

Departamento de Arquitectura

Instituto de Computación

Universidad de la República

Montevideo - Uruguay

# **Notas de Teórico**

## **Entrada / Salida**

**Arquitectura de Computadoras**

(Versión 4.3 - 2012)

## 15 ENTRADA/SALIDA

### 15.1 Introducción

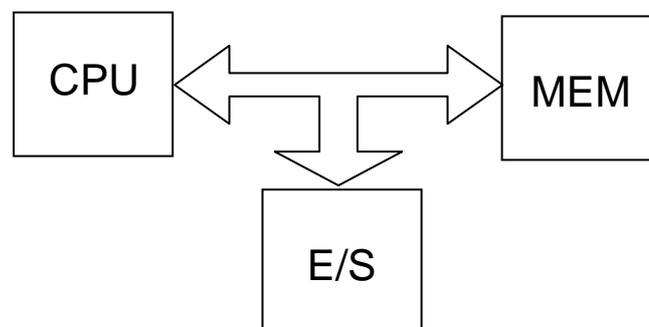
En este capítulo vamos a completar la presentación de la arquitectura von Neuman, introduciendo la parte de entrada y salida, encargada de vincular a la computadora (hasta ahora conjunto CPU+Memoria) con el mundo exterior.

La entrada/salida (habitualmente anotada E/S) actúa como interfaz entre los programas que ejecuta la CPU en base a las instrucciones y los datos almacenados en la memoria, con los dispositivos que permitirán interactuar con el mundo físico, permitiendo así por un lado introducir la información a procesar en el sistema (entrada) y por otro presentar el resultado de ese proceso de modo inteligible para el ser humano (o al menos de manera más amigable que la de medir diferencias de potencial en los circuitos).

Los dispositivos que permiten la interacción de los usuarios con el sistema se denominan **periféricos**. Esta denominación obedece a razones históricas, que se remontan a la época en que un computador ocupaba una sala entera, donde la capacidad de proceso se ubicaba en el centro (de allí Unidad Central de Proceso) y los dispositivos de interacción con el usuario estaban a su alrededor (en la periferia).

### 15.2 Buses

La conexión de la CPU, la Memoria y la Entrada/Salida, se realiza mediante grupos de "líneas" que comunican tres tipos de información: dirección, datos y control. Esta agrupación se denomina **bus** (su nombre hace alusión al hecho que recorre todos los subsistemas a conectar).



Por lo indicado en este bus principal pueden distinguirse a su vez tres buses secundarios:

- bus de direcciones:
  - formado por las líneas de conexión que transportan las direcciones de memoria o E/S a ser accedidas durante la transferencia
- bus de datos:
  - formado por las líneas de conexión que transportan la información que es transferida sobre el bus entre los distintos componentes conectados
- bus de control:
  - formado por las líneas de conexión que transportan señales que controlan el uso del bus y la comunicación sobre el mismo. Algunas de las señales típicas presentes en este bus son:
    - ✘ Memory\_Read (indica una operación de lectura sobre la memoria)

- ✘ Memory\_Write (indica una operación de escritura sobre la memoria)
- ✘ I/O\_Read (indica una operación de lectura sobre la entrada/salida)
- ✘ I/O\_Write (indica una operación de escritura sobre la entrada/salida)
- ✘ Bus\_Request (indica que un sub-sistema desea tomar control del bus para iniciar una transferencia)
- ✘ Bus\_Grant (confirma que el bus está disponible para quien lo solicitó)
- ✘ Transfer\_ACK (confirma la recepción de una transferencia de información)
- ✘ Interrupt\_Request (indica un pedido de interrupción hacia un sub-sistema)
- ✘ Interrupt\_ACK (confirma la aceptación del pedido de interrupción)
- ✘ Clock (sincroniza las actividades del bus y sus señales)
- ✘ Reset (fuerza el reset de todos los componentes conectados al bus)

El bus de direcciones transporta información que permite identificar y seleccionar las posiciones de memoria a leer ó escribir y/o las entidades de E/S con las que se va a interactuar. El bus de datos transporta los datos leídos o a escribir ya sea en la memoria o en los dispositivos de E/S (traspaso de información entre los sub-sistemas). El bus de control permite la administración de la comunicación entre la CPU, la Memoria y la E/S.

En general un bus está definido por un conjunto de características que incluyen:

- especificaciones mecánicas (conectores)
- especificaciones eléctricas (niveles de voltajes y/o corriente), bits de dirección y de datos y consideraciones de "timing" (tiempo) de las señales involucradas incluyendo el reloj de sincronismo
- protocolo de comunicaciones

En particular cuando hablamos del "protocolo de comunicaciones" nos estamos refiriendo al conjunto de reglas que debe seguir cualquier entidad conectada al bus para hacer uso del mismo a los efectos de realizar un intercambio de información con otra entidad.

Según sea el tamaño del sub-bus de datos se habla de buses de 16, 32 o 64 bits (estos son los tamaños habituales al presente). También caracteriza a un bus si el sub-bus de datos es separado del sub-bus de direcciones o está multiplexado en el tiempo con él. En particular estas dos características, unidas a la frecuencia del reloj utilizado para el sincronismo, determinan la capacidad de transferencia de información del bus.

Según sean las características del protocolo de comunicaciones se distinguen distintos tipos de buses. Una primera clasificación posible es en base a su complejidad:

- **Bus Simple**

Este tipo de buses se implementan en una filosofía "Master-Slave" ("Maestro-Escavo"). Existe una sola entidad que controla el uso del bus en todo momento (el "Master"), típicamente la CPU, quien inicia, controla y participa de todas las transferencias que se realizan en el bus. Todas las transacciones sobre el bus se realizan bajo su supervisión. Las entidades "esclavas" que deseen utilizar el bus tienen que esperar ser habilitados por el "maestro", el cual o bien utiliza técnicas de "polling" (consulta) para determinar si una entidad esclava desea hacer una transferencia o bien dispone de algún mecanismo por el cual la entidad interesada en realizar una transferencia le avisa de este interés (ej: por un pedido de interrupción).

- **Bus Inteligente**

En estos buses cualquier entidad conectada puede tener, al menos potencialmente, la capacidad de convertirse en el "Master" del bus (se dice que tiene la capacidad de "bus mastering") y comandar una transferencia. Al momento que una o varias entidades requieren hacer uso del bus ocurre una competencia por el uso del mismo, la cual es resuelta por un mecanismo de arbitraje que determina cuál de los contendientes pasa a ser el "master", mientras que el resto permanecerá como "slave" hasta que se termine la transacción.

En función de cómo sea este mecanismo de arbitraje se distinguen, a su vez, las siguientes categorías de bus:

- **Bus con Arbitraje Centralizado**

Un bus con este tipo de arbitraje se denomina Jerárquico. En estos buses hay una entidad que tiene mayor jerarquía que las demás (típicamente la CPU) y es la responsable de entregar el control del bus a las entidades que lo requieran. La entidad interesada en utilizar el bus le avisa de este interés ("bus request") y el árbitro la habilita a utilizarlo ("bus grant").

- **Bus con Arbitraje Distribuido**

Un bus con este tipo de arbitraje es denominado Equitativo. En estos buses todas las entidades conectadas tienen el mismo "rango" y no existe un árbitro central, sino que el mecanismo de asignación del bus tiene su inteligencia distribuida entre todas las entidades con capacidad de "bus mastering". Cuando dos o más entidades requieren simultáneamente del bus se aplica un mecanismo de arbitraje que determina cuál de los contendientes por el uso del bus tendrá el control del mismo en ese momento y cuál(es) tendrá(n) que esperar hasta que quede liberado nuevamente. Este arbitraje, a diferencia del caso anterior, se resuelve por una especie de votación entre los distintos contendientes, sin la necesidad de contar con un árbitro fijo y central.

Existe otra clasificación de los buses que apunta a categorizarlos según sea la aplicación a la que se destinan, en función de la ubicación de las entidades a conectar en relación a lo que se podría definir como "frontera" del sistema. Si bien esta clasificación es discutible y eventualmente varía con el tiempo ó con el alcance de lo que consideremos "sistema", los buses se suelen distinguir entre:

- **Internos** cuando se utilizan para conectar sub-sistemas dentro de la "frontera" del sistema. El caso más típico e indiscutible de un bus interno es el "bus de memoria" (que conecta la CPU con los bancos de memoria)
- **Externos** cuando se utilizan para conectar sub-sistemas fuera de dicha frontera. Los casos más típicos y claros de este tipo de buses lo constituyen el bus SCSI (Small Computer System Interface) que se puede utilizar para conectar básicamente dispositivos de almacenamiento masivo (discos rígidos y unidades de cinta), y el bus USB (Universal Serial Bus) que se puede utilizar para conectar una variedad muy amplia de dispositivos (discos, impresoras, "mouses", escáners, etc).

### 15.3 Buses de Expansión

Por otra parte están los que podemos denominar **buses de expansión**, que se

utilizan para conectar placas de circuito adicionales (tarjetas de expansión ó **expansion boards** ó **expansion cards**) a la placa de circuito principal del sistema (placa madre o **motherboard**). En la casi totalidad de las computadoras actuales estos buses de expansión están contenidos dentro del chasis del equipo por lo que podríamos considerarlos *internos*, aunque si el "sistema" que consideráramos fuera la placa principal, entonces habría que clasificarlos como *externos*.

Los buses de expansión se identifican claramente en un sistema porque poseen un conector "hembra" característico, normalmente diseñado para insertar en él una placa de circuito con sus contactos en la propia placa, sin necesidad de un conector "macho" (aunque hay buses que utilizan conectores tanto en la placa madre como en la de expansión).

Es destacable el hecho que, si bien como concepto arquitectónico el bus de expansión estuvo presente en la mayoría de los diseños de computadoras desde los inicios de la industria, es a partir de la mitad de la década de los '70 que toma un interés especial. A partir de la microcomputadora Altair 8000, pero en especial a partir de la Apple II, el bus de expansión se convierte en un actor clave en la industria de la computación, ya que al poner en "dominio público" la especificación del mismo, se abre la posibilidad que fabricantes distintos del diseñador original del computador puedan producir "tarjetas de expansión" compatibles con dicho sistema, formándose una industria completa que gira alrededor de las ampliaciones y opciones para los sistemas básicos. Entre otras cosas el éxito de la estrategia de Apple y su "diseño de bus abierto" forzó a IBM a incursionar en la misma senda al lanzar su PC en 1980, lo que terminó de cambiar las reglas de una industria que, hasta ese momento, basaba la diferenciación de su oferta y la competitividad en diseños propietarios y cerrados a otros actores.

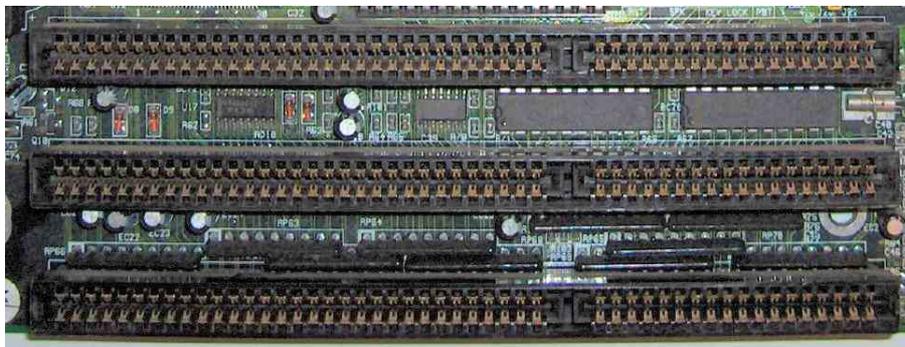
A continuación veremos algunos ejemplos de buses de expansión.

### **Buses de Expansión en los PCs**

- **ISA (Industry Standard Architecture)**

Este bus corresponde al incorporado por IBM en su modelo PC-AT en 1984. Es un bus de 16 bits de datos y 24 bits de direcciones no multiplexados, del tipo "maestro-esclavo". Básicamente es una extensión del bus del microprocesador 80286 utilizado en esos sistemas, con poca inteligencia y un reloj de sincronismo de 8 MHz, con una tasa de transferencia máxima de 16 MBps.

La siguiente fotografía muestra los conectores de la placa madre de un computador con bus ISA.



*Tomado de Wikipedia - Foto tomada por Jorge González*

- **MCA (Micro Channel Architecture)**

Este bus fue incorporado por IBM en su frustrada línea de computadoras PS/2 en 1987, en un intento de retomar el control de la industria de tarjetas de expansión y, en especial, de frenar el fenómeno de los sistemas "clone" (clonado). En un bus inteligente con capacidad de "bus mastering" y arbitraje centralizado, de 32 bits de datos y 32 bits de direcciones, no multiplexados, independiente del tipo de procesador de la placa madre y un reloj inicial de 10 MHz (tasa de transferencia de 20 MBps a 40 MBps).

- **EISA (Extended Industry Standard Architecture)**

Este bus fue diseñado por los distintos fabricantes de PCs "compatibles" en reacción al Microcanal de IBM. Es una extensión del bus ISA a 32 bits (32 bits de datos y 32 bits de direcciones no multiplexados), con dos modalidades de funcionamiento: una con capacidad de "bus mastering" (arbitraje centralizado) y otra del tipo master-slave para funcionar como un ISA. El diseño de su conector, así como la posibilidad de trabajar en las dos modalidades, le permitió soportar tarjetas ISA al mismo tiempo que las "nuevas" tarjetas EISA (aunque sacrificando prestaciones). Si bien esto no fue suficiente para sobrevivir al paso del tiempo, logró su objetivo inmediato: destruyó totalmente la estrategia de IBM y la obligó a abandonar la MCA. El reloj del bus era de 8 MHz (como el ISA), con una tasa de transferencia máxima de 32 MBps.

- **VLB (VESA Local Bus - VESA = Video Electronics Standards Association)**

Este bus fue una propuesta impulsada por los fabricantes de tarjetas de video que necesitaban de un mayor ancho de banda de transferencia que el ofrecido por el bus EISA, principalmente para la conexión de la tarjeta de video gráfica. Consistió básicamente en volver a la idea de "amplificar" el bus interno del microprocesador, en este caso del 80486. Es de 32 bits y trabaja a 33 MHz (igual que dicho micro).

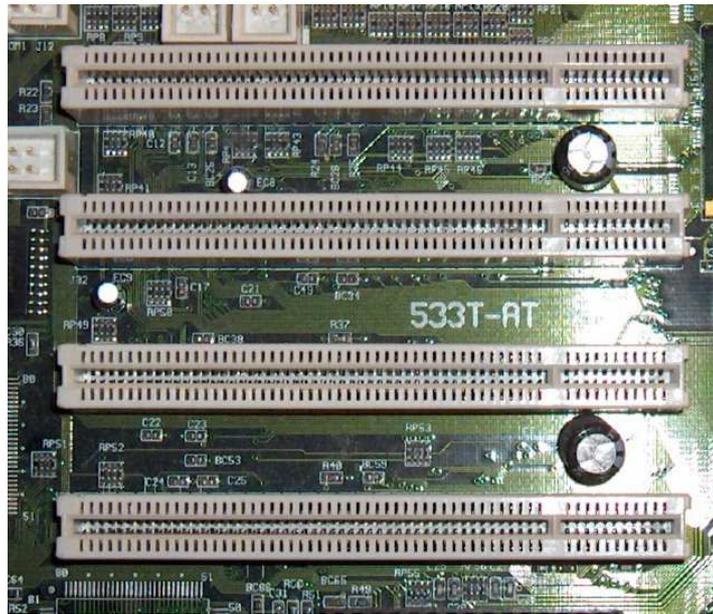
- **PCI (Peripheral Component Interconnect)**

Este bus fue desarrollado inicialmente por Intel y apareció por 1993 (su versión inicial fue la 2.0). Actualmente sus especificaciones son mantenidas por PCI-SIG (PCI Special Interest Group) un consorcio de empresas entre las que, naturalmente, está Intel. Es un bus que acepta configuraciones de 32 bits de datos y 32 bits de direcciones ó, con un conector más grande de 64 bits, multiplexados. Su primera versión fue con un reloj de 33.33 MHz, para una tasa de transferencia de 133 MBps (266 MBps cuando se implementaba en 64 bits). Al igual que el MCA y el EISA es un bus con capacidad de "bus mastering" y arbitraje centralizado. La versión 2.1 de la especificación pasó a soportar un reloj de 66.67 MHz, con una tasa de transferencia de 266 MBps y comenzó a soportar tarjetas con voltajes de 3.3 Volts, además de los tradicionales 5 Volts. La especificación 3.0 pasó a usar solamente 3.3 Volts.

Su evolución, el PCI-X (propuesto por Compaq, IBM y HP en 1998), utiliza un reloj de 133 MHz, lleva la tasa de transferencia a 1014 MBps y el PCI-X 2.0 utiliza un reloj de 266 MHz para una tasa de transferencia de 2035 MBps.

A fines de la década de los '90 y comienzos de este siglo, el bus PCI terminó por desplazar completamente a los buses MCA y EISA, convirtiéndose en el único estándar en materia de buses de expansión de la arquitectura PC, con la excepción del bus AGP.

La siguiente fotografía muestra los conectores de la placa madre de un computador con bus PCI.



*Tomado de Wikipedia - Foto tomada por Jorge González*

- **AGP (Accelerated Graphic Port - Advanced Graphic Port)**

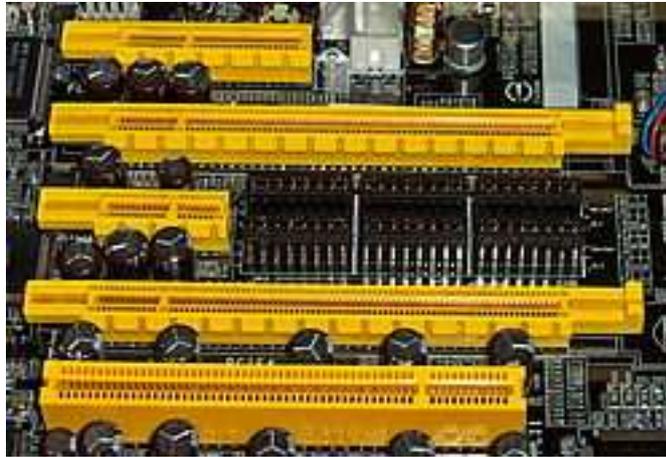
Este bus fue diseñado por Intel teniendo en mente la conexión de tarjetas gráficas de video (como su nombre lo indica). Desde 1997 se sucedieron las especificaciones AGP 1.0 de 32 bits y 66 MHz para una tasa de transferencia de 266 MBps, el AGP 2.0 de 32 bits y 133 MHz, llevando la tasa de transferencia a 533 MBps, el AGP 4x con 32 bits y 133 MHz pero utilizando la técnica de "double pump" (utiliza ambos flancos del reloj para transferir datos) lleva la tasa de transferencia a 1066 MBps y finalmente la AGP 8x que utiliza la técnica "quad pump" (utiliza flancos desfasados del reloj principal) y alcanza 2133 MBps.

- **PCI Express (PCIe)**

Este bus fue desarrollado por el consorcio PCI-SIG como evolución del PCI. Tiene un diseño radicalmente distinto: es un bus de múltiples conexiones serie en lugar de ser un bus paralelo como el PCI y sus antecesores. Esta condición le permite dos características interesantes al momento de evaluar rendimientos: puede transmitir en forma "full-duplex" en cada conexión ("lane" en la terminología PCIe) es decir los dos nodos que se comunican pueden transmitir simultáneamente, y puede utilizar velocidades de reloj muy altas. Tener presente que los buses paralelo tienen una limitante en la velocidad de reloj porque padecen el fenómeno del "skew" que consiste en que las señales transmitidas por conductores paralelos pueden llegar desfasados en el tiempo si los conductores tienen distinto largo, llevando a confundir bits de una palabra con bits de la siguiente. A medida que la frecuencia sube las diferencias de largo de los conductores se hace cada vez más crítica e imposibles de resolver en la práctica. Usando técnicas de paralelismo asíncrono el PCIe puede transmitir por varias conexiones en forma simultánea, multiplicando la capacidad de transferencia de información por la cantidad de conexiones ("lanes") disponibles. Es así que surgen las variantes x1, x2, x4, x8, x16 y x32 del bus, en función de la cantidad de conexiones serie en paralelo que posee el bus. Las implementaciones actuales no incluyen la x32 (aunque sí está en la especificación). Una tarjeta PCIe puede insertarse en un slot de mayor paralelismo (ej: una x1 se puede colocar en

una ranura x4). Lo inverso también es cierto, ya que el protocolo negocia el funcionamiento en función de las conexiones eléctricas disponibles, pero para esto el conector del slot debe tener un diseño especial (abierto en un extremo).

La siguiente foto muestra distintos tipos de slots PCIe (de arriba hacia abajo: x4, x16, x1, x16 y un PCI estándar de 32 bits):



*Tomado de Wikipedia - Foto tomada por Snickerdo*

### **Otros Buses de Expansión**

Obviamente los buses de expansión que se han utilizado y se utilizan en los computadores personales no son los únicos existentes. De hecho no fueron los PCs los primeros sistemas en contar con esta forma de conectar tarjetas de circuito adicionales y hay buses que perduran hasta hoy en día, aunque opacados por la difusión de los vinculados al mundo del PC.

Veamos un par de ejemplos relevantes:

- **VMEbus (VERSA Module Eurocard Bus)**

Es un bus histórico que tiene sus orígenes en la década del 70. Fue diseñado por Motorola, originalmente para su familia de microprocesadores 68000. Mantiene su aplicabilidad en sistemas embebidos y sistemas de control en general. Es un bus inteligente con arbitraje centralizado, realizado por la tarjeta que se inserte en la primera ranura, con prioridad jerárquica o circular. Utiliza conectores (formato DIN europeo) tanto en la tarjeta de expansión como en el "backplane" (conjunto de conectores "macho" con alimentación que implementa el bus). Un sistema construido con VMEbus dispone de un chasis con el "backplane" y la fuente de alimentación en el cual se insertan las tarjetas. El procesador (tarjeta que contiene el microprocesador y la memoria) es una tarjeta que se inserta como cualquier otra de expansión (normalmente en el primer slot para oficiar como árbitro del bus).

La siguiente fotografía muestra un chasis con bus VME y una tarjeta para el mismo.



*Tomado de la página web de ITCN*

- **NuBus**

Este bus fue originalmente desarrollado por el MIT (Massachusetts Institute of Technology) como parte de un proyecto de un sistema denominado NuMachine que nunca fue completado. El diseño fue tomado primeramente por Texas Instruments para su sistema TI Explorer (una máquina diseñada para trabajar en lenguaje LISP) y su sistema S1500 (un multiprocesador basado en UNIX). Luego Apple adoptó una versión modificada (con un solo conector) para su familia de computadores Macintosh (luego Apple adoptó el bus PCI y abandonó el NuBus).

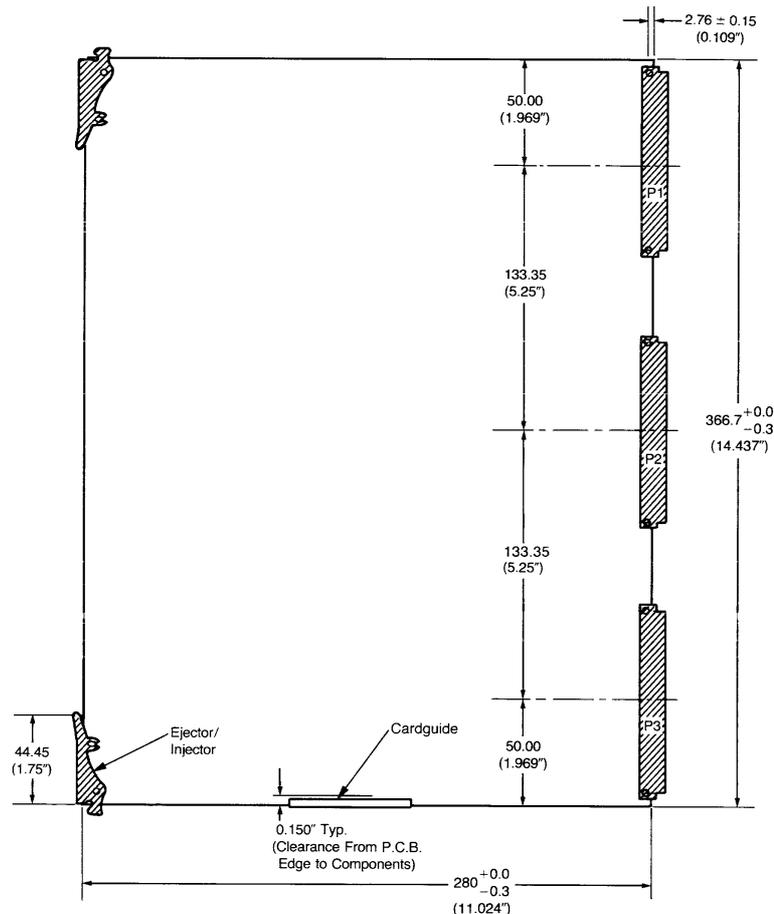
Es un bus inteligente, de 32 bits, con capacidad de bus mastering, con arbitraje distribuido. Utiliza conectores tanto en la tarjeta como en la placa principal. Texas Instruments lo implementó en formato Eurocard (similar al VMEbus) con backplane pasivo, mientras que Apple lo implementó con un único conector por tarjeta (en vez de tres) y sobre la placa principal de las Macs.

La siguiente fotografía muestra una tarjeta de expansión para una computadora Apple Macintosh basada en este bus:



*Tomado de Wikipedia - Foto tomada por Danamania*

El siguiente esquema muestra la especificación mecánica del NuBus original de TI:



Tomado del manual de Texas Instruments 2242825-0001 NuBus Specifications

## 15.4 Periféricos de Entrada/Salida

Los dispositivos periféricos de entrada/salida son los que realizan el vínculo del computador con el mundo exterior. A través de ellos se realiza el ingreso de programas y datos a procesar (mediante teclados, lectores de distinto tipo, etc) y se obtienen los resultados del proceso de la información en un formato que sea "legible" para el ser humano (impreso en papel, desplegado en una pantalla, etc) ó se provoca alguna alteración del mundo físico circundante (ej: encendido de una luz). A este tipo de periféricos a veces se los denomina "de interfaz" o "transductores".

Si bien no son estrictamente dispositivos de "entrada/salida" también se incluyen entre ellos a los dispositivos de almacenamiento masivo de información (discos, cintas). También están los dispositivos de comunicación, que permiten intercambiar información con otros sistemas.

Algunos ejemplos de estos dispositivos son:

### Transductores:

- Teclado
- Apuntador (más conocido como Ratón)
- Impresora (Matriz, Chorro de Tinta, Laser)

- Monitor (CRT, LCD, Plasma)
- Escáner
- Tableta de Digitalización

#### Almacenamiento:

- Diskettera
- Disco Magnético
- Disco Optico (CD, DVD)
- Cinta (QIC, DAT, DLT)

#### Comunicación:

- Modem (Analógico, RDSI, ADSL)
- Red Ethernet

## 15.5 Controladores de E/S

La arquitectura de E/S maneja el concepto de “controlador de E/S”. Un controlador de E/S es la parte del periférico que contiene su inteligencia y por tanto lo “controla”. La denominación es heredada del pasado cuando una impresora o una unidad de cinta tenían por un lado un “armario” que contenía el mecanismo (martillos de impresión, motores de arrastre, cabezas magnéticas, etc) y por otro lado otro armario que contenía los circuitos que manejaban esos mecanismos, accionando los martillos, los motores, decodificando señales, etc. Este último “armario” era el “controlador”.

En la actualidad la parte mecánica del dispositivo y la inteligencia que lo controla están en el mismo gabinete, o al menos la inteligencia está distribuida entre el gabinete del periférico y el gabinete de la computadora. Dos ejemplos de casos extremos pueden ser por un lado un monitor gráfico (la caja del periférico tiene casi ninguna inteligencia vinculada a la generación de las imágenes que muestra y toda la inteligencia se concentra en el “controlador de video gráfico” que reside en el computador) y por el otro lado una impresora láser (que incluye una verdadera computadora interna para generar las imágenes que debe imprimir y manejar todo el complejo mecanismo de impresión y en el computador solo reside una “interfaz” para comunicar los datos a imprimir).

### 15.5.1 Conexión de los Controladores de E/S

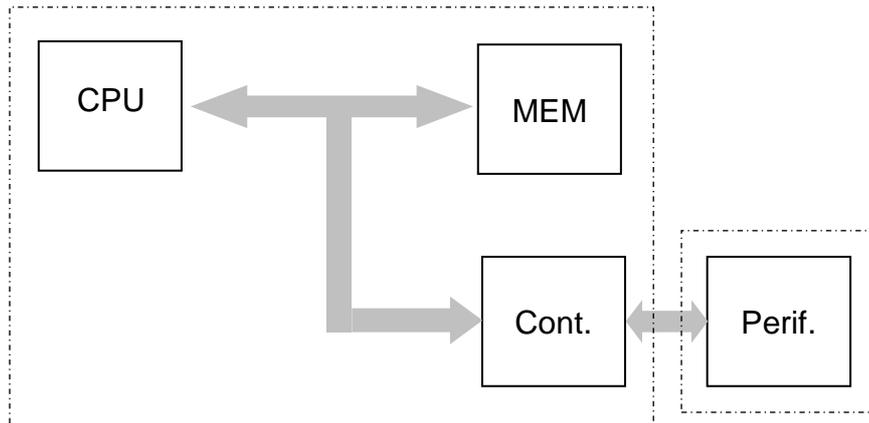
Existen distintas variedades en la forma que un controlador se conecta con un computador, en particular teniendo en cuenta su lugar de residencia: si en el gabinete del periférico, si en el gabinete del computador o distribuido entre ambos.

Veremos a continuación la representación esquemática de algunos de estos casos:

#### 1) El controlador de E/S está contenido en el gabinete del computador

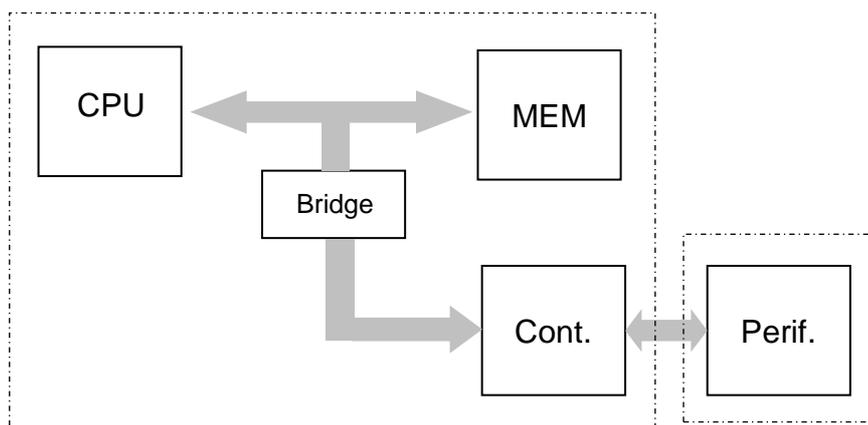
- a) El controlador de E/S está conectado directamente al bus interno del sistema

Este es el caso del controlador de video de un computador personal basado en bus ISA (Nota: en estos ejemplos estamos teniendo en cuenta que el bus ISA es prácticamente el bus interno con sus señales amplificadas).



- b) El controlador de E/S está conectado a un bus distinto al bus interno del sistema, el cual se conecta al bus interno a través de un adaptador

Es el caso de un controlador de video en un computador personal con bus PCI ó AGP. Es de notar que es habitual en la terminología asociada a la tecnología PCI denominar "bridge" (puente) a este adaptador.



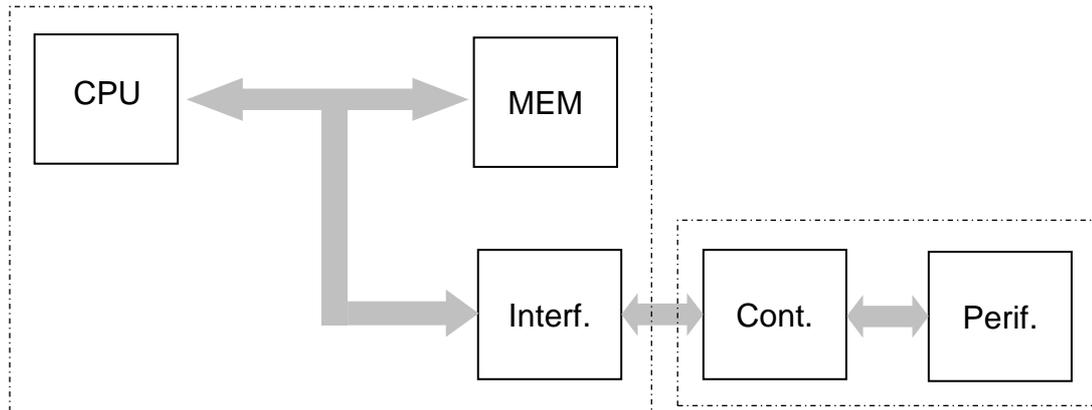
## 2) El controlador de E/S está contenido en el gabinete del periférico

En este caso no se da la situación que el controlador de E/S pueda estar conectado directamente al bus interno del sistema porque dicho bus no está diseñado para soportar largos recorridos de la señal. Por eso se requiere una interfaz ó un bus del tipo externo que habilite la conexión. La diferencia entre interfaz y bus es que la interfaz permite la conexión de un único dispositivo, por lo que su protocolo de comunicaciones es más sencillo y no necesita de un algoritmo de administración del uso compartido.

Podemos distinguir las situaciones:

- a) Se utiliza una interfaz conectada directamente al bus interno del sistema

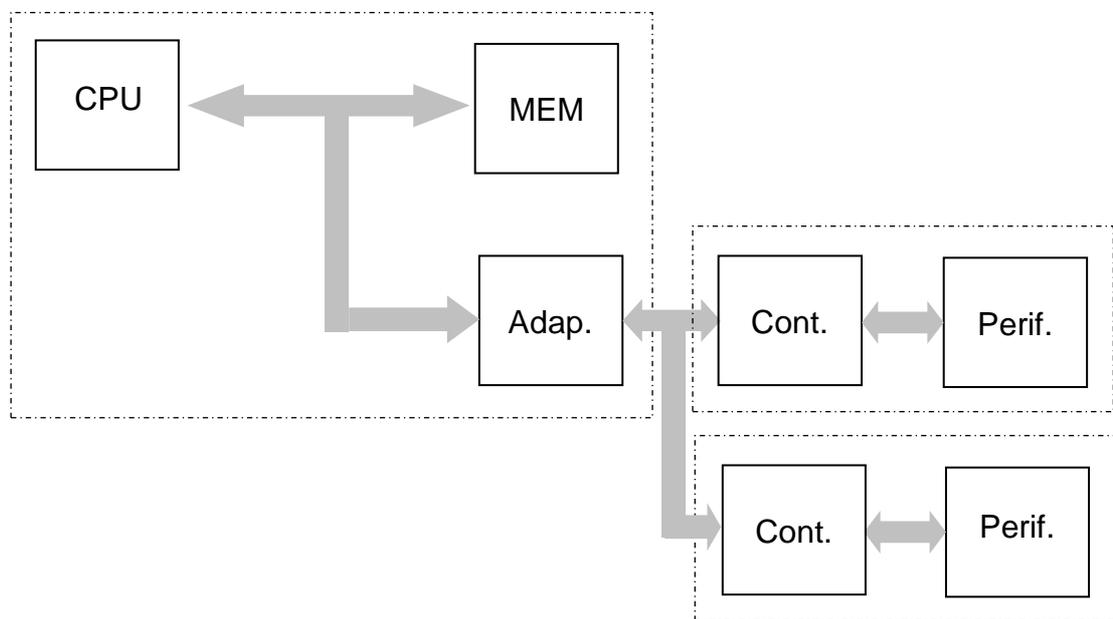
Es el caso de una impresora conectada por un puerto paralelo a un computador personal con bus ISA.



- b) Se utiliza un adaptador de bus conectado directamente al bus interno del sistema

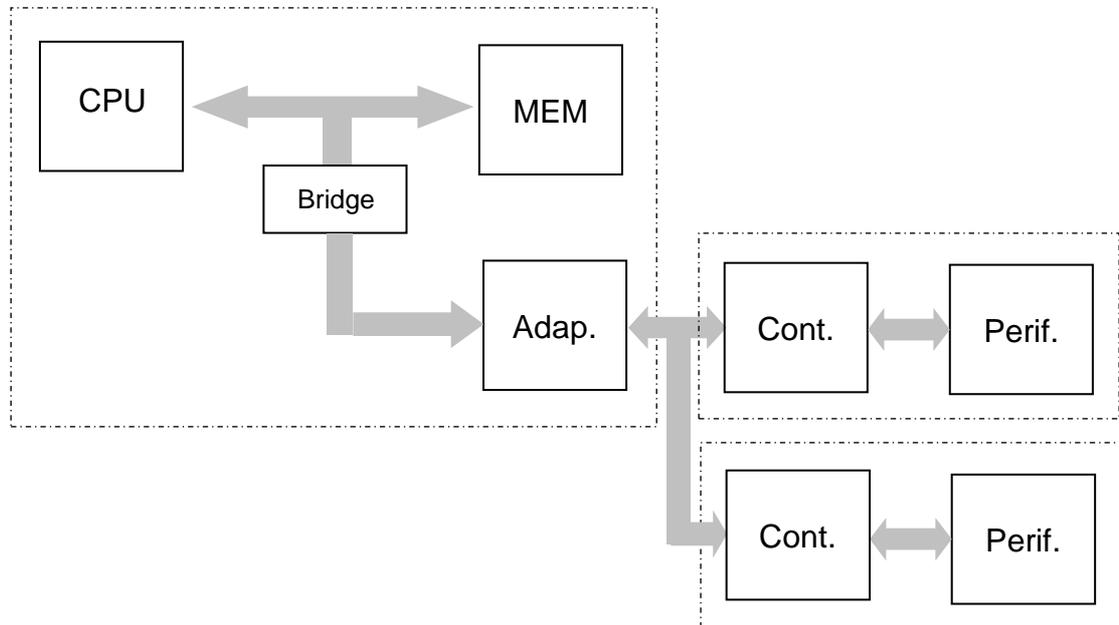
Es el caso de un disco SCSI (Small Computer System Interface) en un computador con bus ISA.

Estamos considerando que el hecho que el disco de un computador personal está contenido en el gabinete esto no modifica el hecho que también puede estar fuera, cuando se utiliza el estándar SCSI. En este caso la ubicación tiene que ver con aspectos mas vinculados a la comodidad y a la economía que a requerimientos técnicos.



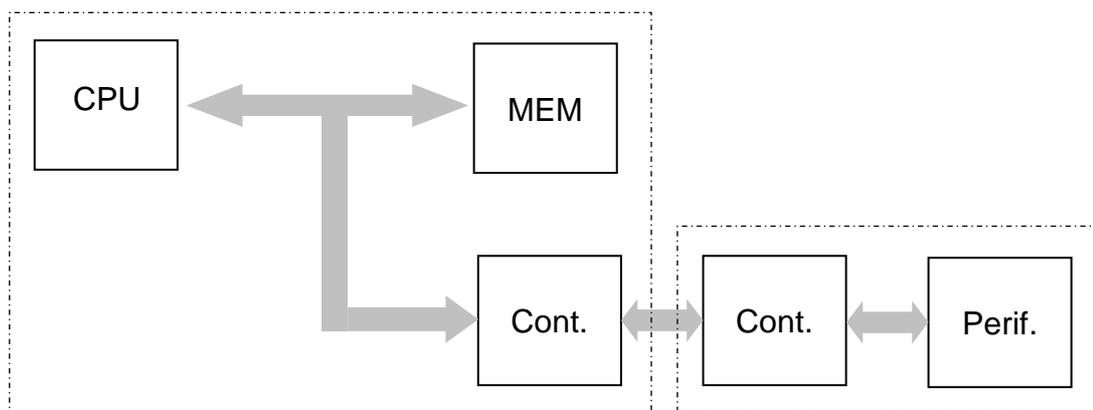
- c) Se utiliza un adaptador de bus conectado a un bus distinto del bus interno

Es el caso de un disco SCSI (Small Computer System Interface) en un computador con bus PCI.



- 3) El controlador de E/S está distribuido entre el gabinete del computador y el periférico

Es el caso del teclado de un computador personal, donde parte de la inteligencia reside en el propio teclado (posee un microcontrolador que realiza el barrido de las teclas para determinar si hay alguna presionada) y parte en la placa principal del computador (donde otro microcontrolador realiza algunas funciones de conversión de códigos y repetición de teclas).



### 15.5.2 Acceso a los Controladores de E/S

La comunicación de la CPU con los controladores de E/S se realiza a través de posiciones de memoria especiales de estos dispositivos, las que son accesibles también para la CPU. Existen dos tipos de arquitecturas de E/S respecto a la forma de acceso desde la CPU:

- 1) La CPU dispone de un espacio de direcciones reservado para la E/S, el cuál es accedido por instrucciones especialmente destinadas para ese fin (típicamente identificadas como **in** y **out**). Un ejemplo de este enfoque es la familia de procesadores Intel.
- 2) La CPU accede a los controladores de E/S como si se trataran de posiciones normales de memoria, utilizando para tales fines cualquier instrucción que acceda a memoria. Un ejemplo de este diseño es la familia de procesadores SPARC de Sun Microsystems (y en general todos los diseños RISC).

Estas memorias especiales de los controladores de E/S se llaman genéricamente **registros**. Normalmente un controlador de E/S tiene un conjunto relativamente pequeño de registros para implementar el intercambio de información con la CPU. Una excepción a esta regla la constituyen los controladores de video que por requerir grandes volúmenes de información necesitan espacios de direcciones amplios. Esto representa alguna dificultad de implementación en las arquitecturas Intel (porque el espacio de direcciones reservado a E/S es pequeño) por lo que en los PCs se adoptó una estrategia de mapeo de las memorias de video en el espacio de direcciones de memoria de la CPU (en vez de utilizar el espacio de E/S).

Una propiedad interesante de los registros de E/S es que aún en el caso que se accedan como memoria, **no** se comportan como memoria, pudiendo tener comportamientos bien diferentes tales como:

**sólo lectura:**

el registro solo puede ser leído y si se escribe en él no se logra ningún efecto (es decir si luego de escribir se lee, lo que se lee no es lo que se escribió).

**sólo escritura:**

el registro solo puede ser escrito y si se lee se obtiene un resultado impredecible (los datos que se leen pueden ser cualquier cosa).

**lectura/escritura independiente:**

en este caso se tienen dos registros diferentes, uno de sólo lectura y otro de sólo escritura accesibles en la misma dirección de E/S. Por lo que si bien se puede escribir y leer en la misma dirección, las posiciones de memoria accedidas son separadas e independientes, por lo que, obviamente, lo que se escribe no puede ser leído posteriormente.

**lectura/escritura normal:**

estos registros de E/S se comportan como una posición de memoria normal, lo que se escribe puede ser leído más tarde

Otra característica importante a tener en cuenta cuando se accede a E/S es que muchas veces hay bits que no están definidos, por lo que al leerlos juntos en un byte o una palabra pueden tomar cualquier valor, con lo que la comparación del contenido del byte ó

palabra con valores debe hacerse mediante el uso de **máscaras** (haciendo el AND bit a bit con un valor que tenga en uno aquellos bits que nos interesan, de forma que el resultado tenga únicamente en cuenta los bits que sí están definidos ó aquellos que nos interesan en cada momento). Algo similar ocurre al escribir, en algunos registros determinados bits que no están definidos deben, de todos modos, escribirse en un valor determinado (0 ó 1).

La cantidad de registros y su función dependen fuertemente de cada controlador y de la aplicación que tenga el mismo (no es igual un controlador de comunicaciones que uno de video). De todos modos hay algunos registros característicos que están presentes en muchos de ellos:

**Datos Entrada:**

este registro contiene un dato destinado a la CPU, proveniente del periférico, del propio controlador o de la línea de comunicaciones.

**Datos Salida:**

este registro contiene un dato proveniente de la CPU y destinado al periférico, al propio controlador o a la línea de comunicaciones.

**Estado:**

este registro contiene bits que indican el estado del controlador en sí mismo o del periférico que controla (ej; si hay un dato en el registro de entrada, si está libre el registro de salida, si hay un pedido de interrupción pendiente, si hay alguna condición de error en el controlador ó el periférico, etc).

**Control:**

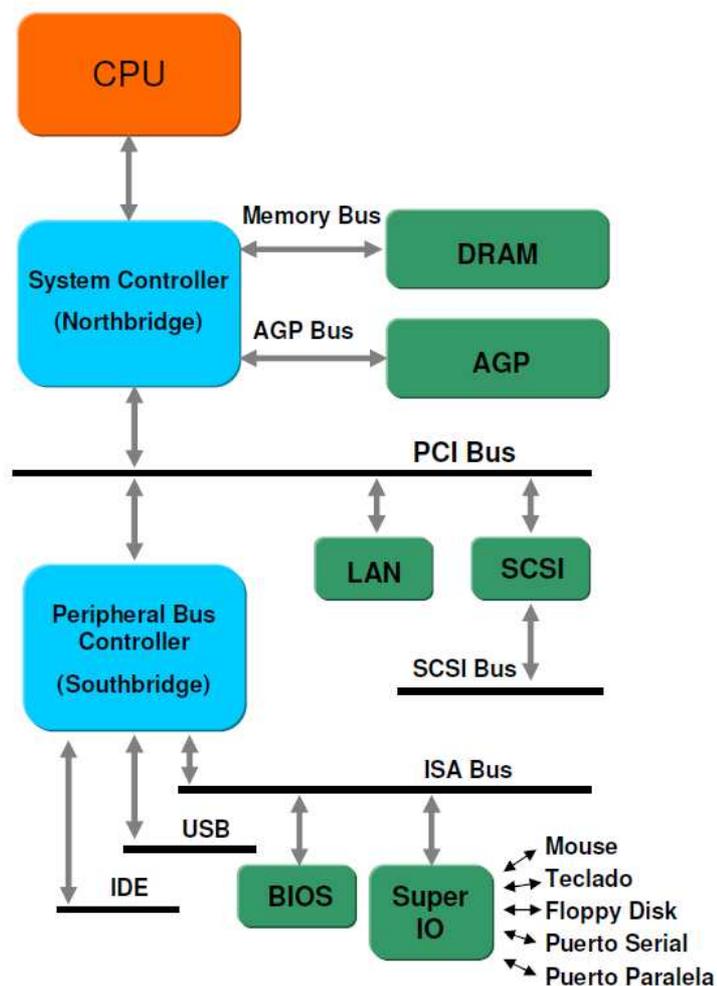
este registro contiene bits que le indican al controlador de E/S ó al periférico realizar determinada acción (ej: ponerse en línea, reiniciarse, que lea el dato del registro de salida, etc). En algunos controladores de E/S antiguos también se controlaban qué registros eran accesible en cada momento en una dirección de E/S específica (por ejemplo se implementaban stacks de registros y con un bit del registro de control se hacía el "push" y con otro el "pop").

## 15.6 Arquitectura de E/S y Buses de un PC

La arquitectura de buses de los Computadores Personales (PCs) ha evolucionado con el tiempo. En un inicio (etapa del Bus ISA) el esquema era básicamente el que hemos venido considerando: la CPU directamente conectada a la memoria y a la E/S.

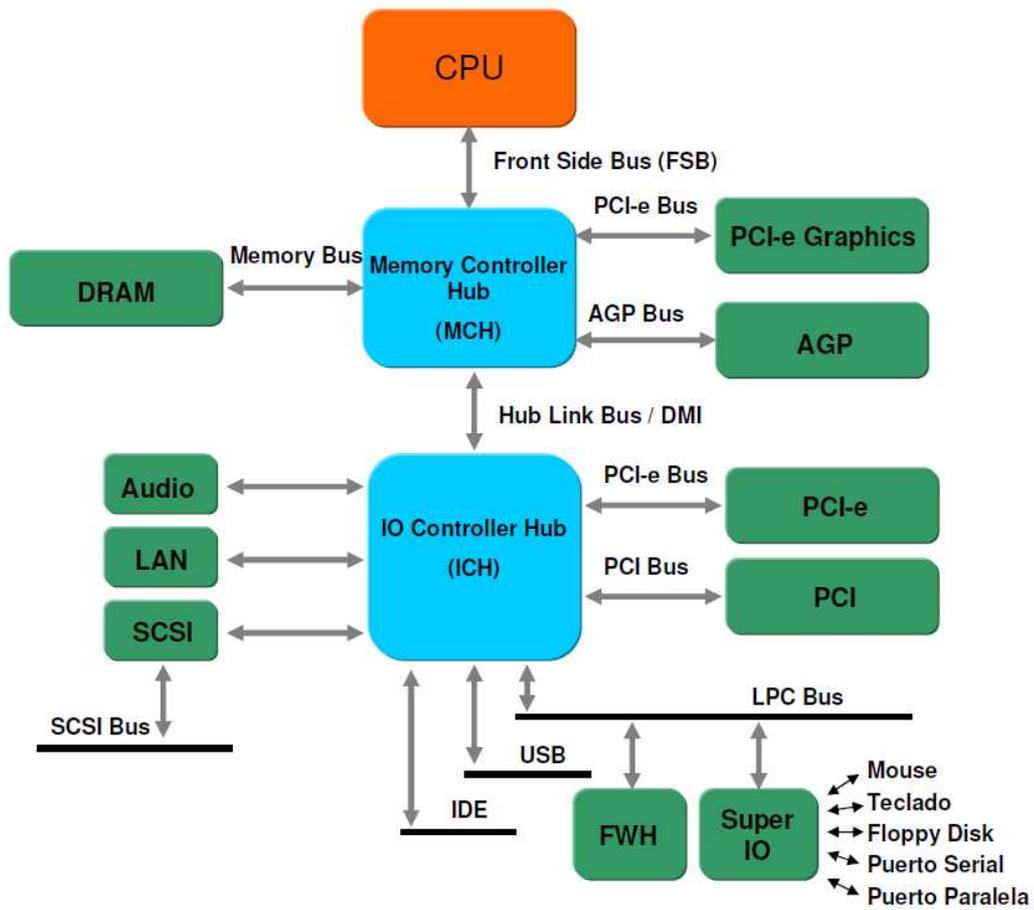
Con la aparición del bus PCI en 1993 y el bus AGP en 1997 la topología cambió y la mayor complejidad del sistema de memoria y la necesidad de un mejor manejo del DMA (Direct Memory Access) llevó a la aparición de dos controladores con funciones especializadas: el "Northbridge" que se encarga del control de los accesos a la memoria y de los buses especializados (como el AGP) y el "Southbridge" responsable del control de los demás buses y periféricos.

El esquema es:



A partir del año 2000, Intel introduce cambios y deja de utilizar el PCI como “bus central” del sistema, buscando lograr mejores velocidades de transferencia entre los distintos componentes. De allí que reemplaza la terminología “northbridge” y “southbridge” por la de “hubs”: MCH (Memory Controller Hub = Northbridge) y ICH (IO Controller Hub = Southbridge).

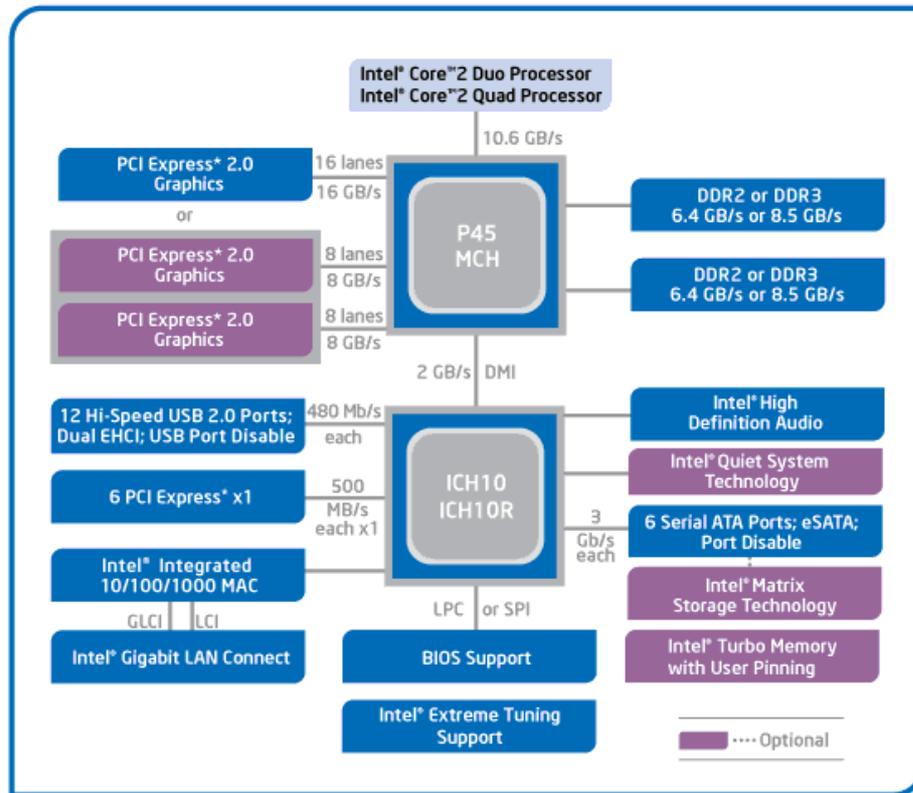
Estos dos controladores están inicialmente unidos por el “Hub Link Bus”, actualmente reemplazado por el DMI (Direct Memory Interface). Otra característica importante de este re-diseño es que desaparece el ISA Bus, y las conexiones con dispositivos de baja velocidad (el Super IO y el Firmware = BIOS) se hace a través de un nuevo bus: el LPC (Low Pin Count).



La tendencia a partir de 2007 fue eliminar el uso de los buses AGP y PCI y basar toda la E/S en el bus PCI-Express (PCI-e). Esta tendencia aparece asociada al desarrollo de los "chipsets" vinculados a los procesadores "Core 2" de Intel.

Por ejemplo en la figura siguiente se observa el esquema de bloques del "chipset" P45 de Intel.

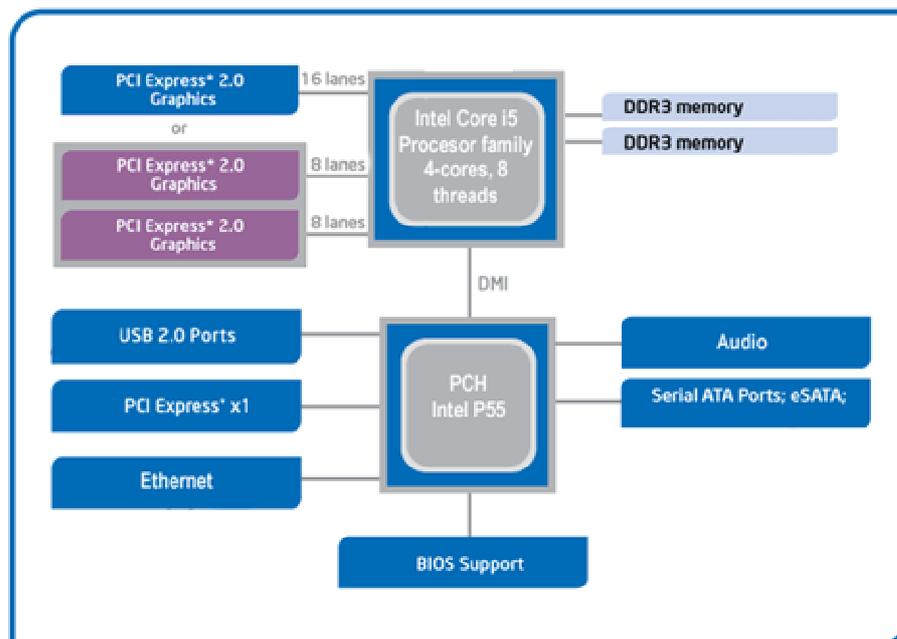
Ç



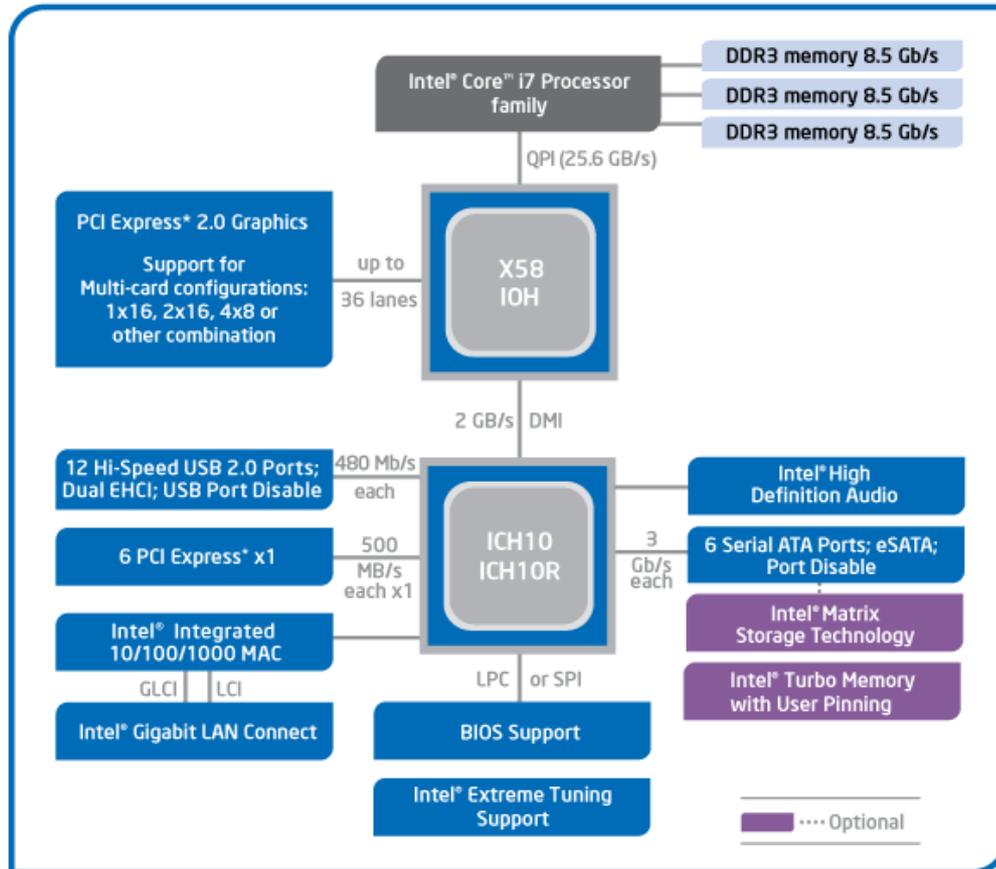
Intel® P45 Express Chipset Block Diagram

A su vez en la arquitectura de los procesadores “Core i5/i7” se coloca el MCH dentro de la CPU.

En el caso del Core i5 se busca un diseño simple de bajo costo, realizable con básicamente dos chips: la CPU y un PCH (Peripheral Controller Hub) que reemplaza al ICH.



Para el Core i7 se busca un diseño de alta performance con un IOH (Input Output Hub) destinado en exclusivo para los dispositivos de muy altos requerimientos de transferencia de datos, como los controladores gráficos.



Intel® X58 Express Chipset Block Diagram