Wireless USB Protocol

John S. Howard

Senior Architect

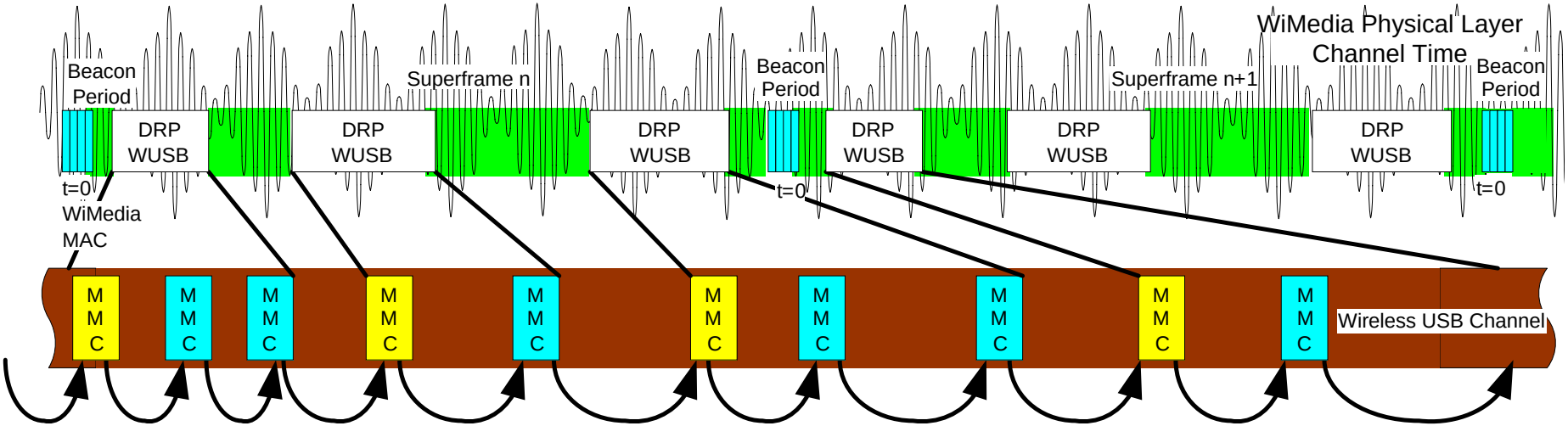
Intel Corporation

Agenda

- Protocol Components
- Data Transfers (general model)
- Device Notifications
- Flow Control

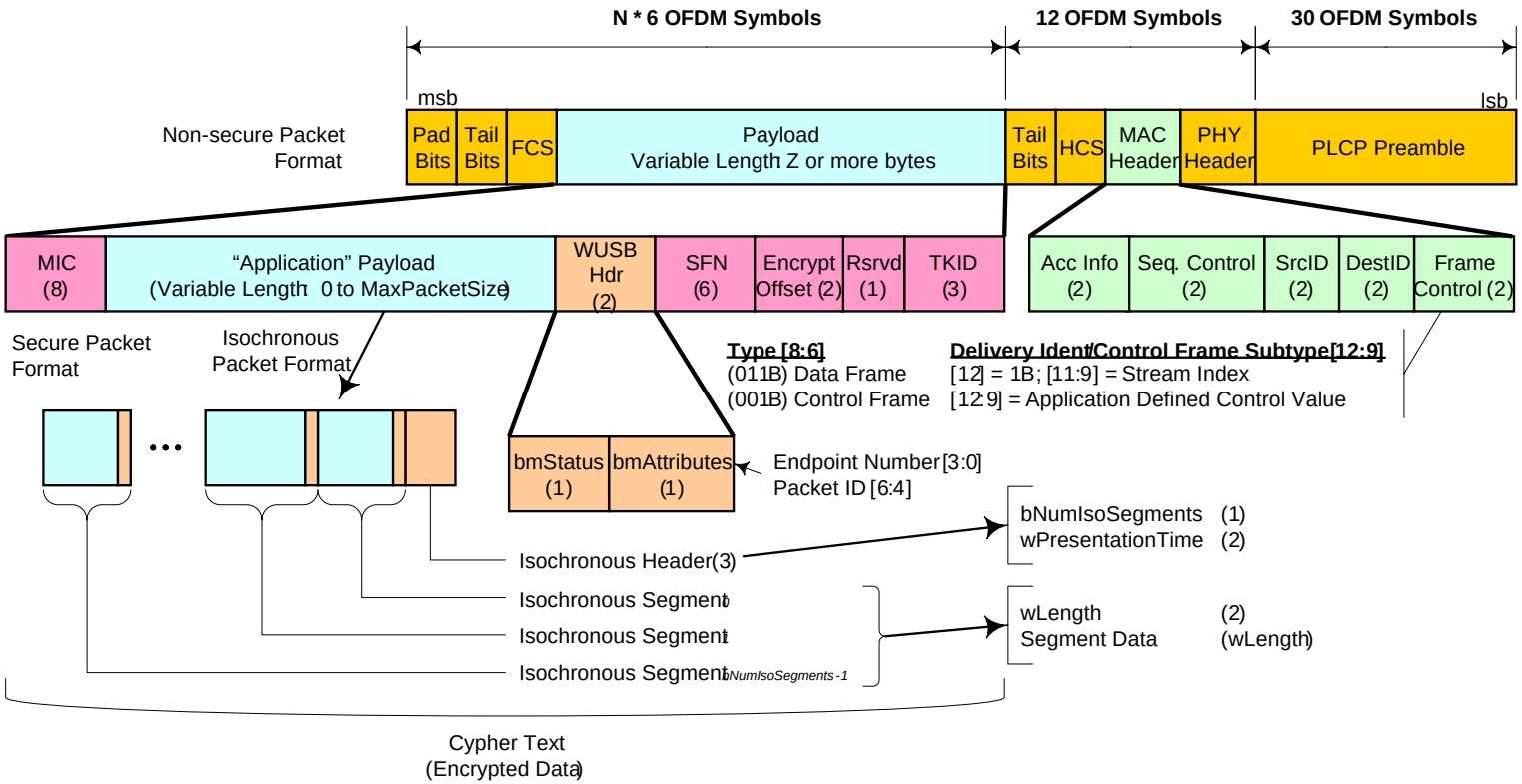
Wireless USB Channel

Review



- The Wireless USB Channel is encapsulated by the WiMedia ‘channel’
- Wireless USB channel is continuous sequence of linked control packets transmitted by the Host during reserved WiMedia channel time
  - WUSB time is reserved from WiMedia channel time (DRPs)
  - Called MMCs – Micro-schedule Management Commands
- All Wireless USB Data communications are over the Wireless USB Channel

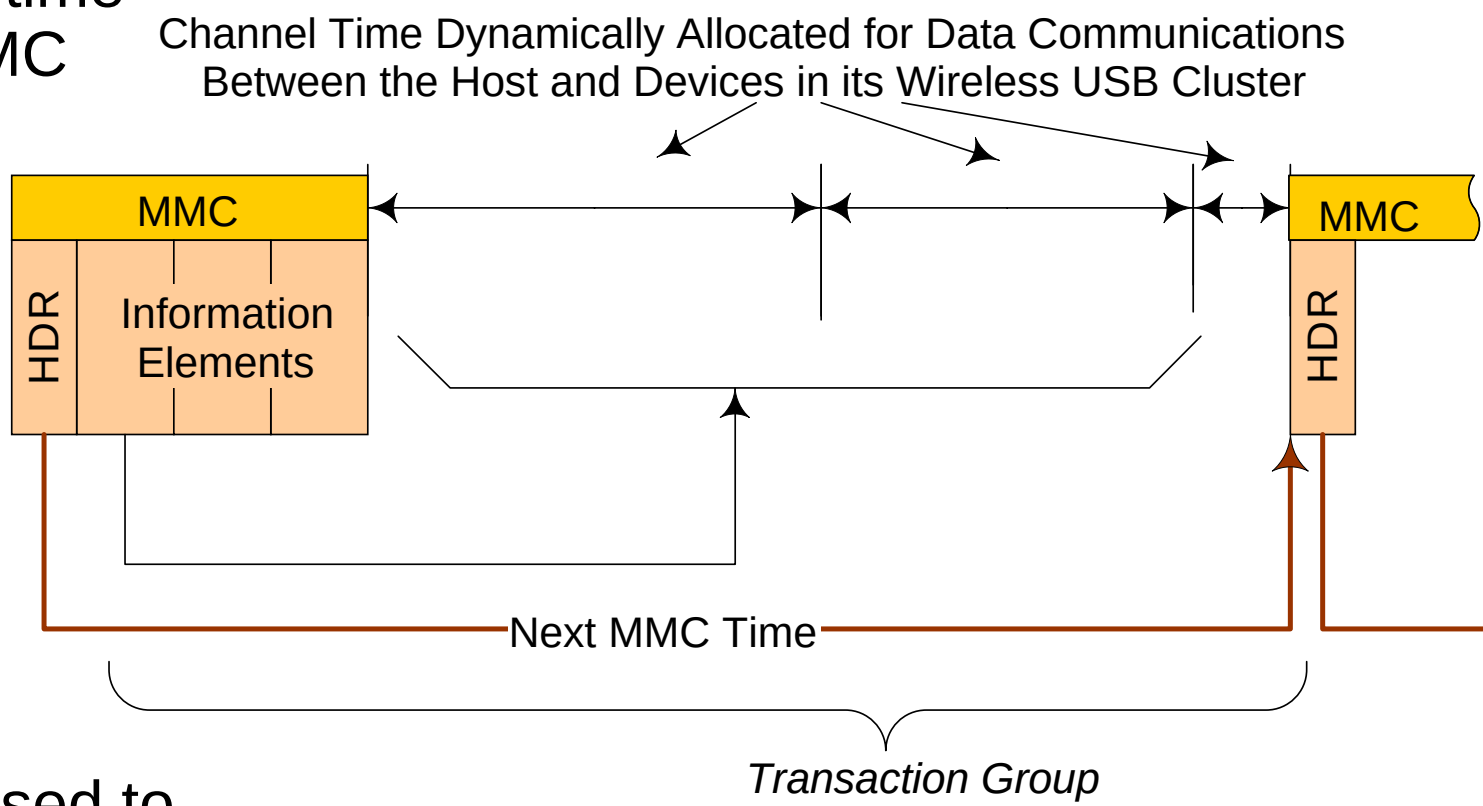
Packet Layout



- WiMedia MAC Packet format
- Utilizes Secure 'frame' format

Wireless USB Channel

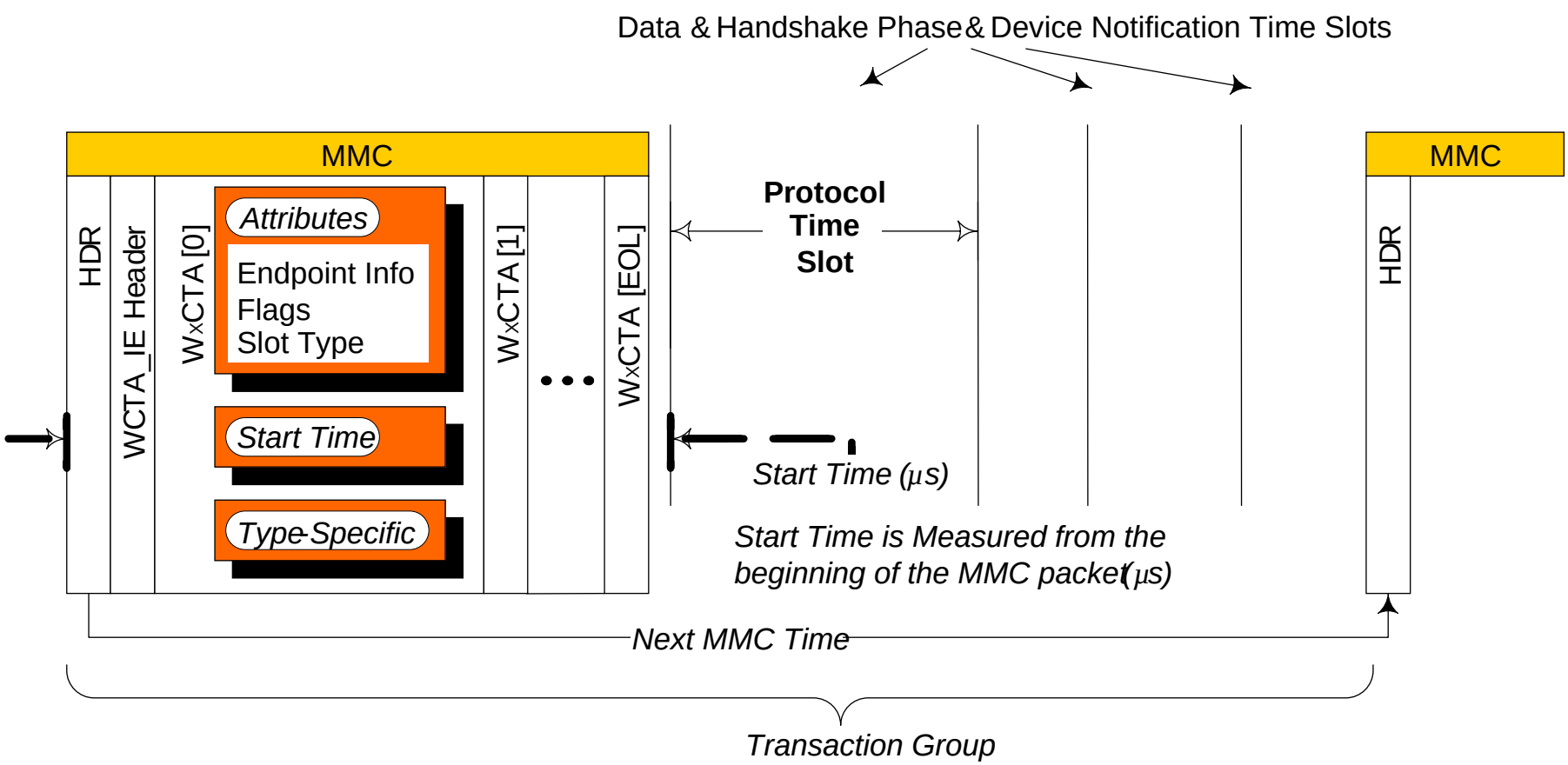
- MMC plus time till next MMC



- Basic structure used to maintain Wireless USB Channel

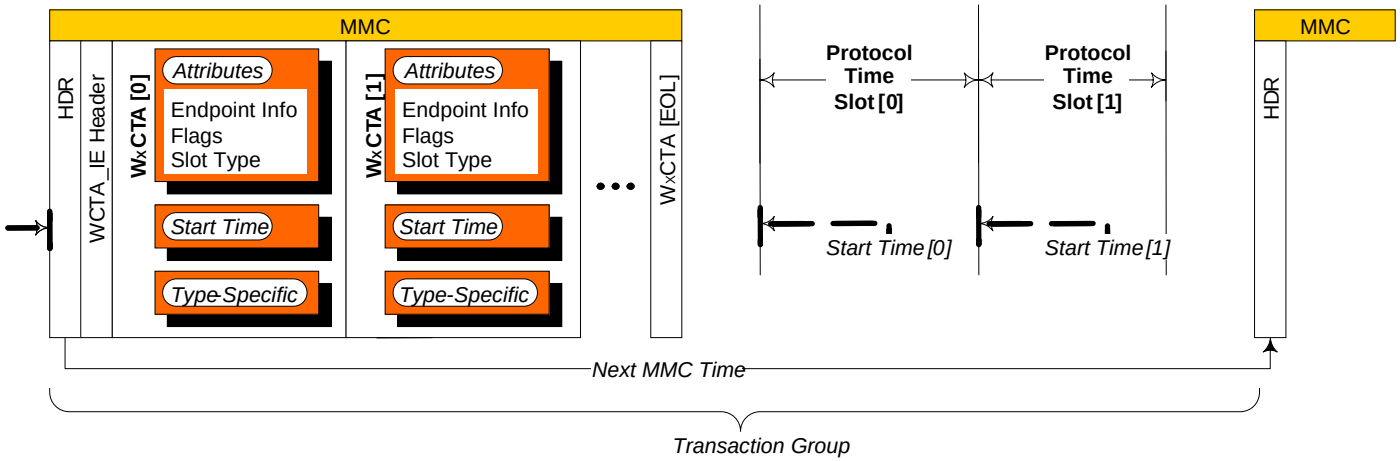
Transaction Group

Channel Time Allocations [WCTA_IE]



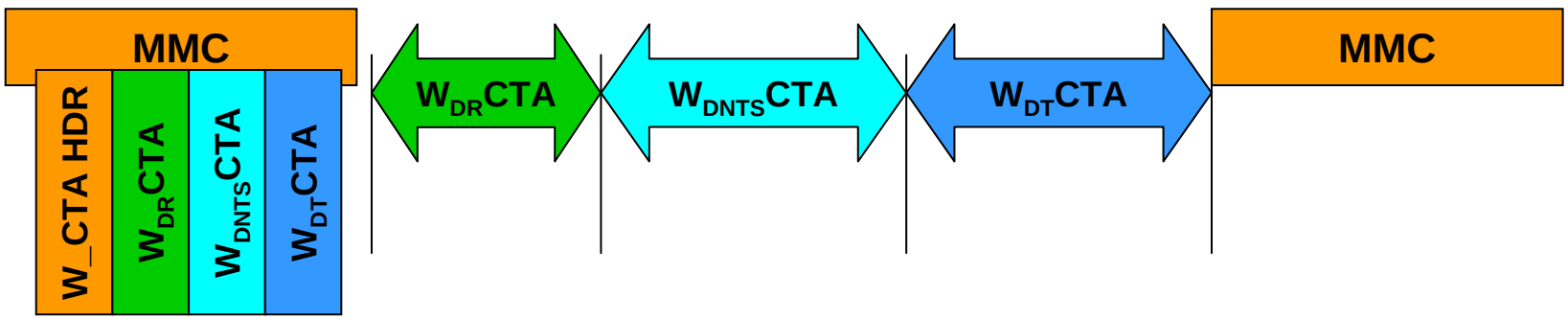
Transaction Group
Slot Durations

- Slot time durations are derived from adjacent W_XCTA blocks
  - OUTs device know when they can stop listening
  - INs devices transmit what was requested
- EOL W_XCTA is always required to be present



Time Slot Allocations

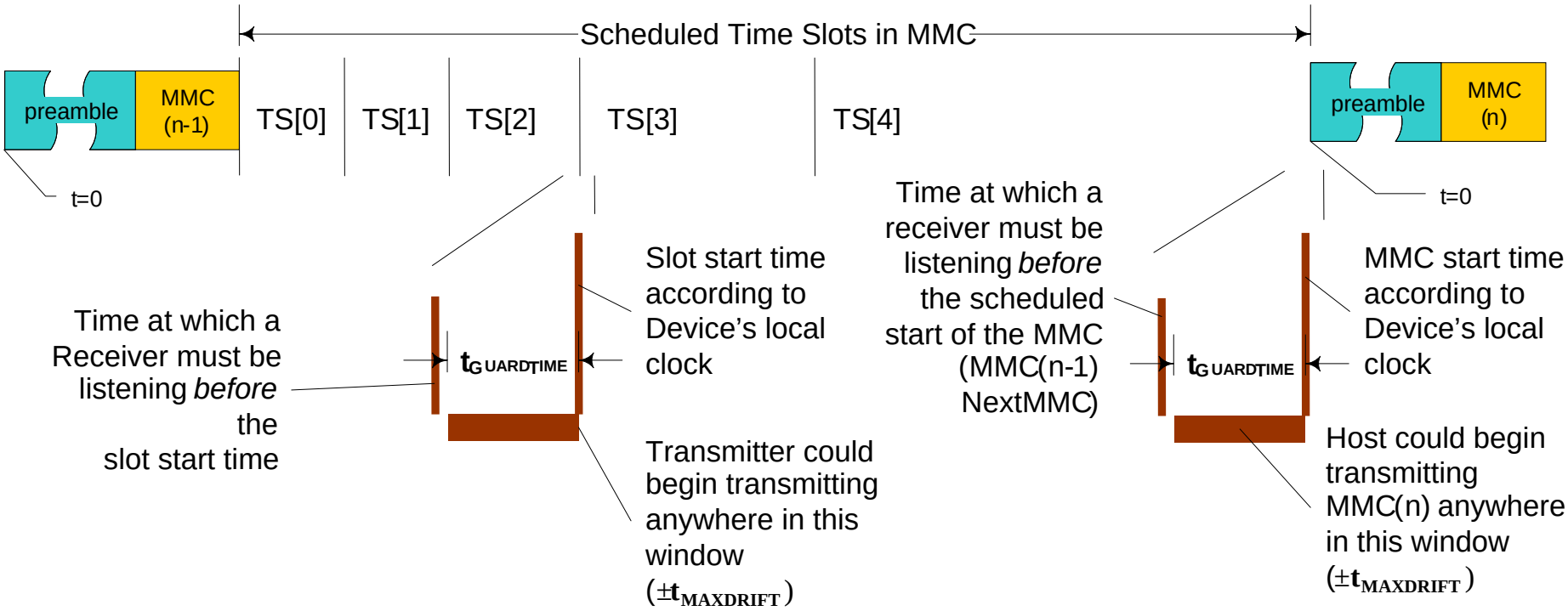
- Three types of time slots:
  - W_{DR}CTA (Device Receive)
  - W_{DT}CTA (Device Transmit)
  - W_{DNTS}CTA (Device Notification) time slot



Time Slot Allocations [cont.]

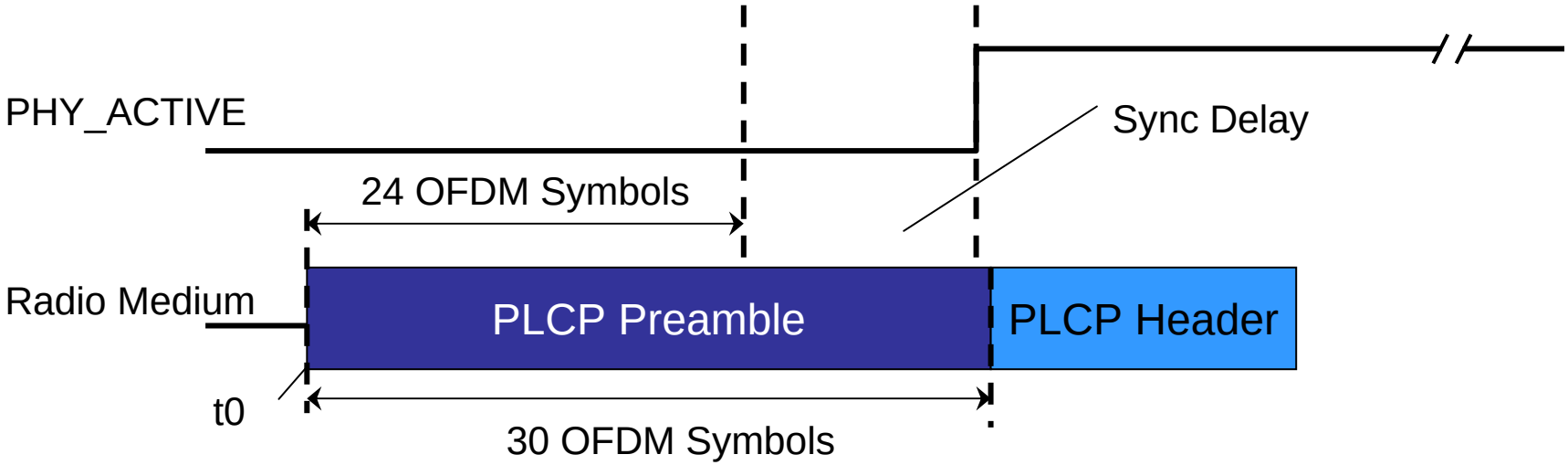
- W_{DR}CTA, W_{DT}CTAs are used for Transaction Protocol
  - Control, Bulk, Interrupt and Isochronous data streams
  - Access is : guaranteed time slot
  - Schedule based on client application data flow needs
- W_{DNTS}CTAs used to emulate USB signaling events
  - Connect, disconnect, etc.
  - Access is : contention based - Slotted Aloha
  - Host schedules sufficient number of slots to provide adequate service for the cluster

Transaction Group
Protocol Synchronization



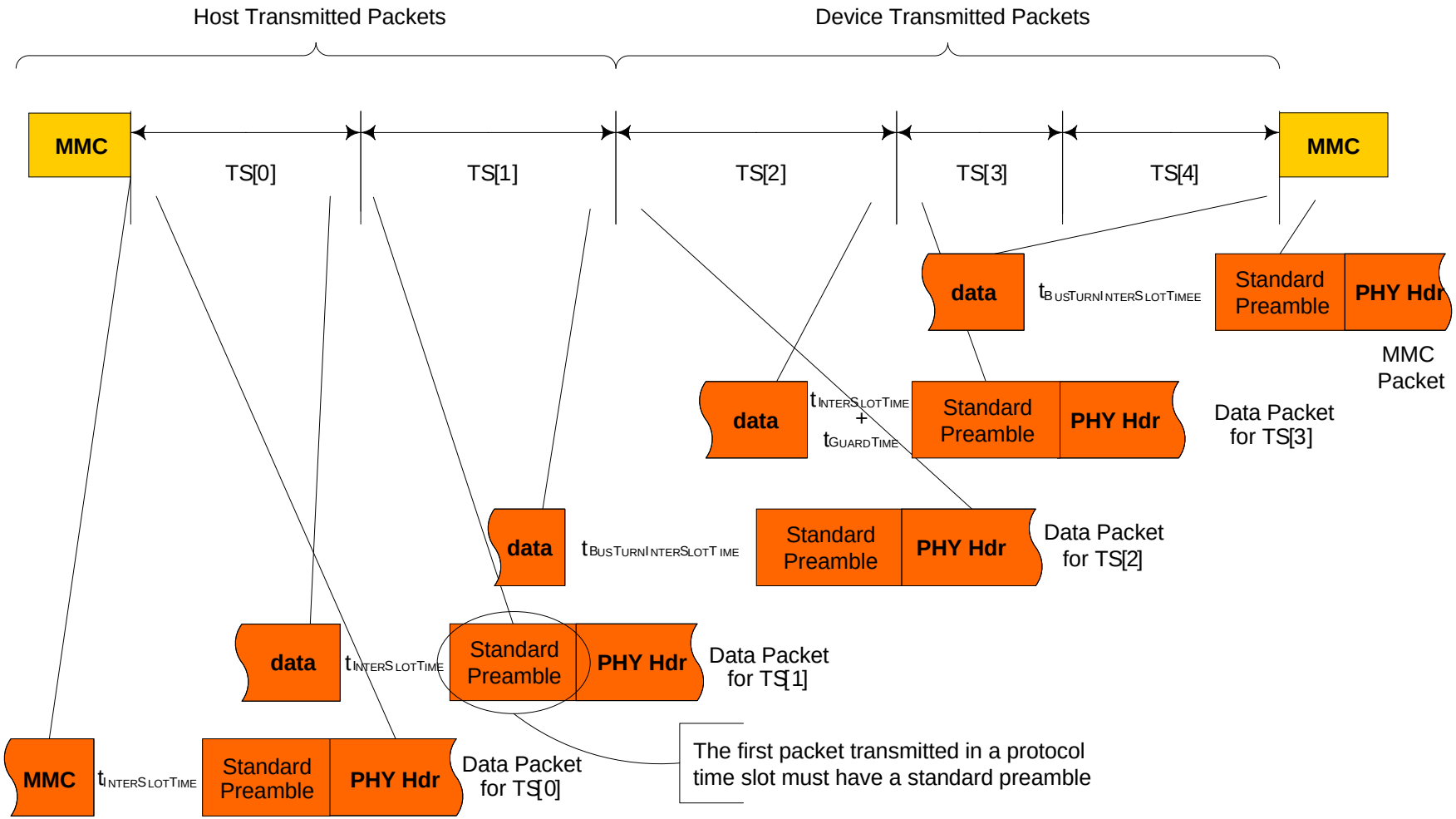
Transaction Group

Protocol Synchronization (cont.)



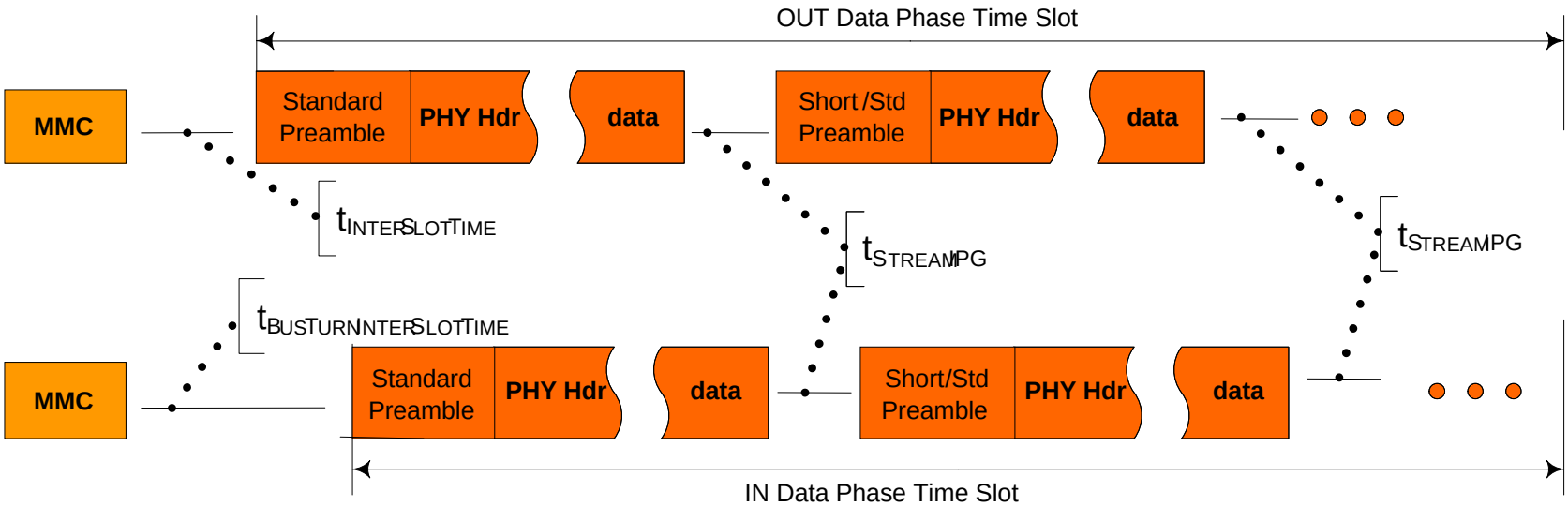
- t0 is the time of the first symbol of the MMC's preamble
- Devices can 'calculate' this from the edge of PHY_ACTIVE

Time Slot Scheduling Requirements



Data Burst Timing

- MMC to first Data OUT : MIFS (1.875 µs)
- MMC to first Data IN : Bus Turn Time
  - Bus Turn + Guard Time (SIFS + 1µs) (11 µs)
- IPGs between burst packets is fixed (MIFS)
- First packet Preamble must be Standard Length
  - Host determines subsequent preamble to use

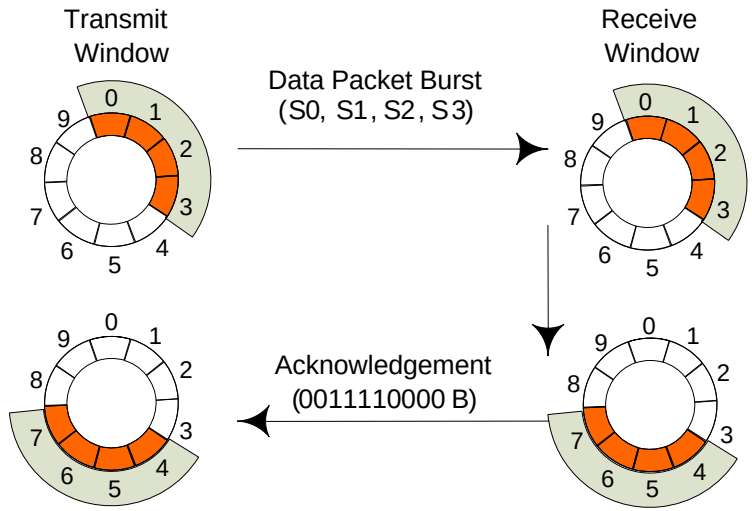


Agenda

- Protocol Components
- Data Transfers (general model)
- Device Notifications
- Flow Control

Data Stream Synchronization
Basic Model

- Protocol-level support for reliable data delivery
- Identifies:
  - Data ordering requirements
  - Guarantees advancement of data stream only after reliable data delivery
- Terms:
  - Maximum Burst Size
  - Maximum Sequence
  - Maximum Sequence Distance



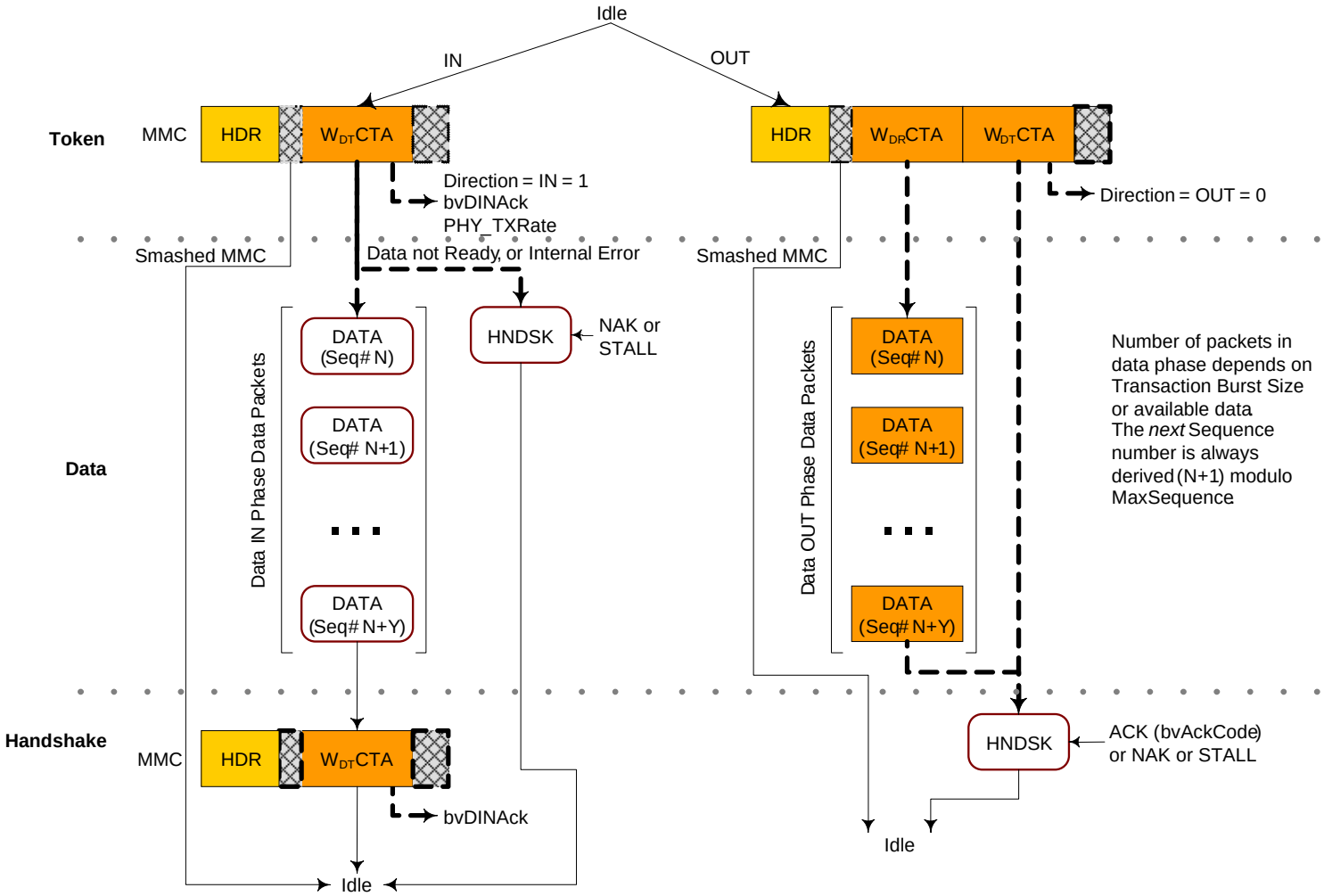
- Transmitter sends data packets associated with Transmit window Sequence numbers
- Receiver acknowledges with new receive window (what is available now)
- Protocol rules for recovering lost packets, avoiding sequence range wrap, etc.
  - See Bursting Session

Transaction Transmit Bit Rate

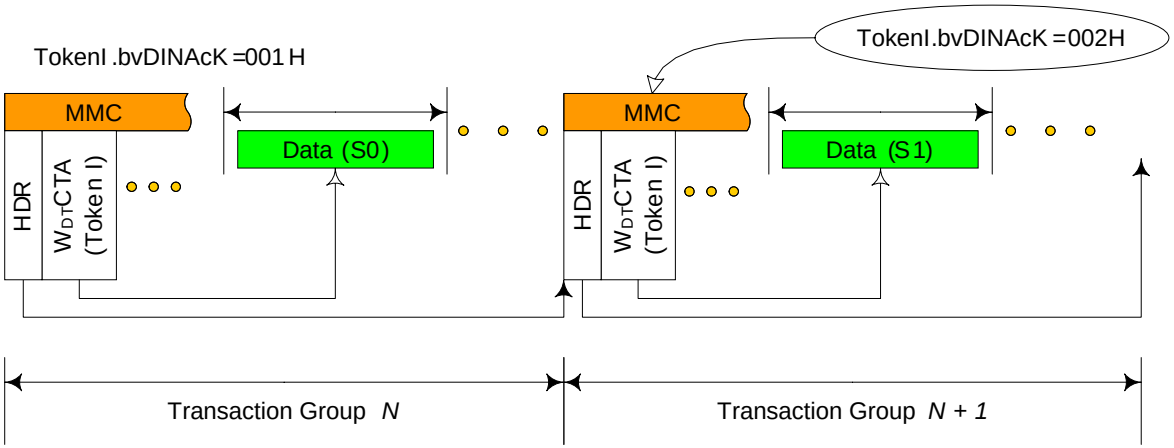
- MMCs are transmitted at ‘base’ signaling rate (53.3 Mb/s)
- Host dictates transmit bit rate for data and handshake phase transmissions
  - Transmit bit rate applies to entire phase
  - OUTs : host uses only TBRs supported by device (assumes Tx/Rx symmetry)
  - INs : WDTCTA.PHY_TXRate directs which TBR device must use during protocol time slot
- Handshakes:
  - Host must use ‘base’ signaling rate for a handshake time slot (OUTs)
  - Host must ensure a data phase time slot is always large enough to accommodate a handshake transmitted at ‘base’ signaling rate

Transaction Format

General Model



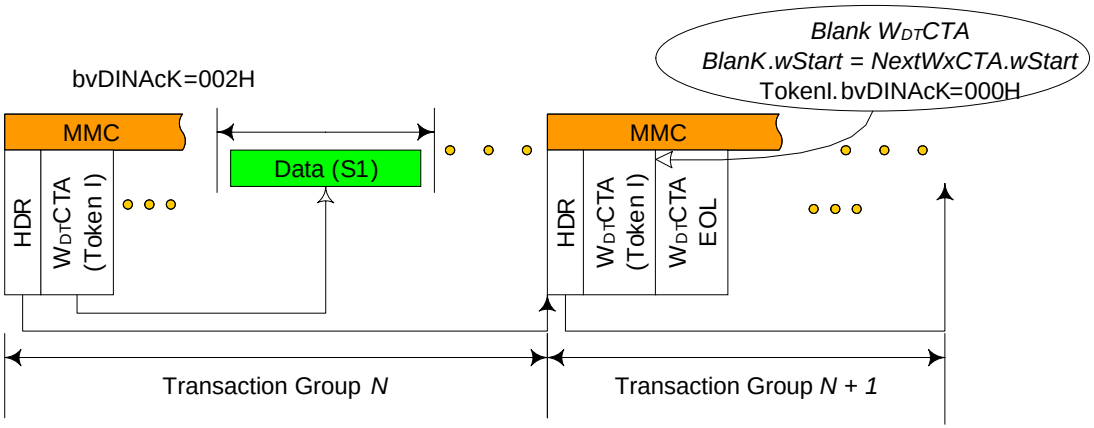
Data IN Transfer
Streaming Example



		Transaction 1			
Transaction 0				Transaction 2 …	
W_{DT}CTA (IN) bvDINAck = 001H	DATA(S0) (Use Data)	W_{DT}CTA (IN) bvDINAck = 002H	DATA(S1) (Use Data)	W_{DT}CTA (IN) bvDINAck = 001H	…
MMC	Device EP TX	MMC	Device EP TX	Host TX (MMC)	

Data IN Transfer

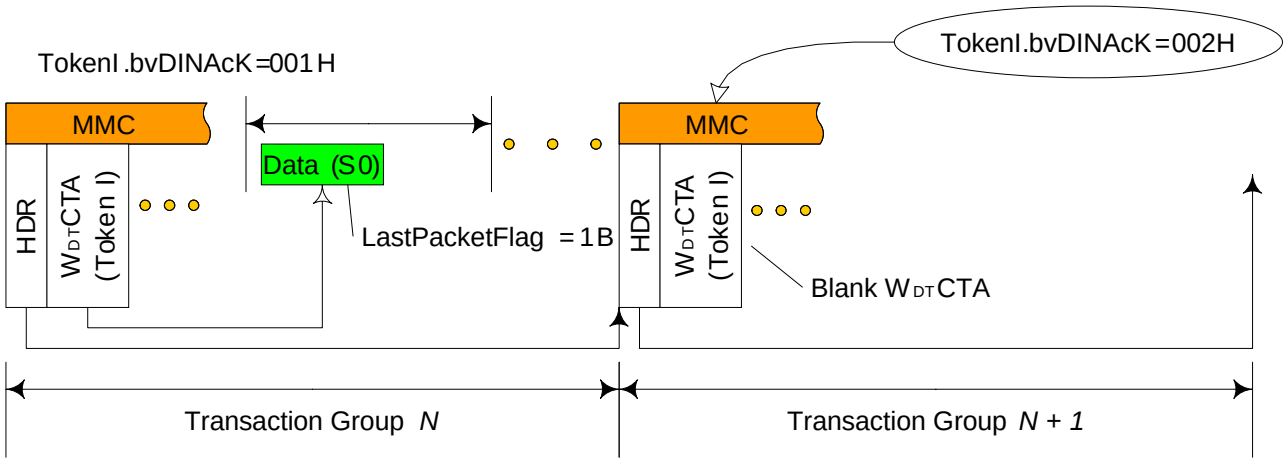
End Of Transfer Example



	Transaction N	
Transaction N-1		
W_{DT}CTA (IN) bvDINAck = 002H	DATA(S1) (Use Data)	**Blank W_{DT}CTA (IN)** bvDINAck = 000H wStart = NextWxCTA.wStart End of List WxCTA (EOL)	…
MMC	Device EP TX	MMC	

Data IN Transfer

Short Packet Example

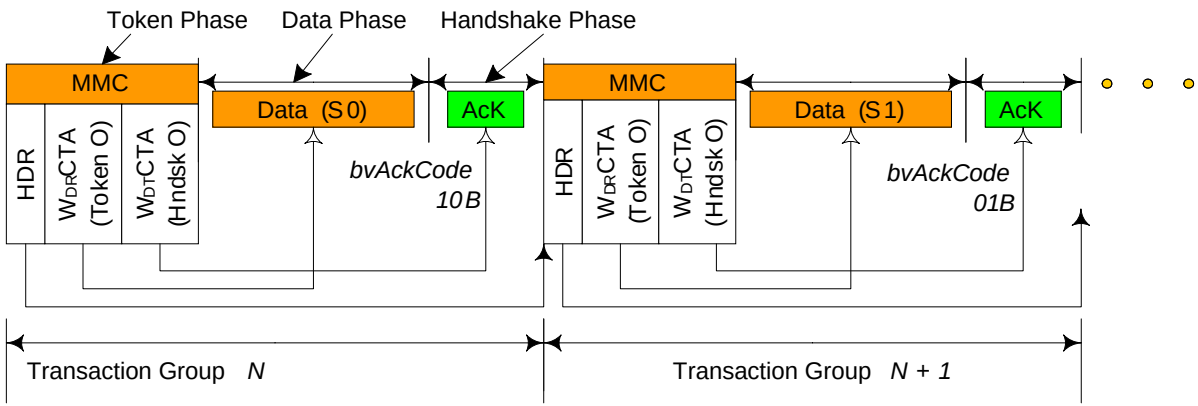


Transaction 0		Transaction 1	
W_{DT}CTA (IN) bvDINAck = 001H	DATA(S0) LastPacketFlag = 1 (Use Data)	**W_{DT}CTA (IN)** bvDINAck = 002H	…
MMC	Device EP TX	MMC	

Host may commence next transfer immediately (if buffer available)

Data OUT Transfers

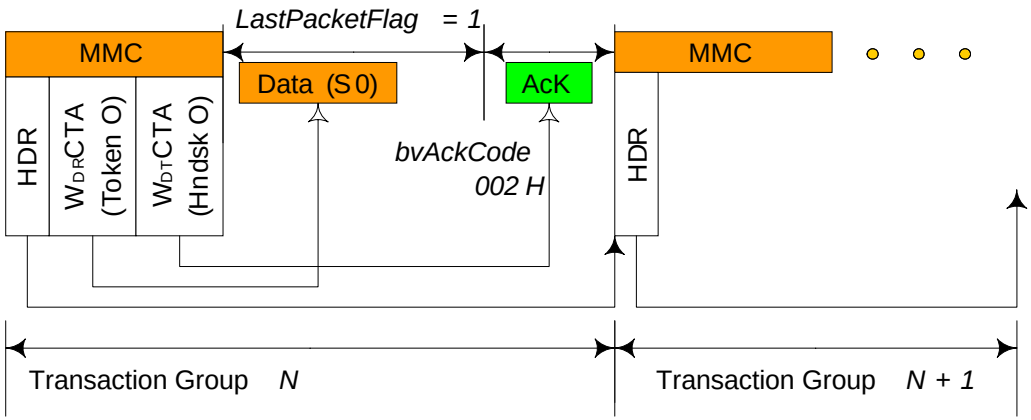
Streaming Example



			Transaction 1		
Transaction 0					
W_{DR}CTA (OUT) W_{DT}CTA (Handshake)	DATA(0) (Use Data)	HNDSHK bvAckCode = 02H	W_{DR}CTA (OUT) W_{DT}CTA (Handshake)	DATA(1) (Use Data)	HNDSHK bvAckCode = 01H
Host TX (MMC)	Host TX	Device EP TX	Host TX (MMC)	Host TX	Device EP TX

Data OUT Transfers

Short Packet Example

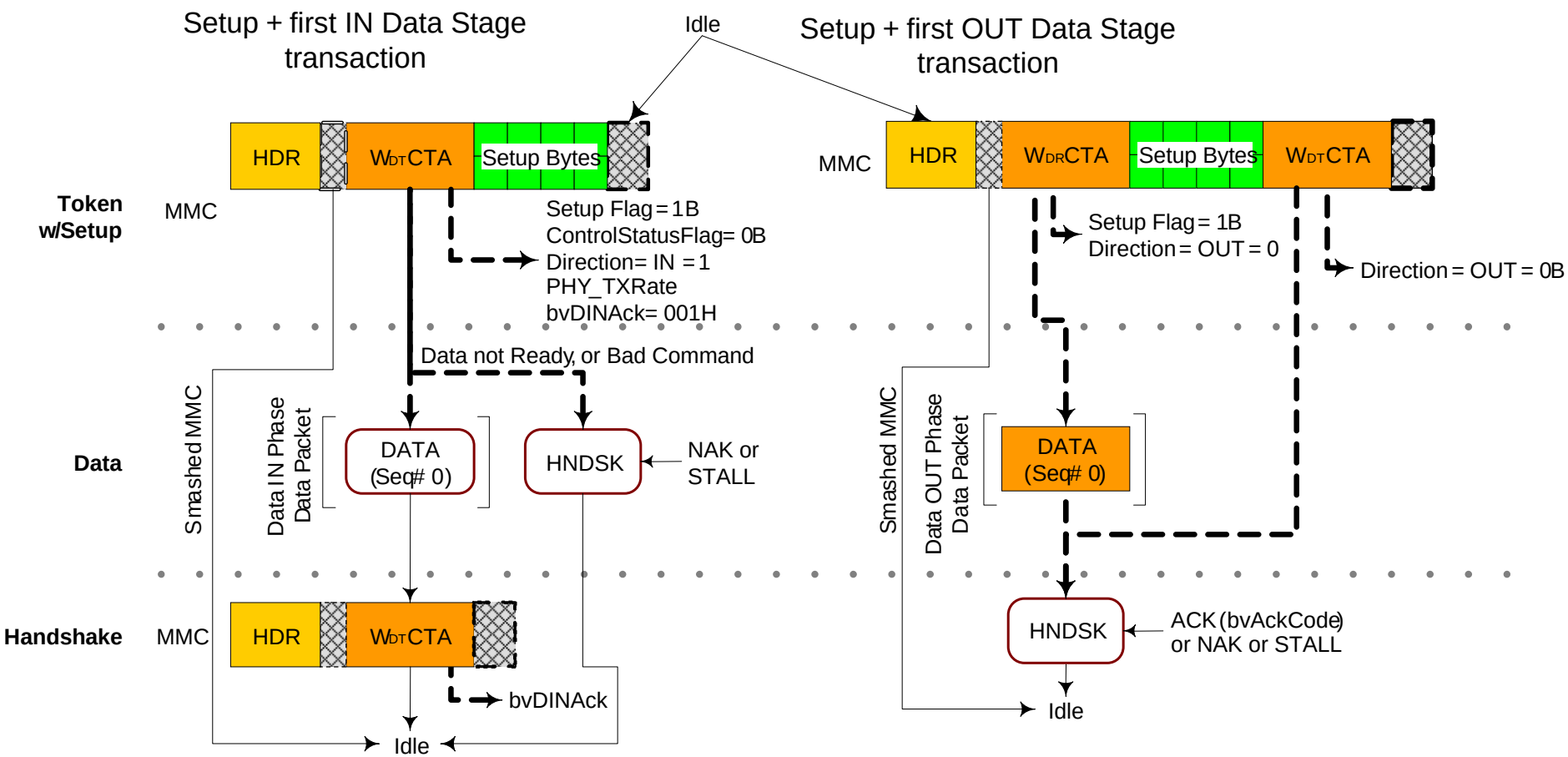


			Transaction 1
Transaction 0			
W_{DR}CTA (OUT) W_{DT}CTA (Handshake)	DATA(0) LastPacketFlag = 1 (Use Data)	HNDSHK bvAckCode = 02H	
Host TX (MMC)	Host TX	Device EP TX	Host TX (MMC)

Host may commence next transfer immediately (if data available)

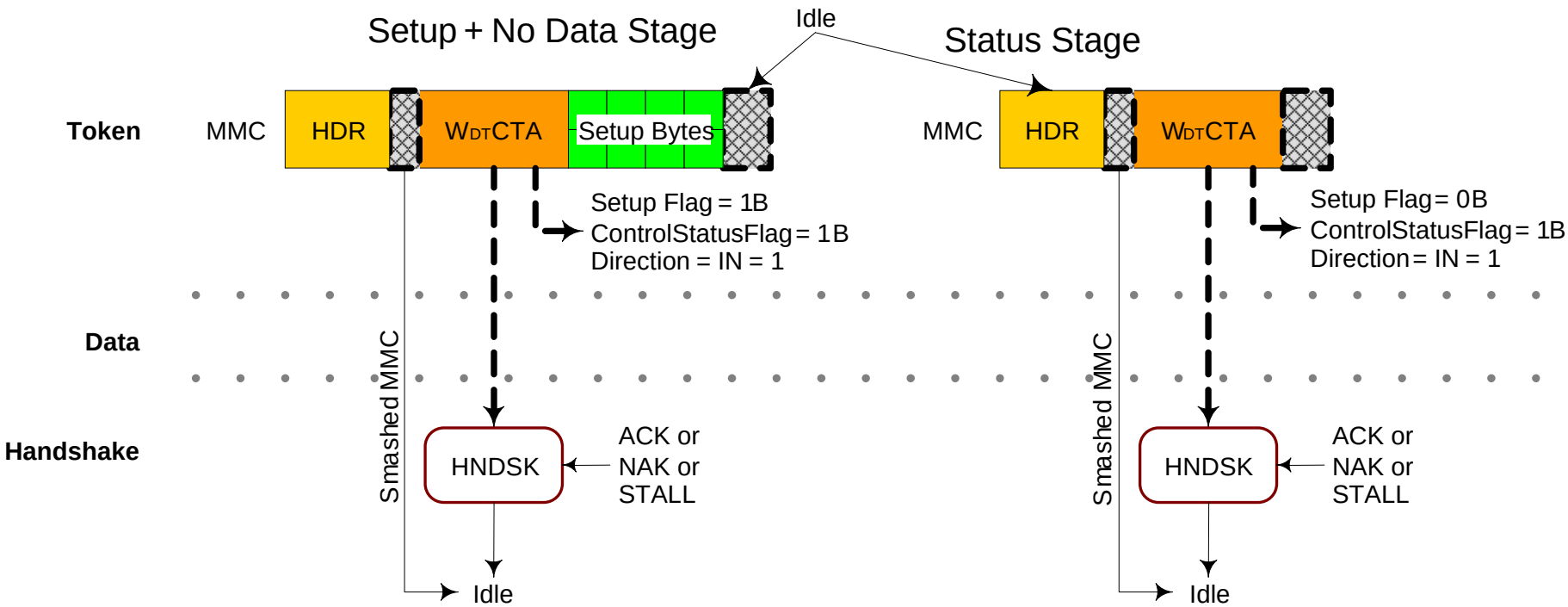
Control Transaction Format

Setup Stage w/Data Stage



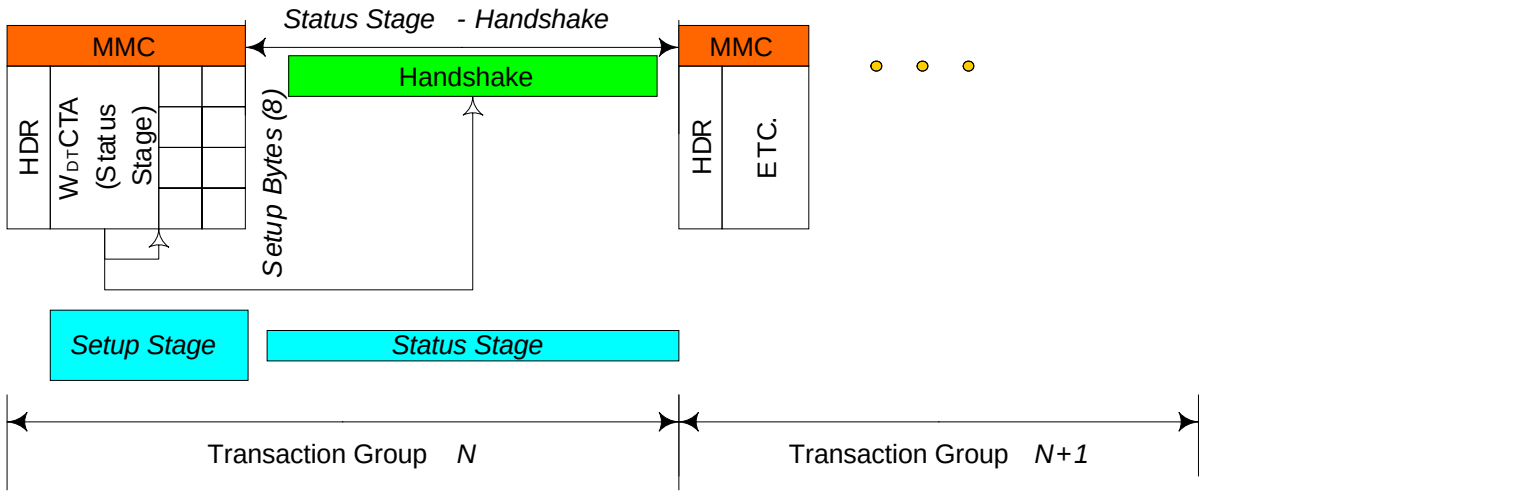
Control Transaction Format

Setup w/No Data Stage; Bare Status Stage



Control Transfer

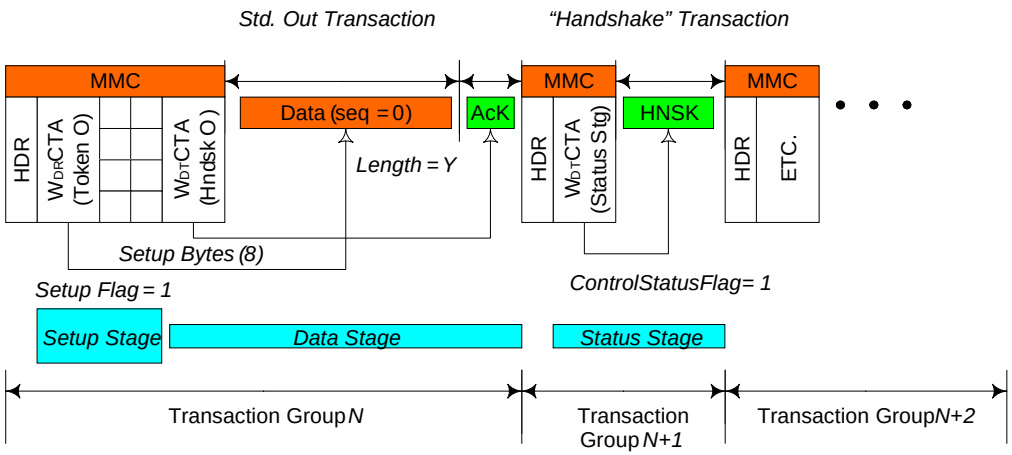
Example (Set Address - No Data Stage)



Transaction 0		…
W_{DT}CTA SetupFlag = 1B Ctrl/StatusFlg = 1B (Handshake) Setup Bytes (8) – set address	HNDSHK Hndshk code = ACK bvAckCode = 00H Direction = IN	
Host TX (MMC)	Device EP TX	Host TX (MMC)
Setup Stage	Status Stage	

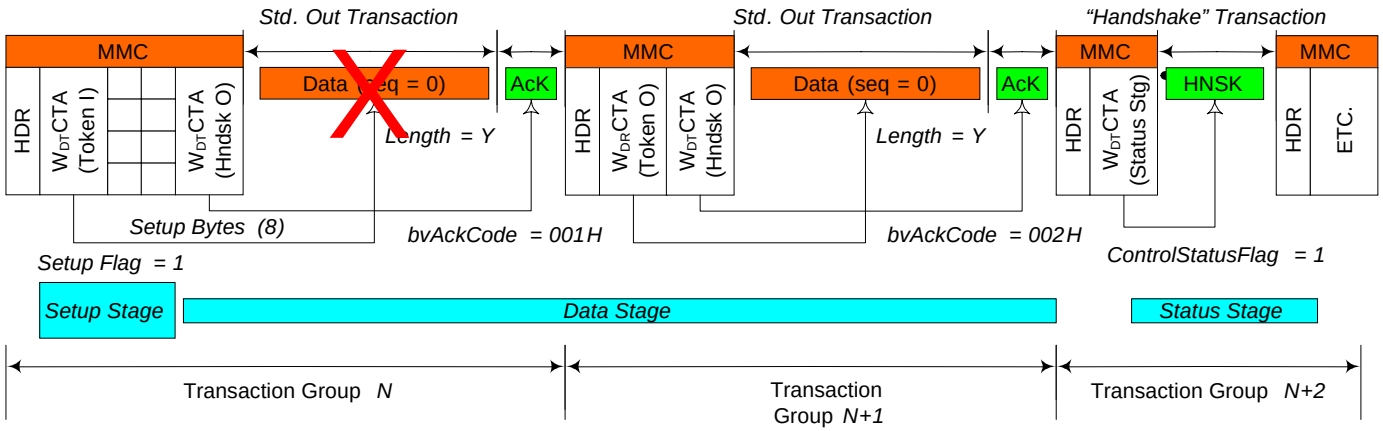
Control Transfer

Example (Control Write - SetKey)



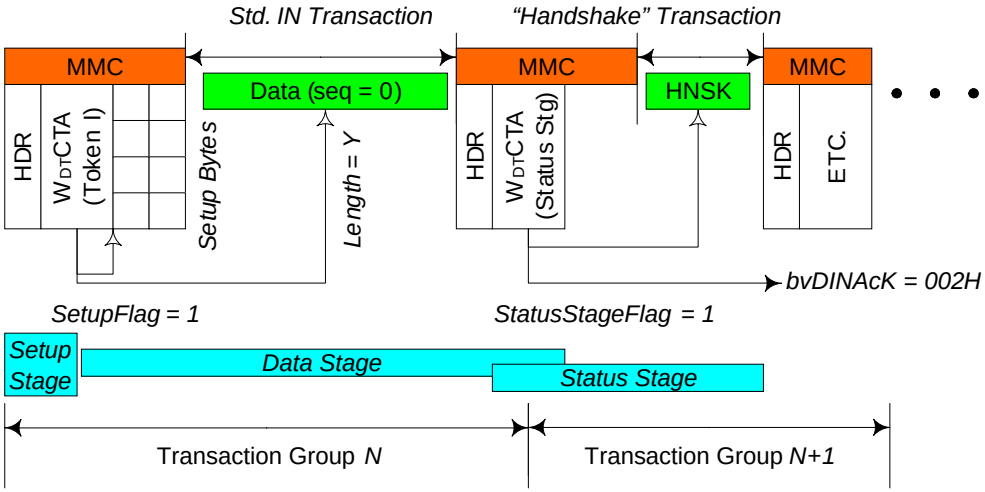
Transaction 0			Transaction 1		
W_{DR}CTA SetupFlag = 1B Ctrl/StatusFlag = 0B (Handshake) Setup Bytes (8) – set Key W_{DT}CTA - handshake	Data Out Seq = 0B (Set Key)	HNDSHK Hndshk code = ACK bvAckCode = 02H	W_{DT}CTA SetupFlag = 0B Ctrl/StatusFlag = 1B (Handshake)	HNDSHK Hndshk code = ACK bvAckCode = 00H Direction = IN	
Host TX (MMC)	Host TX	Device EP TX	Host TX (MMC)	Device EP TX	
Setup Stage	Data Stage		Status Stage		

Control Transfer
(Control Write – SetKey lost data packet)



Transaction 0			Transaction 1		
W_{DR}CTA SetupFlag = 1B Ctrl/StatusFlag = 0B (Handshake) Setup Bytes (8) – set Key W_{DT}CTA - handshake	**Data Out Seq = 0B (Set Key)**	HNDSHK Hndshk code = ACK bvAckCode = 01H	W_{DR}CTA SetupFlag = 0B Ctrl/StatusFlag = 0B W_{DT}CTA - handshake	Data Out Seq = 0B (Set Key	HNDSHK Hndshk code = ACK bvAckCode = 02H
Host TX (MMC)	Host TX	Device EP TX	Host TX (MMC)	Host TX	Device EP TX
Setup Stage	Data Stage				

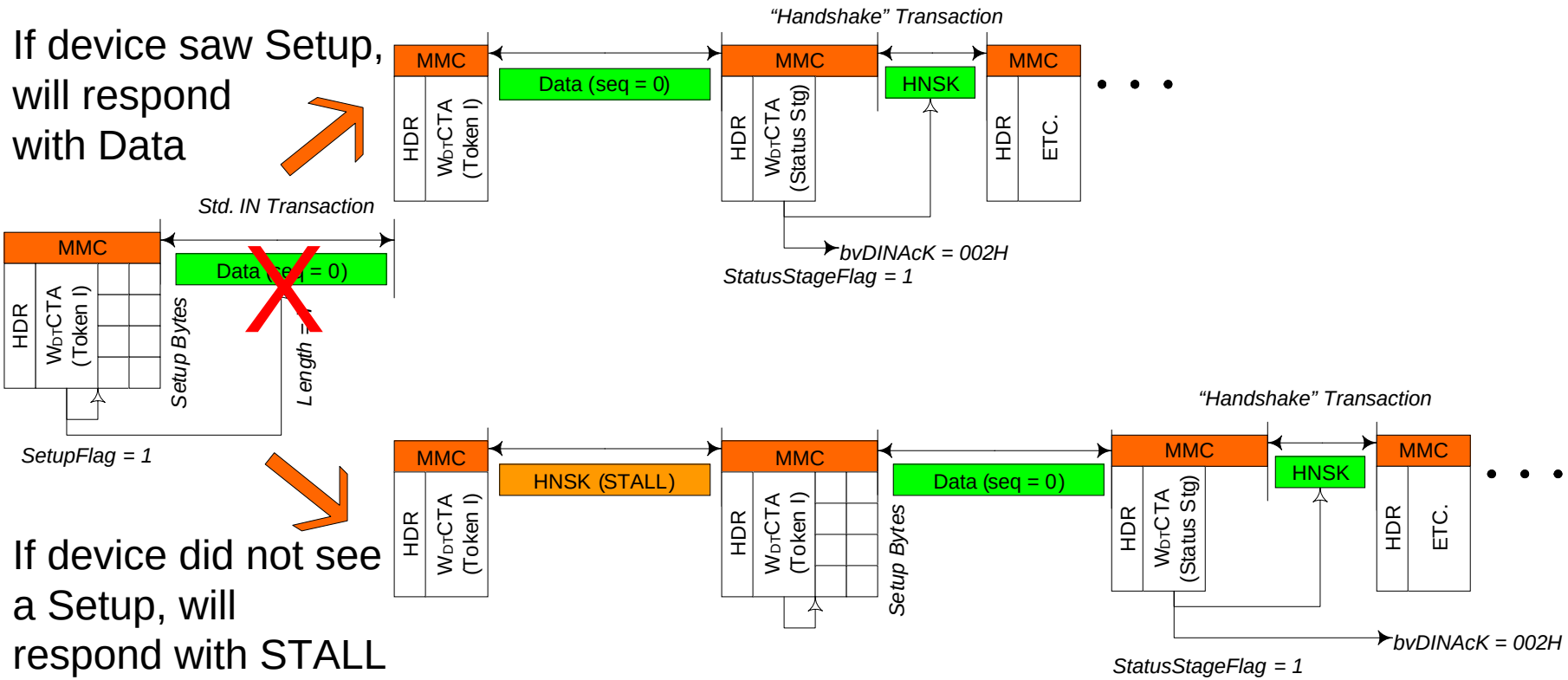
Control Transfer
Example (Control Read – Get Descriptor)



Transaction 0		Transaction 1		
W_{DT}CTA SetupFlag = 1B Ctrl/StatusFlag = 0B bvDINAck = 001H Setup Bytes (8) Get Descriptor	Data IN Seq = 1B (Get Descriptor Data)	W_{DT}CTA SetupFlag = 0B Ctrl/StatusFlag = 1B bvDINAck = 002H (Handshake)	HNDSHK Hndshk code = ACK bvAckCode = 00H Direction = IN	
Host TX (MMC)	Device EP TX	Host TX (MMC)	Device EP TX	
Setup Stage	Data Stage			
		Status Stage		

Control Transfer
Example (Control Read – Get Descriptor)

- Data packet gets smashed
- Host will retry Data Stage



Agenda

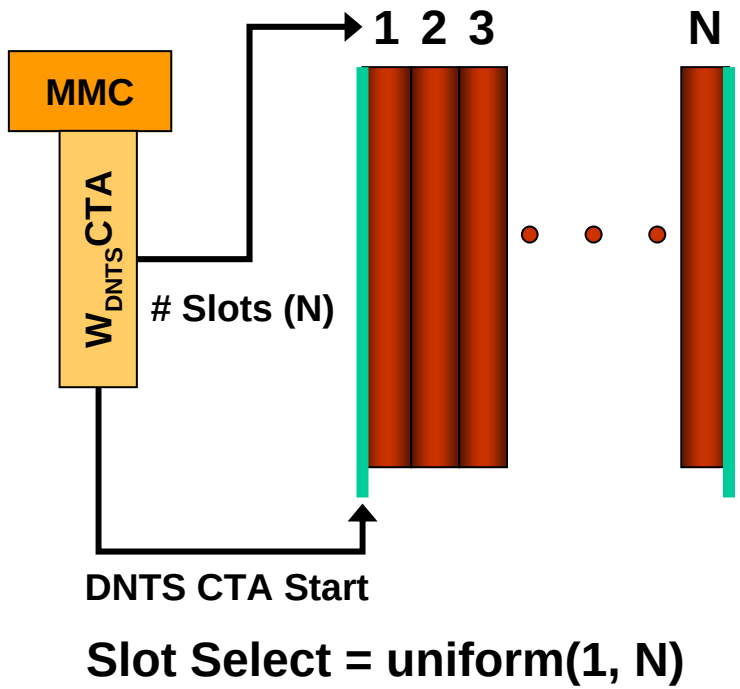
- Packet Layouts
- Transaction Groups
- Timing Constraints
- Data Transfers (general model)
- Device Notifications
- Flow Control

Device Notifications Time Slots
DNTS

- Devices are only allowed transmitters
- Device notification messages (to the WUSB Host) are only allowed data communications
- Nature of notification messages are that they are asynchronous and infrequent
- Fixed Maximum Size for notification messages
  - Take advantage of this in the access method
- Host policy to manage ‘efficiently’
  - Unused time is just lost to WUSB data stream

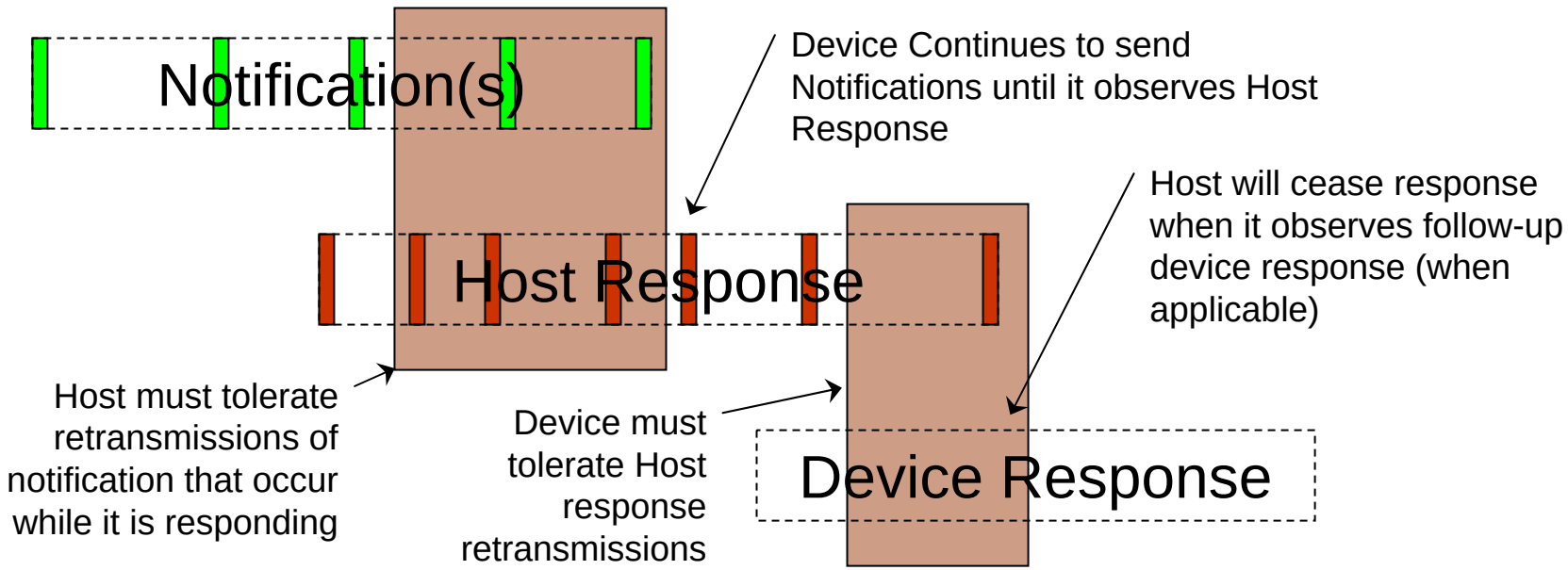
DNTS Access Method
Contention Based : Slotted Aloha

- Fixed (maximum) sized message
  - DNTS is a window of uniform sized message slots
- W_{DNTS}CTA indicates number of message slots
- Device selects a message slot using a uniformly distributed random integer value (in range 1 to N)
- Device transmits message a message slot start time



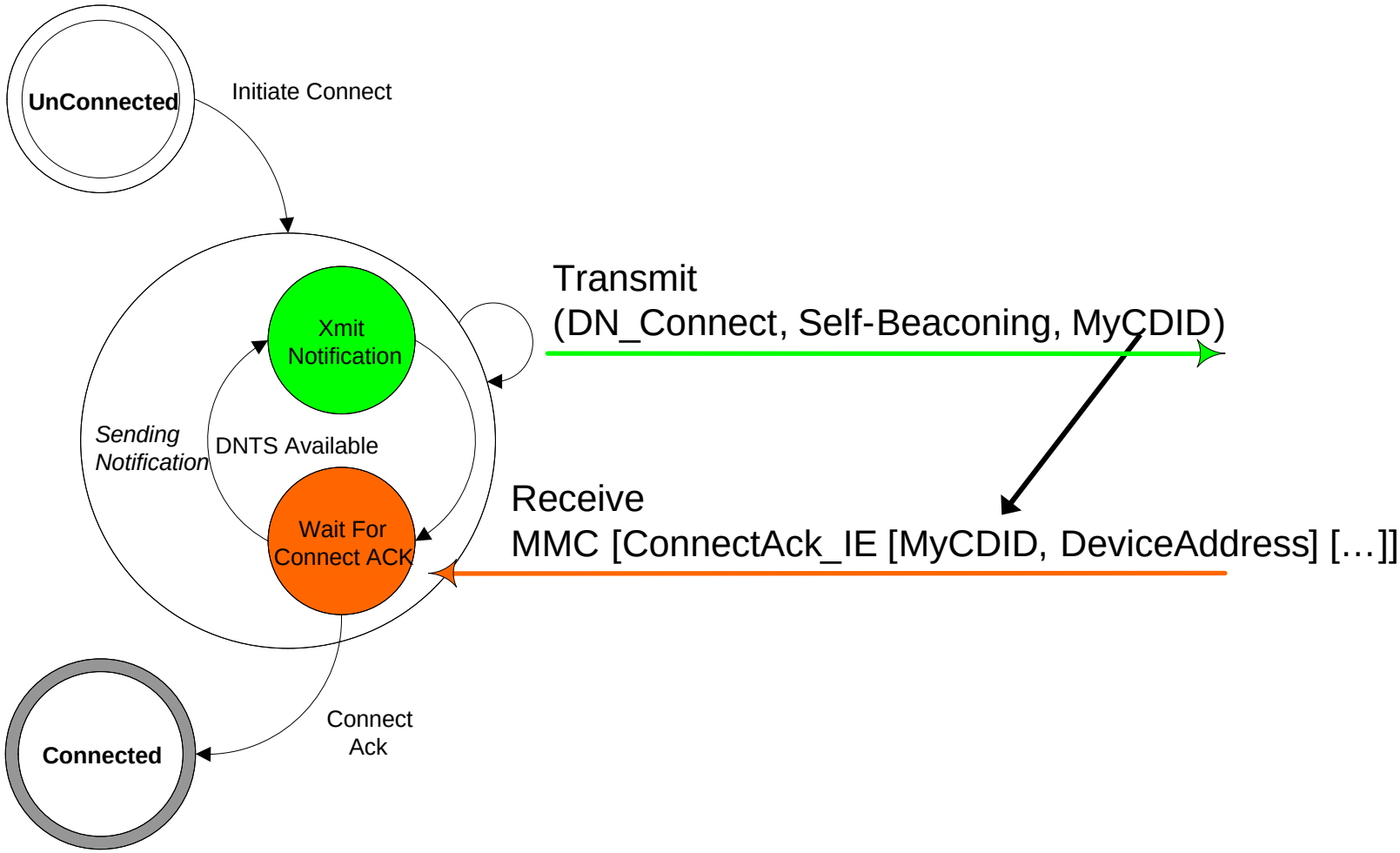
Device Notifications
Reliability

- Protocol for Device Notifications is designed for unreliability:



Device Notification

Example (Device Perspective) - Connect



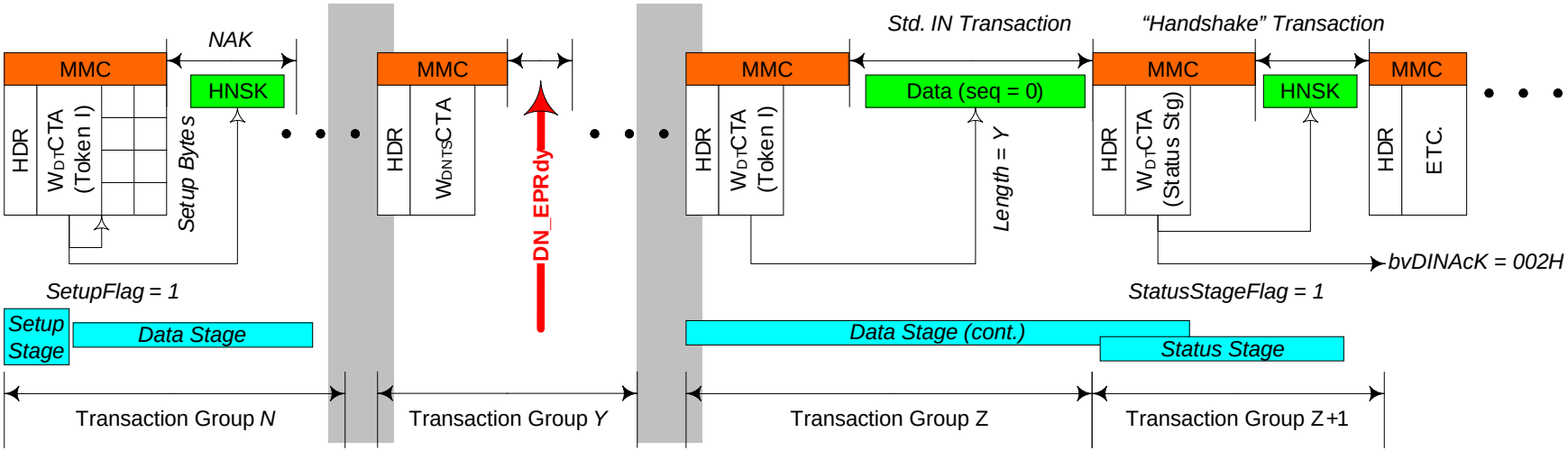
Agenda

- Packet Layouts
- Transaction Groups
- Timing Constraints
- Data Transfers (general model)
- Device Notifications
- Flow Control

Flow Control

- Used on All transfer types
- Device provides flow-control information in it's response to the host
  - *bmStatus* field in the WUSB Header of a Data, IData or Handshake
  - OUTs, the flow control response is in the AcK handshake
  - INs, the flow control response is in the Data(x) packet
  - Handshake encoded with NAKs always available
- Flow control status from Endpoint results in removal of endpoint from 'Active' schedule
- Resumption of service to a flow-controlled endpoint depends on transfer type
  - Bulk & Control
    - Device delivers a Endpoint Ready (DN_EPRdy) notification to the host during an DNTS
  - Interrupt & Isochronous
    - Next service interval

Control Transfer
Example (Control Read – Flow Control)



- Host stops polling endpoint on NAK
- Device notifies it is ready with DN_EPRdy
- Host resumes polling endpoint
  - Retries data phase transaction in this example

Track II Announcement