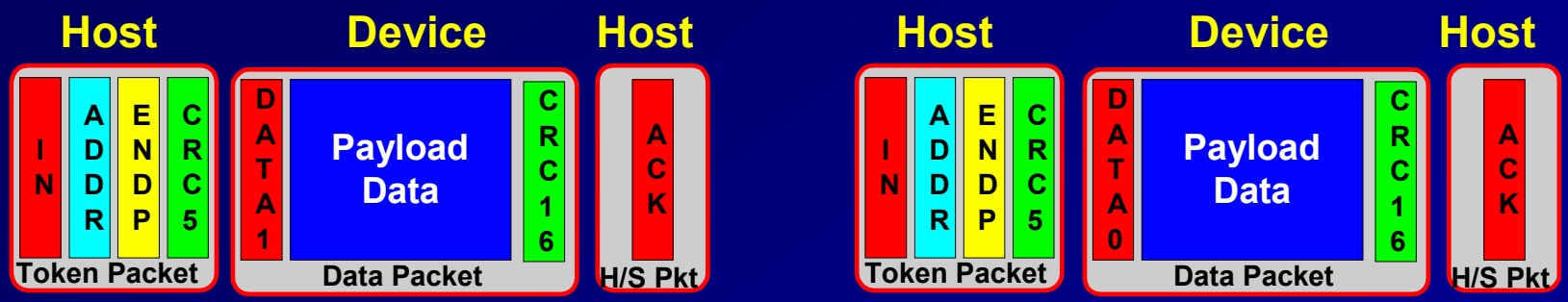
Interrupt transfers

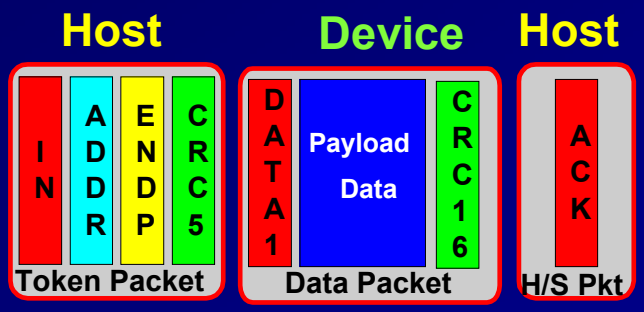
- Guaranteed Latency
- Polling interval is programmable:
  - 1 mS to 255 mS (full speed)
  - 10 mS to 255 mS (low speed)
- Unidirectional
- Error detection and next period retry.
- Maximum data payload size
  - Low-speed - 8 bytes – 8,000 bytes/sec.
  - Full-speed - 64 bytes – 64,000 bytes/sec.
  - High-speed - 1024 bytes – 8,192,000 bytes/sec.

Interrupt transfers

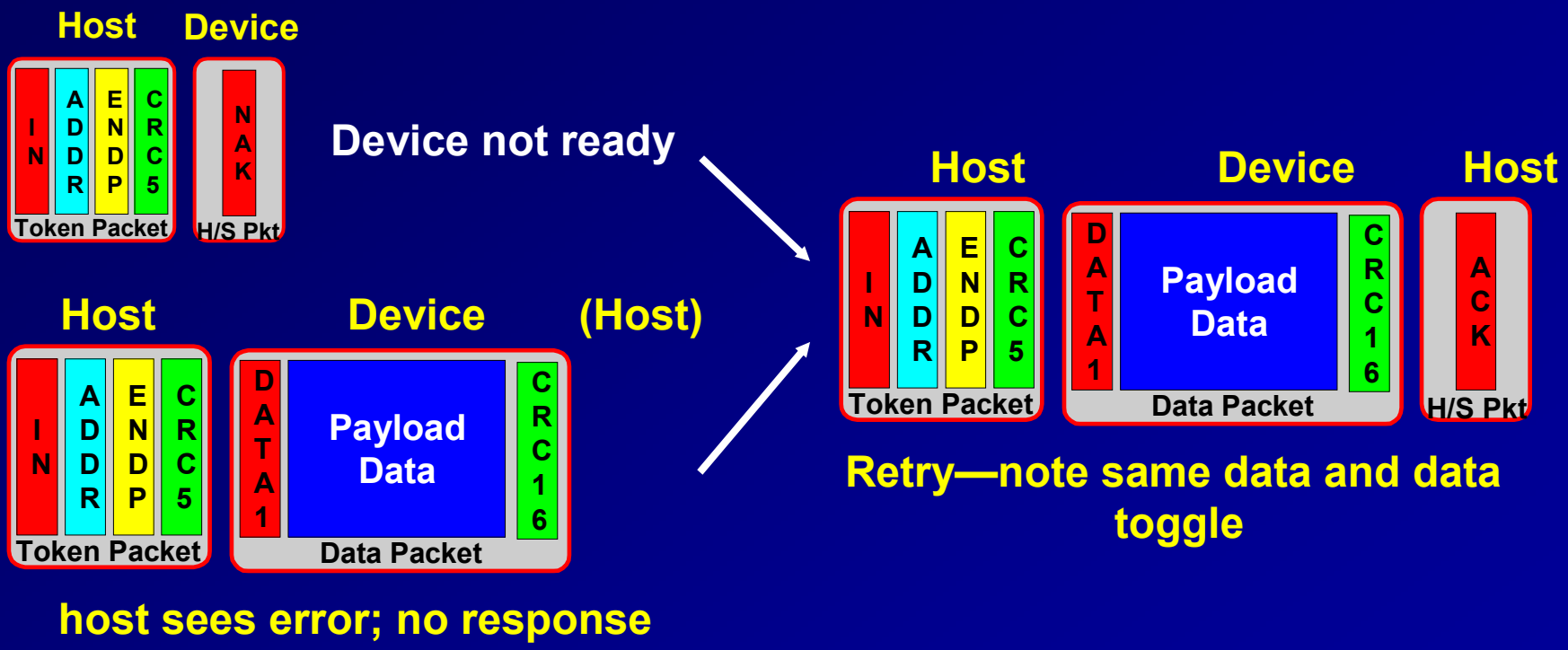
- During enumeration, each interrupt endpoint declares its payload size and polling interval.
- Host may reject devices if available bandwidth is not enough.
- When an interrupt transfer involves more data than can fit in one data payload of the currently established maximum size, all data payloads are required to be maximum-sized except for the last data payload, which will contain the remaining data.

Interrupt IN Transfer

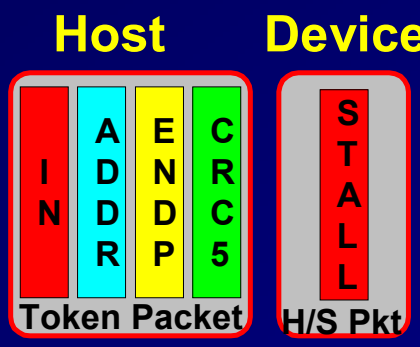




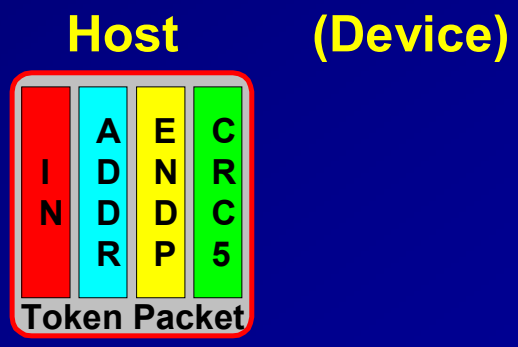
Interrupt IN Transfer



Interrupt IN Transfer

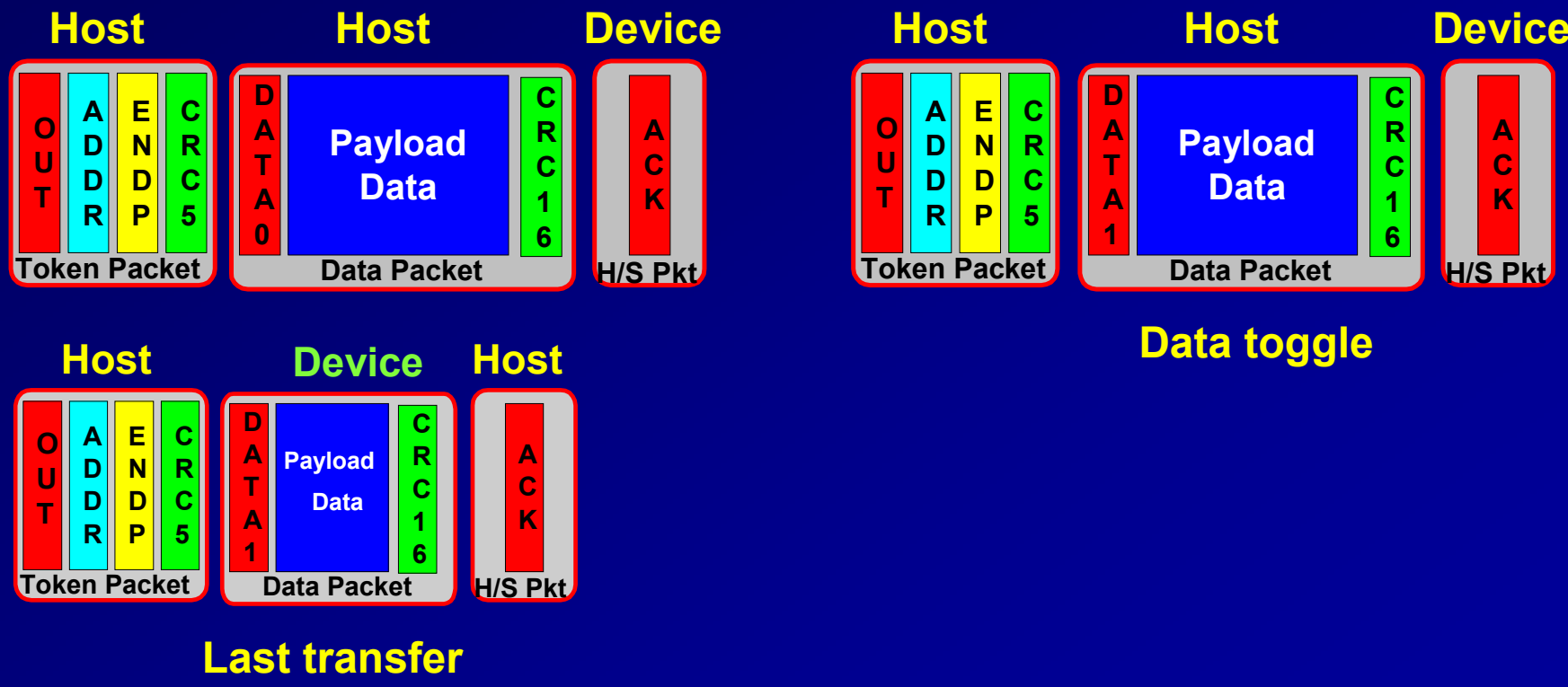


device has a problem

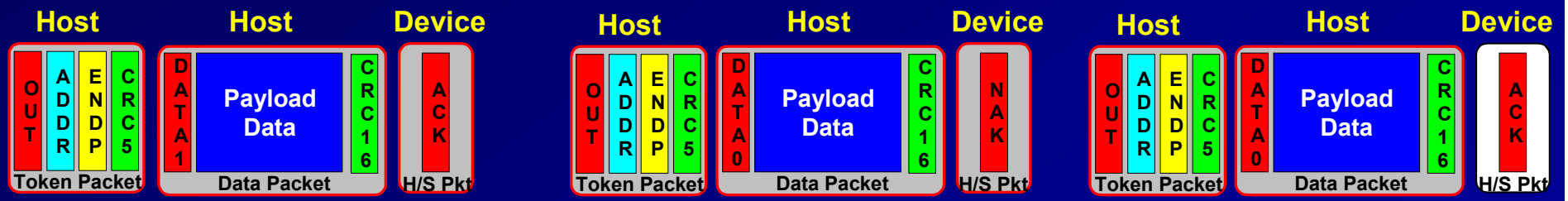


device detects token error
or does not respond

Interrupt OUT Transfer



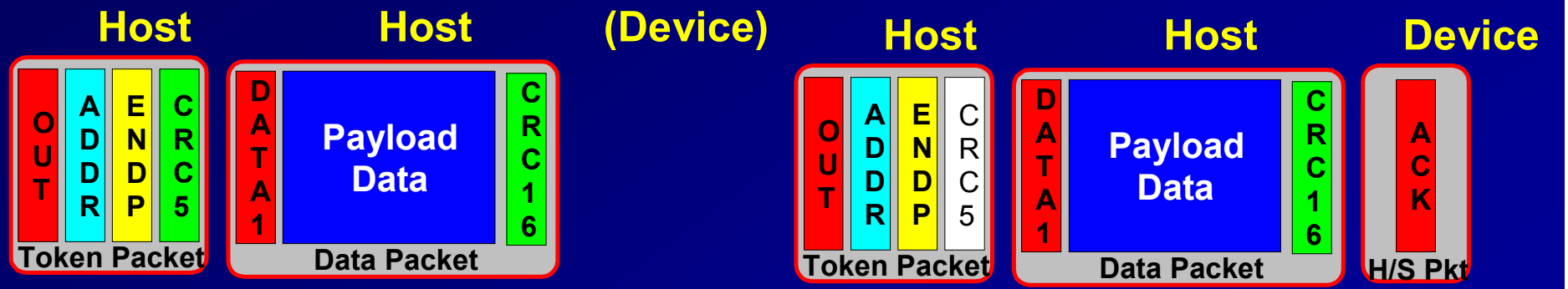
Interrupt OUT Transfer



device not ready for data

retry

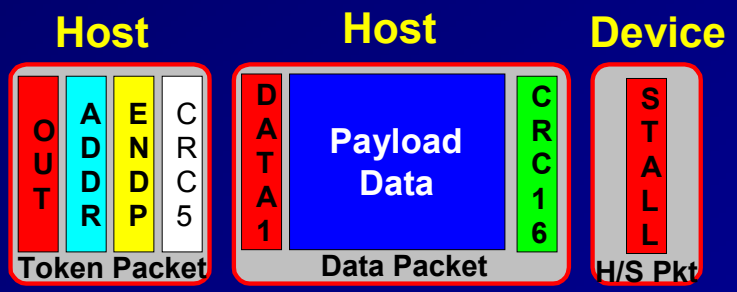
Interrupt OUT Transfer



device sees error; no response

Retry

Interrupt OUT Transfer



device has problem

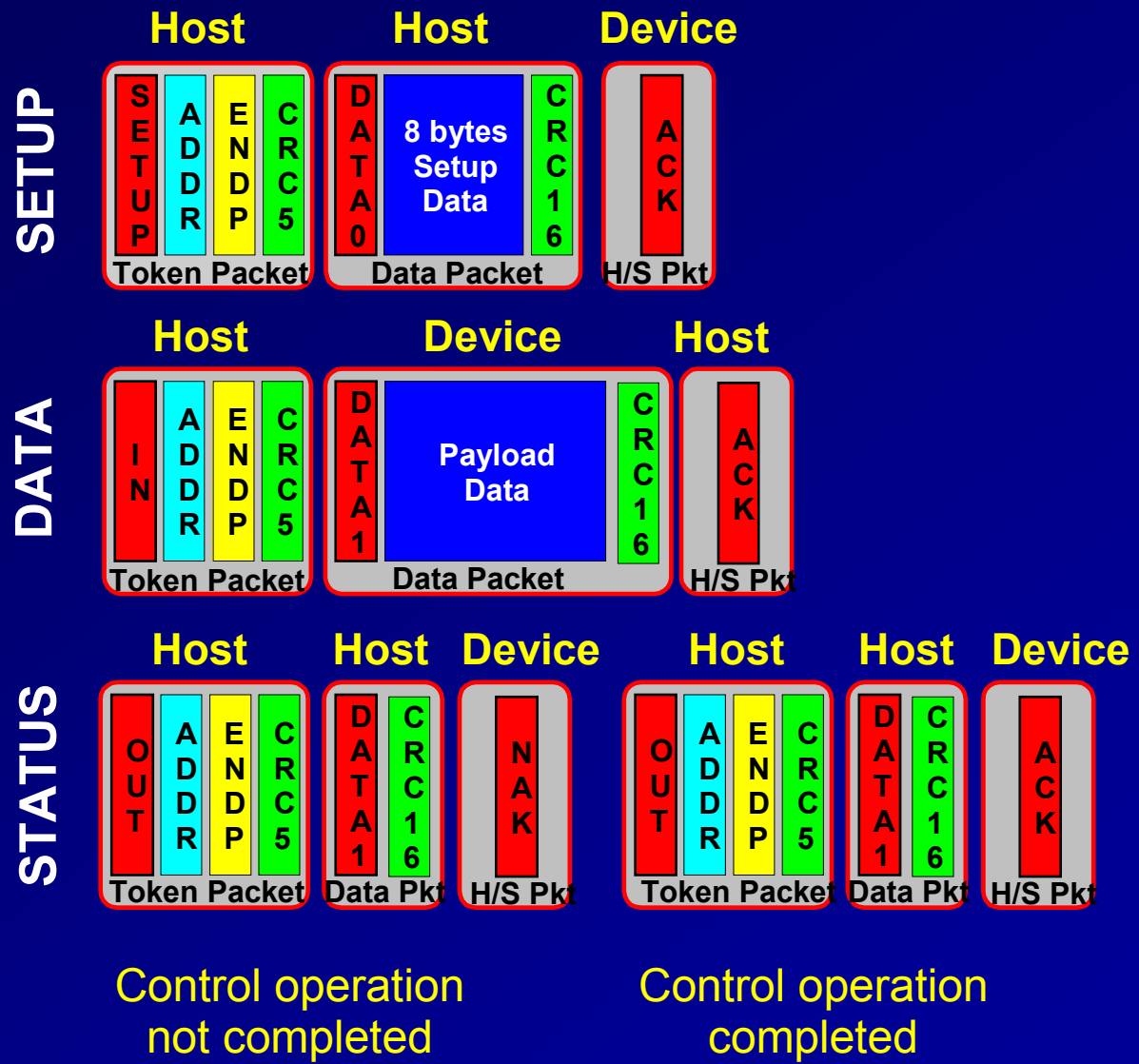
Bulk transfer

- Similar to Interrupt transfer, with no periodic polling
- FS and HS only
- No guarantee of bandwidth or minimum latency
- Unidirectional
- Error detection and next period retry.
- Maximum data payload size
  - Full-speed - 64 bytes – 64,000 bytes/sec.
  - High-speed - 512 bytes – 4,096,000 bytes/sec

Control transfers

- Used for command and status operations
- Essential to set up a USB device with all enumeration functions
- The packet length of control transfers in low speed devices must be 8 bytes, full speed devices allow a packet size of 8, 16, 32 or 64 bytes and high speed devices must have a packet size of 64 bytes.
- 3 stages – Setup, Data and Status

Control Transfer



Control transfers - Setup

- Setup token - contains the address and endpoint number.
- Data packet - PID type of data0 and includes 8 byte Setup data which details the type of request.
- Handshake packet - If the function successfully receives the setup data it responds with ACK, otherwise it ignores the data and doesn't send a handshake packet.
- Functions cannot issue a STALL or NAK packet in response to a setup packet.

We shall visit the setup data when discussing the enumeration process

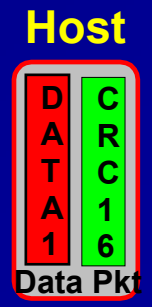
Control transfers - Data

- One or multiple IN or OUT transfers
- Most transfers are IN, where the device reports its capabilities to the host.
- Setup request indicates the amount of data to be transmitted in this stage.
- If it exceeds the maximum packet size, data will be sent in multiple transfers each being the maximum packet length except for the last packet.

We shall visit the capabilities data when discussing the enumeration process

Control transfers – Status IN tokens during the data stage

- Host sent IN tokens during the data stage to receive data
- Host acknowledge the successful receipt of this data by sending an OUT token followed by a zero length data packet.
- The function can now report its status in the handshaking stage.
  - ACK indicates completion of command and ready to accept another command.
  - STALL - An error occurred during the processing of this command
  - NAK - Function is still processing



Control transfers – Status OUT tokens during the data stage

- Host sent OUT tokens during the data stage to sent data
- Host send IN token and the function acknowledges the successful receipt of data by sending a zero length packet in response to an IN token.
  - STALL - An error occurred during the processing of this command
  - NAK - Function is still processing
- If success host send ACK