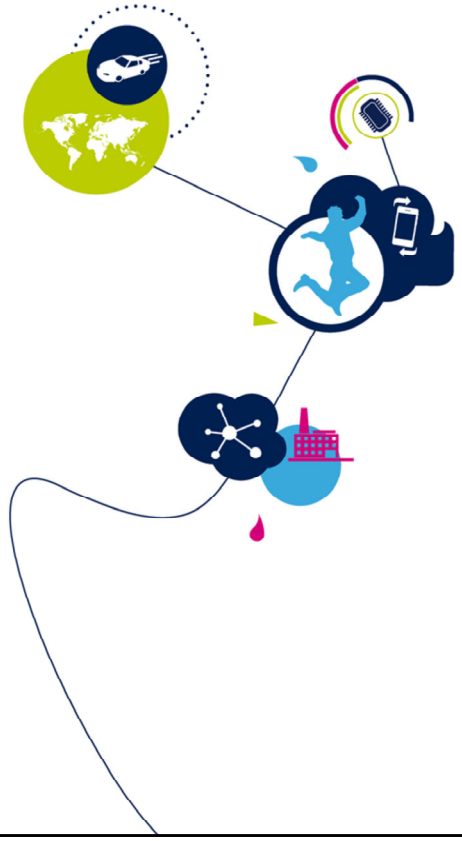
STM32WB – USB device

USB 2.0 Full Speed device interface

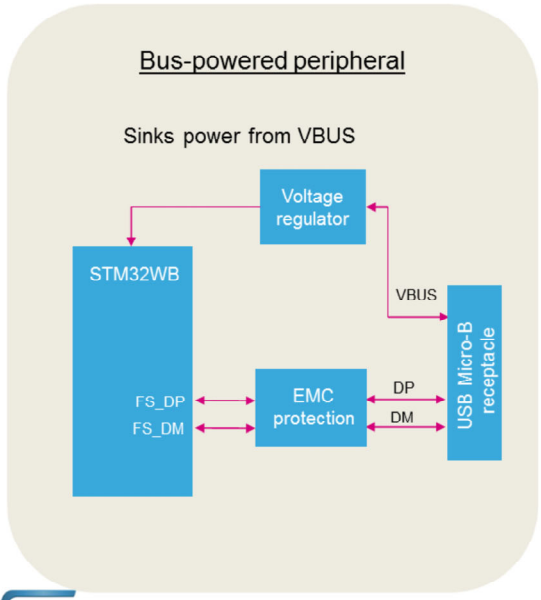
Revision 1.0



life.augmented

Hello, and welcome to this presentation of the STM32WB USB 2.0 Full Speed interface. It covers the features of this interface, which is widely used to interface with a PC.

Overview



- Provides USB 2.0 Full Speed interface
- Total of sixteen endpoints (can provide eight bi-directional endpoints)
- Battery Charger Detection v1.2 supported

Application benefits

- Crystal-less operation
- Low-power implementation
- Battery Charger Detection hardware recognizes chargers, allowing more efficient charging

life.augmented

This figure shows the connections between an STM32WB microcontroller and a USB connector.
The STM32WB features a USB 2.0 Full Speed communication interface, allowing the microcontroller to communicate typically with a PC. The simplest implementation is a USB peripheral device.
It provides a 16--endpoint capability which can be configured for example as 8 bi-directional endpoints.
It also supports the Battery Charging Detection specification v1.2.
Application benefits include crystal-less operation, a low-power implementation, and faster charging thanks to the charger detection function.

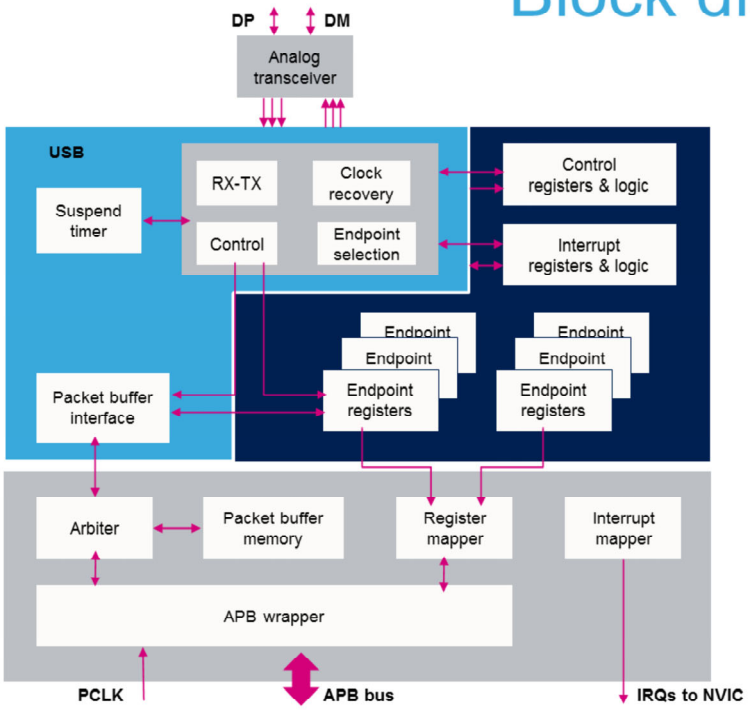
Key features

- USB 2.0 Full Speed (12 Mbit/s)
  - Implements a USB Full Speed device
  - Crystal-less operation is supported due to Clock Recovery System
  - Up to 16 endpoints (e.g. 8 bi-directional) supported
    - Isochronous endpoints supported
    - Bulk endpoint performance optimization using a double-buffering mode
  - Supports Link Power Management (LPM)
  - Includes battery charger detection hardware
    - Latest specification, BC1.2 supported (implementation requires software)
    - By using this specification assisted by hardware, more current (up to 1.5 A) can be safely drawn from BC1.2-compliant chargers, hence decreasing battery recharge time

life.augmented

Let' s look at some of the key features of this USB Full Speed interface, which is a USB 2.0 compliant interface that operates at a 12 Megabits per second bit rate.
A USB FS device can be implemented.
Crystal-less operation is supported.
A total of 16 endpoints (8 bi-directional) can be supported.
Full support for isochronous endpoints.
Bulk endpoints are able to use a double-buffering mode offering higher performance.
In-built support for Link Power Management adds enhanced power modes on top of the USB 2.0 specification.
Battery charger detection allows for increased current to be drawn from BC1.2 compliant chargers allowing up to 1.5 amp charging.

Block diagram (USB)



life.augmented

The block diagram of the USB Full Speed device controller shows the various building blocks inside along with its analog transceiver, which manages the physical layer, shown above.

The physical layer (or PHY handles the analog signal levels including specific level detections as well as Battery Charger Detection functions.

The USB interrupt goes to the Cortex processor to signal various USB events.

The AHB peripheral bus (APB) enables read/write access of the controller.

Other key elements are the packet buffer memory and the suspend timer allowing low-power operation.

Related peripherals

- CRS (clock recovery system)
  - The remote host's start of frame (SOF) timing is extracted from the device controller and the CRS can use it to drive the internal clock generator (HSI RC 48 MHz) allowing USB timing requirements to be met without the need for an external crystal
- NVIC (interrupts)
  - A single interrupt line is connected to the NVIC to generate events
- EXTI (events)
  - A single event line is connected to the EXTI to enable the system to be woken from Stop mode when the USB activity resumes (leaving Suspend mode)

life.augmented

Several related peripherals work in conjunction with the USB device controller to link the USB activity to the system power mode and the requirements of the software.
The clock recovery system allows operation without an external crystal, using the integrated HSI oscillator as the main clock source.
The interrupt events are sent to the non-vectored interrupt controller via a single line.
The system events can cause the system to wake up from Stop mode, for example at the moment we resume from USB Suspend mode.

Interrupts

Interrupt event	Description
Packet memory area over / underrun	Signals that the microcontroller has failed to respond to a USB memory request in time.
Error	General-purpose error signal that is useful for software development. The signaled errors (several types) are typically handled in the protocol by retransmission. This interrupt serves to indicate that these conditions are in fact happening.
Wakeup	Used in Suspend mode to enable wakeup of the microcontroller.
Suspend mode request	Detects a 3 ms absence of host traffic which is the condition in the USB protocol requiring device to adopt" suspend behavior.
USB reset request	Detects the USB reset signaling from the host.
Start of frame	Informs the system of each new frame from the host.
Expected start of frame	Informs the system when a new frame was expected but not received.
LPM L1 state request	Signals the reception and acknowledgement of LPM L1 state.

The USB device controller generates interrupts in different circumstances which generally require a handling by software.
The first two listed interrupts cover various error and warning conditions.
The remaining interrupts correspond to normal USB protocol events.

Low-power modes

MCU mode	Description	USB availability
Run	MCU fully active.	Required until USB enters Suspend mode.
Sleep	Peripheral interrupts cause the device to exit Sleep mode. Peripheral registers content is kept.	Available while USB is in Suspend mode.
Stop 0,1	Peripheral interrupts cause the device to exit Stop mode. Peripheral registers content is kept.	Available while USB is in Suspend mode, offers optimal power reduction.
Stop 2	Peripheral interrupts cause the device to exit Stop mode. Peripheral registers content is kept.	Not compatible with USB applications.
Standby	Powered-down. The peripheral must be reinitialized after exiting Standby mode.	Not compatible with USB applications.

life.augmented

The USB peripheral is fully active in Run mode. After a Suspend event, Sleep mode and also Stop 0 and 1 modes are available. Stop 2 and Standby modes should not be used.

Debug information

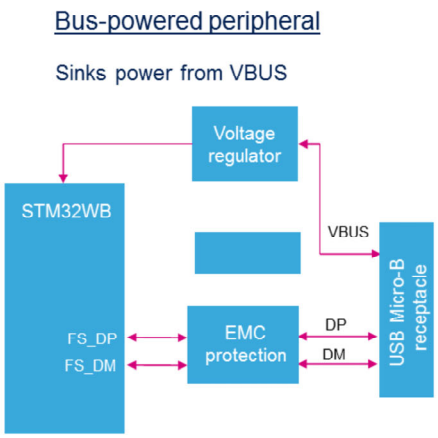
- Dedicated bits in the status and frame number registers may assist debugging

Debug bit	Description
Error (ERR)	Error bit (ERR), also listed in the interrupts slide, enables visibility of a number of events (which are not protocol errors) • No answer • Bit-stuffing error • CRC error • Framing format Violation
RXDP ; RXDM	Receive line state for D+ and D- pins

life.augmented

Some debug help is available using a single status bit which corresponds to an interrupt event.
Within the USB device controller, the dedicated ERR status bit provides some debug functionality in a USB application in relation to various events. The events that can trigger this debug bit are listed in this table.
There is also a way to directly read the states of the D+ and D- lines.

Application: low-power device



- The schematic shows a low power design example for a peripheral device application
- In this example, the device is bus-powered, drawing power only from the USB VBUS pin.
- No crystal is required. The timing is derived from the remote host's frames by making use of the CRS block.

life.augmented

Here is an application example of a low-power peripheral device.
Power is drawn directly from the USB VBUS signal.
No crystal is required.

References

- For more details, please refer to the following pages
  - www.usb.org: usb20_docs (hosts a ZIP file containing):
    - USB2.0 specification
    - USB2.0 ECN: Link Power Management Addendum
  - www.usb.org: devclass_docs
    - Battery charger v1.2 specification
- Additional information can be found in these application notes:
  - Application note AN4879: USB hardware and PCB guidelines using STM32 MCUs
  - Application note AN4775: Basics and low-cost solution proposals to move from a legacy USB 2.0 connector to USB Type-C™ connector with STM32 devices

life.augmented

For complete USB specification documents, please refer to USB.org.
The USB 2.0 document home page has a ZIP file containing the USB 2.0 and OTG 2.0 specifications and an Engineering Change Notice (ECN) for Link Power Management (LPM).
The USB device class documents page has the battery charger specification.
Additional information can be found in these applications notes.