



ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO

FACULTAD DE INFORMÁTICA Y ELECTRÓNICA

ESCUELA DE INGENIERÍA ELECTRÓNICA

**“MONITORIZACIÓN Y CONTROL DE UN ROBOT RECOLECTOR DE
DATOS MEDIANTE EL PUERTO USB Y COMUNICACIÓN INALÁMBRICA”**

TESIS DE GRADO

PREVIO A LA OBTENCIÓN DE TÍTULO DE:

INGENIERO EN ELECTRÓNICA Y COMPUTACIÓN

PRESENTADO POR:

JOSÉ AMABLE ESCOBAR LOPEZ

RIOBAMBA – ECUADOR

2010

Agradezco a Dios del Cielo por medio
del niño Jesús por todas las bendiciones recibidas
al Ing. Franklin Moreno Director de Tesis por su valiosa
colaboración para la realización de este trabajo.

A MIS PADRES

A MIS HERMANOS

Por su gran apoyo de todos los días que han sabido brindarme, en especial a Mi madre por su gran sacrificio e innegable apoyo.

NOMBRE	<u>FIRMA</u>	FECHA
Dr. Ms.c Romeo Rodríguez DECANO FACULTAD DE INFORMÁTICA Y ELECTRÓNICA
Ing. Paúl Romero DIRECTOR DE LA ESCUELA DE INGENIERÍA ELECTRÓNICA
Ing. Franklin Moreno DIRECTOR DE TESIS
Ing. Hugo Moreno MIEMBRO DEL TRIBUNAL
Tlgo. Carlos Rodríguez DIRECTOR DPTO. DOCUMENTACION
NOTA DE LA TESIS	

“Yo, José Escobar soy responsable de las ideas, doctrinas y resultados expuestos en esta tesis; y el patrimonio intelectual de la Tesis de Grado pertenece a la ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO”.

José Escobar

INDICE GENERAL

CAPÍTULO I

1. TECNOLOGÍA USB	12
1.1. USB Normalización	14
1.2. FUNCIONES USB	15
1.3. Host USB - Hardware y software:	16
1.4. Características principales	17
1.4.1. Plug and Play	17
1.4.2. Hot plugginn (conectar en caliente).....	18
1.4.3. Recursos del dispositivo	18
1.4.4. Simplicidad	18
1.4.5. Dispositivos	19
1.4.6. Velocidad	20
1.4.7. Plataforma.....	20
1.5. Los Hub's USB	21
1.6. NIVEL FISICO.....	23
1.7. Conectores USB de Seria "A" y serie "B"	24
1.8. Interfase física (eléctrica).....	26
1.9. Protocolo del bus	28
1.10.1. Transmisión asincrónica	29
1.10.2. Transmisión sincrónica:.....	31
1.10.3. Transmisiones isocrónicas	31
1.10.4. Transmisión bulk:	32
1.10.5. Transmisiones de control:.....	32
1.10.6. Transmisiones de interrupción:.....	32
1.11. CLASES USB.....	33
1.12. USB 1.1.....	33
1.13. USB 2.0.....	34
1.14. USB 1.0 y USB 2.0.....	34
1.15. USB sin cable.....	35

CAPÍTULO II

2. Modulo de Radio Frecuencia Xbee	36
2.1. Introducción	36
2.2. El Coordinador.....	38
2.3. Los Routers.....	38
2.4. End Device.....	38
2.5. Conexión.....	39
2.6. Circuito básico para el Xbee.....	41
2.7. Modos de Operación.....	42
2.7.1. Modo RECIBIR / TRANSMITIR.....	43
2.7.2. Modo de Bajo Consumo (Sleep Mode)	44
2.7.3. Modo de Comando.....	45
2.7.4. Modo Transparente	50
2.8. Modo de operación API.....	52
2.9. IDLE	53
2.10. Configuración del Módulo Xbee.	53

2.11.	Direccionamiento de los módulos.....	53
2.12.	Direccionamiento de 16 bit.....	54
2.13.	Direccionamiento de 64 bits.	54
2.14.	Modo de Conexión Transparente.....	55
2.15.	Punto a Punto.	56
2.16.	Punto a Multipunto.	58
2.17.	Broadcast	61
2.18.	Cable Virtual I/O	63
2.19.	Conexión NonBeacon c/Coordinador.....	67
2.20.	Comandos más importantes.....	68
2.21.	Detalles Físicos.....	75

CAPÍTULO III

	Dispositivos utilizados en el proyecto	76
3.	PIC18F2550	76
3.1.	Estado y control del módulo USB:	79
3.2.	Registro de configuración del USB:	82
3.3.	Características principales del pic 18f2550	86
3.4.	Universal Serial Bus	87
3.5.	Sensor de Temperatura LM 35	90
3.6.	Características principales del LM35	90
3.7.	Fotorresistencia	93
3.8.	Características de una Fotorresistencia.....	93
3.9	Sensor de Humedad	93
3.9.1	Características Eléctricas	94

CAPÍTULO IV

4.	Diseño E Implementación del Prototipo.....	966
4.1.	Implementación del Hardware.....	988
4.2.	Implementación del Software	1033
4.3.	Plataforma móvil.....	104
4.4.	PRUEBAS Y RESULTADOS.....	105

CONCLUSIONES

RECOMENDACIONES

BILIOGRAFIA

BIBLIOGRAFÍA DE INTERNET

SUMARY

ANEXO

INDICE DE FIGURAS

Figura I.1	Puerto USB	14
Figura I.2	Periféricos o dispositivos USB	15
Figura I.3	Esquema de tres puertos de la PC	16
Figura I.4	Vista de un Hub USB	22
Figura I.5	Puertos de un Hub USB	23
Figura I.6	Cable USB con conectores tipo A-A, A-B y , B-B respectivamente	25
Figura I.7	Ranuras serie A y serie B	25
Figura I.8	Ranuras serie A y serie B conectores serie A y serie B	26
Figura I.9	Simbología para identificación de puerto USB	26
Figura I.10	Pines del puerto USB	26
Figura I.11	Conexión interna del cable USB	27
Figura I.12	Conexión del PC al hub	28
Figura I.13	Esquema de datos de la Transmisión asincrónica	30
Figura I.14	Esquema de datos de la Transmisión sincrónica	31
Figura II.1	Conexión típica usando Xbee	37
Figura II.2	Coordinador PAN con múltiples nodos	39
Figura II.3	Elementos del Xbee	40
Figura II.4	Red Mesh para módulos Xbee PRO Serie 2	41
Figura II.5	Conexiones mínimas requeridas para el Xbee	42
Figura II.6	Modos de operación del modulo Xbee.	43
Figura II.7	Ejemplo Comando AT	46
Figura II.8	Ejemplo lectura de parámetros	47
Figura II.9	Ejemplo del comando ATMY	48
Figura II.10	Forma normal de escribir	49
Figura II.11	Forma abreviada para varios comandos al mismo tiempo	50
Figura II.12	Correspondencia de Buffer	51
Figura II.13	Configuración del modulo 1	56
Figura II.14	Configuración del modulo 2	57
Figura II.15	Ejemplo direccionamiento 16 y 64 bit	58
Figura II.16	Canales disponibles para el protocolo IEEE 802.15.4	59
Figura II.17	Configuración punto-a-multipunto	61
Figura II.18	Red de Broadcast	62
Figura II.19	Diagrama de pines del módulo Xbee. Vista Superior	64
Figura II.20	Formato del Frame para Cable Virtual	67
Figura II.21	Diagrama de Pines del modulo Xbee	72
Figura II.22	Detalles eléctricos del modulo del Xbee	73
Figura II.23	Características de operación ADC del modulo del Xbee	73
Figura II.24	Detalles Técnicos del modulo Xbee	73
Figura II.25	Vista Superior y lateral	75
Figura III.1	Pines del Pic18f2550	76
Figura III.2	Nivel de conexionado del USB con el exterior	78
Figura III.3	Puente bidireccional	79
Figura III.4	Registro UCON	80
Figura III.5	Registro UCFG	83

Figura III.6	Registro UPUEN	85
Figura III.7	Diagrama interno del PIC 18F2550	88
Figura III.8	Mapa de la memoria del PIC 18F2550	89
Figura III.9	Sensor de Temperatura	90
Figura III.10	Pines del sensor de Temperatura	92
Figura III.11	Fotorresistencia	93
Figura III.12	Sensor de humedad HU-10S	93
Figura IV.1	Diagrama de Bloques Del Circuito	95
Figura IV.2	Diagrama de Flujo del proyecto	97
Figura IV.3	Diseño que permite la comunicación entre PC y circuito electrónico	98
Figura IV.4	Circuito que permite la activación de los motores y envío de datos de sensores	99
Figura IV.5	Circuito de sensor de luz	100
Figura IV.6	Circuito de sensor de Temperatura	100
Figura IV.7	Diseño 2 con todas las etapas	101
Figura IV.8	Programa final realizado en Visual Basic 6.0	103
Figura IV.9	Plataforma móvil para montar el circuito 2	103
Figura IV.10	Circuito2 armado en el proyect-board	104
Figura IV.11	Circuito1 armado en baquelita	104
Figura IV.12	Identificación del hardware cuando es conectado al PC	105
Figura IV.13	Identificación del interfaz humana	106
Figura IV.14	Controladores del interfaz humana instalados automáticamente	106
Figura IV.15	Activando los movimientos del robot	107
Figura IV.16	Figura recibiendo los datos de los sensores luz, temperatura, humedad	107

INDICE DE TABLAS

Tabla II.1	Tabla Modo Sleep y consumos de corriente	45
Tabla II.2	Frecuencia de Canales y su respectivo Comando AT	60
Tabla II.3	Comandos para Cable Virtual.	65
Tabla II.4	Configuración para Cable Virtual	66
Tabla II.5	Resumen de Comandos	72
Tabla II.6	6 Especificaciones Técnicas	74
Tabla III.1	Bit de registro UCON	80
Tabla III.2	líneas D+ y D-	84
Tabla III.3	Configuraciones registro UPUEN	85

INTRODUCCIÓN

La comunicación inalámbrica ha tenido una gran aceptación en la comunicación entre diferentes dispositivos como por ejemplo Redes Inalámbricas, Sistemas Domóticos, Bluetooth, Infrarrojos, Radio Frecuencia, etc.

Su aceptación tuvo éxito por su fácil manipulación, es decir poder movilizar sin los molestos cables dentro de su distancia permitida.

Con el avance de la tecnología las PC actuales han ido reemplazando los puertos Paralelo y Serie por el puerto USB por sus ventajas que ofrece con relación a los otros puertos. Empresas fabricantes de circuitos electrónicos han adoptado este puerto USB para la comunicación entre los distintos sistemas electrónicos que se crean o que ya están en el mercado. Como por ejemplo teclados, ratones, cámaras, joysticks, Módems, escáneres, impresoras, discos duros externos, altavoces, etc; poseen conexión USB debido a la velocidad y simplicidad que dispone este puerto.

Con la comunicación inalámbrica y con el puerto USB y algunos dispositivos que se pudo crear el proyecto "MONITORIZACIÓN Y CONTROL DE UN ROBOT RECOLECTOR DE DATOS MEDIANTE EL PUERTO USB Y COMUNICACIÓN INALÁMBRICA ".El proyecto abarca lo que es la comunicación inalámbrica en la frecuencia de banda libre de 2.4Ghz con un alcance de 300 metros en línea de vista y de 30 a 50 metros sin línea de vista. El circuito transmite datos de temperatura, de luz, y de humedad que son recolectados por un robot móvil que igualmente su movimiento es controlado inalámbricamente.

Una vez obtenidos los datos estos llegan a la PC mediante el puerto USB y con la ayuda del programa de Visual Basic podemos interpretar los datos gráficamente para la visualización del usuario.

CAPÍTULO I

1. TECNOLOGÍA USB

Originalmente publicado en 1995, los puertos USB comenzaron a aparecer en los nuevos equipos fabricados en 1997. Windows 98 fue la primera plataforma de Windows de apoyo a la tecnología USB. No pasó mucho tiempo antes de la popularidad de USB creció hasta el punto de que todas las anteriores normas de la conectividad se sustituyeron - se ha ido el anticuado ahora los puertos serie y paralelo la mayoría de los equipos utilizados hasta ese momento. Hoy en día las computadoras se suelen ofrecer un mínimo de cuatro puertos USB como característica estándar.

Un solo puerto USB se pueden conectar más de un centenar de dispositivos intercambiables, como módem, teclado, mouse, joysticks, escáneres, impresoras, cámaras digitales, y de almacenamiento externo. USB también soporta Plug-and-Play de instalación y permite a los dispositivos que se conecta en caliente. USB se ha convertido en el estándar para la conexión de varios dispositivos tales como cámaras digitales y escáneres. USB también goza de un amplio reconocimiento de marca del cliente y una reputación de alto rendimiento y facilidad de uso.

USB es una especificación de las empresas Compaq, Intel, Microsoft y NEC, que describe un canal serie que soporta una gran variedad de periféricos de media y baja velocidad, con soporte integral para transferencias en tiempo real (isócronas) como voz, audio y vídeo comprimido, y que permite mezclar dispositivos y aplicaciones isócronas y asíncronas. Por lo tanto, entre los dispositivos USB más característicos se pueden citar teclados, ratones, joysticks, tabletas gráficas, monitores, módems, impresoras, escáneres, CD-ROMs, dispositivos de audio (como micrófonos o altavoces digitales), cámaras digitales y otros dispositivos multimedia. La versión 1.1 (Sep/98) establece:

- Un acceso al bus gestionado directamente por el Controlador USB, para permitir transferencias isócronas y eliminar los tiempos de arbitración.
- Una velocidad de 12 Mbps (Full Speed o FS) y un subcanal de 1,5 Mbps (Low Speed o LS) para los dispositivos más lentos, como ratones y joysticks. La coexistencia en un mismo sistema de dispositivos FS y LS se maneja mediante conmutación automática y dinámica de velocidad entre unas transferencias y otras.
- Una conectividad excepcional, ya que puede manejar hasta 127 dispositivos simultáneamente que se pueden conectar y desconectar en caliente, sin tener que reiniciar el sistema.
- Una configuración automática de dispositivos, que elimina la necesidad de realizar configuraciones manuales por medio de puentes o conmutadores.
- La coexistencia de dispositivos isócronos y asíncronos. Los dispositivos isócronos se atienden en función del ancho de banda y latencia requeridos, y los asíncronos se atienden durante el tiempo restante no consumido por los dispositivos isócronos.

- Una distribución de alimentación desde el Controlador USB, que permite la conexión tanto de dispositivos alimentados desde el bus como autoalimentados.
- Una arquitectura fácilmente escalable para permitir la existencia de varios Controladores USB en un sistema. USB se soporta en Win95 2.1 (4.00.950B) y superior, Win98, Win2000 y MacOS 8.5 y superior.

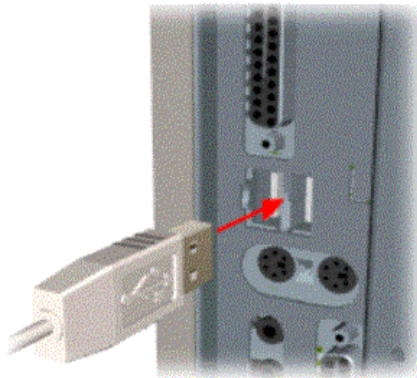


Figura I.1 Puerto USB

1.1. USB Normalización

USB se ha convertido en el estándar de la industria, principalmente porque todos los grandes fabricantes acordadas en el convenio colectivo para utilizar el diseño y la función de esta tecnología de conexión rápida. Algunos de los notables que las empresas se comprometieron a aplicar la tecnología USB incluyen Intel, Apple, Microsoft, Hewlett-Packard, NEC, y Agere. El actual USB versión 2.0 fue liberado en la primavera de 2000 y ha proporcionado importantes ventajas a través del uso de su característica de compatibilidad hacia atrás. Esto permite que las versiones anteriores de USB para ajustarse a las nuevas aplicaciones.

1.2. FUNCIONES USB

Dentro de la terminología USB, todos los dispositivos que pueden ser conectados a este bus, a excepción de los Hubs, se denominan Funciones. Son funciones típicas: el ratón, el monitor, altoparlantes, MODEM, etc.

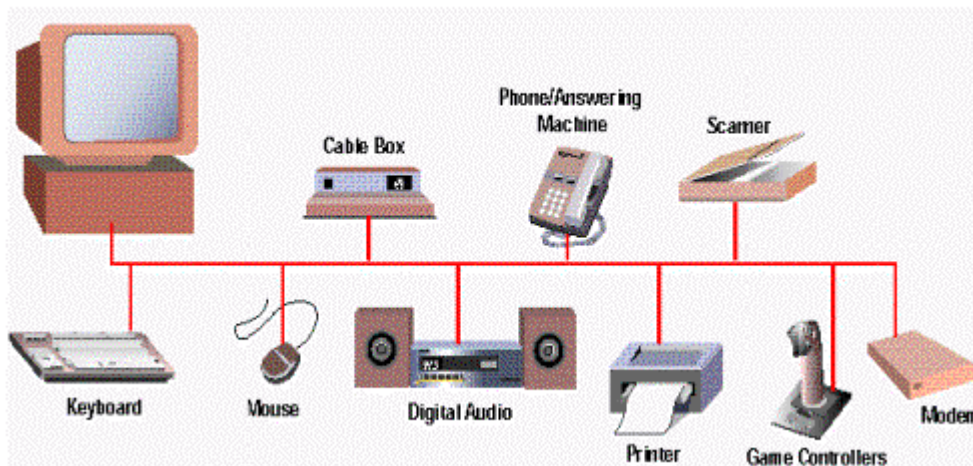


Figura I 2. Periféricos o dispositivos USB

Las funciones o dispositivos periféricos, son capaces de recibir y transmitir información, ya sea del usuario o de control. El común denominador de todas las funciones USB es su cable y el conector del mismo, diseñado y fabricado de acuerdo a las especificaciones del bus, por lo que no cabe preocuparse por la compatibilidad entre equipos de diferentes fabricantes; solamente hay que recordar las empresas que respaldan esta tecnología.

Un aspecto interesante de las funciones, es que pueden ser a la vez nuevos hubs. La siguiente figura muestra un esquema en el que la PC tiene tres puertos, la siguiente función que puede ser un monitor 4, el siguiente 3 y adicionalmente un Hub, provee 4

puertos mas, es un esquema tan sencillo, donde existen 14 puertos disponibles para todo tipo de periférico, entre los que podemos citar: ratón, tablilla digitalizadora, lápiz óptico, teclado impresora, un teléfono ISDN, etc.

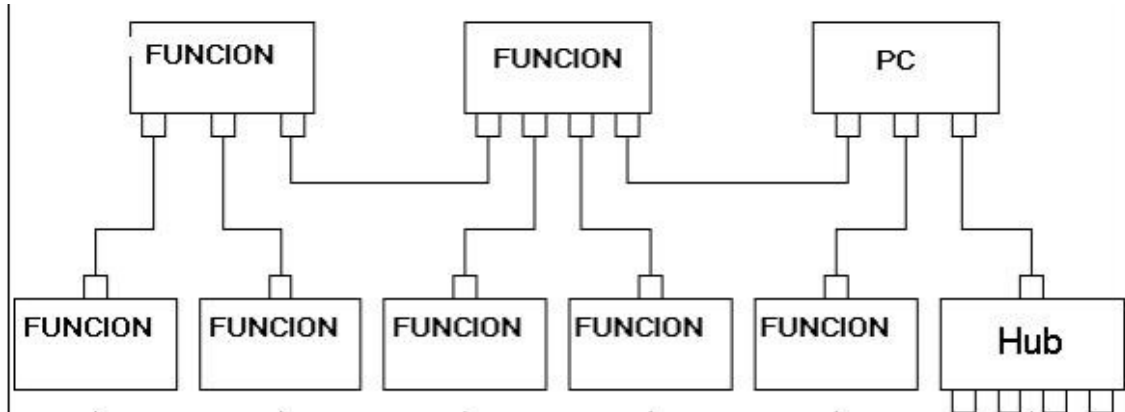


Figura I 3. Esquema de tres puertos de la PC

1.3. Host USB - Hardware y software

El computador o Host USB trabaja con los diferentes dispositivos valiéndose del controlador de host compuesto por una parte de hardware y otra de software, de esta forma conjunta, el host es responsable al nivel de hardware, de los siguientes aspectos dentro del sistema USB:

- Detectar tanto la conexión de nuevos dispositivos USB al sistema como la remoción de aquellos ya conectados, y por supuesto, configurarlos y ponerlos a disposición del usuario, tarea que involucra acciones por software.
- Administrar y controlar el flujo de datos entre el host y los dispositivos USB, es decir el movimiento de información generada por el mismo usuario.

- Administrar y regular los flujos de control entre el host y los dispositivos USB, es decir la información que se mueve con el objeto de mantener el orden dentro de los elementos del sistema.
- Recolectar y resumir estadísticas de actividad y estado de los elementos del sistema.
- Proveer de una cantidad limitada de energía eléctrica para aquellos dispositivos que pueden abastecerse con tan solo la energía proveniente del computador (teclado, ratón son dos ejemplos claros).

Por otra parte, a nivel de software las funciones del controlador de Host se incrementan y complican:

- Enumeración y configuración de los dispositivos conectados al sistema
- Administración y control de transferencias isocrónicas de información
- Administración y control de transferencias asincrónicas
- Administración avanzada de suministro eléctrico a los diferentes dispositivos
- Administración de la información del bus y los dispositivos USB

1.4. Características principales

1.4.1. Plug and Play

El puerto USB, y por tanto todos los dispositivos con conexión USB, son verdaderamente Plug and play, es decir, el dispositivo es detectado automáticamente al conectarlo al equipo y el sistema operativo instala el driver adecuado o nos pide el disco de instalación. Aún más, no es necesario apagar, ni siquiera re iniciar, el equipo para

conectar o desconectar los dispositivos, cargándose y descargándose automáticamente de memoria el driver correspondiente.

1.4.2. Hot plugginn (conectar en caliente)

No es necesario apagar, ni siquiera reinicial, el equipo para instalar o desinstalar un dispositivo USB. Algo especialmente interesante si olvidamos conectar un dispositivo al encender el equipo o si simplemente no tenemos suficientes conectores para todos los dispositivos USB. Además, esto permite conservar recursos de memoria, pues cada dispositivo conectado requiere un driver residente, que de este modo sólo se carga cuando se necesita y se descarga al dejar de necesitarlo.

1.4.3. Recursos del dispositivo

Quizá una de las mayores ventajas para los equipos actuales es el hecho de que el puerto USB solamente necesita una IRQ y una dirección de memoria y todos los dispositivos conectados a él, solamente necesitan una ID para su identificación (como en el bus SCSI) dentro de la cadena de 127 dispositivos, sin necesitar más recursos. Si tenemos en cuenta que los puertos estándar (dos serie, un paralelo, un PS/2 para ratón, uno para teclado, un puerto para joystick) consumen 5 IRQs, algunas DMA y múltiples direcciones de memoria, al utilizar dispositivos USB nos estamos ahorrando valiosos recursos del sistema.

1.4.4. Simplicidad

El manejo de los dispositivos USB se hace por software, concretamente por el propio sistema operativo, por lo que los dispositivos USB son más fáciles de fabricar y por

tanto más baratos. Además, USB es una tecnología abierta por la que no hay que pagar derechos, lo que siempre abarata los costos de fabricación.

1.4.5. Dispositivos

Se puede conectar hasta un total de 127 dispositivos en cadena o utilizando HUBs USB (concentradores de puertos USB), y cada dispositivo puede tener un cable de hasta 5 metros de longitud, frente a 1 metro para el puerto serie y 4 metros para el puerto paralelo. Además, conectándolos en cadena, el último dispositivo puede estar a 635 metros del ordenador.

Actualmente se encuentran en el mercado monitores, teclados, ratones, cámaras, joysticks, módem, escáneres, impresoras e incluso altavoces (sin necesidad de tarjeta de sonido) con conexión USB, dispositivos de almacenamiento unidad ZIP, unidad LS-120, CD-ROMs, discos duros externos, etc.

Lo que se debe olvidar a la hora de comprar un dispositivo USB es que cada dispositivo puede funcionar como HUB, es decir, incluir uno o más conectores USB, de modo que podamos conectar un dispositivo a otro en cadena, y así, por ejemplo un teclado, puede incluir dos conectores USB, uno para el ratón y otro para el joystick, de igual modo el monitor puede servir de HUB y permitir conectar a él por ejemplo los altavoces, o el teclado, al cual a su vez se conectan el ratón y el joystick, etc. Hay que tener en cuenta que muchos dispositivos USB actuales no son más que conversiones de dispositivos existentes por lo que mucho aún no implementan su uso como HUBs, por lo que quizás valga la pena esperar un poco a que haya más dispositivos disponibles.

1.4.6. Velocidad

El puerto serie es capaz de transmitir hasta 112,5 KB/s y el puerto paralelo entre 600KB/s y 15MB/s, pero el puerto USB es capaz de llegar a alcanzar entre 1,5MB/s y 12MB/s, por lo que es la conexión ideal para módem de 56K, escáneres (como alternativa de similar costo a los de puerto paralelo), CD-ROMs externos, dispositivos de copia de seguridad externos, etc. Sin embargo, para dispositivos de almacenamiento (especialmente discos duros externos), conexiones de red, cámaras, etc. quizás sea interesante también mostrar, la nueva versión del estándar de conexión, FIREWIRE (IEEE 1394), que maneja transferencias entre 100MB/s y 400MB/s, que permite conectar hasta 63 dispositivos y un cable de 4.5 metros por dispositivo, permitiendo al igual que el USB la conexión en caliente. El puerto USB no está destinado a desaparecer (de hecho no hay en el mercado placas con este conector pues los chipsets actuales no lo soportan), sin que se destinará a cierto tipo de dispositivos que requieren una transferencia de datos baja o media (teclados, ratones, joysticks, etc.) mientras que el bus FIREWIRE se destinará a dispositivos que requieren una alta transferencia (escáneres, impresoras, disco duro, DVD, etc.). Sin embargo, no hay que olvidar, que USB está modificando sus características, para poder absorber velocidades hasta los 480 Mbps, lo cual dará una dura pelea al estándar IEEE 1394 (del cual se hablará más adelante)

1.4.7. Plataforma

Para poder utilizar dispositivos USB, hay recordar que el sistema operativo instalado en nuestro equipo debe soportar este nuevo bus. Windows 95 en sus versiones OSR2.1 y OSR2.5 detecta el puerto USB y soporta dispositivos USB (la versión OSR2.0 también

añadiendo el SUPLEMENTO USB), pero es realmente con el sistema operativo Windows 98 que los ya abundantes dispositivos USB no han dado problemas de instalación y funcionamiento, no dejando de lado las ultimas versiones de Windows NT.

Si queremos que los dispositivos USB funcionen, además en la BIOS de nuestra tarjeta madre debemos habilitar la opción ASSIGN USB IRQ - ENABLED. Si queremos utilizar un teclado USB con nuestro equipo, debemos habilitar USB KEYBOARD SUPPORT VIA BIOS en vez de VIA OS si queremos utilizar el teclado cuando "Reiniciamos equipo en modo MS-DOS" para ejecutar programas de diagnóstico. Si no lo hacemos así, por ejemplo, no accederemos a la BIOS del sistema al pulsar SUPR/DEL, ni anularemos el test de memoria al pulsar ESC, ni accederemos al "Menú de inicio" al pulsar F8, aunque el teclado funcionará correctamente en Windows 9X.

Uno de los problemas del puerto USB es que suministra solamente 500 miliamperios de electricidad para los dispositivos conectados, que aunque es suficiente potencia para la mayoría de los dispositivos que se conectan a este puerto, resulta escaso cuando conectamos varios dispositivos sin fuente de alimentación propia. Lo que sí podemos hacer es comprar un HUB USB con toma de alimentación eléctrica, para proporcionar la potencia necesaria a aquellos dispositivos que lo requieran (especialmente escáneres e impresoras).

1.5. Los Hub's USB

Los Hubs son elementos claves dentro de la arquitectura Conectar y utilizar de USB. Adicionalmente, simplifican de gran manera la sencillez de la interconexión de

dispositivos al computador. Las siguientes Figuras muestran hubs USB disponibles en el mercado.

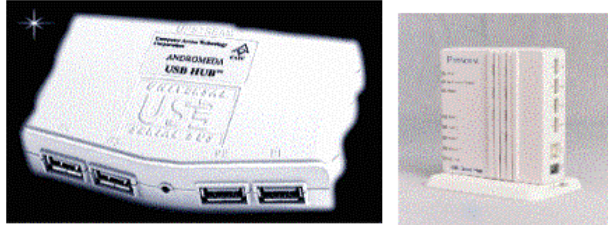


Figura I 4. Vista de un Hub USB

Bajo una óptica eléctrica e informática, los hubs son concentradores cableados que permiten múltiples conexiones simultáneas. Su aspecto más interesante es la concatenación, función por la que a un hub se puede conectar otro y otro, ampliando la cantidad de puertos disponibles para periféricos

El hub USB tiene la capacidad de detectar si un periférico ha sido conectado a uno de sus puertos, notificando de inmediato al Controlador de Host en el computador, proceso que desata la configuración del equipo nuevo; adicionalmente, los hubs también son capaces de detectar la desconexión de un dispositivo, notificando al Controlador de Host que debe remover las estructuras de datos y programas de administración (drivers) del dispositivo retirado.

Otra de las funciones importantes de los hubs es la de aislar a los puertos de baja velocidad de las transferencias a alta velocidad, proceso sin el cual todos los dispositivos de baja velocidad conectados al bus entrarían en colapso. La protección de los dispositivos lentos de los rápidos ha sido siempre un problema serio dentro de las redes mixtas, como es USB.

El hub está compuesto por dos partes importantes: El Controlador del Hub y el Repetidor del Hub. El Repetidor del Hub tiene la función de analizar, corregir y retransmitir la información que llega al hub, hacia los puertos del mismo. Mantiene una memoria consistente en varios registros de interfaz que le permiten sostener diálogos con el host y llevar adelante algunas funciones administrativas además de las meramente operativas; mientras que el Controlador de Hub puede asemejarse a una pequeña CPU de supervisión de las múltiples funciones que deben desempeñar un hub.

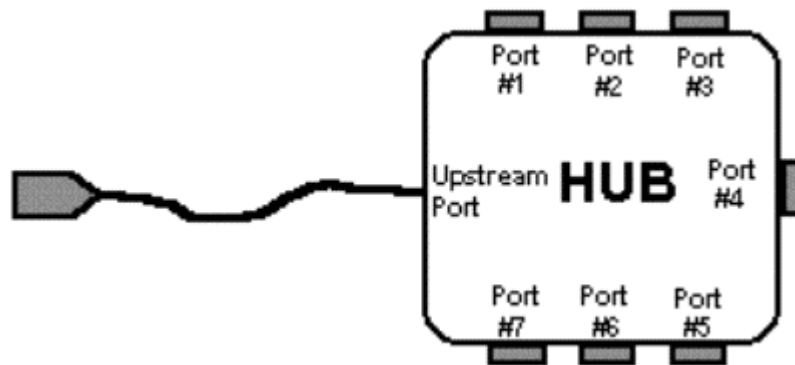


Figura I 5. Puertos de un Hub USB

1.6. NIVEL FISICO

A nivel físico, USB utiliza un cable de 4 conductores para transmitir una señal diferencial (D+ y D-) y alimentación (VBus = 5V y GND) por medio de conexiones punto a punto. Los dispositivos LS van obligatoriamente equipados con un cable de longitud adecuada (hasta unos 3 m, dependiendo de sus características eléctricas), mientras que los FS pueden ir equipados con un cable o utilizar cables independientes de hasta 5 m (también dependiendo de sus características eléctricas).

La comunicación es bidireccional y semi-dúplex, y utiliza codificación auto reloj NRZI (la línea cambia de nivel si se transmite un 0 y no cambia si transmite un 1) con "bit

stuffing" (inserción de un cero tras la transmisión de 6 unos, para asegurar transiciones en la línea y permitir que la PLL del receptor se mantenga sincronizada). Los dispositivos disponen de un transmisor diferencial, receptores diferencial y S/E y resistencias de terminación con los que pueden transmitir y detectar varios estados eléctricos distintos en la línea:

- Transmisión/Recepción diferencial de bits: Estados DIFF0 y DIFF1, denominados también estados J y K.
- SE0 (Single-Ended 0): Ambas señales D+ y D- a 0V. Se utiliza para detectar la conexión/desconexión de dispositivos, para indicar el EOP (fin de paquete) y para generar reset.
- IDLE: reposo o línea en alta impedancia, necesario para permitir transferencias semi-dúplex, detectar la conexión y desconexión de dispositivos y discriminar entre dispositivos FS y LS.
- El SOP (principio de paquete) se indica mediante una transición IDLE a K.
- El EOP (fin de paquete) se indica mediante una secuencia SE0 (2 bits) + J (1 bit) + IDLE. Detección de dispositivo y discriminación FS/LS: cuando el transmisor deja la línea en IDLE, si hay un dispositivo conectado su polarización fuerza un estado J (DIFF0 si LS ó DIFF1 si FS), y si no lo hay, la polarización del transmisor fuerza un estado SE0.
- Reset: transmisión de SE0 durante ≥ 10 ms.

1.7. Conectores USB de Seria "A" y serie "B"

Existen dos tipos de conectores dentro del Bus Serial Universal. El conector Serie A está pensado para todos los dispositivos USB que trabajen sobre plataformas de PCs.

Serán bastante comunes dentro de los dispositivos listos para ser empleados con host PCs, y lo más probables es que tengan sus propios cables con su conector serie A. Sin embargo, esto no se dará en todos los casos, existirán dispositivos USB que no posean cable incorporado, para los cuales el conector Serie B será una característica. Sin embargo este no es un problema, ya que ambos conectores son estructuralmente diferentes e insertarlos de forma equívoca será imposible por la forma de las ranuras. La primera figura muestra los diferentes tipos de conectores USB, y la Figura siguiente las respectivas ranuras.

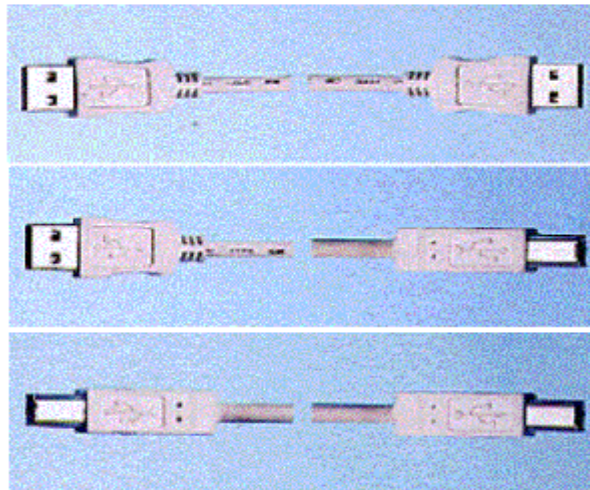


Figura I 6 Cable USB con conectores tipo A-A, A-B y, B-B respectivamente

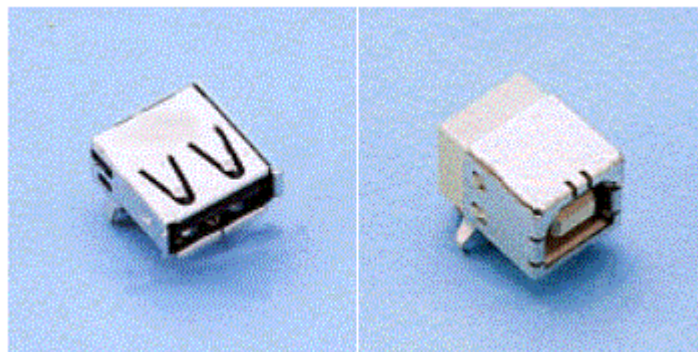


Figura I 7 Ranuras serie A y serie B



Figura I 8 Ranuras serie A y serie B conectores serie A y serie B

1.8. Interfase física (eléctrica)

El interfaz de Bus Universal en Serie (USB) se identifica con este icono que se encuentra en la parte posterior de la computadora:



Figura I 9 Simbología para identificación de puerto USB

Los pines del conector se identifican a continuación.

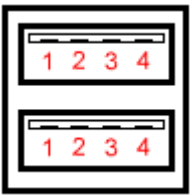
Conector	Pin	Señal
	1	+5V
	2	Datos -
	3	Datos +
	4	A Tierra

Figura I 10 Pines del puerto USB

El Bus Serial Universal transfiere señales de información y energía eléctrica a través de 4 cables, cuya disposición se muestra en las siguientes figuras.

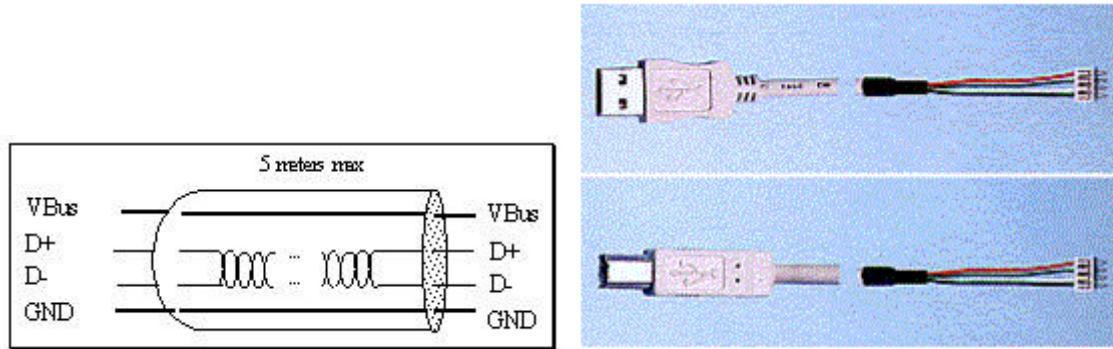


Figura I 11 Conexión interna del cable USB

Por su parte las señales se mueven sobre dos cables y entre segmentos comprendidos entre un par de dispositivos USB, con rangos de velocidad de 12Mbps o 1.5Mbps, para transmisiones de alta y baja velocidad respectivamente. Ambos modos de transmisión son controlados automáticamente por medio de los dispositivos USB de manera transparente al usuario. Es importante notar que siempre ha sido un serio problema manejar velocidades diferentes de transmisión de datos por un mismo cable, y esto no sería posible sin que todos los dispositivos estén preparados para tal efecto.

Los pulsos de reloj o sincronismo son transmitidos en la misma señal de forma codificada bajo el esquema NRZI (Non Return To Zero Invert), uno de los más interesantes sistemas de codificación de información que no vale la pena mencionarlo ahora por su complejidad eléctrica-electrónica.

Los otros dos cables VBus y GND tienen la misión de llevar suministro eléctrico a los dispositivos, con una potencia de +5V para VBus. Los cables USB permiten una

distancia que va de los pocos centímetros a varios metros, más específicamente 5 metros de distancia máxima entre un dispositivo USB y el siguiente. La Figura siguiente muestra el detalle.

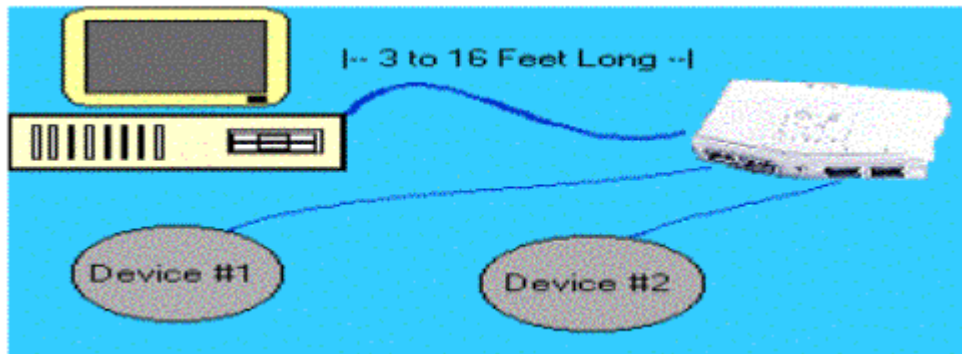


Figura I 12 Conexión del PC al hub

Es importante indicar que los cables USB tienen protectores de voltaje a fin de evitar cualquier daño a los equipos, son estos mismos protectores los que permiten detectar un dispositivo nuevo conectado al sistema y su velocidad de trabajo.

1.9. Protocolo del bus

Toda transferencia de datos o transacción que emplee el bus, involucra al menos tres paquetes de datos. Cada transacción se da cuando el Controlador de Host decide qué dispositivo hará uso del bus, para ello envía un paquete al dispositivo específico. Cada uno de los mismos tiene un número de identificación, otorgado por Controlador de Host cuando el computador arranca o bien cuando un dispositivo nuevo es conectado al sistema. De esta forma, cada uno de los periféricos puede determinar si un paquete de datos es o no para sí. Técnicamente este paquete de datos se denomina Paquete Ficha o Token Packet. Una vez que el periférico afectado recibe el permiso de transmitir,

arranca la comunicación y sus tareas específicas; el mismo informará al host con otro paquete que ya no tiene más datos que enviar y el proceso continuará con el siguiente dispositivo.

Este protocolo tiene un sistema muy eficiente de recuperación de errores, empleando uno de los modelos más seguros como es el CRC (Código de Redundancia Cíclica). Y puede estar implementado al nivel de software y/o hardware de manera configurable. De hecho si el control es al nivel de hardware, no vale la pena activar el control por software, ya que sería duplicar tareas innecesariamente.

1.10. Transmisión del USB

1.10.1. Transmisión asincrónica

Las distintas formas de transmisión de datos a distancia siempre fueron seriales, ya que el desfase de tiempos ocasionada por la transmisión paralela en distancias grandes impide pensar en esta última como apta para cubrir longitudes mayores a algunos pocos metros.

Sobre ello, la transmisión serial ha topado con el problema de que la información generada en el transmisor sea recuperada en la misma forma en el receptor, para lo cual es necesario ajustar adecuadamente un sincronismo entre ambos extremos de la comunicación. Para ello, tanto el receptor como el transmisor deben disponer de relojes que funcionen a la misma frecuencia y posibilite una transmisión exitosa. Como respuesta a este problema surgió la transmisión asincrónica, empleada masivamente años atrás para la comunicación entre los equipos servidores conocidos como hosts y sus terminales.

En este modelo cabe entender que ambos equipos poseen relojes funcionando a la misma frecuencia, por lo cual, cuando uno de ellos desea transmitir, prepara un grupo de bits encabezados por un BIT conocido como de arranque, un conjunto de 7 u 8 bits de datos, un BIT de paridad (para control de errores), y uno o dos bits de parada. El primero de los bits enviados anuncia al receptor la llegada de los siguientes, y la recepción de los mismos es efectuada. El receptor conocer perfectamente cuántos bits le llegarán, y da por recibida la información cuando verifica la llegada de los bits de parada. El esquema de los datos se muestra en la Figura siguiente.

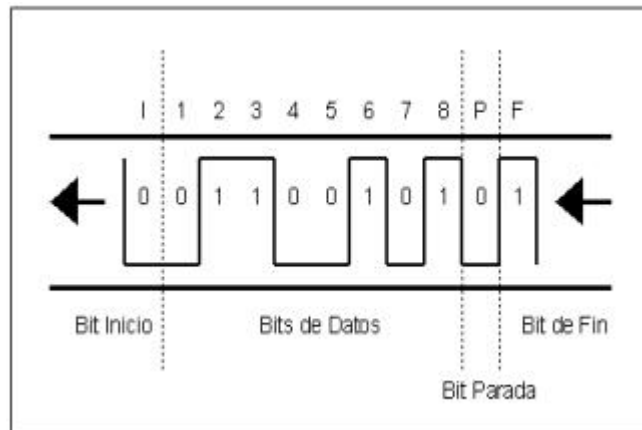


Figura I 13 Esquema de datos de la Transmisión asincrónica

Se denomina transmisión asincrónica no porque no exista ningún tipo de sincronismo, sino porque el sincronismo no se halla en la señal misma, más bien son los equipos mismos los que poseen relojes o clocks que posibilitan la sincronización. La sincronía o asincronía siempre se comprende a partir de la señal, no de los equipos de transmisión o recepción.

están solamente separados por un número entero de intervalos, medidos a partir de la duración de los bits. Contrasta con la transmisión asincrónica en la cual los caracteres pueden estar separados por intervalos aleatorios. La transferencia isocrónica provee comunicación continua y periódica entre el host y el dispositivo, con el fin de mover información relevante a un cierto momento. La transmisión isocrónica se encarga de mover información relevante a algún tipo de transmisión, particularmente audio y video.

1.10.4. Transmisión bulk

La transmisión Bulk, es una comunicación no periódica, explosiva típicamente empleada por transferencias que requieren usar todo el ancho de banda disponible o en su defecto son demoradas hasta que el ancho de banda completo esté disponible. Esto implica particularmente movimientos de imágenes o video, donde se requiere de gran potencial de transferencia en poco tiempo. USB permite dos tipos más de transferencias de datos:

1.10.5. Transmisiones de control

Es un tipo de comunicación exclusivamente entre el host y el dispositivo que permite configurar este último, sus paquetes de datos son de 8, 16, 32 o 64 bytes, dependiendo de la velocidad del dispositivo que se pretende controlar.

1.10.6. Transmisiones de interrupción

Este tipo de comunicación está disponible para aquellos dispositivos que demandan mover muy poca información y poco frecuentemente. Tiene la particularidad de ser unidireccional, es decir del dispositivo al host, notificando de algún evento o solicitando

alguna información. Su paquete de datos tiene las mismas dimensiones que el de las transmisiones de control.

1.11. CLASES USB

Una Clase USB es un grupo de dispositivos (o interfaces) con atributos o características similares. Las especificaciones para cada Clase permiten el desarrollo de dispositivos que pueden controlarse por medio de un manejador adaptativo, es decir, que se configura según la Clase reportada por el dispositivo. Dos dispositivos (o interfaces) pertenecen a la misma Clase si por ejemplo utilizan una misma forma de comunicarse con el sistema, o si por ejemplo utilizan el mismo formato de datos.

Las Clases USB se usan principalmente para describir la manera en que los dispositivos (o interfaces) se comunican con el sistema, incluyendo los mecanismos de control y datos, y adicionalmente algunas Clases se usan para identificar en todo o en parte la funcionalidad del dispositivo (o interfaz). En este caso, la Clase se puede utilizar para identificar qué manejador debe controlar dicho dispositivo (o interfaz).

Adicionalmente, los dispositivos de una Clase pueden agruparse en Subclases, lo que facilita aún más el que los manejadores puedan explorar el bus y seleccionar todos aquellos dispositivos que pueda controlar.

1.12. USB 1.1

El USB 1.1 es un bus externo que soporta tasas de transferencia de datos de 12 Mbps. Un solo puerto USB se puede utilizar para conectar hasta 127 dispositivos periféricos, tales como ratones, módems, y teclados. El USB también soporta la instalación *Plug-and-Play* y el *hot plugging*. Empezó a utilizarse en 1996, algunos fabricantes de

ordenadores empezaron a incluir soporte para USB en sus nuevas máquinas. Con el lanzamiento del iMac en 1998 el uso del USB se extendió. Se espera que substituya totalmente a los puertos de serie y paralelos.

1.13. USB 2.0

La especificación del USB 2.0 fue lanzada en abril de 2000. También conocido como *USB de alta velocidad*, el USB 2.0 es un bus externo que soporta tasas de transferencia de datos de hasta 480Mbps, es decir, 40 veces más rápido que USB 1.1. Será compatible con dispositivos USB 1.1 y utilizará los mismos cables y conectores

El USB 2.0 es una extensión del USB 1.1, utiliza los mismos cables y conectores y es completamente compatible con USB 1.1. Hewlett-Packard, Intel, Lucent, Microsoft, NEC y Philips tomaron juntos la iniciativa para desarrollar una tasa de transferencia de datos más alta que la del USB 1.1 para resolver las necesidades de ancho de banda de las nuevas tecnologías.

1.14. USB 1.0 y USB 2.0

Las versiones anteriores de USB, USB 1.0 y USB 1.1, viene con la capacidad para apoyar a tasas de transferencia de datos hasta 12 Mbps. Mientras USB 2.0 de alta velocidad, el último modelo de éxito a las calles, opera a 480 Mbps otros dispositivos pueden utilizar todo el ancho de banda disponible de USB. Baja velocidad de los dispositivos suelen utilizar un 1,5 Mb / s sub-canal.

1.15. USB sin cable

Wireless USB es una nueva aplicación diseñada para extender el uso y reconocimiento de la marca de USB. Esta nueva norma también permite la compatibilidad hacia atrás con USB 1.1 y USB 2.0 en el nivel de protocolo. Wireless USB está desarrollando como un cable de sustitución de la tecnología, y Ultra utilizar la tecnología inalámbrica de banda ancha para transmisiones de datos a velocidades de hasta 480 Mbps. Wireless USB es ideal para conexión inalámbrica de determinados dispositivos de ordenador portátil que permite transferencia de datos a ocurrir sin el uso de un cable.

Sólo tomó una década para más de 2 millones de ordenadores y dispositivos móviles para utilizar la tecnología USB. Consolas de videojuegos como Microsoft la Xbox 360, Sony PlayStation 2, Nintendo de la Revolución, y PDA's han hecho que la necesidad de la tecnología USB excepcionalmente fuerte para no mencionar las empresas innovadoras en los ámbitos de televisores y equipos estéreo casa adicional que permite la comunicación y las posibilidades de entretenimiento.

CAPÍTULO II

2. Modulo de Radio Frecuencia Xbee

2.1. Introducción

Zigbee es un protocolo de comunicaciones inalámbrico basado en el estándar de comunicaciones para redes inalámbricas IEEE_802.15.4. Creado por Zigbee Alliance, una organización, teóricamente sin ánimo de lucro, de más de 200 grandes empresas (destacan Mitsubishi, Honeywell, Philips, _ ODEM_ do, Invensys, entre otras), muchas de ellas fabricantes de semiconductores. Zigbee permite que dispositivos electrónicos de bajo consumo puedan realizar sus comunicaciones inalámbricas. Es especialmente útil para redes de sensores en entornos industriales, médicos y, sobre todo, domóticos. Las comunicaciones Zigbee se realizan en la banda libre de 2.4GHz. A diferencia de bluetooth, este protocolo no utiliza FHSS (Frequency hopping), sino que realiza las comunicaciones a través de una única frecuencia, es decir, de un canal. Normalmente puede escogerse un canal de entre 16 posibles. El alcance depende de la potencia de transmisión del dispositivo así como también del tipo de antenas utilizadas (cerámicas, dipolos, etc) El alcance normal con antena dipolo en línea vista es de aproximadamente

(tomando como ejemplo el caso de MaxStream, en la versión de 1mW de potencia) de 100m y en interiores de unos 30m. La velocidad de transmisión de datos de una red Zigbee es de hasta 256kbps. Una red Zigbee la pueden formar, teóricamente, hasta 65535 equipos, es decir, el protocolo está preparado para poder controlar en la misma red esta cantidad enorme de dispositivos. Entre las necesidades que satisface el módulo se encuentran:

- Bajo costo.
- Ultra-bajo consumo de potencia.
- Uso de bandas de radio libres y sin necesidad de licencias.
- Instalación barata y simple.
- Redes flexibles y extensibles.

El uso del protocolo Zigbee va desde reemplazar un cable por una comunicación serial inalámbrica, hasta el desarrollo de configuraciones punto a punto, multipunto, peer-to peer (todos los nodos conectados entre sí) o redes complejas de sensores. Una conexión típica se muestra en la figura 1-1, donde se observa que cada módulo Xbee posee algún tipo de sensor, el cual entrega los datos para ser enviados a través de la red a un Centro que administre la información.

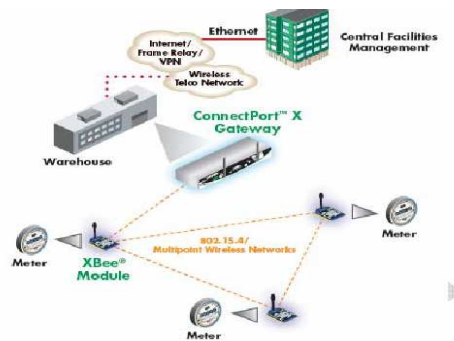


Figura II.1 Conexión típica usando Xbee.

Una red Zigbee la forman conexión 3 tipos de elementos. Un único dispositivo Coordinador, dispositivos Routers y dispositivos finales (end points).

2.2. El Coordinador.

Es el nodo de la red que tiene la única función de formar una red. Es el responsable de establecer el canal de conexión y del PAN ID (identificador de red) para toda la red. Una vez establecidos estos parámetros, el Coordinador puede formar una red, permitiendo unirse a él a dispositivos Routers y End Points. Una vez formada la red, el Coordinador hace las funciones de Router, esto es, participar en el enrutado de paquetes y ser origen y/o destinatario de información.

2.3. Los Routers.

Es un nodo que crea y mantiene conexión sobre la red para determinar la mejor ruta para transmitir un paquete de conexión. Lógicamente un router debe unirse a una red Zigbee antes de poder actuar como Router retransmitiendo paquetes de otros routers o de End points.

2.4. End Device.

Los dispositivos finales no tienen capacidad de enrutar paquetes. Deben interactuar siempre a través de su nodo padre, ya sea este un Coordinador o un Router, es decir, no puede enviar conexión directamente a otro end device. Normalmente estos equipos van alimentados a baterías. El consumo es menor al no tener que realizar funciones de enrutamiento

2.5. Conexión

Cada módulo Zigbee, al igual que ocurre con las direcciones MAC de los dispositivos ethernet, tiene una dirección única. En el caso de los módulos Zigbee cada uno de ellos tiene una dirección única de 64bits que viene grabada de fábrica. Por otro lado, la red Zigbee, utiliza para sus algoritmos de ruteo direcciones de 16 bits. Cada vez que un dispositivo se asocia a una red Zigbee, el Coordinador al cual se asocia le asigna una dirección única en toda la red de 16bits. Por eso el número máximo teórico de elementos que puede haber en una red Zigbee es de $2^{16} = 65535$, que es el nº máximo de direcciones de red que se pueden asignar.

Estos módulos Xbee, pueden ser ajustados para usarse en redes de conexión punto a punto, punto-a-multipunto o peer-to-peer. Un ejemplo se muestra en la Figura 2 1, donde se muestra una conexión multipunto, con un coordinador, conectado a varios nodos. **Digi2** tiene 2 series de módulos disponibles. Los módulos Xbee Serie 2 permite hacer redes mesh, La serie 1 no.

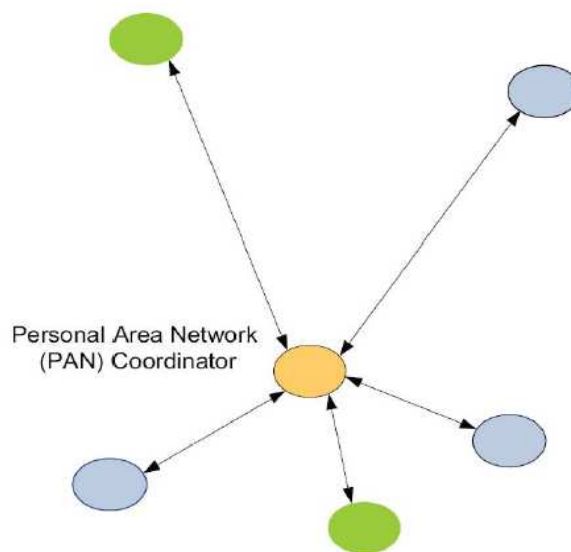


Figura II.2 Coordinador PAN con múltiples nodos

En la Figura I.3 se observan los elementos del Xbee. El Chip de la antena en la parte superior, el conector para la antena RF, y el conector para una antena integrada Whip.

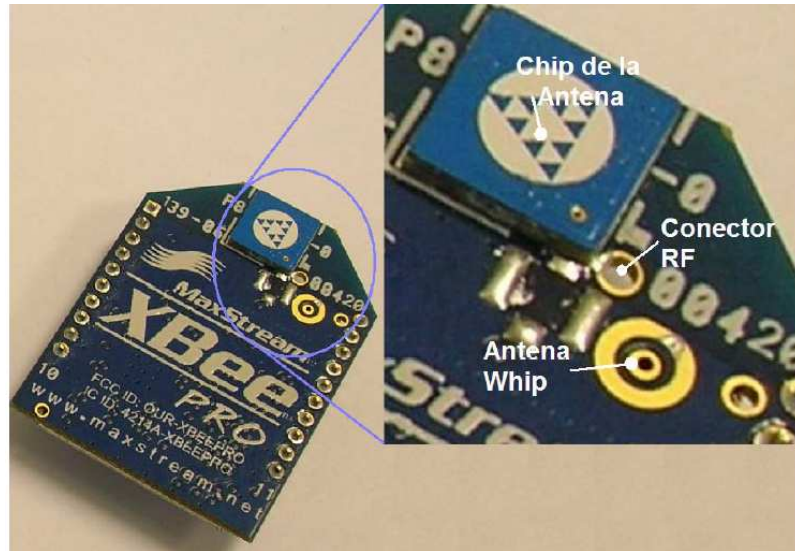


Figura II.3 Elementos del Xbee.

También existen los llamados módulos Xbee PRO de la Serie 1 que se diferencian en la capacidad de alcance, permitiendo en algunos casos doblar la distancia de transmisión, ya que poseen una mayor potencia en la señal. Con los módulos Xbee PRO de la Serie 2, es posible crear redes más complejas, como las llamadas MESH. Estas permiten acceder a un punto remoto, utilizando módulos intermedios para llegar como routers. Además los módulos automáticamente generaran la red entre ellos, sin intervención humana alguna, permitiendo la reparación de la red en caso de que algún nodo falle. Al mismo tiempo la red por sí sola resuelve la mejor ruta para un determinado paquete.

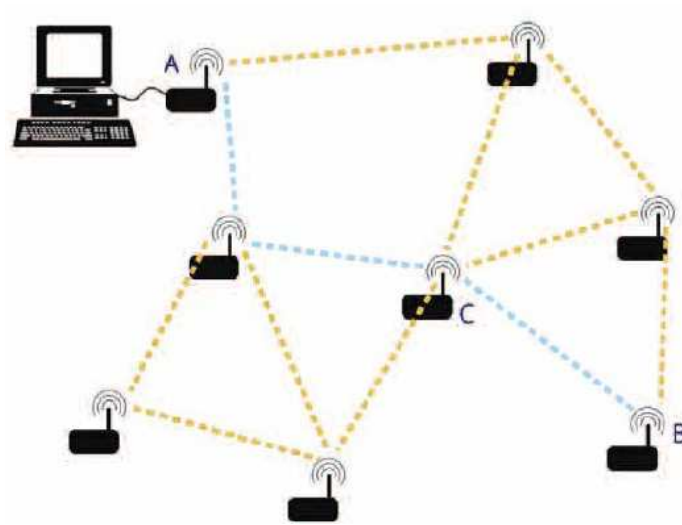


Figura II.4 Red Mesh para módulos Xbee PRO Serie 2.

La figura anterior muestra un ejemplo de una red MESH. Se observa que se desea acceder al punto B a partir del punto A. Suponiendo que la distancia entre A y B es demasiado para que alcance la señal, se utiliza la red MESH para poder alcanzarla, así cada nodo ubicado en medio del camino mostrado en celeste, funciona como módulo transparente, donde todo lo que le llega es retransmitido hacia el punto A. Esto funciona para cada uno de los módulos de la red.

2.6. Circuito básico para el Xbee.

La figura 3-1 muestra las conexiones mínimas que necesita el módulo Xbee para poder ser utilizado. Luego de esto, se debe configurar según el modo de operación adecuado para la aplicación requerida por el usuario.

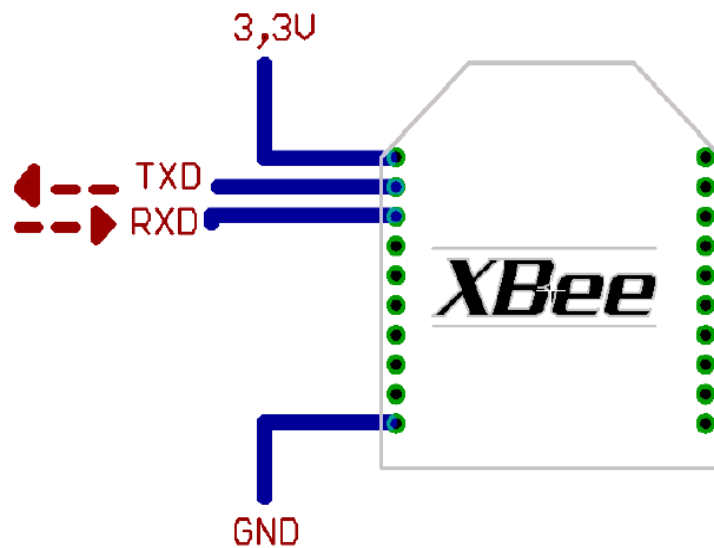


Figura II.5 Conexiones mínimas requeridas para el Xbee.

El módulo requiere una alimentación desde 2.8 a 3.4 V, la conexión a tierra y las líneas de transmisión de datos por medio del UART (TXD y RXD) para comunicarse con un microcontrolador, o directamente a un puerto serial utilizando algún conversor adecuado para los niveles de voltaje. Esta configuración, no permite el uso de Control de Flujo (RTS & CTS), por lo que ésta opción debe estar desactivada en el `_ODEM_` do y en el módulo Xbee. En caso de que se envíe una gran cantidad de información, el buffer del módulo se puede sobrepasar. Para evitar esto existen dos alternativas:

- bajar la tasa de transmisión
- activar el control de flujo.

2.7. Modos de Operación.

Los módulos Xbee, pueden operar en los siguientes 5 modos:

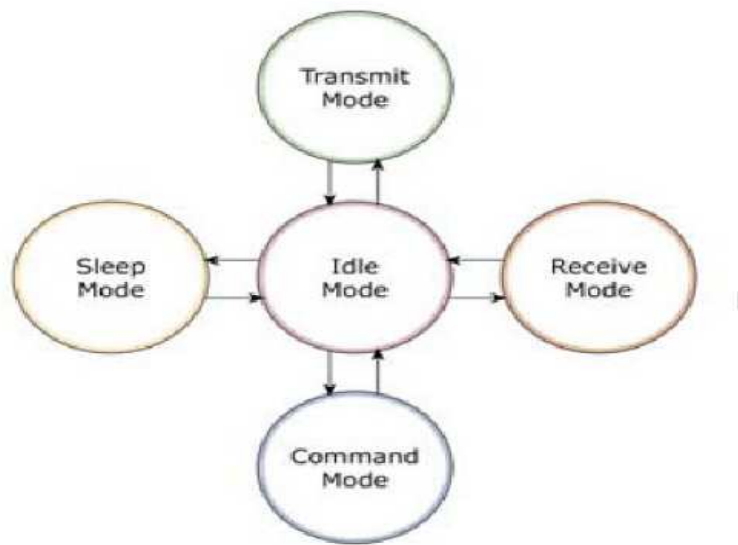


Figura II.6 Modos de operación del módulo Xbee.

2.7.1. Modo RECIBIR / TRANSMITIR

Se encuentra en estos modos cuando el módulo recibe algún paquete RF a través de la antena(modos Receive) o cuando se envía información serial al buffer del pin 3 (UART Data in) que luego será transmitida (modo Transmit).

La información transmitida puede ser Directa o Indirecta. En el modo directo la información se envía inmediatamente a la dirección de destino. En el modo Indirecto la información es retenida por el módulo durante un período de tiempo y es enviada sólo cuando la dirección de destino la solicita.

Además es posible enviar información por dos formas diferentes. Unicast y Broadcast.

Por la primera, la comunicación es desde un punto a otro, y es el único modo que permite respuesta de quien recibe el paquete RF, es decir, quien recibe debe enviar un **ACK** (paquete llamado así, y que indica que recibió el paquete, el usuario no puede

verlo, es interno de los módulos) a la dirección de origen. Quien envió el paquete, espera recibir un **ACK**, en caso de que no le llegue, reenviará el paquete hasta 3 veces o hasta que reciba el **ACK**. Después de los 3 intentos se incrementa el registro de fallas **ACK** en uno. Para ver el conteo se utiliza el comando EA (**ATEA** en el Modo de Comandos– ACK Failures), el cual se satura hasta 0xFFFF (65535 decimal). Para resetear el registro se debe ingresar **ATEA0**. En el modo Broadcast la comunicación es entre un nodo y a todos los nodos de la red. En este modo, no hay confirmación por **ACK**.

2.7.2. Modo de Bajo Consumo (Sleep Mode)

El modo de Bajo Consumo o modo Sleep, corresponde cuando el módulo entra en un estado de bajo consumo de energía. Esto depende de la configuración en la que se encuentra. Se debe indicar que el ahorro de energía depende enormemente del voltaje de alimentación³.

Para entrar a este modo se debe configurar el comando **SM** (ATSM – Sleep Mode). Si SM=4 o SM=5, el módulo entra al modo SLEEP cuando transcurre un período dado por el comando **ST** (ATST-Time Before Sleeping-tiempo antes de dormir). Donde ST posee un rango entre 0 y 0xFFFF (x 1ms). Si SM=4, el módem despertará cada cierto tiempo, dado por **SP** (ATSP – Cyclic Sleep Period), cuyo rango es entre 0 y 0x68B0 (x10 ms). Cuando despierte buscará por datos entrantes desde un módulo configurado como **Coordinador**, en caso de no haber nada, el módem volverá al modo SLEEP. Si SM=5, el módulo realzará lo mismo que antes, pero despertará de acuerdo al estado del pin **SLEEP_RQ** (pin 9). Cuando el pin pase a LOW (el pin es disparado por detección de

borde, no por nivel), despertará, buscará por datos entrantes, y si no hay nada volverá a cero el timer del comando ST, por lo que sólo volverá a dormir cuando transcurra el período dado por éste. Mientras transcurra ese período, cualquier actividad en el pin SPEEL_RQ será ignorada, hasta que vuelva al modo SLEEP. Si SM=1, el módulo entrará al estado de hibernación. En este estado, si el pin SLEEP_RQ está en HIGH, el módulo cortará cualquier actividad entrante, ya sea de transmisión, recepción o de asociación (se verá más adelante) y entrará al modo SLEEP, y no saldrá de ahí hasta que el pin SLEEP_RQ vuelva a estar en LOW. Mientras se encuentre en el estado de hibernación, no responderá a ninguna actividad serial ni a ningún paquete RF entrante, simplemente los desechará. En el modo hibernación el sistema ahorra una gran cantidad de potencia. Si SM=2, el sistema se comporta igual que el modo de hibernación, pero sin tanto ahorro de energía y además despierta mucho más rápido que el modo anterior. La siguiente tabla explica lo anterior:

Modo	Consumo alimentación			Modo Sleep	Modo Wake-up
	2.8 – 3 V	3.2 V	3.4V		
SM=1	<3 uA	32 uA	255 uA	Sleep_RQ	Sleep_RQ
SM=2	<35 uA	48 uA	170 uA	Sleep_RQ	Sleep_RQ
SM=3	(reservado)			(reservado)	(reservado)
SM=4	<34 uA	49 uA	240 uA	Comando ST	Comando SP
SM=5	<34 uA	49 uA	240 uA	Comando ST	Sleep RQ

Tabla II.1 Tabla Modo Sleep y consumos de corriente.

2.7.3. Modo de Comando.

Este modo permite ingresar comandos AT al módulo Xbee, para configurar, ajustar o

modificar parámetros. Permite ajustar parámetros como la dirección propia o la de destino, así como su modo de operación entre otras cosas. Para poder ingresar los comandos AT es necesario utilizar el Hyperterminal de Windows, el programa X-CTU4 o algún microcontrolador que maneje UART y tenga los comandos guardados en memoria o los adquiera de alguna otra forma.

Para ingresar a este modo se debe esperar un tiempo dado por el comando GT (Guard Time, por defecto ATGT=0x3E85 que equivalen a 1000ms) luego ingresar +++ y luego esperar otro tiempo GT. Como respuesta el módulo entregará un **OK**. El módulo Xbee viene por defecto con una velocidad de 9600bps. En caso de no poder ingresar al modo de comandos, es posible que sea debido a la diferencia de velocidades entre el módulo y la interfaz que se comunica vía serial.

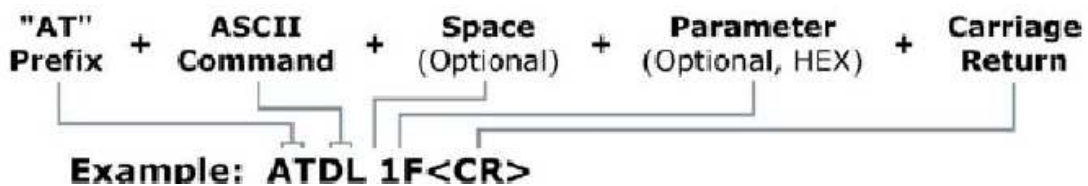


Figura II.7 Ejemplo Comando AT.

En la figura anterior se muestra la sintaxis de un comando AT. Luego de ingresar a este modo, se debe ingresar el comando deseado para ajustar los parámetros del módulo Xbee. La lista de comandos se encuentra en las siguientes secciones.

Por ejemplo si se desea modificar la dirección de origen del módulo de 16-bit (con el comando MY), se debe ingresar:

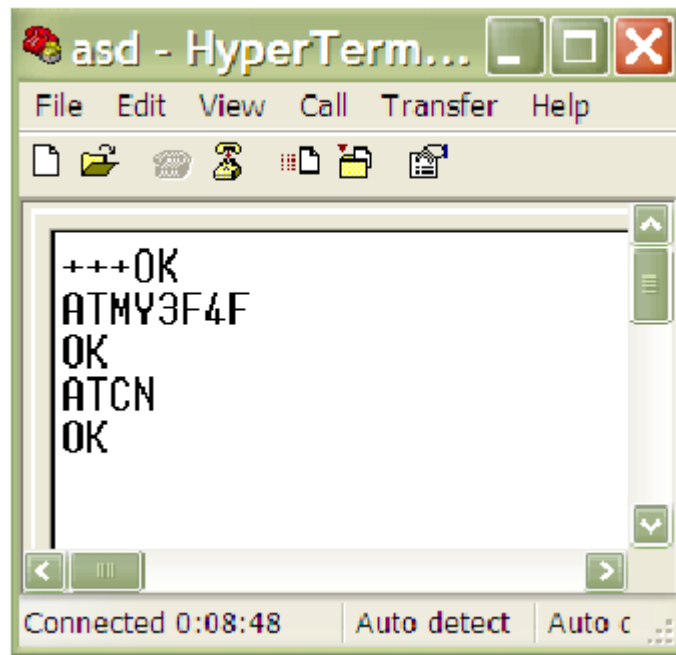


Figura II.8 Ejemplo escritura de parámetros.

En el ejemplo anterior, la dirección asignada equivale a 0x3F4F. Se observa que primero se ingresa al modo de comandos AT, recibiendo un **OK** de respuesta. Luego se ingresa el comando **ATMY3F4F** y se presiona **ENTER** o carácter <CR><LF> (Carrier Return y Line Feed) si se maneja desde un microcontrolador. Con ello se recibe un <CR><LF>**OK**<CR><LF> como respuesta. Para salir del modo de Comandos se ingresa **ATCN** y se presiona **ENTER**. En caso de que no se ingrese ningún comando AT válido durante el tiempo determinado por CT (Command Mode Timeout), el módulo se saldrá automáticamente del modo de comandos. Para que los cambios realizados tengan efecto se debe ingresar el comando **ATCN** (sale del modo de comandos) o **ATAC** (aplica los cambios inmediatamente). Con el comando **ATWR**, se guardan los cambios en la memoria no volátil del módulo, pero sólo tendrán efecto una vez ingresado el comando AC o CN.

Para consultar el valor asignado a un comando, éste de debe ingresar directamente sin

ningún parámetro. En el ejemplo anterior se ingresaría **ATMY**. La siguiente figura muestra lo anterior:

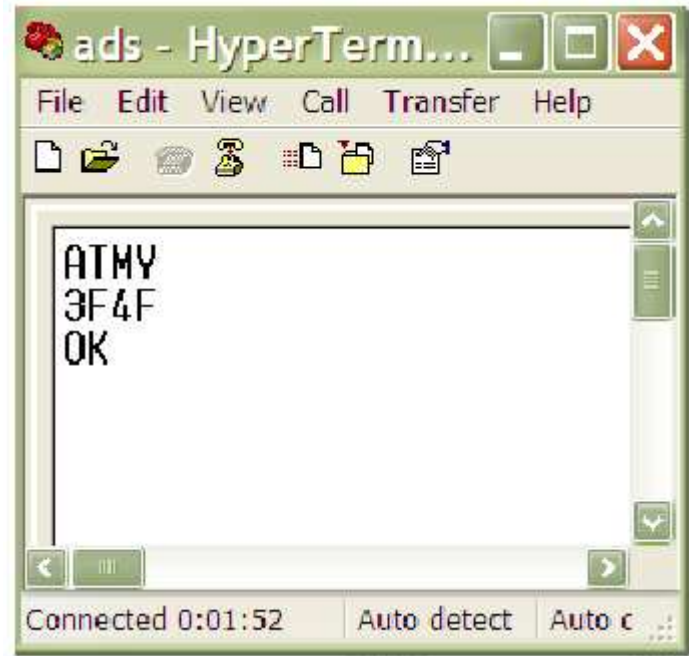


Figura II.9 Ejemplo del comando ATMY

Se observa que se ingresa **ATMY**, donde el que módulo responde con un **3F4F**, que es la dirección que tiene configurada, y luego con un **OK**. Otra forma de configurar el módulo por comandos AT, es ingresando varios comandos separados por *coma* (`,`). Esto se muestra a continuación de las dos formas:

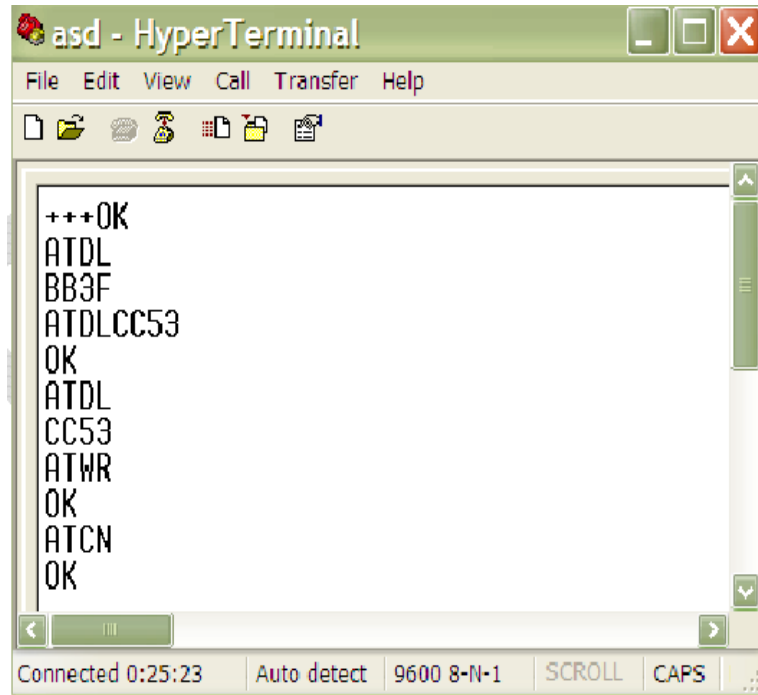


Figura II.10 Forma normal de escribir.

Se observa que luego de ingresar al modo de Comandos (+++), se pregunta por la dirección de destino (**ATDL**), para lo cual se tiene como respuesta la dirección 0xBB3F. Con ello se modifica esta dirección por 0xCC53 ingresando **ATDLCC53**, obteniendo un **OK** como respuesta. Se confirma el correcto ajuste preguntando nuevamente y luego se guarda la configuración en la memoria no volátil del módulo usando **ATWR**, para lo cual se vuelve a obtener un **OK**. Posteriormente se saldrá del modo de comando utilizando **ATCN** obteniendo otro **OK**. Otra forma de hacer lo mismo se muestra en la siguiente figura:

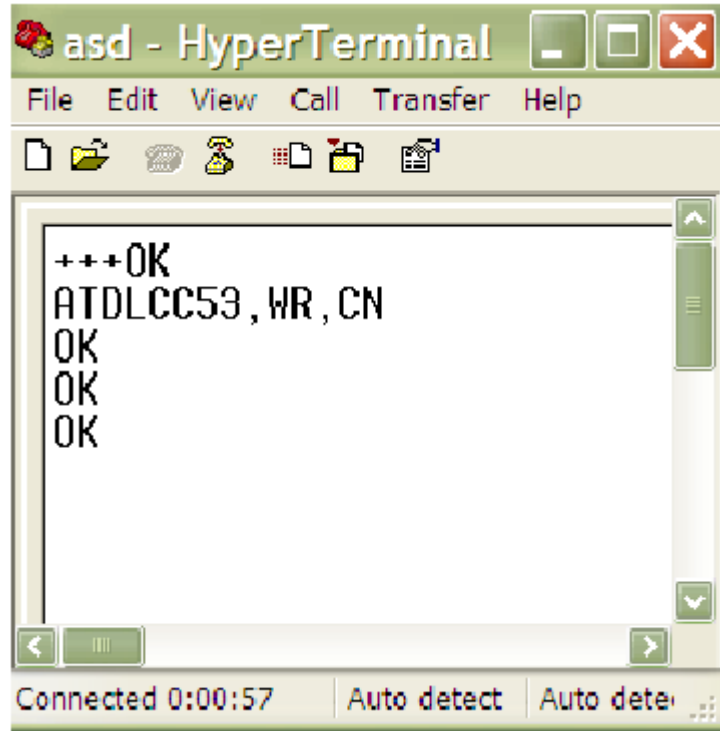


Figura II.11 Forma abreviada para varios comandos al mismo tiempo.

Se observa que luego de ingresar al modo de comandos, se ingresa **ATDLCC35,WR,CN** que indica que se ingresarán tres comandos en uno, y éstos serán **ATDLCC35**, **ATWR** y **ATCN**. Luego se obtendrá la respuesta para cada comando en el mismo orden en que fueron ingresados. Además de los modos anteriores, existen otros modos referentes a lo que son la forma de transmitir la información. Estos modos son: Modo Transparente y Modo API.

2.7.4. Modo Transparente

En este modo todo lo que ingresa por el pin 3 (Data in), es guardado en el buffer de entrada y luego transmitido y todo lo que ingresa como paquete RF, es guardado en el buffer de salida y luego enviado por el pin 2 (Data out). El modo Transparente viene por

defecto en los módulos Xbee. Este modo está destinado principalmente a la comunicación punto a punto, donde no es necesario ningún tipo de control. También se usa para reemplazar alguna conexión serial por cable, ya que es la configuración más sencilla posible y no requiere una mayor configuración.

En este modo, la información es recibida por el pin 3 del módulo Xbee, y guardada en el buffer de entrada. Dependiendo de cómo se configure el comando RO, se puede transmitir la información apenas llegue un carácter (RO=0) o después de un tiempo dado sin recibir ningún carácter serial por el pin 3.

En ese momento, se toma lo que se tenga en el buffer de entrada, se empaqueta, es decir, se integra a un paquete RF, y se transmite. Otra condición que puede cumplirse para la transmisión es cuando el buffer de entrada se llena, esto es, más de 100 bytes de información.

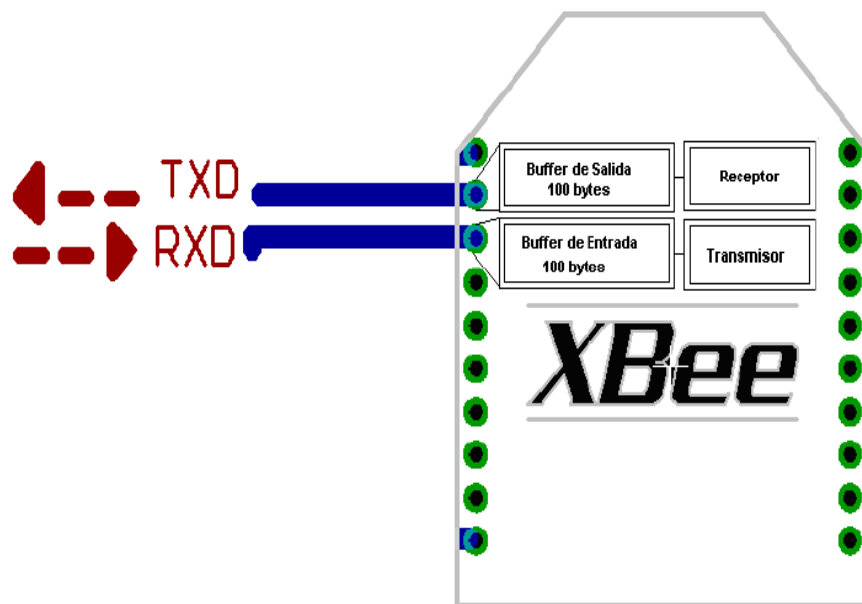


Figura II.12 Correspondencia de Buffer.

2.8. Modo de operación API

Este modo es más complejo, pero permite el uso de frames con cabeceras que aseguran la transmisión de los datos, al estilo TCP. Extiende el nivel en el cual la aplicación del cliente, puede interactuar con las capacidades de red del módulo.

Cuando el módulo Xbee se encuentra en este modo, toda la información que entra y sale, es empaquetada en frames, que definen operaciones y eventos dentro del módulo. Así, un frame de Transmisión de Información (información recibida por el pin 3 o DIN) incluye:

- Frame de información RF transmitida.
- Frame de comandos (equivalente a comandos AT).

Mientras que un Frame de Recepción de Información incluye:

- Frame de información RF recibida.
- Comando de respuesta.
- Notificaciones de eventos como Reset, _ODEM_ do, Disassociate, etc.

Esta API, provee alternativas para la configuración del módulo y ruteo de la información en la capa de aplicación del cliente. Un cliente puede enviar información al módulo Xbee. Estos datos serán contenidos en un frame cuya cabecera tendrá información útil referente el módulo. Esta información además se podrá configurar, esto es, en vez de estar usando el modo de comandos para modificar las direcciones, la API lo realiza automáticamente. El módulo así enviará paquetes de datos contenidos en frames a otros módulos de destino, con información a sus respectivas aplicaciones, conteniendo paquetes de estado, así como el origen, RSSI (potencia de la señal de recepción) e información de la carga útil de los paquetes recibidos. Entre las opciones que permite la API, se tienen:

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

2.9. IDLE

Cuando el módulo no se está en ninguno de los otros modos, se encuentra en éste. Es decir, si no está ni transmitiendo ni recibiendo, ni ahorrando energía ni en el modo de comandos, entonces se dice que se encuentra en un estado al que se le llama IDLE.

2.10. Configuración del Módulo Xbee.

En este capítulo se explicará cómo configurar los módulos para los distintos tipos de redes que soporta Xbee. Además se mostrarán algunos ejemplos de topologías de redes. Para que los cambios realizados tengan efecto se debe ingresar ATCN (sale del modo de comandos) o ATAC (aplica los cambios inmediatamente).

2.11. Direccionamiento de los módulos

Los módulos permiten 2 tipos de direccionamiento. La de 16 bit y la de 64 bits. La principal diferencia es que en la de 64 bit, es posible obtener una mayor cantidad de direcciones y por lo tanto, una mayor cantidad de nodos o equipos funcionando en la misma red. Son a través de estas direcciones que los módulos se comunican entre sí.

La dirección de origen de 16 bits del módulo se define arbitrariamente con el comando MY (ATMY – 16 bit Source Address). La de destino con los comandos DL (ATDL Destination Address Low) y DH (Destination Address High).

2.12. Direccionamiento de 16 bit.

El comando MY, define un número de 16 bit como dirección del módulo dentro de la red. El rango se encuentra entre 0x0 y 0xFFFFE (la dirección 0xFFFF y 0xFFFFE son para habilitar la dirección de 64-bit, por lo que si se desea utilizar direccionamiento de 16 bits, estos valores no deben ser usados). Para definirla se ingresa ATMY y el número en formato hexadecimal, pero sin el '0x'. Por ejemplo si a un módulo se le quiere asignar la dirección 0x3BF1 (15345 en decimal), entonces se debe ingresar el comando ATMY3BF1. El comando DL, permite definir un número de 16 bit como dirección del módulo de destino dentro de la red al cual se va a realizar la comunicación. El rango debe estar entre 0x0 y 0xFFFFE (las direcciones 0xFFFFE y 0xFFFF se utilizan para direccionamiento de 64 bits). Así para habilitar el direccionamiento de 16 bit, se debe utilizar una dirección menor a 0xFFFFE con el comando MY, de igual modo para DL y se debe dejar en cero el comando DH=0 (ATDH0). No se permite usar la dirección 0xFFFFE ni 0xFFFF para el direccionamiento de 16 bits.

2.13. Direccionamiento de 64 bits.

El número 0xFFFF y 0xFFFFE del comando MY, se usa cuando se desea desactivar el direccionamiento de 16 bit, y se habilita el uso de la dirección de 64 bit. Con este direccionamiento ya no es posible definir la dirección de origen del módulo, ya que ésta se asigna automáticamente. En este caso, la dirección del módulo corresponde a su número serial, que viene de fábrica y el cual es imposible de cambiar. Este número se encuentra guardado en dos variables de 32 bit cada una (SL y SH) y es único. SL lee los 32 bit menos significativos del número serial y SH los 32 más significativos

Cuando se utiliza direccionamiento de 64 bit, para asignar una dirección de destino, se utilizan los comandos DL y DH. Éstos son de 32 bit cada uno (para el direccionamiento de 16 bit, DL se maneja como uno de 16, mientras que DH se mantiene en cero) y juntos (DL+DH) forman el número de 64 bit que debe corresponder con el número serial de otro módulo formado por SL+SH. Así para `_ ODEM_ dor_ 1` algún dato, DL debe ser igual a SL y DH debe ser igual a SH, donde SL+SH corresponden al número serial de un módulo destino configurado para direccionamiento de 64 bits.

Para el direccionamiento de 64 bit, se debe dejar MY como 0xFFFF (ATMYFFFF) o 0xFFFE (ATMYFFFE) y elegir una dirección de destino usando DL+DH, que debe corresponder a una dirección de 64 bit de otro módulo, indicando su número serial dado por SL+SH. Para consultar este número se debe ingresar ATSL (32 bit menos significativos) y luego ATSH (32 bit más significativos), entregando como respuesta los números seriales en formato hexadecimal.

2.14. Modo de Conexión Transparente.

Esta es la conexión que viene por defecto y es la forma más sencilla de configurar el módem. Básicamente todo lo que pasa por el puerto UART (DIN, pin 3), es enviado al módulo deseado, y lo recibido en el módulo, es enviado devuelta por el mismo puerto UART (DOOUT, pin2). Existen básicamente 4 tipos de conexión transparente. La diferencia principal radica en el número de nodos o puntos de acceso, y la forma en que éstos interactúan entre sí.

2.15. Punto a Punto.

Es la conexión ideal para reemplazar comunicación serial por un cable. Sólo se debe configurar la dirección. Para ello se utilizan los comandos MY y el DL. La idea, es definir arbitrariamente una dirección para un módulo, usando el comando MY, el cual se va a comunicar con otro que tiene la dirección DL, también definida arbitrariamente. Con esto cada módulo define su dirección con MY, y escribe la dirección del módulo al cual se desea conectar usando DL.

En este modo, el módulo receptor del mensaje envía un paquete al módulo de origen llamado ACK (viene de Acknowledgment) que indica que el mensaje se recibió correctamente. En la siguiente figura se muestra un pequeño ejemplo donde las direcciones se eligieron arbitrariamente:

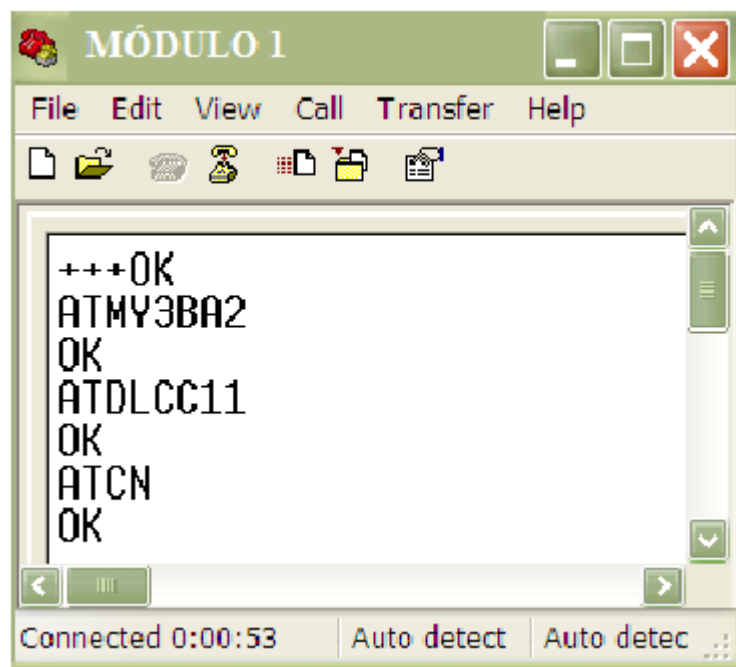


Figura II.13 Configuración del modulo 1.

Se observa que en el módulo 1, se ajustó la dirección de origen como 0x3BA2 (ATMY3BA2), mientras que la dirección de destino se asignó como 0xCC11 (ATDLCC11) que corresponde al módulo 2. En el otro módulo se observa lo siguiente:

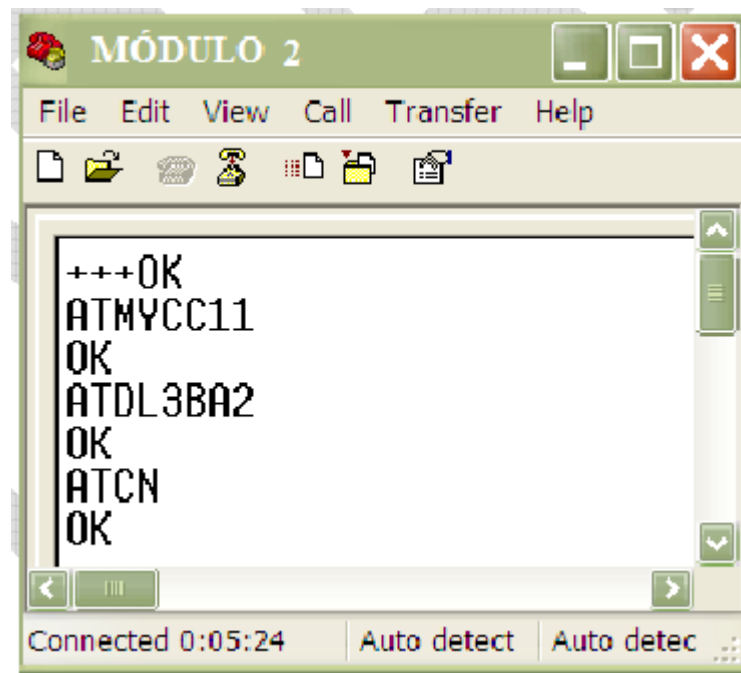


Figura II.14 Configuración del modulo 2.

En este módulo se asignó como dirección de origen 0xCC11 (ATMYCC11) y como dirección de destino 0x3BA2 (ATDL3BA2) que corresponde al módulo 1. La siguiente figura muestra un ejemplo gráfico de lo anterior.

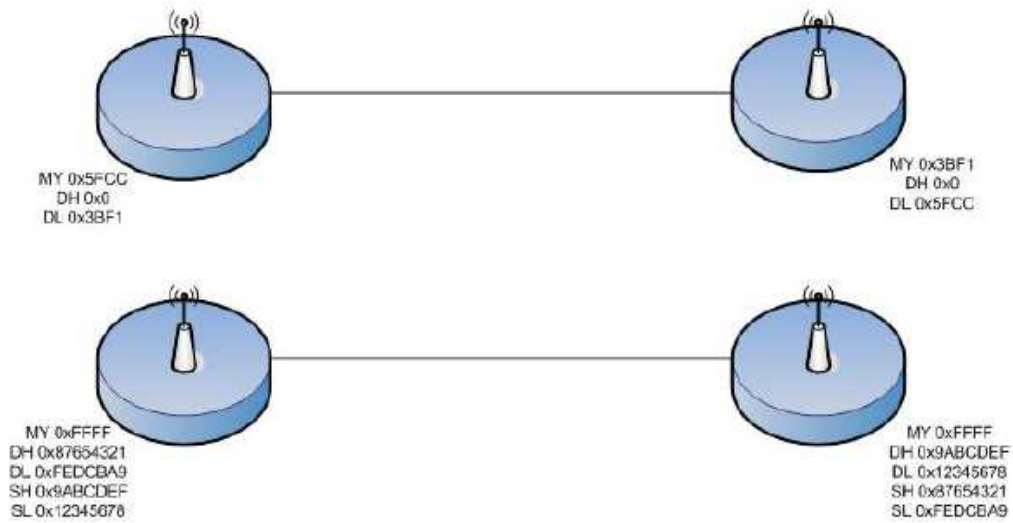


Figura II.15 Ejemplo direccionamiento 16 y 64 bit.

En la figura anterior se muestra que la primera conexión es una punto a punto utilizando direccionamiento de 16 bit, mientras que la segunda utiliza direccionamiento de 64 bits. Una vez configurado, el módem se encuentra listo para funcionar. Así todo lo que se transmite por el pin DIN de un módulo, es recibido por el pin DOUT del otro. Para que el modo Punto a Punto funcione, los módulos deben pertenecer a la misma PAN ID y al mismo canal. Más adelante se explica cómo configurar la PAN ID y el Canal.

2.16. Punto a Multipunto.

Esta conexión, permite prestaciones extras. Se diferencia del Broadcast, en que permite transmitir información, desde la entrada serial de un módulo (DIN, pin 3) a uno o varios módulos conectados a la misma red de manera más controlada, ya que se necesitan las direcciones de los

otros módulos, por lo que existe mayor seguridad. Para esto se necesitan dos comandos más aparte de MY y DL. Se utilizará el direccionamiento de 16 bits.

El primer comando es el ID de la PAN (Personal Area Network- Red de Área Personal). Todos los módulos que tengan idéntico PAN ID, pertenecerán a la misma red. El comando para configurar este valor es ID, es decir, **ATID**, y su rango va entre 0x0 y 0xFFFF. Por ejemplo si queremos ajustar el PAN ID como 0x3332, se debe ingresar ATID3332. Este parámetro también es arbitrario, al igual que MY y DL.

El otro comando corresponde al canal por el cual se va a comunicar. Según la Figura 5 4, se disponen de 16 canales según el protocolo IEEE 802.15.4. Este estándar indica que entre cada canal, deben existir 5 MHz de diferencia, partiendo de la frecuencia base 2.405 GHz, se llegan hasta los 2.480 GHz.

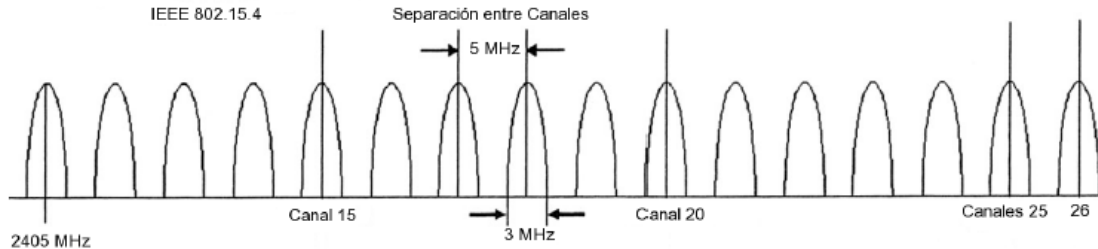


Figura II.16 Canales disponibles para el protocolo IEEE 802.15.4

Se observa que hay 16 canales disponibles, sin embargo, los valores se asignan desde el 11 hasta el 26. Para calcular la frecuencia central se utiliza la siguiente fórmula:

$$Canal = 2.405 + (CH - 11) \times 0.005 \text{ [GHz]}$$

Donde CH equivale al número del canal entre 11 y 26. Así para cambiar de canal se utiliza el comando CH con el número de canal en formato hexadecimal. Es decir, si se

desea ocupar el canal 15 (0x10), se ingresa ATCH10. La siguiente tabla muestra la frecuencia central de cada canal, así como su límite inferior y superior:

Frecuencia [GHz]					
canal	hexadecimal	inferior	central	superior	Comando AT
11	0x0B	2,4025	2,4050	2,4075	ATCH0B
12	0x0C	2,4075	2,4100	2,4125	ATCH0C
13	0x0D	2,4125	2,4150	2,4175	ATCH0D
14	0x0E	2,4175	2,4200	2,4225	ATCH0E
15	0x0F	2,4225	2,4250	2,4275	ATCH0F
15	0x10	2,4225	2,4250	2,4275	ATCH10
17	0x11	2,4325	2,4350	2,4375	ATCH11
18	0x12	2,4375	2,4400	2,4425	ATCH12
19	0x13	2,4425	2,4450	2,4475	ATCH13
20	0x14	2,4475	2,4500	2,4525	ATCH14
21	0x15	2,4525	2,4550	2,4575	ATCH15
22	0x16	2,4575	2,4600	2,4625	ATCH16
23	0x17	2,4625	2,4650	2,4675	ATCH17
24	0x18	2,4675	2,4700	2,4725	ATCH18
25	0x19	2,4725	2,4750	2,4775	ATCH19
26	0x1A	2,4775	2,4800	2,4825	ATCH1A
Frecuencia Base		2,405	GHz		

Tabla II.2 Frecuencia de Canales y su respectivo Comando AT.

La elección del canal debe ser cuidadosa, ya que otras tecnologías como WI-FI o Bluetooth utilizan el mismo espectro de frecuencias, por lo que se podría producir interferencia. Con todo lo anterior, es posible configurar una PAN y hacer una conexión punto a multipunto. La red se vería como:

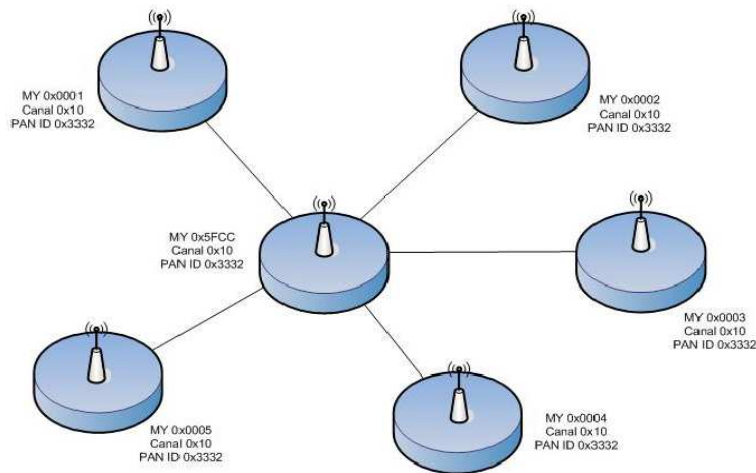


Figura II.17 Configuración punto-a-multipunto.

Así en cada nodo se configura una dirección MY distinta, pero utilizando el mismo canal y el mismo PAN ID, que en la figura corresponden al canal 15 (0x10) y al ID 0x3332 de la PAN. Para que cada módulo reciba la información, debe ser estrictamente necesario que tengan tanto el mismo canal, como el mismo PAN ID. Incluso si se trabaja en Broadcast o punto a punto los módulos deben coincidir en ello. Los módulos vienen por defecto configurados con el canal 0x0C y el PAN ID 0x3332 por defecto. Esta configuración, permite enviar información más controlada, ya que es necesario pertenecer tanto al mismo canal, como a la misma red. Además para enviar información se debe ingresar la dirección del módulo de destino, por lo que es necesario el conocimiento completo de la red. En los módulos más avanzados, como Xbee PRO, el reconocimiento de la red se realiza automáticamente.

2.17. Broadcast

Esta configuración permite el envío de información desde un nodo a varios nodos en una misma red PAN. La información recibida es la misma para todos los nodos. Para

que un módulo entregue datos a todos los nodos, es necesario ajustarlo con la dirección de Broadcast. Cualquier módulo que reciba un paquete con una dirección de destino de Broadcast será aceptado.

La dirección de Broadcast es:

- DL=0x0000FFFF
- DH=0x00000000

Esta dirección puede ser configurada en todos los nodos de la red, ya sea que estén en direccionamiento de 16 o 64 bits. Así se puede ingresar **ATDH0** y **ATDL0000FFFF** en todos los módulos para que el modo broadcast esté habilitado. Para que este modo funcione, los módulos deben pertenecer a la misma PAN ID y al mismo canal. La siguiente figura muestra una red de Broadcast en cada nodo:

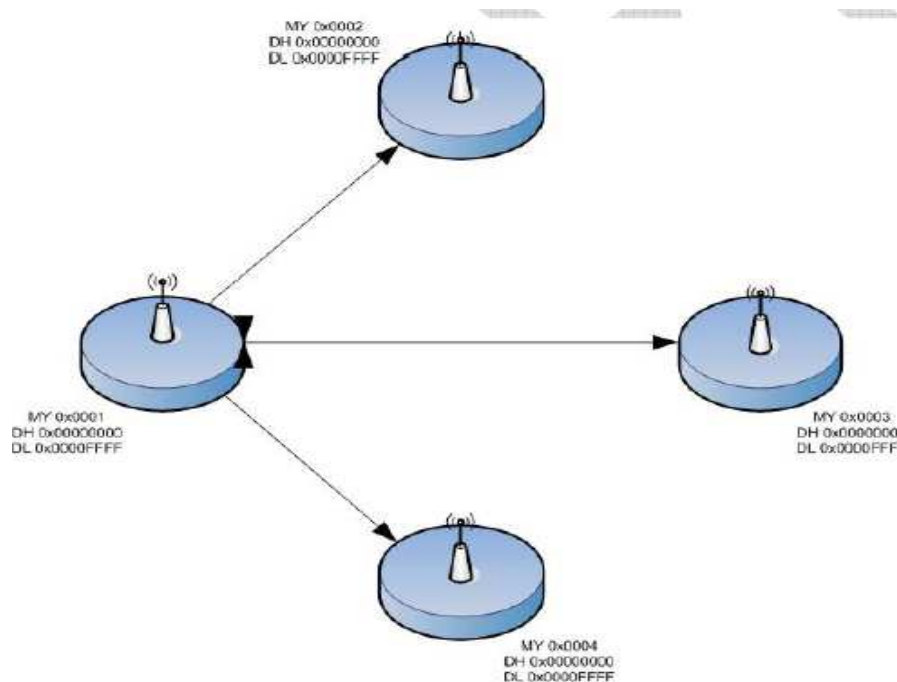


Figura II.18 Red de Broadcast.

Se observa en la figura anterior la configuración de Broadcast. Si se envía algún dato por el módulo 0x0001, la información enviada será recibida por igual en el resto de los

módulos (0x0002, 0x0003 y 0x0004). Del mismo modo si se envía algún dato por otro módulo, por ejemplo por el 0x0004, este dato le llegará al resto, es decir, al 0x0001, 0x0002 y al 0x0003. Cabe mencionar que este tipo de red o de envío de datos, no entrega respuesta de recibo o ACK, por lo que no es posible saber si el paquete fue entregado correctamente o si es que llegó. Si se ajusta la dirección PAN ID del módulo como ID=0xFFFF, se produce Broadcast a todas las redes PAN. Esto es, los datos son transmitidos a las distintas redes PAN, pero no se confirma la entrega de éstos (no se recibe ACK). Si se ingresa ID=0xFFFF y además DL=0xFFFF se realiza doble broadcast, es decir, además de transmitirse los datos a todas las redes PAN, el mensaje es transmitido a todos los módulos de cada una de ellas. Si se ingresa ID=0xFFFF y DL=0xAAAA (dirección arbitraria), los datos son transmitidos a todos los módulos que posean la dirección AAAA, pero que no necesariamente se encuentren en la misma red PAN.

2.18. Cable Virtual I/O

Esta opción de configuración permite crear los llamados Cables Virtuales. Se utilizan para crear un canal de comunicación de manera transparente entre los pines de un módulo y otro. Cada pin de entrada tiene su propio pin de salida ya definido entre nodos, esto permite una forma totalmente simple de enviar información, controlar o medir de manera sencilla y rápida, sin necesidad de complicadas configuraciones. El esquemático de los pines se muestra en la siguiente figura:

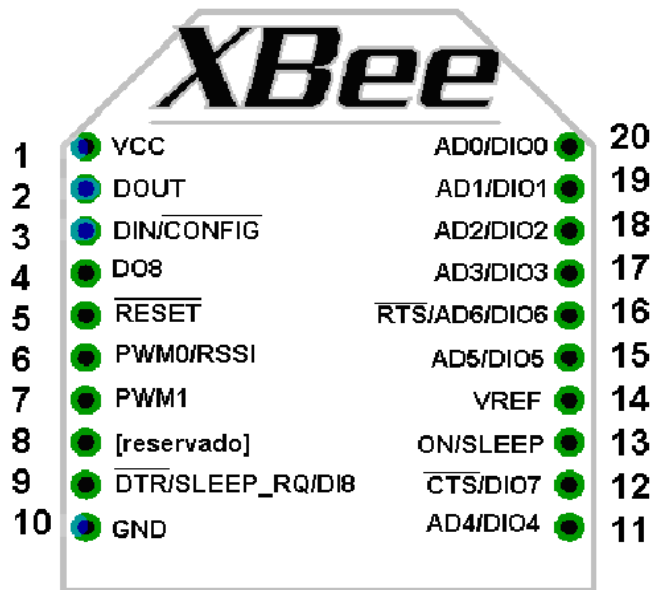


Figura I.I.19 Diagrama de pines del módulo Xbee. Vista Superior.

Como se dijo, el envío es totalmente transparente para el usuario, ya que el paquete recibido, puede o no ser entregado por el pin DOUT (pin 3) dependiendo de la configuración, lo que permite utilizar ese pin para el envío de otro tipo de información e incluso seguir recibiendo desde otros módulos.

Para crear un cable virtual, se debe notar que los pines de entrada/salida o I/O, están asociados entre sí, es decir, vienen en pares. Por esto, sólo se pueden crear cables entre estos pares de pines de distintos módulos. Por ejemplo, una entrada digital DI5, sale por una salida digital DO5 y una entrada analógica ADC 0, sale por una salida analógica PWM 0. Algunas entradas y salidas tienen el mismo pin asignado, pero en otros como los analógicos, éstas encuentran en diferentes pines. Por ejemplo DI-0 y DO-0 están en el pin 20 (DIO-20). Pero la entrada ADC-1 tiene la salida PWM-1, asignados a los pines 19 (AD1) y 6 (PWM1) respectivamente. Para el caso analógico, la salida sólo corresponde con los pines PWM, es decir, AD0 (pin 20) con PWM0 (pin 6) y AD1 (pin

19) con PWM1 (pin 7). Cabe mencionar que DIO8 (ni DO8 ni DI8, pines 4 y 9 respectivamente) no puede usarse como cable virtual.

Para configurar los pines se tienen los siguientes comandos:

Configuración I/O		
Nombre Pin	Valor	Descripción
Dn (con n entre 0 y 8)	0	Deshabilitado
	1	(se vera luego)
	2	ADC (excepto 7 y 8)
	3	Entrada Digital
	4	Salida Digital 'L' (LOW)
	5	Salida Digital 'H' (HIGH)
Pn (con n entre 0 y 1)	0	Deshabilitado
	1	RSSI
	2	PWM
IR	0x0 – 0xFFFF (x 1 ms)	Ajusta la tasa de muestreo de los conversores ADC y digitales DIO. Si IR = 0x0A (10 en decimal), entonces la tasa será de 10 ms o 100 Hz.
IT	0x0 – 0xFF	Número de muestras que guarda en buffer antes de enviarlas. Buffer del módulo es cercano a 93 bytes de datos de muestra. Como los conversores son de 10 bit, cada muestra tiene 2 bytes, por lo que deja un máximo de 46 muestras (IT=0x2C) para el tamaño del buffer.
IA	0x0 – 0xFFFFFFFFFFFFFFFF	Indica la dirección del módulo al cual se le aceptarán los
		paquetes para cambiar las salidas I/O. Acepta direcciones de 16 y 64 bits. Si IA=0xFFFFFFFFFFFFFFFF no se aceptarán paquetes de ningún módulo. Ésta, es la configuración por defecto. Si IA=0xFFFF, se aceptarán cambios de todos los paquetes
IU	0	Este comando habilita o no la salida por UART de los paquetes recibidos.
	1	Deshabilitado Habilitado.

Tabla II.3 Comandos para Cable Virtual.

Para su uso se debe escribir AT, más el comando y luego el valor que se desea asignar.

Por ejemplo si se quiere configurar el pin 19 como conversor analógico, se debe ingresar **ATD12**, donde D1 indica el conversor AD1 ubicado en el pin 19, y el 2 indica que se utilizará como conversor analógico según la tabla anterior. Si se desea configurar el muestreo de los conversores cada 1000 ms, se debe ingresar **ATIR3E8**, donde IR indica que se ajustará el parámetro de muestreo y 3E8 corresponde al valor hexadecimal para 1000 (0x3E8). Si se desea consultar el valor de cierto parámetro, se ingresa el comando sin ningún valor, por ejemplo, si se desea consultar la tasa actual de muestreo de los conversores se ingresa **ATIR**, a lo cual entregará como respuesta, si lo hacemos luego del ejemplo anterior, el valor 3E8. Se muestra la siguiente tabla con una configuración de ejemplo:

MÓDULO 1		MÓDULO 2	
MY	0x1234	MY	0x5678
DL	0x5678	DL	0x1234
D0	2	P0	2
D1	2	P1	2
D2	3	D2	4
IR	0x14	IU	1
IT	5	IA	0x1234 (O 0xFFFF)

Tabla II.4 Configuración para Cable Virtual.

Se observa que se parte con un direccionamiento de 16 bit, como el mostrado anteriormente. Luego en el módulo 1, se configuran 3 entradas, la entrada D0 y D1 como analógicas y la D2 como digital. En el módulo 2, se configura la salida P0 y P1 como salida PWM y la salida D2 como salida digital. Luego se configura una tasa de 20 ms (IR=0x14, 20 en decimal) donde se guardan 5 muestras (IT=5) por entrada antes de

enviar. Como se muestrea cada 20 ms y se almacenan 5 muestras, éstas serán enviadas cada 100 ms. Luego cada muestra ADC pesa 2 bytes y la digital 1 byte, por lo que son 5 bytes, cada 20 ms, lo que hace que se envíen 25 bytes (5 muestras cada 20 ms) cada 100 ms. Además se deben contar los bytes de la cabecera de los paquetes, los que son 12 bytes, por lo que en total se enviarán 37 bytes (25 datos + 12 cabecera del frame) cada 100 ms.

Con el comando IU, es posible observar la salida de los cables virtuales que llegan al módulo. Si IU=1 (ATIU1), se habilita la salida por el pin DOUT de los frames con los datos de todos los cables virtuales que llegan al módulo. Los datos vienen contenidos en un frame, no en formato ASCII, sino en hexadecimal. El formato es el mismo que el de los frame de la API y es el siguiente:

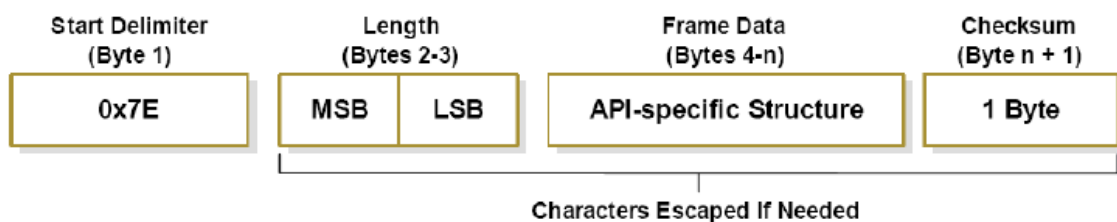


Figura II.20 Formato del Frame para Cable Virtual.

2.19. Conexión NonBeacon c/Coordinador.

Es básicamente lo mismo que una red punto-multipunto, con la diferencia de que existe un módulo central que posee ciertas propiedades y características que le permiten administrar la red. En esta red, el módulo central es llamado **Coordinador**, mientras que el resto de módulos son llamados **Dispositivos Terminales** (End Device). Un mismo módulo Xbee puede ser configurado para funcionar como Coordinador o como

Dispositivo Terminal. Para configurar esta red, todos los módulos deben tener el mismo canal (ATCH) y la misma PAN (ATID). El módulo Coordinador se configura como ATCE=1 (ATCE1), mientras que todos los demás, los cuales serán llamados Dispositivos Terminales, se configuran como ATCE=0 (ATCE0).

En este tipo de red, los Coordinadores pueden ser usados para usar transmisiones directas o indirectas. En las directas, la información es enviada de inmediato, mientras que en la indirecta, la información es guardada un tiempo dado por el parámetro SP (Cyclic Sleep Period). Si SP=0, la transmisión es directa. Si SP está en un rango entre 1 y 0x68B0 (x10 milisegundos), es el tiempo que espera antes de enviar.

Para este tipo de configuraciones NonBeacon con Coordinador, se requiere crear una relación llamada Asociación. Ésta mantiene un control del Coordinador sobre los Dispositivos Terminales. Este tipo de configuración con un Coordinador se utiliza cuando se requiere una unidad central para enviar mensajes a varios módulos, o juntar información proveniente de varios Dispositivos Terminales, asignar canales o ID de redes PAN.

Una red de datos RF consistente de un Coordinador y uno o varios Dispositivos Terminales, forman lo que se llama una PAN (Personal Area Network). Cada dispositivo en una PAN tiene un identificador llamado ID (ATID), el cual debe ser el mismo para todos los módulos de la misma PAN.

2.20. Comandos más importantes.

A continuación se muestra una tabla resumen con los comandos AT para el módulo Xbee más importantes. El contenido se muestra con el rango permitido por el comando, una descripción, y las configuraciones para cada valor del parámetro. Para utilizar el

programa se debe ingresar AT y luego, sin espacios, el comando a configurar y el valor del parámetro en caso de que se quiera ajustar, o sin nada en caso de que se quiera consultar el valor de ese parámetro.

Comando	AT	Rango	Descripción
	A1	0 - 0x0F	Describe el modo de Asociación de un módulo. utilizado como Dispositivo Terminal (CE=0). Defecto=0.
	A2	0 - 0x0F	Describe el modo de Asociación de un módulo utilizado como Coordinador (CE=1). Defecto=0.
	AC	-	Aplica los cambios realizados explícitamente en l configuración.
	AP	0 - 0x02	Habilita el modo de operación API. Defecto=0. 0 Modo API Deshabilitado. 1 Modo API habilitado. 2 Modo API habilitado con carácter de escape. Ajusta la tasa de transmisión entre el módulo y su cliente conectado a través de la interfaz serial. Para valores no-estándar revisar el manual. Defecto=3.
	BD	0 - 0x07	0 1200 1 2400 2 4800 3 9600 4 19200 5 38400 6 57600 7 115200
	CC	0 - 0xFF	Establece el carácter de secuencia a ser usado entre tiempos de esperas para entrar al modo de comandos. Defecto=0x2B (carácter ASCII +)
	CH	0x0B - 0x1A	Establece el canal por el cual se realiza la conexión RF entre módulos. Verificar Tabla 5-1 Frecuencia de Canales para configurar este parámetro. Defecto=0x0C.
	CE	0 - 1	Indica el comportamiento del módulo. Defecto=0. 0 Dispositivo Terminal. 1 Coordinador.
	CN	-	Sale del modo de Comando.
	D0-D4	0 - 5	Ajusta la configuración de los pines I/O.

		0 Deshabilitado.
		1 --
		2 ADC.
		3 Entrada Digital.
		4 Salida Digital LOW.
		5 Salida Digital HIGH.
D5	0 - 5	Mismas funciones que D0 - D4, exceptuando lo siguiente:
D6	0 - 5	1 Indicador de Asociación. Mismas funciones que D0 - D4, exceptuando lo siguiente:
		1 Control de Flujo RTS.
		2 No tiene conversor ADC.
D7	0 - 5	Mismas funciones que D0 - D4, exceptuando lo siguiente:
D8	Sólo 0 y 3.	1 Control de Flujo CTS. 2 No tiene conversor ADC. Ajusta la configuración del pin DI-8 (pin 9).
		0 Deshabilitado.
		3 Entrada Digital.
DB	0x17 - 0x5C (x-1dBm)	Lee la potencia de la señal del módulo del cual provino el último paquete RF recibido.
DL	0 - 0xFFFFFFFF	Ajusta los 32 bits menos significativos para direccionamiento. Defecto = 0.
DH	0 - 0xFFFFFFFF	Ajusta los 32 bits más significativos para direccionamiento. Defecto = 0.
GT	2 - 0x0CE4 (x 1 ms)	Tiempo de espera antes y despues de ingresar el carácter de secuencia para entrar al modo de comandos. Defecto = 0x3E8.
IA	0 - 0xFFFFFFFF	Utilizado para crear el Cable Virtual. Indica la dirección del módulo de origen de los datos. Defecto= 0xFFFFFFFF (no permite el recibo de ningún paquete para cambiar las salidas.)
ID	0 - 0xFFFF	Ajusta la dirección PAN del módulo. Defecto = 0x3332
IR	0 - 0xFFFF (x1 ms)	Ajusta la tasa de muestreo de los pines I/O. Defecto = 0.
IS	1 - 0xFF	Forza al módulo a leer todos sus pines I/O. Si AP=0, el resultado se retorna del siguiente modo: - Número de Muestras. - Máscara de Canal. - Datos DIO.
		- Datos conversores ADC (se repite por cada conversor habilitado).
IT	1 - 0xFF	Número de muestras DIO y ADC que se deben

		esperar, antes de transmitir. Defecto = 1.
IO	8 bits	Ajusta los niveles de las salidas digitales. Cada bits representa el nivel de los pines I/O configurados como salida .
M0 - M1	0 - 0x03FF	Ajusta el ciclo de trabajo de la salida PWM0 y PWM1. Si Mn=0 (0% PWM), Mn=0x01FF (50% PWM) y si Mn=0x03FF (100% PWM). Defecto=0.
MY	0 - 0xFFFF	Configura la dirección de 16 bits para el módulo. Si My=0xFFFF o 0xFFFE, se habilita el modo de direccionamiento de 64 bit. Defecto = 0.
NB	0 - 4	Ajusta la Paridad para la comunicación serial UART del módulo. Defecto =0 .
		0 8 bit sin paridad o 7 bit con cualquier paridad.
		1 8 bit even.
		2 8 bit odd.
		3 8 bit mark.
		4 8 bit space.
ND	-	Reporta todos los dispositivos que se encuentren en el mismo canal y en la misma PAN que el módulo. El formato de respuesta es el siguiente cuando se encuentra en el modo Transparente.
		- MY (dirección de 16 bit)
		- SH (Serial Number High)
		- SL (Serial Nmuber Low).
		- DB (Fuerza de la señal proveniente de este módulo)
		- NI (Identificador del Nodo)
NI	String de 20 caracteres ASCII	Define con un String el nodo o módulo .
P0 - P1	0 - 2	Configura el pin PWM0 y PWM1. Defecto P0 =1, Defecto P1=0.
		0 Deshabilitado.
		1 RSSI.
		2 PWM habilitado.
RE	-	Restaura los valores de los parámetros a los valores por defecto que vienen de fábrica.
SM	0 - 6	Configura el modo de operación SLEEP. Defecto = 0.
		0 Deshabilitado.
		1 Pin de Hibernado.
		2 Pin Doze.
		3 (reservado)
		4 Remoto Cyclic SLEEP.
		5 Remoto Cyclic SLEEP (con pin Wake-up).
		6 SLEEP Cordinador.
SL	0 - 0xFFFFFFFF	Entrega los 32 bit menos significativos del Número Serial del módulo.

SH	0 - 0xFFFFFFFF	Entrega los 32 bit más significativos del Número Serial del módulo.
SP	1 - 0x68B0 (x10 ms)	Ajusta el tiempo de duración en que un módulo duerme o se mantiene en el modo SLEEP. Una vez terminado el período, busca por data entrante, si no hay nada vuelve a dormir y espera por un nuevo ciclo.
ST	1 - 0xFFFF (x1 ms)	Ajusta el tiempo de inactividad (datos ni recibidos ni enviados ya sea por RF o serial) antes de que el módulo ingrese al modo SLEEP. Defecto = 0x1388. Tiempo de espera de apagado para los Cables Virtuales. Si luego de este tiempo no hay cambios en alguna salida I/O, éstas cambian a su valor por defecto. Defecto = 0xFF.
T0 - T7	0 - 0xFF (x100 ms)	
IU	0 - 1	Habilita o no la salida I/O UART. 0 Deshabilitado. Paquetes RF recibidos no serán enviados por UART. 1 Habilitado. Paquetes RF recibidos serán enviados por UART.
VL	-	Entrega la versión del Firmware de forma Verbal. La respuesta entrega fecha de compilación de la aplicación, MAC, PHY y versión del bootloader y sus fechas de compilación.
VR	0 - 0xFFFF	Indica cual versión de firmware se encuentra actualmente en el módulo.
WR	-	Guarda en la memoria no-volátil del módulo, todos los valores de los parámetros.

. Tabla II.5 Resumen de Comandos.

En la figura se muestra la especificación de cada uno de los pines del modulo Xbee

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

Figura II.21 Diagrama de Pines del modulo Xbee

También tenemos a continuación el detalle eléctrico del modulo Xbee

DC Characteristics (VCC = 2.8 - 3.4 VDC)

Symbol	Characteristic	Condition	Min	Typical	Max	Unit
V _{IL}	Input Low Voltage	All Digital Inputs	-	-	0.35 * VCC	V
V _{IH}	Input High Voltage	All Digital Inputs	0.7 * VCC	-	-	V
V _{OL}	Output Low Voltage	I _{OL} = 2 mA, VCC >= 2.7 V	-	-	0.5	V
V _{OH}	Output High Voltage	I _{OH} = -2 mA, VCC >= 2.7 V	VCC - 0.5	-	-	V
I _{IN}	Input Leakage Current	V _{IN} = VCC or GND, all inputs, per pin	-	0.025	1	µA
I _{OZ}	High Impedance Leakage Current	V _{IN} = VCC or GND, all I/O High-Z, per pin	-	0.025	1	µA
TX	Transmit Current	VCC = 3.3 V	-	45 (XBee) 215 (PRO)	-	mA
RX	Receive Current	VCC = 3.3 V	-	50 (XBee) 55 (PRO)	-	mA
PWR-DWN	Power-down Current	SM parameter = 1	-	< 10	-	µA

Figura II.22 Detalles eléctricos del modulo del Xbee

ADC Characteristics (Operating)

Symbol	Characteristic	Condition	Min	Typical	Max	Unit
V _{REFH}	VREF - Analog-to-Digital converter reference range		2.08	-	V _{DDAD}	V
I _{REF}	VREF - Reference Supply Current	Enabled	-	200	-	µA
		Disabled or Sleep Mode	-	< 0.01	0.02	µA
V _{INDC}	Analog Input Voltage ¹		V _{SSAD} - 0.3	-	V _{DDAD} + 0.3	V

1. Maximum electrical operating range, not valid conversion range.

Figura II.23 Características de operación ADC del modulo del Xbee

ADC Timing/Performance Characteristics¹

Symbol	Characteristic	Condition	Min	Typical	Max	Unit
R _{AS}	Source Impedance at Input ²		-	-	10	kΩ
V _{A IN}	Analog Input Voltage ³		V _{REFL}	-	V _{REFH}	V
RES	Ideal Resolution (1 LSB) ⁴	2.08V ≤ V _{DDAD} ≤ 3.6V	2.031	-	3.516	mV
DNL	Differential Non-linearity ⁵		-	±0.5	±1.0	LSB
INL	Integral Non-linearity ⁶		-	±0.5	±1.0	LSB
E _{ZS}	Zero-scale Error ⁷		-	±0.4	±1.0	LSB
F _{FS}	Full-scale Error ⁸		-	±0.4	±1.0	LSB
E _{IL}	Input Leakage Error ⁹		-	±0.05	±5.0	LSB
E _{TU}	Total Unadjusted Error ¹⁰		-	±1.1	±2.5	LSB

Figura II.24 Detalles Técnicos del modulo Xbee

Detalles	XBee 802.15.4 (Series 1)
Funciones	
Tasa de datos	254 Kbps
Rango recinto interior	30 m
Rango línea de visión	100 m
Potencia de Transmisión	1 mW (+0 dBm)
Sensibilidad del receptor (1% PER⁸)	-92 dBm
Características	
Interfaz de Datos Serial	3.3V CMOS UART
Métodos de Configuración	Comandos AT o API, local o sobre el aire.
Banda de frecuencia	2.4 GHz
Inmunidad a la interferencia	DSSS (Direct Sequence Spread Spread Spectrum)
Tasa de Comunicación Serial	1200 bps a 250 Kbps
Conversores ADC	Conversores de 10 bit (hasta 7 disponibles). ⁹
I/O Digitales	Hasta 8 disponibles.
PWM (Pulse Width Modulation)	Hasta 2 disponibles.
Opciones de Antena	Chip, Wire Chip, U.FL y RPSMA.
Redes y Seguridad.	
Encriptación	128 bits
Entrega confiable de paquetes.	Retries/Acknowledgments (Reintentos/ aviso de recibo de paquetes)
ID's y canales.	PAN ID (personal Area Network), 64-bit IEEE MAC, 16 canales.
Requerimientos de poder	
Voltaje de Alimentación	2.8 a 3.4 Vdc
Corriente al transmitir.	45 mA @ 3.3 Vdc
Corriente al recibir.	50 mA @ 3.3 Vdc
Corriente de apagado.	<10uA @ 25 °C
Regulaciones aprobadas.	
FCC (USA)	OUR-XBee
IC (Canadá)	4214A-XBee
ETSI (Europa)	Si
C-TICK Australia	Si
Telec (Japón)	Si

Tabla II.6 Especificaciones Técnicas.

2.21. Detalles Físicos.

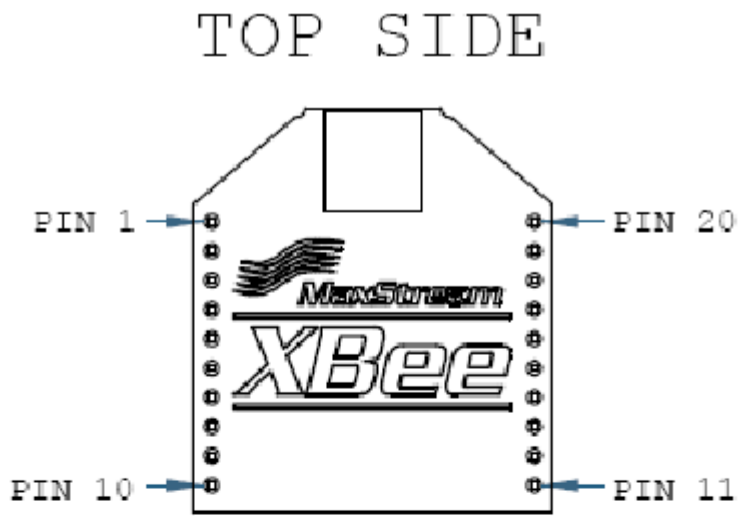


Figura 9-3 Vista superior.

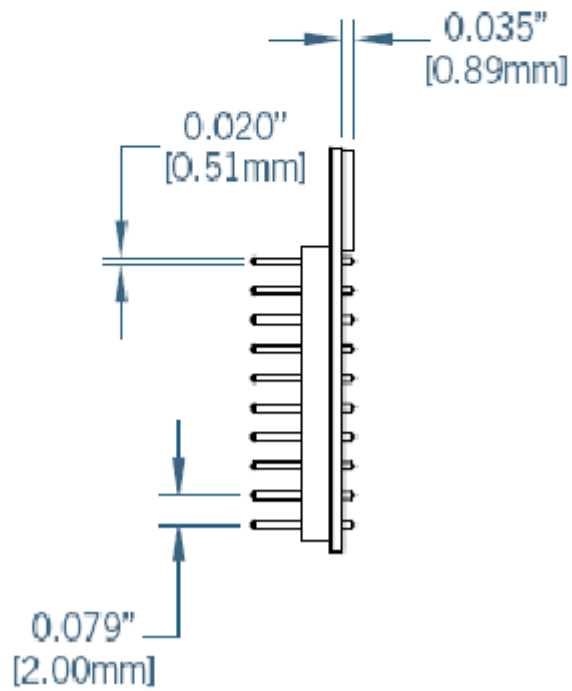


Figura II.25 Vista Superior y lateral

CAPÍTULO III

Dispositivos utilizados en el proyecto

3. PIC18F2550

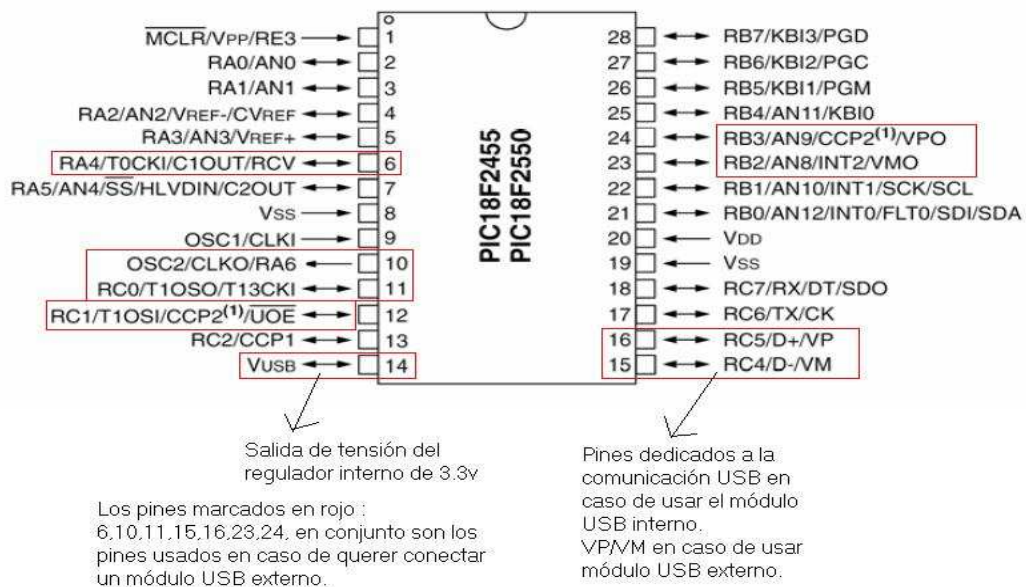


Figura III.1 Pines del Pic18f2550

La familia de dispositivos PIC18FX455/X550 contiene una interfaz serie compatible con el SIE (serial interface engine o máquina con comunicación serie en español) USB

“full-speed” (2.0) y “de poca velocidad” (1.0) que permite la comunicación rápida entre cualquier dispositivo USB y el microcontrolador PIC®.

El SIE puede interconectarse directamente al USB, utilizando el transmisor - receptor interno, o puede conectarse a través un transmisor-receptor externo. El PIC tiene un regulador interno de 3,3V para alimentar el transmisor-receptor interno en aplicaciones de 5V.

Se han incluido algunas características especiales en el hardware para mejorar el funcionamiento. Se proporciona memoria de puerto dual en la memoria de datos del dispositivo (RAM del USB) para tener acceso directo a la memoria desde el núcleo del microcontrolador y desde el SIE. También se proporcionan unos buffer para que el programador elija libremente el uso de la memoria dentro del espacio de la RAM del USB. Existe un puerto paralelo para transmitir grandes cantidades de datos (SSP), para esto se ha proporcionado la ayuda de transferencia ininterrumpida de volúmenes de datos grandes, por ejemplo datos síncronos, a los buffer de memoria externos.

Ahora veremos en detalle como se comunica la SIE con el exterior y las diferentes opciones a nivel conexionado del USB con el exterior:

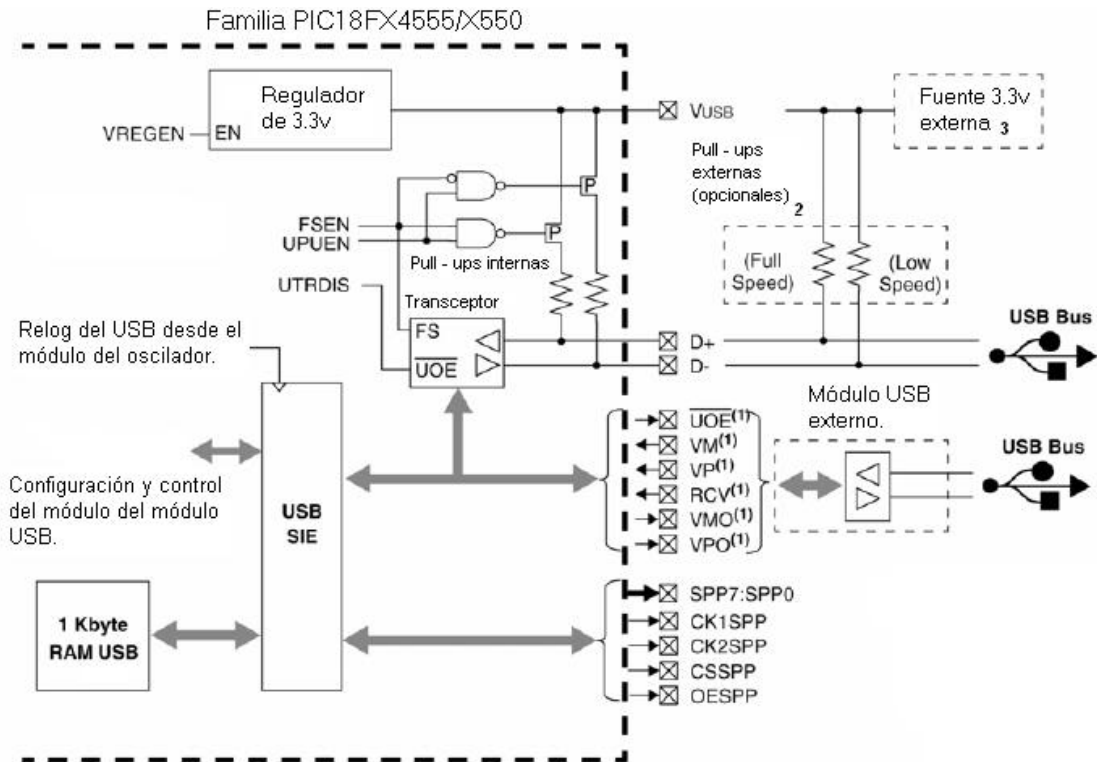


Figura III.2 Nivel de conexionado del USB con el exterior:

Nota 1: Las pull - ups externas son opcionales , pero si queremos usarlas hay que desactivar las propias resistencias de pull - up internas que tiene el microcontrolador. La activación o desactivación de estas resistencias se hace mediante el bit **UPUEN**...que se verá más adelante.

Nota 2: El regulador de voltaje de 3.3v, nos provee de tensión al módulo USB en aplicaciones de 3.3v. En caso de tener una fuente externa de 3.3v conectada a Vusb, tenemos que desactivar el módulo interno del regulador.

Vemos en la imagen los diferentes pines que según configuración de la SIE , podemos usar un transceptor externo.

Algo importante que tenemos que ver es el puente bidireccional entre la SIE y el SSP este puente nos va a permitir el transporte a alta velocidad de grandes flujos de datos en modo sincrónico.

la siguiente imagen se muestra el puente bidireccional:

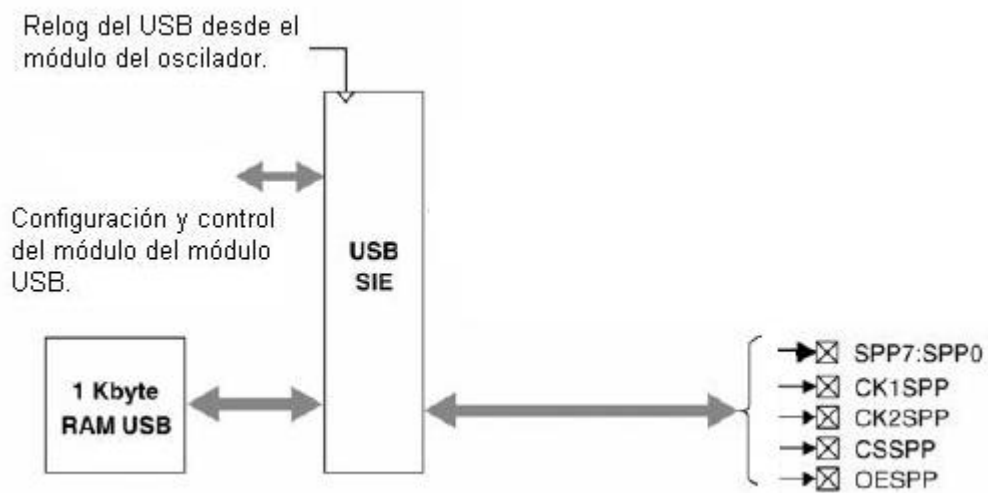


Figura III.3 Puente bidireccional

3.1. Estado y control del módulo USB:

En total hay 22 registros para manejar las transacciones del USB. Los registros son:

- Registro de control del USB (**UCON**)
- Registro de configuración del USB (**UCFG**)
- Registro de estado de la transferencia del USB (**USTAT**)
- Registro de dirección de dispositivo USB (**UADDR**)
- Registros del número del frame (**UFRMH: UFRML**)
- Registros activadores de los Endpoints de 0 a 15 (**UEPN**)

El primer registro que estudiaremos será el UCON (registro de control del USB):

En la siguiente imagen veremos como está constituido:

U-0	R/W-0	R-x	R/C-0	R/W-0	R/W-0	R/W-0	U-0
—	PPBRST	SE0	PKTDIS	USBEN	RESUME	SUSPND	—
bit 7							bit 0

Figura III.4 Registro UCON

Este registro de control contiene los bits necesarios para determinar el comportamiento del bus durante las transferencias.

7	6	5	4	3	2	1	0
—	PPBRST	SE0	PKTDIS	USBEN	RESUME	SUSPND	—
U0	R/W	R-X	R/C-0	R/W-0	R/W-0	R/W-0	U0

Tabla III.1 Bit de registro UCON

El registro contiene los bits que gobiernan lo siguiente

- Permiso del periférico principal del USB
- Reset de los punteros tipo ping-pong
- Control del modo al suspender
- Desactivar la transferencia de paquetes

El significado de los diferentes bits se muestra a continuación:

BIT7: Bit no implementado se lee como "0"

BIT6: El bit **PPBRST**, se encarga de controlar el estado del reset, cuando se utiliza el modo Doble - Buffering o modo Ping - Pong. Cuando se activa el bit **PPBRST**, todos

los buffers Ping - Pong se fijan a los buffers intermedios. El bit **PPBRST**, tiene que borrarse por firmware. Este bit se ignora en los modos de buffer que no usen el buffer ping-pong.

BIT5: El bit **SE0**, se utiliza para indicar el estado del bus, si sólo se manda un cero. Cuando se permite el módulo USB, este bit debe supervisarse para determinar si las líneas de datos han salido de una condición de single-ended cero (sólo se manda cero). Esto ayuda a distinguir el estado de ciclo inicial de la señal de reset del USB.

BIT4: El bit **PKTDIS**, es un flag que indica si el SIE ha inhabilitado la transmisión y la recepción de paquetes. Este bit activa el SIE cuando recibe un SETUP para permitir su procesamiento. Este bit no lo puede activar el microcontrolador, sólo borrar; al despejarlo el SIE continúa la transmisión y/o recepción. Cualquier acontecimiento pendiente dentro del buffer intermedio todavía estará disponible, indicado dentro del registro USTAT en el buffer FIFO.

BIT3: El bit **USBEN**, la operación total del módulo USB se controla con este bit. Activar este bit setea el módulo y resetea todos los bits PPBI en el Buffer a '0'. Este bit también activa el regulador de tensión del chip y conecta las resistencias pull-up, si se permiten. Así, este bit puede utilizarse como una unión / separación al USB. Aunque se ignoran todos los estados y bits de control si este bit está borrado, el módulo necesita preconfigurarse antes de activar este bit.

BIT2: El bit **RESUME**, permite al periférico realizar un reinicio ejecutando la señal resume. Para generar un reinicio válido, por firmware se debe activar el RESUME durante 10ms y entonces borrar el bit.

BIT1: El bit **SUSPEND**, coloca el módulo y soporte del circuito (es decir, regulador de tensión) en un modo de baja potencia. El reloj de entrada al SIE se desactiva. Este bit

debe activarse por software dentro de la respuesta a una interrupción IDLEIF. Debe borrarse por firmware después de observar una interrupción ACTVIF. Cuando este bit está activo, sigue estando el dispositivo unido al bus pero las salidas del transmisor-receptor permanecen en reposo. La tensión en el pin VUSB puede variar dependiendo del valor de este bit. Activar este bit antes de un IDLEIF dará lugar a comportamiento imprevisible del bus.

Nota: Cuando esté en el modo de suspensión, el bus del dispositivo USB se limita al 500µA de corriente. Ésta es la corriente completa cedida por el dispositivo PIC y su circuito de soporte. Hay que tener cuidado de ceder la corriente mínima cuando el dispositivo entre en el modo de suspensión.

BIT 0: Bit no implementado se lee como "0"

3.2. Registro de configuración del USB:

Antes de comunicarse con el USB, se tiene que configurar el módulo del hardware interno y/o externo. La mayor parte de la configuración se realiza con el registro **UCFG**. El regulador de tensión del USB se maneja con la configuración de los registros.

El registro **UFCG** contiene la mayor parte de los bits que dirigen el comportamiento del módulo USB. Esto incluyen:

Velocidad del bus (“Alta velocidad” vs “Baja velocidad”)

Permiso de las resistencias pull-up del chip

Permiso del transmisor del chip

Uso del buffer ping-pong

El registro **UCFG** también contiene dos bits que ayudan a probar el módulo, eliminando errores y certificaciones del USB. La salida de control de estos bits permite el monitor de estado y generación de patrones “de ojo”.

R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
UTEYE	UOEMON ⁽¹⁾	—	UPUEN ^(2,3)	UTRDIS ⁽²⁾	FSEN ⁽²⁾	PPB1	PPB0
bit 7							bit 0

Figura III.5 Registro UCFG

BIT 7: UTEYE: Bit de permiso del test patrón “de ojo” del USB.

1 = prueba del patrón “de ojo” permitido. 0 = prueba del patrón “de ojo” inhabilitado.

BIT 6: UOEMON: Bit de permiso del monitor OE del USB.

1 = señal del UOE* activa; indica los intervalos durante los cuales las líneas D+/D- están conduciendo. 0 = señales del UOE* inactivas.

BIT 4: UPUEN: Bit de permiso del pull-up del chip. 1 = Pull-up del chip permitido (pull-up en D+ con FSEN=1 o en D- con FSEN=0).

0 = Pull-up del chip desactivado.

BIT 3: UTRDIS: Bit inhabilitador del transmisor del chip 1 = Transmisor del chip inhabilitado; la interfaz digital del transmisor permitida. 0 = transmisor del chip activo.

BIT 2: FSEN: Bit de permiso del Full-Speed. 1 = dispositivo Full-speed: controla los flancos del transmisor; requiere un reloj de 48MHz. 0 = dispositivo de poca velocidad:

controla los flancos del transmisor; requiere un reloj de 6MHz. **BIT 1-0 PPB1:PPB0:**

Bits de configuración de los buffer ping-pong. 11 = buffers ping-pong permitidos en los End-Points 1 a 15. 10 = buffers ping-pong permitidos en todos los End-Points.

01 = buffer ping-pong permitido en los End-Points de salida 0.

00 = buffers ping-pong inhabilitados.

Nota 1: 1: Si se activa UTRDIS, la señal UOE* estará activa independiente del ajuste del bit UOEMON. 2: Los bits UPUEN, UTRDIS y FSEN no deben cambiarse mientras el módulo USB esté permitido. Estos valores se deben preconfigurar antes de permitir el módulo.

3: Este bit solamente es válido cuando el transmisor del chip está activo (UTRDIS = 0); si no, se ignora.

Nota 2: La velocidad del USB, la transmisión y las pull-up deben configurarse solamente durante la fase de activación del módulo. No se recomienda cambiar estos ajustes mientras que el módulo esté funcionando. Viendo un poco el registro UCFG, en donde se encuentra el bit UTEYE, que es un bit de control de estado del USB que vigila los errores y es un generador de patrones. Luego tenemos al bit UOEMON. Este bit nos muestra cuando las líneas D+ y D- están conduciendo y lo que estos valores representan según el siguiente cuadro

D +	D -	Estado del Bus
0	0	Single-Ended Cero
0	1	Baja velocidad
1	0	Alta velocidad
1	1	Error

Tabla III.2 líneas D+ y D-

Ahora tenemos 2 bit que trabajan de forma conjunta FSEN y UPUEN, según sea su configuración activarán o desactivarán las Resistencias Pull - Up internas del módulo USB o configuraran el módulo USB para trabajar a full - speed o slow - speed. Veamos el circuito lógico que realiza estas acciones y una pequeña tabla donde se muestran los siguientes posibles configuraciones

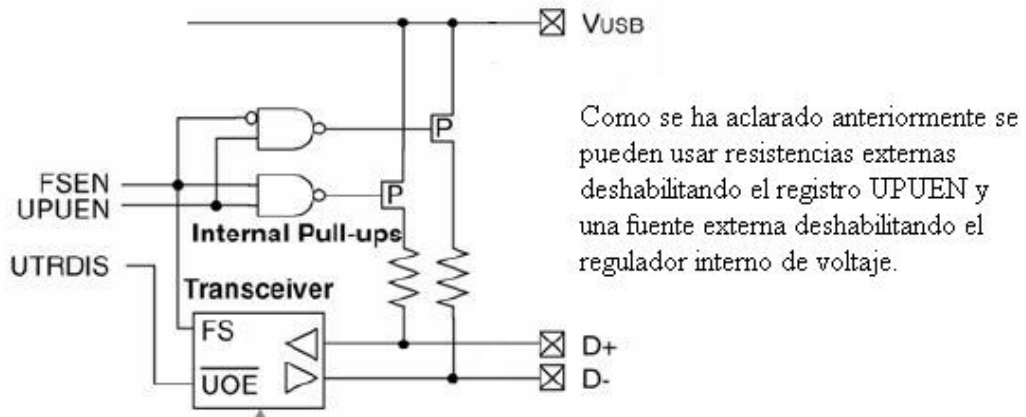


Figura III.6 Registro UPUEN

Bit FSEN	Bit UPUEN	Velocidad FS (Full speed) o LS (low speed).	Resistencias de Pull - Up internas.
0	0	Ninguna	Desconectadas
1	0	Ninguna	Desconectadas
0	1	LS	R en D -
1	1	FS	R en D +

Tabla III.3 Configuraciones registro UPUEN

Luego tenemos **UTRDIS**, que habilita o deshabilita el transmisor del chip, en caso de que el bit sea 0 , activamos el transceptor y los pines C4 y C5 son usados para conectar el dispositivo USB. En caso de que el bit sea seteado a 1 , el transceptor es desactivado y las líneas C4 y C5 quedan para poder ser usadas como entradas digitales únicamente.

Por último tenemos los bits de configuración **BIT 1-0 PPB1:PPB0**: que establecen la configuración del modo Ping - Pong.

3.3. Características principales del pic 18f2550

Posee 32Kbyte de memoria flash y de EEPROM de 256kByte

24 Puerto de entrada salida

100,000 Borrar/Escribir Cycle Ampliada Memoria típica de programa

1,000,000 Borrar/Escribir Ciclos de datos EEPROM la memoria típica

Flash/Data EEPROM Retención mayor a 40 años

Prioridad para los niveles de interrupción

8 x 8 Simple ciclo Hardware multiplicador

Extendido Watchdog Timer (WDT):

- Programable periodo desde 41 ms a 131s Programable código de protección
- Depurar el circuito (ICD) vía dos pines

Rango amplio de operación (2.0V z 5.5V)

Alta corriente de suministro 25 mA/25 mA

Tres Interrupciones externas

Cuatro módulos timer (Timer0 to Timer3)

Dos módulos Capture/Compare/PWM (CCP)

- Capture es 16-bit, máximo. Resolución 5.2 ns (TCY/16)
- Compare es 16-bit, máximo. Resolución 83.3 ns (TCY)
- PWM output: PWM resolución is 1 to 10-bit

Mejorado módulo Capture/Compare/PWM (ECCP)

- Múltiples modos de salida
- Seleccionar polaridad
- Programable tempo inactivo
- Auto apagado del sistema y auto reset

Mejorado modulo USART

- soporta bus LIN

10-bit, a 13-canales de conversor Analógico a Digital Modulo convertidor (A/D) con tiempo de adquisición programable

Doble Comparador Analógico con entradas multiplexadas

Los dispositivos de la familia PIC18FX455/X550 contienen velocidad mínima y máxima compatible con el USB una interface de procesamiento Serial que permite la rápida comunicación entre cualquier host USB y el microcontrolador PIC. La interface de procesamiento serial puede se interface directamente a el USB, utilizando el transmisor-receptor interno, o se puede conectar atreves de externo. Un regulador interno de 3.3v esta también disponible para aplicaciones en 5v.

Algunas características especiales del hardware estuvieron incluidas para mejorar el rendimiento. Doble puerto de memoria en el espacio de memoria de datos del dispositivo espacio (USB RAM) Se ha facilitado a compartir entre el acceso directo a memoria y el núcleo del Microcontrolador y la SIE. Buffer descritos son también proporcionan, permitiendo a usuarios libremente programar puntos de memoria usada con el USB RAM.

3.4. Universal Serial Bus

USB V2.0 Completamente

Velocidad baja (1.5 Mb/s) y Máxima velocidad (12 Mb/s)

Soporta alrededor de 32 productos (16 bidireccional)

1-Kbyte Dual Acceso a la RAM desde USB

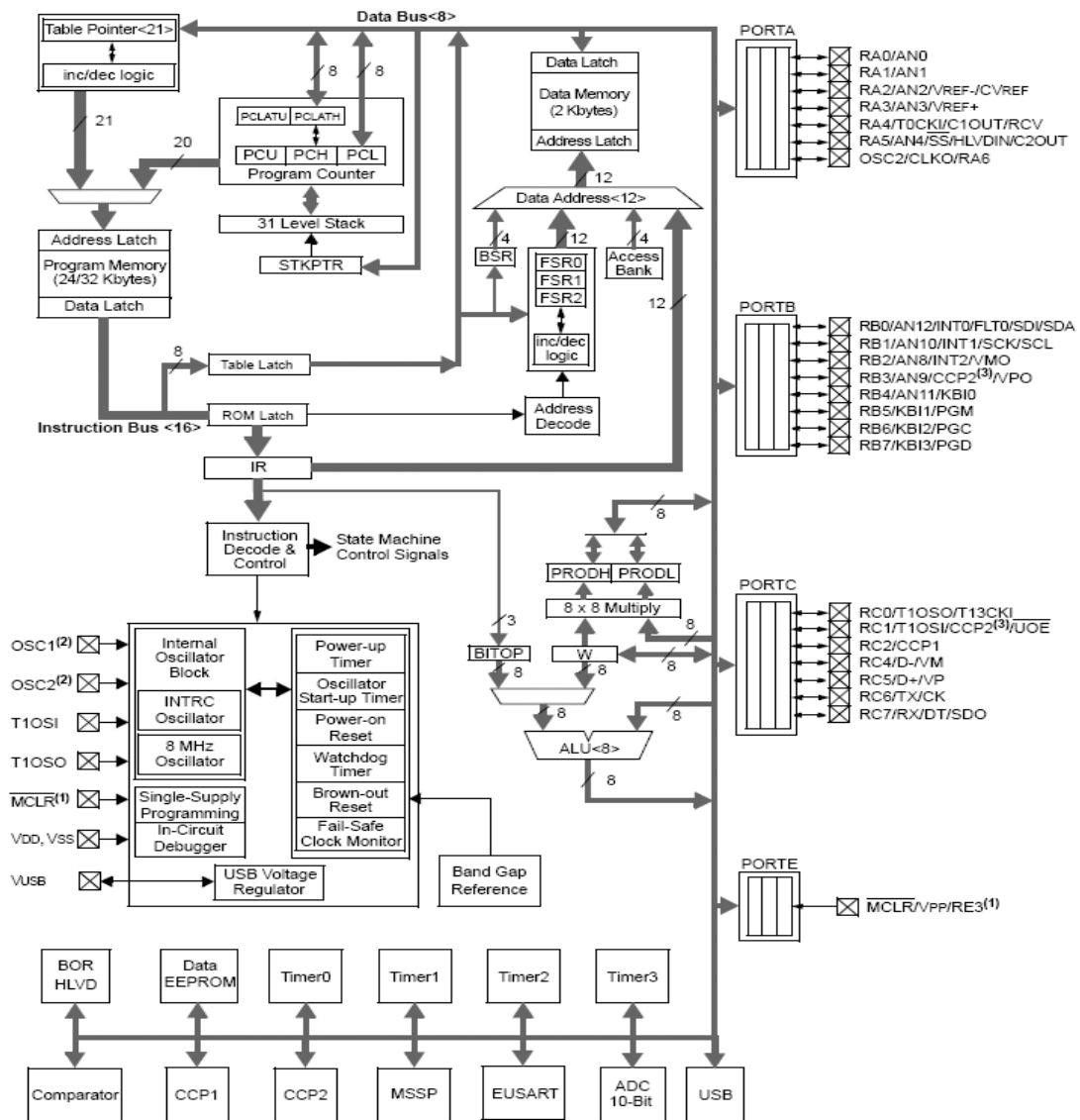


Figura III.7 Diagrama interno del PIC 18F2550

Address	Name	Address	Name	Address	Name	Address	Name	Address	Name
FFFh	TOSU	FDfH	INDF2 ⁽¹⁾	FBFh	CCPR1H	F9Fh	IPR1	F7Fh	UEP15
FFEh	TOSH	FDEh	POSTINC2 ⁽¹⁾	FBEh	CCPR1L	F9Eh	PIR1	F7Eh	UEP14
FFDh	TOSL	FDDh	POSTDEC2 ⁽¹⁾	FBDh	CCP1CON	F9Dh	PIE1	F7Dh	UEP13
FFCh	STKPTR	FDCh	PREINC2 ⁽¹⁾	FBCh	CCPR2H	F9Ch	— ⁽²⁾	F7Ch	UEP12
FFBh	PCLATU	FDBh	PLUSW2 ⁽¹⁾	FBHh	CCPR2L	F9Bh	OSCTUNE	F7Bh	UEP11
FFAh	PCLATH	FDAh	FSR2H	FBAh	CCP2CON	F9Ah	— ⁽²⁾	F7Ah	UEP10
FF9h	PCL	FD9h	FSR2L	FB9h	— ⁽²⁾	F99h	— ⁽²⁾	F79h	UEP9
FF8h	TBLPTRU	FD8h	STATUS	FB8h	BAUDCON	F98h	— ⁽²⁾	F78h	UEP8
FF7h	TBLPTRH	FD7h	TMR0H	FB7h	ECCP1DEL	F97h	— ⁽²⁾	F77h	UEP7
FF6h	TBLPTRL	FD6h	TMR0L	FB6h	ECCP1AS	F96h	TRISE ⁽³⁾	F76h	UEP6
FF5h	TABLAT	FD5h	T0CON	FB5h	CVRCON	F95h	TRISD ⁽³⁾	F75h	UEP5
FF4h	PRODH	FD4h	— ⁽²⁾	FB4h	CMCON	F94h	TRISC	F74h	UEP4
FF3h	PRODL	FD3h	OSCCON	FB3h	TMR3H	F93h	TRISB	F73h	UEP3
FF2h	INTCON	FD2h	HLVDCON	FB2h	TMR3L	F92h	TRISA	F72h	UEP2
FF1h	INTCON2	FD1h	WDTCON	FB1h	T3CON	F91h	— ⁽²⁾	F71h	UEP1
FF0h	INTCON3	FD0h	RCON	FB0h	SPBRGH	F90h	— ⁽²⁾	F70h	UEP0
FEFh	INDF0 ⁽¹⁾	FCFh	TMR1H	FAFh	SPBRG	F8Fh	— ⁽²⁾	F6Fh	UCFG
FEeh	POSTINC0 ⁽¹⁾	FCEh	TMR1L	FAEh	RCREG	F8Eh	— ⁽²⁾	F6Eh	UADDR
FEDh	POSTDEC0 ⁽¹⁾	FCDh	T1CON	FADh	TXREG	F8Dh	LATE ⁽³⁾	F6Dh	UCON
FECh	PREINC0 ⁽¹⁾	FCCh	TMR2	FACH	TXSTA	F8Ch	LATD ⁽³⁾	F6Ch	USTAT
FEBh	PLUSW0 ⁽¹⁾	FCBh	PR2	FABh	RCSTA	F8Bh	LATC	F6Bh	UEIE
FEAh	FSR0H	FCAh	T2CON	FAAh	— ⁽²⁾	F8Ah	LATB	F6Ah	UEIR
FE9h	FSR0L	FC9h	SSPBUF	FA9h	EEADR	F89h	LATA	F69h	UIE
FE8h	WREG	FC8h	SSPADD	FA8h	EEDATA	F88h	— ⁽²⁾	F68h	UIR
FE7h	INDF1 ⁽¹⁾	FC7h	SSPSTAT	FA7h	EECON2 ⁽¹⁾	F87h	— ⁽²⁾	F67h	UFRMH
FE6h	POSTINC1 ⁽¹⁾	FC6h	SSPCON1	FA6h	EECON1	F86h	— ⁽²⁾	F66h	UFRML
FE5h	POSTDEC1 ⁽¹⁾	FC5h	SSPCON2	FA5h	— ⁽²⁾	F85h	— ⁽²⁾	F65h	SPPCON ⁽³⁾
FE4h	PREINC1 ⁽¹⁾	FC4h	ADRESH	FA4h	— ⁽²⁾	F84h	PORTE	F64h	SPPEPS ⁽³⁾
FE3h	PLUSW1 ⁽¹⁾	FC3h	ADRESL	FA3h	— ⁽²⁾	F83h	PORTD ⁽³⁾	F63h	SPPCFG ⁽³⁾
FE2h	FSR1H	FC2h	ADCON0	FA2h	IPR2	F82h	PORTC	F62h	SPPDATA ⁽³⁾
FE1h	FSR1L	FC1h	ADCON1	FA1h	PIR2	F81h	PORTB	F61h	— ⁽²⁾
FE0h	BSR	FC0h	ADCON2	FA0h	PIE2	F80h	PORTA	F60h	— ⁽²⁾

Figura III.8 Mapa de la memoria del PIC 18F2550

- Amplio rango de funcionamiento (desde -55° a + 150°C).
- Bajo costo.
- Baja impedancia de salida.

El **LM35** es un sensor de temperatura con una precisión calibrada de 1°C. Puede medir temperaturas en el rango que abarca desde -55° a + 150°C. La salida es muy lineal y cada grado centígrado equivale a 10 mV en la salida.

El sensor se presenta en diferentes encapsulados pero el mas común es el to-92 de igual forma que un típico transistor con 3 patas, dos de ellas para alimentarlo y la tercera nos entrega un valor de tensión proporcional a la temperatura medida por el dispositivo. Con el LM35 sobre la mesa las patillas hacia nosotros y las letras del encapsulado hacia arriba tenemos que de izquierda a derecha los pines son: VCC - Vout - GND.

La salida es lineal y equivale a 10mV/°C por lo tanto:

$$+1500\text{mV} = 150^{\circ}\text{C}$$

$$+250\text{mV} = 25^{\circ}\text{C}$$

$$-550\text{mV} = -55^{\circ}\text{C}$$

3.7. Fotorresistencia



Figura III.11 Fotorresistencia

Una fotorresistencia, ldr o fotocelda es un componente electrónico cuya resistencia disminuye con el aumento de intensidad de luz incidente. Puede también ser llamado fotorresistor, fotoconductor, célula fotoeléctrica o resistor dependiente de la luz, cuya siglas, LDR, se originan de su nombre en inglés *light-dependent resistor*. Su cuerpo está formado por una célula o celda y dos patillas. En la siguiente imagen se muestra su símbolo eléctrico.

El valor de resistencia eléctrica de un LDR es bajo cuando hay luz incidiendo en él (puede descender hasta 50 ohms) y muy alto cuando está a oscuras (varios megaohms).

3.8. Características de una Fotorresistencia

Un fotorresistor está hecho de un semiconductor de alta resistencia como el sulfuro de cadmio, CdS. Si la luz que incide en el dispositivo es de alta frecuencia, los fotones son absorbidos por la elasticidad del semiconductor dando a los electrones la suficiente energía para saltar la banda de conducción. El electrón libre que resulta, y su hueco asociado, conducen la electricidad, de tal modo que disminuye la resistencia. Los valores típicos varían entre 1 M Ω , o más, en la oscuridad y 100 Ω con luz brillante.

Las células de sulfuro del cadmio se basan en la capacidad del cadmio de variar su resistencia según la cantidad de luz que incide la célula. Cuanto más luz incide, más baja es la resistencia. Las células son también capaces de reaccionar a una amplia gama de frecuencias, incluyendo infrarrojo (IR), luz visible, y ultravioleta (UV).

La variación del valor de la resistencia tiene cierto retardo, diferente si se pasa de oscuro a iluminado o de iluminado a oscuro. Esto limita a no usar los LDR en aplicaciones en las que la señal luminosa varía con rapidez. El tiempo de respuesta típico de un LDR está en el orden de una décima de segundo. Esta lentitud da ventaja en algunas aplicaciones, ya que se filtran variaciones rápidas de iluminación que podrían hacer inestable un sensor (ej. tubo fluorescente alimentado por corriente alterna). En otras aplicaciones (saber si es de día o es de noche) la lentitud de la detección no es importante.

Se fabrican en diversos tipos y pueden encontrarse en muchos artículos de consumo, como por ejemplo en cámaras, medidores de luz, relojes con radio, alarmas de seguridad o sistemas de encendido y apagado del alumbrado de calles.

También se fabrican fotoconductores de Ge :Cu que funcionan dentro de la gama más baja "radiación infrarroja".

3.9. Módulo Sensor de Humedad HU-10S

Sensor de Humedad para sistemas de aire acondicionado y monitorización, registradores de humedad, provee valores de humedad a través de una salida digital. No

requiere componentes externos, y puede ser conectado directamente a un microcontrolador



Figura III.11 Sensor de humedad HU-10S

3.9.1. Características Eléctricas

Referencia Eléctrica	Módulo HU-10S
Rango de Voltaje de Operación	Voltaje DC 5.0 ±0.1V
Rango de Voltaje de Salida	Voltaje DC 1.0 - 3.0V
Corriente de Operación (Max)	2mA
Rango de Humedad de Almacenamiento (%RH)	0 – 95
Rango de Humedad de Operación (%RH)	20 – 90
Rango de Histéresis de humedad (RH @25°C)	Max 2% RH
Estabilidad en trabajo de larga duración	±1.5%
Linealidad	Salida Lineal
Tiempo de Respuesta (63% del alcance)	1 min
Tamaño (Longitud x Anchura)	34mm x 22mm

CAPÍTULO IV

4. Diseño E Implementación del Prototipo

El Hardware consta de un sensor de temperatura, sensor de luz, y un sensor de humedad los cuales son los encargados de monitorear el estado ambiental, un integrado L293d para controlar el movimiento del robot conocido como el puente H y dos módulos Xbee para la comunicación inalámbrica. Y un codificador y decodificador USB, una PC para monitorear las variaciones de los sensores y controlar el movimiento del robot..

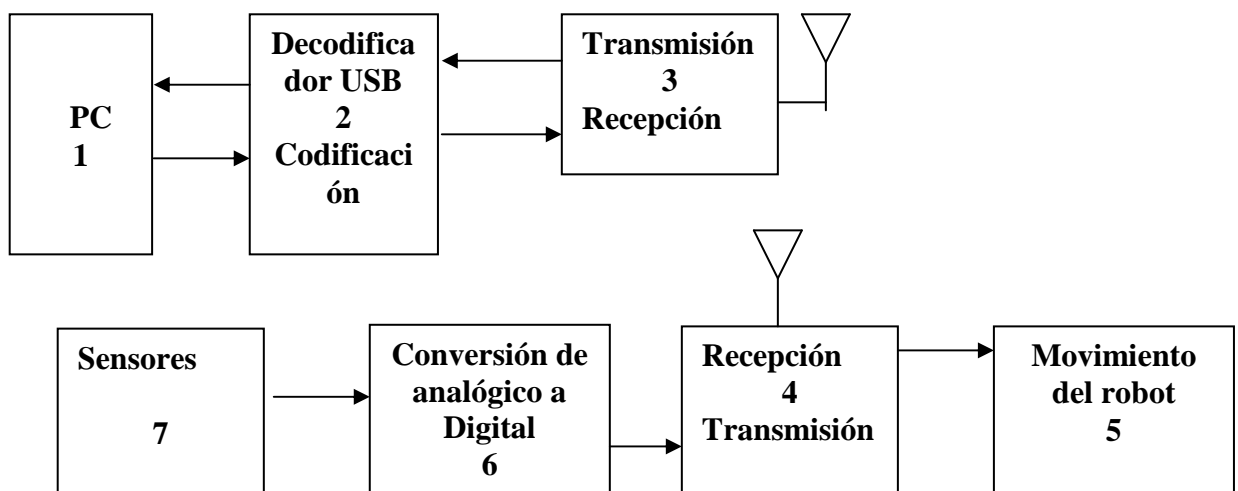


Figura IV.1 Diagrama de Bloques Del Circuito

En el bloque 1 encontramos la PC o computadora es donde vamos enviar y recibir información aquí se encuentra el programa en ejecución.

En el segundo tenemos la interfaz electrónica que permite la comunicación entre el circuito electrónico y la computadora. Ya los datos decodificados en el bloque tres son enviados inalámbricamente por un transceiver hacia el otro circuito

En el bloque cuatro recibe información específicamente esta información enviada desde la PC son datos de movimiento del robot

Cuando desde la PC solicitamos información de los sensores en ese momento el modulo 1 deja de transmitir para poder receptar los datos obtenidos del bloque 7 que son convertidos de voltajes analógicos a digitales como esta en el bloque 6 .El modulo Xbee2 también cambia su opción de receptar para transmitir la información (bloque4) dependiendo del dato de control.

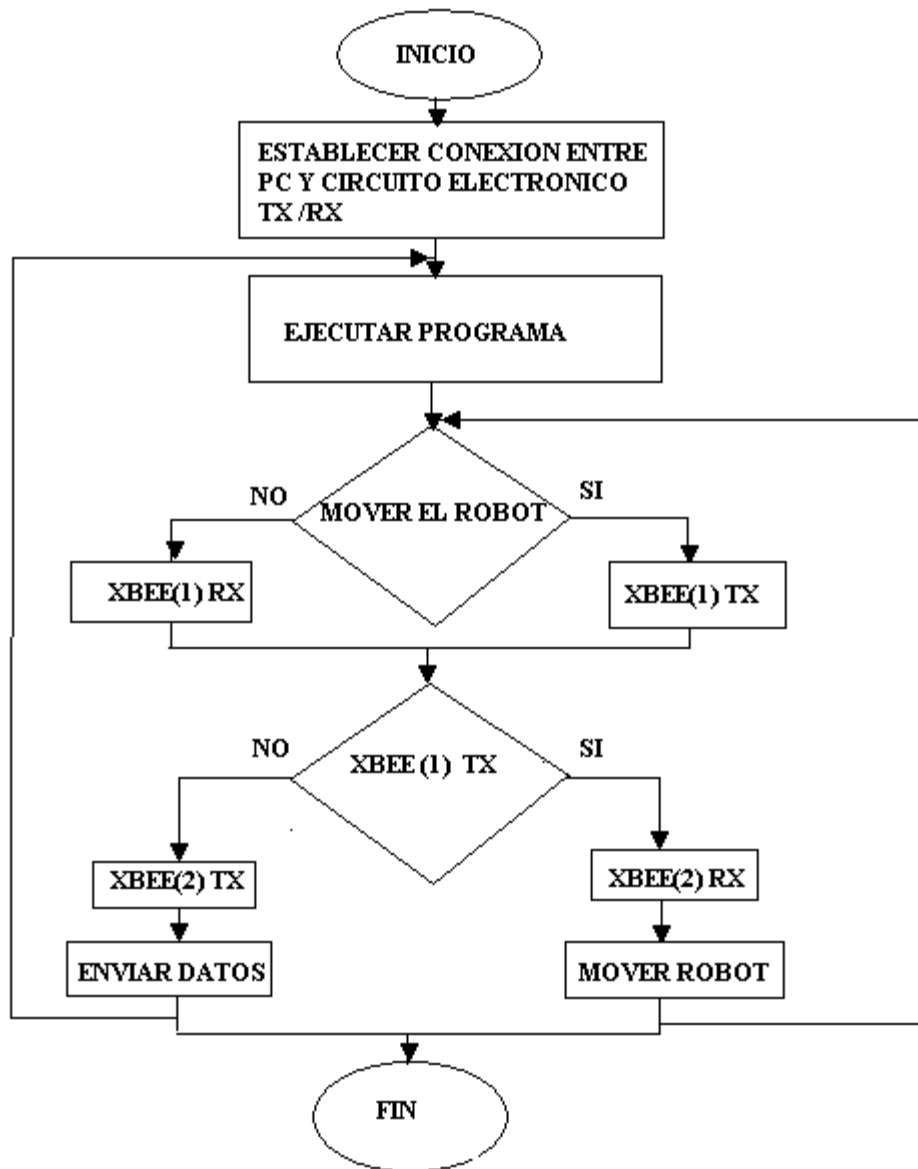


Figura IV.2 Diagrama de Flujo del proyecto

En este diagrama se muestran todas las condiciones por las que debe operar el proyecto dentro del programa de Visual Basic.

4.1. Implementación del Hardware

Algunos programas están diseñados para la simulación electrónica dentro de los cuales tenemos Workbench, Crocodile, Proteus, etc.

Para la simulación se escogió el programa Proteus porque es un poco mas completo con relación a los dispositivos electrónicos que se encuentran en el mercado

Mediante el programa Proteus se realizo el diseño de los circuitos para la simulación respectiva como son el circuito electrónico para la comunicación con la PC, el circuito del sensor de temperatura, circuito del sensor de humedad, circuito de temperatura, control de los motores.

En este programa se tiene casi todos los componentes que se utilizaron en este proyecto a excepción de los módulos Xbee : Estos módulos son dispositivos nuevos en el mercado electrónico aun no constan dentro de los componentes de proteus para la simulación. A continuación el circuito que permite la comunicación con la PC.

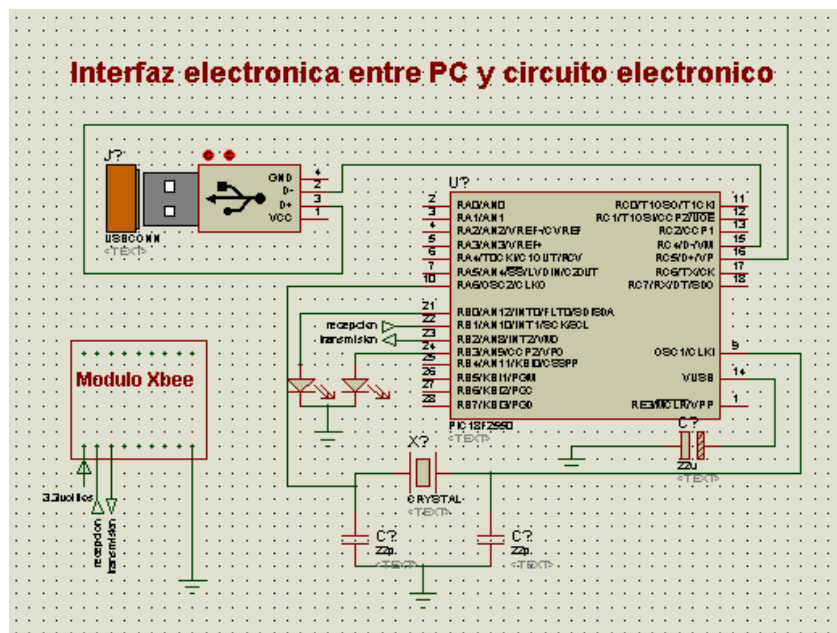


Figura IV.3 Diseño que permite la comunicación entre PC y circuito electrónico

En este diseño electrónico permite el envío y recepción(Xbee1) de datos mediante el puerto USB. El envío de datos de la PC llega al circuito electrónico y el circuito se

encarga de enviar la información inalámbricamente a través del módulo de radio frecuencia Xbee a un receptor que procesa la información y realiza las acciones que le sean asignadas y mediante un control deja de transmitir para que el módulo reciba datos inalámbricamente del módulo 2 dentro de estos datos tenemos de temperatura, luz, humedad.

En el siguiente diseño esta conformado por el modulo Xbee2, el pic 16f877 el integrado L293d .Este diseño también actúa como transmisor y receptor pero uno a la vez en el modo de receptor recibe un dato que es procesado en el pic16f877, y el pic se encarga de realizar ordenes con la información recibida.

Esta información permite activar los motores para el movimiento del robot como son : adelante, atrás, izquierda, derecha.

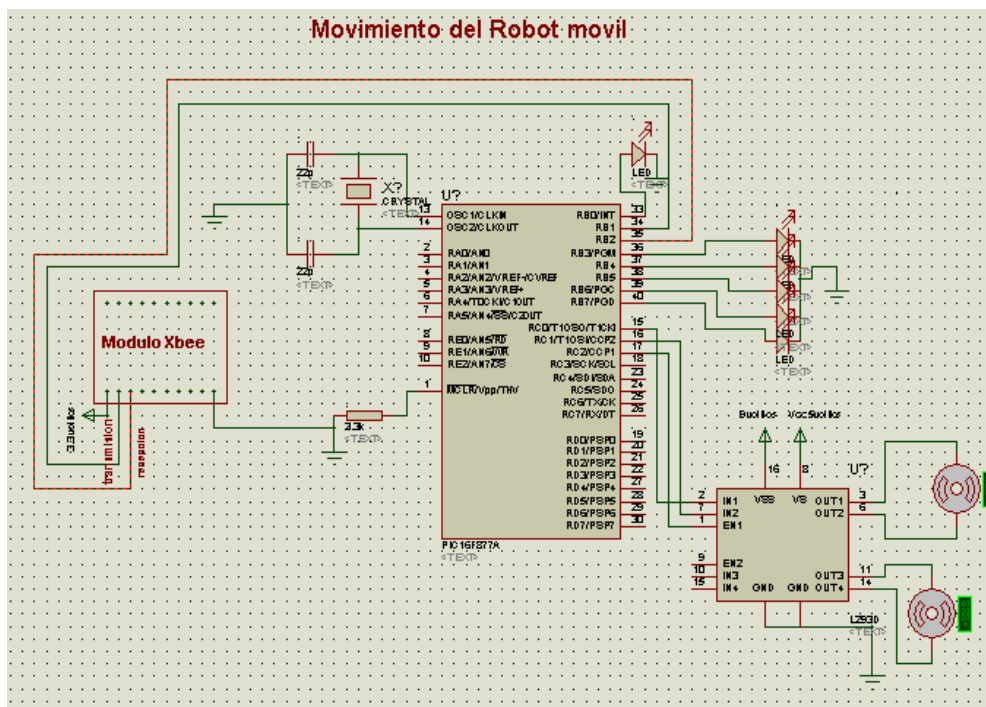


Figura IV.4 Circuito que permite la activación de los motores y envío de datos de sensores

El siguiente diseño es el circuito del sensor de luz que permite reconocer si hay o no incidencia de luz en la fotorresistencia. El circuito nos entrega un voltaje alto o mínimo de acuerdo a la claridad u oscuridad que se le administre al sensor de luz. Este voltaje ingresa al pic16f877 para ser procesado.

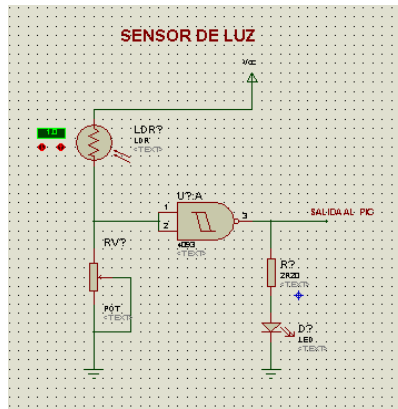


Figura IV.5 Circuito de sensor de luz

El circuito de sensor de temperatura esta compuesto por el dispositivo LM35 y el voltaje que nos entrega es muy bajo por lo que necesitamos un amplificador de voltaje y utilizamos el LM358.

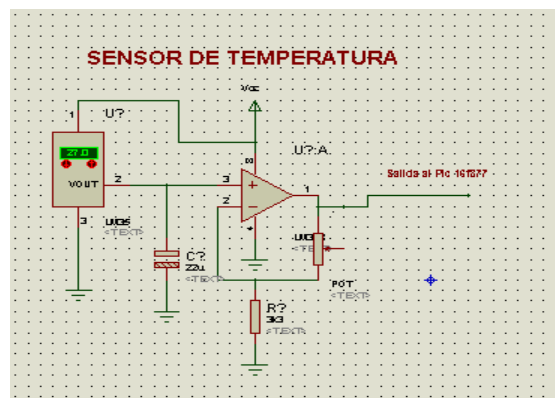


Figura IV.6 Circuito de sensor de Temperatura

programación de alto nivel en el Microcode transformamos el voltaje analógico en voltaje digital para poder ser enviada al modulo Xbee para su transmisión.

4.2. Implementación del Software

Para la programación interna de los microprocesadores tanto del Pic18f2550 y del Pic1f877 se utilizo el programa de Microcode que utiliza lenguaje de lato nivel.

Una vez programado y compilado en el Microcode se crea un archivo .hex y mediante otro programa y un hardware apto para montar el pic se procede a quemar la información dentro del pic.

En el caso del Pic16f877 de utilizo el Ic-prog y el quemador JDM-programer para quemar la información en el pic. El Ic-prog utiliza el puerto serial y no tiene en su configuración para quemar el pic18F2550 para este pic utilizamos otro quemador.

Y para el pic18f2550 utilizamos el software winpic800 y como hardware el Pickit2e este hardware utiliza el puerto USB.

El diseño del programa se puede realizar en Delphi, Visual Basic, Visual C +++ estos programas permiten la comunicación con el puerto USB . El programa se lo realizo en Visual Basic 6.0 ya que es uno de los programas de fácil manejo

A continuación el programa ejecutado.

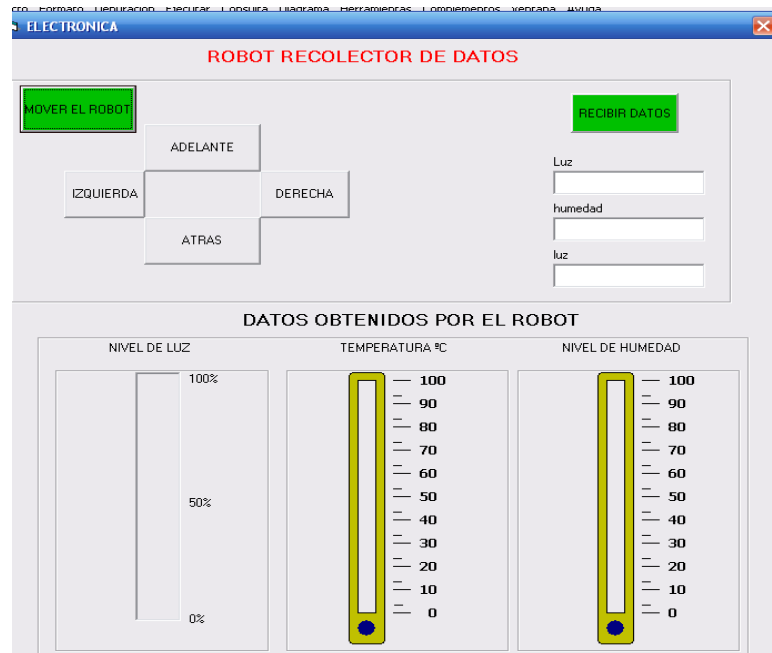


Figura IV.8 Programa final realizado en Visual Basic 6.0

Como se ve en la figura tenemos las opciones de los botones adelante, atrás, izquierda, derecha que es para el movimiento del robot. El botón recibir datos es para recibir los datos censados por el robot. Y esta información es visualizada en la ventana del programa.

En la siguiente figura observamos el móvil en donde se implementara el proyecto

4.3. Plataforma móvil



Figura IV.9. Plataforma móvil para montar el circuito 2

4.4. PRUEBAS Y RESULTADOS

En la siguiente figura muestra el circuito 2 armado en el proyect-board para las pruebas

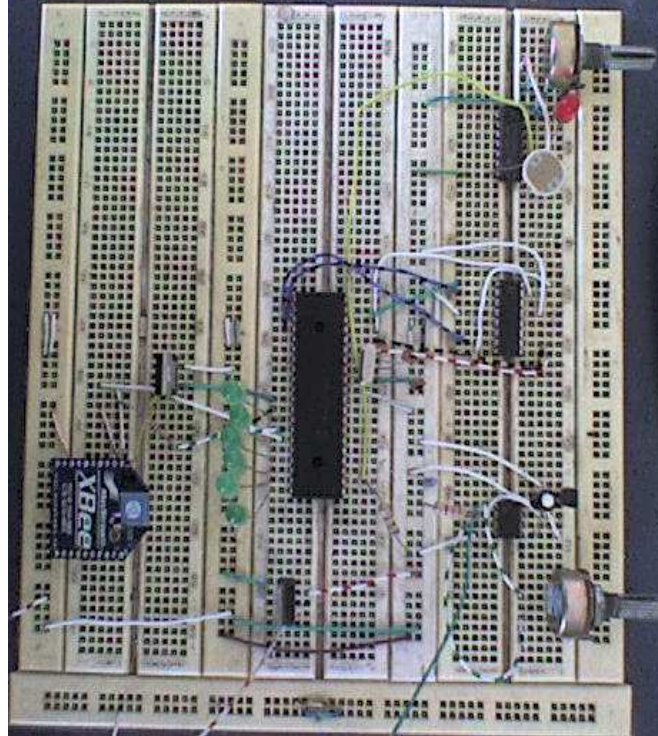


Figura IV.10. Circuito2 armado en el proyect-board

El circuito siguiente es el circuito 1 ya montado en baquelita ya funcionando

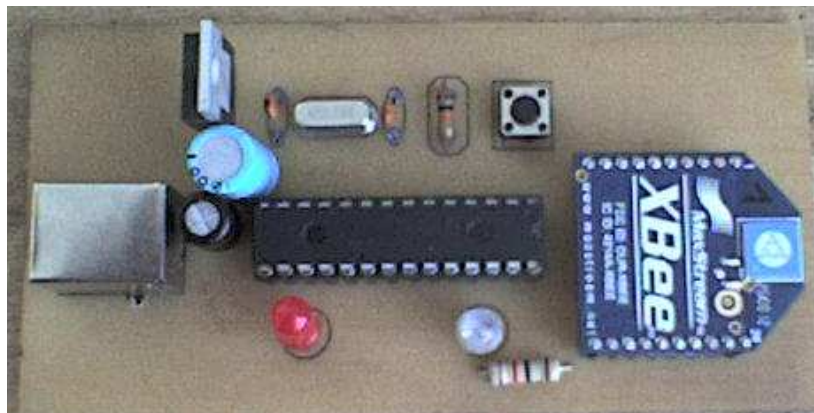


Figura IV.11. Circuito1 en baquelita

4.5. Proyecto Final

Una vez ya armados los circuito1 y circuito2 procedemos a la puesta en marcha de todo el proyecto primero conectamos el cable USB al circuito1 y vemos unos mensajes igual como si se ingresara una flash memory por primera vez.

El circuito 2 lo alimentamos con 5voltios de unas baterías en serie pero si se sobrepasa el voltaje el circuito esta protegida por el CI 7805 que es un regulador de voltaje de 5 Voltios. De igual manera el modulo Xbee necesita una fuente de voltaje de 3.3 voltios y para ello un regulador de voltaje CI LM1117 que nos entrega 3.3 voltios

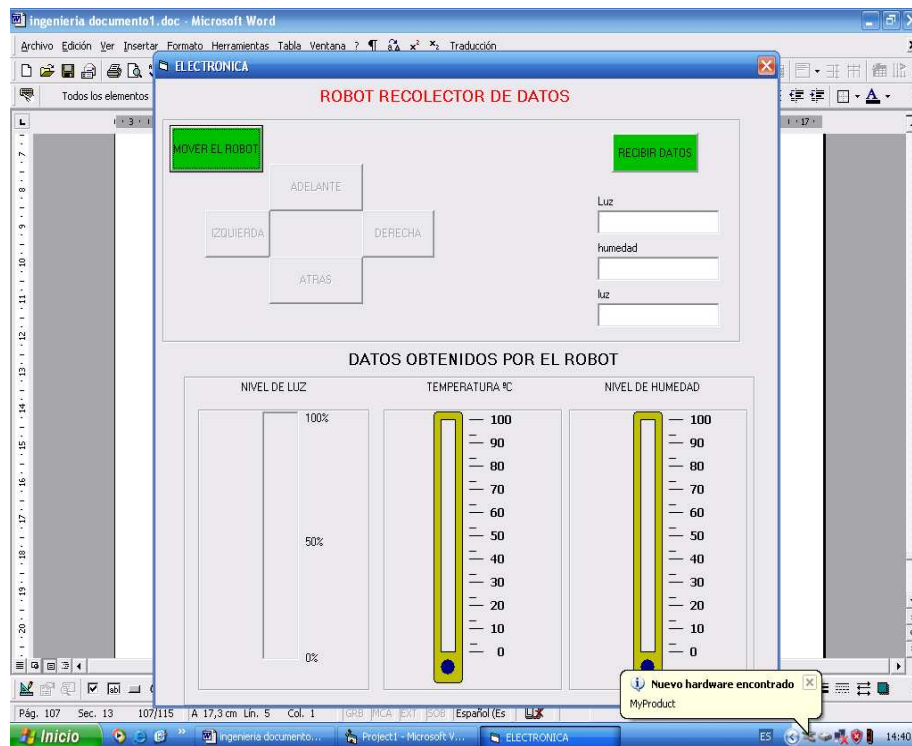


Figura IV.12. Identificación del hardware cuando es conectado al PC

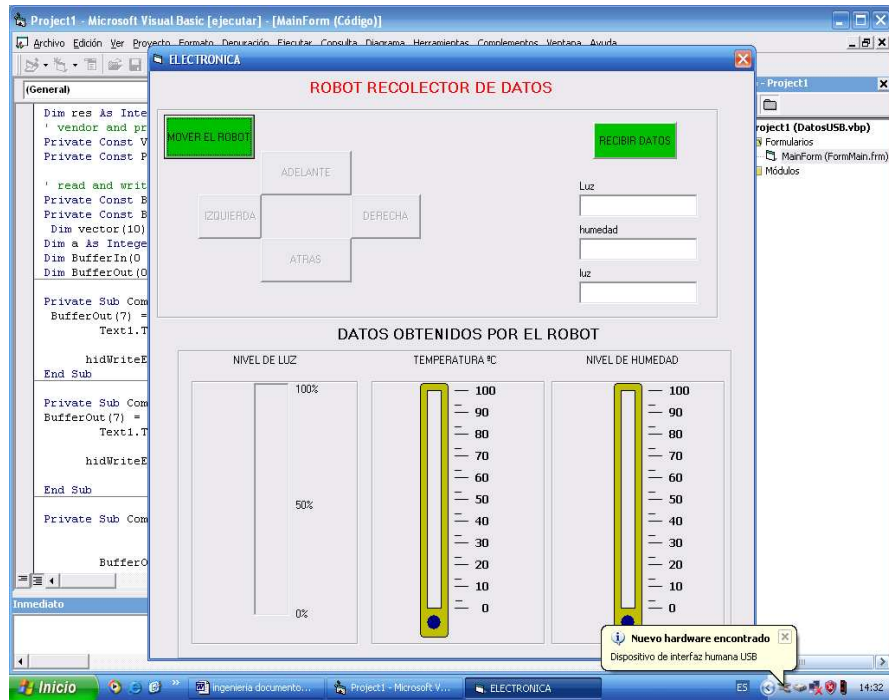


Figura IV.13. Identificación de la interfaz humana

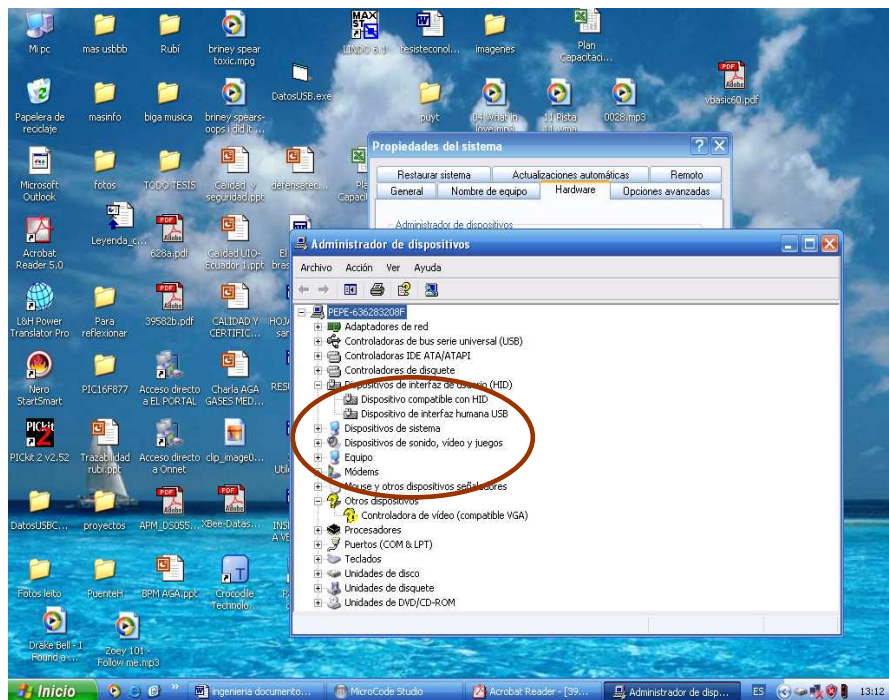


Figura IV.14. Controladores del interfaz humana instalados automáticamente

Ya instalados los controladores procedemos a abrir el programa en Visual Basic como se muestra a continuación.

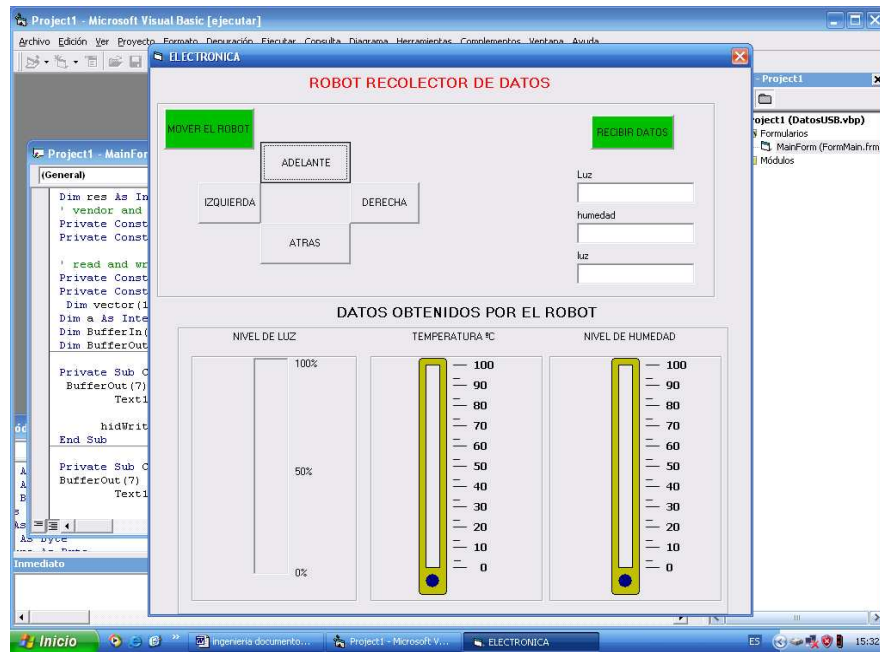


Figura IV.15. Activando los movimientos del robot

En la siguiente ventana estamos recogiendo información de los sensores.

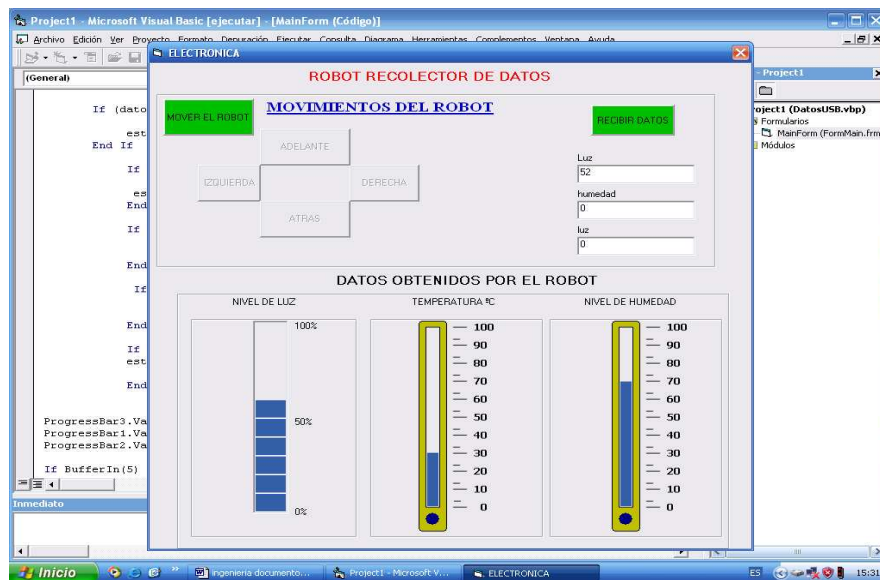


Figura IV.16. Figura recibiendo los datos de los sensores luz, temperatura, humedad

CONCLUSIONES

- El puerto USB brinda una excelente posibilidad de enviar y recibir datos y de esta manera poder controlar cualquier circuito electrónico.
- Los módulos de radio frecuencia Xbee brindan una alta confiabilidad en la transmisión y recepción de datos.
- El pic16F877 al tener 28 pines de entrada y salida se pueden colocar sensores o dispositivos para controlar circuitos de robótica sin dificultad alguna.
- El alcance de transmisión supera los 300 metros sin dificultad y se podría aumentar más la distancia con los módulos Xbee Pro.
- Los módulos Xbee son transceiver bidireccionales half duplex podemos transmitir información pero no recibir al mismo tiempo y viceversa.
- La transmisión de datos entre PC- HW_Xbee deben ser configurados para transmitir a la misma velocidad.
- Con el proyecto realizado se puede explorar áreas difíciles para conocer su temperatura, pudiendo ser analizadas si es o no un peligro para el ser humano.
- La lectura del puerto USB es flexible desde varios SW existentes en el informática.

RECOMENDACIONES

- Realizar la simulación del circuito para comprobar su respectivo funcionamiento.
- Ir comprobando el funcionamiento de la simulación con el circuito ya realizado para evitar errores de funcionamiento.
- Revisar el conector USB tipo B del circuito electrónico si no hay conexión entre la PC y el circuito electrónico
- Estudiar las formas de transformar señales analógicas en señales digitales entorno al Microcode para poder transmitir mediante los módulos Xbee..
- Para un correcto enlace entre los módulos Xbee primero se deben configurar mediante el hiperterminal o por el serial communicator de Microcode la dirección origen y la dirección destino en cada modulo, conectando los módulos al puerto serial del Pc mediante el max 232.
- Asegurar el voltaje adecuado para el funcionamiento correcto de el circuito con el regulador de voltaje 7805. Y en el caso de los módulos Xbee con el regulador LM1117.

BIBLIOGRAFIA

- ANGULO, J y otros. Microcontroladores PIC: la solución en un chip.
Barcelona: Paraninfo, 1997. pp. 204.
- BOYLESTAD, R y Nashelsky, L. Electrónica: teoría de circuitos. México:
Prentice Hall Hispanoamericana, 1998. pp . 99 - 129.
- REYES A, Carlos. Miconroladores PIC: programación en BASIC. 2ª.ed.
Quito: Rispergraf, 2006. pp. 211
- TOCCI R, Widmer. Sistemas Digitales: principios y aplicaciones. 8^{va}.ed
México: Prentice Hall, 2003. pp. 673-674

BIBLIOGRAFÍA DE INTERNET

- DATASHEETS DE DISPOSITIVOS
<http://www.alldatasheet.com>
<http://www.microchip.com/>
2009/10/10
- PROGRAMACION EN MICROCODE
<http://www.mecanique.co.uk/code-studio>
2010/1/5
- INFORMACION DEL PUERTO USB
<http://www.usb.org>
2009/8/12
www.monografias.com
2009/9/4

RESUMEN

Se diseñó e implementó la monitorización y control de un robot recolector de datos mediante los sensores con la finalidad de obtener información de temperatura, luz y humedad.

Con el Microprocesador 18F2550 se logró la comunicación entre la PC y el circuito electrónico por medio del puerto USB. Desde la Pc y mediante el programa realizado en el software de Visual Basic se envió y recibió datos. Si se envía datos desde la PC sirve para controlar el movimiento del robot y con un dato de control se recibe información de los sensores ubicados en el robot móvil.

En el robot móvil los dispositivos lm35,ldr, HS1101,HU-10S son los sensores utilizados para la temperatura, luz y humedad respectivamente. En el robot móvil se recibe y se envía información por los módulos de radio frecuencia(Xbee). Cuando el módulo recibe información es enviada al pic 16F877 donde se analiza la información si es de movimiento del robot móvil o para enviar datos de temperatura, humedad y luz.

Con la ejecución de todo proyecto se realizó pruebas de funcionamiento en áreas con distintos niveles ambientales, con los datos obtenidos por el robot móvil se observó que su porcentaje de veracidad se acerca al 90%.

Este proyecto está apto para tomar mediciones en lugares donde se cree que los niveles normales de temperatura, luz y humedad excedan el límite de lo normal que puedan afectar al ser humano; por ejemplo áreas volcánicas, refinerías.

SUMMARY

The monitoring and control of a data collector robot through sensors were designed and implemented to obtain temperature, light and humidity information. With the microprocessor 18f2550 communication between the PC and the electronic circuit was achieved through the port USB. From the PC and through the program carried out in the Visual Basic software data were sent and received. If data are sent from the Pc it serves to control the robot movement and with a control datum information is received from the sensors used for temperature, light and humidity respectively. In the mobile robot information is received and sent through the modules of radio frequency(Xbee). When the module receives information it is sent to the pic16f877 where the information is analyzed if it is from the mobile robot movement or to send temperature, light and humidity data. With the execution of the whole project functioning tests were carried out in areas with different levels. With the mobile robot data it was observed that is veracity percentage reaches 90%. This project is capable of taking measurements in places where the normal temperature, light and humidity levels are believed to exceed the normal limit affecting the human being such as volcanic areas and refineries.

ANEXO

A continuación las partes más importantes del programa realizado en código fuente de Visual Basic

```
Public Sub OnPlugged(ByVal pHandle As Long)
```

```
    If hidGetVendorID(pHandle) = VendorID And hidGetProductID(pHandle) =  
ProductID Then
```

```
        '** YOUR CODE HERE **
```

```
    End If
```

```
End Sub
```

```
*****
```

```
' a HID device has been unplugged...
```

```
*****
```

```
Public Sub OnUnplugged(ByVal pHandle As Long)
```

```
    If hidGetVendorID(pHandle) = VendorID And hidGetProductID(pHandle) =  
ProductID Then
```

```
        '** YOUR CODE HERE **
```

```
    End If
```

```
End Sub
```

```
*****
```

```

' controller changed notification - called
' after ALL HID devices are plugged or unplugged

*****

Public Sub OnChanged()

    Dim DeviceHandle As Long

    ' get the handle of the device we are interested in, then set
    ' its read notify flag to true - this ensures you get a read
    ' notification message when there is some data to read...

    DeviceHandle = hidGetHandle(VendorID, ProductID)

    hidSetReadNotify DeviceHandle, True

End Sub

```

Anexo de Microcode

Este código permite la comunicación entre el PC y el circuito electrónico

```

DEFINE OSC 48

'DEFINE LOADER_USED 1

INCLUDE "MODEDEFS.BAS"

USBBufferSizeMax con 8 ' maximum buffer size

USBBufferSizeTX con 8 ' input

USBBufferSizeRX con 8 ' output

' the USB buffer...

USBBuffer Var BYTE[USBBufferSizeMax]

USBBufferCount Var BYTE

```