

BLOQUE III.

Nivel físico

INTERFACES DE NIVEL FÍSICO (II)



Contenidos

1. Introducción
2. Medios guiados
 1. Pares trenzados
 2. Cable coaxial
 3. Fibra óptica
3. Medios no guiados
 1. Sistemas de microondas terrestres
 2. Sistemas de microondas satélite
 3. Ondas de radio
 4. Infrarrojos

Contenidos

4. Sistemas de cableado estructurado
 1. Introducción
 2. Topologías
 3. Subsistemas de cableado estructurado
 4. Normativa
 5. Concentradores

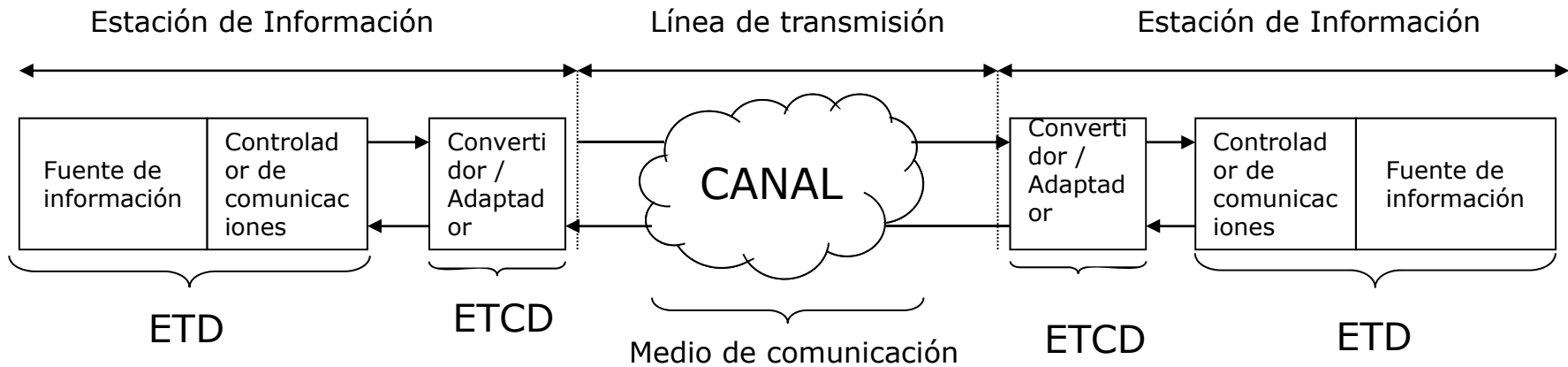
5. Interfaces de nivel físico
 1. RS232
 2. USB

Contenidos

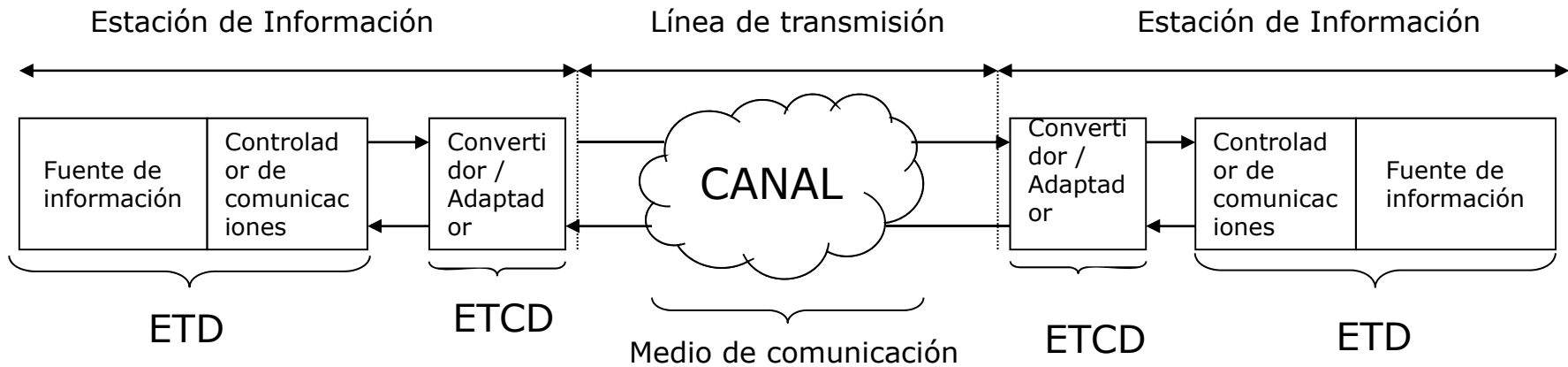
4. Sistemas de cableado estructurado
 1. Introducción
 2. Topologías
 3. Subsistemas de cableado estructurado
 4. Normativa
 5. Concentradores

5. Interfaces de nivel físico
 1. RS232
 2. USB

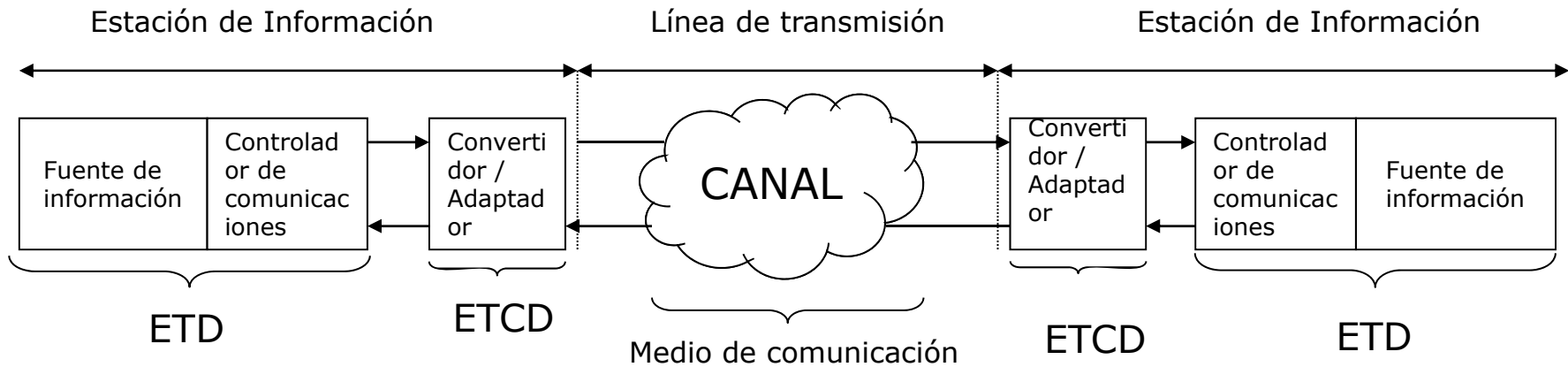
5.2 USB



5.2 USB



5.2 USB



5.2 USB

- **¿Por qué aparece la interfaz USB?**
 - Necesitamos *facilidad de uso*
 - Las interfaces de entrada/salida de los PC no eran plug-and-play
 - Cada periférico tenía su propia interfaz
 - Necesitamos *rapidez*
 - Que no sea un cuello de botella
 - Necesitamos *confiabilidad*
 - Que no ocurran muchos errores y si ocurren que se solucionen automáticamente
 - Necesitamos *versatilidad*
 - Distintos periféricos puedan usar la misma interfaz
 - Necesitamos heterogeneidad
 - Que pueda funcionar con Windows o con otros S.O.
 - Necesitamos *expansión de puertos*
 - Poder agregar periféricos al PC pasa por disponer de suficientes puertos en el equipo
 - Necesitamos puertos bidireccionales, de bajo coste, de velocidad media (o en algunos casos alta),...
 - Poco consumo energético

5.2 USB

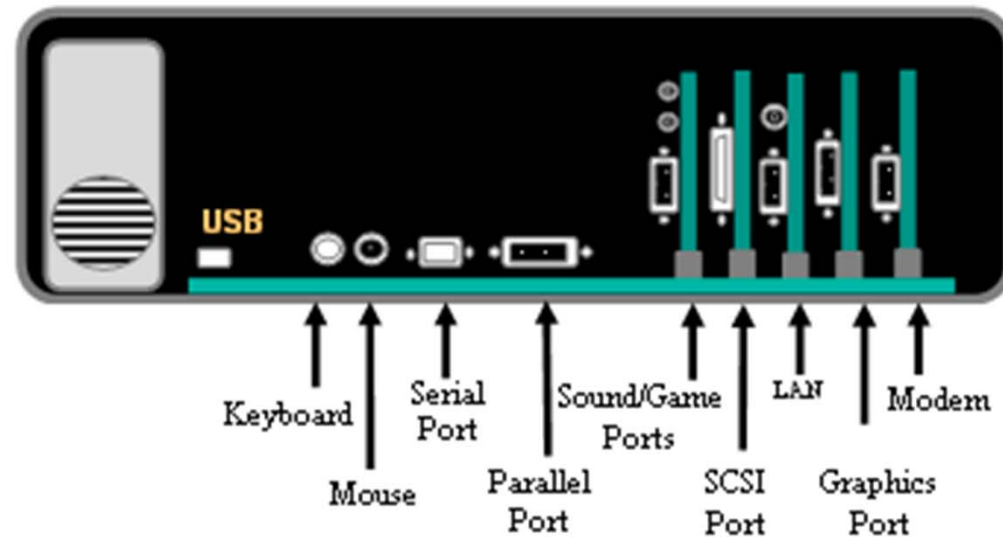
- ¿Por qué aparece la interfaz USB?



5.2 USB

□ ¿Por qué aparece la interfaz USB?

USB Ahora: Se introdujo inicialmente como un conector adicional para nuevas aplicaciones.



USB Futuro: PC evoluciona hacia un equipo más simple y sencillo de utilizar.



5.2 USB

BENEFICIOS PARA USUARIO

Facilidad de uso

Permite conectar a través de la misma interfaz infinidad de periféricos.

Ofrece una configuración automática. Al conectar dispositivo, S.O. detecta el periférico y carga controlador software apropiado. No necesidad de reiniciar equipo.

Fácil de conectar, no es necesario abrir el equipo y añadir tarjetas de expansión por cada periférico. PC típico 4 puertos USB (expandir con hubs USB).

Cableado sencillo, imposible conectar USB de forma errónea. Longitud máxima 5 metros (con hubs 30m). Conectores pequeños.

Se puede conectar/desconectar cuando queramos.

Sin opciones de configuración para el usuario.

(A veces) no requieren fuente de energía, propia interfaz incluye hilos para alimentación (+5V) a través de propio equipo.

5.2 USB

BENEFICIOS PARA USUARIO

Velocidad

Se puede usar a tres velocidades: alta (480Mbps), completa (12Mbps) y baja (1,5Mbps). Las tres soportadas en equipos actuales.

Velocidades (velocidades de bus): tasa a la que los datos viajan por el bus. Además de datos, el bus lleva señales de estado, de control y de comprobación de errores. Si varios periféricos comparten bus, velocidades más bajas.

Baja velocidad para el ratón: necesita cables flexibles. Los cables *low-speed* no necesitan estar trenzados ni apantallados ⇒ +flexibles y +económicos.

Velocidad media para reemplazar periféricos típicos (RS232 y paralelo) ofreciendo velocidades iguales o mejores.

Alta velocidad con USB 2.0.

5.2 USB

BENEFICIOS PARA USUARIO

Fiabilidad

Gracias al hardware y a los protocolos de transferencia de datos. Especificaciones hardware (controladores, receptores y cables) aseguran una interfaz con ruido↓. Protocolo USB permite detectar errores en datos recibidos y notificar a emisor para reenvío. Detección, notificación y reenvío es hardware (sin intervención del usuario).

Coste

Más compleja que predecesores pero igual o más económica.

Bajo consumo

Circuitaría de alimentación ⇒ desconectarlos cuando no están en uso.

Comunicaciones inalámbricas

Existen opciones para dispositivos inalámbricos que usan puertos USB para comunicarse con PC

5.2 USB

BENEFICIOS PARA DESARROLLADORES

Además de los anteriores:

- Flexibilidad de protocolos de USB
- Soporte a diferentes S. O.
- Ayuda del *USB Implementers Forum* (www.usb.org)

5.2 USB

LIMITACIONES

Velocidad

Otras interfaces (IEEE1394-b) → 3,2Gbps

Distancia

Otras interfaces (RS232) permite mayores distancias

Peer-to-peer

Comunicación USB entre host y periférico ⇒ periféricos no pueden hablar entre sí directamente.

Solución parcial: USB On-the-go ⇒ dispositivo puede funcionar como periférico o como host con capacidades limitadas que se pueden comunicar con otros periféricos

Broadcast

USB no proporciona ningún método para hacer *broadcast* a varios dispositivos en el bus (se debe hacer individualmente)

5.2 USB

□ Evolución

■ USB original (v1, 1996)

- v1.1 1998, añade transferencia *Interrupt Out*
- Primeros PC compatibles → Windows95 (uso limitado, no periféricos)
- Con Windows98 empiezan a aparecer más periféricos → USB más popular
- A partir de Windows98 todas las versiones dan soporte a USB

■ USB (v2.0, 2000)

- ↑ Popularidad ⇒ necesidad de velocidad mayor
- Alta velocidad (*high speed*) de 480 Mbps (impresoras, escáneres, video, ...)
- Compatible con versiones anteriores (mismos cables, ...)
- Funcionamiento *high speed* ⇒ dispositivo compatible se debe conectar a host con puerto USB 2.0 (idem hub intermedios)

5.2 USB

□ Evolución

■ USB On-the-go (v1, 2001)

- Interfaz más utilizado, ¿poder conectar periféricos entre sí?
- Periférico tiene funciones limitadas de host y permite comunicación entre periféricos
- Otras versiones v1.3 2006

■ Wireless USB (v1, 2006)

- Permite comunicación inalámbrica con dispositivos periféricos llegando a velocidades de 480 Mbps
- v1.1 2008
 - HP, Intel Corporation, LSI, Logic, Microsoft Corporation, NEC Corporation, NXP Semiconductors, and Samsung Electronics
 - 480 Mbps a 3 metros

ENTREGABLE

5.2 USB

□ Objetivos de diseño del USB (Std.)

1. Facilidad de uso
2. Solución de bajo coste que soporte velocidades de hasta 480 Mbps (2.0)
3. Soporte completo para aplicaciones en tiempo real de voz, audio y video
4. Protocolo flexible para transferencias de datos isócronas y envío asíncrono de mensajes
5. Fácil integración
6. Rápida difusión a través de un estándar
7. Permitir nuevos periféricos
8. Compatibilidad con versiones anteriores (2.0)

5.2 USB

- ❑ ¿Con qué aplicaciones se puede emplear?
- ❑ Se emplea en transmisiones con velocidades de hasta 480Mbps (USB 2.0 Standard, 2000)

PRESTACIONES

Low-Speed

- Dispositivos interactivos
- 10 kbps – 100 kbps

Full-Speed

- Teléfono, Audio, Video comprimido
- 500 kbps – 10 Mbps

High-Speed (USB 2.0)

- Video, almacenamiento
- 25 Mbps – 480 Mbps

APLICACIONES

Teclados, ratones, periféricos de juegos, periféricos de realidad virtual

Servicio de telefonía clásica, audio, micrófonos, ...

Video, almacenamiento, imágenes, ...

ATRIBUTOS

Menor coste
Fácil de usar
Inserción/extracción dinámica
Múltiples periféricos

Menor coste
Fácil de usar
Inserción/extracción dinámica
Múltiples periféricos
Ancho de banda garantizado
Latencia garantizada

Menor coste
Fácil de usar
Inserción/extracción dinámica
Múltiples periféricos
Ancho de banda garantizado
Latencia garantizada
Gran ancho de banda

5.2 USB

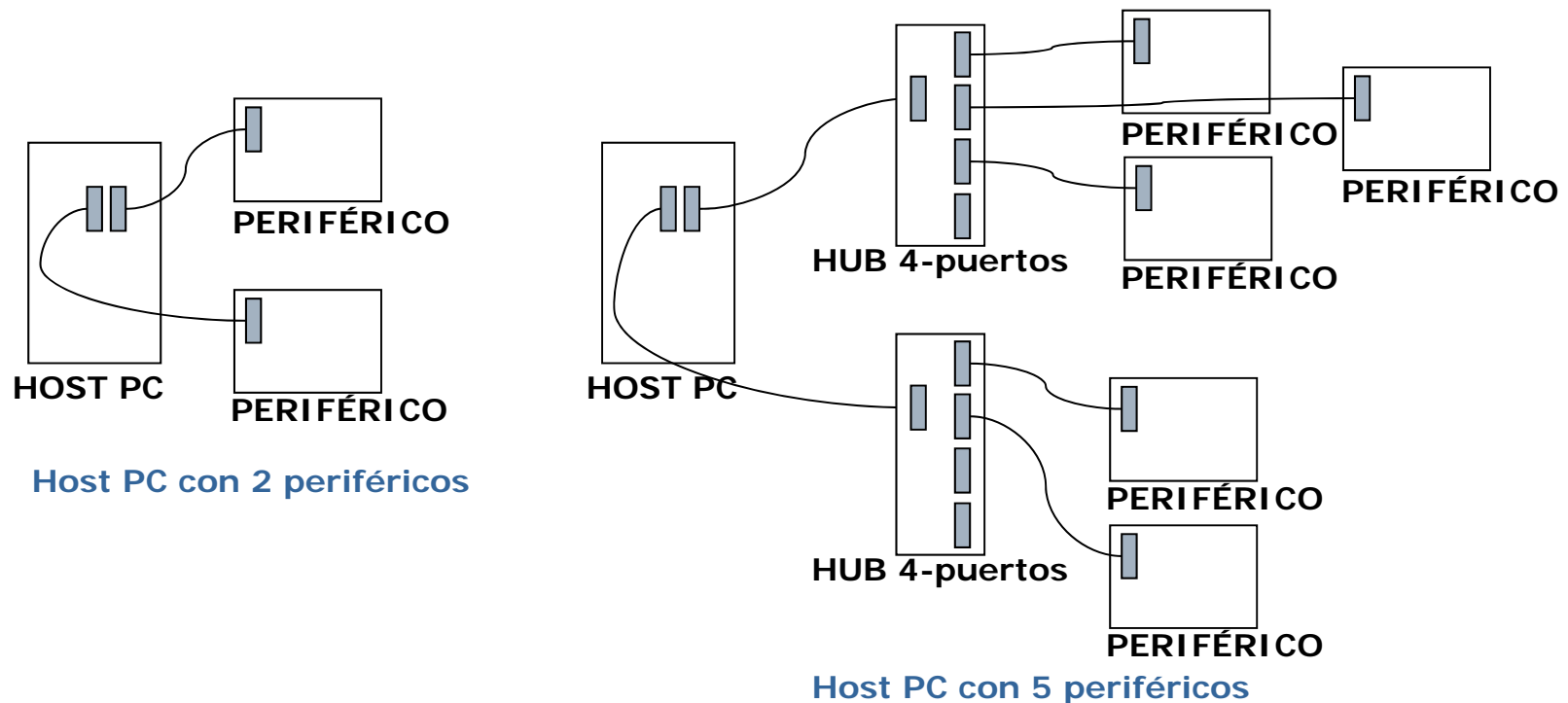
□ Topología

- **Host Controller**, interfaz USB-host
 - Sólo puede haber un host (PC, *host controller+root hub*) en cada sistema (en cada estrella extendida)
 - Da formato a los datos que se van a transmitir en el bus
 - Traduce datos recibidos a un formato entendible por S.O.
 - Tareas de gestión de comunicaciones en el bus
- **Root Hub**
 - Se integra en el *host controller* para permitir conectar más de un dispositivo USB a la vez (tiene uno o más conectores)
 - Junto con *host controller* detecta nuevos dispositivos conectados/desconectados
 - Lleva a cabo peticiones del *host controller*
 - Pasa los datos entre *host controller* y dispositivos
- **Dispositivos**
 - Periféricos o hub adicionales que se conectan al bus
 - Hub tiene uno o más puertos para conectar dispositivos

5.2 USB

□ Topología

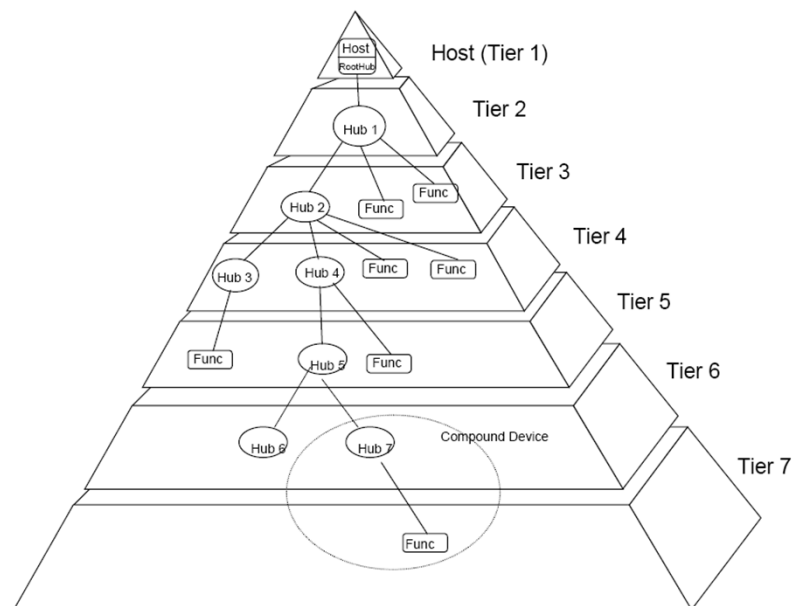
- Sólo un dispositivo se puede comunicar con *host controller* cada vez
- Para aumentar disponibilidad host podría tener varios *host controller*



5.2 USB

□ Topología

- Es una estrella extendida
- En el centro de cada estrella tenemos un hub
- Cada punto de la estrella es un dispositivo USB que se conecta a un puerto USB
 - En cada estrella 2, 4 o 7 puertos
 - Número máximo de estrellas por limitaciones de tiempo: 7



5.2 USB

- División de tareas entre host y dispositivo
 - Host se encarga de gestión de comunicaciones
 - Dispositivo suficientemente inteligente para responder a comunicación y eventos del bus

TAREAS DEL HOST

PC necesita hardware y software que permitan al PC funcionar como host USB

El host a cargo del bus:

- ¿Qué dispositivos hay en el bus?
- Características/capacidades de cada uno
- No inanición (enviar/recibir)

HW: *host controller + root hub*

SW: S.O. mecanismos para que controladores se comuniquen con HW

5.2 USB

TAREAS DEL HOST

Detectar Dispositivos

- Encendido → cada hub indica a host que tiene conectados dispositivos
- *Enumeration* ≡ host asigna dirección a dispositivo y les solicita información
- Tras encendido → si conexión/ desconexión de dispositivo host asigna/elimina dirección

Gestionar flujo de datos en bus

- Múltiples periféricos quieren enviar de forma simultánea
- *Host controller* → divide tiempo en segmentos (tramas, microtramas) y asigna a cada dispositivo una porción de trama (microtrama)
- Transferencias deben ocurrir a tasa específica, garantizado su espacio de tiempo en cada trama
- *Enumeration* ≡ controlador de dispositivo pide BW que necesita para transferencia con restricción temporal. Si BW no garantizado → no comunicación. Transferencias sin restricciones temporales resto del tiempo de la trama, esperan si bus ocupado

5.2 USB

TAREAS DEL HOST

Comprobación de errores

- Cuando transferencia de datos, host añade bits de comprobación de error
 - Al recibir datos, dispositivo comprueba si datos recibidos correctamente
 - Si fallo, dispositivo descarta datos y no los reconoce → host hará retransmisión
 - USB incluye tipo de datos donde no hay retransmisiones ¿por qué?
 - Host también comprueba datos que recibe de periférico
- Host también puede recibir indicación de dispositivo: no enviar/recibir datos
 - Notificar a aplicación

Proporcionar energía

- Además de hilos para datos ⇒ dos hilos para alimentación (5V, tierra) para proporcionar energía a dispositivos
- Host intenta conservar energía si es posible
- Algunos dispositivos su propia alimentación

5.2 USB

TAREAS DEL PERIFÉRICO

Detectar comunicaciones para el dispositivo

- Dispositivo monitoriza dirección de dispositivo contenida en comunicaciones del bus
- Si no se corresponde con dirección almacenada → ignorar
- Si coincide → dispositivo almacena datos en buffer y genera señal interrupción (datos recibidos)

Responder a preguntas estándar

- Cuando dispositivo enciende/conecta al host → responder a preguntas de *enumeration*
- Especificación define 11 preguntas estándar (no es necesaria respuesta siempre pero hay que responder)

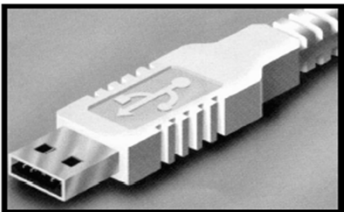
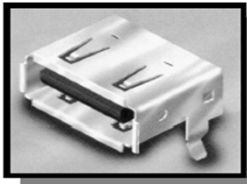
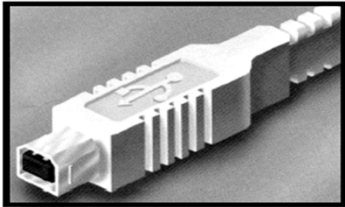
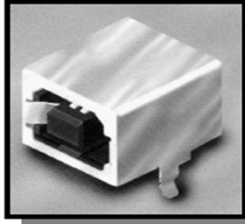
Otras:

- Comprobación de errores
- Gestión de energía
- Intercambiar datos con host
- Tareas propias de cada periférico

5.2 USB

1 Especificaciones mecánicas

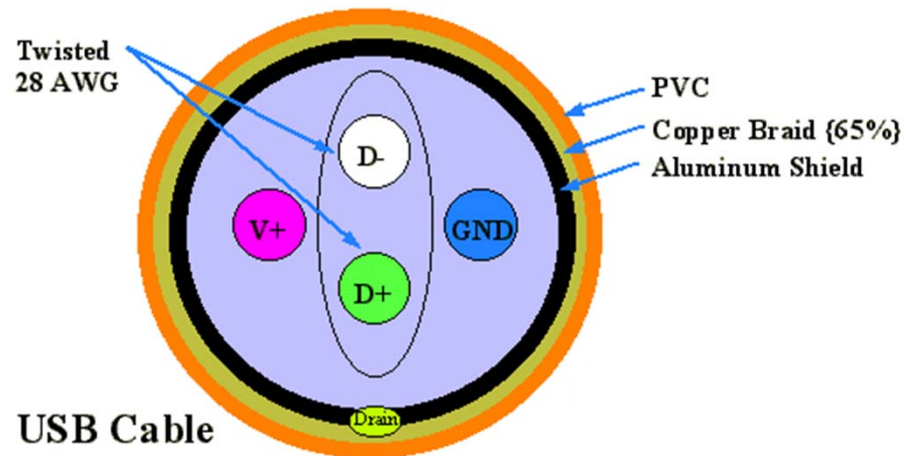
- Icono USB de los conectores para identificar mediante el tacto cómo conectar
- Longitud máxima 5 metros

Conectores Serie "A"	Conectores Serie "B"
<p>◆ Conectores Serie "A" siempre son <i>upstream</i> hacia el Host</p>  <p>Conector "A" (desde el dispositivo USB)</p> <p>Receptáculo "A" (salida <i>downstream</i> desde el Host o desde el hub)</p> 	<p>◆ Conectores Serie "B" siempre son <i>downstream</i> hacia el dispositivo USB</p>  <p>Conector "B" (desde el Host)</p> <p>Receptáculo "A" (entrada <i>downstream</i> hacia el dispositivo USB o Hub)</p> 

5.2 USB

1 Especificaciones mecánicas

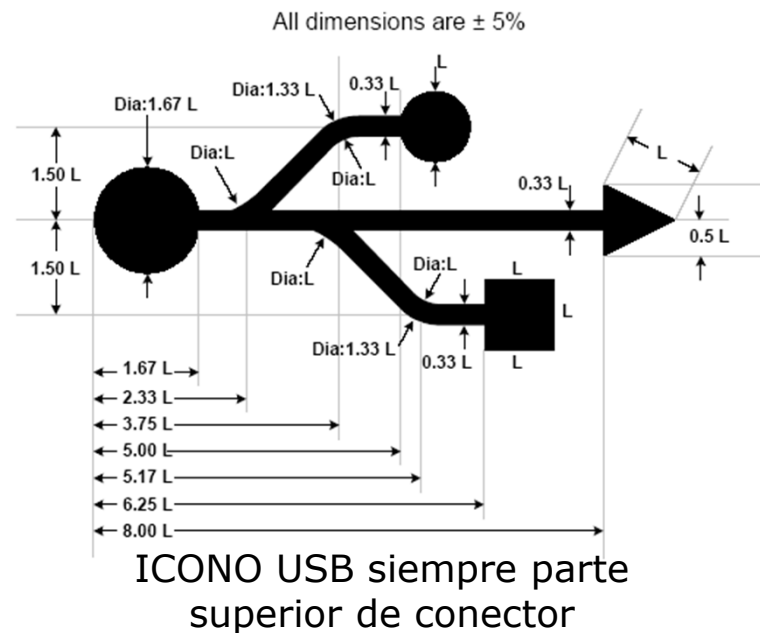
- Cables de 4 hilos
 - Dos para datos (trenzados si es *high speed*) de 28 AWG con cubierta protectora de poliéster metalizado con aluminio
 - Dos para la alimentación no trenzados de 28 a 20 AWG
 - Cubierta de PVC



5.2 USB

1 Especificaciones mecánicas

- Dos tipos de cables
 - *High/full speed*, en un extremo es conector serie A y en el otro extremo en conector serie B (*detachable*)
 - *High/full speed* o *Low speed*, en un extremo es conector serie A y en el otro específico de vendedor

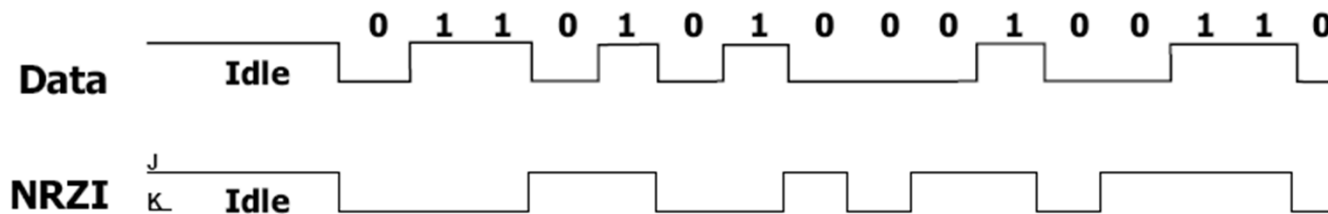


5.2 USB

2 Especificaciones eléctricas

□ Codificación NRZI (bit stuffing)

- En forma balanceada \Rightarrow si se transmite un 1 \Rightarrow en una línea van +2,5V y en la otra -2,5 V)



"0" Cambio
"1" No hay cambio
Bit Stuffing \Rightarrow Cada 6 "1" se manda un "0"

5.2 USB

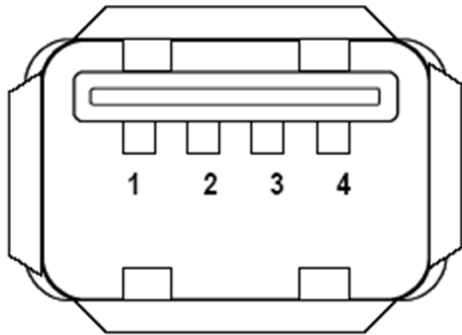
2 Especificaciones eléctricas

- Retardo de propagación máximo permitido es 30ns (26+3+1)
- Atenuación

Frequency (MHz)	Attenuation (maximum) dB/cable
0.064	0.08
0.256	0.11
0.512	0.13
0.772	0.15
1.000	0.20
4.000	0.39
8.000	0.57
12.000	0.67
24.000	0.95
48.000	1.35
96.000	1.9
200.00	3.2
400.00	5.8

5.2 USB

3 Especificaciones funcionales



PIN	Señal	Descripción
1	VBUS	Alimentación +5 V
2	D-	Transmisión y Recepción Diferencial -
3	D+	Transmisión y Recepción Diferencial +
4	Tierra	Masa de referencia alimentación

5.2 USB

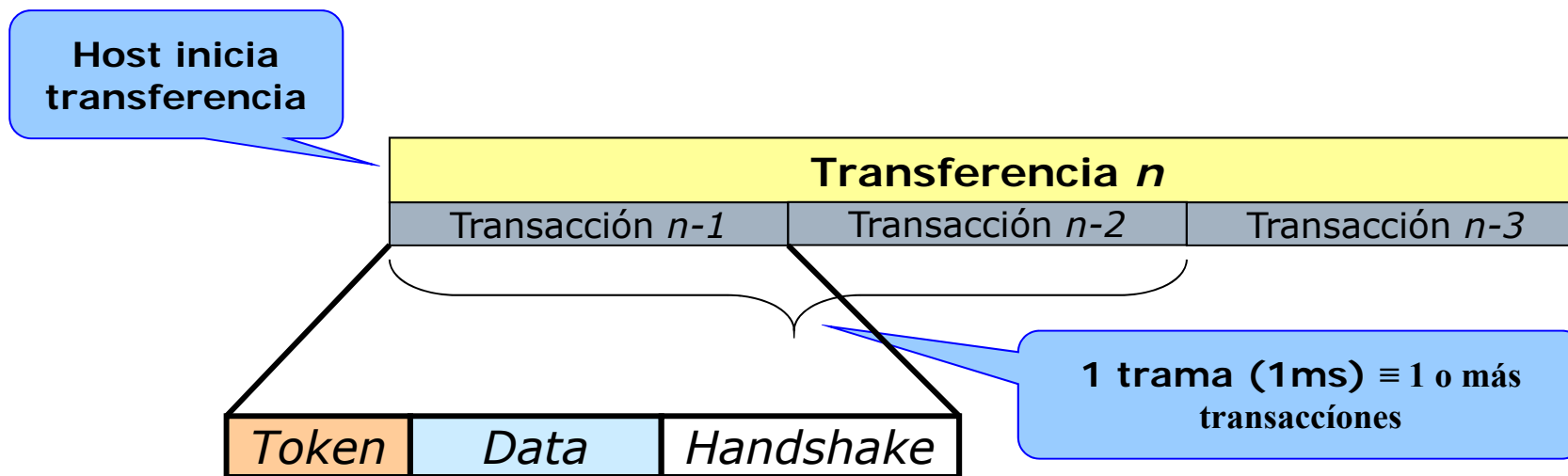
4 Especificaciones de procedimiento

- Los dispositivos USB se configuran automáticamente tan pronto como se conectan físicamente
- Normalmente se comunica un dispositivo con un PC (host) ⇒ host controla la comunicación:
 - Host envía datos ≡ **Transacción OUT**
 - Host solicita datos ≡ **Transacción IN**
 - Host especifica origen/destino de los datos con un ***Device Address*** y ***Endpoint Number***
 - Host planifica las transacciones y reserva recursos
- El host asigna una dirección única a cada dispositivo
- Comunicaciones half-duplex

5.2 USB

4 Especificaciones de procedimiento

- Cada **transacción** consiste en una serie de paquetes:
 - **Token**, define tipo de transacción
 - **Data**, lleva la información
 - **Handshake**, *feed-back* para saber si la transacción fue correcta.



5.2 USB

4 Especificaciones de procedimiento



Token Packet



SOF Token Packet



Data Packet



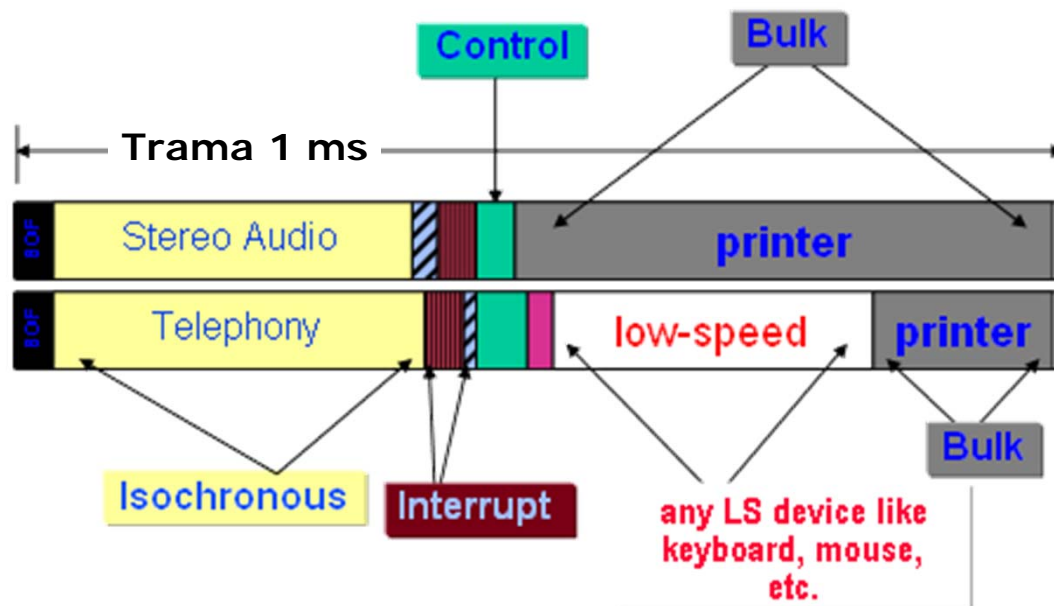
Handshake/
Low Speed Preamble

5.2 USB

4 Especificaciones de procedimiento

□ Cuatro tipos de transferencias:

- **Control** (bidireccional), mensajes de configuración.
- **Interrupt** (unidireccional), datos que envía el ratón, el teclado, etc.
- **Isochronous** (unidireccional), audio, telefonía,...
- **Bulk** (unidireccional), impresora, cámaras, etc...



5.2 USB

4 Especificaciones de procedimiento

□ Descriptores:

- Descriptores de Dispositivo \equiv ¿Quién eres? ¿Qué *driver* hay que cargar?
 - Versión del USB; Clase de Dispositivo/Subclase/Protocolo; ID Vendedor/ID Producto.
- Descriptores de Configuración \equiv ¿Qué recursos necesitas?
 - Bus Powered/ Self Powered; Current Consumption; Number of configurations/Number of Interfaces.
- Descriptores de Interfaz
- Descriptores de punto-final (*endpoint*)
- Descriptores String

**Descriptores
se almacenan
en ROM**

5.2 USB

4 Especificaciones de procedimiento

- Una vez el USB se conecta al host (*enumeration*):
 1. El host lee los 8 primeros bytes del descriptor de dispositivo
 2. El host asigna una dirección al dispositivo
 3. El host lee por completo el descriptor de dispositivo
 4. El host lee el descriptor de configuración (*interfaz y endpoint*)
 5. EL host envía el comando SET_CONFIGURATION
 6. El host carga el controlador (*driver*)

5.2 USB

- ¿Qué otras interfaces se emplean además de RS232 y USB?
 - Norma V.35 para transmisiones síncronas a 48kbps.
 - Norma X.21 para la transmisión en la red pública de datos.
 - Norma RS-449, distancias de hasta 1200 m, y hasta 24 Mbps, con señales balanceadas ó 40 kbps si no lo están.
 - En algunos casos se necesitan adaptadores de interfaz para DB-9 o RJ-45 en vez de DB-25.
 - Existen otras interfaces paralelo como el CENTRONICS que tiene 36 pines:
 - Transmisión es en paralelo
 - Se usa para la conexión de impresoras (LPR)
 - En general, los PC tienen un conector DB-25 y la impresora un conector CENTRONICS ⇨ cables especiales

Próximo día

BLOQUE IV. Nivel de enlace de datos