Protocolo

USB

INGENIERIA EN MICROCONTROLADORES

Protocolo USB (UNIVERSAL SERIAL BUS)

Teoría

PROTOCOLO USB

www.i-micro.com

Teléfono 044 55 11 29 55 05
E-mail: cursos@i-micro.com
elp@i-micro.com

Capitulo 1

1

El Protocolo USB

Escrito por: **Eric López Pérez**
Ing. En Comunicaciones y Electrónica
México D.F. elp@i-micro.com

EL siguiente articulo, tiene como objetivo, proporcionar las bases teóricas que hacen posible el entendimiento de este protocolo, que rápidamente esta desplazando al ya conocido RS-232.

Mi principal función al escribir este articulo es sentar las bases a ingenieros, estudiantes, hobistas, etc, para que puedan realizar en futuro diversas aplicaciones utilizando este protocolo.

Este articulo es una recopilación de diversas fuentes, en donde la mayoría se encuentran en ingles por lo que en algunos casos resulta difícil de entender, si no se cuenta con los conocimientos necesarios en programación de microcontroladores, redes, programación en algún lenguaje de alto nivel y ensamblador, etc.

Debido a la complejidad para entender este protocolo, este articulo esta dividido en tres partes donde la primera se puede descargar gratis de nuestro web site. A su vez se le extiende una cordial invitación a nuestro curso **Protocolo USB para aplicaciones Electrónicas.**

Para poder desarrollar cualquier aplicación se deben seleccionar una familia de microcontroladores la cual cuente con CI que contengan registros propio para el manejo de este protocolo. En este caso se escogieron los microcontroladores **PIC de la empresa Microchip.**

Requerimientos Para desarrollar una aplicación USB

- Un microcontrolador o CI que soporte la interfase USB
- Un programa sobre el periférico para transmitir la información en cualquier lenguaje de Programación.
- Una computadora que con puertos Usb.
- Conocimientos en electrónica digital.
- Conocimientos en Microcontroladores.
- Herramientas para programar los Microcontroladores.
- Conocimiento en Redes de Microcontroladores, (Otros protocolos por ejemplo RS-232, RS-485)
- Teoría del Protocolo USB.
- Diseño de Hardware .
- Conocimientos en lenguaje de programación de Alto nivel (Vbasic, VC++, etc)
- Conocimientos en Windows.

Además de contar con las herramientas necesarias como lo son:
Un osciloscopio, una computadora con puerto USB, borrador de memorias, algún kit de desarrollo, etc.

Las principales características del bus son :

- Banda de paso, disponibilidad desde algunos kilobits a varios megabits;
- Transferencia isócrona y asíncrona en el mismo bus;
- Varios tipos de periféricos en el mismo bus;
- Posibilidad de conectar hasta 127 perifericos;
- Tiempo de respuesta garantizado (para audio y vídeo)
- Flexibilidad a nivel de banda de paso;
- Fiabilidad, control de errores;

- Perfectamente integrado en el PC, *plug and play* (conectar y usar)
- Coste reducido en la versión de baja velocidad (1,5 Mbits/s);
- Posible expansión del bus

Descripción del sistema USB

El USB es un bus punto a punto: dado que el lugar de partida es el *host (PC o hub*), el destino es un periférico u otro *hub.* No hay más que un único *host* (PC) en una arquitectura USB.

Los PC estándar tienen dos tomas USB, lo que implica que, para permitir más de dos periférico simultáneamente, es necesario un *hub.* Algunos periféricos incluyen un *hub* integrado, por ejemplo, el teclado USB, al que se le puede conectar un Mouse USB.

Los periféricos comparten la banda de paso del USB. El protocolo se basa en el llamado *paso de testigo (token).* El ordenados proporciona el testigo al periférico seleccionado y seguidamente, éste le devuelve el testigo en su respuesta.

Este bus permite la conexión y la des-conexión en cualquier momento sin necesidad de apagar el equipo.

A continuación se describen los primeros aspectos de este protocolo.

Interfaz física

Aspecto eléctrico

A nivel eléctrico, el cable USB transfiere la señal y la alimentación **sobre 4 hilos.**

A nivel de alimentación, el cable proporciona la tensión nominal de 5 V. Es necesario definir correctamente el diámetro del hilo con el fin de que no se produzca una caída de tensión demasiado importante en el cable. Una resistencia de terminación instalada en la línea de datos permite detectar el puerto y conocer su configuración (1,5 o 12 Mbits/s).

A nivel de señal, se trata de un par trenzado con una impedancia característica de 90 Ω La velocidad puede ser tanto de 12 Mbits/s como de 1,5 Mbits/s. La sensibilidad del receptor puede ser de, al menos, 200mV y debe poder admitir un buen factor de rechazo de tensión en modo común. El reloj se transmite en el flow de datos, la codificación es de tipo NRZI, existiendo un dispositivo que genera un bit de relleno (bit stuffing) que garantiza que la frecuencia de reloj permanezca constante. Cada paquete va precedido por un campo de sincronismo.

Consumo

Cada sección puede proporcionar una determinada potencia máxima siendo el PC el encargado de suministrar la energía. Además, el periférico puede estar autoalimentado ***(self powered).***

Control de consumo
El ordenador gestiona el consumo, teniendo capacidad de poner en reposo (*suspend)* o en marcha a un periférico USB. En reposo, este reduce su consumo (si puede),quedándose la parte USB funcional. Esta gestión está orientada especialmente a los equipos portátiles.

Terminología USB

Host: Dispositivo maestro que inicia la comunicación (Generalmente la computadora).
Hub: Dispositivo que contiene uno o mas conectores o conexiones internas hacia otros dispositivos usb, el cual habilita la comunicación entre el host y con diversos dispositivos. Cada conector representa un puerto USB.

Dispositivo compuesto: Es aquel dispositivo con múltiples interfaces independientes. Cada una tiene una dirección sobre el bus pera cada interfase puede tener un diferente driver device en el host.

Puerto USB: Cada host soporta solo un bus, cada conector en el bus representa un puerto USB por lo tanto sobre le bus puede haber un varios conectores , pero solo existe una ruta y solo un dispositivo puede transmitir información a un tiempo.

Driver: es un programa que habilita aplicaciones para poderse comunicar con el dispositivo. Cada dispositivo sobre el bus debe tener un driver, algunos periféricos utilizan los drivers que trae Windows.

Puntos terminales (Endpoints): Es una localidad especifica dentro del dispositivo. El **Endpoint** es un buffer que almacena múltiples bytes, típicamente es un bloque de la memoria de datos o un registro dentro del microcontrolador. Todos lo dispositivos deben soportar el punto terminal 0. Este punto terminal es el que recibe todo el control y la peticiones de estado durante la enumeración cuando el dispositivo esta sobre el bus.

Tuberías (Pipes): Es un enlace virtual entre el host (la PC) y el dispositivo USB, este enlace configura los parámetros asociados con el ancho de banda que tipo de transferencia se va a utilizar (**Control, Bulk, Isocrona o Interrupt**) dirección del flujo de datos y el máximo y/o mínimo tamaño de los paquetes/buffers.
Cada enlace está caracterizado por su banda de paso (Token), su tipo de servicio, el número de punto terminal (End Point) y el tamaño de los paquetes.

Estos enlaces se definen y crean durante la inicialización del USB . Siempre existe un enlace virtual 0 que permite tener acceso a la información de configuración del periférico USB (estado, control e información).

La norma USB define 2 tipos de enlaces virtuales (pipe); *stream y message*.

Stream Pipes: se trata de un flujo sin formato USB definido, esto significa que se puede enviar cualquier tipo de dato. Este tipo de pipe soporta las transferencias **bulk, isocronas, y interrupt**. Además tanto el host como el dispositivo USB pueden controlar.

Message Pipes: este tipo de enlace virtual si tiene un formato USB definido y solo puede soportar la transferencia **Control**.

TIPOS DE TRANSFERECIA

El enlace virtual (pipe) puede ser de cuatro tipos:

Control: Modo utilizado para realizar configuraciones: existe siempre sobre el Punto terminal 0 (EndPoint 0). Todos los dispositivos USB deben soportar este tipo de transferencia.

Los datos de control sirven para configurar el periférico en el momento de conectarse al USB. Algunos drivers específicos pueden utilizar este enlace para transmitir su propia información de control. Este enlace no tiene pérdida de datos, puesto que los dispositivos de detección de recuperación de errores están activos a nivel USB.

Bulk: Este modo se utiliza para la transmisión de importantes cantidades de información. Como el tipo control, este enlace no tiene pérdida de datos. Este tipo de transferencia es útil cuando la razón de transferencia no es critica como por ejemplo , el envió de un archivo a imprimir o la recepción de datos desde un escáner.
En estas aplicaciones, la transferencia es rápida, pero puede espera si fuera necesario.
Solo los dispositivos de media y alta velocidad utilizan este tipo de transferencia.

Interrupt, modo utilizado para transmisiones de pequeños paquetes, rápidos, orientados a percepciones humanas (ratón, punteros).
Este tipo de transferencia son para dispositivos que deben recibir atención periódicamente y lo utilizan los dispositivos de baja velocidad

Este tipo de transmisión garantiza la transferencia de pequeñas cantidades de datos. El tiempo de respuesta no puede ser inferior al valor especificado por la interfaz. El ratón o cualquier otro dispositivo apuntador es una aplicación típica de este modo de transmisión.

Isochronous o Flujo en tiempo real: modo utilizado para la transmisión de audio o video comprimido. Este tipo de transmisión funciona en tiempo real. Este es el modo de mayor prioridad.

La transmisión de la voz es un ejemplo de esta aplicación. Si ésta no se transmite correctamente, pueden llegar a oírse parásitos (*glich)* y la aplicación puede detectar ciertos errores de los llamados *underruns*

Enumeración

Cuando se conecta un dispositivo USB a la PC se produce el **Proceso de Enumeración,** el cual consiste en que el host le pregunta al dispositivo que se presente y le diga cuales son sus parámetros, tales como:

- Consumo de energía expresada en unidades de Carga
- Numero y tipos de Puntos terminales
- Clase del producto.
- Tipo de transferencia
- Razón de escrutinio, etc.

El proceso de enumeración es inicializado por el host cuando detecta que un nuevo dispositivo que ha sido adjuntado al Bus. El host le asigna una dirección al dispositivo adjuntado al bus y habilita su configuración permitiendo la transferencia de datos sobre el bus.

Bibliografía

EL BUS USB
Guía del Desarrollador
Xavier Fenard
Editorial: Paraninfo

MICROCHIP
Usb Firmware Users Guide
Datasheet PIC16C745/765
8-bit CMOS Microcontrollers with USB

USB in a Nutshell
www.beyondlogic.org

USB DESIGN BY EXAMPLE
A practical guide to building I/O devices
John Hyde Wiley