



ESCUELA POLITÉCNICA SUPERIOR DE ALCOY  
LABORATORIO DE SISTEMAS ELECTRONICOS DIGITALES

INTERCONEXION BLUETOOTH CON DSPIC30F4013



PRESENTADO POR:

Yeimy De La Cruz  
Carolina Vargas Barros  
Tsun-Yeng Sierra Marín  
Lynda Evelin Acuña Hernández

Miercoles 31 de marzo de 2010.

## INTRODUCCION

Actualmente existen en el mercado una serie de tecnologías mediante las cuales podemos enviar información o incluso realizar conexiones entre diferentes aparatos electrónicos sin la necesidad del uso de cables, los cuales funcionan de forma eficiente pero su instalación y configuración es bastante engorrosa.

Hoy en día la sociedad tiende a informatizarse cada vez más y ello conlleva la necesidad de tener que interconectar los diferentes aparatos que existen en el mercado de una forma fácil y sencilla.

En el mercado ya existen tecnologías inalámbricas como IrDa, pero aspectos como su ancho de banda o su operatividad (tiene que haber línea visual entre los dispositivos que se quieren comunicar) no han permitido el asentamiento esperado en el mercado. Por ello, se llegó a la conclusión que la solución era un sistema vía radio de bajo coste que permitiera la conectividad entre dispositivos en ausencia de cables y sin la necesidad de que hubiera línea visual entre emisor y receptor.

Actualmente el Bluetooth es una tecnología en auge que ofrece muchas ventajas respecto al resto de tecnologías y en la que parece que todas las empresas del sector de las telecomunicaciones y la informática están muy interesadas.

La evolución de la electrónica desde la aparición del circuito integrado ha sido constante. Actualmente podemos encontrar dispositivos cada vez más complejos ubicados en encapsulados cada vez más pequeños, un ejemplo de esto son los microcontroladores.

Un microcontrolador, es un dispositivo electrónico encapsulado en un chip, capaz de ejecutar un programa. El microcontrolador reúne en un solo integrado: microprocesador, memoria de programa, memoria de datos y puertos de entrada/salida. Además, también suelen disponer de otras características especiales como: puertos serie, comparadores, convertidores analógico-digital, etc. En el mercado existen gran variedad de microcontroladores de marcas y características distintas que podremos utilizar dependiendo de la aplicación a realizar.

Un microcontrolador ejecuta instrucciones. El conjunto de instrucciones es lo que llamamos programa. Las instrucciones son leídas de la memoria de programa para ejecutarlas una detrás de otra. La memoria de programa contiene las instrucciones que queremos que el microcontrolador ejecute.

Programar un microcontrolador consiste en introducir el programa en la memoria del microcontrolador. Las instrucciones son operaciones simples como sumar, restar, escribir en un puerto, activar un bit de un dato, etc. Mediante estas instrucciones básicas podemos realizar operaciones más complejas y así llegar al objetivo de la aplicación. En este TFC nos vamos a centrar en los microprocesadores de la casa Microchip Technology, es decir los PICs. Este tipo de microprocesadores están muy extendidos actualmente en el mercado gracias a su gran variedad y bajo coste. Otra razón del éxito de los PICs es su utilización, ya que una vez se ha aprendido a utilizar uno, conociendo su arquitectura y juego de instrucciones, es muy fácil emplear otro modelo diferente.

En el presente trabajo explicaremos la forma de conectar un dispositivo Bluetooth con un microcontrolador DSPic30F4013. Hemos escogido un módulo Bluetooth para conectarlo con el DS\_PIC llamado el Parani SD 100/110, para el cual explicaremos sus características técnicas y de conexión y programación.

## Bluetooth

Bluetooth es una especificación industrial para Redes Inalámbricas de Área Personal (WPANs) que posibilita la transmisión de voz y datos entre diferentes dispositivos mediante un enlace por radiofrecuencia segura y globalmente libre (2,4 GHz.).

Los principales objetivos que se pretende conseguir con esta norma son:

- Facilitar las comunicaciones entre equipos móviles y fijos.
- Eliminar cables y conectores entre éstos.
- Ofrecer la posibilidad de crear pequeñas redes inalámbricas.
- Facilitar la sincronización de datos entre nuestros equipos personales.

Los dispositivos que con mayor intensidad utilizan esta tecnología son los de los sectores de las telecomunicaciones y la informática personal, como PDAs, teléfonos móviles, computadoras portátiles, ordenadores personales, impresoras y cámaras digitales

### Origen del nombre y logo:

El nombre procede del rey danés y noruego Harald Blåtand cuya traducción al inglés sería Harold Bluetooth (Diente Azul, aunque en lengua danesa significa 'de tez oscura') conocido por buen comunicador y por unificar las tribus noruegas, suecas y danesas.

De la misma manera, Bluetooth intenta unir diferentes tecnologías como las de las computadoras, los teléfonos móviles y el resto de periféricos.

El símbolo de Bluetooth es la unión de las runas nórdicas análogas a las letras B y H: ✞ (Hagall) y Bjarkan (Berkanan).



### Historia

En 1994 la empresa sueca Ericsson inició un estudio para investigar la viabilidad de una interfaz vía radio, de bajos costo y consumo, para la interconexión entre teléfonos móviles y otros accesorios con el objetivo de eliminar cables entre aparatos. El estudio partía de un largo proyecto que investigaba sobre unos multicomunicadores conectados a una red celular, hasta que se llegó a un enlace de radio de corto alcance llamado MC link. Con el avance del proyecto quedó claro que este tipo de enlace podía ser utilizado en un gran número de aplicaciones, pues poseía como ventaja principal el hecho de basarse en un chip de radio relativamente económico.

A principios de 1997, otros fabricantes de equipos portátiles despertaron su interés por el avance del proyecto MC link y para que el sistema tuviera éxito, un gran número de equipos debería estar formado con esta tecnología.

A principios de 1998, la creación de un Grupo de Especial Interés en Bluetooth (SIG), formado por cinco promotores y que fueron Ericsson, Nokia, IBM, Toshiba e Intel. La idea era lograr un conjunto adecuado de áreas de negocio, ya que se hallaban en el grupo dos líderes del mercado de las telecomunicaciones, dos del mercado de las PCs portátiles y un líder de la fabricación de chips. El propósito principal del consorcio fue y es, el establecer un estándar para la interface aérea junto con su software de control, con el fin de asegurar la interoperabilidad de los equipos entre los diversos fabricantes.

Más tarde en diciembre de 1999 se unieron a la iniciativa Microsoft, 3com, Axis Communication, Compaq, Dell, Motorola, Xircom, Lucent y otros publicando a su vez una nueva versión de la tecnología conocida como 1.0B.

En febrero del 2000 como prueba del gran desarrollo de Bluetooth ya son más de 1500 empresas las adheridas al grupo SIG. Y por supuesto siguen publicándose nuevas versiones adaptando las nuevas mejoras que van apareciendo, la última publicación se trata de la versión 2.0.

## **El SIG de Bluetooth**

El Bluetooth SIG es una asociación privada sin ánimo de lucro con sede en Bellevue, Washington. A fecha de septiembre de 2007, el SIG está formado por más de 9000 compañías de telecomunicaciones, informática, automovilismo, música, textil, automatización industrial y tecnologías de red. Tiene pequeños grupos de personal dedicado al grupo en Hong Kong, Suecia y Estados Unidos.

Los miembros del SIG dirigen el desarrollo de la tecnología inalámbrica Bluetooth, además de implementar y comercializar la tecnología en sus productos. El Bluetooth SIG por sí mismo no fabrica ni vende dispositivos Bluetooth.

Bluetooth se denomina al protocolo de comunicaciones diseñado especialmente para dispositivos de bajo consumo, con una cobertura baja y basados en transceptores de bajo coste. Gracias a este protocolo, los dispositivos que lo implementan pueden comunicarse entre ellos cuando se encuentran dentro de su alcance.

Las comunicaciones se realizan por radiofrecuencia de forma que los dispositivos no tienen por qué estar alineados, pueden incluso estar en habitaciones separadas si la potencia de transmisión lo permite.

La clasificación de los dispositivos Bluetooth como "Clase 1", "Clase 2" o "Clase 3" es únicamente una referencia de la potencia de transmisión del dispositivo, siendo totalmente compatibles los dispositivos de una clase con los de la otra.

| Clase   | Potencia máxima permitida (mW) | Potencia máxima permitida (dBm) | Rango (aproximado) |
|---------|--------------------------------|---------------------------------|--------------------|
| Clase 1 | 100 mW                         | 20 dBm                          | ~100 metros        |
| Clase 2 | 2.5 mW                         | 4 dBm                           | ~25 metros         |
| Clase 3 | 1 mW                           | 0 dBm                           | ~1 metro           |

La cobertura efectiva de un dispositivo de clase 2 se extiende cuando se conecta a un transceptor de clase 1. Esto es así gracias a la mayor sensibilidad y potencia de transmisión del dispositivo de clase 1. Es decir, la mayor potencia de transmisión del dispositivo de clase 1 permite que la señal llegue con energía suficiente hasta el de clase 2. Por otra parte la mayor sensibilidad del dispositivo de clase 1 permite recibir la señal del otro pese a ser más débil.

Los dispositivos con Bluetooth también pueden clasificarse según su ancho de banda:

| Versión                   | Ancho de banda  |
|---------------------------|-----------------|
| Versión 1.2               | 1 Mbit/s        |
| Versión 2.0 + EDR         | 3 Mbit/s        |
| UWB Bluetooth (propuesto) | 53 - 480 Mbit/s |

### Perfiles Bluetooth:

Para utilizar Bluetooth, un dispositivo debe implementar alguno de los perfiles Bluetooth. Estos definen el uso del canal Bluetooth.

### Lista de aplicaciones

- Conexión sin cables entre los celulares y equipos de manos libres y kit para vehículos.
- Red inalámbrica en espacios reducidos donde no sea tan importante un ancho de banda grande.
- Comunicación sin cables entre la computadora y dispositivos de entrada y salida.
- Mayormente impresora, teclado y mouse.
- Transferencia de ficheros entre dispositivos vía OBEX.
- Transferencia de fichas de contactos, citas y recordatorios entre dispositivos vía OBEX.
- Reemplazo de la tradicional comunicación por cable entre equipos GPS y equipamiento médico.
- Controles remotos (tradicionalmente dominado por el infrarrojo).

- Enviar pequeñas publicidades desde anunciantes a dispositivos con Bluetooth. Un negocio podría enviar publicidad a teléfonos móviles cuyo Bluetooth (los que lo posean) estuviera activado al pasar cerca.
- Las consolas Sony PlayStation 3 y Nintendo Wii incorporan Bluetooth, lo que les permite utilizar mandos inalámbricos.

### Información técnica

La especificación de Bluetooth define un canal de comunicación de máximo 720 kb/s (1 Mbps de capacidad bruta) con rango óptimo de 10 m (opcionalmente 100 m con repetidores).

La frecuencia de radio con la que trabaja está en el rango de 2,4 a 2,48 GHz con amplio espectro y saltos de frecuencia con posibilidad de transmitir en Full Duplex con un máximo de 1600 saltos/s. Los saltos de frecuencia se dan entre un total de 79 frecuencias con intervalos de 1Mhz; esto permite dar seguridad y robustez.

La potencia de salida para transmitir a una distancia máxima de 10 metros es de 0 dBm (1 mW), mientras que la versión de largo alcance transmite entre 20 y 30 dBm (entre 100 mW y 1 W).

Para lograr alcanzar el objetivo de bajo consumo y bajo costo, se ideó una solución que se puede implementar en un solo chip utilizando circuitos CMOS. De esta manera, se logró crear una solución de 9x9 mm y que consume aproximadamente 97% menos energía que un teléfono celular común.

El protocolo de banda base (canales simples por línea) combina conmutación de circuitos y paquetes. Para asegurar que los paquetes no lleguen fuera de orden, los slots pueden ser reservados por paquetes síncronos, un salto diferente de señal es usado para cada paquete. Por otro lado, la conmutación de circuitos puede ser asíncrona o síncrona. Tres canales de datos síncronos (voz), o un canal de datos síncrono y uno asíncrono, pueden ser soportados en un solo canal. Cada canal de voz puede soportar una tasa de transferencia de 64 kb/s en cada sentido, la cual es suficientemente adecuada para la transmisión de voz. Un canal asíncrono puede transmitir como mucho 721 kb/s en una dirección y 56 kb/s en la dirección opuesta, sin embargo, para una conexión síncrona es posible soportar 432,6 kb/s en ambas direcciones si el enlace es simétrico.

### VERSIONES DEL BLUETOOTH

- **Bluetooth v.1.1**

En 1994, [Ericsson](#) inició un estudio para investigar la viabilidad de una nueva interfaz de bajo costo y consumo para la interconexión vía [radio](#) (eliminando así cables) entre dispositivos como [teléfonos](#) móviles y otros accesorios.

El estudio partía de un largo proyecto que investigaba unos multicomunicadores conectados a una red celular, hasta que se llegó a un enlace de radio de corto alcance, llamado *MC link*. Conforme este proyecto avanzaba se fue haciendo claro que éste tipo de enlace podía ser utilizado ampliamente en un gran número de aplicaciones, ya que tenía como principal virtud que se basaba en un [chip](#) de radio.

- **Bluetooth v.1.2**

A diferencia de la V.1.1, provee una solución inalámbrica complementaria para coexistir Bluetooth y [Wi-Fi](#) en el espectro de los 2.4 GHz, sin interferencia entre ellos. La versión 1.2 usa la técnica “*función de salto adaptable de frecuencia (AFH)*” que ejecuta una transmisión más eficiente y una encriptación más segura.

Esta tecnología inalámbrica *Bluetooth* se diseñó expresamente para reducir las interferencias de las tecnologías inalámbricas que comparten el espectro de 2.4 GHz. La “*función de salto adaptable de frecuencia (AFH)*” utiliza la frecuencia disponible dentro del espectro.

Para ello, detecta los dispositivos conectados y descarta las frecuencias que éstos estén utilizando. Este salto adaptable permite unas transmisiones más eficaces dentro del espectro, por lo que se mejora el funcionamiento del dispositivo, incluso si el usuario utiliza otras tecnologías al mismo tiempo.

La señal salta entre 79 frecuencias en intervalos de 1 MHz para tener un alto grado de tolerancia a las interferencias.

La **v.1.2** se diseña para mejorar las experiencias de los usuarios, ofrece una calidad de voz con menor ruido ambiental y provee una más rápida configuración de la comunicación con los otros dispositivos bluetooth dentro del rango del alcance, como pueden ser PDAs, ordenadores portátiles, ordenadores de sobremesa, impresoras y teléfonos móviles.

- **Bluetooth v.2.0**

Esta versión de la especificación Bluetooth fue lanzada el 10 de noviembre de 2004. Es compatible con la versión anterior 1.2. La principal diferencia es la introducción de una velocidad de datos mejorada “*Tasa de Datos Mejorada (EDR)*” para la rápida transferencia de datos.

La tasa nominal de EDR es de 3Mbps, aunque la tasa de transferencia de datos de prácticas es de 2,1 Mbps.

De acuerdo con la v.2.0 + EDR, EDR proporciona las siguientes ventajas:

- Tres veces la velocidad de transmisión (2,1 Mbit / s) en algunos casos.
- Reducción de la complejidad de múltiples conexiones simultáneas debido al ancho de banda adicional.
- Bajo consumo de energía a través de un ciclo de trabajo reducido.

Esta versión intenta mejorar algunos errores de la v.1.2.

- **Bluetooth v.2.1**

Versión 2.1 + EDR es totalmente compatible con 1.2, fue adoptada el 26 de julio de 2007. Soporta velocidades de transferencia de datos teóricos de hasta 3 Mbit / s. Esta especificación incluye las siguientes características:

- **Respuesta de investigación extendido (EIR)**

Proporciona más información durante el procedimiento de investigación para permitir un mejor filtrado de dispositivos antes de la conexión. Esta información puede incluir el nombre del dispositivo, una lista de servicios que el dispositivo es compatible, el nivel

de potencia de transmisión utilizada para las respuestas de la investigación, y el fabricante de datos definidos.

- **Sniff subrating**

Reduce el consumo de energía cuando los dispositivos están en modo de bajo consumo, especialmente con flujos de datos asimétricos. “*Dispositivos de interfaz humana (HID)*” estos dispositivos son los más beneficiados con esta tecnología como el ratón y el teclado.

- **Pausa de cifrado / CV (EPR)**

Permite una clave de cifrado con menos gestiones requeridas para el anfitrión Bluetooth.

- **Vinculación segura y confirmada (SSP)**

Mejora radicalmente la experiencia de vinculación de dispositivos Bluetooth el aumento del uso y seguridad.

- **Comunicación de campo cercano (NFC)**

Garantiza las conexiones Bluetooth cuando la interfaz de radio “*Near Field Communication (NFC)*” es un [protocolo](#) basado en una [interfaz inalámbrica](#), la [comunicación](#) se realiza entre dos entidades ([peer-to-peer](#)), el protocolo establece conexión wireless entre las aplicaciones de la red y los dispositivos electrónicos también está disponible.

Un ejemplo es la carga automática de fotos desde un teléfono móvil o a un marco de fotos digital.

- **Bluetooth v.3.0**

Esta nueva versión de *Bluetooth* se basa en las cualidades de la versión 2.1 EDR, incluyendo la seguridad automática, como con todas las versiones del *Bluetooth* la v. 3,0 proporciona a los desarrolladores, fabricantes y consumidores el beneficio de la compatibilidad con las demás versiones, permitiendo la expansión y mejora de esta tecnología.

## **Aplicaciones**

Usa el protocolo de conexión **802.11 (WiFi)**, lo que conlleva una ganancia en la transferencia de datos de **hasta unos 24 Mbps**. Con la disponibilidad de *Bluetooth* versión 3.0 HS, los consumidores pueden mover archivos de gran tamaño como videos, música y fotos sin necesidad de cables y alambres. Algunas aplicaciones son:

- Sincronizar de forma inalámbrica las bibliotecas de música entre el PC y reproductor de música o teléfono
- Descargar fotos a una impresora o PC
- Enviar archivos de vídeo desde una cámara o teléfono a la computadora o la televisión.

## Especificaciones técnicas

- Suplente MAC / PHY (AMP)
- 802,11 Protocolo de Capa de Adaptación (PAL)
- Genérico metodología de la prueba
- Control mejorado de energía
- Unicast a la conexión de datos

- **Bluetooth V4.0**

El 17 de diciembre del 2009, el Bluetooth SIG Bluetooth aprobó la “energía baja (BLE)” como el rasgo distintivo de la versión 4.0. Wibree y Bluetooth ULP (Ultra Low Power).

El 12 de junio de 2007, Nokia y Bluetooth SIG anunciaron que Wibree será parte de la especificación Bluetooth, Wibree es una nueva tecnología digital de radio interoperable para pequeños dispositivos. Wibree es la primera tecnología abierta de comunicación [inalámbrica](#), que ofrece comunicación entre dispositivos móviles o computadores y otros dispositivos más pequeños (de pila de botón).

Las aplicaciones de esta nueva versión se desarrollaran sobre todo para la salud, la seguridad entre otras, en las aplicaciones del **hogar digital lo han enfocado en el entretenimiento.**

### Bajo consumo para los equipos con bluetooth

Si hasta ahora muchos dispositivos electrónicos no adoptaban la conectividad Bluetooth, la causa tenemos que buscarla en el consumo. Con la nueva especificación esto ya no será un problema. En teoría.

Tampoco será excusa la seguridad pues la nueva especificación llega con **cifrado AES 128 usando CCM.**



Y si en nuestro entorno tenemos más equipos conectándose vía Bluetooth la versión 4.0 también ha pensado en ello, habiendo **mejorado la comunicación sin problemas entre equipos de diferentes marcas**, además de ampliarse el **alcance a 100 metros**. Podrán usarse de forma simultánea prácticamente con los dispositivos que queramos, pues con el direccionamiento de 32 bits las cifras hablan de miles de millones de equipos usándose de forma simultánea.

En cuanto a las velocidades de transferencia, los datos a transmitir son cada día mayores, **Bluetooth 4.0 permite transferencias de hasta 1 Mbps**.

Los primeros equipos con Bluetooth low energy llegarán al mercado en **2010**, pues ya están siendo intensamente probados por los diferentes fabricantes.

## FUTURO DE BLUETOOTH

**Los estándares Bluetooth y WiFi parecen compartir algo más que la banda de frecuencias en un futuro cercano, la idea es fusionar tecnologías de forma inteligente.**

La tecnología Bluetooth es un protocolo para transferencia de datos de forma inalámbrica y de corto alcance (10 metros). El concepto era disponer de la transferencia inalámbrica de archivos en dispositivos de bajo consumo (dispositivos que funcionan con baterías y pocas veces conectados a la red eléctrica). Operando bajo la banda de frecuencias de 2.4Ghz pero con una implementación de saltos frecuenciales que le permitiera ser lo más inmune posible a las interferencias, Bluetooth se ha impuesto como la tecnología dominante en transferencia de audio más que de datos debido a su bajo ancho de banda, la mejora de sus especificaciones con audio estéreo (Bluetooth 2.0 con AD2P) y alcances de hasta 100 metros ha hecho que esa potencia mínima que lo abanderó, sea ahora más comparable al despilfarro energético de WiFi que a la idea original de Bluetooth.

WiFi sin embargo nació de un conglomerado de tecnologías inalámbricas poco estandarizadas para transferencias elevadas de datos entre dispositivos estáticos. Operando también en la banda de 2,4GHz pero sin calidad de servicio alguna a diferencia del Bluetooth. Se ha impuesto como estándar de transferencia inalámbrica con su última revisión 802.11n y hasta la llegada de WIMAX promete seguir siéndolo algunos años.

Ahora se especula que formaran una nueva especificación que será un híbrido de ambas tecnologías, MAC/PHY será el nombre que reciba el nuevo protocolo que implementará Bluetooth sobre WiFi a grosso modo. Es decir, bluetooth se usará para establecer las conexiones y mantener el enlace vivo entre dispositivos, mientras que WiFi tomará el control una vez se establezca la transferencia de datos masiva. Al finalizar el proceso de transferencia se devolverá el control a la tecnología Bluetooth para mantener activo el enlace en tareas de sincronización típicas. De esta manera, aparte de un considerable ahorro energético, se podrá integrar en un sólo chip lo que hasta hoy se hacía con dos, y viendo la importancia de la miniaturización de dispositivos, giros de este tipo se hacen indispensables en tecnología.

Desde el punto de vista de la utilidad, el implementar esto en los dispositivos actuales haría más sencilla la sincronización de música en MP4's con el PC, de terminales móviles e incluso GPS's con los mapas. Sin duda libraría del estorbo de tener que

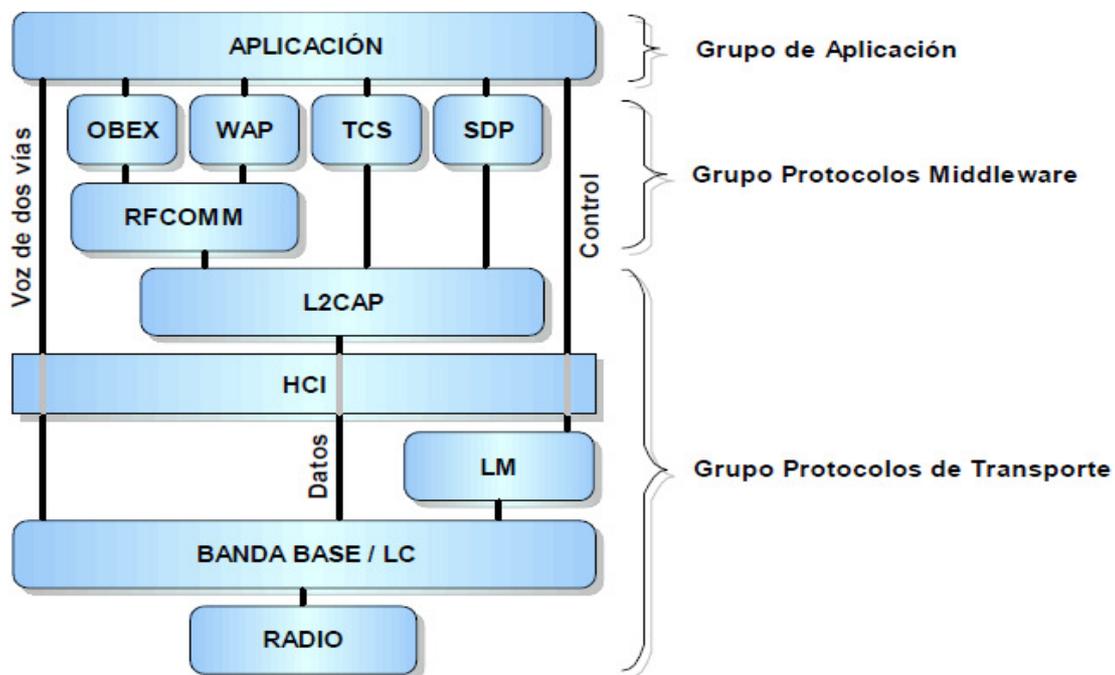
comprar un stick USB con bluetooth para el ordenador al poder integrarlo en la placa junto al chip WiFi. Porque en los sobremesas es bastante extraño aún ver el Bluetooth integrado.

### PILA DE PROTOCOLOS BLUETOOTH

La pila de protocolos del bluetooth fue desarrollado en su totalidad por la S16. Y su objetivo es conseguir que aplicaciones de diferentes fabricantes mantengan una comunicación fluida, para conseguirla llegaron a una conclusión que el receptor y el transmisor se debe de ejecutarse sobre la misma pila de protocolos.

La pila está constituida de varias capas entre las cuales, a excepción de las capas más bajas, existen protocolos que permiten su interrupción.

Los datos de las pilas fluyen a través de todas las capas, excepto de la información de audio que va directa desde la banda base hasta la aplicación que va con alto grado de prioridad para garantizar la calidad del servicio.



### PROTOCOLO DE TRANSPORTE

Está integrada por un conjunto de protocolos que permiten la búsqueda de dispositivos bluetooth.

El manejo y la configuración del enlace lógico y físicos que conforman una tubería virtual, por lo cual fluyen los datos desde y hacia las capas más altas. Estos protocolos de transporte pueden ser implementados tanto en hardware como en firmware.

**LC:** Controlador de enlace

**LM:** Manejador de enlace

**HCI:** Interfaz controlador de host

**L2CAP:** Protocolo de adaptación y control de enlaces lógicos, proporciona servicios de datos tanto orientados a conexiones a los protocolos de capas superiores.

**LC2CAP** se basa en el concepto de canales. Un canal es una conexión lógica que se situá sobre la conexión de la banda base, todo canal se asocia a un único protocolo. Cada paquete **LC2CAP** que si recibe un canal se dirige al protocolo superior correspondiente.

**FIRMWARE:** Este consiste en agrupar bloques para así dar instrucciones que este integrado en la electrónica del dispositivo, es parte hardware y también software ya que también proporciona lógica y se dispone en algún tipo de lenguaje de programación.

### **PROTOCOLOS MIDDLEWARE (Protocolos de capa media).**

Es un software de conectividad, funciona entre las capas de aplicaciones y las capas inferiores. Presenta interfaces estándares para el flujo de datos desde el grupo de transporte hacia la capa de aplicación. Esta funciona como protocolos intermediarios.

### **RFCOMM (Comunicación por radio frecuencia).**

Es un conjunto simple de protocolos de transporte, construido sobre el protocolo **L2CAP** lo cual proporciona sesenta conexiones simultaneas para dispositivo bluetooth. Emulando puertos, series **RS-232**.

**OBEX:** Intercambio de datos

**WAP:** Aplicación inalámbrica

**TCS:** Protocolo de control de teléfono

**SDP:** Protocolo de descubrimiento de servicios.

### **Aplicación**

Este grupo se refiere al software y es suministrado por el fabricante del aparato o vendidos por desarrolladores independientes, estos programas se acompañan con la pila del protocolo bluetooth para obtener beneficios del producto.

Para una aplicación determinada, no es necesario utilizar todas las capas de la pila. El fabricante es el que indica que capas en particular se deben usar y de que manera según la aplicación.

### **PARALELO ENTRE OSI Y BLUETOOTH**



La pila de protocolos Bluetooth no sigue exactamente el modelo OSI, aunque si se hace un seguimiento gradual de las capas se pueden encontrar transiciones en la implementación de cada una de ellas desde **hardware**, pasando por **firmware**, y finalmente llegando a **software**. Bluetooth exhibe este mismo comportamiento.

La capa física define las características eléctricas del medio de transporte para los datos, por lo que el radio y parte de la banda base cubren esta capa. La capa de enlace de datos es responsable de la transmisión, marco y control de error sobre un particular enlace, por lo que traslapa la tarea del controlador de enlace y parte final de la banda base, incluyendo chequeo y corrección de error.

Una de las razones por la cual la pila Bluetooth no es igual al modelo **OSI** es la interoperabilidad. Dado que las capas altas del modelo **OSI** dan libertad a los fabricantes de implementarlas como desearán, dispositivos de diferentes fabricantes posiblemente no se comunicaban adecuadamente. El Bluetooth **SIG** anticipando el problema, tuvo en cuenta esta situación, así que la especificación junto con otros documentos relacionados, fue construida de tal forma que se requería la interoperabilidad no importando quien diseñaba y construía el dispositivo.

## MODULOS BLUETHOOT PARA CONECTAR CON EL DSPIC

Existe una amplia gama de dispositivos Bluetooth los cuales sirven para realizar una conexión al DSPIC30F4013 en el mercado, los podemos escoger de acuerdo a las diferentes especificaciones técnicas y económicas. En nuestro caso hemos de explicar cómo se conecta uno de estos dispositivos, para el cual hemos escogido el Parani SD 100/110, donde explicaremos su funcionamiento programación y conexión.

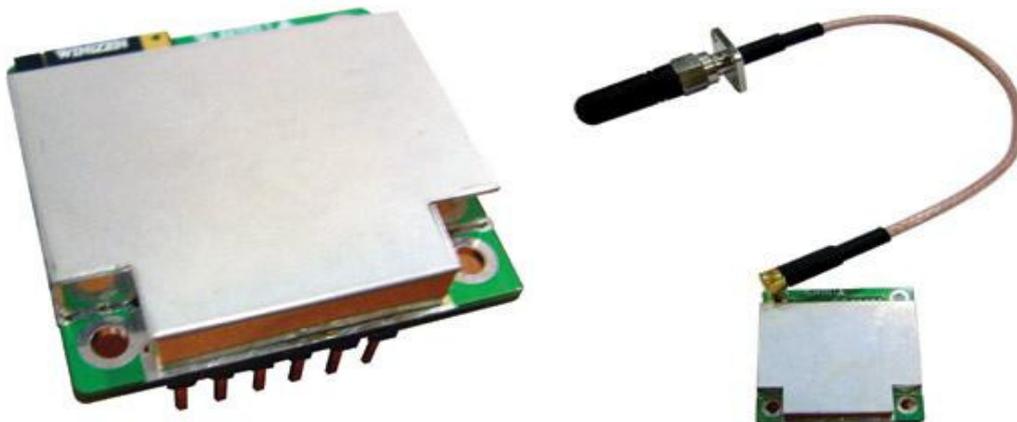
### Descripción del Modulo OEM Bluetooth

El Parani-ESD es un convertidor embebido serial a bluetooth, especial para aquellos usuarios que necesitan incorporar conectividad bluetooth directamente en los circuitos impresos de equipos basados en RS-232 como sistemas de seguridad, POS, maquinaria industrial y dispositivos medicos. Puede ser conectado al dispositivo usando la interfase UART que viene con el módulo y a su vez éste con otros dispositivos bluetooth como teléfonos móviles, computadores portátiles, PDAs y Handhelds entre otros.

No necesita software de configuración, ya que un dip switch externo permite la configuración instantánea de un par de dispositivos Parani-SD para comunicación punto a punto y además viene equipado con opciones flexibles de alimentación.

Puede ser conectado al dispositivo usando la interfase UART que viene con el módulo y a su vez éste con otros dispositivos bluetooth como teléfonos móviles, computadores portátiles, PDAs y Handhelds entre otros.

Posee un compacto diseño, que le permite ser implementado convenientemente dentro de cualquier dispositivo equipo, su antena desmontable optimiza la calidad y distancia en comunicaciones wireless.



El modulo puede ser configurado y controlado mediante comandos AT, los usuario pueden configurar de manera fácil el modulo utilizando un hyperterminal, además de contar con los comandos At básicos, el parani posee una expansión de dichos comandos para varias funciones.

Los comandos AT son instrucciones codificadas que conforman un lenguaje de comunicación entre el hombre y un Terminal MODEM. Los comandos AT fueron desarrollados en 1977 por Dennis Hayes como un interfaz de comunicación con un MODEM para así poder configurarlo y proporcionarle instrucciones.

## Características Técnicas Parani SD 100/110

|                              |  |
|------------------------------|--|
| Interface de salida          | UART, cumple con el estándar v1.2 de Bluetooth mejorado AFH (Adaptive Frequency Hopping), rápida conexión.                                       |
| Potencia de transmisión      | ESD100/110 : Max. +18dBm.  |
| Sensibilidad de recepción    | ESD100/110 : -88dBm(0.1%BER)   |
| Tamaño compacto              | ESD100/110: 27.5 x 27.7 x 14(mm)   |
| Plataforma de Comunicaciones | Soporta Bluetooth Serial Port Profile.   |
| Interoperabilidad            | con PDAs, portátiles, etc.   |
| Antena superficial           | Incluida   |
| Actualizaciones              | De firmware usando herramientas de software basada en Windows (Parani Updater).  |
| Distancia de operación       | (en línea de vista) Parani-ESD100: Clase 1, Nominal 100 metros Parani-ESD110: Clase 1, Nominal 100 metros, hasta 1000m usando antena tipo Patch. |
| Herramienta de configuración | Basada en Windows amigable al usuario No se requieren drivers externos.  |

## DESCRIPCION DE COMANDOS AT DEL PARANI SD 100/110

### ATZ

|              |  |
|--------------|--|
| Respuesta SD | <input type="checkbox"/> OK <input type="checkbox"/> |
| Propósito    | Software reset                                       |
| Referencia   | AT&F, AT+BTCSD, AT+UARTCONFIG                        |

### AT&F

|              |  |
|--------------|--|
| Respuesta SD | <input type="checkbox"/> OK <input type="checkbox"/> |
| Propósito    | Hardware reset.                                      |
| Referencia   | ATZ  |

### AT

|              |  |
|--------------|--|
| Respuesta SD | <input type="checkbox"/> OK <input type="checkbox"/> |
| Propósito    | Check the connection status with host equipment      |
| Referencia   | AT+UARTCONFIG, ATZ, AT&F                             |

### AT+UARTCONFIG, Baudrate, Parity, Stopbit, Hwf

|              |  |
|--------------|--|
| Respuesta SD | <input type="checkbox"/> OK <input type="checkbox"/> |
| Propósito    | Set Serial parameters.                               |
| Referencia   | AT, ATZ, AT&F, ATS                                   |

### AT+BTINFO?

|              |  |
|--------------|--|
| Respuesta SD | <input type="checkbox"/> 112233445566,DeviceName,Mode,Status,Auth,Encryp,FlowContr |
|--------------|--|

|            |  |
|------------|--|
|            | oI□<br>□OK□                            |
| Propósito  | Display Bluetooth settings.            |
| Referencia | AT+BTNAME, AT+BTMODE, AT+BTSEC, ATS14? |

### AT+BTINQ?

|              |   |
|--------------|---|
| Respuesta SD | □112233445566,FriendlyName,CoD□<br>□112233445566,FriendlyName,CoD□<br>□112233445566,FriendlyName,CoD□<br>□OK□ |
| Propósito    | Search Bluetooth devices nearby.  |
| Referencia   | AT+BTSCAN, ATD, AT+BTINFO?  |

### AT+BTLAST?

|              |  |
|--------------|--|
| Respuesta SD | □112233445566□                                       |
| Propósito    | Display the BD address of the last connected device. |
| Referencia   | AT+BTSCAN, ATD, AT+BTINFO?, AT+BTINQ?                |

### AT+BTVER?

|              |                                  |
|--------------|----------------------------------|
| Respuesta SD | □SD100v1.0.0□<br>□OK□            |
| Propósito    | Display device firmware versión. |
| Referencia   | AT+BTINFO?                       |

### AT+BTRSSI,n

|              |   |
|--------------|---|
| Respuesta SD | □OK□<br>□0,255,0,0□ (repeatedly).                                 |
| Propósito    | Test signal strength.   |
| Parámetro    | n=0: Start signal strength test<br>n=1: Stop signal strength test |

### AT+BTMODE,n

|              |  |
|--------------|--|
| Respuesta SD | □OK□   |
| Propósito    | Set operation mode.  |
| Parámetro    | n=0: MODE0 (Default)<br>n=1: MODE1<br>n=2: MODE2<br>n=3: MODE3 |

+++

|              |  |
|--------------|--|
| Respuesta SD | □OK□   |
| Propósito    | Convert the operation status of 'Connect'to 'Standby'. |
| Referencia   | AT+SETESC, ATO, AT+BTCANCEL                            |

### AT+SETESC,nn

|              |                                       |
|--------------|---------------------------------------|
| Respuesta SD | □OK□                                  |
| Propósito    | Change the escape sequence character. |
| Referencia   | +++ , ATO                             |

**ATO**

|              |  |
|--------------|--|
| Respuesta SD | None   |
| Propósito    | Convert the operation status of 'Standby'to 'Connect'. |
| Referencia   | +++ , AT+SETESC  |

**AT+BTCANCEL**

|              |  |
|--------------|--|
| Respuesta SD | <input type="checkbox"/> OK <input type="checkbox"/> |
| Propósito    | Terminate the current executing task.                |
| Referencia   | AT+BTSCAN, ATD, AT+BTINQ?                            |

**AT+BTSCAN**

|              |  |
|--------------|--|
| Respuesta SD | <input type="checkbox"/> OK <input type="checkbox"/><br><input type="checkbox"/> CONNECT 112233445566 <input type="checkbox"/> |
| Propósito    | Wait for inquiry and connection from other Bluetooth devices.  |
| Referencia   | ATD, AT+BTINQ?, AT+BTCANCEL  |

**AT+BTSCAN,n,to**

|              |   |
|--------------|---|
| Respuesta SD | <input type="checkbox"/> OK <input type="checkbox"/><br><input type="checkbox"/> CONNECT 112233445566 <input type="checkbox"/><br>or<br><input type="checkbox"/> OK <input type="checkbox"/><br><input type="checkbox"/> ERROR <input type="checkbox"/> |
| Propósito    | Wait for inquiry and connection from other Bluetooth devices for a given duration.  |
| Parámetros   | n=1: Allows Inquiry scan<br>n=2: Allows Page scan<br>n=3: Allows both of Inquiry scan and Page scan<br>to= Time duration in seconds   |
| Referencia   | ATD, AT+BTINQ?, AT+BTCANCEL   |

**AT+BTSCAN112233445566,to**

|              |   |
|--------------|---|
| Respuesta SD | <input type="checkbox"/> OK <input type="checkbox"/><br><input type="checkbox"/> CONNECT 112233445566 <input type="checkbox"/><br>or<br><input type="checkbox"/> OK <input type="checkbox"/><br><input type="checkbox"/> ERROR <input type="checkbox"/> |
| Propósito    | Wait for connection by the Bluetooth device with given BD address.  |
| Parametros   | 112233445566=BD address to= time duration in seconds.   |
| Referencia   | ATD, AT+BTINQ?, AT+BTCANCEL   |

**ATD**

|              |  |
|--------------|--|
| Respuesta SD | <input type="checkbox"/> OK <input type="checkbox"/><br><input type="checkbox"/> CONNECT 112233445566 <input type="checkbox"/> |
|--------------|--|

|            |   |
|------------|---|
|            | or<br><input type="checkbox"/> OK<br><input type="checkbox"/> ERROR |
| Propósito  | Connect to the last connected Bluetooth device.                     |
| Referencia | AT+BTINQ?, AT+BTSCAN  |

#### **ATD112233445566**

|              |   |
|--------------|---|
| Respuesta SD | <input type="checkbox"/> OK<br><input type="checkbox"/> CONNECT 112233445566<br>or<br><input type="checkbox"/> OK<br><input type="checkbox"/> ERROR |
| Propósito    | Connect to a specific Bluetooth device with a given BD address.   |
| Parametros   | 112233445566=BD address.  |
| Referencia   | AT+BTINQ?, AT+BTSCAN  |

#### **ATH**

|              |  |
|--------------|--|
| Respuesta SD | <input type="checkbox"/> OK<br><input type="checkbox"/> DISCONNECT |
| Propósito    | Release the current connection.                                    |
| Referencia   | ATD, AT+BTSCAN   |

#### **AT+BTKEY=\$string**

|              |  |
|--------------|--|
| Respuesta SD | <input type="checkbox"/> OK                      |
| Propósito    | Change pin code.                                 |
| Parametros   | <i>\$string</i> = New pin code (Default="1234"). |
| Referencia   | AT+BTCSD, AT+BTFP, AT+BTSD?, AT+BTSEC, ATZ, AT&F |

#### **AT+BTSD?**

|              |  |
|--------------|--|
| Respuesta SD | <input type="checkbox"/> 112233445566<br><input type="checkbox"/> OK |
| Propósito    | Display a list of Bluetooth devices sharing the same pin code.       |
| Referencia   | AT+BTCSD, AT+BTFP, AT+BTKEY, AT+BTSEC, ATZ, AT&F                     |

#### **AT+BTCSD**

|              |  |
|--------------|--|
| Respuesta SD | <input type="checkbox"/> OK                                    |
| Propósito    | Clear the list of Bluetooth devices sharing the same pin code. |
| Referencia   | AT+BTFP, AT+BTKEY, AT+BTSD?, AT+BTSEC, ATZ, AT&F               |

#### **AT+BTFP,n**

|              |  |
|--------------|--|
| Respuesta SD | <input type="checkbox"/> OK                            |
| Propósito    | Set generation of link key every time of connection.   |
| Parámetros   | n=0: Inactivate (Default)<br>n=1: Activate             |
| Referencia   | AT+BTCSD, AT+BTKEY, AT+BTSD?, AT+BTSEC, ATD, ATZ, AT&F |

## AT+BTSEC,Authentication,Encryption

|              |  |
|--------------|--|
| Respuesta SD | <input type="checkbox"/> OK <input type="checkbox"/>   |
| Propósito    | Set authentication and data encryption.  |
| Parámetros   | <i>Authentication</i> =0: Inactivate (Default)<br><i>Authentication</i> =1: Activate<br><i>Encryption</i> =0: Inactivate (Default)<br><i>Encryption</i> =1: Activate |
| Referencia   | AT+BTCSD, AT+BTFP, AT+BTSD?, AT+BTSD?, ATZ, AT&F   |

## AT+BTNAME=\$string

|              |  |
|--------------|--|
| Respuesta SD | <input type="checkbox"/> OK <input type="checkbox"/>         |
| Propósito    | Change device name.  |
| Parámetros   | <i>\$string</i> = New device name (Default="PSDv3b-445566"). |
| Referencia   | AT+BTINFO?, AT+BTINQ?  |

## AT+BTLPM,n

|              |  |
|--------------|--|
| Respuesta SD | <input type="checkbox"/> OK <input type="checkbox"/> |
| Propósito    | Set low power mode.                                  |
| Parámetros   | n=0: Inactivate (Default)<br>n=1: Activate           |

## AT+DFU

|              |                                       |
|--------------|---------------------------------------|
| Respuesta SD | Display garbage messages repeatedly). |
| Propósito    | Device firmware update.               |

## AT&V

|              |   |
|--------------|---|
| Respuesta SD | <input type="checkbox"/> S0:m0;S1:m1; ...Sn:mn <input type="checkbox"/><br><input type="checkbox"/> OK <input type="checkbox"/> |
| Propósito    | Display all the S-register  |
| Referencia   | ATS   |

## ATSnn?

|              |   |
|--------------|---|
| Respuesta SD | <input type="checkbox"/> value <input type="checkbox"/><br><input type="checkbox"/> OK <input type="checkbox"/> |
| Propósito    | Display a given S-register.   |
| Parámetros   | Nn = Address of S-register.   |
| Referencia   | AT&V  |

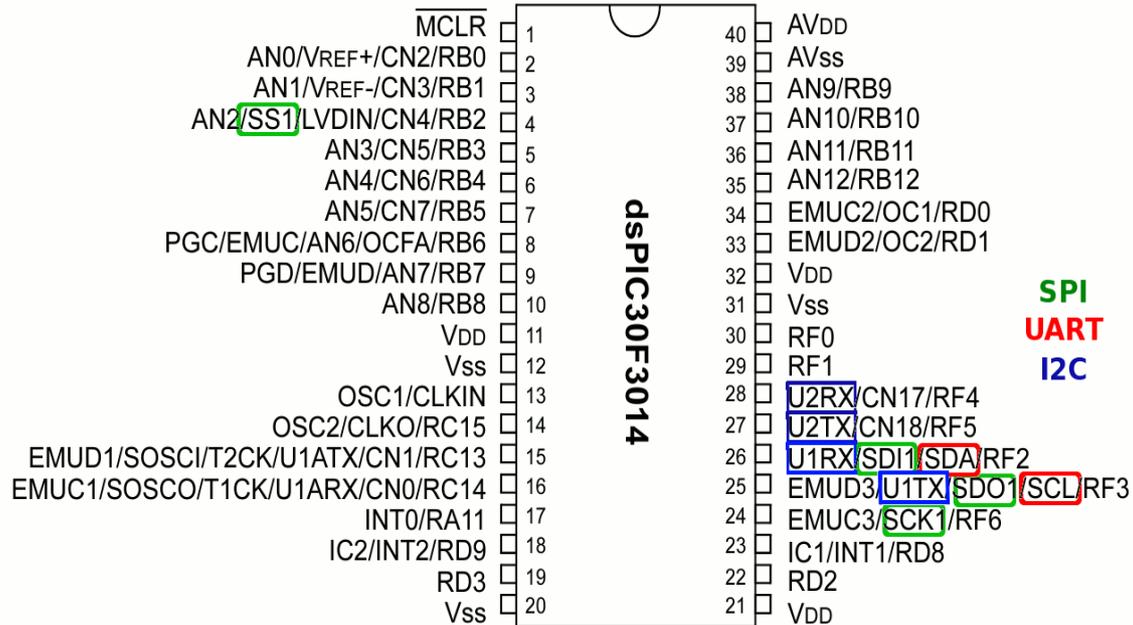
## ATSnn=mm

|              |  |
|--------------|--|
| Respuesta SD | <input type="checkbox"/> OK <input type="checkbox"/>       |
| Propósito    | Change S-register value.                                   |
| Parámetros   | nn= Address of S-register.<br>mm= New value of S-register. |
| Referencia   | AT&V   |

## CONEXIÓN DE DISPOSITIVOS

### Conexiones en el DSP:

En toda comunicación se debe establecer un protocolo de comunicación definido, y tal como ocurre en nuestra sesión de comunicación, tenemos una emulación serial mediante el protocolo RFCOMM integrado en el dispositivo de radio Bluetooth y el módulo UART integrado en el dispositivo de procesamiento DSP. Los puertos que utilizaremos para la conexión a bluetooth que es la aplicación que dada a nuestro trabajo son 25 y 26 (**Módulo UART (Módulo de Transmisión y Recepción Universal Asíncrono)**).



La figura anterior muestra cómo es posible conectar el Bluetooth a los diferentes módulos del DSPIC lo son el Módulo SPI y I<sup>2</sup>C, los cuales también se pueden conectar a diferentes tipos de módulos Bluetooth que existen en el mercado pero que no cumplen el propósito de nuestra aplicación.

### Módulo UART (Módulo de Transmisión y Recepción Universal Asíncrono):

Dentro de las características principales del modulo encontramos:

- Comunicación Full-Duplex de 8 ó 9 bits de datos en comunicación.
- Diferentes modos de paridad (Par o Impar) disponible para 8 bits de datos.
- Configuración de uno o dos bits de parada.
- Generador de Tasa de Baudios completamente integrado con un preescaler de 16 bits.
- Rangos de configuración para la Tasa de Baudios: Desde 38 bps hasta 1875 Mbps a una frecuencia de 30 Mhz.
- Profundidad en el buffer de transmisión de datos de 4 palabras.
- Profundidad en el buffer de recepción de datos de 4 palabras.
- Paridad, entramado y buffer de detección de sobreflujo.
- Soporte para interrupciones sólo en detección de dirección.
- Interrupciones de transmisión y recepción de forma separada.
- Modo de retroalimentación para soporte de diagnóstico.
- Dos opciones para escoger libremente los pines de TX/RX sobre el módulo UART1.

Las transferencias de información se realizan sobre dos líneas, UTX (transmisión) y URX (recepción), saliendo y entrando los bits por dichas líneas al ritmo de una frecuencia controlada internamente por el módulo UART y específicamente de común acuerdo por la tasa de Baudios acordada para una transmisión.

Cada palabra o dato de información se envía independientemente de los demás. Suele constar de 8 ó 9 bits y van precedidos por un bit de inicio y detrás de ellos se coloca un bit de stop ó parada, de acuerdo a las normas del formato estándar NRZ. Los bits son transmitidos a una frecuencia fija y normalizada.

El módulo UART está formado por tres complementos específicos:

- Generador de Baudios.
- Transmisor Asíncrono.
- Receptor Asíncrono.

## **Modelo Interno de Funcionamiento**

### **Perspectiva General**

#### **Transmisión:**

La transmisión asíncrona se describe de la siguiente manera:

El dato que se desea transmitir por el módulo UART de transmisión se deposita en el registro UTREG y a continuación se pasa al registro de desplazamiento UTSR, que va sacando los bits secuencialmente y a la frecuencia establecida.

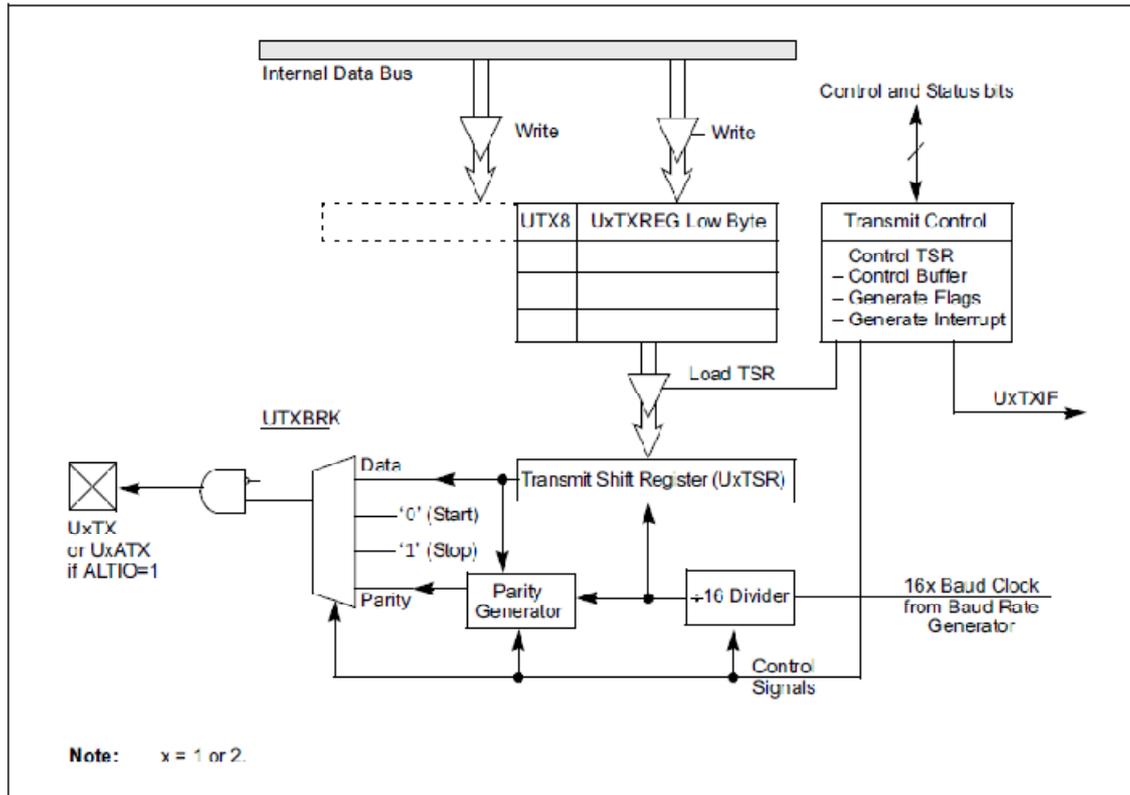
Además, antes de los bits del dato de información existe un bit de INICIO y después de todos los bits de datos se añade un bit de PARADA. La UART de recepción recibe uno a uno los bits, elimina los dos de control y una vez que los bits de información han llenado el registro de desplazamiento URSR, se trasladan automáticamente al registro URREG, donde quedan disponibles para su posterior procesamiento.

El núcleo del módulo UART está constituido por el registro de desplazamiento UTSR, que obtiene el dato del registro UTREG y luego lo va sacando bit a bit. El primer bit en salir es el de menos peso. El dato a transferir se carga mediante software en UTREG y se transfiere al UTSR en cuanto se haya transmitido el bit de PARADA del dato anterior. La transferencia entre los dos registros se realiza en un ciclo de reloj, tras ello un señalizador se activa indicando que el registro de transmisión se ha ido vaciando conforme avanza la transmisión. También en ese momento puede producirse una interrupción si se ha programado el bit correspondiente para ello. El bit TRMT sirve para indicar el estado del registro UTSR y vale "1" cuando está vacío.

La secuencia de pasos a seguir para implementar una transmisión con el módulo UART es la siguiente:

1. Si se desea trabajar con interrupción se coloca el bit UTISEL a 1, además de habilitar las interrupciones en general.
2. Se carga el valor adecuado en el registro UBRG, para generar la frecuencia de trabajo deseada.
3. Poner a 1 el bit UTEN para permitir la transmisión.
4. Cargar en UTREG el dato a transmitir. Comienza la transmisión.

**FIGURE 16-1: UART TRANSMITTER BLOCK DIAGRAM**



### Recepción:

La recepción asíncrona se describe de la siguiente manera:

Los datos se reciben en serie bit a bit y se van introduciendo secuencialmente en el registro de desplazamiento URSR, que funciona a una frecuencia 16 veces más rápida que la de trabajo. Cuando un procesador maestro intenta enviar información a uno de los esclavos, primero envía un byte de dirección que identifica al destinatario. El byte de dirección se identifica porque el noveno bit que llega vale "1". Si el bit ADDEN es igual a "1", en el esclavo se ignorarán todos los bytes de datos. Pero si el noveno bit que se recibe vale "1", quiere decir que se trata de una dirección y el esclavo provocará una interrupción y se transferirá el contenido del registro URSR al buffer de recepción. Tras la interrupción, el esclavo deberá examinar la dirección y si coincide con la suya debe poner ADDEN a 0 para poder recibir datos del maestro.

Si ADDEN vale 1, todos los datos son ignorados, el bit de PARADA no se carga en el registro URSR, por lo que este hecho no produciría una interrupción.

Los pasos a seguir en el modo de interrupción son los siguientes:

1. Se carga con el valor X al registro UBRG, para trabajar con la frecuencia deseada.
2. Si se desea que se genere una interrupción con el bit de PARADA, se pone a "1" el bit URISSEL.
3. Si se desea trabajar con detección de dirección ponemos el bit ADDEN a "1".
4. Se habilita la recepción.
5. Al completarse la recepción se producirá una interrupción si se había habilitado.
6. Se leen los bits PERR y FERR (Errores de Paridad y entramado respectivamente) para determinar si se ha producido algún error.
7. Determinar si el dispositivo ha sido direccionado.
8. Si ha sido direccionado el dispositivo, poner ADDEN=0 para permitir la recepción de información.

## CONFIGURACIÓN GENERAL DEL MÓDULO UART

### Habilitación del módulo UART:

Para habilitar el módulo UART sólo es necesario escribir en “1” el bit UARTEN en el registro UxMODE (donde claramente tenemos que “x” puede ser 1 ó 2).

TABLE 16-1: dsPIC30F3014/4013 UART1 REGISTER MAP

| SFR Name | Addr. | Bit 15                        | Bit 14 | Bit 13 | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 | Bit 7             | Bit 6    | Bit 5 | Bit 4 | Bit 3 | Bit 2  | Bit 1  | Bit 0 | Reset State         |
|----------|-------|-------------------------------|--------|--------|--------|--------|--------|-------|-------|-------------------|----------|-------|-------|-------|--------|--------|-------|---------------------|
| U1MODE   | 020C  | UARTEN                        | —      | USIDL  | —      | —      | ALTIO  | —     | —     | WAKE              | LPBACK   | ABAUD | —     | —     | PDSEL1 | PDSEL0 | STSEL | 0000 0000 0000 0000 |
| U1STA    | 020E  | UTXISEL                       | —      | —      | —      | UTXBRK | UTXEN  | UTXBF | TRMT  | URXISEL1          | URXISEL0 | ADDEN | RIDLE | PERR  | FERR   | OERR   | URXDA | 0000 0001 0001 0000 |
| U1TXREG  | 0210  | —                             | —      | —      | —      | —      | —      | —     | UTX8  | Transmit Register |          |       |       |       |        |        |       | 0000 0000 0000 0000 |
| U1RXREG  | 0212  | —                             | —      | —      | —      | —      | —      | —     | URX8  | Receive Register  |          |       |       |       |        |        |       | 0000 0000 0000 0000 |
| U1BRG    | 0214  | Baud Rate Generator Prescaler |        |        |        |        |        |       |       |                   |          |       |       |       |        |        |       | 0000 0000 0000 0000 |

Legend: — = uninitialized bit

TABLE 16-2: dsPIC30F3014/4013 UART2 REGISTER MAP

| SFR Name | Addr. | Bit 15                        | Bit 14 | Bit 13 | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 | Bit 7             | Bit 6    | Bit 5 | Bit 4 | Bit 3 | Bit 2  | Bit 1  | Bit 0 | Reset State         |
|----------|-------|-------------------------------|--------|--------|--------|--------|--------|-------|-------|-------------------|----------|-------|-------|-------|--------|--------|-------|---------------------|
| U2MODE   | 0210  | UARTEN                        | —      | USIDL  | —      | —      | —      | —     | —     | WAKE              | LPBACK   | ABAUD | —     | —     | PDSEL1 | PDSEL0 | STSEL | 0000 0000 0000 0000 |
| U2STA    | 0218  | UTXISEL                       | —      | —      | —      | UTXBRK | UTXEN  | UTXBF | TRMT  | URXISEL1          | URXISEL0 | ADDEN | RIDLE | PERR  | FERR   | OERR   | URXDA | 0000 0001 0001 0000 |
| U2TXREG  | 021A  | —                             | —      | —      | —      | —      | —      | —     | UTX8  | Transmit Register |          |       |       |       |        |        |       | 0000 0000 0000 0000 |
| U2RXREG  | 021C  | —                             | —      | —      | —      | —      | —      | —     | URX8  | Receive Register  |          |       |       |       |        |        |       | 0000 0000 0000 0000 |
| U2BRG    | 021E  | Baud Rate Generator Prescaler |        |        |        |        |        |       |       |                   |          |       |       |       |        |        |       | 0000 0000 0000 0000 |

Legend: — = uninitialized bit

Note: Refer to dsPIC30F Family Reference Manual (DS70040) for descriptions of register bit fields.

Una vez habilitados, los pines UxTX y UxRX se configuran como salidas y entradas respectivamente, generando los cambios en los registros de TRIS (para la habilitación de dirección de los puertos que usa la UART) y LATCH para los correspondientes puertos de entrada/salida. El pin UxTX se encuentra en “1” lógico cuando no se lleva a cabo ninguna transmisión.

### Deshabilitación del módulo UART:

El módulo UART se deshabilita limpiando o colocando un “0” en el bit UARTEN del registro UxMODE. Éste es el estado por defecto de dicho bit después de haber ocurrido un Reset en todo el dispositivo. Si el módulo UART está deshabilitado, todos los pines de entrada y salida operan como pines de puerto bajo el control del LATCH de puerto ordinario, y de los bits de TRIS de los correspondientes puertos que corresponden coincidentalmente a los que usa el módulo UART. Cuando se deshabilita el módulo UART igualmente tiene lugar un reset del buffer pasándolo a un estado vacío. Cualquier dato ó carácter que estuviera almacenado en el buffer se perderá e igualmente el contador de tasa de baudios irá a su estado común de reset, perdiendo toda configuración anterior.

Todas las banderas de error y de estado asociadas con el módulo UART son llevadas a reset cuando el módulo se deshabilita. Los bits URXDA, OERR, FERR, PERR, UTXEN, UTXBRK y UTXBF son llevados a “0”, mientras que los bits RIDLE y TRMT son llevados a “1”. Otros bits de control que participan en el funcionamiento del módulo, incluyendo los bits ADDEN, URXISEL<1:0>, UTXISEL, como también los registros UxMODE y UxBRG, no son afectados por una operación de reset.

Si Limpiamos (colocando un “0”) el bit UARTEN mientras que el módulo UART está funcionando, podemos tener un aborto en todas las transmisiones y recepciones pendientes y la operación de reset tendrá lugar como se explico anteriormente afectando de tal manera al módulo. Cuando se re-habilita el módulo UART se restablece con la misma configuración.

### Entradas y Salidas Alternativas del Módulo UART:

La función de alternar E/S se habilita cuando se coloca el bit ALTIO en “1” (UxMODE<10>). Si el bit ALTIO es igual a “1”, los pines UxATX y UxARX (pin de

transmisión alterna y pin de recepción alterna) son usados por el módulo UART en vez de los pines UxTX y UxRX. Si por el contrario el bit ALTIO es igual a "0", los pines UxTX y UxRX son usados como pines principales por el módulo UART.

### **Configurar los bits para la selección de datos, paridad y paradas:**

Los bits de control PDSEL<1:0> ubicados en el registro UxMODE son usados para seleccionar el tamaño de los datos y la paridad usada en la transmisión. El tamaño de los datos puede ser el mismo de 8 bits con paridad par, impar o sin paridad; ó 9 bits sin paridad. El bit de STSEL determina cuando se usa uno o dos bits de PARADA, necesarios durante la transmisión de datos. La configuración "por defecto" (en frío) de la UART es de 8 bits, sin paridad y con un bit de parada (típicamente representado como 8, N, 1).

## **TRANSMISIÓN DE DATOS**

Transmitiendo en el modo de transmisión para 8 bits de datos tenemos:

### 1. Configuración de la UART:

Primero, los bits de tamaño de datos, paridad y cantidad de bits de parada tienen que seleccionarse. Luego, los bits para habilitar las interrupciones de transmisión y recepción deben configurarse en los registros UxMODE y UxSTA. Igualmente es pertinente configurar el baud rate adecuado escribiendo su valor en el registro UxBRG.

2. Después hay que habilitar la UART colocando en "1" el bit UARTEN (UxMODE<15>).

3. Igualmente colocar un "1" en el bit de UTXEN (UxSTA<10>), de éste modo se habilitará la transmisión de datos.

4. Después de haber realizado la configuración anterior hay que escribir el byte que se desea transmitir en el byte inferior del registro UxTXREG. Ese valor será transmitido inmediatamente al registro de desplazamiento de transmisión (UxTSR) y de ese modo el chorro de datos seriales comenzará durante los próximas sucesiones del reloj de baud rate en sus flancos de subida. Alternativamente a la transmisión, el bit de datos puede ser modificado mientras que el bit se encuentre en "0" (UTXEN=0), seguidamente, el usuario puede escribir en el mismo registro un "1" según la pertinencia de la comunicación deseada. Sin embargo esto puede causar que la transmisión del stream de datos serial comience inmediatamente, ya que esto ha de tener efecto cuando la instrucción se configure y debido a que el reloj de baudios comienza su cuenta desde el estado "0" de dicho bit.

5. Una interrupción de transmisión ocurrirá dependiendo el valor del bit de control de interrupciones UTXISEL(UxSTA<15>).

## **TRANSMITIENDO EN EL MODO DE 9 BIT`S DE DATOS:**

La secuencia de pasos involucrada en éste proceso de transmisión es similar a la usada anteriormente, sin embargo hace excepción en que una palabra de 16 bit's de datos (los cuales los 7 bits superiores siempre son llevados a "0" para completar los 9 bit's de datos) tiene que ser escrita antes en el registro UxTXREG.

## **BUFFER DE TRANSMISIÓN (UxTXB):**

El buffer de transmisión es de 9 bits de extensión y de 4 caracteres de profundidad. Incluyendo el registro de desplazamiento de transmisión (UxTSR), el usuario puede tener efectivamente un buffer FIFO (First In First Out) con profundidad de 5. El bit de estado UTXBF (UxSTA<9>) indica cuando el buffer de transmisión esta lleno. Si el programador intenta escribir en un buffer completamente lleno, los nuevos datos no serán aceptados en el FIFO, y ningún corrimiento de datos nuevos ocurrirá en el buffer ya que es una operación ilegal; Esta situación habilita la recuperación desde una condición de Sobrecarga del buffer. El buffer FIFO se resetea durante cualquier "Reset" que tenga lugar en el dispositivo en general pero no es afectado cuando el dispositivo opera en un modo de bajo consumo.

### **INTERRUPCIÓN DE TRANSMISIÓN:**

La bandera o señalador de interrupción de transmisión (U1TXIF ó U2TXIF) está ubicada en el correspondiente registro que maneja las banderas de interrupción de todo el dispositivo. EL transmisor genera una opción para llevar a "1" el bit UxTXIF. La condición para generar interrupciones depende de la configuración previa del bit de control UTXISEL:

- Si UTXISEL = 0, una interrupción es generada cuando una palabra es transferida desde el buffer de transmisión hasta el registro de desplazamiento de transmisión (UxTSR); esto implica que el buffer de transmisión tiene como mucho una palabra vacía que transmitir.
- Si UTXISEL = 1, una interrupción es generada cuando una palabra es transferida desde el buffer de transmisión hasta el registro de desplazamiento de transmisión (UxTSR) y el buffer de transmisión está vacío. Al Intercambiar los dos modos de interrupción durante una operación de transmisión es posible obtener más flexibilidad a la hora de transmitir datos.

### **PAUSA DE TRANSMISIÓN:**

Colocando en "1" el bit UTXBRK (UxSTA<11>) se puede hacer que la línea de transmisión UxTX sea conducida hacia un "0" lógico. El bit de break ó UTXBRK sobrescribe toda actividad de transmisión. Desde luego, el programador generalmente debe esperar a que el transmisor pase a un estado de receso, parada ó ausencia antes de llevar a un "1" lógico el bit de UTXBRK. Para enviar un carácter de pausa, el bit UTXBRK debe colocarse en "1" por software y debe recordarse su estado en el mismo programa por un mínimo de 13 ciclos de reloj en baudios. Luego de ello, el bit UTXBRK es llevado a "0" por software para generar los bits de parada correspondientes. El programador debe esperar por lo menos la duración de 1 ó dos ciclos de reloj en baudios para asegurarse que el bit de parada ó los bit's de parada sean válidos antes de recargar el registro UxTXB y también antes comenzar cualquier otra actividad de transmisión. La transmisión de un carácter de pausa ó espera no genera una interrupción de transmisión.

### **RECEPCIÓN DATOS**

#### **Recepción de datos en los modos de 8 o 9 bits:**

Los siguientes pasos deben seguirse mientras se reciben datos en formatos tanto de 8 como de 9 bits:

1. Activar el módulo UART como vimos antes.
2. Habilitar el módulo UART.
3. Una interrupción de recepción se generará cuando uno o más palabras de datos sean recibidas, dependiendo cómo se hallan especificado las propiedades de interrupción por los bits URXISEL del registro UxSTA <7:6>.
4. Leer los datos recibidos desde el registro UxRXREG. El acto de leer los datos de ese registro consiste simplemente en mover la siguiente palabra en la parte alta de la pila FIFO de recepción, y los valores PEERR y FERR se actualizarán.

#### **BUFFER DE RECEPCIÓN (UxRXB):**

El buffer de recepción tiene una profundidad de 4 palabras. Incluyendo el registro de desplazamiento de recepción (UxRSR), el programador efectivamente tiene 5 palabras de profundidad en la pila FIFO del buffer.

El bit URXDA del registro UxSTA (<0>) = 1 indica que el buffer de recepción tiene datos recibidos disponibles para su lectura. En el caso contrario ("0") implica que el buffer de recepción o entrada de datos está vacío. Si se percibe la lectura de un buffer

vacio, los valores antiguos en el buffer deberán ser leídos y por consiguiente ningún corrimiento de datos tendrá lugar en el FIFO.

### INTERRUPCIÓN DE RECEPCIÓN:

Como vimos en la transmisión, la recepción es similar. La bandera de interrupción de recepción (U1RXIF ó U2RXIF) puede leerse desde el correspondiente registro que maneja las interrupciones. La bandera se coloca en "1" por una condición de recepción generada por el receptor. La condición para colocar dicha bandera a "1" es que depende de la configuración inicial adoptada por los bits de control URXISEL <1:0> (UxSTA<7:6>).

- Si URXISEL <1:0> = 00 ó 01, una interrupción es generada todo el tiempo dando lugar a la transferencia de una palabra de datos desde el registro de desplazamiento de recepción (UxRSR) hasta el buffer de recepción. Pueden haber uno o más caracteres en el buffer de recepción.
- Si URXISEL <1:0> = 10, una interrupción es generada cuando una palabra es transferida desde el registro de desplazamiento de recepción (UxRSR) hasta el buffer de recepción, el cual como resultado de la transferencia, contendrá 3 caracteres.
- Si URXISEL <1:0> = 11, la interrupción se activa cuando la palabra es transferida desde el registro de desplazamiento de recepción (UxRSR) hasta el buffer de recepción, el cual como resultado contiene 4 caracteres (por ejemplo se llena).

### GENERADOR DE BAUD RATE:

El módulo UART tiene un generador de Baud Rate de 16 bit's para permitir una máxima flexibilidad en dicha generación. El registro de generación de Baud Rate (UxBRG) es escribible y permite su lectura. El baud rate se calcula de la siguiente manera:

BRG = Valor de 16 bits que mantiene el registro UxBRG. (Se encuentra entre 0 y 65535).

FCY = Frecuencia de Reloj para cada instrucción (1/TCY).

La Ecuación para hallar el Baud Rate la obtenemos de la siguiente manera:

$$\text{Baud Rate} = \frac{F_{CY}}{16 * (BRG + 1)}$$

Desde luego el máximo baud rate posible es  $\frac{F_{CY}}{16}$  (si BRG=0), y el mínimo baud rate

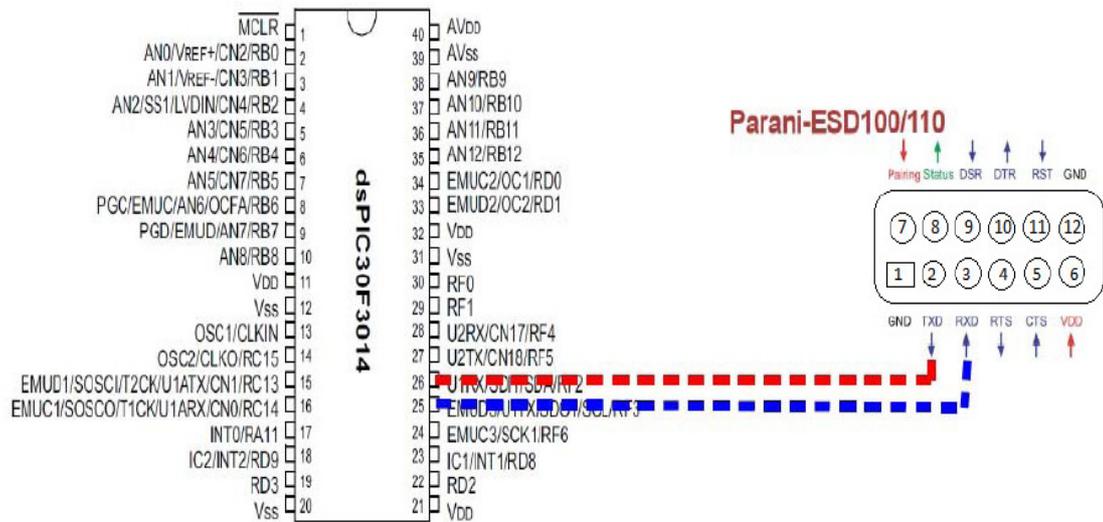
posible es  $\frac{F_{CY}}{16 * 65536}$

.Con un generador de baud rate a su máxima capacidad ó a 30 MIP's de operación, el mínimo baud rate alcanzable es 28.5 bps.

### Configuración Automática del Baud Rate:

Para permitirle al sistema determinar automáticamente el baud rate gracias a los caracteres recibidos, la entrada puede ser opcionalmente conectada con el modo de captura (IC1 para la UART1, IC2 para la UART2). Y desde luego para habilitar este modo, el programador tiene que programar el modo de captura de entrada para hacer autodetección de flancos tanto de subida como de bajada del bit de inicio.

Dadas las anteriores indicaciones, el programa se construye en base a:



Conexión Física de los módulos (Sólo se toma en cuenta la transmisión asíncrona):

- Activación y habilitación del módulo UART
- Configuración del Baud Rate (Se puede trabajar con 9600 bps).
- Secuencia de de Iniciación en envío de secuencias de palabras con el contenido de comandos AT de configuración del dispositivo de radio Bluetooth.
- Habilitación de recepción.
- Estableción de sesión.
- Envío de datos.

## BIBLIOGRAFIA

<http://en.wikipedia.org/wiki/Bluetooth>  
<http://spanish.bluetooth.com/Bluetooth/Technology/>  
[http://html.rincondelvago.com/bluetooth\\_1.html](http://html.rincondelvago.com/bluetooth_1.html)  
<http://upcommons.upc.edu/pfc/bitstream/2099.1/3607/2/36416-2.pdf>  
<http://www.microchip.com/wwwproducts/Devices.aspx?dDocName=en010345>  
<http://www.dspace.espol.edu.ec/bitstream/123456789/7825/1/D-39558.pdf>  
<http://server-die.alc.upv.es/>  
<http://server-die.alc.upv.es/assignaturas/PAEEES/2008-09/A08/Index.htm>  
[http://es.wikipedia.org/wiki/Protocolos\\_Bluetooth](http://es.wikipedia.org/wiki/Protocolos_Bluetooth)  
Tecnología bluetooth /Nathan j Muller  
<http://ww1.microchip.com/downloads/en/devicedoc/70138c.pdf>