Universal Serial Bus (USB)

- Asynchronous, serial bus for communication
- Rev 1.1 released in 1998

RS-232 vs. USB

Characteristics	RS-232	USB
Speed	20Kbps (115K with some drivers)	1.5Mbps, 12Mbps, 480Mbps
Max. Cable Length (feet)	50-100	16(or upto 96 ft. with 5 hubs)
Data Formats	Sent as ASCII sequence of characters	Well structured as packets
Max. number of devices	2	127
Port Concept	Different data paths	Common data paths, shared across transfer types
Typical use	Modem, mouse, printer etc…	Mouse, keyboard, disk drive, modem

USB system architecture

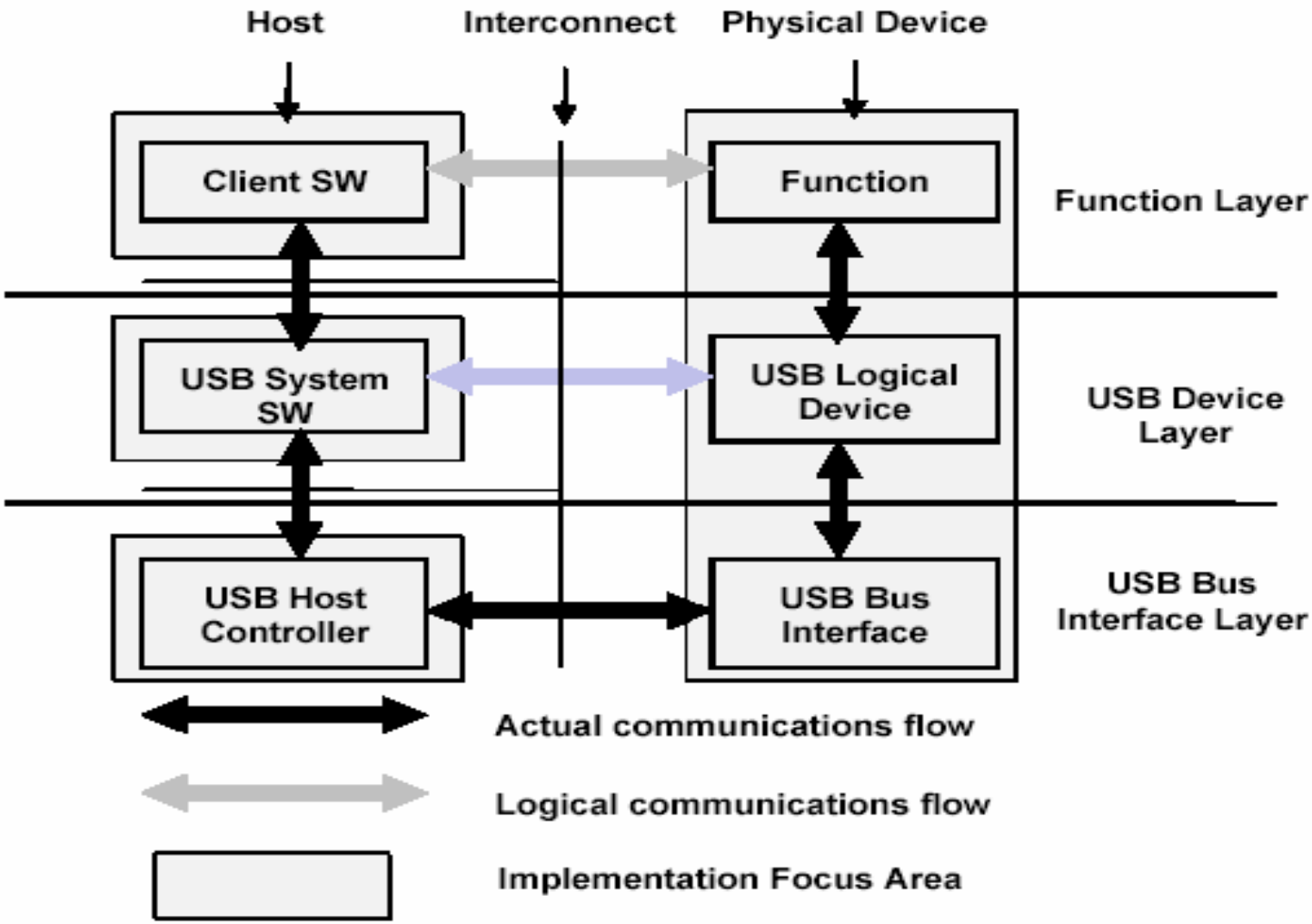


Figure 5-2. USB Implementation Areas

Communication Flow in a USB

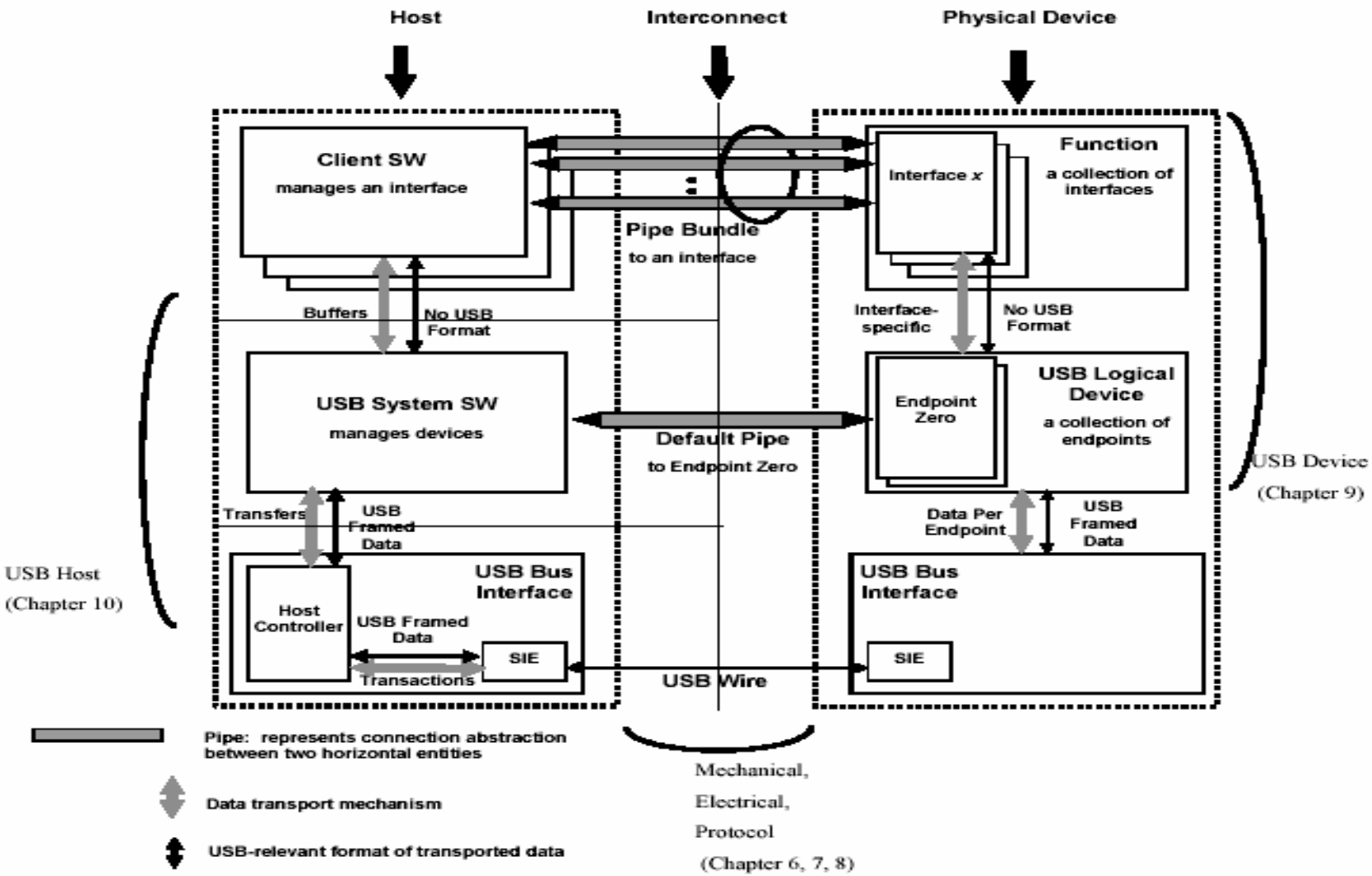
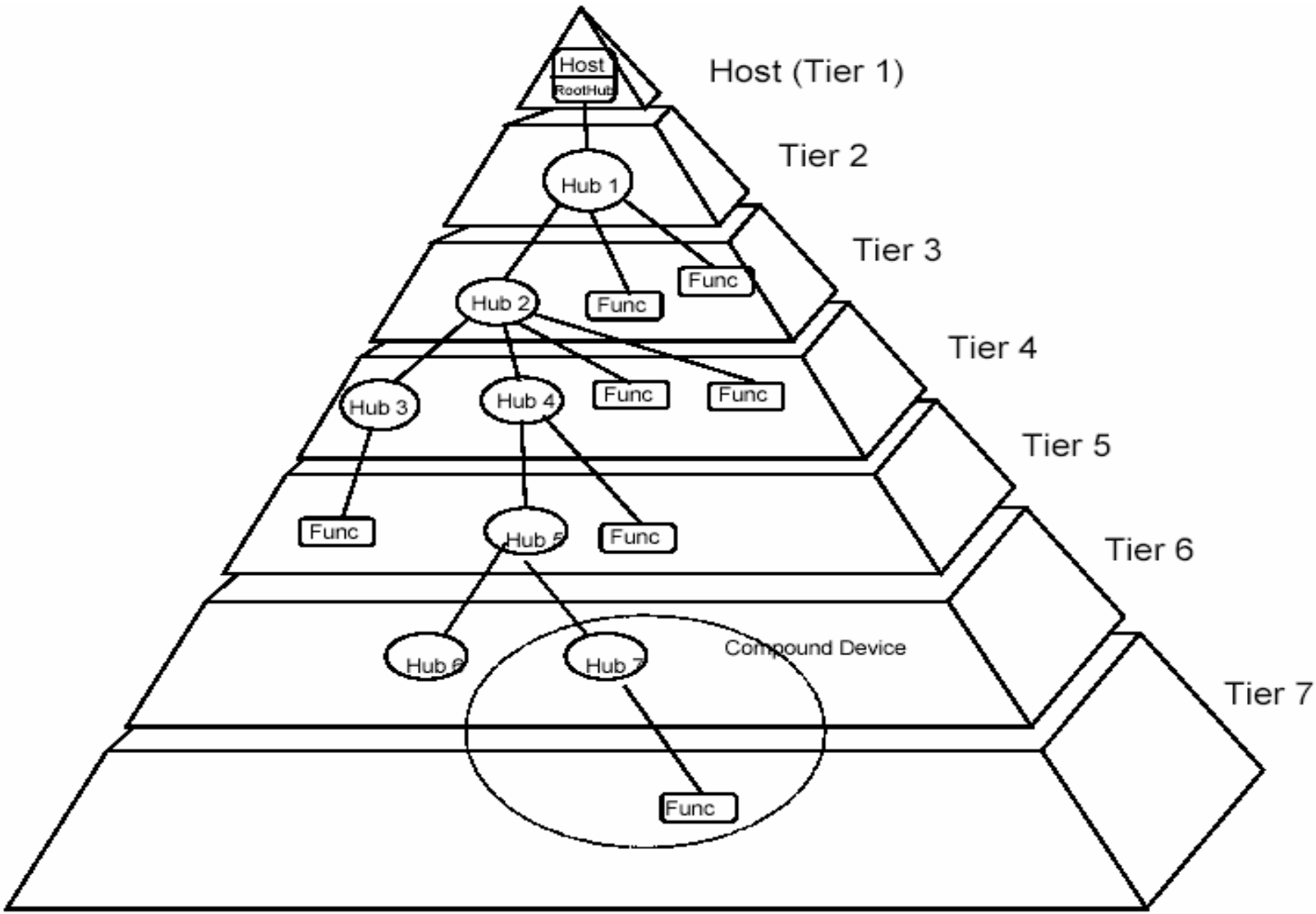
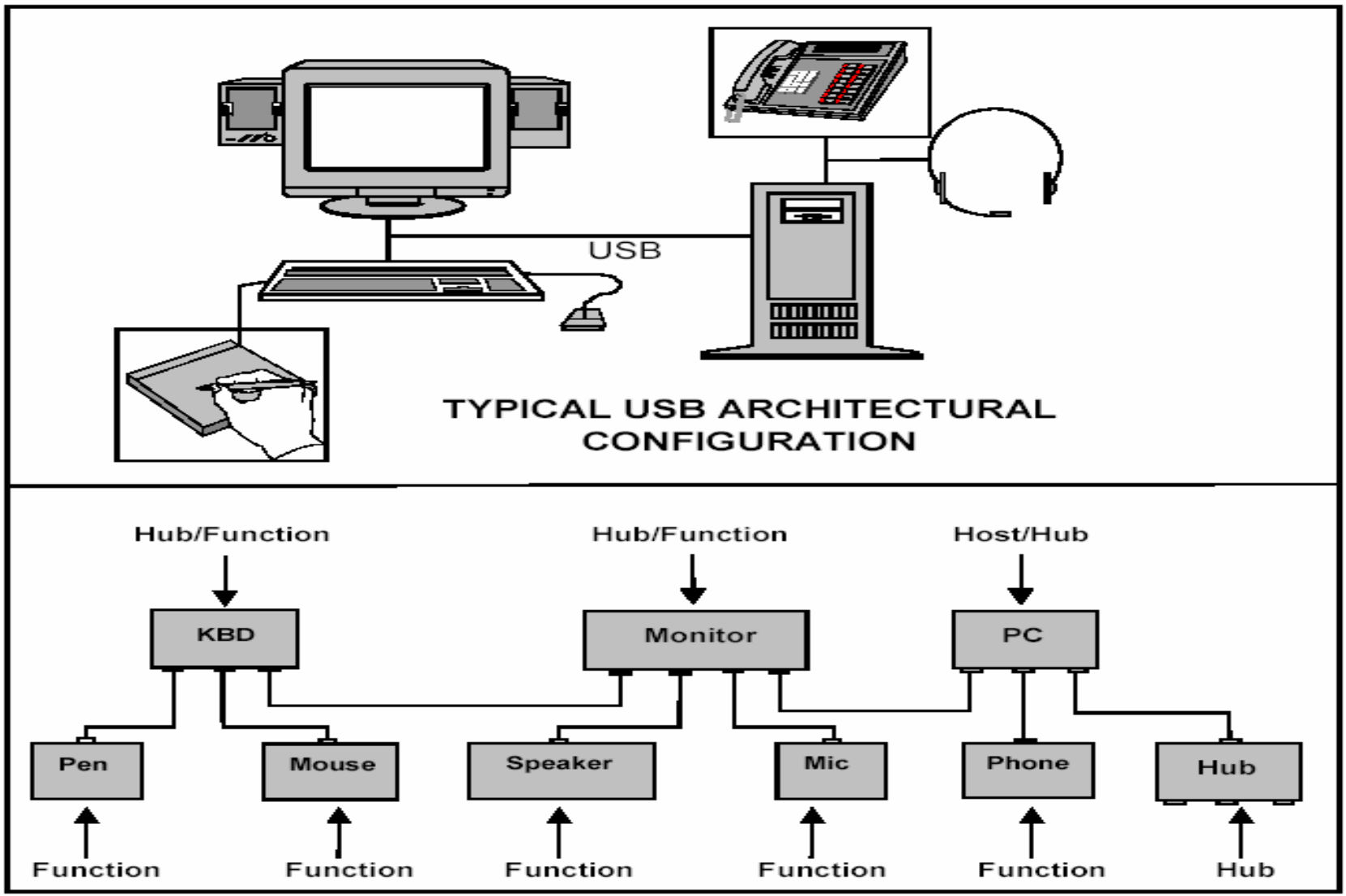


Figure 5-9. USB Host/Device Detailed View

Tiered Topology



Example Architecture



Benefits of USB

- Ease of use
  - One interface for many devices
  - Automatic configuration
  - No user settings
  - Simple cables
  - Hot pluggable
- Speed
  - Low speed – 1.5Mbps (USB v1.1)
  - Full speed – 12Mbps (USB v1.1)
  - High speed – 480Mbps (USBv2.0)

Benefits of USB (contd…)

- Flexibility
  - Four transfer types and three speeds make it feasible for many types of peripherals
- OS support
  - Win NT4 doesn’t support USB!

Challenges of USB

- Lack of support for “legacy” hardware
  - Need converters for use with peripherals with RS-232 or Centronics-type parallel ports
- Distance Limits
  - Max cable segment length of 5m
  - Can be made as much as 30m using 5 hubs and a device
- Protocol complexity

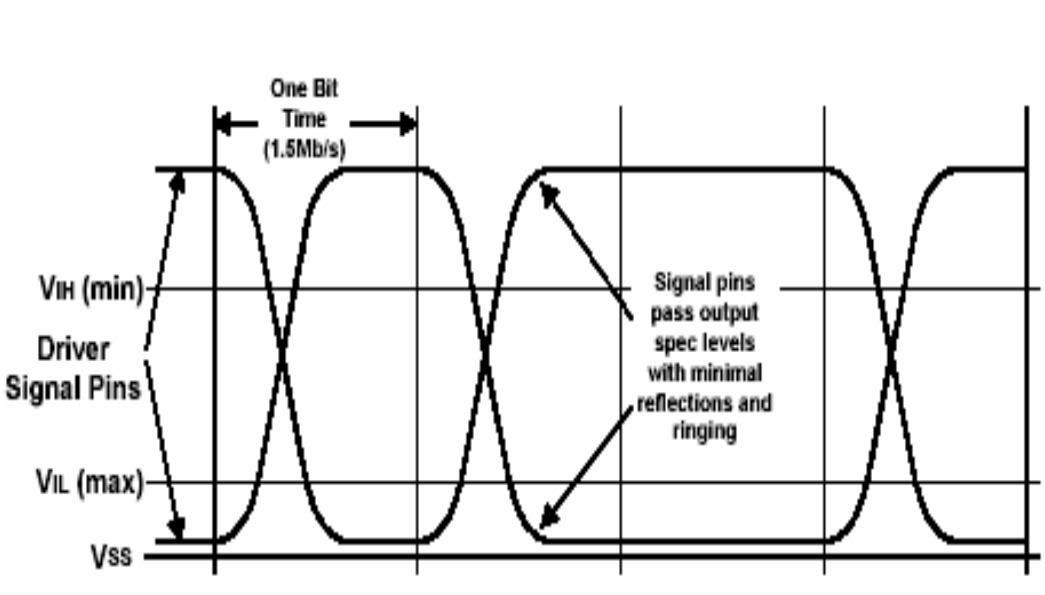
Transaction Types

- Control:
  - Request/Response type, non-periodic
  - guaranteed 10% frame time and reliable data transfer
  - USB structured data: control transfers, device configuration, device specific purposes
- Bulk transfer:
  - Large, non-periodic packets,guaranteed data delivery
  - access on available bandwidth
- Interrupt transfer
  - Bounded latency but reliable transfer
  - Small infrequent data – to poll event driven peripherals
- Isochronous transfers
  - Guaranteed bandwidth and latency
  - Periodic, error tolerant.
  - 90% frame time for periodic transfers

Signaling

- 4 wired bus: Vbus, Gnd, D+, D-
- Differential Signaling
- Signal levels:-
  - 'J' state
    - differential '0' for FS
    - differential '1' for LS
    - 'High speed J state' -> High speed differential "1"
  - 'K' state
    - differential '0' for LS
    - differential '1' for FS
    - 'High speed k state' -> High speed differential "0"

Differential Signaling



Differential Signaling

- At the driver side,
  - Differential ‘1’ when D+>=2.8V and D-<=0.3V
  - Differential ‘0’ when D->=2.8V and D+<=0.3V

- At the receiver side,
  - Differential ‘1’ when D+>=2V and diff(D+,D-)>=200mV
  - Differential ‘0’ when D->=2V and diff(D-,D+) >=200mV

Signaling (contd…)

Bus State	Signaling Levels		
	At originating source connector (at end of bit time)	At final target connector	
		Required	Acceptable
Differential "1"	D+ > V_{OH} (min) and D- < V_{OL} (max)	(D+) - (D-) > 200 mV and D+ > V_{IH} (min)	(D+) - (D-) > 200 mV
Differential "0"	D- > V_{OH} (min) and D+ < V_{OL} (max)	(D-) - (D+) > 200 mV and D- > V_{IH} (min)	(D-) - (D+) > 200 mV
Single-ended 0 (SE0)	D+ and D- < V_{OL} (max)	D+ and D- < V_{IL} (max)	D+ and D- < V_{IH} (min)
Single-ended 1 (SE1)	D+ and D- > V_{OSE1} (min)	D+ and D- > V_{IL} (max)	

Signaling (contd…)

	At originating source connector	At final target connector	
Data J state: Low-speed Full-speed	 Differential "0" Differential "1"	 Differential "0" Differential "1"	
Data K state: Low-speed Full-speed	 Differential "1" Differential "0"	 Differential "1" Differential "0"	
Idle state: Low-speed Full-speed	NA	 D- > V_{IHZ} (min) and D+ < V_{IL} (max) D+ > V_{IHZ} (min) and D- < V_{IL} (max)	 D- > V_{IHZ} (min) and D+ < V_{IH} (min) D+ > V_{IHZ} (min) and D- < V_{IH} (min)
Resume state	Data K state	Data K state	

Signaling (contd…)

	At originating source connector	At final target connector	
Start-of-Packet (SOP)	Data lines switch from Idle to K state		
End-of-Packet (EOP)[4]	SE0 for approximately 2 bit times[1] followed by a J for 1 bit time[3]	SE0 for ≥ 1 bit time[2] followed by a J state for 1 bit time	SE0 for ≥ 1 bit time[2] followed by a J state
Disconnect (at downstream port)	NA	SE0 for ≥2.5 µs	
Connect (at downstream port)	NA	Idle for ≥2 ms	Idle for ≥2.5 µs
Reset	D+ and D- < VOL (max) for ≥10ms	D+ and D- < VIL (max) for ≥10 ms	D+ and D- < VIL (max) for ≥2.5 µs

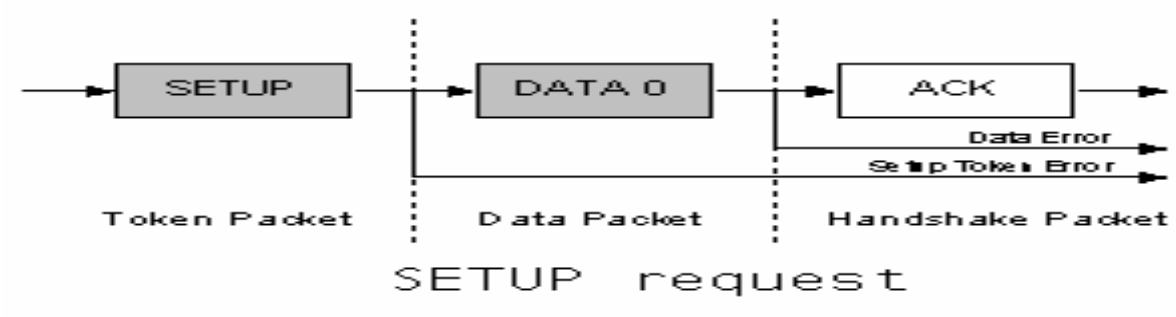
Packet Types in USB

(Relevant to LS devices)

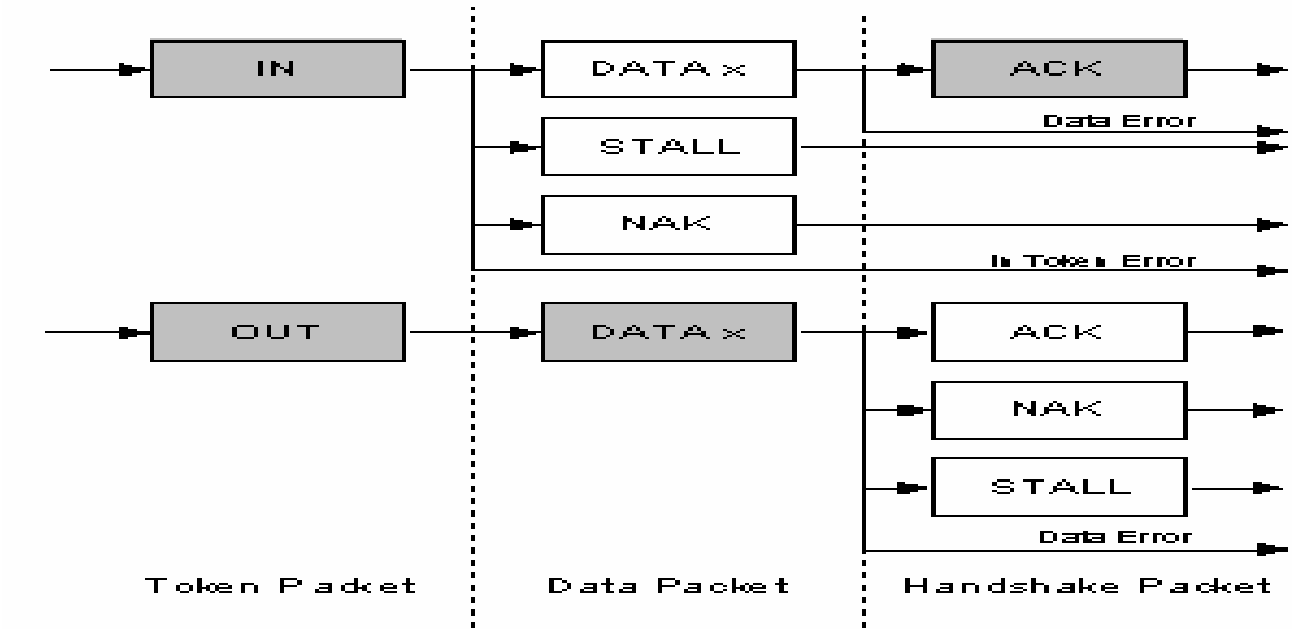
- **TOKEN Packet (PID + Address)**
  - **SETUP**
  - **IN**
  - **OUT**
- **DATA Packet (PID + DATA)**
  - **DATA0**
  - **DATA1**
- **HANDSHAKE Packet (PID Only)**
  - **ACK**
  - **NAK**
  - **STALL**

Control SETUP Transaction

- **SETUP Stage**

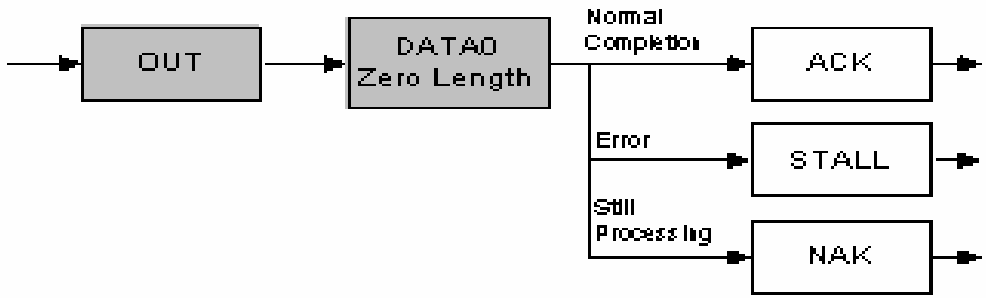


- **DATA Stage**

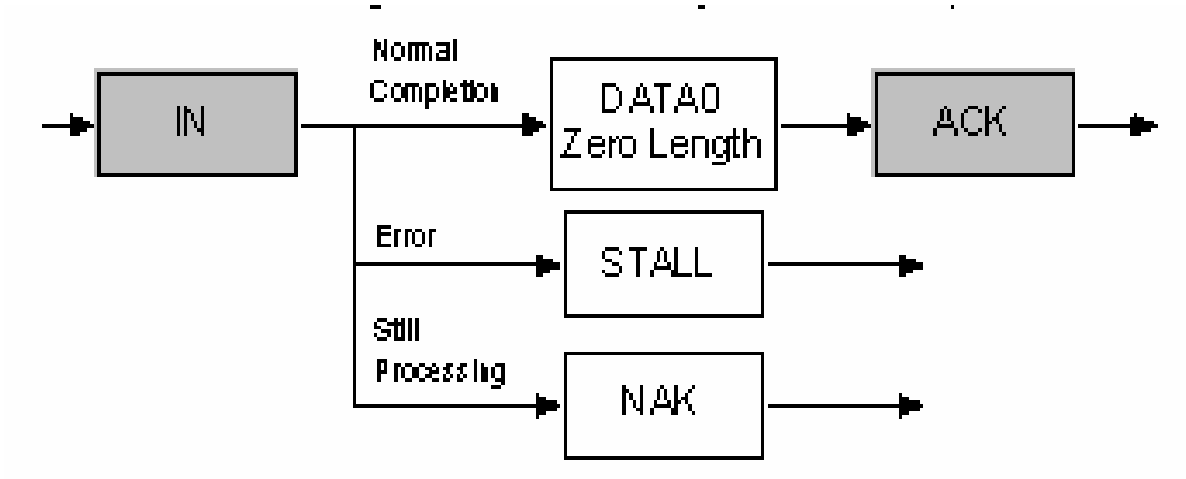


Control SETUP Transaction (Cntd..)

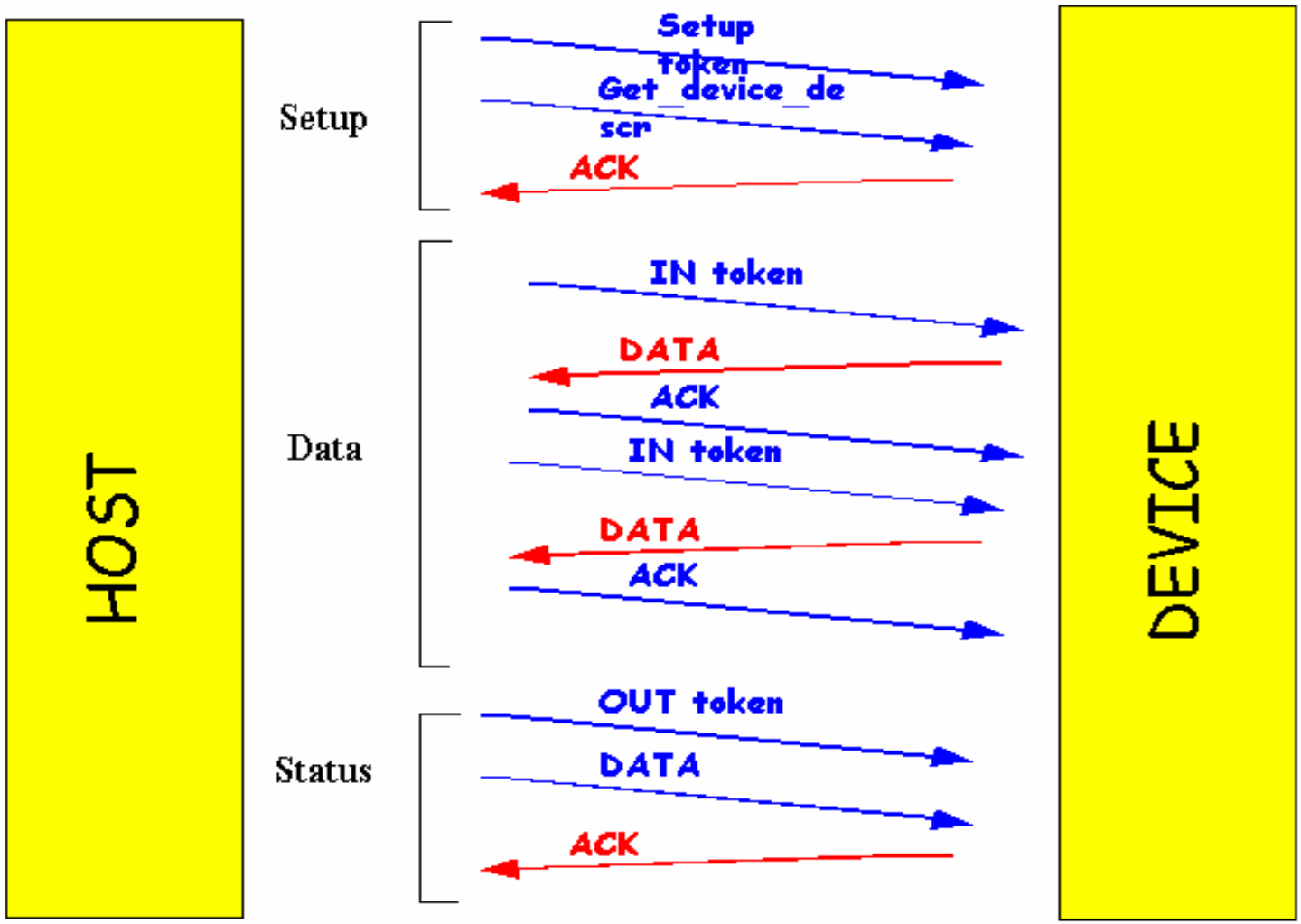
- **STATUS Stage**
  - **For Control Read**



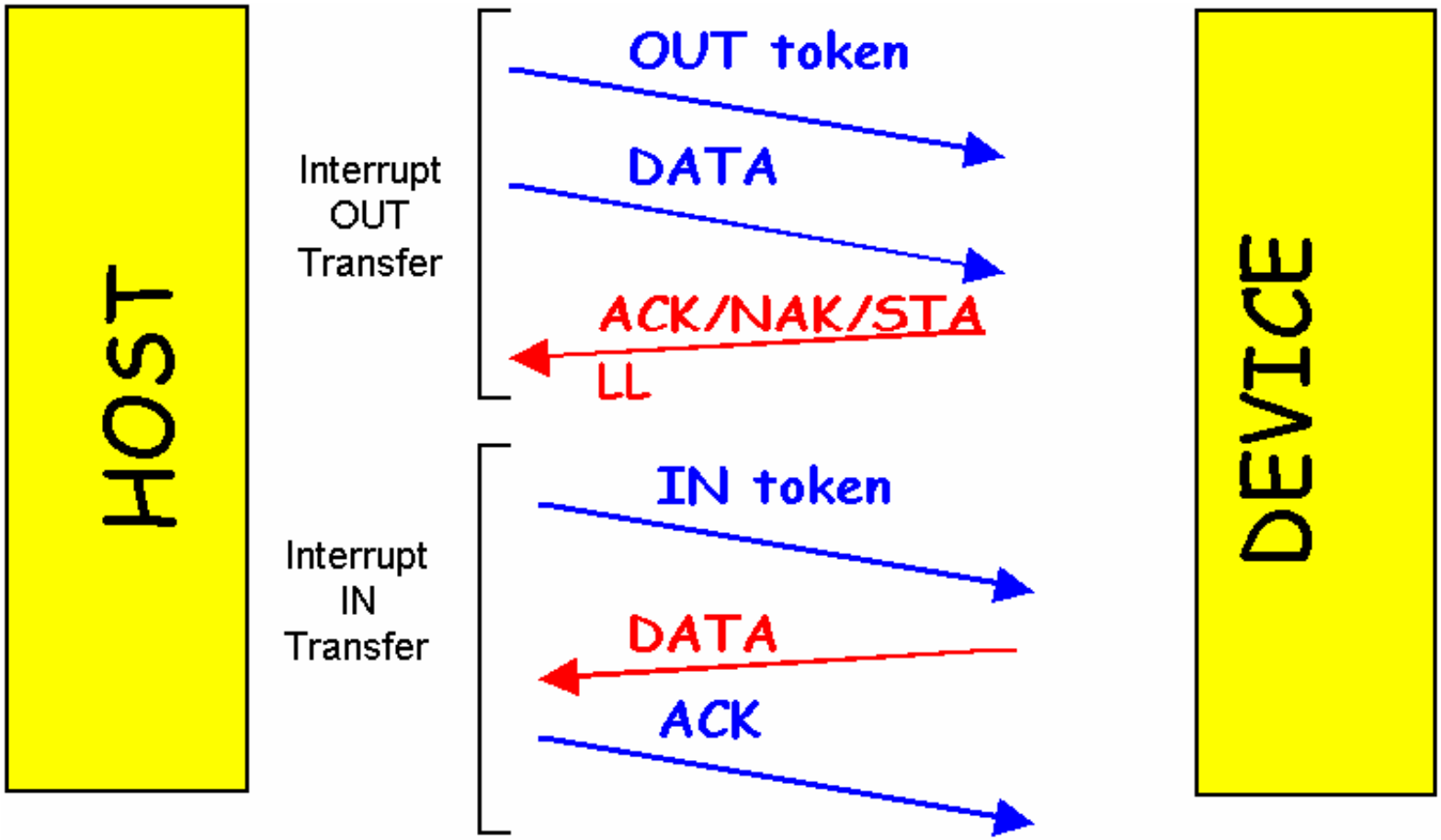
  - **For Control Writes**



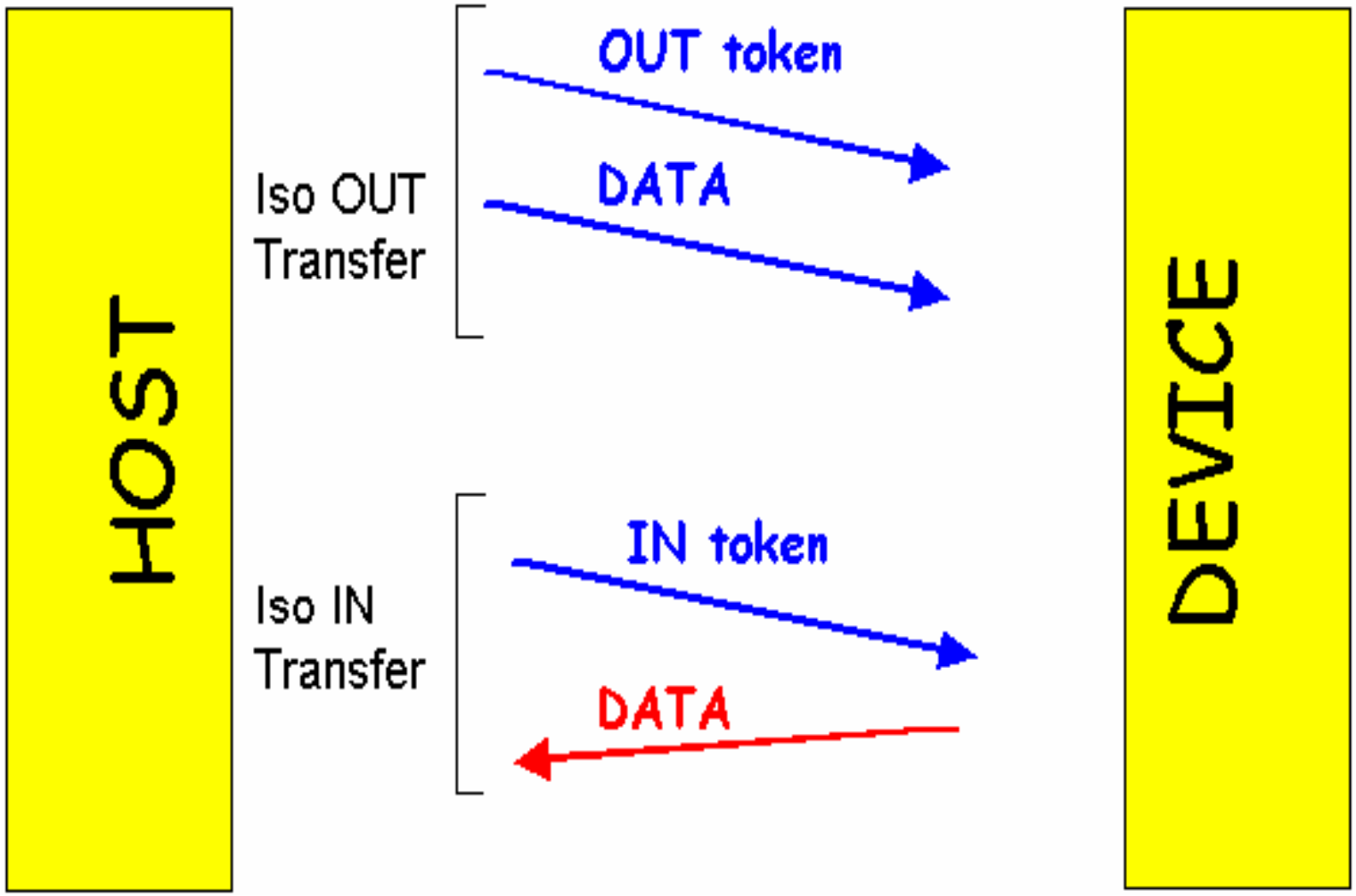
Sample Transfers #1 (Get_Device_Descriptor Request)



Sample Transfers #2 (Interrupt Type)



Sample Transfers #3
(Isochronous Type)



Packet Formats

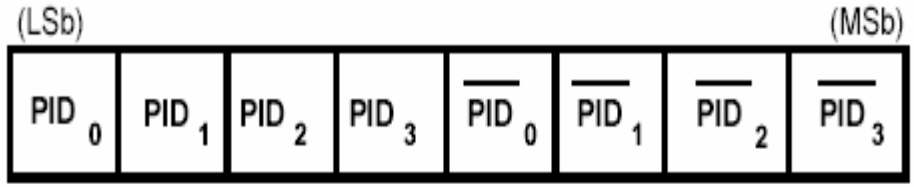


Figure 8-1. PID Format

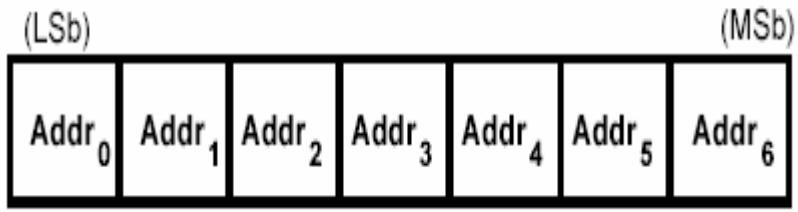


Figure 8-2. ADDR Field

PID Types

PID Type	PID Name	PID<3:0>*	Description
Token	OUT	0001B	Address + endpoint number in host-to-function transaction
	IN	1001B	Address + endpoint number in function-to-host transaction
	SOF	0101B	Start-of-Frame marker and frame number
	SETUP	1101B	Address + endpoint number in host-to-function transaction for SETUP to a control pipe
Data	DATA0	0011B	Data packet PID even
	DATA1	1011B	Data packet PID odd
	DATA2	0111B	Data packet PID high-speed, high bandwidth isochronous transaction in a microframe (see Section 5.9.2 for more information)
	MDATA	1111B	Data packet PID high-speed for split and high bandwidth isochronous transactions (see Sections 5.9.2, 11.20, and 11.21 for more information)

PID Types (contd…)

PID Type	PID Name	PID<3:0>*	Description
Handshake	ACK	0010B	Receiver accepts error-free data packet
	NAK	1010B	Receiving device cannot accept data or transmitting device cannot send data
	STALL	1110B	Endpoint is halted or a control pipe request is not supported
	NYET	0110B	No response yet from receiver (see Sections 8.5.1 and 11.17-11.21)
Special	PRE	1100B	(Token) Host-issued preamble. Enables downstream bus traffic to low-speed devices.
	ERR	1100B	(Handshake) Split Transaction Error Handshake (reuses PRE value)
	SPLIT	1000B	(Token) High-speed Split Transaction Token (see Section 8.4.2)
	PING	0100B	(Token) High-speed flow control probe for a bulk/control endpoint (see Section 8.5.1)
	Reserved	0000B	Reserved PID

Packet Formats (contd…)

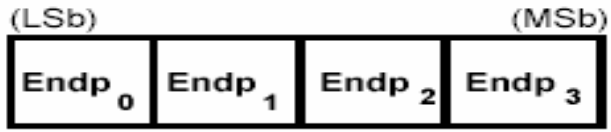


Figure 8-3. Endpoint Field

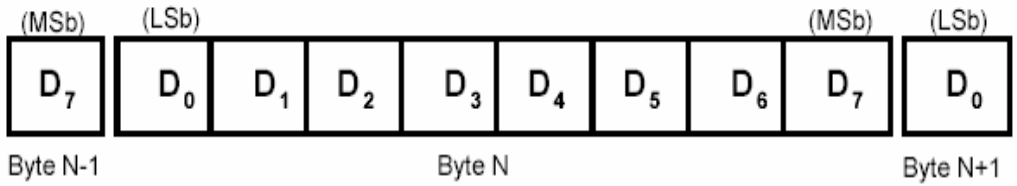


Figure 8-4. Data Field Format

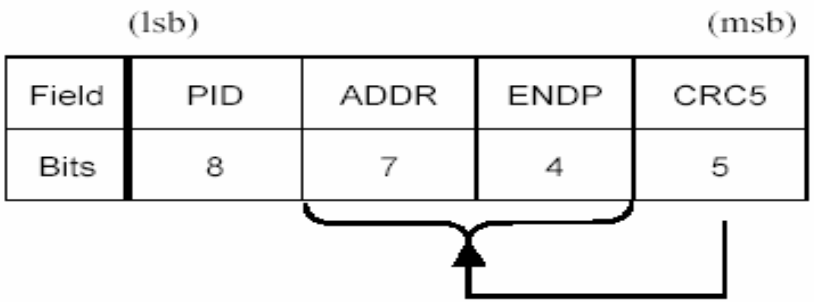


Figure 8-5. Token Format

Packet Formats (contd…)

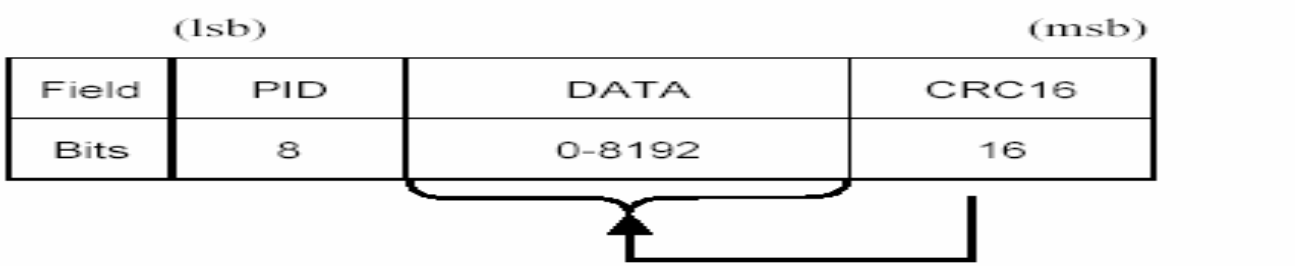


Figure 8-15. Data Packet Format

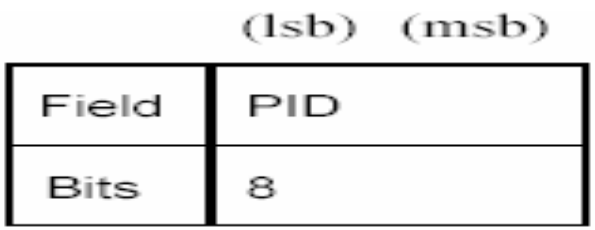


Figure 8-16. Handshake Packet

Host responses to IN transactions

Table 8-5. Host Responses to IN Transactions

Data Packet Corrupted	Host Can Accept Data	Handshake Returned by Host
Yes	N/A	Discard data, return no response
No	No	Discard data, return no response
No	Yes	Accept data, issue ACK

Function Responses to IN transactions

Table 8-4. Function Responses to IN Transactions

Token Received Corrupted	Function Tx Endpoint Halt Feature	Function Can Transmit Data	Action Taken
Yes	Don't care	Don't care	Return no response
No	Set	Don't care	Issue STALL handshake
No	Not set	No	Issue NAK handshake
No	Not set	Yes	Issue data packet

Data Encoding/Decoding

- Uses NRZI encoding and decoding
- Uses bit stuffing
  - Stuff a ‘0’ after every 6 consecutive ‘1’s

Synchronization

- Bit stuffing
- Sync pattern of KJKJKJKK

- CRC calculation
  - Polynomials for data and address
  - Address CRC Generator Polynomial $x^5 + x^2 + 1$
  - Data CRC Generator Polynomial $x^{16} + x^{15} + x^2 + 1$