

# S8-Buses-del-Sistema

Elizabeth Lopez, Francisco Morales, Juan Murillo , Angel Falcon,  
Danny Tipan, Anderson Herrera, Sebastian Chicaiza, Jairo Angulo

2025-01-13

# Outline

- 1 Buses del Sistema (Sección)
- 2 Interconexión con Buses (Sección)
- 3 Buses del Sistema (Sección)
- 4 Interconexión punto a punto
- 5 Introducción QPI
- 6 QuickPath Interconnect (QPI)
- 7 PCI Express (E4, 11)
- 8 Introducción a PCI Express
- 9 Dispositivo complejo raíz
- 10 Arquitectura
- 11 Capas del Protocolo
- 12 Arquitectura PCIe
- 13 Conclusiones
- 14 Referencias



## Módulos:

- **Memoria:** constituido por N palabras de igual longitud. Se pueden realizar las operaciones Read(Leer) y Write(Escribir). La posición de memoria se especifica mediante una dirección.
- **Módulo de E/S:** se encarga de controlar los dispositivos externos enlazados a los puertos, donde se les asignara una dirección M. Controla los datos de salida y entrada. Realiza las operaciones de lectura y escritura. Envía señales de interrupción.
- **Procesador:** lee instrucciones y datos. Escribe datos después de procesarlos y controla el funcionamiento del sistema. Puede recibir señales de interrupción.

## Intercambios de Datos:

- **Memoria a procesador:** el procesador lee información desde la memoria.
- **Procesador a memoria:** el procesador escribe un dato en la memoria.
- **E/S a procesador:** el procesador lee datos de un dispositivo de E/S.
- **Procesador a E/S:** el procesador envía datos al dispositivo de E/S.
- **Memoria a E/S - E/S a Memoria:** ambos intercambian datos directamente.

# Interconexión con Buses (E1, 7, 99)

- Los buses son caminos de comunicación entre dos o más dispositivos con la habilidad de transmitir señales hacia los demás o recibir las señales emitidas.
- Solo un dispositivo puede emitir la señal en un periodo de tiempo. Si ambos transmiten la señal, esta podría solaparse y distorsionarse.
- Los caminos o líneas del bus transmiten señales binarias ya sea a travez de una sola línea o de varias de manera paralela.
- Existen diferentes tipos de buses para la comunicacion de diversos componentes. El que trabaja con los módulos elementales se denomina (System bus).



- ¿Qué son las líneas de datos?
  - Transporte
  - Datos
  - Memoria -> Procesador



- ¿Qué son las líneas de dirección?
  - Ubicación
  - Memoria
  - Puertos de E/S
  - Anchura del bus

- ¿Qué son las líneas de control?
  - Control
  - Señales
  - Escritura
  - Lectura

# Jerarquía de Buses Múltiples (E2, 7)

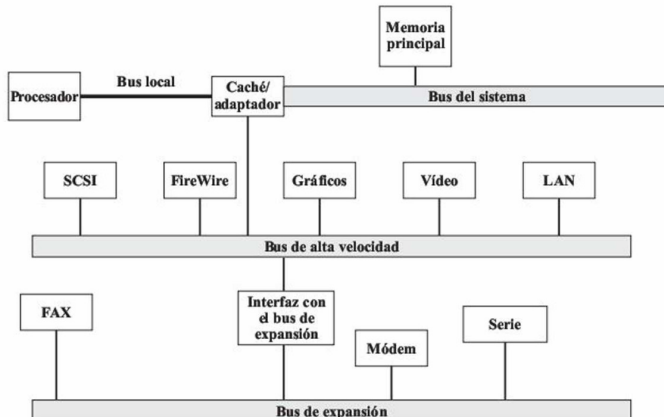
Si se conecta un gran número de dispositivos al bus, las prestaciones pueden disminuir. Hay dos causas principales:

- 1 Mayor retarde de propagación. Este retardo determina el tiempo que necesitan los dispositivos para coordinarse en el uso del bus.
- 2 Posible cuello de botella. Este problema se puede resolver en alguna medida incrementando la velocidad a la que el bus puede transferir los datos y utilizando buses más anchos (por ejemplo incrementando el bus de datos de 32 a 64 bits)



# Jerarquía de buses múltiples

La respuesta común a esta situación, por parte de la industria, ha sido proponer un bus de alta velocidad que está estrechamente integrado con el resto del sistema, y requiere solo un adaptador (bridge) entre el bus del procesador y el bus de alta velocidad. En algunas ocasiones, esta disposición es conocida como arquitectura de entreplanta (mezzanine architecture).



# Jerarquía de buses múltiples

La ventaja de esta organización es que el bus de alta velocidad acerca al procesador los dispositivos que exigen prestaciones elevadas y al mismo tiempo es independiente del procesador. Así, se pueden tolerar las diferencias de velocidad entre el procesador y el bus de altas prestaciones y las variaciones en la definición de las líneas de los buses. Los cambios en la arquitectura del procesador no afectan al bus de alta velocidad, y viceversa.

# Elementos de Diseño de un Bus (E2, 7)

**Tipo**

Dedicado

Multiplexado

**Método de arbitraje**

Centralizado

Distribuido

**Temporización**

Síncrono

Asíncrono

**Anchura del bus**

Dirección

Datos

**Tipo de transferencia de datos**

Lectura

Escritura

Lectura-modificación-escritura

Lectura-después de-escritura

Bloque



# Elementos de Diseño de un Bus (E2, 7)

## 1. Tipo

- **Dedicado:** El bus está reservado para una única función o dispositivo, asegurando una comunicación directa y sin interferencias.
- **Multiplexado:** Se comparte el mismo bus para diferentes tipos de datos o señales (por ejemplo, datos y direcciones), utilizando mecanismos de temporización o control para alternar entre usos.

## 2. Método de arbitraje

- **Centralizado:** Existe un único controlador que decide cuál dispositivo puede usar el bus en cada momento. Es más sencillo de implementar pero puede ser un punto único de fallo.
- **Distribuido:** La responsabilidad del control del bus se reparte entre los dispositivos conectados, permitiendo un diseño más robusto y descentralizado.

## 3. Temporización

- **Síncrono:** Las transferencias de datos se realizan basándose en un reloj común. Es más rápido, pero menos flexible, ya que todos los dispositivos deben funcionar a la misma velocidad.
- **Asíncrono:** No requiere un reloj común, permitiendo que cada dispositivo opere a su propia velocidad. Es más flexible, aunque más complejo de coordinar.

## 4. Anchura del bus

- **Dirección:** Define el número de líneas disponibles para las direcciones, lo que determina la cantidad máxima de dispositivos o memoria que se pueden direccionar.
- **Datos:** Indica el número de bits que se pueden transferir simultáneamente, influyendo en la velocidad del sistema (por ejemplo, un bus de datos de 32 bits permite transferir 32 bits en paralelo).

## 5. Tipo de transferencia de datos

- **Lectura:** Se transfieren datos desde un dispositivo (por ejemplo, memoria) al bus para que sean utilizados por otro dispositivo.
- **Escritura:** Se envían datos desde un dispositivo al bus para almacenarlos en otro (como en memoria).
- **Lectura-modificación-escritura:** Un ciclo donde se lee un dato, se modifica y luego se escribe de nuevo en memoria o dispositivo.
- **Lectura-después-de-escritura:** Se escriben los datos en un dispositivo y luego se leen para verificar la operación.
- **Bloque:** Se transfieren varios datos consecutivos en una sola operación, optimizando la eficiencia para grandes volúmenes de datos.



- Elimina cuellos de botella asociados con buses compartidos.
- Mejora la escalabilidad en sistemas multiprocesador, ya que cada procesador puede comunicarse con otros sin interferencias.
- Aumenta la velocidad de transferencia y reduce la latencia.

Fue creado para solucionar los problemas de los buses compartidos, ofreciendo una conexión directa y eficiente entre los componentes. Este tipo de interconexión mejora el rendimiento al permitir una comunicación más rápida y efectiva entre los procesadores y otros dispositivos, sin las restricciones de los buses.

# Características QPI:

- Múltiples conexiones directas
- Arquitectura de protocolo en capas
- Transferencia de datos en paquetes

# Características de QPI

- **Múltiples conexiones directas:**

- Cada componente (como el procesador, la memoria o los dispositivos de entrada/salida) se conecta directamente con otros componentes de manera individual, sin tener que compartir el mismo canal.
- Al tener conexiones directas, cada componente puede enviar y recibir datos sin esperar turno, lo que hace que todo funcione de manera más rápida y eficiente.

- **Arquitectura de protocolo en capas:**

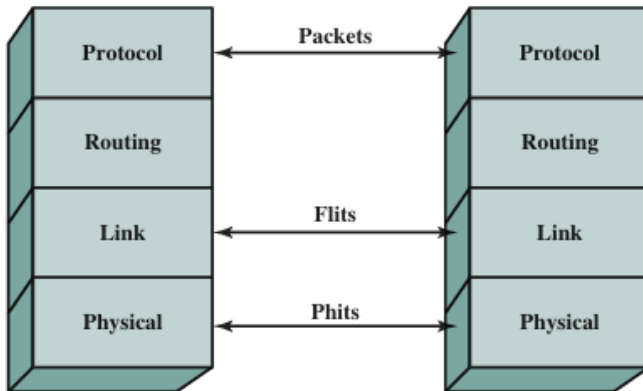
- Se usan protocolos como TCP/IP para organizar y manejar la comunicación. En lugar de enviar un mensaje de una sola forma simple, se utilizan diferentes pasos o etapas para asegurar que el mensaje llegue correctamente.

- **Transferencia en paquetes:**

- Los datos no se envían de manera continua, sino que se dividen en paquetes.
- Cada paquete contiene una parte de los datos y también incluye información adicional, como encabezados de control para saber a dónde deben ir los datos y códigos de control de errores para asegurarse de que los datos no se pierdan o se dañen durante el envío.



# QPI una arquitectura de protocolo de cuatro capas



# Arquitectura de protocolo QPI

## ● Capa física:

- Está formada por 84 enlaces individuales, cada camino de datos consta de un par de cables, llamados "carriles", que transmiten un bit a la vez.
- Hay 20 carriles en cada dirección: una para enviar datos y otra para recibir.
- Cada conjunto de 20 bits que se transmite se llama "phit", con una velocidad de transferencia de 6.4 giga transferencias por segundo (GT/s).

## ● Capa de enlace:

- Realiza dos funciones clave: control de flujo y control de errores. Estas se aplican a cada "flit"(unidad de control de flujo).
- Cada flit tiene una carga útil de 72 bits, que contiene los datos o mensajes.
- Los flits de datos transportan los bits reales entre los procesadores y el controlador de entrada/salida.
- Los flits de mensaje se utilizan para funciones como el control de flujo y el control de errores.
- El control de flujo asegura que el transmisor no envíe datos más rápido de lo que el receptor puede procesar.

# ¿Qué es PCI Express?

PCI Express representa la evolución natural del bus PCI tradicional. Esta tecnología surgió como respuesta a las crecientes demandas de velocidad y eficiencia en la transferencia de datos. A diferencia de su predecesor, PCIe implementa un esquema de interconexión punto a punto, abandonando la arquitectura de bus compartido para ofrecer mayor rendimiento y flexibilidad.

La necesidad de PCIe surgió cuando los esquemas basados en bus tradicionales alcanzaron sus límites de capacidad. Los dispositivos modernos, como las tarjetas gráficas y las unidades de almacenamiento de alta velocidad, demandaban un nuevo estándar capaz de manejar mayores velocidades de transferencia y múltiples flujos de datos simultáneos.

En el corazón de la arquitectura PCIe encontramos el Complejo Raíz, un componente crucial que actúa como puente entre el procesador, la memoria y el tejido de conmutación PCIe. Este elemento no solo gestiona las conexiones, sino que también armoniza las diferentes velocidades de transferencia entre los componentes del sistema.



# Dispositivos que implementan PCIe

**Conmutador:** El conmutador gestiona múltiples flujos PCIe.

**Punto final PCIe:** Dispositivo o controlador de E/S que implementa PCIe, como un conmutador Gigabit ethernet, un controlador gráfico o de vídeo, una interfaz de disco o un controlador de comunicaciones.

**Endpoint heredado:** La categoría de punto final heredado está pensada para diseños existentes que se han migrado a PCI , y permite comportamientos heredados como uso de espacio de E/S y transacciones bloqueadas. No se permite que los puntos finales PCI Express requieran el uso de espacio de E/S en tiempo de ejecución y no deben utilizar transacciones bloqueadas.

**Puente PCIe/PCI:** Permite conectar dispositivos PCI antiguos a sistemas basados en PCIe.

# La arquitectura del protocolo PCIe

La arquitectura del protocolo PCIe abarca las siguientes capas:

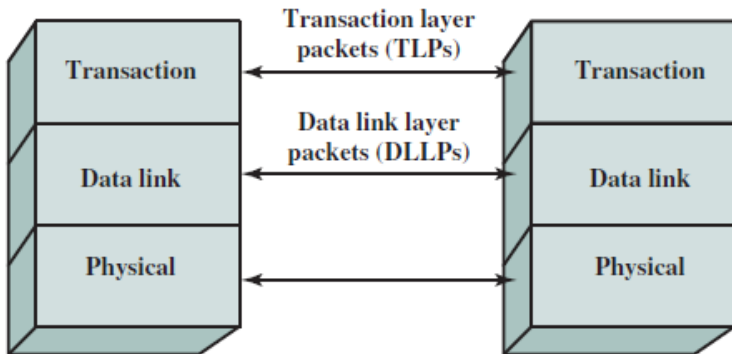
**Físico:** Consiste en los propios cables que transportan las señales, así como los circuitos y la lógica que soportan las funciones auxiliares necesarias en la transmisión y recepción de los 1 y los 0.

**Enlace de datos:** Es responsable de la transmisión fiable y del control de flujo. Los paquetes de datos generados y consumidos por la DLL se denominan paquetes de capa de enlace de datos (DLLP).

**Transacción:** Genera y consume paquetes de datos utilizados para implementar mecanismos de transferencia de datos de carga/almacenamiento y también gestiona el control de flujo de dichos paquetes entre los dos componentes de un enlace. Los paquetes de datos generados y consumidos por la TL se denominan paquetes de capa de transacción (TLP).



# La arquitectura del protocolo PCIe



La capa física de PCIe maneja la transmisión real de datos a través del medio físico. Emplea técnicas sofisticadas de codificación 128b/130b y scrambling para mantener la integridad de la señal y la sincronización. Estas técnicas permiten alcanzar velocidades de transferencia de hasta 16 GB/s, garantizando la fiabilidad en la transmisión de datos.

# Capa de transacción PCIe

La capa de transacciones (TL) recibe peticiones de lectura y escritura del software por encima de la TL y crea paquetes de peticiones para su transmisión a un destino a través de la capa de enlace. La mayoría de las transacciones utilizan una técnica de transacción dividida. Con la técnica de transacción dividida, la finalización está separada en el tiempo de la solicitud, en contraste con una operación de bus típica en la que ambos lados de una transacción deben estar disponibles para aprovechar y utilizar el bus. Entre la solicitud y la finalización, otro tráfico PCIe puede utilizar el enlace.

# Espacios de dirección

La TL admite cuatro espacios de dirección:

**Memoria:** El espacio de memoria incluye la memoria principal del sistema. También incluye los dispositivos de E/S PCIe. Ciertos rangos de direcciones de memoria se asignan a dispositivos de E/S.

**E/S:** Este espacio de direcciones se utiliza para dispositivos PCI heredados, con rangos de direcciones de memoria reservados utilizados para direccionar dispositivos de E/S heredados.

**Configuración:** Este espacio de direcciones permite a la TL leer/escribir los registros de configuración asociados a los dispositivos de E/S.

**Mensaje:** Este espacio de direcciones es para señales de control relacionadas con interrupciones, manejo de errores y gestión de energía.

# Tipos de transmisión

Espacio de direcciones	Tipo TLP
Memoria	Petición de lectura de memoria Solicitud de bloqueo de lectura de memoria Petición de escritura en memoria
E/S	Solicitud de lectura de E/S Solicitud de escritura de E/S
Configuración	Config Tipo 0 Petición de lectura Config Tipo 0 Petición de escritura Config Tipo 1 Petición de lectura Solicitud de escritura Config Tipo 1
Mensaje	Solicitud de mensaje Solicitud de mensaje con datos
Memoria, E/S, Configuración	Finalización Finalización con datos Finalización bloqueada Finalización bloqueada con datos

# Formato de Unidad de Datos del Protocolo PCIe

Las transacciones PCIe se transmiten utilizando paquetes de capa de transacción. El formato de los paquetes en el protocolo PCIe (Peripheral Component Interconnect Express), desglosado en dos niveles:

1.- Paquete de la Capa de Transacción (Transaction Layer Packet - TLP): Este paquete es generado por la capa de transacción y contiene las siguientes secciones:

**STP framing (Start of TLP framing):** Señal de inicio del paquete para delimitarlo. Ocupa 1 byte.

**Sequence Number:** Un campo que indica el número de secuencia del paquete para garantizar la entrega ordenada. También ocupa 1 byte.

**Header:** Contiene información esencial para procesar el paquete, como dirección, tipo de operación, etc. Puede ocupar 12 o 16 bytes, dependiendo de la operación.

**Data:** La carga útil del paquete, que puede variar entre 0 y 4096 bytes. Es opcional, dependiendo del tipo de transacción.

**ECRC (End-to-End CRC):** Código de redundancia cíclica (CRC) opcional para verificar la integridad de los datos de extremo a extremo. Ocupa 0 o 4 bytes.

2.-Paquete de la Capa de Enlace de Datos (Data Link Layer Packet - DLLP): Este paquete es generado por la capa de enlace y es más pequeño que el TLP.

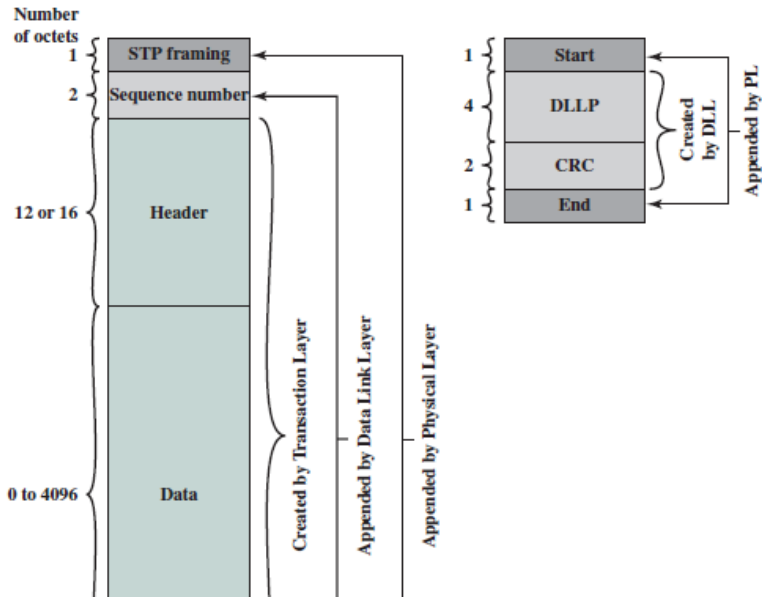
**Start:** Indica el inicio del paquete de la capa de enlace. Ocupa 1 byte.

**DLLP:** Contiene datos de control o gestión del enlace. Este es el núcleo del paquete de la capa de enlace.

**CRC:** Código de redundancia cíclica usado para verificar la integridad de los datos transmitidos. También ocupa 1 byte.

**End:** Indica el final del paquete. Ocupa 1 byte.

# Formato de Unidad de Datos del Protocolo PCIe





Esta capa implementa mecanismos robustos para garantizar la entrega confiable de datos. Gestiona el control de flujo y maneja los paquetes de nivel de enlace (DLLPs). Su sistema de verificación incluye números de secuencia y códigos de verificación, asegurando la integridad de cada transmisión.

- El sistema PCIe consta de varios componentes clave:
  - 1 **Root Complex**: Es el componente principal que se conecta al procesador. Controla el acceso a los dispositivos y coordina las transferencias de datos.
  - 2 **Switches**: Son los dispositivos que gestionan y dirigen el tráfico entre diferentes dispositivos conectados al sistema PCIe.
  - 3 **Endpoints**: Son los dispositivos conectados al bus PCIe, como tarjetas gráficas, SSDs, y otros periféricos.

PCIe ha revolucionado la forma en que los componentes de computadora se comunican entre sí. Su arquitectura escalable y eficiente ha permitido el desarrollo de nuevas tecnologías y aplicaciones, estableciendo un estándar que continúa evolucionando para satisfacer las necesidades futuras de la industria.

Stallings, W. (2006). *Organización y arquitectura de computadores*.  
Pearson Educación.  
<https://books.google.com.ec/books?id=C3HTAAAACAAJ>