USB On-The-Go System Software

Biao Jia

TransDimension Inc.

Agenda

- USB On-The-Go System Software Overview
  - Host, Peripheral, USB On-The-Go
  - Class Drivers
- Software Challenges
- Typical System Requirements
  - CPU
  - Memory
  - Other software components
- USB On-The-Go Application Examples
  - Cell Phone
  - PDA

USB On-The-Go System Software Overview

What kind of USB Software do I need?

- USB Peripheral
  - My device will connect into a host (B receptacle)
  - Implements specific “device” functionality
- USB Host
  - Peripherals will connect to my host (A receptacle)
- USB On-The-Go (mini-AB receptacle)
  - PDA to mobile phone
  - Mobile phone to mobile phone
  - Digital camera to printer

What kind of USB Software do I need?

- **USB Peripheral**
  - Incapable of initiating USB data transfers (only responds to host requests)
  - Application specific (e.g. mass storage, communication device, simple data transfer)
  - May require special Windows Driver (large software investment) Use built-in Windows support where possible.
  - For most USB Full-Speed applications, the software is largely firmware based, typically for micro-controller based systems
  - For USB High-Speed applications, the throughput requirements introduce more complex systems (operating systems, 32 bit microprocessors)
  - Generally, single purpose software

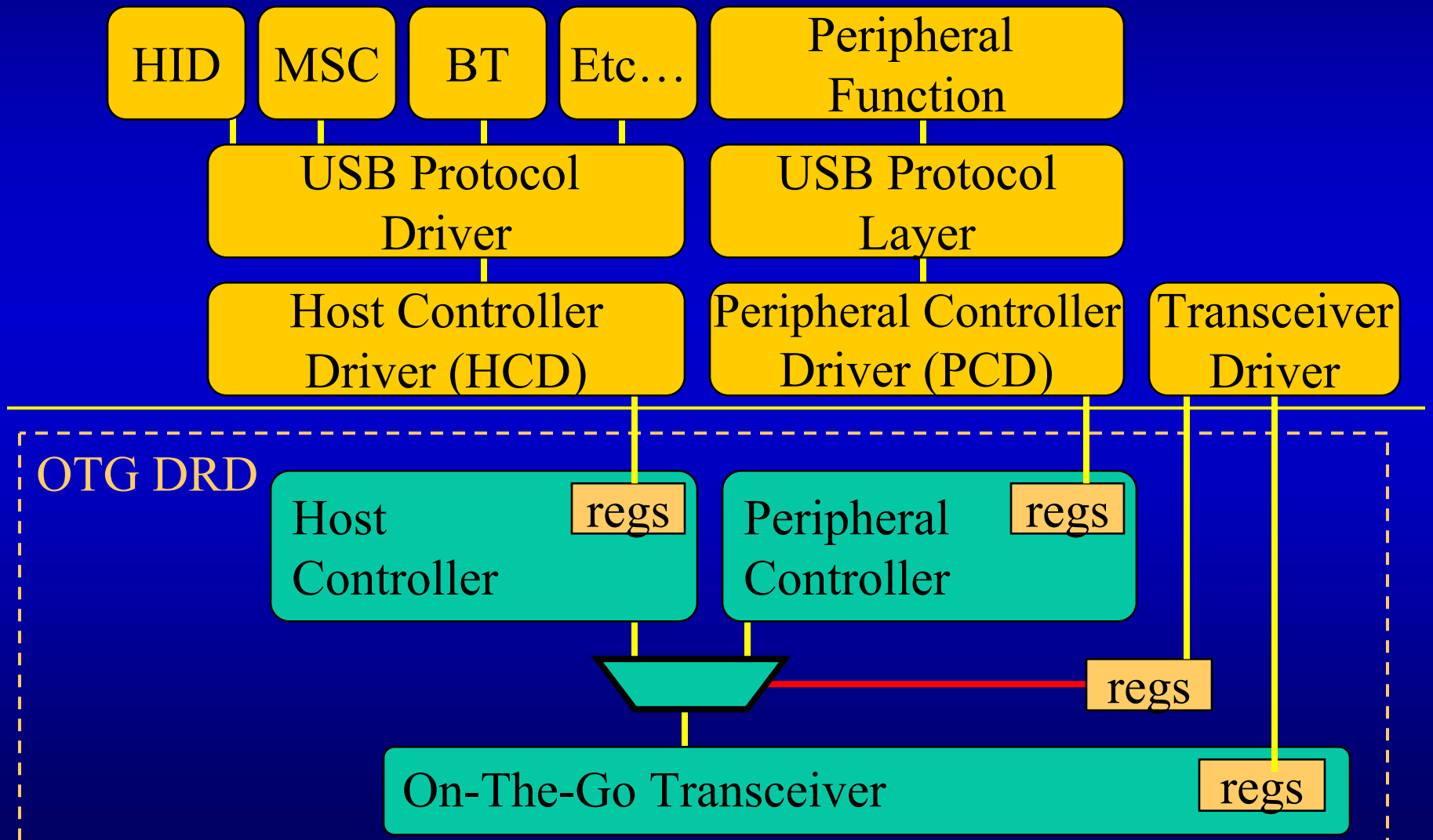
What kind of USB Software do I need?

- USB Host
  - Manages and controls the bus
  - Initiates all data packet activity on the bus
  - Detects peripheral insertion / extraction
  - Enumerates inserted USB devices and matches device to a particular class driver
  - System is typically consists of complex software applications with an operating system and 32 bit microprocessor
  - Class driver needed for each device or class of devices

What kind of USB Software do I need?

- USB On-The-Go
  - Capable of both Host and Peripheral Operation (dual role device or DRD)
  - DRD is a compliant USB peripheral when operating in this mode
  - Transceiver Driver to support new USB On-The-Go protocols
    - Host Negotiation Protocol
    - Session Request Protocol
  - Targeted Peripheral List
    - Manufacturer, model number
  - No Silent Failures
    - API for higher layer software to receive errors for messaging
  - Power Management

Logical USB On-The-Go Architecture



What USB Devices does my design need to support?

USB-IF Standard Classes

Standard USB-IF defined Class Specifications

- HID (Human Interface Device)
  - Mouse, keyboard, joystick, etc.
- Hub (defined in the USB 2.0 Specification)
- Printer
- Digital Image Capture Class (a.k.a. PTP, PIMA)
- Audio Class
  - Speakers, Microphones, MIDI Devices
- Mass storage
  - USB hard drives, Zip drives, flash card readers, some digital cameras
- Communications
  - ACM (analog modems, cell-phones)
  - ECM (cable modems)
  - Others…
- USB Chip/Smart Card Interface Devices
- IRDA Bridge Device Class
- USB Monitor Control Class
- Video Class

Proprietary Protocols

Other Standards Bodies

- USB Bluetooth HCI (defined in Bluetooth Specification Appendix)
- MS RNDIS (cable modems, other “ethernet” type devices)

Manufacture Specific

- USB-to-Serial (legacy adapters, PDAs)
- USB-to-Ethernet (chipset specific, some require firmware)
- MP3 Players (some use Mass Storage)
- Some Digital Still Cameras
- Web Cams
- Video Capture Devices
- Host-to-Host Cables

What protocol should I implement as a peripheral?

- Simple PC Transfer
  - Mass Storage Class (MSC)
  - Media Transfer Protocol (support in Longhorn)
  - Active Sync, Palm Hot Sync, Symbian Connect…
- Application Specific Implementations
  - USB-to-Ethernet (Comm Class, RNDIS, other)
  - USB-to-Serial (ACM-Serial Emulation other)
  - Video Cameras (Video Class, proprietary)
  - Audio devices (Audio Class, proprietary)
  - DSC (DSC Class based on PTP Protocol)

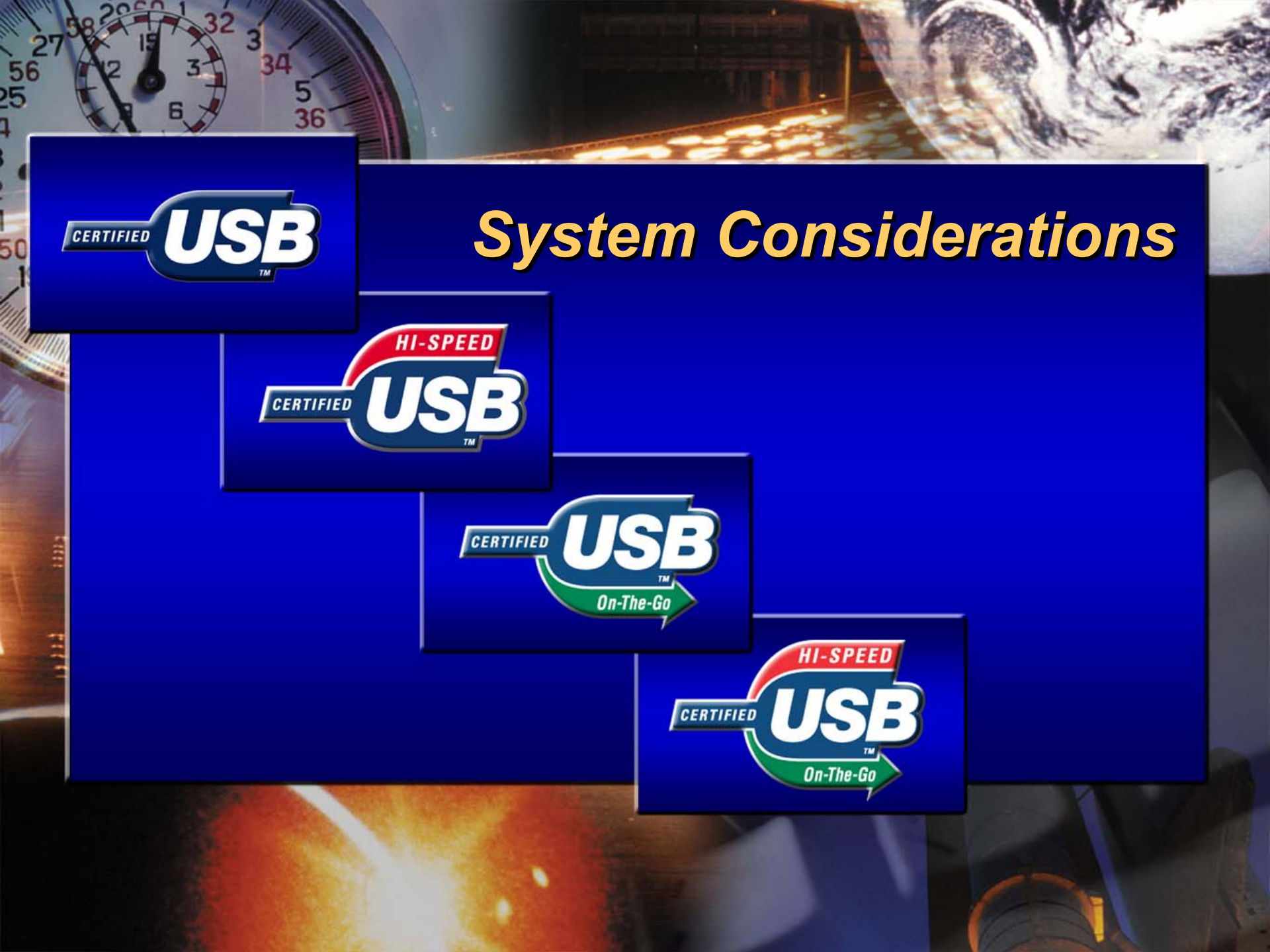

CERTIFIED USB
Software Challenges
HI-SPEED
CERTIFIED USB
CERTIFIED USB
On-The-Go
HI-SPEED
CERTIFIED USB
On-The-Go

Software Challenges

- No native support for USB On-The-Go in any OS today
  - Third party solutions available
- No “Open” Standard for USB On-The-Go Controllers
  - Discrete Chips (proprietary interfaces)
    - TransDimension TD243, TD242
    - Philips 1362
    - Others…
  - Silicon IP
    - TransDimension OTGIP
    - Synopsys DesignWare USB OTG Component
    - Sciworx USB 1.1 Combo
    - Mentor Graphics Inventra USB OTG IP
    - Others…

Software Challenges (continued)

- USB touches on many different aspects of a system:
  - File System for USB hard drives, card readers, or CD-ROMs
  - TCP/IP Networking for ECM or USB-to-Ethernet support
  - Printer drivers for USB printing applications
  - Comm port for ACM (USB modem) connectivity
  - Audio subsystem for playing music (USB speakers) or recording sound (USB microphones)
- So many USB devices, so little time
  - USB-IF defined classes vs. proprietary classes
- Embedded designs have a tendency to change over time
  - Different operating systems
  - Different CPUs
  - Different USB On-The-Go hardware


CERTIFIED USB
System Considerations
HI-SPEED
CERTIFIED USB
CERTIFIED USB
On-The-Go
HI-SPEED
CERTIFIED USB
On-The-Go

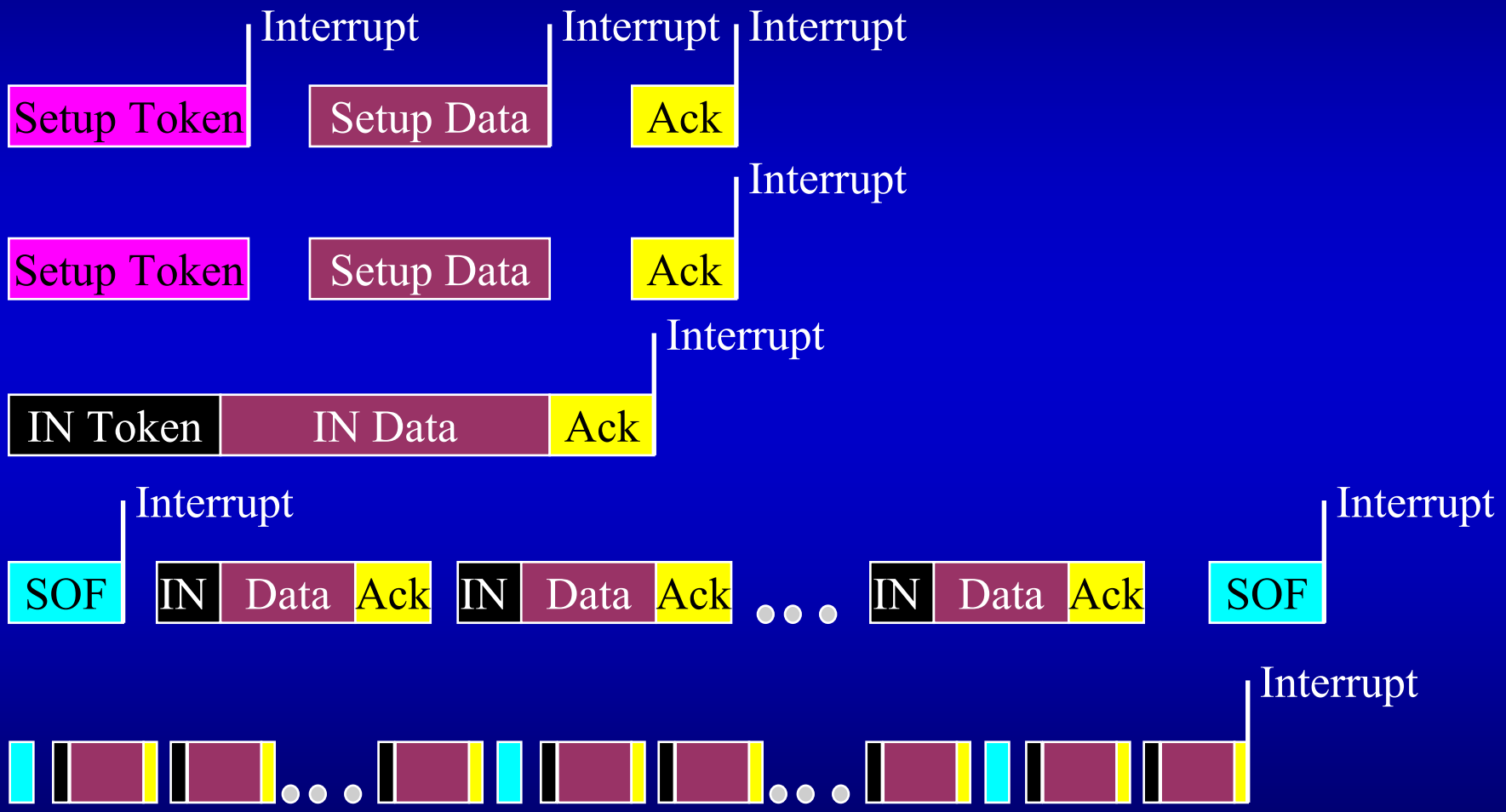
Common System Challenges

- Software Complexity
  - Host applications generally require significantly more software than traditional peripheral firmware.
  - Many Classes of Devices requiring many different device drivers
- Power
  - USB On-The-Go Specification calls for a minimum of 8mA
  - Most devices require a minimum of 100mA
  - Look at targeted peripheral list to determine amount of power required for your application
- CPU Utilization
  - Directly related to USB throughput
  - Many USB controllers interrupt frequently

System Requirements

- CPU Utilization
  - Frequency of Interrupts (fewer is better)
  - Data Movement
    - Data bus size (8 bit vs. 16 bit vs. 32 bit)
  - Direct Memory Access (DMA)
    - Slave DMA
    - Bus Mastering DMA
- Memory Requirements
  - Single while loop vs. operating system
  - Targeted Peripheral List (i.e. number of devices supported)
- Additional Software beyond USB
  - File system (Mass Storage Applications)
  - PCL (Printer Control Language for graphics printing)
  - TCP/IP Stack (Networking applications, USB-to-Ethernet)
  - Bluetooth Stack (USB-to-Bluetooth)

System Requirements
Interrupt Frequency



System Requirements
Direct Memory Access (DMA)

- Bus Mastering
  - Controller is capable of accessing system memory directly
  - Possible in SOC designs (Silicon IP), less likely in discrete chip implementations
- DMA Slave
  - Controller requires a DMA Master to move data to and from the controller
  - Supported in most discrete USB On-The-Go implementations
    - Adds a level of complexity to the software for Host transfer scheduling
    - Reduces CPU utilization as data movement is off loaded from CPU
    - Reduces USB throughput as transaction scheduling is less deterministic

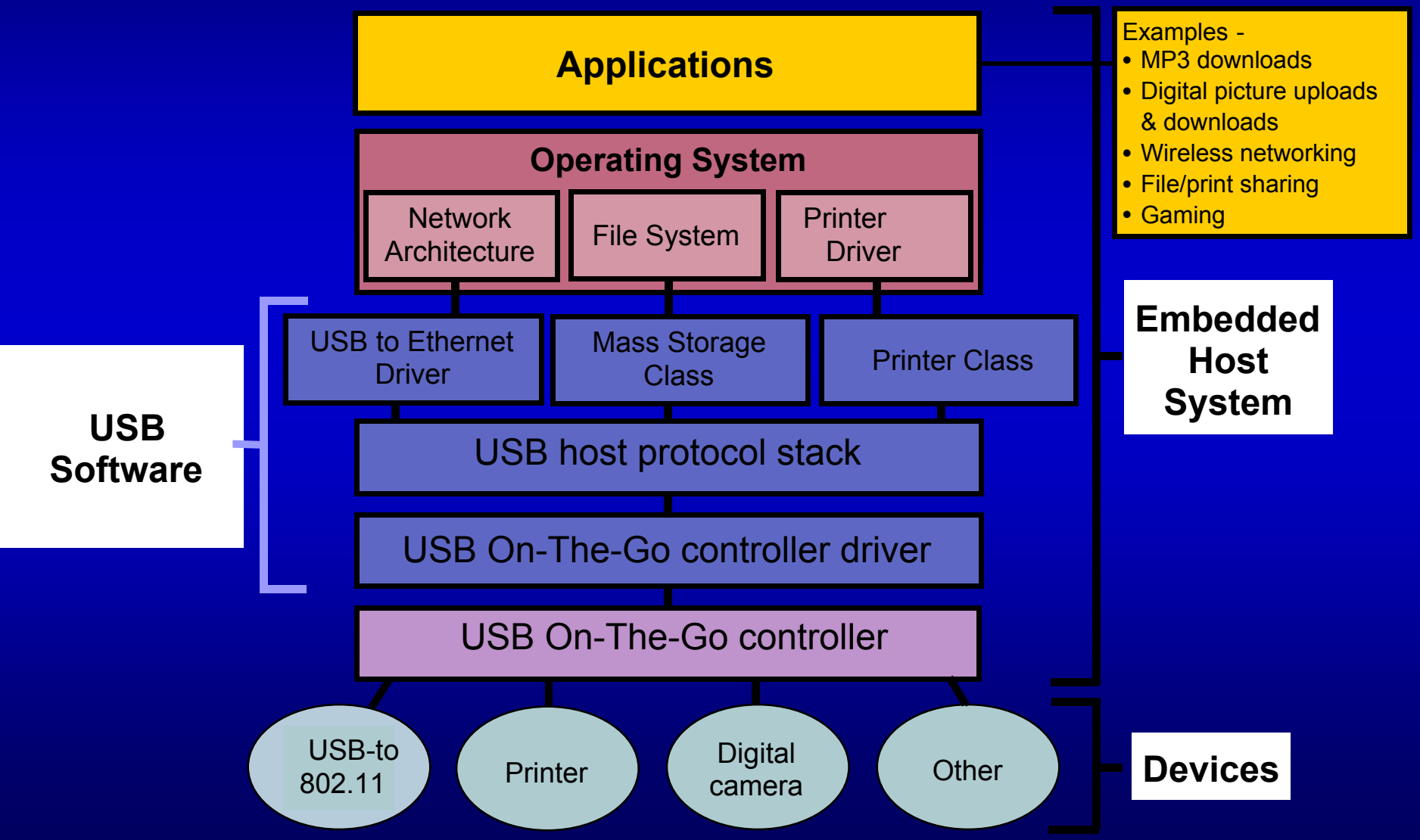
System Requirements
Memory Requirements

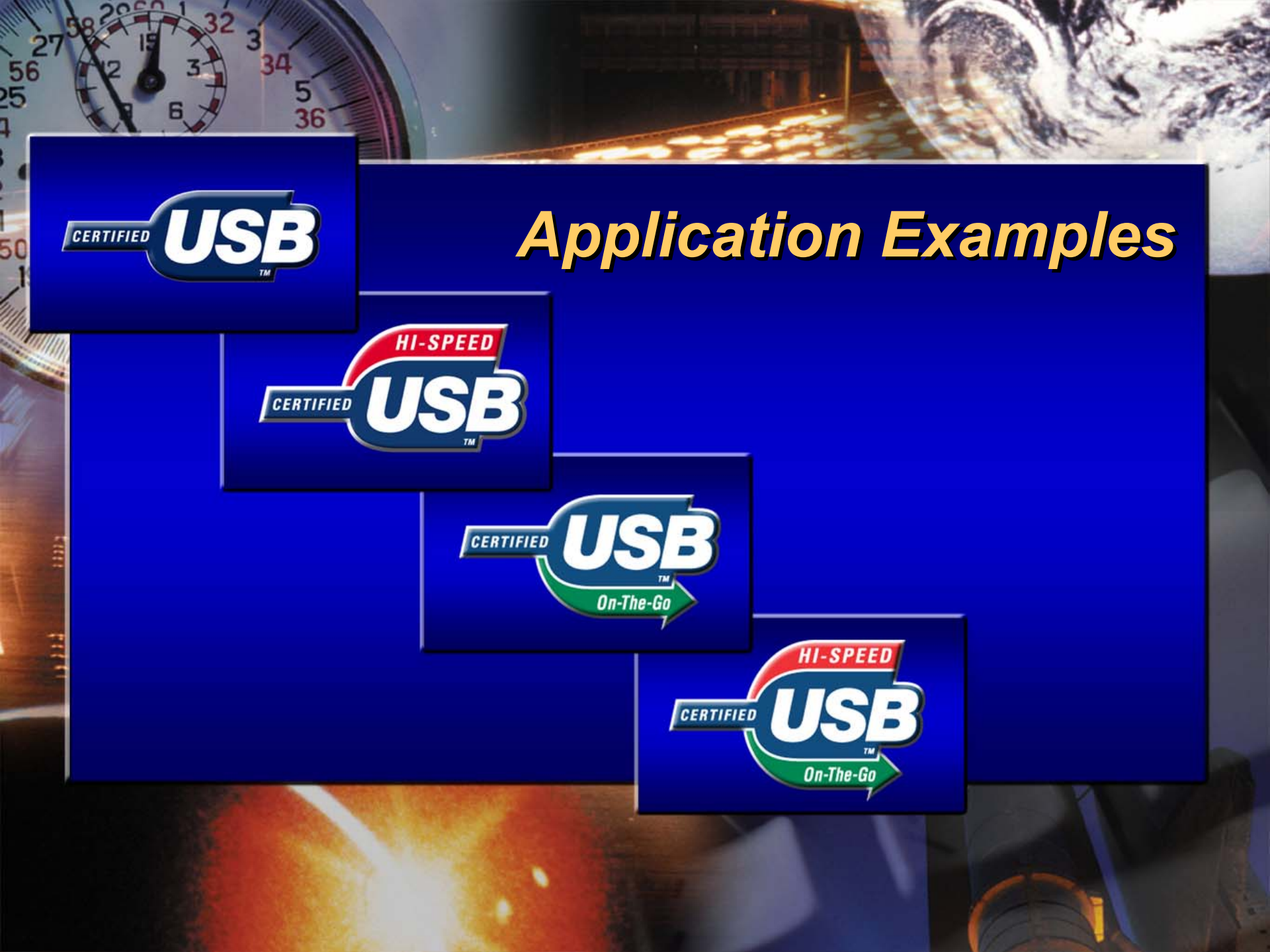
- USB On-The-Go Implementations
  - Simple USB On-The-Go implementations may utilize while(1)
  - OS likely to be present
  - RAM Requirements (32-128K)
  - ROM Requirements (70-150K)
- Targeted Peripheral List
  - Manufacturer, model number list
  - The more device classes are supported the more ROM and RAM required in the system
  - Lists can become outdated quickly
  - Standards required to help alleviate software burden

System Requirements
Other Software Required

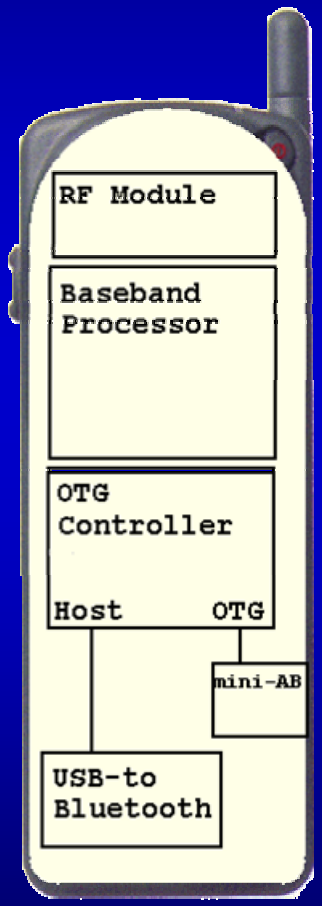
- USB-to-Bluetooth
  - Bluetooth Stack
- Mass Storage Devices
  - File system
  - FAT 12/16, FAT 32 for most HDs and memory media
  - UDF
  - ISO9660
- Networking Devices
  - USB-to-Ethernet
  - USB-to-802.11
  - Communications Class Devices
- USB Printers
  - PCL (Printer Control Language) Driver
  - Direct Print Standard (DPS), Direct print from DSC to Printer

System Requirements
Other Software Required



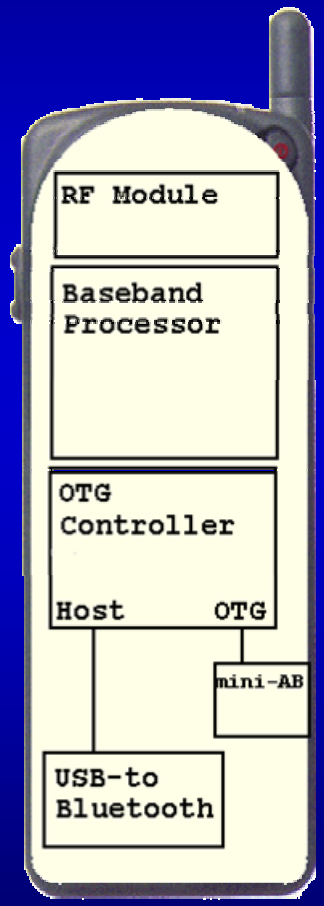

CERTIFIED USB
Application Examples
HI-SPEED
CERTIFIED USB
CERTIFIED USB
On-The-Go
HI-SPEED
CERTIFIED USB
On-The-Go

Mobile Handset Application



- USB On-The-Go Controller (2 ports)
  - Internal Host port
    - USB-to-Bluetooth chipset
  - External Mini-AB
    - Connection as a peripheral
      - Communication Class
    - Connection as Host
      - Support for Disk-on-Key (MSC)
      - Mobile-to-Mobile Connectivity (Proprietary)
- Power
  - External mini-AB to supply 100mA

Mobile Handset App - Software



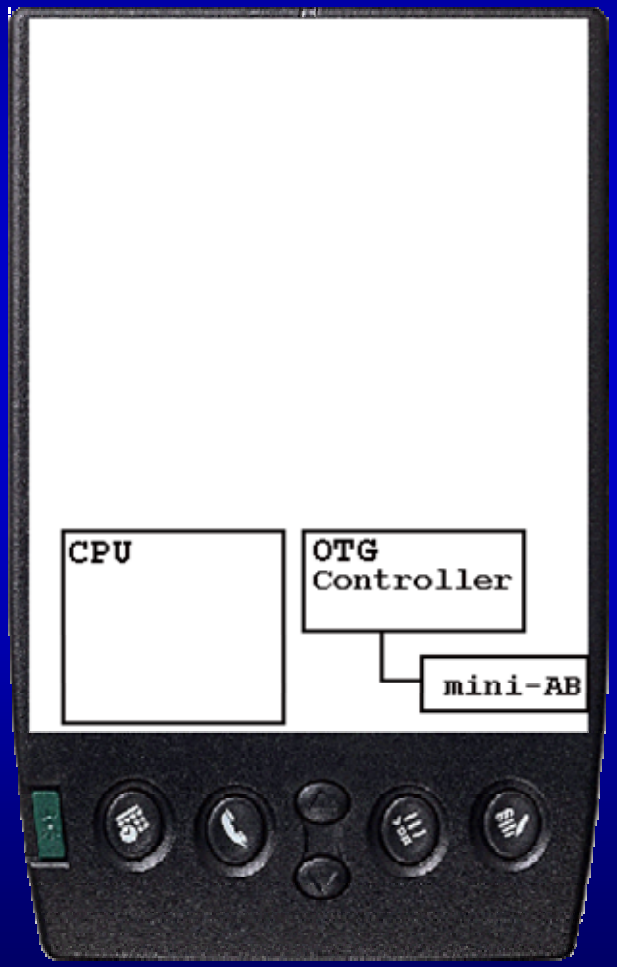
Host Software

- USB-to-Bluetooth
  - Requires tie-in to Bluetooth Stack
- Mass Storage Devices
  - Requires tie-in to File System

Peripheral

- As a peripheral, the device controller supports the Communications Class Specification as an ACM device to be used as a modem call out device.

PDA Application



USB On-The-Go Hardware

- Discrete Hardware Implementation
  - USB On-The-Go controller connected to CPU system bus.
- External mini-AB connector for operation as a Dual Role Device (DRD)

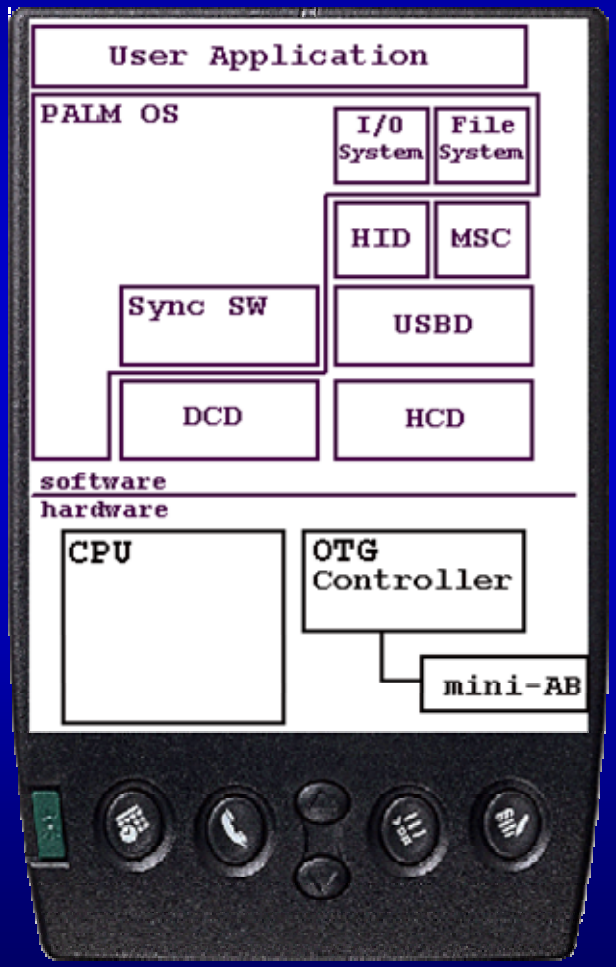
Applications

- As a peripheral, to support the existing PALM PC Sync software.
- As a Host, to allow Disk-on-key (thumb drives) to transfer files to/from the PALM device.

Power

- External mini-AB to supply 100mA

PDA Application - Software



Host Software

- HID Support (USB Keyboard, mouse)
  - Requires tie-in to input device subsystem
- Mass Storage Devices
  - Requires tie-in to PALM File System
  - PALM natively supports FAT12/16

Peripheral

- As a peripheral, the device controller driver is tied directly into the pre-existing PALM Sync subsystem