Future Technology Devices International Ltd.

Technical Note TN_110

What is USB?

Document Reference No.: DocNo. FT_000156

Version 1.0

Issue Date: 2009-08-07

The purpose of this technical note is to provide a brief introduction into the Universal Serial Bus (USB) Interface.

Future Technology Devices International Limited (FTDI)

Unit 1, 2 Seaward Place, Centurion Business Park, Glasgow G41 1HH, United Kingdom
Tel.: +44 (0) 141 429 2777 Fax: + 44 (0) 141 429 2758
E-Mail (Support): Support1.support@ftdichip.com Web: http://www.ftdichip.com

Table of Contents

1 Introduction

The Universal Serial Bus (USB) interface is rapidly becoming the DeFacto interface for PCs and replacing many other forms of interface such as RS232 or parallel ports that were common a few years ago.

The purpose of this technical note is to provide a brief introduction to the interface to explain its operation.

2 Hardware Interface

2.1 Host or Peripheral

USB designs have 2 distinct interface types. A host and a device (peripheral) interface.

The host must detect when a device is plugged in and then interrogate the device to determine what it is and load the relevant driver. This process is known as enumeration and will be discussed later.

A host will control the device that has been connected to it and it may also provide power to it.

2.2 Power

A device may be powered from its own power supply and is referred to as a self powered device.

The alternative is to power the device from the USB host and this is known as a bus powered device.

The power output pin from a USB host provides power at 5V levels.

A low power, bus powered device may consume up to 100mA from this power pin.

A high power, bus powered device may consume up to 500mA from this power pin, however there are limitations to the availability of this power.

Up to 100mA will always be available on this pin.

Greater than 100mA is only made available to a device from a host after enumeration is successfully complete and the device is considered as "awake".

If the device was not enumerated or had gone to a suspend state then the extra current is not available.

If a device is using its own power supply then the device can draw as much power as the power supply is capable of supplying. This is termed a self powered design.

Also note a bus powered hub cannot provide more than 100mA on its ports.

2.3 Signals

The USB data bus is a 2-wire differential interface with a USB D+ and USB D- signal.

Each signal line is bi-directional and operates between 0V and 3V3.

2.4 Connectors

The USB standard defines a set of USB connectors to ensure a minimal set of cables are required for correct interconnect. This standardisation has allowed for simpler interoperability and cheaper manufacturing costs.

2.4.1 Type A

Type A connectors are the most common USB connector . An example of a Type A receptacle can be seen on most host PCs.

The Type A plug is generally used on a product with a tethered cable.

The Type A connector shape is rectangular with the pins all in a row. The PWR and GND pins should be prominent to allow for mating before the signal pins.

The pinout is as shown below:

Pin 1	VCC (5V)
Pin 2	USB D-
Pin 3	USB D+
Pin 4	GND

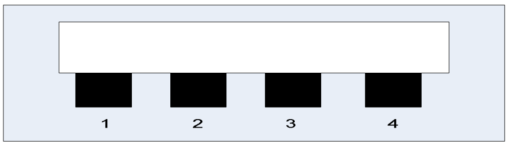

Figure 1: Type A Receptacle

2.4.2 Type B

The Type B connectors are most common on peripheral products that are too large to connect to the PC host directly and require an extra cable known as a Type A-B cable to connect to the host.

The receptacle will be on the peripheral side and the plug is on the cable.

The Type B connector shape is square with a pin in each corner.

The pinout is as per the type A connector.

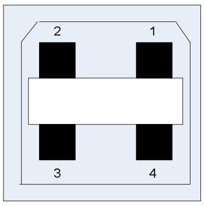

Figure 2: Type B Receptacle

2.4.3 Mini Connector

The mini connector is most commonly used in On-The-Go (OTG) or in space critical designs due to its smaller size.

The receptacle will be on the peripheral side and the plug is on the cable.

The shape is similar to the Type A but smaller.

The pinout is as shown below:

Pin 1	VCC (5V)
Pin 2	USB D-
Pin 3	USB D+
Pin 4	ID*
Pin 5	GND

*Pin 4 may be used in OTG designs to identify if the port should be operating as a host or a device. It is frequently left unconnected when used in designs that are not OTG.

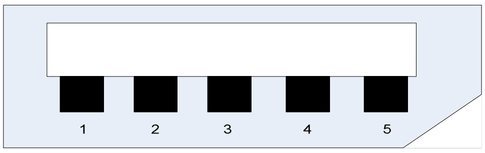

Figure 3: Mini Receptacle

2.5 USB Speed

The USB interface operates at 3 defined speeds, with a 4^{th} generation being ratified at the time of writing.

The USB 1.0 version of the spec defined the interface as a 1.5Mbit/s interface. This is know an as a "low speed".

A low speed device will be recognised by the USB host as being low speed by the presence of a 1k5 ohm pullup resistor to 3V3 on the USB D- signal line.

The USB 1.1 version of the spec defined the interface as a 12Mbit/s interface. This is know an as a "full speed".

A full speed device will be recognised by the USB host as being full speed by the presence of a 1k5 ohm pullup resistor to 3V3 on the USB D+ signal line.

The USB 2.0 version of the spec defined the interface as a 480Mbit/s interface. This is know an as a "high speed".

A high speed device will present itself as full speed on initial connection and during enumeration the host will check if the device is capable of high speed.

Thus a high speed device must support high speed and full speed operation, but a USB 2.0 compliant full speed device does not need to support high speed.

A USB 2.0 compliant host will support all 3 speeds.

The USB 3.0 superspeed specification is being ratified for speeds upto 4.8Gbit/s. At the time of writing there are limited commercial examples available.

USB Specification	Speed name	Speed
USB 1.0	Low Speed	1.5Mbits/s
USB 1.1	Full Speed	12 Mbit/s
USB 2.0	High Speed	480 Mbit/s
USB 3.0	Super Speed	4.8Gbit/s

2.6 Suspend Mode

The USB host controls the power state of the device.

If the USB bus is detected as being idle for >3ms then the host will request to put the device to a suspend state to conserve power.

In suspend mode the current drawn by a device is limited to 500uA.

3 Software Interface

3.1 Enumeration

Enumeration is the process of a host detecting that a USB device has been connected and then loading the relevant device drivers. After detecting the presence of a device the host will initiate a transfer with the device to determine what it is. The host does this by asking for device descriptors which define the device class and what drivers need to be loaded.

3.2 Device Classes

Not all USB devices are from the same class. This allows different peripheral types to be identified by a host when they are plugged in. For example a USB printer is a different class from a USB mouse (HID class) and so requires different drivers as it performs a different task.

For a full list of device classes see: http://www.usb.org/developers/defined_class

The main classes have default drivers on a Windows machine which is why you often do not need to load a separate driver.

USB to serial converters do not have a standard class which is why FTDI products are categorised as device class 0xFF (Vendor specific). This allows FTDI to create their own drivers which they provide and support royalty free. Theoretically a filter driver could be created to allow the FTDI devices to appear under any of the other common classes, but this has only been done for serial ports so far.

3.3 VID/PID

The device VendorID (VID) and Product(ID) (PID), along with the address of the host port are used as an address by the host PC to identify a unique device to match the driver to the device.

The VID is assigned to a company by the USB Implementers Forum (www.usb.org), who ensure each company applying for a VID gets a unique value.

With a VID a company has access to PIDs from 0x0000 to 0xFFFF. A company with their own VID can choose what to do with each PID.

FTDI have been assigned the VID 0x0403 from the USB Implementers Forum and have reserved a few PIDs for their own use. The remaining PIDs can be allocated to customers who wish to customise their designs and the driver. However it should be noted that it is not essential that every single device uses a different VID/PID combination.

3.4 Transfer Modes

There are 3 main types of transfer mode for moving data between the host and the peripheral device. These are bulk mode, interrupt mode and isochronous mode, and are explained in the next sections.

3.4.1 Bulk Mode

Bulk mode is based on the fact you intend to send a set amount of data in a packet every time.

Full speed devices typically use 8, 16, 32 or 64 byte packets as the default size.

High speed devices may use up to 512 byte packet as the default size.

In bulk mode there will usually be no transfer of data until the packet size threshold (64/512) has been reached. However there are conditions where you may never reach this amount of data and you need to transfer a “short” packet. The mechanism to achieve this is called a latency timer timeout. This is simply a mechanism to flush the buffer of the device back to the PC after a pre-set time delay.

Bulk transfers are good for large transfers of data that are perhaps happening in bursts as opposed to streamed. The data is checked for errors using CRCs, and delivery is guaranteed. The downside of bulk transfers is having less control over the bandwidth and latency as the data may transfer in bursts.

FTDI devices support bulk transfer mode.

3.4.2 Interrupt Mode

Interrupt mode is a preferred method of transfer if you have infrequent data that you want to send in a controlled period of time.

Interrupt mode also contains error detection with a retry to ensure the correct data is received.

The maximum packet size for full speed devices in this mode is again 64 bytes and high speed is 1024 bytes.

An interrupt request will be queued for the next time a device is polled by the host to ensure the data is transferred as soon as possible.

3.4.3 Isochronous Mode

Isochronous mode is a preferred method of transfer for streaming continuous data such as audio.

This uses the principal that the bandwidth will be evenly distributed to allow a guaranteed amount of data at a set period of time. However there is no retry, and as such, data delivery is not guaranteed.

3.5 Endpoints

Endpoints are the destinations that the USB host links with for data transfer.

Every device has a control endpoint. This is usually endpoint 0 and this is used for obtaining the device descriptors and configuring a device. Ie control data as opposed to application data.

There will be at least one IN endpoint and this is used to transfer application data from the peripheral device to the host.

There will be at least one OUT endpoint and this is used to transfer application data from the host to the peripheral device.

The IN and OUT endpoints may be of type bulk, interrupt or isochronous depending on the transfer mode used by the USB device.

FTDI devices use a control endpoint and one bulk IN and one bulk OUT endpoint for each port in their devices.

E.g. The FT232R is a single port device and has 1 Control + 1 IN + 1 OUT endpoint.

4 How FTDI "Make USB Easy"

This document has been designed to give a very basic overview of some of the main USB terms.

It is not intended to teach designers how to design a USB device or USB drivers.

FTDI devices will connect and enumerate with no additional effort from the product designer and if connected to a mainstream Operating System (e.g. Windows) then FTDI drivers are already available - free.

As well as providing the ICs to connect USB to other interfaces, FTDI have produced a vast range of development modules based on the FTDI chipsets. These modules allow easy development of USB to TTL Serial, RS232, RS422, RS485, JTAG, SPI, I2C and many other interfaces.

5 Contact Information

Head Office – Glasgow, UK

Future Technology Devices International Limited

Unit 1, 2 Seaward Place, Centurion Business Park

Glasgow G41 1HH
United Kingdom

Tel: +44 (0) 141 429 2777
Fax: +44 (0) 141 429 2758

E-mail (Sales) sales1@ftdichip.com
E-mail (Support) support1@ftdichip.com
E-mail (General Enquiries) admin1@ftdichip.com
Web Site URL http://www.ftdichip.com
Web Shop URL http://www.ftdichip.com

Branch Office – Taipei, Taiwan

Future Technology Devices International Limited (Taiwan)
2F, No 516, Sec. 1 NeiHu Road
Taipei 114
Taiwan, R.O.C.
Tel: +886 (0) 2 8797 1330
Fax: +886 (0) 2 8751 9737

E-mail (Sales) tw.sales1@ftdichip.com
E-mail (Support) tw.support1@ftdichip.com
E-mail (General Enquiries) tw.admin1@ftdichip.com
Web Site URL http://www.ftdichip.com

Branch Office – Hillsboro, Oregon, USA

Future Technology Devices International Limited (USA)
7235 NW Evergreen Parkway, Suite 600
Hillsboro, OR 97123-5803
USA
Tel: +1 (503) 547 0988
Fax: +1 (503) 547 0987

E-Mail (Sales) us.sales@ftdichip.com
E-Mail (Support) us.admin@ftdichip.com
Web Site URL http://www.ftdichip.com

Branch Office – Shanghai, China

Future Technology Devices International Limited (China)
Room 408, 317 Xianxia Road,
ChangNing District,
ShangHai, China

Tel: +86 (21) 62351596
Fax: +86(21) 62351595

E-Mail (Sales): cn.sales@ftdichip.com
E-Mail (Support): cn.support@ftdichip.com
E-Mail (General Enquiries): cn.admin1@ftdichip.com
Web Site URL: http://www.ftdichip.com

Distributor and Sales Representatives

Please visit the Sales Network page of the FTDI Web site for the contact details of our distributor(s) and sales representative(s) in your country.

Appendix A - References

FTDI website: www.ftdichip.com

USB Implementers forum: www.usb.org

Appendix B - Revision History

Revision History

Draft	Initial Draft	June, 2009
1.0	First Release	7th Aug 2009