la actualidad, que hacen posible la implementación de redes inalámbricas personales, las cuales debido a su eficiencia y desempeño, son de gran utilidad para la prestación de servicios.

El Proyecto de titulación consta de las siguientes partes:

En el capítulo uno se analiza las tecnologías *Bluetooth* y *Wi-Fi*, se realiza un estudio de las características fundamentales que posee cada tecnología y sus posibles aplicaciones. Además se realiza un estudio de los factores de propagación inalámbrica y de los modelos para el cálculo del enlace.

En el capítulo dos se realiza el diseño de los prototipos inalámbricos con tecnología *Bluetooth* y *Wi-Fi*, se analizan las alternativas del acceso inalámbrico con estas tecnologías y se realizan las pruebas correspondientes, en las que se puede evaluar la correcta operación de los prototipos implementados, al igual que se determinan las limitaciones de los mismos.

En el capítulo tres se simula el funcionamiento de los prototipos, en la simulación se puede verificar cómo varía la velocidad, potencia, relación señal a ruido en función de la distancia.

Para la simulación de los prototipos de prueba se utiliza el ns-2, que permite simular eventos discretos, redes telemáticas e inalámbricas.

En el capítulo cuatro se realiza la comparación de resultados obtenidos en la implementación práctica y en la simulación de los prototipos. En base a estos resultados se determina cuál de las tecnologías se comporta de mejor manera en ambientes de corto alcance.

En el capítulo cinco se comentan las diferentes conclusiones y recomendaciones concernientes al desarrollo del proyecto de titulación.

En los anexos del proyecto de titulación se encuentra información utilizada en el desarrollo del mismo.

PRESENTACIÓN

La posibilidad de comunicarse en cualquier momento y desde cualquier lugar sin estar sujeto a un sitio específico, ha sido desde siempre uno de los principales objetivos de los sistemas de comunicación. En los últimos años se han ido desarrollando diversas aplicaciones para tecnologías inalámbricas las cuales facilitan a sus usuarios acceder a información y recursos en tiempo real sin necesidad de estar conectados a una red determinada.

Para el logro de los objetivos planteados en este proyecto se ha recopilado una cantidad valiosa de información de las tecnologías inalámbricas *Bluetooth* y *Wi-Fi*, que puede ser útil al momento de realizar consultas por los estudiantes.

El presente proyecto contiene las pautas necesarias para el diseño e implementación de prototipos inalámbricos con tecnología *Bluetooth* y *Wi-Fi*, en ambientes de corto alcance, además brinda al público en general la posibilidad de escoger qué tecnología utilizar de acuerdo a sus requerimientos.

Para la simulación de los prototipos *Bluetooth* y *Wi-Fi* se utilizó el simulador ns-2, conjuntamente con la librería *UCBT*. Este simulador puede convertirse en una excelente herramienta de aprendizaje para los estudiantes de ingeniería al momento de simular redes inalámbricas.

CAPÍTULO 1

1. ESTUDIO DE LAS TEGNOLOGÍAS BLUETOOTH Y WI-FI

En los últimos años las redes de área local inalámbricas *WLAN*, están ganando mucha aceptación, que va creciendo conforme aumentan sus prestaciones y se descubren nuevas aplicaciones para ellas. Las *WLAN* permiten a sus usuarios acceder a información y recursos en tiempo real sin necesidad de estar conectados a una red determinada.

Con las *WLAN* se elimina la necesidad de usar cables y establece nuevas aplicaciones añadiendo flexibilidad a la red. Un usuario dentro de una red *WLAN* puede transmitir y recibir voz, datos y vídeo dentro de edificios, entre edificios o campus e incluso sobre áreas metropolitanas a velocidades de 11 *Mbps* o superiores.

Las *WLAN* ofrecen una fácil incorporación de nuevos usuarios a la red, movilidad de los usuarios, bajo costo de implementación comparado con los sistemas cableados, velocidades de transmisión aceptables y áreas de cobertura que dependen de la potencia del interfaz utilizado, etc.

Es por esta razón que el objetivo principal del proyecto de titulación es la comparación de las tecnologías inalámbricas *Bluetooth* y *Wi-Fi*, en ambientes de corto alcance, es decir en distancias limitadas, ya que en la actualidad estas tecnologías son las utilizadas para este propósito.

1.1 BLUETOOTH [17]

Bluetooth es una tecnología estandarizada por la *IEEE 802.15.1*, define un estándar global de comunicaciones inalámbricas de corto alcance, que posibilita la transmisión de voz y datos tanto para estaciones fijas y móviles (*PCs*, teléfonos móviles, *PDA* (Asistentes Personales Digitales), etc.).

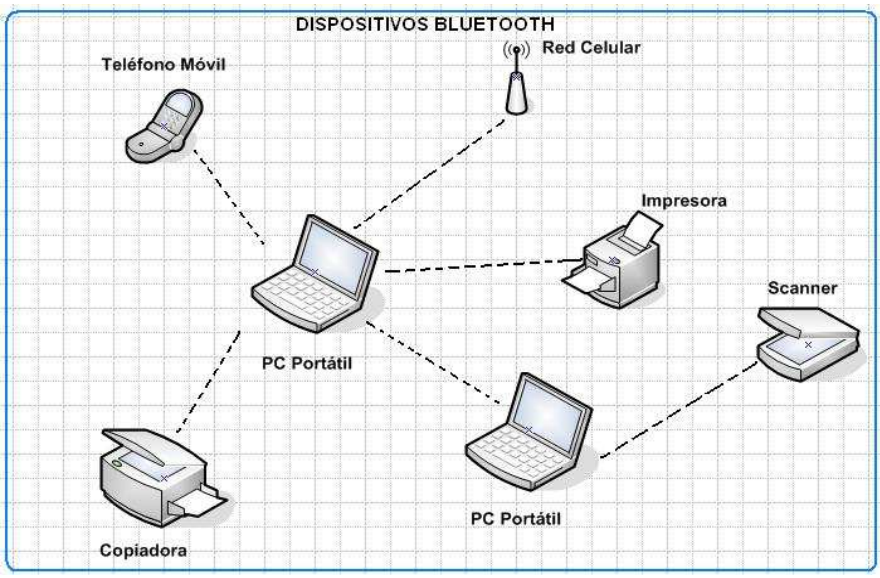


Figura 1.1 Equipos *Bluetooth*

Los objetivos que se persigue con este estándar son:

- Proporcionar comunicaciones entre equipos móviles y fijos
- Excluir cables y conectores entre éstos.
- Brindar la posibilidad de crear pequeñas redes inalámbricas y facilitar la sincronización de datos entre los equipos de trabajo.

1.1.1 EVOLUCIÓN [15] [17] [33]

En 1994 *Ericsson* investigó la viabilidad de utilizar un interfaz de radio, para la interconexión de teléfonos móviles y otros accesorios, con el fin de eliminar los cables existentes entre aparatos relativamente cercanos. El estudio partía de un largo proyecto que investigaba sobre multi-comunicadores conectados a una red celular, hasta que se llegó a un enlace de radio de corto alcance. Conforme avanzaba este proyecto, se estableció que este tipo de enlace podía ser utilizado en un gran número de aplicaciones, ya que tenia como principal virtud el que se basaba en un chip de radio de bajo costo y de corto alcance.

A principios de 1998 se creó el *SIG* (*Special Interest Group*, Grupo de Interés Especial) y estuvo integrado por 5 promotores que fueron: *Ericsson, Nokia, IBM, Toshiba e Intel*. La idea era lograr un conjunto adecuado de áreas de negocio.

Bluetooth se basó en el *SIG* y definió el estándar *IEEE 802.15.1* con el propósito principal, de establecer una interfaz aérea junto con su *software* de control, con el fin de asegurar la interoperabilidad de los equipos entre los diversos fabricantes.

"En la actualidad el *SIG* cuenta con miembros tales como *Motorola, 3Com*, *Lucent* y *Microsoft*, el respaldo de 1900 empresas de tecnología y 2000 empleados (delegados en el Congreso convocado por el *SIG*) de otras tantas empresas que investigan productos y servicios con aplicaciones *Bluetooth*." [13]

1.1.2 ARQUITECTURA BLUETOOTH [34]

La especificación *Bluetooth* utiliza una arquitectura de protocolos que divide las diversas funciones de red en un sistema de niveles. En conjunto, permiten el intercambio transparente de información entre aplicaciones diseñadas de acuerdo con dicha especificación y fomentan la interoperabilidad entre los productos de diferentes fabricantes.

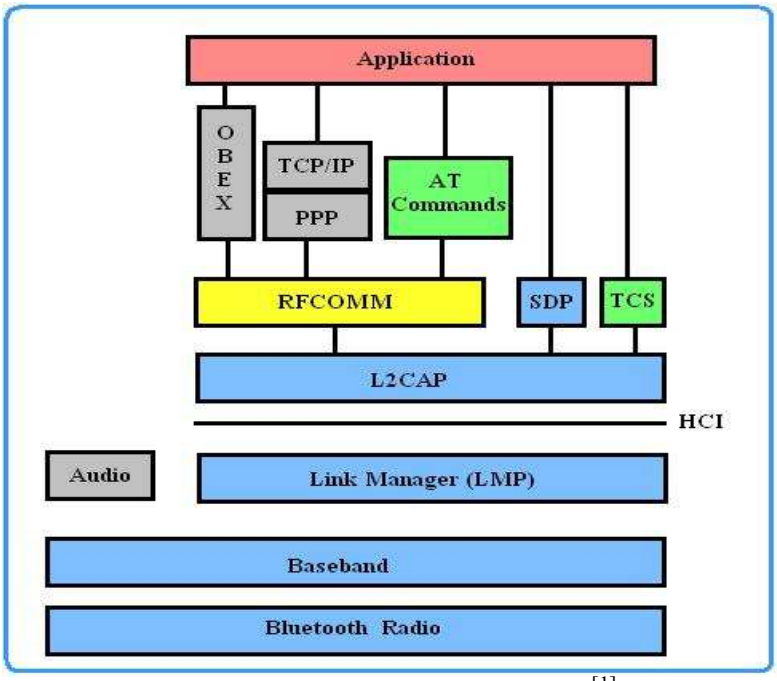


Figura 1.2 Arquitectura *Bluetooth* [1]

Cada aplicación puede operar bajo una estructura de protocolos definida por cada columna como se observa en la figura 1.2, o por un conjunto de ellas. Algunas columnas son usadas sólo como soporte de la aplicación principal, como lo son el

SDP (Protocolo de Descubrimiento de Servicio) y el *TCS* (Especificación de Control Telefónico).

La especificación es abierta, lo que permite el desarrollo de nuevos protocolos de aplicación en las capas superiores, lo cual se traduce en el desarrollo de una gran variedad de servicios por parte de los fabricantes.

Los protocolos pueden ser divididos de la siguiente forma:

- Protocolos Bluetooth Centrales (*BaseBand*, *LMP, L2CAP, SDP*).
- Protocolos de Reemplazo de Cable (*RFCOMM*).
- Protocolos de control de Telefonía (*TCS Binary, AT-Commands*).
- Protocolos Adaptados (*PPP, UDP/TCP/IP, OBEX, WAP, vCard, vCal, IrMC, WAE*).

El Grupo *Bluetooth SIG*, ha desarrollado los protocolos de la primera capa, los cuales son usados por la mayoría de los dispositivos *Bluetooth*. Por otra parte, el *RFCOMM* y el *TCS Binary* fueron desarrollados por el *SIG*, basándose en las especificaciones *ETSI-TS 07.10* y la *ITU-T Q.931*, respectivamente.

1.1.2.1 Capa Radio *Bluetooth*

En este nivel se especifica detalles del interfaz aire como: bandas de frecuencia, arreglos de canales, saltos de frecuencia, esquema de modulación y niveles de potencia.

1.1.2.1.1 Banda de Frecuencia utilizada por Bluetooth [2]

Para que *Bluetooth* opere globalmente, es indispensable que trabaje en una banda no licitada. La banda *ISM* (*Industrial, Scientific and Medical*, Industrial, Científica y Médica) de 2,45 *GHz* cumple con este requisito, con rangos que van de los 2.4 *GHz* a los 2.5 *GHz*, con algunas limitaciones en países como Francia, España y Japón. En la tabla 1.1 se muestra los rangos de frecuencia permitidos en diferentes regiones del mundo.

Ubicación Geográfica	Rango Regulatorio	Canales RF
USA, Europa	2.400 – 2.4835 GHz	F = 2402+ K*MHz , K = 0,....,78
España	2.445 – 2.475 GHz	F = 2449+ K*MHz , K = 0,....,22
Francia	2.4465 – 2.4835 GHz	F = 2454+ K*MHz , K = 0,....,22

Tabla 1.1 Bandas de frecuencia y canales de *RF Bluetooth* [2]

1.1.2.1.2 Espectro Ensanchado por Salto de Frecuencia [24]

Puesto que la banda *ISM* puede ser accedida sin necesidad de licencia, el sistema de radio *Bluetooth* deberá estar capacitado para evitar las múltiples interferencias que se pueden producir. Éstas pueden ser evitadas utilizando un sistema de Espectro Ensanchado por Salto de Frecuencia.

Este sistema divide la banda de frecuencia en 79 canales con un ancho de banda de 1 *MHz* cada canal, donde, los transceptores, durante la conexión van cambiando de uno a otro canal de salto de manera pseudo-aleatoria. Con lo que se consigue que el ancho de banda instantáneo sea muy pequeño y se tenga una propagación efectiva sobre el total del ancho de banda. En la figura 1.3 se observa el esquema de funcionamiento del sistema de (*FHSS)* Espectro Ensanchado por Salto de Frecuencia para *Bluetooth*.

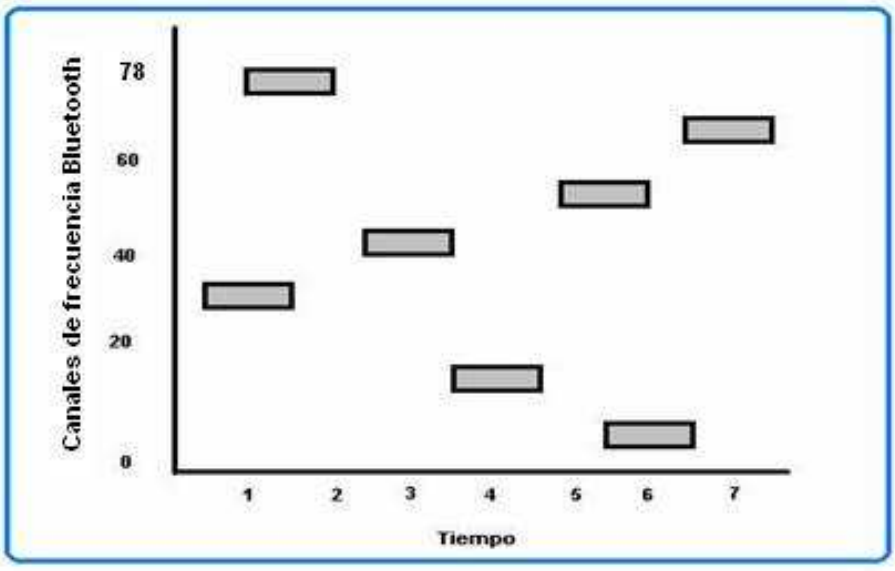


Figura 1.3 *FHSS* Salto de Frecuencia [26]

En conclusión, con el sistema *FHSS*, se pueden conseguir transceptores de banda estrecha con una gran inmunidad a las interferencias.

1.1.2.1.3 Modulación [3] [13]

En la banda de 2.4 *GHz* el ancho de banda para los sistemas *FH* está limitada en 1 *MHz*. El ancho de banda disponible es de 79 *MHz*, por lo que se dispone de 79 canales de salto en América.

Bluetooth utiliza una modulación *GFSK* (*Gaussian Frecuency Shift Keying*, Modulación por Desplazamiento de Frecuencia Gausiana) con un índice de modulación K=0.3. En donde un "1" binario representa una desviación de frecuencia positiva, y un "0" binario representa una desviación de frecuencia negativa. La desviación máxima de frecuencia está entre 140 *KHz* y 175 *KHz*.

La elección de este esquema radica en su robustez y simplicidad de implementación del mismo.

A continuación se detalla el proceso de transmisión entre maestro y esclavo, en una *piconet*.[1]

El canal está dividido en ranuras de tiempo, cada ranura corresponde a una frecuencia de salto y tiene una longitud de 625 *us*.

Cada secuencia de salto en una *piconet* está determinada por la dirección del maestro de la *piconet.* Todos los dispositivos conectados a la *piconet* están sincronizados con el canal en salto y tiempo.

En una transmisión, cada paquete debe estar alineado con el inicio de una ranura y puede tener una duración de 1, 3 o 5 ranuras de tiempo. Durante la transmisión de un paquete la frecuencia es fija. Para evitar fallas en la transmisión, el maestro

[1] Piconet: Dos o más dispositivos *Bluetooth* que comparten un mismo canal forman una *piconet*.

inicia enviando en las ranuras de tiempo pares y los esclavos en las ranuras de tiempo impares. En la figura 1.4 se puede observar este esquema de transmisión.

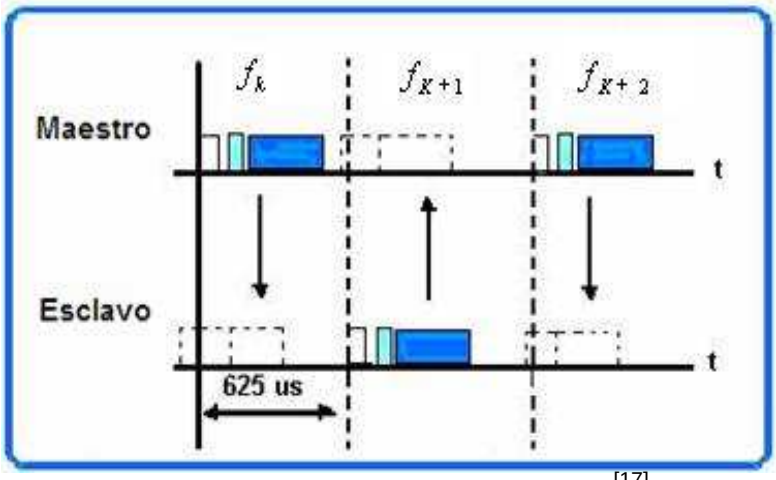


Figura 1.4 Transmisión en una *piconet* [17]

1.1.2.1.4 Potencia [16]

De acuerdo a las especificaciones *Bluetooth* los dispositivos de transmisión se dividen en tres grupos tal como se muestra en la tabla 1.2.

Clase de Transmisor	Potencia Máxima	Potencia Mínima	Alcance Máximo	Control de Potencia
Clase 1	100 mW	1 mW	100 m	Obligatorio
Clase 2	2.5 mW	0.25mW	10 m	Opcional
Clase 3	1mW	----------	10 cm.	----------

Tabla 1.2 Niveles de Emisión en *Bluetooth* [16]

“El equipo receptor debe poseer una sensibilidad de al menos -70 dBm y la tasa de error admisible debe ser menor o igual a 0.1 %” [16]

Los dispositivos de radio usados son de clase 2 que tienen una potencia de transmisión de 2.5 *mW*. La tecnología *Bluetooth* está diseñada para tener un consumo de potencia muy bajo. La tabla 1.3 representa la máxima potencia de salida permitida por regiones de acuerdo a la ubicación geográfica.

Máxima potencia de salida	Localización Geográfica	Documento de Complacencia
1000 mW	NORTE AMERICA	FCC 15.247
100 mW	EUROPA	ETS 300-328
10 mW/MHz	JAPÓN	MPT ordinance 79

Tabla 1.3 Niveles de Potencia de Transmisión para diferentes Regiones [2]

1.1.2.2 Capa Banda Base

En esta segunda capa se define el descubrimiento de dispositivos, establecimientos de conexión de una *piconet*, direccionamiento, formato de paquetes, temporización, control de potencia y comunicaciones asíncronas y síncronas entre pares.

1.1.2.2.1 Piconet [17]

Dos o más dispositivos *Bluetooth* que comparten un mismo canal forman una *piconet*. Para regular el tráfico en el canal cada *piconet* debe tener un maestro y puede tener hasta siete esclavos activos, además pueden haber muchos más esclavos en estado *parked*[2]. Los participantes podrían intercambiar los papeles si una unidad esclava quisiera asumir el papel de maestra. Sin embargo sólo puede haber un maestro en la *piconet* al mismo tiempo. En la figura 1.5 se puede observar una *piconet*.

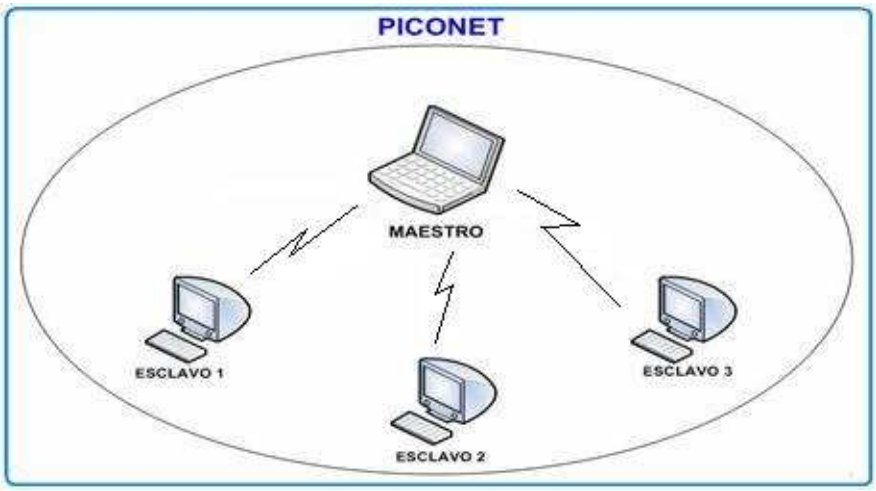


Figura 1.5 *Piconet*

[2] Parked: una unidad en una piconet se encuentra en este modo cuando está sincronizada pero no tiene una dirección MAC.

1.1.2.2.2 Scatternet [3] [17]

Dos o más p*iconets* que comparten una parte de su espacio físico de transmisión (*canal de transmisión*) forman una *scatternet*.

Las s*catternet* permiten aprovechar mejor el ancho de banda, y la velocidad efectiva individual de los usuarios es mucho mayor en la *scatternet* que si todos los usuarios estuviesen conectados a una misma *piconet.*

En la figura 1.6 se puede observar un ejemplo de una red *scatternet*

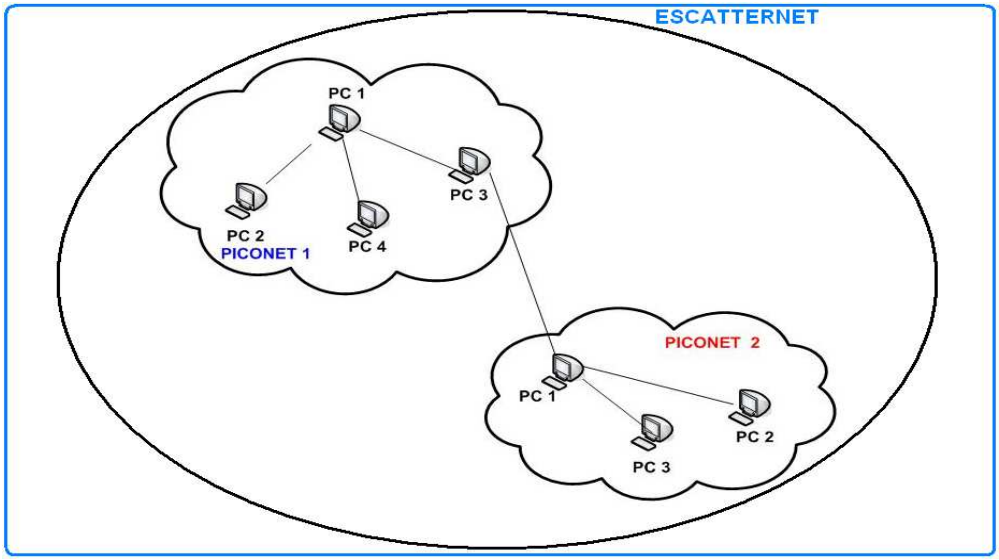


Figura 1.6 *Scatternet*

Las unidades que se encuentran en el mismo radio de cobertura pueden establecer potencialmente comunicaciones entre ellas. Sin embargo, sólo aquellas unidades que realmente quieran intercambiar información comparten un mismo canal creando la *piconet*. Este hecho permite que se creen varias *piconets* en áreas de cobertura superpuestas. A un grupo de *piconets* se le llama *scatternet.*

El rendimiento, en conjunto e individualmente de los usuarios de una *scatternet* es mayor que el que tiene cada usuario cuando participa en un mismo canal de 1 *MHz*.

Se debe tener en cuenta que cuantas más *piconets* se añaden a la *scatternet* su velocidad efectiva disminuye poco a poco, existiendo una reducción por término medio del 10%. Sin embargo el rendimiento que finalmente se obtiene de múltiples *piconets* supera al de una simple *piconet*.

Una estimación bastante simplificada de la velocidad efectiva normalizada es:

$$TH = \left(1 - \frac{1}{79}\right)^{N-1}$$ **Ecuación 1.1** Velocidad Efectiva Normalizada [3]

Donde: TH = Velocidad Efectiva
N = Número de *Piconets*

La información intercambiada por una *piconet* sólo es compartida por los miembros de la *piconet*, no por toda la *scatternet*. Una unidad puede participar en distintas *piconets* por medio de *TDD* (Duplexación por división de tiempo) pero esta unidad solo puede ser maestra en una sola *piconet*.

1.1.2.2.3 Enlace Físico [3] [17]

En la especificación *Bluetooth* se han definido dos tipos de enlace que permitan soportar incluso aplicaciones multimedia:

- Enlace sincrónico orientado a conexión (*SCO*)
- Enlace asíncrono no orientado a conexión (*ACL*)

a. Enlace Sincrónico Orientado a Conexión (*SCO*)

Los enlaces *SCO* soportan conexiones simétricas, punto a punto y conmutación de circuitos, estos enlaces se usan en conexiones de voz, estos enlaces se encuentran definidos en el canal de transmisión, reservándose dos ranuras consecutivas (envío y retorno) en intervalos fijos. La reserva de las ranuras la realiza el maestro cuando se establece la conexión entre el maestro y el esclavo. La conexión *SCO* debe establecerse explícitamente después de que se ha creado la *piconet*.

El maestro envía un mensaje de establecimiento a conexión al esclavo, usando el protocolo de gestión del enlace (*Link Management*). En una conexión *SCO* no se permiten paquetes multi-ranura es decir paquetes que ocupen 2 o más ranuras consecutivas. En una *piconet* el maestro puede contener hasta tres enlaces *SCO* con un solo esclavo o con esclavos diferentes y el esclavo puede mantener hasta dos enlaces *SCO* si los maestros con los que se comunica son diferentes.

b. Enlace Asíncrono no Orientado a Conexión (*ACL*)

Los enlaces *ACL* soportan conexiones simétricas o asimétricas, punto a multipunto y con conmutación de paquetes, típicamente usadas en la transmisión de datos. Por defecto cuando, se establece la *piconet* la unidad maestra establece una conexión *ACL* con las unidades esclavas.

Un enlace *ACL* no reserva ancho de banda, usa ranuras por demanda de 1, 3 y 5 ranuras consecutivas también usa control de errores para garantizar la entrega de los paquetes. La máxima velocidad de transmisión se obtiene enviando paquetes sin protección, de 5 ranuras, con capacidad de asignación asimétrica 721 *Kbps* en un sentido y 57,6 *Kbps* en el otro.

1.1.2.2.4 Formato del Paquete Bluetooth [13]

La figura 1.7, representa el formato de paquete general el cual consta de tres campos: código de acceso, cabecera y carga útil.

Figura 1.7 Paquete *Bluetooth* [17]

a. Código de acceso [14]

Usado para sincronización, identificación. Todos los paquetes que son enviados sobre el canal de transmisión en la *piconet* están antepuestos del mismo código de acceso. Existen tres tipos diferentes de código de acceso:

- **Código de Acceso al Canal (*CAC*):** identifica una *piconet* y es incluido en todos los paquetes intercambiados dentro de la *piconet*.

- **Código de Acceso de Dispositivo (*DAC*):** para procedimientos de señalización especiales, *paging* (utilizado para la localización y señalización de un dispositivo), entre otros.

- **Código de Acceso de Búsqueda (*IAC*): *(IAC) de tipo general*** se usa para descubrir otros dispositivos *Bluetooth* dentro de una *piconet*, o ***(IAC) dedicado*** cuando se quiere descubrir dispositivos *Bluetooth* específicos.

b. Cabecera de paquete [13]

La cabecera contiene información del control del enlace y está formada por seis campos:

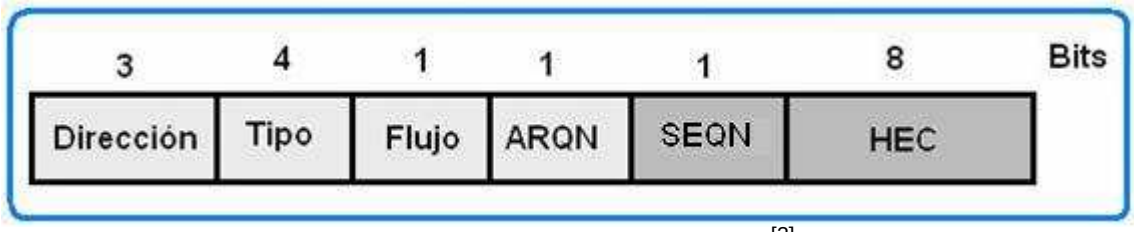


Figura 1.8 Cabecera del Paquete [3]

- **Dirección:** una dirección de dispositivo para distinguirlo de los demás dispositivos activos en la *piconet.* Se tienen 3 bits porque se pueden tener hasta 7 dispositivos activos en la *piconet*. El valor cero se tiene reservado por el maestro para enviar información a todos los esclavos en la *piconet*.

Dirección *M_ADDR*: permite identificar a los esclavos que están activos dentro de una *piconet*. Si la información está dirigida a todos los esclavos entonces los 3 bits son ceros.

- **Tipo:** define el tipo de paquete enviado. Éste dependerá del enlace asociado al paquete (*SCO* o *ACL*). El campo *Tipo* también indica el número de ranuras que ocupa el paquete actual que puede ser de 1, 3 o 5 ranuras consecutivas en caso de un enlace *ACL*.

- **Flujo:** usado para el control de flujo de los paquetes sobre el enlace *ACL*, para notificar al emisor cuando el *buffer* del receptor está lleno.

  Si el *buffer* del receptor para el enlace *ACL* está lleno, se devuelve una señal de parada para detener la transmisión de datos (Flujo = 0), esta señal de parada sólo se aplica a paquetes *ACL*, los paquetes con información de control de enlace y *SCO* se siguen recibiendo normalmente. Cuando se vacía el *buffer* del receptor, se devuelve una señal de continuar (Flujo = 1).

- ***ARQN*:** usado para informar si una transferencia es exitosa o no, *ARQN* puede ser un *ACK* (*ARQN*=1) si la recepción fue exitosa, o un *NAK* (**ARQN**=0) si la recepción fue errónea, en este caso el paquete se retransmite.

- ***SEQN*:** permite distinguir si el paquete enviado es nuevo o es un paquete retransmitido.

- ***HEC*:** permite verificar la integridad de la cabecera del paquete.

c. Carga útil [4]

La carga útil de un paquete puede ser dividida en dos campos:

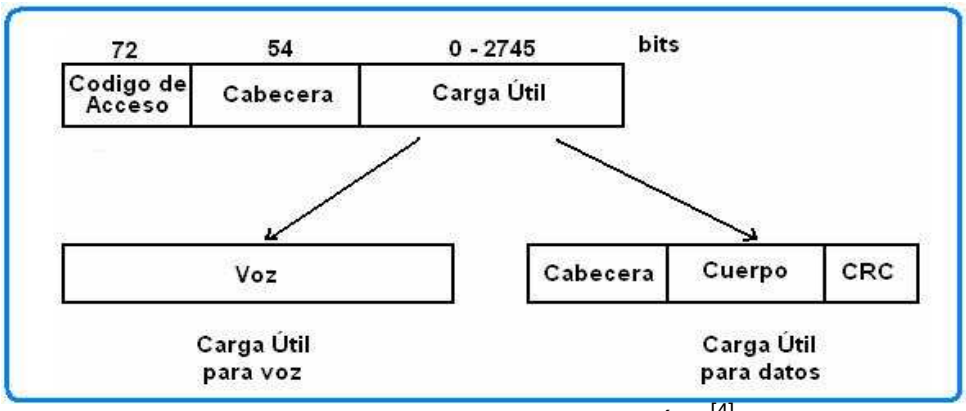


Figura 1.9 Tipos de Datos en la Carga Útil [4]

- **Campo de Voz:** consta de una carga útil, exclusiva para la transmisión de voz. Este campo no posee una cabecera, no realiza chequeo de errores.

- **Campo de Datos:** consta de tres partes, cabecera de carga útil, datos de carga útil, y código *CRC.*

c.1) División de la Carga Útil [4]

Como se puede observar en la figura 1.10, la carga útil destinada para el envío de datos se divide en tres campos que son cabecera, cuerpo y *CRC*.

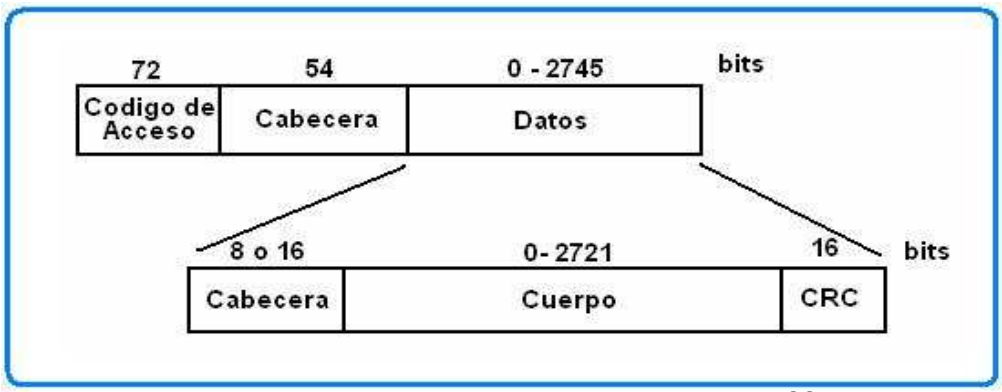


Figura 1.10 División de la Carga Útil para Datos [4]

- **Cabecera de la Carga Útil:** consta de 8 o 16 bits dependiendo si son paquetes de una sola ranura o paquetes multi-ranura.

Figura 1.11 Cabecera de la Carga Útil para Datos [4]

La cabecera de la carga útil consta de los siguientes campos:

- **L_CH:** consta de 2 bits y especifica el canal lógico.

- **Flujo:** campo de un 1 bit y permite realizar el control de flujo de los canales lógicos a nivel de *L2CAP*.

- **Longitud:** la longitud en bytes del cuerpo. Consta de 5 o 9 bits dependiendo si es un paquete que ocupa una sola ranura o es un paquete multi-ranura.

- **Cuerpo de la Carga Útil:** es un campo que va depender del número de bits que se van a transmitir y su longitud puede variar entre 0 y 2721 bits.

- **CRC:** es un campo de 16 bits para chequeo de errores.

1.1.2.2.5 Tipos de Paquetes [3]

Los tipos de paquetes se dividen en paquetes de control y de información.

Los paquetes de control son de cuatro tipos:

- **ID:** paquete de identificación. Consiste solo en el código de acceso.

- **NULL:** consiste en el código de acceso y la cabecera. Sirve para llevar información solo de control.

- **POLL:** similar al anterior; usado por el maestro para invitar a los esclavos.

- **FHS:** paquete de sincronización. Sirve para intercambiar información de identidad e información del reloj.

Los 12 códigos de paquetes restantes sirven para definir el tipo de servicio que se entrega (sincrónico o asincrónico) y el tamaño en ranuras del paquete. Los datos pueden o no ser protegidos con *FEC* (1/3 o 2/3).

Considerando la transmisión sin *FEC* se puede lograr una máxima tasa asimétrica de 723.2 Kbps con un enlace de retorno de 57.6 Kbps.

En la tabla 1.4 se presentan algunos paquetes para transmisión sincrónica y asincrónica con sus respectivas velocidades.

PAQUETES			
TIPO	**SIMÉTRICO**	**ASIMÉTRICO**	
		ENVÍO	**RETORNO**
DM1	108.8	108.8	108.8
DH1	172.8	172.8	172.8
DM3	256.0	384.0	54.4
DH3	384.0	576.0	86.4
DM5	286.7	477.8	36.3
DH5	432.8	721.0	57.6

Tabla 1.4 Paquetes para Transmisión Simétrico y Asimétrico [3]

Máximas tasas de transmisión promedio en Kbps.

- **DM**x: Paquetes de largo x slots, con *FEC*
- **DH**x: Paquetes de largo x slots, sin protección

1.1.2.2.6 Canales Lógicos [4]

Los canales lógicos definidos en *Bluetooth* son usados para actividades de control y para transporte de datos de usuario. Estos canales lógicos existen sobre los canales físicos *SCO* (Enlace Sincrónico Orientado a Conexión) o *ACL* (Enlace Asincrónico no Orientado a Conexión).

- **Canal de Control LC (*Link Control*):** implementado a través de la cabecera del paquete excepto en los paquetes de identificación (*ID*) que carecen de encabezado. Se encarga de transportar información de bajo nivel tal como:

  - ✓ Caracterización de la carga útil es decir el tipo de paquete que se envía.
  - ✓ Peticiones de repetición automática
  - ✓ Control de flujo.

- **Canal de Control LM (*Link Manager*):** transporta información de control para la administración del enlace entre el maestro y uno o más esclavos. Este canal lógico es transportado en la carga útil y puede estar presente en enlaces *SCO* o *ACL* soportando tráfico *LMP* (Protocolo de Administración del Enlace)

- **Canal de Usuario UA (*User Asynchronous*)**: transporta datos asíncronos de usuario. Generalmente es transportado en un enlace *ACL*.

- **Canal de usuario UI (*User Isochronous*):** se encarga de transportar datos isócronos de usuario. Estos datos se caracterizan porque la información de temporización está incluida en la cadena de datos. Los datos isócronos necesitan temporización de forma precisa como es el caso de enviar audio comprimido sobre un enlace *ACL*.

- **Canal de Usuario US (*User Sinchronous*):** transporta datos de usuario síncronos y están presentes sobre enlaces físicos *SCO*.

1.1.2.2.7 Establecimiento de la Conexión [4]

Para el establecimiento de una conexión en *Bluetooth* los dispositivos pueden estar en ciertos estados como son:

- **Inquiry:** el estado de *Inquiry* es un estado de búsqueda que es utilizado para descubrir otros dispositivos.

- **Scan:** cuando un dispositivo *Bluetooth* está en modo *STANDBY* (dormida), periódicamente escucha el canal, esperando a ser descubiertos por otros dispositivos.

- **Paging:** este estado es también llamado como un estado de paginación o localización. Es utilizado, generalmente, luego del estado *Inquiry* para establecer las conexiones.

a. Conexión entre dos dispositivos:

Figura 1.12 Conexión de Dispositivos

En una conexión de dispositivos *Bluetooth*, estos dispositivos pueden estar en diferentes estados como son: estado *Inquiry*, *Scan*, *Paging*. Dependiendo del estado de cada dispositivo la conexión se la puede realizar utilizando los siguientes procedimientos.

a.1. Inquiry [4]

Durante el procedimiento *Inquiry* el nodo fuente invita al nodo destino con un mensaje *inquiry*, luego de escuchar el mensaje de invitación el nodo destino responde la invitación aceptando la comunicación.

La información que envía en el mensaje el nodo fuente es un paquete de *ID* (paquete de identificación) con un código *IAC* (Código de Acceso de Búsqueda). El paquete *ID* es un paquete que no tiene cabecera ni carga útil y el código *IAC* es un código común para todos los dispositivos *Bluetooth*.

Si el destino recibe el mensaje *Inquiry* responde con un paquete *FHS* (Paquete de Sincronización) que contiene la dirección del dispositivo e información del reloj. Para evitar colisiones, los destinos difieren sus respuestas utilizando un temporizador de *backoff*.

La fuente captura información básica enviada por el destino y esta información es utilizada para hacer el *paging* con el dispositivo destino seleccionado.

A continuación en la figura 1.13 se muestra el proceso de *Inquiry*:

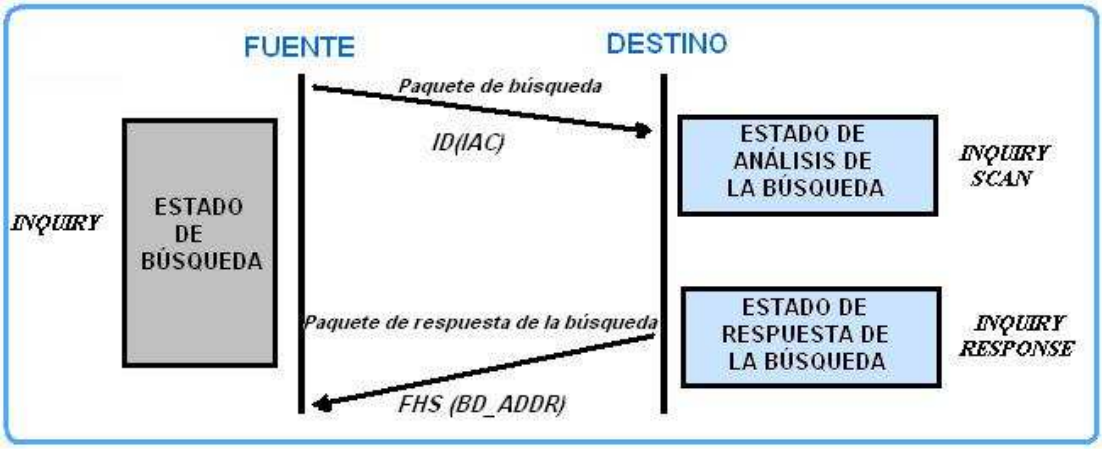


Figura 1.13 Proceso de *Inquiry* [4]

Pasos para el procedimiento del *Inquiry*:

- La fuente envía los paquetes de *inquiry*, que llevan la información de identificación y un código de acceso común.

- El destino que recibe los paquetes de *inquiry*, debe estar en el estado *Scan Inquiry*, en el que está atento a recibir los paquetes de *Inquiry*.

- El destino entra al estado de respuesta de la búsqueda o también llamado *Inquiry Response*, en el cual manda una contestación al mensaje de *Inquiry* enviado por la fuente.

- Una vez que el nodo destino responde a un *Inquiry*, se mueve al estado *Page Scan*

a.2. Paging [4]

La fuente envía un mensaje de *paging* que es único al destino. El destino contesta inmediatamente sin necesidad de esperar un periodo de *backoff*.

Para establecer una conexión es necesario conocer la dirección del dispositivo *Bluetooth* (*BD_ADDR*). Esta dirección es también utilizada para el cálculo de la secuencia de salto del *paging* (*page frequency- hopping sequence*), con el cual se contacta al dispositivo durante el *paging.*

El paquete que envía la fuente es un paquete *ID* (Paquete de Identificación) al cual se le ha añadido un Código de Acceso del Dispositivo (*DAC*) este código de acceso del dispositivo contiene una parte de la dirección del dispositivo (*BD-ADDR*).

Luego de recibir la respuesta al *paging*, la fuente se convierte en el maestro y el destino en esclavo de la nueva *piconet*.

A continuación se representa en la figura 1.14 el procedimiento del *paging*:

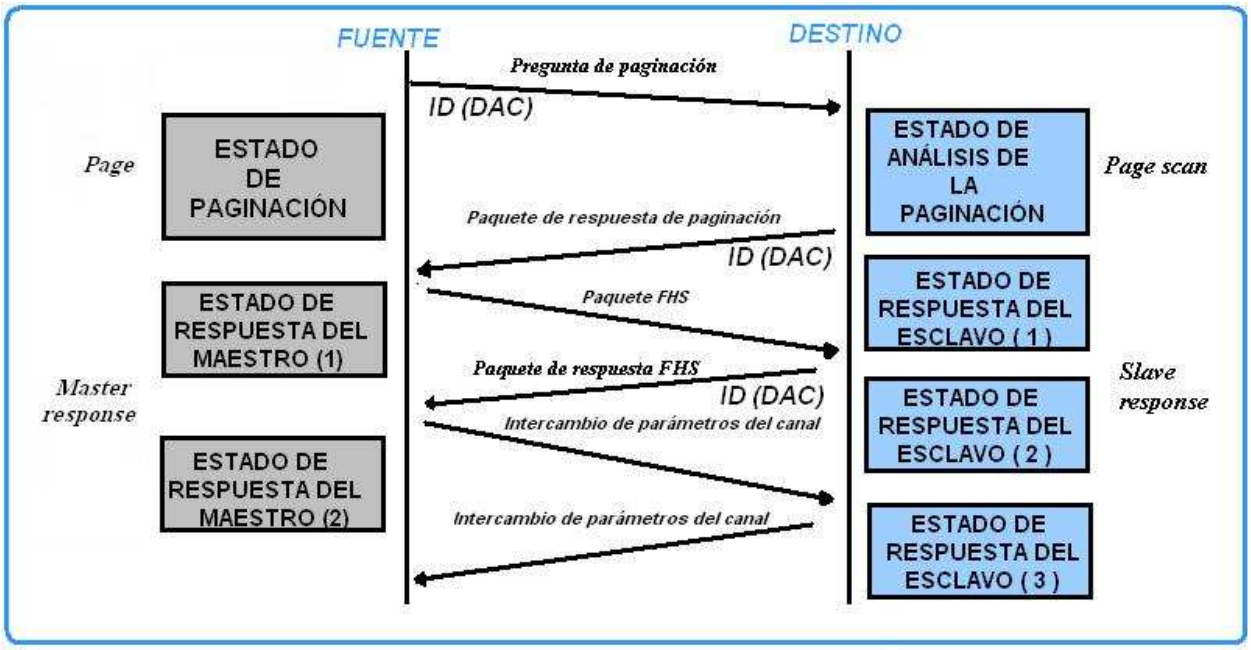


Figura 1.14 Proceso de *Paging* [4]

Procedimiento de *paging*:

- La fuente que se encuentra en el estado de *Paging* envía un *page* al destino, que contiene un paquete de Identificación (*ID*) pero esta vez con un Código de Acceso del Dispositivo.

- El destino que se encuentra en el estado de *Page Scan*[3] recibe el *page*.

- El destino manda una contestación a la fuente que se encuentra en el estado de *Master Response*. Esta contestación contienen un paquete idéntico al recibido (*ID*)

- La fuente que está en el estado de *Master Response* manda un paquete de sincronización (*FHS*) al destino. Este paquete de sincronización contiene la dirección del dispositivo *Bluetooth* fuente y el valor de su reloj de tiempo real *Bluetooth*.

- El destino que se encuentra en el estado *Slave Response* manda una segunda contestación a la fuente, confirmando la recepción del paquete de

[3] **Page Scan**: Subestado en el cual el esclavo escucha el código de acceso durante el tiempo que dura una ventana de scan.

Sincronización (*FHS*). Esta respuesta es el mismo paquete de identificación con el mismo Código de Acceso del dispositivo.

- La fuente y el destino que se encuentran en los estados *Master Response* y *Slave Response* respectivamente, intercambian los parámetros de sincronización del Canal. El destino empieza a utilizar la secuencia de salto definida por el maestro.

El maestro puede continuar realizando invitaciones a otros dispositivos.

a.3. Admisión de un nuevo esclavo [4]

Los pasos para la admisión de un nuevo esclavo son relativamente complejos. El maestro podría tomar una de las siguientes opciones:

- Empezar a descubrir nuevos nodos e invitarlos a unirse a la *piconet*.
- Esperar a ser descubierto por otros nodos, permaneciendo en un estado *Scan*.

En ambos casos, la comunicación en la *piconet* debe suspenderse durante el proceso de *Inquiry* y *Paging*. El retardo involucrado en la admisión de un nuevo nodo puede ser grande, especialmente si el maestro no pasa al estado de *Inquiry* o *Scan* frecuentemente; esto provoca una degradación en la capacidad de la *piconet*.

La operación de una *piconet* puede entenderse en base a “dos estados de operación” principales que se definen durante el establecimiento de una conexión.

- ***Standby:*** estado en el que se encuentra una unidad *Bluetooth* por defecto, es un estado de bajo consumo de potencia en el cual solo el reloj local está activo. Los dispositivos que se encuentran en este estado periódicamente entran al estado *Inquiry Scan.*

- ***Connection:*** en este estado una estación *Bluetooth* puede estar conectada a una *piconet* ya sea como esclavo o como maestro.

Todos los procedimientos se pueden combinar en un solo diagrama de estados, el mismo que se muestra en la figura 1.15.

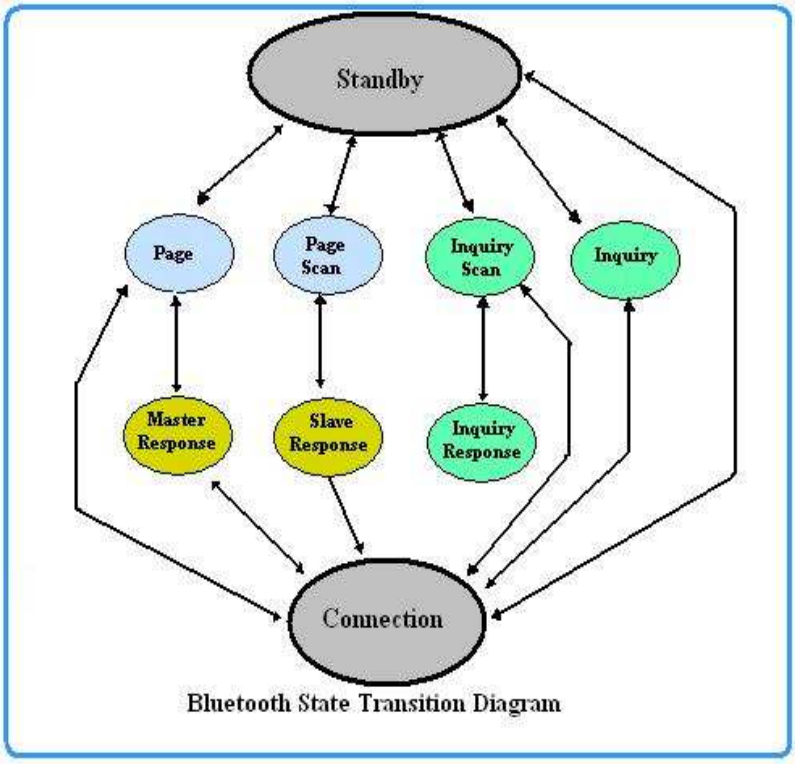


Figura 1.15 Diagrama de Estados de Transición *Bluetooth* [4]

1.1.2.2.8 Modos de Ahorro de Potencia [3] [14]

Bluetooth es un estándar que permite un ahorro considerable de energía no solo por la baja potencia que utiliza cada dispositivo sino también por la manera como éstos establecen una conexión.

Una unidad en modo *SCAN* se activa durante un pequeño periodo de tiempo de 10 [*ms*] para reducir el consumo de potencia mientras está en modo "*STANDBY*".

El modo *PAGE* necesita un consumo mayor de energía debido a que lleva el peso de la incerteza de frecuencia y tiempo. Se prefiere esta configuración debido a que el modo *PAGE* es más infrecuente que el modo *STANDBY*. También para

ahorrar energía, las unidades *Bluetooth* que detectan un paquete que no está dirigido hacia ellas siguen en estado *STANDBY*.

Además el esquema *FH* (Salto de Frecuencia) es robusto en términos de sincronización por lo que no es necesario que se estén constantemente enviando señales de temporización, lo que también reduce el consumo de energía. *Bluetooth* además define una serie de técnicas para ahorrar energía:

- **MODO *HOLD*:** el maestro puede ordenar al esclavo quedarse en modo *HOLD*. Durante este periodo no hay comunicación posible entre esclavo y maestro. Cuando el periodo expira el esclavo vuelve al canal y permanece sincronizado.

- **MODO *PARK*:** el esclavo también puede ser puesto en modo *PARK*. En este caso el esclavo entra a un ciclo de trabajo en donde los intervalos de escucha del maestro son más largos.

- **MODO *SNIFF*:** el esclavo no escucha todas las ranuras de tiempo, sino que solo escucha algunas. Para entrar al modo *SNIFF*, el esclavo y maestro deben acordar en qué ranuras el esclavo pondrá atención al canal.

1.1.2.3 Protocolo de Administración del Enlace *LMP* [14]

El protocolo *LMP* se usa para establecer y controlar un enlace. Las señales son interpretadas y filtradas por el protocolo *LMP* en el lado del receptor, y no se propagan a las capas superiores.

Los mensajes *LMP* son usados para el inicio, seguridad y control del enlace. Estos mensajes de administración del enlace tienen una prioridad mayor que los datos del usuario.

1.1.2.3.1 Emparejamiento con el Protocolo LMP

Si en el momento de iniciar el emparejamiento dos dispositivos *Bluetooth* no tienen una clave de enlace común, entonces se crea una clave de inicialización denominada *Kinit* basada en un número *PIN*, un número aleatorio y una dirección del dispositivo *Bluetooth* (*BD_ADDR*).

Para el procedimiento de emparejamiento existen cinco posibilidades:

- Contestador acepta el procedimiento “emparejamiento”
- Contestador tiene un número *PIN*.
- Contestador rechaza el procedimiento “emparejamiento”
- Creación de una clave de enlace.
- Intentos repetidos

1.1.2.3.2 Características Soportadas en el enlace

La radiocomunicación *Bluetooth* y el controlador de enlace pueden soportar solo una parte de los tipos de paquetes y características descritas en las especificaciones de Banda Base y Radio de *Bluetooth*. Las características soportadas pueden ser requeridas en cualquier momento, siguiendo un procedimiento exitoso de búsqueda en banda base. Cuando se hace un requerimiento, éste debe ser compatible con las características soportadas del otro dispositivo.

1.1.2.3.3 Requerimiento de Nombre del enlace LMP

El protocolo *LMP* soporta requerimiento de nombre a otro dispositivo *Bluetooth*, y es un nombre de usuario-amigo asociado al dispositivo.

1.1.2.3.4 Terminación del emparejamiento de LMP

La conexión entre dos dispositivos *Bluetooth* puede ser terminada en cualquier momento por el dispositivo maestro o por el dispositivo esclavo.

1.1.2.3.5 Establecimiento de Conexión de LMP

Después del procedimiento de búsqueda, el dispositivo maestro debe invitar al dispositivo esclavo.

1.1.2.3.6 Modos de Prueba de LMP

Este protocolo tiene *PDUs* para soportar diferentes modos de prueba *Bluetooth*, los cuales se usan para certificaciones y pruebas de cumplimiento de banda base y radio *Bluetooth*.

1.1.2.4 Interfaz del Controlador de Host (*HCI*) [13] [14]

Permite acceder al estado y los registros de control del aparato, además de proporcionar un método uniforme de acceder a todas las funciones de la banda base *Bluetooth*.

La sección *HCI* tiene dos funciones en la especificación *Bluetooth*:

- Definir las bases de una interfaz física para un módulo externo *Bluetooth*.
- Definir las funciones de control necesarias para todas las implementaciones *Bluetooth*.

El computador recibe notificaciones asincrónicas de eventos *HCI* independientemente de que capa de transporte se usa. Los eventos *HCI* son usados para notificar al computador cuando algo ocurre. Al descubrir éste, que ha ocurrido un evento, analizará un paquete recibido para determinar qué tipo de evento *HCI* se tiene.

1.1.2.5 Protocolo de Control y Adaptación de Enlace Lógico (*L2CAP*) [13]

L2CAP se encuentra sobre el protocolo de gestión de enlace (*LMP*) y reside en la capa de enlace de datos. *L2CAP* permite a protocolos de niveles superiores y a aplicaciones, la transmisión y recepción de paquetes de datos *L2CAP* de hasta 64

Kbytes, con capacidad de multiplexación de protocolo, segmentación y reensamble, y abstracción de grupos.

Para cumplir sus funciones, *L2CAP* espera que la banda base suministre paquetes de datos en los dos sentidos al mismo tiempo, que realice el chequeo de integridad de los datos y que reenvíe los datos hasta que hayan sido reconocidos satisfactoriamente. Las capas superiores que se comunican con *L2CAP* son por ejemplo el protocolo de descubrimiento de servicio (*SDP*), el *RFCOMM* y el control de telefonía (*TCS*).

1.1.2.5.1 Canales Lógicos de L2CAP

L2CAP está basado en el concepto de canales. Se asocia un identificador de canal, *CID*, a cada uno de los puntos finales de un canal *L2CAP*. Los *CIDs* están divididos en dos grupos, uno con identificadores reservados para funciones *L2CAP* y otro con identificadores libres para implementaciones particulares. Los canales de datos orientados a la conexión representan una conexión entre dos dispositivos, donde un *CID* identifica cada punto final del canal.

Los canales no orientados a la conexión limitan el flujo de datos a una sola dirección. La señalización de canal es un ejemplo de un canal reservado. Este canal es usado para crear y establecer canales de datos orientados a la conexión y para negociar cambios en las características de esos canales.

1.1.2.5.2 Multiplexación de Protocolo

L2CAP soporta Multiplexación de Protocolo, ya que el protocolo de banda base no soporta ningún campo "*TYPE*" que identifica al protocolo de capa superior como protocolos de descubriendo de servicio *SDP, RFCOMM* y control de telefonía.

1.1.2.5.3 Segmentación y Reensamblado

Los paquetes de datos definidos por el protocolo banda base están limitados en tamaño. Los paquetes grandes *L2CAP* deben ser segmentados en varios

paquetes más pequeños antes de transmitirse y luego deben ser enviados a la gestión de enlace.

En el receptor los paquetes pequeños recibidos de la banda base son reensamblados en paquetes *L2CAP* más grandes. Varios paquetes banda base recibidos pueden ser reensamblados en un solo paquete *L2CAP* seguido de un simple chequeo de integridad.

La segmentación y reensamblado de paquetes, es necesaria para soportar protocolos con paquetes grandes, que los soportados por la banda base, la figura 1.16 muestra la segmentación *L2CAP*.

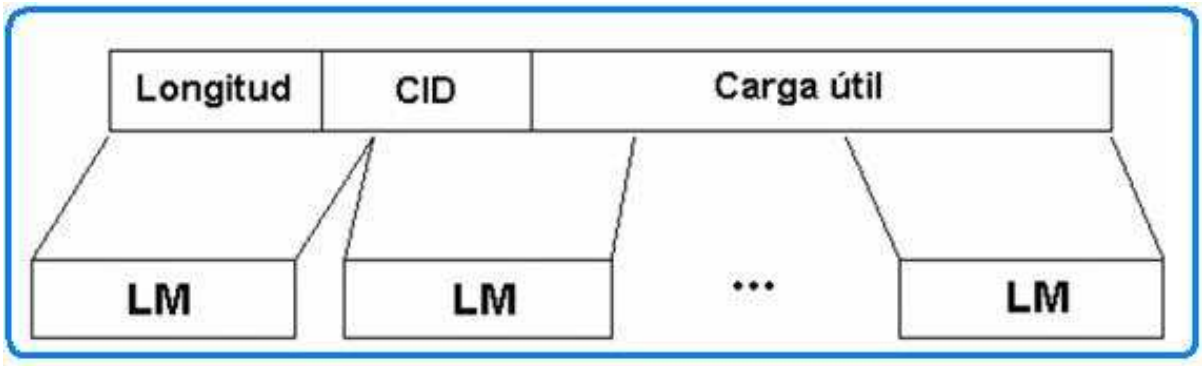


Figura 1.16 Segmentación L2CAP [13]

1.1.2.5.4 Eventos de L2CAP

Todos los mensajes que entran en la capa *L2CAP*, son llamados eventos. Los eventos se encuentran divididos en cinco categorías: indicaciones y confirmaciones de capas inferiores, peticiones de señal y respuestas de capas *L2CAP*, datos de capas *L2CAP*, peticiones y respuestas de capas superiores, y eventos causados por expiraciones de tiempo.

1.1.2.5.5 Formato del paquete de datos

L2CAP está basado en paquetes pero sigue un modelo de comunicación basado en canales. Como se puede observar en la figura 1.17, los paquetes de canal

orientado a la conexión están divididos en tres campos: longitud de la información, identificador de canal, e información.

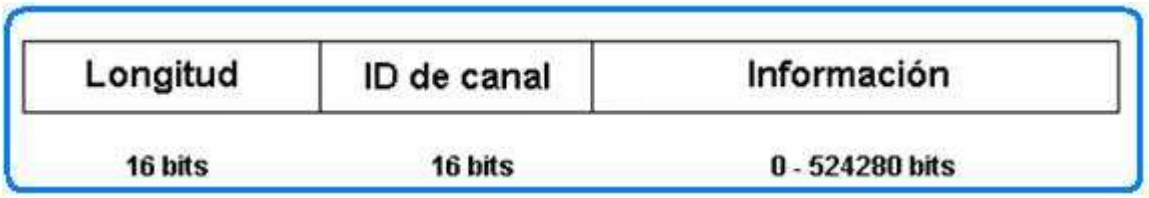


Figura 1.17 Paquete *L2CAP* [13]

Los paquetes de canal de datos no orientados a conexión son iguales a los paquetes orientados a conexión pero adicionalmente incluyen un campo con información multiplexada de protocolo.

1.1.2.6 Protocolo de Descubrimiento de Servicio SDP [4] [13]

El protocolo de descubrimiento de servicio, brinda a las aplicaciones recursos para descubrir qué servicios están disponibles y determinar las características de dichos servicios.

1.1.2.6.1 Descripción General

El *SDP* ofrece a los clientes la facilidad de averiguar sobre servicios que sean requeridos, basándose en la clase de servicio o propiedades específicas de estos servicios.

Los dispositivos *Bluetooth* que usan el *SDP* pueden ser vistos como un servidor y un cliente. El servidor posee los servicios y el cliente es quien desea acceder a ellos. En el *SDP* esto es posible ya que el cliente envía una petición al servidor y el servidor responde con un mensaje. El *SDP* solamente soporta el descubrimiento del servicio, no la llamada del servicio.

1.1.2.6.2 Registros de Servicio

Los registros de servicio contienen propiedades que describen un servicio determinado. Cada propiedad de un registro de servicio consta de dos partes, un identificador de propiedad y un valor de propiedad. El identificador de propiedad

es un número único de 16 bits que distingue cada propiedad de servicio de otro dentro de un registro. El valor de propiedad es un campo de longitud variable que contiene la información.

1.1.2.7 Capa *RFCOMM* [5] [13]

El protocolo *RFCOMM* brinda emulación de puertos seriales sobre el protocolo *L2CAP*, éste soporta hasta 60 puertos emulados simultáneamente. Dos unidades *Bluetooth* que usen *RFCOMM* en su comunicación pueden abrir varios puertos seriales emulados, los cuales son multiplexados entre sí, la figura 1.18 muestra el esquema de emulación para varios puertos seriales.

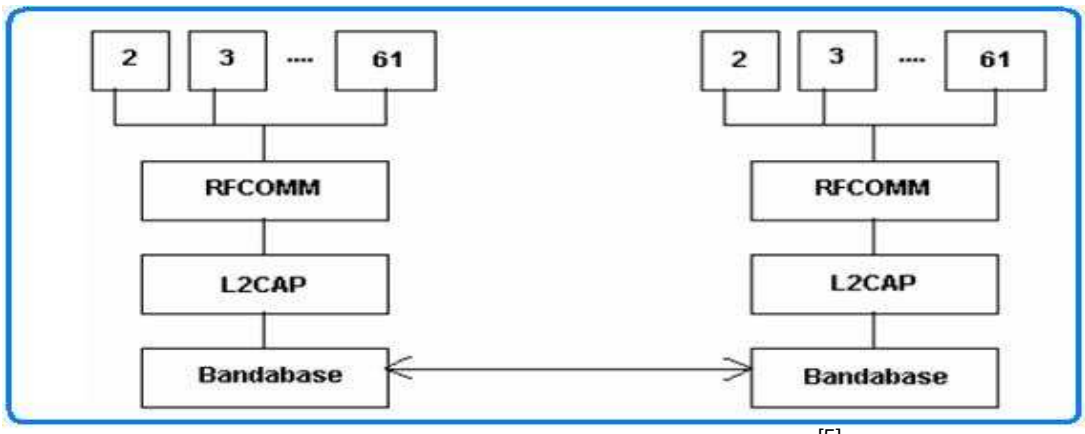


Figura 1.18 Puertos emulados por *RFCOMM* [5]

Muchas aplicaciones hacen uso de puertos seriales. El *RFCOMM* está orientado a hacer más flexibles estos dispositivos, soportando una fácil adaptación de comunicaciones seriales utilizando *Bluetooth*. Un ejemplo de una aplicación de comunicación serial es el protocolo punto-a-punto (*PPP*). El *RFCOMM* tiene construido un esquema para emular el cable que se utiliza en una transmisión serial de datos y usa a *L2CAP* para cumplir con el control de flujo requerido por alguna estación.

1.1.3 PERFILES BLUETOOTH [13]

Desde que se inició la especificación del estándar *Bluetooth*, una de las principales preocupaciones de *SIG* fue garantizar la interoperabilidad total entre

dispositivos de distintos fabricantes, siempre que éstos compartan iguales perfiles.

Los perfiles especifican cómo utilizar el conjunto de protocolos *Bluetooth* para implementar una solución que trabaje sin problemas con otras marcas. En cada uno se establecen opciones y parámetros, además de detallar cómo usar los distintos procedimientos de los diversos estándares que se encuentren implicados. En la figura 1.19 se observa la estructura de los perfiles *Bluetooth*.

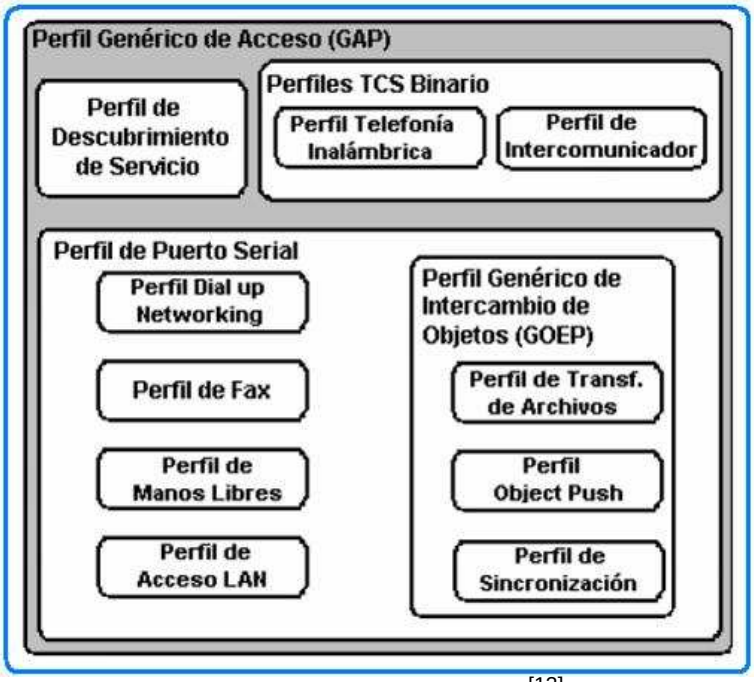


Figura 1.19 Perfiles *Bluetooth* [13]

A continuación se hace una breve descripción de algunos de los perfiles *Bluetooth*:

- **Perfil Genérico de Acceso**

  Define los procedimientos generales para el descubrimiento y establecimiento de conexión entre dispositivos *Bluetooth. A*segurando que cualquier par de dispositivos *Bluetooth* puedan intercambiar información para descubrir qué tipo de aplicaciones soportan las unidades.

- **Perfil Genérico de Intercambio de Objetos *GOEP***

  Este perfil puntualiza los protocolos y procedimientos usados por aplicaciones para ofrecer características de intercambio de objetos. Los dispositivos más comunes que usan este modelo son agendas electrónicas, *PDAs*, teléfonos celulares y teléfonos móviles. El *GOEP* es dependiente del perfil de puerto serial.

- **Transferencia de Archivos**

  Soporta la transferencia de directorios, archivos, documentos, imágenes, y formatos de *streaming*[4] Además soporta la exploración de directorios en el dispositivo remoto.

- **Perfil de Puerto Serial**

  Define los requerimientos necesarios para establecer una conexión de cable serial emulada usando *RFCOMM* entre dos dispositivos *Bluetooth* similares.

- **Perfil de Acceso a una *LAN***

  Permite a los dispositivos de una *piconet* conectarse a una *LAN* como que estuviera conectado a un cable. Usando el protocolo punto-a-punto, *PPP* sobre *RFCOMM*.

- **Perfil de Aplicación de Descubrimiento de Servicio**

  Define los protocolos y procedimientos para descubrir los servicios proporcionados por otra unidad *Bluetooth*. El Perfil de Aplicación de Descubrimiento de Servicio es dependiente del Perfil Genérico de Acceso.

[4] *Streaming:* este término se refiere a ver u oír un archivo directamente sin necedad de descargarlo antes al computador.

- **Perfil de Telefonía Inalámbrica**

  Define cómo un teléfono móvil puede ser usado para acceder a un servicio de telefonía de red fija a través de una estación base. El perfil incluye llamadas a través de una estación base, haciendo llamadas de intercomunicación directa entre dos terminales y accediendo adicionalmente a redes externas.

- **Perfil de Manos Libres**

  Este perfil precisa los requerimientos, para que los dispositivos *Bluetooth,* soporten el uso de manos libres. En este caso el dispositivo puede ser usado como una unidad de audio inalámbrico de entrada/salida.

- **Perfil *Dial-up Networking***

  Este perfil define los protocolos y procedimientos que deben ser usados por dispositivos que implementen el uso del modelo llamado Puente *Internet.* Este perfil es aplicado cuando un teléfono celular es usado como un *modem* inalámbrico.

- **Perfil de *Fax***

  Precisa los protocolos y procedimientos que deben ser usados por dispositivos que implementen el uso de *fax*. El *software* de *fax* opera directamente sobre *RFCOMM*.

- **Perfil de Intercomunicador**

  Permite a dos teléfonos móviles establecer enlaces de conversación directa. El enlace directo es establecido usando señalización de telefonía sobre *Bluetooth.*

- **Perfil *Object Push***

  Este perfil define protocolos y procedimientos usados en el modelo *object push.* En el modelo *object push* hay procedimientos para introducir en el *inbox*, sacar e intercambiar objetos con otro dispositivo *Bluetooth.*

- **Perfil de Sincronización**

  Define protocolos y procedimientos usados en el modelo de sincronización. Éste usa el *GOEP*, el modelo soporta intercambios de información, por ejemplo para sincronizar calendarios de diferentes dispositivos.

1.1.4 SEGURIDAD EN BLUETOOTH [17] [34]

Para asegurar la protección de la información se ha definido un nivel básico de encripción, incluido en el diseño del *chip* de radio para proveer seguridad en equipos que carezcan de capacidad de procesamiento, las principales medidas de seguridad son:

- Una rutina de pregunta-respuesta, para autentificación
- Una corriente cifrada de datos, para encripción
- Generación de una clave de sesión (que puede ser cambiada durante la conexión)

Tres entidades son utilizadas en los algoritmos de seguridad: la dirección de la unidad *Bluetooth*, que es una entidad pública; una clave de usuario privada, como una entidad secreta; y un número aleatorio, que es diferente en cada nueva transacción.

La dirección *Bluetooth* se puede obtener a través de un procedimiento de consulta. La clave privada se deriva durante la inicialización y no es revelada posteriormente. El número aleatorio se genera en un proceso pseudo-aleatorio en cada unidad *Bluetooth*.

La especificación *Bluetooth* detalla tres modos de seguridad bajo los que el protocolo puede operar.

- **Modo1: Sin seguridad.** Todos los mecanismos de seguridad (autenticación y cifrado) están deshabilitados. Además el dispositivo se sitúa en un modo en el cual, permite que todos los dispositivos *Bluetooth* se conecten a él.

- **Modo2: Seguridad a Nivel de Servicio**. Este modo permite un acceso más flexible. Este modo es el más apropiado para ejecutar varias aplicaciones en paralelo con diferentes requerimientos de seguridad. Este modo de seguridad es impuesto después del establecimiento del canal.

- **Modo3: Seguridad a Nivel de Enlace.** El dispositivo instala procedimientos de seguridad antes del establecimiento del canal.

Los dispositivos *Bluetooth* solo pueden estar en un solo modo de seguridad en un momento determinado. Un dispositivo que opere en modo 3 no podrá autenticarse ante otros dispositivos de forma selectiva, sino que tratará de autenticarse ante todos los dispositivos que intenten comunicarse con él.

1.1.4.1 Seguridad con el Emparejamiento de dispositivos

Al inicio de una comunicación entre dispositivos *Bluetooth* la comunicación entre éstos está protegida, motivo por el que todos los dispositivos pueden comunicarse. Un nodo *Bluetooth* puede solicitar autenticación para realizar un determinado servicio.

Para la autenticación es necesario un código *PIN*. El código *PIN* tiene una longitud de hasta 16 caracteres *ASCII*. Para que los dispositivos se comuniquen se debe ingresar en ambos lados el mismo código *PIN*. Una vez que el usuario ha introducido el *PIN* adecuado ambos dispositivos generan una clave de enlace. Una vez generada, la clave se puede almacenar en el propio dispositivo o en un dispositivo de almacenamiento externo. La siguiente vez que se comuniquen

ambos nodos se utilizará la misma clave.

El procedimiento descrito hasta este punto se denomina emparejamiento. Es importante recordar que si la clave de enlace se pierde en alguno de los dispositivos involucrados se debe volver a ejecutar el procedimiento de emparejamiento.

No existe ninguna limitación en los códigos *PIN* a excepción de su longitud.

1.1.4.2 Autenticación *Bluetooth*

Los dispositivos *Bluetooth* se autentican siguiendo el esquema "*challenge-response",* desafío-respuesta. A continuación en la figura 1.20 se puede observar el procedimiento de autenticación.

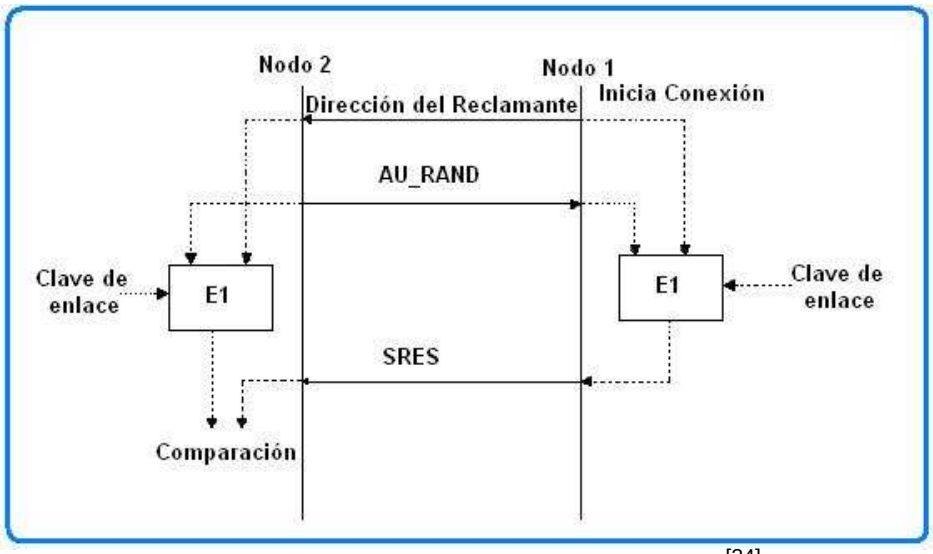


Figura 1.20 Proceso de autenticación *Bluetooth* [34]

Pasos para la autenticación *Bluetooth*:

- El nodo 1 inicia la conexión transfiere su dirección de 48 bits (*BD_ADDR*) al nodo 2, esta dirección es única e identifica al dispositivo.

- En respuesta el otro nodo 2 le transfiere un "desafío" aleatorio de 128 bits (*AU_RAND*) al nodo 1 que inició la conexión.

- El nodo 2 usa el algoritmo E1 para generar el “response” de autenticación, usando como parámetros la dirección del que inició la conexión, *BD_ADDR*, la clave de enlace, *Kab*, y el desafío. El nodo 1 realiza la misma operación.

- El nodo 1 que inició la conexión le devuelve el “*response*”, *SRES*, al otro nodo.

- El nodo 2 compara el *SRES* del demandante con el que él ha calculado.

- Si los valores de los 32 bits de los *SRES* son idénticos, el verificador establece la conexión.

1.1.5 APLICACIONES DE *BLUETOOTH*

En la actualidad se encuentra una gran cantidad de dispositivos *Bluetooth* que ofrecen aplicaciones muy variadas, permitiendo cambiar radicalmente la forma como los usuarios interactúan con los dispositivos que se encuentran relativamente cerca. Dentro de las aplicaciones se mencionan las siguientes:

- **Transferencia de archivos:** permite la transferencia de archivos sean estos: documentos en *Word*, imágenes, presentaciones en *Power Point*, etc. Además ofrece la posibilidad de ver el contenido de las carpetas existentes en otros dispositivos.

- **Conexión a Internet:** permite tener acceso inalámbrico a Internet mediante un teléfono móvil el cual actúa como si fuera una línea telefónica fija.

- **Escritorio Inalámbrico:** permite remplazar todos los dispositivos que utilizan cables permitiendo una comunicación vía radio. Utilizando desde un teclado inalámbrico hasta incluso utilizar un disco duro portátil que emplee esta tecnología para comunicarse.

- **Acceso inalámbrico a *LAN*:** a través de esta aplicación un grupo de dispositivos *Bluetooth* podrían conectarse a la red de Área Local a través de los llamados *LAP* (Puntos de Acceso *LAN*) y compartir los recursos

- **Sincronización Automática:** este servicio permite sincronizar automáticamente y de manera continua la Información de Administración Personal (*PIM*) con otros dispositivos *Bluetooth*; la información que se actualiza es la concerniente a calendario, lista de direcciones y teléfonos, mensajes y notas.

- **Dispositivo Manos Libres Inalámbrico***:* el dispositivo manos libres puede conectarse de manera inalámbrica al teléfono móvil, al ordenador portátil u otro móvil, con el fin de actuar como un dispositivo remoto con entrada y salida de audio.

1.2 IEEE 802.11

Los principales objetivos que se pretende conseguir con esta norma son la movilidad y la flexibilidad, los cuales vienen a ser fuertes argumentos a favor de la implementación de una Red Local Inalámbrica, *Wireless Fidelilty (Wi-Fi)*, en cualquier ambiente:

- Movilidad
- Simplicidad y rapidez en la instalación
- Flexibilidad en la instalación
- Costo de propiedad reducido
- Escalabilidad

1.2.1 EVOLUCIÓN [2] [27]

Las redes de área local inalámbricas funcionan desde hace más de quince años en entornos industriales y de investigación. Fueron implementadas por primera vez en el año 1979.

En marzo de 1985 la Comisión Federal de Comunicaciones, asignó a los sistemas *WLAN* las bandas de frecuencia 902-928 *MHz,* 2.4-2.48 *GHz* y 5.72-5.85 *GHz* también conocidas como *ISM.*

Desde 1985 hasta 1990 se siguió trabajando más en la fase de desarrollo hasta que en mayo de 1991 se publicaron varios trabajos referentes a *WLAN* operativas que superaban la velocidad de 1 *Mbps*, el mínimo establecido por el *IEEE 802* para que la red sea considerada realmente una *LAN*, con aplicación empresarial.

En junio del año 1997 el *IEEE* (Instituto de Ingenieros Electrónicos y Eléctricos) ratificó el estándar para *WLAN IEEE 802.11*, con velocidades de 1 y 2 *Mbps*, modulación de señal de espectro expandido por secuencia directa (*DSSS*), aunque también contempla la opción de espectro expandido por salto de frecuencia (*FHSS)* en la banda de 2,4 *GHz*, y se definió el funcionamiento y la interoperabilidad entre redes inalámbricas.

En el año 1999, se aprobó el estándar *802.11b*, una extensión del *IEEE 802.11* para *WLAN* empresariales, con una velocidad de 11 *Mbps* (otras velocidades normalizadas a nivel físico son: 5.5, 2 y 1 *Mbps*) y un alcance de 100 metros, que al igual que *Bluetooth* y *Home RF*, también emplea la banda de *ISM* de 2,4 GHz, pero en lugar de una simple modulación de radio digital y salto de frecuencia (*FH/* salto de frecuencia), utiliza técnicas de espectro expandido por secuencia directa (*DSSS*).

En julio de 1999 la *IEEE* publicó el suplemento del estándar en *802.11a*, que con modulación *OFDM* (*Orthogonal Frequency Division Multiplexing,* Modulación Ortogonal en División de Frecuencia) alcanza una velocidad de hasta 54 *Mbps* en la banda de 5 *GHz,* un alcance limitado a 30 metros dependiendo de la potencia de transmisión de los dispositivos utilizados.

“La banda de 5 *GHz* que utiliza se denomina *UNII* (Infraestructura de Información Nacional sin Licencia), que en los Estados Unidos está regulada por la *FCC*, el cual ha asignado un total de 300 *MHz*, cuatro veces más de lo que tiene la banda

ISM, para uso sin licencia, en tres bloques de 100 *MHz,* siendo en el primero la potencia máxima de 50 *mW,* en el segundo de 250 *mW*, y en el tercero puede llegar hasta 1 *W*, por lo que se reserva para aplicaciones en el exterior.” [2]

En el año 2003, se aprobó el estándar *IEEE 802.11g*, compatible con el *IEEE 802.11b*, capaz de alcanzar una velocidad de 54 *Mbps*, para competir con los otros estándares que prometen velocidades mucho más elevadas pero que son incompatibles con los equipos *802.11b* ya instalados, aunque pueden coexistir en el mismo entorno debido a que las bandas de frecuencias que emplean son distintas.

Estándar	Velocidad máxima	Interface de aire	Ancho de banda de canal	Frecuencia	Disponibilidad
802.11b	11 Mbps	DSSS	22 MHz	2.4 GHz	Ahora
802.11a	54 Mbps	OFDM	20 MHz	5.0 GHz	Ahora
802.11g	54 Mbps	OFDM/DSSS	22 MHz	2.4 GHz	Ahora

Tabla 1.5 Estandarización de *IEEE 802.11* [27]

1.2.2 ARQUITECTURA *Wi-Fi* [7]

La arquitectura básica *Wi-Fi* está definida por el estándar original *802.11.* Las especificaciones del estándar *802.11b* afectan únicamente a la capa física, añadiendo velocidades mayores y una conectividad más robusta.

El estándar *802.11* se centra en los dos niveles inferiores del modelo *OSI*, el físico y el de enlace de datos como se muestra en la figura 1.21

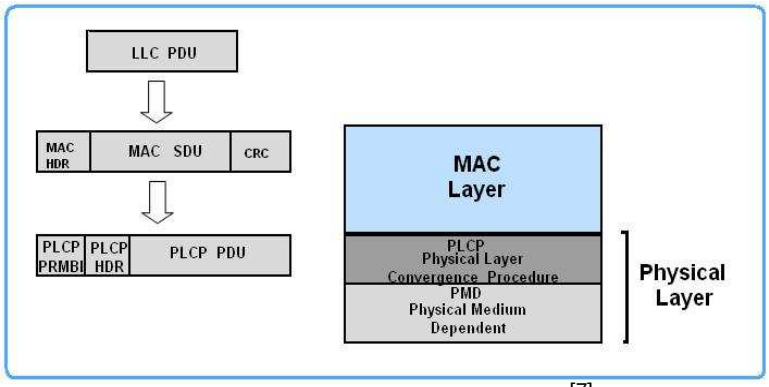


Figura 1.21 Arquitectura *Wi-Fi* [7]

Las tres capas físicas originalmente definidas en el *802.11* incluyen dos espectros de radio y una especificación de infrarrojos. Los estándares basados en radio operan dentro de la banda 2.4 *GHz*. Estas bandas de frecuencia son reconocidas por los reguladores internacionales, como *FCC* (USA), *ETSI* (Europa), y la *MKK* (Japón), como operaciones de radio sin licencia, para usos científicos, militares e industriales.

1.2.2.1 Capa Física [7]

En la Capa Física se define la modulación, señalización, características de la transmisión de datos, dos posibles topologías, tres tipos de medios inalámbricos que funcionan a cuatro posibles velocidades, potencia y banda de frecuencia.

1.2.2.1.1 Topologías que utiliza IEEE 802.11

La *IEEE 802.11* define la topología *Ad-hoc* y la topología Infraestructura.

- **Topología *Ad-Hoc***

Esta topología se caracteriza porque no existe punto de acceso (*AP*), las estaciones se comunican directamente entre si (punto a punto), de esta manera el área de cobertura está limitada por el alcance de cada estación individual.

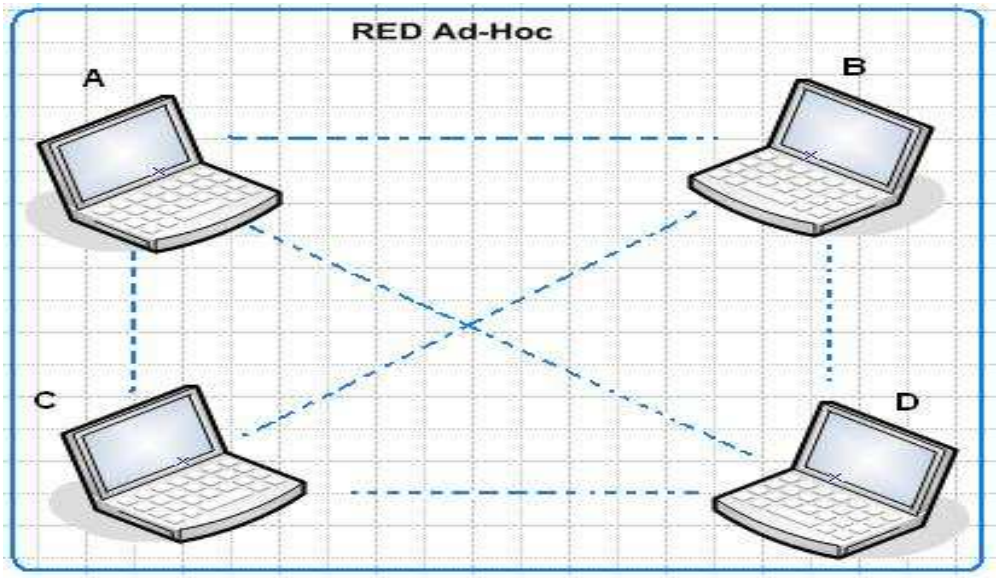


Figura 1.22 Red Ad-Hoc

- **Topología Infraestructura**

Esta topología se caracteriza por tener un punto de acceso (AP), el mismo que controla la comunicación entre estaciones, de esta manera el área de cobertura está limitada por el alcance del AP.

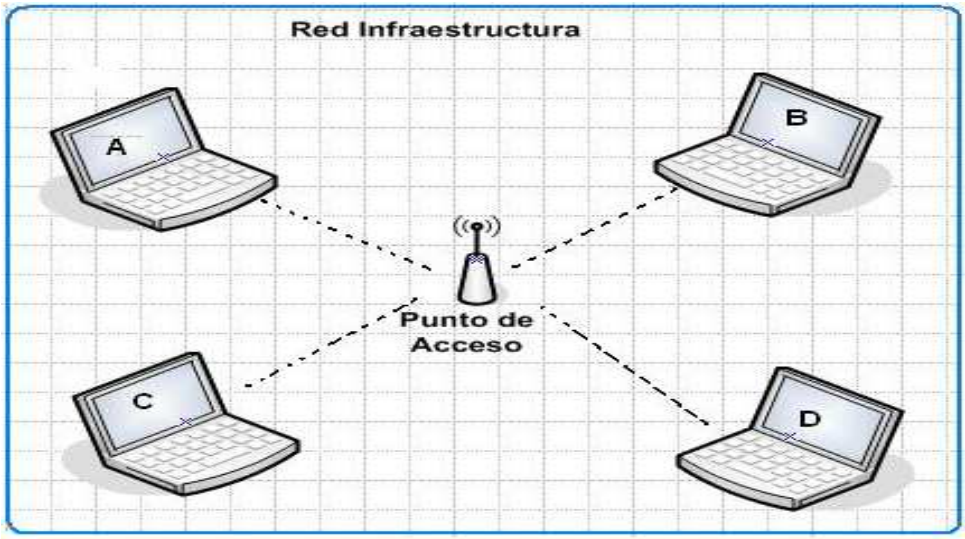


Figura 1.23 Red Infraestructura

a. Descripción general de componentes de las topologías [2]

- **Conjunto de servicios básicos (BSS):** es el bloque básico de construcción de una *LAN 802.11*. En el caso de tratarse de 2 estaciones se denomina *IBSS* (Independiente *BSS*), es lo que a menudo se denomina "red *Ad Hoc*".

- **Sistema de distribución (DS):** es la arquitectura que se utiliza para interconectar distintos *BSS*. El *AP* es el encargado de proveer acceso al *DS*, todos los datos que se mueven entre *BSS* y *DS* se hacen a través de estos *AP*.

- **Conjunto de servicios extendidos (ESS):** tanto *BSS* como *DS* permiten crear redes inalámbricas de tamaño arbitrario, este tipo de redes se denominan redes *ESS*.

La integración entre una red *802.11* y una no *802.11* se realiza

mediante un Portal. Es posible que un mismo dispositivo cumpla las funciones de *AP* y Portal.

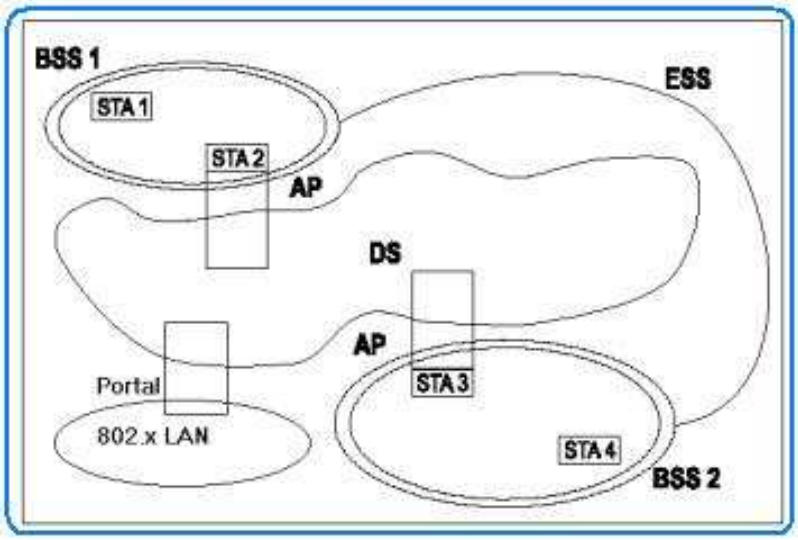


Figura 1.24 Componentes de la Arquitectura [2]

b. Servicios del Sistema de Distribución [2]

Tiene que ver con la administración de los miembros dentro de una celda y con la movilidad de las estaciones conforme entran y salen de la misma.

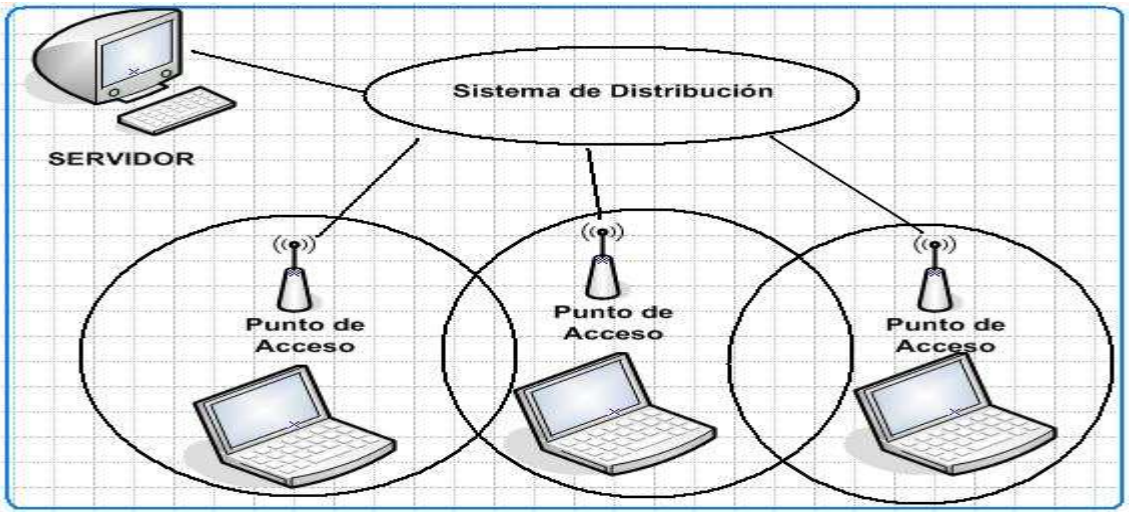


Figura 1.25 Direccionamiento en Modo Infraestructura

- **Distribución:** se encarga de llevar un paquete del punto de acceso de origen al de destino.

- **Integración:** se encarga de acoplar un sistema *IEEE 802.11* con otros sistemas *IEEE 802.x.*

  En concreto, define el componente portal que se encargará de aspectos necesarios como redireccionamiento.

- **Asociación:** servicio necesario para que una estación pueda adherirse al modo infraestructura y utilizar sus servicios.

- **Reasociación:** consiste en el campo de punto de acceso al que se asocia la estación para adherirse al modo infraestructura. También se utiliza para modificar las características de la asociación.

- **Autenticación y Desautenticación:** proceso necesario para que la estación se pueda conectar a la W*LAN* y consiste en la identificación de la estación.

- **Privacidad:** este servicio utilizará *WEP* para el encriptado de los datos en el medio.

- **Reparto de MSDUs entre STAs:** este es el servicio básico de intercambio.

c. Servicios de estación en el Sistema de Distribución [2]

Se relacionan con la actividad dentro de una sola celda:

- **Autenticación:** proceso necesario para que la estación pueda conectarse a la red *WLAN.*

- **Desautenticación:** cuando una estación previamente autenticada quiere abandonar la red.

- **Entrega de datos:** se realiza la entrega de datos entre las diferentes estaciones de trabajo.

1.2.2.1.2 Banda de Frecuencia utilizada por IEEE 802.11 [2]

Wi-Fi opera en la banda *ISM*, la cual es una banda de frecuencia abierta a cualquier sistema de radio independientemente del lugar del planeta donde se encuentre implementada. Sólo la banda *ISM* de 2,45 *GHz* cumple con este requisito, con rangos que van de los 2.4 *GHz* a los 2.5 *GHz*, y solo con algunas restricciones en países como Francia, España y Japón que se muestran en la tabla 1.6.

Localización Geográfica	**Rango Regulatorio**	**Canales RF**
USA, Europa	2.400 – 2.4835 GHz	F = 2402+ K*MHz , K = 0,....,76
España	2.445 – 2.475 GHz	F = 2449+ K*MHz , K = 0,....,22
Francia	2.4465 – 2.4835 GHz	F = 2454+ K*MHz , K = 0,....,22

Tabla 1.6 Banda de Frecuencia *Wi-Fi* [2]

1.2.2.1.3 Modulación [7]

El *IEEE 802.11* define tres tipos de modulación en la capa física para la transmisión y recepción de tramas.

- Espectro expandido por secuencia directa *DSSS*
- Espectro expandido por salto de frecuencias *FHSS*
- Luz Infrarroja en banda base sin modular

La explicación de cada modulación se encuentra en el *ANEXO H*.

1.2.2.1.4 Potencia [2]

Las antenas operan con un determinado nivel de potencia entregado por el transmisor. En el caso de *IEEE 802.11* se ajustan normalmente a 100 *mW* que es la potencia máxima permitida en Europa para la emisión de puntos de acceso o *NIC Wi-Fi*.

La limitación de potencia impuesta por las distintas autoridades influye evidentemente en la cobertura inalámbrica. A continuación se muestra los niveles de potencia permitidos en cada una de las regiones para la banda de 2.4 *Ghz*.

El nivel de potencia máximo permitido en este rango de frecuencias varía de un país a otro según sus normas regulatorias.

- Estados Unidos la *FCC* (*Federal Communication Commission*) a través de la norma Part. 15.247 limita la radiación de antena a 1*W* de potencia.
- En Japón el *MPT ordinance* 79, fija el nivel de potencia a 10 *mW* por 1 *MHz*
- En Europa el *ETS300-328 ETS96* (*European Telecommunications Standards Institute*) es el encargado de regular los límites de potencia de emisión, ésta limita la potencia hasta 100 *mW*.

Máxima potencia de salida	Localización Geográfica	Documento de Complacencia
1000 mW	EE.UU.	FCC 15.247
100 mW	EUROPA	ETS 300-328
10 mW/MHz	JAPÓN	MPT ordinance 79

Tabla 1.7 Niveles de Potencia de Transmisión para diferentes Regiones [2]

1.2.2.2 Capa Enlace de Datos [7]

La principal función de esta capa es el control de acceso al medio, realizando funciones de fragmentación, encriptación, gestión de alimentación eléctrica, sincronización y soporte de *roaming* entre múltiples *APs.*

La capa enlace de datos se divide en dos subcapas: Control de Enlace Lógico (*LLC*) y Control de Acceso al Medio (*MAC*). El estándar *802.11* utiliza el mismo *LLC* que el *802.2*, pero el nivel *MAC* es diferente.

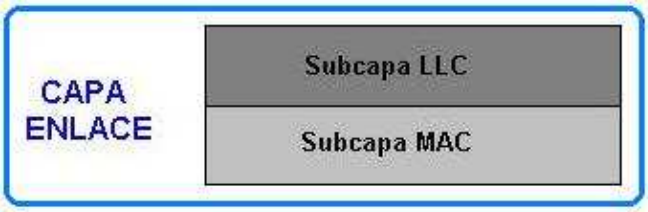


Figura 1.26 Capa Enlace de Datos [7]

1.2.2.2.1 Estructura de trama de la Capa de Enlace de Datos [2]

El estándar *802.11* define 3 clases diferentes de tramas: trama de datos, de control y de administración. Cada una de ellas tiene un encabezado con campos utilizados dentro de la subcapa *MAC* y algunos usados por la capa física.

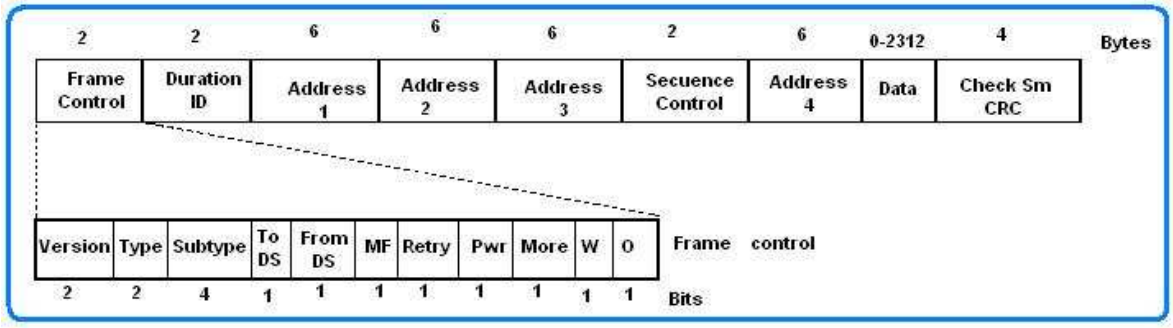


Figura 1.27 Trama Capa Enlace de Datos [2]

- **Frame Control:** tiene 11 subcampos:
  - ✓ **Versión del protocolo**: indica el tipo de protocolo de control con el que se trabaja
  - ✓ **Tipo:** tipo de datos, de Control o de Administración: ejemplo. *RTS* o *CTS.*
  - ✓ **To DS:** indica que la trama va al Sistema de Distribución.
  - ✓ **From DS:** indica que la trama viene del Sistema de Distribución.
  - ✓ **MF:** indica que siguen más fragmentos.
  - ✓ **Retransmisión:** indica que ésta es la retransmisión de una trama.

- ✓ **Pwr:** usado por la estación base para poner a una estación en estado de escucha o sacarla de este estado.
- ✓ **More:** indica que el emisor tiene tramas adicionales para el receptor.
- ✓ **W:** especifica que el cuerpo de la trama se ha empleado usando el algoritmo *WEP.*
- ✓ **O:** una secuencia de tramas que tenga este bit encendido debe procesarse en orden.

- **Duración:** indica cuanto tiempo ocuparán el canal, la trama y su *ACK*.
- **Direcciones:** 2 son para origen y destino y las otras 2 se usan para las estaciones base origen y destino, en el caso de tráfico entre celdas.
- **Secuencia:** permite numerar los fragmentos, 12 bits identifican la trama y 4 al fragmento.
- **Datos:** contiene la carga útil, hasta 2312 *bytes.*
- ***Checksum:*** contiene la suma de verificación.

1.2.2.2.2 Control de Acceso al Medio (MAC) [2] [7]

La principal función de esta capa es el control de acceso al medio, realizando funciones de fragmentación, encripción y sincronización.

- Acceso al canal
- Direccionamiento de las *PDU*
- Formato de las tramas
- Comprobación de errores
- Fragmentación y ensamblado de las *MSDU*
- Autenticación y privacidad para permitir servicios seguros
- Servicios de gestión *MAC* para permitir *Roaming* dentro de un *ESS* y para control de potencia de estaciones.

El estándar *IEEE* define dos modos de operación posibles:

- *DCF* (Función de Coordinación Distribuida)
- *PCF* (Función de Coordinación Puntual)

b. PCF Función de Coordinación Puntual [8]

Permite proporcionar diferenciación de servicios para soportar aplicaciones en tiempo real, pero resulta bastante ineficiente y compleja de implementar, *PCF* se puede implementar mediante un *AP* (Punto de Acceso), para permitir transmisión orientada a conexión de tramas *MAC* dentro de un intervalo de tiempo máximo.

PCF puede alternar entre:

- *CP* (*Contention Period*)
- *CFP* (*Contention - Free Period*): la utilización del medio está controlada por el *AP*, con el que se elimina la necesidad que las estaciones luchen por conseguir acceso al canal.

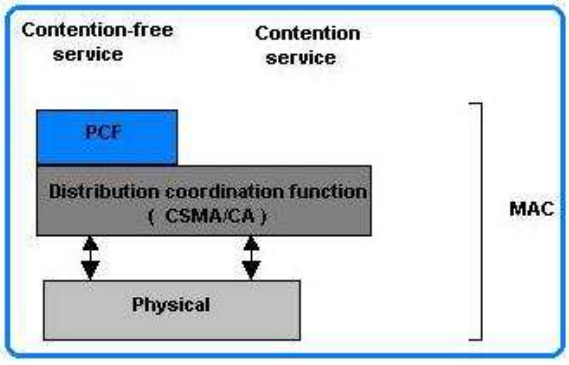


Figura 1.28 Estructura MAC [7]

c. DCF Función de Coordinación Distribuida [8]

Permite la transmisión de datos empleando el método del mejor esfuerzo, este método de contención se basa *CSMA/CA*. En redes *Ad-hoc* se hace uso exclusivo de *DCF*.

b.1 Modo de contención CSMA/CA (*Carrier Sense Multiple Access, Collision Avoidance*). [8]

Esta clase de métodos de acceso, denominados protocolos de acceso por contienda, son muy efectivos si la carga de uso del medio no es muy alta, ya que esto permitirá a las estaciones transmitir con un retardo mínimo. Hay que tener en cuenta además que pueden producirse colisiones debido a la posibilidad, que 2 estaciones “escuchen” el medio simultáneamente, detectando que esté libre e iniciando su transmisión al mismo tiempo.

El método que más se utiliza en redes inalámbricas es el *CSMA/CA*. Este protocolo evita colisiones en lugar de descubrir una colisión, como el algoritmo usado en la *IEEE 802.3*. En una red inalámbrica es difícil descubrir colisiones. Es por ello que se utiliza el *CSMA/CA* y no el *CSMA/CD* debido a que entre el final y el principio de una transmisión suelen provocarse colisiones en el medio. En *CSMA/CA*, cuando una estación identifica el fin de una transmisión espera un tiempo aleatorio antes de transmitir su información, disminuyendo así la posibilidad de colisiones.

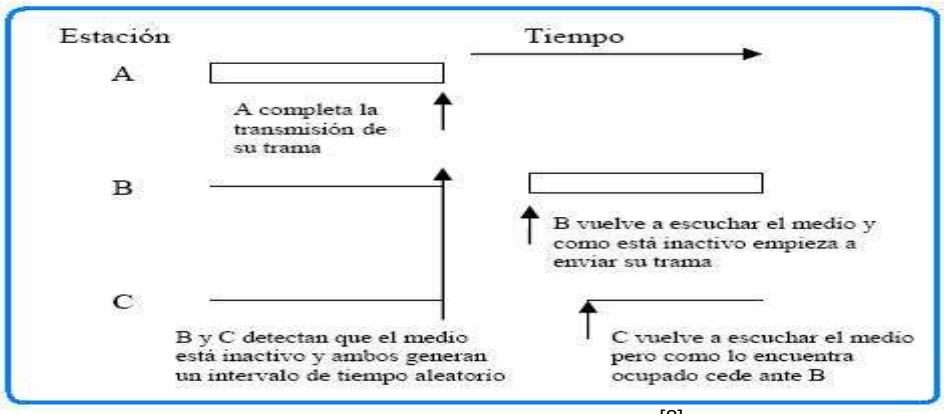


Figura 1.29 Método CSMA/CA [8]

b.1.1 Funcionamiento de CSMA/CA [8] [9]

Antes de transmitir información, la estación debe sensar el medio, o canal inalámbrico, para determinar su estado (libre/ocupado). Si el medio está desocupado la estación ejecuta una espera adicional llamada espaciado entre

tramas (*IFS*). Si durante este intervalo temporal, el medio continúa ocupado, la estación debe esperar hasta que finalice la transmisión antes de realizar cualquier otra acción.

Una vez finalizada esta espera, la estación ejecuta el algoritmo de *Backoff*, para determinar una espera adicional y aleatoria escogida uniformemente en un intervalo llamado ventana de contienda (*CW*). El algoritmo de *Backoff* da un número aleatorio y entero de ranuras temporales y su función es la de reducir la probabilidad de colisión, que es máxima cuando varias estaciones están esperando a que el medio quede libre para transmitir.

Mientras se ejecuta el algoritmo de *Backoff*, la estación continúa escuchando el medio hasta que el medio este libre. Si el medio continúa ocupado durante un tiempo igual o superior a *IFS*, el algoritmo de *Backoff* queda suspendido hasta que se cumpla esta condición. Cada retransmisión provocará que el valor de *CW*, que se encontrará entre *CWmin* y *CWmax* se duplique hasta llegar al valor máximo, el valor de las ranuras temporales para *IEEE 802.11b* es 20 *µseg.*

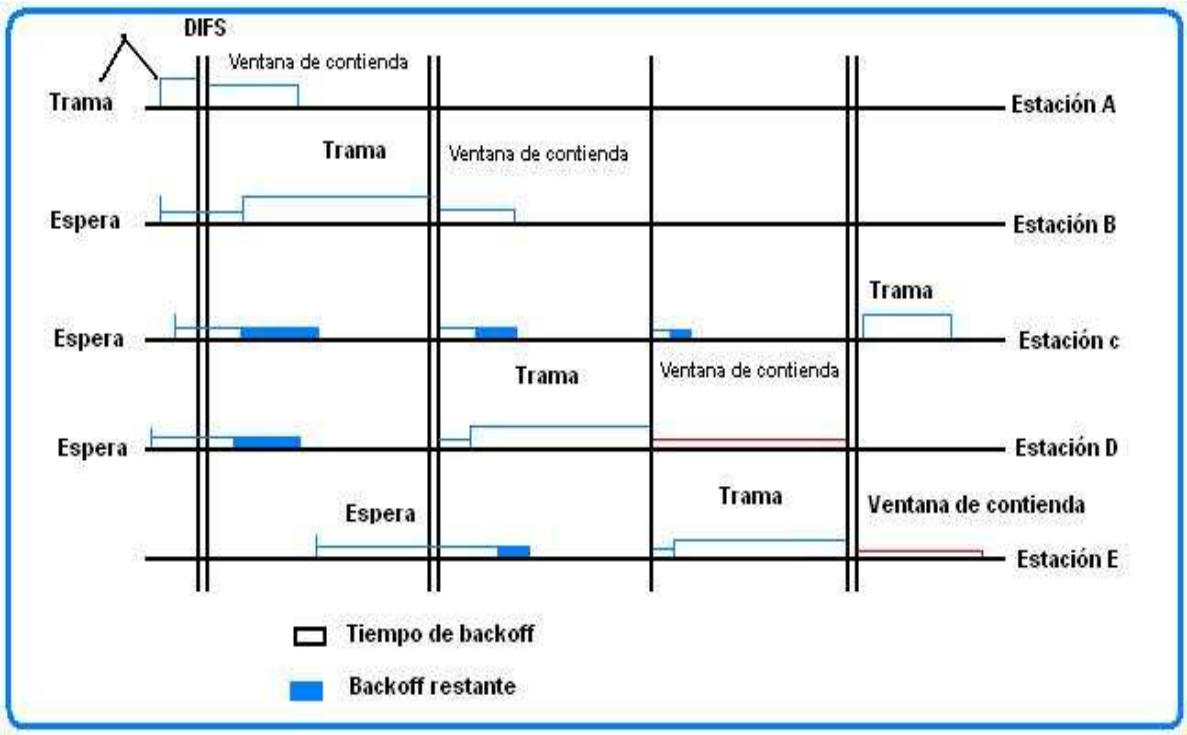


Figura 1.30 Acceso CSMA/CA [8]

Sin embargo, *CSMA/CA* en un entorno inalámbrico presenta una serie de problemas que se intentan resolver con alguna modificación. Los dos principales problemas que se pueden detectar son:

- "Nodos ocultos: una estación cree que el canal está libre, pero en realidad está ocupado por otro nodo que no oye." [9]

- "Nodos expuestos: una estación cree que el canal está ocupado, pero en realidad está libre pues el nodo al que oye no le interferiría para transmitir a otro destino, la solución que propone *802.11* es *MACA (Multiple Acces with Collision Avoidance)"*. [9]

La figura 1.31 representa u ejemplo de nodo escondido.

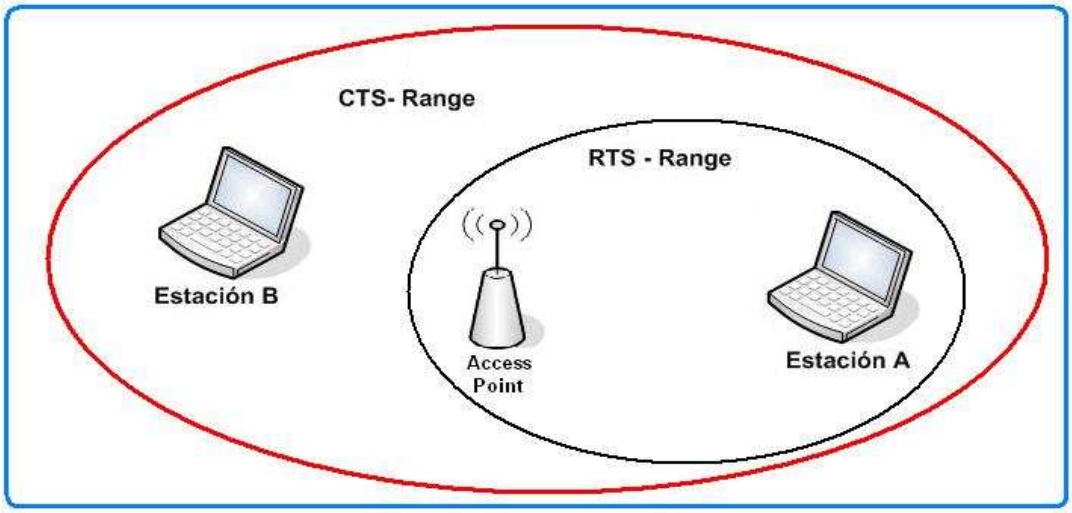


Figura 1.31 Ejemplo de nodo escondido

Un dispositivo inalámbrico puede transmitir con la potencia suficiente para que sea escuchado por un nodo receptor, pero no por otra estación que también desea transmitir y que por tanto no detecta la transmisión.

Para resolver este problema, la norma *IEEE 802.11* ha añadido al protocolo de acceso *CSMA/CA* un mecanismo de intercambio de mensajes con reconocimiento positivo, al que denomina *Reservation - Based Protocol*.

b.2 Modo de contención CSMA/CA con RTS/CTS [8]

Cuando una estación está lista para transmitir, primero envía una solicitud (*RTS*) al punto de acceso quien difunde el *NAV* (*Network Allocation Vector*) a todos los demás nodos para que queden informados de que se va a transmitir y cuál va a ser la duración de la transmisión. Si no encuentra problemas, responde con una autorización *(CTS)* que permite al solicitante enviar su trama (datos). Si no se recibe la trama *CTS,* se supone que ocurrió una colisión y los procesos *RTS* empiezan de nuevo.

Después de que se reciba la trama de los datos, se devuelve una trama de reconocimiento (*ACK*) notificando al transmisor que se ha recibido correctamente la información.

Aun así permanece el problema de que las tramas *RTS* sean enviadas por varias estaciones a la vez, sin embargo estas colisiones son menos dañinas ya que el tiempo de duración de estas tramas es relativamente corto.

Este protocolo también puede utilizarse si no existen dispositivos auxiliares en las redes *Ad-hoc*, en este caso no aparecería la trama *NAV*.

Se definen cuatro espaciados entre tramas (*IFS*) para dar prioridad de acceso al medio inalámbrico.

- **SIFS (*Short IFS*).** Es el tiempo de espera más corto, provoca transmisiones inmediatas. Usado para transmisiones de *ACKs, RTS* y *CTS*

- **DIFS (*DCF*).** Es el tiempo de espera habitual en las contiendas con mecanismo MACA.

La figura 1.32 representa el modo de contención CSMA/CA con RTS/CTS

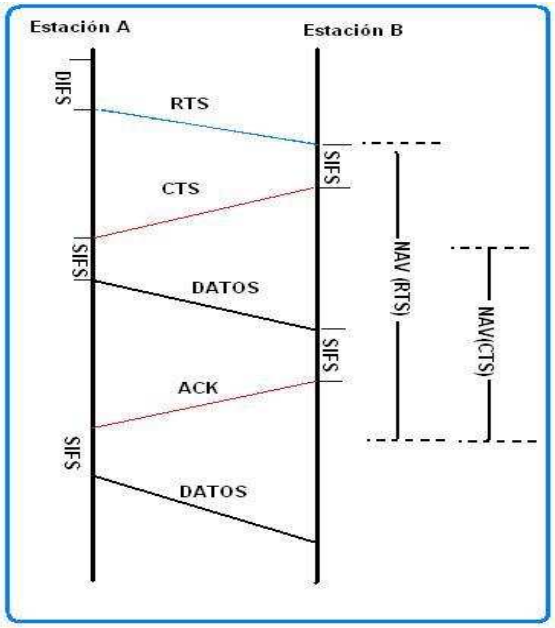


Figura 1.32 Modo de contención *CSMA/CA* con *RTS/CTS* [7]

b.3 Control de Acceso al Medio MACAW (*Multiple Acces with Collision Avoidance*) [8]

Con base en estudios de simulación de *MACA, Bharghavan* (1994) afinaron el *MACA* para mejorar su desempeño y llamaron *MACAW* a su nuevo protocolo. En *MACAW* las tramas perdidas no son retransmitidas en el nivel de enlace sino que se debe esperar a que la información llegue al nivel de transporte. Esto es resuelto obligando al receptor a enviar una trama *ACK* de datos correctamente enviados.

Además el algoritmo de *Backoff* se ejecuta para cada trama de datos fuente-destino para cada estación. Este cambio provoca la imparcialidad del protocolo. Finalmente, se añade un mecanismo para estaciones que intercambian información sobre la congestión y para hacer que el algoritmo de *backoff* no reaccione tan violentamente ante problemas temporales.

b.3.1. Funcionamiento de MACAW

MACAW funciona de manera similar a CSMA/CA, *MACAW* utiliza un intercambio de mensajes *RTS/CTS/DS* (*Data Send*) - DATA-ACK, además de implementar modificaciones al algoritmo de retransmisión o de *backoff.*

La utilización de un *ACK* en este nivel mejora los tiempos de respuesta, comparándolos con los que se obtendría si se dejara manejar la situación por el protocolo de nivel de transporte.

El nuevo *DS* permite distribuir la información de sincronización sobre los períodos de contienda, de forma que los nodos puedan “pelear" de igual forma por una ranura de tiempo para solicitar la transmisión.

La transmisión se lleva a cabo de la siguiente manera, el emisor envía un *RTS* al receptor, quien responderá con un *CTS*, una vez recibido el *CTS,* el emisor envía un *DS* seguido de los datos a transmitir. En caso de recibirse correctamente los datos el receptor devuelve un *ACK*, caso contrario no lo hace y se retransmite la información siguiendo el mismo esquema partiendo con el *RTS.*

En el caso de que el *ACK* se pierda, se enviará un nuevo *RTS* al cual se le responderá nuevamente con el mismo *ACK*.

1.2.3 Seguridad en Wi-Fi

1.2.3.1 WEP (*Wired Equivalent Protocol*) [25]

Es un sistema de encriptación estándar propuesto por el comité *802.11*, implementando a nivel de la capa *MAC* del modelo *OSI.* Dicho estándar comprime y cifra los datos que se envían a través de las ondas de radio.

WEP ofrece dos niveles de seguridad, encriptación a 64 o 128 bit. La encriptación usa un sistema de claves. La clave de la tarjeta de red del cliente debe coincidir con la clave del *AP*.

WEP utiliza una palabra clave que va a ser utilizada para autenticarse en redes cerradas y para cifrar los mensajes de la comunicación. Para generar la clave, en muchos *AP* se pide una frase y luego a partir de ella se generan 4 claves distintas para garantizar el máximo azar en la elección de la misma, pero en otros

simplemente se pide que se introduzca una clave con las restricciones de longitud que se configure y listo.

Para el cifrado de cada trama se añadirá una secuencia cambiante de bits, que se llama Vector de Inicialización *(IV),* para que no se utilice siempre la misma clave de cifrado y descifrado. Así, dos mensajes iguales no generarán el mismo resultado cifrado ya que la clave va cambiando. Cuando se tiene 4 claves, se debe marcar cual es la que se utiliza ya que *WEP* sólo utiliza 1 clave para todo. Si se ha seleccionado una opción de clave *WEP* de 64 bits, se tendrá 5 octetos (40 bits) son la clave y los 24 bits restantes serán el Vector de Inicialización *(IV)*. Es decir, en una comunicación normal se tendrá 2^{24} claves distintas de cifrado.

En el caso de *WEP* de 128 bits se tiene 13 octetos fijos (104 *bytes*) y 24 bits cambiantes, es decir, se tendrá el mismo número de claves pero de mayor longitud.

La figura 1.33 representa la creación de claves en seguridad con *WEP*

a. Creación de las claves en *WEP*

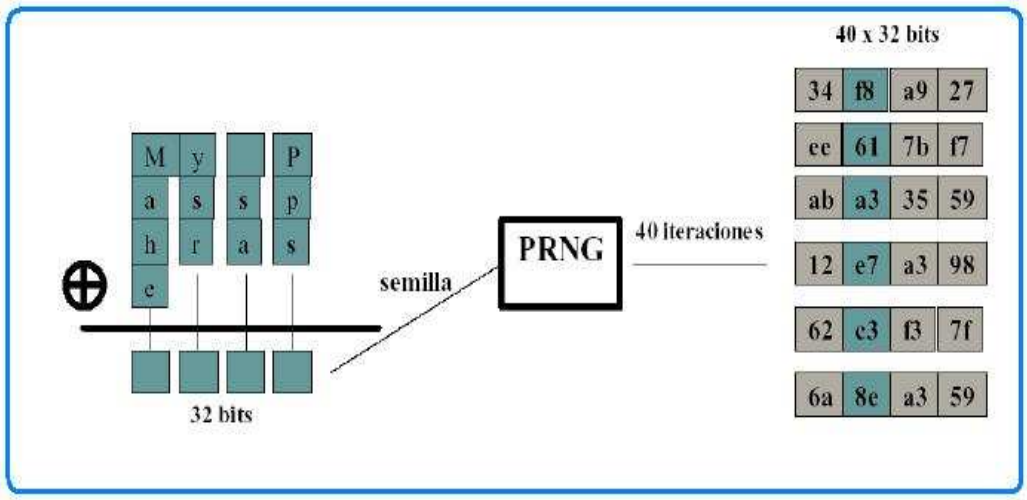


Figura 1.33 Creación de claves en WEP [25]

WEP utiliza el algoritmo *RC4* para la encriptación con claves de 64 bits, aunque existe también la posibilidad de utilizar claves de 128 bits. En realidad son 40 y 104 bits, ya que los otros 24 van en el paquete como Vector de Inicialización *(IV)*.

La clave de 40 o 104 bits, se genera a partir de una clave (*passphrase*) estática de forma automática, aunque existe *software* que permite introducir esta clave manualmente. La clave debe ser conocida por todos los clientes que quieran conectarse a la red inalámbrica que utiliza *WEP*, esto implica que muchas veces se utilice una clave fácil de recordar y que no se cambie de forma frecuente. A partir de la clave se generan 4 claves de 40 bits, sólo una de ellas se utilizará para la encriptación *WEP*.

Para generar las claves se hace una operación *XOR* con la cadena *ASCII* (*My Passphrase*) que queda transformada en una secuencia de 32 bits que utilizará el generador de números *pseudoaleatorios (PRNG)* para generar 40 cadenas de 32 bits cada una. Se toma un bit de cada una de las 40 cadenas generadas por el *PRNG* para construir una clave y se generan 4 claves de 40 bits.

1.2.3.2 WPA (*Wireless Application Protocol*) [25]

WPA emplea el cifrado de clave dinámico, lo que significa que la clave está cambiando constantemente y hacen que las incursiones en la red inalámbrica sean más difíciles que con *WEP*. *WPA* está considerado como uno de los más altos niveles de seguridad inalámbrica para su red, es el método recomendado si su dispositivo es compatible con este tipo de cifrado. Las claves se insertan como dígitos alfanuméricos, sin restricción de longitud, en la que se recomienda utilizar caracteres especiales, números, mayúsculas y minúsculas, y palabras difíciles de asociar entre ellas o con información personal.

Dentro de *WPA*, hay dos versiones de *WPA*, que utilizan distintos procesos de autenticación:

- **Para el uso personal doméstico:** el protocolo de integridad de claves temporales *(TKIP)* es un tipo de mecanismo empleado para crear el cifrado de clave dinámico y autenticación mutua. *TKIP* aporta las características de seguridad que corrige las limitaciones de *WEP*. Debido a que las claves están en constante cambio, ofrecen un alto nivel de seguridad para su red.

- **Para el uso empresarial de negocios:** el protocolo de autenticación extensible *(EAP)* se emplea para el intercambio de mensajes durante el proceso de autenticación. Emplea la tecnología de servidor *802.1x* para autenticar los usuarios a través de un servidor *RADIUS* (Servicio de usuario de marcado con autenticación remota). Esto aporta una seguridad de fuerza industrial para su red.

  Servidor *RADIUS* distribuye claves diferentes a cada usuario a través del protocolo *802.1x.*

  En la figura 1.34 se puede ver un ejemplo de autenticación *WPA*

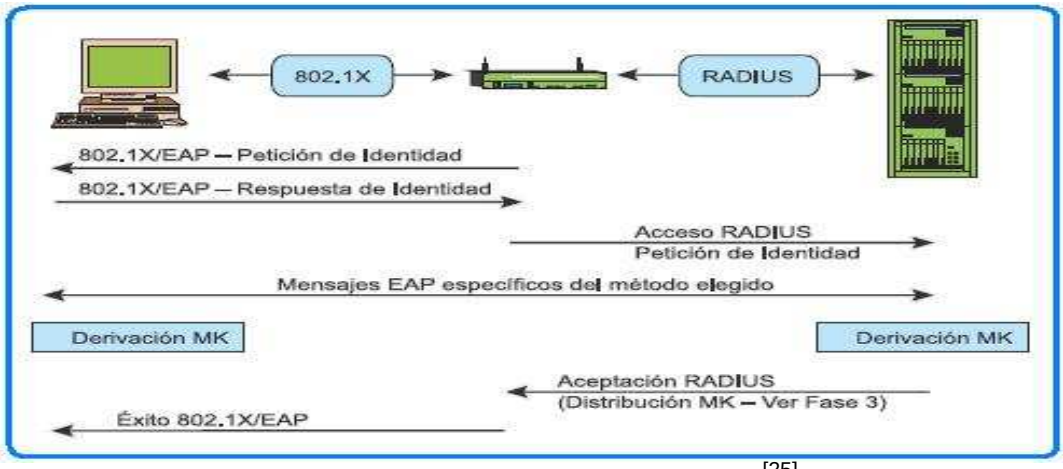


Figura 1.34 WPA (Protocolo de Autenticación) [25]

1. El cliente se asocia con el Punto de Acceso, que bloquea su tráfico
2. El cliente presenta credenciales que son autentificadas por *RADIUS*
3. El cliente autentifica al servidor *RADIUS* (*EAP-MD5* no válido)
4. Cliente y *RADIUS* derivan clave *WEP Unicast* (clave inicial *TKIP*)
5. Punto de acceso envía clave *WEP broadcast* cifrada con *WEP unicast*
6. Punto de acceso y cliente cifran sus comunicaciones

1.2.3.3 802.11i o WPA2 [25]

El *Task Group* de *IEEE 802.11i*, se conformó en el año 2001, con la intención de analizar una arquitectura de seguridad más robusta y escalable, debido a la inminente demanda del mercado en este tema y en julio de 2004 aprobó este

estándar. Por su parte la *Wi-Fi Alliance* lo lanzó al mercado en septiembre de ese año. En forma resumida, este nuevo estándar, propone a *802.1x* como protocolo de autenticación, pudiendo trabajar con su referencia *EAP*, éste último proporciona una gran flexibilidad en la metodología de autenticación.

Previo al estándar, *Cisco Systems* ofreció el primer tipo de autenticación que se denominó *LEAP* (*Lightweight EAP*), protocolo que inicialmente fue propietario de *Cisco*, pero en la actualidad lo emplean varios fabricantes.

Por su parte *Microsoft,* inicialmente junto con *Windows XP* lanzó al mercado su protocolo denominado *EAP/TLS* (*EAP with Transport Layer Security*), y fue aceptado por *IEEE*, se basa en certificados en lugar de contraseñas como credenciales de autenticación. Otros fabricantes han presentado *EAP/TTLS* (*EAP with Tunneling Transport Layer Security*), el cual realiza un túnel de nivel 2 entre el cliente y el *AP*, una vez establecido el túnel, *EAP/TTLS* opera sobre él, lo cual facilita el empleo de varios tipos de credenciales de autenticación que incluyen contraseñas y certificados, en realidad no deja de ser una variante de *EAP/TLS*.

La última variante es *PEAP* (*Protected Extensible Authentication Protocol*), inicialmente fue la versión “0” y ya está vigente la versión “1”, el cual aplica una metodología muy similar a *EAP/TTLS* en cuanto al empleo de túnel y sobre el una amplia variedad de credenciales de autenticación, este último ya está soportado por los más importantes fabricantes. En general, se considera que *PEAP* es el método más seguro del momento. Este protocolo fue desarrollado por *Microsoft, Cisco* y *RSA*.

802.1x: Este estándar no fue presentado para *Wi-Fi*, sino para el acceso seguro *PPP* (en tecnologías de cable). Una de las grandes características de *Wi-Fi* es emplear todas las herramientas que ya existen y pueden prestar utilidad al mismo. *802.1x* es uno de los mejores ejemplos de esto. La arquitectura *802.1x* está compuesta por tres partes:

- **Solicitante:** generalmente se trata del cliente *Wi-Fi*

- **Autenticador:** suele ser el *AP*, que actúa como traspaso de datos y como bloqueo hasta que se autoriza su acceso (importante esto último).

- **Servidor de autenticación:** suele ser un Servidor *RADIUS* o *Kerberos,* que intercambiará el nombre y credencial de cada usuario. El almacenamiento de las mismas puede ser local o remoto en otro servidor de *LDAP*, de base de datos o directorio activo.

Una de las grandes ventajas de emplear *802.1x* es que el servidor de autenticación, permite también generar claves de cifrado muy robustas.

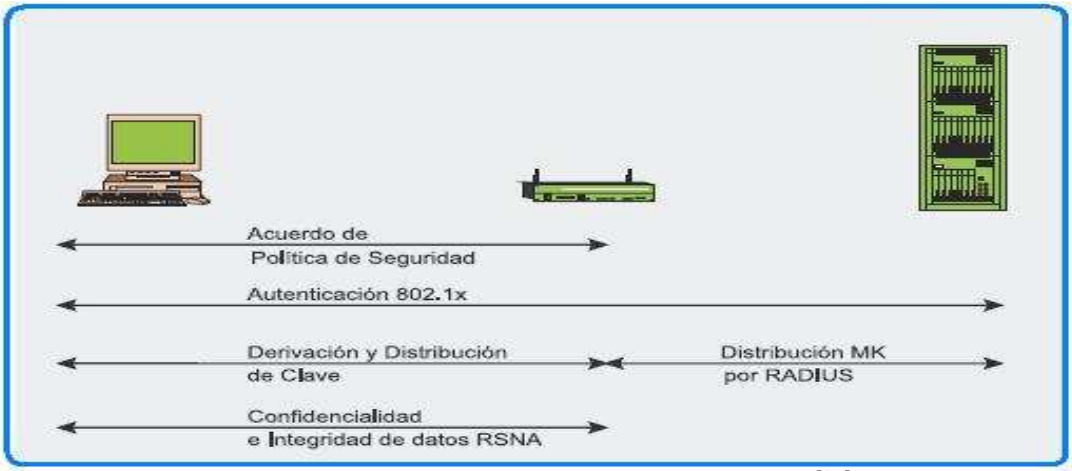


Figura 1.35 WPA2 (Protocolo de Autenticación) [25]

1.2.4 APLICACIONES

Hoy en día *Wi-Fi* amplia más sus aplicaciones gracias a que cada vez son más los dispositivos móviles que salen al mercado (ordenadores portátiles, agendas electrónicas, teléfonos). Las redes inalámbricas son la mejor solución para conectarlos a la red, aprovechando al máximo todas sus ventajas, sin perder la flexibilidad y movilidad que proporciona no necesitar cables.

- **Conexión a Internet:** La solución inalámbrica es idónea para la prestación de servicios de acceso a *Internet* o a *Intranets* en lugares donde no existe una infraestructura moderna de comunicaciones e incluso donde ya existe esta infraestructura para aportar más servicios y solucionar problemas antiguos.

Con la tecnología satélite con *Wi-fi* o *Wimax* con *Wi-Fi*, un punto de acceso es capaz de dar cobertura a varias decenas de ordenadores con un control total en el acceso como en el ancho de banda. La conexión a Internet es permanente y el usuario dispone de 11 *Mbps*.

- **Telefonía IP:** la integración de las comunicaciones de voz en las redes informáticas empresariales aporta enormes ventajas en cuanto a productividad y ahorro en comunicaciones. Soluciones de telefonía fija *IP* y de telefonía móvil *IP* basada en la norma *802.11b*. Las redes *Wi-Fi* en conjunto con la telefonía *IP* permiten que particulares o empresas de la misma población se comuniquen sin costo.

- **Redes privadas:** la ventaja que ofrece *Wi-Fi* de llegar a cualquier parte sin cables permite interconectar, a través de la red ya creada, las instalaciones o delegaciones de su empresa para la prestación de servicios de transmisión de datos a alta velocidad eliminando los costos mensuales por tráfico. La interconexión se puede realizar utilizando tantos enlaces punto a punto como sea necesario o utilizando nodos intermedios.

- **CCTV (Circuito Cerrado de Televisión):** las tecnologías inalámbricas ofrecen avanzados sistemas de seguridad. Con la llegada al mercado de las cámaras *IP* y la elevada capacidad de transmisión de la red *Wi-Fi* ha permitido desarrollar nuevos sistemas de video vigilancia remota a través de *Internet*.

  La posibilidad de obtener y enviar datos como video digital abre muchas posibilidades en el campo de la seguridad al permitir que unidades móviles dispongan a tiempo real de toda la información necesaria y de un completo control de las instalaciones.

- **Gestión de datos:** las soluciones móviles están aportando beneficios gracias a una mejor gestión de las empresas y mejora de la productividad. Además implementan nuevos servicios y reducen costos.

Las *PDAs* con conexión inalámbrica a *Internet* son utilizadas por empresas de logística para control de almacén, restaurantes, hoteles y cafeterías, para control de pedido.

1.3 FACTORES DE PROPAGACIÓN INALÁMBRICA

Son varios los factores a considerar a la hora de diseñar un sistema inalámbrico, algunos de los aspectos a tener en cuenta son los siguientes:

- Cobertura
- Rendimiento
- Interferencia

Entre los diferentes factores que afectan a la cobertura y rendimiento de un sistema inalámbrico se puede mencionar los siguientes:

1.3.1 ATENUACIÓN Y ABSORCIÓN DE ONDAS [10]

El espacio libre puede ser considerado como el vacío y no se consideran pérdidas. Cuando las ondas electromagnéticas se encuentran en el vacío, se llegan a dispersar y se reduce la intensidad de potencia a lo que es llamado atenuación. La atenuación se presenta tanto en el espacio libre como en la atmósfera terrestre. La atmósfera terrestre no se le considera vacío debido a que contiene partículas que pueden absorber la energía electromagnética y a este tipo de reducción de potencia se le llama pérdida por absorción la cual no se presenta cuando las ondas viajan afuera de la atmósfera terrestre

1.3.1.1 Atenuación [10]

La reducción de la intensidad de potencia con la distancia equivale a una pérdida de potencia y se suele llamar atenuación de la onda electromagnética. La atenuación de la onda se expresa en general en función del logaritmo de la relación de intensidades de potencia (pérdida en dB), la definición matemática de la atenuación es:

$$\gamma_a = 10\log\frac{P_1}{P_2}$$ **Ecuación 1.2** [10]

La reducción de la intensidad de potencia debida a la propagación en el espacio no libre se llama absorción.

1.3.1.2 Absorción [10]

Las ondas de radio que viajan por la atmósfera terrestre son atenuadas o debilitadas mediante la transferencia de energía a este medio. Entre los diferentes materiales que pueden absorber las ondas electromagnéticas se puede mencionar los siguientes: rocas, ladrillos, concreto, madera, árboles y otros materiales.

La absorción de onda por la atmósfera es análoga a una pérdida de potencia I^2R. Una vez absorbida la energía de onda ésta se pierde, y causa una atenuación en las intensidades de campo eléctrico, campo magnético, y una reducción en intensidad de potencia.

1.3.2 PÉRDIDAS EXISTENTES EN UN RADIO ENLACE

1.3.2.1 Pérdidas en la Trayectoria en el Espacio Libre (L_{patch}) [10]

Se define como la pérdida incurrida por una onda electromagnética al propagarse en línea recta a través del vacío, sin energías de absorción o reflexión debidas a objetos cercanos. Estas pérdidas dependen de la frecuencia, y aumentan con la distancia. La ecuación para determinar estas pérdidas es la siguiente:

$$L_{patch} = \left(\frac{4\pi \bullet R}{\lambda}\right)^2 = \left(\frac{4\pi \bullet f \bullet R}{c}\right)^2$$ **Ecuación 1.3** [10]

Donde:

L_{patch} = pérdidas en la trayectoria en espacio libre (adimensional)

R = distancia máxima entre dispositivos (metros)

f = frecuencia (hertz)

λ = longitud de onda (metros)

c = velocidad de la luz en el espacio libre ($3*10^8$ m/s)

Al pasar a dB se obtiene

$$L_{p(dB)} = 10 * \log\left(\frac{4\pi \bullet f \bullet R}{c}\right)^2$$

$$L_{p(dB)} = 20 * \log\left(\frac{4\pi \bullet f \bullet R}{c}\right)$$

$$L_{p(dB)} = 20 * \log\left(\frac{4\pi}{c}\right) + 20 * \log f + 20 * \log R \quad \textbf{Ecuación 1.4} \ [10]$$

1.3.2.2 Desvanecimiento por Múltiple Trayectoria (L_{fade}) [10]

En esencia el desvanecimiento por multitrayectoria es un factor ficticio que se incluye en la ecuación del cálculo de potencia de recepción (ecuación 1.6), para tener en cuenta las características no ideales y menos predecibles de la propagación de las ondas de radio. Estas reflexiones multitrayectoria de la onda transmitida se deben a obstáculos naturales o a objetos que actúan como dispersores, entre estos se puede citar: muebles, ventanas, paredes, puertas metálicas, etc. Pequeñas variaciones en la distancia entre emisor y receptor, del orden de una cuarta parte de la longitud de onda por ejemplo, pueden causar grandes cambios en la amplitud o en la fase de la señal.

Estas características de desvanecimientos son propias del área donde se quiera implementar la red inalámbrica, estos desvanecimientos alteran las pérdidas en la trayectoria en espacio libre, y por lo general, son perjudiciales para la eficiencia general del sistema.

Cuando la señal electromagnética se propaga por una estancia es afectada por múltiples fenómenos debido a los diferentes tipos de obstáculos que debe atravesar o en los cuales es reflejado Es por tanto imprescindible tener en cuenta estos fenómenos, que causan atenuaciones y desvanecimientos de la señal original, a la hora de diseñar un enlace inalámbrico en ambientes internos.

1.3.2.2.1 Reflexión [2]

Cuando una onda electromagnética que se propaga por el aire incide contra un objeto de grandes dimensiones en comparación con la longitud de onda se dice que ocurre reflexión. La consecuencia de este fenómeno es que la señal puede ser absorbida, reflejada o a su vez una combinación de ambas. Esta reacción depende principalmente de:

- Propiedades de la señal, como orientación, ángulo de incidencia y longitud de onda.
- Propiedades físicas del obstáculo, como su geometría, textura y composición.

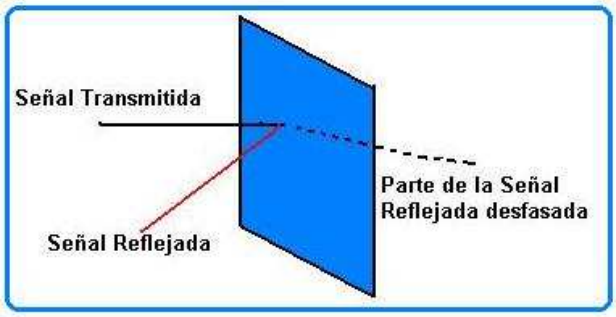


Figura 1.36 Reflexión de una señal [2]

1.3.2.2.2 Penetración [2]

La penetración es la capacidad de transmisión de una señal cuando ésta se encuentra en su camino con un obstáculo, cuando una señal penetra un obstáculo experimenta una pérdida, la cual será función del tipo de objeto.

En la tabla 1.8 se encuentran las pérdidas de penetración predecibles dependiendo del material.

Tipo de obstáculo	Pérdida (dB)
Espacio abierto	0
Ventana metálica (tintado no metálico)	3
Ventana metálica (tintado metálico)	5-8
Muros finos	5-8
Muros medios de madera	10
Muros gruesos	15-20
Muros muy gruesos	20-25
Suelo/Techo grueso	15-20
Suelo/Techo muy grueso	20-25

Tabla 1.8 Penetración a través de diferentes tipos de materiales [2]

1.3.2.2.3 Difracción [2]

La difracción ocurre cuando los obstáculos son impenetrables por las ondas de radio, el resultado de este fenómeno son ondas secundarias alrededor y detrás del obstáculo. La señal difractada depende de la geometría del objeto así como la amplitud, fase y polarización de la onda incidente en el punto de difracción.

Figura 1.37 Difracción de Señal [2]

1.3.2.2.4 Dispersión [2]

Cuando una señal transmitida se encuentra en el camino con objetos cuyas dimensiones son pequeñas con relación a la longitud de onda ocurre una dispersión. El resultado es que el frente de onda se rompe o se dispersa en múltiples direcciones. La dispersión en la práctica es provocada por:

- Señales de tráfico, focos.
- Conductos para los servicios eléctricos
- Estructuras de hierro forjado y tuberías.

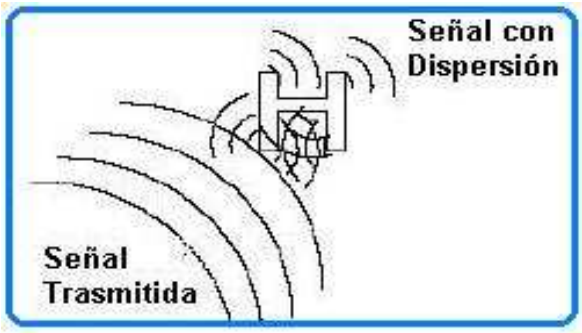


Figura 1.38 Dispersión de Señal [2]

1.3.2.2.5 Interferencia [2]

Las interferencias de radio frecuencia son uno de los asuntos más importantes a tener en cuenta para el éxito en el diseño, operación y mantenimiento de sistemas inalámbricos.

La señal de interferencia se caracteriza por ser una señal de naturaleza similar a la deseada, la misma que perturba a la señal que se desea transmitir. Se produce interferencia eléctrica cuando las señales de información de una fuente producen frecuencias que caen fuera de su ancho de banda asignado, e interfieren con otras señales de otra fuente.

“Las fuentes potenciales de interferencia de este tipo son numerosas: materiales metálicos, aislamientos, pinturas de plomo, etc. y pueden reducir la calidad de la señal radioeléctrica.” [2]

Existen otros dispositivos que utilizan la misma banda de frecuencia que también pueden ser fuente de interferencias como hornos microondas y ciertos teléfonos inalámbricos.

1.3.3 GANANCIA DE LA ANTENA [2]

La ganancia de la antena es la relación entre la intensidad de potencia radiada por la antena en una dirección específica y la intensidad de potencia radiada por una antena isotrópica alimentada con la misma potencia.

La mayoría de fabricantes especifica la ganancia de la antena en dBi (decibelios isotrópicos). Cada antena posee una ganancia diferente, la cual depende; de los materiales de elaboración, método de propagación (omnidereccionales o direccionales), es por eso que es necesario definir las características importantes de las antenas.

1.3.3.1 Características de las antenas

1.3.3.1.1 Diagrama de Radiación [2] [10]

El diagrama de radiación o lóbulo de radiación es la forma como se propaga la onda electromagnética. De acuerdo al diagrama de radiación existen dos tipos básicos de antenas que son; omnidireccionales y direccionales.

- **Antenas Omnidireccionales**

  Una antena omnidireccional es aquella diseñada para proveer un patrón de radiación de 360º. Propagan la señal de *RF* en todas las direcciones en el plano horizontal aunque tienen un rango limitado en el plano vertical. Son las más comunes en *WLAN* y se utilizan cuando se requiere dotar en un plano cobertura en todas las direcciones. Proporcionan la cobertura más amplia dentro de edificios, pudiendo formar celdas circulares mínimamente solapadas a lo largo del edificio.

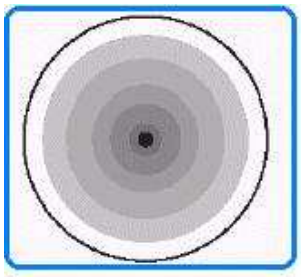


Figura 1.39 Diagrama de Radiación de una Antena Omnidireccional [2]

- **Antenas Direccionales**

  Las antenas direccionales son aquellas que han sido concebidas y construidas para que la mayor parte de la energía sea radiada en una dirección en concreto. Puede darse el caso en que se desee emitir en varias direcciones,

pero siempre que se este hablando de un número de direcciones determinado donde se encontrarán el lóbulo principal y los secundarios. Existen diferentes tipos de antena direccionales, cada una con una forma y estilo determinado, incluyendo *yagis*, antenas *patch* y parabólicas.

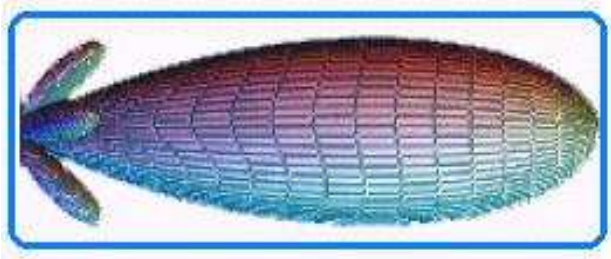


Figura 1.40 Diagrama de Radiación de una Antena Direccional [10]

1.3.3.1.2 Polarización de la Antena [10]

La polarización de una antena se refiere sólo a la orientación del campo eléctrico radiado desde ésta. Una antena puede polarizarse en forma lineal (en general, polarizada horizontal o vertical), en forma elíptica o circular.

Si una antena irradia una onda electromagnética polarizada verticalmente perpendicular al suelo, la antena se encuentra polarizada verticalmente; si la antena irradia una onda electromagnética polarizada horizontalmente paralela al suelo, se dice que la antena está polarizada horizontalmente.

1.3.3.1.3 Ancho de Banda [10]

El ancho de banda de la antena se define como el rango de frecuencias sobre las cuales la operación de la antena es "satisfactoria". Esto, por lo general, se toma entre los puntos de media potencia, pero a veces se refiere a las variaciones en la impedancia de entrada de la antena.

En *WLANs* las antenas tienen que estar sintonizadas para la banda de 2.4 *GHz* (*802.11b/g*) o 5 *GHz* (*802.11a*). Una antena funcionará de modo eficiente sólo si su ancho de banda está dentro de las frecuencias de radio utilizadas.

1.4 MODELOS PARA EL CÁLCULO DEL ENLACE

El cálculo del enlace consiste en calcular la potencia de recepción tomando en cuenta los diferentes obstáculos y dispositivos que pueden interferir con la señal transmitida y compararla con la sensibilidad del dispositivo, para determinar si estos cumplen con los requerimientos de cobertura del sistema inalámbrico.

1.4.1 MODELOS PARA EL CÁLCULO DEL ENLACE BLUETOOTH

Para el cálculo del enlace *Bluetooth* se consideró modelos para ambientes internos, ya que la implementación y las pruebas de los prototipos se las realizó en este tipo de entornos. A continuación se presentan los modelos analizados:

1.4.1.1 Modelo que considera las Pérdidas en la Trayectoria y Desvanecimientos Multitrayectoria [3] [35]

En este modelo la señal está sujeta a pérdidas en la trayectoria (L_{patch}) y desvanecimientos multitrayectoria (L_{fade}) debido a la presencia de obstáculos. Para el cálculo de la potencia que se espera recibir en el receptor se debe considerar además las ganancias de las antenas tanto transmisora como receptora, dando como resultado la siguiente expresión:

$$\mathrm{P_{RX}}[mW] = \frac{P_{TX}[mW] \bullet G_{TX} \bullet G_{RX}}{L_{patch} \bullet L_{fade}}$$ **Ecuación 1.5** [3]

Donde:

$\mathrm{P_{RX}}[mW]$ = Potencia de recepción

$P_{TX}[mW]$ = Potencia de transmisión

G_{TX} = Ganancia de la antena transmisora

G_{RX} = Ganancia de la antena receptora

L_{patch} = Pérdidas en la trayectoria

L_{fade} = Desvanecimientos por multitrayectoria

Si se transforma la ecuación 1.5 a dB se tiene:

$$P_{RX}[dBm] = P_{TX}[dBm] + G_{TX}[dB] + G_{TX}[dB] - L_{patch}[dB] - L_{fade}[dB] \quad \textbf{Ecuación 1.6} \ [3]$$

Las antenas utilizadas en la banda *ISM* no permiten antenas de alta directividad por lo que se usan antenas *omnidireccionales* con ganancia unitaria, con lo cual la ecuación 1.6 queda de la siguiente forma:

$$P_{RX}[dB] = P_{TX}[dBm] - L_{patch}[dB] - L_{fade}[dB] \quad \textbf{Ecuación 1.7} \ [3]$$

Las pérdidas por trayectoria se asumen a través de estimaciones de la siguiente forma:

$$L_{patch} = 20\log\left(\frac{4 \bullet \pi \bullet R}{\lambda}\right) \approx 40 + 20\log(R) \qquad R \leq 8.5[m] \quad \textbf{Ecuación 1.8} \ [3]$$

$$L_{patch} = 36\log\left(\frac{4 \bullet \pi \bullet R}{\lambda}\right) - 46.7[dB] \approx 25.3 + 36\log(R) \qquad R > 8.5[m] \quad \textbf{Ecuación 1.9} \ [3]$$

"Las pérdidas por desvanecimientos (L_{fade}) se asumen de 8[dB], se asume este valor ya que se estima una confiabilidad del sistema igual al 90%" [35]. Los desvanecimientos más profundos serán absorbidos por la diversidad de frecuencia del canal y causarán solo interrupciones ocasionales.

Entonces la ecuación para el cálculo del enlace para *Bluetooth* es la siguiente:

$$P_{RX}[dB] = P_{TX}[dBm] - L_{patch}[dB] - 8[dB] \quad \textbf{Ecuación 1.10} \ [3]$$

1.4.1.2 Modelo de Atenuación Lineal por Trayectoria

Este modelo es realmente sencillo en cuanto a su parte de aplicación teórica, pues la parte real de mediciones por pérdidas por trayectoria tienen un gran peso. Andelman lo propuso en el 2004 como un modelo a utilizarse cuando el transmisor y el receptor se encuentran en el mismo piso.

Este modelo de pérdidas por trayectoria lineal toma en cuenta trayectorias para interiores a partir de la potencia radiada, estas pérdidas están dadas por las pérdidas en el modelo del espacio libre, más el factor lineal que se obtiene experimentalmente.

La ecuación que describe las pérdidas en este modelo es la siguiente:

$$L = L_{FS} + a * d \quad \textbf{Ecuación 1.11} \ [18]$$

Donde:

L_{FS} = Pérdidas en el espacio libre

a = Coeficiente de atenuación lineal

d = distancia entre el transmisor y el receptor

"El coeficiente de atenuación lineal para un ambiente de oficinas es a=0.47 [dB/m]" [18]. Introduciendo la ecuación 1.4 en la ecuación 1.11 se tiene la siguiente ecuación:

$$L = 20 * \log\left(\frac{4\pi * f}{C}\right) + 20 * \log(d) + a * d \quad \textbf{Ecuación 1.12}$$

Para una frecuencia f=2.4 *GHz*, la velocidad de la luz C =3 * 10^8 m/s y el coeficiente de atenuación lineal a = 0.47 se tiene:

$$L = 40.1 + 20 * \log(d) + 0.47 * d \quad \textbf{Ecuación 1.13}$$

Tomando en cuenta que se usan antenas *omnidireccionales* con ganancia unitaria y las pérdidas totales la ecuación para el cálculo del enlace para *Bluetooth* es la siguiente:

$$P_{RX}[dBm] = P_{TX}[dBm] - 20 * \log(d)[dB] - 0.47 * d[dB] - 40.1[dB] \quad \textbf{Ecuación 1.14}$$

1.4.2 MODELOS PARA EL CÁLCULO DEL ENLACE Wi-Fi

Para el cálculo del enlace *Wi-Fi* al igual que en *Bluetooth* se utilizó modelos de propagación para ambientes internos, ya que las pruebas del prototipo se las realizó en este tipo de entornos. A continuación se analizarán los siguientes modelos:

1.4.2.1 Modelo de Pérdidas de Propagación de una Pendiente (1SM: *one-slope model)* [19] [36]

Este modelo (*one-slope model*), supone una dependencia lineal entre las pérdidas del trayecto en dB, y el logaritmo de la distancia.

$$L[dB] = L_0[dB] + 10 \bullet n \bullet \log(d)[dB]$$ **Ecuación 1.15** [19]

Donde:

L_0 = atenuación del trayecto para a una distancia de 1m.

n = exponente de pérdidas

d = distancia entre el transmisor y el receptor en (m)

Si se considera las pérdidas en el espacio libre:

$$L[dB] = 32.45 + 20 * \log(d) + 20 * \log(f)$$ **Ecuación 1.16** [36]

Para una distancia de un metro Lo:

$$Lo\ [dB] = 32.45 + 20 * \log(f)$$; *f* en *GHz*

Es el modelo más sencillo de utilizar pero necesita de una adecuada clasificación del tipo de edificio, para obtener un exponente de pérdidas para cada entorno que minimice la desviación típica. El principal problema de este modelo se da cuando se pretende utilizar en edificios de varias plantas, donde da lugar a grandes errores, por lo que se suele incluir un factor de pérdidas por penetración en el suelo.

La ecuación para el cálculo de las pérdidas para la banda de 2.4 *Ghz* es:

$$L[dB] = 40.1[dB] + 10 \bullet n \bullet \log(d)[dB] \quad \textbf{Ecuación 1.17} \ [19]$$

Tomando en cuenta que se usan antenas *omnidireccionales* con ganancia unitaria para la banda *ISM* la ecuación para el cálculo del enlace para *Wi-Fi* es la siguiente:

$$P_{RX}[dBm] = P_{TX}[dBm] - 10 \bullet n \bullet \log(d)[dB] - 40.1[dB] \quad \textbf{Ecuación 1.18}$$

Donde n toman los siguientes valores:

AMBIENTE	n
1 piso	4
atraviesa 2 paredes	5.2
Atraviesa más de 2 paredes	5.4
Abierto	1.9
Grande	2
Corredor	1.4

Tabla 1.9 Exponente de Pérdidas [19]

1.4.2.2 Modelo de Pérdidas con Factores de Atenuación por Suelo y Pared (MWM) [12] [20]

El modelo de *Keenan-Motley,* llamado también como modelo multi-pared (*MWM*), añade las pérdidas introducidas por las paredes y los suelos que atraviesa la onda directa entre el transmisor y el receptor. Su formulación más general viene dada por la siguiente expresión:

$$L = L_o + 10 * n * \log(d) + \sum_{i=1}^{I} K_{wi} * L_{wi} + \sum_{j=1}^{J} K_{fj} * L_{fj} \quad \textbf{Ecuación 1.19} \ [12]$$

L_o = pérdidas de referencia a 1 m en espacio libre.

n = pendiente de pérdidas con la distancia (n=2).

d = distancia entre el transmisor y el receptor.

I = número de categorías de paredes.

K_{wi} = número de paredes de la categoría *i*.

L_{wi} = pérdidas de la pared tipo *i*.

J = número de tipos de suelos.

K_{fj} = número de suelos de tipo *j*.

L_{fj} = pérdidas del suelo de tipo *j*.

La pérdida por suelos es una función no lineal del número de suelos atravesados. Esto se puede tener en cuenta introduciendo un factor empírico adicional. Es necesario indicar también que los factores de pérdidas de paredes y suelos no son pérdidas físicas reales, sino coeficientes del modelo, optimizados mediante procesos de medida.

Las normas *UMTS* de la *ETSI,* reconocen el modelo de *Keenan-Motley* modificado como una solución adecuada para el cálculo de la propagación en el interior de edificios, donde se incluye además del factor de pérdidas de penetración por suelos, un factor adicional de pérdidas de penetración por paredes u obstáculos. En la tabla 1.10 se puede observar algunas pérdidas que se deben considerar en el modelo de *Keenan-Motley.*

TIPO DE PÉRDIDA	DESCRIPCIÓN	FACTOR (dB)
L_f	**Suelos** • Baldosas • Revestimiento de hormigón • Espesor típico < 30 cm	18.3
L_{w1}	**Muros internos finos** • Yeso • Muros con ventanas	3.4
L_{w2}	**Muros internos** • Hormigón, ladrillos • Mínimo número de ventanas	6.9

Tabla 1.10 Factores de pérdidas según categoría [20]

La ecuación de *Keenan-Motley* para el caso de la banda de 2.4 GHz y considerando una pendiente de pérdidas n=2 es:

$$L = 40.1[dB] + 20 * \log(d) + \sum_{i=1}^{I} K_{wi} * L_{wi} + \sum_{j=1}^{J} K_{fj} * L_{fj}$$ **Ecuación 1.20** [12]

Tomando en cuenta que se usan antenas omnidireccionales con ganancia unitaria y las pérdidas de *Keenan-Motley,* la ecuación para el cálculo del enlace para Wi Fi es la siguiente:

$$P_{RX}[dBm] = P_{TX}[dBm] - \left(20 * \log(d) + \sum_{i=1}^{I} K_{wi} * L_{wi} + \sum_{j=1}^{J} K_{fj} * L_{fj} \right)[dB] - 40.1[dB]$$ **Ecuación 1.21** [12]

- **Tabla comparativa de las principales características de *Bluetooth* y *Wi-Fi***

En la tabla 1.11 se muestra la comparación de las principales características entre las tecnologías *Bluetooth* y *Wi-Fi.*

TEGNOLOGÍA	BLUETOOTH	WI-FI
Estándar	802.15.1	802.11 b
Velocidad	1 Mbps	11 Mbps
Medio	Inalámbrico	Inalámbrico
Topología	Ad-hoc e Infraestructura	Ad-hoc e Infraestructura
Modulación	GFSK	DBPSK (1 Mbps), DQPSK (2 Mbps), CCK (5.5 y 11 Mbps), DSSS
Potencia	2.5 mW	100 mW
Alcance	10 m	100 m
Banda de Frecuencia	ISM 2.4 GHz – 2.5 GHZ	ISM 2.4 GHz – 2.5 GHZ
Tipo de Enlace	Asincrónico y Sincrónico	Sincrónico
Ancho de Banda por canal	1 MHZ	22 MHz
Número de canales	79/23canales	77/23 canales
Consumo de energía	Baja (1/5 del consumo de Wi-Fi)	Alto
Seguridad	Obligatoria	Opcional

Tabla 1.11 Comparación teórica entre Bluetooth y Wi-Fi

CAPÍTULO 2

2. DISEÑO E IMPLEMENTACIÓN DE LOS PROTOTIPOS

Al momento de realizar el diseño de los prototipos inalámbricos *Bluetooth* y *Wi-Fi* se deben considerar los diferentes requerimientos para su correcto funcionamiento, entre éstos están: la cobertura, estándar, modo de operación y topología.

El área de cobertura en la cual se realizó el análisis de los prototipos es de 10 m, la frecuencia de trabajo es 2.4 *GHz ISM*, la topología utilizada es *Ad-Hoc*, el estándar para *Bluetooth* es *IEEE 802.15.1* y para *Wi-Fi IEEE 802.11 b.*

La implementación de los prototipos se la realizó en la *SUPTEL* (Superintendencia de Telecomunicaciones), debido a que en este lugar se cuenta con el equipo necesario para realizar medidas de potencia de transmisión de las interfaces utilizadas en los prototipos los cuales fueron facilitados para la realización del proyecto.

2.1 DISEÑO DE LOS PROTOTIPOS BLUETOOTH y WI-FI

Para el diseño del prototipo tanto *Bluetooth como Wi-Fi* se debe realizar un análisis del sitio en el cual se va a implementar, en base a un plano de construcción del lugar se definen las posibles ubicaciones de las estaciones dentro del área de cobertura deseada.

El principal objetivo de hacer un análisis del sitio es determinar las posibles causas y factores por las que se puede degradar el funcionamiento de los prototipos y de esta manera establecer si los interfaces que se van a utilizar están en la capacidad de satisfacer los requisitos de un enlace inalámbrico, esto es, alta capacidad, cobertura pequeña, conectividad de las estaciones, seguridad, facilidad de desplazamiento, etc.

El análisis de interferencias del sitio puede hacerse a través del *software* que viene incorporado en las NIC (Tarjeta de Interfaz de Red) inalámbricas, éstas permiten determinar el alcance y la calidad de un enlace inalámbrico, también se puede determinar la conectividad del enlace ejecutando un *ping*[5].

2.1.1 PLANO DE PLANTA ALTA DE LA *SUPTEL* (Superintendencia de Telecomunicaciones)

La figura 2.1 representa el plano arquitectónico de una de las instalaciones de la SUPTEL (Superintendencia de Telecomunicaciones), específicamente de la última planta, lugar donde se realizaron las pruebas de los prototipos.

[5] *Ping:* El comando ping es un programa básico que verifica conectividad en una red de computadoras enviando petición de envío y respuesta a través de datagramas de petición de eco.

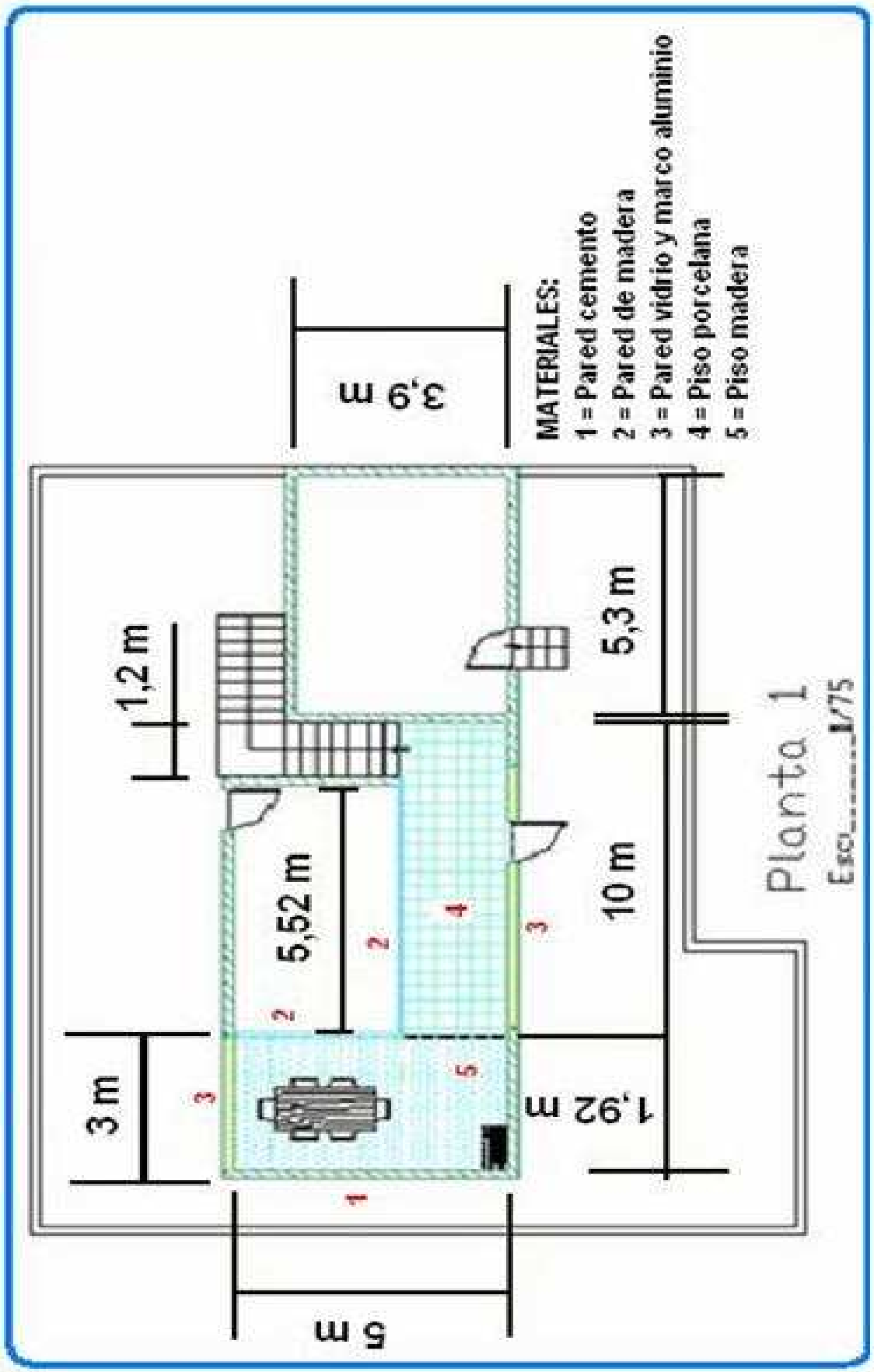


Figura 2.1 Plano de la planta alta de la SUPTEL (2D)

La figura 2.2 representa el plano arquitectónico del sitio en tres dimensiones, de lo que se puede apreciar en ésta y de la inspección que se hizo, se pudo determinar que existen paredes de distinto material los mismos que van a absorber o reflejar la señal en menor o mayor grado. También se observó que existen muebles y escritorios que van a degradar la señal así como un horno microondas que funciona en la banda de 2.4 *GHz* el mismo que cuando esté en funcionamiento podría causar interferencias.

Por último se observó que existen otras redes con tecnología *Wi-Fi* en el lugar, pertenecientes a Andinanet y el Banco del Pichincha que podrían interferir con los prototipos ya que funcionan en la misma banda de frecuencia de *2.4 GHz*.

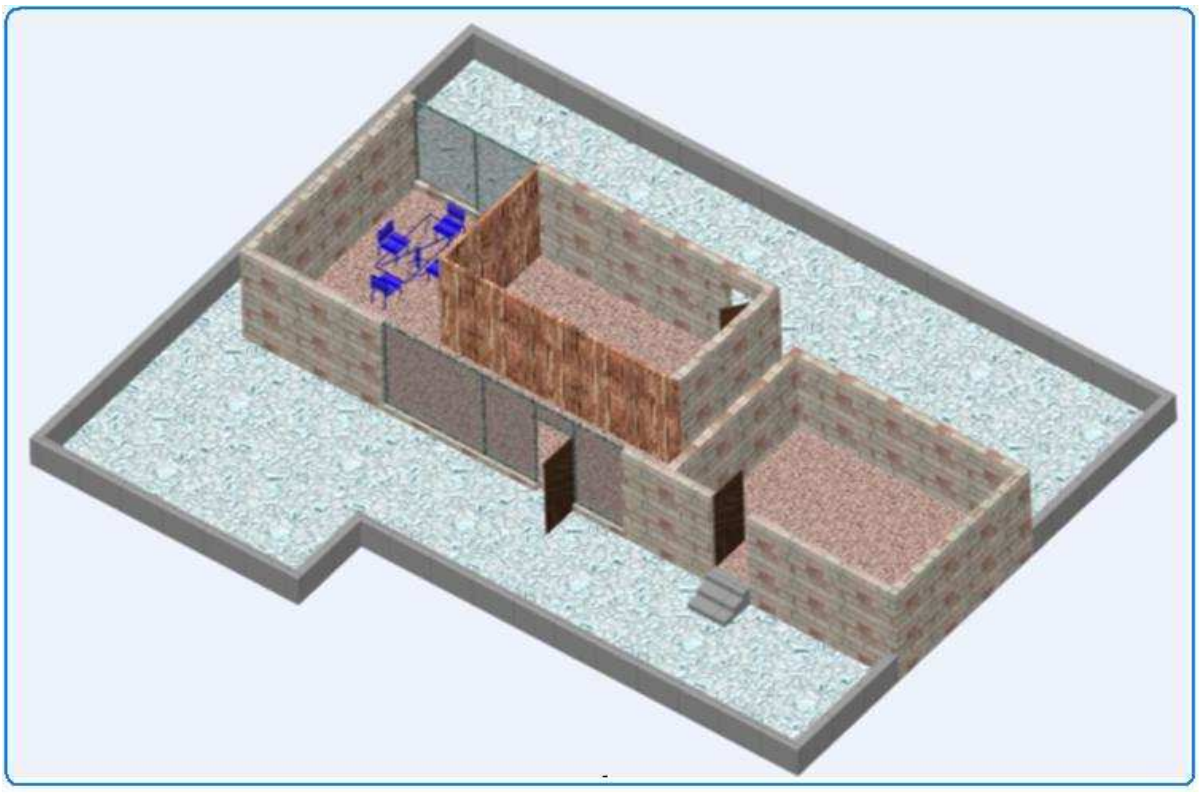


Figura 2.2 Plano Arquitectónico del lugar (3D)

2.1.2 IDENTIFICACIÓN DEL ÁREA DE COBERTURA

Como se puede apreciar en la figura 2.3 el área que se desea cubrir con los prototipos tanto *Bluetooth* como *Wi-Fi* es aproximadamente 10 metros.

Figura 2.3 Zona de Cobertura de los Prototipos

2.1.3 UBICACIÓN DE LAS ESTACIONES

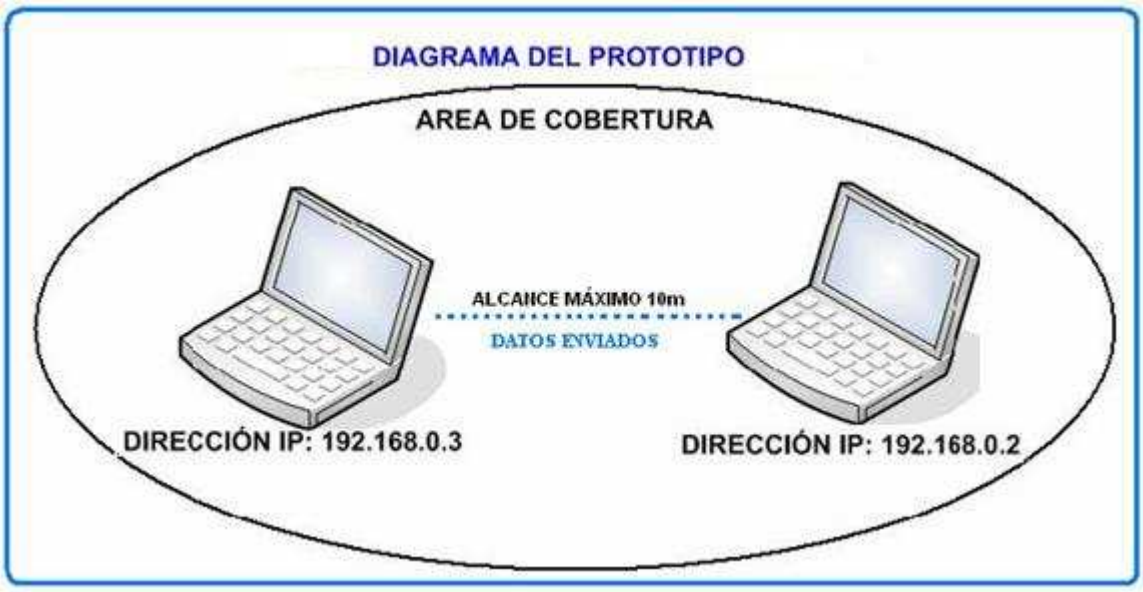


Figura 2.4 Diagrama de los Prototipos a Implementarse

La ubicación de las estaciones en una red *Ad-hoc* es esencial, ya que si existen obstáculos entre éstas la señal de transmisión podría degradarse e incluso podría perderse la comunicación entre las mismas.

Para determinar que los prototipos puedan cubrir un área determinada, se lo puede hacer en base a mediciones de campo, disponibilidad de la conexión entre los dispositivos de la red y mediante *software* que viene incorporado en las interfaces utilizadas para el prototipo *Bluetooth* y el prototipo *Wi-Fi*.

2.1.4 EQUIPOS A UTILIZARSE EN EL DISEÑO DE LOS PROTOTIPOS

Para la implementación de los prototipos inalámbricos, es necesario incorporar a las estaciones de trabajo dos adaptadores inalámbricos con tecnología *Bluetooth* para el prototipo *Bluetooth* y dos adaptadores inalámbricos con tecnología *Wi-Fi* para el prototipo *Wi-Fi*; estos dispositivos pueden ser de varios tipos y modelos, su elección dependerá de las características técnicas de cada uno, en la actualidad en el mercado existen gran variedad de adaptadores inalámbricos *PCMCIA, PCI* o *USB.*

2.1.4.1 Adaptadores Inalámbricos *Bluetooth y Wi-Fi*

En la tabla 2.1 se muestran los adaptadores con tecnología *Bluetooth* con sus principales características técnicas. Se eligió adaptadores *USB* debido a su fácil conexión.

	BELKIN. F8T001v	**D–Link** DBT–122	**ANYCOM** USB - 120
			
Especificaciones Técnicas			
Operación	Bluetooth 1.1	Bluetooth 1.2	Bluetooth 1.1
Almacenaje	Punto a Multipunto	Punto a Multipunto	Punto a Multipunto
Alcance de Cobertura			
Distancia	100 m	10 m	30 m
Potencia de Transmisión			
Potencia de transmisión	0 a 20 dBm	-6 a 4 dBm	0 a 13 dBm
Banda de Frecuencia			
Rango	2.4 – 2.483 GHz	2.4 – 2.483 GHz	2.4 – 2.483 GHz
Velocidad Máxima de transmisión de datos	56/723 Kbps asincrónico 400/420 Kbps sincrónico	723 Kbps asincrónico 433.9 Kbps sincrónico	1 Mbps

Tabla 2.1 Datos Técnicos de los Adaptadores *USB Bluetooth* [36] [37] [38]

2.1.4.2 Adaptadores Inalámbricos *Wi-Fi*

En la tabla 2.2 se puede observar las principales características técnicas de adaptadores inalámbricos *con tecnología Wi-Fi.*

	AIR802	DWL – G122	WUSB54GC
			

Especificaciones Técnicas

Operación	802.11 b/g 1.1	802.11 b/g 1.1	802.11 b 1.1
Almacenaje	Punto a Multipunto	Punto a Multipunto	Punto a Multipunto

Alcance de Cobertura

Distancia	300 m	100 m	100 m

Potencia de Transmisión

Clase 1	14 dBm ± 2dBm 802.11g 17 dBm ± 2dBm 802.11b	14 dBm ± 2dBm 802.11g	14 dBm ± 2dBm 802.11g

Banda de Frecuencia

Rango	2.4 a 2.497 GHz 2.4 a 2.4835 GHz	2.4 GHz	2.4 GHz
Velocidad Máxima de transmisión de datos	6, 9, 12, 18,24, 36, 48, 54 Mbps	54 Mbps	54 Mbps

Tabla 2.2 Datos Técnicos de los Adaptadores *USB Wi-Fi* [36] [39] [40]

2.1.4.3 Comparación de los Equipos

El análisis técnico comparativo se basa en los siguientes parámetros: potencia, velocidad, frecuencia de operación y cobertura de los dispositivos; estos parámetros permiten determinar qué dispositivo se debe utilizar de acuerdo a las necesidades del enlace inalámbrico.

2.1.4.3.1 Comparación de los Equipos Bluetooth

Como se puede observar en la tabla 2.1 todos los dispositivos trabajan en la banda de frecuencia de 2.4 *GHz,* la velocidad de los dispositivos *D-Link* y *BELKIN* es 723 kbps, *ANYCOM* presenta una velocidad de 1 *Mbps*.

El área de cobertura a cubrir con los dispositivos *Bluetooth* es 10 m, en la tabla 2.1 se puede apreciar que todos los fabricantes citados cumplen con éste requerimiento.

2.1.4.3.2 Comparación de los Equipos Wi-Fi

Como se puede observar en la tabla 2.2 todos los dispositivos *Wi-Fi* pueden cubrir el área de cobertura deseada, la banda de frecuencia de trabajo es 2.4 *GHz*, la velocidad máxima es 54 *Mbps* para estos tres fabricantes.

2.1.4.4 Costos Referenciales de los Equipos

La disponibilidad de los equipos *Bluetooth* y *Wi-FI* en el mercado no es la misma; existen marcas que se comercializan localmente en el país, y otras que no se comercializan localmente, los cuales deben ser adquiridos fuera del país.

El precio de los equipos varía de acuerdo a los fabricantes y si están disponibles a nivel local o no, a continuación se presenta en la tabla 2.3 y en la tabla 2.4 los precios de los equipos que se podrían utilizar para los prototipos *Bluetooth* y *Wi-Fi* respectivamente. Cabe mencionar que todos los costos incluyen IVA y para los equipos que no son comercializados localmente se incluye el costo de envío.

En la tabla 2.3 y 2.4 todos los precios indicados incluyen IVA.

FABRICANTE	MODELO	PRECIO UNITARIO (USD)	PRECIO DE ENVIO (USD)	COSTO POR UNIDAD (USD)
ANYCOM	USB - 120	50	0	50
BELKIN	F8T001v	50	50	100
D – LINK	DBT - 122	28.605	0	28.605

Tabla 2.3 Precios de los Equipos *Bluetooth* (fecha 25/06/2006) [36]

FABRICANTE	MODELO	PRECIO UNITARIO (USD)	PRECIO DE ENVIO (USD)	COSTO POR UNIDAD (USD)
D – LINK	DWL – G122	35.615	0	35.615
EVCOM	AIR802	38	50	88
LINKSYS	WUSB54GC	30	50	80

Tabla 2.4 Precios de los Equipos *Wi–Fi* (fecha 25/06/2006) [36]

2.1.4.5 Selección de los equipos a utilizarse en la implementación de los Prototipos

Técnicamente los equipos *Bluetooth* y *Wi-Fi* de los diferentes fabricantes tienen las mismas características aplicables al proyecto, por lo que todos ellos podrían ser de utilidad para realizar la implementación de los prototipos; la elección del equipo se basa principalmente en la disponibilidad en el mercado, facilidad de adquisición y costo.

Se ha seleccionado al equipo *DBT-122* para el prototipo *Bluetooth* y el equipo *DWL-G122 para el prototipo Wi-Fi* ambos modelos son del fabricante *D-Link,* ya que es una marca comercial en Ecuador, tiene gran disponibilidad de sus equipos y se los puede adquirir fácil e inmediatamente. Otra razón para la selección del fabricante es el costo, *D-Link* ofrece precios accesibles, y cumplen con los requerimientos de diseño.

2.1.5 CÁLCULO DEL ÁREA DE COBERTURA

Para el cálculo del área de cobertura para los prototipos *Bluetooth* y *Wi-Fi* se utilizaron dos modelos para cada prototipo, cada modelo utilizado se adapta de mejor manera para cada tecnología.

2.1.5.1 Cálculo del área de cobertura para Bluetooth

A continuación se procede a calcular el área de cobertura para el prototipo *Bluetooth.*

- **MODELO 1: Modelo que considera las Pérdidas en la Trayectoria y Desvanecimientos Multitrayectoria.**

Para el cálculo de la potencia de recepción se utiliza la ecuación 1.10

$$P_{RX}[dBm] = P_{TX}[dBm] - L_{patch}[dB] - 8[dB]$$

R = 10 m distancia de separación máxima entre estaciones
P_{TX} = - 6 dBm potencia especificada por la interfaz *DBT-122*

Cálculo de la potencia de recepción

Como se tiene una distancia mayor a 8.5 m se utiliza la ecuación 1.9 que permite calcular las pérdidas en la trayectoria L_{patch}

$$L_{patch} = 25.3 + 36\log(R)$$
$$L_{patch} = 25.3 + 36\log(10)$$
$$L_{patch} = 61.3\ [dB]$$
$$P_{RX}[dBm] = P_{TX}[dBm] - L_{patch}[dB] - 8[dB]$$
$$P_{RX} = -\ 6\ [dBm] - 61.3\ [dB] - 8\ [dB]$$
$$P_{RX} = -75.3\ [dBm]$$

- **MODELO 2: Modelo de Atenuación Lineal por Trayectoria**

Para el cálculo de la potencia de recepción se utiliza la ecuación 1.14

$$P_{RX}[dBm] = P_{TX}[dBm] - 20 * \log(d)[dB] - 0.47 * d[dB] - 40.1[dB]$$

R = 10 m distancia de separación máxima entre estaciones
P_{TX} = - 6 dBm potencia especificada por la interfaz *DBT-122*

Cálculo de la potencia de recepción

$$P_{RX} = -6[dBm] - 20 * \log(10)[dB] - 0.47 * 10[dB] - 40.1[dB]$$
$$P_{RX} = -6[dBm] - 20[dB] - 4.7[dB] - 40.1[dB]$$
$$P_{RX} = -70.8[dBm]$$

2.1.5.2 Cálculo el área de cobertura para Wi Fi

A continuación se procede a calcular el área de cobertura para el prototipo *Wi-Fi* con modelos que se adaptan de mejor manera a esta tecnología.

- **MODELO 1: Modelo de pérdida de propagación de una pendiente.**

Para el cálculo del enlace se considera un exponente de pérdidas n=4 que se establece en la tabla 1.9 y la ecuación 1.18.

$$P_{RX}[dBm] = P_{TX}[dBm] - 10 \bullet n \bullet \log(d)[dB] - 40.1[dB]$$

d = 10 m distancia de separación máxima entre estaciones
P_{TX} = 14 dBm potencia especificada por la interfaz *DWL-G122*

$$P_{RX} = 14[dBm] - 10 \bullet 4 \bullet \log(10)[dB] - 40.1[dB]$$
$$P_{RX} = -66.1\ [dBm]$$

- **MODELO 2: Pérdidas con factores de atenuación por suelo y pared**

Para el cálculo del enlace con este modelo, de la figura 2.2 correspondiente al plano arquitectónico del lugar se determina que en las peores condiciones la señal transmitida debe atravesar dos paredes, el índice de pérdidas por pared es $L_{wi} = 3.4$ que se encuentra en la tabla 1.10.

El cálculo se lo realiza en base a la ecuación 1.21.

$$P_{RX}[dBm] = P_{TX}[dBm] - \left(20 * \log(d) + \sum_{i=1}^{I} K_{wi} * L_{wi} + \sum_{j=1}^{J} K_{fj} * L_{fj} \right)[dB] - 40.1[dB]$$

d = 10 m distancia de separación máxima entre estaciones

P_{TX} = 14 dBm potencia especificada por la interfaz *DWL-G122*

$$P_{RX} = 14[dBm] - (20 * \log(10) + 2 \bullet 3.4 + 0)[dB] - 40.1[dB]$$

$$P_{RX} = -52.9[dBm]$$

2.1.5.3 Análisis de Resultados del Cálculo del Área de Cobertura

Una vez realizado el cálculo del enlace para *Bluetooth* y *Wi-Fi* en base a los dos modelos descritos para cada tecnología, se puede concluir que los dispositivos *DBT-122* elegidos para el prototipo *Bluetooth* y *DWL-G122* para el prototipo *Wi-Fi* cumplen con los requerimientos de cobertura deseada. Esto se puede observar en la tabla 2.5 y la tabla 2.6, la cual indica que la potencia de recepción calculada es mayor que la sensibilidad del receptor especificada en los dispositivos.

MODELOS BLUETOOTH	POTENCIA TX (dBm) DEL DBT -122	DISTANCIA MÁXIMA (m)	POTENCIA Rx Calculada (dBm)	Sensibilidad del DBT -122 (dBm)
MODELO 1	- 6	10	-75.3	-80
MODELO 2	- 6	10	-70.8	-80

Tabla 2.5 Cálculo del área de cobertura *Bluetooth*

MODELOS WI-FI	POTENCIA TX (dBm) DEL DWL-G122	DISTANCIA MÁXIMA (m)	POTENCIA Rx Calculada (dBm)	Sensibilidad del DWL -G122 (dBm)
MODELO 1	14	10	-66.1	-82
MODELO 2	14	10	-52.9	-82

Tabla 2.6 Cálculo del área de cobertura *Wi-Fi*

2.1.6 ESTÁNDAR DE LOS PROTOTIPOS

A pesar que los prototipos operen en la misma banda de frecuencia no significa que utilicen la misma tecnología y estándar, es por esta razón que se describe el estándar utilizado por cada prototipo.

2.1.6.1 Estándar del Prototipo Bluetooth

El estándar *IEEE 802.15.1* trabaja en la banda de frecuencia *ISM* de 2.4 *GHz*, en esta banda se puede implementar libremente cualquier red *WLAN*, la modulación utilizada por *IEEE 802.15.1* es *GFSK.*

La máxima velocidad que permite el estándar es de 723 *Kbps*, en un enlace asimétrico, el cual cumple con los requerimientos del diseño en cuanto a la transferencia de datos, lo cual es aceptable en una red inalámbrica *Ad-hoc*.

2.1.6.2 Estándar del Prototipo Wi-Fi

El estándar *IEEE 802.11b* trabaja en la banda de frecuencia *ISM* de 2.4 GHz, en la cual el prototipo *Wi-Fi* puede ser implementado libremente, la modulación utilizada por *IEEE 802.11b* es *DSSS*, la máxima velocidad teórica que permite el estándar es de 11 *Mbps.*

2.1.7 MODO DE OPERACIÓN Y TOPOLOGÍA

La topología utilizada en el diseño de los prototipos *Bluetooth* y *Wi-Fi* es la topología *Ad-Hoc*, la cual permite trabajar con sistemas inalámbricos punto a punto, estos sistemas se ajustan a los requerimientos de movilidad e independencia de las estaciones de trabajo ya que posee una arquitectura dinámica.

La ubicación de las estaciones en una red *Ad-hoc* es importante ya que al ser una red nómada estas pueden movilizarse de un lugar a otro dentro del área de

cobertura, motivo por el cual es indispensable ubicar las estaciones en sitios donde no existan obstáculos que impidan el correcto desempeño del prototipo.

- **Flexibilidad:** la flexibilidad en una red *Ad-hoc* es aceptable, ya que en este tipo de red las estaciones pueden cambiar fácilmente su topología física, es decir las estaciones no están sujetas a una posición fija.

- **Escalabilidad:** la escalabilidad en una red *Ad-hoc* para las tecnologías *Bluetooth y Wi-Fi* es aceptable, ya que permite incrementar el número de usuarios sin cambiar su topología, al incrementar el número de usuarios disminuye la velocidad de transmisión de datos, lo cual viene a ser una limitación de los prototipos.

2.1.8 SEGURIDAD

Bluetooth implementa un sistema de seguridad obligatorio, el cual se lo realiza al momento de emparejamiento de las estaciones. Mientras que *Wi-Fi* no utiliza un sistema de seguridad obligatorio motivo por el cual no se lo describe.

A continuación se explica el proceso de emparejamiento ofrecido por el interfaz *Bluetooth DBT-122* utilizado:

1. Identificar y visualizar todos los dispositivos *Bluetooth* que se encuentran en el área de cobertura del sistema, esto permite tener una lista de todos los usuarios que podrían tener acceso al sistema.

2. Para que un usuario acceda al sistema, es necesario ingresar un código *PIN* (número de identificación personal) de hasta 16 caracteres, el cual permite verificar la identidad de los usuarios que intervienen en la comunicación.

3. Autorización, aquí se permite o se niega el acceso del usuario a la red.

Luego de ejecutado este proceso se realiza la transmisión de datos. La figura 2.5 representa la solicitud del código *PIN* en la interfaz *DBT-122*

Una guía de este procedimiento se lo presenta en el ANEXO A

Figura 2.5 Ingreso del Código PIN

2.2 VERIFICACIÓN DEL FUNCIONAMIENTO DE LOS PROTOTIPOS

Para verificar el funcionamiento del los prototipos se realizó las siguientes pruebas: transmisión de datos y conectividad entre estaciones.

2.2.1 PRUEBAS DEL PROTOTIPO *BLUETOOTH*

Las pruebas que se realizaron consisten en transmitir paquetes a través del comando *ping* cada metro, desde una distancia mínima de un metro hasta una distancia máxima de 15 m. como se muestra en la figura 2.4

2.2.1.1 Conectividad de las estaciones

La verificación de conectividad entre estaciones en el prototipo *Bluetooth* se la realiza verificando los tiempos de respuesta a través del comando *ping*.

El comando *ping* entrega la siguiente información: tiempos de respuesta mínimo, máximo y promedio, y pérdida de paquetes. En la figura 2.6 se observa el funcionamiento del comando *ping* con el cual se verifica la conectividad existente entre las estaciones.

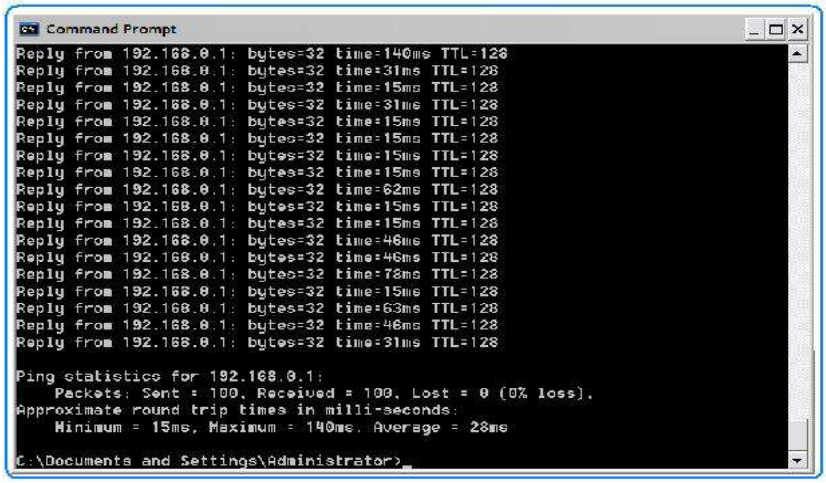


Figura 2.6 Ping entre las Estaciones del Prototipo *Bluetooth*

2.2.1.2 Estado de conexión Bluetooth

El estado de la conexión se verifica con el *software* incorporado en la interfaz *DBT-122* utilizada en el prototipo, mediante este se determina el nivel de señal recibido. En la figura 2.7 se observa que la señal recibida es buena.

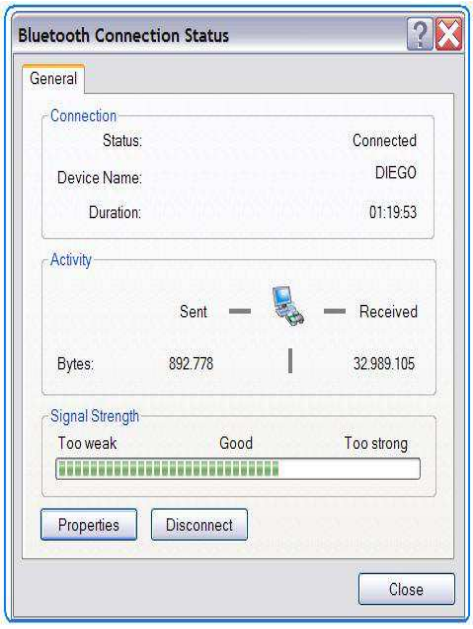


Figura 2.7 Estado de conexión

2.2.1.3 Transmisión de Datos Bluetooth

La prueba de transmisión de datos se la realizó mediante el comando *ping* entre las estaciones, el comando *ping* consiste en enviar 100 paquetes de 32 *bytes* y esperar una respuesta a esta petición.

Esta prueba se la realizó con una distancia de separación mínima entre estaciones de 1 m, hasta una distancia máxima de 15 m que supera el área cobertura de 10 m calculada en el diseño. Estas medidas se las realizaron ya que los dispositivos utilizados permitieron un alcance máximo de 15 m, esto también permitió observar la degradación del prototipo *bluetooth*.

En cada posición se realizó la toma de datos obtenidos con el comando *ping*, mediante éstos se puede comprobar si existe pérdidas de datos y los retardos entre estaciones.

Para visualizar de mejor manera los resultados de las pruebas del prototipo *Bluetooth* se las agrupó en la tabla 2.7.

Distancia (m)	Paquetes enviados	Paquetes recibidos	Paquetes perdidos (%)	Tiempo de respuesta mínimo (ms)	Tiempo de respuesta máximo (ms)	Tiempo de respuesta medio (ms)
1	100	100	0	19	120	37
2	100	100	0	19	120	39
3	100	100	0	19	130	40
4	100	100	0	19	160	38
5	100	100	0	19	120	39
6	100	100	0	19	140	39
7	100	100	0	12	140	38
8	100	100	0	19	140	40
9	100	100	0	20	140	40
10	100	100	0	19	160	38
12	100	100	0	19	130	40
15	100	100	0	19	562	55

Tabla 2.7 Resultados obtenidos en el Prototipo *Bluetooth*

El tiempo de respuesta es el tiempo que se tarda desde que la petición es enviada hasta recibir una respuesta a dicha petición, la cual se la realiza con el comando *ping*.

En la tabla 2.7 se puede observar que existen tiempos de respuesta altos, estos valores se deben a interferencias de otras redes y obstáculos existentes en el sitio donde se realizaron las pruebas, también se observa que en el escenario donde se realizaron las pruebas el prototipo *Bluetooth* no presenta pérdidas de datos

2.2.1.4 Representación Gráfica de los Resultados Obtenidos en el Prototipo Bluetooth

La figura 2.8 representa el tiempo mínimo de respuesta en función de la distancia. El tiempo mínimo de respuesta es el menor tiempo que tarda la estación en recibir una respuesta a su petición.

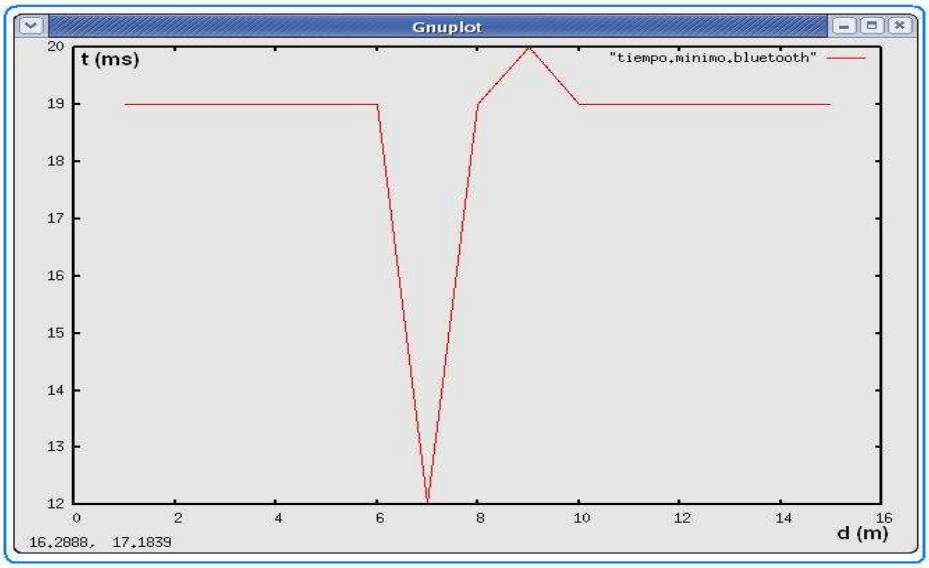


Figura 2.8 Tiempo de respuesta mínimo vs Distancia (*BLUETOOTH*)

La figura 2.9 representa la variación del tiempo de respuesta máximo en función de la distancia. El tiempo máximo de respuesta es el mayor tiempo que tarda la estación en recibir una respuesta a su petición con el comando *ping*.

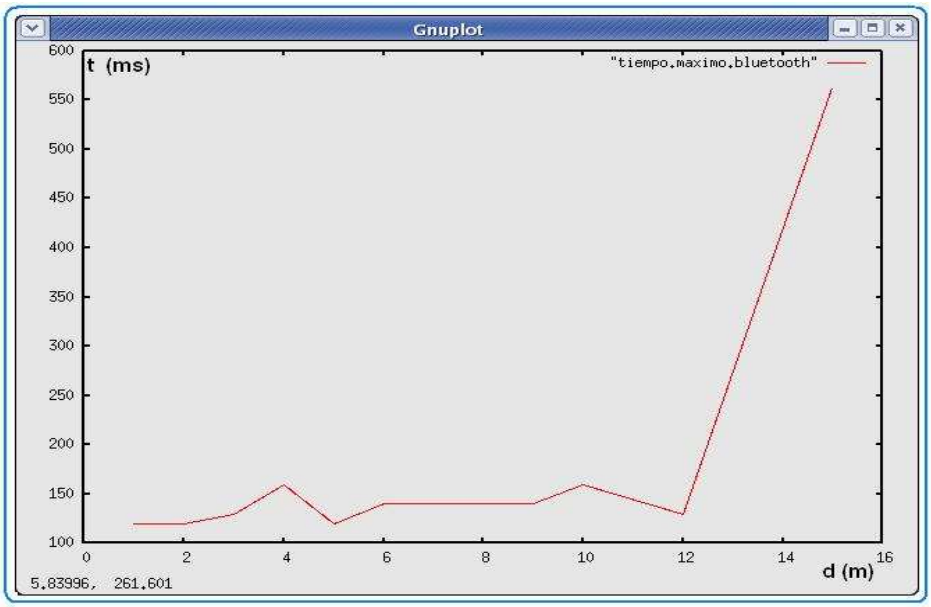


Figura 2.9 Tiempo de respuesta máximo vs Distancia (*BLUETOOTH*)

La figura 2.10 representa la variación del tiempo de respuesta medio en función de la distancia. El tiempo medio de respuesta es el promedio de todos los tiempos de respuesta que se tienen con el comando *ping*.

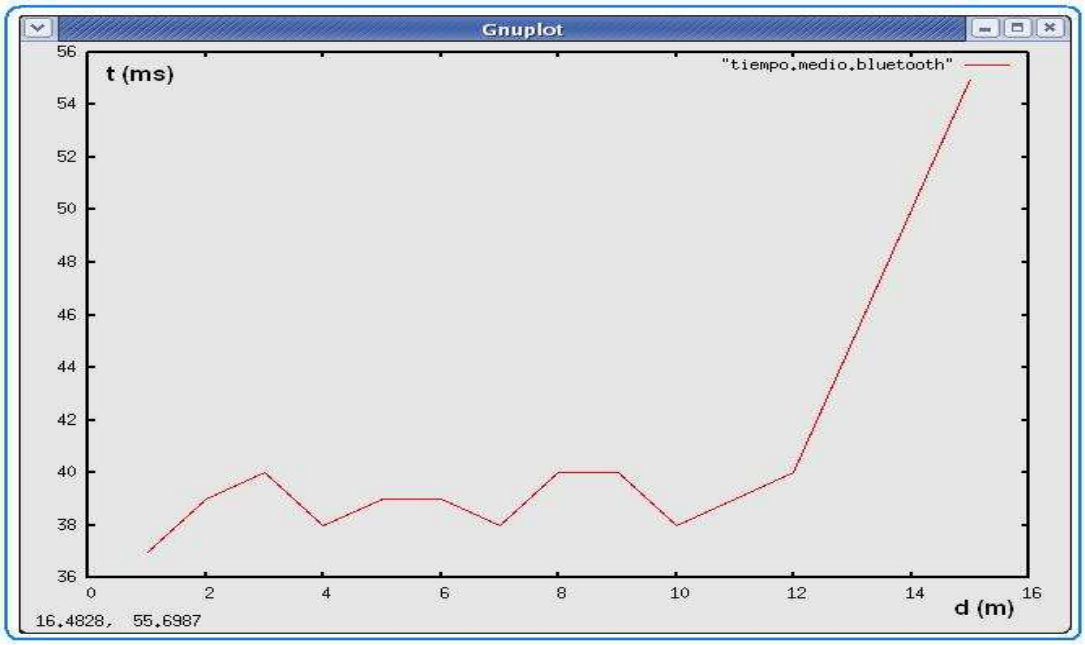


Figura 2.10 Tiempo de respuesta medio vs Distancia (*BLUETOOTH*)

De la figura 2.9 y figura 2.10 se concluye que, a medida que aumenta la distancia el tiempo de respuesta aumenta.

Mediante el *software* de la interfaz *DBT-122* se comprobó que el enlace está disponible en cada posición y también indica una velocidad de 700 *Kbps*.

2.2.2 PRUEBAS DEL PROTOTIPO WI-FI

Las pruebas que se realizaron con el prototipo *Wi-Fi* consisten en transmitir paquetes a través del comando ping cada metro, desde una distancia mínima de un metro hasta una distancia máxima de 15 m. como se muestra en la figura 2.4

2.2.2.1 Conectividad de las estaciones

La verificación de conectividad entre estaciones en el prototipo *Wi-Fi* se la realiza verificando los tiempos de respuesta a través del comando *ping.*

El comando *ping* entrega la siguiente información: tiempos de respuesta mínimo, máximo y promedio, y pérdida de paquetes. En la figura 2.11 se observa el funcionamiento del comando *ping* con el cual se verifica la conectividad existente entre las estaciones.

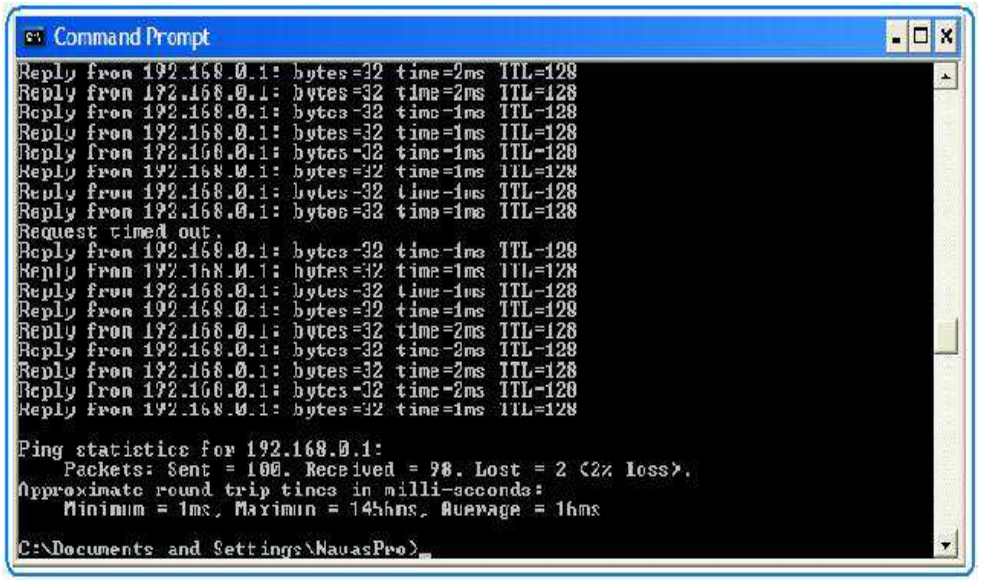


Figura 2.11 Ping entre las Estaciones del Prototipo Wi -Fi

2.2.2.2 Estado de conexión Wi-Fi

El estado de la conexión se lo verifica con el *software* incorporado en la interfaz *DWL-G122* utilizada en el prototipo, en la figura 2.12 se observa que el nivel de señal recibido es bueno.

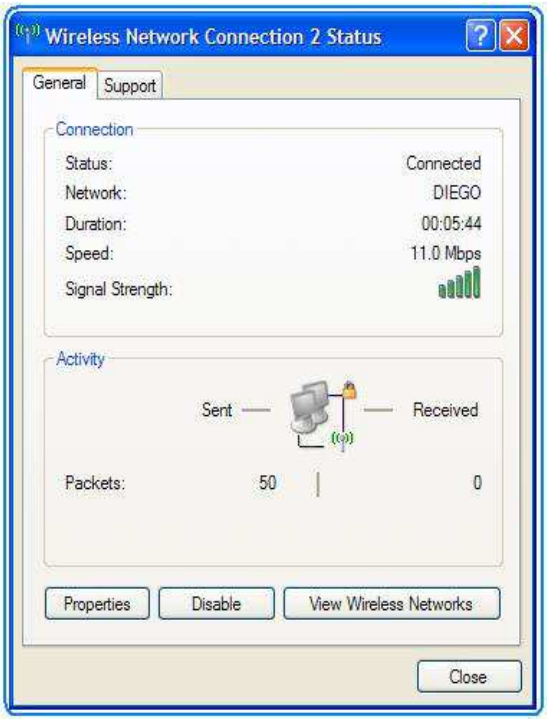


Figura 2.12 Estado de conexión *Wi-Fi*

2.2.2.3 Transmisión de Datos Wi - Fi

La prueba de transmisión de datos *Wi-Fi* se la realizó mediante el comando *ping* entre las estaciones, el comando *ping* consiste en enviar 100 paquetes de 32 *bytes* y esperar una respuesta a esta petición.

Esta prueba se la realizó con una distancia de separación mínima entre estaciones de 1 m, hasta una distancia máxima de 15 m que supera el área cobertura de 10 m calculada en el diseño. Estas medidas se las realizaron ya que los dispositivos utilizados permitieron un alcance mayor, esto también permitió observar la degradación del prototipo *Wi-Fi*.

En cada posición se realizo la toma de datos obtenidos con el comando *ping*, mediante éstos se puede comprobar si existe pérdidas de datos y los retardos entre estaciones.

Para visualizar de mejor manera los resultados de las pruebas del prototipo *Wi-Fi* se las agrupó en la tabla 2.8.

Distancia (m)	Paquetes enviados	Paquetes recibidos	Paquetes perdidos (%)	Tiempo mínimo (ms)	Tiempo máximo (ms)	Tiempo medio (ms)
1	100	99	1	1	2441	58
2	100	98	2	1	2189	23
3	100	99	1	1	2381	27
4	100	99	1	1	2650	58
5	100	97	3	1	2601	50
6	100	98	2	1	2610	54
7	100	99	1	1	2370	25
8	100	98	2	1	2359	61
9	100	96	4	1	2611	81
10	100	97	3	1	2631	53
12	100	81	19	1	3155	322
15	100	30	70	1	3878	894

Tabla 2.8 Resultados obtenidos en el Prototipo *Wi-Fi*

El tiempo de respuesta es el tiempo que se tarda desde que la petición es enviada hasta recibir una respuesta a dicha petición, la cual se la realiza con el comando *ping*.

En la tabla 2.8 se puede observar que existen tiempos de respuesta altos, estos valores se deben a interferencias de otras redes y obstáculos existentes en el sitio donde se realizaron las pruebas, también se observa que el prototipo *Wi-Fi* presenta mayor pérdida de datos a medida que las estaciones se alejan.

2.2.2.4 Representación Gráfica de los Resultados Obtenidos en el prototipo Wi-Fi

La figura 2.13 representa el tiempo mínimo de respuesta en función de la distancia. El tiempo mínimo de respuesta es el menor tiempo que tarda la estación en recibir una respuesta a su petición.

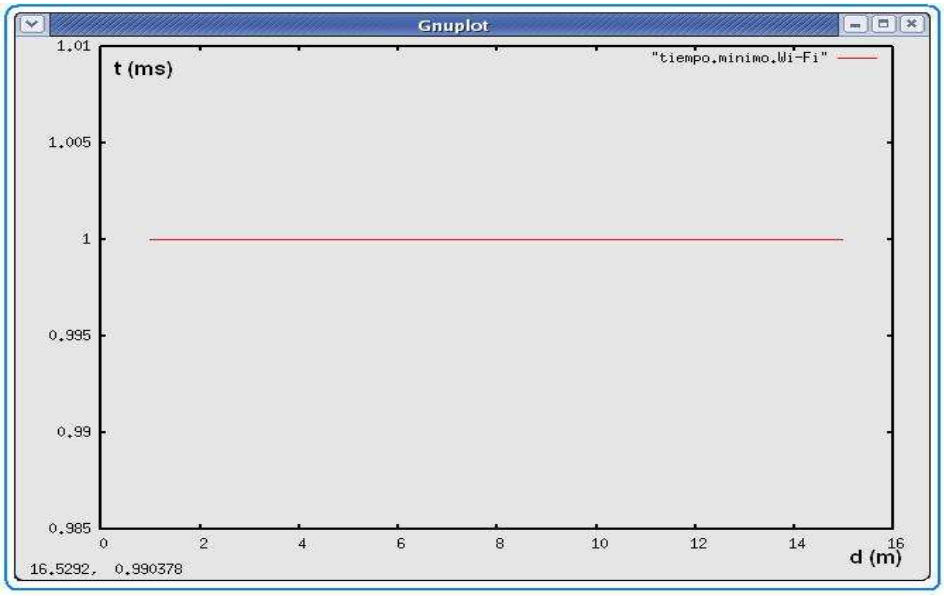


Figura 2.13 Tiempo de respuesta mínimo vs Distancia (*Wi-Fi*)

La figura 2.14 representa la variación del tiempo de respuesta máximo en función de la distancia. El tiempo máximo de respuesta es el mayor tiempo que tarda la estación en recibir una respuesta a su petición con el comando *ping*.

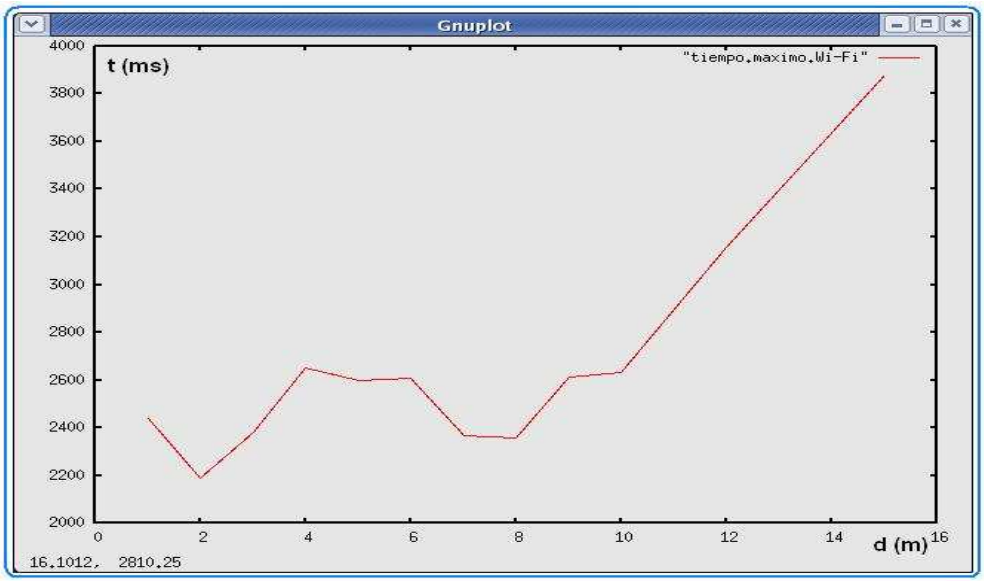


Figura 2.14 Tiempo de respuesta máximo vs Distancia (*Wi-Fi*)

La figura 2.15 representa la variación del tiempo de respuesta medio en función de la distancia. El tiempo medio de respuesta es el promedio de todos los tiempos de respuesta que se tienen con el comando *ping*.

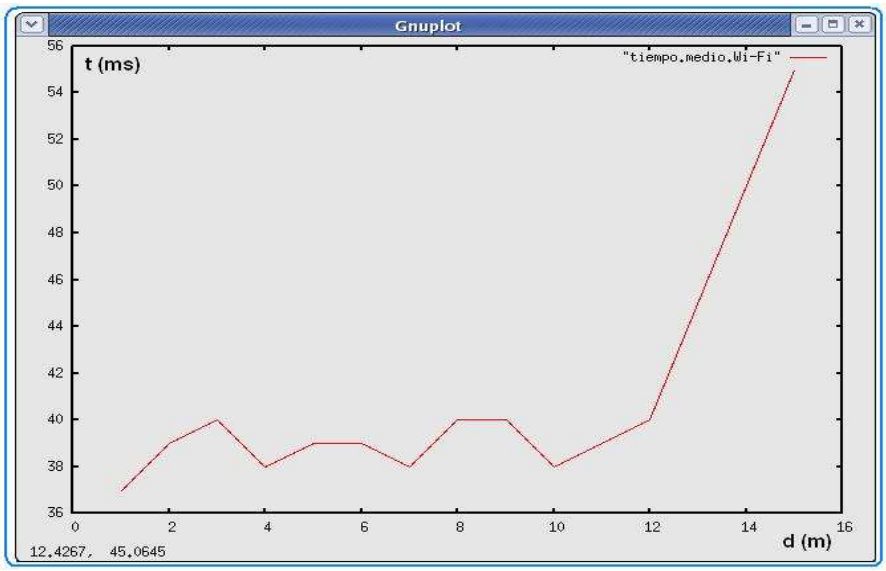


Figura 2.15 Tiempo de respuesta medio vs Distancia (*Wi-Fi*)

De la figura 2.14 y figura 2.15 se concluye que, a medida que aumenta la distancia el tiempo de respuesta aumenta.

La tabla 2.9 representa el estado de conexión del enlace *Wi-Fi* y la intensidad de potencia de recepción que indica el *software* de la interfaz *DWL-G122* utilizada en el prototipo. También este software indica una velocidad de transmisión de 11 *Mbps* para cada posición.

Distancia (m)	Estado conectado	Intensidad potencia (%)
1	SI	100
2	SI	100
3	SI	100
4	SI	100
5	SI	80
6	SI	80
7	SI	80
8	SI	60
9	SI	40
10	SI	40
12	SI	20
15	SI/NO	0-20

Tabla 2.9 Estado de conexión y velocidad de la interfaz

De las pruebas realizadas, se puede apreciar que el prototipo *Wi-Fi*, posee una mayor tasa de transferencia pero presenta una mayor pérdida de datos. Por otro lado *Bluetoot*h tiene menor velocidad de transferencia, pero para este caso el prototipo *Bluetooth* no presenta pérdida de datos.

2.3 COSTO REFERENCIAL DE LOS PROTOTIPOS

Una vez realizada la fase de diseño de los prototipos, es necesario hacer una investigación de costos referenciales para la implementación de los mismos, esta investigación servirá como un referente económico para la implementación de los prototipos.

En la fase de diseño de los prototipos se realizó previamente una comparación entre diferentes marcas de dispositivos *Bluetooth* y *Wi-Fi* disponibles en el mercado Nacional e Internacional para de esta manera conocer sus características técnicas y costos de cada uno de ellos.

2.3.1 COSTO REFERENCIAL DE EQUIPAMIENTO DEL PROTOTIPO CON TECNOLOGÍA *BLUETOOTH*

El interfaz seleccionado es el *DBT-122* (*Bluetooth* 1.2 Adaptador USB para PC) de marca *D-Link*, estos son equipos que se pueden encontrar con facilidad en el mercado nacional y en el exterior. Por lo tanto se puede obtener un amplio mercado de equipos de esta marca suficiente para tener buenas ofertas de diferentes proveedores. En este caso los precios son una medida media de los productos que existen en el mercado.

Se consideran la cantidad de clientes inalámbricos de acuerdo a lo mencionado en el diseño de los prototipos inalámbricos en el cual se consideran 2 tarjetas inalámbricas para el prototipo *Bluetooth*. A continuación se presenta en la tabla 2.10 los precios de estos productos, cabe destacar que estos precios incluyen IVA.

Equipo	Cantidad	Unidad (USD)	Precio(USD)
Tarjetas Inalámbricos USB Bluetooth	2	28.605	57.21
Subtotal			**57.21**

Tabla 2.10 Costos de equipamiento con tecnología *Bluetooth*

2.3.2 COSTO REFERENCIAL DE EQUIPAMIENTO DEL PROTOTIPO CON TECNOLOGÍA *WI-FI*

El interfaz seleccionado es el *DWL–G122* (Adaptador USB para PC) de marca *D-Link*, estos son equipos que se pueden encontrar con facilidad en el mercado nacional y en exterior. Por lo tanto se puede obtener un amplio mercado de equipos de esta marca suficiente para tener buenas ofertas de diferentes proveedores. En este caso los precios son una medida media de los productos que existen en el mercado.

Para la implementación del prototipo *Wi-Fi* es necesario utilizar dos interfaces con tecnología *Wi-Fi*, a continuación en la tabla 2.11 se presenta los precios de estos productos, cabe destacar que estos precios incluyen IVA

.

Equipo	Cantidad	Unidad (USD)	Precio(USD)
Tarjetas Inalámbricos USB Wi-Fi	2	35.615	71.23
Subtotal			**71.23**

Tabla 2.11 Costos de equipamiento con tecnología *Wi-Fi*

CAPÍTULO 3

3. SIMULACIÓN DE LOS PROTOTIPOS

Uno de los principales objetivos del proyecto, es la comparación de los resultados de la simulación de los prototipos *Bluetoot*h y *Wi-Fi* con la implementación de los mismos. Para la simulación de los prototipos se utilizó el simulador *ns-2, la versión que se utilizo es ns-2.29.3*

Antes de realizar la simulación es indispensable hacer una pequeña introducción del programa *ns-2*.

3.1 SIMULADOR ns-2 [21]

El *ns-2* es un simulador de eventos discretos creado para la investigación de redes telemáticas y esta disponible en múltiples plataformas. Probablemente *ns-2* es el simulador de redes gratuito más extendido tanto en investigación como para propósitos docentes.

Este simulador se empezó a desarrollar en 1989 como una variante del simulador *Real Network Simulator* y ha ido evolucionando substancialmente en los últimos años. En 1995, el desarrollo estaba bajo la supervisión del proyecto *VINT* (*Virtual InterNetwork Testbed*), finalmente su investigación acabó en manos de un grupo de investigadores y desarrolladores de la Universidad de California en *Berkeley*, el *LBL* (*Lawrence Berkeley Laboratory*), *XEROX Parc* y *USC/ISI* (*University of Southern California/ Information Sciences Institute*).

Entre los usos más habituales que posee *ns-2* se puede destacar los siguientes:

- Simular estructuras y protocolos de redes de todo tipo (satélite, *wireless*, cableadas, etc.)
- Desarrollar nuevos protocolos, algoritmos y comprobar su funcionamiento.

- Comparar distintos protocolos en cuanto a prestaciones.

- **UCBT - Extensión de Bluetooth para ns-2** [32]

UCBT (Extensión de Bluetooth para *ns-2* en la universidad de *Cincinnati*) es un módulo basado en *ns-2* para simular redes con tecnología *Bluetooth*. La mayoría de las especificaciones en la banda base y capas superiores como *LMP* (Protocolo de Administración del Enlace), *L2CAP* (Protocolo de Control y Adaptación de Enlace Lógico), *BNEP* (Protocolo de encapsulamiento de red bluetooth) se han simulado en *UCBT*, incluyendo el esquema de la utilización de frecuencia, descubrimiento del dispositivo, establecimiento de conexión, negociación del paquete de la multi-ranura, conexión de la voz utilizando *SCO* (Enlace sincrónico orientado a la conexión), etc.

UCBT no es el primer simulador utilizado por ns-2 para simular redes *Bluetooth.* Existen otros como *BlueHoc* creado por *IBM* y su extensión del *scatternet, Blueware*, ambos fueron desarrollados antes que el *UCBT*. Sin embargo, *UCBT* es el simulador más exacto, completo y actualizado de *Bluetooth*.

UCBT permite simular las versiones 1.1, 1.2 de *Bluetooth* cuya velocidad es menor a 1*Mbps*, también permite simular nuevas versiones como *bluetooth* 2.0 que incorpora la técnica "*Enhanced Data Rate*" (*EDR*) que permite mejorar la velocidad de transmisión hasta 2 o 3 Mbps. Una de las principales contribuciones del *UCBT* es que proporciona un marco flexible a la investigación de redes *scatternet*.

La única falencia que posee *UCBT*, es que no permite el posicionamiento ni la movilidad de los nodos, éste permite simular el funcionamiento de las capas de *Bluetooth,* esto es una desventaja con respecto a la librería *MAC/802.11* en el ns-2 ya que ésta, a más de simular la transmisión de datos permite dar movimiento a los nodos.

3.1.1 INSTALACIÓN DE LINUX

El *software ns-2* utilizado en la simulación se puede ejecutar en cualquier versión del sistema operativo *Linux*. Para el presente proyecto se utilizó la versión *Linux Centos-4i386*.

Es importante mencionar que para una correcta instalación del *ns-2* se requiere que *Linux* sea instalado con la opción *EVERYTHING* (total), de esta manera se copian las librerías necesarias para el correcto desempeño de *ns-2*, caso contrario, se necesitará un conocimiento avanzado acerca de la programación en *Linux*, para obtener dichas librerías y adjuntarlas en el entorno de trabajo.

Con la opción de instalación *EVERYTHING*, es conveniente disponer de características mínimas presentes en una PC, como por ejemplo 5 *Gbytes* de espacio libre en disco, memoria *RAM* de 256 *Mbytes* y un procesador *Pentium III*.

3.1.2 INSTALACIÓN DEL SIMULADOR Y LIBRERÍA UCBT

Para instalar el simulador *ns-2* conjuntamente con la librería *UCBT*, es necesario obtener el paquete "todo en uno del *ns-2*" que contiene todas las librerías, también es necesario obtener la librería *UCBT*. La versión del *ns-2* utilizada en el proyecto es ***ns-2.29.3***, para obtener este paquete hay que dirigirse a la siguiente página de *Internet* http://www.isi.edu/nsnam/. La versión del *UCBT* utilizada es ***UCBT 0.9.9.2*** para obtener esta librería hay que dirigirse a la siguiente página de *Internet* http://www.ececs.uc.edu/~cdmc/ucbt/.

Una vez obtenidos los paquetes *ns-2* y *UCBT*, lo primero que se debe hacer es descomprimir el *ns–2* en el directorio que se desee instalar, en este caso se escogió el directorio raíz dentro de la carpeta simulador de la siguiente forma:

```
#mkdir / simulador
#cd / simulador
# tar zxvf ns-allinone-2.29.3.tar.gz
```

Luego de descomprimir el paquete se crea el directorio *ns-allinone-2.29,* aquí se encuentra el directorio *ns-2.29*, dentro de éste se debe descomprimir la librería *UCBT* de la siguiente forma:

```
#cd / simulador
#ls
ns-allinone-2.29
#cd / simulador  / ns-allinone-2.29  / ns-2.29
# tar zxvf ucbt - 0.9.9.2 .gz
```

Una vez descomprimido el *UCBT* se debe ingresar a este directorio y ejecutar el siguiente comando:

```
#cd / simulador / ns-allinone-2.29 / ns-2.29 /ucbt – 0.9.9.2
./install-bt
```

Para utilizar las herramientas del simulador *ns-2* es necesario agregar un *PATH* permanente en el archivo */etc/profile* de la siguiente forma:

```
export PATH=”$PATH:/ simulador/ns-allinone-2.29/bin:/simulador/ns-allinone-2.29/tcl8.4.11/unix:/simulador/ns-allinone-2.29/tk8.4.11/unix
```

Además se debe realizar los siguientes enlaces simbólicos, a través de estos enlaces se crean puentes hacia los directorios de origen, permitiendo ejecutar estos programas desde cualquier parte:

```
cd /usr/bin
ln -s  /simulador/ns-allinone-2.29./ns-2.29/ns ns
ln –s  /simulador/ns-allinone-2.29/nam-1.11/nam nam
ln –s  /simulador/ns-allinone-2.29/xgraph-12.1/xgraph xgraph
```

Una vez instalado el simulador *ns-2* con la librería *UCBT*, éste debe ser validado de la siguiente forma:

```
./validate
```

Este proceso inicia una serie de pruebas propias de *ns-2* que toman varios minutos y que comprueban la correcta instalación del *software*. Ya validado el *ns-2*, se puede iniciar la implementación de las simulaciones.

3.2 SIMULACIÓN DE LOS PROTOTIPOS

Las simulaciones de los prototipos se basan en una red *Ad-hoc* punto a punto de corto alcance con tecnología inalámbrica. Los resultados se enfocan a determinar la velocidad efectiva, la relación señal a ruido y niveles de potencia en función de la distancia de cada uno de los prototipos.

La versión utilizada del ns-2 es *ns 2.29.3* se puede destacar que el uso de este simulador no ha sido sencillo debido a la dificultad de manejo, respecto a instalación, simulación de redes *wireless* e incompatibilidad de módulos y versiones.

Concretamente la simulación de redes *wireless* en *ns-2* es complicada debido a que existe poca documentación sobre el tema.

3.2.1 ESCENARIOS

Las simulaciones se han dividido en 2 escenarios diferentes:

- El primer escenario consta de dos nodos inalámbricos con tecnología *Bluetooth* y un enlace punto a punto unidireccional (figura. 3.1); en este escenario se realizarán diversas pruebas en relación a la capacidad del enlace como son: velocidad efectiva, niveles de potencia y la relación señal a ruido.

- El segundo escenario consta de dos nodos inalámbricos con tecnología *Wi-Fi* y un enlace punto a punto unidireccional (figura 3.8); en este escenario se realizarán diversas pruebas en relación a la capacidad del enlace como son: velocidad efectiva, niveles de potencia y la relación señal a ruido.

3.2.1.1 Escenario Bluetooth

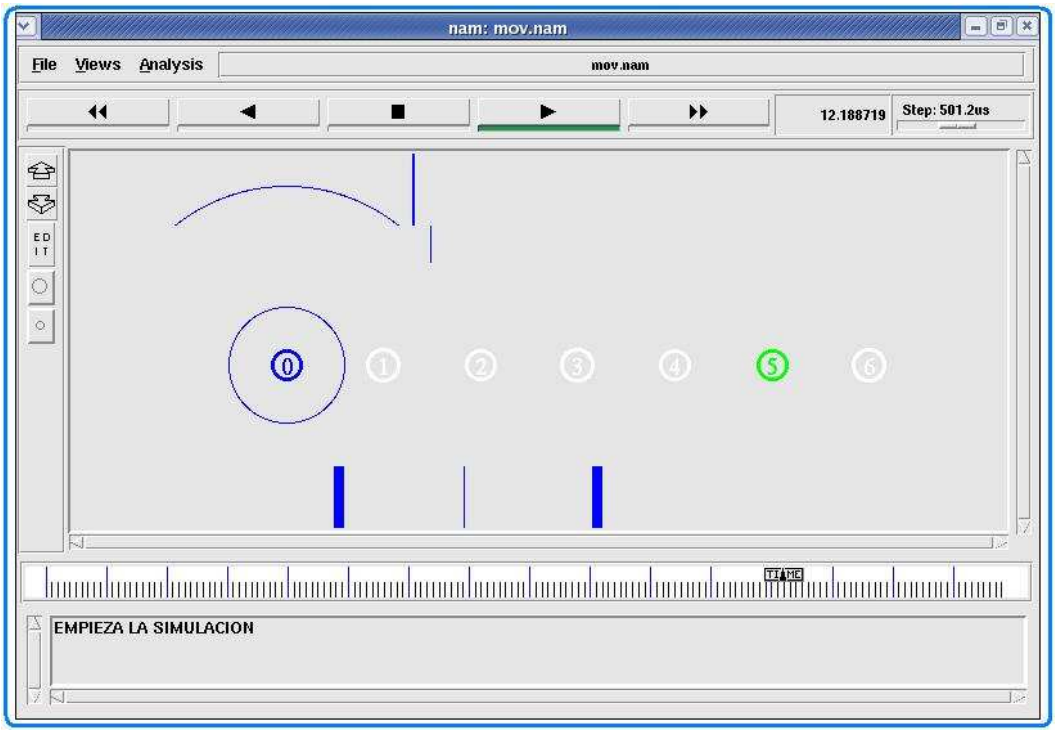


Figura 3.1 Escenario *Bluetooth*

3.2.1.1.1 Simulación Bluetooth

Para tomar las mediciones de campo cada dos metros en la simulación, se creó siete nodos inalámbricos con tecnología *Bluetooth,* la comunicación entre estos nodos es unidireccional (enlace *simplex*).

El motivo de crear siete nodos es emular el movimiento de estos, los nodos se van activando cada dos segundos, dando así un efecto de movilidad. Todo esto se realizó debido a que la librería *UCBT* no permite la movilidad y posicionamiento de los nodos. Para posicionar los nodos cada dos metros se modificó el archivo *wnode.cc.*

En la simulación se puede verificar la variación de la velocidad efectiva, la potencia y la relación señal a ruido en función de la distancia. El nodo *cero* recibirá paquetes *TCP* con tráfico *FTP* del nodo *uno*, luego del nodo *dos* y así respectivamente hasta el nodo *seis*.

El *script* que se utilizó en la simulación es fruto de varios *scripts* de prueba que se han ido utilizando a lo largo de todo el proyecto. Este código se ha modificado para adecuarlo a las necesidades de este proyecto.

A continuación se muestra el *script* utilizado para la simulación, en el cual se realiza un comentario de los aspectos más importantes.

- **SCRIPT BLUETOOTH**

Los siguientes comandos permiten almacenar en la variable *v*, la velocidad con la cuál se quiere realizar la simulación.

```
=========================================================================
#Las siguientes líneas permiten mostrar una ayuda al usuario en el caso
de que no sepa como utilizar el script
=========================================================================

if {$argc != 1} {
        puts ""
        puts "ERROR"
        puts ""
        puts "UTILICE EL SCRIPT DE LA SIGUIENTE FORMA"
        puts ""
        puts "ns bluetooth1.tcl velocidad "
        puts ""
        puts "VELOCIDAD:DH1,DH3,DH5,DM1,DM3,DM5"
        puts ""

        exit 1

} else {
        set v [lindex $argv 0]

}
```

Definición de las variables que se utilizan en la simulación como: la capa *MAC* que en este caso es *BNEP* para *Bluetooth* y el número de nodos que intervienen en la simulación.

```
=========================================================================
# Declaración de Variables a Utilizar para la presente Simulación
=========================================================================

set val(mac) Mac/BNEP    ;#Variable que identifica el tipo de nodo (BT)
set val(nn)    7         ;#Variable que especifica el número de nodos BT
```

Se crea una instancia a la clase simulador, para que se pueda realizar la simulación.

```
#=====================================================================
#Declaraciones Típicas de NS2 y NAM
#=====================================================================

set ns  [new Simulator]    ;#Definición de ns como nueva instancia de NS2
```

Se configura el nodo con el tipo de MAC definido anteriormente

```
$ns node-config -macType $val(mac) ;#Definición de nodos a usarse q sean
                                     BT
```

Se crean en modo escritura los ficheros que se van a utilizar:

- **mov.tr:** archivo de texto donde se almacenan las trazas generadas en la simulación. Se puede observar la evolución del envío de cada trama. Con este archivo y el archivo *efectiva.pl* se logra generar otro archivo que mediante el *xgraph* permite visualizar como varia la velocidad en función de la distancia.

- **mov.nam:** archivo de texto donde se almacenan las trazas *nam*, que permiten visualizar el escenario de simulación.

- **potencia.tr:** archivo de texto que se va a generar en la simulación y que mediante el *xgraph* permiten visualizar como varía la potencia en función de la distancia.

- **señalruido.tr:** archivo de texto que se va a generar en la simulación y que mediante el *xgraph* permite visualizar como varía la relación señal a ruido en función de la distancia.

```
set tracefile [open mov.tr w]       ;#Definición y creación de un archivo
                                    de trazas

set namfile   [open mov.nam w]      ;#Definición y creación de un archivo
                                    NAM (ambiente grafico)
set pot       [open ./potencia.tr w]
set SN        [open ./señalruido.tr w]
```

Estos comandos permiten generar las trazas adecuadas tanto en el archivo *mov.tr* y en el archivo *mov.nam* y habilitar su uso en el *nam*.

```
$ns trace-all $tracefile                    ;#Declaración del formato del
                                              archivo de trazas

$ns namtrace-all-wireless $namfile 7 7      ;#Declaración del tipo de
                                              ambiente y sus dimensiones

Simulator set MacTrace_ ON                  ;#Comando de habilitación para
                                              uso del NAM

Simulator set RouterTrace_ ON               ;#Comando de habilitación para
                                              el uso wireless en el NAM


$ns color 0 blue
```

Las siguientes líneas permiten crear los siete nodos, asignarles el protocolo de *routing AODV,* se utilizó este protocolo por ser uno de los más utilizados en este tipo de simulaciones; se activan los nodos a los cero segundos, se establece el tamaño de la cola en 30 paquetes para *L2CAP* y por último se asignan valores típicos de *INQUIRY* y *PAGE* a todos los nodos excepto al nodo 0.

El protocolo de enrutamiento *AODV* es un protocolo que no requiere que los nodos mantengan las rutas de los destinos que no tienen una conexión activa. Este protocolo usa números de secuencia de destino para cada ruta de entrada, el número de secuencia del destino es creado por el destino y este es enviado por cualquier ruta de información. Los números de secuencia nos permiten determinar cuál de las diversas rutas es la más accesible.

```
# ======================================================================
# Proceso de creación de Nodos Bluetooth
# ======================================================================

for {set i 0} {$i < $val(nn) } {incr i} {
    set node($i) [$ns node $i ]
  $node($i) rt AODV                 ;#se asigna el agente de ruteo AODV
  $ns at 0.0 "$node($i) on"         ;#se encienden todos los nodos al
                                      mismo tiempo en 0.0seg

  [$node($i) set l2cap_] set ifq_limit_ 30    ;#se establece el tamaño de
                                               la cola para la capa L2CAP

   if {$i > 0} {
      $node($i) inqscan 4096 2048    ;#se asignan los valores típicos de
                                      inquiry
```

```
      $node($i) pagescan 4096 2048  ;#se asignan los valores típicos de
                                      page

    }
}
```

Estas líneas asignan al nodo 0 el tipo de modelo inalámbrico adecuado para *Bluetooth,* la visualización de los paquetes *MAC* y se configura al Protocolo de Administración de Enlace para que el nodo cero realice una sola vez el *INQUIRY*

```
$node(0) LossMod BlueHoc         ;#se asigna el tipo de modelo
                                   Inalámbrico con o sin pérdidas

$node(0) trace-all-in-air on    ;#se asigna la visualización de tipos de
                                   paquetes MAC

[$node(0) set lmp_] set scan_after_inq_ 0    ;#se configura al LMP con
                                               comandos para q solo
                                               realice una vez inquiry
```

El procedimiento de configuración del enlace, tráfico y aplicaciones se lo hace para los nodos 1, 2, 3, 4, 5, 6 con el nodo cero. A continuación se explica el procedimiento de configuración del enlace entre el nodo “0” con el nodo “1”

Se crea un agente tipo *TCP* para el nodo 1, este nodo será el encargado de generar tráfico tcp y enviarlo al destino nodo 0.

```
=========================================================================
# Configuración de enlaces, trafico y aplicaciones                *
=========================================================================

set tcp0 [new Agent/TCP]      ;#Declaración de un agente de trafico TCP

$ns attach-agent $node(1) $tcp0    #Unión del agente con el nodo
                                     correspondiente (tx)
```

Comandos que permiten generar tráfico *FTP* sobre la conexión *TCP* que ya fue creada.

```
set ftp0 [new Application/FTP]  ;#Declaración de una aplicación soportada
                                   por el agente de trafico TCP

$ftp0 attach-agent $tcp0      ;#unión de la aplicación al agente de trafico
```

Se define la conducta del nodo destino y se le asigna a la fuente llamada *sink*. Este nodo destino es el encargado de generar *acks* (acuses de recibo) que garantizan el arribo de todos los paquetes al nodo 0.

```
set null0 [new Agent/TCPSink]   ;#Declaración del repositorio del agente
                                  de trafico TCP

$ns attach-agent $node(0) $null0  ;#Unión del repositorio con el nodo
                                   correspondiente (rx)
```

Se realiza la conexión entre el nodo 0 y el nodo 1.

```
$ns connect $tcp0 $null0     #unión del agente de trafico con el
                              repositorio
```

Estos comandos establecen un tamaño de 20 paquetes en el buffer esto indica que si el limite de paquetes es sobrepasado los paquetes serán descartados

```
set ifq [new Queue/DropTail]            ;#Declaración de la cola o Buffer
$ifq set limit_ 20                      ;#Límite de la cola (paquetes)
```

Variables creadas para iniciar la transferencia de tráfico *FTP* que son usadas más adelante para generar un efecto de movilidad

```
# Variables creadas para iniciar y terminar la transferencia de paquetes
de las aplicaciones

set nscmd0 "$ftp0 start"
set nscmd1 "$ftp1 start"
set nscmd2 "$ftp2 start"
set nscmd3 "$ftp3 start"
set nscmd4 "$ftp4 start"
set nscmd5 "$ftp5 start"
```

Variables creadas para finalizar la transferencia de tráfico *FTP* que son usadas más adelante para generar un efecto de movilidad.

```
set nscmd00 "$ftp0 stop"
set nscmd01 "$ftp1 stop"
set nscmd02 "$ftp2 stop"
set nscmd03 "$ftp3 stop"
set nscmd04 "$ftp4 stop"
```

Esta línea permite al tiempo 0.000001 visualizar en el *nam* el mensaje “EMPIEZA LA SIMULACIÓN”

```
#=========================================================================
# Organizador de Eventos                                                 *
#=========================================================================

$ns at 0.000001 "$ns trace-annotate \" EMPIEZA LA SIMULACION \""
```

Estas líneas permiten en un tiempo dado iniciar la conexión entre el nodo 0 y los otros nodos indicando el tipo de paquetes que envía y paquetes que recibe, estos paquetes serán ingresados por el usuario que pueden ser: *DH5*, *DH3*, *DH1*, *DM5*, *DM3* y *DM1*, las velocidades que se alcanzan cuando se transmiten estos paquetes se muestran en la tabla 1.4; también se establece el tamaño de la cola en el *buffer* que fue definido anteriormente. El tiempo en que se establece la conexión de todos los nodos depende del número de nodos que van a establecer una conexión, para este caso es de 4 segundos.

```
$ns at 0.1 "$node(0) make-bnep-connection $node(1) $v $v  noqos $ifq"
$ns at 0.2 "$node(0) make-bnep-connection $node(2) $v $v noqos $ifq1"
$ns at 0.3 "$node(0) make-bnep-connection $node(3) $v $v noqos $ifq2"
$ns at 0.4 "$node(0) make-bnep-connection $node(4) $v $v noqos $ifq3"
$ns at 0.5 "$node(0) make-bnep-connection $node(5) $v $v noqos $ifq4"
```

Esta línea de comando permite determinar el alcance máximo del simulador para bluetooth en el caso de que se la active no se podrá realizar la conexión entre los nodos y por ende no se inicia la transmisión *FTP*.

```
#$ns at 0.6 "$node(0) make-bnep-connection $node(6) DH5 DH5 noqos $ifq5"
```

Estas líneas permiten al simulador en un tiempo dado iniciar y terminar la transferencia de datos dando un efecto de movilidad entre el nodo cero y los demás.

```
$ns at 4.0 "$nscmd0"
$ns at 6.0 "$nscmd00"

$ns at 6.1 "$nscmd1"
$ns at 8.0 "$nscmd01"

$ns at 8.1 "$nscmd2"
$ns at 10.0 "$nscmd02"

$ns at 10.1 "$nscmd3"
$ns at 12.0 "$nscmd03"

$ns at 12.1 "$nscmd4"
$ns at 13.5 "$nscmd04"
```

Al tiempo t=4 segundos se llama a la función *record,* esta función es la encargada del cálculo del enlace.

```
=========================================================================
# Procedimiento para llamar a la función record la cual se encargara del
cálculo  de la potencia y de la relación señal a ruido
=========================================================================

#A los 4.0 segundos llamo a la función record
$ns at 4.0 "record"
```

Se define dos variables locales para la función *record* la una llamada *pot* y la otra llamada *SN*

```
proc record {} {
        global sink pot
        global sink SN
```

Indica la granularidad de 2 segundos y se almacena en la variable *time*

```
        set ns [Simulator instance]
        set time 2.0
```

Este comando permite determinar en que tiempo se encuentra la simulación

```
#Calculo de la distancia
        set now [$ns now]
```

Se calcula la distancia cada 2 segundos debido a que es el tiempo en el cual el nodo va a desplazarse de una posición a otra.

```
        set distancia [expr $now*1.0 - 2.0 ]
```

Se realiza el cálculo de las pérdidas en la trayectoria y el cálculo de la potencia para *Bluetooth* de acuerdo a las siguientes ecuaciones: ecuación 1.8 y ecuación 1.10. Para el cálculo de la potencia de transmisión se toma el valor de 4dBm que es la especificada para la interfaz *DBT -122.*

```
#Comparo la distancia de acuerdo a una referencia para el cálculo de la
potencia.
        if {$distancia <= 8.5} {

#Calculo de las pérdidas en dB para una distancia menor a 8.5 m

                set pérdidas [ expr 40.0 + 20.0*log10($distancia)]

#Cálculo de la potencia
           set potencia [expr 4-pérdidas-8.0]
```

Con la potencia calculada para una distancia menor a 8.5m se calcula la relación señal a ruido. El ruido se cálculo en base a las ecuaciones que se encuentran en el ANEXO F:

Para una temperatura de 27℃ y un ancho de banda de 1 *MHz* correspondiente a *Bluetooth* y transformando este valor a decibelios se obtuvo un ruido de -154.28 dB.

```
set sn [expr $potencia+154.28]
```

Se realiza el cálculo de las pérdidas en la trayectoria y el cálculo de la potencia para *Bluetooth* de acuerdo a las siguientes ecuaciones: ecuación 1.9 y ecuación 1.10. Para el cálculo de la potencia de transmisión se toma el valor de 4dBm que es la especificada para la interfaz *DBT -122.*

```
} else {
        set perdidas [ expr 25.3+36*log10($distancia)]
        set potencia [expr 4-$perdidas-8.0]
```

Con la potencia calculada para una distancia mayor a 8.5m se prosigue a calcular la relación señal a ruido como se hizo anteriormente.

```
        set sn [expr $potencia+154.28]
        }
```

Se imprime en el archivo *potencia.tr* la distancia y la potencia. En el archivo *señallruido.tr* se imprime la distancia y la relación señal a ruido.

```
puts $pot "$distancia  $potencia    "
puts $SN "$distancia  $sn "
```

Se llama a la función *record* cada dos segundos, para que realice el cálculo de los parámetros requeridos, esto se debe ya que cada dos segundos se activa el siguiente nodo que se encuentra en una posición diferente.

```
$ns at [expr $now+$time] "record"
}
```

Se indica el tiempo en el cual la simulación finaliza.

```
$ns at 15.9 "finish"
```

Se llama a la función *finish* que permite realizar todos los procesos para finalizar la simulación y permite visualizar con el *xgraph* los resultados obtenidos en la misma.

Para el cálculo de la velocidad efectiva se ejecuta el comando *perl* que conjuntamente con el archivo *efectiva.pl*, con el archivo *mov.tr*, indicando el nodo donde se necesita analizar el resultado y la granularidad generan el archivo *velo.tr* en el cual se almacena la velocidad efectiva de *Bluetooth* en función de la distancia. El archivo *efectiva.pl* se encuentra en el ANEXO I

```
#=========================================================================
# Procedimiento Final                                                    *
#=========================================================================

proc finish {} {
        global node
        $node(0) print-all-stat

        exec nam mov.nam &

        exec perl efectiva.pl mov.tr _0_ 0.1 > velo.tr & \

        exec xgraph velo.tr -t "VELOCIDAD BLUETOOTH vs DISTANCIA" -x
                                "DISTANCIA m" -y "VELOCIDAD bps" &
      exec xgraph potencia.tr -geometry "750x500" -P -t
" POTENCIA BLUETOOTH vs DISTANCIA" -x "DISTANCIA m" -y "POTENCIA dBm" &

        exec xgraph señalruido.tr -geometry "750x500" -P -t "S/N
BLUETOOTH vs DISTANCIA" -x "DISTANCIA m" -y "S/N dBm" &

        exit 0
}

$ns run
```

3.2.1.1.2 Ejemplo de Simulación del Prototipo Bluetooth

Para realizar la simulación el usuario debe ingresar al directorio en el cual se encuentran ubicados los s*cripts bluetooth.tcl* y *efectiva.pl* en este caso los *scripts* se encuentran en el directorio *Bluetooth*

Para acceder al directorio el usuario debe ingresar los siguientes comandos en el terminal de *Linux.*

cd /root/Desktop/tesis/Bluetooth/

Para obtener ayuda del uso del *script* ingresar el siguiente comando.

ns bluetooth.tcl

Después de ejecutar el comando *ns bluetooth.tcl* se despliega la siguiente pantalla, la cual indica la forma de utilizar el *script* para iniciar la simulación.

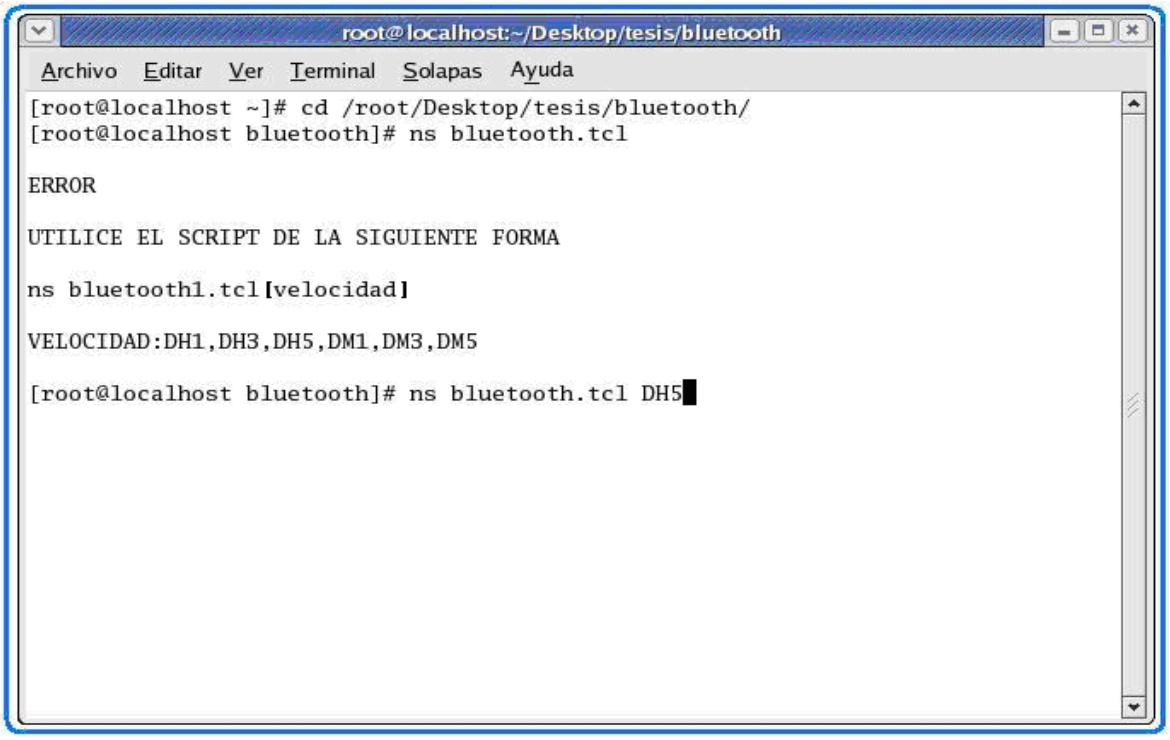


Figura 3.2 Ayuda para la simulación *Bluetooth*

Aquí el usuario puede escoger la velocidad para realizar la simulación. En la figura 3.2 la simulación se realizó con una velocidad *DH5*.

DH5 indica que se utiliza para la transmisión paquetes *DH5* que se transmiten a una velocidad de 721 Kbps, los paquetes y sus velocidades se los pueden visualizar en la tabla 1.4.

Una vez que se ejecute el programa se visualizará el escenario en el *nam* y los resultados que se obtienen de la simulación en el *xgraph* como son: potencia, relación señal a ruido y la velocidad efectiva.

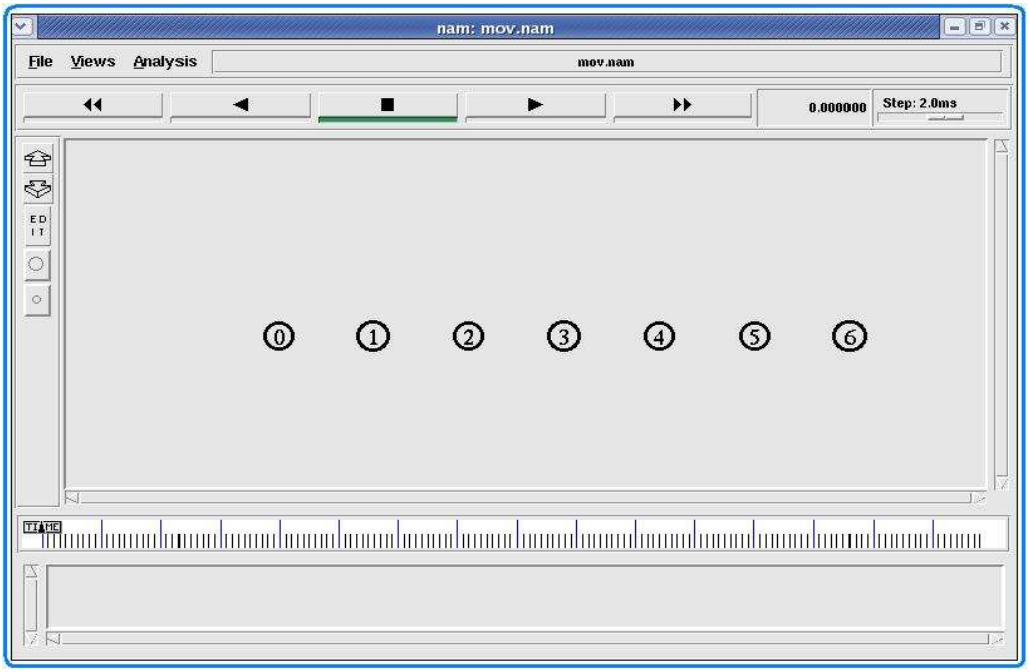


Figura 3.3 Pantalla inicial del nam *Bluetooth*

Luego de iniciar la simulación en el *nam* se visualiza como los paquetes son enviados.

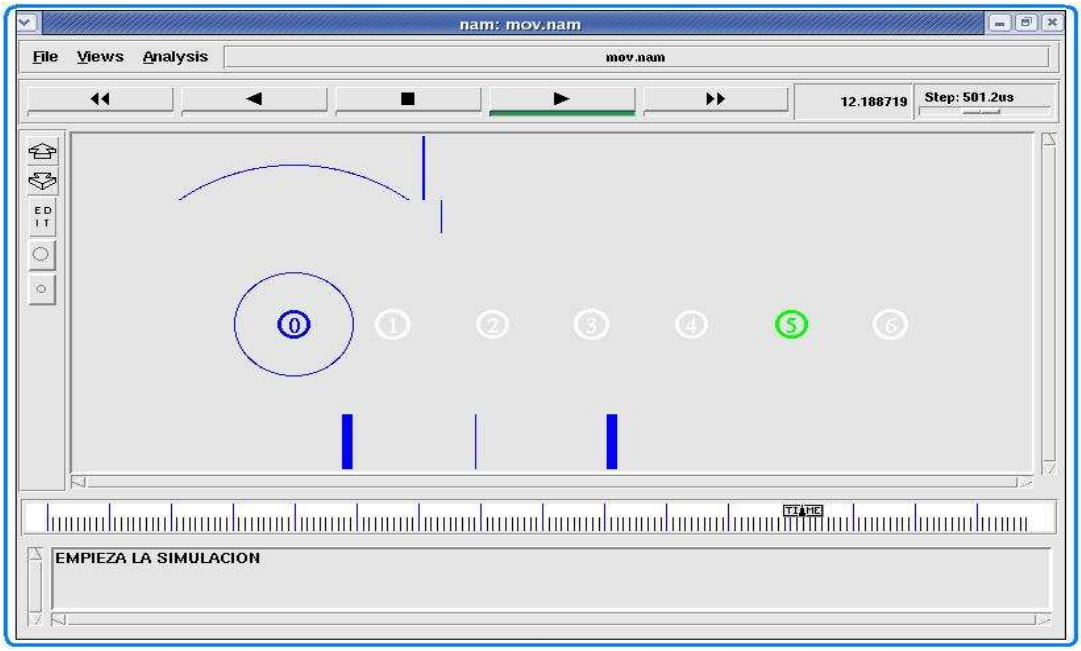


Figura 3.4 Simulación *Bluetooth* en el *nam*

Con la ayuda del *XGraph* y el archivo *potencia.tr* que se genera en la simulación, visualizamos como la potencia cambia en función de la distancia.

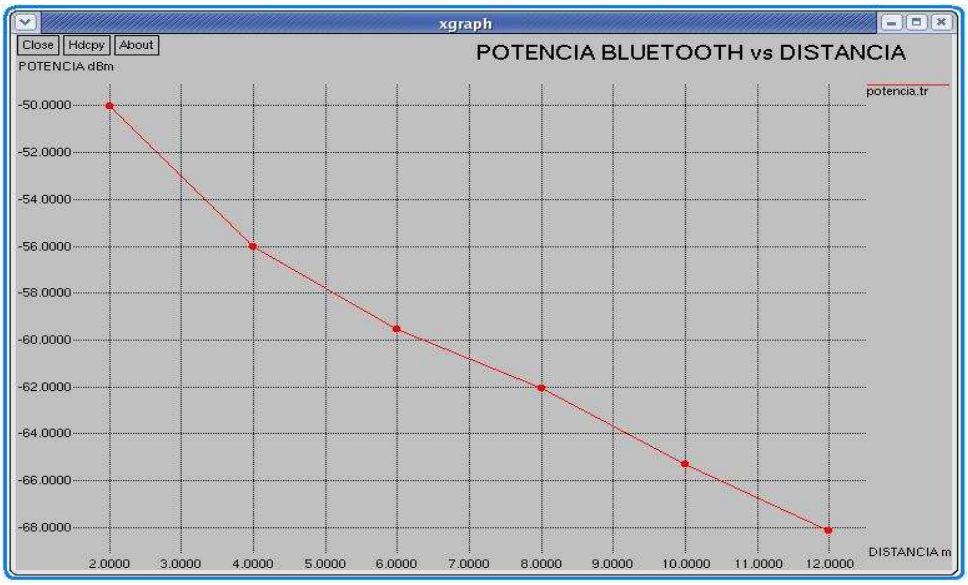


Figura 3.5 Potencia *Bluetooth* de la Simulación

Con la ayuda del *XGraph* y el archivo *señalruido.tr* generado en la simulación, se visualiza la relación señal a ruido en función de la distancia.

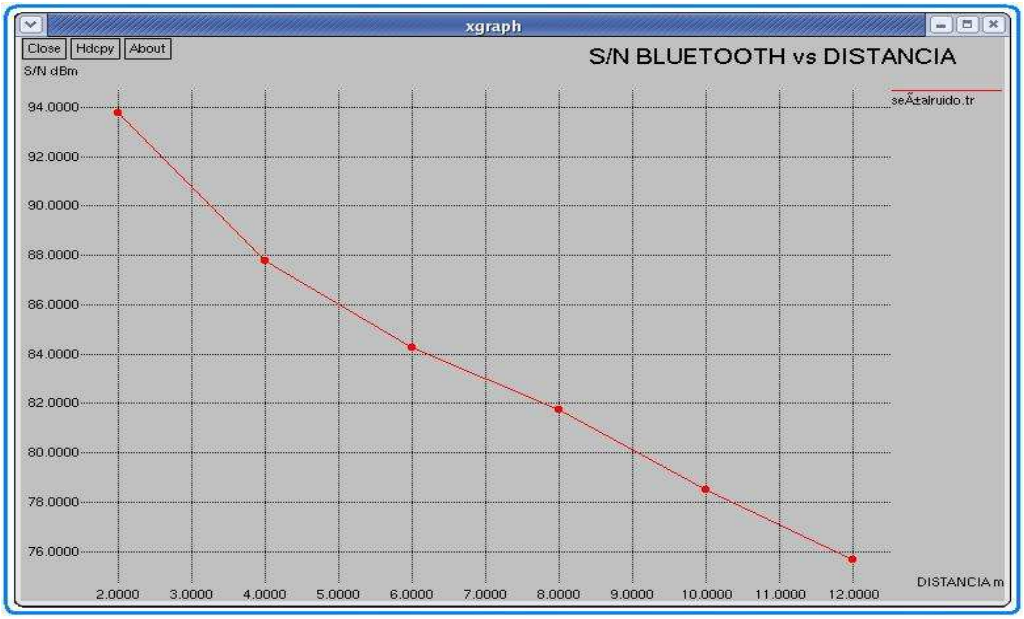


Figura 3.6 Señal a ruido *Bluetooth* de la Simulación

El archivo *efectiva.pl* es un programa que permite procesar el archivo *blue2.tr* generado en la simulación. Este programa permite crear un nuevo archivo con la velocidad efectiva en el nodo que recibe los datos.

Con la ayuda del xg*raph* y el nuevo archivo creado se genera la siguiente gráfica de la velocidad efectiva para *Bluetooth.*

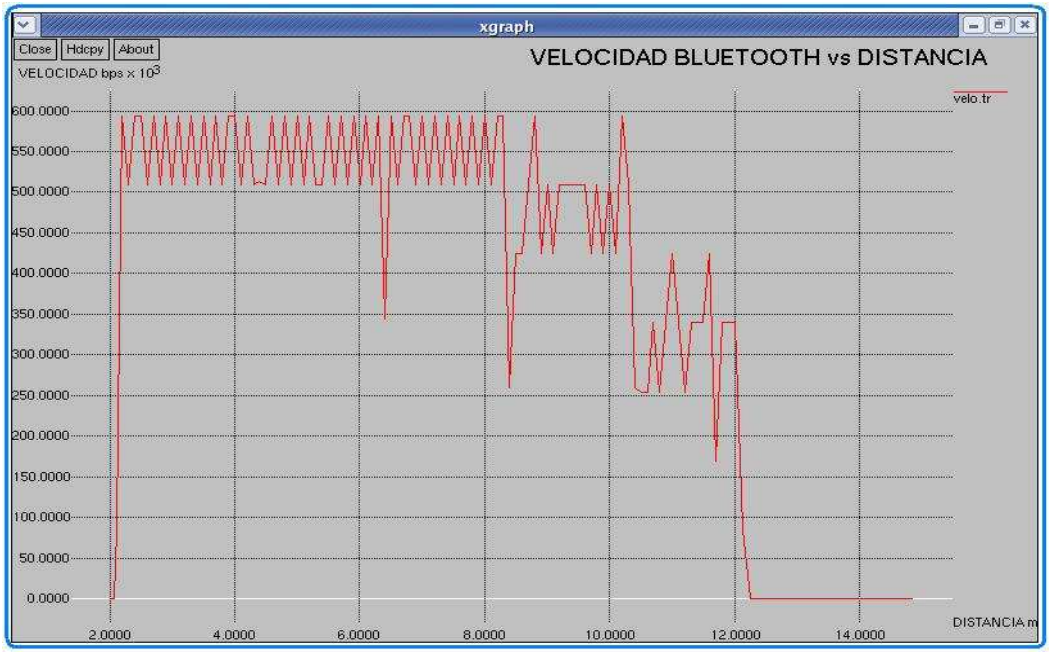


Figura 3.7 Velocidad *Bluetooth* de la Simulación

3.2.1.2 Escenario Wi-Fi

El segundo escenario consta de dos nodos inalámbricos con tecnología *Wi-Fi* (*802.11b*) y un enlace punto a punto unidireccional figura. 3.8. En este escenario se realizarán diversas pruebas en relación a la capacidad del enlace como son: velocidad efectiva, y niveles de potencia.

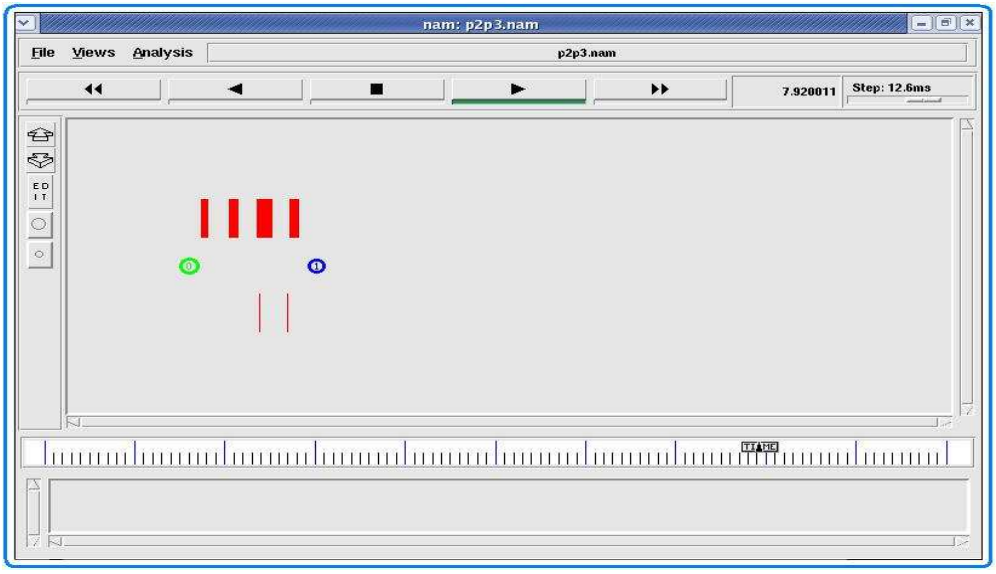


Figura 3.8 Escenario Wi-Fi

3.2.1.2.1 Simulación Wi-Fi

La simulación consiste en crear dos nodos inalámbricos *Wi-Fi* los mismos que se comunican con un enlace *simplex* e ir verificando como varia el *throughput*, la potencia, y la relación señal a ruido en función de la distancia. El nodo uno emitirá paquetes *TCP* con tráfico *FTP* al nodo dos mientras este se va alejando hasta una distancia máxima de 12m. Con la simulación se obtendrá un gráfico de todos los parámetros a analizar en función de la distancia.

El *script* que se ha utilizado en la simulación es fruto de varios *scripts* de prueba que se han ido utilizando a lo largo de todo el proyecto. El código base se ha reutilizado de los ejemplos que vienen con el *ns-2* y que a su vez también se utilizan en el tutorial de Marc Greis. Este código se ha modificado para adecuarlo a las necesidades de este proyecto.

A continuación se muestra el *script* utilizado para la simulación en el cual se comentaran los aspectos más importantes.

- **SCRIPT WI-FI**

Estas líneas permite almacenar en la variable *x*, la distancia y en la variable *n* el coeficiente de pérdidas para *Wi-Fi* con los cuales se quiere realizar la simulación

```
# ======================================================================
#Las siguientes líneas permiten mostrar una ayuda al usuario en el caso
de que no sepa como utilizar el script
# ======================================================================

if {$argc != 2} {

        puts ""
        puts "ERROR"
        puts ""
        puts "UTILICE EL SCRIPT DE LA SIGUIENTE FORMA"
        puts ""
        puts "ns wifi.tcl distancia n"
        puts ""
        puts "n: COEFICIENTE DE PERDIDAS "
        puts ""
        puts "n=1.4 Corredor,n=2 Grandes,n=1.9 Abierto,n=4 Oficinas,n=5.2
             Atraviesa una pared, n=5.4 Atraviesa mas de una pared"
        puts ""
```

```
        puts "DISTANCIAS: En metros"
        puts ""

        exit 1

} else {
  set x [lindex $argv 0]
  set n [lindex $argv 1]
}
```

Las siguientes líneas permiten imprimir en el *terminal* la distancia y el exponente de pérdidas.

```
puts "distancia= $x"
puts "n exponente de perdidas = $n"
```

Las siguientes líneas permiten realizar el cálculo de la potencia de recepción para la distancia ingresada e imprimir la misma en el *terminal*. Más adelante se indica como se realiza el cálculo de la potencia de recepción.

```
 set px [ expr 10*$n*log10($x)]

#Cálculo de las pérdidas totales

        set pT [expr 40.1+$px]

#Cálculo de la potencia
# La potencia para la  tarjeta DWL-G122 es de 14dbm +/- 2dB
        set P [expr 14.0 -$pT]

puts "POTENCIA = $P dbm"
```

Estas líneas permiten poner la velocidad con las que trabaja el interfaz *DWL-G122* en base a los niveles de sensibilidad especificados por la interfaz.

```
if {$P > -78} {
set velocidad 11M
puts "VELOCIDAD = 11 Mbps"

} else {

        if {$P > -82} {

        set velocidad 5.5M
        puts "VELOCIDAD = 5.5 Mbps"

        } else {

                if {$P > -85} {

                set velocidad 2M
                puts "VELOCIDAD = 2 Mbps"
```

```
            } else {

                    if {$P > -87} {

                    set velocidad 1M
                    puts "VELOCIDAD = 1 Mbps"

                    } else {
                            set velocidad 0M
                            puts " NO EXISTE CONEXION "
                            puts " NODOS FUERA DE COBERTURA "

                            }
                    }
            }

    }
```

Se define el canal inalámbrico, el modelo de propagación *Two Ray Ground*, el nivel físico y la *MAC* adecuados para trabajar con el *IEEE 802.11*. El *ns-2* no cambia la velocidad automáticamente a medida que los nodos se alejan es por ello que esto se trata de controlar con las primeras líneas de código.

El modelo de colas utilizado es el *DropTail*, que consiste en una cola simple *FIFO* en la que se descartan los paquetes que sobrepasen la capacidad del tamaño del *buffer* de la cola. El tamaño del *buffer* se ha delimitado a 41 paquetes. Además, al modelo *DropTail* se ha añadido la clase *PriQueue*, que significa que se está dando prioridad a los paquetes que se han enviado utilizando protocolos de enrutamiento.

El protocolo de enrutamiento utilizado es el *DSDV,* se utilizó este protocolo por ser uno de los más utilizados en este tipo de simulaciones. En este protocolo, los nodos vecinos van enviando mensajes de enrutamiento los unos a los otros. Se construye una tabla de enrutamiento en la que se actualizan los cambios. Si se da la situación de que llega un paquete del que no se conoce el destino, se envía a los nodos vecinos un mensaje de solicitud de ruta y se retiene el paquete hasta obtener una respuesta.

Otros parámetros que se definen son el tipo de antena (*omnidireccional*), el número de nodos inalámbricos de la simulación (2), las dimensiones del *nam* para poder posicionar a los nodos, el instante en el que finaliza la simulación (por

ejemplo, a los $x segundos). En este caso el tiempo de simulación coincide con la distancia ingresada.

```
# Define options
set val(chan)           Channel/WirelessChannel
set val(prop)           Propagation/TwoRayGround
set val(netif)          Phy/WirelessPhy
set val(mac)            Mac/802_11
set val(ifq)            Queue/DropTail/PriQueue
set val(ll)             LL
set val(ant)            Antenna/OmniAntenna
set val(ifqlen)         41
set val(nn)             2
set val(rp)             DSDV
set val(x)              50
set val(y)              10
set val(finish)         $x
```

La velocidad a la que se envían los datos por defecto en el *ns-2* es 2*Mbps*. Si se desea trabajar con las distintas velocidades del *IEEE 802.11b* es necesario modificar en el *script* la velocidad. A través de la variable *velocidad* se controla la velocidad a la que se envían los datos. Las líneas de código que empiezan por *Phy/WirelessPhy* o por *Mac/802_11* hacen referencia a variables del código fuente que están en los archivos *wireless-phy.cc* y *mac-802_11.cc.* Entonces, según el valor que se de a la variable velocidad, se acabará trabajando con una modulación o con otra.

Se ha cambiado el valor de la potencia de transmisión a 25 *mW* (14dBm) que es la potencia de transmisión del *DWL-G122*. El simulador utiliza por defecto 282 *mW* (24.5 dBm), que es un valor unos 10 dB superior a los valores especificados en las tarjetas más modernas y además está fuera del margen de niveles de potencia con los que se trabaja en Europa.

A pesar de que la mayoría de fabricantes de tarjetas tienen la funcionalidad de envío de tramas *RTS/CTS* desactivada, el *ns-2* por defecto la tiene activada. Esto ralentiza bastante el envío de información útil, sobretodo en escenarios sencillos como el que se ha planteado. Por este motivo se ha decidido prescindir del envío de este tipo de tramas fijando el parámetro *RTSThreshold_* a 3000 *bytes*. Esto significa que las tramas *RTS/CTS* se enviarán cuando se envíe una trama de

datos de más de 3000 *bytes*. Si por otro lado se fija la longitud de la trama a menos de 3000 *bytes,* este caso no se dará nunca.

Todos los paquetes se envían con un preámbulo, que es una cantidad de bits conocida que se envía al inicio de cada paquete. Esto permite que el receptor se sincronice y así esté listo para la recepción de los datos reales. El preámbulo se envía siempre a 1*Mbps* y puede ser de 144 bits o de 72 bits. El ns-2 utiliza por defecto el valor mayor, aunque en este escenario se ha fijado el valor del preámbulo al valor mínimo para así obtener mejores resultados.

```
Phy/WirelessPhy set bandwidth_ $velocidad
Phy/WirelessPhy set rate_ftp_ $velocidad
Phy/WirelessPhy set Pt_ 0.025

Mac/802_11 set dataRate_ $velocidad
Mac/802_11 set basicRate_ $velocidad
Mac/802_11 set RTSThreshold_ 3000
Mac/802_11 set PreambleLength_ 72
```

Se crea una instancia a la clase simulador, para que se pueda realizar la simulación.

```
set ns [new Simulator]
```

Se crean en modo escritura los ficheros que se utilizan:

- **p2p3.tr:** archivo de texto donde se almacenan las trazas generadas en la simulación. Se puede observar la evolución del envío de cada trama.

- **p2p3.nam:** archivo de texto donde se almacenan las trazas *nam* que nos van a permitir visualizar el escenario de simulación.

- **potenciaw.tr:** archivo de texto que se genera en la simulación y que mediante el *xgraph* permite visualizar como varía la potencia en función de la distancia.

- **señalruidow.tr:** archivo de texto que se genera en la simulación y que mediante el *xgraph* permite visualizar como varía la relación señal a ruido en función de la distancia.

```
set tracefd  [open p2p3.tr w]
set namtrace [open p2p3.nam w]
set pot      [open ./potenciaw.tr w]
set SN       [open ./señalruidow.tr w]
```

Tiene como parámetro el nombre de archivo donde las trazas deberían ir. Con este comando se trazan todos los eventos de acuerdo a un formato específico.

```
$ns trace-all $tracefd
$ns namtrace-all-wireless $namtrace $val(x) $val(y)
```

Se crea la topología en nam con los valores de ‘X’, ‘Y’ definidos anteriormente

```
#set up topology object
set topo [new Topography]
$topo load_flatgrid $val(x) $val(y)
create-god $val(nn)
# Create nn mobilenodes [$val(nn)] and attach them to the channel.
```

Se configuran los nodos con los valores de las variables definidos al inicio del *script*. Además, se activan todas las trazas salvo la *macTrace*. Estos datos se podrán visualizar en el archivo *p2p3.tr*

```
# configure the nodes

        $ns node-config -adhocRouting $val(rp) \
                        -llType $val(ll) \
                        -macType $val(mac) \
                        -ifqType $val(ifq) \
                        -ifqLen $val(ifqlen) \
                        -antType $val(ant) \
                        -propType $val(prop) \
                        -phyType $val(netif) \
                        -channelType $val(chan) \
                        -topoInstance $topo \
                        -agentTrace ON \
                        -routerTrace ON \
                        -macTrace OFF \
                        -movementTrace ON\
```

Se crean los nodos móviles y se les asigna a una variable.

```
for {set i 0} {$i < $val(nn)} {incr i} {
set node_($i) [$ns node]
}
```

Se posiciona el nodo cero en la posición (5, 5, 0) y el nodo 1 a un metro de separación del nodo 0 en la posición (6, 5, 0) en el *nam*.

```
# Provide initial location of mobilenodes
$node_(0) set X_ 5.0
$node_(0) set Y_ 5.0
$node_(0) set Z_ 0.0

$node_(1) set X_ 6.0
$node_(1) set Y_ 5.0
$node_(1) set Z_ 0.0
```

Comando que permite calcular la posición final en “x” del nodo 1

```
#Generation of movements
set posicion [expr 6 + $x]
```

Cuando transcurre 1 segundo, el nodo 1 empieza a desplazarse desde el punto donde está posicionado, que es a un metro de la ubicación del nodo 0 (posición 6.0, 5.0) a la posición x = $posicion, y = 5.0, a una velocidad de 1m/s.

```
$ns at 1.0 "$node_(1) setdest $posicion 5.0 1.0"
```

Se crea un agente tipo *TCP* para el nodo 0, este nodo será el encargado de generar tráfico *tcp* y enviarlo al destino. Se define la conducta del nodo destino y se le asigna a un puntero llamado *sink* este nodo destino es el encargado de generar *acks* que garantizan el arribo de todos los paquetes al nodo 1. Por último se realiza la conexión entre el nodo 0 y el nodo 1.

```
# Set a TCP connection between node_0 and node_(1)

set tcp [new Agent/TCP/Newreno]
$tcp set class_ 2
set sink [new Agent/TCPSink]
$ns attach-agent $node_(0) $tcp
$ns attach-agent $node_(1) $sink
$ns connect $tcp $sink
```

Las siguientes líneas generan tráfico *FTP* sobre una conexión *TCP ns-2* posee muchos parámetros por defecto que pueden ser cambiados. Como por ejemplo el tamaño del paquete *ftp* que en este caso es de 1024 bytes y la velocidad que ha sido cambiada al parámetro *velocidad*.

```
set ftp [new Application/FTP]
$ftp attach-agent $tcp
$ftp set packetSize 1024
```

```
$ftp set rate_ $velocidad
```

Al tiempo t = 2.0 segundos se empieza a generar tráfico *FTP*.

```
$ns at 2.0  "$ftp start"
```

Se define el tamaño del nodo en el *nam* en este caso de 1.0 para los dos nodos.

```
#Define node initial position in nam
for {set i 0} {$i < $val(nn)} {incr i} {
$ns initial_node_pos $node_($i) 1.0
}
```

A los dos segundos se llama a la función *record*

```
#Al 2.0 segundos se llama a la función record
$ns at 2.0  "record"
```

Se define tres variables locales para la función *record pot, SN* y *n.*

```
proc record {} {
        global sink pot
        global sink SN
        global sink n
```

Indica la granularidad de 0.25 segundos y se almacena en la variable *time*

```
    set ns_ [Simulator instance]
        set time 0.25
```

Indica el instante en el que se encuentra la simulación.

```
#Calculo de la distancia

        set now [$ns_ now]
```

Se calcula la distancia cada 0.25 segundos y se almacena en la variable distancia. Para el cálculo de la distancia se multiplica el tiempo por la velocidad de desplazamiento del nodo “1”

```
#Para el cálculo de la distancia multiplico el tiempo x la velocidad de
desplazamiento

        set distancia [expr $now*1.0 ]
```

Se realiza el cálculo de las pérdidas en la trayectoria y el cálculo de la potencia para *Wi-Fi* utilizando la ecuación 1.18. Para el cálculo de la potencia de transmisión se toma el valor de 14dBm que es la especificada para la interfaz *DWL-G122.*

El coeficiente de pérdidas *n* será ingresado por el usuario. Estos parámetros dependerán del lugar donde se realicen las pruebas.

```
#Calculo de las pérdidas en función de la distancia.

        set perdidas [ expr 10*$n*log10($distancia)]

#Calculo de las pérdidas totales

        set pérdidasT [expr 40.1+$pérdidas]

#Cálculo de la potencia
# La potencia para la  tarjeta DWL-G122 es de 14dbm +/- 2dB

        set potencia [expr 14.0 - $perdidasT]
```

La siguiente línea permite calcular la relación señal a ruido y almacenarla en la variable *sn.* El ruido se calculó en base a las ecuaciones que se encuentran en el ANEXO F.

Para una temperatura de 27℃ y un ancho de banda de 22 *MHz* correspondiente para *Wi-Fi* se obtuvo un ruido de -140.86 dB.

```
        set sn  [expr $potencia+140.86]
```

Se imprime en el archivo *potenciaw.tr* la distancia y la potencia. En el archivo *señallruidow.tr* se imprime la distancia y la relación señal a ruido.

```
#Imprimo en el archivo potecia.tr la distancia y la Prx en dBm

        puts $pot "$distancia    $potencia  "
        puts $SN  "$distancia    $sn "
```

Cada 0.25 segundos llamamos a la función *record* para que realice el cálculo de los parámetros requeridos.

```
        $ns_ at [expr $now+$time] "record"

}
```

Se comunica a los nodos cuando finaliza la simulación.

```
#telling nodes when the simulation ends

for {set i 0} { $i < $val(nn)} { incr i} {
```

```
        $ns at $val(finish) "$node_($i) reset"
}
```

Por último, se finaliza la simulación y se ejecuta el archivo *p2p3.nam*. El *script* acaba llamando a la instrucción *run*, que permite ejecutar todo el código.

```
#ending nam and the simulation
$ns at $val(finish) "$ns nam-end-wireless $val(finish)"
$ns at $val(finish) "finish"

set terminar [expr $x + 0.01]

$ns at $terminar   "puts \"end simulation\"; $ns halt"

proc finish {} {
        global node
        exec nam p2p3.nam &

}

$ns run
```

Las siguientes líneas son necesarias para visualizar automáticamente los resultados de la simulación en el *xgraph* como: potencia, velocidad efectiva, relación señal a ruido en función de la distancia.

Para el cálculo de la velocidad efectiva se ejecuta el comando *perl* que conjuntamente con el archivo *throughput.pl* y con el archivo *p2p3.tr*, indicando el nodo donde se necesita visualizar el tráfico y la granularidad, generan un archivo *thpwifi.tr* en el cual se almacena la velocidad efectiva de wi-fi en función de la distancia. El archivo *throughput.pl* se encuentra en el *ANEXO I*

```
#Comandos necesarios para visualizar resultados de la simulacion

  exec perl throughput1.pl p2p3.tr _1_ 0.1 > thpwifi.tr & \

  exec xgraph thpwifi.tr -t        "VELOCIDAD WI-FI VS DISTANCIA" -x
                                   "DISTANCIA m" -y "VELOCIDAD bps" &

  exec xgraph potenciaw.tr -t      "POTENCIA WI-FI VS DISTANCIA" -x
                                   "DISTANCIA m" -y "POTENCIA dBm" &


  exec xgraph señalruidow.tr -t    "S/N WI-FI VS DISTANCIA" -x "DISTANCIA
                                   m" -y "S/N dBm" &

exit 0
```

3.2.1.2.2 Ejemplo de Simulación del Prototipo Wi-Fi

Para realizar la simulación el usuario debe ingresar al directorio en el cual se encuentran los *scripts wifi.tcl* y *throughput1.pl* en este caso los *scripts* se encuentran en el directorio *wi-fi*

En este caso para acceder al directorio el usuario debe ingresar los siguientes comandos en el *terminal* de *Linux.*

cd/root/Desktop/BLUETOOTH1/definitivo/wi-fi/

Para obtener ayuda del uso del *script* ingresar el siguiente comando.

ns wifi.tcl

Después de ejecutar el comando ns *wifi.tcl* se despliega la siguiente pantalla, la cual indica la forma de utilizar el *script* para iniciar la simulación.

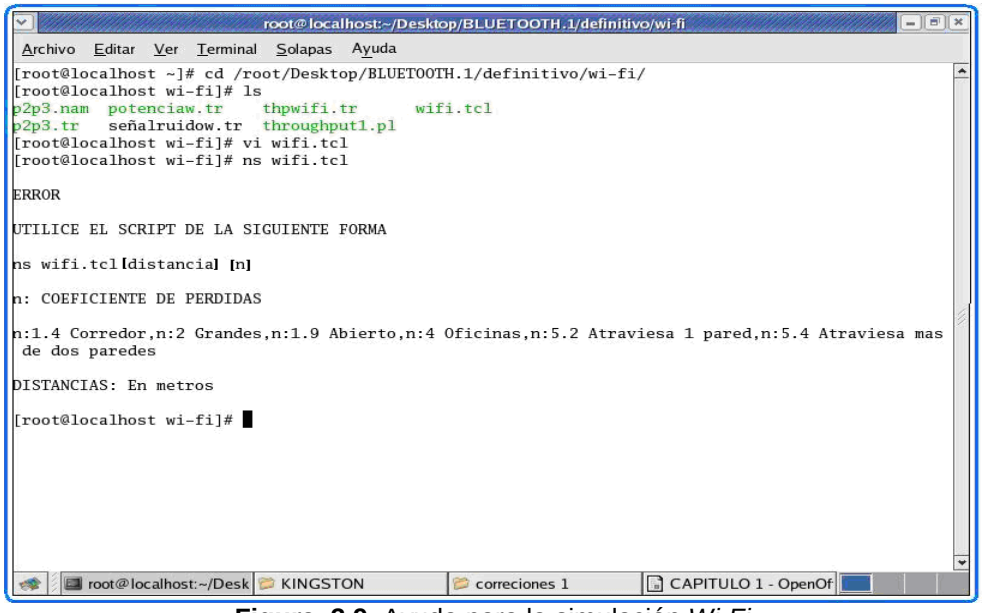


Figura 3.9 Ayuda para la simulación *Wi-Fi*

Aquí el usuario puede escoger la distancia y el coeficiente de pérdidas con el que se desea realizar la simulación. Para ejecutar el *script* ingresar el siguiente comando.

ns wifi.tcl 10 5.2

Una vez que se ejecute el programa se visualizará en el *terminal* de *Linux* la siguiente información. La simulación se realizó con una distancia igual 10 m y un coeficiente de pérdidas igual a 5.2

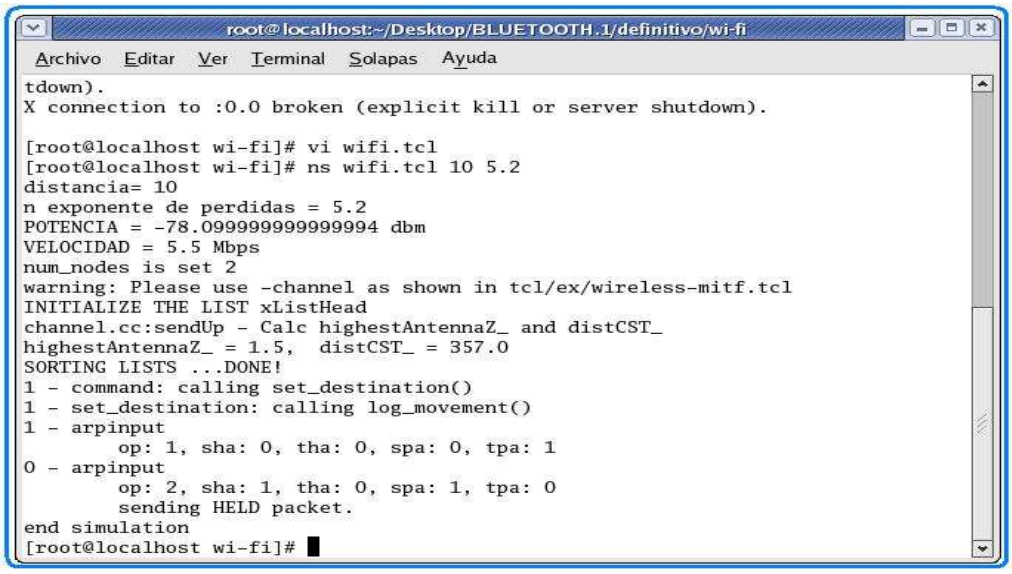


Figura 3.10 Información de la simulación *Wi-Fi*

También se visualizará de forma automática el escenario en el *nam* y los resultados que se obtienen de la simulación en el *xgraph* como son: potencia, relación señal a ruido y velocidad efectiva.

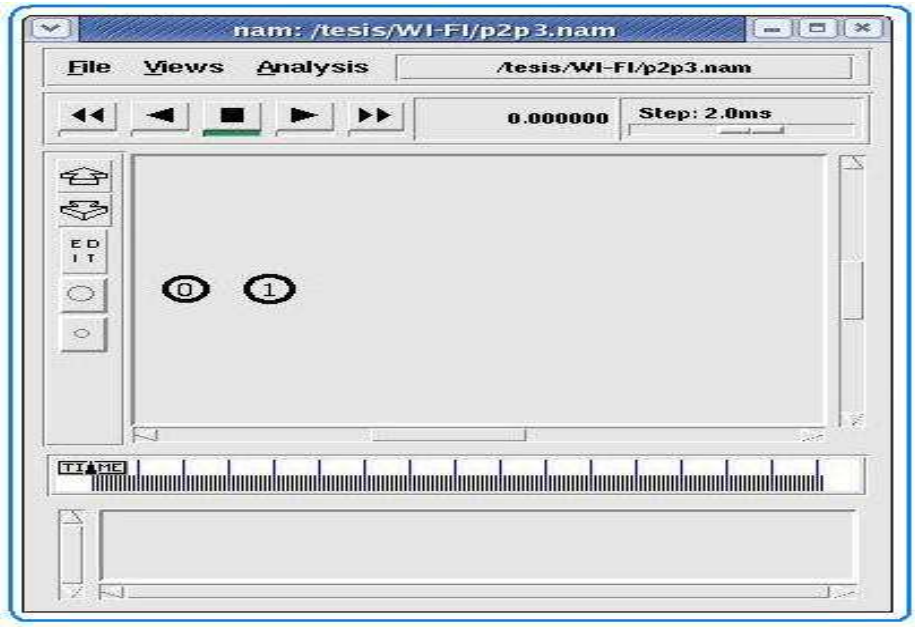


Figura 3.11 Pantalla inicial del nam W*i-Fi*

Luego de iniciar la simulación en el *nam* se visualiza como los paquetes son enviados.

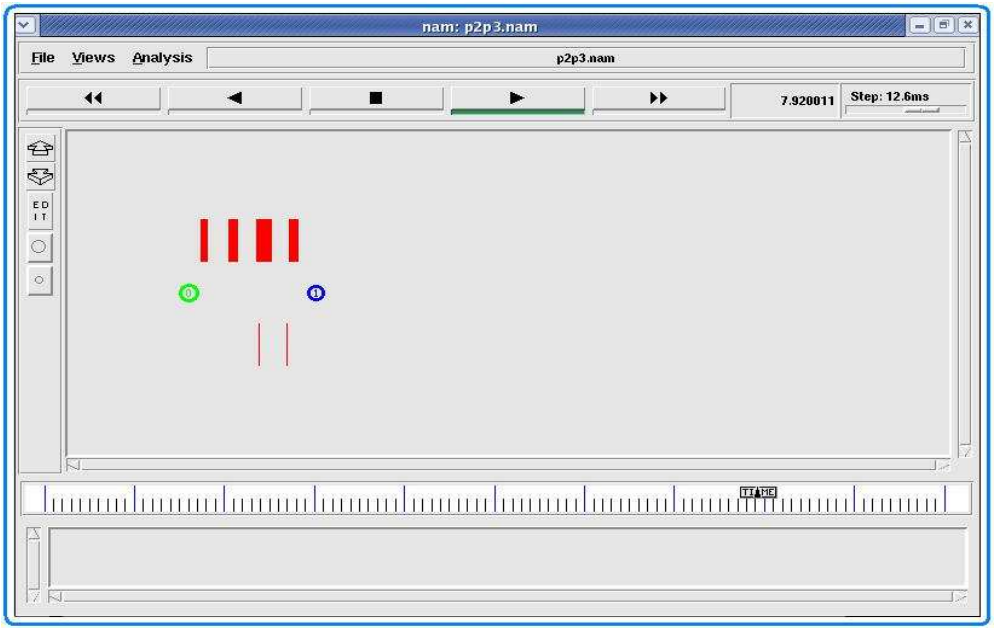


Figura 3.12 Simulación *Wi-Fi* en el *nam*

Con la ayuda del xg*raph* y el archivo *potenciaw.tr* que se genera en la simulación, visualizamos como la potencia cambia en función de la distancia.

Figura 3.13 Potencia *Wi-Fi* de la Simulación

Con la ayuda del *xgraph* y el archivo *señalruidow.tr* generado en la simulación, se visualiza la relación señal a ruido en función de la distancia.

Figura 3.14 Señal a ruido *Wi-Fi* de la Simulación

El archivo *througtput1*.pl que es un programa que permite procesar el archivo *p2p3.tr* generado en la simulación. Este programa permite crear un nuevo archivo con la velocidad efectiva en el nodo que recibe los datos.

Con la ayuda del *xgraph* y el nuevo archivo creado se genera la siguiente gráfica de la velocidad efectiva.

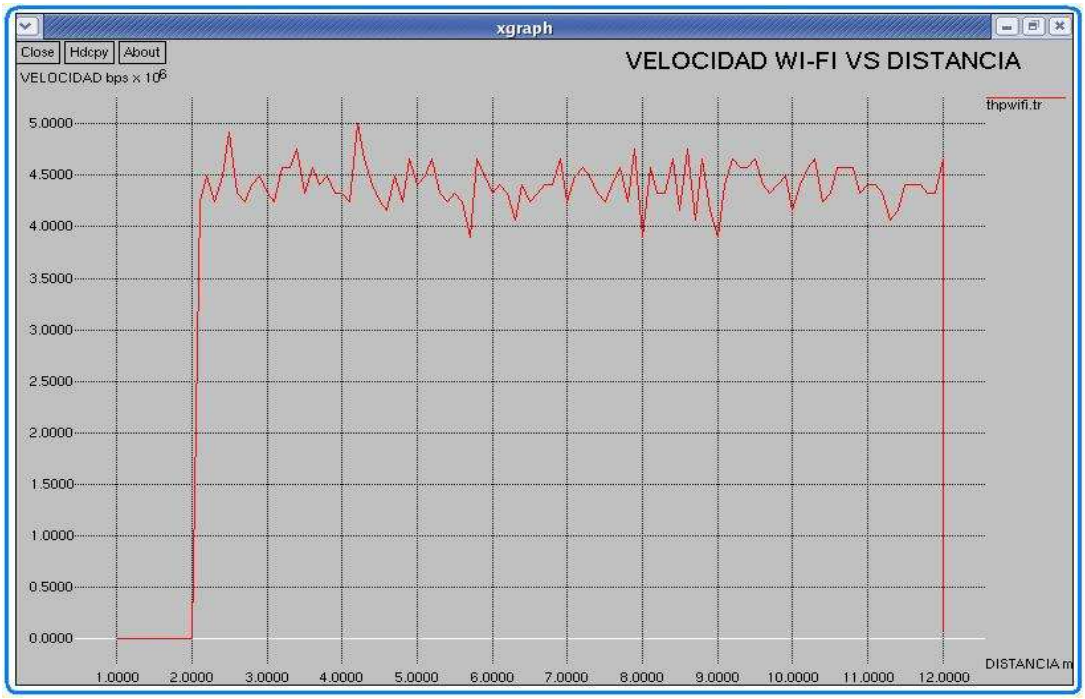


Figura 3.15 Velocidad Efectiva *Wi-Fi* de la Simulación

CAPÍTULO 4

PRUEBAS Y ANÁLISIS DE RESULTADOS

4.1 INTRODUCCIÓN

Uno de los objetivos principales del proyecto es la parte de pruebas, en este capítulo se realizarán las pruebas correspondientes que permitan evaluar el correcto funcionamiento del sistema implementado.

En base a estas pruebas se realizará la comparación de las tecnologías *Bluetooth* y *Wi-Fi*, para así determinar las ventajas y desventajas que implican el uso de estas. Las pruebas a realizarse se separan en dos partes:

- La primera se relaciona a todas las pruebas prácticas correspondientes a los prototipos inalámbricos con tecnología *Bluetooth* y *Wi-Fi*, entre las distintas pruebas a comprobarse están las siguientes: conectividad, nivel de potencia, velocidad, y pérdida de datos.

- La segunda corresponde a la simulación de cada prototipo en la cual se podrá ver como varía la velocidad efectiva, nivel de potencia de recepción y la relación señal a ruido. Para realizar esta prueba se utilizó el simulador *ns-2* versión (2.29.3).

4.2 PRUEBAS PRÁCTICAS

A continuación se detallan las pruebas prácticas correspondientes a los prototipos, las pruebas han sido realizadas iniciando con una distancia de separación mínima de un metro hasta una distancia de separación máxima de quince metros.

4.2.1 PRUEBAS PRÁCTICAS *BLUETOOTH*

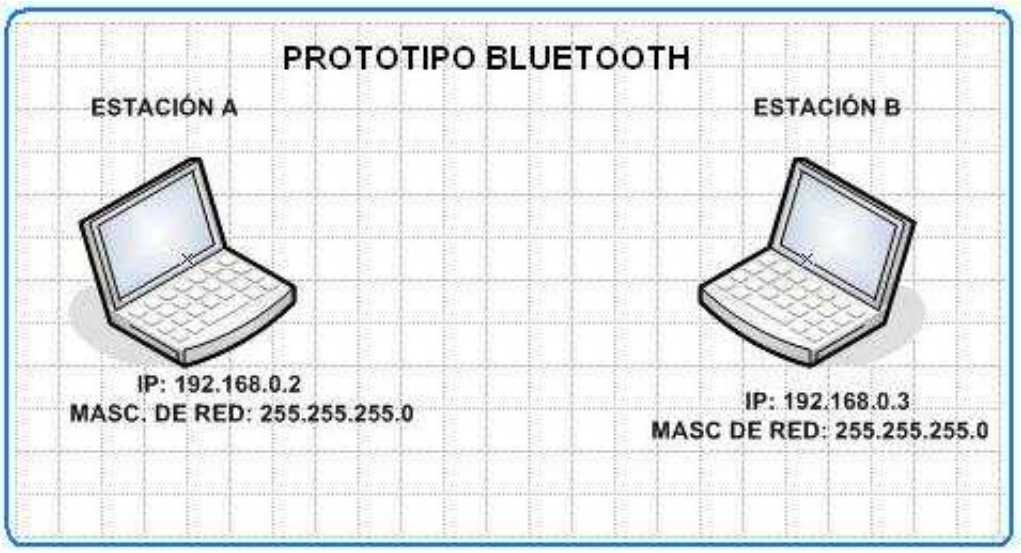


Figura 4.1 Prototipo *Bluetooth*

Se presenta el detalle de las medidas realizadas a la distancia de 1m, todo el conjunto de medidas hasta los 15m se encuentran en el ANEXO D

Medidas a 1 metro de separación

La figura 4.2 indica el nivel de señal detectada por el *softwar*e del interfaz *DBT-122*.

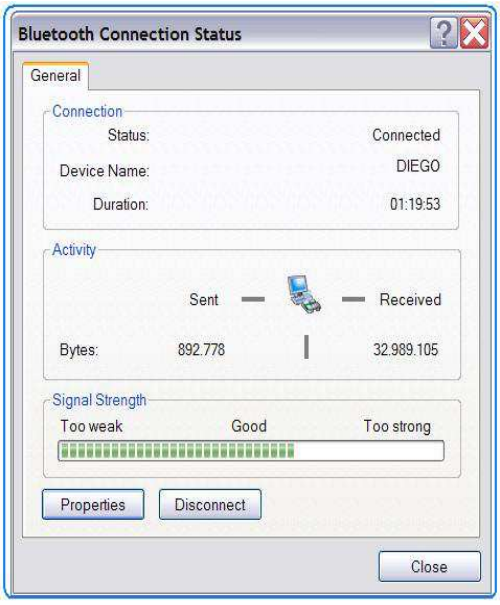


Figura 4.2 Estado de conexión 1m

La figura 4.3 representa el nivel de potencia de la señal medida con el analizador de espectros (d = 1 m, p = - 54.83 dBm, f = 2.4155 GHz)

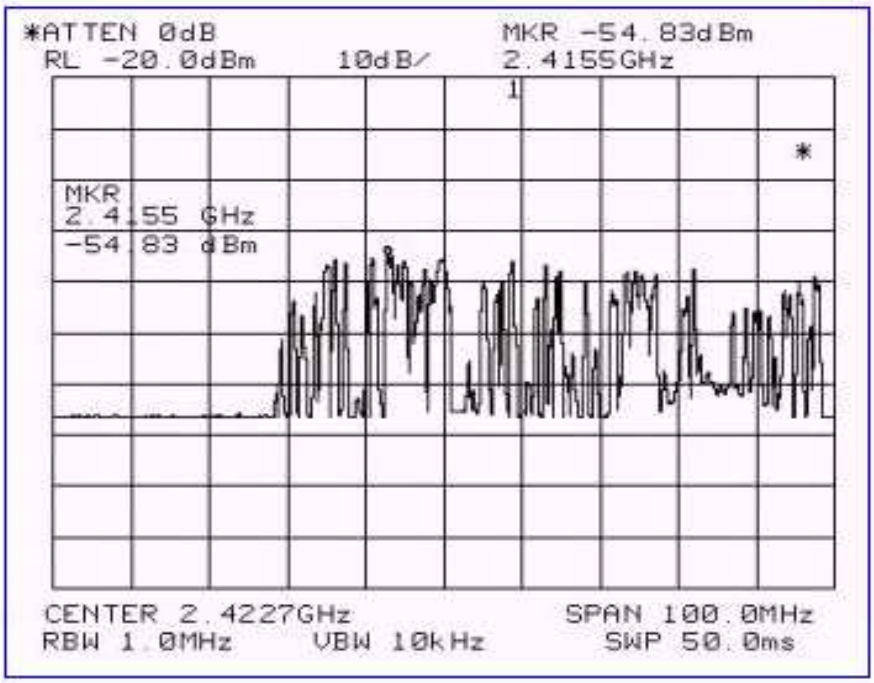


Figura 4.3 Nivel de potencia 1m

La figura 4.4 indica la respuesta entre las estaciones a través del comando *ping*.

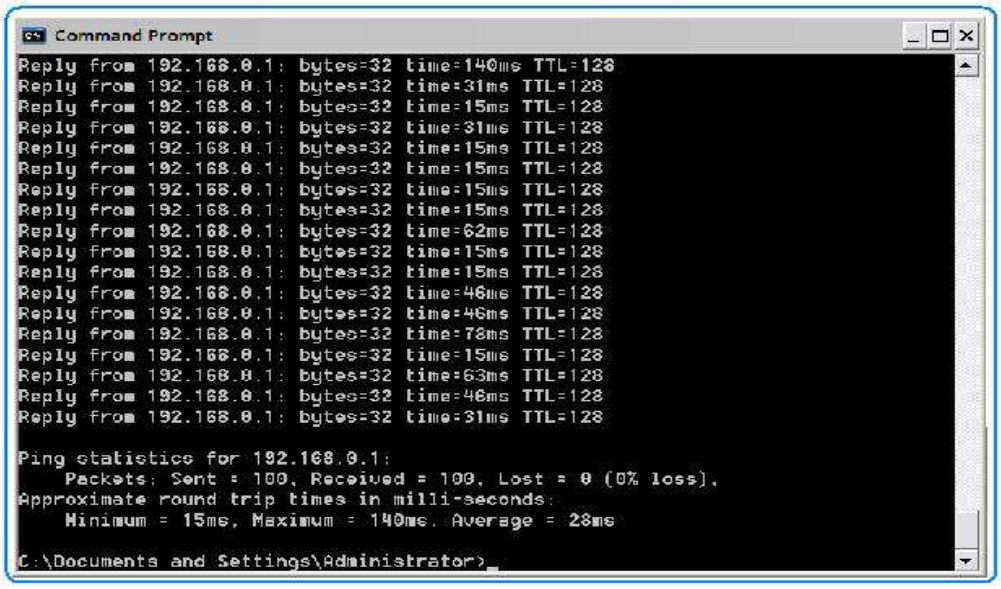


Figura 4.4 *Ping* entre las estaciones 1m

La figura 4.5 indica la velocidad que se obtiene al transferir un archivo en función del tiempo, utilizando el *software Smart FTP.* La velocidad esta expresada en *Kbytes/s.* la figura 4.6 indica la velocidad promedio y el tamaño del archivo transferido.

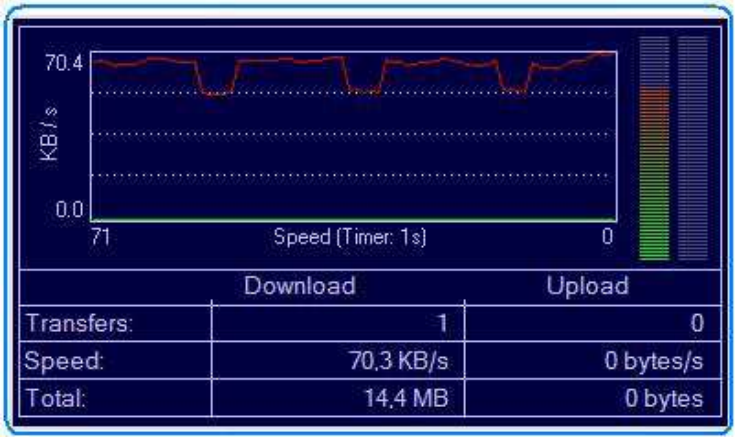


Figura 4.5 Velocidad a un metro de separación

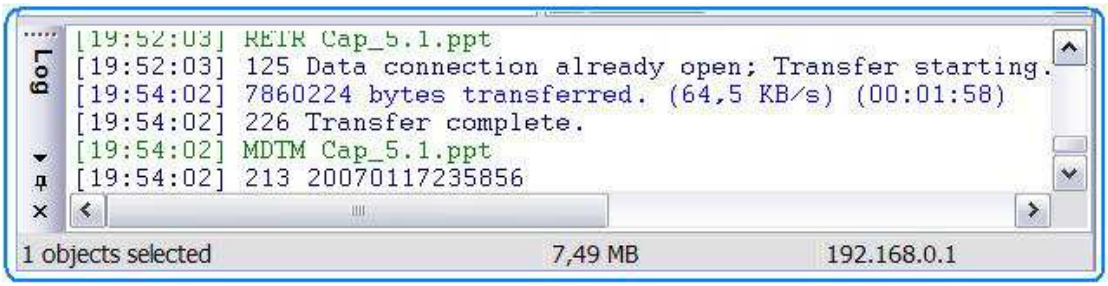


Figura 4.6 Velocidad Promedio a un metro de separación

4.2.2 PRUEBAS PRÁCTICAS *Wi-Fi*

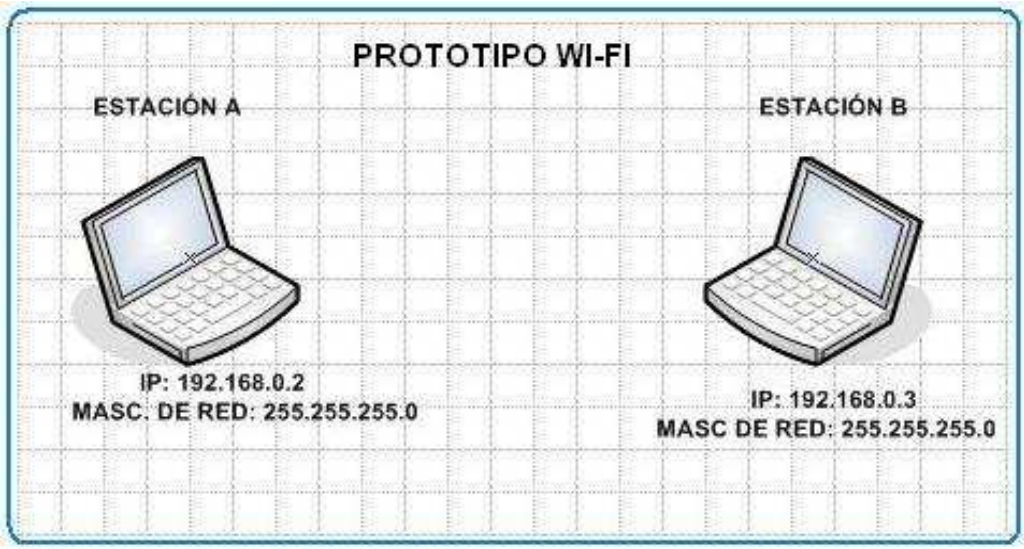


Figura 4.7 Prototipo *Wi-Fi*

Se presenta el detalle de las medidas realizadas a la distancia de 1m, todo el conjunto de medidas hasta los 15m se encuentran en el ANEXO E

Medidas a 1 metro de separación

La figura 4.8 indica el nivel de señal detectada por el *softwar*e del interfaz *DWL-G122*.

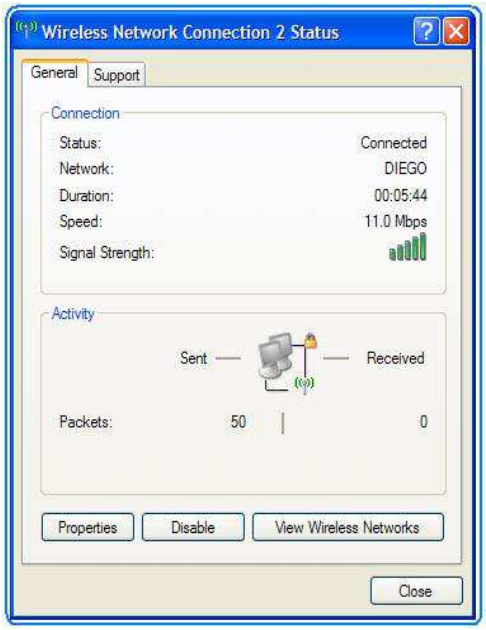


Figura 4.8 Estado de conexión 1m

La figura 4.9 representa el nivel de potencia de la señal medida con el analizador de espectros (d = 1 m, p = - 47.33 dBm, f = 2.4403 GHz)

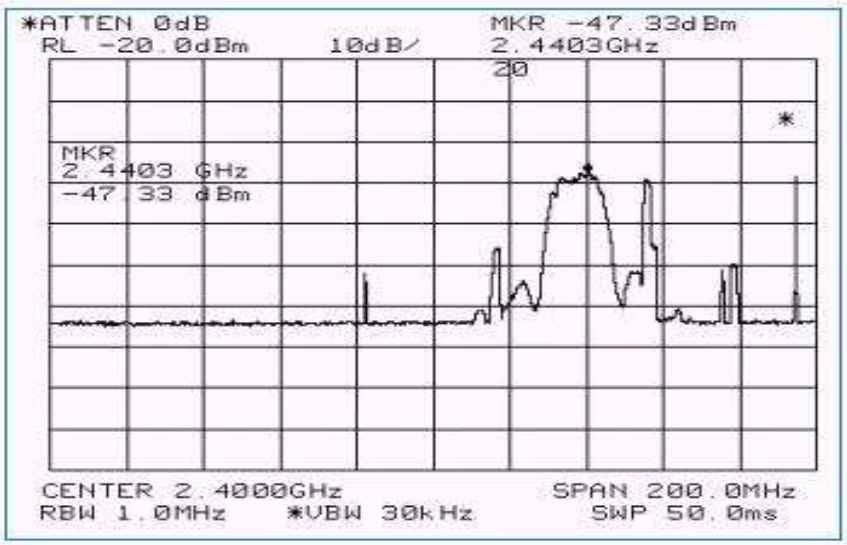


Figura 4.9 Nivel de potencia 1m

La figura 4.10 indica la respuesta entre las estaciones a través del comando *ping*

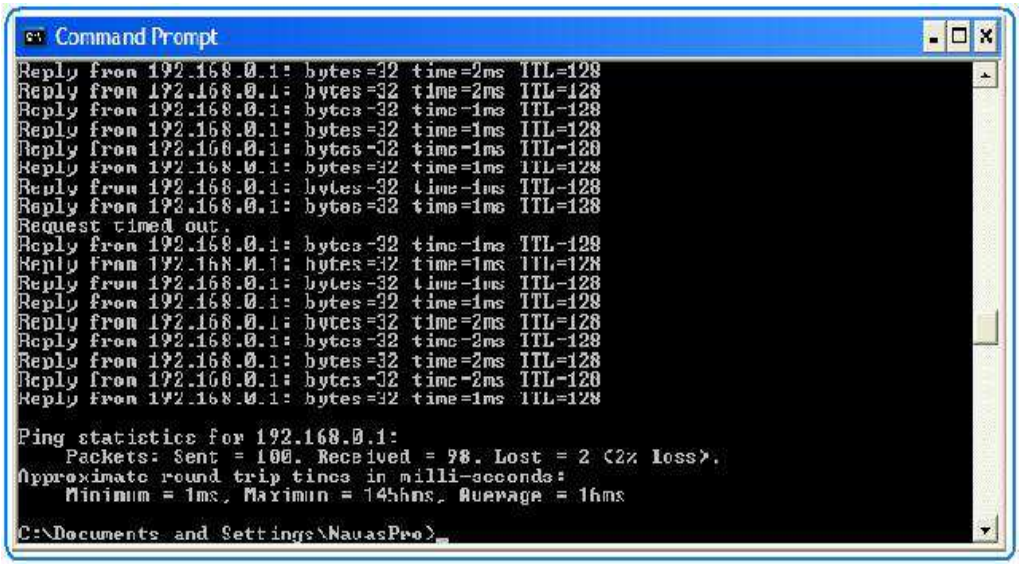


Figura 4.10 *Ping* entre las estaciones 1m

La figura 4.11 indica la velocidad que se obtiene al transferir un archivo en función del tiempo, utilizando el *software Smart FTP.* La velocidad esta expresada en *Kbytes/s.* la figura 4.12 indica la velocidad promedio y el tamaño del archivo transferido.

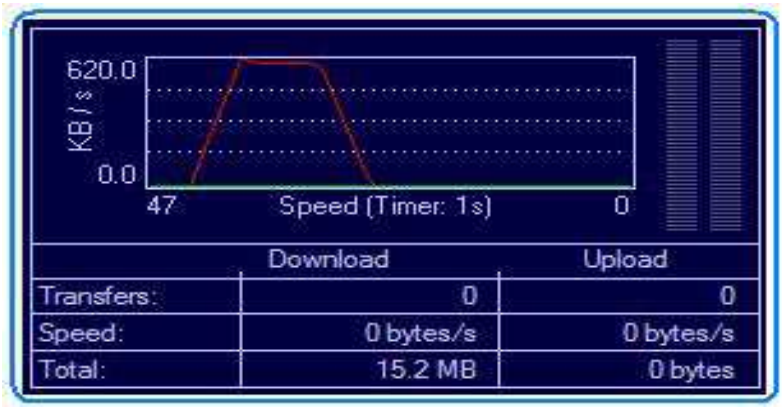


Figura 4.11 Velocidad 1m

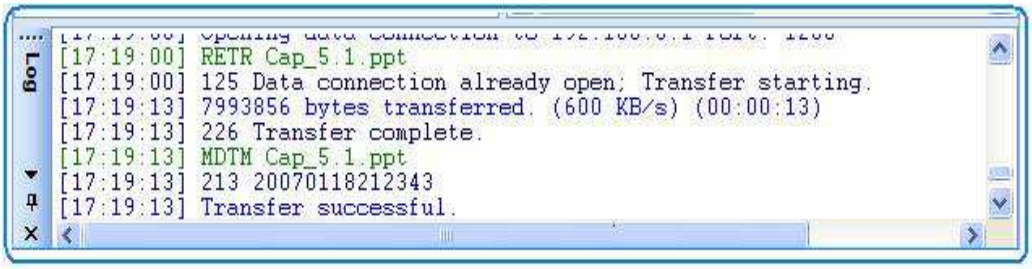


Figura 4.12 Velocidad Promedio 1m

4.2.2.1 Comparación y Análisis de Resultados de Pruebas Prácticas

Para realizar el análisis de los resultados obtenidos en las pruebas prácticas de *Bluetooth* y *Wi-Fi* de mejor manera, se los agrupó en la tabla 4.1 y tabla 4.2 respectivamente y además se realizó la representación gráfica de los datos obtenidos.

No. de prueba	Distancia (m)	Potencia (dBm)	Velocidad promedio (KB/s)	Velocidad promedio (Kbps)	Frecuencia (GHz)	Pérdida de datos %
1	1	-54.83	64.5	528.38	2.4155	0
2	2	-55.67	63.4	519.37	2.4747	0
3	3	-58.83	63.0	516.09	2.4740	0
4	4	-56.83	58.1	475.95	2.4537	0
5	5	-57.17	63.9	523.46	2.4557	0
6	6	-56.83	57.7	472.67	2.4420	0
7	7	-58.17	62.0	507.90	2.4287	0
8	8	-58.00	39.9	326.86	2.4277	0
9	9	-57.33	50.1	410.42	2.4177	0
10	10	-58.50	30.6	250.67	2.4307	0
11	12	-60.00	23.6	193.33	2.4397	0
12	15	-61.83	3.15	25.8	2.4207	0

Tabla 4.1 Resultados obtenidos en las pruebas de *Bluetooth*

No. de prueba	Distancia (m)	Potencia (dBm)	Velocidad promedio (KB/s))	Velocidad promedio (Mbps)	Frecuencia (GHz)	Pérdida de datos %
1	1	-47.33	600.0	4.91	2.4403	1
2	2	-59.00	613.0	5.02	2.4340	2
3	3	-60.83	610.0	4.99	2.4410	1
4	4	-56.83	557.0	4.56	2.4383	1
5	5	-58.33	589.0	4.82	2.4383	3
6	6	-62.50	587.0	4.80	2.4337	2
7	7	-58.50	594.0	4.86	2.4383	1
8	8	-66.83	604.0	4.94	2.4323	2
9	9	-66.50	607.0	4.97	2.4347	4
10	10	-68.83	610.0	4.99	2.4207	3
11	12	-69.39	627.2	5.13	2.4397	9
12	15	-70.50	561	4.59	2.4400	19

Tabla 4.2 Resultados obtenidos en las pruebas de *Wi-Fi*

4.2.2.2 Representación gráfica de resultados obtenidos en las pruebas prácticas

La figura 4.13 representa la pérdida de paquetes en función de la distancia para el prototipo *Bluetooth*, se puede observar que en este caso no existe pérdida de datos, por lo que se puede decir que este enlace es confiable.

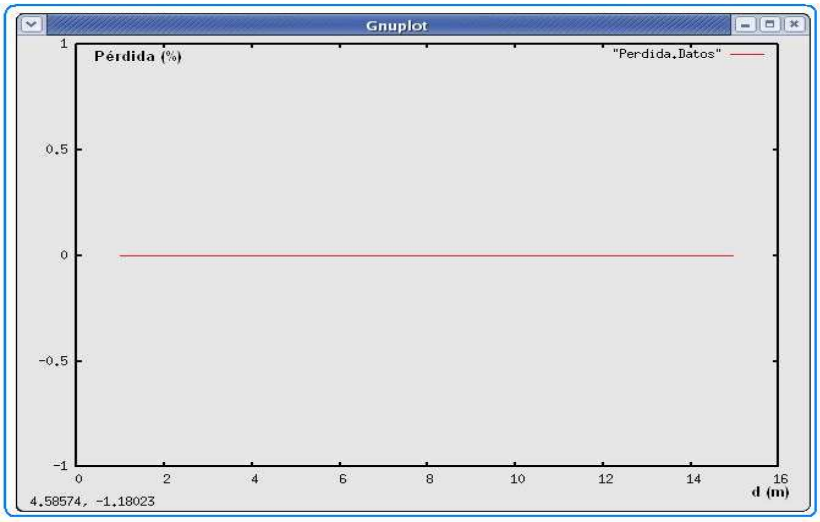


Figura 4.13 Pérdida de Datos vs Distancia de *Bluetooth*

La figura 4.14 representa la pérdida de paquetes en función de la distancia del prototipo *Wi-Fi*, aquí se observa que en distancias cortas las pérdidas son despreciables, pero a medida que aumenta la distancia estas pérdidas aumentan lo que ocasiona que el sistema sea desconfiable.

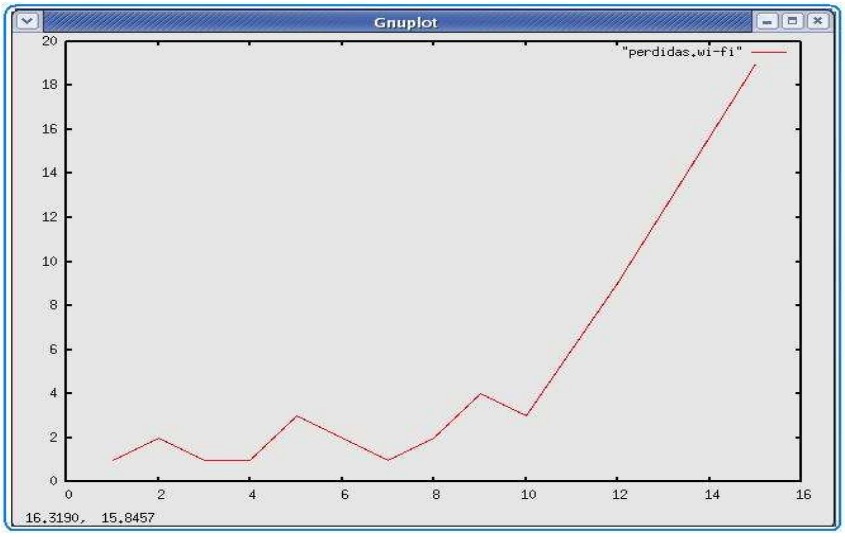


Figura 4.14 Pérdida de Datos vs Distancia de *Wi-Fi*

La figura 4.15 indica la variación del nivel de potencia en función de la distancia del prototipo *Bluetooth*. Se puede observar que a distancias menores a 10 metros los niveles de potencia no sufren atenuaciones considerables, pero a partir de la misma distancia el enlace sufre una atenuación considerable en la señal de potencia.

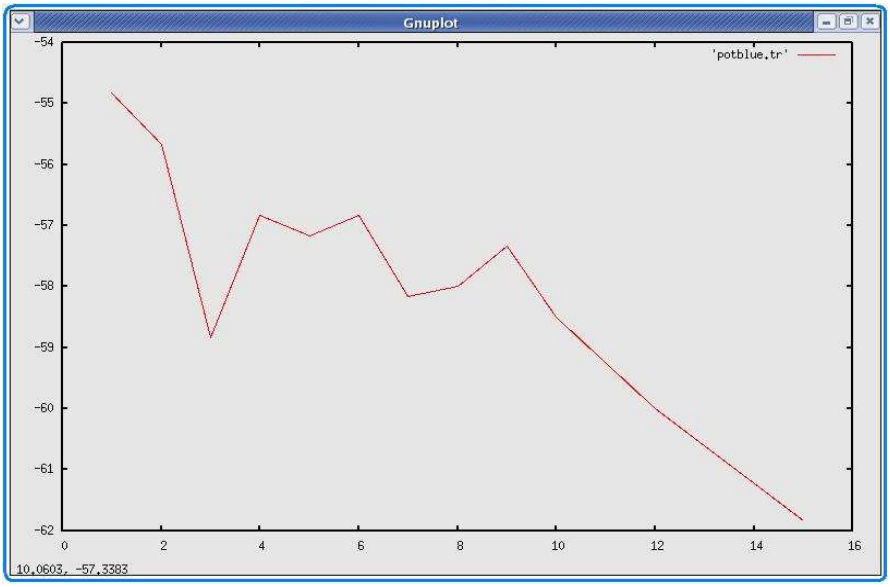


Figura 4.15 Potencia vs Distancia de *Bluetooth*

La figura 4.16 indica la variación del nivel de potencia del prototipo *Wi-Fi*, como se puede ver este sistema en distancias menores a 10 metros tiene atenuaciones considerables, pero a partir de la misma distancia el nivel de señal se mantiene estable.

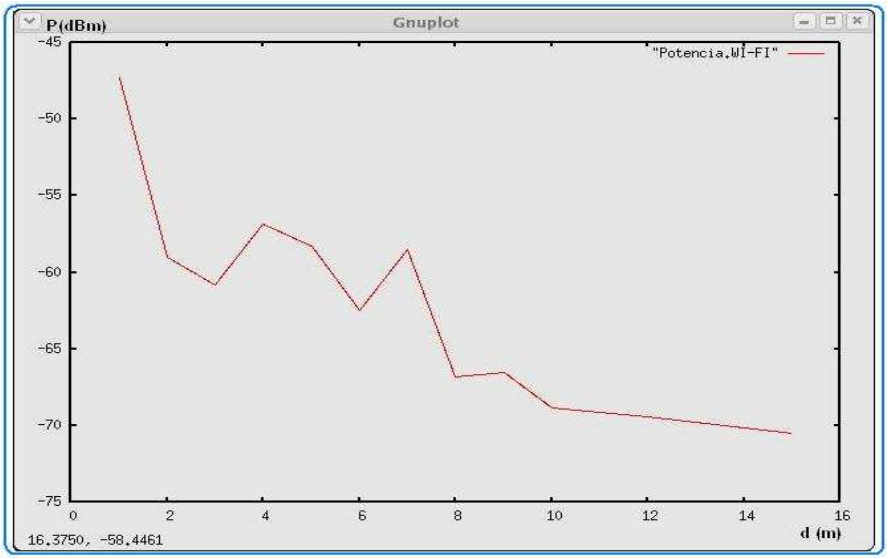


Figura 4.16 Potencia vs Distancia de *Wi-Fi*

La figura 4.17 indica la variación de la velocidad de transmisión en función de la distancia del prototipo *Bluetooth,* se puede observar que en distancias menores a los 6 metros la velocidad es estable, pero a distancias mayores a 7 metros esta velocidad disminuye notablemente.

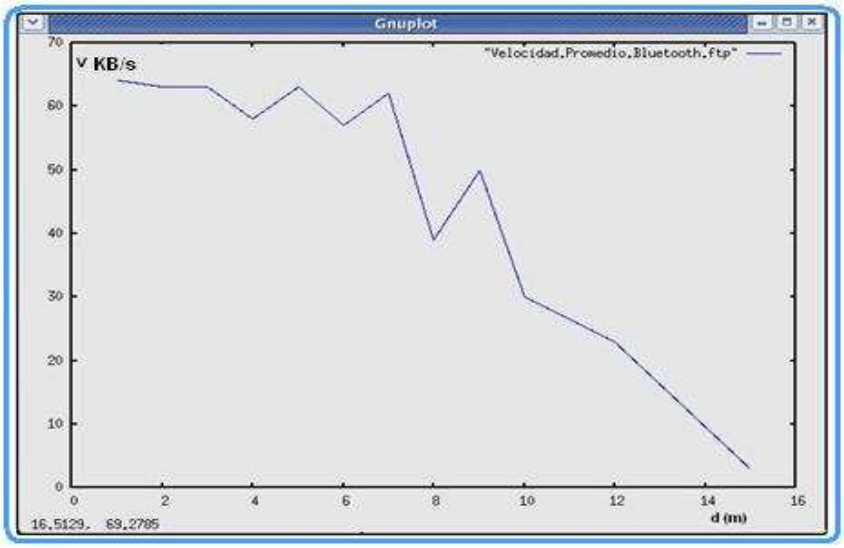


Figura 4.17 Velocidad Promedio vs Distancia de *Bluetooth*

La figura 4.18 indica la variación de la velocidad de transmisión de datos del prototipo *Wi-Fi,* aquí se puede observar que la velocidad esta dentro de los márgenes de velocidad aceptables. Cabe mencionar que para la distancia igual 4 m esta velocidad disminuye debido a obstáculos existentes donde se realizaron las pruebas que hacen que la señal se degrade.

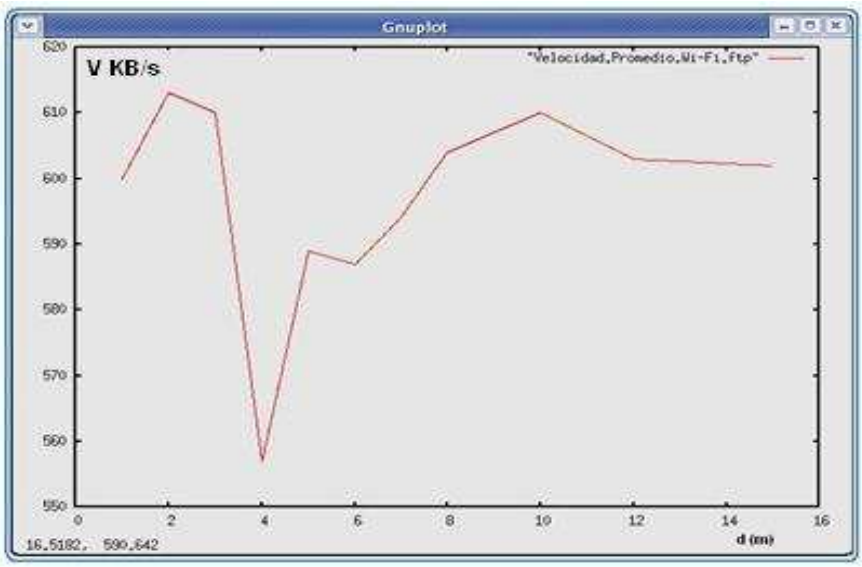


Figura 4.18 Velocidad Promedio vs Distancia de *Wi-Fi*

4.2.2.3 Comparación de pruebas prácticas

La figura 4.19 representa la comparación entre *Bluetooth* y *Wi-Fi* con respecto a pérdida de datos en función de la distancia, aquí se observa que *Bluetooth* no tiene pérdida de datos para este caso, ya que en el sitio donde se realizaron las pruebas no existían redes que utilicen este tipo de tecnología, mientras que en *Wi-Fi* la pérdida de datos ocurre a medida que aumenta la distancia. Como conclusión se puede decir que *Bluetooth* es más estable que *Wi-Fi* con respecto a pérdida de datos.

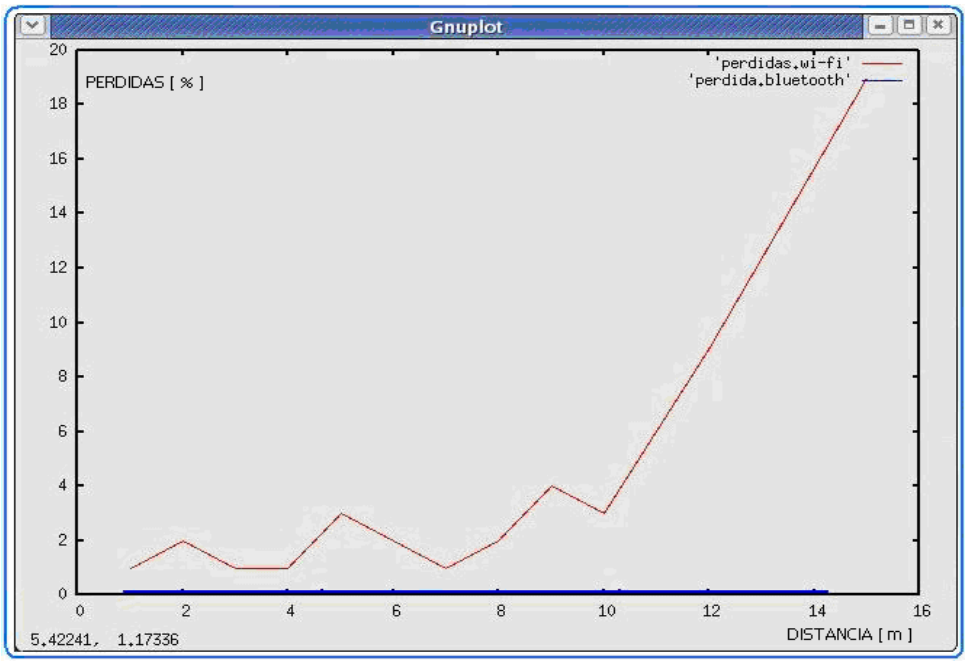


Figura 4.19 Pérdida de Datos *Bluetooth* y *Wi-Fi*

La figura 4.20 representa la comparación de los niveles de potencia para *Bluetooth* y *Wi-Fi* en función de la distancia, se observa que *Bluetooth* presenta mayor estabilidad que *Wi-Fi*. La curva que representa la potencia de *Wi-Fi* decrece más rápidamente que la de *Bluetooth*, porque los dispositivos *Wi-Fi* son más propensos a receptar mayores interferencias existentes en el medio.

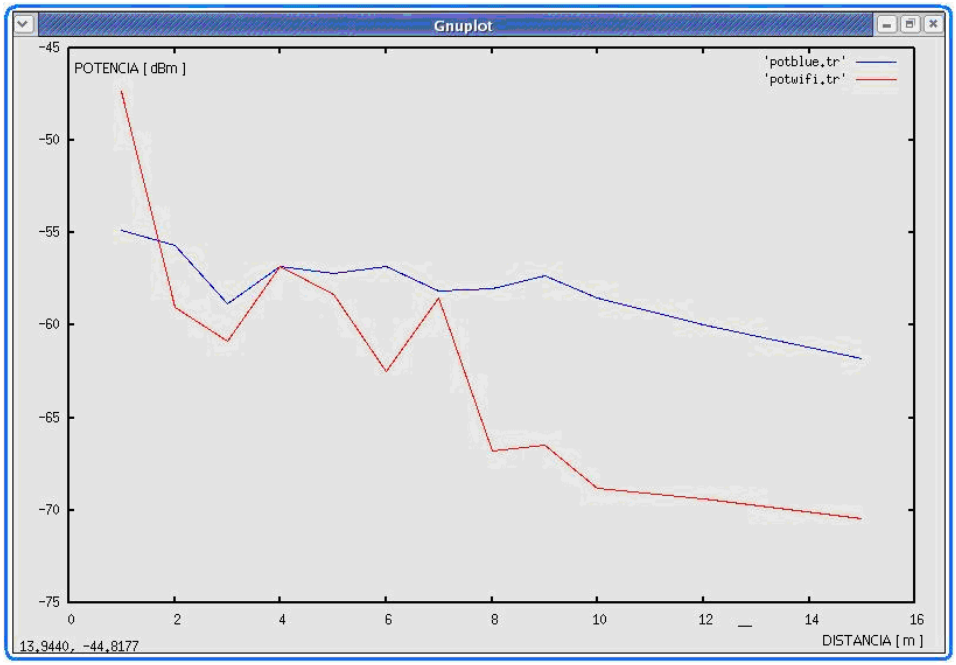


Figura 4.20 Potencia Práctica *Bluetooth* y *Wi-Fi*

En la figura 4.21 se puede observar que *Wi-Fi* tiene una mayor velocidad de transmisión que *Bluetooth*. Esto se debe ya que *Wi-Fi* transmite a una velocidad de 11 *Mbps* mientras que *Bluetooth* transmite a una velocidad máxima de 723 *Kbps.*

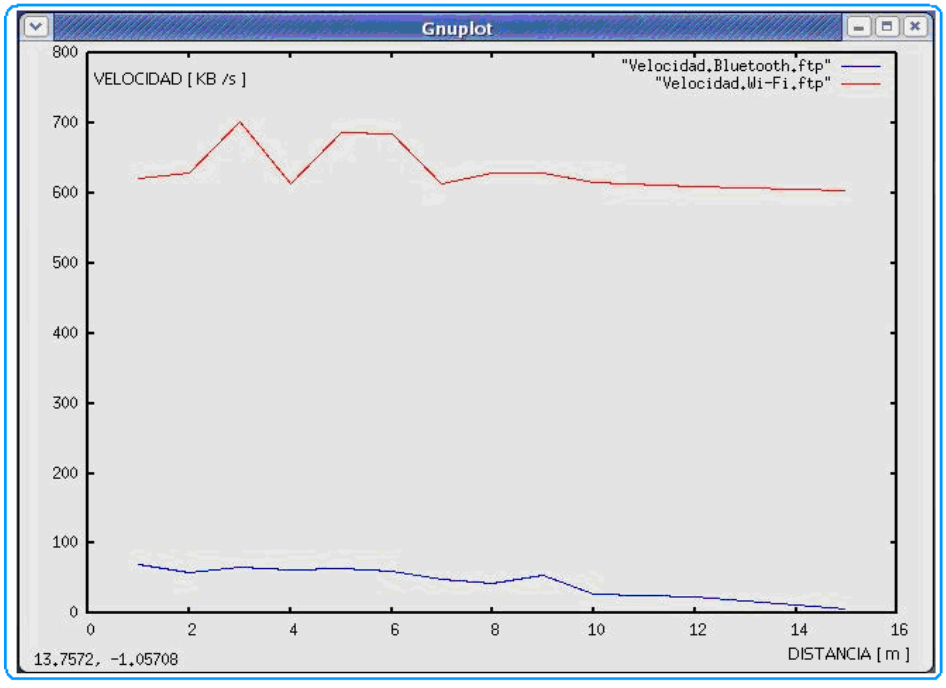


Figura 4.21 Velocidad Promedio *Bluetooth* y *Wi-Fi*

4.2.2.4 Análisis de resultados de las pruebas prácticas

Luego de realizar las pruebas y analizar los resultados obtenidos en la implementación de los prototipos se concluye que el prototipo *Bluetooth* es más estable que *Wi-Fi*, ya que la señal de potencia del prototipo *Bluetooth* sufre menos atenuaciones que *Wi-Fi.*

En el prototipo *Bluetooth* no existen pérdidas de datos lo que hace que este prototipo sea confiable al momento de transmitir datos, mientras que el prototipo *Wi-Fi* tiene pérdidas de datos considerables a partir de los 10 metros.

.

Con respecto a la velocidad de transmisión de datos, en el enlace con tecnología *Bluetooth* la velocidad varia a medida que aumenta la distancia, mientras que en el prototipo *Wi-Fi* la velocidad se mantiene estable.

4.3 PRUEBAS SIMULADAS

Mediante la simulación de los prototipos se trató de obtener las diferentes respuestas de potencia, señal a ruido, velocidad efectiva, lo más cercano a la implementación práctica a medida que el simulador permite. Debido a las limitaciones existentes para la simulación de *Bluetooth* la distancia varía desde los 2 m hasta los 12 m para ambos casos. A continuación se presenta los resultados de la simulación.

4.3.1 BLUETOOTH

La figura 4.22 indica la variación de la potencia en función de la distancia, aquí se observa que a medida que se alejan las estaciones la potencia disminuye tal como ocurre en la realidad.

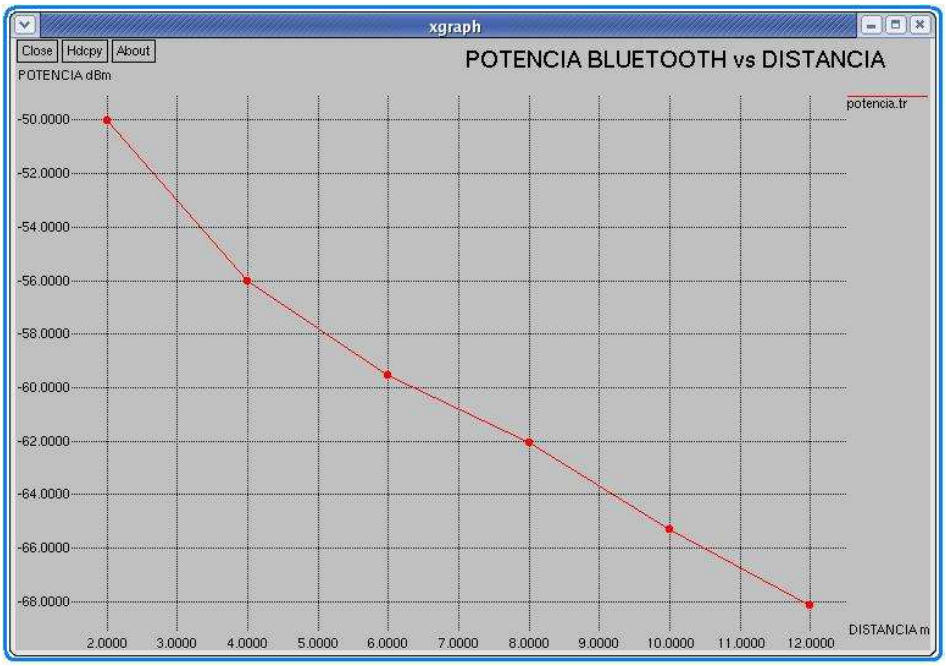


Figura 4.22 Potencia *Bluetooth* de la Simulación

La figura 4.23 representa la variación de la relación señal a ruido de la simulación en función de la distancia, a medida que se alejan las estaciones la relación señal a ruido decrece.

Figura 4.23 Señal a Ruido *Bluetooth* de la Simulación

La figura 4.24 representa la variación de la velocidad efectiva en función de la distancia, en este gráfico se puede observar valores máximos y mínimos de la velocidad esta variación depende de la cola del buffer, es decir si existe congestión en el enlace se emite un mensaje para detener el envió momentáneo de datos.

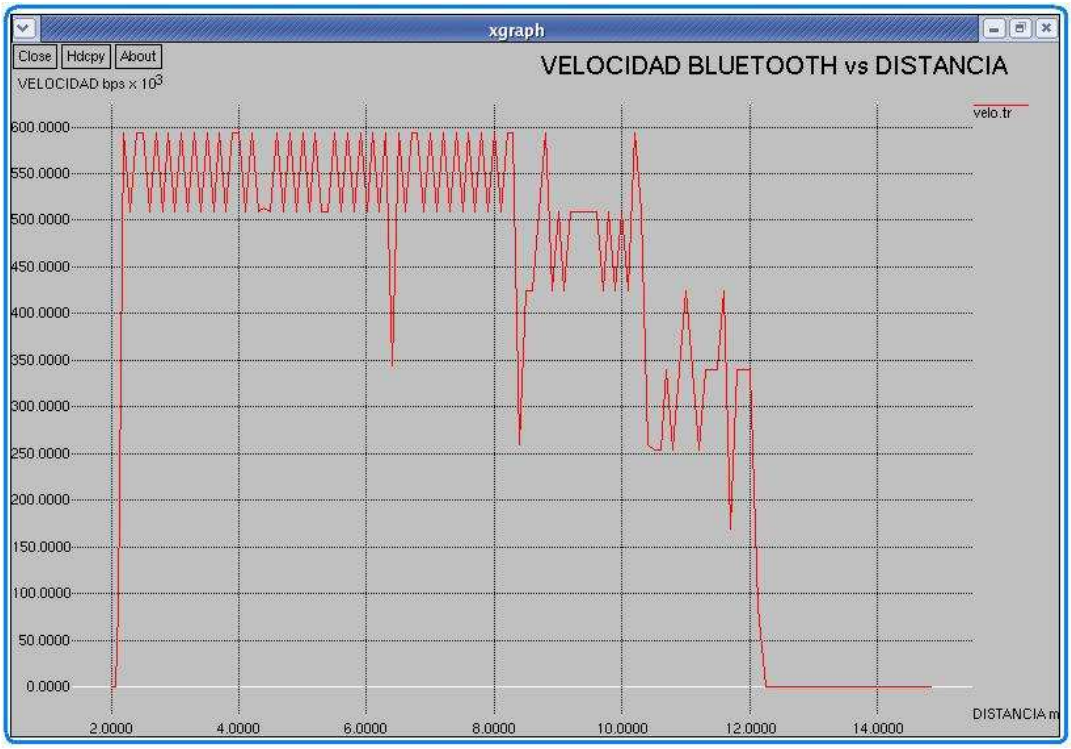


Figura 4.24 Velocidad *Bluetooth* de la Simulación

4.3.2 WI-FI

La figura 4.25 representa la variación de la potencia en función de la distancia, aquí se observa que a medida que se alejan las estaciones la potencia disminuye tal como ocurre en la realidad.

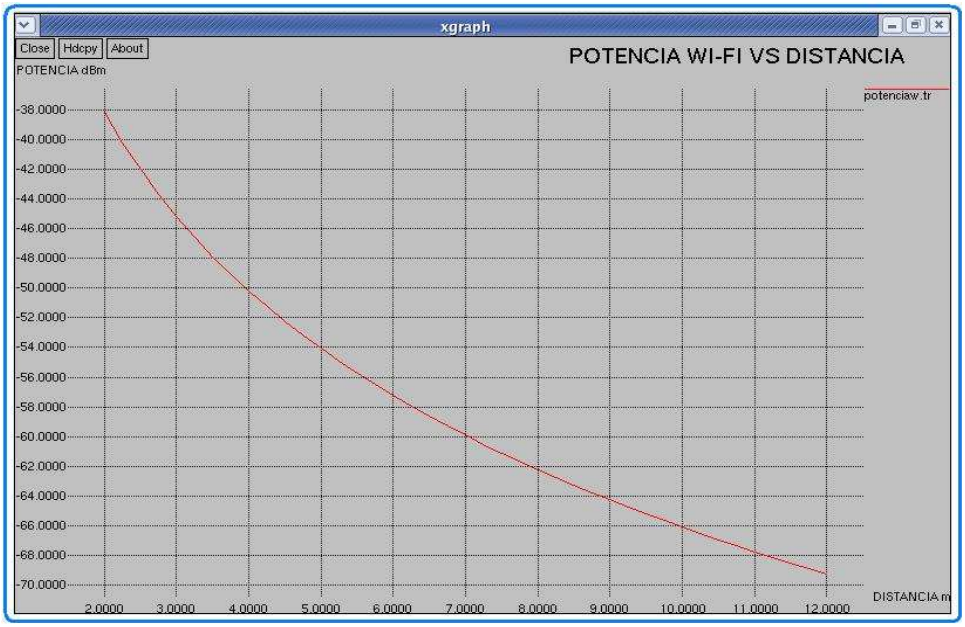


Figura 4.25 Potencia *Wi-Fi* de la Simulación

La figura 4.26 indica la variación de la relación señal a ruido de la simulación en función de la distancia para el prototipo *Wi-Fi,* este resultado depende de la distancia de separación de los dispositivos y del ruido existente

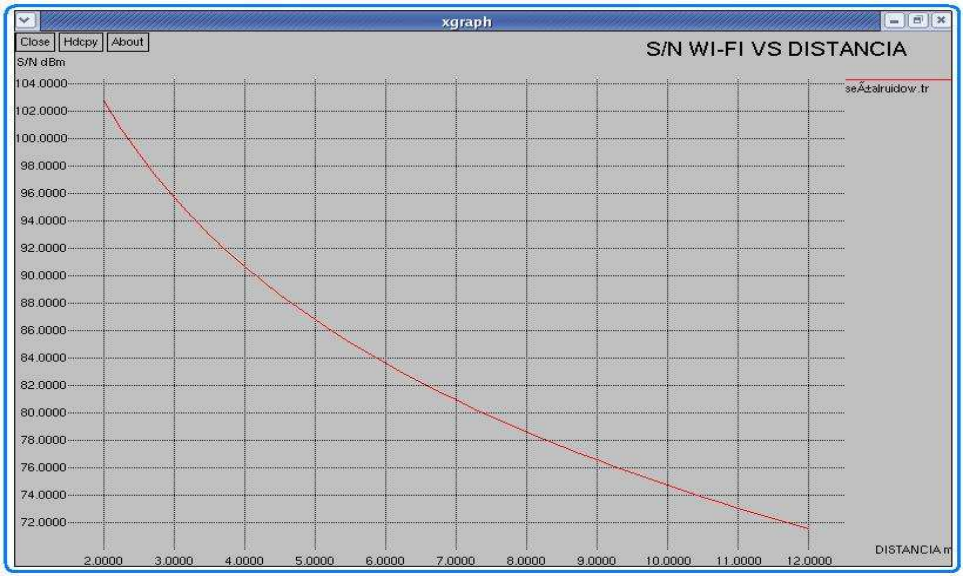


Figura 4.26 Señal a ruido *Wi-Fi* de la Simulación

En la figura 4.27 representa la variación de la velocidad efectiva de *Wi-Fi* en función de la distancia, los valores máximos y mínimos en la velocidad dependen de la cola del *buffer*, es decir si existe congestión en el enlace se emite un mensaje para detener el envío momentáneo de datos.

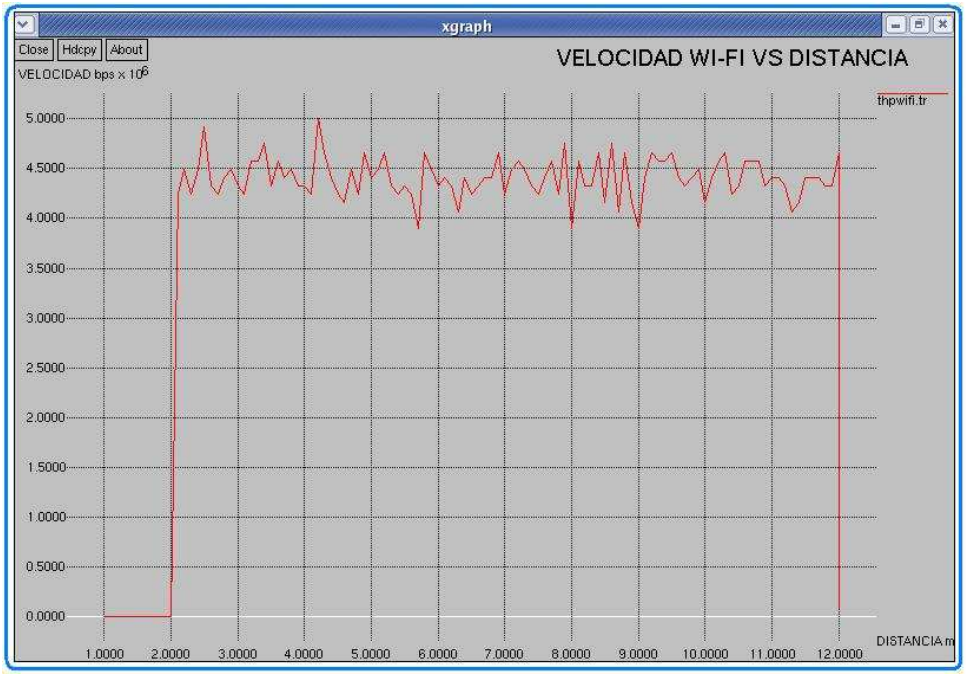


Figura 4.27 Velocidad Efectiva *Wi-Fi* de la Simulación

4.3.3 COMPARACIÓN GRÁFICA DE LAS SIMULACIONES

La figura 4.28 representa la comparación entre *Bluetooth* y *Wi-Fi* con respecto a la potencia, como se puede ver *Wi-Fi* posee niveles de potencia más altos que *Bluetooth* en distancias cortas, pero a medida que se alejan las estaciones *Bluetooth* se comporta de mejor manera.

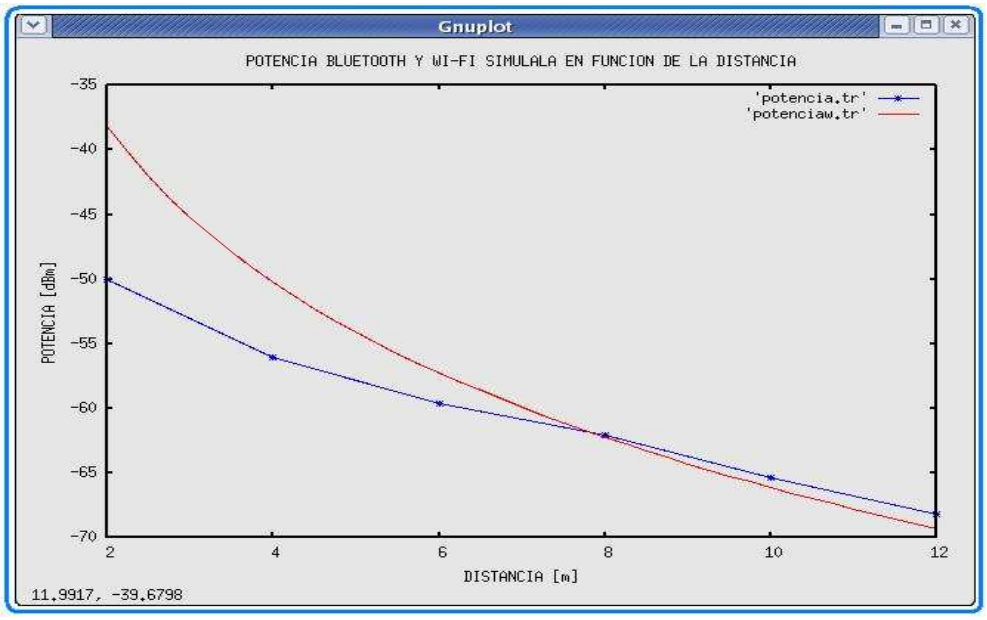


Figura 4.28 Potencia *Bluetooth* y *Wi-Fi* de la Simulación

La figura 4.29 representa la comparación de la relación señal a ruido entre *Bluetooth* y *Wi-Fi*, aquí se observa que *Bluetooth* tiene una mejor relación señal a ruido que *Wi-Fi.*

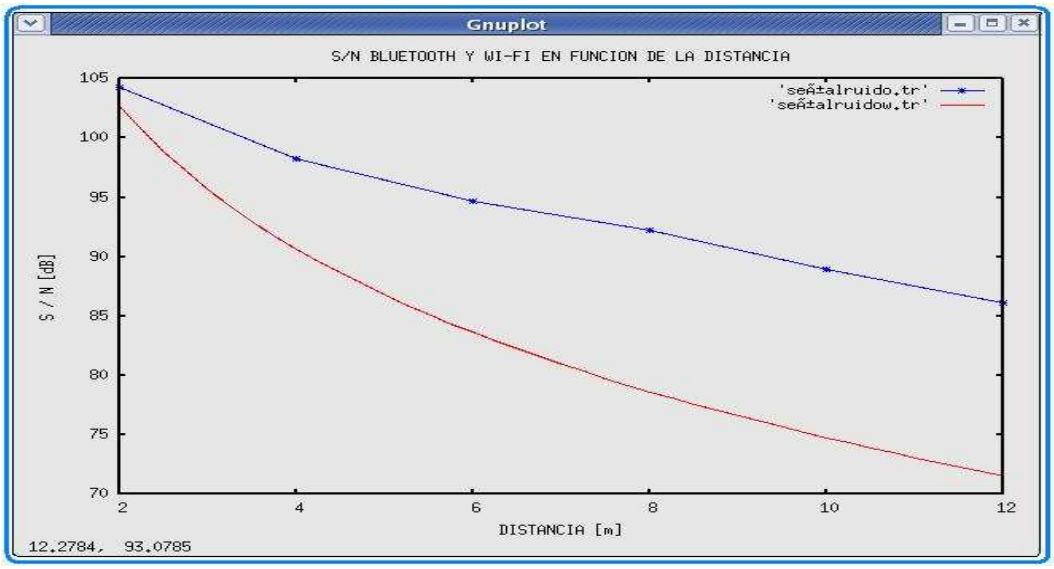


Figura 4.29 Señal a ruido *Bluetooth* y *Wi-Fi* simuladas

La figura 4.30 indica la comparación entre *Bluetooth* y *Wi-Fi* con respecto a la velocidad efectiva, aquí se puede ver claramente que *Wi-Fi* posee una mayor tasa de transferencia que *Bluetooth*, como ocurre en la práctica.

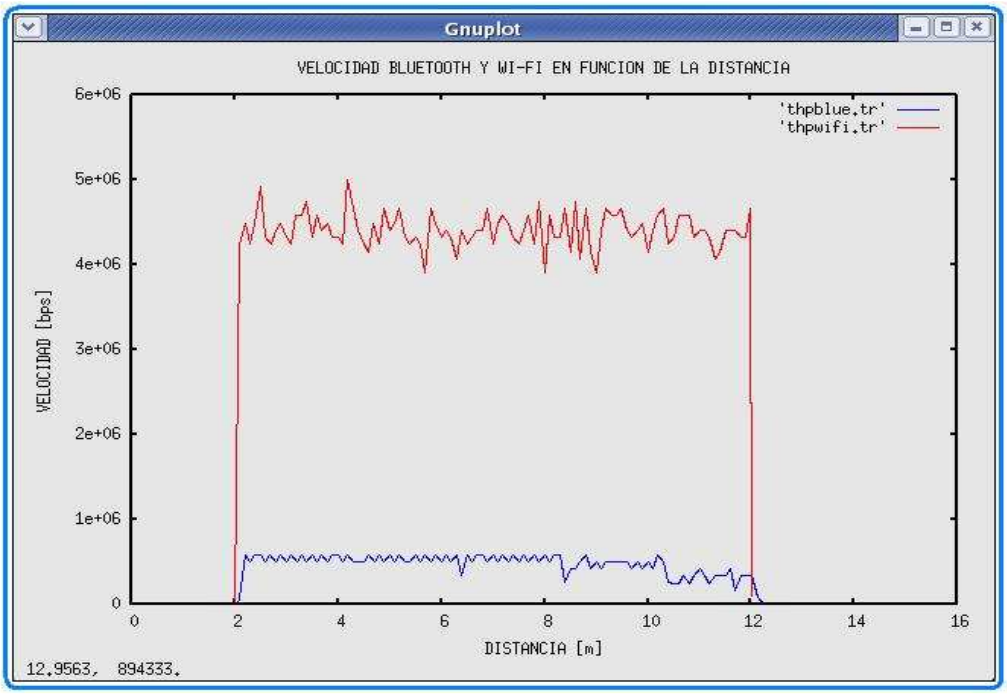


Figura 4.30 Velocidad *Bluetooth* y *Wi-Fi* simuladas

4.4 COMPARACIÓN PRUEBAS PRÁCTICAS CON SIMULADAS

A continuación se realiza una comparación gráfica entre las pruebas prácticas y simuladas en función de la distancia, desde los 2 metros hasta el alcance máximo de *Bluetooth* en el simulador que es de 12 m.

La figura 4.31 representa la potencia práctica y simulada para *Bluetooth,* como se puede apreciar la potencia simulada se asemeja a un decrecimiento logarítmico ya que por ser la simulación, ésta no va a ser inestable como en el caso práctico en el cual la señal está variando debido a que ésta depende del ambiente externo, como temperatura, objetos que se encuentran en el medio a más de otros ambientes inalámbricos.

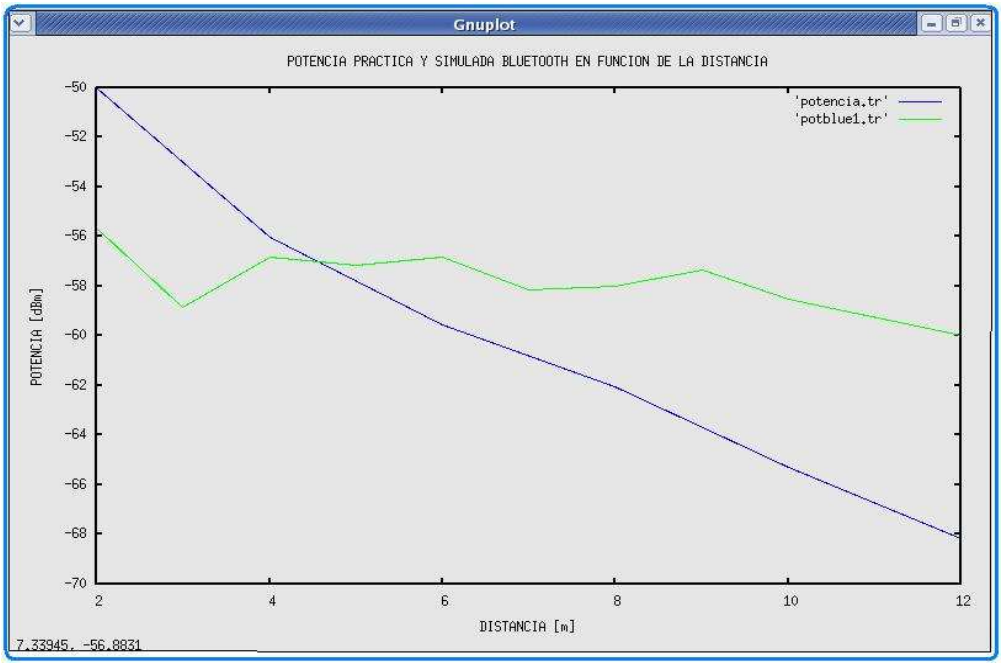


Figura 4.31 Potencia Práctica y Simulada *Bluetooth*

La figura 4.32 representa la velocidad práctica y simulada para *Bluetooth*, se puede observar claramente que tanto la velocidad práctica como la simulada varían con respecto a la distancia.

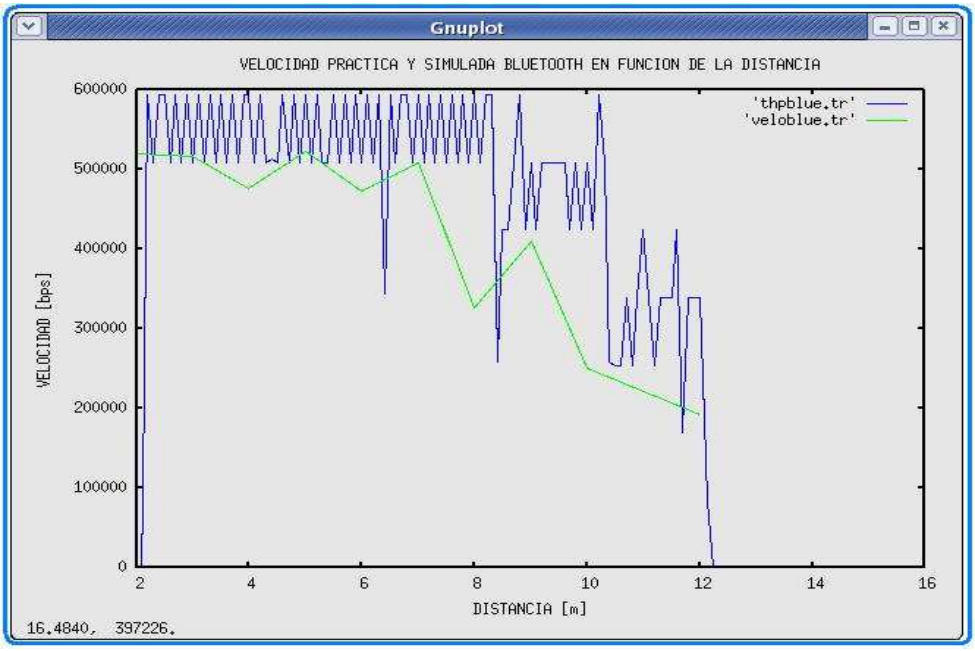


Figura 4.32 Velocidad Práctica y Simulada *Bluetooth*

La figura 4.33 representa la potencia práctica y simulada para *Wi-Fi,* como se puede apreciar la potencia simulada tiene un decrecimiento logarítmico ya que por ser la simulación esta no va a ser inestable como en el caso práctico en el cual la señal esta variando debido a que ésta depende el ambiente externo, como temperatura, objetos que se encuentran en el medio a más de otros ambientes inalámbricos.

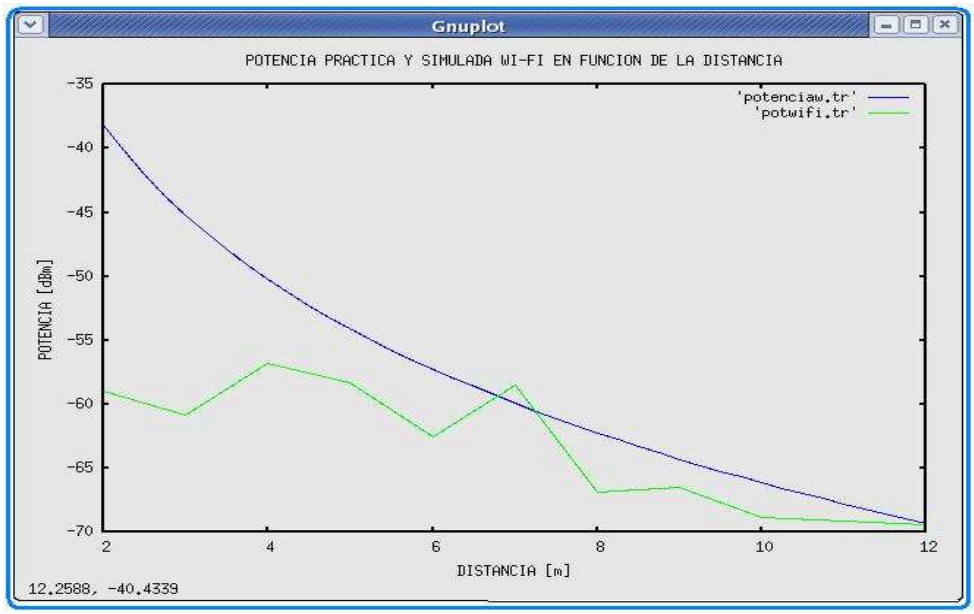


Figura 4.33 Potencia Práctica y Simulada *Wi-Fi*

La figura 4.34 representa las velocidades práctica y simulada *Wi-Fi*, se puede ver claramente que la velocidad simulada se mantiene en un rango aceptable, mientras que la velocidad práctica se encuentra variando en función de la distancia. También se observa que la velocidad simulada es similar a la práctica.

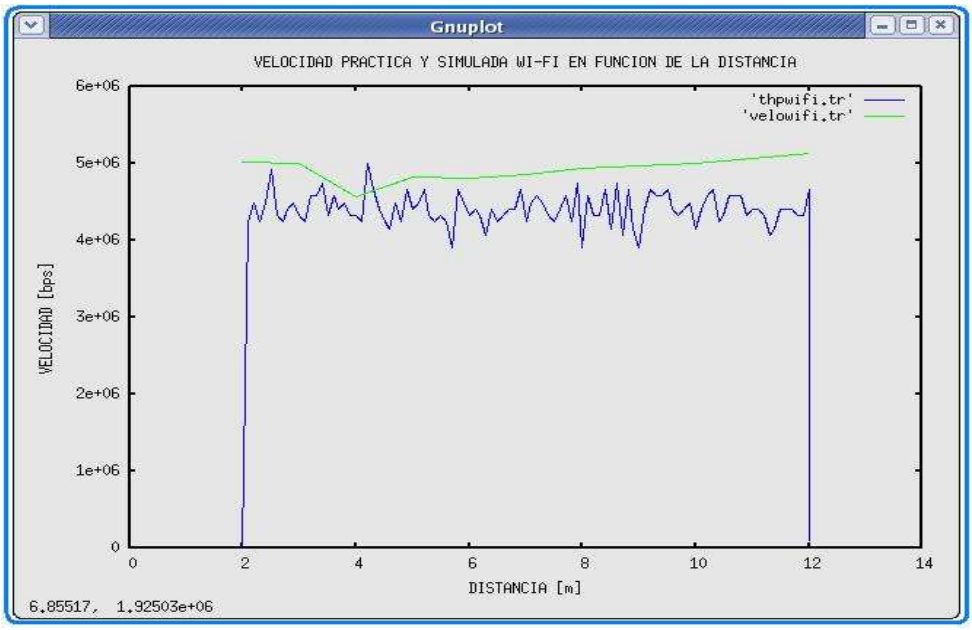


Figura 4.34 Velocidad Práctica y Simulada *Wi-Fi*

CAPÍTULO 5

CONCLUSIONES Y RECOMENDACIONES

5.1 CONCLUSIONES

- Hoy en día la tecnología *Bluetooth* está siendo adoptada por un buen número de fabricantes de *hardware*, ganando cada vez mayor aceptación y penetración en el mercado, que trae como consecuencia la disminución de sus costos de comercialización, lo cual beneficia al usuario final de esta tecnología.

- *Bluetooth* a más de prescindir de los tradicionales y molestos cables empleados para conectar dispositivos digitales entre sí (computadores de escritorio, *PC* portátiles, impresoras, teléfonos móviles, *PDAs*, etc.), permite la conformación de grupos cerrados de usuarios de manera dinámica, este tipo de usuarios operan en redes con infraestructura no fijas, y proporciona una interfaz universal que permite la interoperabilidad entre los diferentes dispositivos, gracias al carácter abierto de la especificación, por todas estas bondades mencionadas se puede decir que *Bluetooth* es un sistema revolucionario, comparado con los sistemas de comunicaciones inalámbricos existentes en la actualidad.

- Una de las principales desventajas que posee *Bluetooth,* frente a otras tecnologías inalámbricas es su alcance, velocidad de transmisión de datos, ya que esta tecnología fue creada para trabajar en ambientes de redes *WPAN.*

- La gran ventaja que posee *Bluetooth,* sobre otras tecnologías es que está diseñada para entregar servicio inalámbrico a dispositivos de gran movilidad, de reducido tamaño y bajo consumo de potencia que les proporcionen portabilidad e independencia de una fuente de energía fija; dispositivos de tipo periférico o que a su vez trabajen en ambientes de

corto alcance. Por esta razón se puede decir que las aplicaciones de esta tecnología es inmensa y que su límite se encuentra en la imaginación del que crea el sistema inalámbrico.

- Una de las principales limitaciones que actualmente posee *Wi-Fi*, es que ésta permite la interoperabilidad únicamente entre computadores de escritorio con interfaces inalámbricas *Wi-Fi 802.11 b,g* y *PC* portátiles que posean esta interfaz.

- *Wi-Fi* ofrece mayor cobertura y velocidades de transmisión relativamente altas. Por todo esto se puede decir que *Wi-Fi* lleva una ligera ventaja, frente a otras tecnologías inalámbricas, otra de las ventajas que posee *Wi-Fi* es que esta tecnología permite expandir una red cableada o a su vez reemplazarla.

- Luego de realizadas las pruebas prácticas de *Bluetooth* y *Wi-Fi* se puede concluir que en ambientes pequeños la señal de potencia de *Bluetooth* sufre menor atenuación que *Wi-Fi*, también se puede decir que en cuanto a estabilidad, pérdida de datos, *Bluetooth* tiene un mejor desempeño que *Wi-Fi*.

- En cuanto al consumo de potencia, las interfaces *Bluetooth* tiene menor consumo que las interfaces *Wi-Fi*, por este motivo se puede decir que las interfaces *Bluetooth* son aptos para ser usados en *PCs* Portátiles que necesitan accesorios externos como Mouse, permitiendo así una mayor duración de la batería y otorgando así mayor tiempo de duración de trabajo sin necesidad de alimentación externa.

- *Bluetooth* presenta excelente robustez frente al ruido, lo que hace que los sistemas que trabajen con esta tecnología, sean más estables y seguros que los demás, pero por su baja velocidad de transmisión de datos hace que en la actualidad *Bluetooth* no sea muy utilizado para redes inalámbricas de corto alcance.

- Se pudo comprobar prácticamente para *Bluetooth* y para *Wi-Fi* que los parámetros de campo como; velocidad efectiva, nivel de potencia y relación señal a ruido varían directamente en función de la distancia, es decir a medida que aumentaba la distancia de separación entre las estaciones estos parámetros decrecen.

- Como era de esperarse en el enlace *Wi-Fi* a medida que se alejan las estaciones éste presenta mejor velocidad efectiva que *Bluetooth* ya que el estándar *802.11b* se caracteriza por tener mayor alcance, velocidad, que el estándar *802.15.1*

- Se pudo comprobar que las tarjetas *Wi-Fi* utilizados en el proyecto, no cumplen con el alcance especificado en las mismas, motivo por el cual estas tarjetas son muy inestables, a partir de distancias mayores a 14 metros. Esto se debe a la baja sensibilidad de las tarjetas, pues en las pruebas de medición de potencia realizadas con el analizador de espectros se observa que la potencia transmitida por el interfaz llega al receptor con un valor aceptable mayor a la sensibilidad especificada por el interfaz.

- Para la simulación de los prototipos, se utilizó el programa *Network Simulator*, que además de ser un *software* muy utilizado en propósitos similares (redes satelitales, redes inalámbricas *Ad-Hoc* e Infraestructura, etc.) por diversos organismos en casi todo el mundo que se dedican a la investigación de redes, se puede afirmar que este simulador es muy confiable al momento de realizar las distintas simulaciones pero tiene la desventaja de ser complicado al momento de instalarlo y realizar las distintas simulaciones.

- En la simulación se pudo verificar que el enlace *Bluetooth* es más robusto que *Wi-Fi* frente al ruido. Esto se debe ya que *Bluetooth* utiliza un ancho de banda pequeño con respecto a *Wi-Fi* y es más inmune a la interferencia.

5.2 RECOMENDACIONES

- Una vez culminado el proyecto y realizadas las distintas pruebas que abarcaron el desarrollo del mismo, se recomienda implementar redes inalámbricas con tecnología *Bluetooth* en oficinas, lugares cerrados en general que tengan un diámetro no mayor a diez metros y en donde la velocidad de transmisión no sea importante, ya que Bluetooth en distancias menores o iguales a esta, tiene un buen desempeño en lo que corresponde a nivel de potencia, pérdida de datos.

- También se recomienda utilizar la tecnología *Bluetooth* en sistemas inalámbricos en los cuales la pérdida de datos deba ser mínima al momento de transmitir, ya que *Bluetooth* presenta una pérdida de datos mínima. Como es el caso de sensores inalámbricos.

- Una recomendación final con respecto a la implementación de sistemas inalámbricos con tecnología *Bluetooth*, es que *Bluetooth* en la actualidad puede interactuar con dispositivos móviles, celulares, impresoras, *mouse*, *PDAs*, etc. por lo que es recomendable implementar esta tecnología en lugares que posean este tipo de dispositivos.

- Se recomienda utilizar *Wi-Fi* en sistemas inalámbricos en los cuales no sea importante la pérdida de datos al momento de la transmisión de estos y la tasa de transferencia sea alta, ya que esta tecnología posee una pérdida de datos considerable a distancias mayores a los doce metros, pero a su vez posee alta velocidad de transmisión de datos.

- Se recomienda descargar el paquete completo del programa allinone del ns e instalarlo en su totalidad, para evitar futuros problemas en el momento de ejecutar sus librerías. Se puede obtener el programa simple e ir instalando las librerías una por una de acuerdo a las necesidades que se tenga.

- Se recomienda en un proyecto futuro se modifiquen las librerías del *UCBT* para lograr el posicionamiento y movimiento de los nodos ya que en el desarrollo de este trabajo se logró posicionar los nodos pero no dar movimiento a estos.

BIBLIOGRAFÍA

[1] Jorge Alfonso Briones García, “Un *middleware* para el desarrollo de aplicaciones en redes espontáneas de dispositivos móviles Bluetooth”, Tesis de Maestría, Centro de Investigación y de Estudios Avanzados del IPN Departamento de Ingeniería Eléctrica Sección de Computación.

[2] Néstor García Fernández, “Modelo de Cobertura en Redes Inalámbricas basado en Radiosidad por Refinamiento Progresivo”, Tesis Doctoral, Universidad de Oviedo, marzo 2006

[3] Msc. Soraya Sinche M. “Apuntes de Comunicaciones Inalámbricas”, Escuela Politécnica Nacional, Marzo-Agosto 2005.

[4] PhD. Iván Bernal,” Apuntes de Comunicaciones Inalámbricas”, Escuela Politécnica Nacional, Diciembre 2005

[5] BLUETOOTH SPECIAL INTEREST GROUP. “Bluetooth Core”, Specification of the Bluetooth System, Version 1.1, 22 de febrero de 2003

[6] Carlos García García, Doctorado PCSM “Protocolo de Comunicaciones para Sistemas Móviles”

[7] Ing. Pablo Hidalgo, “Apuntes de Telemática”, Escuela Politécnica Nacional Marzo 2005

[8] Mari Carmen Domingo, “Diferenciación de servicios y mejora de la supervivencia en redes Ad-hoc conectadas a redes fijas”, Tesis Doctoral, Universidad Politécnica de Cataluña, Año 2006

[9] Andrew S. Tanenbaum,”Redes de Computadoras”, Tercera edición, Prentice-Hall, páginas 263-265, 1997.

[10] Wayne Tomasi, “Sistemas de Comunicaciones Electrónicas”, Pearson Educación de México, Cuarta Edición 2003

[11] Msc. Maria Soledad Jiménez Jiménez,” Apuntes de Comunicación Digital”, Escuela Politécnica Nacional, 2004.

[12] A.J. Motley, J.M. Keeman, “Radio Coverage in Buildings”. British Telecom, Technology Journal, Vol. 8, Enero 1990.

[13] Oscar Darío Rodríguez Calvachi y Ricardo Andrés Maya Coral Implementación de una red Inalámbrica Bluetooth”, Tesis, Universidad del Valle, Santiago de Cali, 2003.

[14] Roberto Carlos Ortega y Wladimir Valdivieso, Diseño e Implementación de un acceso inalámbrico e interfaz al sistema de administración estudiantil (SAE) y biblioteca, basado en la tecnología Bluetooth, ubicado en el edifico antiguo de Ingeniería Eléctrica

[15] http://www.mastermagazine.info/articulo/3125.php

[16] http://www:bluetooth.com/Bluetooth/Learn/

[17] http://www.zonabluetooth.com/que_es_bluetooth2.htm

[18] http://www:catarina.udlap.mx/capitulo4.pdf

[19] http://www.senacitel.cl/downloads/senacitel2000/IDO33.pdf

[20] http://www.di.uniovi.es/investigacion/tesis/Nestor.pdf

[21] http://www.isi.edu/nsnsm/ns/.

[22] http://www.isi.edu/nsnam/ns/tutorial/index.html

[23] http://www.nile.wpi.edu/NS/

[24] http://www.bluetooth.com/Bluetooth/Learn/Security/

[25] http://www.haking9.org

[26] http://www.Wireless LAN Alliance

[27] http://www.eveliux.com

[28] http://www.isi.edu/nsnam/ns/ns-documentation.html

[29] http://www.ns-allinone-2.1b8a/ns-2.1b8a/tcl/lib

[30] http://www.expansys.es

[31] http://www.isi.edu/nsnam/

[32] http://www.ececs.uc.edu/~cdmc/ucbt/

[33] http://www.bluetooth.com/sig/membership/membership.asp

[34] http://bluehack.elhacker.net/proyectos/bluesec/bluesec.html

[35] http:// www.guw.cl/foros/

[36] INTCOMEX DEL ECUADOR S.A.

[37] http://www.dooyoo.es/targetas-de-red/anycom-blue-usb-120/precios/

[38] http://www.preciomania.com/search_attrib.php/vededorlds%5B%

[39] http://www.air802.com/home.php

[38] http://www.pixmania.com/es/es/597652/linksys/adaptador-usb-wifi-54-mb.html

GLOSARIO DE TÉRMINOS

GLOSARIO

1SM:	Modelo de una Sola Pendiente
ACL:	Enlace Asíncrono no Orientado a la Conexión
AP:	Punto de Acceso
AT- Commands:	Comandos de Telefonía
Backoff:	Espera aleatoria antes de transmitir un paquete
BD_ADDR:	Dirección del Dispositivo *Bluetooth*
BSS:	Conjunto de Servicios Básicos
CAC:	Código de Acceso al Canal
CCK	Modulación de Código Complementario
CFP:	Periodo Libre de Contención
CHIP:	Circuito Integrado
CID:	Identificador de Canal Lógico
CP:	Periodo de Contención
CRC:	Código de Redundancia Cíclica
CSMA/CA:	Acceso Múltiple con Escucha de Portadora Evitando Colisiones
CTS:	Limpieza de Envío
CW:	Ventana de Contienda
DAC:	Código de Acceso de Dispositivo
dBi:	Decibelios Isotrópicos
DCF:	Función de Coordinación Distribuida
DIFS:	Espaciado entre Tramas Distribuido
DS:	Sistema de Distribución
DS:	Envío de Datos
DSSS:	Espectro Expandido por Secuencia Directa
DBPSK	Modulación por Desplazamiento de Fase Bivalente Diferencial
DQPSK	Modulación por Desplazamiento de Fase Cuadrivalente Diferencial
EAP:	Protocolo de Autenticación Extensible
ESS:	Conjunto de Servicios Extendidos
ETSI:	Instituto de Estándares de Telecomunicaciones de Europa
FCC:	Comisión Federal de Comunicaciones
FEC:	Corrección Directa de Errores
FH:	Salto de Frecuencia

FHS:	Paquete de Sincronización
FHSS:	Espectro Expandido por Salto de Frecuencia
GFSK:	Modulación por Desplazamiento de Frecuencia Gausiana
GOEP:	Perfil Genérico de Intercambio de Objetos
HCI:	Interfaz de Controlador de Host
IAC:	Código de Acceso de Búsqueda
IBSS:	Independiente BSS
ID:	Paquete de Identificación
IEEE:	Instituto de Ingenieros Electrónicos y Eléctricos
IFS:	Espaciado entre Tramas
IP:	Protocolo de Internet
ISM:	Industrial, Científica y Médica
ITU:	Unión Internacional de Telecomunicaciones
IV:	Vector de Inicialización
IVA:	Impuesto de Valor Agregado
L2CAP:	Protocolo de Control y Adaptación de Enlace Lógico
LAN:	Red de Área Local
LAP:	Puntos de Acceso LAN
LC:	Control de Enlace
LEAP:	EAP Liviano
LLC:	Control de Enlace lógico
LM:	Enlace de Administración
LMP:	Protocolo de Administración del Enlace
M_ADDR:	Dirección que identifica a los esclavos activos en una *piconet*
MAC:	Control de Acceso al Medio
MACA:	Acceso Múltiple Evitando Colisiones
MSDU:	Unidad de Datos de Servicio MAC
NAV:	Vector de Localización de Red
NIC:	Tarjetas de Interfaz de Red
OBEX:	Protocolo para el Intercambio de Objetos
OFDM:	Multiplexación por División de Frecuencia Ortogonal
PC:	Computador Personal
PCF:	Función de Coordinación Puntual

PCs:	Computadora
PDA:	Asistentes Personales Digitales
PDU:	Unidades de Datos de Protocolo
PIM:	Información de Administración Personal
PIN:	Numero de identificación Personal
PPP:	Protocolo Punto a Punto
RADIUS:	Servicio de Usuario de Autenticación Remota
RF:	Radiofrecuencia
RFCOMM:	Protocolo para Emulación de Puertos Seriales sobre *L2CAP*
RTS:	Preguntar para Enviar
S/N:	Relación Señal a Ruido
SCO:	Enlace Sincrónico Orientado a Conexión
SDP:	Protocolo de Descubrimiento de Servicio
SIFS:	Espaciado entre Tramas Corto
SIG:	Grupo de Interés Especial
TCP:	Protocolo de Control de Transmisión
TCS:	Especificación de Control Telefónico
TDD:	Duplexación por División de Tiempo
TKIP:	Protocolo de Integridad de Claves Temporales
UA:	Datos Asíncronos de usuario
UDP:	Protocolo de Datagrama de Usuario
UI:	Datos Isócrono de Usuario
UMTS:	Sistema de Telecomunicaciones Móbiles Universal
UNII:	Infraestructura de Información Nacional sin Licencia
US:	Datos Síncrono de Usuario
vCard:	Visualizador de tarjetas virtuales
WAE:	Entorno Inalámbrico de Aplicación
WAP:	Protocolo de Aplicación Inalámbrica
WEP:	Privacidad Equivalente a la Cableada
Wi-Fi	Fidelidad Inalámbrica
WLAN:	Red de Área Local Inalámbrica
WPA:	Protocolo de Aplicación Inalámbrica

ANEXOS

ANEXO C

Módulos del ns-2 [21]

Módulos del ns-2

ns-2 dispone de varios módulos, a continuación se describirán los más importantes y aquellos que han sido utilizados en este proyecto.

- **Simulador ns**

Este módulo está desarrollado en C++, dispone de un núcleo principal, donde están definidos todos los protocolos. Por otra parte, los scripts donde se configuran los escenarios de la simulación se deben programar en el lenguaje OTcl, que al igual que C++ también es un lenguaje de programación orientado a objetos. Por lo tanto, para desarrollar las simulaciones en el ns-2 es necesario programar en los dos lenguajes.

C++ es más rápido al momento de ejecutar las simulaciones, pero tiene la desventaja que es más lento al momento de modificar que OTcl, entonces para ejecutar protocolos es mejor utilizar C++. Para la generación de los scripts es más fácil utilizar OTcl ya que este es más rápido y fácil de modificar.

El simulador se invoca introduciendo en la línea de comandos el siguiente comando.

./ns nom_fichero.tcl

Las simulaciones bajo ns-2 se basan en scripts programados por el propio usuario, se puede definir la topología de red y las características de los enlaces entre los nodos.

Los resultados obtenidos en la simulación se pueden visualizar utilizando las herramientas **nam** o **xgraph**.

- **Nam (*Network Animator*)**

Permite representar gráficamente la red diseñada, visualizar dinámicamente los resultados de la simulación realizada. Para ejecutar el nam se escribe en la línea de comandos el siguiente comando.

nam nombre_fichero.nam

En la figura 3.1 se puede ver la captura de un ejemplo de simulación realizada con el nam.

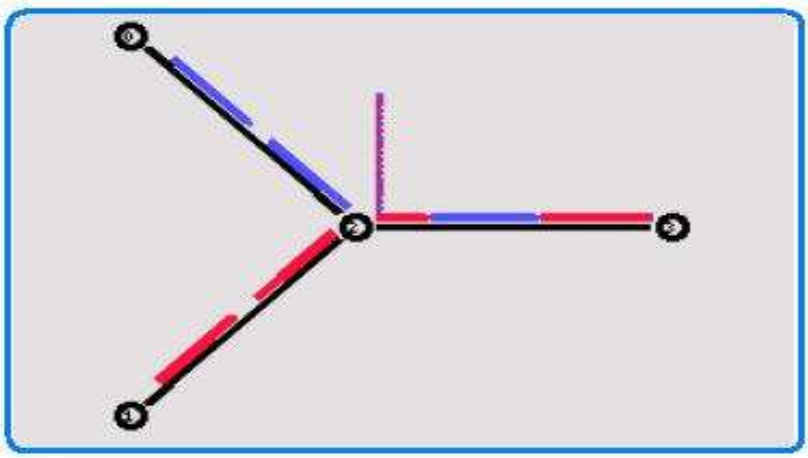


Figura 1 Simulación con nam

- **Xgraph**

Xgraph es una herramienta de graficación proporcionada por el ns-2. Además permite crear archivos postscript, Tgif, y otros. Puede ser invocado dentro de los comandos tcl para desplegar inmediatamente después de que la simulación haya concluido.

El comando xgraph espera como entradas uno o más archivos ASCII que contengan datos en forma de pares ordenados x-y en cada línea. Por ejemplo, xgraph f1 f2 imprimirá en la misma figura los archivos f1 y f2.
Algunas opciones que posee el xgraph son:

- Título: -t “título”

- Tamaño: -geometry xsize x ysize.
- Títulos de ejes: -x “xtitle” (para el título del eje x) y -y “ytitle” (para el título del eje y).
- Color de texto y grilla: con la bandera -v.

A continuación se muestra como se debe introducir el comando para realizar la graficación mediante el xgraf:

xgraph f1 f2 -geometry 800x400 -t “Loss rates” -x “time” -y “Lost packets”

Un ejemplo de uso del Xgraph, se indica en la figura 3.2, de acuerdo al siguiente código:

xgraph out.tr -geometry 800x400 -t "Ejemplo de uso del Xgraph" -x "tiempo" -y Paquetes perdidos

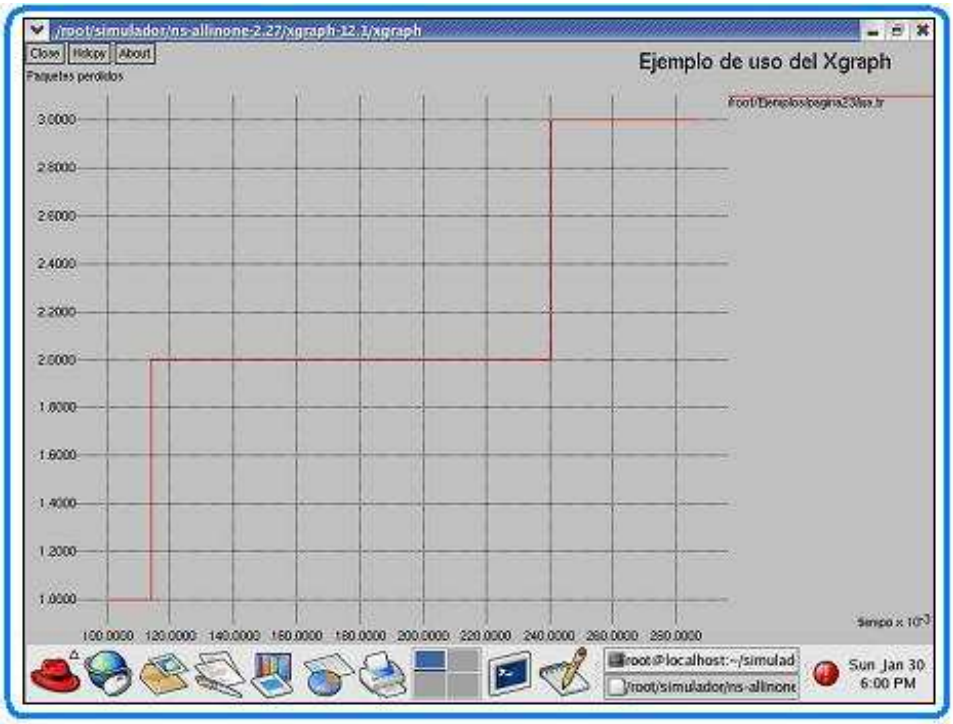


Figura 2 Ejemplo de uso del XGraph.

- **Gnuplot**

Gnuplot es un programa muy flexible creado para generar gráficas de funciones y datos. Este programa es compatible con muchos sistemas operativos como son

(*Linux, Unix, Windows,*). Gnuplot puede graficar sus resultados directamente en pantalla, así como en multitud de formatos de imagen, como PNG, EPS, SVG, JPEG, etc. Gnuplot se puede usar en forma interactiva o en modo por lotes usando scripts.

Para acceder al *gnuplot,* se ejecuta el siguiente comando.

Gnuplot
Plot nom_fichero.tr

La figura 3.3 representa en ejemplo de graficación utilizando gnuplot

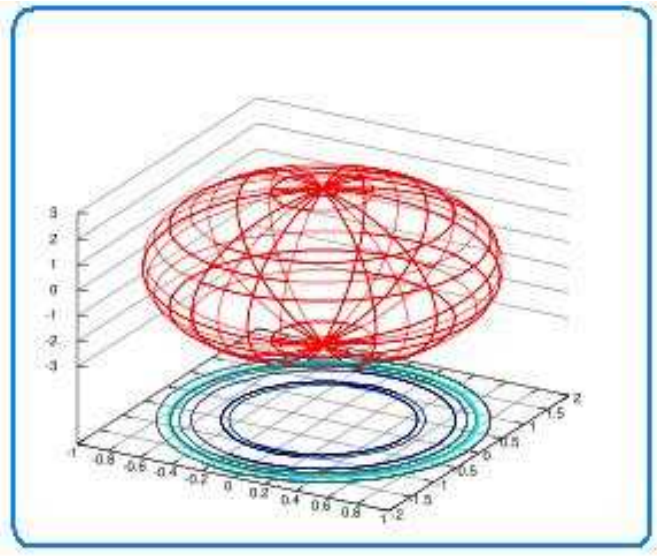


Figura 3 Ejemplo de gráfica utilizando gnuplot

ANEXO D

Pruebas Prácticas Bluetooth

Pruebas Prácticas de Bluetooth

Medidas a 2 metros de separación

Señal detectada

Figura 1 Estado de conexión 2m

Nivel de potencia (d = 2 m, p = - 55.67 dBm, f = 2.4747 GHz)

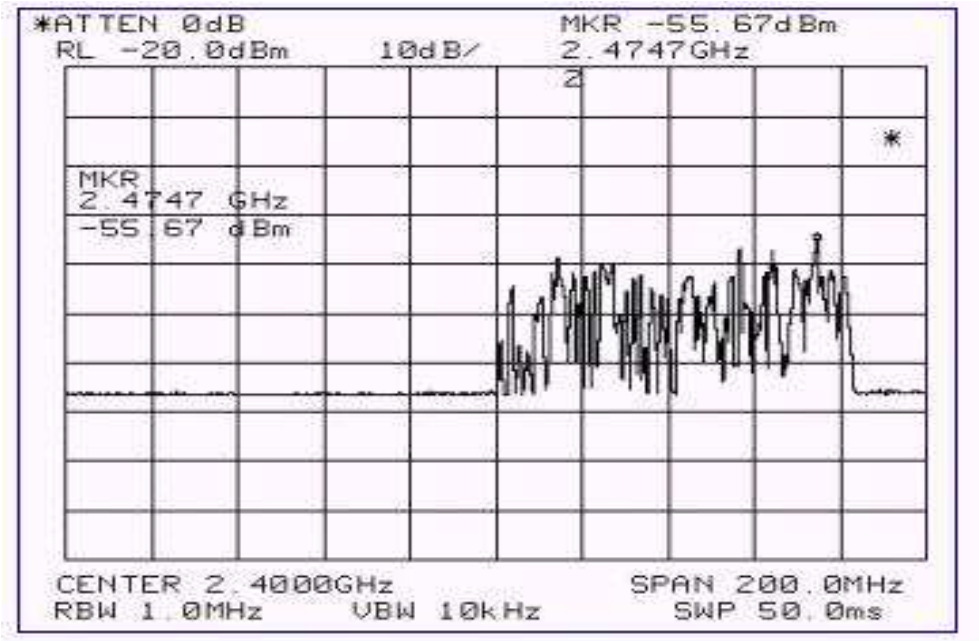


Figura 2 Nivel de potencia 2m

Respuesta entre las estaciones

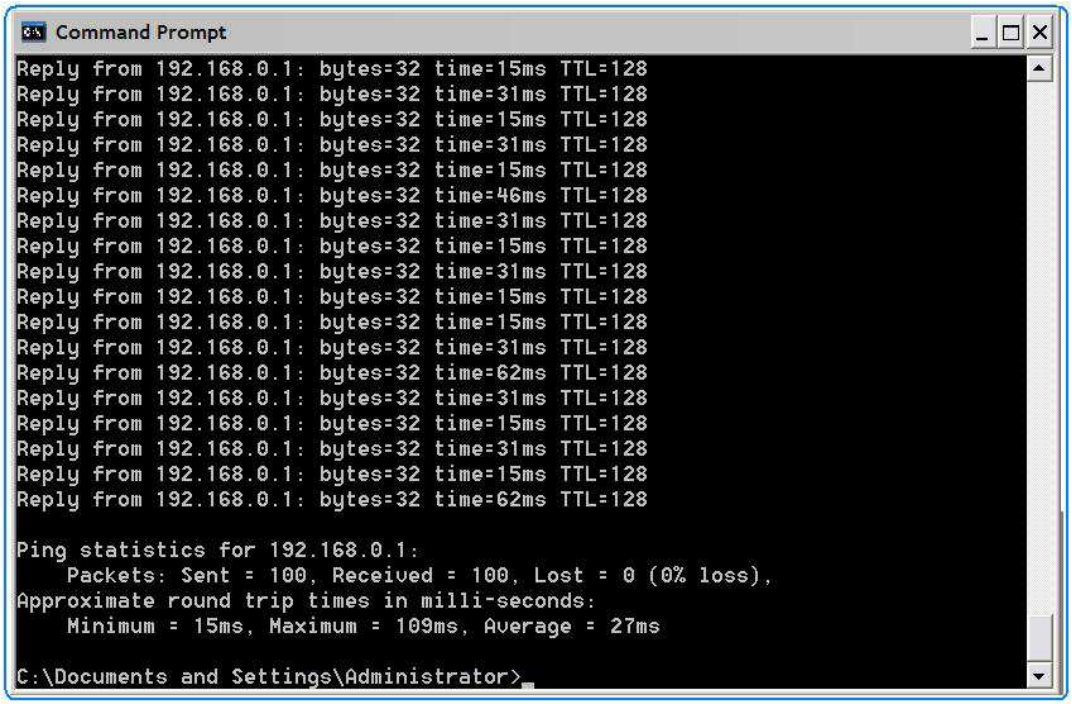


Figura 3 Ping entre las estaciones 2m

Velocidad

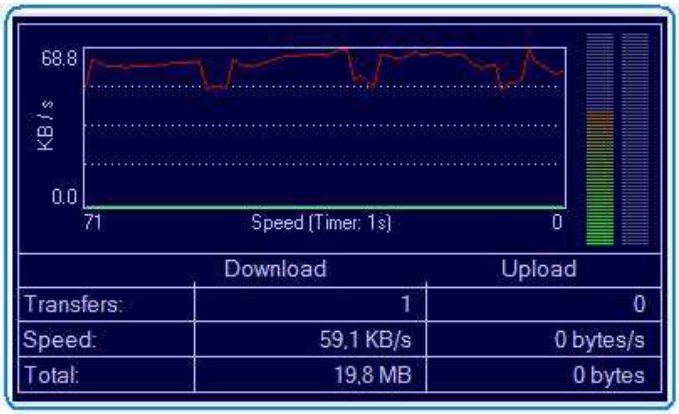


Figura 4 Velocidad 2m

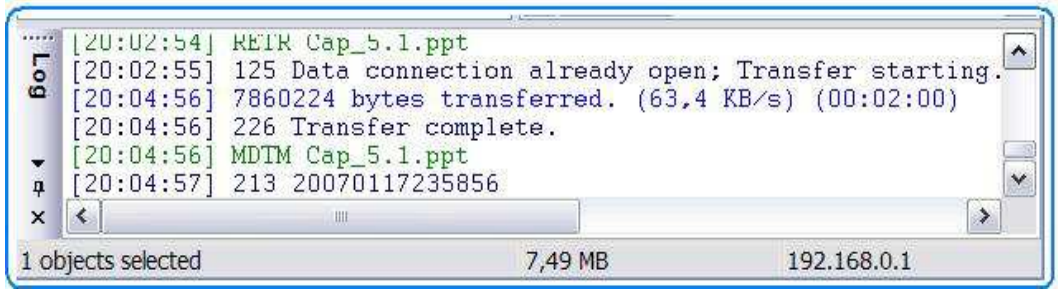


Figura 5 Velocidad Promedio 2m

Medidas a 3 metros de separación

Señal detectada

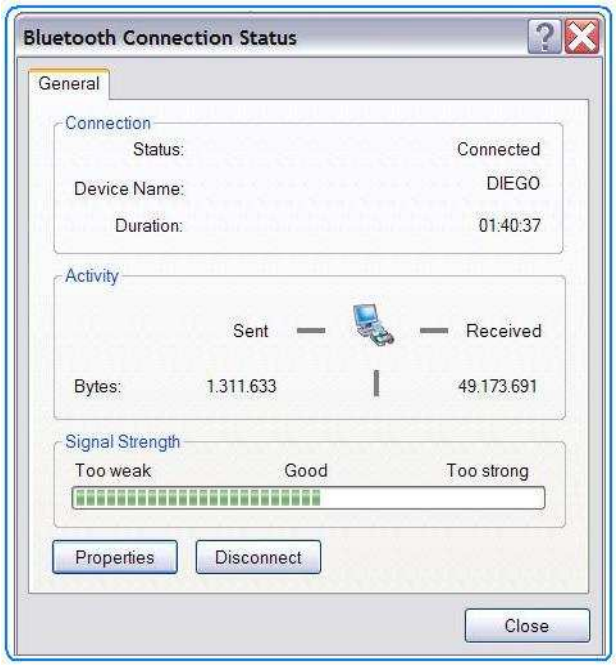


Figura 6 Estado de conexión 3m

Nivel de potencia (d = 3 m, p = - 58.83 dBm, f = 2.4740 GHz)

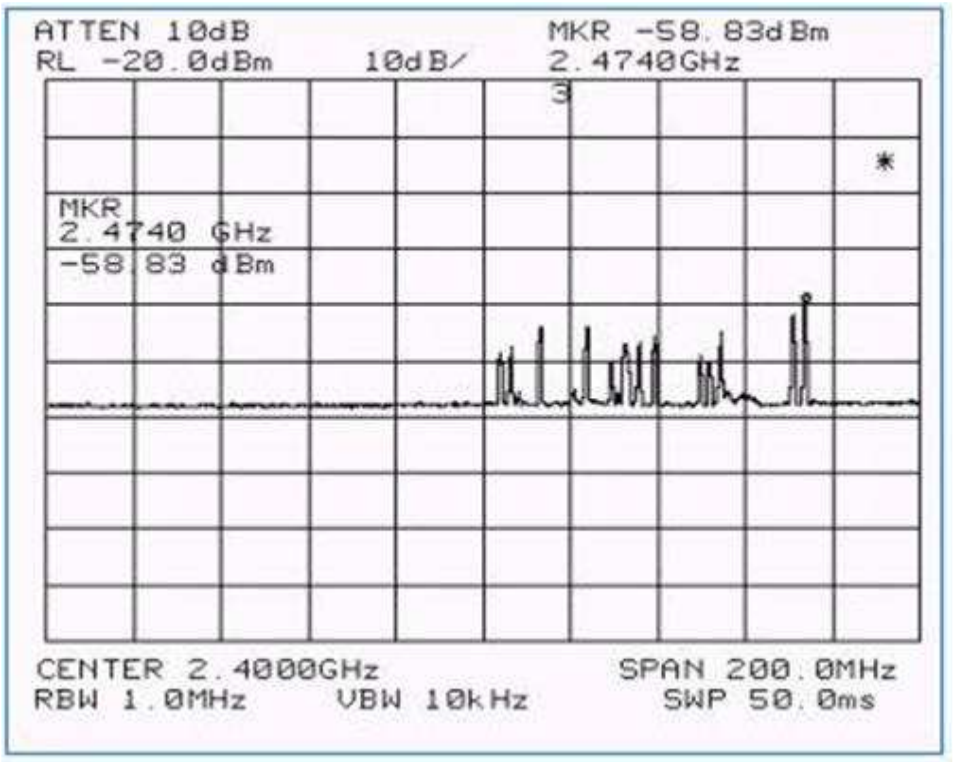


Figura 7 Nivel de potencia 3m

Respuesta entre las estaciones

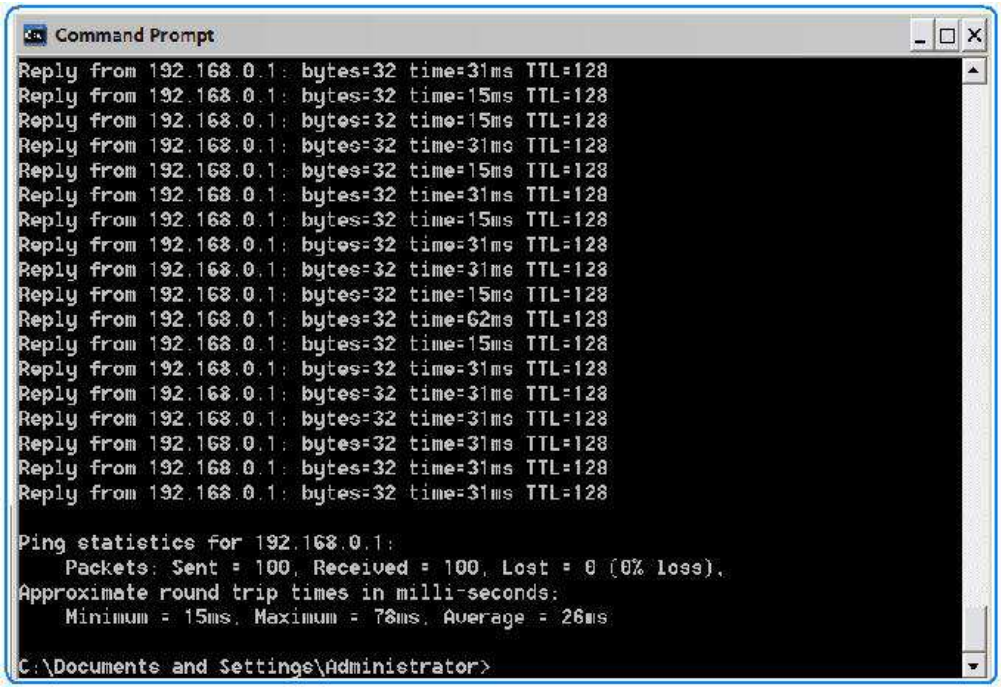


Figura 8 Ping entre las estaciones 3m

Velocidad

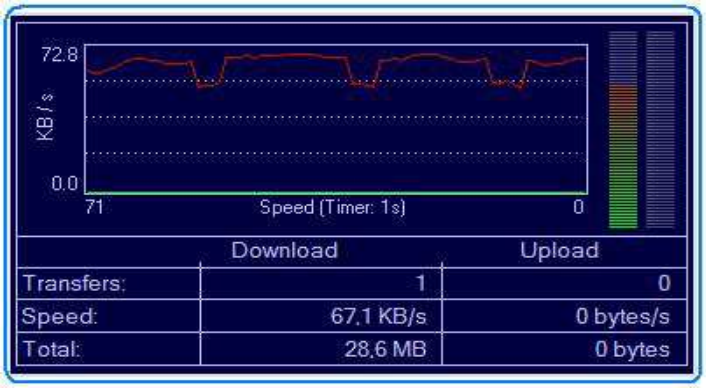


Figura 9 Velocidad 3m

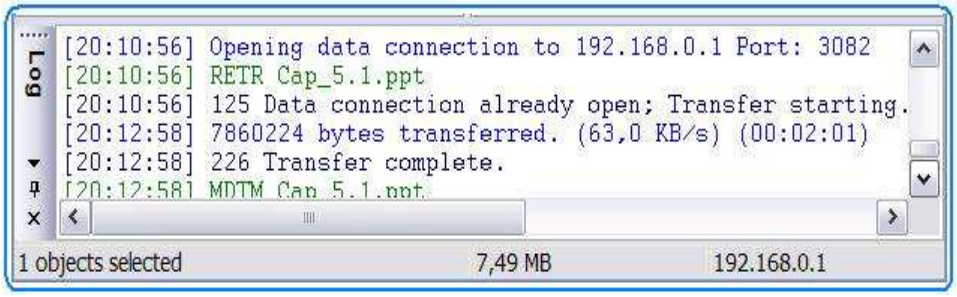


Figura 10 Velocidad Promedio 3m

Medidas a 4 metros de separación

Señal detectada

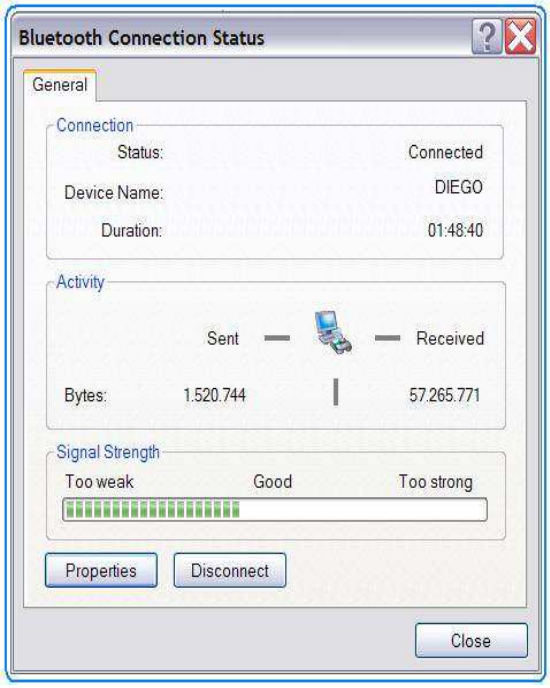


Figura 11 Estado de conexión 4m

Nivel de potencia (d = 4 m, p = - 56.83 dBm, f = 2.4537 GHz)

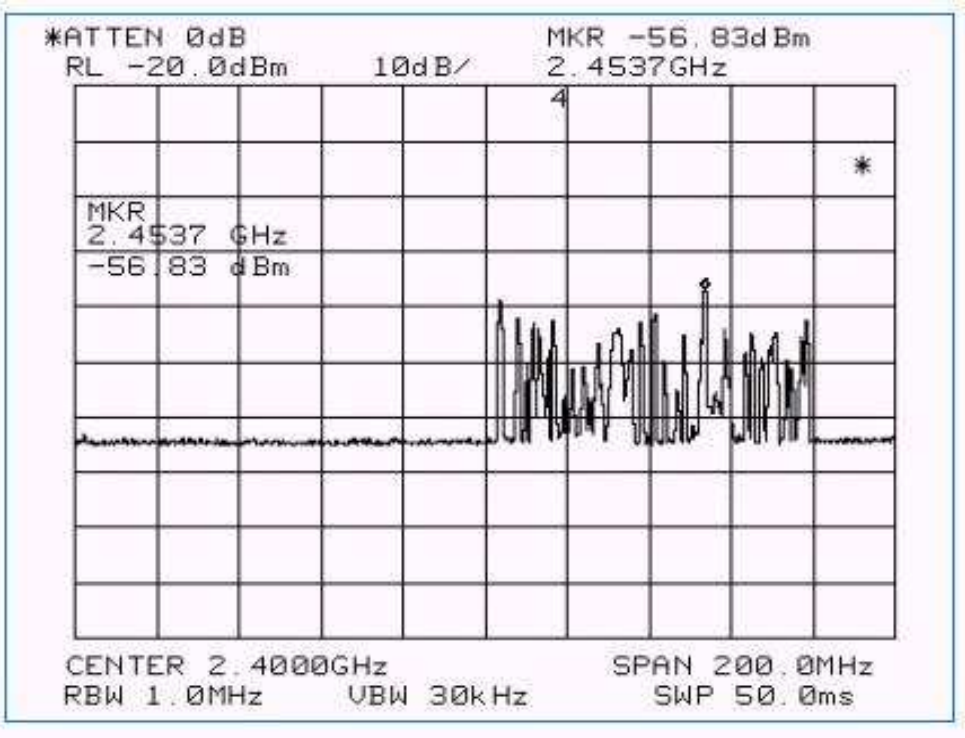


Figura 12 Nivel de potencia 4m

Respuesta entre las estaciones

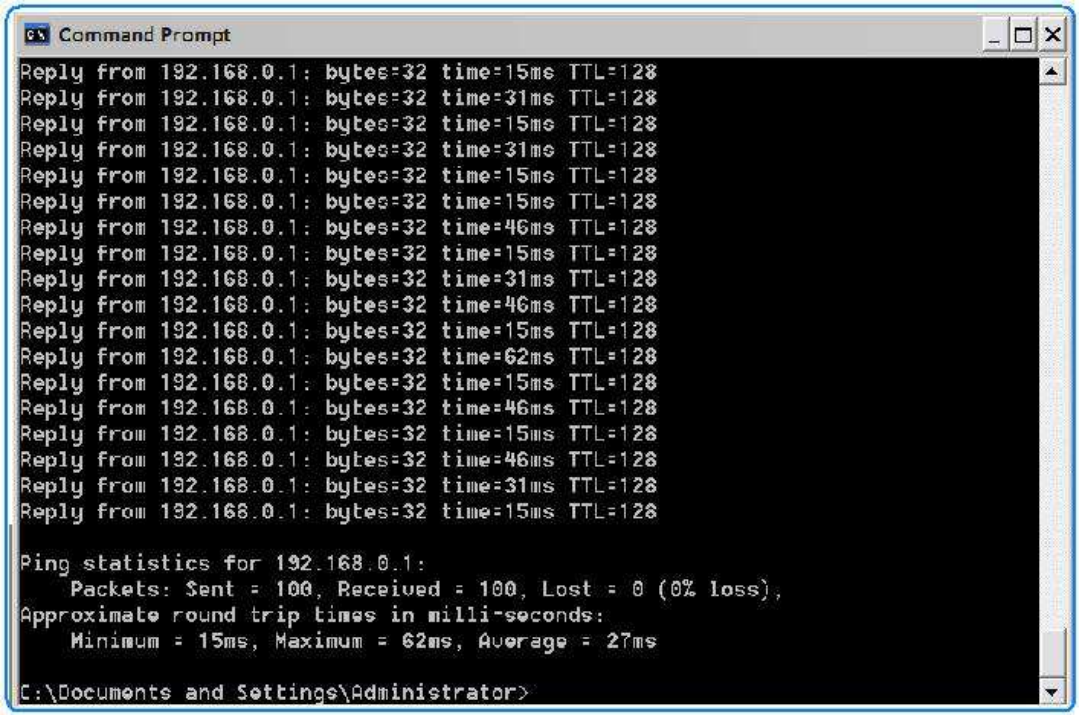


Figura 13 Ping entre las estaciones 4m

Velocidad

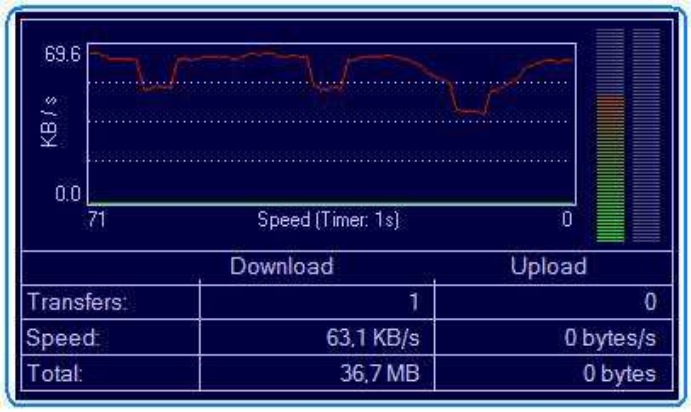


Figura 14 Velocidad 4m

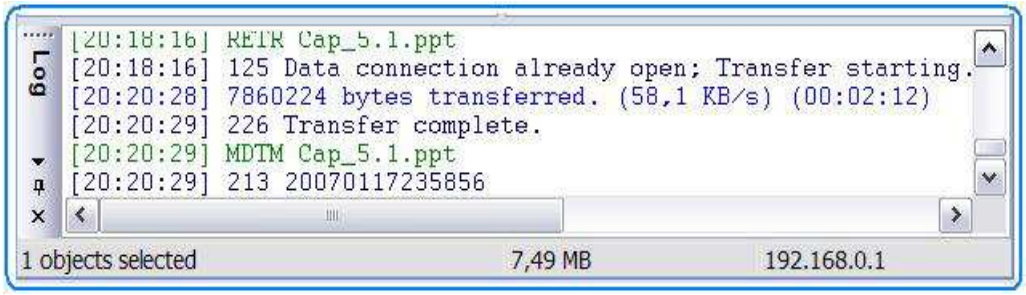


Figura 15 Velocidad 4m

Medidas a 5 metros de separación

Señal detectada

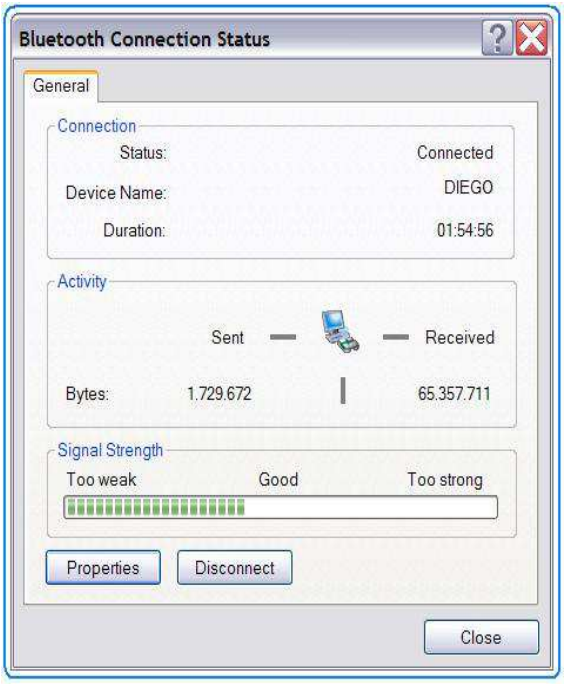


Figura 16 Estado de conexión 5m

Nivel de potencia (d = 5 m, p = - 57.17 dBm, f = 2.4557 GHz)

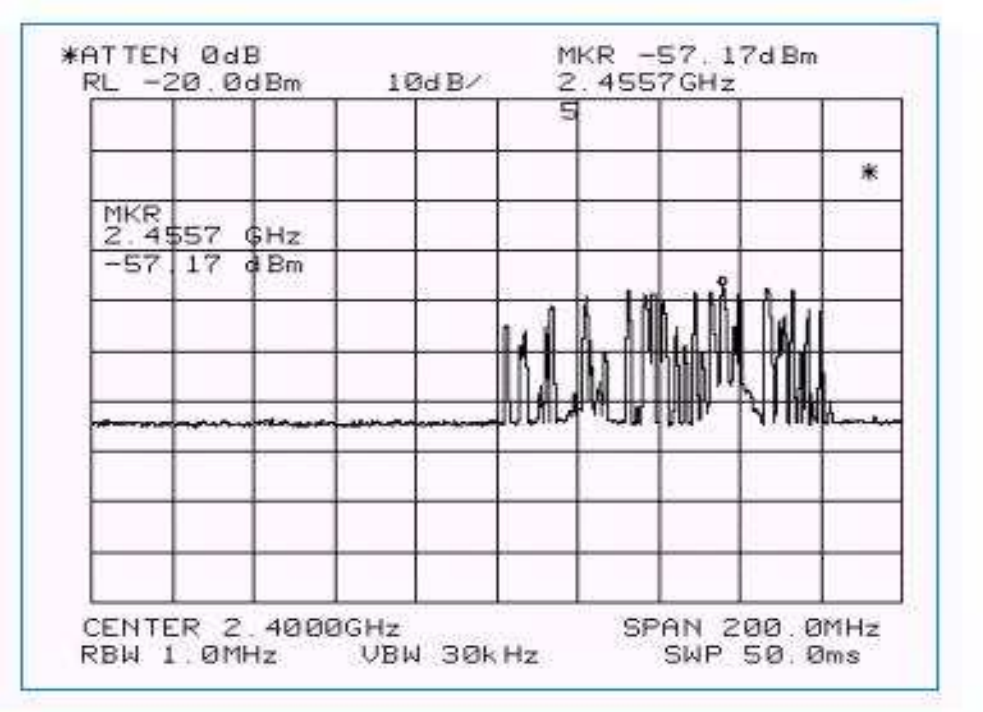


Figura 17 Nivel de potencia 5m

Respuesta entre las estaciones

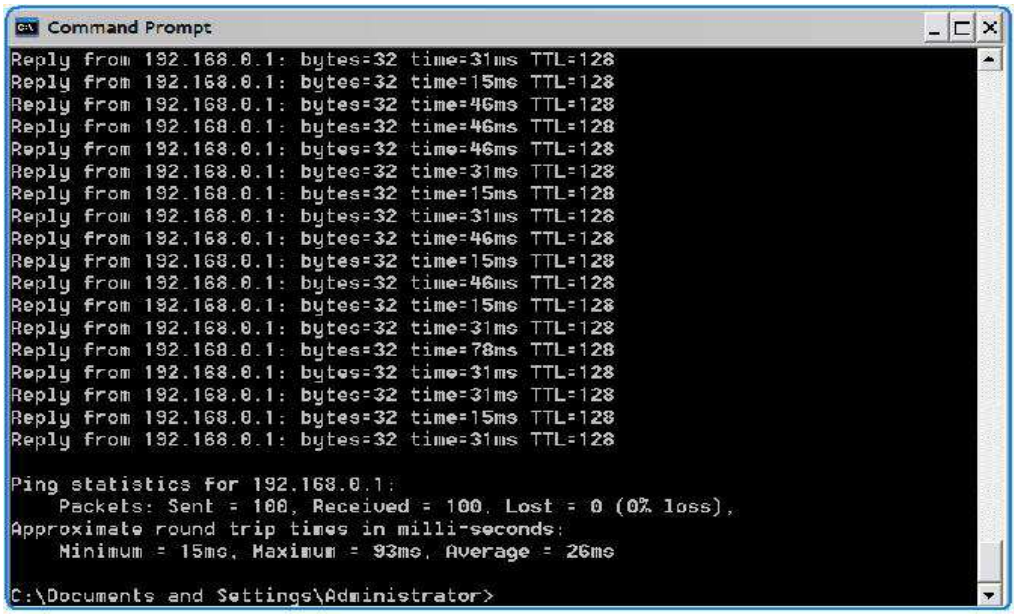


Figura 18 Ping entre las estaciones 5m

Velocidad

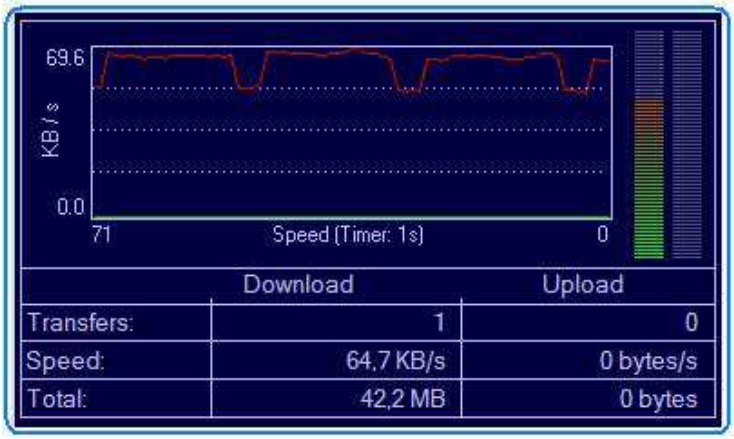


Figura 19 Velocidad 5m

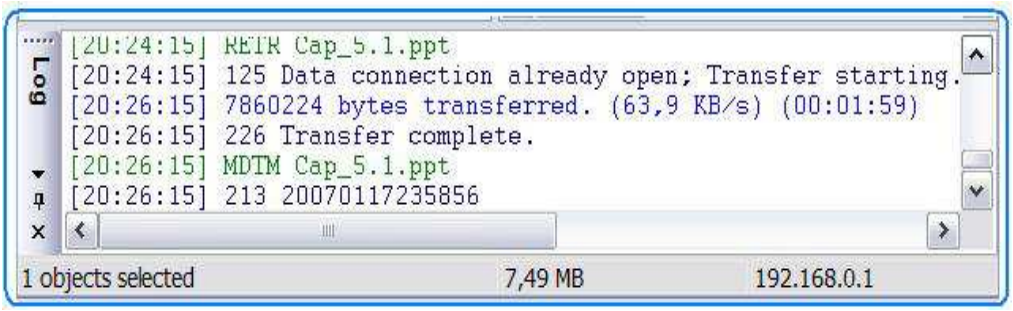


Figura 20 Velocidad Promedio 5m

Medidas a 6 metros de separación

Señal detectada

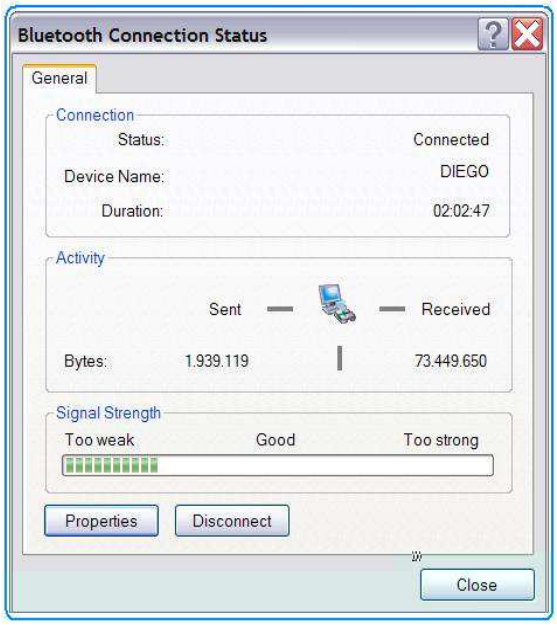


Figura 21 Estado de conexión 6m

Nivel de potencia (d = 6 m, p = - 56.83 dBm, f = 2.4420 GHz)

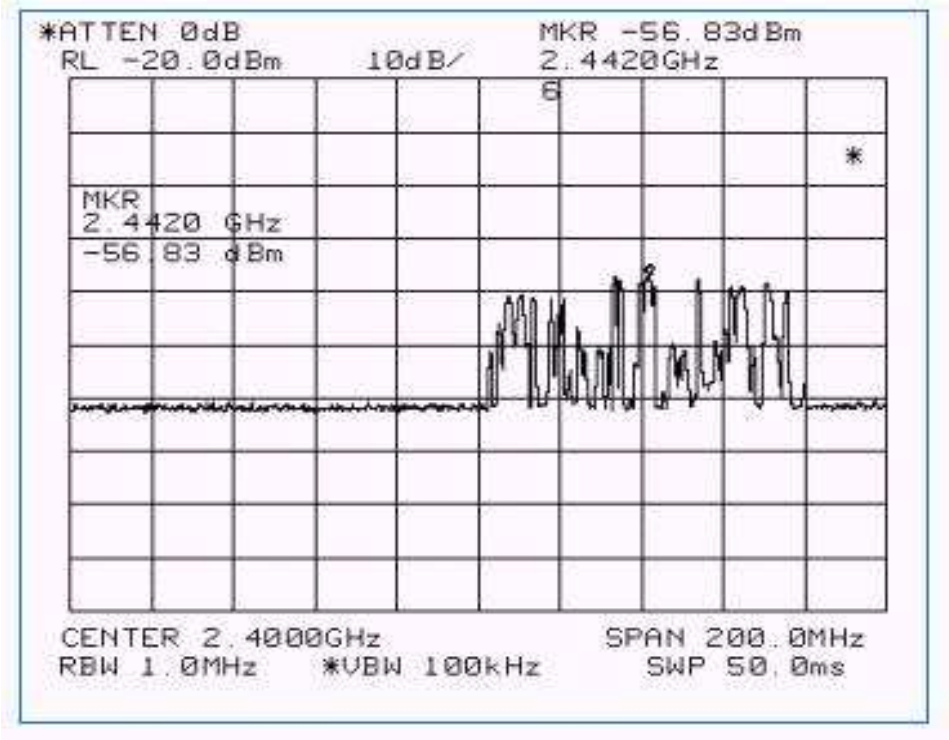


Figura 22 Nivel de potencia 6m

Respuesta entre las estaciones

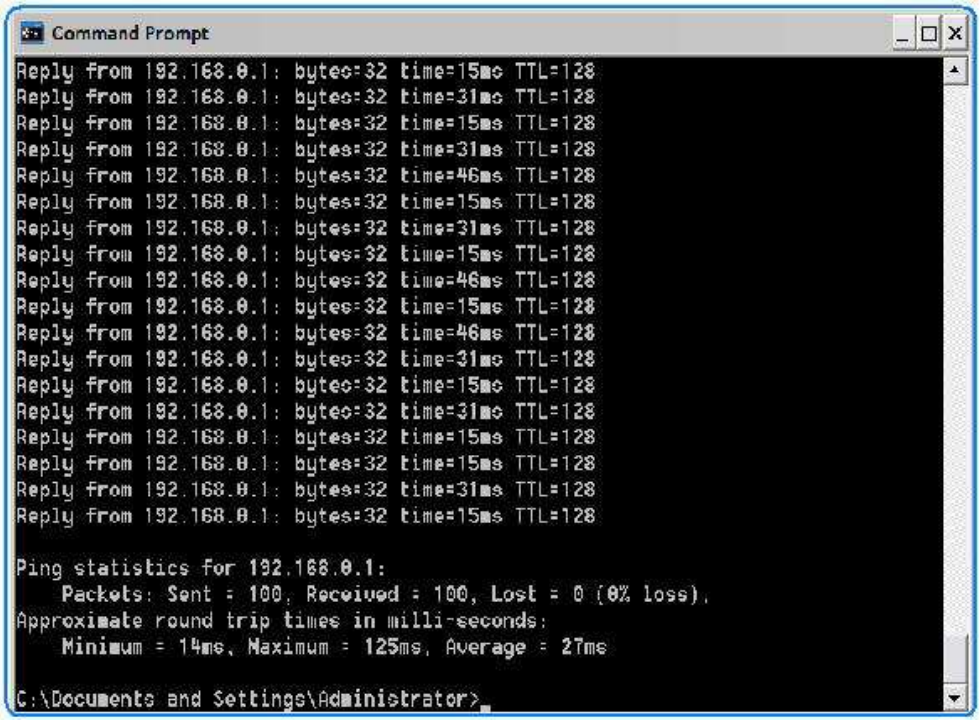


Figura 23 Ping entre las estaciones 6m

Velocidad

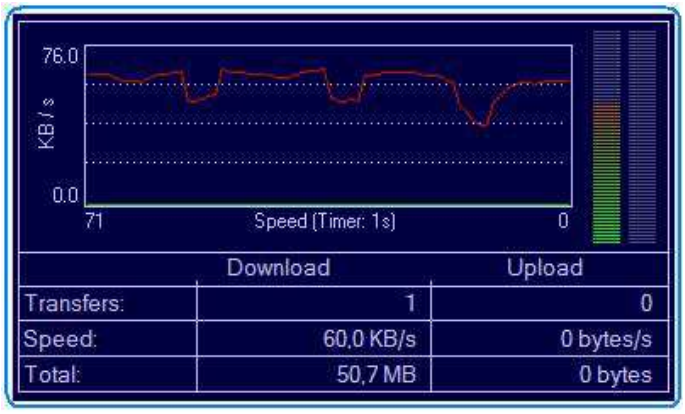


Figura 24 Velocidad 6m

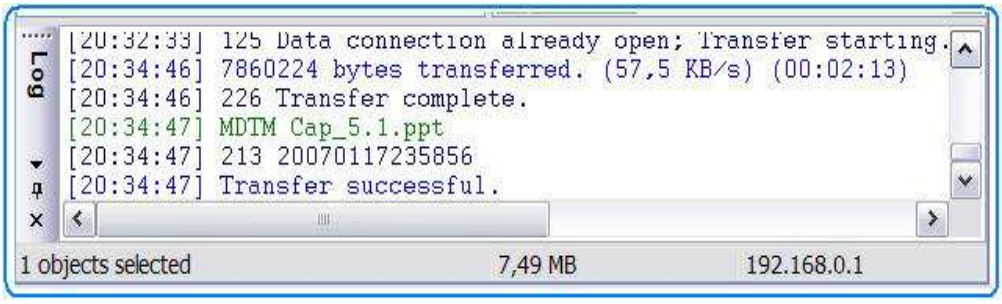


Figura 25 Velocidad Promedio 6m

Medidas a 7 metros de separación

Señal detectada

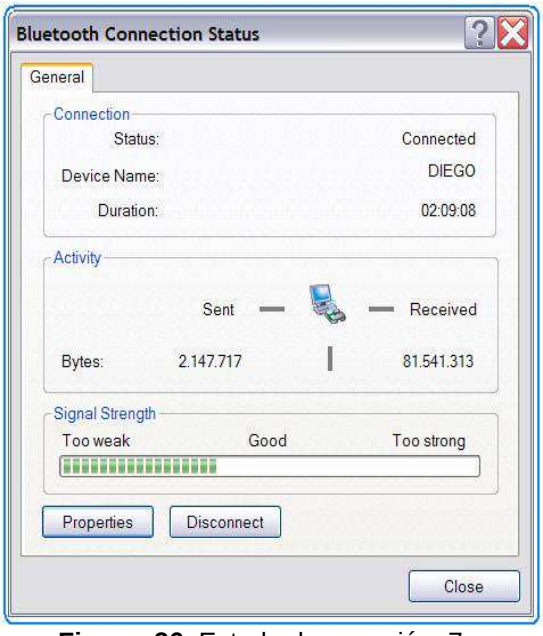


Figura 26 Estado de conexión 7m

Nivel de potencia (d = 7 m, p = - 58.17 dBm, f = 2.4287 GHz)

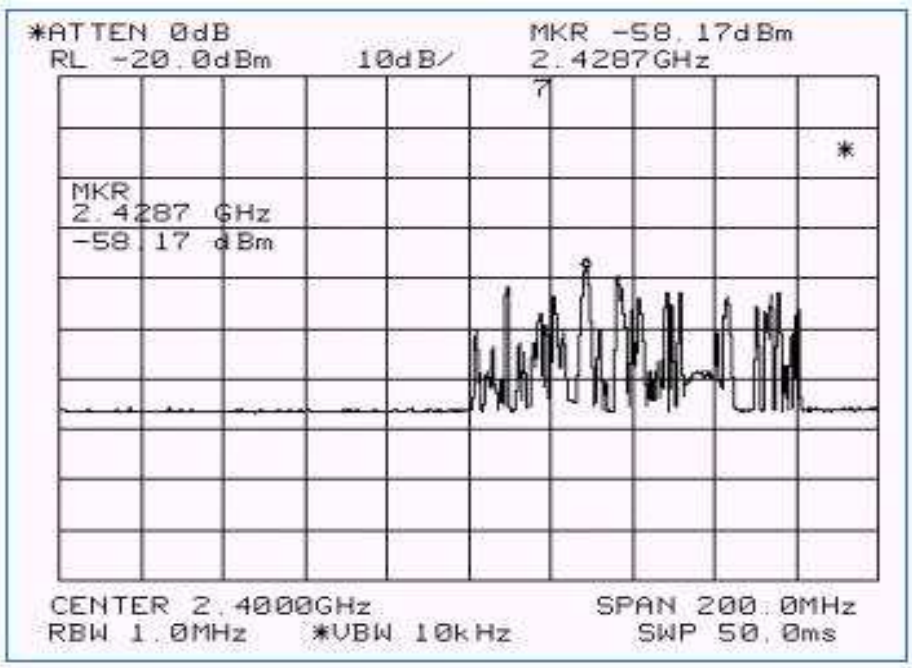


Figura 27 Nivel de potencia 7m

Respuesta entre las estaciones

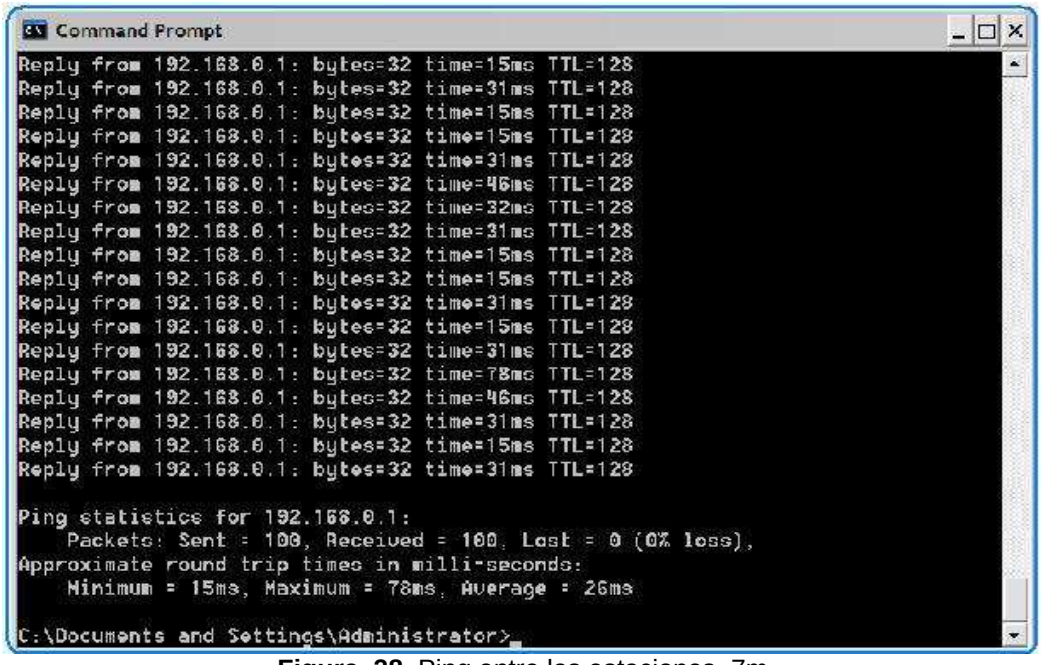


Figura 28 Ping entre las estaciones 7m

Velocidad

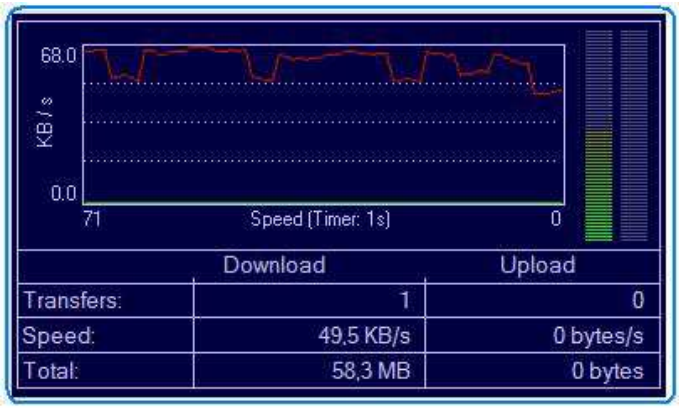


Figura 29 Velocidad 7m

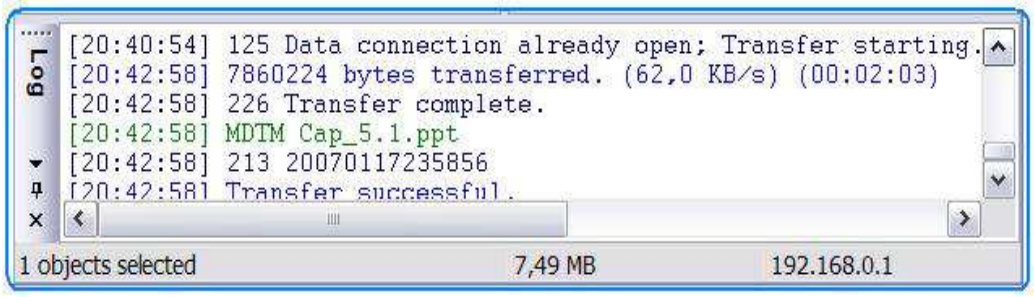


Figura 30 Velocidad Promedio 7m

Medidas a 8 metros de separación

Señal detectada

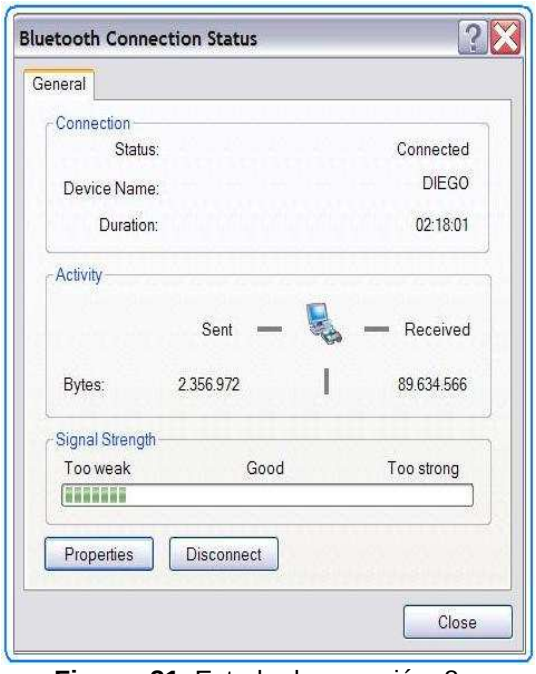


Figura 31 Estado de conexión 8m

Nivel de potencia (d = 8 m, p = - 58.00 dBm, f = 2.4277 GHz)

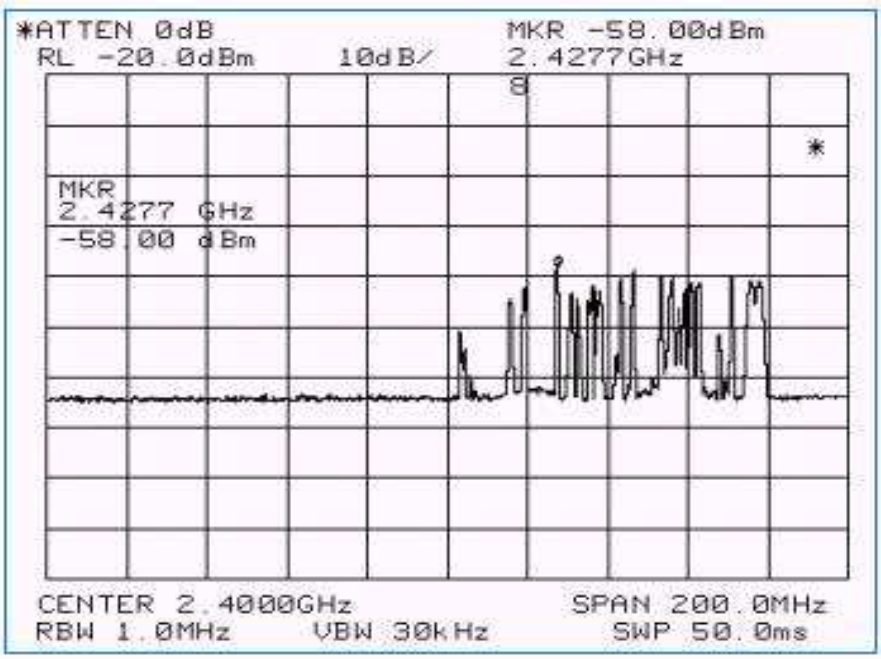


Figura 32 Nivel de potencia 8m

Respuesta entre las estaciones

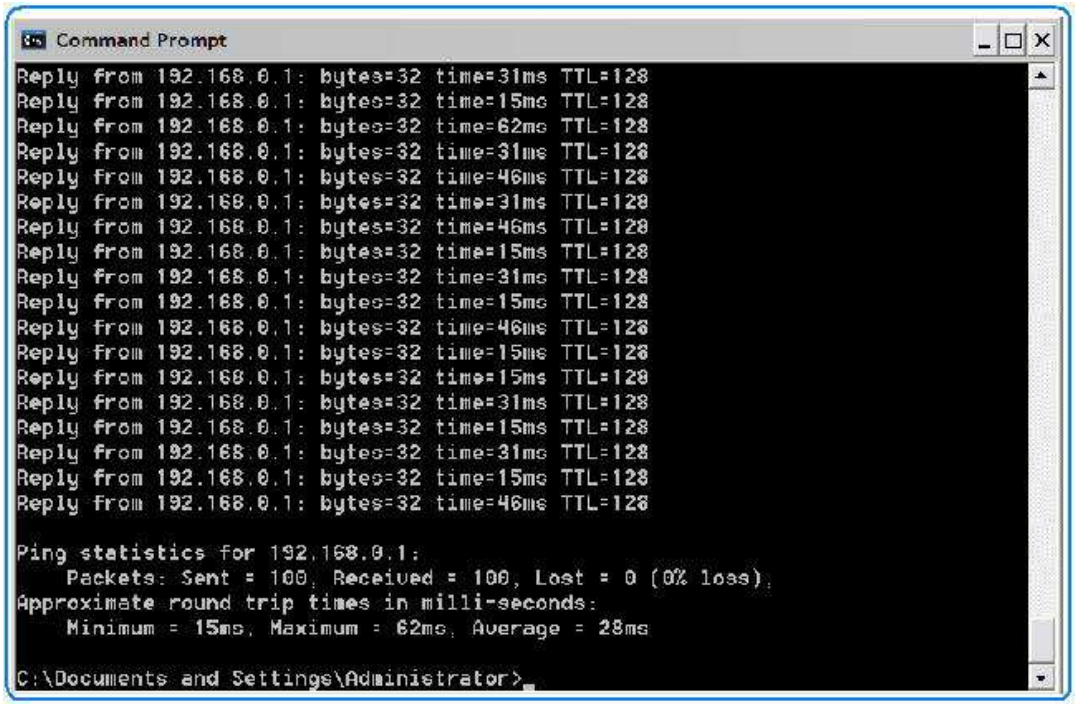


Figura 33 Ping entre las estaciones 8m

Velocidad

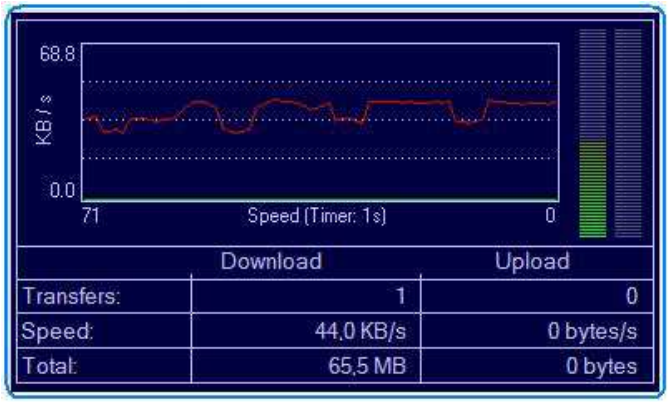


Figura 34 Velocidad 8m

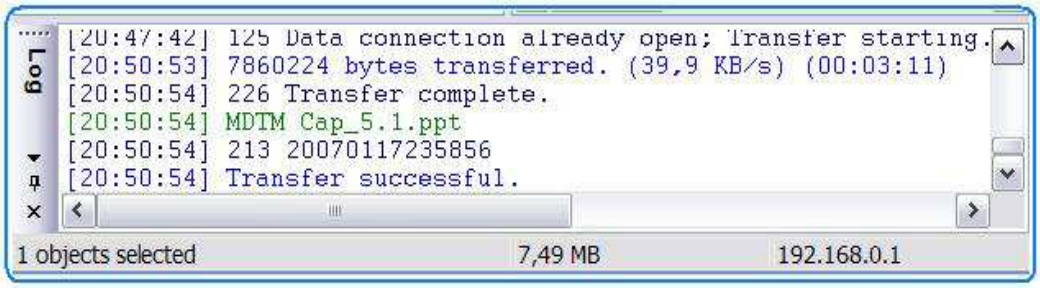


Figura 35 Velocidad Promedio 8m

Medidas a 9 metros de separación

Señal detectada

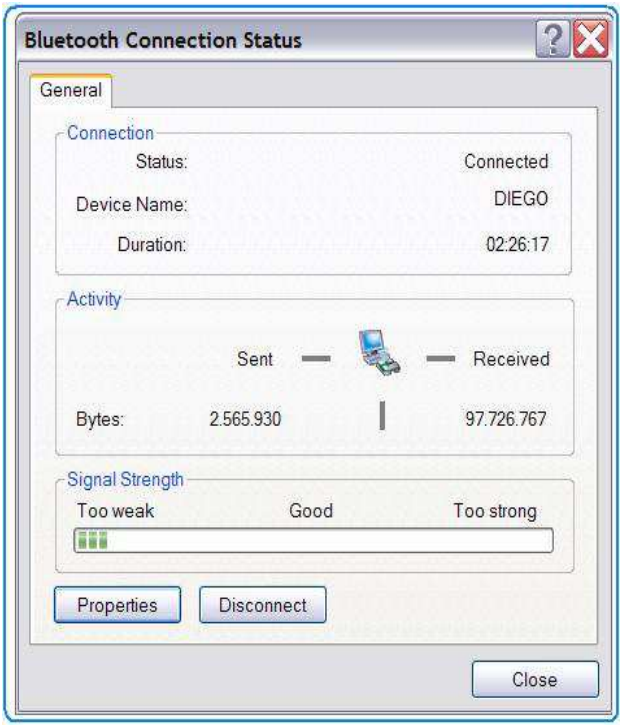


Figura 36 Estado de conexión 9m

Nivel de potencia (d = 9 m, p = - 57.33 dBm, f = 2.4177 GHz)

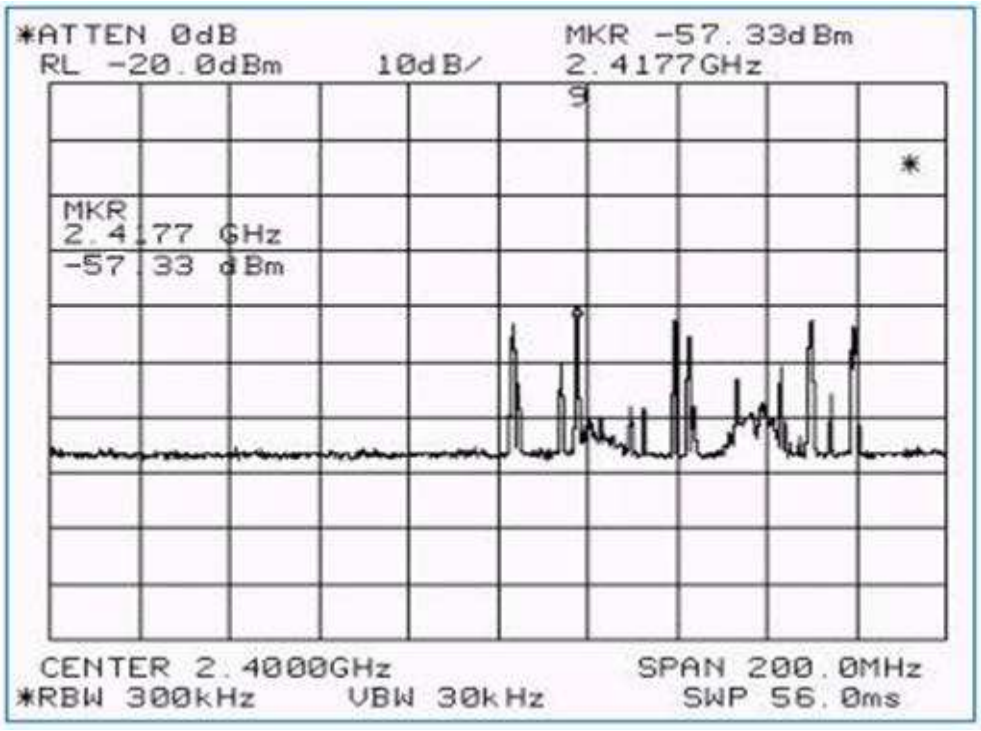


Figura 37 Nivel de potencia 9m

Respuesta entre las estaciones

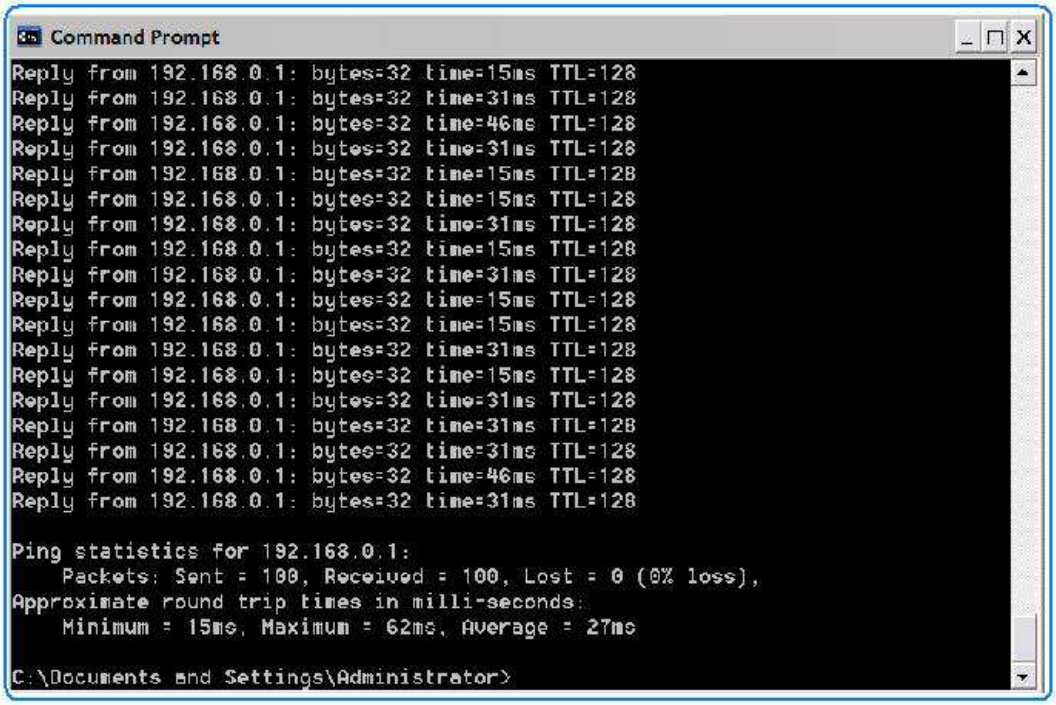


Figura 38 Ping entre las estaciones 9m

Velocidad

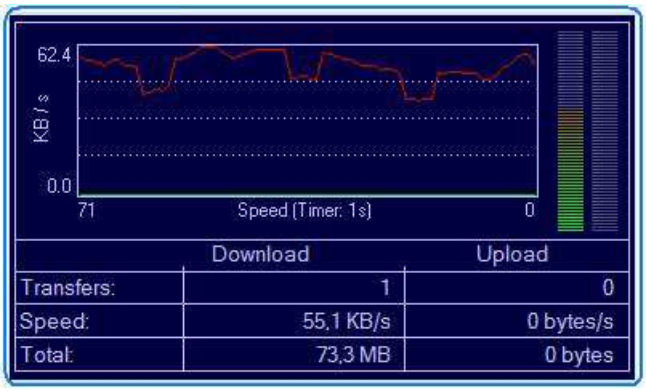


Figura 39 Velocidad 9m

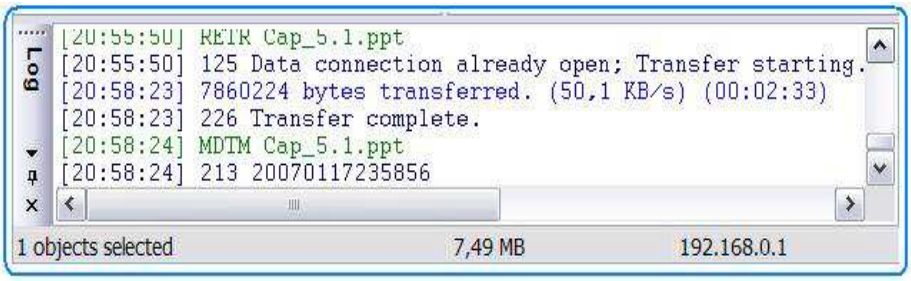


Figura 40 Velocidad Promedio 9m

Medidas a 10 metros de separación

Señal detectada

Figura 41 Estado de conexión 10m

Nivel de potencia (d = 10 m, p = - 58.50 dBm, f = 2.4307 GHz)

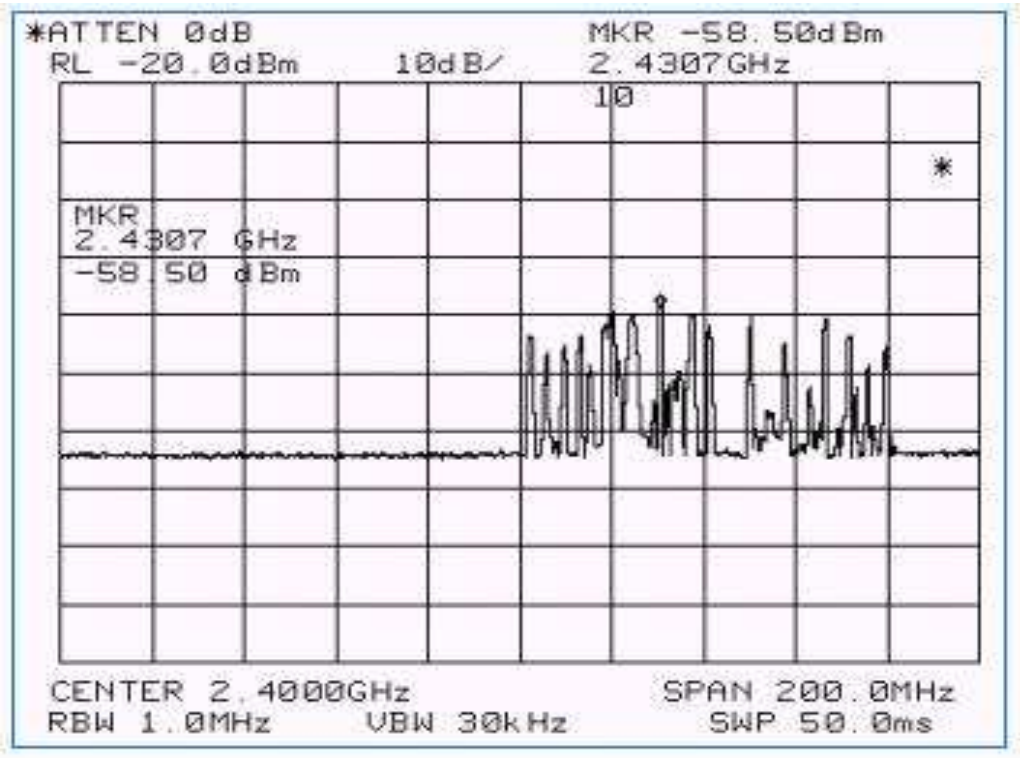


Figura 42 Nivel de potencia 10m

Respuesta entre las estaciones

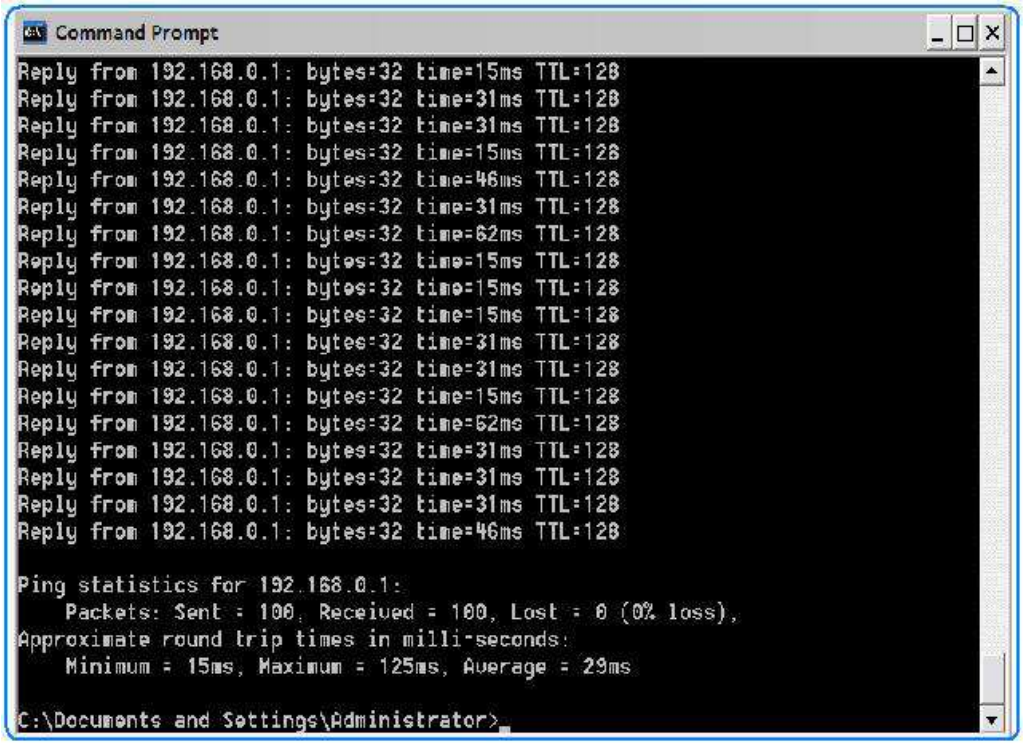


Figura 43 Ping entre las estaciones 10m

Velocidad

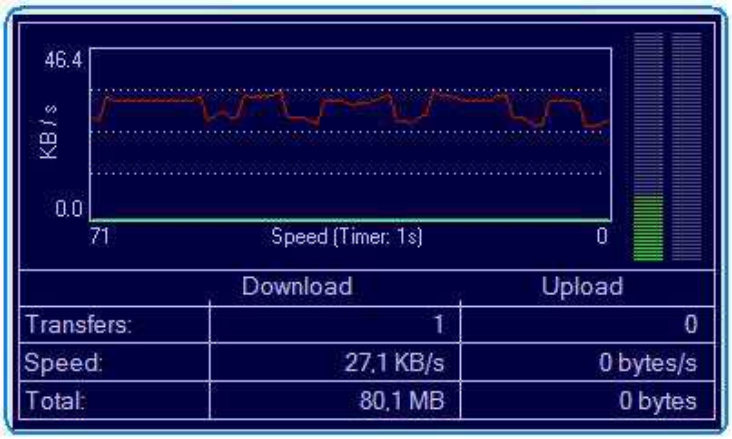


Figura 44 Velocidad 10m

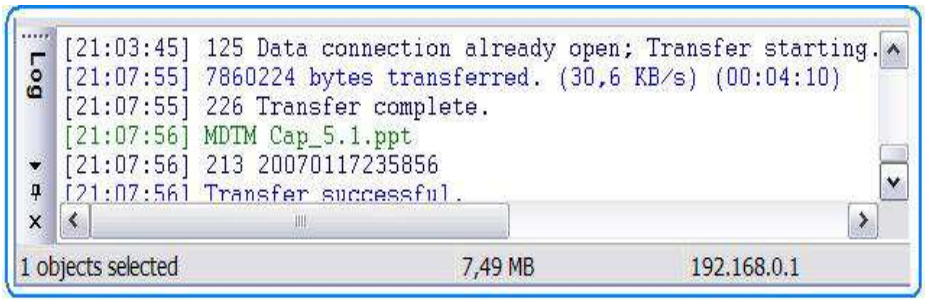


Figura 45 Velocidad Promedio 10m

Medidas a 12 metros de separación

Señal detectada

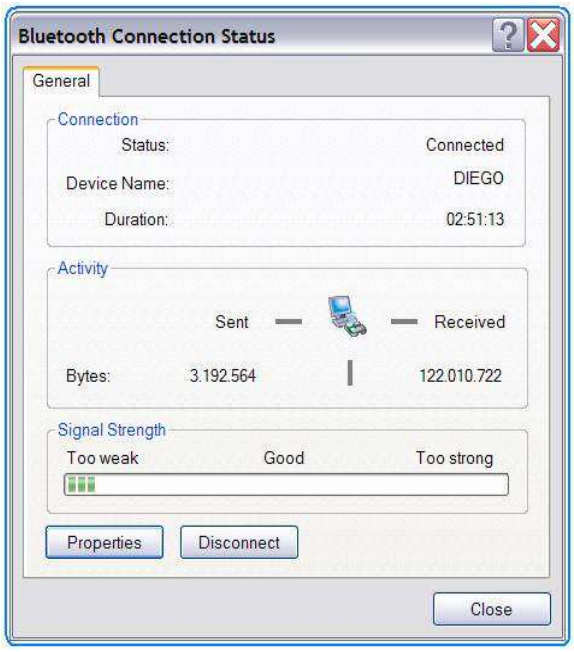


Figura 46 Estado de conexión 12m

Nivel de potencia (d = 12 m, p = - 60.00 dBm, f = 2.4397 GHz)

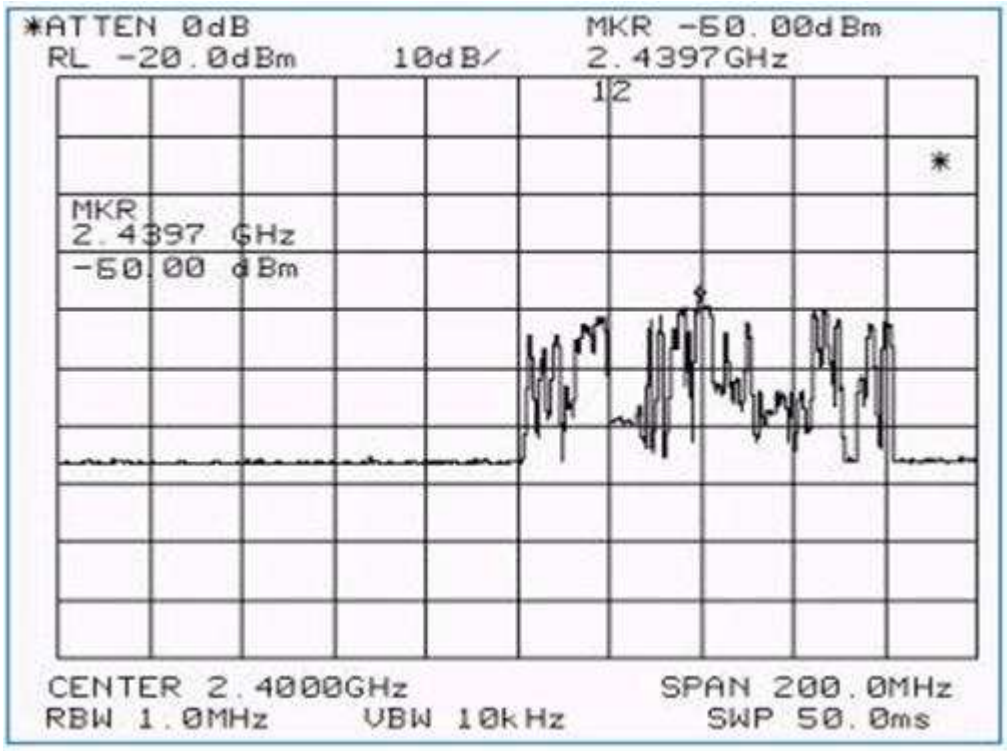


Figura 47 Nivel de potencia 12m

Respuesta entre las estaciones

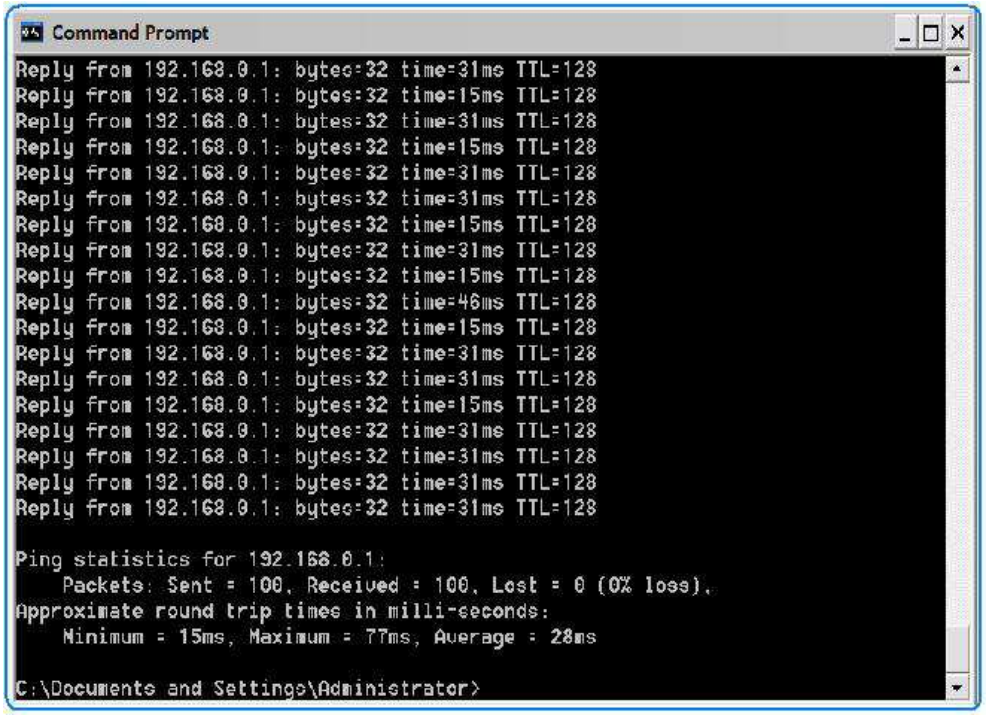


Figura 48 Ping entre las estaciones 12m

Velocidad

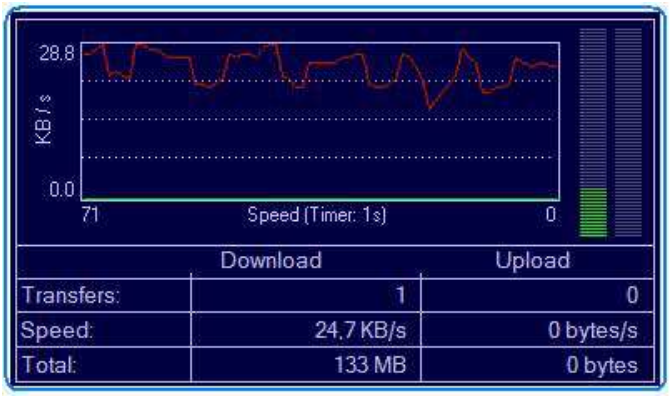


Figura 49 Velocidad 12m

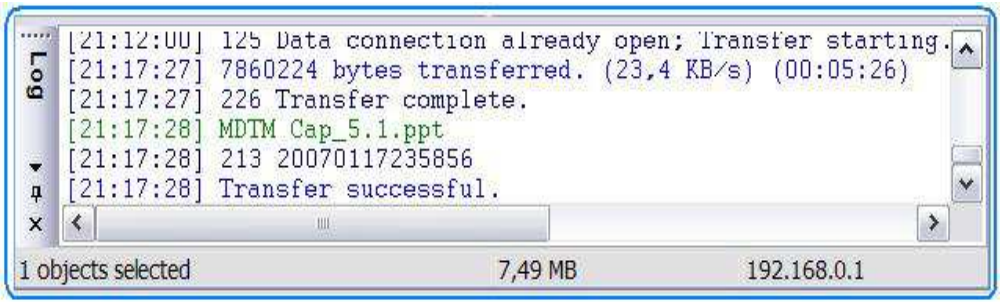


Figura 50 Velocidad Promedio 12m

Medidas a 15 metros de separación

Señal detectada

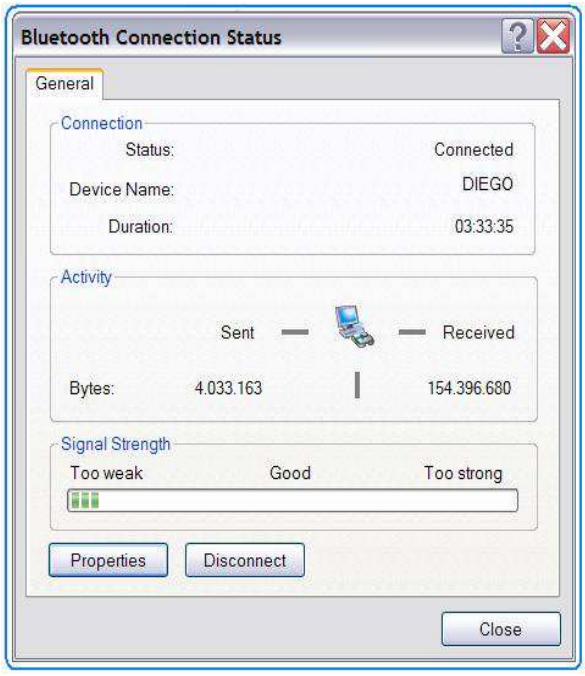


Figura 51 Estado de conexión 15m

Nivel de potencia (d = 15 m, p = - 61.83 dBm, f = 2.4207 GHz)

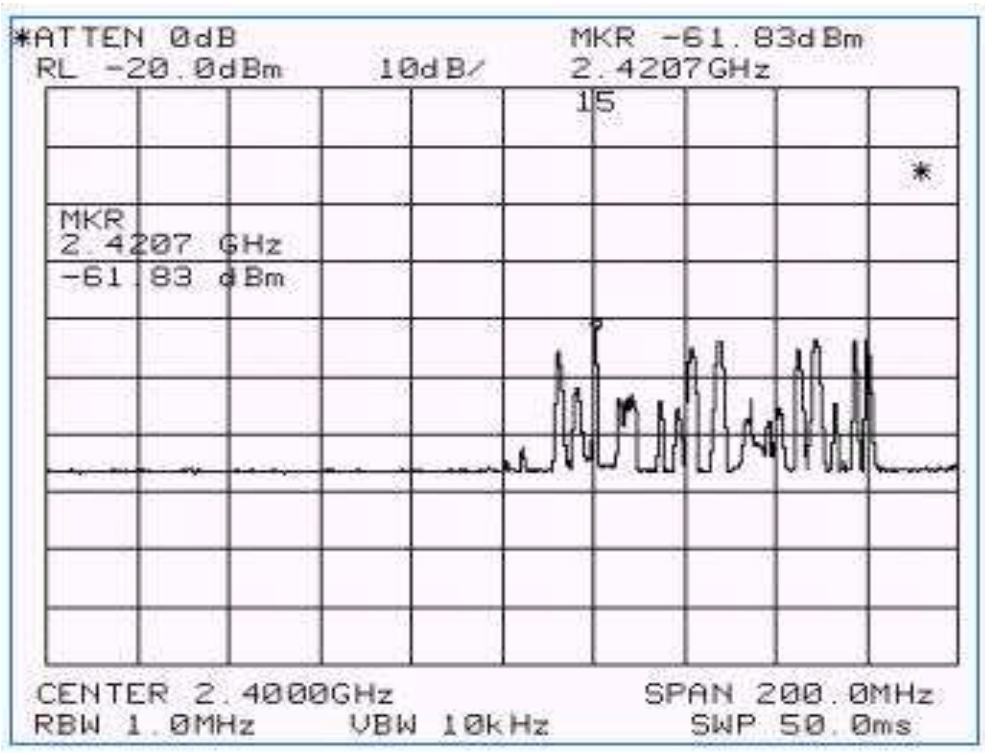


Figura 52 Nivel de potencia 15m

Respuesta entre las estaciones

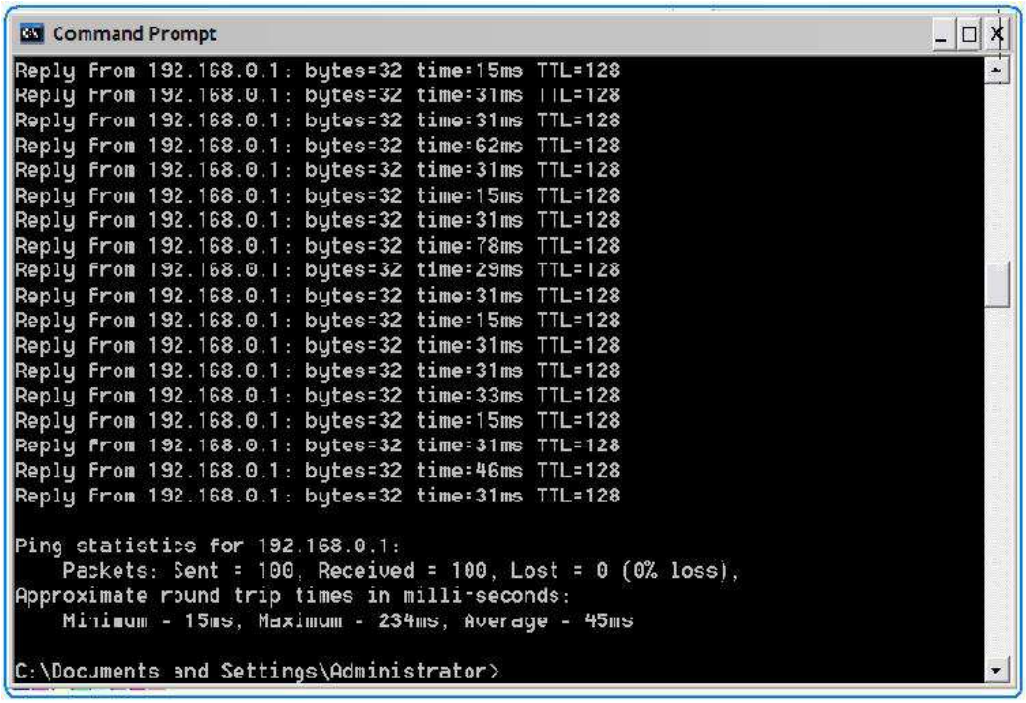


Figura 53 Ping entre las estaciones 15m

Velocidad

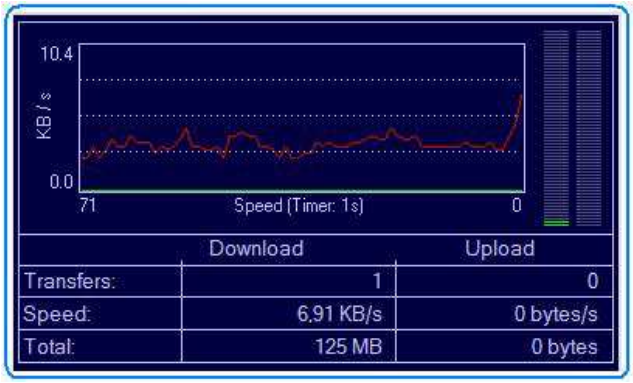


Figura 54 Velocidad 15m

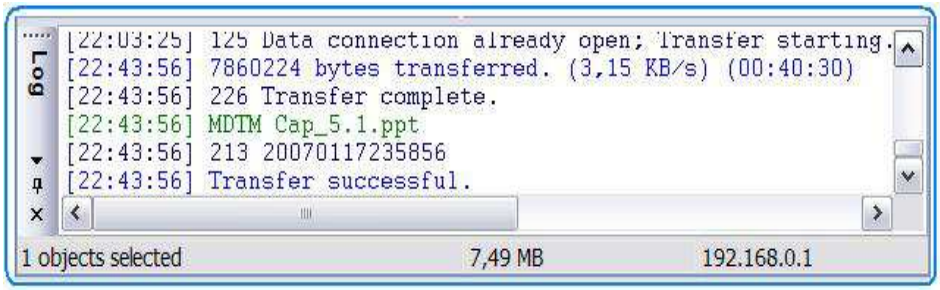


Figura 55 Velocidad 15m

ANEXO E

Pruebas Prácticas Wi-Fi

Pruebas Prácticas de Wi-Fi

Medidas a 2 metros de separación

Señal detectada

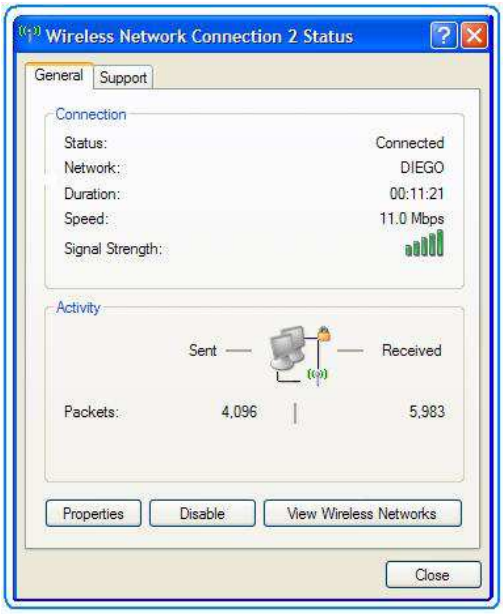


Figura 1 Estado de conexión 2m

Nivel de potencia (d = 2 m, p = - 59.00 dBm, f = 2.4340 GHz)

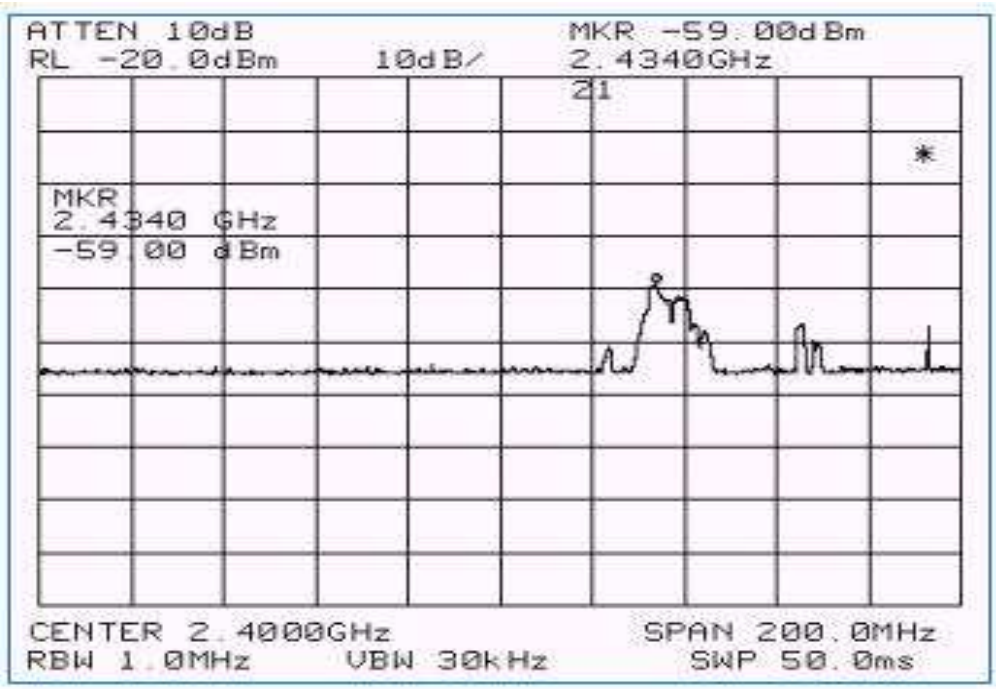


Figura 2 Nivel de potencia 2m

Respuesta entre las estaciones

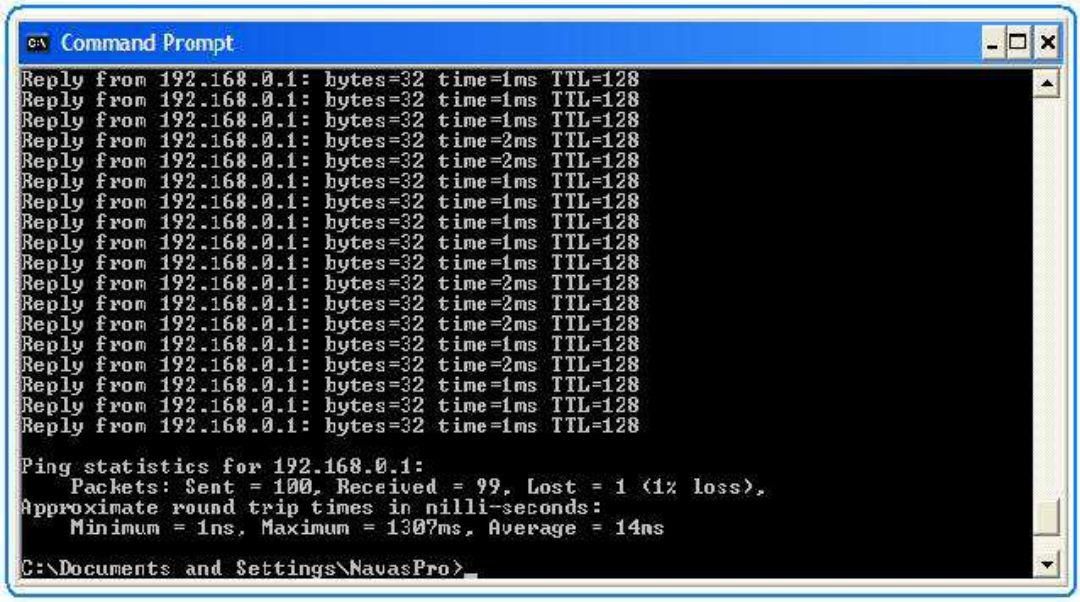


Figura 3 Ping entre las estaciones 2m

Velocidad

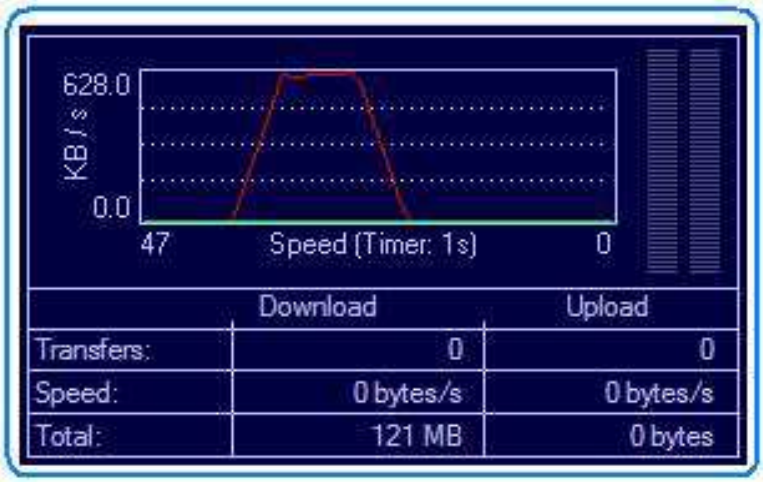


Figura 4 Velocidad 2m

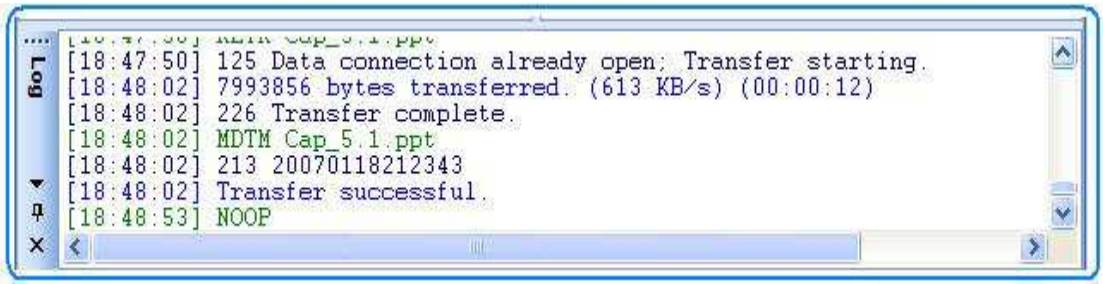


Figura 5 Velocidad Promedio 2m

Medidas a 3 metros de separación

Señal detectada

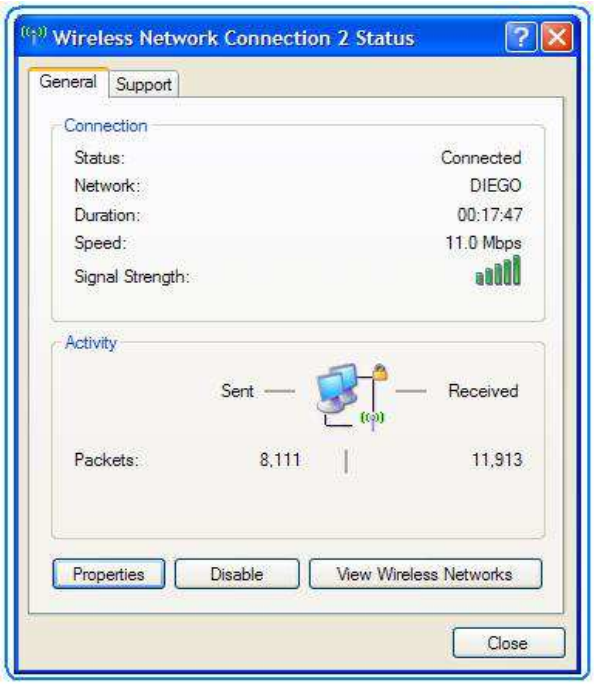


Figura 6 Estado de conexión 3m

Nivel de potencia (d = 3 m, p = - 60.83 dBm, f = 2.4410 GHz)

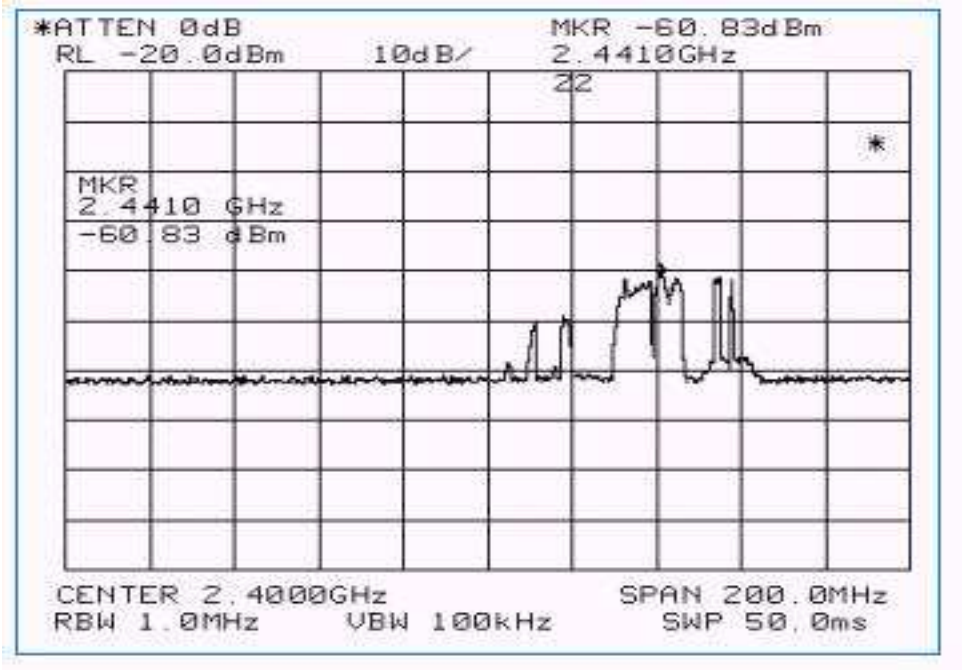


Figura 7 Nivel de potencia 3m

Respuesta entre las estaciones

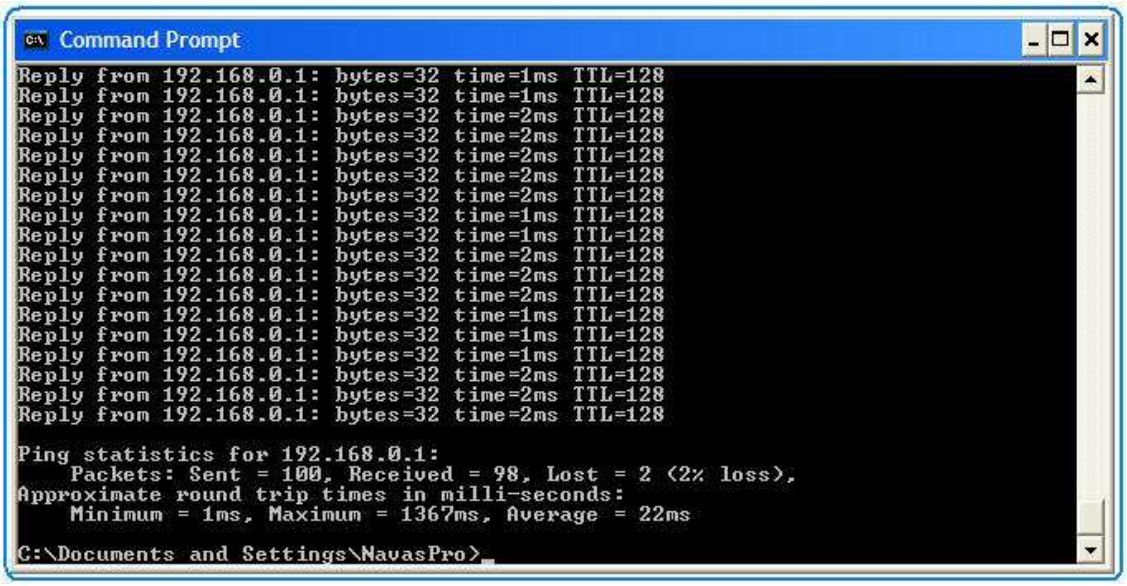


Figura 8 Ping entre las estaciones 3m

Velocidad

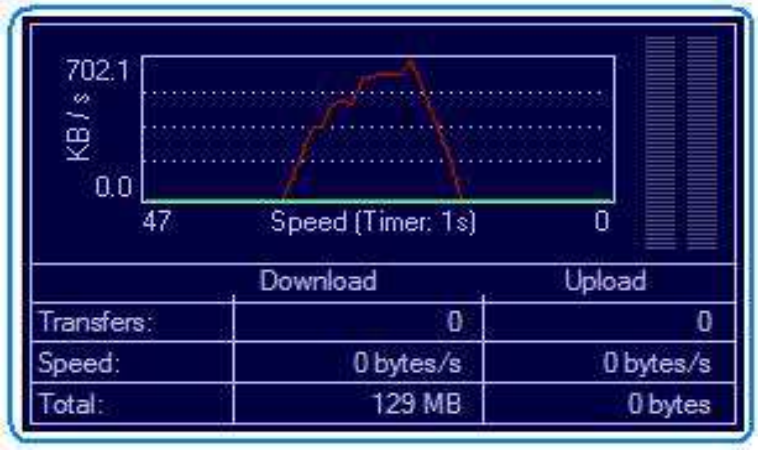


Figura 9 Velocidad 3m

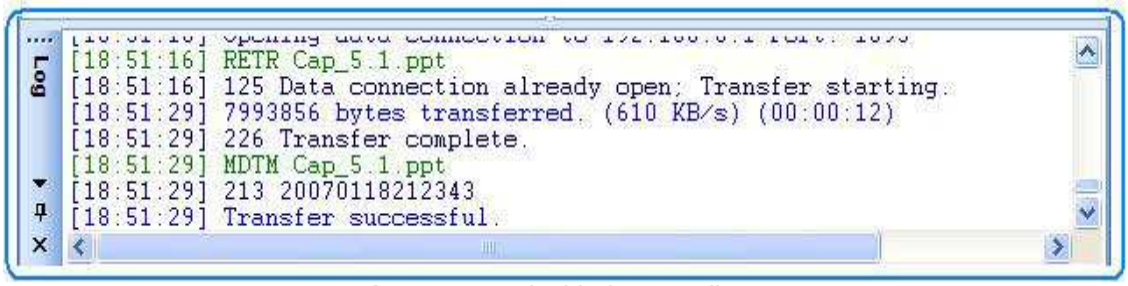


Figura 10 Velocidad Promedio 3m

Medidas a 4 metros de separación

Señal detectada

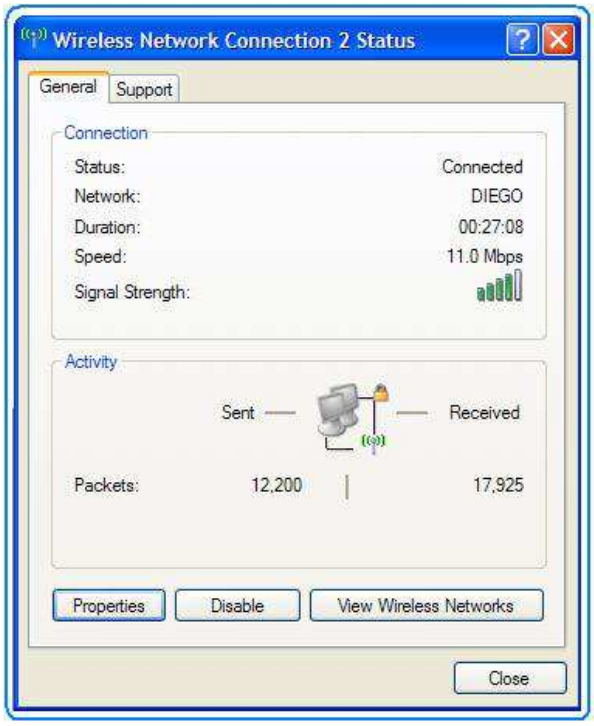


Figura 11 Estado de conexión 4m

Nivel de potencia (d = 4 m, p = - 56.83 dBm, f = 2.4383 GHz)

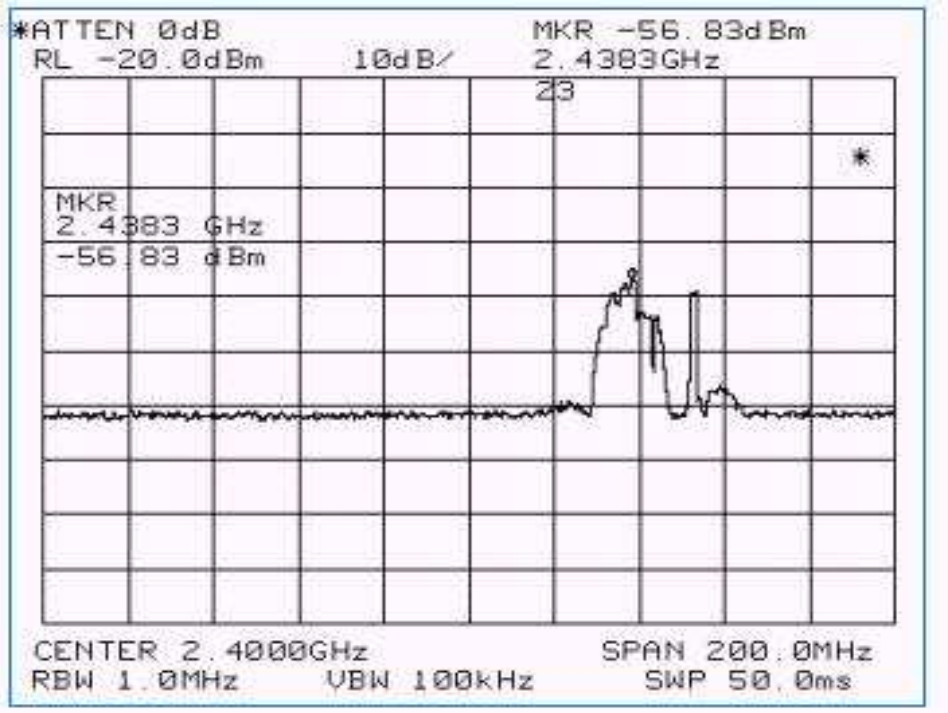


Figura 12 Nivel de potencia 4m

Respuesta entre las estaciones

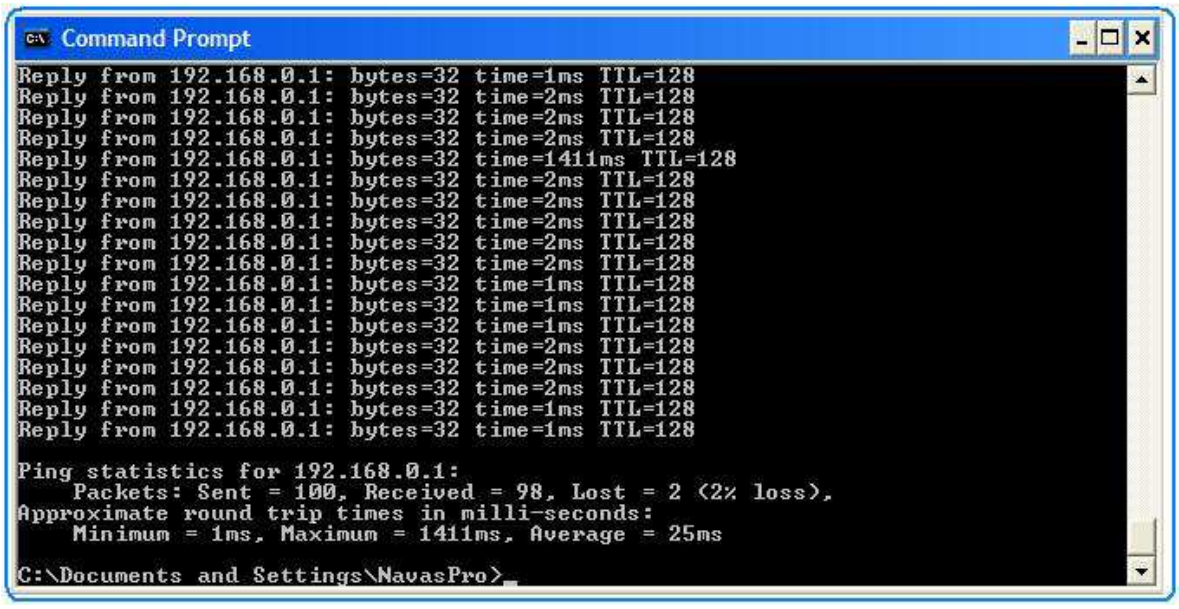


Figura 13 Ping entre las estaciones 4m

Velocidad

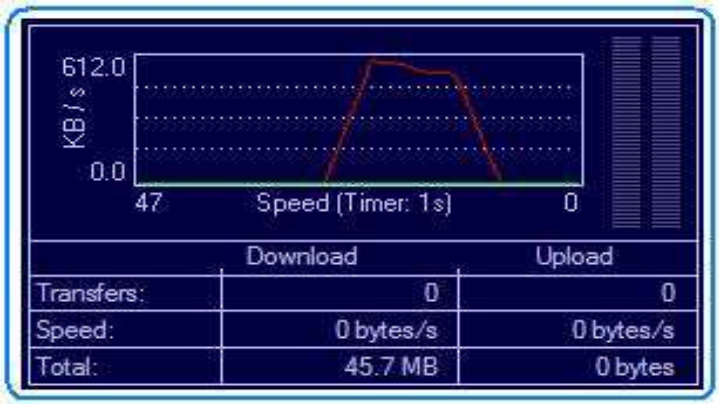


Figura 14 Velocidad 4m

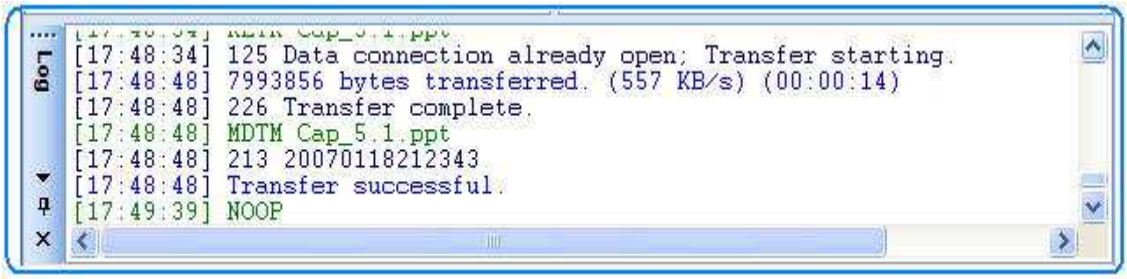


Figura 15 Velocidad Promedio 4m

Medidas a 5 metros de separación

Señal detectada

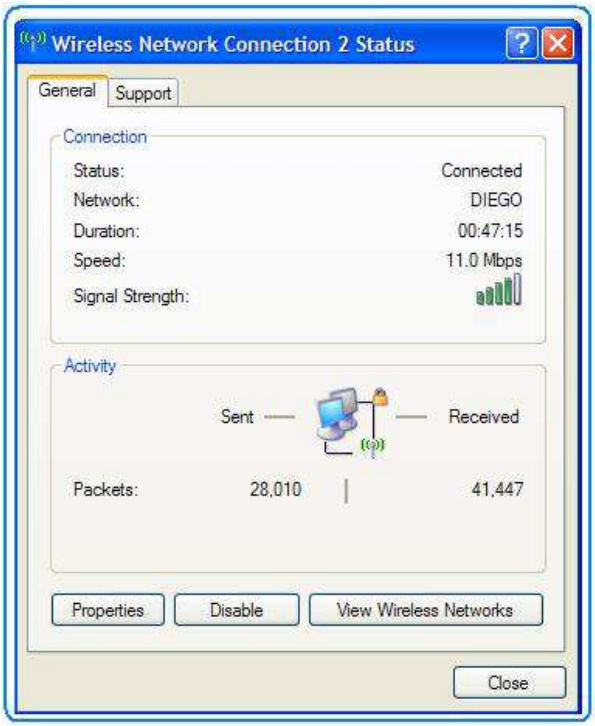


Figura 16 Estado de conexión 5m

Nivel de potencia (d = 5 m, p = - 58.33 dBm, f = 2.4383 GHz)

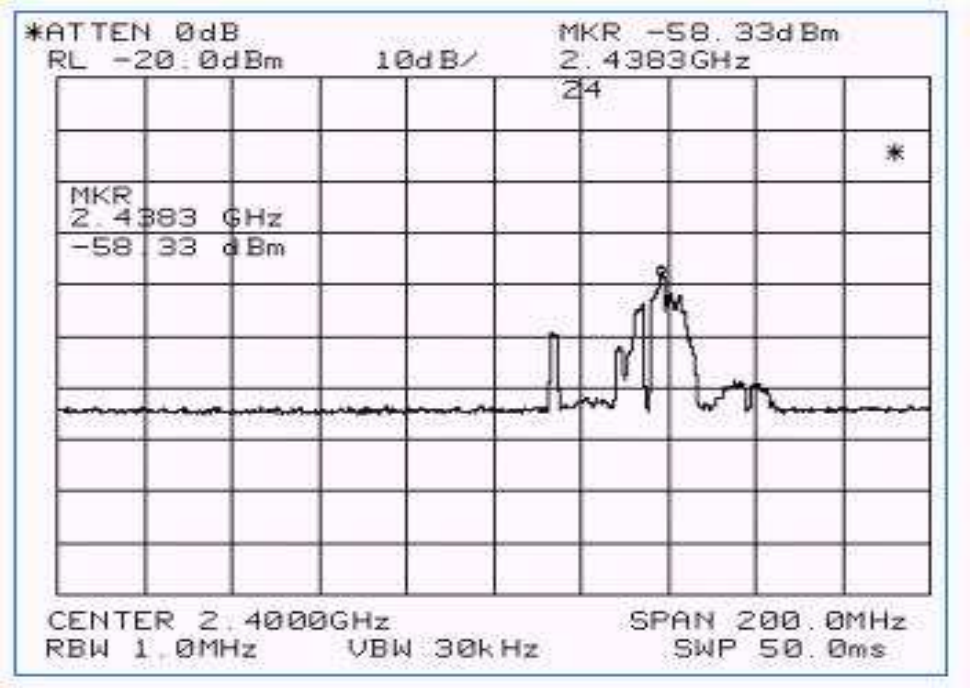


Figura 17 Nivel de potencia 5m

Respuesta entre las estaciones

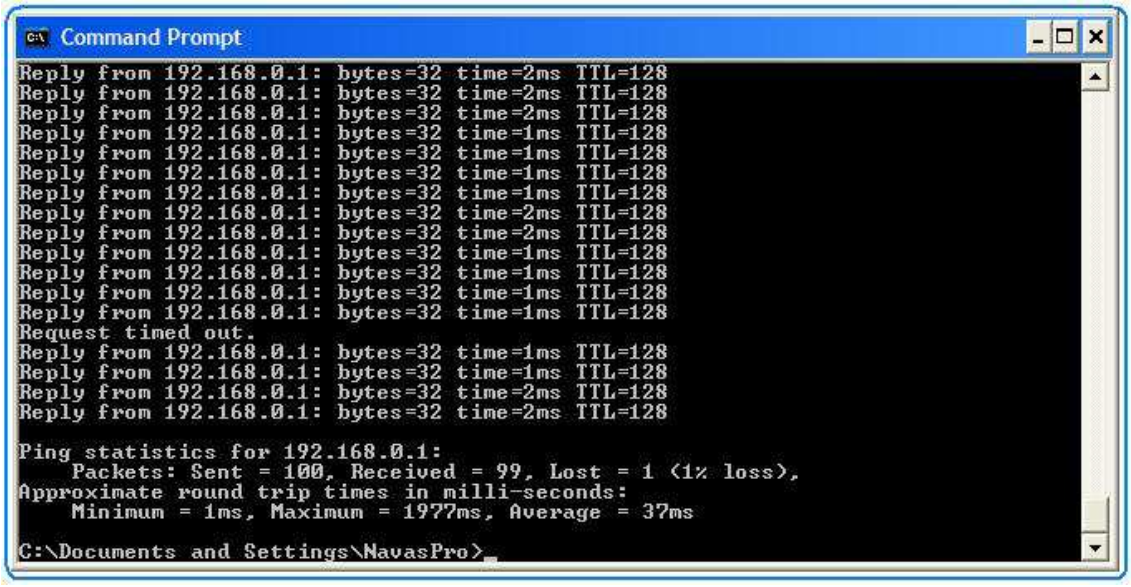


Figura 18 Ping entre las estaciones 5m

Velocidad

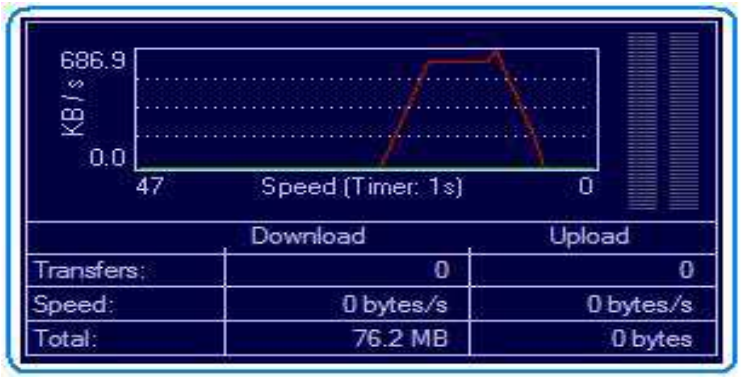


Figura 19 Velocidad 5m

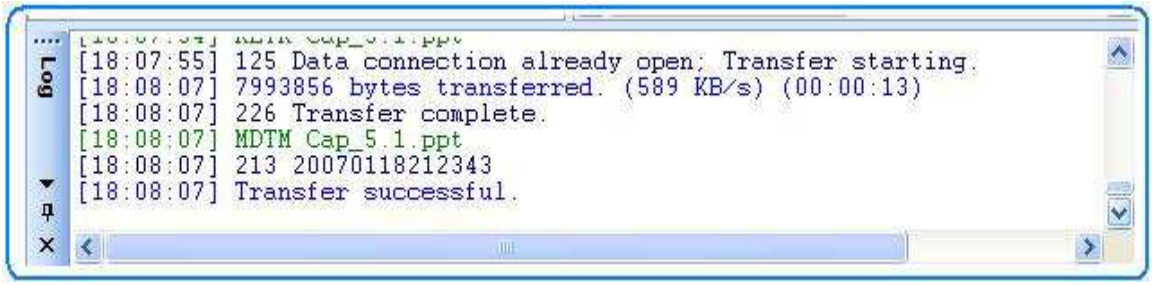


Figura 20 Velocidad Promedio 5m

Medidas a 6 metros de separación

Señal detectada

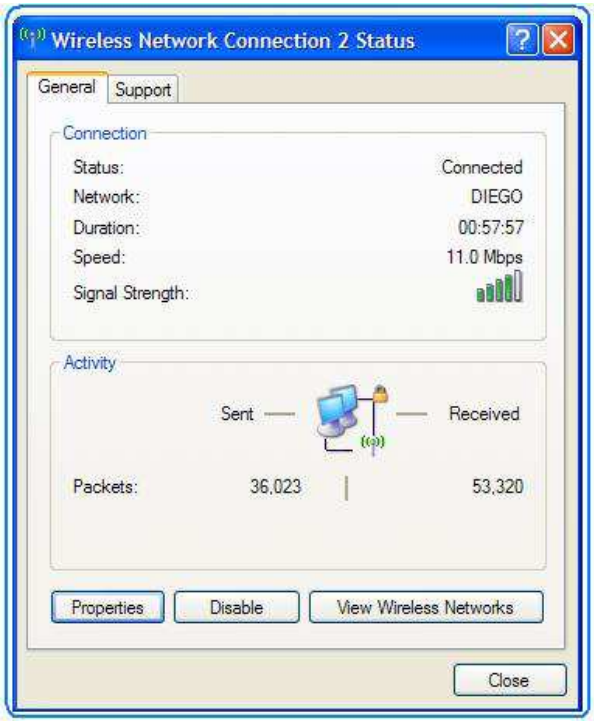


Figura 21 Estado de conexión 6m

Nivel de potencia (d = 6 m, p = - 62.50 dBm, f = 2.4337 GHz)

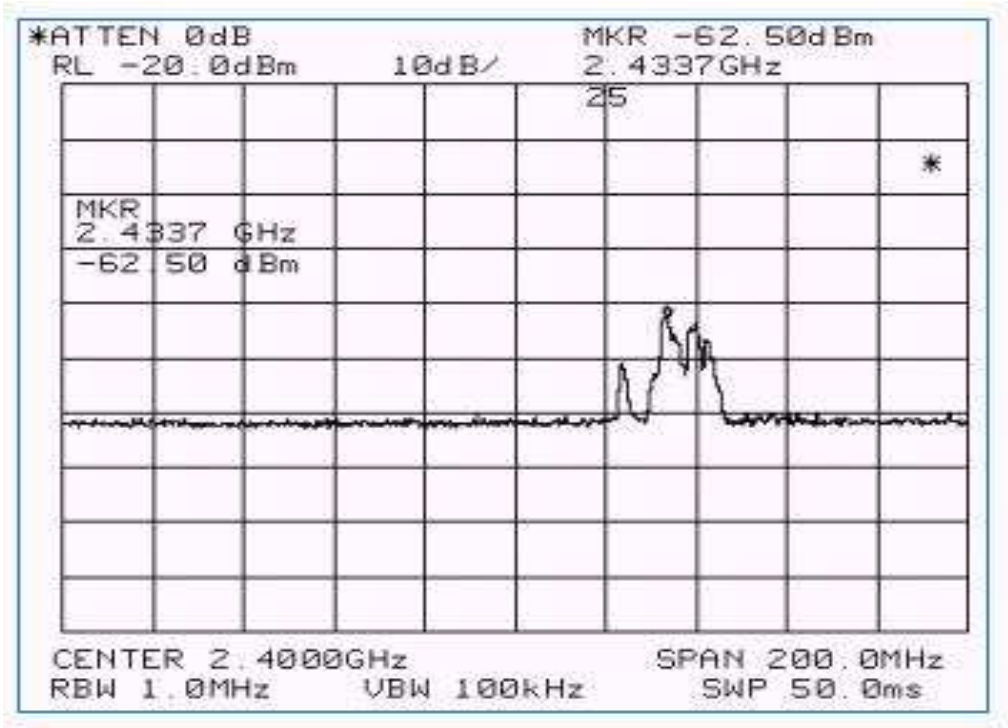


Figura 22 Nivel de potencia 6m

Respuesta entre las estaciones

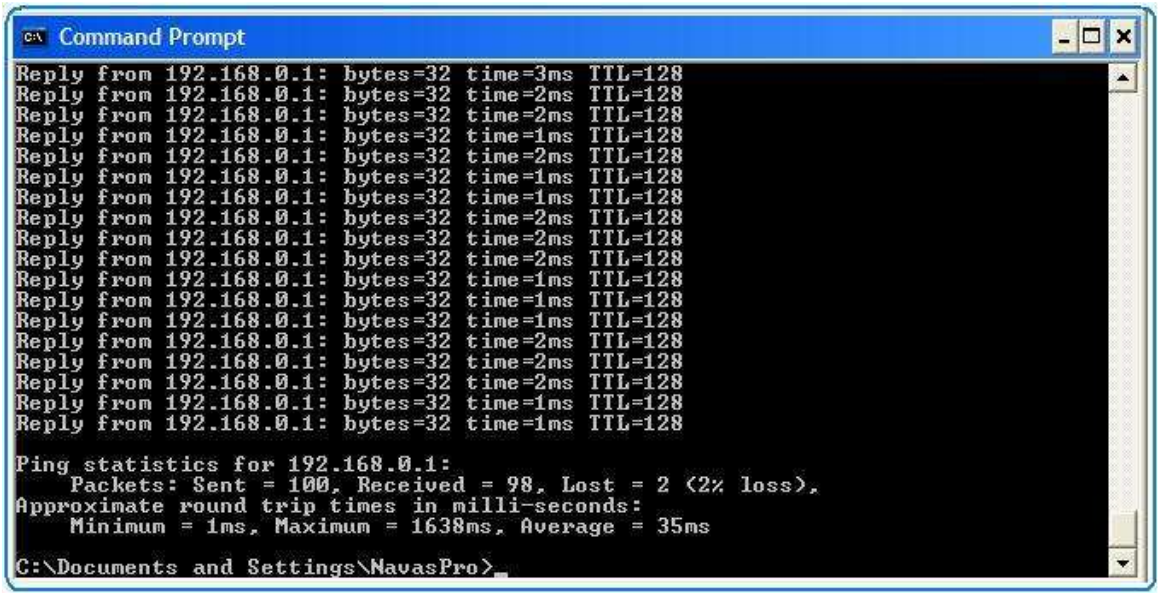


Figura 23 Ping entre las estaciones 6m

Velocidad

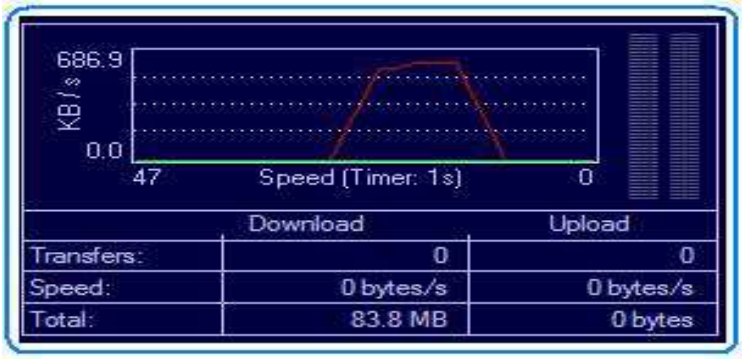


Figura 24 Velocidad 6m

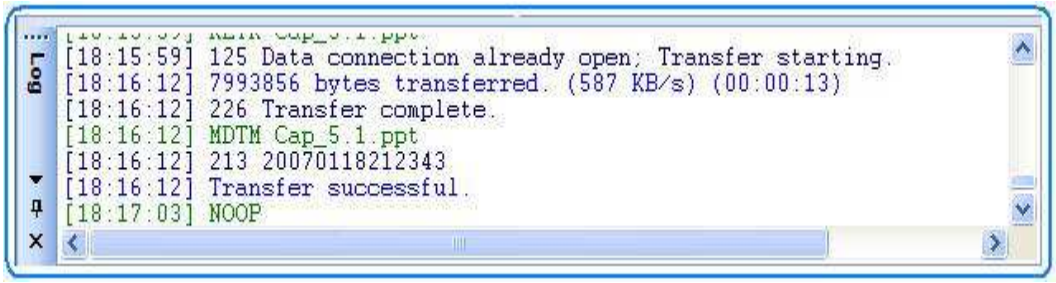


Figura 25 Velocidad Promedio 6m

Medidas a 7 metros de separación

Señal detectada

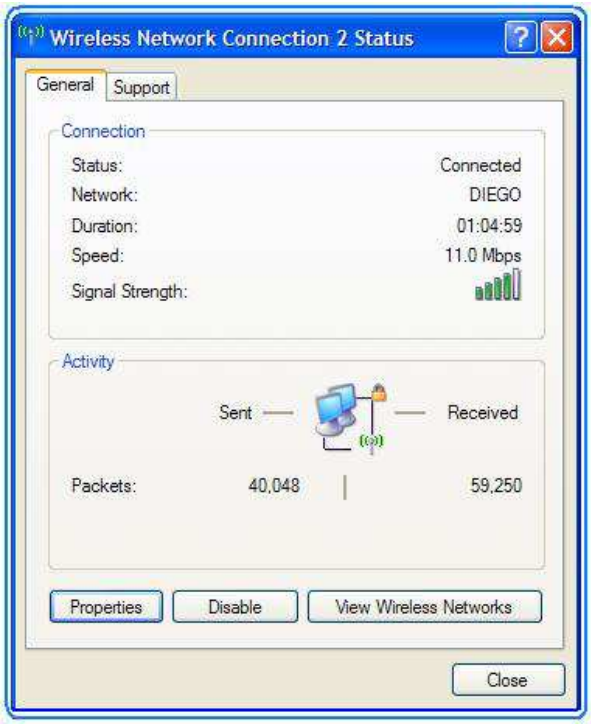


Figura 26 Estado de conexión 7m

Nivel de potencia (d = 7 m, p = - 58.50 dBm, f = 2.4383 GHz)

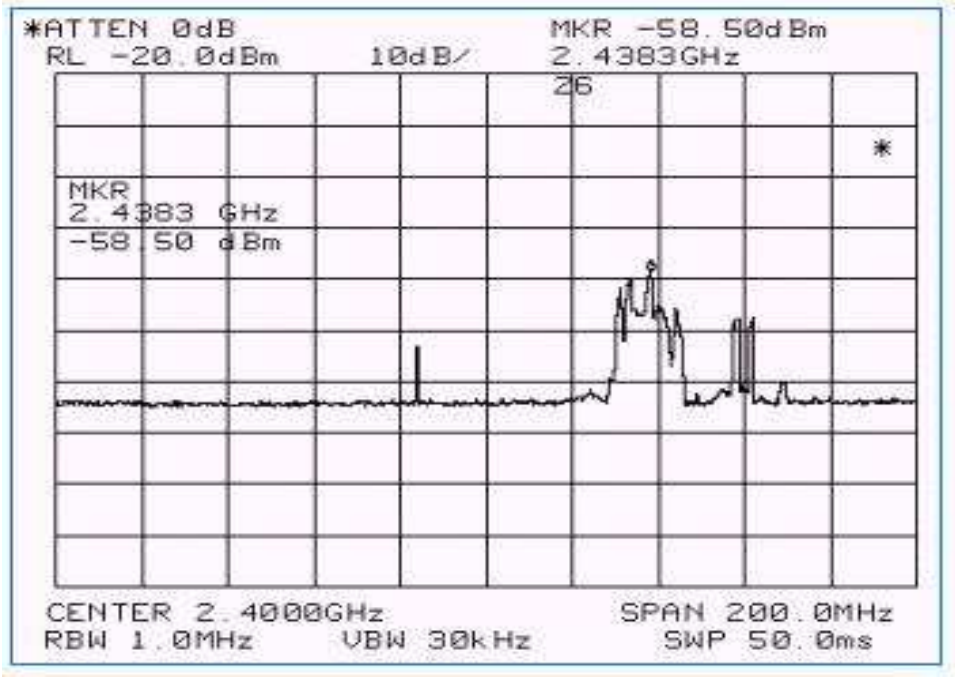


Figura 27 Nivel de potencia 7m

Respuesta entre las estaciones

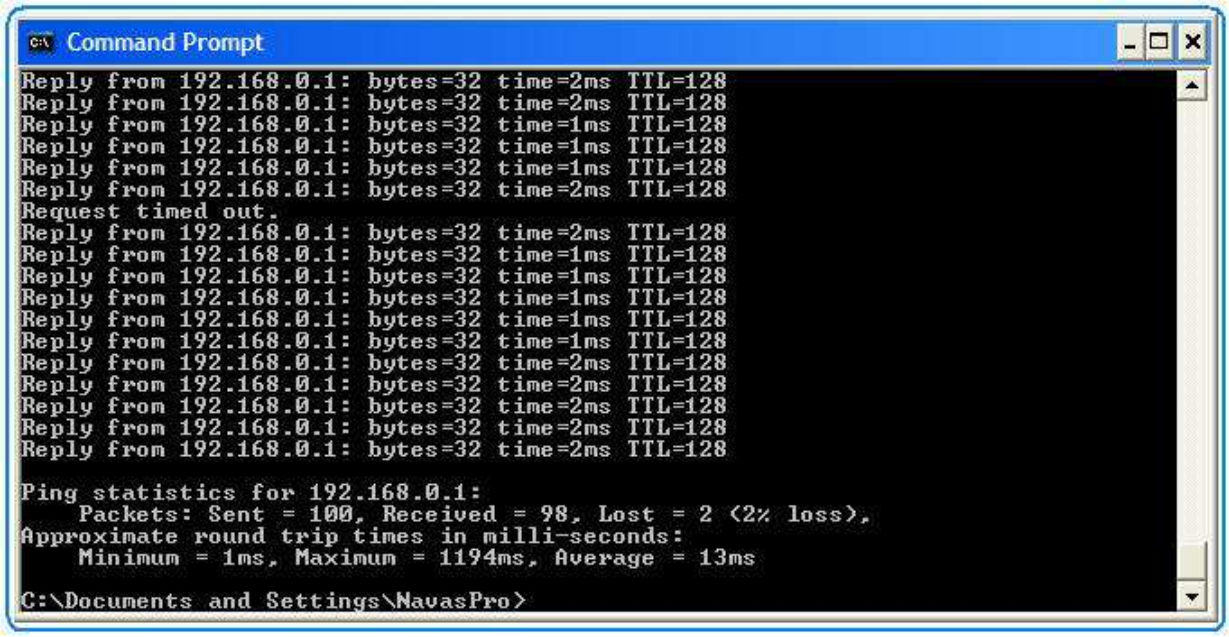


Figura 28 Ping entre las estaciones 7m

Velocidad

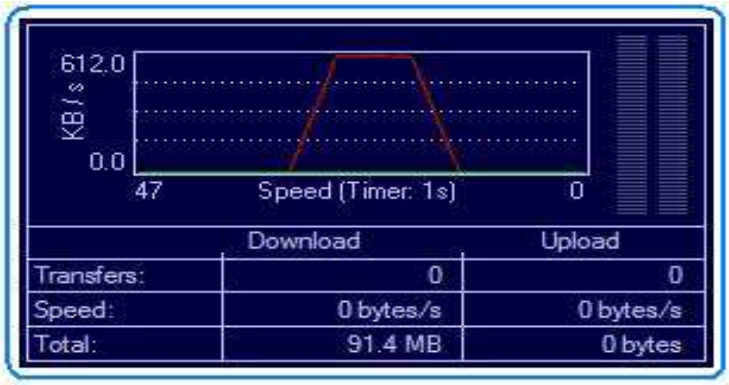


Figura 29 Velocidad 7m

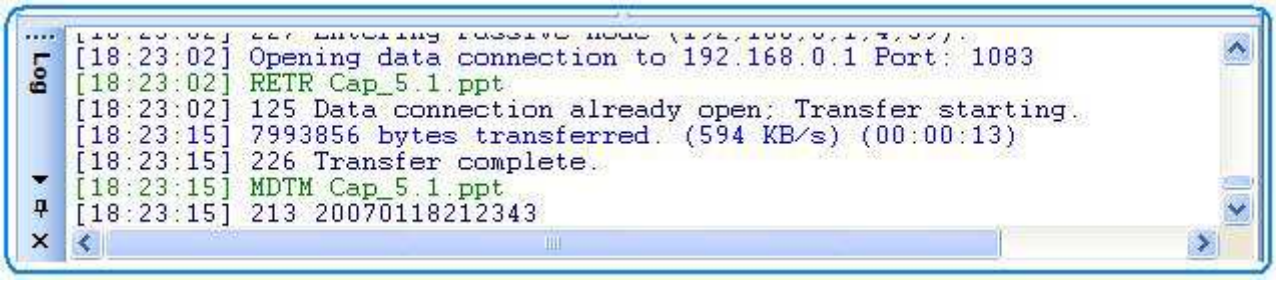


Figura 30 Velocidad Promedio 7m

Medidas a 8 metros de separación

Señal detectada

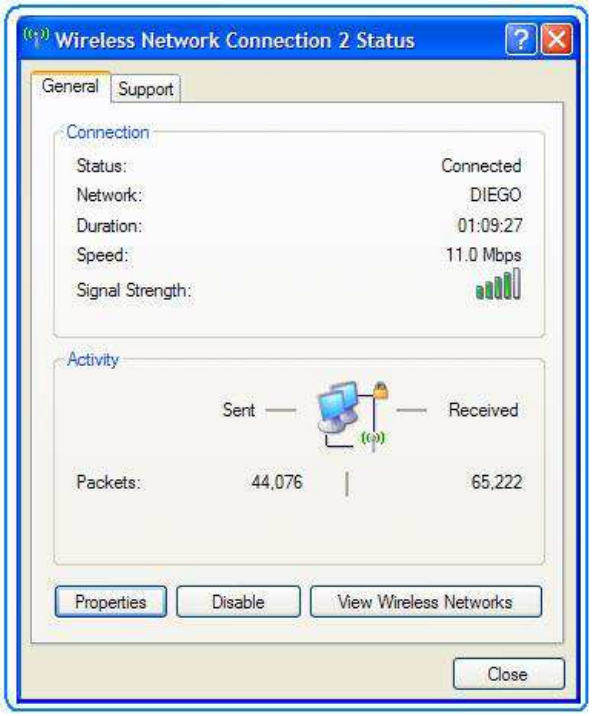


Figura 31 Estado de conexión 8m

Nivel de potencia (d = 8 m, p = - 66.83 dBm, f = 2.4323 GHz)

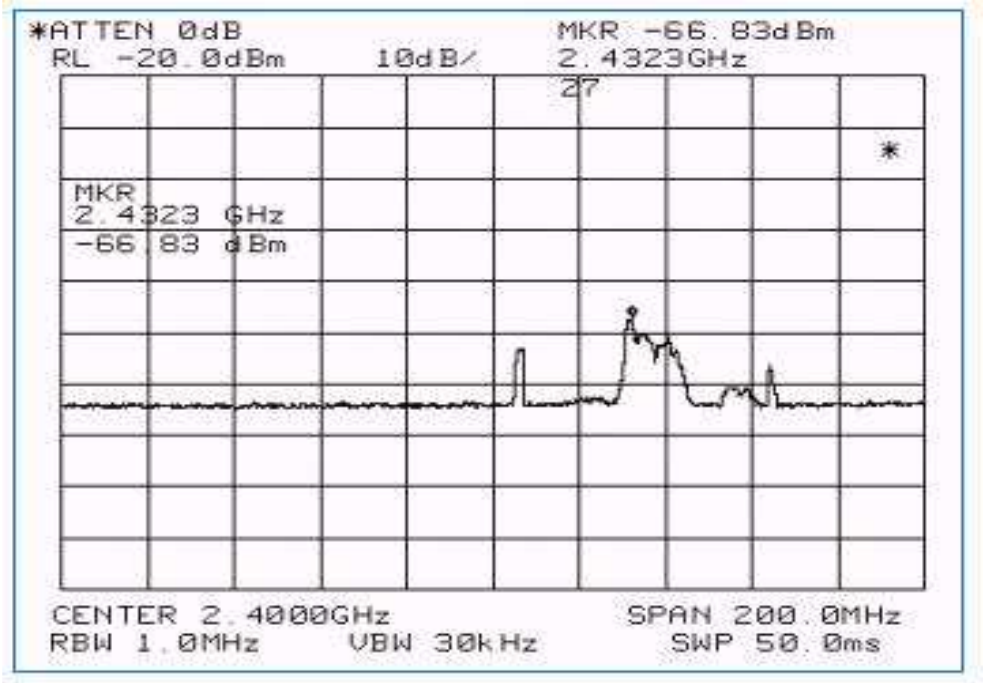


Figura 32 Nivel de potencia 8m

Respuesta entre las estaciones

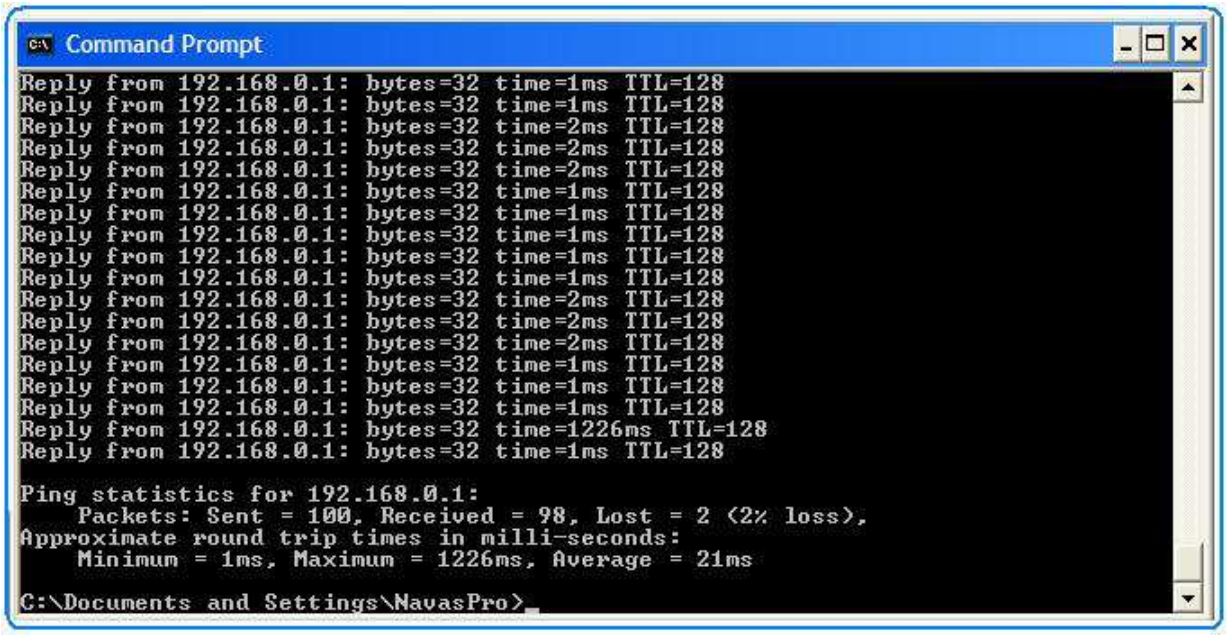


Figura 33 Ping entre las estaciones 8m

Velocidad

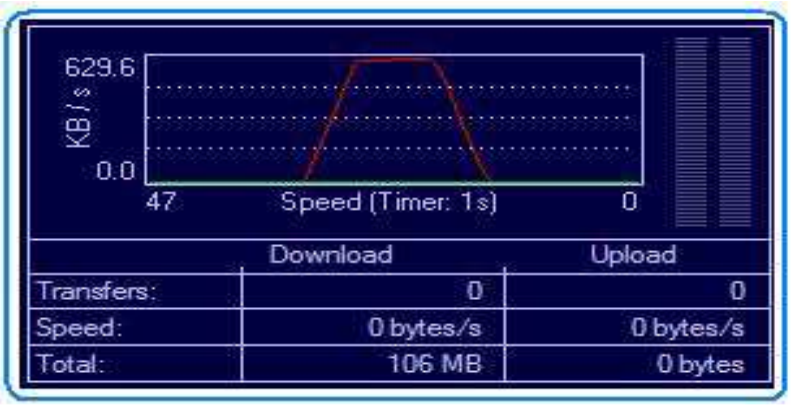


Figura 34 Velocidad 8m

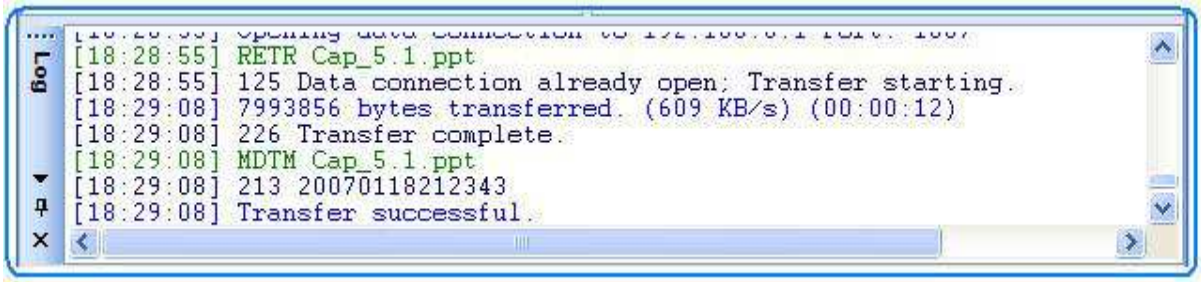


Figura 35 Velocidad 8m

Medidas a 9 metros de separación

Señal detectada

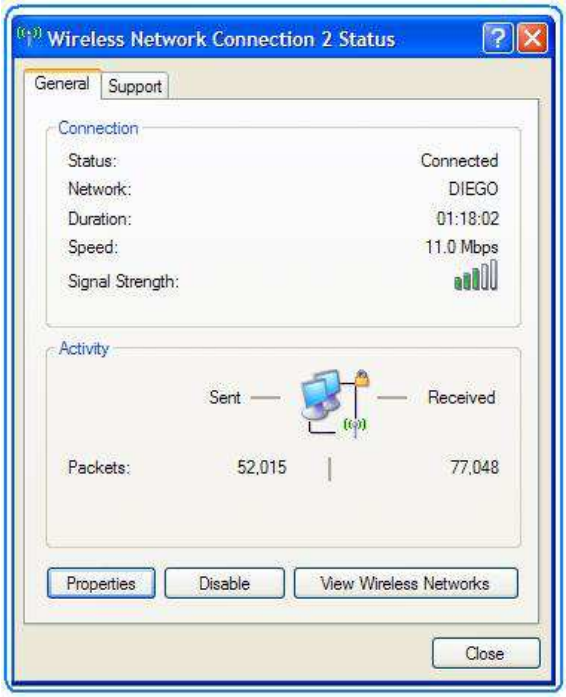


Figura 36 Estado de conexión 9m

Nivel de potencia (d = 9 m, p = - 66.50 dBm, f = 2.4347 GHz)

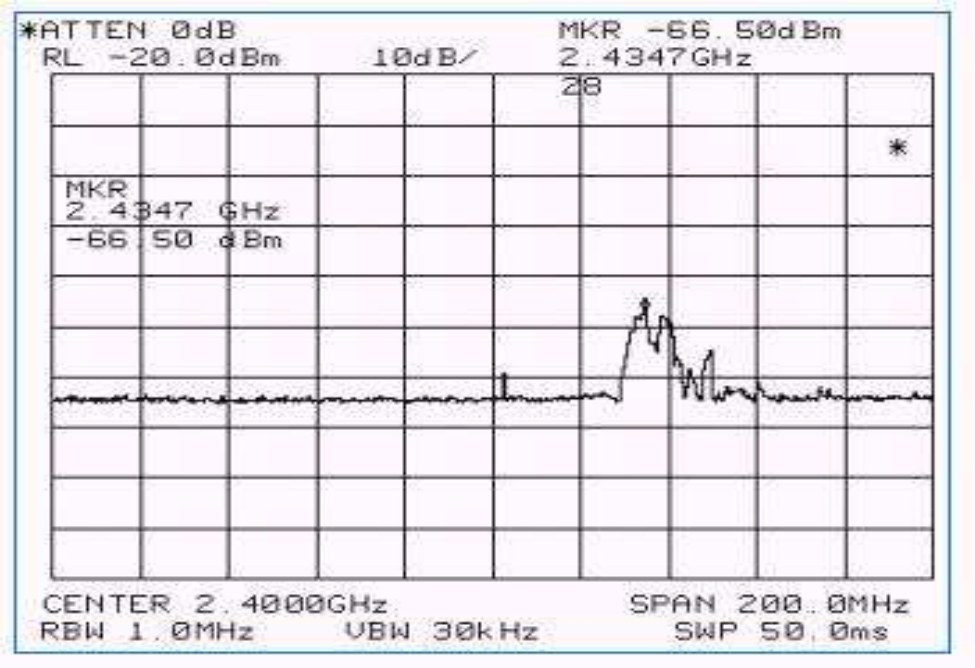


Figura 37 Nivel de potencia 9m

Respuesta entre las estaciones

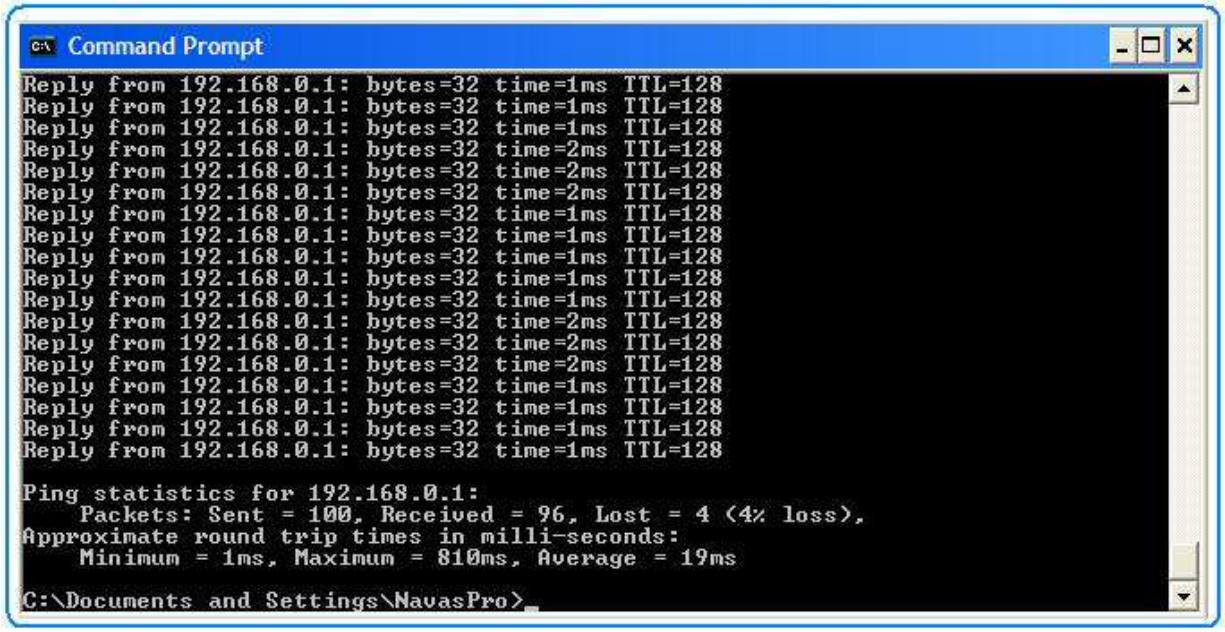


Figura 38 Ping entre las estaciones 9m

Velocidad

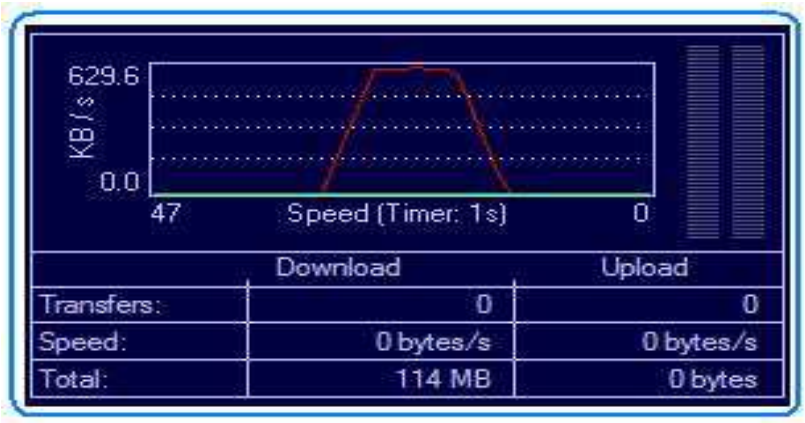


Figura 39 Velocidad 9m

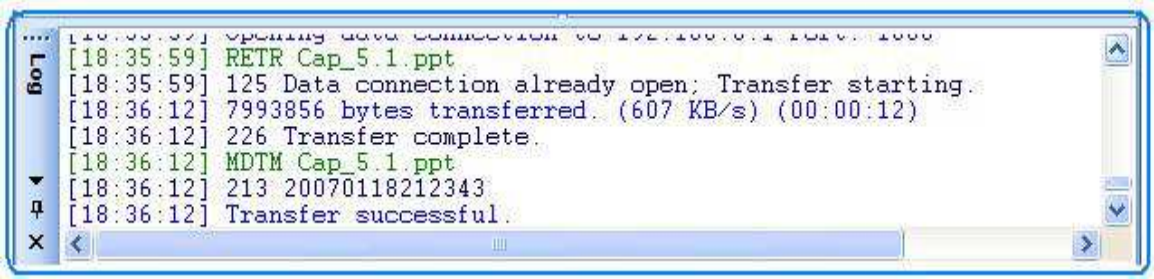


Figura 40 Velocidad Promedio 9m

Medidas a 10 metros de separación

Señal detectada

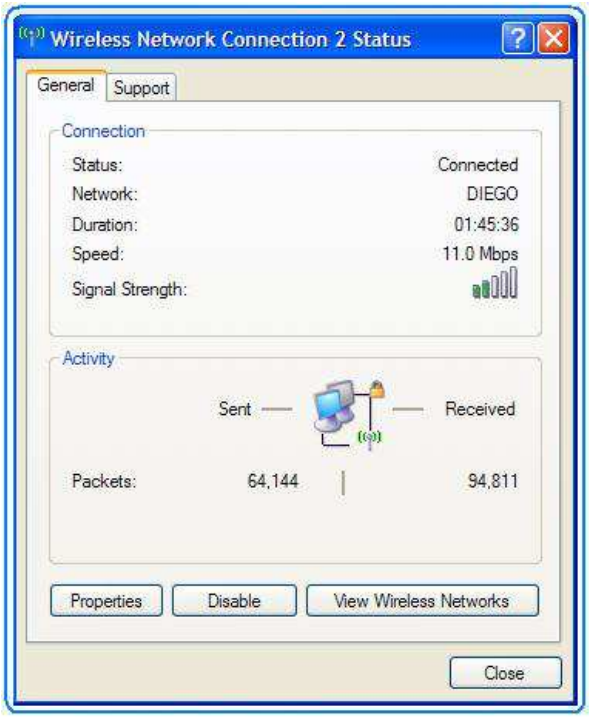


Figura 41 Estado de conexión 10m

Nivel de potencia (d = 10 m, p = - 68.83 dBm, f = 2.4207 GHz)

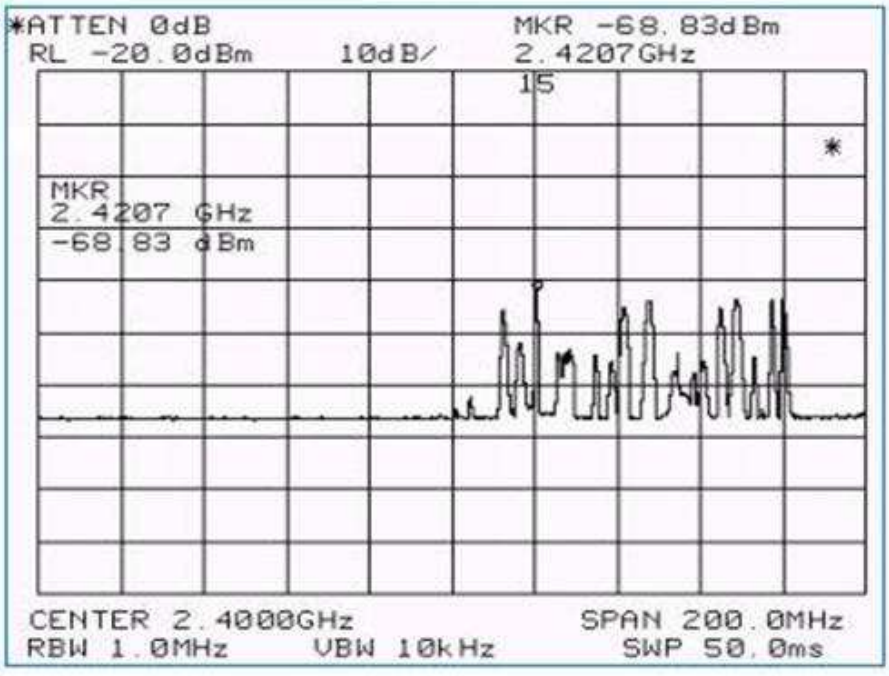


Figura 42 Nivel de potencia 10m

Respuesta entre las estaciones

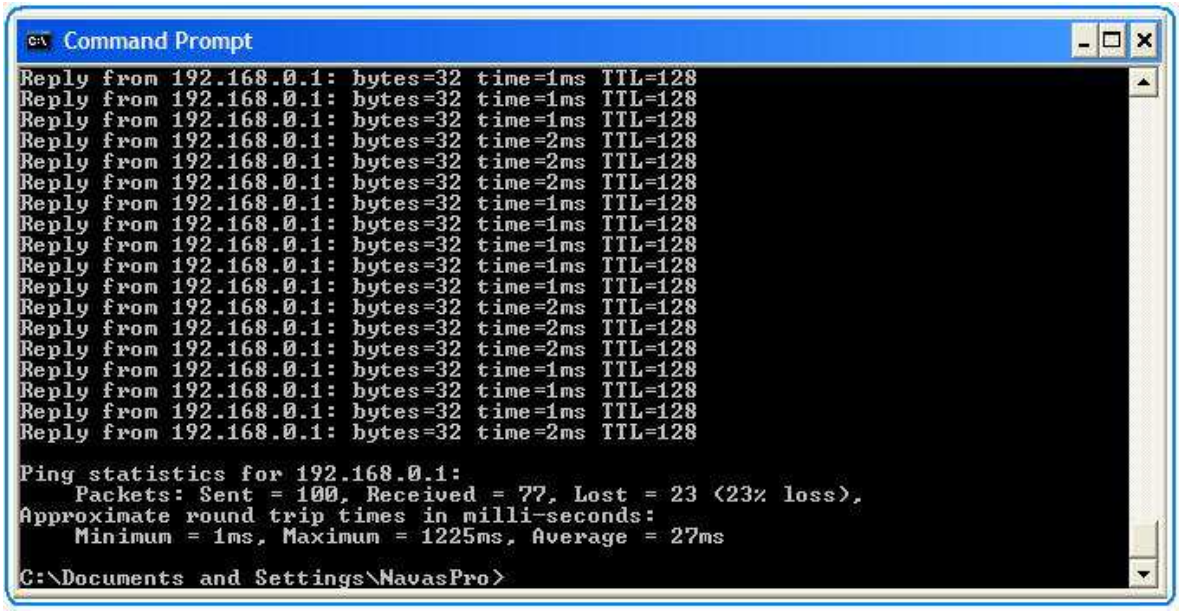


Figura 43 Ping entre las estaciones 10m

Velocidad

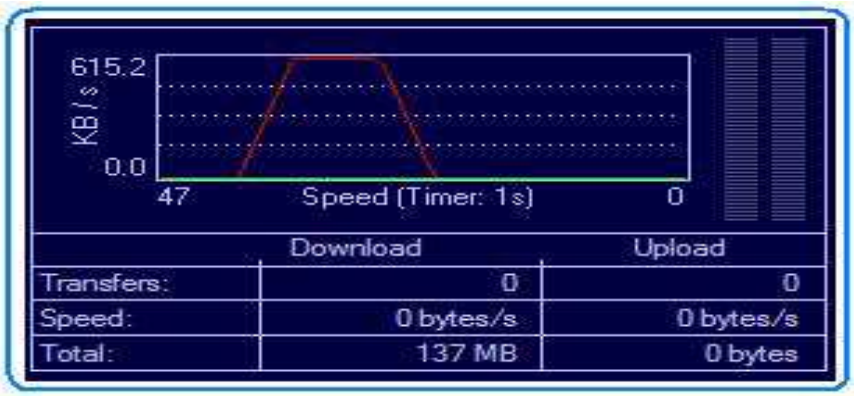


Figura 44 Velocidad 10m

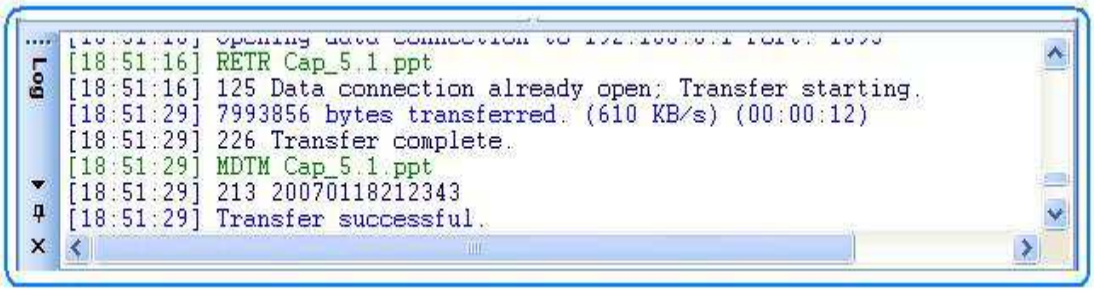


Figura 45 Velocidad 10m

Medidas a 12 metros de separación

Señal detectada

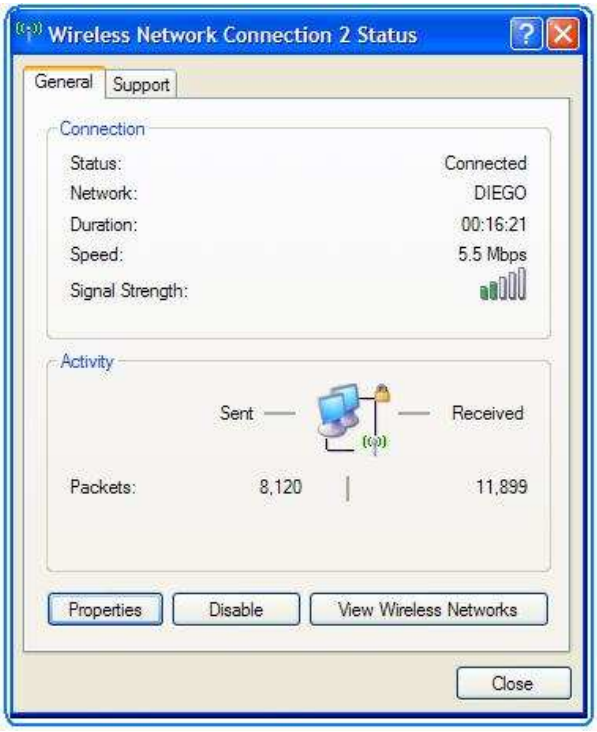


Figura 46 Estado de conexión 12m

Nivel de potencia (d = 12 m, p = - 69.39 dBm, f = 2.4397 GHz)

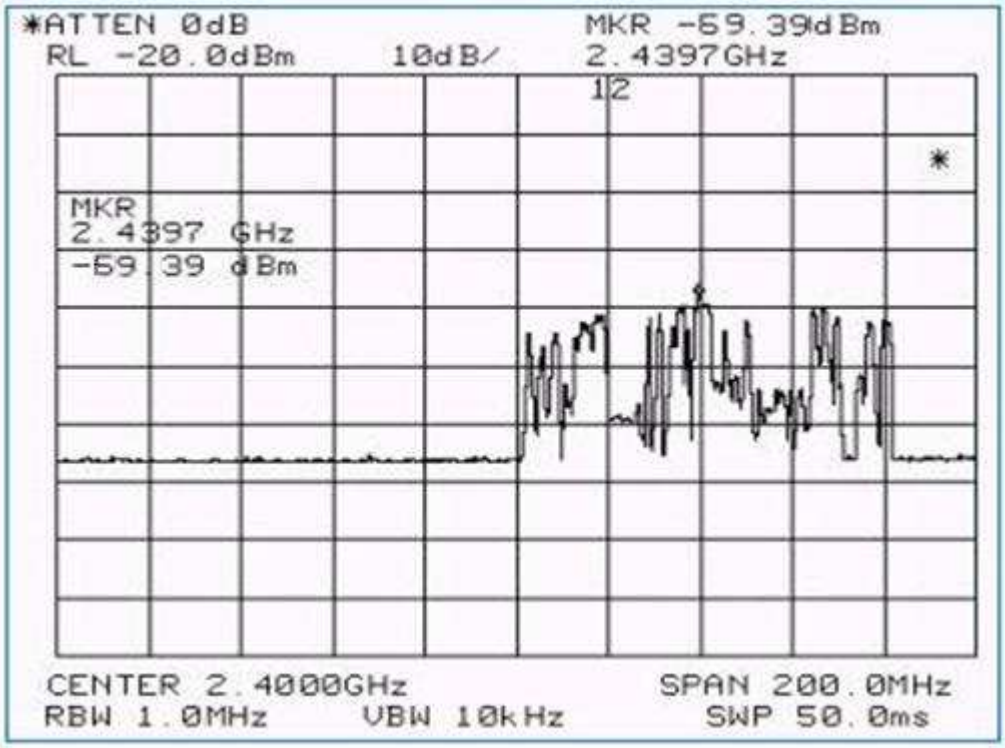


Figura 47 Nivel de potencia 12m

Respuesta entre las estaciones

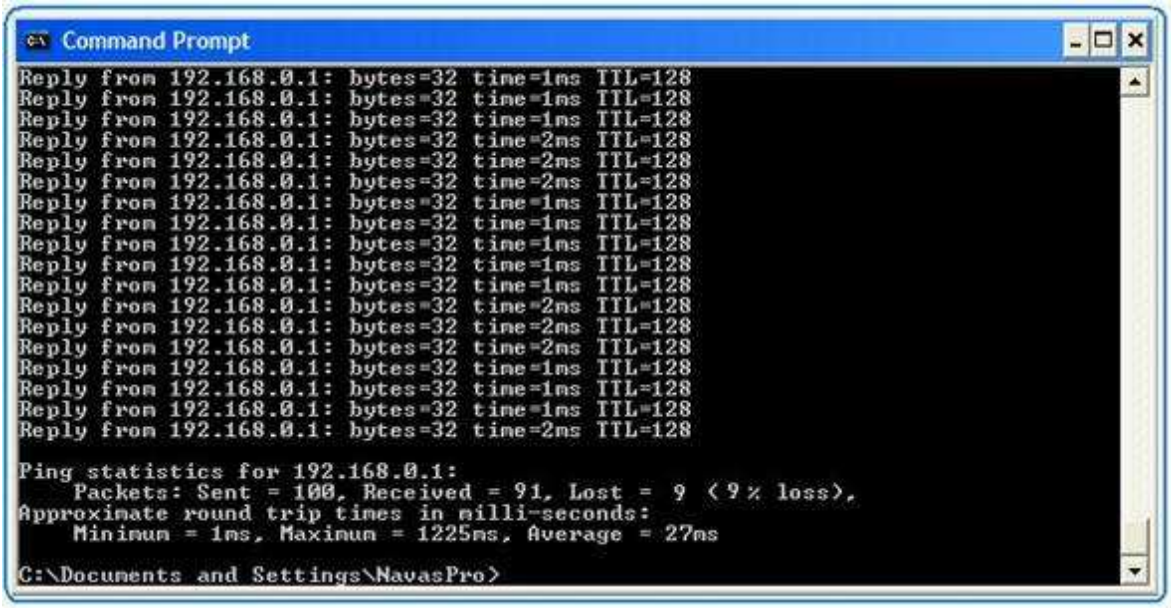


Figura 48 Ping entre las estaciones 12m

Velocidad

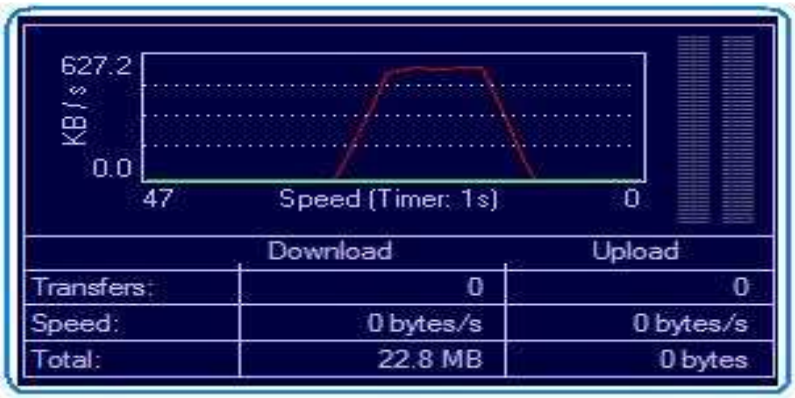


Figura 49 Velocidad 12m

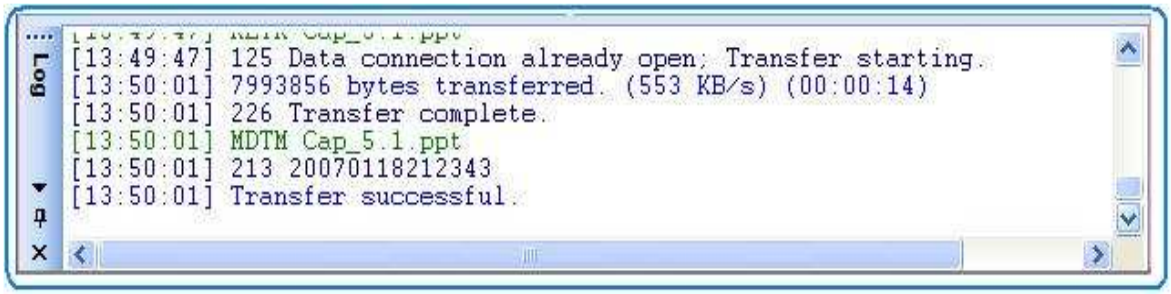


Figura 50 Velocidad Promedio 12m

Medidas a 15 metros de separación

Señal detectada

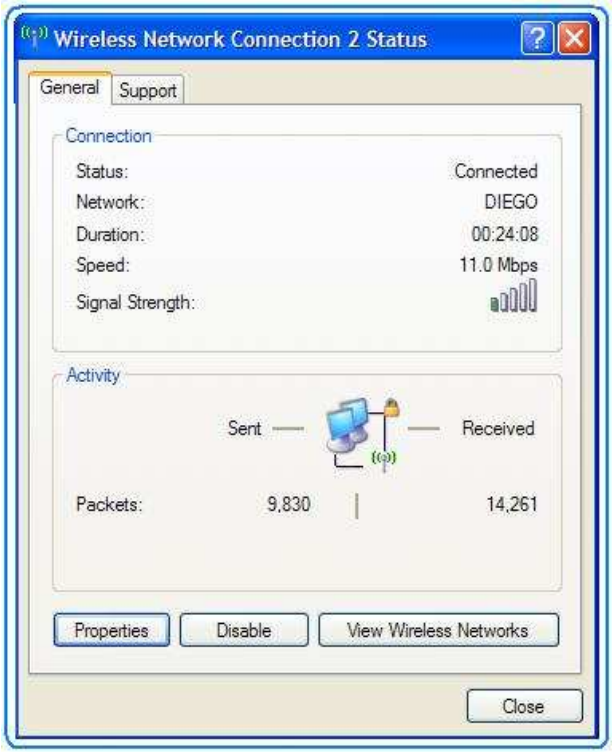


Figura 51 Estado de conexión 15m

Nivel de potencia (d = 15 m, p = - 70.50 dBm, f = 2.4400 GHz)

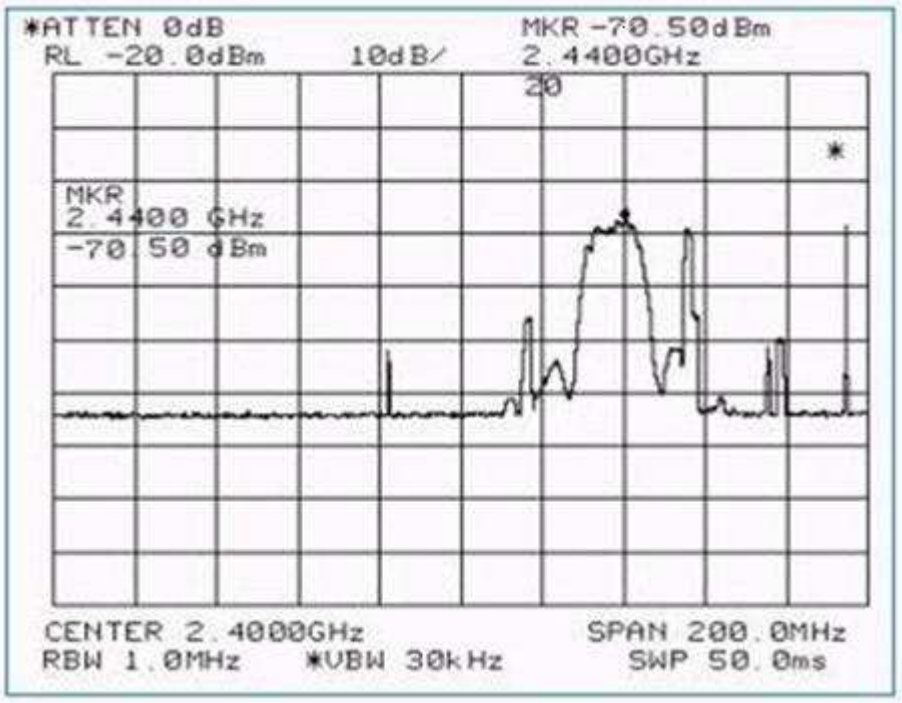


Figura 52 Nivel de potencia 15m

Respuesta entre las estaciones

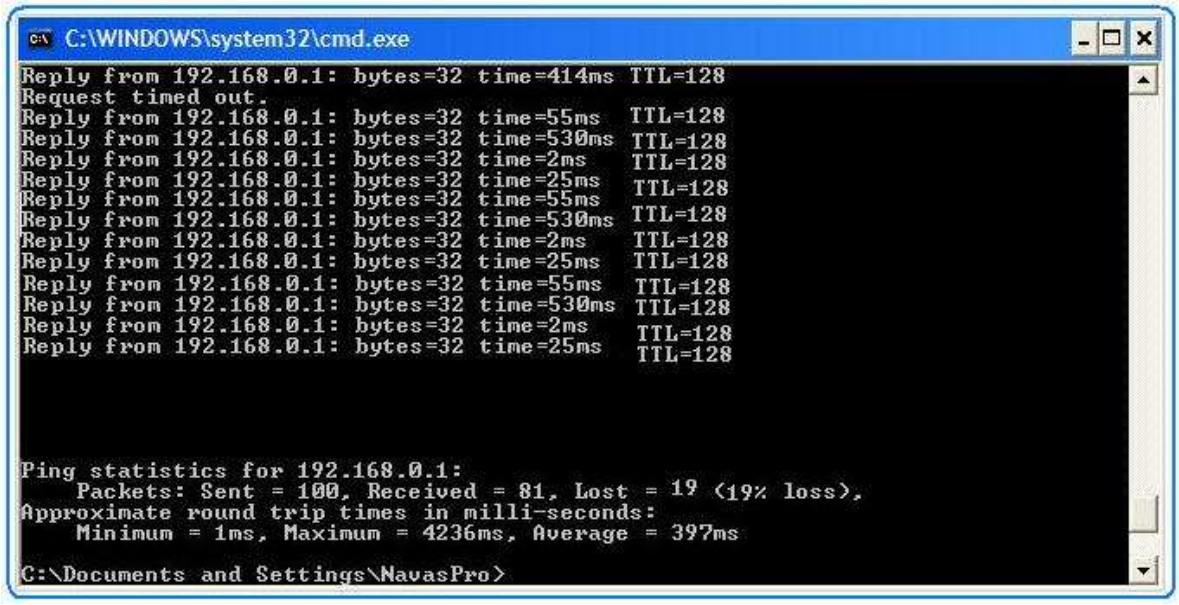


Figura 53 Ping entre las estaciones 15m

Velocidad

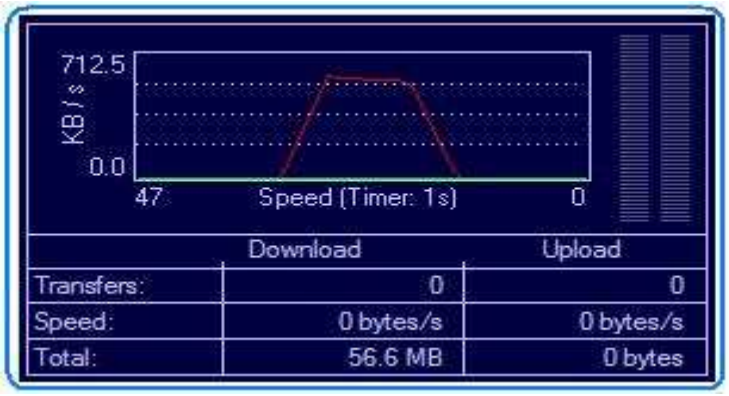


Figura 54 Velocidad 15m

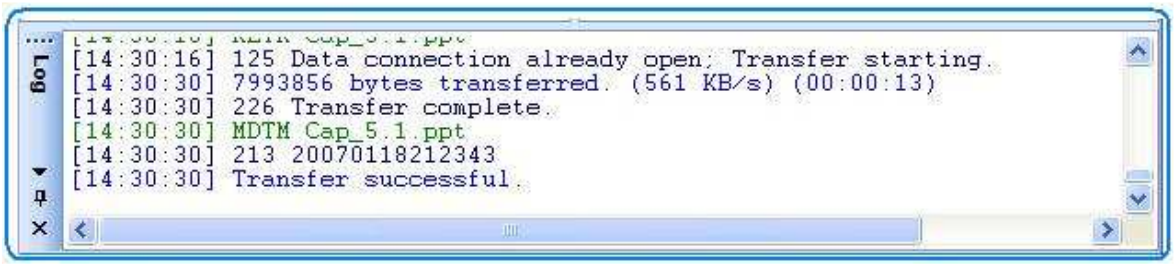


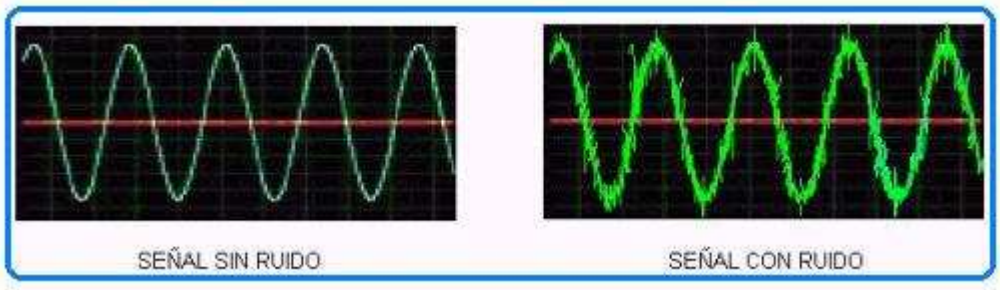
Figura 55 Velocidad Promedio 15m

ANEXO F

Ruido [11]

Ruido

El ruido es una señal de naturaleza aleatoria que contamina a la señal deseada.



Señal sin Ruido ***Señal con Ruido***

Figura. 1 Señal con ruido

El ruido reduce la capacidad del receptor para reconocer correctamente los símbolos, limitando de esta manera la velocidad de transmisión. Existe una gran cantidad de causas por las que el ruido se puede incrementar; a continuación se presenta una tabla resumida de las diferentes fuentes de ruido.

FUENTES DE RUIDO	**CAUSAS**
EXTERNAS	▪ Atmosféricas ▪ Galácticas ▪ Ruido generado por el hombre: Ignición, motores, otros.
INTERNAS	▪ Pérdidas disipativas ▪ Pérdidas de dispositivos
TÉRMICAS	▪ Por movimiento de electrones en los conductores ▪ Agitación térmica en todos los componentes

Tabla. 1 Fuentes de Ruido

A continuación solo definiremos el ruido térmico debido a que es el ruido con mayor importancia en un sistema electrónico de comunicaciones.

- **RUIDO TÉRMICO:** Este ruido se asocia con el movimiento rápido y aleatorio de los electrones dentro de un conductor, producido por la agitación térmica.

El ruido térmico es también llamado Ruido Blanco o de Jonson, esta presente en todas las componentes de frecuencia. Este ruido es predecible, aditivo, y esta presente en todos los dispositivos; es por ello que es el mas importante de todos los ruidos.

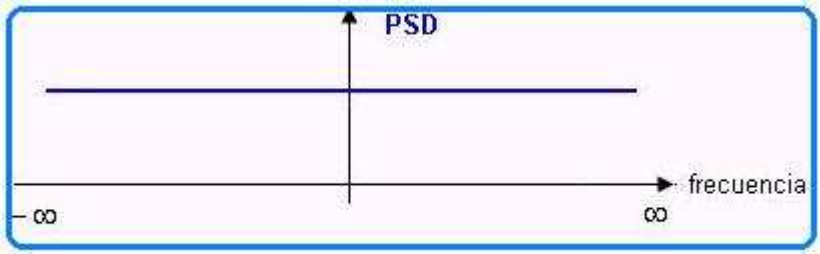


Figura. 2 Función Densidad Espectral de Potencia

$$PSD = N_0 = K \bullet T$$

T = Temperatura absoluta ($^0K = 273^0 + ^0C$)

K = Constante de Boltzman ($1.38 \times 10^{-}23$ J / ºK)

La potencia de ruido viene dado por la siguiente expresión:

$$N = N_0 \bullet AB$$

AB = ancho de banda (hertz)

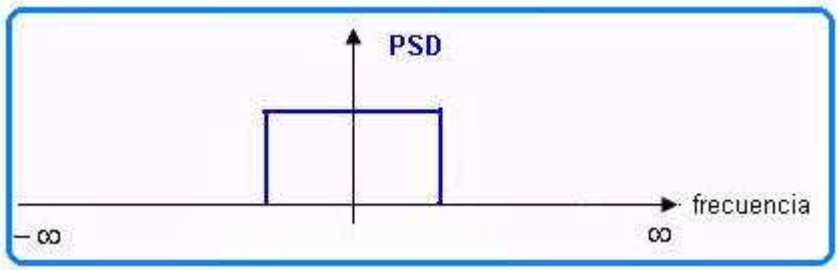


Figura. 3 Ruido Coloriado

ANEXO G

Modulación por división ortogonal de frecuencias (OFDM) [2]

Modulación por división ortogonal de frecuencias (OFDM)

Esta tecnología sólo está presente en 802.11a y en 802.11g como principal variación respecto a 802.11 y 802.11b. Se observa que la modulación pasa a ser OFDM (Orthogonal Frequency Division Multiplexing), en vez de la clásica y más fiable hasta entonces CCK (Complimentary Code Keying); aunque esta norma pueda coexistir en los puntos de acceso 802.11g, conservando a su vez la banda de los 2.4Ghz (precedido de un CCK RTS).

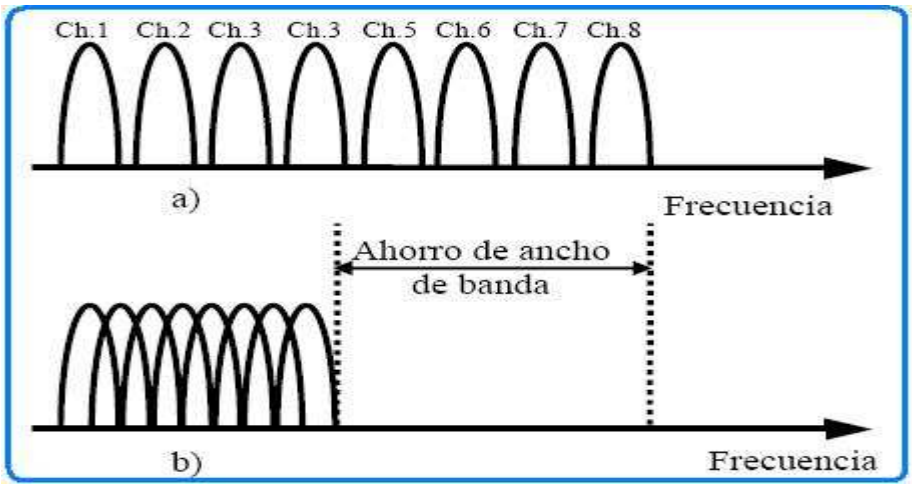


Figura 1 OFDM Orthogonal Frequency division Multiplexing

- Técnica multiportadora original.
- Modulación de portadoras ortogonales**.**

Durante los últimos años, se ha aceptado OFDM como tecnología de base para el 802.16a, que es un estándar de IEEE para redes de área metropolitana inalámbrica; y que puede proveer extensión inalámbrica para acceso de última milla de banda ancha en instalaciones de cable y DSL. El mismo cubre el rango de frecuencias de 2 a 11 GHz y alcanza hasta 50 kilómetros lineales, brindando conectividad de banda ancha inalámbrica sin necesidad de que exista una línea directa de visión a la estación de base. La velocidad de transmisión de datos puede llegar a 70 Mbps.

Una estación de base típica puede albergar hasta seis sectores. La calidad de servicio está integrada dentro del MAC, permitiendo la diferenciación de los niveles de servicio.

El origen del OFDM está en las décadas de los 50 y 60 en aplicaciones de uso militar, y trabaja dividiendo el espectro disponible en múltiples subportadoras. La transmisión sin línea de visión ocurre cuando entre el receptor y el transmisor existen reflexiones o absorciones de la señal, lo que resulta en una degradación de la señal recibida, que se manifiesta por medio de los siguientes efectos: atenuación plana, atenuación selectiva en frecuencia o interferencia inter-símbolo.

Estos efectos se mantienen bajo control con el W-OFDM, que es una tecnología propietaria de Wi-LAN, quién recibió en 1994 la patente 5.282.222 para comunicaciones inalámbricas de dos vías y banda ancha OFDM (WOFDM). Esta patente es la base para los estándares 802.11a, 802.11g, 802.11a R/A, 802.16a, estándares para HiperMAN.

Los sistemas W-OFDM incorporan además: estimación de canal, prefijos cíclicos y códigos Reed-Solomon de corrección de errores. Wi-LAN introdujo su línea de productos BWS 3000 basada en W-OFDM en octubre del 2001.

Actualmente ya ha introducido al mercado la tercera generación de equipos OFDM siendo el único proveedor mundial con una sólida experiencia en esta tecnología probada a través de la excelencia de sus productos.

Las tecnologías 802.11a y 802.11b definen una capa física diferente. Los emisores 802.11b transmiten a 2.4 GHz y envían datos a tasas tan altas como 11 Mbps usando modulación DSSS; mientras que los emisores 802.11a y 802.11g transmiten a 5 y 2,4 GHz respectivamente y envían datos a tasas de hasta 54 Mbps usando OFDM.

OFDM es una tecnología de modulación digital, una forma especial de modulación multi-portadora (multi-carrier) considerada la piedra angular de la próxima

generación de productos y servicios de radio frecuencia de alta velocidad, para uso tanto personal como corporativo. La técnica de espectro disperso de OFDM distribuye los datos en un gran número de portadoras (carriers) que están espaciados entre sí en distintas frecuencias precisas. Ese espaciado evita que los demoduladores vean frecuencias distintas a las suyas propias.

OFDM tiene una alta eficiencia de espectro y menor distorsión multi-ruta. Actualmente OFDM no sólo se usa en las redes inalámbricas LAN 802.11a y 802.11g, si no también en comunicaciones de alta velocidad por vía telefónica como las ADSL y en difusión de señales de televisión digital terrestre en Europa, Japón y Australia.

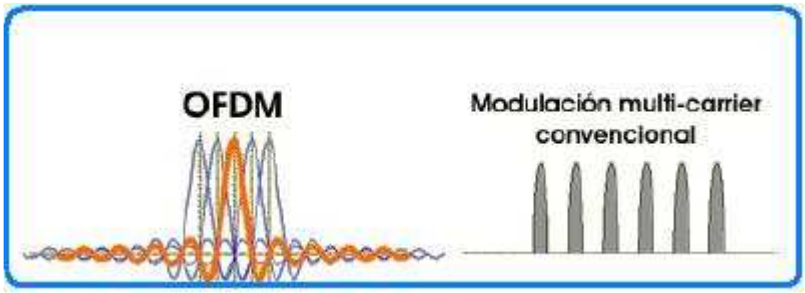


Figura 2 Espectro de OFDM solapado

ANEXO H

Modulación Wi-fi [2]

DSSS

FHSS

INFRARROJOS

- **Espectro Expandido por Salto de Frecuencia (FHSS)**

La tecnología de espectro expandido por salto en frecuencia (FHSS) consiste en transmitir una parte de la información en una determinada frecuencia durante un intervalo de tiempo llamado dwell time inferior a 400 ms. Pasado este tiempo se cambia la frecuencia de emisión y se sigue transmitiendo a otra frecuencia. De esta manera cada tramo de información se va transmitiendo en una frecuencia distinta durante un intervalo muy corto de tiempo.

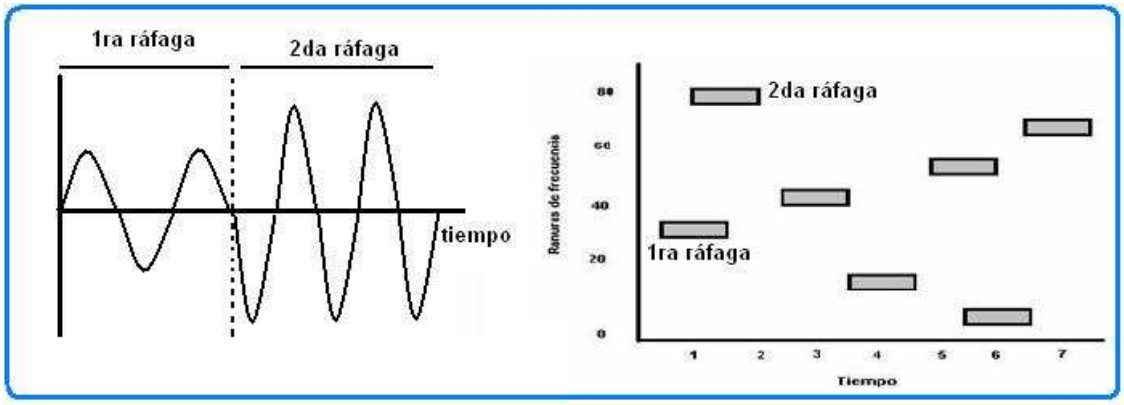


Figura 1 Codificación con Salto en Frecuencia

El orden en los saltos en frecuencia se determina según una secuencia pseudo aleatoria almacenada en unas tablas, que tanto el emisor y el receptor deben conocer.

Si se mantiene la sincronización en los saltos de frecuencias se consigue que, aunque en el tiempo se cambie de canal físico, a nivel lógico se mantiene un solo canal por el que se realiza la comunicación.

Esta técnica utiliza la zona de los 2.4GHz, la cual organiza en 79 canales con un ancho de banda de 1MHz cada uno. No obstante el número real de canales que son usados se regula por las autoridades competentes de cada país. El número de saltos por segundo está también regulado en cada país, así, por ejemplo, Estados Unidos fija una tasa mínima de saltos de 2,5 por segundo.
El estándar IEEE 802.11 define la modulación aplicable en este caso. Se utiliza la modulación en frecuencia FSK (Frequency Shift Keying), con una velocidad

de 1 Mbps ampliable a 2 Mbps. En la revisión 802.11b del estándar, la velocidad también ha aumentado a 11Mbps.

Formato de la trama FHSS

Figura 2 Trama FHSS

- **Preámbulo:** contiene dos subcampos separados: el campo de preámbulo de sincronización (SYNC) y el delimitador de comienzo de trama (Start Frame Delimiter).

- **Sincronismo:** contiene 80 bits con un patrón alternativo de unos-ceros, comenzando con cero y terminando con uno. Se usa para detectar una señal potencialmente válida, seleccionar una de las antenas si se usa un sistema de diversidad y sincronizarse temporalmente.

- **Delimitador de comienzo de trama (SFD):** contiene un patrón de 16 bits con patrón 0000 1100 1011 1101 que define el tempo de la trama.

- **Cabecera:** contiene 3 subcampos: Longitud de 12 bits, Señalización de 12 bits y Control de Errores de 16 bits.

- **Longitud:** indica la longitud del campo de datos que puede ser de hasta 4095 octetos.

- **Señalización:** campo de 4 bits que indica la velocidad de transmisión de los datos desde 1 Mbps a 4.5 Mbps en incrementos de 0.5 Mbps.

- **Control de errores (HEC):** campo de 16 bits para detección de errores que utiliza el polinomio generador CCITT CRC-16 $G(x) = X^{16} + X^{12} + X^{15} + 1$

El preámbulo y la cabecera son siempre transmitidos a 1 Mbps. El resto de la trama es transmitido a la velocidad indicada en el campo de señalización. Para minimizar el efecto de las reflexiones multitrayecto el FHSS tiene un salto de distancia mínima entre frecuencias. Esto es debido a que las reflexiones del salto anterior tienen un efecto mínimo sobre el siguiente salto debido a que, transcurrido el retardo producido por la reflexión hasta llegar al receptor, éste se encontrará entonces esperando por información en una frecuencia diferente.

❖ Espectro Expandido por Secuencia Directa (DSSS)

DSSS es el segundo tipo de modulación soportado por el IEEE 802.11 y el único especificado en el IEEE 802.11b, soportando velocidades de transmisión de 5.5 y 11Mbps.

En el caso de Estados Unidos y Europa la tecnología DSSS utiliza un rango de frecuencias que va desde los 2,4 GHz hasta los 2,4835 GHz, lo que permite tener un ancho de banda total de 83,5 MHz. Este ancho de banda se subdivide en canales de 5 MHz, lo que hace un total de 14 canales independientes. Cada país esta autorizado a utilizar un subconjunto de estos canales. En Europa existen 13 canales disponibles, excepto en Francia de los cuales solo 3 no están solapados.

Con arreglo a IEEE 802.11 debe existir una separación de 30 MHz entre las frecuencias centrales de los canales si las celdas se solapan y/o son adyacentes para no causar interferencias. En IEEE 802.11b la separación se reduce a 25 MHz. Esto significa que pueden existir 3 celdas con zonas solapadas y/o adyacentes sin causar interferencias entre ellas, tal y como se muestra en la Figura. 1.28

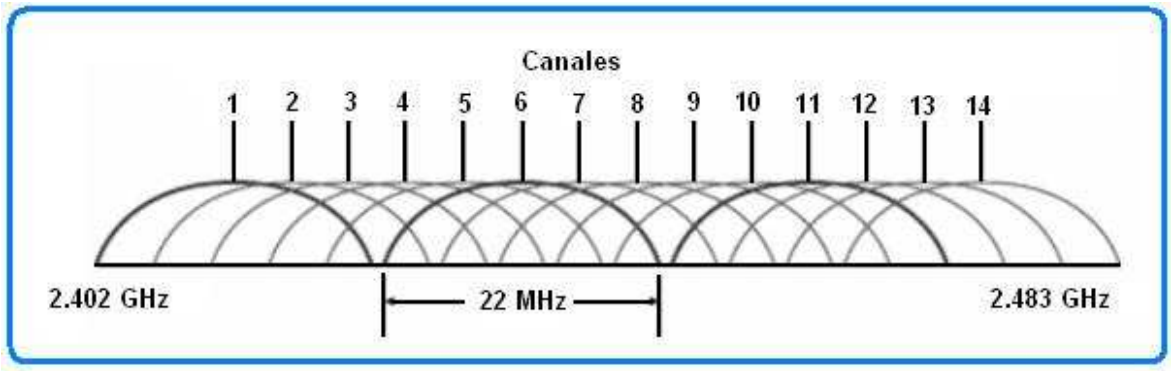


Figura 3 Canales DSSS

En configuraciones donde existan mas de una celda, éstas pueden operar simultáneamente y sin interferencias, siempre y cuando la diferencia entre las frecuencias centrales de las distintas celdas sea de al menos 30 MHz, lo que reduce a tres el número de canales independientes y funcionando simultáneamente en el ancho de banda total de 83,5 MHz.

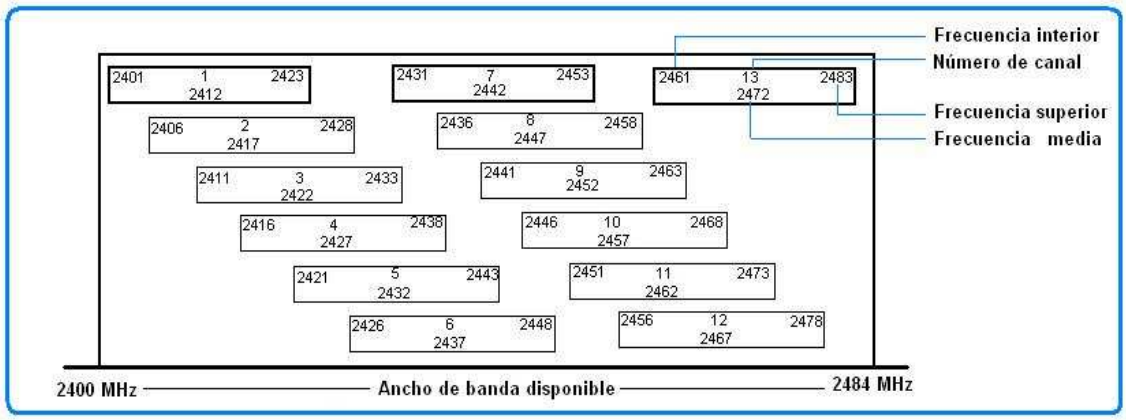


Figura 4 Tabla de Frecuencias DSSS

Formato de la trama DSSS

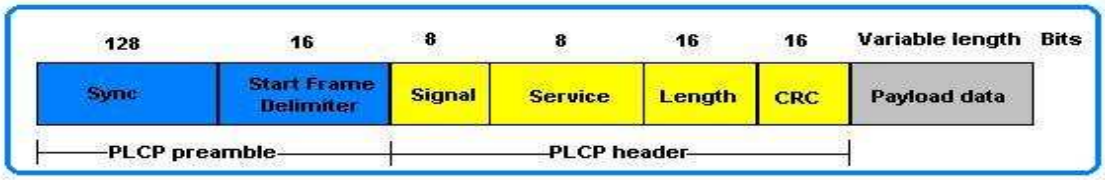


Figura 5 Trama DSSS

Al igual que en FHSS el preámbulo y la cabecera se transmiten siempre a 1 Mbps y el campo de señalización indica la velocidad de transmisión de los

datos. En el 802.11b este campo soporta velocidades mayores que el original de 1 y 2 Mbps (5,5 Mbps y 11 Mbps).

- **Sincronismo:** contiene una codificación de 128 bits que garantiza la sincronización previa del receptor.

- **Delimitador de comienzo de trama (SFD):** señaliza el comienzo de la trama real después del preámbulo.

- **Señal:** indica a la capa física que tipo de modulación se utilizara en la transmisión. La velocidad será igual al valor de este campo multiplicado por 1000Kbps.

- **Servicio:** reservado para usos futuros.

- **Longitud:** entero sin signo de 16 bits que indica el número de microsegundos requerido para transmitir los datos.

- **CRC:** los campos de cabecera están protegidos por una secuencia de verificación de trama CRC-16.

Proceso de modulación DSSS

En esta técnica se genera un patrón de bits redundante (señal de chip) para cada uno de los bits que componen la señal. Cuanto mayor sea esta señal, mayor será la resistencia de la señal a las interferencias. El estándar IEEE 802.11 recomienda un tamaño de 11 bits, pero el óptimo es de 100. En recepción es necesario realizar el proceso inverso para obtener la información original.

La secuencia de bits utilizada para modular los bits se conoce como secuencia de Barker (también llamado código de dispersión o Pseudo Noise). Es una secuencia rápida diseñada para que aparezca aproximadamente la misma

cantidad de 1 que de 0. Un ejemplo de esta secuencia es el siguiente:+1 -1 +1 +1 -1 +1 +1 +1 -1 -1 -1 -1.

Solo los receptores a los que el emisor haya enviado previamente la secuencia podrán recomponer la señal original. Además, al sustituir cada bit de datos a transmitir por una secuencia de 11 bits equivalente, aunque parte de la señal de transmisión se vea afectada por interferencias, el receptor aún puede reconstruir fácilmente la información a partir de la señal recibida.

A continuación podemos observar como se utiliza la secuencia de Barker para codificar la señal original a transmitir:

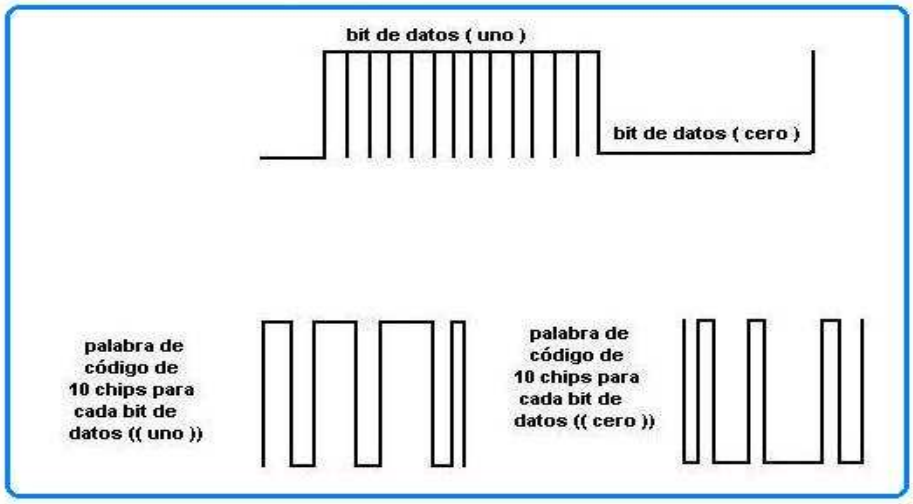


Figura 6 Secuencia Directa (DSSS)

Una vez aplicada la señal de chip, el estándar IEEE 802.11 ha definido dos tipos de modulación para la técnica de espectro ensanchado por secuencia directa (DSSS), la modulación DBPSK (Differential Binary Phase Shift Keying) y la modulación DQPSK (Differential Quadrate Phase Shift Keying), que proporcionan una velocidad de transferencia de 1 y 2 Mbps respectivamente.

- **Infrarrojo**

La verdad es que IEEE 802.11 no ha desarrollado todavía en profundidad esta área y solo menciona las características principales de la misma:

Entornos muy localizados, un aula concreta, un laboratorio, un edificio.

- Modulaciones de 16-PPM y 4-PPM que permiten 1 y 2 Mbps de transmisión.
- Longitudes de onda de 850 a 950 nanómetros de rango.
- Frecuencias de emisión entre 3,15*10E14 Hz y 3,52*10E14 Hz.

Las WLAN por infrarrojos son aquellas que usan el rango infrarrojo del espectro electromagnético para transmitir información mediante ondas por el espacio libre. Los sistemas de infrarrojos se sitúan en altas frecuencias, justo por debajo del rango de frecuencias de la luz visible. Las propiedades de los infrarrojos son, por tanto, similares a las que tiene la luz visible. De esta forma los infrarrojos son susceptibles de ser interrumpidos por cuerpos opacos pero se pueden reflejar en determinadas superficies.

Para describir esta capa física seguiremos las especificaciones del IrDA (Infrared Data Association) organismo que ha estado desarrollando estándares para conexiones basadas en infrarrojos.

Para la capa infrarroja tenemos las siguientes velocidades de transmisión:

- 1 y 2 Mbps Infrarrojos de modulación directa.
- 4 Mbps mediante Infrarrojos portadora modulada.
- 10 Mbps Infrarrojos con modulación de múltiples portadoras.

c.1 Clasificación de Infrarrojo

De acuerdo al ángulo de apertura con que se emite la información en el transmisor, los sistemas infrarrojos pueden clasificarse en sistemas de corta apertura, también llamados de rayo dirigido o de línea de vista (*line of sight*, LOS) y en sistemas de gran apertura, reflejados o difusos (*diffused*).

- **Los sistemas infrarrojos de corta apertura:** están constituidos por un cono de haz infrarrojo altamente direccional y funcionan de manera

similar a los controles remotos de las televisiones: el emisor debe orientarse hacia el receptor antes de empezar a transferir información, limitando por tanto su funcionalidad. Resulta muy complicado utilizar esta tecnología en dispositivos móviles, pues el emisor debe reorientarse constantemente. Este mecanismo solo es operativo en enlaces punto a punto exclusivamente. Por ello se considera que es un sistema inalámbrico pero no móvil, o sea que está más orientado a la portabilidad que a la movilidad.

- **Los sistemas de gran apertura:** permiten la información en ángulo mucho más amplio por lo que el transmisor no tiene que estar alineado con el receptor. Una topología muy común para redes locales inalámbricas basadas en esta tecnología, consiste en colocar en el techo de la oficina un nodo central llamado punto de acceso, hacia el cual dirigen los dispositivos inalámbricos su información, y desde el cual ésta es difundida hacia esos mismos dispositivos.

  La dispersión utilizada en los sistemas de gran apertura, hace que la señal transmitida rebote en techos y paredes, introduciendo un efecto de interferencia en el receptor, que limita la velocidad de transmisión (la trayectoria reflejada llega con un retraso al receptor). Esta es una de las dificultades que han retrasado el desarrollo del sistema infrarrojo en la norma 802.11.

ANEXO I

SCRIPTS

SCRIPT efectiva.pl

```
#PAGINA  35
# type: perl Throughput.pl <trace flie> <requerid node> <granularity> file

$infile=$ARGV[0];
$tonode=$ARGV[1];
$granularity=$ARGV[2];


#calculamos  cuantos bytes fueron transmitidos durante el intervalo de tiempo especificado
#Por el parametro granularity en segundos

$sum=0;
$clock=0;

        open (DATA,"<$infile")
          || die "Can't open $infile $!";

        while (<DATA>) {
                @x= split(' ');


#if ($x[1] >= 4.0)
#{


#columna 1 es el tiempo

if ($x[1]-$clock <= $granularity)
{

#chequeo si los eventos corresponden a recividos

if ($x[0] eq 'r')
{

#OJO AQUI
#chequeo si el destino corresponde al primer argumento
if ($x[2] eq $tonode)
{

#chequeo si el paquete es TCP
if ($x[6] eq 'tcp')
{

        $sum=$sum+$x[7];

}
```

```
}
}
}

else
{       $throughput=8.0*$sum/$granularity;


#       $dis=$x[1]-2.0;


        if ($x[1] >= 4.0)
{

       $dis=$x[1]-2.0;
        print STDOUT "$dis $throughput\n";
        $clock=$clock+$granularity;
        $sum=0;
}
}

}

        $throughput=8.0*$sum/$granularity;

#       $dis=$x[1]-2.0;

        print STDOUT "$x[1] $throughput\n";
        $clock=$clock+$granularity;
        $sum=0;


        close DATA;

#}

exit(0);
```

SCRIPT wi-fi.tcl

```
#PAGINA  35
# type: perl Throughput.pl <trace flie> <requerid node> <granularity> file

$infile=$ARGV[0];
$tonode=$ARGV[1];
$granularity=$ARGV[2];


#calculamos  cuantos bytes fueron transmitidos durante el intervalo de tiempo especificado
#Por el parametro granularity en segundos

$sum=0;
$clock=0;

      open (DATA,"<$infile")
        || die "Can't open $infile $!";

      while (<DATA>) {
            @x= split(' ');

#columna 1 es el tiempo

if ($x[1]-$clock <= $granularity)
{

#chequeo si los eventos corresponden a recividos

if ($x[0] eq 'r')
{

#OJO AQUI
#chequeo si el destino corresponde al primer argumento
if ($x[2] eq $tonode)
{
#chequeo si el paquete es TCP
if ($x[6] eq 'tcp')
{

      $sum=$sum+$x[7];

}
}
}
```

```
}

else
{       $throughput=8.0*$sum/$granularity;


        print STDOUT "$x[1] $throughput\n";
        $clock=$clock+$granularity;
        $sum=0;
}

}

        $throughput=8.0*$sum/$granularity;

        print STDOUT "$x[1] $throughput\n";
        $clock=$clock+$granularity;
        $sum=0;


        close DATA;

exit(0);
```