



Universal Serial Bus *USB*

Walter Fetter Lages

`w.fetter@ieee.org`

Universidade Federal do Rio Grande do Sul

Escola de Engenharia

Departamento de Engenharia Elétrica

Microprocessadores II



Introdução

- Soluções anteriores para conexão de periféricos ao PC
 - Não eram fáceis de usar (*plug-and-play*) do ponto de vista do usuário
 - Limitado número de portas que não podem ser facilmente aumentado
 - Falta de uma porta bidirecional, de baixo custo com uma taxa de transferência média
 - Para cada novo dispositivo uma nova interface era definida
- O USB está para a RS232 assim como o PCI está para o ISA



Objetivos

- Facilidade de conexão de periféricos ao PC
- Solução de baixo custo para até 480 Mb/s
- Suporte para voz, áudio e vídeo em tempo real
- Protocolo flexível com mistura de modos isócrono e assíncrono
- Integração com dispositivos de consumo
- Padrão capaz de ser facilmente adotado em produtos
- Permitir novas classes de dispositivos periféricos



Características

- Facilidade de uso para o usuário final
 - Um único modelo para o cabeamento e conectores
 - Detalhes elétricos invisíveis para o usuário (terminadores de barramento)
 - Periféricos auto-identificáveis, com configuração e seleção de *driver* automáticas
 - Periféricos *hot-pluggable*



Características

- Grande variedade de aplicações
 - Adequada para taxas de alguns kb/s até várias centenas de Mb/s
 - Transferências isócronas e assíncronas
 - Operação concorrente em vários dispositivos
 - Até 127 dispositivos físicos
 - Suporta dispositivos compostos (que possuem diversas funções)
 - Baixo *overhead* do protocolo
 - Alimentação no mesmo cabo +5V @ 100-500mA, dependendo do hub
 - Dispositivos entram no modo de economia após 3ms de inatividade no barramento



Características

- Flexibilidade
 - Variedade de tamanhos de pacotes
 - Permite variação nas taxas de dados dos periféricos
 - O protocolo implementa controle de fluxo
- Robustez
 - Tratamento e recuperação de erros incluídos no protocolo
 - Adição e remoção de dispositivos percebidos pelo usuário como sendo em tempo real

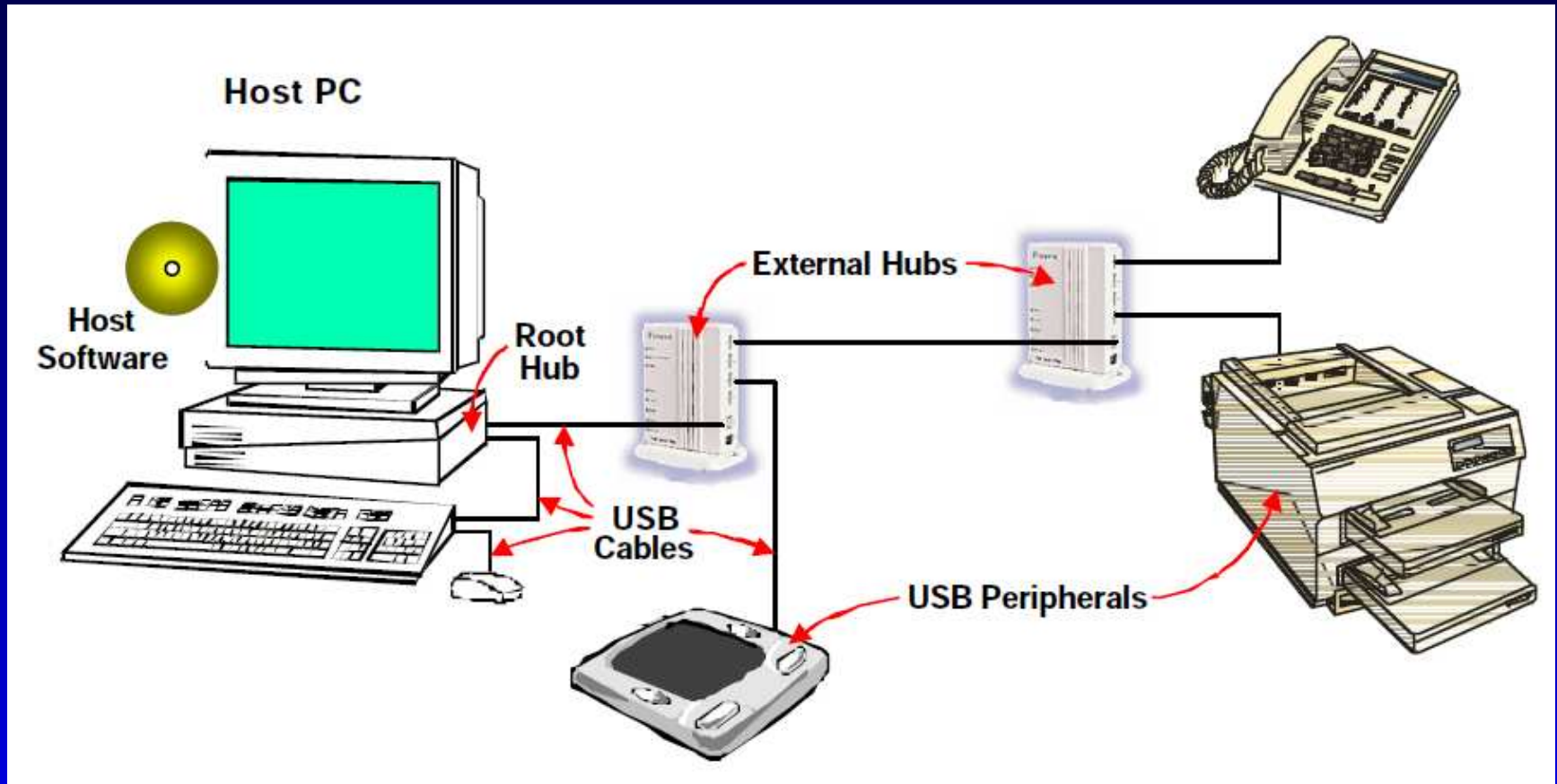


Versões do USB

- USB 1.1
 - Low-speed: 1.5 Mbps
 - Full-speed: 12 Mbps
- USB 2.0
 - High-speed: 480 Mbps
- USB On-The-GO
- Wireless USB
 - 3 m: 480 Mbps
 - 10 m: 110 Mbps

Arquitetura

- Fisicamente não é uma topologia em barramento





Sistema USB

- Dispositivos USB
 - Hubs
 - Funções
- *Host* USB
- Interconexão USB
 - Topologia do barramento
 - Relações entre camadas
 - Modelo de fluxo de dados
 - Escalonamento do meio de transmissão

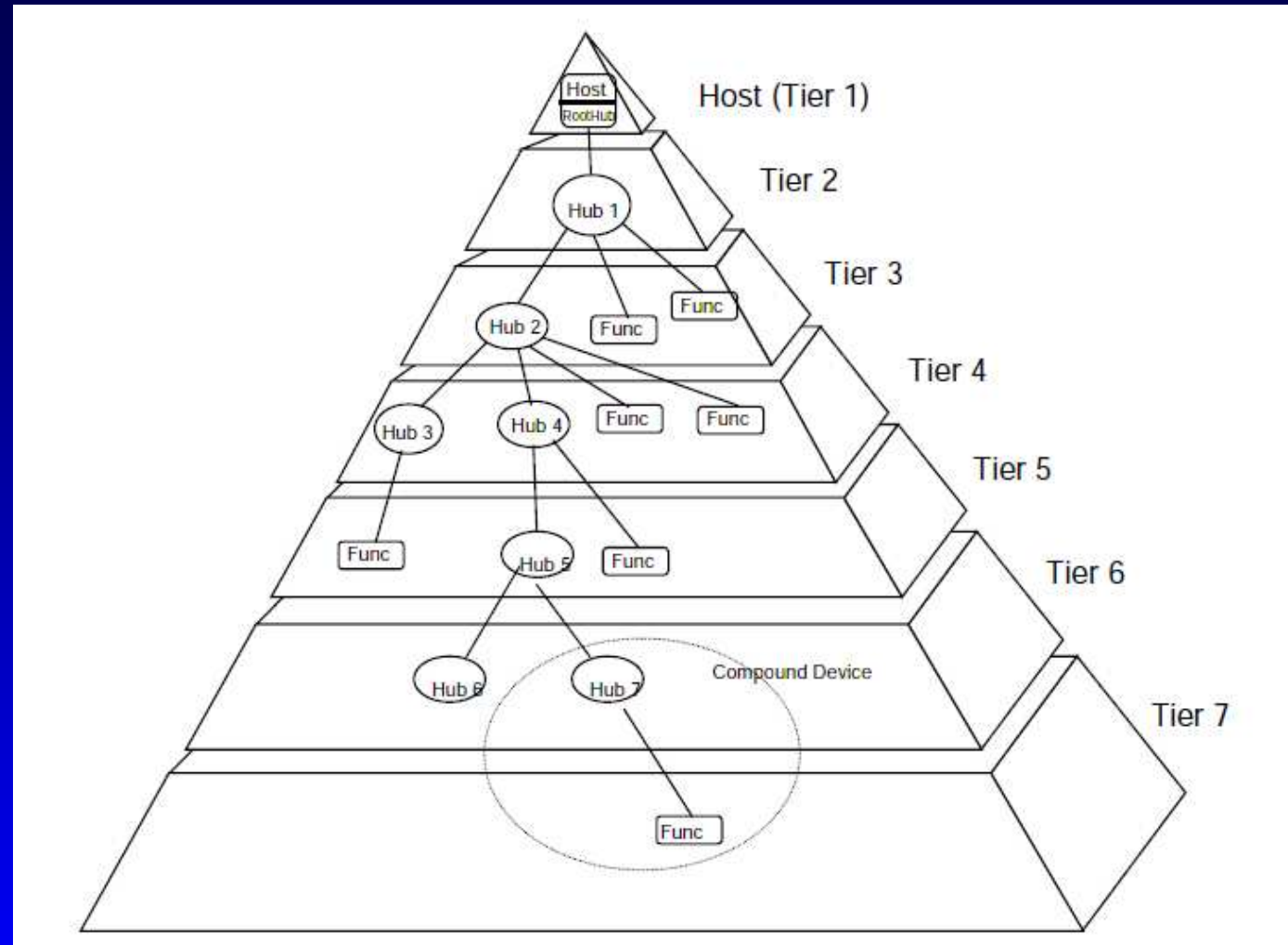


Dispositivos USB

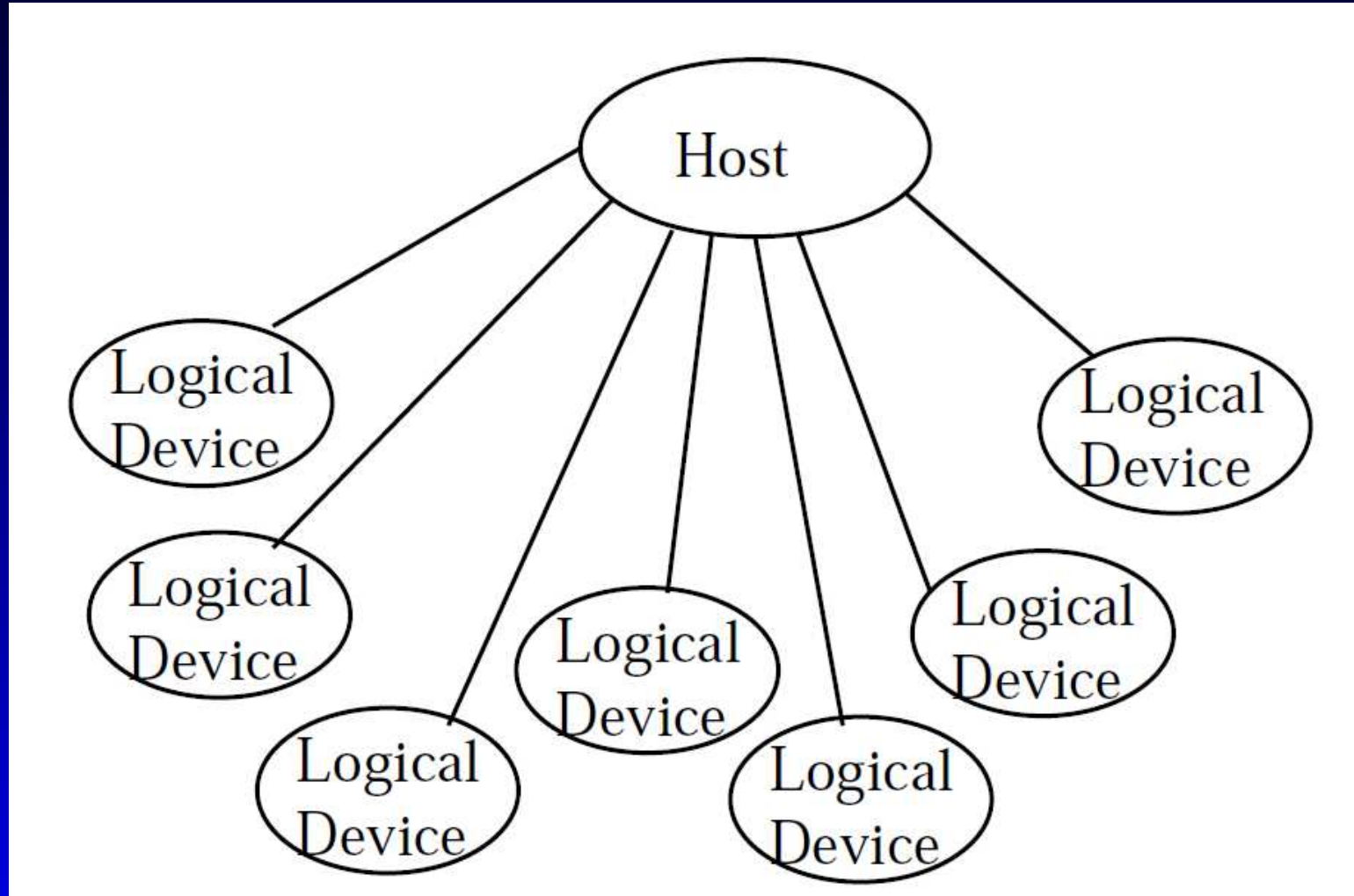
- Dispositivos USB apresentam uma interface padrão USB
 - Entendem o protocolo USB
 - Respondem a operações padrão USB como configuração e reset
 - Possuem informação de descrição
 - Versão do padrão USB
 - Classe de dispositivo
 - *hub*
 - HCI
 - impressora
 - dispositivo de imagem
 - dispositivo de armazenamento
 - Fabricante

Topologia Física

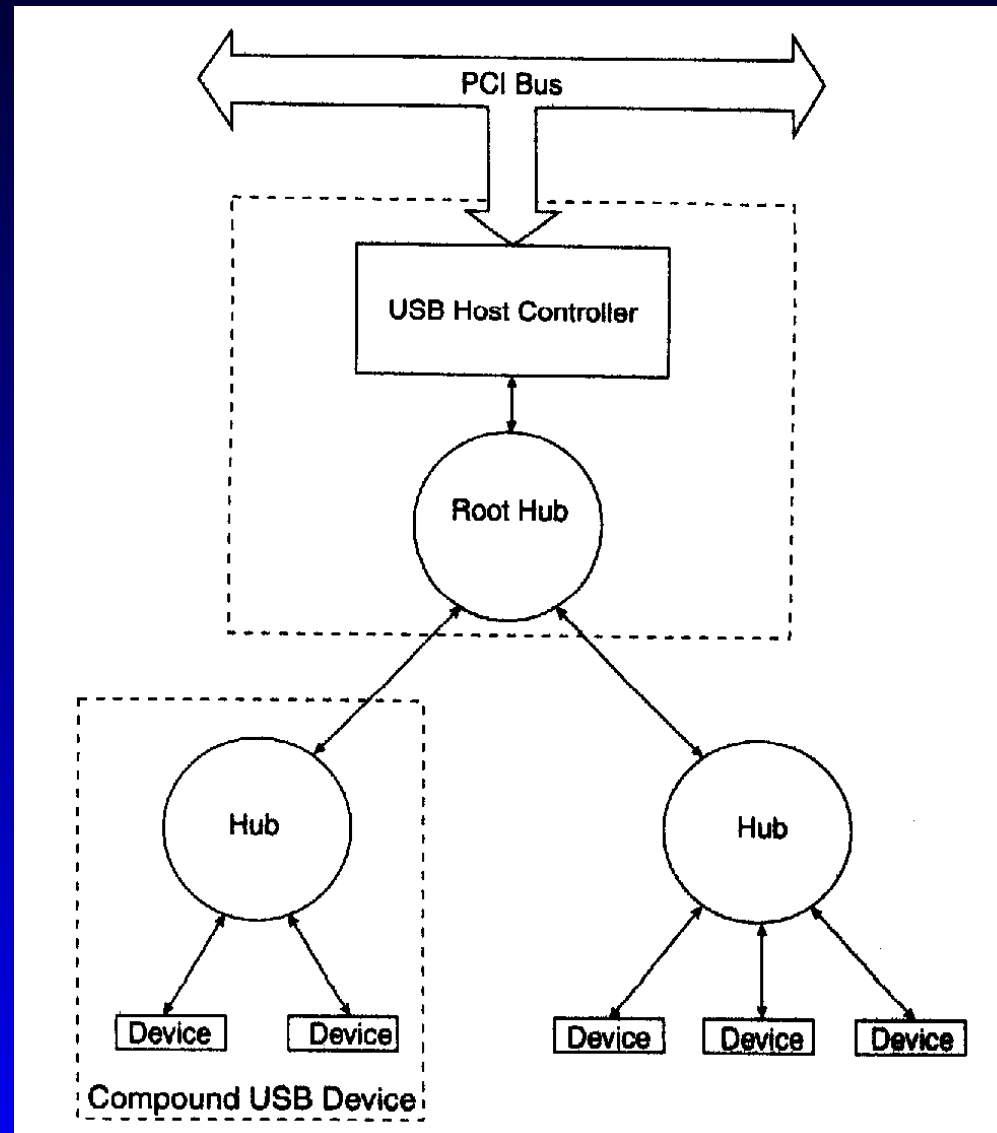
- No máximo 7 níveis devido a temporização



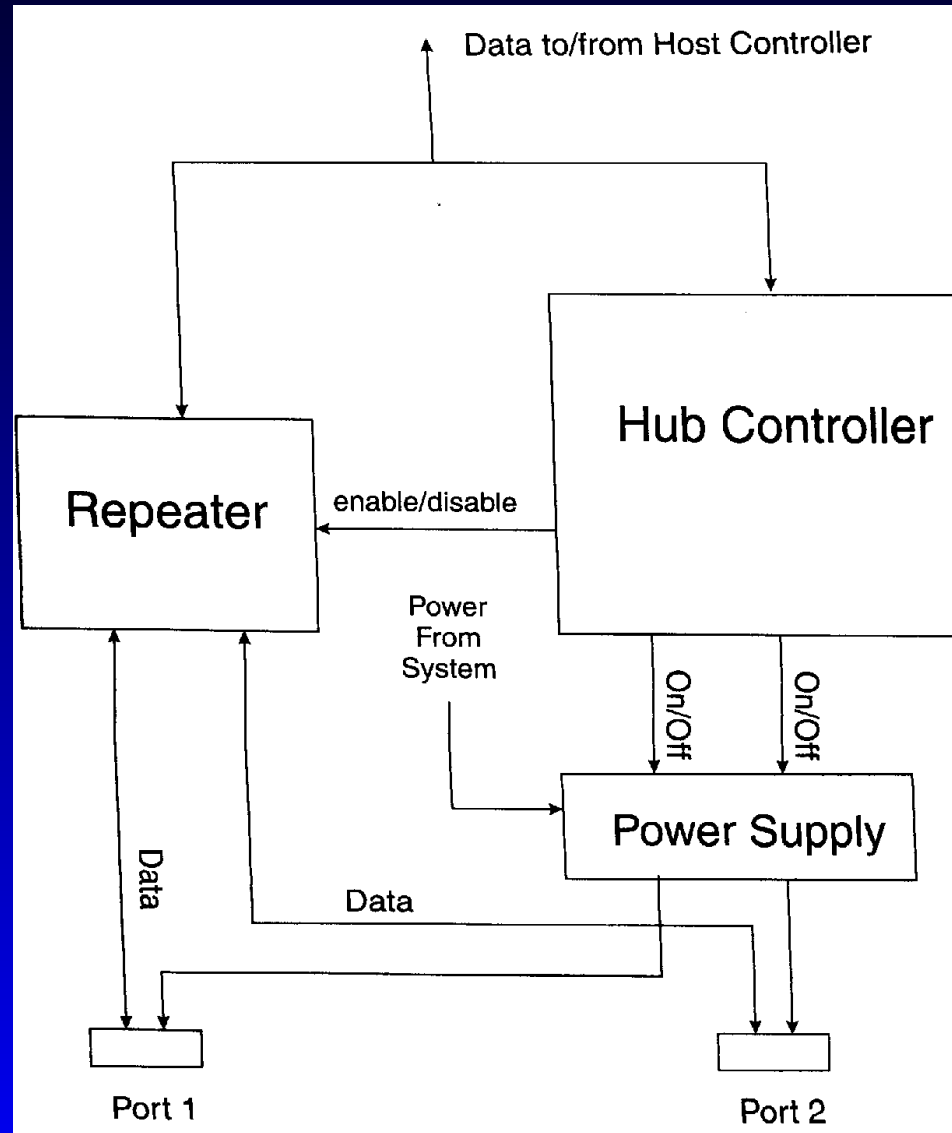
Topologia Lógica



Hub USB



Hub raiz





Hardware USB

- Controlador USB do host
 - Faz a conversão serial/paralelo
 - Gera as transações USB
 - Envia as transações para o hub raíz
- Hub raíz
 - Implementa as portas no host
 - Controla a alimentação das portas USB
 - Habilita/desabilita portas
 - Reconhece dispositivos conectados às portas
 - Reporta status associado a cada porta

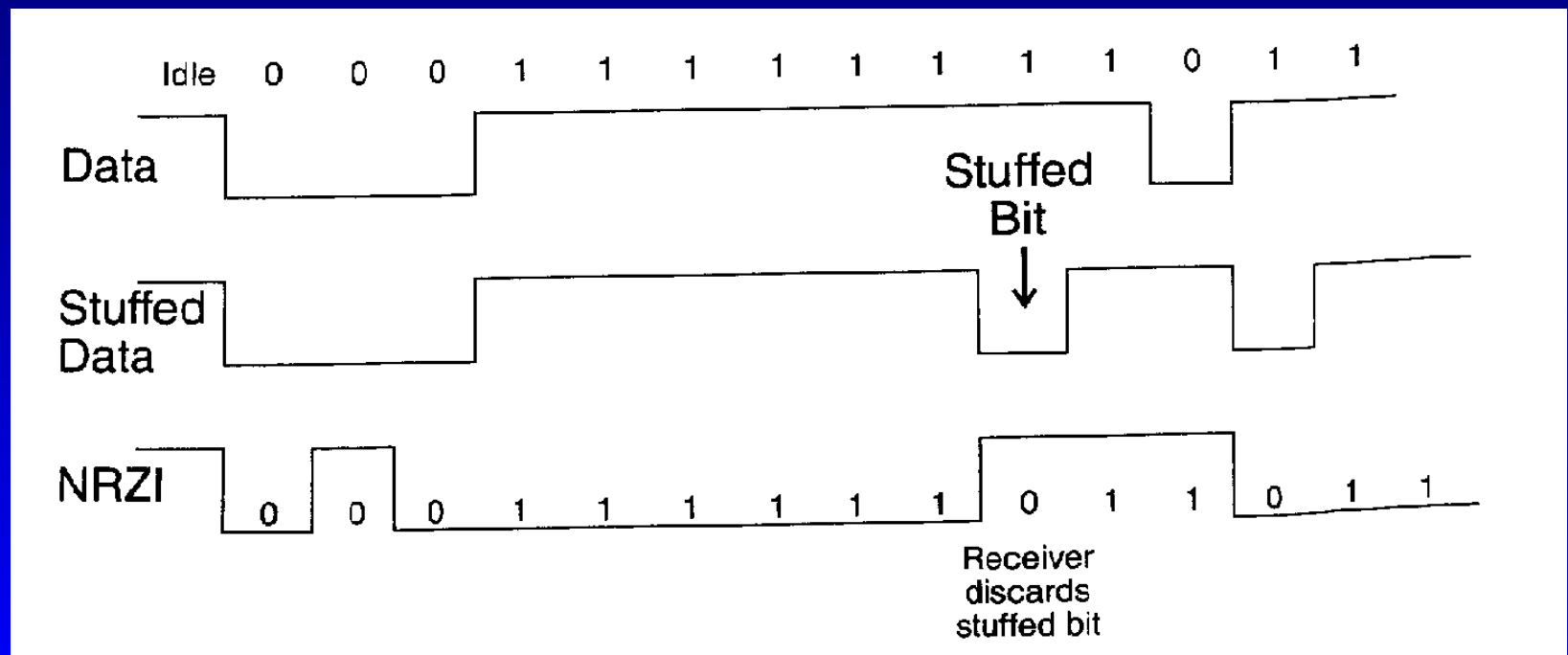


Hardware USB

- Hubs USB
 - Permite a extensão do barramento USB
 - Pode ser integrado a dispositivos ou stand-alone
 - Deve detectar quando dispositivos são conectados ou desconectados nas suas portas
- Dispositivos USB
 - *High-speed* 480 Mbps
 - *Full-speed* 12 Mbps
 - *Low-speed* 1.5 Mbps
 - Portas de baixa velocidade ficam desabilitadas quando há uma transmissão de alta velocidade

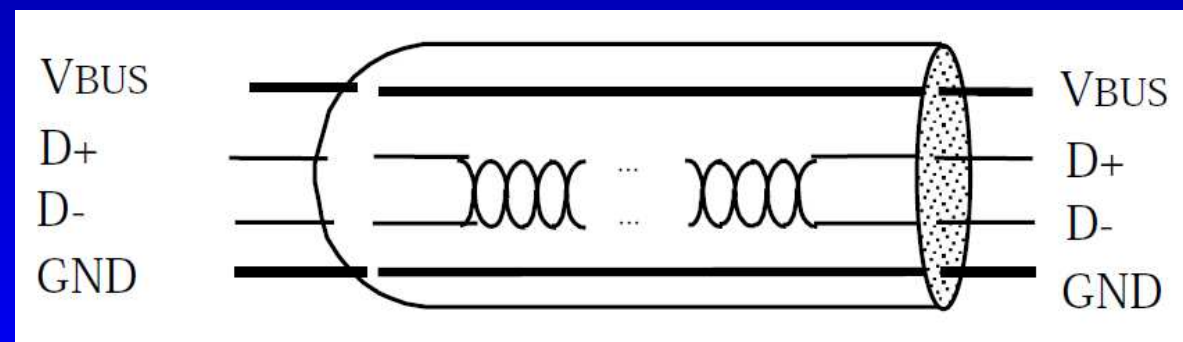
Interface Física

- Sinais e alimentação em um cabo com 4 vias
- Dados codificados em NRZI com *bit stuffing* após seis 1s consecutivos
 - *clock* pode ser recuperado do sinal
 - Uma campo de SYNC precede cada pacote, para sincronizar o *clock*

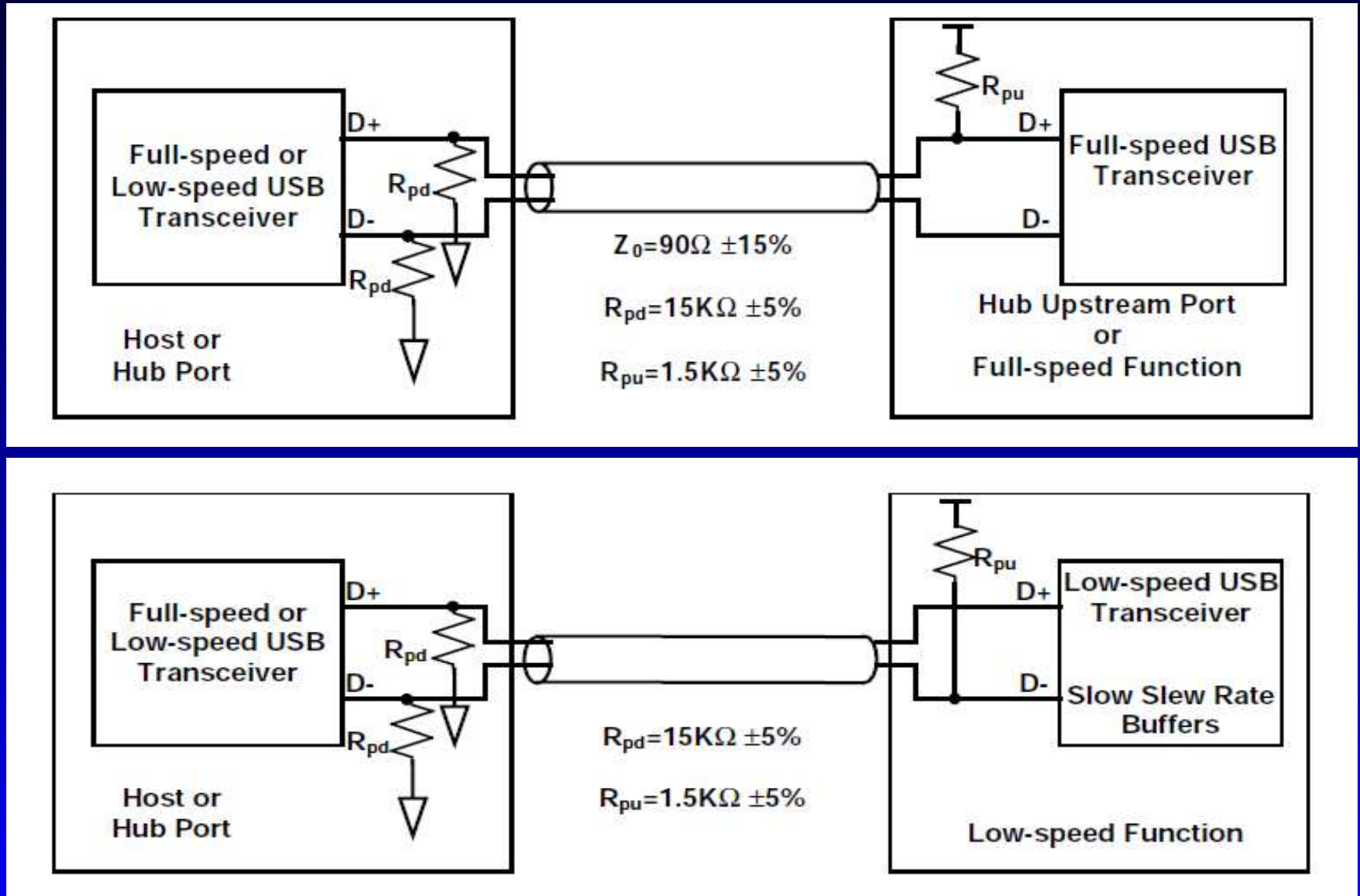


Interface Física

- São utilizados terminadores ativos em ambos os extremos do cabo
 - Detecção de conexão e desconexão de dispositivos
 - Diferenciação entre dispositivos *high/full-speed* e *low-speed*
- Todos os dispositivos possuem conexão *upstream*
- Conectores para *upstream* e *downstream* são diferentes

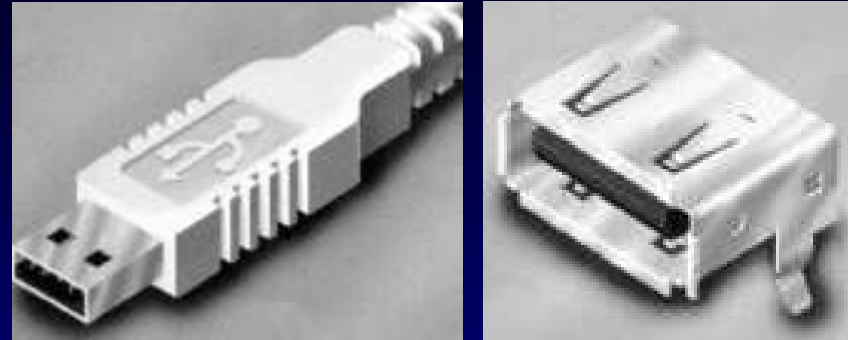


Detecção de Taxa de Transferência

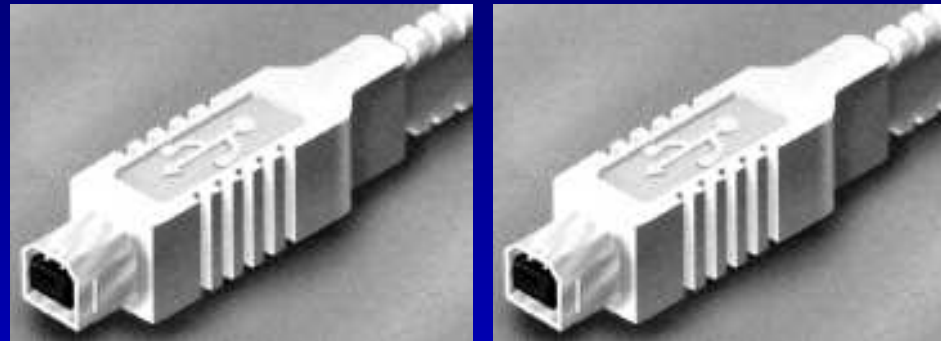


Conectores USB

- Tipo A, *Upstream*



- Tipo B, *Downstream*



Pino	cor	signal
1	vermelho	+VCC
2	branco	-Data
3	verde	+data
4	preto	-VCC



Software USB

- Driver de dispositivo USB
 - Solicita transferências USB para o driver USB
 - Faz a interface com o software cliente, tipicamente o sistema operacional
- Driver USB
 - Gerencia a comunicação USB
 - Gera transações USB a serem executadas como uma série de quadros de 1ms
- Driver do controlador do host
 - Escalona as transações para serem difundidas no barramento USB



Protocolo

- O barramento funciona por *polling*
- O controlador do *host* inicia todas as transferências de dados
- Cada transação no barramento envolve a transmissão de três pacotes
 - *token*
 - Enviado pelo controlador do *host*
 - Tipo e direção da transação e endereço do dispositivo
 - dados
 - reconhecimento
- Algumas transações envolvem a transmissão de quatro pacotes



Pipes

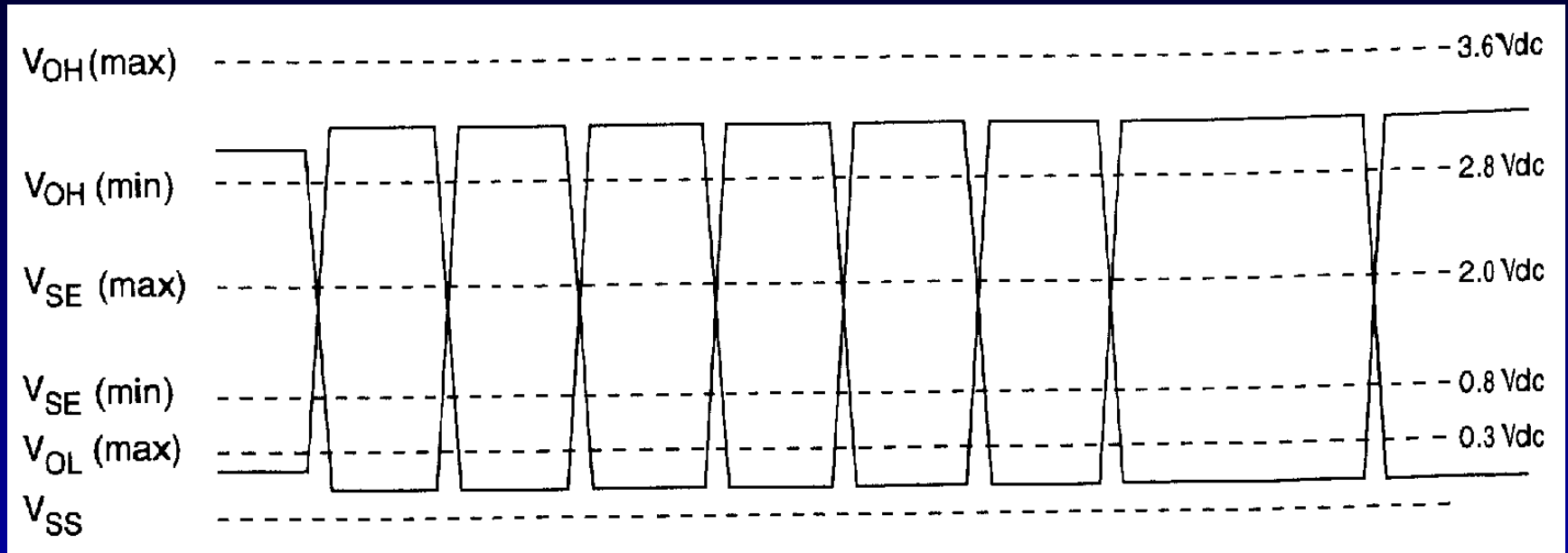
- Transferência de dados, que pode ser de dois tipos
- *Stream*, sem estrutura definida
- Mensagem, com estrutura definida
- Possuem associações de
 - Largura de banda
 - Tipo de serviço
 - Direcionalidade
 - Tamanho de *buffer*
- Usualmente são criadas quando da configuração do dispositivo



Robustez

- É utilizada sinalização diferencial e blindagem
- Os pacotes possuem CRC separados para os campos de controle e de dados
 - 100% de cobertura em erros de bit simples e duplos
- O protocolo realiza recuperação, utilizando *timeouts* para pacotes perdidos
 - O controlador tenta retransmissões 3 vezes antes de reportar erro para a aplicação

Níveis de Sinais



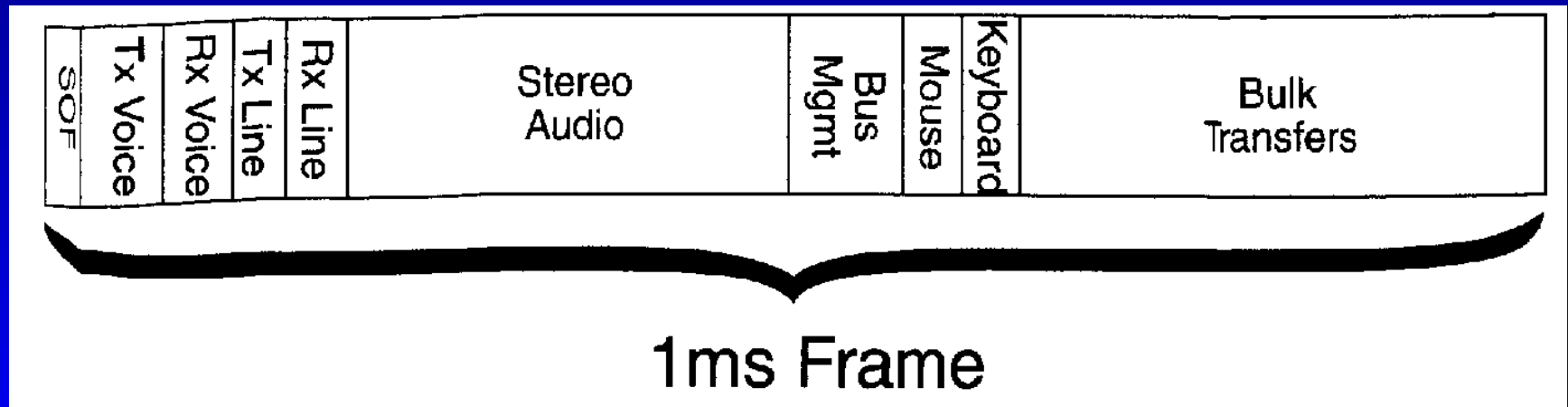
Tipos de Transferências de Dados



- Transferência isócrona
 - Transferências a uma taxa constante
- Transferência em lote
 - Transferências sem exigências de taxa
- Transferência de Interrupção
 - Usadas para consultar os dispositivos sobre interrupções pendentes
- Transferência de Controle
 - Configuração dos dispositivos USB

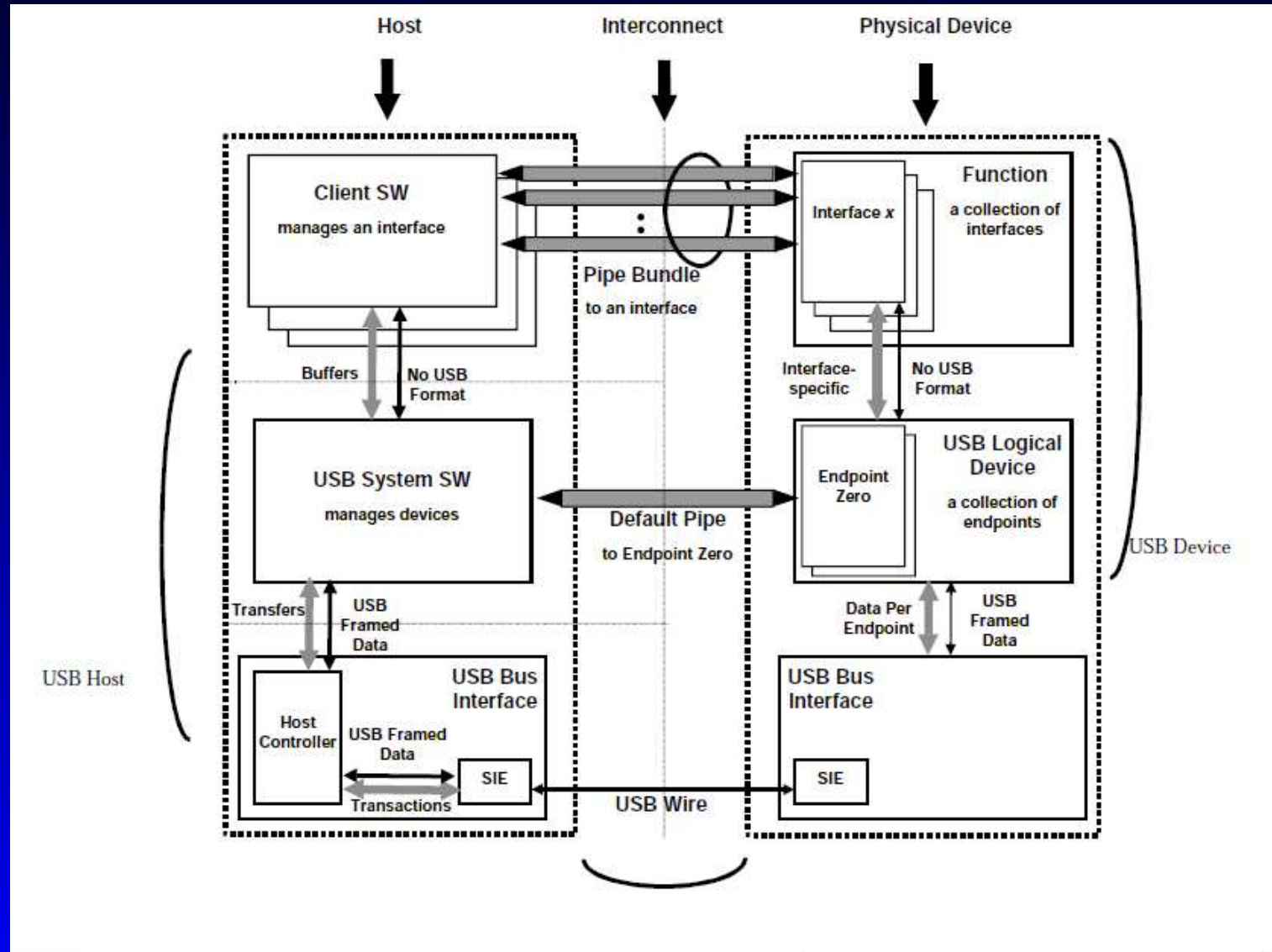
Frames

- A comunicação no barramento é baseada em frames de 1 ms
- Cada dispositivo solicita que uma fração da largura de banda seja alocada nestes quadros
- A alocação depende da necessidade do dispositivo e é feita durante a configuração
- Se não for possível garantir a largura de banda necessária o dispositivo não é configurado





Visão Completa





Implementação

- *Open Host Controller Interface (OHCI)*
 - *Scatter/gather bus master*
 - Minimiza o uso da CPU
 - Implementa transferências isócronas
- *Universal Host Controller Interface (UHCI)*
 - Minimiza a complexidade do *hardware*
 - Utiliza a CPU para controlar o barramento
- *Extended Host Controller Interface (EHCI)*
 - USB 2.0
 - Funciona em paralelo com os anteriores



EHCI

