

## Anexo B

### Comunicaciones entre $\mu$ C y PC

En este apartado se hará hincapié en los comandos para el manejo del módulo de comunicaciones desde el PC. Conociendo estos comando se podrá realizar una aplicación sobre PC para la gestión de la programación y depuración del sistema. Así mismo es recomendable leerla antes de intentar comprender el código fuente del programa de gestión del  $\mu$ C.

El módulo de gestión empotrado en el  $\mu$ C ofrece unos servicios a través del puerto serie, mediante comandos ascii, que están disponibles cuando el programa se encuentra en modo programación<sup>1</sup>. La comunicación a través del puerto serie tiene las siguientes características:

**Baudios: 115200**  
**Bits de datos: 8**  
**Bits de stop: 1**  
**Paridad: Ninguna**  
**Control de flujo: Ninguno**

Pasamos a describir estos servicios y los comandos para solicitarlos, para ello seguiremos la nomenclatura:

[xxx]  $\Rightarrow$  "aaa"

Donde "xxx" representa el emisor de la cadena ascii (cliente, por ejemplo un PC, y servidor que es el  $\mu$ C). Y "aaa" es un ejemplo de cadena que se envía desde [xxx].

#### 1. Versión de software ("software")

Este servicio solicita la versión del módulo de gestión del  $\mu$ C. El modo de uso es el siguiente:

[Cliente]  $\Rightarrow$  "software"  
[Servidor]  $\Rightarrow$  "VX.X"

El  $\mu$ C envía la versión de software con una "V" seguida de un número, un punto y otro número, por ejemplo "V3.0". La utilidad de este comando es doble. Por un lado pueden existir diferentes versiones con diferentes servicios, por lo cual debemos impedir en el software del cliente el uso de servicios no implementados. Por otro lado permite conocer el estado de la línea, es decir, si se solicita este

---

<sup>1</sup> El modo programación se habilita mediante el switch 1 de la placa TC-FPGA, ó mediante el programa de aplicación si este lo realiza.

servicio y se no se obtiene una respuesta adecuada, sabremos que la línea o el  $\mu$ C no se encuentran disponibles.

## 2. Cargar configuración de la FPGA (“FPGA”)

Este servicio nos permite enviar una configuración para la FPGA. Esta configuración siempre es almacenada en la zona de pruebas. El uso es el siguiente:

[Cliente]  $\Rightarrow$  “FPGA”  
[Servidor]  $\Rightarrow$  “F”

Una vez que se confirma el servicio, se puede enviar el archivo con la configuración. Para ello se enviará en paquetes de 64 bytes, esperando la confirmación para la recepción de un nuevo paquete. En el software del PC debe implementarse un timer que permita no dejar colgado el programa esperando una confirmación que puede no llegar por ruptura de las comunicaciones. Cuando el  $\mu$ C está preparado para recibir un nuevo paquete nos lo comunica mediante el envío de una “R”.

[Cliente]  $\Rightarrow$  Paquete de 64 bytes  
[Servidor]  $\Rightarrow$  “R”  
[Cliente]  $\Rightarrow$  Paquete de 64 bytes  
[Servidor]  $\Rightarrow$  “R”  
•  
•  
•  
Hasta el final del fichero

Por último debe tenerse en cuenta que el fichero con la configuración es del tipo \*.RBF (Raw Binary File), así pues debe ser leído como archivo binario, así mismo tendrá siempre un tamaño fijo de 14751 bytes. Por lo que se deben añadir 33 ceros al último paquete de 64 bytes antes de ser enviado. Como el tamaño del archivo siempre es el mismo, no es necesario identificar el final del fichero, ya que el programa gestor lo conoce. Por tanto, basta esperar la confirmación tras el último paquete para asegurar que este se recibió correctamente. En caso de fallo en las comunicaciones el programa no se habría almacenado correctamente, por lo que no se tomará por bueno y el programa de gestión<sup>1</sup> no intentará jamás configurar la FPGA con el mismo.

---

<sup>1</sup> Implementado en el  $\mu$ C de la TC\_FPGA

### 3. Cargar programa de aplicación para el **mC** (“PIC”)

Permite el envío de un programa de aplicación para el  $\mu$ C. El programa se ejecutará cuando el módulo de gestión se encuentre en el modo normal. Para pedir el servicio utilizamos:

[Cliente]  $\Rightarrow$  “PIC”  
[Servidor]  $\Rightarrow$  “P”

Tras la confirmación del servicio se puede enviar el fichero que contendrá el programa del microcontrolador. El fichero tendrá formato **INTEL 8 bits**<sup>1</sup>, este formato está compuesto por caracteres ASCII y cada línea del mismo tiene el siguiente aspecto:

**:BBAAAATTHHHH.....HHHCC**

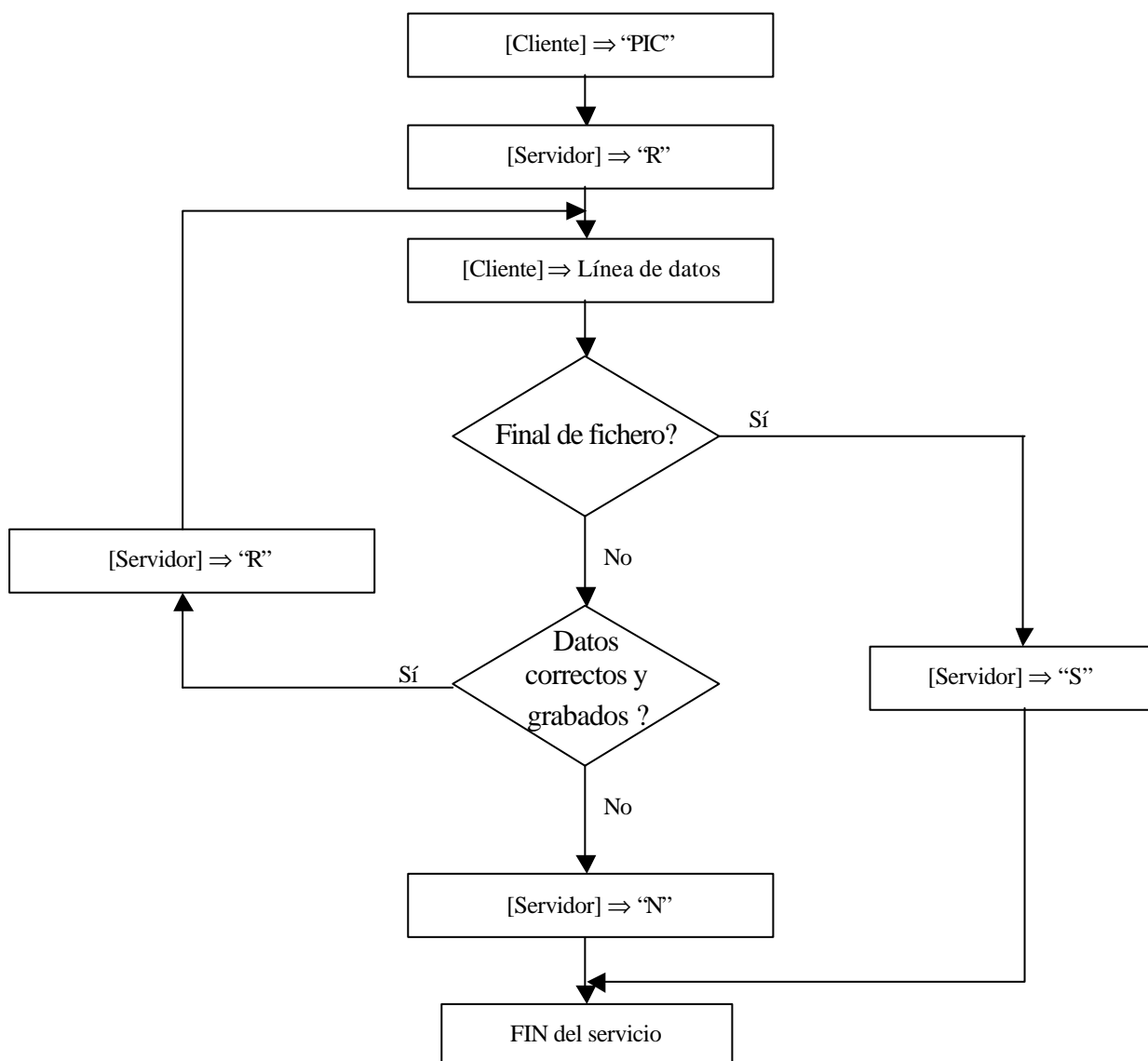
Cada línea de datos para grabar comienza con 9 caracteres y termina con 2 de suma de comprobación. Toda línea se inicia con el carácter “:”. El significado de cada elemento es:

- **BB** Representan el número de bytes de datos que aparecen en la línea en hexadecimal. Si lo dividimos entre 2 obtenemos el número de palabras (16 bits) de datos de la línea.
- **AAAA** Son 4 dígitos hexadecimales que indican la dirección de comienzo en que se deben grabar los datos que contiene la línea. En el caso de este  $\mu$ C los datos son de 16 bits, por lo que será necesario dividir la dirección entre 2 para hallar la dirección real de grabación.
- **TT** Indican el tipo de grabación, solo pueden tener 2 valores, 00 para grabación normal y 01 para indicar el final de fichero.
- **HHHH** Son 4 dígitos hexadecimales que representan una palabra de datos. Primero aparece el byte menos significativo y después el más significativo. Tendremos BB/2 palabras siguiendo a TT.
- **.....** Se ha colocado en el ejemplo de forma simbólica para indicar que el número de dígitos hexadecimales con formato **HHHH** es variable.
- **CC** Son los dígitos de control de errores mediante suma de comprobación. CC se obtiene realizando la suma en complemento a 2 de todos los datos de la línea, exceptuando el carácter “:”.

---

<sup>1</sup> Este es un formato estándar para los ficheros hexadecimales. El entorno de desarrollo MPLAB genera automáticamente este formato, salvo que se le indique otro. Para más información sobre este consultar la nota de aplicación AN732 de microchip, incluido en el CD.

Para enviar este fichero se realizará en paquetes, estando formado cada paquete por una línea completa de este formato. Por lo tanto el tamaño de cada paquete no será fijo, sino que dependerá del fichero generado. Igual que en el caso del envío a la FPGA será necesario recibir la confirmación para poder enviar el siguiente paquete. Pero en este caso el fichero tiene controles de integridad como es el *check sum* o que las direcciones de grabación sean correctas y el numero de datos en la línea corresponda con BB... Por ello el programa detecta estos posibles fallos para no programar una línea mal recibida. Por este motivo cambia el formato de transmisión:

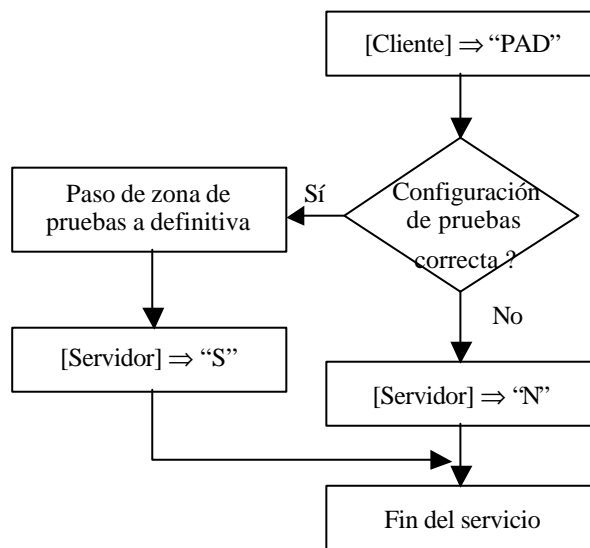


Por tanto el programa del PC deberá tener en cuenta tanto las posibles interrupciones del canal de comunicaciones como el fallos de las tramas. En una versión futura del software de gestión se preverá la solicitud de un paquete en caso de producirse errores, pues no se ha podido realizar en el proyecto por problemas de tiempo.

Por otro lado se realiza un control de los programas descargados, de modo que si un programa no se transfirió correctamente hasta el final, no se tomará como bueno y no intentará ejecutarlo.

#### 4. Hacer definitivo el programa de la zona de pruebas (“PAD”)

Todas las configuraciones descargadas para la FPGA son almacenadas en el banco de pruebas, de este modo se evita dejar el sistema sin ninguna configuración correcta. Por ello este es el único mecanismo que permite validar una configuración para pasarla a la zona definitiva de trabajo. El modo de solicitar el servicio es el siguiente:



Por tanto primero se comprueba que existe una configuración válida en la zona de pruebas y si este es el caso se pasa a la zona definitiva. En ambos casos se comunica el resultado al cliente.

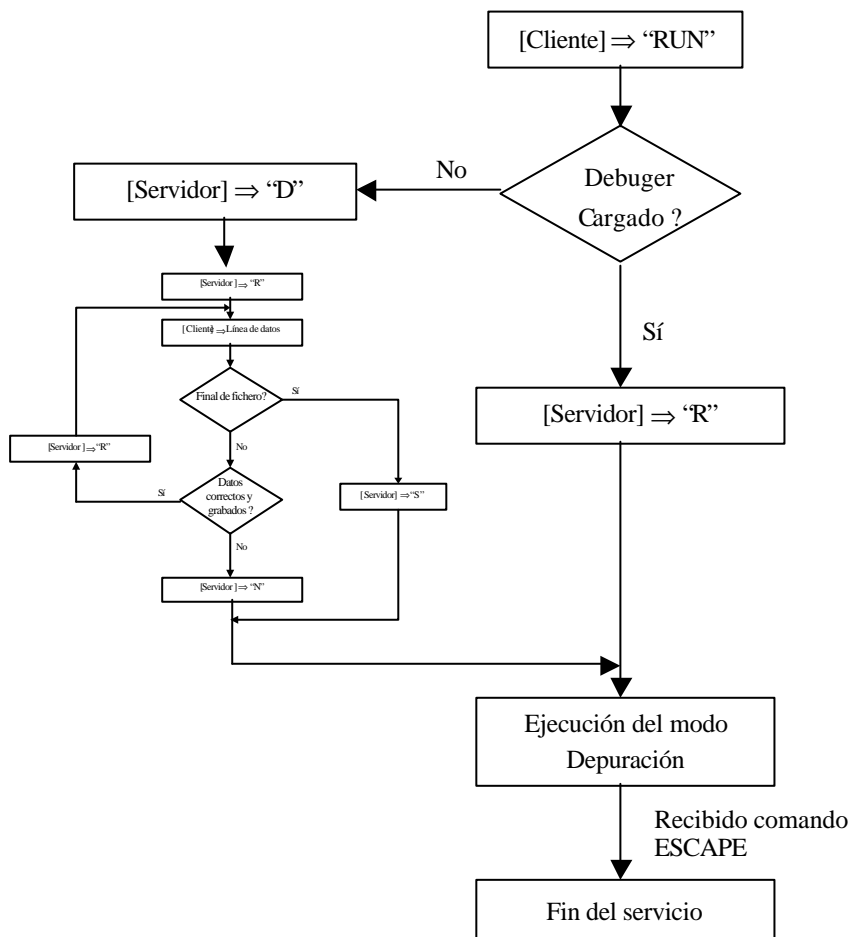
#### 5. Ejecutar el modo depuración (“RUN”)

En realidad el modulo de depuración debería estar contenido en el módulo de gestión, sin embargo se precisaba del uso de las interrupciones del  $\mu C$ , y por la arquitectura de este es imposible crear un módulo que permita descargar programas

sobre el propio  $\mu C$  usando al mismo tiempo las interrupciones si queremos seguir usándolas en la aplicación descargada.

Por este motivo el módulo de depuración debe ser cargado como un programa de aplicación para el  $\mu C$ . Lo que conlleva por un lado que un programa de aplicación ya cargado sea borrado, y que para cada depuración sea descargado el módulo de depuración en la zona de aplicaciones. Para solventar esto último, dicho módulo indica su presencia al módulo de gestión del  $\mu C$  mediante una variable de la memoria EEPROM de datos del  $\mu C$ . De este modo el gestor indicará al cliente que le descargue el módulo de gestión<sup>1</sup> únicamente si este no se encuentra presente.

Al contrario que los otros servicios, este último no finaliza hasta que el cliente lo decide. Para ello se debe enviar el comando "ESCAPE" del estándar ASCII que tiene el valor 0x18 en hexadecimal. Mostramos el funcionamiento del servicio:



<sup>1</sup> El módulo de gestión se encuentra en formato intel de 8 bits en el CD con el nombre "debugger.hex".

Como se puede observar se sigue el mismo protocolo para descargar el módulo de depuración que en el servicio de carga de aplicaciones para el  $\mu$ C, solo que no es necesario solicitar este servicio.

Ahora se describe el modo de entregar y recibir información con el módulo de depuración. Este es bastante sencillo, ya que solamente hay que enviar el contenido de los registros de lectura de la FPGA con un formato, el mismo formato se empleará para recibir la información de los registros de escritura de la FPGA, este es:

### \$XYYY

- \$ Carácter de sincronización, indica el inicio de un dato
- X Indica el numero del registro al cual se refieren los datos de la trama, se envía en formato ASCII y puede tomar los valores de 1 a 8.
- YYY Es el valor del registro en ASCII y formato decimal de 3 dígitos. Puede tomar valores de 000 a 255.

Este formato es seguido tanto para el envío por parte del Cliente como del Servidor. No es necesario enviar todos los registros ni tampoco hacerlo en orden. Para finalizar el servicio, se envía el carácter ESCAPE, como hemos indicado, en cualquier momento. Al finalizar este servicio se encuentran disponibles de nuevo todos los servicios del módulo de gestión.

### Tabla de Servicios

Para finalizar mostramos una tabla resumen de los comandos para los servicios del módulo de gestión (V3.0).

Servicio	Comando de solicitud "software"	Comando de confirmación "VX.X"	Comentarios
Versión software			Servicio para contemplar diferentes servicios con diferentes versiones
Descarga aplicación FPGA	"FPGA"	"F"	Descarga de aplicaciones a la zona de pruebas. Tras la confirmación se descarga el fichero *.rbf en paquetes de 64 bytes
Descarga aplicación $\mu$ C	"PIC"	"P"	Descarga de aplicaciones para el micro. Tras la confirmación se descarga en formato intel 8 bits
Paso a la zona definitiva	"PAD"	"S"	En caso de no poder ofrecer el servicio devuelve "N", ocurre cuando no hay programa en la zona de pruebas.
Modo depuración	"RUN"	"D" ó "R"	Con "D" es necesario descargar el programa "debuger.hex" antes de entrar en el modo.

Para más información sobre el comportamiento de estos módulos (gestión del PC, gestión del  $\mu$ C y debug) consultar los códigos fuente de estos programas, contenidos en el CD.