

Connecting Processors, Memory and I/O Devices

Conexión de Procesadores, Memoria y Dispositivos de Entrada y Salida

Yeimy T. Albarracín Serrano, Anthony A. Caicedo Amorochó, y Johan M. Cepeda Ortíz
Universidad Industrial de Santander

Abstract—the following document is focused on describe the different media and protocols that can be used in the communication between the different interfaces and devices with the computer. (Abstract)

Keywords—interfaces; net; I/O(input / output); devices; bridge(key words)

Resumen—El siguiente documento está enfocado en describir los diferentes medios y protocolos que pueden ser usados en la comunicación entre las diferentes interfaces y dispositivos con el computador.

Keywords—interfaces; net; I/O(input / output); devices; bridge(key words)

I. INTRODUCTION (HEADING 1)

Un sistema computacional está formado tanto por interfaces como por protocolos que le permiten a las interfaces comunicarse entre sí enviando y recibiendo información comúnmente a través de buses, pero los buses tienen una serie de limitaciones que no nos permiten ampliar su capacidad de flujo de información más allá de sus limitaciones físicas, por esta razón se hace necesario encontrar otra forma de llevar los datos de una interfaz a otra para que se comuniquen entre sí, es así como en la industria de la computación se introduce una nueva manera de comunicar dos interfaces creando una conexión punto a punto de alta velocidad entre ellas superando de esta manera las limitaciones físicas que tienen los buses, estas nuevas conexiones punto a punto crean una interconexión de I / O que sirve para conectar nuevos periféricos y expandir la máquina pero para que estos periféricos puedan conectarse con distintas máquinas fue necesario estandarizar el tipo de conexión que hay entre los periféricos y las máquinas de esta manera el usuario final puede conectar un periférico con una máquina aunque no sean del mismo fabricante, de esta manera, gracias a las múltiples formas de intercambiar información entre las distintas interfaces que conforman un sistema computacional se abre la posibilidad de formar diferentes

configuraciones de una máquina, bajando los costos de una máquina de propósito general para el usuario común y potenciando la capacidad de las máquinas de propósito específico de los usuarios especializados.

II. ESTADO DEL ARTE

En la arquitectura tradicional de Intel (que es la más usada hasta modelos recientes), ese bus se llama front-side bus y es de tipo paralelo con 64 líneas de datos, 32 de direcciones además de múltiples líneas de control que permiten la transmisión de datos entre el procesador y el resto del sistema. Este esquema se ha utilizado desde el primer procesador de la historia, con mejoras en la señalización que le permite funcionar con relojes de 333 MHz haciendo 4 transferencias por ciclo.

En algunos procesadores de AMD y en el Intel Core i7 se han usado otros tipos para el bus principal de tipo serial. Entre estos se encuentra el bus HyperTransport de AMD, que maneja los datos en forma de paquetes usando una cantidad menor de líneas de comunicación, permitiendo frecuencias de funcionamiento más altas y en el caso de Intel, Quickpath

Los procesadores de Intel y de AMD poseen además un controlador de memoria de acceso aleatorio en el interior del encapsulado lo que hace necesario la implementación de buses de memoria del procesador hacia los módulos. Ese bus está de acuerdo a los estándares DDR de JEDEC y consisten en líneas de bus paralelo, para datos, direcciones y control. Dependiendo de la cantidad de canales pueden existir de 1 a 4 buses de memoria.

III. MARCO TEÓRICO

INTERCONEXIÓN CON BUSES:

Un bus es un camino de comunicación entre dos o más dispositivos. Una característica clave de un bus es que se trata

de un medio de transmisión compartido. Al bus se conectan varios dispositivos, y cualquier señal transmitida por uno de esos dispositivos está disponible para que los otros dispositivos conectados al bus puedan acceder a ella. [1]

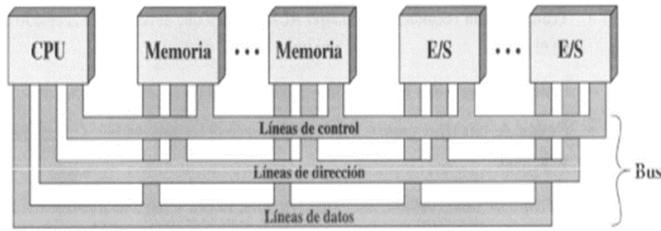


Fig. 1. Esquema de interconexión mediante un bus.

Los buses se clasifican tradicionalmente como buses de memoria de procesador o buses de I / O. [3]

- Los buses de memoria del procesador son cortos, generalmente de alta velocidad, y memoria para maximizar el ancho de banda del procesador de memoria.
- Buses de I / O, por contraste, puede ser largo, puede tener muchos tipos de dispositivos conectados a ellos, y a menudo tienen una amplia gama en el ancho de banda de datos de los dispositivos conectados a ellos.

TEMPORIZACIÓN:

El término temporización hace referencia a la forma en la que se coordinan los eventos en el bus. Los buses utilizan temporización síncrona o asíncrona [1]

- SÍNCRONA:** La presencia de un evento en el bus está determinada por un reloj. El bus incluye una línea de reloj a través de la que se transmite una secuencia en la que se alternan intervalos regulares de igual duración a uno y a cero.
- ASÍNCRONA:** La presencia de un evento en el bus es consecuencia y depende de que se produzca un evento previo.

I/O:

En arquitectura de computadores la unidad de entrada/salida (input/output) se encarga de interconectar la computadora con los dispositivos externos (periféricos) accesibles al usuario, como son el teclado, el monitor, la impresora, etc., por los cuales podemos ingresar o extraer información a o desde la CPU. Las I/O cuentan con interfaces (que generalmente realizan la adaptación de niveles eléctricos de las señales) y CONTROLADORES que son sistemas digitales para controlar periféricos específicos. [2]



Fig. 2. Conectores.

BACKPLANE

Un backplane es una placa de circuito (por lo general, una placa de circuito impreso) que conecta varios conectores en paralelo uno con otro, de tal modo que cada pin de un conector esté conectado al mismo pin relativo del resto de conectores, formando un bus de ordenador.

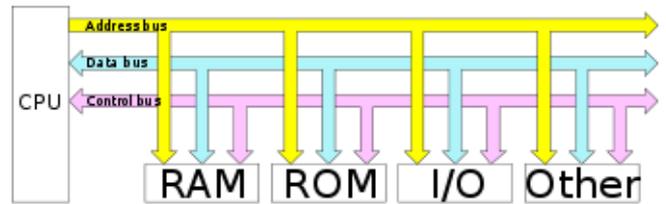


Fig. 3. Diagrama de bus backplane como extensión del bus del microprocesador donde se grafican los buses de direcciones, de datos, y de control, que van desde la CPU a la RAM, ROM, E/S y otros.

IV. COMUNICACIÓN INTERNA DE UNA COMPUTADORA

Un sistema informático está compuesto por diversos subsistemas que deben tener interfaces entre sí. Las tres unidades básicas en una computadora: la CPU, la memoria principal (MP) y el subsistema de E/S necesitan comunicarse. Durante muchos años, esto se ha hecho con buses, debido a su versatilidad y el bajo costo. Al definir un esquema de conexión único, los nuevos dispositivos se pueden agregar fácilmente, y los periféricos pueden incluso moverse entre los sistemas informáticos que utilizan el mismo tipo de bus. Además, los buses son rentables, porque un solo conjunto de cables se comparte de múltiples maneras.

Los primeros computadores tenían dos sistemas de buses, uno para la memoria y otro para los demás dispositivos. La CPU tenía que acceder a dos sistemas con instrucciones para cada uno, protocolos y sincronizaciones diferentes. La empresa DEC notó que el uso de dos buses no era necesario si se combinaban las direcciones de memoria con las de los periféricos en un solo espacio de memoria (mapeo). Entonces se basaron en basaban en la conexión de varias tarjetas de circuito impreso a un bus Backplane pasivo que servía de eje al sistema. En ese bus se conectaba la tarjeta de CPU que realiza las funciones de árbitro de las comunicaciones con las demás tarjetas de dispositivo conectadas; las tarjetas incluían la memoria, controladoras de disquete y disco, adaptadores de

vídeo. La CPU escribía o leía los datos apuntando a la dirección que tuviera el dispositivo buscado en el espacio único de direcciones haciendo que la información fluyera a través del bus principal.

El hecho de que el bus fuera pasivo y que usara la CPU como control, representaba varios problemas para la ampliación y modernización de cualquier sistema con esa arquitectura. Además, que la CPU utilizaba una parte considerable de su potencia en controlar el bus. Desde que los procesadores empezaron a funcionar con frecuencias más altas, se hizo necesario jerarquizar los buses de acuerdo a su frecuencia: se creó el concepto de bus de sistema (conexión entre el procesador y la RAM) y de buses de expansión, haciendo necesario el uso de un chipset.

La principal desventaja de un bus es que crea un cuello de botella de comunicación, posiblemente limitando el rendimiento máximo de I/O. Cuando I/O debe pasar a través de un único bus, el ancho de banda de ese bus limita el rendimiento máximo de I/O.

Una de las razones por las que el diseño del bus es tan difícil es que la velocidad máxima del bus es mayormente limitada por factores físicos: la longitud del bus y el número de dispositivos. Estos límites físicos nos impiden correr el bus arbitrariamente rápido. Además, el Necesitan soportar una gama de dispositivos con latencias y transferencias de datos muy variadas también hace que el diseño de buses sea un desafío.

Como se hizo difícil de ejecutar muchos cables paralelos a alta velocidad debido a la inclinación del reloj y la reflexión, la industria pasó de los buses a las interconexiones punto a punto de alta velocidad de la velocidad con los interruptores. Así, tales redes de I/O generalmente han reemplazado los buses de I/O en nuestros sistemas. En la evolución de los buses se les llama buses de tercera generación.

Los buses de tercera generación se caracterizan por tener conexiones punto a punto, a diferencia de los buses arriba nombrados en los que se comparten señales de reloj. Esto se logra reduciendo fuertemente el número de conexiones que presenta cada dispositivo usando interfaces seriales. Entonces cada dispositivo puede negociar las características de enlace al inicio de la conexión y en algunos casos de manera dinámica, al igual que sucede en las redes de comunicaciones.

La "tercera generación" de buses es ahora en proceso de venir al mercado, incluyendo HyperTransport e InfiniBand. Estos típicamente incluyen características que les permiten correr a las velocidades muy altas necesitadas para soportar memoria y tarjetas de vídeo, mientras también soportando bajas velocidades cuando hablan a dispositivos más lentos tal como unidades de discos. Estos también tienden a ser muy flexibles en términos de sus conexiones físicas, permitiéndolos a usarse ambos como buses internos, así como conectando diferentes maquinas juntas. Esto puede llevar a problemas complejos cuando al intentar servir peticiones diferentes, tanto del trabajo en estos sistemas preocupando el diseño de software, como opuesto a el propio hardware. En general esta tercera generación de buses tiende a parecerse más como una red [network] que el concepto original de un bus, con un protocolo

más alto sobre la cabeza necesitada que los primeros sistemas, mientras también permitiendo dispositivos múltiples para usar el bus en seguida.

LAS INTERCONEXIONES DE I/O DE LOS PROCESADORES

El procesador se conecta a periféricos a través de dos chips principales. El chip junto al procesador es la memoria controladora, comúnmente llamado puente norte, y el que está conectado a él es el concentrador del controlador de E/S, llamado puente sur.

Puente Norte: Es el chip que controla las funciones de acceso desde y hasta microprocesador, AGP o PCI-Express, memoria RAM, vídeo integrado (dependiendo de la placa) y puente sur. Su función principal es la de controlar el funcionamiento del bus del procesador, la memoria y el puerto AGP o PCI-Express. De esa forma, sirve de conexión (de ahí su denominación de "puente") entre la placa madre y los principales componentes de la PC: microprocesador, memoria RAM y tarjeta de vídeo AGP o PCI Express.

El puente norte de una placa madre, es el que determinará el número, velocidad y tipo de CPU y, la cantidad, velocidad y tipo de memoria RAM, que puede usar una computadora. La tecnología de fabricación de un puente norte es muy avanzada, y su complejidad, comparable a la de un microprocesador moderno.

Puente sur: Es también conocido como Concentrador de Controladores de Entrada/Salida (*I/O Controller Hub, ICH*), es un circuito integrado que se encarga de coordinar los diferentes dispositivos de entrada y salida y algunas otras funcionalidades de baja velocidad dentro de la tarjeta madre. El puente sur no está conectado a la CPU y se comunica con ella indirectamente a través del Puente Norte. Adicionalmente el puente sur puede incluir soporte para Ethernet, RAID, USB y Codec de Audio. El puente sur algunas veces incluye soporte para el teclado, el ratón y los puertos seriales, sin embargo, aún en el 2007 los computadores personales (PC) gestionaban esos recursos por medio de otro dispositivo conocido como Súper I/O. En los últimos modelos de placas el puente sur integra cada vez mayor número de dispositivos a conectar y comunicar por lo que fabricantes como AMD o VIA Technologies han desarrollado tecnologías como HyperTransport o Ultra V-Link respectivamente para evitar el efecto cuello de botella que se producía al usar como puente el bus PCI.

V. EJEMPLO HYPERTRANSPORT

HyperTransport (HT), también conocido como Lightning Data Transport (LDT) es una tecnología de comunicaciones bidireccional, que funciona tanto en serie como en paralelo, y que ofrece un gran ancho de banda en conexiones punto a punto de baja latencia. Se publicó el 2 de abril de 2001. Esta tecnología se aplica en la comunicación entre chips de un circuito integrado ofreciendo un enlace (ó bus) avanzado de alta velocidad y alto rendimiento; es una conexión universal que está diseñada para reducir el número de buses dentro de un sistema, suministrando un enlace de alto rendimiento a las aplicaciones incorporadas y facilitando sistemas de multiprocesamiento altamente escalables.

VI. EJEMPLO INFINIBAND

InfiniBand es un bus de comunicaciones serie de alta velocidad, baja latencia y de baja sobrecarga de CPU, diseñado tanto para conexiones internas como externas. Sus especificaciones son desarrolladas y mantenidas por la Infiniband Trade Association (IBTA).

Al igual que Fibre Channel, PCI Express y otros modos de interconexión modernos, Infiniband usa un bus serie bidireccional de tal manera que evita los problemas típicos asociados a buses paralelos en largas distancias (en este contexto, una habitación o edificio). A pesar de ser una conexión serie, es muy rápido, ofreciendo una velocidad bruta de unos 2,5 Gigabits por segundo (Gbps) en cada dirección por enlace. Infiniband también soporta doble e incluso cuádruples tasas de transferencia de datos, llegando a ofrecer 5 Gbps y 10 Gbps respectivamente. Se usa una codificación 8B/10B, con lo que, de cada 10 bits enviados solamente 8 son de datos, de tal manera que la tasa de transmisión útil es 4/5 de la media. Teniendo esto en cuenta, los anchos de banda ofrecidos por los modos simples, doble y cuádruple son de 2, 4 y 8 Gbps respectivamente.

Los enlaces pueden añadirse en grupos de 4 o 12, llamados 4X o 12X. Un enlace 12X a cuádruple ritmo tiene un caudal bruto de 120 gbps, y 96 Gbps de caudal eficaz. Actualmente, la mayoría de los sistemas usan una configuración 4X con ritmo simple, aunque los primeros productos soportando doble ritmo ya están penetrando en el mercado. Los sistemas más grandes, con enlaces 12X se usan típicamente en lugares con gran exigencia de ancho de banda, como clústeres de computadores, interconexión en superordenadores y para interconexión de redes.

La latencia teórica de estos sistemas es de unos 160ns. Las reales están en torno a los 6 μ s, dependiendo bastante del software y el firmware.

Infiniband usa una topología conmutada de forma que varios dispositivos pueden compartir la red al mismo tiempo (en oposición a la topología en bus). Los datos se transmiten en paquetes de hasta 4 kB que se agrupan para formar mensajes. Un mensaje puede ser una operación de acceso directo a memoria de lectura o escritura sobre un nodo remoto (RDMA),

un envío o recepción por el canal, una operación de transacción reversible o una transmisión multicast.

Al igual que en el modelo de canal usado en la mayoría de los mainframes, todas las transmisiones empiezan o terminan con un adaptador de canal. Cada procesador contiene un host channel adapter (HCA) y cada periférico un target channel adapter (TCA). Estos adaptadores también pueden intercambiar información relativa a la seguridad o a la calidad de servicio del enlace.

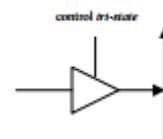


Fig. 4. InfiniBand.

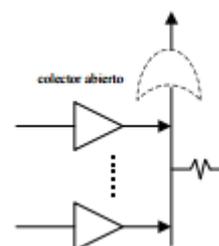
VII. DIRECCIONABILIDAD DE BUSES

Las líneas de un bus podemos también clasificarlas en función de su direccionalidad:

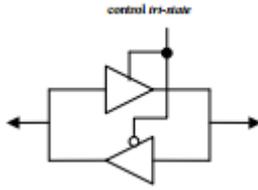
1. Líneas unidireccionales



• Emisor múltiple



2. Líneas bidireccionales



Desde el punto de vista físico un bus es un conjunto de conductores eléctricos paralelos dispuestos sobre una tarjeta de circuito impreso. Los dispositivos del sistema se conectan a través de conectores (slots) dispuestas a intervalos regulares a lo largo del bus.

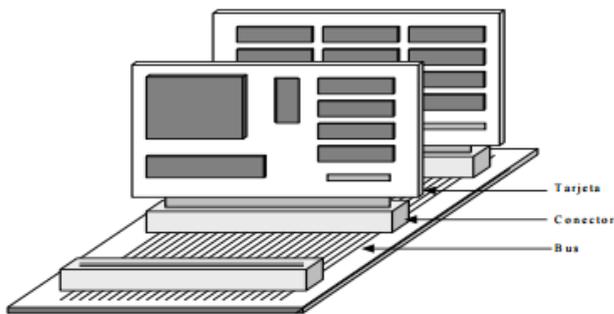


Fig. 5. Buses de sistema

La disposición anterior corresponde generalmente a los buses del sistema. Los denominados buses de entrada/salida conectan las unidades a través de cables.

Los principales parámetros que caracterizan un bus son:

- Ancho de banda: velocidad de transferencia medida en Mb/s
- Anchura del bus: número de líneas que lo componen
- Ancho de datos: número de líneas de datos
- Capacidad de conexión: número de unidades conectadas al bus

VIII. BUSES LOCALES

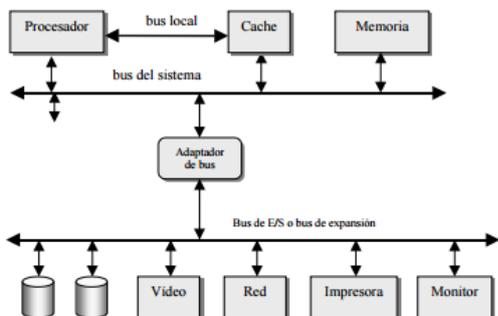


Fig. 6. Diagrama de buses locales.

El bus local es de longitud pequeña, de alta velocidad, y adaptado a la arquitectura particular del sistema para maximizar el ancho de banda entre el procesador y la caché, por eso suele ser un bus propietario. Este bus aísla el tráfico procesador-caché del resto de transferencias del sistema.

IX. BUSES DE E/S O DE EXPANSIÓN

El bus de E/S o de expansión reduce el tráfico en el bus del sistema, de manera que el procesador puede acceder a memoria en un fallo de caché mientras realiza una operación de entrada/salida.

Los buses de expansión son buses estándar o abiertos (ISA, EISA, PCI, etc.) es decir, independientes del computador y con unas características bien definidas en el correspondiente documento de normalización. La existencia de estos buses permite diseñar una amplia gama de controladores de periféricos compatibles.

Para conectar los buses del sistema y de expansión se requiere un Adaptador de Bus, dispositivo que permite adaptar las distintas propiedades de ambos buses: velocidad, carácter síncrono o asíncrono, multiplexación, etc.

El proceso de jerarquización se puede complicar más con otras topologías que den cabida a dispositivos de distinta velocidad. De esta forma se equilibra mejor el tráfico de información en sistemas que operan con muchos dispositivos conectados. En la figura siguiente se muestra una topología jerárquica en la que se contempla un bus de alta velocidad del que cuelgan dos buses, uno de expansión para dispositivos rápidos y otro de E/S para dispositivos lentos.

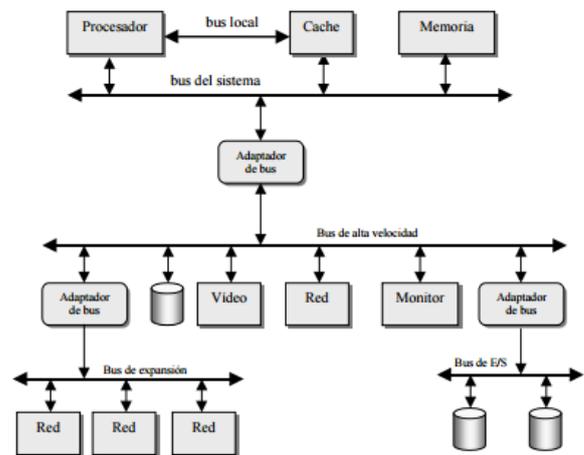


Fig. 7. Diagrama expandido de buses locales.

X. BUSES NORMALIZADOS

Las especificaciones de un bus estándar están perfectamente definidas en un documento de estandarización respaldado por

alguna sociedad de prestigio en el área (IEEE, etc.). En las especificaciones se distinguen varios niveles:

- Nivel mecánico, en el que se recoge la forma y tamaño de los conectores, el número de contactos por conector y el número de dispositivos que soporta el bus.
- Nivel eléctrico, en el que se especifican los valores de tensión de las diferentes señales, polarización, etc.
- Nivel lógico, especifica la función de cada señal del bus: direcciones, datos, control, etc.
- Nivel de temporización o sincronismo, que especifica el protocolo de transferencia
- empleado
- Nivel de arbitraje, especifica el protocolo de arbitraje que utiliza el bus.

1. BUS PCI (PERIPHERAL COMPONENT INTERCONNECT)

El bus PCI es un bus de ancho de banda elevado e independiente del procesador. El estándar actual permite el uso de hasta 64 líneas de datos a 66 MHz, para una velocidad de transferencia de 528 MBytes/s, o 4,224 Gbps. El PCI está diseñado para permitir una cierta variedad de configuraciones basadas en microprocesadores, incluyendo sistemas con uno o varios procesadores. Utiliza temporización semisíncrona y un esquema de arbitraje centralizado.

1.1. Estructura

El bus PCI puede configurarse como un bus de 32 o 64 bits. La siguiente tabla define las líneas más importantes obligatorias en el PCI:

CLK (reloj) Señal de reloj que es muestreada en el flanco de subida.

RST# (reset) Hace que todos los registros y señales específicas del PCI pasen al estado inicial.

Señales de direcciones y datos

AD[31:0] Incluye 32 líneas para datos y direcciones multiplexadas en el tiempo.

C/BE[3:0]# Se utilizan para interpretar y validar las líneas de datos y direcciones.

Señales de control de interfaz

FRAME# Activada por el master para indicar el comienzo y la duración de una transferencia. Las activa al comienzo y la desactiva al final de la fase de datos.

IRDY# Señal de master preparado (Initiator Ready). La proporciona el master actual del bus (el iniciador de la transacción). Durante una lectura, indica que el master está preparado para aceptar datos; durante una escritura indica que el dato válido está en AD.

TRDY# Señal de slave preparado (Target Ready). La activa el slave al principio de la transferencia, y la desactiva cuando no puede completar la transferencia en un solo ciclo de reloj

DEVSEL# Señal de slave (dispositivo) seleccionado (Device Select). Activada por el slave cuando ha reconocido su dirección.

Señales de arbitraje

REO# Indica al árbitro que el dispositivo correspondiente solicita utilizar el bus. Es una línea punto-a-punto específica para cada dispositivo.

GNT# Indica al dispositivo que el árbitro le ha cedido el acceso al bus. Es una línea punto-a-punto específica para cada dispositivo.

1.2. Ordenes

La actividad del bus consiste en transferencias entre dos elementos, denominándose Initiator (master) al que inicia la transacción. Cuando un master del bus adquiere el control del mismo, determina el tipo de transferencia que se producirá a continuación. Los tipos, entre otros son los siguientes:

- Reconocimiento de interrupción
- Lectura de E/S
- Escritura en E/S
- Lectura de memoria
- Escritura en memoria

1.3. Transferencia de datos

Toda transferencia de datos en el bus PCI es una transacción única, que consta de una fase de direccionamiento y una o más fases de datos. La siguiente figura muestra la temporización de una operación de lectura. Todos los eventos se sincronizan en las transiciones de bajada del reloj, cosa que sucede a la mitad de cada ciclo de reloj.

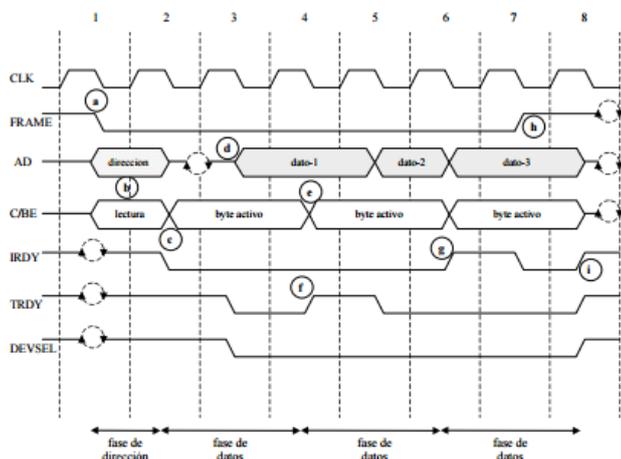


Fig. 8. Tipos de señales.

Los dispositivos interpretan las señales del bus en los flancos de subida, al comienzo del ciclo. A continuación, se describen los eventos significativos señalados en el diagrama:

a) Una vez que el master ha obtenido el control del bus, inicia la transacción:

- activando FRAME, que permanece activa hasta la última fase de datos
- situando la dirección de inicio en el bus de direcciones
- situando la orden de lectura en las líneas C/BE.

b) El slave reconoce su dirección en las líneas AD al comienzo del ciclo de reloj 2.

c) El master deja libre las líneas AD del bus y cambia la información de las líneas C/BE para indicar qué líneas AD se utilizan para transportar datos (de 1 a 4 bytes). También activa IRDY para indicar que está preparado para recibir el primer dato (*).

d) El slave activa DEVSEL para indicar que ha reconocido su dirección. Después sitúa el dato solicitado en las líneas AD y activa TRDY para indicar que hay un dato válido en el bus.

e) El master lee el dato al comienzo del ciclo de reloj 4 y cambia las líneas C/BE según se necesite para la próxima lectura.

f) En este ejemplo el slave necesita algún tiempo para preparar el segundo bloque de datos para la transmisión. Por tanto, desactiva TRDY para señalar al master que no proporcionará un nuevo dato en el próximo ciclo. En consecuencia, el master no lee las

líneas de datos al comienzo del quinto ciclo de reloj y no cambia la señal C/BE durante ese ciclo. El bloque de datos es leído al comienzo del ciclo de reloj 6.

g) Durante el ciclo 6 el slave sitúa el tercer dato en el bus. No obstante, en este ejemplo, el master todavía no está preparado para leer el dato. Para indicarlo, desactiva IRDY. Esto hará que el slave mantenga el tercer dato en el bus durante un ciclo de reloj extra.

h) El master sabe que el tercer dato es el último, y por eso desactiva FRAME para indicárselo al slave. Además, activa IRDY para indicar que está listo para completar esa transferencia.

i) El master desactiva IRDY, haciendo que el bus vuelva a estar libre, y el slave desactiva TRDY y DEVSEL.

1.4. Arbitraje

El bus PCI utiliza un esquema de arbitraje centralizado síncrono, en el que cada maestro tiene una señal propia de petición (REQ) y cesión (GNT) del bus. Estas líneas se conectan a un árbitro central. La especificación PCI no indica un algoritmo particular de arbitraje. El árbitro puede utilizar un procedimiento de primero en llegar primero en servirse, un procedimiento de cesión cíclica (round-robin), o cualquier clase de esquema de prioridad. El maestro del PCI establece, para cada transferencia que desee hacer, si tras la fase de dirección sigue una o más fases de datos consecutivas.

La siguiente figura es un ejemplo en el que se arbitra a cuál de los dispositivos A y B se cede el bus. Se produce la siguiente secuencia:

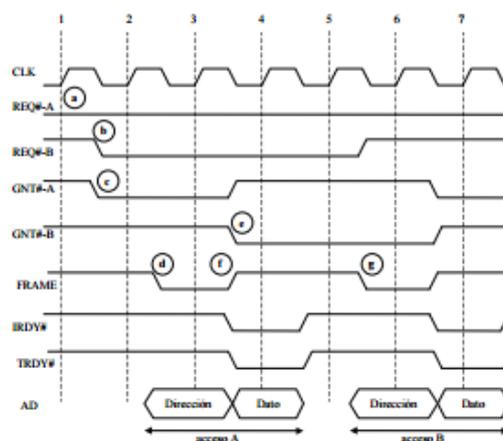


Fig. 9. Señales de arbitraje.

a) En algún momento anterior al comienzo del ciclo de reloj 1, A ha activado su señal REQ. El árbitro muestrea esa señal al comienzo del ciclo de reloj 1.

b) Durante el ciclo de reloj 1, B solicita el uso del bus activando su señal REQ.

c) Al mismo tiempo, el árbitro activa GNT-A para ceder el acceso al bus a A.

d) El maestro del bus A muestrea GNT-A al comienzo del ciclo de reloj 2 y conoce que se le ha cedido el acceso al bus. Además, encuentra IRDY y TRDY desactivados, indicando que el bus está libre. En consecuencia, activa FRAME y coloca la información de dirección en el bus de direcciones, y la orden correspondiente en las líneas C/BE. Además mantiene activa REQ-A, puesto que tiene que realizar otra transferencia después de la actual.

e) El árbitro del bus muestrea todas las líneas GNT al comienzo del ciclo 3, y toma la decisión de ceder el bus a B para la siguiente transacción. Entonces activa GNT-B y desactiva GNT-A. B no podrá utilizar el bus hasta que éste no vuelva a estar libre.

f) A desactiva FRAME para indicar que la última transferencia de datos está en marcha. Pone los datos en el bus de datos y se lo indica al dispositivo destino con IRDY. El dispositivo lee el dato al comienzo del siguiente ciclo de reloj.

g) Al comienzo del ciclo 5, B encuentra IRDY y FRAME desactivados y, por consiguiente, puede tomar el control del bus activando FRAME. Además, desactiva su línea REQ, puesto que sólo deseaba realizar una transferencia.

Hay que resaltar que el arbitraje se produce al mismo tiempo que el maestro actual del bus está realizando su transferencia de datos. Por consiguiente, no se pierden ciclos de bus en realizar el arbitraje. Esto se conoce como arbitraje oculto o solapado (hidden arbitration).

XI. BUS USB (UNIVERSAL SERIAL BUS)

El bus USB (Universal Serial Bus) es un bus normalizado para la conexión de periféricos, desarrollado por un conjunto de empresas de informática y telecomunicaciones (7 compañías: Compaq, DEC, IBM, Intel, Microsoft, NEC y Northern Telecom). Permite conectar de forma sencilla dispositivos periféricos al computador, sin necesidad de reiniciarlo ni de configurar el sistema. Se pueden conectar hasta 127 dispositivos, con una longitud máxima de cable de 5 metros

para cada uno, con lo que una conexión en cadena permitiría que el último dispositivo estuviese a 635 metros del ordenador. Trabaja en dos modos, a baja velocidad, 1,5 Mbps, para dispositivos lentos como teclados y ratones, y a alta velocidad, 12 Mbps, para dispositivos rápidos, como CDROM, módems, etc. Utiliza un cable de cuatro hilos, dos para datos y dos para alimentación. El bus USB está organizado en una estructura de árbol descendente, con unos elementos especiales, llamados hubs que encaminan las señales desde un dispositivo al host o viceversa. En la raíz está el host, que es el interfaz entre el bus USB y el bus del ordenador. De él cuelgan los dispositivos USB y los hubs, que también son dispositivos USB. A un hub se puede conectar uno o más dispositivos, que a su vez pueden ser otros hubs.

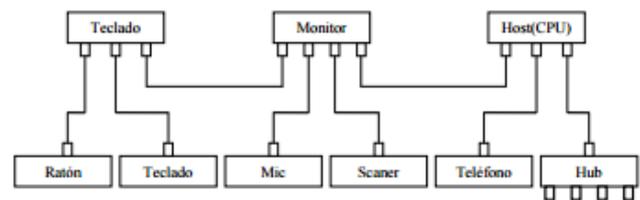


Fig. 10. Diagrama de conexiones seriales.

Cuando se conecta un dispositivo, se detecta la diferencia de tensión en la red USB y procede a determinar las características del dispositivo (vendedor, funcionalidad, ancho de banda requerido, etc.). El host le asigna una dirección única ID para diferenciarlo del resto de los dispositivos de la red USB. Después el SO carga los drivers del dispositivo, solicitándolos al usuario si es necesario. Cuando se desconecta el dispositivo, el host lo detecta y descarga los drivers. El host USB tiene, entre otras, las siguientes funciones:

- Detectar la conexión de nuevos dispositivos al sistema.
- Detectar la desconexión de dispositivos previamente conectados,
- Enumerar y configurar los dispositivos conectados al sistema.
- Administrar y controlar el flujo de datos entre el host y los dispositivos USB.
- Administrar y controlar las transferencias síncronas y asíncronas de información.
- Recoger y resumir estadísticas de actividad y estado de los elementos del sistema.
- Proporcionar energía eléctrica a algunos dispositivos (teclado, ratón, etc.).

Un puerto serie es capaz de transmitir hasta 112,5 KB/s y un puerto paralelo entre 600KB/s y 15MB/s, sin embargo, la velocidad de transferencia de un puerto USB está entre 1,5MB/s y 12MB/s. El FIREWIRE (IEEE 1394), maneja transferencias entre 100MB/s y 400MB/s, que permite conectar hasta 63 dispositivos y un cable de 4.5 metros por dispositivo, permitiendo al igual que el USB la conexión en

operación. Uno de los problemas del puerto USB es que suministra solamente 500 miliamperios de corriente para los dispositivos conectados, que, aunque es suficiente para la mayoría de los dispositivos que se conectan a este puerto, resulta pequeña cuando conectamos varios dispositivos sin fuente de alimentación propia. Estructura de Computadores, Facultad de Informática, UCM, Curso 11-12 16 Otra de las funciones importantes de los hubs es la de aislar a los puertos de baja velocidad de las transferencias de alta velocidad, proceso sin el cual todos los dispositivos de baja velocidad conectados al bus entrarían en colapso. La protección de los dispositivos lentos de los rápidos ha sido siempre un problema importante en el diseño redes mixtas, como es USB. El hub está compuesto por dos unidades principales: el Controlador y el Repetidor. El Repetidor tiene la función de analizar, corregir y retransmitir la información que llega al hub hacia los puertos del mismo. Contiene una memoria de varios registros para sus funciones. El Controlador es una pequeña CPU de supervisión de las múltiples funciones realiza un hub. Todos los dispositivos conectados acceden al canal o medio para transmitir sus datos siguiendo un protocolo análogo al token ring, que consecutivamente va dando la posibilidad de transmisión a cada periférico.

XII. EJEMPLO DE CONEXIÓN PCIEX16 (TARJETA GRAFICA)

En la actualidad las tarjetas gráficas usan una conexión de bus PCIex16 y las más avanzadas usan PCIex16 2.0 que ya maneja un ancho de banda de 3200 / 6400 MB/s el cual es el ancho de banda más alto actualmente, las tarjetas gráficas son capaces de transportar datos a estas velocidades ya que tienen una conexión con la cpu a través del puente norte lo cual hace que la distancia entra la gpu y la cpu sea mucho menor y por tanto los datos puedan viajar a altas velocidades, de esta manera la gpu y la cpu pueden también compartir el acceso a la memoria RAM para realizar sus respectivos procesos e incluso algunas tarjetas gráficas pueden trabajar en paralelo junto con otras tarjetas gráficas a través de tecnologías SLI o CrossFire dependiendo del chip de procesamiento gráfico que use cada tarjeta grafica

XIII. EJEMPLO DE CONEXIÓN EN SERIE (MONITOR)

La conexión con el monitor es una conexión unidireccional ya que los datos solo son enviados en un solo sentido desde la cpu al monitor debido a que el monitor no es un dispositivo de entrada sino de salida, pero en la actualidad existen monitores que cuentan con puertos usb dichos puertos no envían la información de los dispositivos que se conectan a ellos a través del cable de interface visual que use el correspondiente

monitor sino a través de otro cable usb que se conecta del monitor a otro puerto usb del computador, en otras palabras dichos monitores cuentan con una bahía de expansión para puertos usb pero dicha comunicación sigue siendo a través del puerto usb de la computadora y por tanto la comunicación de la cpu con el monitor para mostrar la imagen en pantalla sigue siendo en un solo sentido.

XIV. CONCLUSIONES

- Se puede concluir que debido a que la velocidad máxima del bus es mayormente limitada por factores físicos, impide correr el bus arbitrariamente rápido.
- Las interconexiones punto a punto, se implementaron debido a que los buses que tradicionalmente se utilizaban para la comunicación, hacía que fuera difícil ejecutar muchos cables paralelos a alta velocidad.
- Se concluye que la temporización síncrona en un bus implica poca lógica, además lo hace funcionar rápidamente y la lógica de la interfaz puede ser pequeño.
- Debido a que las interconexiones asíncronas no están sincronizadas, permiten una amplia variedad de dispositivos y además el bus puede alargarse sin preocuparse por los problemas de desviación o sincronización del reloj. Permitiendo la comunicación de nuevos periféricos que no se hubieran podido comunicar con el anterior método de temporización (síncrono).

REFERENCIAS

- [1] William Stallings. (2005) Organización y Arquitectura de Computadores – 7° edición, Madrid. pp.77-83.
- [2] Santiago Cristobal Pérez, Higinio Facchini, Daniel M. Argüello. (2015) Arquitectura de computadoras, Argentina. pp.110-111.
- [3] David A. Patterson, John L. Hennessy. Computer Organization and Design, *The hardware / Software interface* pp. 582-586.
- [4] [https://es.wikipedia.org/wiki/Bus_\(inform%C3%A1tica\)#Primera_generaci.C3.B3n](https://es.wikipedia.org/wiki/Bus_(inform%C3%A1tica)#Primera_generaci.C3.B3n)
- [5] <https://es.wikipedia.org/wiki/HyperTransport>
- [6] <https://es.wikipedia.org/wiki/InfiniBand>
- [7] <http://www.upv.es/amiga/275.htm#0>
- [8] M. Young, *The Technical Writer's Handbook*. Mill Valley, CA: University Science, 1989.