

UNIVERSIDAD POLITECNICA DE VALENCIA

ESCUELA POLITECNICA SUPERIOR DE GANDIA

Grado en Ing. Sist. de Telecom., Sonido e Imagen

---



UNIVERSIDAD  
POLITECNICA  
DE VALENCIA



ESCUELA POLITECNICA  
SUPERIOR DE GANDIA

# “Comunicación Bluetooth entre microprocesador ATMEL y un pc”

**TRABAJO FINAL DE GRADO**

Autor/a:

**F. Borja Gimeno Belda**

Tutor/a:

**María Asunción Pérez Pascual**

**GANDIA, 2015**

# Comunicación Bluetooth entre microprocesador ATMELE y un pc

[Dpto. de Electrónica](#)



Escuela Politécnica Superior de Gandía

## Trabajo final de grado

---

### Contenido

<b>1.</b>	<b>INTRODUCCIÓN .....</b>	<b>3</b>
<b>2.</b>	<b>ESPECIFICACIONES DEL SISTEMA .....</b>	<b>4</b>
<b>3.</b>	<b>DIAGRAMA BLOQUES DEL SISTEMA .....</b>	<b>5</b>
<b>4.</b>	<b>DISEÑO CIRCUITOS.....</b>	<b>6</b>
4.1.	CONECTORES PROGRAMACIÓN / COMUNICACIÓN .....	6
4.2.	MODULO BLUETOOTH.....	11
4.2.1.	UART .....	13
4.2.2.	BLUETOOTH.....	13
4.2.3.	RS-232.....	14
<b>5.</b>	<b>CONSTRUCCIÓN PROTOTIPOS.....</b>	<b>15</b>
5.1.	DISEÑO PROTOTIPO COMUNICACIÓN.....	15
5.2.	DISEÑO PROTOTIPO REAL .....	16
<b>6.</b>	<b>PRUEBA PROTOTIPOS.....</b>	<b>18</b>
6.1.	TESTEO PROTOTIPO COMUNICACIÓN.....	18
6.2.	TESTEO PROTOTIPO REAL .....	23
6.3.	MEJORAS AÑADIDAS A LA IMPLEMENTACIÓN.....	24
6.3.1.	SOLUCIÓN BACK-UP .....	25
<b>7.</b>	<b>DISEÑO FINAL .....</b>	<b>31</b>
<b>8.</b>	<b>CONCLUSIONES Y TRABAJO FUTURO .....</b>	<b>34</b>
<b>9.</b>	<b>ANEXOS .....</b>	<b>35</b>
9.1.	ART FILM – TOP .....	35
9.2.	ART FILM – BOTTOM.....	36
9.3.	ART FILM - SOLDERMASK TOP.....	37
9.4.	ART FILM - PASTEMASK TOP .....	38
9.5.	ART FILM - SILKSCREEN TOP.....	39
9.6.	ART FILM - ASSEMBLY TOP .....	40
9.7.	ART FILM - SOLDERMASK BOTTOM .....	41
9.8.	ART FILM - BOARD OUTLINE .....	42
9.9.	MAIN.C PRUEBA PROTOTIPO REAL.....	43
9.10.	BILL OF MATERIALS.....	44

## Objetivo del TFG

El trabajo tiene por objetivo la implementación de un sistema de comunicación inalámbrico entre distintos dispositivos, ya que hasta este momento se estaba haciendo mediante cables. Del resultado del trabajo realizado para la implementación del sistema, se ha redactado un documento técnico donde se recogen las especificaciones y procedimientos desarrollados para la consecución exitosa del sistema.

Además al tratarse del trabajo final de un ciclo académico, se ha intentado utilizar conceptos y conocimientos adquiridos durante el transcurso de éste. También se ha contado con la motivación de tratarse de un proyecto dentro de una empresa del sector tecnológico.

*The target of document is the implementation of a wireless communication system among different devices. Until this moment, it was been done through cables. From the result of the document done for the implementation of the system, a technician paper has been written, which collects the developing specifications and the developing process for the successful achievement of the system.*

*Being a final document of an academic cycle, it was tried to employ concepts and knowledge acquired during the course. Moreover, the motivation of participating in a project for a company in the technology sector was a plus.*

Palabras clave: *Bluetooth, ATMEL, Cadence, PCB, Wireless.*

# 1. Introducción

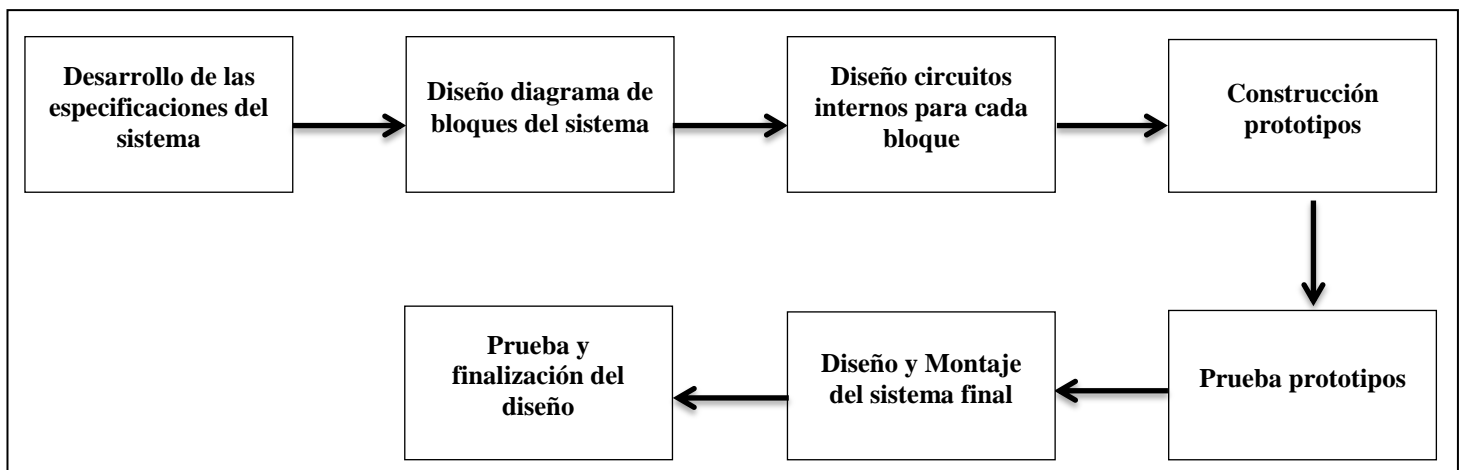
Este trabajo final de grado trata sobre el diseño de un circuito impreso (*PCB*), para la comunicación bluetooth entre distintos microprocesadores ATMEL [F] y un ordenador.

El trabajo está englobado dentro del proyecto que está desarrollando la empresa *Senia Technologies [A]*, este proyecto versa en el desarrollo de microelectrónica por medio de semiconductores, para la construcción de chips que puedan implementar pantallas led extremadamente finas, pudiendo llegar a ser incluso flexibles.

El objetivo del circuito desarrollado es la eliminación del cableado en la conexión entre el microprocesador y el ordenador, la finalidad de dicha conexión es la realización de pruebas en los circuitos donde se encuentran los microprocesadores, debido a que estos envían informes al ordenador dependiendo de su estado, o de la función que están llevando a cabo.

Se dispone de numerosos módulos led, donde cada uno de ellos cuenta con un microprocesador, todos ellos crearán una conexión bluetooth con el ordenador, y por medio del protocolo bluetooth enviarán los informes al ordenador, finalmente éste mostrará los mensajes por el monitor del puerto serie.

Para el diseño del circuito electrónico se ha seguido el siguiente diagrama de bloques, por ello el trabajo está estructurado de la siguiente manera.



Esquema 1[Diagrama bloques para diseño del circuito]

## 2. Especificaciones del sistema

Las especificaciones técnicas vienen dadas por los siguientes requisitos:

- Conexión bluetooth.

Es la especificación más relevante, ya que satisface el principal objetivo del circuito, que no es otro que la eliminación de cables en las pruebas de los módulos led a testear.

- Dimensiones.

El circuito a diseñar estará emplazado en un encapsulado de dimensiones acotadas, por ello el circuito impreso debe adaptarse a ellas.

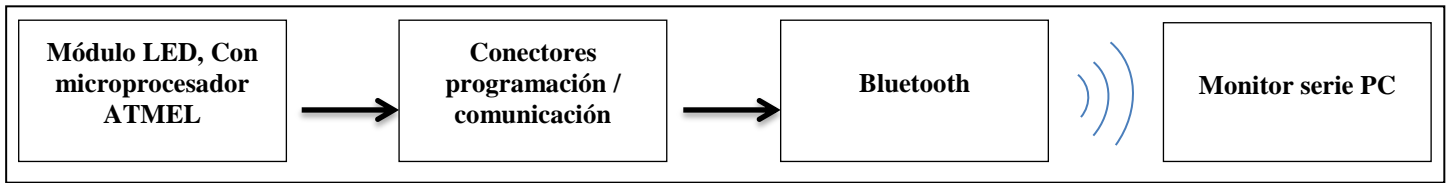
- Diferentes conectores para los módulos led.

Se han empleado distintos conectores para la programación, en cada uno de los módulos led que contienen los microprocesadores, para así experimentar los beneficios e inconvenientes de cada uno de ellos. Por ello se precisará de diferentes conectores, para la conexión entre el programador y los microprocesadores emplazados en los módulos. Por medio de dichos conectores también serán transmitidos los mensajes que se quieren enviar desde el microprocesador ATMEL al pc, por medio de bluetooth.

- Fuentes de alimentación.

La fuente de alimentación será suministrada a través del módulo led, tendrá una tensión de 3.3V

### 3. Diagrama bloques del sistema



Esquema 2[Diagrama bloques del sistema]

El primer bloque no será materia de diseño en este trabajo final de grado, se incluye dentro del diagrama para una mayor comprensión del sistema a diseñar. De este primer bloque saldrá un cable para la conexión con el módulo bluetooth, como ya se ha comentado anteriormente, se necesita de un intermediario para que adapte el tipo de conector, ya que no todos los módulos led dispondrán del mismo.

Una vez los datos lleguen al bloque bluetooth serán enviados al pc, que contará con un módulo bluetooth integrado.

Por tanto se va a implementar el diseño de dos bloques, es decir, de dos circuitos electrónicos por separado. El primero de ellos se trata del bloque conectores programación / comunicación, que nos servirá para adaptar el tipo de conector en el módulo led, para su conexión al bloque bluetooth. Por otra parte se diseñará el circuito electrónico que contendrá el módulo bluetooth.

## 4. Diseño circuitos

### 4.1. Conectores programación / comunicación

En este apartado se van a presentar los distintos conectores [B], que van a formar este primer circuito, en la tabla 1 podemos observar las señales que deberán tener asignadas los contactos, en cada uno de los diferentes conectores.

Contacto	Descripción
VDIG	Se trata de la referencia de voltaje que se utiliza para crear el nivel lógico de referencia (VDIG/2) para la entrada del programador, también proporcionará alimentación para los componentes del módulo bluetooth una vez la programación del micro controlador haya sido realizada. En nuestro caso se tratará de una tensión de 3.3 voltios.
TCK	Es la señal que proporciona el programador para sincronizar con el micro controlador.
TMS	Se trata de la señal de control para el controlador TAP (Test Access Point).
TDO	Es la señal de datos del micro controlador para el programador.
NRST	Señal de reinicio a nivel lógico bajo.
DRXD	Esta señal nos proporciona la comunicación entre el modulo Bluetooth y los módulos led, por esta línea se recibirán mensajes de control.
DTXD	Al igual que la señal anterior nos proporciona la comunicación con el modulo Bluetooth, en este caso por esta conexión se enviarán los informes, que deberán ser mostrados por el monitor del Puerto serie en el pc.
TDI	Se trata de la señal de datos del programador al procesador.

Tabla 1 [Señales para programación micro ATMEL]

En las siguientes tablas se muestran una descripción física de los conectores, así como sus características y la referencia que le ha sido asignada en el esquema del circuito:

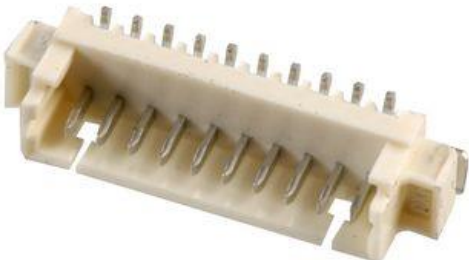

Conector "J2"		
Fotografía	Pinout	
	 <p>VDIG GND TCK TMS TDO NRST GND DRXD DTXD TDI</p>	Espaciado entre contactos de 1.25 mm
		Montaje de superficie
		Posición vertical
		10 Contactos

Tabla 2 [Conector J2]

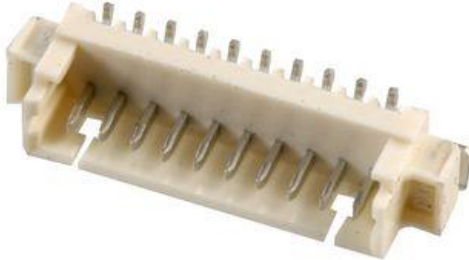

Conector "J3"		
Fotografía	Pinout	
	 <p>DTXD NC VDIG NRST TDI TDO TMS TCK GND</p>	Espaciado entre contactos de 1.25 mm
		Montaje de superficie
		Posición vertical
		9 Contactos

Tabla 3[Conector J3]





Conector "J4"		
Fotografía	Pinout	
		Espaciado entre contactos de 0.8 mm
		Montaje de superficie
		Posición horizontal
		8 Contactos

Tabla 4 [Conector J4]

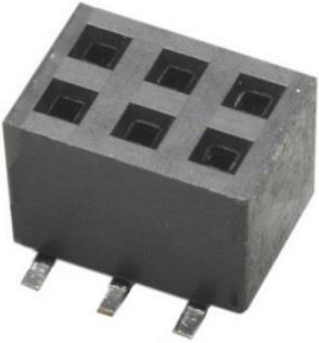
Conector "J5" / "J6"														
Fotografía	Pinout													
	<table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>VDIG</td> <td>GND</td> </tr> <tr> <td>TDO</td> <td>NRST</td> </tr> <tr> <td colspan="2"> </td> </tr> <tr> <td>TCK</td> <td>TDI</td> </tr> <tr> <td>TMS</td> <td>DTXD</td> </tr> <tr> <td>NC</td> <td>NC</td> </tr> </table>	VDIG	GND	TDO	NRST			TCK	TDI	TMS	DTXD	NC	NC	Espaciado entre contactos de 2 mm
		VDIG	GND											
		TDO	NRST											
TCK	TDI													
TMS	DTXD													
NC	NC													
Montaje de superficie, con agujero pasante														
Posición vertical														
6 Contactos + 4 Contactos														

Tabla 5 [Conector J5 y J6]

Para la conexión entre el bloque buetooth y el bloque de conectores programación/comunicación, se ha utilizado un conector de 14 contactos, por el cual se establecerá la comunicación entre el módulo led y el bluetooth, también se suministrarán los 3.3 voltios para la alimentación del módulo bluetooth y demás componentes pasivos presentes en el circuito.

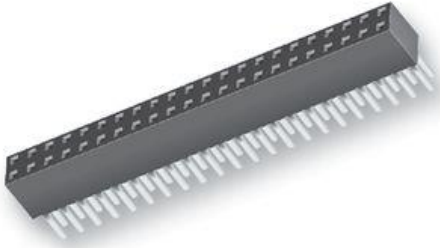
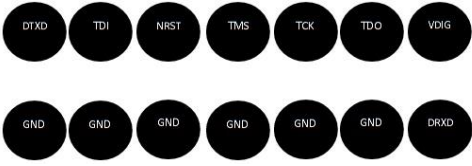
Conector "J1"		
Fotografía	Pinout	
		Espaciado entre contactos de 2.54 mm
		Agujero pasante
		Posición vertical
		14 Contactos

Tabla 6 [Conector J1]

Finalmente se expone el esquemático para el circuito de conectores programación/comunicación, en este circuito no se precisa ningún tipo de componente discreto, ya que simplemente nos servirá para el transporte de las señales anteriormente descritas, con la interconexión de éstas, el circuito deseado estará finalizado.

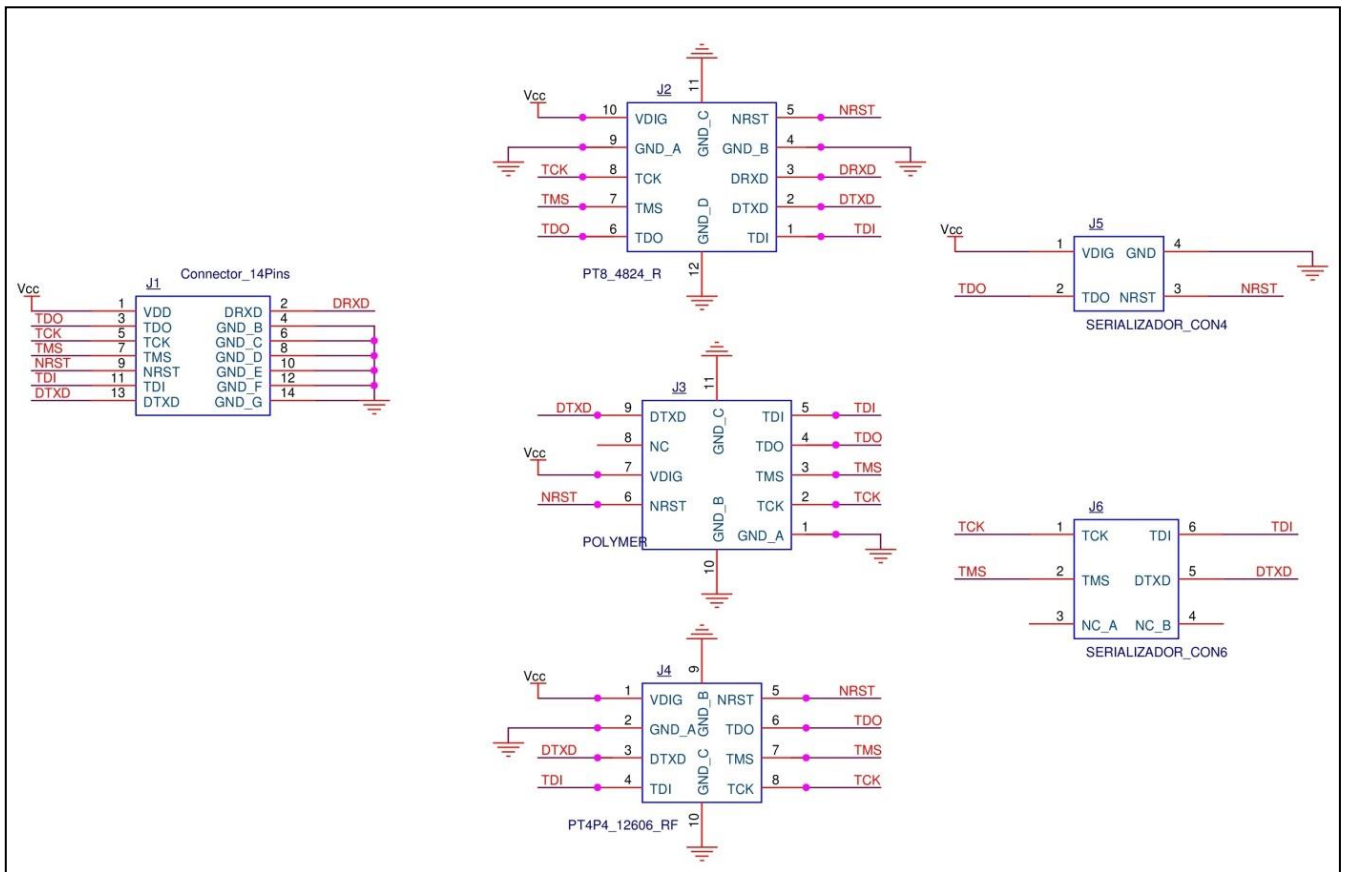


Imagen 1 [Esquemático circuito conectores programación/comunicación]

## 4.2. Modulo bluetooth

Para el diseño del circuito bluetooth [D], se ha optado por la elección de un módulo de la compañía *Microchip* [G], se trata del módulo bluetooth RN-42. Este módulo cuenta con una antena integrada de dimensiones reducidas, trabaja con la clase 2 del protocolo bluetooth, pudiendo llegar a los 3 Mbps de velocidad de transmisión, y a un alcance de 20 metros.

Debido a que primero se debe tener una buena comprensión del funcionamiento del módulo bluetooth a utilizar, por el momento no se va a exponer ningún esquemático del circuito a implementar, primero se van a desarrollar dos prototipos para experimentar con el integrado.

Un primer prototipo se le denominara de comunicación, y se intentará que sea lo más sencillo posible, se tendrá por objetivo poder entrar por medio de comandos al menú de configuración del dispositivo, y establecer una conexión bluetooth con un ordenador así como establecer una comunicación.

El segundo prototipo ya será una aproximación real al circuito final, se diseñará con la finalidad de poder conectarse mediante cable, al módulo led que contiene el microprocesador ATMEL y que éste envíe por medio la UART mensajes al módulo RN-41, finalmente éste por medio bluetooth enviará al ordenador los informes, que serán mostrados por medio del monitor serie.

El modelo RN-42 será utilizado para el circuito final, para los prototipos se utilizará el módulo de la serie RN-41, éste contiene un chip que implementa la comunicación del protocolo RS-232, de esta forma podremos emular el módulo led mediante un ordenador convencional, para así realizar las pruebas de creación de puertos virtuales y la comunicación mediante el monitor del puerto serie.

El RN-41 coge los datos de entrada y salida de la UART y los traduce al estándar RS-232, de esta forma serán enviados por medio de un conector DB-9 al ordenador por el puerto serie.

En las siguientes tablas se exponen los símbolos para el esquemático, y una descripción de los terminales que son de interés para el diseño del circuito.

Símbolo RN-42	Numero terminal	Nombre terminal	Descripción
	5	RESET	Reinicio.
	1,12,28 y 29	GND	Conexión a la masa del circuito.
	11	VDD	Alimentación del módulo a 3.3 V.
	13	UART_RX	Mensajes recibidos para la UART.
	14	UART_TX	Mensajes enviados desde la UART.
	15	UART_RTS	Señalización de la UART.
	16	UART_CTS	Señalización de la UART.
	19	GPIO2	Señalización de la conexión.
	21	GPIO5	Estado de la conexión.
	22	GPIO4	Reinicio de fábrica.

Tabla 7[Símbolo RN-42]

Símbolo RN-41	Numero terminal	Nombre terminal	Descripción
	4	RX-RS232	Mensajes recibidos desde el puerto serie.
	5	TX-RS232	Mensajes enviados al puerto serie.
	6	RTS-RS232	Señalización RS 232.
	7	CTS-RS232	Señalización RS 232.
	10	PRS	Alimentación para el chip RS-232.
	11	VCC	Alimentación del módulo a 3.3 V.
	12	GND	Conexión a la masa del circuito.

Tabla 8[Símbolo RN-41]

En los siguientes apartados se presenta una descripción de los protocolos de comunicación utilizados en el proyecto.

### 4.2.1. UART

En este caso no utilizaremos los puertos de señalización CTS y RTS, por lo que se trata de una transmisión-recepción asíncrona (UART). Ya que los datos fluirán en una sola dirección, no se precisa de dicha sincronización para que no se produzcan colisiones desde el módulo led hacia el ordenador. De lo contrario se denominaría USART donde si hay cierta sincronización.

El objetivo de la UART [L] es tomar bytes de datos y descomponerlos en bits de forma secuencial para su emisión, una vez recibidos los bits volverán a ser ensamblados en bytes, de esta forma se consigue la transmisión de datos por un solo hilo. También son añadidos dos bits de inicio y fin de palabra, que indicaran al receptor donde empieza y acaba cada palabra de 8 bits para que este pueda conformar el byte.

Bit Inicio	Data 0	Data 1	Data 2	Data 3	Data 4	Data 5	Data 6	Data 7	Bit Parada
---------------	-----------	-----------	-----------	-----------	-----------	-----------	-----------	-----------	---------------

Imagen 2[Trama de datos UART]

Para nuestro proyecto sólo se reciben mensajes desde el módulo led, por lo que se utilizara el terminal de RX, el terminal TX se contempla para futuras aplicaciones. Una vez el módulo bluetooth reciba los datos a través de la UART, creará el paquete de datos que será enviado mediante bluetooth al ordenador.

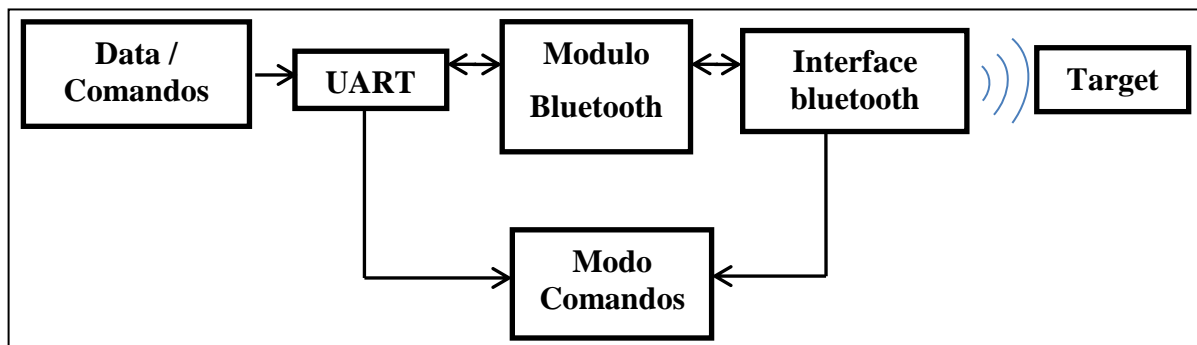
### 4.2.2. Bluetooth

Se trata de una norma para crear redes inalámbricas de área personal (WPAN), utiliza enlaces por radiofrecuencia en la banda ISM de los 2,4 GHz, en nuestro caso los módulos utilizados son de clase 2, que proporciona una potencia máxima de transmisión de 2.5 mW (4dBm ).

El protocolo actúa dentro del rango de frecuencias de 2,4 GHz a 2,48 GHz, donde tiene la posibilidad de realizar 79 saltos de 1MHz de intervalo, lo que nos asegura robustez y fiabilidad entre las comunicaciones.

En los módulos RN-42 y RN-41 a nivel de hardware, se cuenta con un dispositivo de radio encargado de modular en GFSK y transmitir la señal, y un controlador digital implementado por un procesador de señales digitales, este se encargará de construir las instrucciones del protocolo bluetooth.

A nivel de software los módulos tienen dos tipos de funcionamiento, modo datos y modo comandos. En modo comandos no existe ninguna comunicación sobre bluetooth, sino que con este modo se consigue la configuración del módulo a través de comandos, por ejemplo se estipula el nombre del dispositivo y se pueden revisar parámetros del mismo. En el modo datos el módulo actúa como una tubería, cuando el módulo recibe un mensaje por el contacto TX de la UART, éste coge dicha información y crea un paquete bluetooth, posteriormente el paquete es enviado a través de la conexión establecida. En modo datos, el proceso de envío y recepción de información por medio de bluetooth, es totalmente transparente al usuario.



Esquema 3[Diagrama Hardware módulo bluetooth]

### 4.2.3. RS-232

El estándar RS-232 [1] será utilizado en el prototipo. Como antes ya se ha expuesto, el módulo RN-41 cuenta con un circuito integrado que convierte las señales del puerto RS-232, que pueden alcanzar  $\pm 15V$ , en señales lógicas CMOS que tienen un nivel entre 3V y 15V. De esta forma se emulará el módulo led mediante un ordenador. Por medio del monitor serie se escribirán mensajes que serán enviados por bluetooth al ordenador que actúa como oyente.

## 5. Construcción prototipos

### 5.1. Diseño prototipo comunicación

Para la construcción del prototipo se ha utilizado el módulo bluetooth *RN-41*, que como anteriormente ha sido explicado, cuenta con un chip que nos realizara la translación entre niveles lógicos *CMOS* y niveles *RS-232*.

Con este prototipo se pretende testear la comunicación bluetooth por medio del módulo, así como la configuración del módulo por medio de bluetooth, a continuación se muestra el esquemático del montaje a realizar.

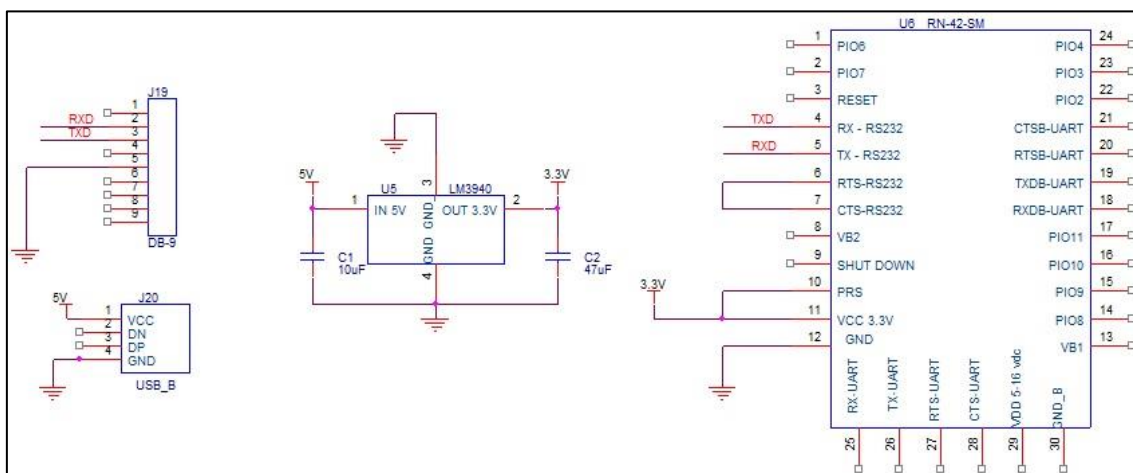


Imagen 3[Esquemático prototipo comunicación]

La alimentación para el circuito vendrá dada por medio de un conector *USB*, de éste sólo se utilizarán los pines de *VCC* y *GND*, ya que la comunicación no se realizará mediante *USB*. Debido a que el protocolo *USB* proporciona una tensión de 5V y para el módulo se precisan 3.3V se ha hecho necesario la utilización del regulador lineal *LM3940*, que nos convierta de 5V a 3.3V.

La comunicación se hará por medio de un conector *DB-9*, que irá conectado a los pines de *RX-RS232* y *TX-RS232* del módulo bluetooth, entre las señales *RTS-RS232* y *CTS-RS232* se ha provocado un cortocircuito ya que la comunicación es asíncrona.

Para finalizar se ha proporcionado alimentación al chip *RS-232* cortocircuitando la pata 10 del módulo *RN-41* a 3.3V. Se ha reutilizado un circuito impreso que ya contaba con un conector *DB-9* instalado. A continuación se muestran las fotografías del prototipo finalizado.



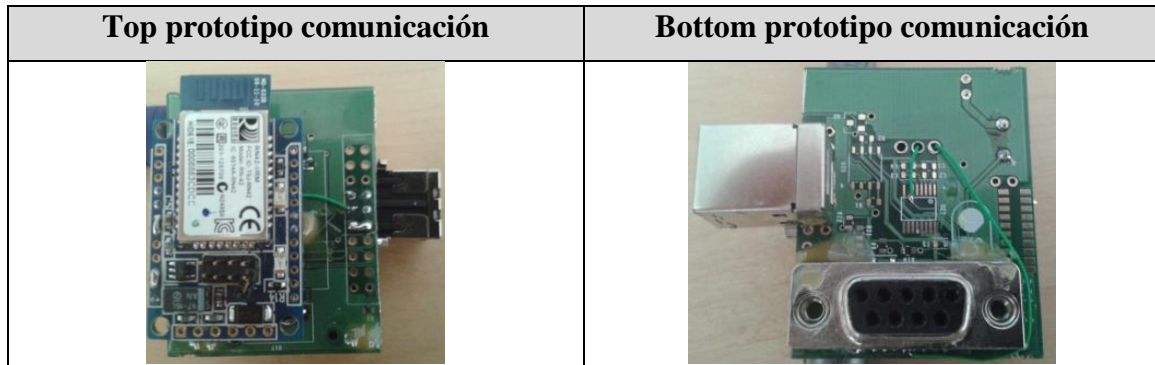


Tabla 9[Prototipo comunicación]

## 5.2. Diseño prototipo real

En la siguiente imagen se muestra el esquemático a montar para la construcción de este prototipo, como se puede observar la comunicación se realiza a través de los pines de la *UART*, no por los pines del chip *RS-232* como se había hecho con el anterior prototipo, esto se debe a que ahora la comunicación es directa con el microprocesador y no se precisa de la conversión de niveles lógicos *CMOS* a *RS-232*.

Se va a utilizar un conector de 10 pines, de los cuales sólo cuatro son de nuestro interés, estos pines son los dos de alimentación y de comunicación, el voltaje vendrá suministrado a través de la tarjeta que contenga el microprocesador. Ha sido añadido un condensador de desacoplo para filtrar altas frecuencias en la alimentación continua.

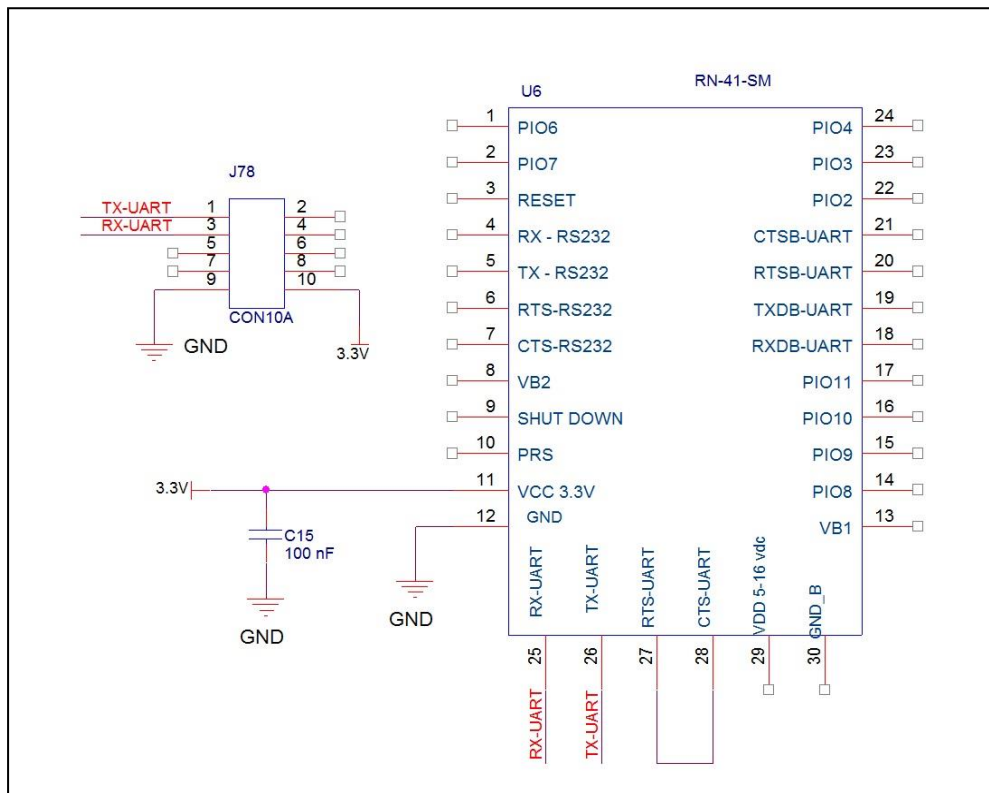


Imagen 4[Esquemático prototipo real]

Para la utilización de los pines 25, 26, 27 y 28 se ha debido quitar dos resistencias existentes en la parte inferior del módulo bluetooth, de este modo se ha desconectado el chip RS-232, habilitando la comunicación por medio de la UART.

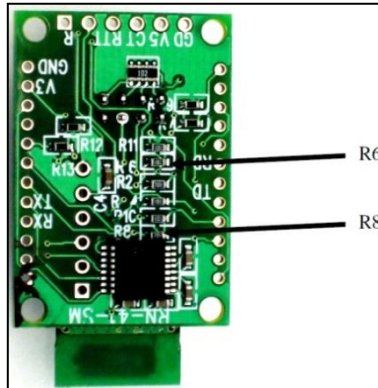


Imagen 5[Resistencias retiradas para prototipo real]

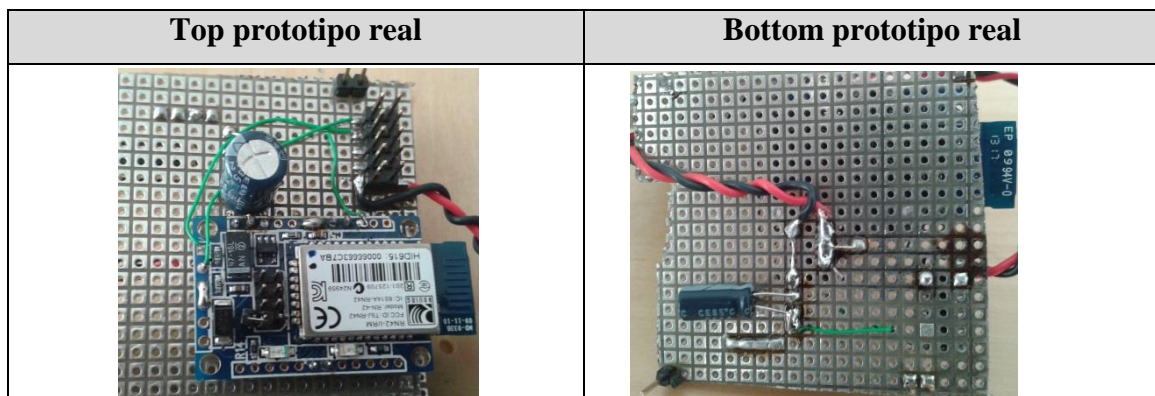
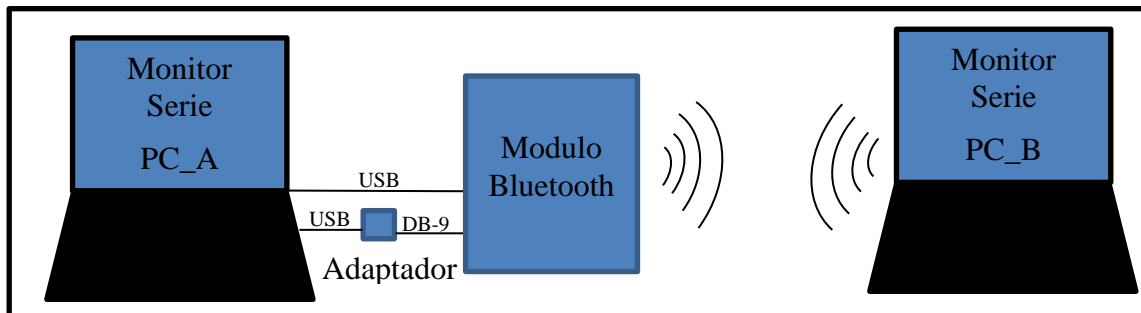


Tabla 10 [Prototipo real]

## 6. Prueba prototipos

### 6.1. Testeo prototipo comunicación

Para la realización de las pruebas se va a utilizar un ordenador (PC\_A) al cual se conectara el prototipo, como se muestra en el esquema 4, éste simulará el microprocesador que más adelante será el encargado de enviar los informes. Por otro lado, contamos con un ordenador con bluetooth integrado, que deberá coger el paquete bluetooth enviado desde el módulo *RN-41* y mostrarlo por el monitor serie. También se comprobará que desde el PC\_B se puede entrar en el modo comandos del *RN-41* y poder ajustar parámetros.



Esquema 4 [Montaje prueba prototipo comunicación]

Se precisa de un conversor de puerto serie a *USB*, ya que el PC\_A no dispone de puerto serie para su uso, por medio de este adaptador se conecta a un puerto *USB* el módulo y se crea un puerto de comunicaciones serie, en la siguientes capturas de pantalla se muestra el puerto serie creado y la configuración deseada en el monitor serie del PC\_A.

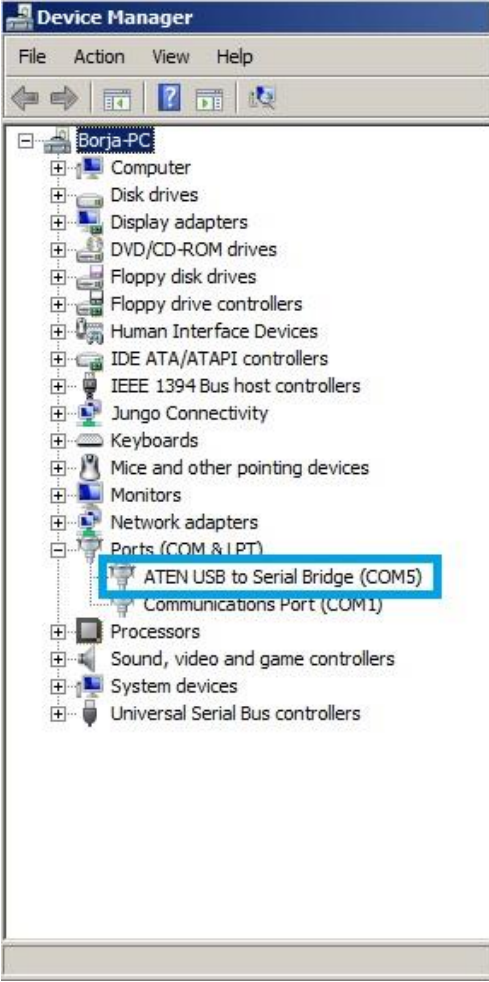

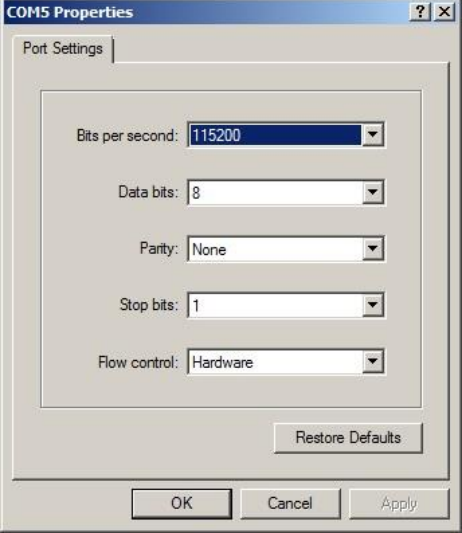
Puerto serie creado por el adaptador	Selección puerto en el monitor serie
 <p>The screenshot shows the Windows Device Manager window for 'Borja-PC'. The 'Ports (COM &amp; LPT)' category is expanded, and 'ATEN USB to Serial Bridge (COM5)' is highlighted with a blue selection box. Other visible ports include 'Communications Port (COM1)'.</p>	 <p>The screenshot shows the 'Connect To' dialog box. The 'Country/region' is set to 'Spain (34)'. The 'Connect using' dropdown menu is open, showing 'COM1', 'COM5' (which is selected), and 'TCP/IP (Winsock)'.</p>
Configuración monitor serie	
 <p>The screenshot shows the 'COM5 Properties' dialog box, 'Port Settings' tab. The configuration is as follows:         <ul style="list-style-type: none"> <li>Bits per second: 115200</li> <li>Data bits: 8</li> <li>Parity: None</li> <li>Stop bits: 1</li> <li>Flow control: Hardware</li> </ul>         Buttons for 'Restore Defaults', 'OK', 'Cancel', and 'Apply' are visible at the bottom.       </p>	

Tabla 11 [Configuración PC\_A prototipo comunicación]

Para realizar la conexión el modulo *RN-41* se dispone como esclavo, mientras que el PC\_B será configurado como maestro, pudiendo conectarse hasta con 7 dispositivos esclavos distintos, esto lo consigue mediante cambios rápidos entre esclavos

conectados, es decir sólo puede estar conectado con un dispositivo en un mismo instante de tiempo. La red creada entre los distintos esclavos y el maestro se denomina *Piconet*.

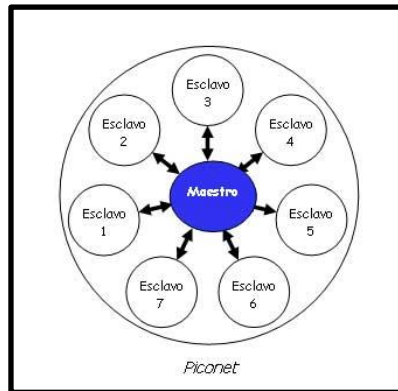


Imagen 6 [Esquema red Piconet]

Para el establecimiento de la conexión se ejecutan las siguientes acciones [C]:

- **Solicitud:** El maestro envía una solicitud al dispositivo y este responde con su dirección MAC.
- **Paginación:** El maestro selecciona una dirección, sincroniza la señal de reloj y frecuencia con el punto de acceso esclavo.
- **Descubrimiento del servicio:** se establece un enlace con el dispositivo esclavo, mediante el protocolo SDP (*Service Discovery Protocol*).
- **Creación de un canal:** se crea un canal de comunicación entre el maestro y el esclavo con el protocolo *L2CAP*. Mediante este protocolo se crea un canal adicional *RFCOMM*, que nos proporcionara un puerto serial virtual, para así poder conectar el monitor serie a este y visualizar la comunicación entre los dispositivos.
- **Emparejamiento:** El dispositivo esclavo envía un código PIN al maestro, que este debe guardar y si es necesario el usuario debe introducirlo para establecer la conexión con el esclavo.
- **Utilización del canal:** En este punto ya puede haber comunicación entre los dispositivos.

A continuación se muestran unas capturas de pantalla realizadas desde el PC\_B, donde se muestran algunos de los pasos anteriormente expuestos, no todos ellos van a ser mostrados, ya que se ha utilizado el asistente de *Windows* para conexiones bluetooth y algunos de los pasos son transparentes para el usuario.

### Selección dispositivo bluetooth al cual se desea emparejar

### Selección clave de paso

### Solicitud / Paginación / Descubrimiento servicio / Creación canal / Emparejamiento





Tabla 12 [Configuración conexión Bluetooth]

Para la configuración del monitor serie en el PC\_B, se ha implementado la misma configuración que la mostrada en la tabla 11 (*Configuración PC\_A prototipo comunicación*) para el PC\_A, exceptuando un cambio en la selección del puerto de comunicaciones, en el caso del PC\_B se debe seleccionar el puerto virtual de salida COM31.

En la siguientes capturas se muestra la configuración realizada en el módulo, dicha configuración se ha realizado mediante el enlace bluetooth anteriormente establecido. Se ha subrayado en rojo los comandos introducidos por el usuario y en azul las respuestas recibidas desde el módulo, al lado de cada comando se puede observar una explicación de ellos, dando a entender su función en la configuración del módulo.

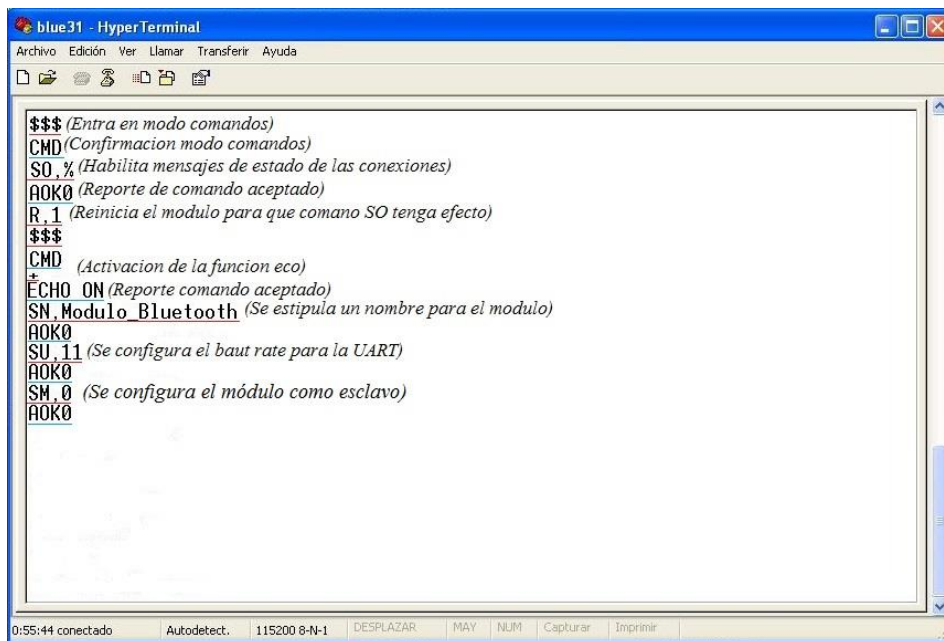
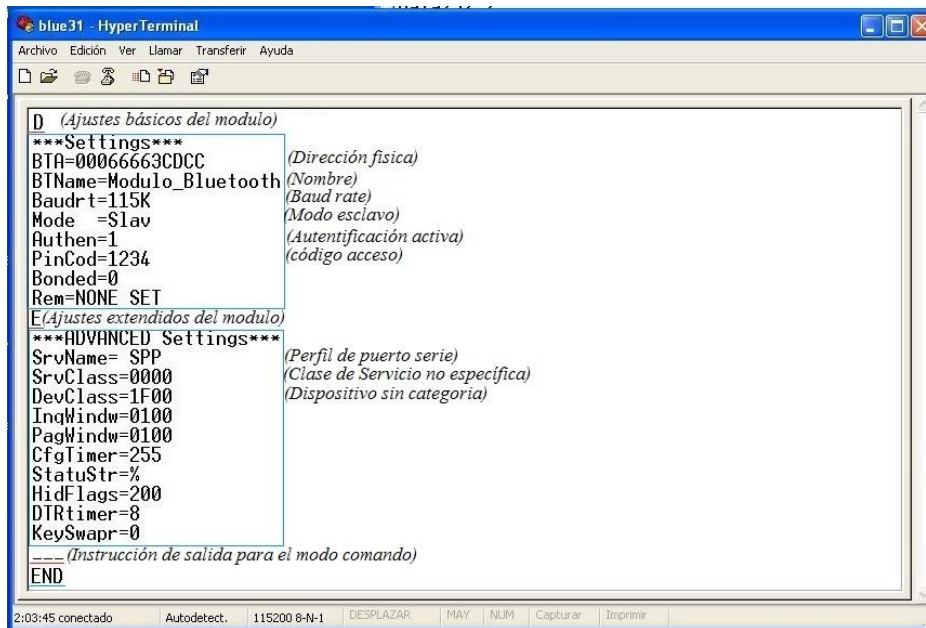


Imagen 7 [Configuración módulo Bluetooth]

Como resultado se ha obtenido la siguiente configuración:



```

blue31 - HyperTerminal
Archivo Edición Ver Llamar Transferir Ayuda
D (Ajustes básicos del modulo)
***Settings***
BTAd=00066663CDCC (Dirección física)
BTName=Modulo_Bluetooth (Nombre)
Baudrt=115K (Baud rate)
Mode =$1av (Modo esclavo)
Authen=1 (Autenticación activa)
PinCod=1234 (código acceso)
Bonded=0
Rem=NONE SET
E (Ajustes extendidos del modulo)
***ADVANCED Settings***
SrvName= SPP (Perfil de puerto serie)
SrvClass=0000 (Clase de Servicio no específica)
DevClass=1F00 (Dispositivo sin categoría)
InqWindw=0100
PagWindw=0100
CfgTimer=255
StatuStr=%
HidFlags=200
DTRtimer=8
KeySwapr=0
--- (Instrucción de salida para el modo comando)
END
2:03:45 conectado Autodetect. 115200 8-N-1 DESPLAZAR MAY NUM Capturar Imprimir

```

Imagen 8 [Configuración resultante en el módulo Bluetooth]

Como prueba final una vez de vuelta al modo bluetooth, todo lo que se escribe en el monitor serie del PC\_A aparece en el monitor del PC\_B y viceversa, esto demuestra que se ha establecido la conexión entre los dos extremos y que se puede llevar a cabo la comunicación con éxito.

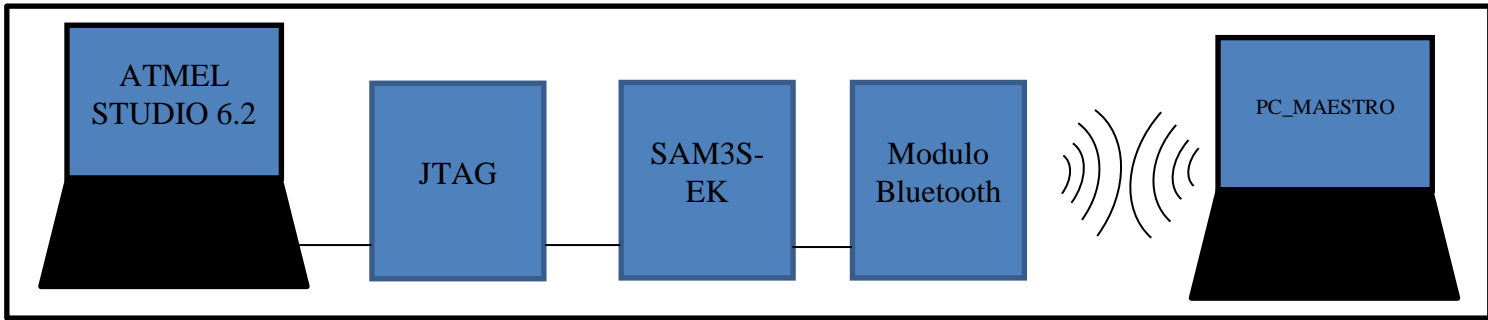
Como conclusión de las pruebas de este prototipo se extrae los conocimientos para realizar la conexión entre un ordenador y un dispositivo bluetooth, así como la monitorización de los mensajes enviados inalámbricamente.

## 6.2. Testeo prototipo real

Para la prueba el prototipo se ha utilizado una tarjeta de evaluación con un microprocesador *ATMEL SAM3S4C*, este micro contiene un puerto de comunicación *UART* que ha sido correspondientemente configurado, para que emita mensajes al módulo bluetooth y éste los reenvíe inalámbricamente al ordenador maestro. Por otro lado para la implementación del software se ha empleado el entorno de *ATMEL STUDIO 6.2*, que proporciona la depuración del código en C y su compilación, en el anexo 9.9 (*Main.c prueba prototipo real*) se puede observar el código implementado.

También ha sido necesario la utilización de un programador *JTAG [E]* para una vez compilado el programa pueda ser enviado al microprocesador, el programador es el encargado de adaptar las señales eléctricas para el *SAM3S4C*. *Tabla 1 (Señales para programación micro ATMEL)*





Esquema 5 [Montaje testeo prototipo real]

El proceso de conexión entre el módulo y el ordenador maestro, es el mismo que el mostrado anteriormente en la tabla 12 (configuración conexión bluetooth), en cambio no se ha realizado la configuración del monitor serie en el otro extremo, ya que no se recibirán mensajes desde el ordenador maestro hacia el modulo bluetooth.

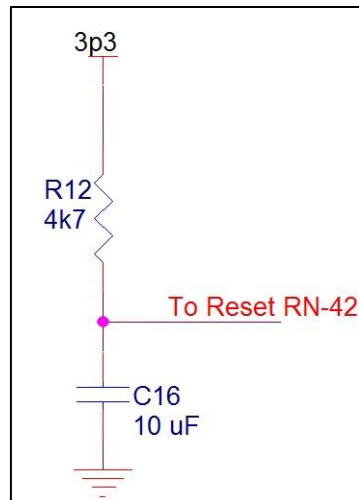
Desde el *SAM3S4C* se ha estado alimentando el modulo bluetooth, además se ha enviado a través de la *UART* el código binario de 8 bits "01000010" (en decimal 66) correspondiente a la letra B en el código ASCII, el módulo ha creado el paquete bluetooth y enviado éste al ordenador maestro. Finalmente el ordenador ha mostrado por el monitor serie dicho carácter, realizando así exitosamente la prueba.

Además se ha retirado la alimentación a la tarjeta de evaluación durante el proceso de comunicación, debido a que en la futura aplicación en algún momento se necesitara reiniciar de esta manera el módulo led a testear, del resultado de efectuar dicho experimento se han extraído unas especificaciones que anteriormente no habían sido contempladas, en el siguiente apartado se exponen y se presentan las soluciones adoptadas para satisfacerlas.

### 6.3. Mejoras añadidas a la implementación

Se ha observado que una vez apagado el sistema, cuando se desea volver a encenderlo hay problemas en el módulo bluetooth, ya que se crea un conflicto entre el ordenador y no se permite reutilizar la conexión anteriormente guardada en el mismo, en esta conexión se mantienen los puertos virtuales creados, permitiendo así la reutilización de la sesión guardada en el monitor serie.

Para resolver dicho problema se probó a provocar un reinicio del módulo bluetooth, en los primeros instantes del encendido, para ello se ha utilizado un circuito de *delay*, compuesto por una resistencia en serie con un condensador.



**Imagen 9 [Esquemático circuito *delay*]**

De esta forma podemos calcular el tiempo de carga del condensador ( $\tau$ ) del circuito, una vez pasado ese tiempo el modulo bluetooth ya estará operativo.

$$\tau = R12 * C16 = 4.7K\Omega * 10 \mu F = 47ms$$

**Formula 1 [Factor tau circuito *delay*]**

Con esta solución y una vez probado se ha dado por resuelto el problema originado.

Debido a la retirada de la alimentación se reinicia también el modulo bluetooth, ya que éste recibe la alimentación por medio del conector proveniente del módulo LED, esto crea el inconveniente de que se pierde la conexión cada vez que se realiza dicho procedimiento, debiendo volver a realizar todo el protocolo de conexión desde el ordenador maestro, además puede cambiar los puertos virtuales creados en la primera conexión obligando a reconfigurar el monitor serie.

Como solución se ha ideado un sistema de back-up para que funcione cada vez que se retira la alimentación, así de esta forma se mantendrá la tensión en el módulo bluetooth. En el siguiente apartado se expone el estudio del circuito implementado para atajar el problema.

### **6.3.1. Solución *back-up***

Para el circuito de back-up se ha empleado el circuito integrado *LTC3226 [H]*, este encapsulado se trata de un cargador de dos súper condensadores en serie, dispone también de un controlador que nos permite conmutar si la alimentación directa falla, a la carga almacenada en los súper condensadores, permitiendo así la alimentación permanente del circuito si falla la fuente de tensión principal.

Símbolo esquemático LTC3226		Símbolo físico LTC3226	
U?			
1	Vout	CPO	16
2	PFO_N	C_positive	15
3	PFI	VMID	14
4	LDO_FB	Vin	13
5	GATE	C_negative	12
6	RST_FB	CAPGOOD	11
7	RST_N	CPO_FB	10
8	EN_CHG	PROG	9

Tabla 13 [Símbolo LTC3226]

A continuación se muestran los cálculos realizados para la correcta configuración del dispositivo, dicha configuración se ha llevado a cabo con el uso de componentes discretos.

#### - Tensión de carga para los súper condensadores.

Los súper condensadores reciben la tensión por medio del contacto *CPO* y se estipula esta tensión por medio de un divisor de tensión entre *CPO* y masa, haciendo la lectura en *CPO\_FB*. Cada súper condensador tiene una tensión nominal de 2.7 V, es por ello que se ha optado por cargarlos con una tensión de 5 V, de este modo se tendrá un tiempo óptimo de carga así como un buen margen para no dañarlos por sobre tensión.

$$V_{cpo} = V_{cpo_{fb}} * \left(1 + \frac{R_{cp1}}{R_{cp2}}\right)$$

Formula 2 [Tensión de carga para los súper condensadores]

Donde  $V_{cpo_{fb}}$  son 1.2 V,  $V_{cpo}$  la tensión a la cual se desea que carguen los condensadores,  $R_{cp1}$  y  $R_{cp2}$  se tratan de las resistencias que nos proporcionaran el divisor de tensión, para su cálculo se ha estipulado el valor de una de ellas y se ha despejado la restante.

$$5V = 1.21V * \left(1 + \frac{3.83M\Omega}{R_{cp2}}\right)$$

**Formula 3 [Resolución tensión de carga para los súper condensadores]**

Finalmente se ha obtenido los siguientes valores de resistencias:

$$R_{cp1} = 3.83M\Omega \quad R_{cp2} = 1.21M\Omega$$

#### - **Limitador de corriente de carga para los súper condensadores**

Esta corriente es programada mediante una resistencia conectada a la patilla *PROG* del encapsulado, en la hoja de características del *LTC3226* se ha extraído que la máxima corriente se consigue con una resistencia de  $33.2K\Omega$ , dando por resultado la siguiente corriente de carga.

$$R_{prog} = 10500 * \frac{1V}{I_{vin}}$$

**Formula 4 [Corriente de carga para los súper condensadores]**

$$I_{vin} = 10500 * \frac{1V}{33.2K\Omega} = 315mA$$

**Formula 5 [Resolución corriente de carga para los súper condensadores]**

#### - **Voltaje de entrada al cual conmuta al modo back-up**

El voltaje de entrada al cual el integrado debe conmutar a modo back-up, se estipula mediante un divisor de tensión desde *Vin* a *GND* realizando la lectura en el contacto *PFI*.

$$V_{in} = V_{pfi} * \left(1 + \frac{R_{pf1}}{R_{pf2}}\right)$$

**Formula 6 [Tensión de entrada para conmutar al modo back-up]**

Donde  $V_{pfi}$  es  $1.2V$ ,  $V_{in}$  es la tensión a la que se desea que conmute y las resistencias  $R_{pf1}$  y  $R_{pf2}$  las encargadas de crear el divisor de tensión. Para nuestro caso se desea que conmute cuando el voltaje de entrada alcance un valor nulo, es por ello que no es un valor de especial relevancia para el diseño. Para la configuración se ha optado por

seleccionar dos resistencias de igual valor, dando como resultado el siguiente valor de tensión a la entrada.

$$V_{in} = 1.2V * \left(1 + \frac{196K\Omega}{196K\Omega}\right) = 2.4V$$

**Formula 7 [Resolución tensión de entrada para conmutar al modo back-up]**

Como resultado cuando el voltaje de entrada sea inferior a 2.4 V, el dispositivo conmutara al modo back-up.

### - Voltaje de transferencia de los súper condensadores a la Salida

La tensión de salida proporcionada por los súper condensadores en modo back-up, puede estar comprendida entre los 2.5 V y 5.3 V, para nuestro caso la tensión deseada es de 3.3V, para tener dicha tensión se precisa de un divisor de tensión entre la patilla  $V_{out}$  y  $GND$  realizando la lectura en el contacto  $LDO_{FB}$ .

$$V_{out} = V_{ldo\_fb} * \left(1 + \frac{R_{fb1}}{R_{fb2}}\right)$$

**Formula 8 [Voltaje de salida en modo back-up]**

Donde  $V_{out}$  son los 3.3 V deseados a la salida,  $V_{ldo\_fb}$  son 0.8 V y  $R_{fb1}$  e  $R_{fb2}$  las resistencias que conforman el divisor de tensión, el valor de dichas resistencias debe estar entre el rango de 40K $\Omega$  y 500K $\Omega$ , del caso contrario un valor menor puede provocar una corriente demasiado elevada y un valor elevado puede crear un polo adicional y crear inestabilidad.

Para la obtención del valor de salida, se ha elegido un valor para la resistencia  $R_{fb1}$  y a partir de este se ha calculado el valor para  $R_{fb2}$ .

$$3.3V = 0.8V * \left(1 + \frac{255K\Omega}{R_{fb2}}\right)$$

**Formula 9 [Resolución voltaje de salida en modo back-up]**

Obteniéndose los siguientes valores para las resistencias.

$$R_{fb1} = 255K\Omega$$

$$R_{fb2} = 80.6K \Omega$$

Finalmente se presenta el esquemático del circuito diseñado.

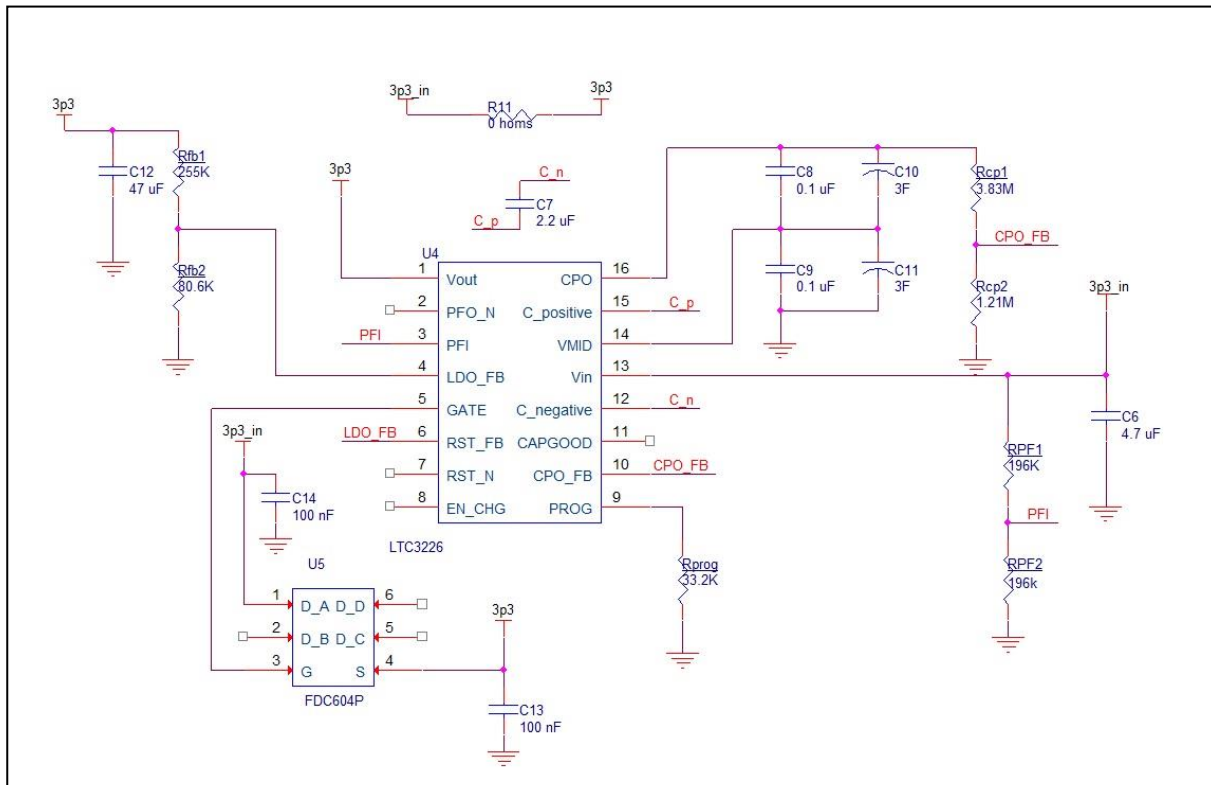


Imagen 10 [Esquemático circuito de back-up]

El encapsulado *FDC604P[K]* se trata de un transistor MOSFET\_PMOS que se encarga de conmutar al modo de back-up, por medio del contacto *GATE* se realiza su polarización y con ella la conmutación entre modos.

La resistencia *R11* ha sido introducida pensando en futuras aplicaciones donde no se precise de un back-up para la alimentación, se trata de una resistencia de valor  $0\Omega$  que nos permitirá realizar un bypass cuando no se requiera hacer uso del circuito.

El condensador *C7* está dispuesto según requiere la hoja de características del dispositivo, así como los condensadores *C12*, *C6*, *C8* y *C9*. El resto de componentes discretos son los anteriormente expuestos.

Para la prueba del circuito diseñado se creó un prototipo donde poder comprobar la funcionalidad, este prototipo se montó sobre una placa de baquelita y se soldó a mano como los anteriores prototipos presentados. Dado que el *LTC3226* viene encapsulado en un *QFN* de 16 patillas, ha sido necesario un adaptador para su montaje en la placa de baquelita, en la siguiente imagen se presenta el adaptador empleado.

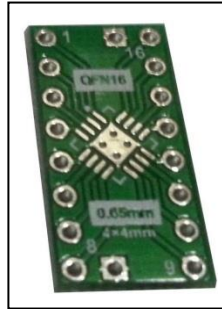
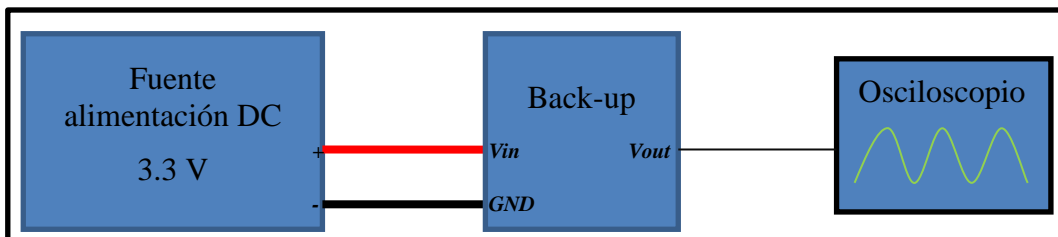


Imagen 11 [Adaptador para QFN]

En la siguiente figura se muestra el resultado final del montaje del circuito y el esquema de la prueba realizada.

Top prototipo back-up	Bottom prototipo back-up

Tabla 14 [Prototipo circuito back-up]



Esquema 6 [Montaje testeo prototipo back-up]

Una vez realizado el montaje mostrado en la figura, se alimentó a 3.3 V el back-up y se comprobó que los súper condensadores alcanzaran la tensión de 5 V esperada en ellos, una vez estaban los condensadores cargados se retiró la tensión y por medio del osciloscopio se comprobó el tiempo de descarga de los condensadores, se observó un tiempo de descarga de 5 segundos, tiempo más que suficiente para satisfacer las especificaciones del sistema.

## 7. Diseño final

Como ya ha sido indicado anteriormente, el modulo bluetooth empleado para el diseño final se trata del RN-42, a éste le ha sido añadido el sistema de back-up presentado en el apartado anterior, a continuación se muestra el esquemático del diseño final del circuito.

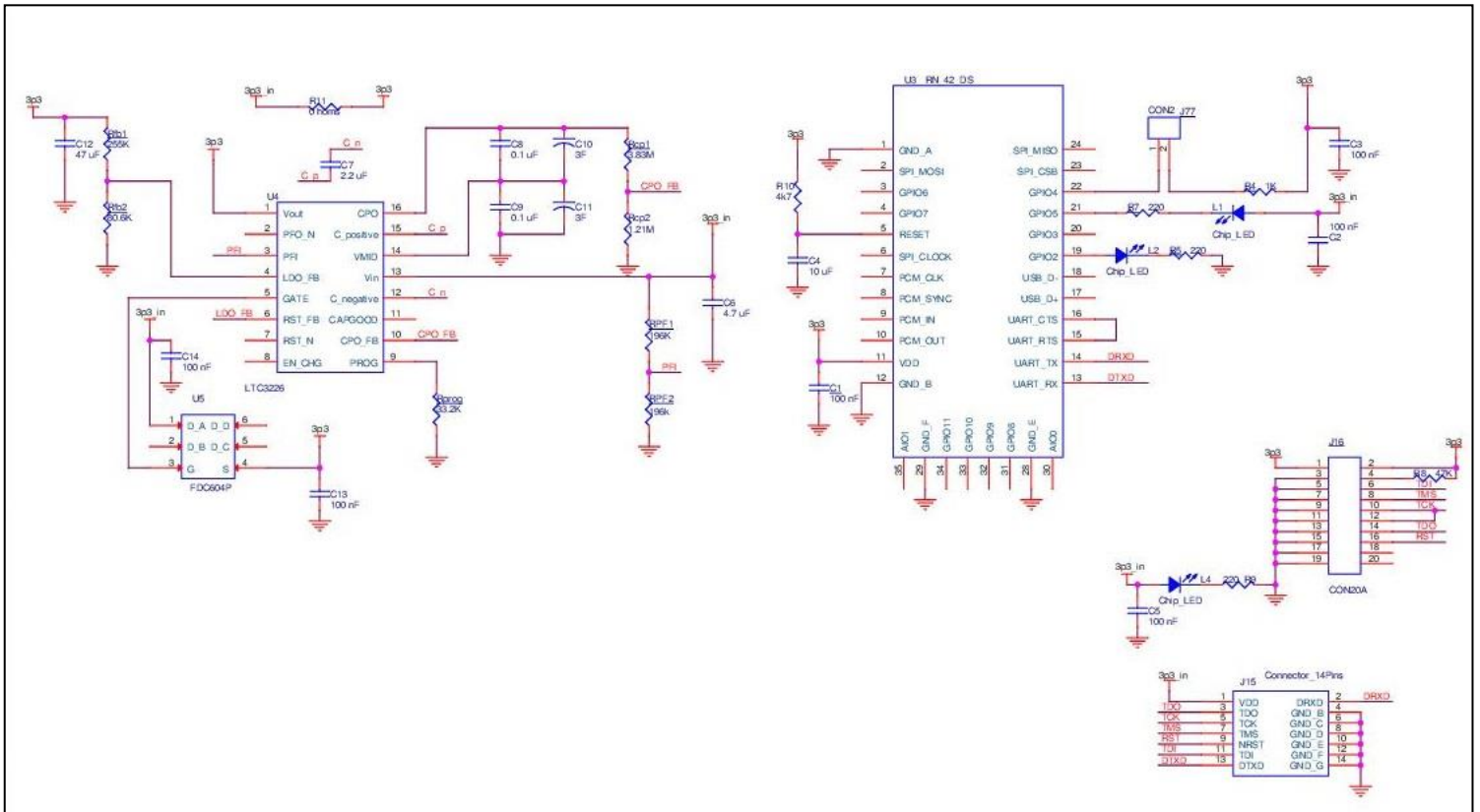


Imagen 12 [Esquemático circuito final]

Al circuito final le han sido añadidos diferentes diodos leds que nos darán información del estado del módulo:

- El led *L4* indicara la existencia de alimentación proveniente de la fuente de alimentación.
- El led *L2* actuara como indicador de conexión si esta encendido el módulo estará conectado a una red *piconet*.
- El led *L1* será un indicador de estado donde dependiendo de su parpadeo indicara que el módulo está a la espera de una conexión o en estado de comandos, si el módulo se encuentra conectado este permanecerá apagado.
- También le ha sido añadido un *jumper (j77)* para poder realizar un reinicio de fábrica en caso de que se efectuó una mala configuración o haya algún inconveniente en el software del integrado.



- El conector *J15* será conectado al módulo led que contendrá el micro procesador, a través de él serán recibidos los mensajes que el modulo bluetooth deberá reenviar inalámbricamente al ordenador oyente.
- El conector *J16* será utilizado para programar el micro controlador por medio el programador *JTAG* anteriormente expuesto en el apartado 6.2.

Una vez el esquemático del circuito final está diseñado se ha realizado el diseño de la *PCB*, estará compuesta por dos capas haciendo ambas de plano de masa estando todos los componentes emplazados en la cara superior de la placa, para el diseño de ésta se ha utilizado el programa *Allegro* de la compañía *Cadence [J]*, en los siguientes puntos se exponen y explican las diferentes capas que componen el circuito. En los anexos se presentan los dos circuitos diseñados, el circuito conectores programador/comunicación (*apartado 4.1*) y circuito módulo bluetooth (*apartado 4.2*).

- **ART FILM – TOP [pág. 35]**

Esta capa muestra las pistas de cobre trazadas para realizar las conexiones entre los distintos componentes que conforman el circuito, también se representan las vías que conectan los dos planos de masa, en ella se puede apreciar que el plano de masa no cubre toda la superficie de la placa, esto es debido a que se ha realizado un vacío en el lugar donde va emplazada la antena del módulo bluetooth para que no provoque interferencias.

- **ART FILM – BOTTOM [pág. 36]**

Al igual que la capa anterior, en ésta se muestran pistas y vías de la capa inferior de la *PCB*. Como ya ha sido comentado anteriormente se puede apreciar que no hay componentes emplazados en esta cara.

- **ART FILM – SOLDERMASK\_TOP [pág. 37]**

Esta capa muestra en que partes de la placa se va a distribuir la máscara *antisoldante*, esta máscara se trata de una pintura especial que brinda una protección frente agentes externos que pueden provocar corrosión u oxidación en el cobre, las áreas comprendidas dentro de los recuadros verdes indica en qué lugar no va a ser distribuida la pintura, estas áreas corresponden al emplazamiento de los *pads* de los componentes ya sean de montaje en superficie o de agujero pasante.

- **ART FILM – PASTEMASK\_TOP [pág. 38]**

Al contrario que la anterior capa, ésta indica en qué lugar de la placa va dispuesta la pasta de estaño para el posterior montaje de los componentes, en las áreas marcadas se encontrara el cobre desnudo para que el estaño pueda fundirse y que los componentes tengan contacto eléctrico.

- **ART FILM – SILKSCREEN\_TOP [pág. 39]**

La capa *Silk Screen* se trata de una impresión en tinta que se realiza sobre la placa, en ella se imprimen contornos de algunos componentes, su referencia en el esquemático y un marcador que indique la posición de la patilla número uno de los encapsulados. Se trata de una máscara de referencia para el montaje y la posterior utilización del circuito. Es importante que esta capa no se solape con la capa *Paste Mask*, ya que sería contraproducente que se pintara encima del cobre desnudo.

- **ART FILM – ASSEMBLY\_TOP [pág. 40]**

El *Assembly Top* no es relevante para la manufacturación de la *PCB*, más bien es una guía para el montador, en ella se puede apreciar las referencias de los componentes en el esquemático, así como la disposición de estos en la placa.

- **ART FILM – SOLDERMASK\_BOTTOM [pág. 41]**

Esta capa es de características idénticas a la explicada anteriormente en *Solder Mask Top*, pero esta vez muestra las áreas que no estarán recubiertas por esta pintura en la cara inferior del circuito.

- **ART FILM – BOARD\_OUTLINE [pág. 42]**

El *Board Outline* se trata de la forma geométrica que debe tener la *PCB*, en el apartado 2 (Especificaciones del sistema) se enumeró como especificación que la placa debía tener dimensiones acotadas ya que estaría emplazada dentro de una caja, por ello esta capa muestra el diseño realizado para que la *PCB* entre dentro del recipiente que la va a contener, en la siguiente imagen se muestran las dimensiones y el aspecto físico de la caja contenedora


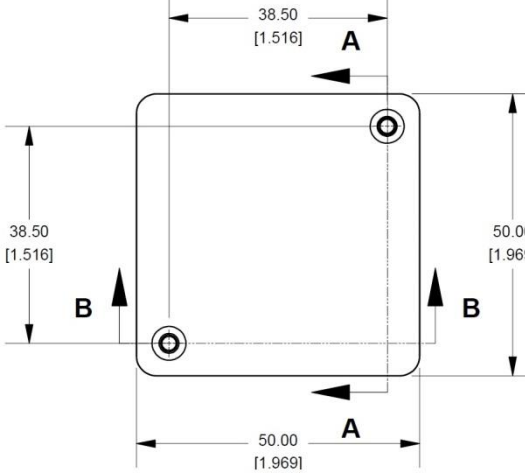
Aspecto físico caja contenedora	Dimensiones caja contenedora
	

Tabla 15 [Aspecto y dimensiones de la caja contenedora]

## **8. Conclusiones y trabajo futuro**

En este Trabajo Fin de Grado se ha realizado la implementación de un sistema de comunicaciones inalámbrico que utiliza el protocolo bluetooth y cuya finalidad es realizar el testeo de una pantalla led.

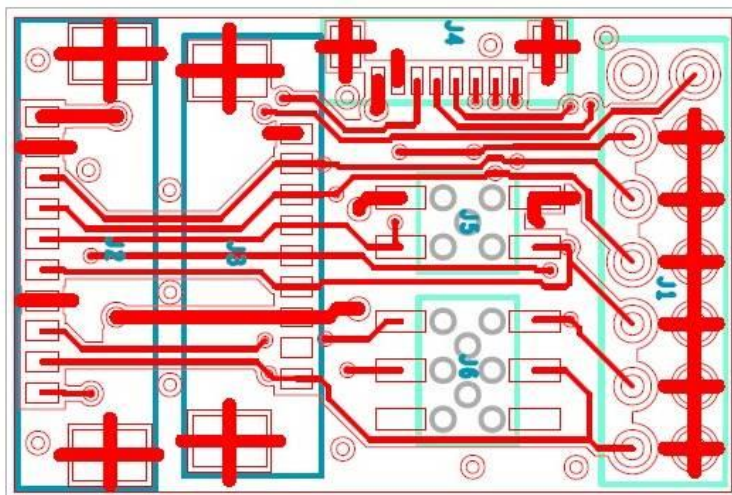
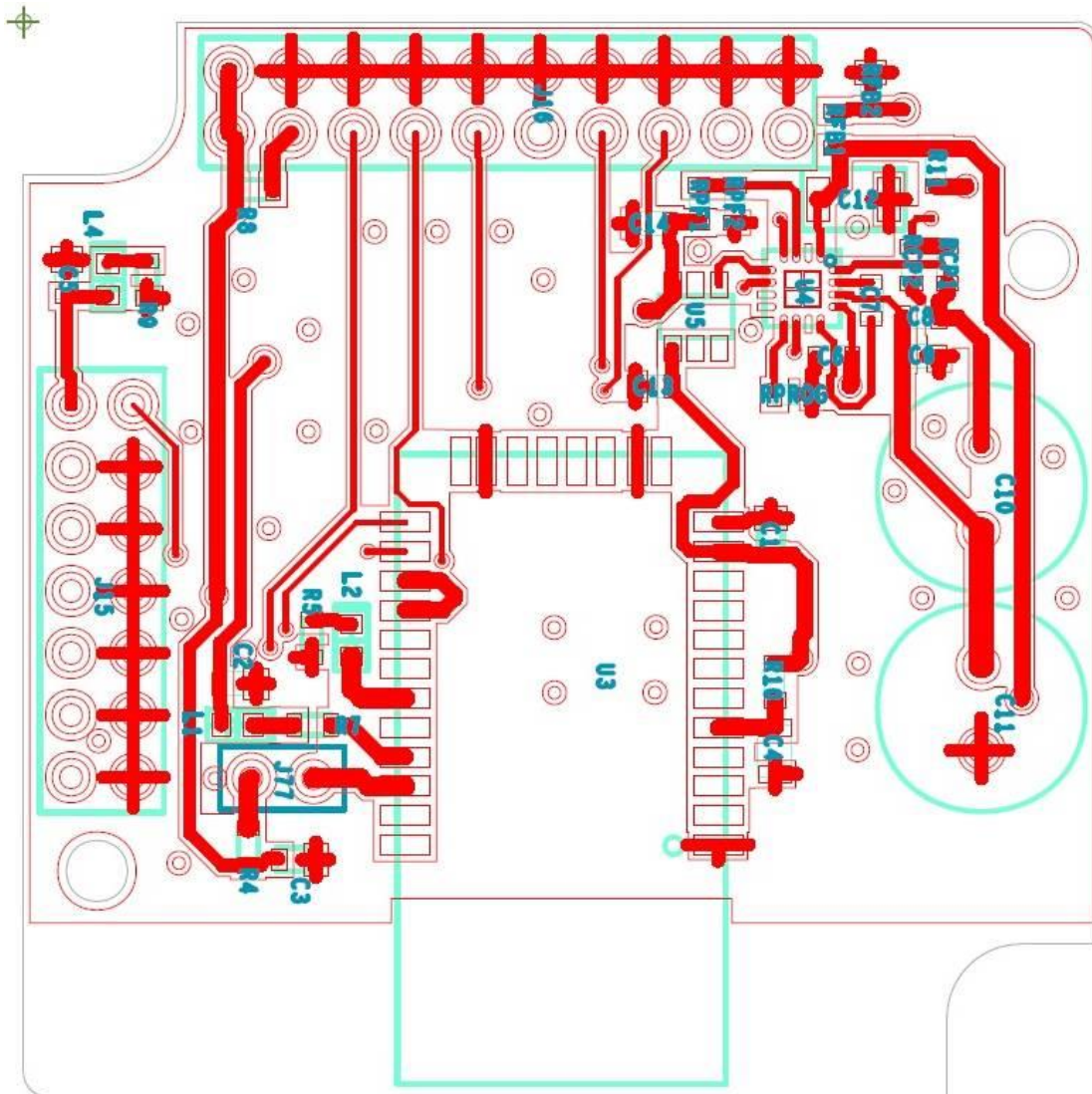
Se han realizado varios prototipos los cuales han sido testeados tras lo cual se han corregido los problemas encontrados, una vez solucionados se ha realizado el diseño de las dos placas, conectores programación/comunicación y módulo bluetooth. El montaje de las placas no ha sido posible mostrarlo en el trabajo, debido a un problema con las fechas de entrega de las placas por parte del proveedor y la entrega del presente trabajo final de grado, es por ello que el montaje de las placas es uno de los trabajos futuros que se llevaran a cabo.

Por conclusión se ha extraído la posibilidad de crear conexión bluetooth en un microprocesador ATMEL haciendo uso de su módulo de comunicación UART, por medio de un módulo bluetooth emisor/receptor estándar disponible en el mercado, además abre un abanico de futuras implementaciones para mejorar la funcionalidad de las pantallas led. Es por ello que se ha contemplado la posibilidad de realizar comunicaciones inalámbricas entre los diferentes módulos que conformaran la pantalla led, esta interconexión se podría llevar a cabo mediante el protocolo bluetooth o wiffi, es por ello que como trabajo futuro se propone la investigación y el desarrollo de prototipos, para poder realizar una experimentación y lograr el objetivo de la comunicación entre módulos led.

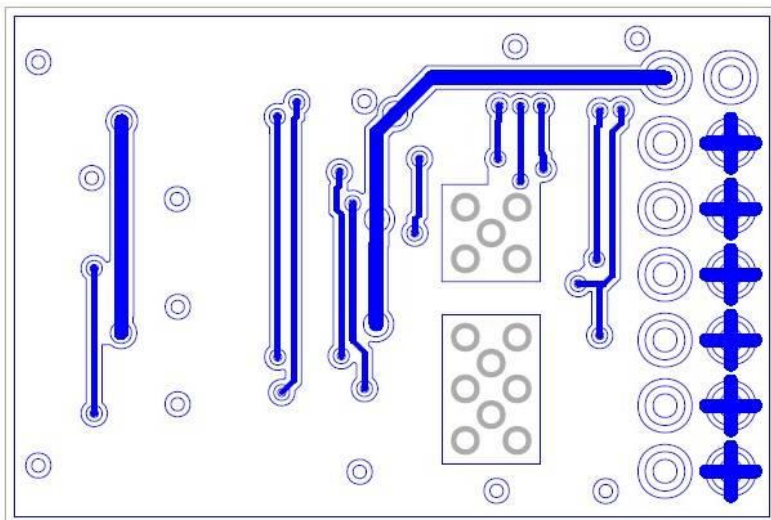
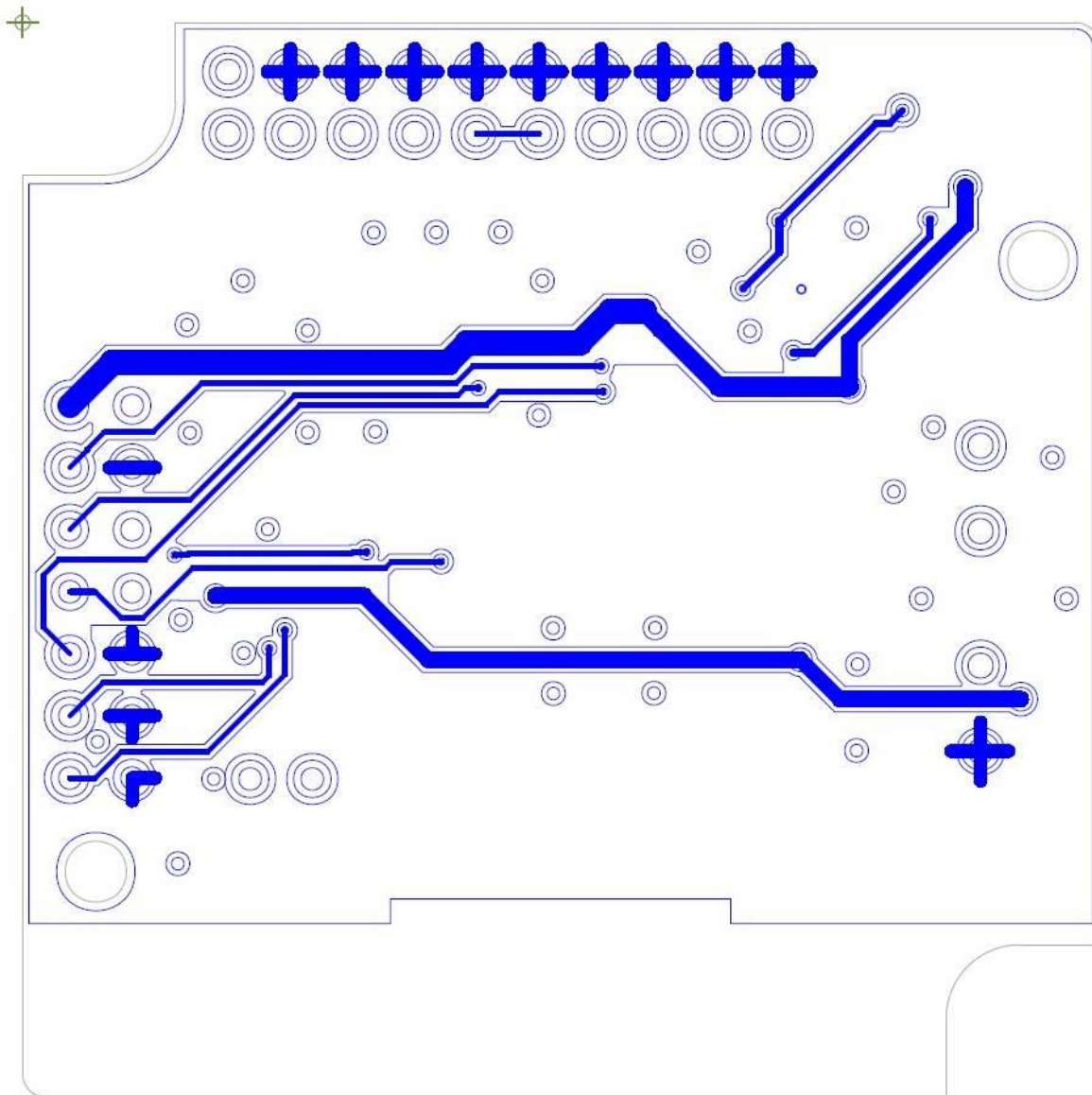
Finalmente se ha completado el objetivo de aplicar conocimientos adquiridos durante la formación académica en el trabajo, como vienen a ser los referentes a electrónica y comunicaciones.

## 9. Anexos

### 9.1. ART FILM – TOP

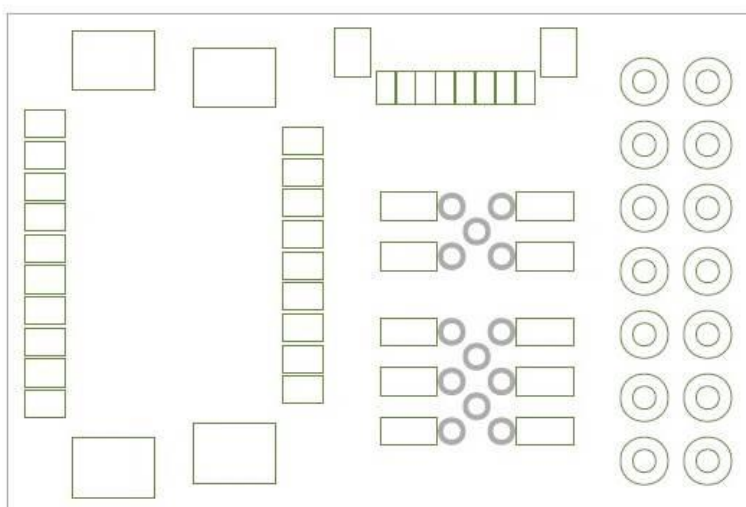
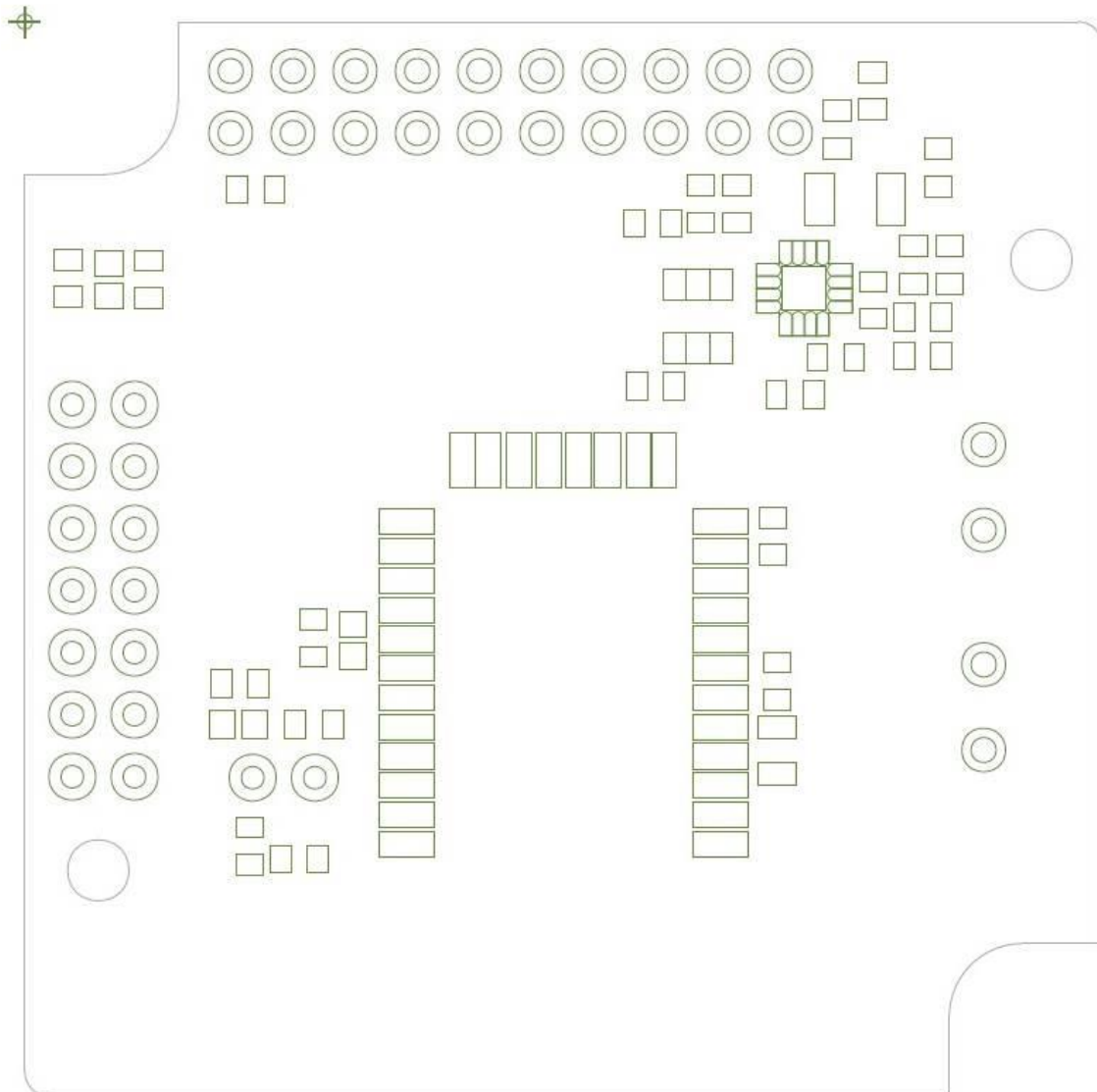


## 9.2. ART FILM – BOTTOM

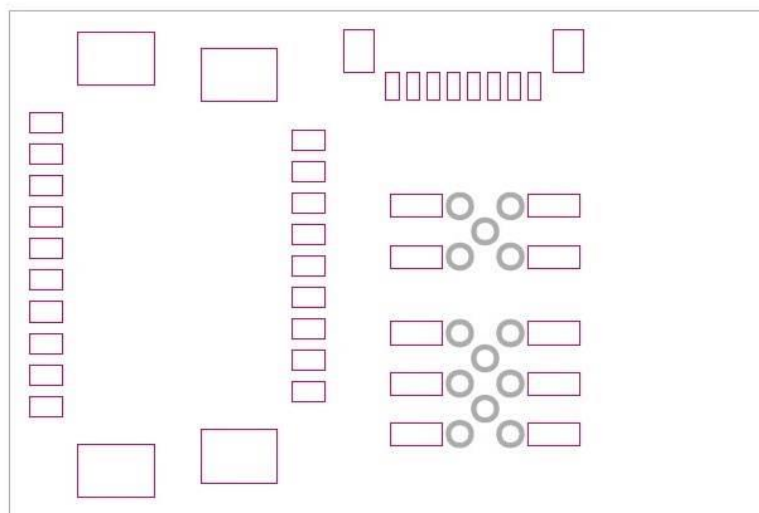
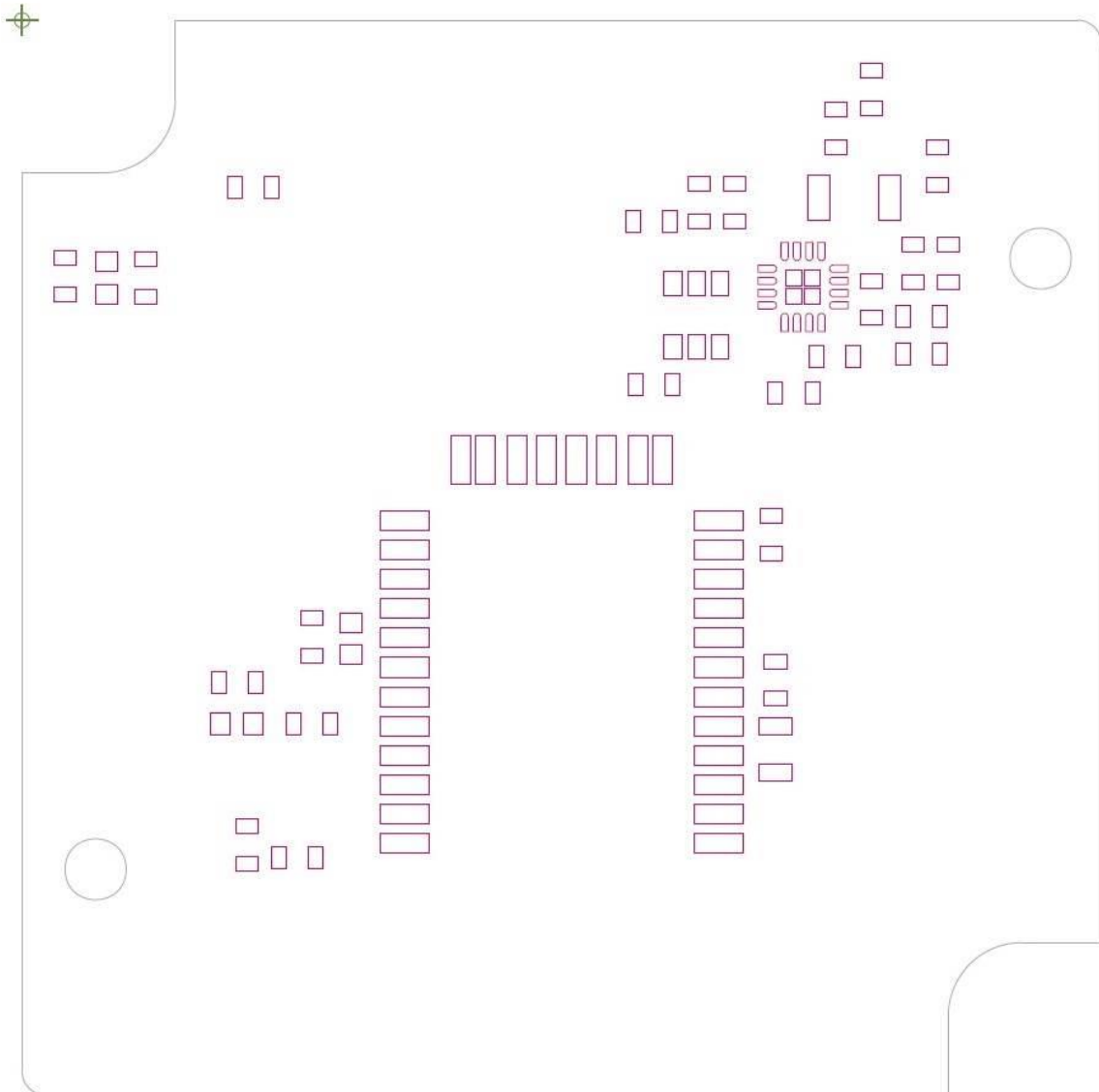




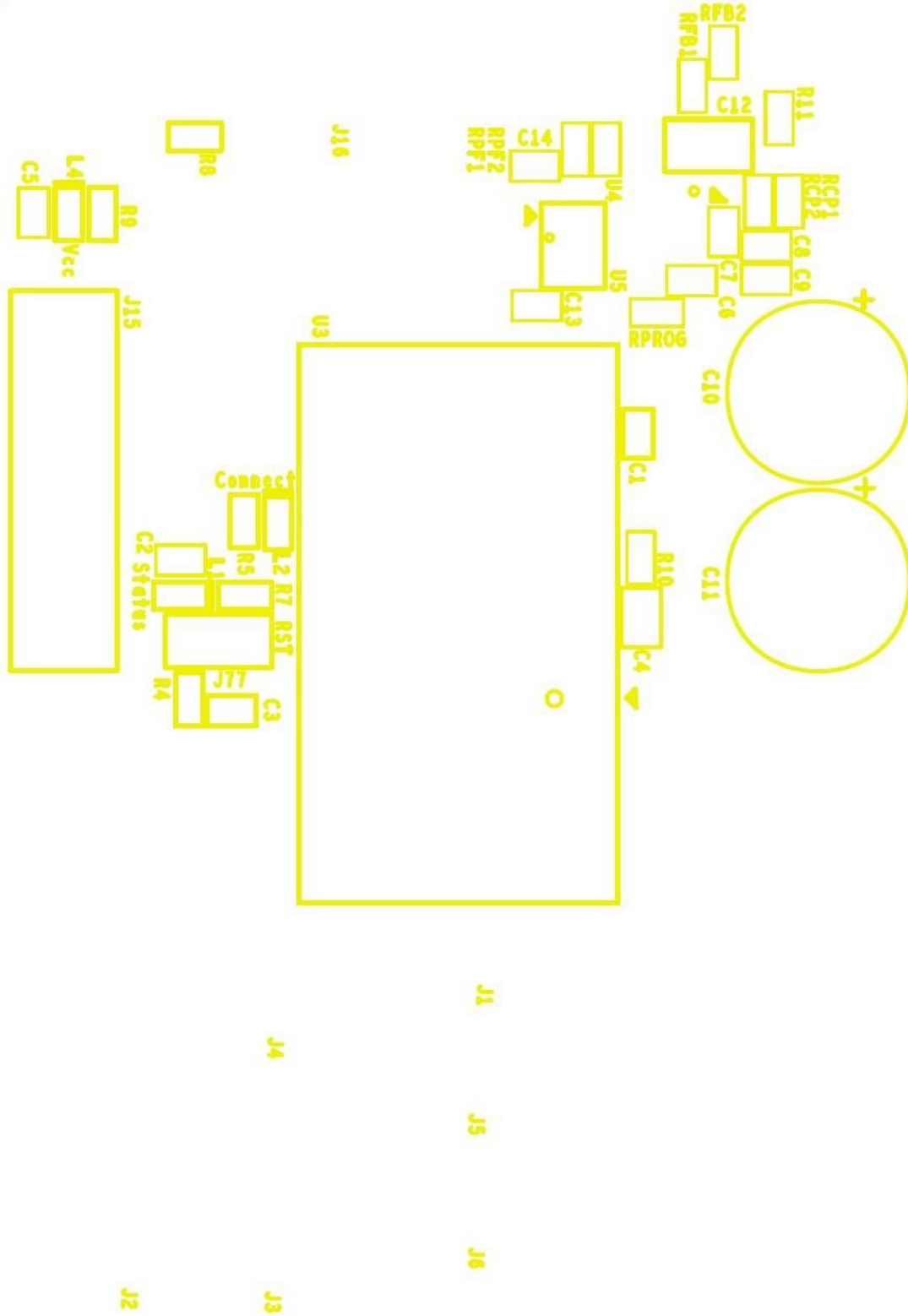
### 9.3. ART FILM - SOLDERMASK\_TOP



### 9.4. ART FILM - PASTEMASK\_TOP



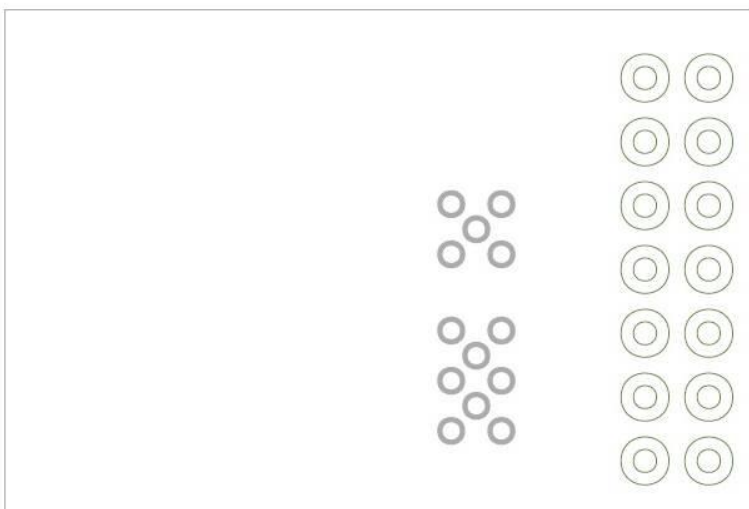
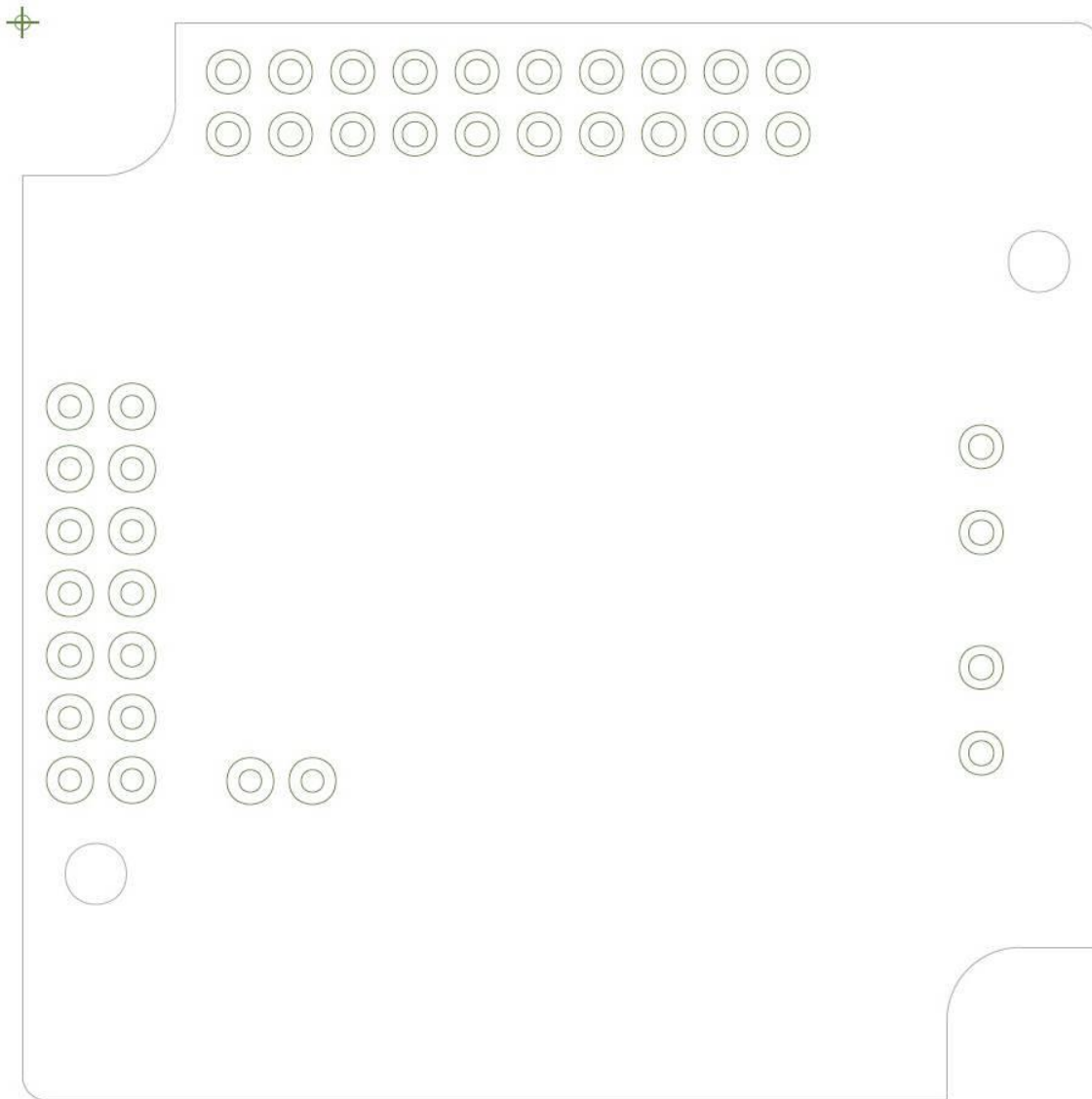
### 9.5. ART FILM - SILKSCREEN\_TOP



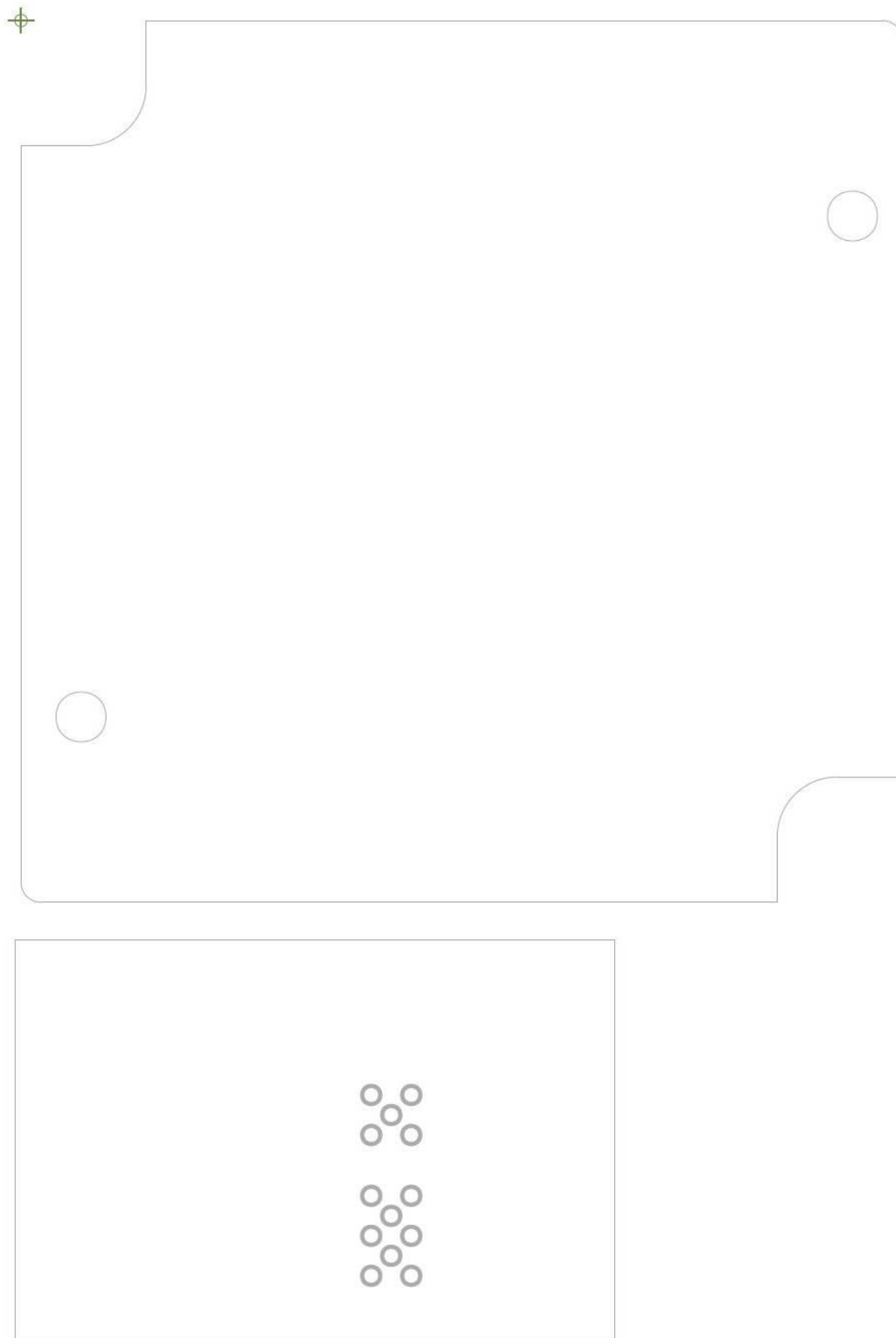




### 9.7. ART FILM - SOLDERMASK\_BOTTOM



## 9.8. ART FILM - BOARD\_OUTLINE



## 9.9. MAIN.C Prueba prototipo real

```

#include "asf.h"
#include "conf_board.h"
#include "delay.h"

#define STRING_EOL    "\r"
#define STRING_HEADER "-- USART Serial Example --\r\n" \
"-- "BOARD_NAME" --\r\n" \
"-- Compiled: "__DATE__" "__TIME__" --"STRING_EOL

static void configure_uart(void)
{
    const sam_uart_opt_t uart_console_settings = {

        sysclk_get_cpu_hz(),
        115200,
        0x00000800

    };

    sysclk_enable_peripheral_clock(UART1_ID);

    uart_init(UART1_MASTER_BASE, &uart_console_settings);

    uart_disable_interrupt(UART1_MASTER_BASE, 0xffffffff);

    UART1_MASTER_BASE->UART_BRGR=0x00000023;

} // configure_console

static void configure_console(void)
{
    const usart_serial_options_t uart_serial_options = {
        .baudrate = CONF_UART_BAUDRATE,
        .paritytype = CONF_UART_PARITY
    };

    /* Configure console UART. */
    sysclk_enable_peripheral_clock(CONSOLE_UART_ID);
    stdio_serial_init(CONF_UART, &uart_serial_options);
}

int main (void)
{
    // Insert system clock initialization code here (sysclk_init()).
    sysclk_init();
    board_init();

    configure_console();

    puts(STRING_HEADER);

    /* Initialize the console UART. */
    configure_uart();
    int i;

    while(1){

        for(i=0;i<9;i++){

            uart_write(UART1_MASTER_BASE,66);
            delay_s(1);

        }

    }
}

```

## 9.10. Bill of materials

Part	Reference	VOLTAGE	PCB footprint	Quantity
100 nF	C1,C2,C3,C5,C13,C14	6.3V	C0603	6
10 uF	C4	6.3V	C0805	1
4.7 uF	C6	6.3V	C0603	1
2.2 uF	C7	10 V	C0603	1
0.1 uF	C8,C9	16V	C0603	2
3 F	C10,C11	2.7V	HV0820_Super_Cap	2
47 uF	C12	6.3 V	C1206	1
Connector_14Pins	J1,J15		CON14_B	2
PT8_4824_R	J2		SMD_HEADER_10	1
POLYMER	J3		SMD_HEADER_9	1
PT4P4_12606_RF	J4		SM08B	1
SERIALIZADOR_CON4	J5		CON4_SMD_BOTTOM	1
SERIALIZADOR_CON6	J6		CON6_SMD_BOTTOM	1
CON20A	J16		ARM_JTAG	1
CON2	J77		CON2	1
Chip_LED	L1,L2,L4		LED_0603	3
196K	RPF1,RPF2		R0603	2
3.83M	Rcp1		R0603	1
1.21M	Rcp2		R0603	1
255K	Rfb1		R0603	1
80.6K	Rfb2		R0603	1
33.2K	Rprog		R0603	1
1K	R4		R0603	1
220	R5,R7,R9		R0603	3
47K	R8		R0603	1
4k7	R10		R0603	1
0 homs	R11		R0603	1
RN_42_DS	U3		RN_42	1
LTC3226	U4		QFN16_B	1
FDC604P	U5		SuperSOT_6	1

## Bibliografía

- A. <http://www.senia-tech.com/>
- B. <http://www.harwin.com/>
- C. <http://es.ccm.net/contents/69-como-funciona-bluetooth>
- D. <http://www.hispazone.com/Guia/72/Bluetooth-El-Futuro-de-las-Comunicaciones-I.html>
- E. [http://www.jtag.com/en?gclid=CjwKEAajw6IauBRCJ3KPXkNro1BoSJA\\_AhXxpyAYo2dUXicX3qC9BlwdUOpZ0lhiWhffboVA0HNGKeRxoCm\\_xXw\\_wcB](http://www.jtag.com/en?gclid=CjwKEAajw6IauBRCJ3KPXkNro1BoSJA_AhXxpyAYo2dUXicX3qC9BlwdUOpZ0lhiWhffboVA0HNGKeRxoCm_xXw_wcB)
- F. <http://www.atmel.com/>
- G. <http://www.microchip.com/pagehandler/en-us/technology/bluetooth/home.html>
- H. <http://www.linear.com/product/LTC3226>
- I. [http://materias.fi.uba.ar/6679/apuntes/RS232\\_V35.pdf](http://materias.fi.uba.ar/6679/apuntes/RS232_V35.pdf)
- J. <http://www.cadence.com/us/Pages/default.aspx>
- K. <https://www.fairchildsemi.com/datasheets/FD/FDC604P.pdf>
- L. [http://www.zator.com/Hardware/H2\\_5\\_1\\_1.html](http://www.zator.com/Hardware/H2_5_1_1.html)
- M. Bray, Jennifer. Charles F. Sturman. Bluetooth: connect whitout cables, Pearson Education. 2001
- N. Allan R. Hambley. Electrónica, Pearson Educación. 2001

- O. Bataller Mascarell, Jordi. Programación en C, Universidad Politécnica de Valencia servicio de publicación. 1999

## Acrónimos

**USB:** *Universal Serial Bus*, es utilizado como estándar para la conexión de periféricos a un ordenador.

**Protocolo Bluetooth:** Habilita la transmisión de voz y datos mediante la creación de una *WPAN*.

**Jtag:** *Join test action group*, habilita al programador para poder entrar dentro el módulo de depuración de la CPU.

**Microprocesador ATMEL:** Microprocesadores basados en arquitectura *ARM* y sus propias arquitecturas *AVR* y *AVR32*.

**SAM3S4C:** Microprocesador de 32 bits basado en la arquitectura *ARM*.

**Cadence capture:** Software dedicado al diseño de esquemas electrónicos, para el posterior diseño de circuitos integrados.

**Allegro:** Software encargado de diseñar *PCBs*.

**Back-up:** Hace referencia a cualquier sistema que implemente una salvaguarda frente a la posible pérdida de datos.

**PCB:** *Printed Circuit Board*, es la superficie de material no conductor sobre la que se trazan las pistas de material conductor, que conforman el circuito.

**UART:** *Universal Asynchronous Receiver-Transmitter*, fragmenta bytes y transmite los bits que lo conforman de forma secuencial.

**Puertos virtuales:** Puertos virtuales son aquellos que se crean para emular el comportamiento del puerto serie de un pc.

**RS-232:** *Recomnded Sandard 232*, designa la norma para el intercambio de datos binarios entre dos equipos.

**Redes inalámbricas de área personal WPAN:** *Wireless Personal Area Network*, red inalámbrica de interconexión entre dispositivos dentro un espacio personal de trabajo.



**GFSK** : Modulación por desplazamiento de frecuencia gaussiana

**DB9**: Conector analógico de 9 contactos, utilizado comúnmente en la norma RS-232

**CMOS**: Tecnología de construcción de circuitos digitales implementada mediante transistores MOS de canal n y canal p.

**Piconet**: *WPAN* creada mediante una conexión bluetooth.

**SDP**: *Service Discovery Protocol*, protocolo de red que permite la detección automática de dispositivos o servicios.

**L2CAP**: *Logical Link Control and Adaptation Protocol*, diseñado para pasar paquetes a sus capas superiores, siendo estos con o sin orientación a la conexión.

**RFCOMM**: *Radio Frequency Communication*, conjunto de protocolos de transporte encargado de la emulación de los puertos series.

**ATMEL STUDIO 6.2**: Software para el desarrollo y depuración de código orientado a microprocesadores de la compañía ATMEL.

**MOSFET PMOS**: Transistor de efecto de campo metal-óxido-semiconductor.

**QFN**: *Quad-flat no-leads*, encapsulado para circuitos electrónicos integrados, de montaje de superficie.

**Jumper**: Conector de dos contactos que actúa como un interruptor dentro el circuito.

# Índice de contenidos

## TABLAS

	<b>Pág.</b>
- Tabla 1 [Señales para programación micro ATMEL]	6
- Tabla 2 [Conector J2]	7
- Tabla 3 [Conector J3]	7
- Tabla 4 [Conector J4]	8
- Tabla 5 [Conector J5 y J6]	8
- Tabla 6 [Conector J1]	9
- Tabla 7 [Símbolo RN-42]	12
- Tabla 8 [Símbolo RN-41]	12
- Tabla 9 [Prototipo comunicación]	16
- Tabla 10 [Prototipo real]	17
- Tabla 11 [Configuración PC_A prototipo comunicación]	19
- Tabla 12 [Configuración conexión Bluetooth]	22
- Tabla 13 [Símbolo LTC3226]	25
- Tabla 14 [Prototipo circuito back-up]	30
- Tabla 15 [Aspecto y dimensiones de la caja contenedora]	33

## IMÁGENES

	<b>Pág.</b>
- Imagen 1 [Esquemático circuito conectores programación/comunicación]	10
- Imagen 2 [Trama de datos UART]	13
- Imagen 3 [Esquemático prototipo comunicación]	15
- Imagen 4 [Esquemático prototipo real]	16
- Imagen 5 [Resistencias retiradas para prototipo real]	17
- Imagen 6 [Esquema red Piconet]	20
- Imagen 7 [Configuración módulo Bluetooth]	22

- Imagen 8 [Configuración resultante en el módulo Bluetooth]	23
- Imagen 9 [Esquemático circuito <i>delay</i> ]	25
- Imagen 10 [Esquemático circuito de back-up]	29
- Imagen 11 [Adaptador para QFN]	30
- Imagen 12 [Esquemático circuito final]	31

## ESQUEMAS

	Pág.
- Esquema 1 [Diagrama bloques para diseño del circuito]	3
- Esquema 2 [Diagrama bloques del sistema]	5
- Esquema 3 [Diagrama Hardware módulo bluetooth]	14
- Esquema 4 [Montaje prueba prototipo comunicación]	18
- Esquema 5 [Montaje testeo prototipo real]	24
- Esquema 6 [Montaje testeo prototipo back-up]	30

## FORMULAS

	Pág.
- Formula 1 [Factor tau circuito <i>delay</i> ]	25
- Formula 2 [Tensión de carga para los súper condensadores]	26
- Formula 3 [Resolución tensión de carga para los súper condensadores]	27
- Formula 4 [Corriente de carga para los súper condensadores]	27
- Formula 5 [Resolución corriente de carga para los súper condensadores]	27
- Formula 6 [Tensión de entrada para conmutar al modo back-up]	27
- Formula 7 [Resolución tensión de entrada para conmutar al modo back-up]	28
- Formula 8 [Voltaje de salida en modo back-up]	28
- Formula 9 [Resolución voltaje de salida en modo back-up]	28