



Aplicaciones para USB con Microcontroladores PIC18F4550

www.i-micro.com



Objetivos

- **Los Asistentes aprenderán los fundamentos del Protocolo USB.**
- **Entenderán las consideraciones mas importantes para una Aplicación USB**
- **Conocerán las Capacidades de las familias PIC 18F4550**
- **Conocerán los diferentes tipos de Clases de Dispositivos USB**
- **Aprenderán a manejar el Stack de Microchip**



Agenda

- **Breve Historia del Protocolo USB (características)**
- **Conceptos Básicos**
 - **Requisitos (Conocimientos Necesarios)**
 - **Principales Características del Bus**
 - **Terminología**
 - **Arquitectura**
 - **Clase de Dispositivos**
 - **Tipos de Tránsito de Datos**
- **Introducción a la Tarjeta de Evaluación FS USB**
- **USB Microchip Firmware**
- **Vbasic.Net Firmware**
- **Conversión de USB a RS232**
- **Dispositivos de Almacenamiento Masivo**



USB Wireless





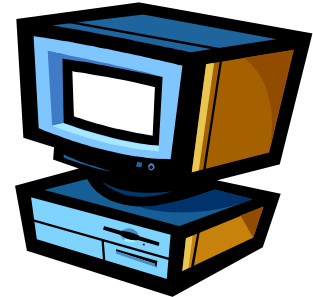
Requisitos

- **Tener conocimientos de un Microcontrolador PIC de Microchip**
- **Conocimiento en Otros Protocolos de Comunicación,**
(Por ejemplo RS-232, RS-485, I2C, SPI, ONE WIRE, etc.)
- **Teoría del Protocolo USB.**
- **Conocimientos en cualquier Lenguaje de programación de Alto nivel (Vbasic, VC++, Delphi, etc).**
- **Sistema de Desarrollo que facilite la programación del Microcontrolador.**
- **Adquirir un Sniffer**
- **Conocimientos en Windows Y Drivers (DDK= Driver Development kit).**



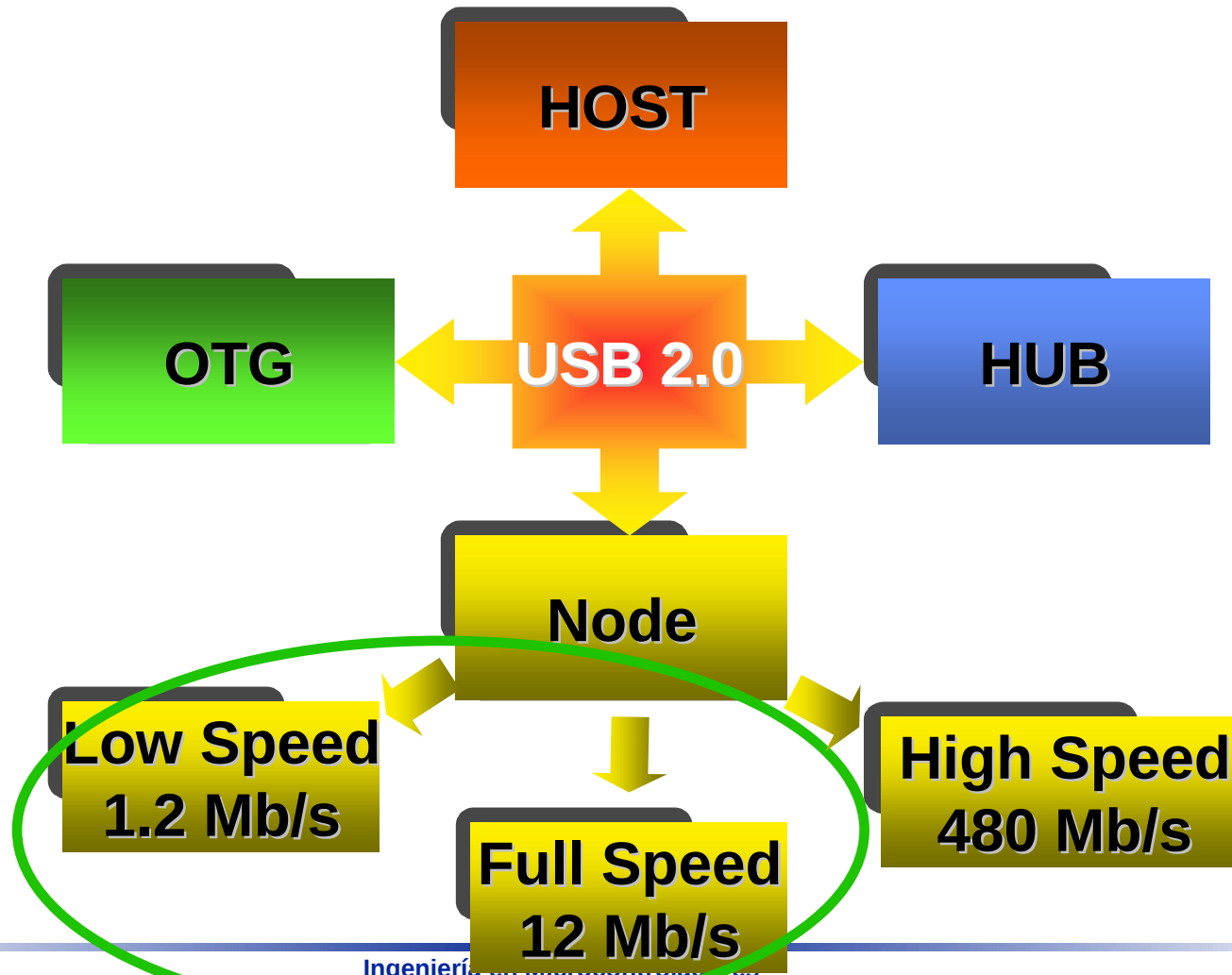
Principales **Características del Bus**

- Auto detección & configuración (Plug & play)
- Fácil expansión utilizando Hubs
- Alimentación propia del Bus
- Posibilidad de conectar hasta 127 periféricos
- Tres tipos de Velocidades
 - ✓ Baja (Low Speed) – 1.5 Mb/s
 - ✓ Mediana (Full Speed)-12 Mb/s
 - ✓ Alta (High Speed) 480 Mb/s



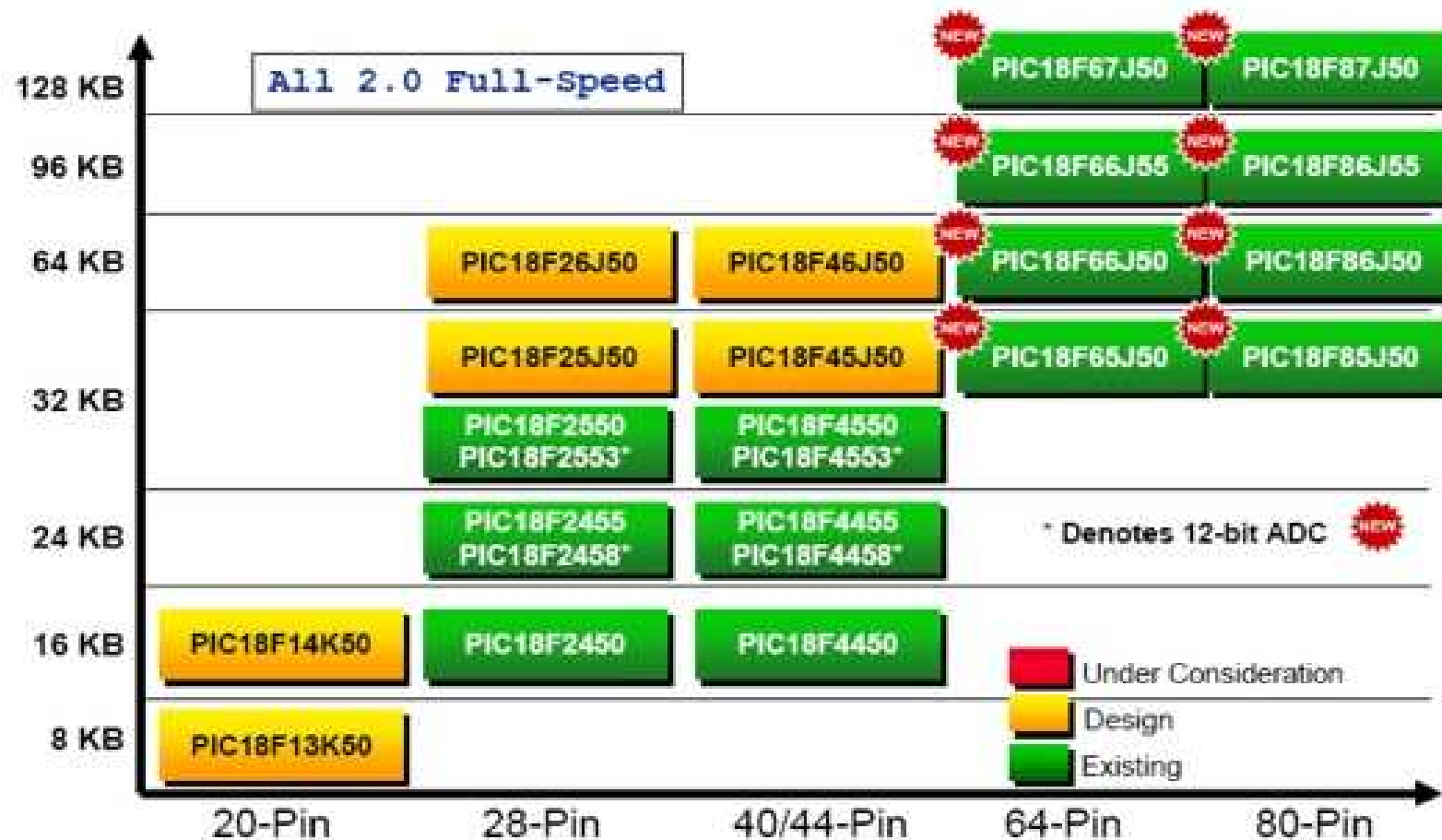


Dispositivos USB





Numero de Parte de Microchip





Historia

- El Protocolo Usb Fue co-desarrollado por un conjunto de compañías como: Compac, Intel, Microsoft, Nec, etc.

- 1991 USB 1.1

- 2000 USB 2.0

- Se necesita de una licencia o Vendor ID (Numero de 16 bits), Requerida para que un producto USB salga al mercado.

- <http://www.usb.org/developers/vendor>

- \$1500 dll



- Problemas legales y Técnicos y si no se usa una VID

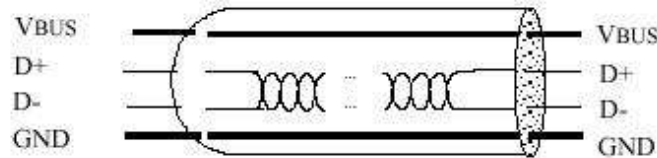
- Se de Contar con un Product ID (PID) = Un numero de 16 bits

- Pasar Pruebas

- Para usar el logotipo se debe pagar Aprox. \$1500 dll



Interfaz Fisica



- A nivel eléctrico, el cable USB transfiere la señal y la alimentación sobre 4 hilos.
- Utiliza una Transmisión de tipo Diferencial
- Consumo: Puede proporcionar una determinada potencia máxima siendo la PC la encargada de suministrar la energía.
(500mA max).
- Además, el periférico puede estar Autoalimentado (Self powered).

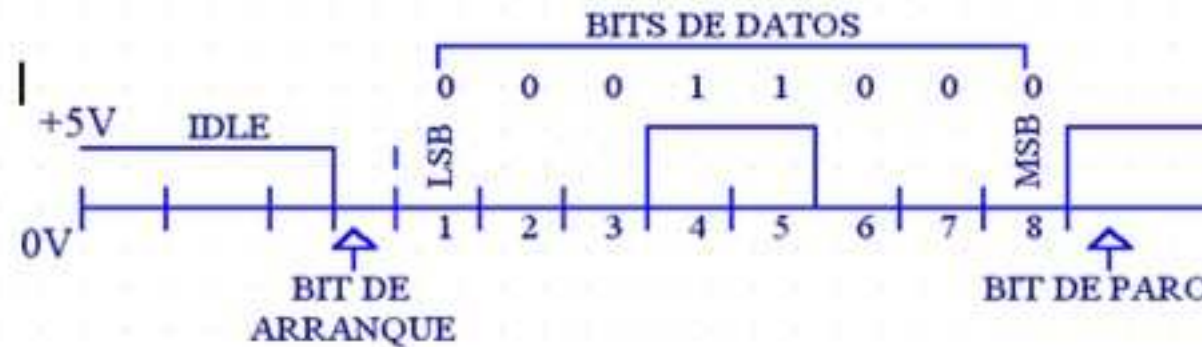


Velocidad de Transmisión en RS-232 @ 9600bps

**¿Cuántos Bytes de información
podemos transmitir en 1 seg @
9600 bps?**



Velocidad de Transmisión en RS-232 @ 9600bps



1seg $\rightarrow$ 9600bits

X $\rightarrow$ 1 bit

1.041ms = 1byte

1 seg = X

1bit = 104.16 us

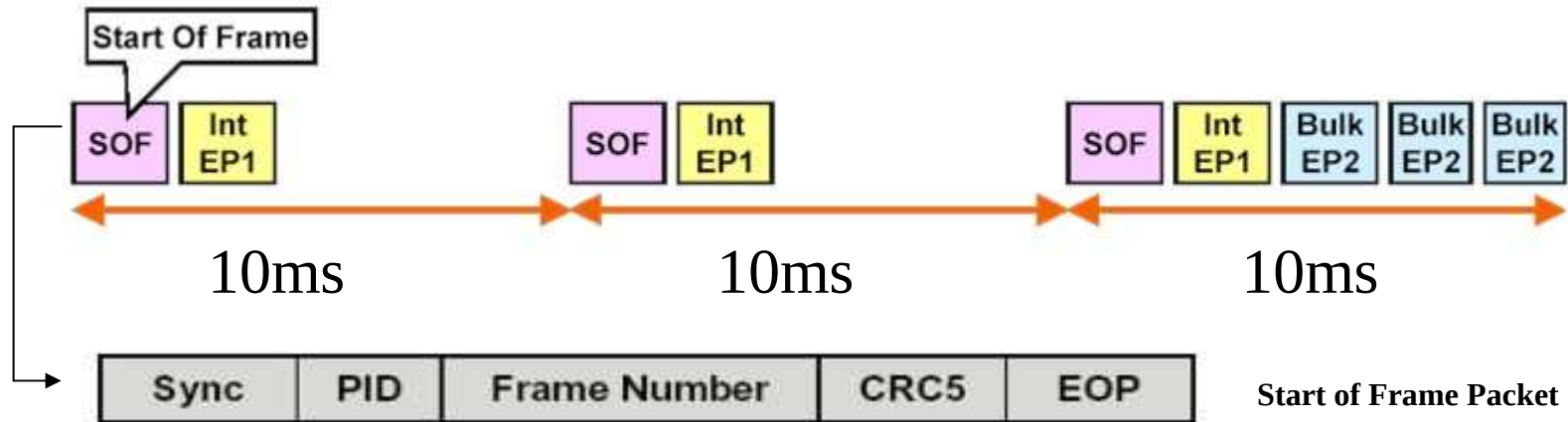
104.16us * 10 bits = 1.041ms = 1byte

Aprox= 960 bytes en 1 seg

Si y solo Si el BIT de Arranque y Paro tienen el mismo ancho



Trama USB para Un Dispositivo Low Speed





Velocidad de Transmisión en un Dispositivo Low Speed

Mito: Un dispositivo de **Baja Velocidad** puede transferir datos arriba de 187.5 KB/s (1.5Mb/s).

Verdad: Imposible debido a restricciones en el protocolo USB (Ver especificaciones)

- **8 bytes se transfieren cada 10ms** = 800 Bytes/seg únicamente.

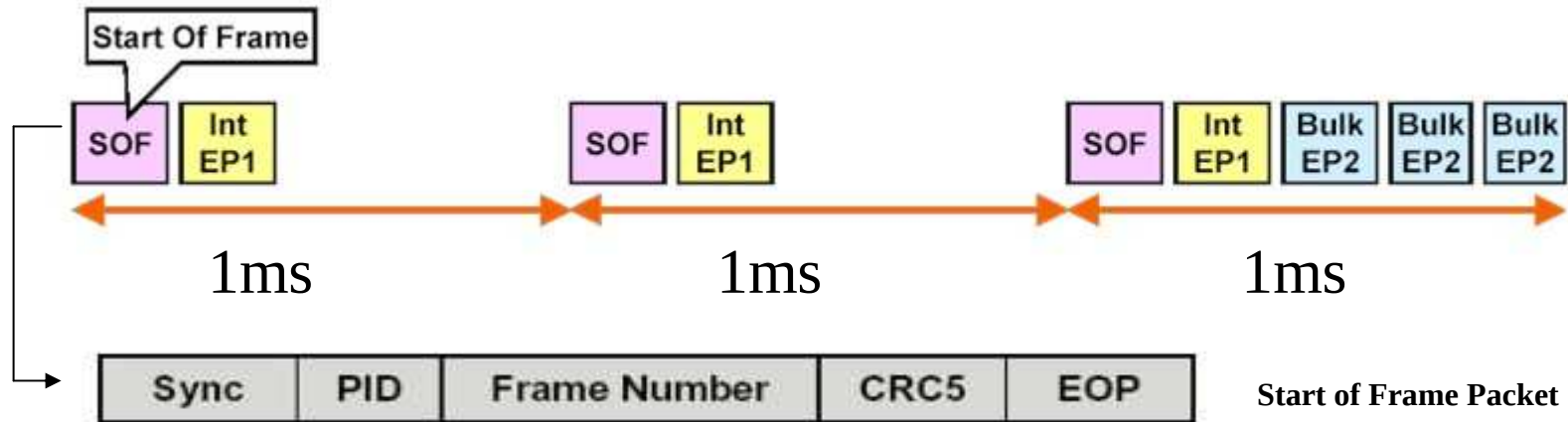
- 8 bytes $\rightarrow$ 10ms = 64bits $\rightarrow$ 10ms

? $\rightarrow$ **1seg = 6400 bits = 800bytes**

- Ejm: un Raton, un Teclado, Joystick, etc.



Trama USB para Un Dispositivo Full Speed



**Un dispositivo USB Full Speed solo puede
Transmitir a lo mas**

64 Kbytes en 1s

Nota: Depende del tipo de transferencia



Velocidad de Transmisión en un Dispositivo Full Speed

Mito: Un dispositivo FULL Speed puede transferir datos arriba de 1.5 MB/s a 12Mb/s.

Verdad: Imposible 1.5Mb/s es el total del ancho de banda

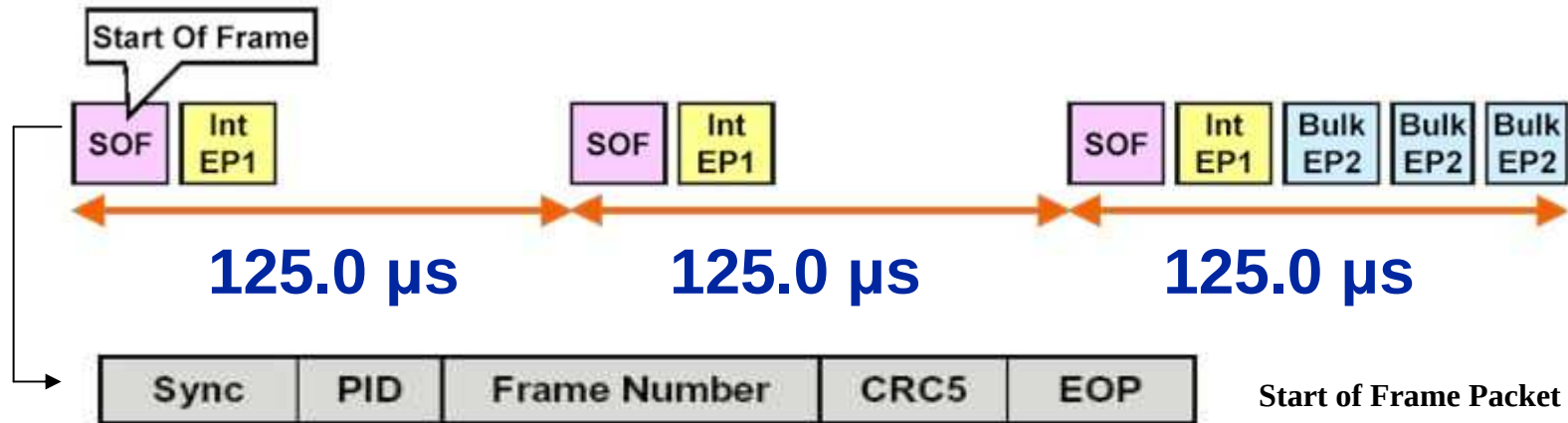
- Se debe compartir con otros periféricos
- Restricciones por parte del protocolo.

64 bytes en 1ms = 512 bits en 1ms = 512000 bits en 1 seg

- Solo 64KB/s en algunos casos
(Depende del tipo de Transferencia)



Trama USB para Un Dispositivo High Speed



Un dispositivo **USB High Speed** solo puede Transmitir a los mas 64Bytes en 125us

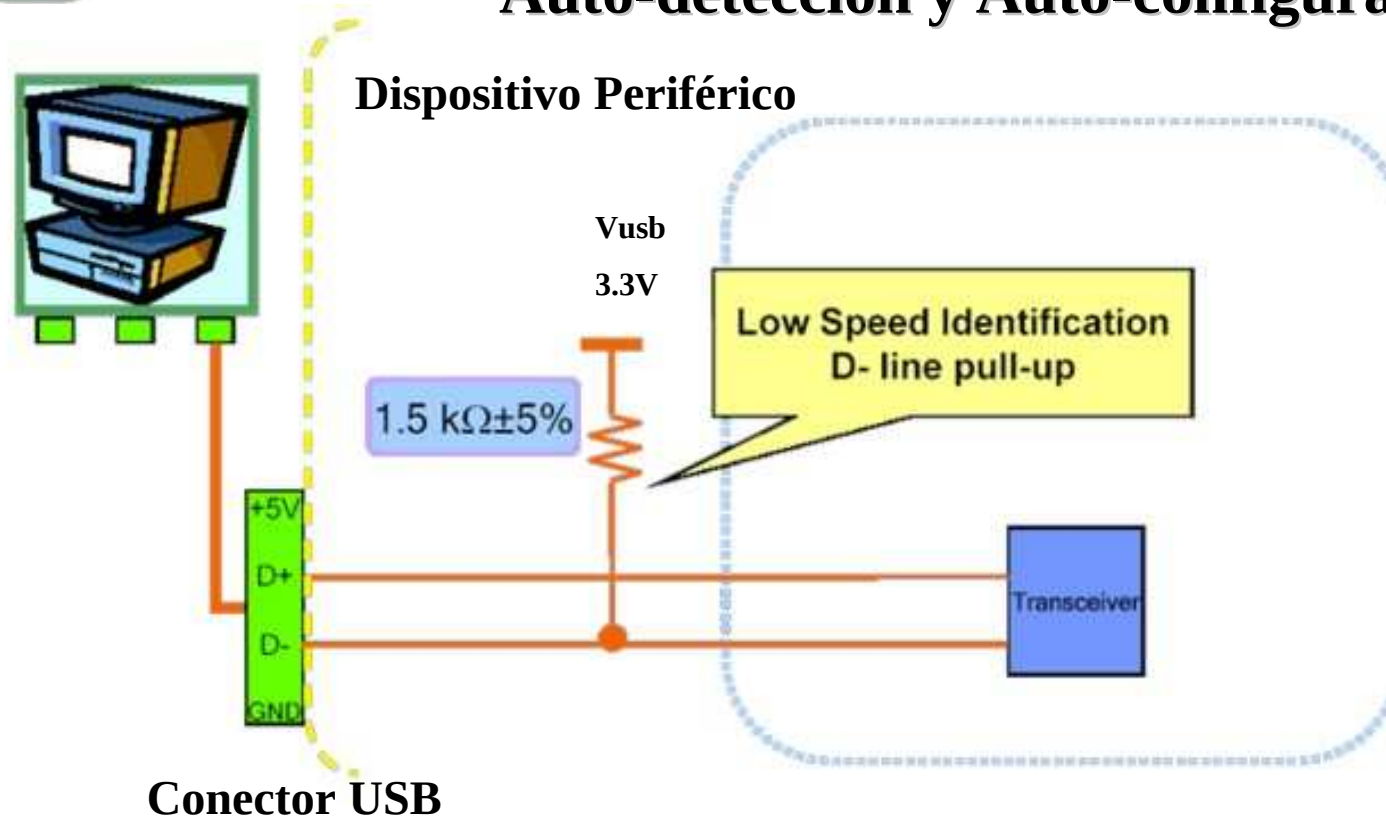
512kbytes en 1 seg

Nota: Depende del tipo de transferencia



La Magia de Plug & Play

Auto-detección y Auto-configuración



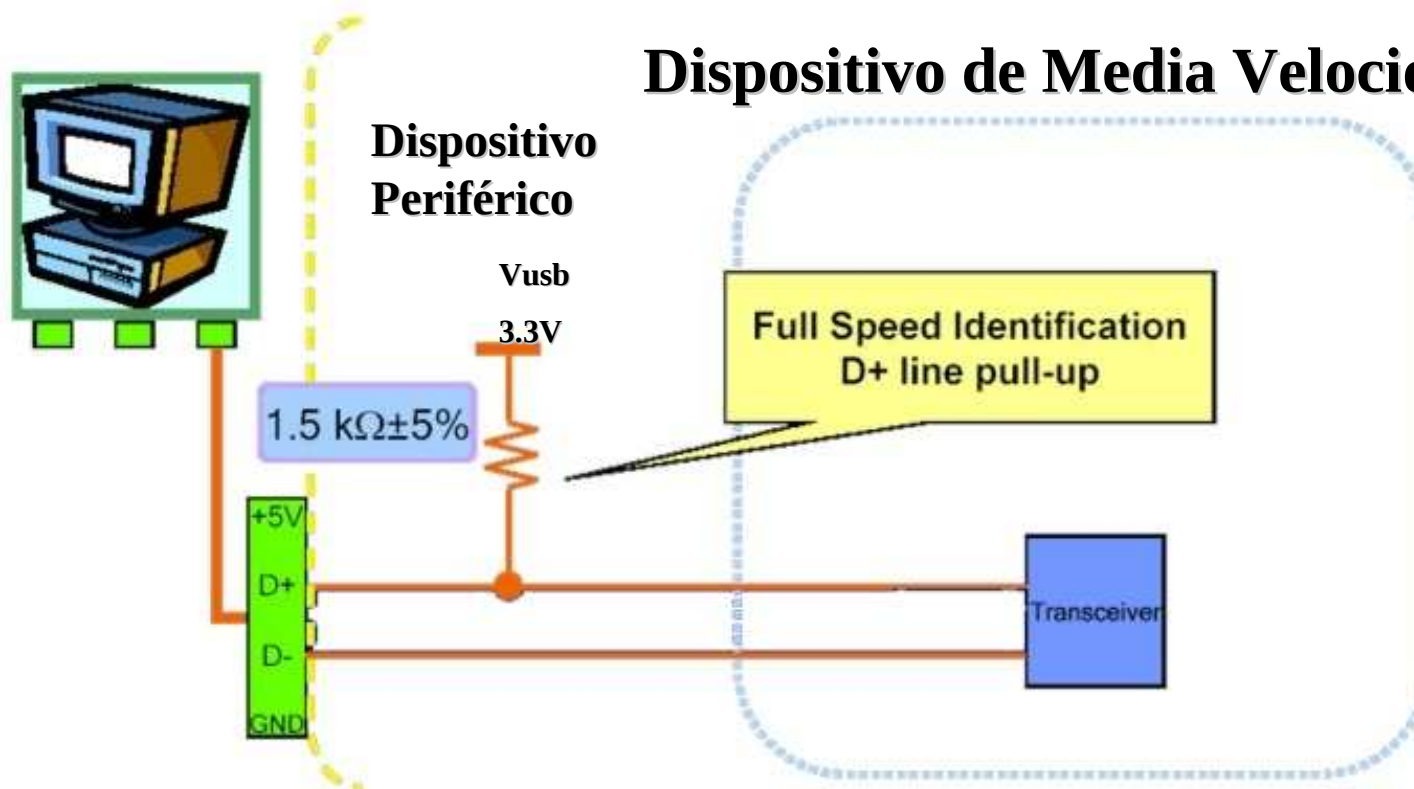
Dispositivo de Baja Velocidad



La Magia de Plug & Play

Auto-detección y Auto-configuración

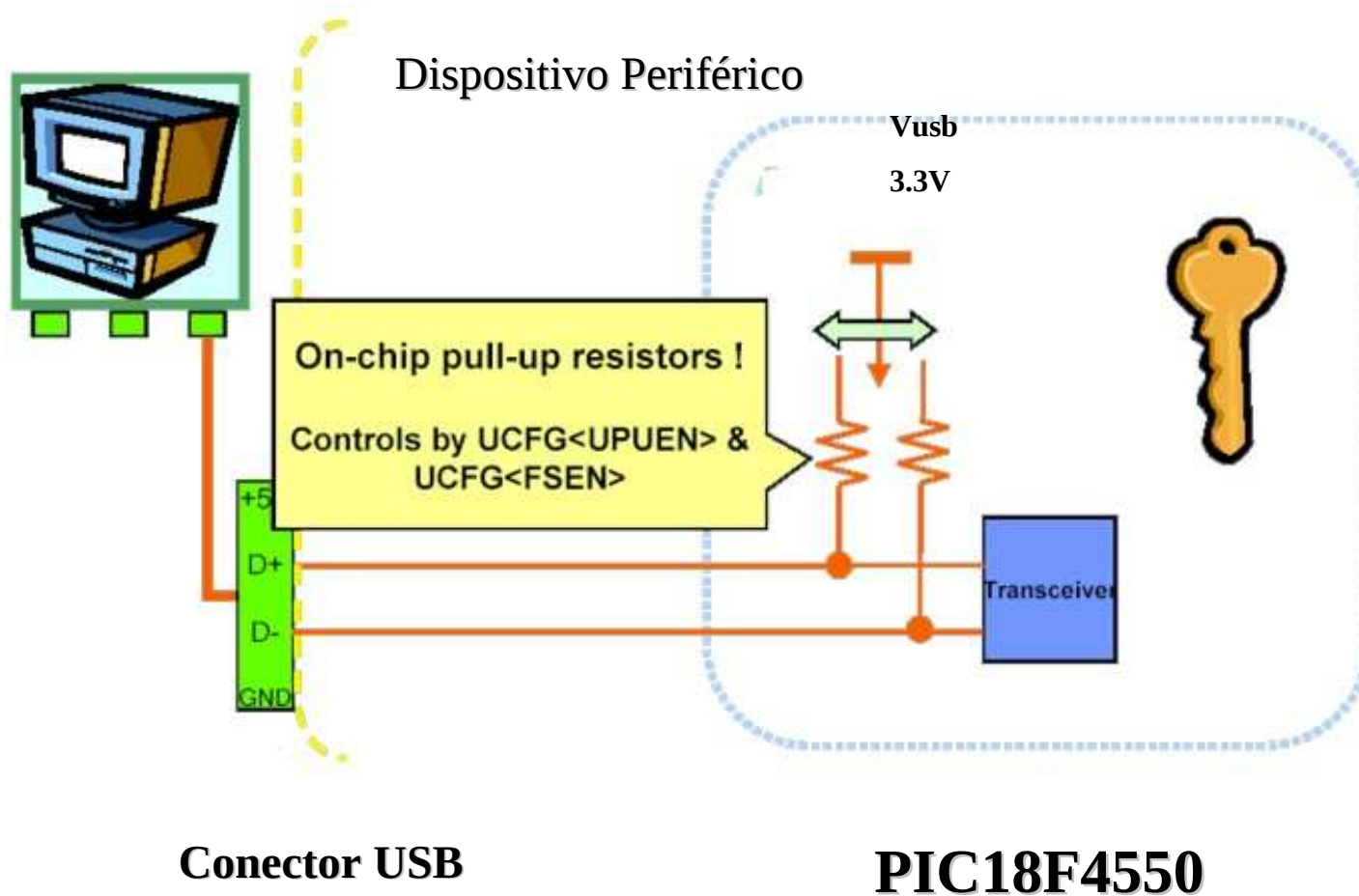
Dispositivo de Media Velocidad



Conector USB



La Magia de Plug & Play





Agenda

- **Breve Historia del Protocolo USB (características)**
- **Conceptos Básicos**
 - **Requisitos (Conocimientos Necesarios)**
 - **Principales Características del Bus**
 - **Terminología**
 - **Arquitectura**
 - **Clase de Dispositivos**
 - **Tipos de Tránsito de Datos**
- **Introducción a la Tarjeta de Evaluación FS USB**
- **USB Microchip Firmware**
- **Vbasic.Net Firmware**
- **Conversión de USB a RS232**
- **Dispositivos de Almacenamiento Masivo**



Terminología

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.

Tuberías (Pipes): Es un enlace virtual entre el host (la PC) y el dispositivo USB.



¿Qué es un Punto Terminal?

Es una localidad específica 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 los dispositivos deben soportar el Punto terminal 0.

Cada Punto Terminal debe tener un número y una Dirección (IN/OUT).

Dispositivo de baja Velocidad solo pueden tener 2 Puntos Terminales. EP0, EP1, EP2



Proceso de 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.

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.



Descriptores

Son Estructuras de Datos o bloques de Información que le permiten al Host (PC) aprender del dispositivo.

El propósito de un descriptor es comunicar la identidad de un periférico en particular con el host (PC)

Los descriptores mas comunes son:

- **Descriptor para Dispositivos**
- **Descriptor de Configuración**
- **Descriptor de Interfaz**
- **Descriptor de Punto terminal**
- **Descriptor de Cadena**



Descriptores

```
File usbdsc.c
/* Device Descriptor */
rom USB_DEV_DSC device_dsc=
{
    sizeof(USB_DEV_DSC), // Size of this descriptor in bytes
    DSC_DEV,             // DEVICE descriptor type
    0x0200,              // USB Spec Release Number in BCD format
    0x00,                // Class Code
    0x00,                // Subclass code
    0x00,                // Protocol code
    EP0_BUFF_SIZE,      // Max packet size for EP0, see usbcfg.h
    0x04D8,              // Vendor ID
    0x000C,              // Product ID: PICDEM FS USB (DEMO Mode)
    0x0000,              // Device release number in BCD format
    0x01,                // Manufacturer string index
    0x02,                // Product string index
    0x00,                // Device serial number string index
    0x01                 // Number of possible configurations
};
```



Arquitectura

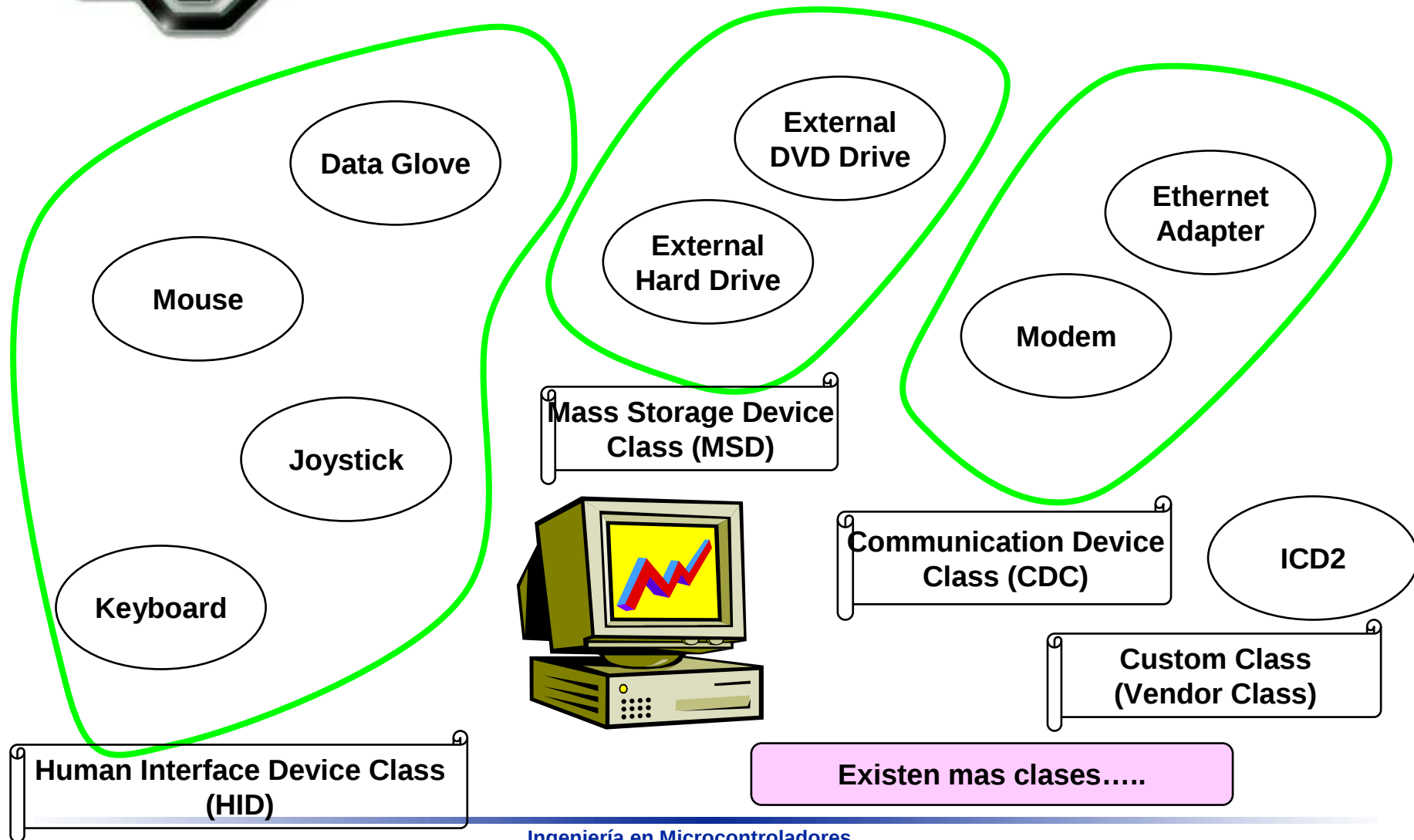
- Es un Bus punto a punto: dado que el lugar de partida es el *host* (PC o *hub*), el destino es un periférico u otro *hub*. No hay más que un único *host* (PC) en una arquitectura USB.

***Nota:USB On The Go**

- Los Ordenadores estándar tienen dos tomas USB, lo que implica que para permitir más de dos periféricos simultáneamente, es necesario un *hub*. Algunos periféricos incluyen un *hub* integrado, por ejemplo, un teclado USB, al que se le puede conectar un Mouse USB.
- Se Pueden Conectar hasta 127 periféricos

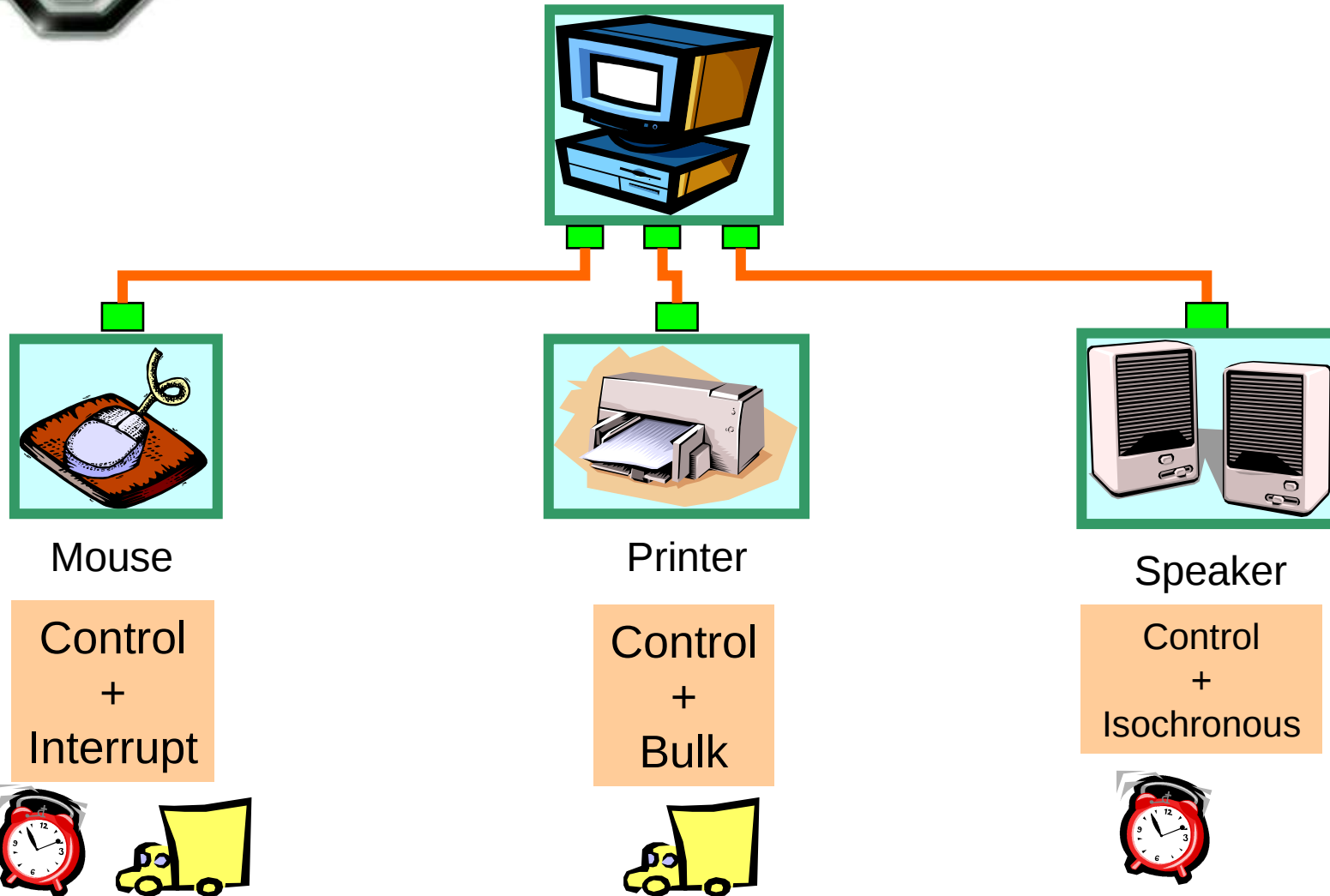


Clases de Dispositivos





Tipos de Transferencia





Tipos de Transferencia

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.**

Bulk: Este modo se utiliza para la transmisión de **importantes cantidades de información**. Este tipo de transferencia es útil cuando la razón de transferencia no es crítica como por ejemplo, el envío de un archivo a imprimir o la recepción de datos desde un escáner



Tipos 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.**

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.**



Tipos de Transferencia

Tamaño del Punto Terminal= Máximo Tamaño del paquete de Datos

	Tipo de Transferencia	Max Tamaño en Bytes
Low Speed	Control	8
	Interrupt	≤ 8
Full Speed	Control	8, 16, 32, 64
	Interrupt	≤ 64
	Bulk	8, 16, 32, 64
	Isochronous	≤ 1023



Microchip PIC18F4550

Function-Specific Features

- Full-speed USB 2.0 compliant interface
 - 12 Mbit/s data transfer
 - 1K bytes USB buffer
 - 32 endpoints (16 in/16 out)
 - Control, interrupt, bulk and isochronous transfer modes
 - Parallel streaming port (DMA)
 - Internal full-speed transceiver and voltage regulator
- nanoWatt Technology
- 48 MHz operation

Basic Feature Set

Memory

32 KB Flash
2 KB data RAM
256B data EE

Analog

13 x 10-bit ADC
2 x comparators

Communication

EUSART, SPI/I2C

Timers

1 x 8-bit, 3 x 16-bit
1 x CCP, 1 x ECCP

Operating Voltage
2.0 – 5.5



Agenda

- **Breve Historia del Protocolo USB (características)**
- **Conceptos Básicos**
 - **Requisitos (Conocimientos Necesarios)**
 - **Principales Características del Bus**
 - **Terminología**
 - **Arquitectura**
 - **Clase de Dispositivos**
 - **Tipos de Tránsito de Datos**
- **Introducción a la Tarjeta de Evaluación FS USB**
- **USB Microchip Firmware**
- **Vbasic.Net Firmware**
- **Conversión de USB a RS232**
- **Dispositivos de Almacenamiento Masivo**



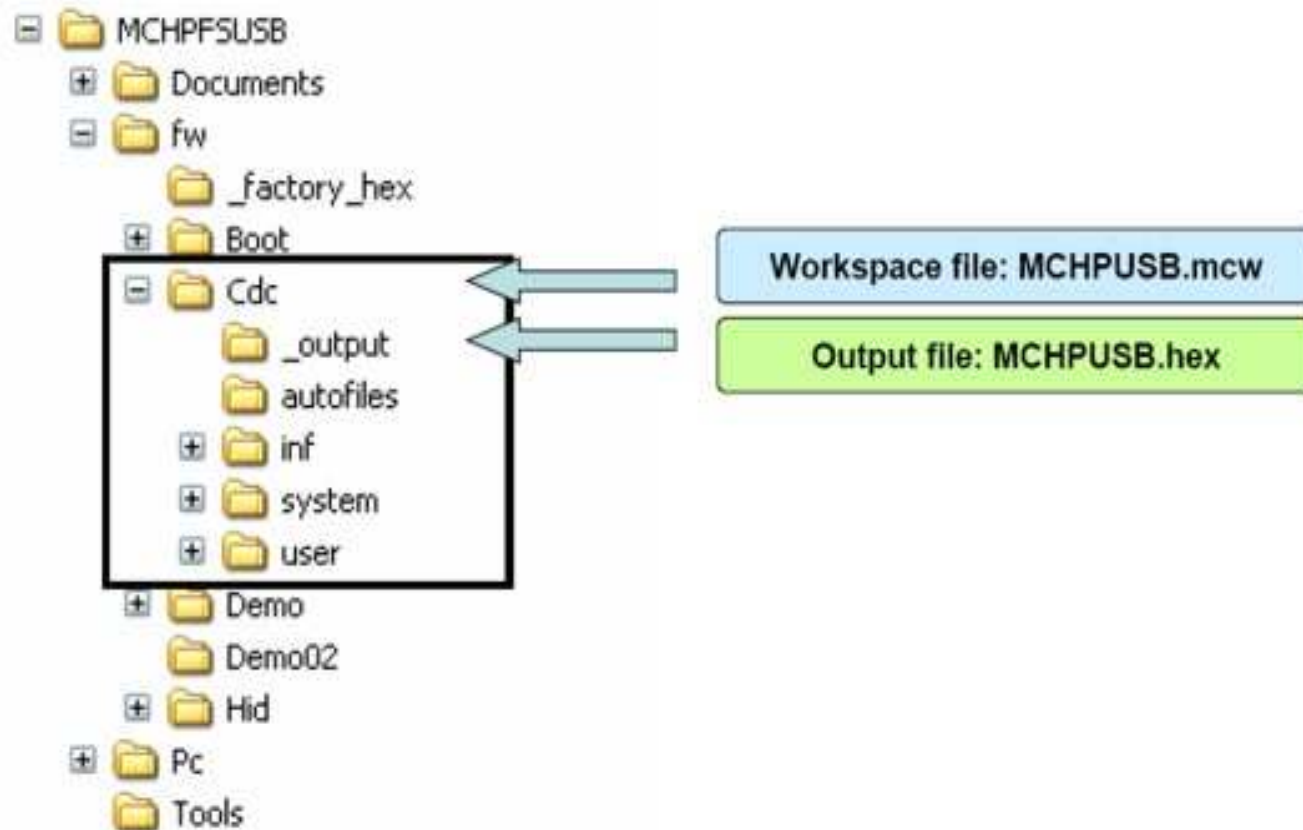
Introducción a la Tarjeta FS USB





Microchip Firmware PIC18F4550

C:\MCHPFSUSB\





Microchip Firmware

PIC18F4550

Subdirectories:

- `_output`: Central location for output files
- `autofiles`: Contains USB global configuration file and descriptors
- `system`: Contains USB firmware
- `user`: Placeholder for the user's firmware

Files:

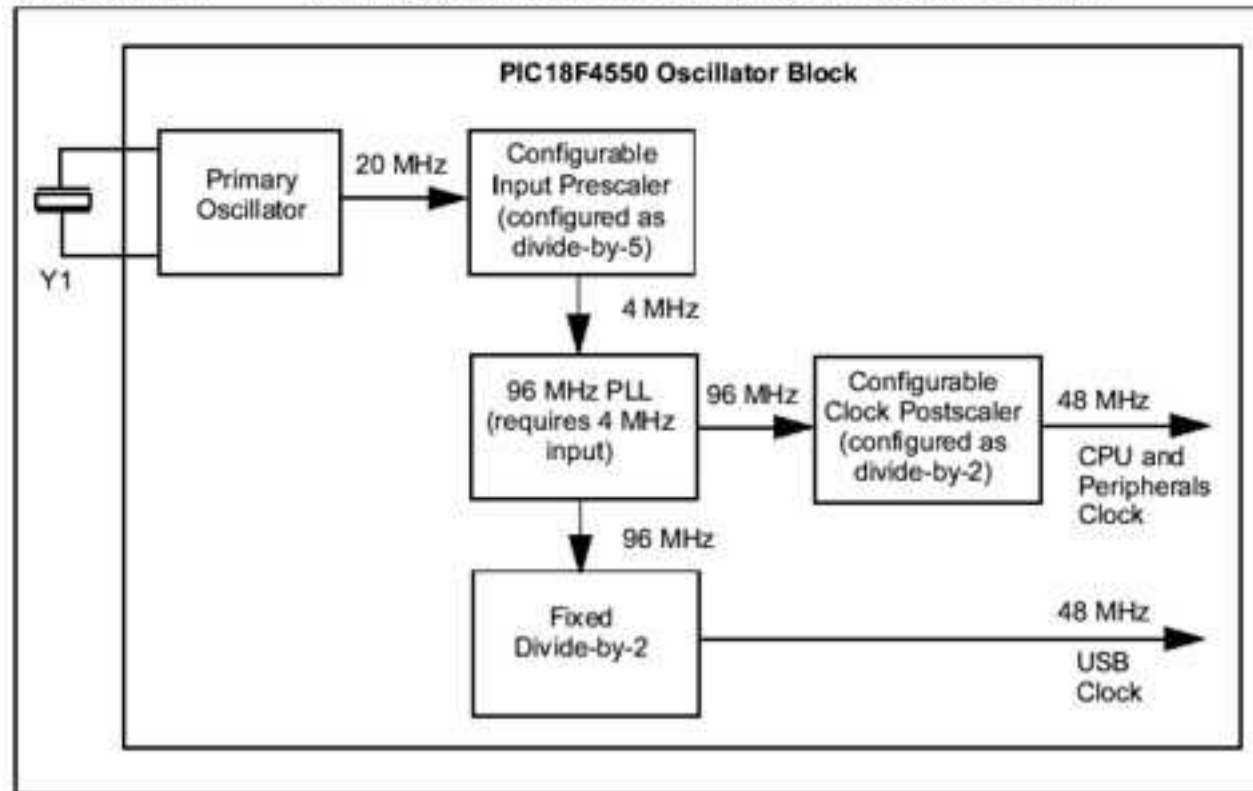
- `CleanUp.bat`: Delete all output files in this folder and any subdirectories.
- `io_cfg.h`: Maps pin names to pin functions; this file should be modified to match each target board. The default example is designed for the PICDEM FS USB demonstration board.
- `main.c`: The file which contains `main()` for the application
- `MCHPUSB.mcp`: MPLAB IDE Project file
- `MCHPUSB.mcw`: MPLAB IDE Workspace file

Ejemplo



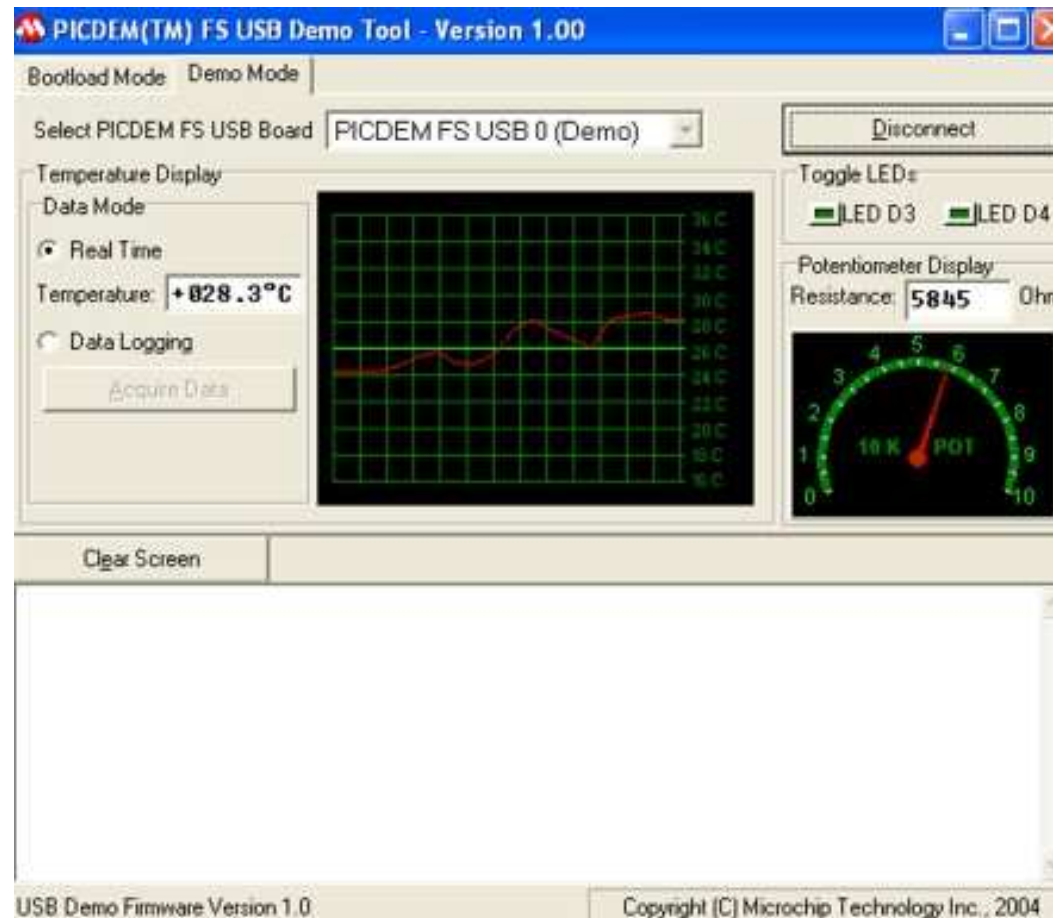
Frecuencia de Reloj PIC18F4550

FIGURE 1-2: PIC18F4550 CLOCK SETUP FOR PICDEM FS USB





Software





Programación en Vbasic.Net

Control Toolbox

Used to create new controls on the form

Main Form

GUI Application Base

Solution Explorer

Provides a list of all files in the project

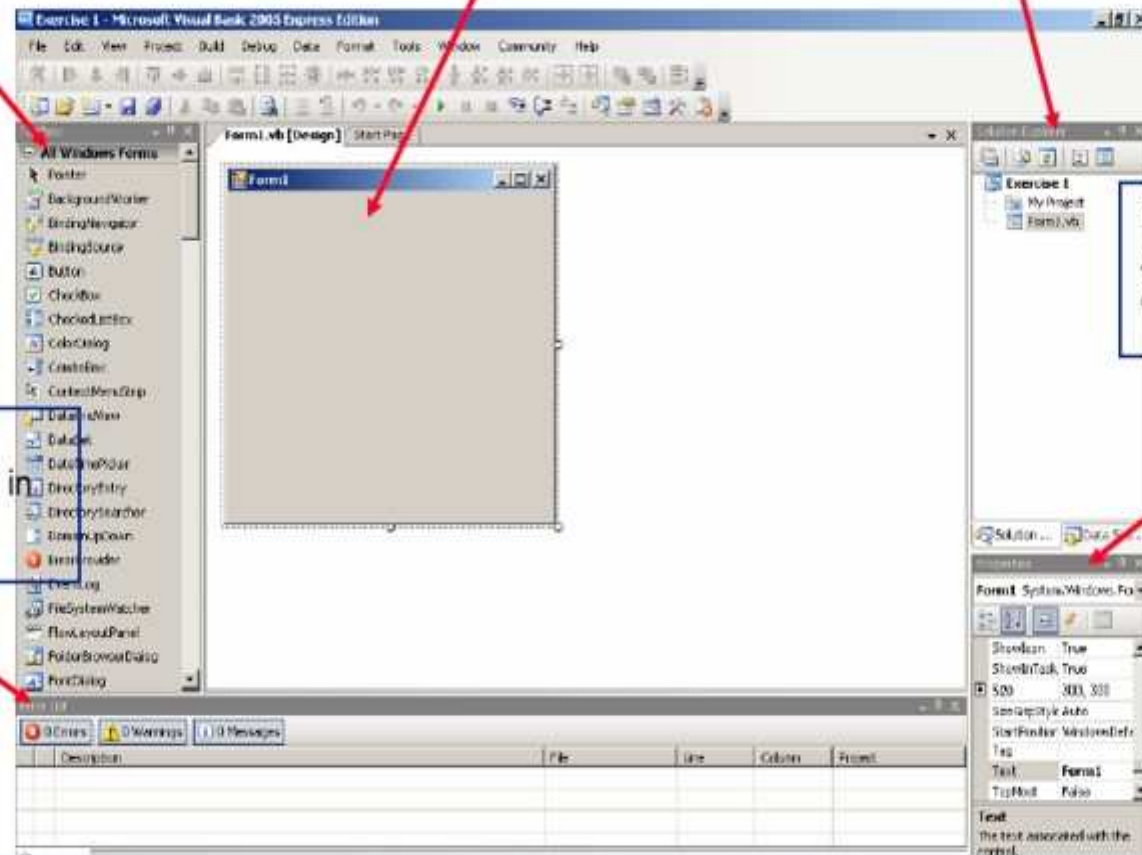
Error List Window

Shows a list of errors in the current project.

Properties Window

Access to the properties of the selected object

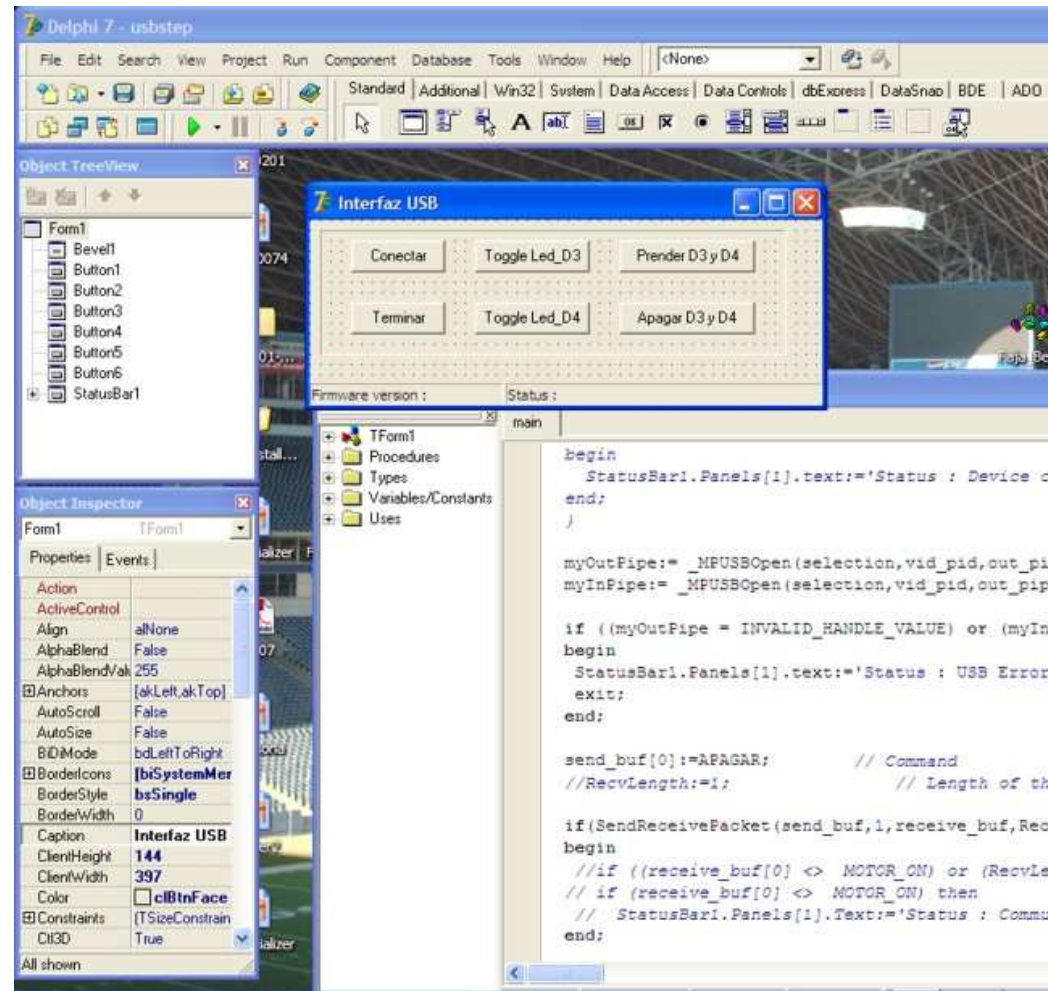
Ejemplo





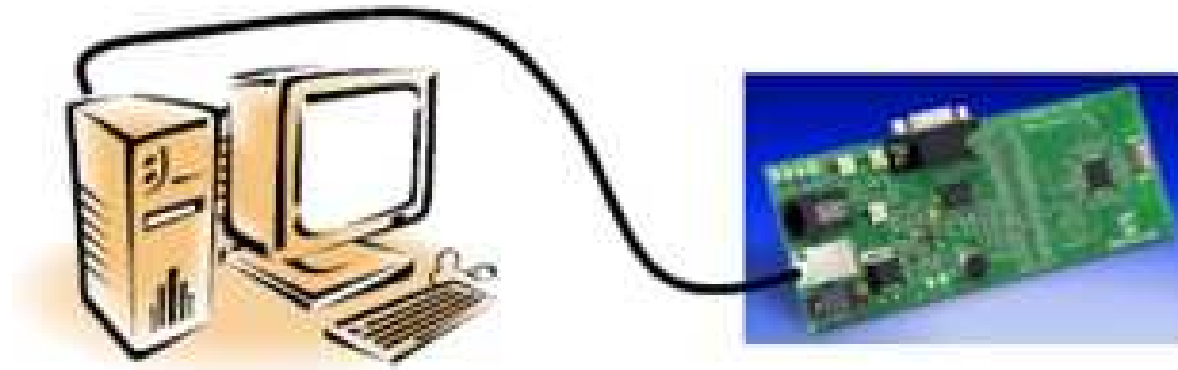
Programación en Delphi

Ejemplo





Conversión RS232 a USB





Típica Aplicación con el Puerto Serial



Bill of Materials:

- 8-bit microcontroller
- RS-232 transceiver

RS-232

±12V
Transceiver
(MAX232)

8-bit MCU

Serial

UART



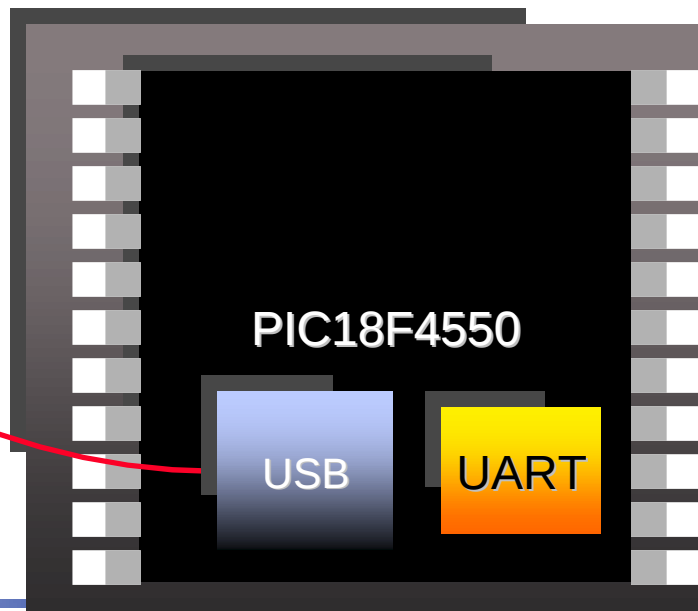
Emulación de un Puerto Serial

Bill of Materials:

- PIC18F4550



FS-USB

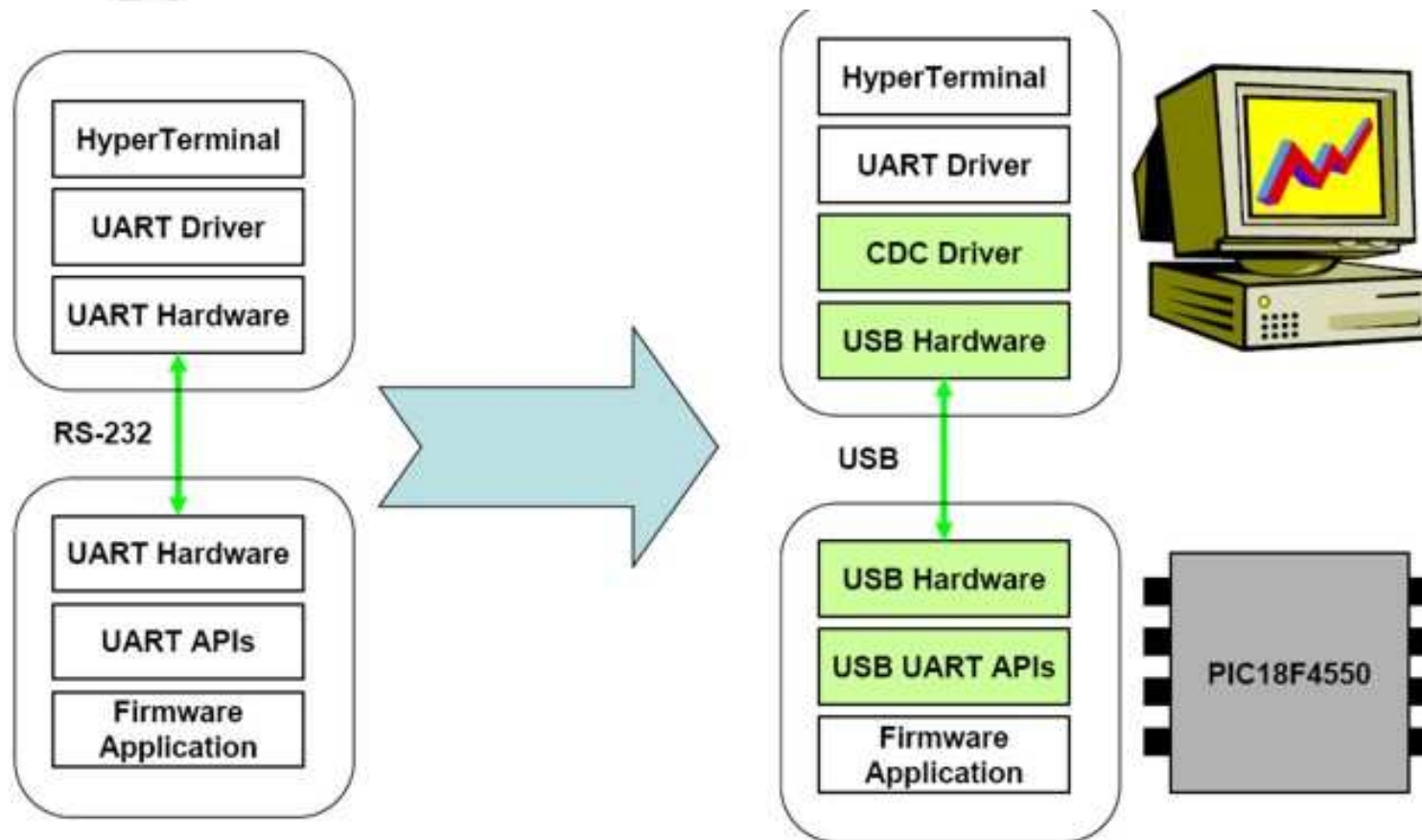


Ingeniería en Microcontroladores

www.i-micro.com



Emulación Un Puerto Serial





Emulación Un Puerto Serial

You should see an additional 'virtual' COM port. Remember the number of the COM port



Ejemplo



EI PICKit Serial Analyzer





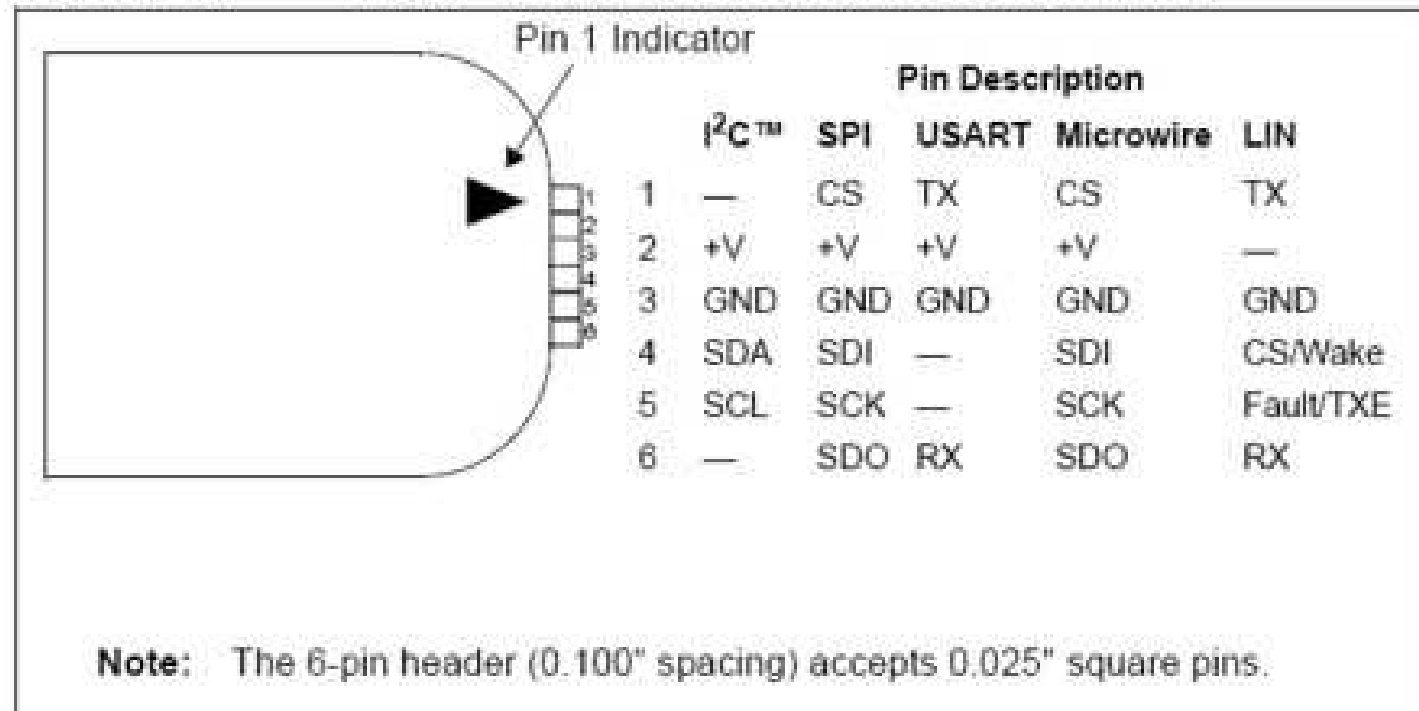
El PICKit Serial Analyzer





EI PICKit Serial Analyzer

FIGURE 1-3: PICKit™ SERIAL ANALYZER PIN ASSIGNMENTS



Ejemplo Displays Seriales



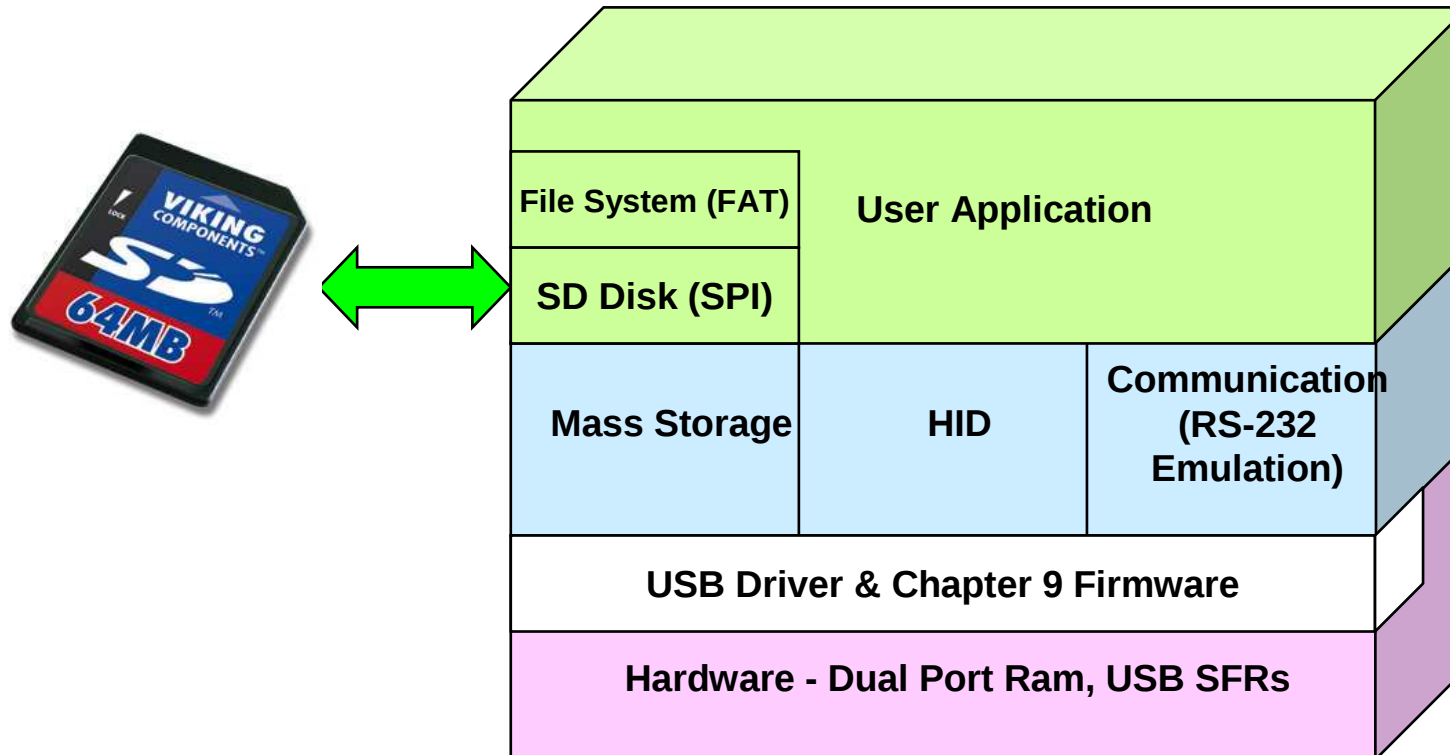
Emulación de un Dispositivo de Almacenamiento Masivo

Part Number – AC164122





Microchip USB Firmware Stack



Ejemplo

PIC18F87J50 FS USB Plug-In Module (PIM)

- PIC18F87J50, 12 MHz xtal
- USB Port, mini-B
- Linear Regulator
- 2 LEDs, 2 Pushbuttons
- Stand alone operation
- Operation with: HPC Explorer Demo Board
- ICSP™ Technology: 6-pin header

Part Number – MA180021



HPC Explorer – DM183022



Part Number – AC164110



Bibliografía

- <http://www.usb.org>
- www.microchip.com/usb
- www.beyondlogic.net (usb-in-a-nutshell.pdf)
- www.lvr.com

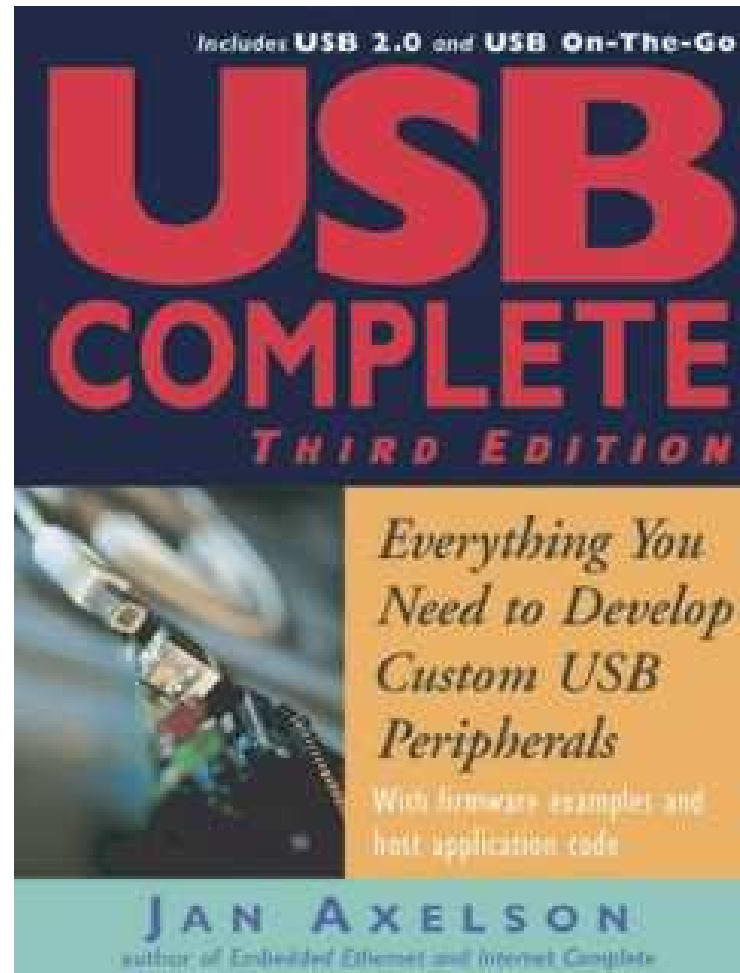
USB Design by Example

A practical guide to building I/O devices

John Hyde Wiley



La Biblia del USB

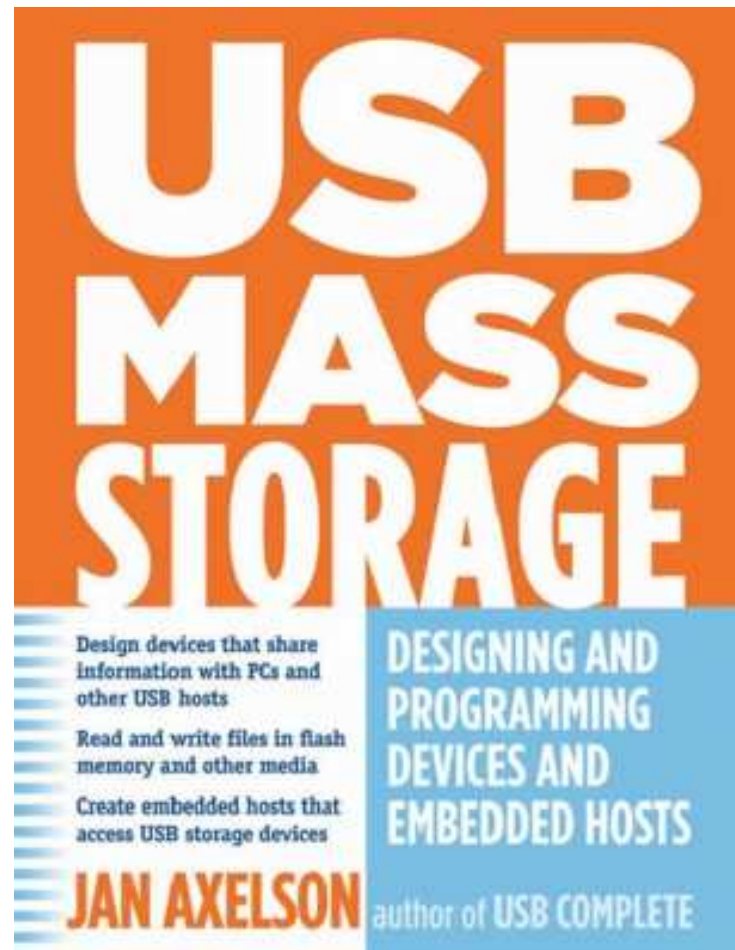


Ingeniería en Microcontroladores

www.i-micro.com



Dispositivos de Almacenamiento Masivo



Ingeniería en Microcontroladores

www.i-micro.com



Notas De Aplicación

DS51526A – FS USB DEMO BOARD

**DS91095A – MODIFYING THE PICDEM USB BOARD
FOR PIC_18**

DS41174A – PICDEM USB USER GUIDE

**AN956 - Migrating Applications to USB from RS-232
UART with Minimal Impact on PC Software.**

AN1003 - USB Mass Storage Device Using a PIC® MCU



¿Dudas?

