

# *Sistema Host USB para gestionar archivos entre dispositivos*

## *Implementación de un sistema Host USB en microcontrolador NXP.*

Luis Antonini; Jorge Osio\*; Jose Rapallini;  
Centro de Técnicas Analógico – Digitales (CeTAD)  
Facultad de Ingeniería – Universidad Nacional de La Plata  
La Plata, Argentina

\*Becario CIC – Comisión de Investigaciones Científicas de la Prov. de Bs. As.  
luis\_antonini@live.com.ar ; Jorge.osio@ing.unlp.edu.ar

*Resumen*— La interfaz USB es lo suficientemente versátil para una amplia gama de dispositivos periféricos, entre ellos los pertenecientes a la clase de dispositivos de almacenamiento masivo o “Mass Storage”. En esta clase de dispositivos se encuentran los pen drives. Aquí surge la motivación de realizar un sistema embebido portátil que permita acceder a su memoria.

Esta implementación se va a realizar utilizando el kit de desarrollo “Kit Educativo LPC2400”. En este kit se implementa un host USB a partir de un microcontrolador NXP, este dispositivo tiene los requerimientos de hardware para implementar el Host USB, un sistema de archivos FAT 32 y las funciones necesarias para controlar el sistema.

El sistema cuenta con una interfaz RS232 para visualización y control desde una PC. Se puede detectar los dispositivos conectados, determinar el espacio libre en dichos dispositivos, leer el listado de archivos y carpetas de cada dispositivo, intercambiar archivos de un dispositivo a otro entre otras funciones.

**Palabras Clave – Sistemas Embebidos, Host USB, Sistema de Archivos.**

### I. INTRODUCCIÓN

Los microcontroladores LPC2400 de NXP son ideales para todo tipo de aplicaciones en el área de comunicaciones, dado que cuentan en forma integrada con numerosos controladores con los cuales se pueden implementar interfaces Ethernet, RS232, USB, entre otras. Por ello la selección de esta familia de microcontroladores para la realización de esta implementación.

El microcontrolador LPC2478 [1] es el que se utiliza para esta aplicación porque es el único de esta familia que además cuenta con interfaz para display gráfico. Para la implementación se utiliza el “Kit Educativo LPC2400” antes nombrado.

Este kit cuenta con los requerimientos de hardware necesarios para la implementación del Host mediante dos conectores USB tipo A de acuerdo a la especificación USB 2.0 [2, pp. 93-98] y cuenta con una interfaz RS232 la cual se va a utilizar para visualización y control desde una PC.

### II. DIFERENCIAS ENTRE HOST Y DISPOSITIVOS USB

Se define un host USB como un dispositivo embebido, una PC o cualquier periférico que contenga hardware específico para la implementación de un host, un hub raíz con uno o más puertos USB y un programa para administrar las comunicaciones y los eventos del bus [3, pp.17-28].

Principalmente el host administra la alimentación y las comunicaciones en el bus, adjuntamente tiene las siguientes responsabilidades:

- **Detección de dispositivos:** Los hubs esperan que cualquier dispositivo USB sea conectado. En el proceso llamado enumeración, el host asigna una dirección y le solicita una serie de estructuras de datos llamados descriptores, los descriptores son propios de cada dispositivo. En cualquier momento cuando un dispositivo es conectado o removido, el host detecta el evento y enumera los nuevos dispositivos.
- **Provee alimentación:** El host provee alimentación a todos los dispositivos cuando se conectan. Algunos dispositivos utilizan la alimentación del bus, mientras otros tienen su propia fuente de alimentación.

- **Administración de tráfico del bus:** El host administra el tráfico de datos del bus dado que múltiples dispositivos quieren transferir datos al mismo tiempo. El controlador del host divide el tiempo disponible en segmentos llamados frames para la velocidad “full speed” o microframes para la velocidad “high speed”. El host otorga a cada transmisión una porción de un frame o microframe.
- **Chequeo de errores:** cuando se transmiten datos el host agrega bits para el chequeo de errores, cuando se reciben datos el host usa esos bits de chequeo de error para detectarlos.

A su vez, los dispositivos, son periféricos o hubs que se conectan al bus del host. Estos dispositivos contienen los circuitos y el código necesario para comunicarse con el host únicamente cuando este se lo solicite. No deben asumir que va a ocurrir ninguna secuencia particular de comunicación o eventos en el bus. En resumen los dispositivos deben:

- **Detectar la tensión del bus:** Una vez detectada la tensión de alimentación, deben cambiar la resistencia de pull-up para anunciar al host la presencia del dispositivo.
- **Administración de la alimentación:** En una operación normal un dispositivo debe limitar la corriente de alimentación a 100 mA, durante la enumeración este valor se puede incrementar a 500 mA.
- **Responder a solicitudes estándar:** Durante el encendido o la conexión al host, los dispositivos deben responder a las solicitudes estándar enviadas por el host durante la enumeración o incluso después de ella.
- **Manejo de errores:** Se adicionan bits para el chequeo de errores de la misma forma que el host.

### III. DESCRIPCIÓN DEL MICROCONTROLADOR LPC2478

Este microcontrolador tiene un núcleo con arquitectura ARM7TDMI-S, diseñado con un alto nivel de integración. Tiene integrado memoria flash de 512 kB de alta velocidad para el programa de aplicación, esta memoria incluye una interfaz especial de 128 bit lo que permite al CPU ejecutar instrucciones secuenciales desde la memoria flash a una velocidad máxima de 72 MHz.

El programa de aplicación se realiza utilizando el lenguaje de programación C y la aplicación *Keil uVision4*, una vez compilado el programa el microcontrolador se programa en sistema (ISP) utilizando la aplicación *Flash Magic*.

El sistema de archivos FAT 32 requiere un reloj de tiempo real para establecer la fecha y hora de creación o modificación de un archivo. Este microcontrolador cuenta con un bloque específico llamado RTC (“Real Time Clock”), el cual va a ser

utilizado en esta implementación para cumplir con este requerimiento.

El bloque controlador de Host USB se encuentra conformado por una interfaz de registros que cumple con la especificación OHCI (“Open Host Controller Interface”), una interfaz serie y un controlador DMA.

El diagrama en bloques del controlador USB se muestra en la figura 1, cuenta con las siguientes características [4, pp.389-390]:

- Cumple con la especificación USB 2.0 (full speed).
- Alimentación independiente de cada puerto.
- Posee memoria RAM integrada de 16 Kb.
- Soporta transferencias del tipo bulk, control, interrupt y Isochronous
- El bus soporta hot plugging, un-plugging y configuración dinámica de dispositivos.

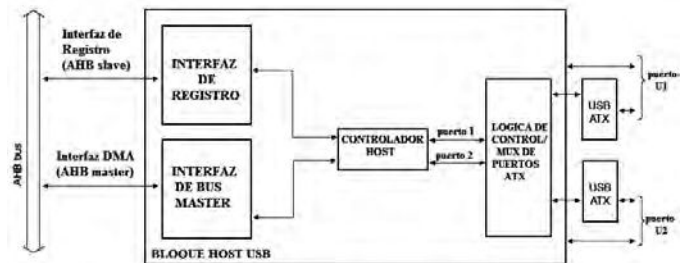


Figura 1. Diagrama en Bloques del controlador USB.

### IV. DESCRIPCIÓN DEL SISTEMA HOST USB

Esta implementación detecta, configura y se comunica únicamente con dispositivos USB pertenecientes a la clase de almacenamiento masivo o “Mass Storage” [5]. La memoria de estos dispositivos debe tener el formato de archivo del tipo FAT 12, 16 o 32.

Dicho programa está compuesto por cinco bloques, los cuales se muestran en la figura 2.

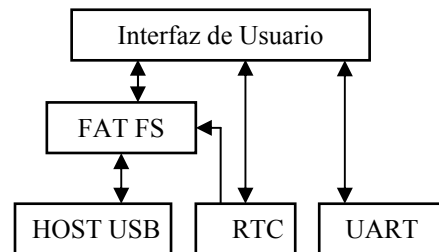


Figura 2. Diagrama en Bloques del programa de aplicación.

Descripción de los bloques:

- En el bloque Host USB se configura el controlador USB del microcontrolador y se desarrollan las funciones necesarias para enviar y recibir datos de dispositivos.
- En el bloque RTC se implementa el sistema de reloj en tiempo real.
- En el bloque UART se configura el controlador RS232 del microcontrolador para comunicarse con un PC.
- En el bloque FAT FS se implementa las funciones para operar con el sistema de archivos Fat 32.
- Finalmente en el bloque de interfaz de usuario se implementan las funciones necesarias para poder controlar el sistema desde una PC.

#### A. El módulo Host USB

Este bloque se divide en dos subsistemas. El primero es el encargado de configurar el controlador OHCI USB del microcontrolador e implementar las funciones para detectar, configurar y administrar las comunicaciones con distintos dispositivos USB conectados al bus.

Cuando se conecta un dispositivo USB al bus, el Host detecta automáticamente su conexión y da inicio al proceso de enumeración. Durante este proceso se corrobora la clase de dispositivo conectado al bus y en caso de no pertenecer a esta clase se inactiva.

En el segundo subsistema se implementan las funciones específicas de la clase de almacenamiento masivo. Una vez configurado el dispositivo, este subsistema otorga al bloque superior las funciones necesarias para comunicarse con los dispositivos. Para la comunicación se utiliza el tipo de transferencia bulk tal como las especificaciones establecen.

El bloque superior accede a la memoria del dispositivo especificando el número de cluster que se quiere leer o escribir.

#### B. Sistema de archivos FAT

El sistema de archivos se implementa en el bloque denominado "FAT FS". Para tal fin se utiliza el módulo genérico FAT FS disponible sin restricciones de uso.

Dicho módulo implementa todas las funciones necesarias para acceder a unidades de memoria con un sistema de archivo FAT 12, 16 ó 32. Este módulo es genérico porque es independiente del protocolo de comunicación utilizado para acceder a la memoria. Para su utilización simplemente hay que desarrollar las funciones necesarias para comunicarse con el tipo de interfaz utilizada.

Para esta implementación se desarrollaron cinco funciones, cuatro de ellas se relacionan con el bloque HOST USB y la restante con el bloque RTC. Las funciones destinadas al primer

bloque inicializan la unidad, determinan su estado y finalmente leen o escriben en su memoria.

La función destinada al bloque RTC obtiene la fecha y la hora en el formato que requiere el sistema FAT FS. Una vez otorgadas dichas funciones el módulo está listo para acceder a la memoria de dispositivos USB conectados al host.

El módulo FAT FS brinda al bloque superior una serie de funciones para acceder y gestionar los archivos y carpetas existentes en la unidad.

#### C. La interfaz RS232

Esta interfaz se implementa en el bloque UART y su objetivo es poder visualizar y controlar el sistema desde una PC. En el kit de desarrollo se implementa la interfaz número cero ya que es la utilizada para la programación del microcontrolador.

Esta interfaz se configura seleccionando adecuadamente el divisor del clock para generar el baud rate apropiado (9600 para esta aplicación). Luego para enviar o recibir un carácter sólo es necesario escribir o leer un registro determinado. La interfaz automáticamente envía o recibe el carácter.

Utilizando esos registros se implementan en este bloque las funciones necesarias para otorgarle al bloque superior la capacidad de recibir o transmitir a un PC un carácter o una serie de caracteres.

En la PC se recibe o envían los caracteres utilizando la aplicación Hyper Terminal o cualquiera similar que pueda acceder al puerto serie.

#### D. La interfaz de usuario

Este bloque es el encargado de controlar el sistema, recibe del puerto serie los comandos a ejecutar retornando el resultado obtenido.

Cuando el usuario conecta al Host un dispositivo USB de la clase de almacenamiento masivo, este lo detecta automáticamente y lo configura quedando listo para gestionar los archivos.

El usuario puede listar y acceder a todos los archivos y carpetas existentes en el directorio raíz o dentro de una carpeta del dispositivo USB conectado. Puede leer cualquier archivo de texto sin importar su tamaño. Tiene la posibilidad de crear carpetas y eliminarlas. También puede eliminar archivos.

Otra posibilidad es copiar un archivo, ya sea de una carpeta a otra en un mismo dispositivo USB o entre dos dispositivos USB distintos.

Los archivos en el sistema FAT tienen ciertos atributos, como la fecha de creación, el tipo de archivo, si es un archivo oculto, solo lectura, etc. El usuario puede acceder a estas características con solo introducir el nombre de archivo o carpeta. También, puede obtener el espacio disponible de la unidad conectada a la interfaz USB.

Finalizada la gestión de archivos el usuario puede retirar en forma segura su dispositivo USB.

Adicionalmente, el usuario puede acceder a la hora y fecha actual y si lo desea puede modificarla o ajustarla.

### V. RESULTADOS OBTENIDOS

En la figura 3 se muestra el sistema Host USB en funcionamiento implementado en el kit de desarrollo antes citado. Se han conectado en sus dos interfaces USB pen drives con diferentes características.



Figura 3. Imagen de la placa de desarrollo con los pen drives conectados.

Al encender la placa de desarrollo, en la ventana del Hyper Terminal, se visualiza un mensaje de bienvenida con la hora y fecha actual. Se indica al usuario que conecte un dispositivo USB y luego presione la tecla “ENTER” para que el sistema configure dicho dispositivo. El sistema automáticamente detecta en que puerto se conectó el dispositivo y lo configura.

Una vez finalizada la configuración, de ser exitosa, se muestra en pantalla el espacio disponible y el espacio total de la unidad, se indica además que la unidad está lista para utilizarse. De ocurrir algún tipo de error se muestra en pantalla de que error se trata.

En la figura 4 se pueden ver los mensajes obtenidos durante el inicio y configuración de un dispositivo mediante un software de interfaz serial.

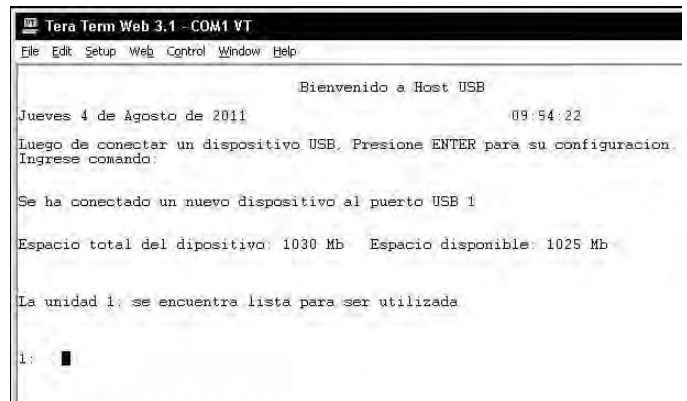


Figura 4. Mensajes obtenidos en la pantalla del Hyper Terminal.

El sistema tiene un comando de ayuda denominado “help” para asistir al usuario, al introducir este comando se muestran en pantalla todos los comandos disponibles y la forma correcta de introducirlos.

Así mismo, cada comando, de necesitar un parámetro adicional lo solicita en forma clara y siempre chequeando si se ingresa correctamente. De ocurrir esta situación se imprime en pantalla un mensaje explicativo del error. El resultado obtenido de introducir un comando se muestra en la figura 5, en este caso se listaron los archivos de la unidad introduciendo el comando “dir”.

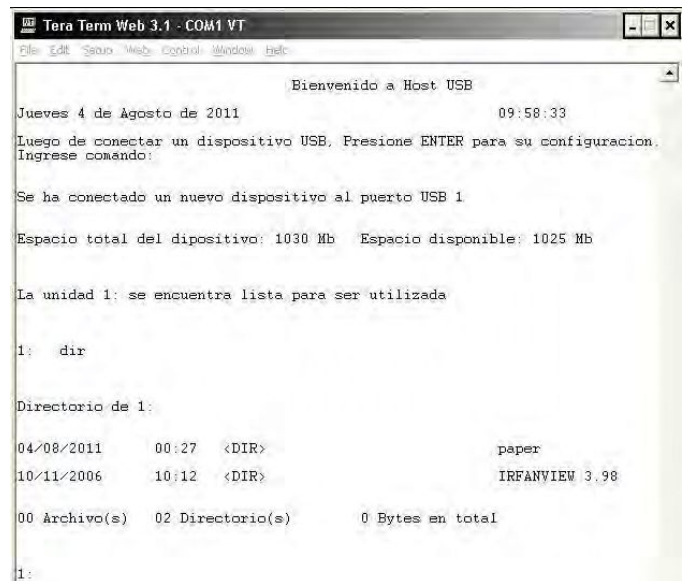


Figura 5. Resultado obtenido al ingresar el comando “dir”.

### VI. CONCLUSIONES

Finalizado el desarrollo del sistema Host USB se realizó el debug del mismo. Las pruebas consistieron en testear el funcionamiento de cada una de las funciones implementadas en el bloque de interfaz de usuario, los resultados obtenidos superaron ampliamente las expectativas. Se logró copiar, crear, leer, renombrar archivos o carpetas en un intervalo de tiempo relativamente corto, similar al que se obtiene

utilizando una PC. Los archivos con que se trabajó en estas pruebas no superaron en tamaño el orden de los Kbytes.

La última prueba relacionada a la implementación del Host USB consistió en transferir un archivo de un pen drive a otro. Se utilizó para esta prueba un archivo de audio con un tamaño total de 9.8 MBytes. Si bien el archivo se copio exitosamente, requirió un tiempo aproximado de tres minutos. El tiempo requerido superó ampliamente el esperado. Este es un aspecto a mejorar constituyendo una nueva tarea a futuro, posiblemente se mejore sustancialmente con una optimización del código del programa.

En relación al microcontrolador empleado los resultados superaron las expectativas. La programación, configuración y utilización de cada controlador integrado es sencilla, la única dificultad que recurrentemente se presentó fue la escasa información sobre algunos aspectos de configuración y los numerosos errores presentes en los códigos encontrados en notas de aplicación y ejemplos de programación.

## VII. TRABAJO A FUTURO

Como primer trabajo a futuro se deberá mejorar el tiempo de transferencia de archivos entre dispositivos optimizando las funciones que manejan los buffers USB para la transferencia.

### REFERENCIAS

- [1] "Data sheet del producto LPC2478", [http://www.nxp.com/documents/data\\_sheet/LPC2478.pdf](http://www.nxp.com/documents/data_sheet/LPC2478.pdf), 29 de septiembre de 2010.
- [2] *Universal Serial Bus Specification*, Compaq, Hewlett-Packard, Intel, Lucent, Microsoft, NEC, Philips, 2000.

Por otro lado se convertirá el sistema en un dispositivo portátil, lo cual se va a lograr en dos etapas.

La primera consiste en la implementación de un teclado en el sistema de forma de ingresar los comandos sin necesidad de utilizar una PC.

La segunda etapa consiste en implementar un display gráfico de forma de poder visualizar los resultados. Se pretende utilizar el controlador de display con el que cuenta el microcontrolador. Una vez concluidas estas etapas el sistema será totalmente portable.

Además, dado que el kit de desarrollo tiene una interfaz SD, se pretende implementar el controlador para dicha interfaz de forma de poder acceder a la memoria de estas tarjetas y poder gestionar sus archivos y copiarlos a los dispositivos USB o desde ellos.

- [3] Jan Axelson, *USB Complete*. Madison, Lakeview research, 1999-2005.
- [4] "UM10237 LPC24XX User Manual", [http://www.nxp.com/documents/user\\_manual/UM10237.pdf](http://www.nxp.com/documents/user_manual/UM10237.pdf), 26 de agosto de 2009.
- [5] Jan Axelson, *USB Mass Storage*. Madison, Lakeview research, 2006.