Developers Conference 2007

Amsterdam, The Netherlands

Software Designs for Certified Wireless USB Systems

David G. Brenner

Stonestreet One, Inc.

Agenda

- Stonestreet One background
- USB-IF wireless USB PDK support
- Host platform architecture and implementation
- Device side implementation
- User considerations
- Wrap up and questions

About Stonestreet One

- Wireless software and services
  - Founded in 1996
  - From standard products to custom designs
- Diverse customer base
  - From semiconductor to Tier 1 OEM’s
- Active participation in standards
  - USB-IF, WiMedia, Bluetooth SIG, …
- Developing UWB software since 2004
  - First products feature Wireless USB support

UltraSuite™ for Wireless USB

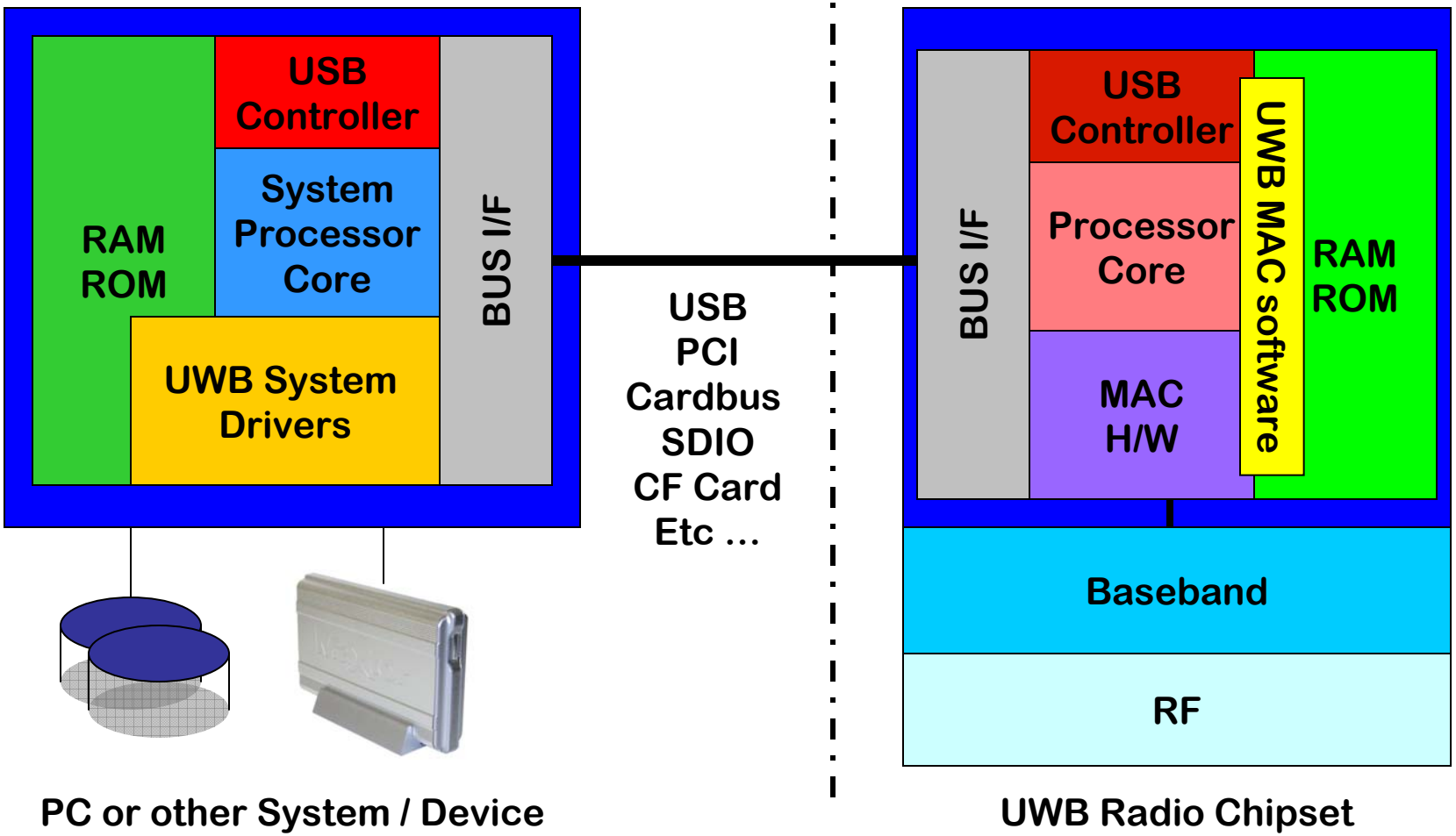
- On-chip protocol adaptation MAC software
  - Wireless USB as well as WLP and Bluetooth 3.0
- Device-side protocol drivers
  - Native device, simple host, dual role
  - OS's include Windows CE / Mobile edition, Linux, NetBSD, OSE, Nucleus, QNX, VxWorks etc.
- Host-side protocol drivers and association
  - Microsoft XP SP2 and Vista
  - HWA, DWA, WHCI, CBAF and NFCD
- Configuration and management services
  - Incorporated into WiCenter application

Stonestreet One and USB-IF

- Ultrasuite™ Windows drivers required by the USB-IF for certification testing
- Drivers for USB-IF HWA and WHCI PDK
  - Testing, development and certification
- Support on an as-is, best effort basis
  - Must have a USB-IF-issued PDK platform
- License agreements required for any production use of the software

Go to www.stonestreetone.com/register

Wireless USB Platform Model

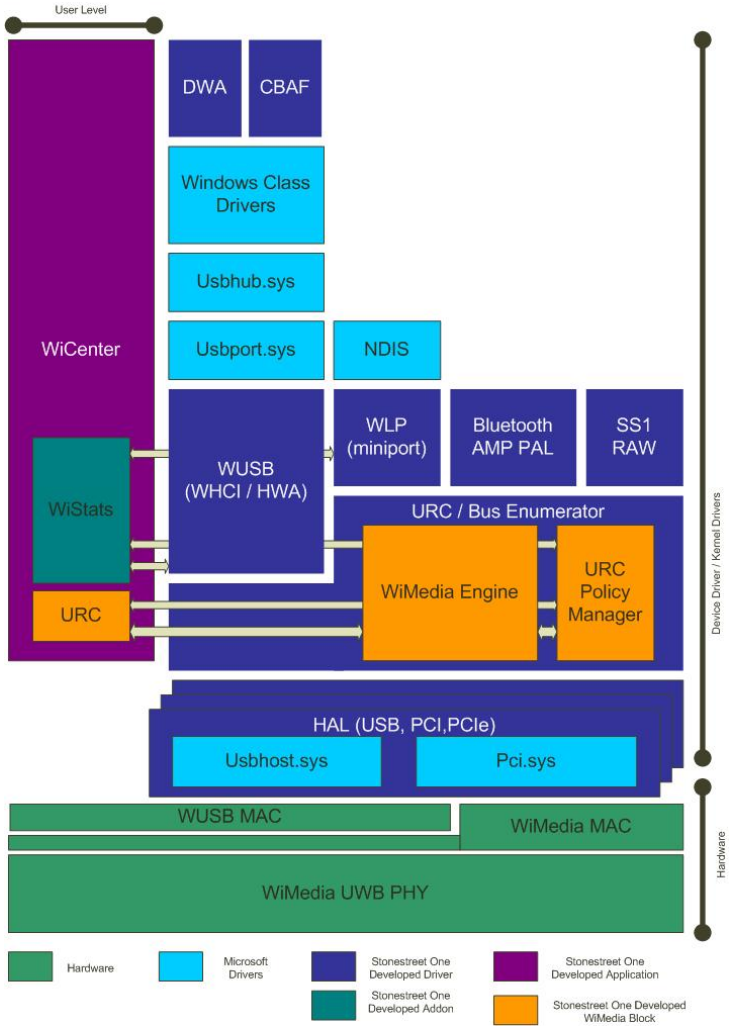


PC Host Software Requirements

- Support the USB-IF Certified WUSB 1.0 and WHCI specifications
- Transparently integrate with existing PC wired USB host environment
- Provide secure association support for Wireless USB device connections
- Enable a true WiMedia multi-protocol adaptation layer capability

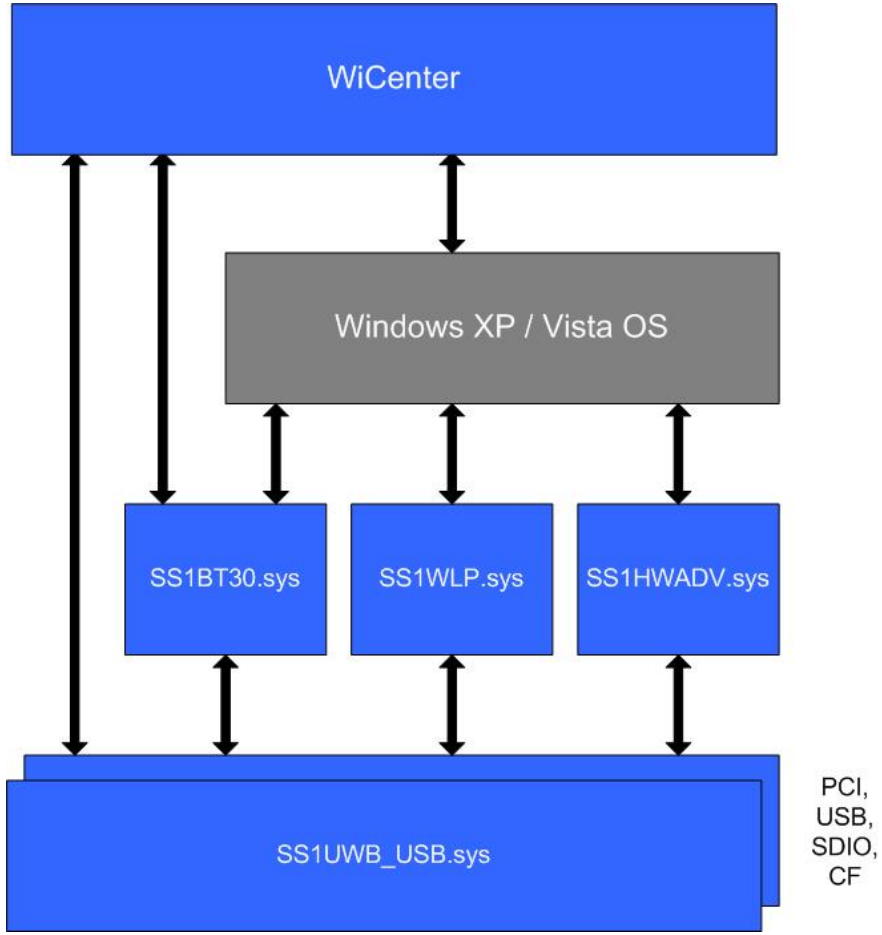
Windows Wireless USB S/W Details

- Driver stack integrates with Microsoft USB driver services
- SS1 drivers are transparent to USB class device drivers enumerated above and below
- Driver adds extensions for security and association
  - Required before the USB wired equivalent connection
- PNP builds the USB stack above the HWA (or WHCI) and DWA
- Design works with existing Microsoft class drivers
- Plugs in cleanly with Windows device manager and USB tree



Windows Host Architecture

- Built on lower layer WiMedia PAL support
- URC driver exposes bus interface(s) for each enumerated WiMedia PAL, allowing extensibility
- Upper layer application interface through IOCTL's
- Transparent device driver communication through IRP messages



UltraSuite UWB Windows Design

- URC, HWA, WHCI, DWA, CBAF are separate, standalone drivers
- Both HWA and DWA driver can support vendor extensions
  - Statistics / performance measuring
  - Firmware loading
  - Video and other services …
- DWA driver works with existing Microsoft native hub driver
- HWA Driver supports automatic reconnection of wireless devices
  - Without the need for upper level application / service

UltraSuite Host Features

- Dynamic / static RPipe management
  - Block size, selection, etc…
- Support multiple requests per pipe
- Device address management
- IE management engine with priority scheme
- IOCTL Interface for numeric (wireless) association
- Tunable debug messaging

SS1HWADV.sys

UltraSuite DWA Features

- Host-side PC driver for downstream Wireless USB DWA system
- Dynamic / static RPipe management
  - Block size, selection, etc…
- Device address management
- Composite device support
- Supports Microsoft native hub driver
- Tunable debug messaging

SS1DWADV.sys

Wire Adapter Transfer Engine

- Common for both DWA and HWA
- Zero copy architecture
  - Supports fast, low latency message and data passing between drivers
- Limited Spinlocks
  - Core design is DPC – event driven
- Highly Optimized
  - Request partitioning
  - Priority device traffic management
- Sideband IOCTL interface
  - Provides driver access for testing, tuning and customer support

UWB Radio Control (URC)

- Dynamic DRP allocation engine
  - Grows and shrinks the number of host MAS reservations
  - Allows for other WiMedia devices to coexist in the same channel
- Implements device address conflict resolution and other WiMedia “good neighbor" policies
- Up-to-date implementation of types as based on WHCI specification
- URC Driver implements the flow as specified in the WHCI specification
- Exposed bus interface allows for dynamic addition and negotiation between different PALS (HWA / WHCI / WLP)

Additional Host Attributes

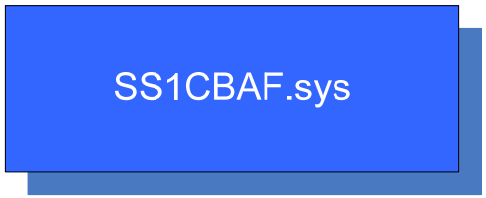
- Link Adaptation
  - Dynamic PHY rate adjustment
    - Per December errata
  - Dynamic max packet size adjustment
- Power Management
  - TX power level used to communicate with device based on LQI and RSSI values
- Suspend / Resume
  - Remote wake and low power modes in place with silicon enablement

Channel Selection Approach

- The drivers search for the best channel at any given time during the connection
  - Applies to FFC (non-hopping) configurations
- If a connection deteriorates, the driver will determine if there is a better channel
  - And will move the connection over as appropriate
- User-selectable / configurable band group policy dictates overall channel selection
  - 3Ghz, >6Ghz etc.
  - Geographic and country-specific regulatory policies

UltraSuite CBAF

- Separate Windows driver from DWA
- Implements the Cable Based Association Framework as defined in the Wireless USB Association specification
- Service attached through IOCTL interface allows for storing connection information
- Tunable debug messaging



Embedded Wireless USB Software

- Services appropriate to the application
  - Native device, simple host, DRD
- A number of examples deploying now
  - Wireless mass storage, multi-function printers ..
- Partitioning will change over time
  - Early designs need to ‘bolt’ to existing SoC USB device cores and implementations
  - Wireless USB cores will penetrate designs and give greater access to native services

Embedded S/W Considerations

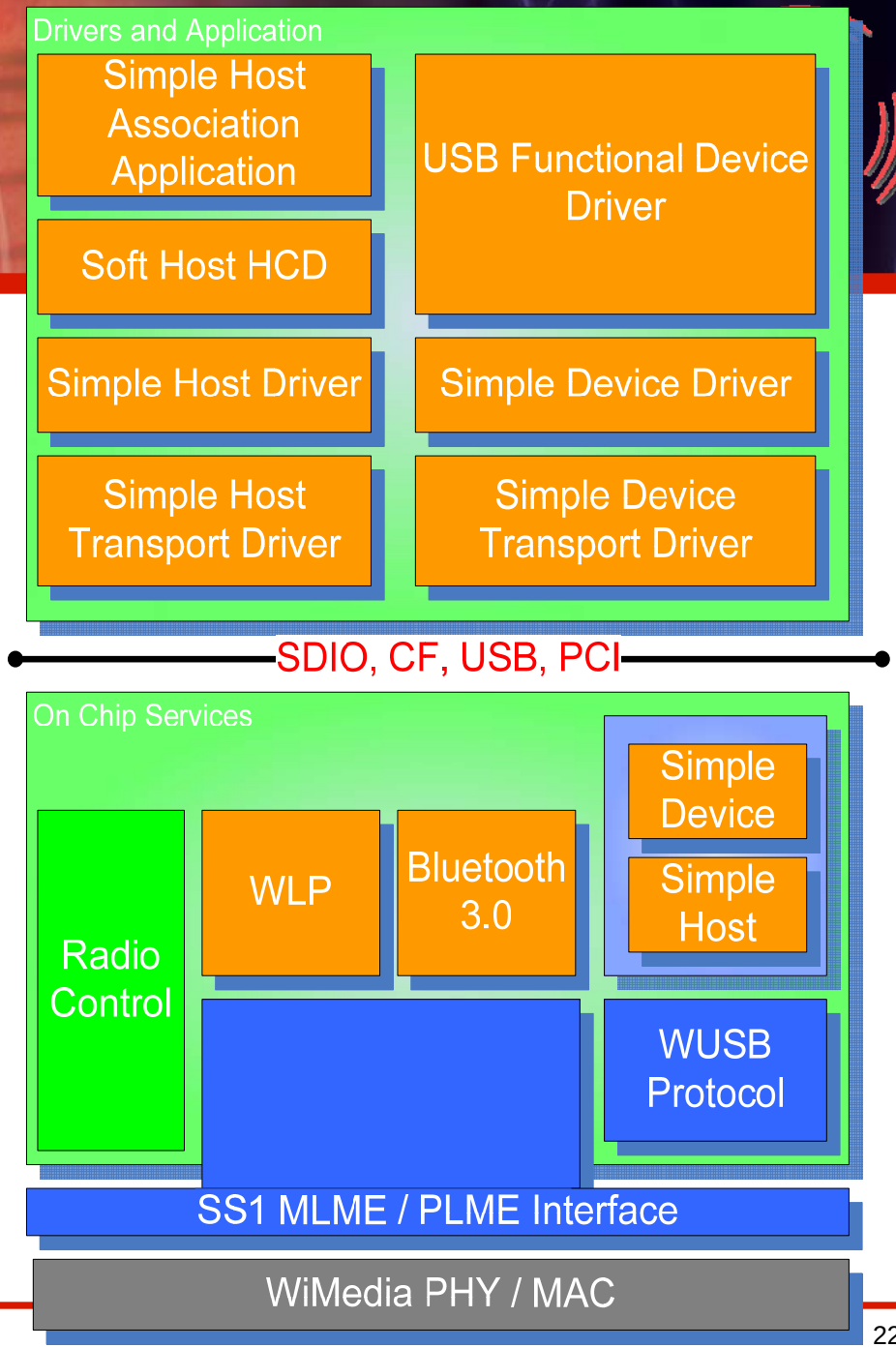
- System processor
- USB controller core
- Operating system and built-in services
- Physical bus and transport models
- WUSB host, device, or both
- Memory availability and impact
- Protocol stack requirements
- Performance needs and goals

Partitioning the Solution

- Varies based on system and architecture
  - i.e. PC notebook versus embedded device
- Hardware-based functions
  - Low layer WiMedia MAC timing
  - Packet encryption
  - Buffering and multi-PAL arbitration
- Software handles higher level protocols
  - Key exchange and related association
  - User configuration, policy, and management

SS1 Embedded Architecture

- Solution enabled by S/W on both host processor and UWB MAC / chipset
- UltraSuite includes standard I/F for MAC-based PAL services
- Wireless USB options ‘plug-in’ depending on system configuration



Native Device Design

- MMC decoder engine
- Rapid MMC recovery
- Dynamic device configuration
  - Mass Storage
  - RNDIS
  - Others …
- Ties into device services
  - Mass storage reference application
  - Active sync support

Simple Host Services

- Complex MMC scheduling engine
- Channel model adaptation service
- Single Wireless USB device support
  - API limitation
  - Platform constraints
- Out of band association support
- Class driver based on connection scenarios
  - Storage, printer, sync etc.

Dual Role Design

- Modal
  - Functions within one reservation during a super frame as either a host or a device
    - Controlled by user selection / configuration
- Concurrent
  - Operates within two private reservations during the same super frame, one reservation as a host and one reservation as a device
    - Limited based on memory, MIPS and code space

UltraSuite MAC Software

- Robust UltraSuite firmware driver support
  - PAL registration
  - MAS reservation
  - Notification of global radio control events
    - Airplane mode, power management etc.
- Straight-forward APIs to driver PAL's
- OS function integration is PAL-specific
  - Does not need to be reworked for integration
- WiCenter control abstracted through "IRadio" objects
  - On a per-PAL basis
  - Easily adapted for new driver PAL's

Performance Considerations

- Driver tuning
  - Dynamic versus static Rpipes
  - Memory usage
  - MAC-specific implementation details
- Bus loading
  - USB bus hub/tree topology
  - Other internal USB device loads
- UWB and WiMedia
  - Picocell co-existence and concurrency

User Perspective and Tools

- Wireless USB association models require active user involvement
  - Cable insert / de-insert
  - PIN code entry (numeric compare)
- Usage considerations
  - Auto-reconnect or manual (every time)
  - Context and location awareness
    - Other devices and connections
- Functional OS transparency
  - From boot time (kernel load) through user mode

Summary

- Stonestreet One supports PC and embedded host / device services
- Architecture varies depending on resources and target system
- Software ranges from on-chip (PAL MAC services) to system processor
- Wireless USB software may need to be native device, simple host, and/or both
- Security model demands a user role

For More Information

- Visit our web site: www.stonestreetone.com
- Email us at info@stonestreetone.com
- For PDK drivers:
  - Register at www.stonestreetone.com/register
- My contact information:
  - Email dbrenner@stonestreetone.com
  - Direct / mobile (214) 738-1099

Developers Conference 2007

Amsterdam, The Netherlands