



Certified Wireless USB Host Controller

Abdul R. Ismail
Intel Corporation

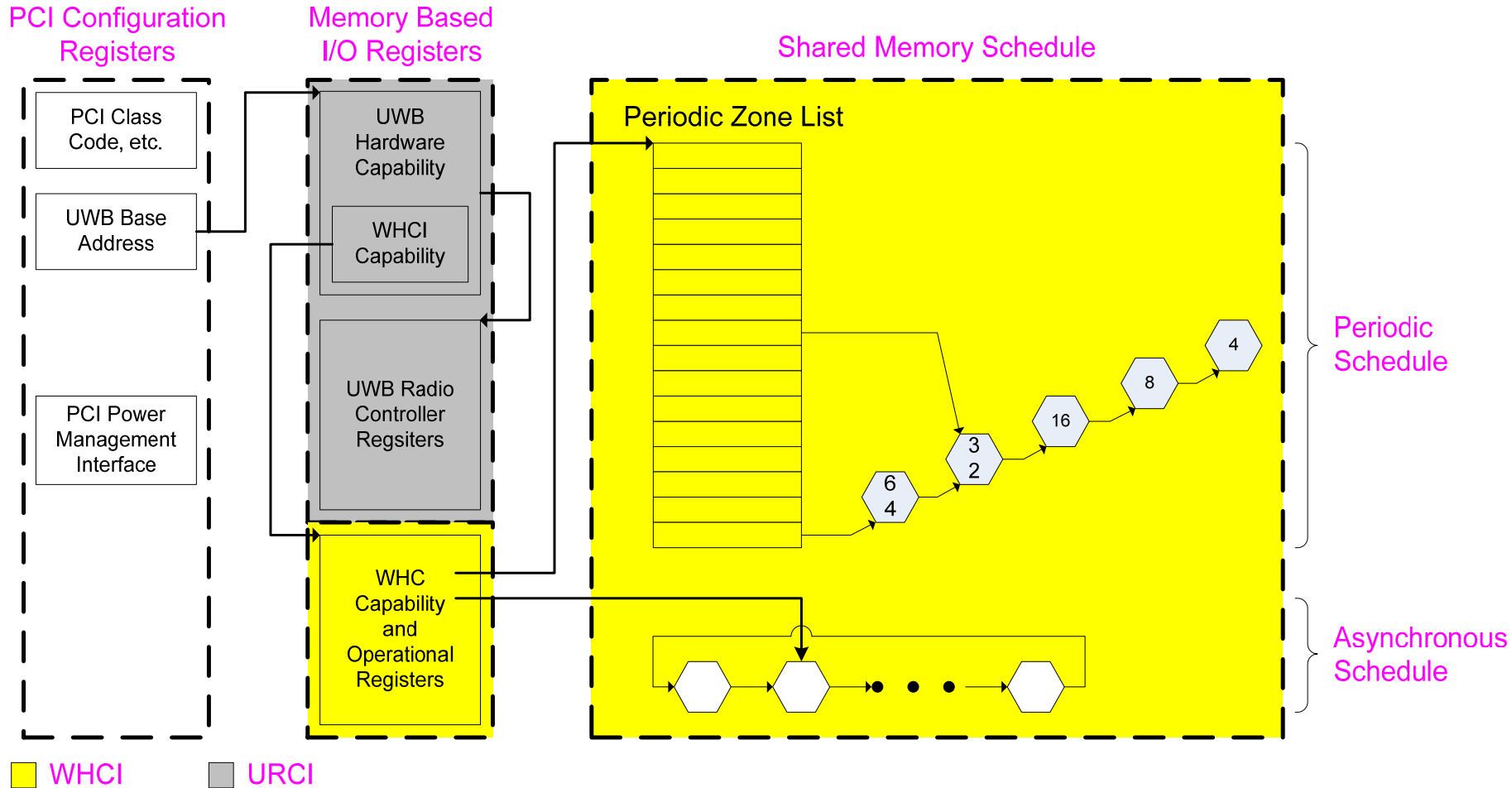
Agenda



- Architectural Overview
- UWB Radio Controller (URC)
- Certified Wireless USB Host Controller (WHC)
- Next Steps

UWB Multi-Interface Controller

Interface Architecture





- One PCI Function
- Multiple, quasi-independent functional units
 - Radio Control (base)
 - Driver provides control services to Applications (control plane)
 - Application-specific Interfaces (WHCI, etc.)
 - Used for Application-specific data plane
- Separate register space 'regions' within the same PCI Function
 - 'Bus' Driver loads for base interface, then loads application-specific drivers and provides them with pointer to their register set region.
 - No driver load order problems
 - ALL PCI device function operations are coordinated through the base driver (Power management, etc.)
- Supports both 32 and 64 bit addressing

Certified Wireless USB

Host Controller Interface (WHCI)

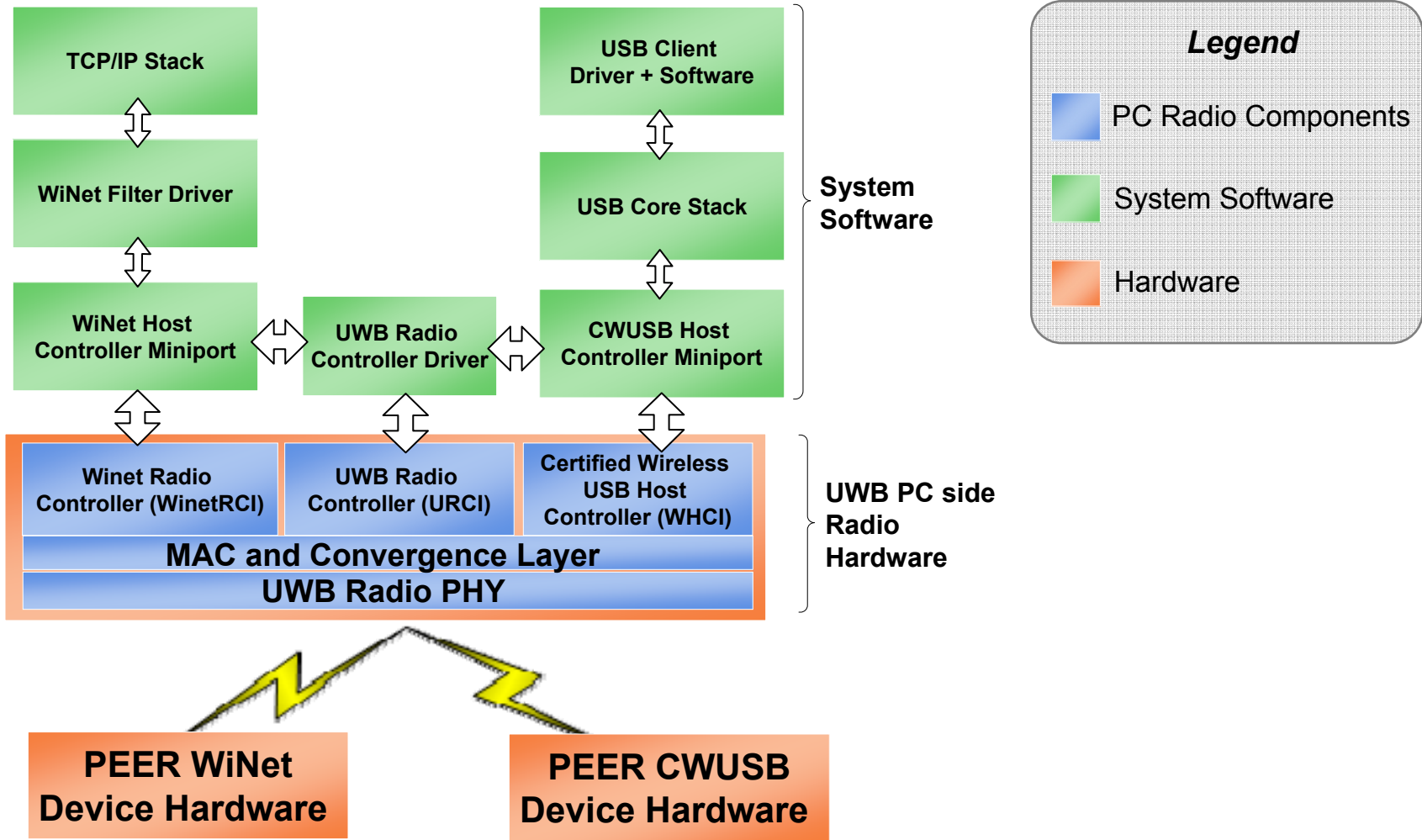


- Architecture derived from EHCI
- Established driver models and hardware implementations
- Certified Wireless USB and WiMedia Policies will effect HC system behavior
- Base WHCI architecture has some needed differences

	USB 2.0	CWUSB	Comment
Periodic Sample Rate (min/max)	125 usec/~4 sec (typical: 4 to 8 ms)	~4ms/~4sec	WiMedia MAC & channel allocation rules limit the minimum periodic rate to ~4ms (with an 8ms gap every 65 ms (superframe))
Protocol	One transaction at a time	Multiple transactions	Split transaction protocol for all transfer types allows multiple transactions active at a time. High performance HC's will tend to fetch and hold more of the scheduling data structures (less frequent memory activity)
Device Notifications	Not Available	Available	Asynchronous device-to-host message protocol (host schedules channel time)
Flow control	Busy-wait	XON/XOFF	Device notifications allow for an Xon/Xoff type flow control protocol

UWB Multi-Interface Controller

Software Architecture





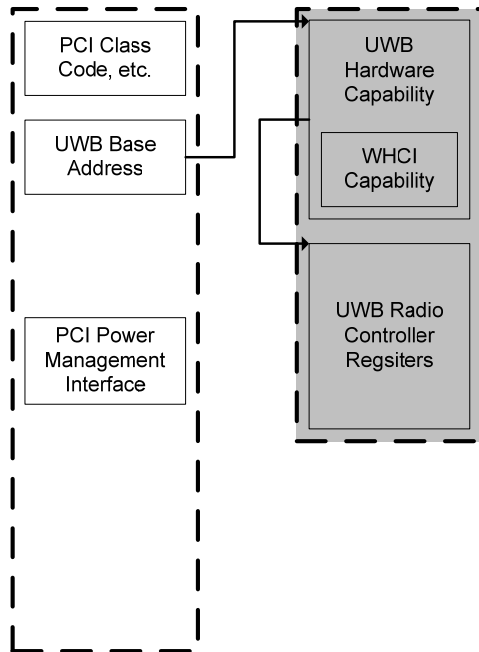
Agenda

- Architectural Overview
- UWB Radio Controller (URC)
- Certified Wireless USB Host Controller (WHC)
- Next Steps

UWB Radio Controller Interface



- Exposes the other capabilities of the UWB Hardware
- Implements all mandatory functionalities defined by WiMedia
- Routes all interrupts to the appropriate interface
 - If enabled via the URC Interrupt Enable Register
- Provides the ability to control the Radio
- Informs system software of asynchronous events

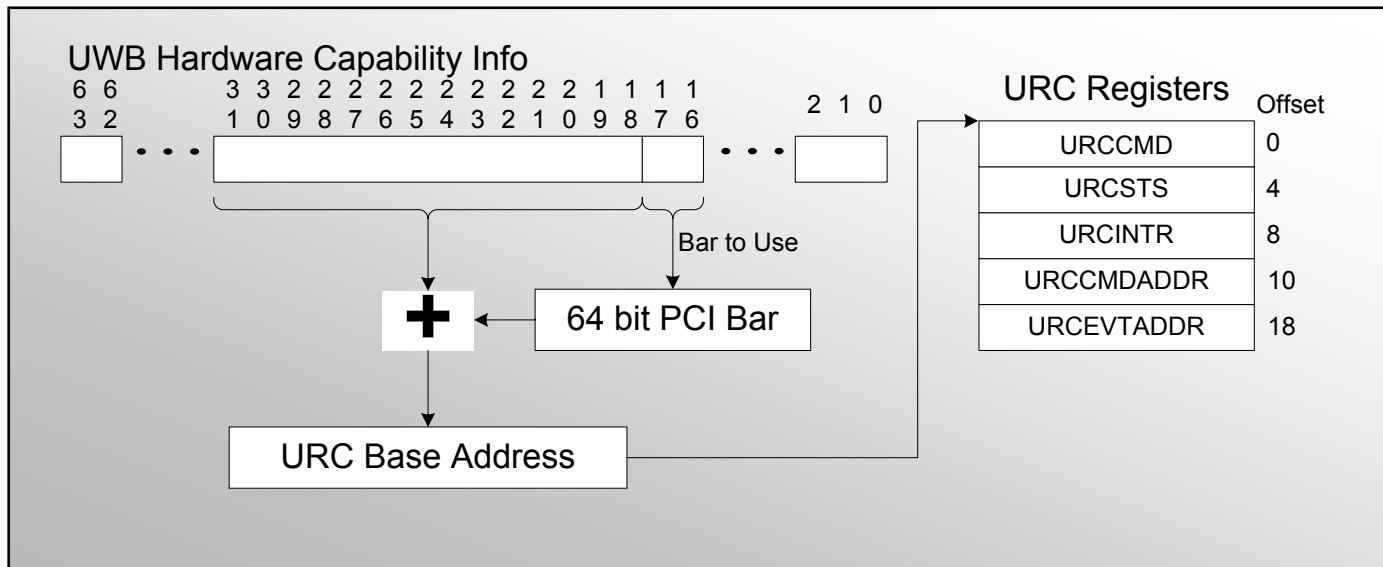


UWB Radio Controller

Registers



- The location of first register is determined by the values in the Offset and Bar to Use fields of the UWBCAPINFO Register
- The offset of each subsequent register is defined by the specification



URC Command Format



CHANNEL_CHANGE	16
DEV_ADDR	17
GET_IE	18
RESET	19
SCAN	20
SET_BEACON_FILTER	21
SET_DRP_IE	22
SET_IE	23
SET_NOTIFICATION_FILTER	24
SET_TX_POWER	25
SLEEP	26
START_BEACONING	27
STOP_BEACONING	28
BP_MERGE	29
SEND_COMMAND_FRAME	30
SET_ASIE_NOTIFICATION	31

Command Type	Command	Command Context	Parameter0	...	ParameterN-1
GENERAL 0	VENDOR SP: F0-FFH [16-31]	ID 1-FEH	Size and data are specific to each command		

- Command Context ID is used to match the event response to a Command
- Parameters are command specific
- The 16 bit Command defines the unique commands for each command type

URC Event Format



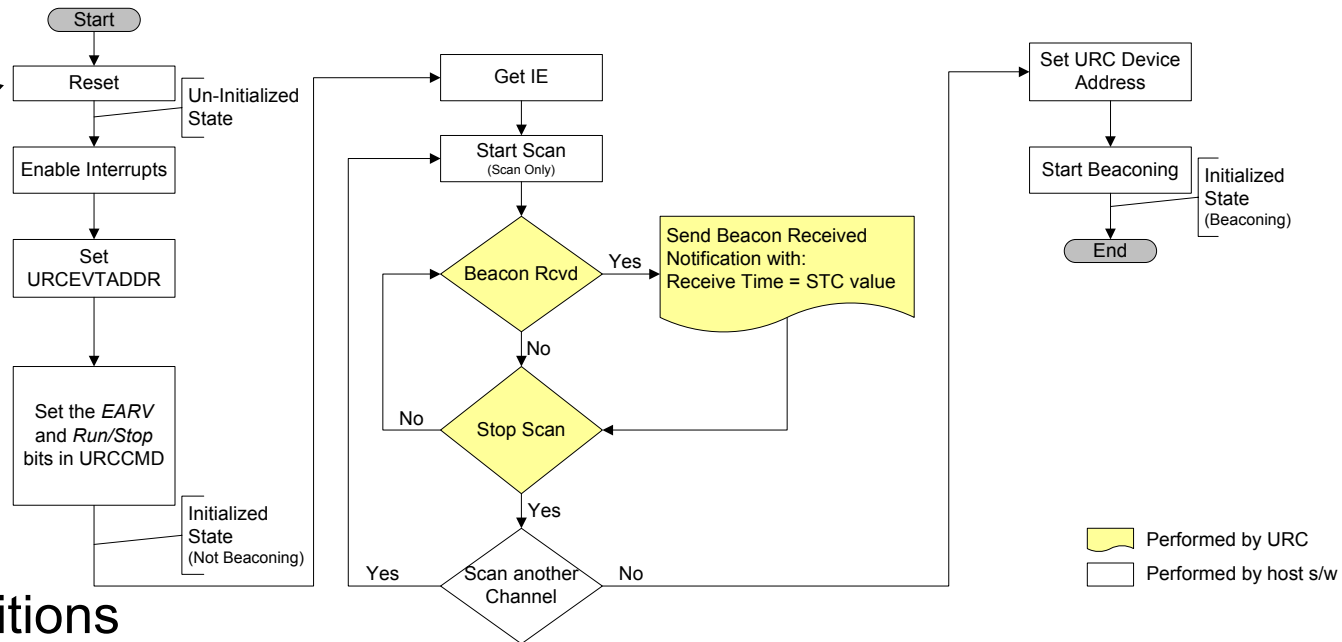
EventType	Event	Event Context	Parameter0	...	ParameterN-1
GENERAL 0 VENDOR SP. F0-FFH	NOTIFICATIONS [0-8] RESULTS [16-28]	EVENT 0 RESULT 1-FEH	Size and data are specific to each command		

IE_RECEIVED	0
BEACON_RECEIVED	1
BEACON_SIZE_CHANGE	2
BPOIE_CHANGE	3
BP_SLOT_CHANGE	4
BP_SWITCH_IE_RECEIVED	5
DEV_ADDR_CONFLICT	6
DRP_AVAILABILITY_CHANGE	7
DRP	8
BP_SWITCH_STATUS	9
COMMAND_FRAME_RECEIVED	10
CHANNEL_CHANGE_IE_RECEIVED	11

CHANNEL_CHANGE	16
DEV_ADDR	17
GET_IE	18
RESET	19
SCAN	20
SET_BEACON_FILTER	21
SET_DRP_IE	22
SET_IE	23
SET_NOTIFICATION_FILTER	24
SET_TX_POWER	25
SLEEP	26
START_BEACONING	27
STOP_BEACONING	28
BP_MERGE	29
SEND_COMMAND_FRAME	30
SET_ASIE_NOTIFICATION	31

- The result of a command is returned to the URCD via an Event Block
- If asynchronous notifications are enabled, these are also sent to system software using an Event Block
- All Events are returned via the Event buffer

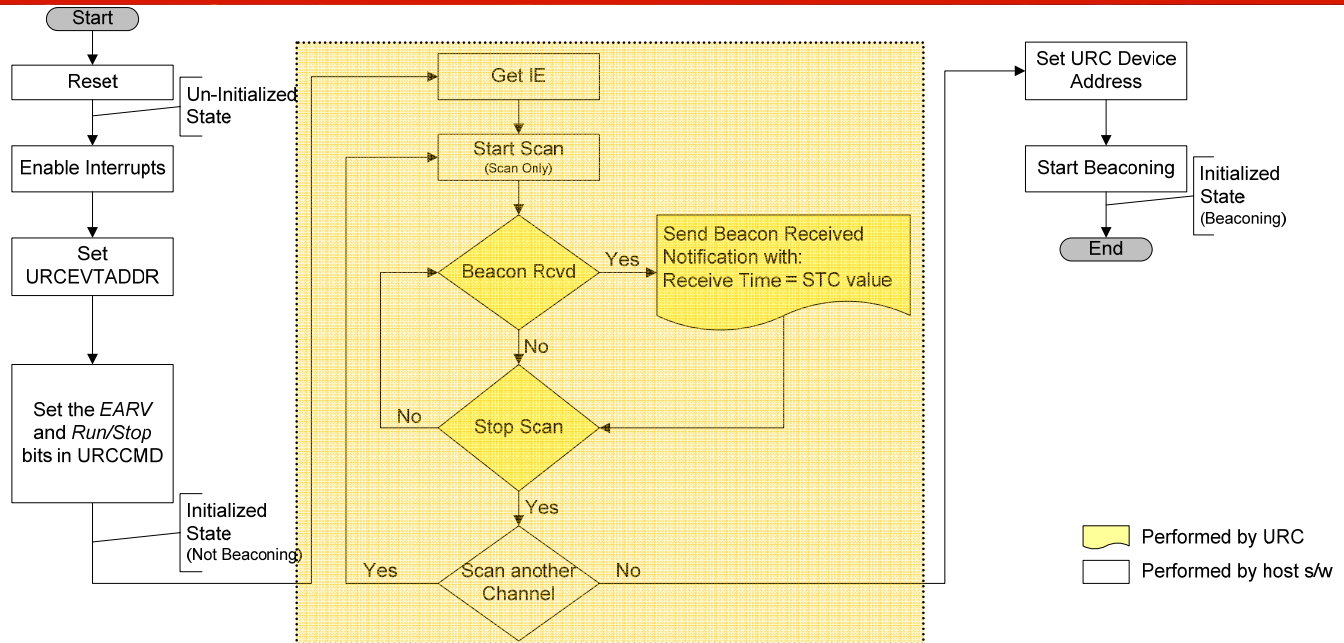
URC Initialization



- Reset conditions

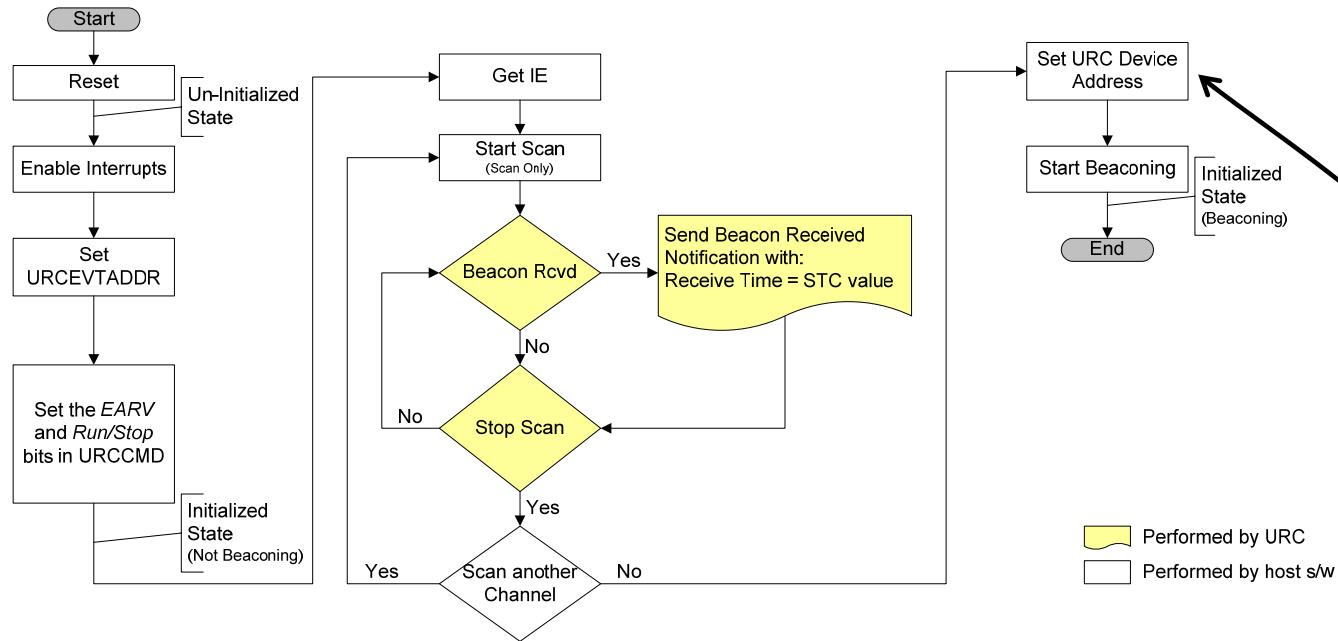
- Highest transmission power level
- Free-running Superframe Time Counter (STC)
- Scan disabled
- Beacon Notification filters off
- Notification filters all off
- Buffers empty

URC Initialization



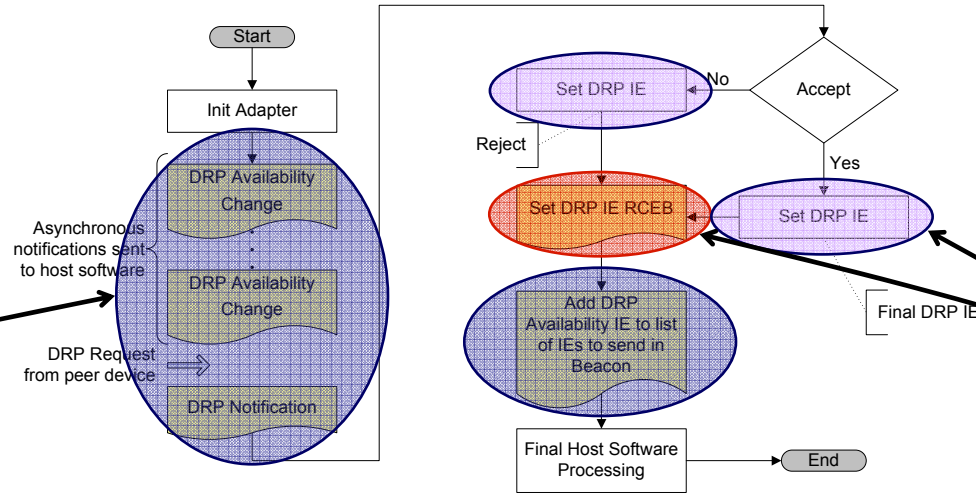
- Get IE
 - Used to derive the UWB channels supported
- RCFD starts Scanning
 - The RC returns Beacon Received notifications generated for every received beacon
- RCFD determines the channel to be used

URC Initialization



- IEEE defined 48-bit extended unique identifier (EUI-48)
 - This can be either statically configured or assigned by the host
- 16-bit device address (DevAddr)
 - It is host's responsibility to assign one to the URC and modify it as needed

Example: Implicit DRP Negotiation



- RCFD chooses which notifications it wants to receive
- By default all notifications are enabled
- Asynchronous notifications
- Commands sent to the URC
- Notification that is the result of a previous Command



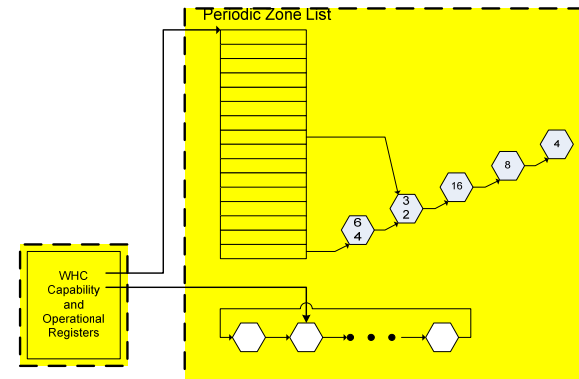
Agenda

- Architectural Overview
- UWB Radio Controller
- Certified Wireless USB Host Controller
- Next Steps

Certified Wireless USB Host Controller Interface



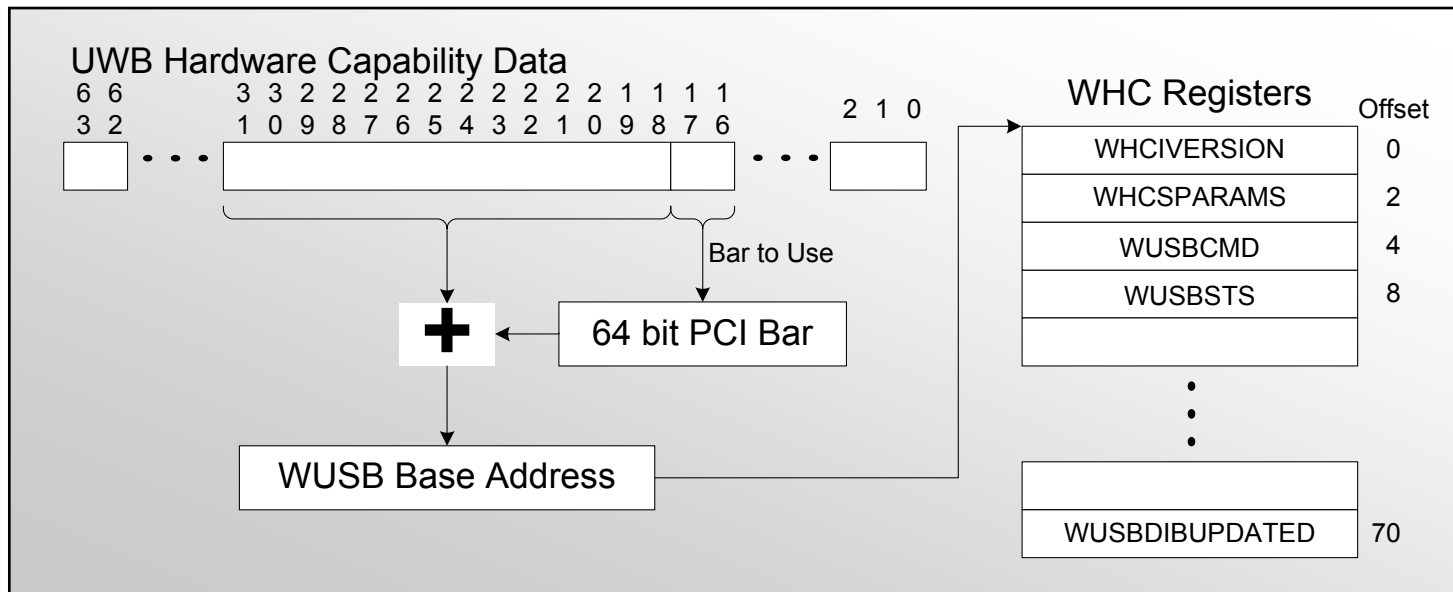
- Provides an EHCI-ish register and data structure interface
- Follows the CWUSB Protocol for OTA communication
- Ability to only Reset the CWUSB hardware
- Provides support for all USB data transfer types
 - Including low power interrupt
- Provides support for device power management (Suspend/Resume/Remote Wake)
- Uses the same interrupt as the UWB Radio Control Unit
- Exposes Capability and Operational Registers



WHC Registers



- The location of first register is determined by the values in the Offset and Bar to Use fields of the UWBCAPDATA Register (WHCI Capability)
- The offset of each subsequent register is defined by the specification



WHC

Initialization



- Enable Interrupts from the WHC
- Set the Device Notification Buffer address
- Set the base address of the Device Information Buffer
- Set the Broadcast Cluster ID, Stream Index and turn the WHC ON
- Set any Information Elements to be broadcast within MMC packets.
- Set the group key to be used for authenticating broadcast packets
- Establish a Wireless USB Reservation



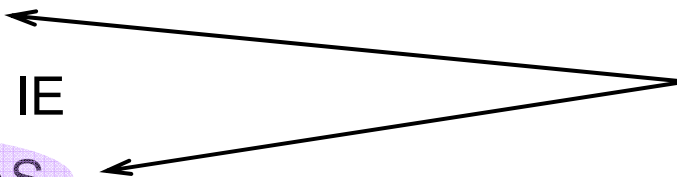
- Interface accessed via 3 Registers

- WUSB Generic Command Status Register
- WUSB Generic Command Parameters Register
- WUSB Generic Address Register

- Commands

- Add MMC IE
- Remove MMC IE
- Set WUSB MAS
- Channel Stop
- Remote Wake Poll Enable

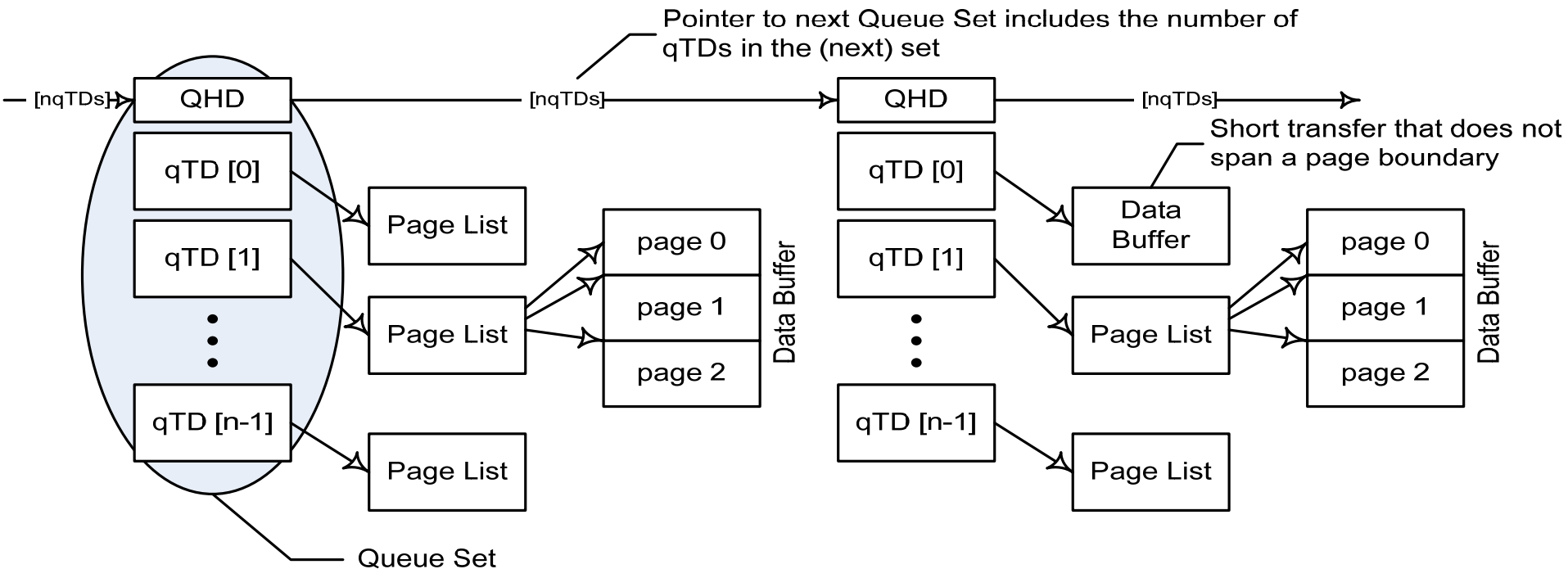
**Use the Generic
Address Register**



- A WHC must support secure transactions
- Interface accessed via 3 Registers
 - WUSB Set Security Key Command Register
 - WUSB Temporal Key ID
 - WUSB Security Key
- Commands
 - Set Encryption Key
 - Erase Encryption Key
 - Set Group Key
 - Erase Group Key

Certified Wireless USB Host Controller

Queue Sets

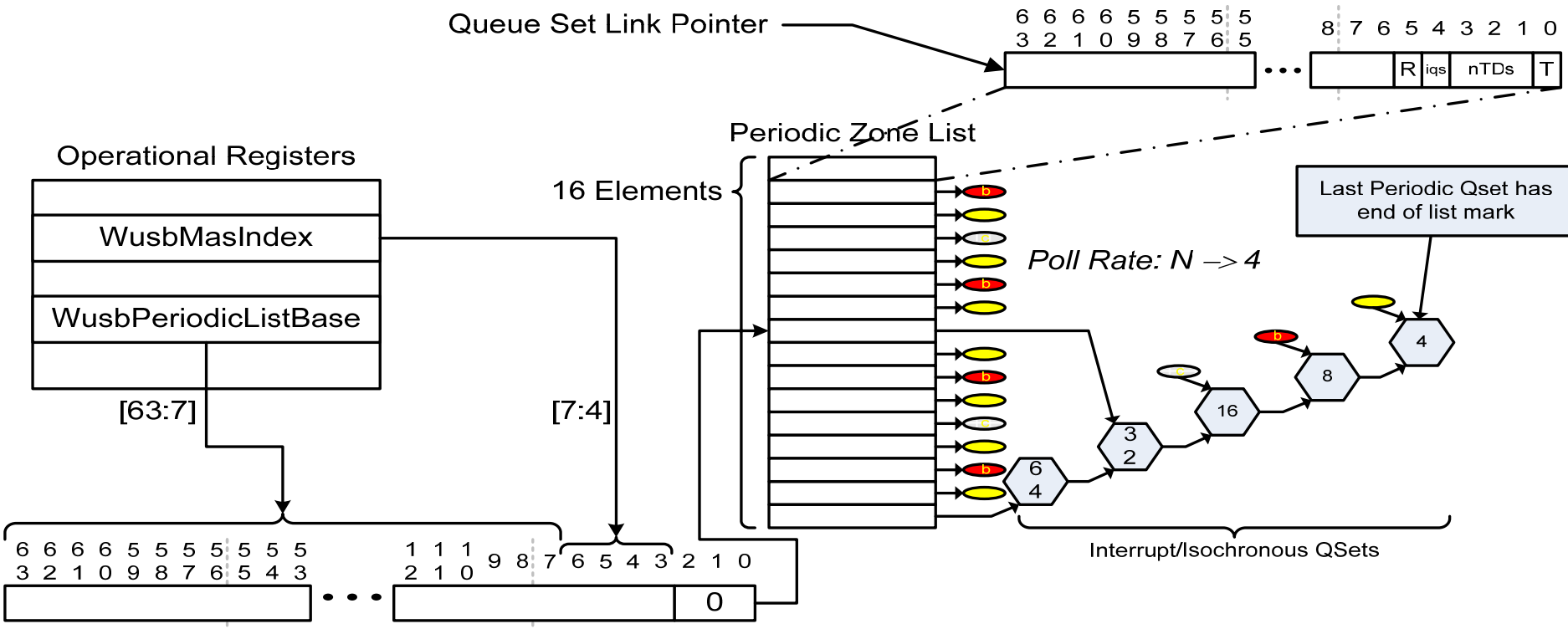


- Set of data structures in contiguous physical memory
- Cannot cross a physical page boundary
- Reduces the number of 64 bit pointers present in the QHead and its associated overlay space

Certified Wireless USB Host Controller



Periodic Schedule



- 128 byte aligned array of Zone List Link pointers
- A Zone consists of 16 Media Access Slots
- No CWUSB transactions occur in the first zone
 - Beacon Period



Agenda

- Architectural Overview
- UWB Radio Controller (URC)
- Certified Wireless USB Host Controller (WHC)
- Next Steps

Next Steps



- Sign the license agreement
- Download from <http://www.intel.com/go/wusb>
- Send feedback to whci@intel.com

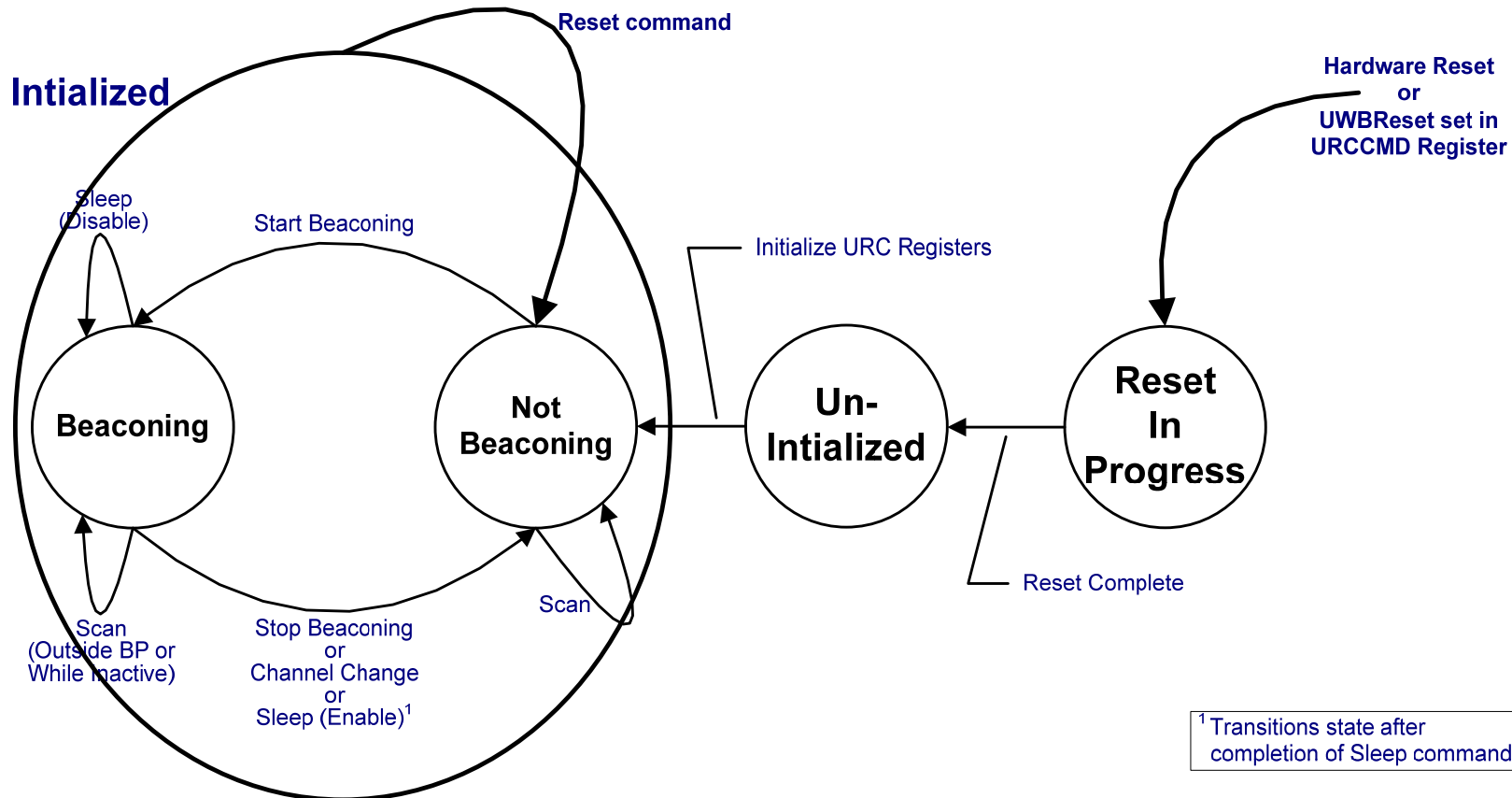
WHCI



Question & Answer

UWB Radio Controller

State Diagram



- All URC commands are executable in the initialized state only



Developers Conference 2006

Taipei, Taiwan