

(12)

EPS



Estructura y Tecnología de Computadores II (2º)

Tema 12 Recursos hardware

Ingeniería Informática
EPS - UAM

20.07.23

(12)

Índice

- Descripción de los terminales del 8086
- Modos de funcionamiento del 8086: Mínimo y Máximo
- El Generador de Reloj 8284
- Arbitraje del Bus (DMA)
- Ejemplo de sistema con un 8086 en modo Mínimo
- El Controlador de Bus 8288
- Ejemplo de sistema con un 8086 en modo Máximo
- Ciclo de Bus
- Estados de Espera
- Buses en la arquitectura PC: ISA (EISA), PCI, IDE, SCSI, AGP, PCMCIA
- Nuevos puertos en el PC: USB, IrDA

20.07.23

(12)

Características del 8086

- En 1997, INTEL presenta el 8086, una de las primeras CPUs de 16 bits y mantiene la compatibilidad con CPUs anteriores de la familia (8085, 8088).
- La compatibilidad obliga a mantener la estructura interna básica de la nueva CPU pero ahora es necesario manejar un bus de direcciones con 20 bits, no 16. Por ello, se mantienen los registros internos de 16 bits del 8085 que ahora pasan a ser los registros SEGMENTO y se añaden nuevos registros de 16 bits: DESPLAZAMIENTO u OFFSET (nueva BIU o Unidad de Intefaz con el Bus), además de introducir un nuevo concepto característico de INTEL: SEGMENTACIÓN DE LA MEMORIA (segmentos de 64 Kbytes)
- El aumento del número de terminales (8 a 16 bits del B. Datos y de 16 a 20 bits del B. Direcciones) se resuelve con terminales MULTIFUNCIÓN y hardware auxiliar externo.

20.07.23

(12)

Descripción del 8086 (I)

Nombre	DIR	Función	Min.	Max.
AD0...AD15	E/S	Bus de Datos y de Dirección Multiplexado	SI	SI
A19/S6	S	B. Dir / No se usa	SI	SI
A18/S5	S	B. Dir / Estado del bit IF (SR)	SI	*
A17/S4	S	B. Dir / Indican el segmento accedido (CS, SS, DS, ES) para control externo memoria	SI	*
A16/S3	S		SI	*
ALE	S	Validación de una dirección en el bus	SI	*
DT/#R	S	Indica sentido de los datos en el bus de datos	SI	*
#DEN	S	Validación de un dato en el bus	SI	*(1)
#BHE/S7	S	Validación de un dato en Byte Alto del bus de datos / S7 no se usa	SI	SI
READY	E	Introducción de ciclos de espera (desactivada)	SI	SI
INTR	E	Entrada de interrupción enmascarable	SI	SI
NMI	E	Entrada de interrupción no enmascarable	SI	SI
#INTA	S	Reconocimiento de interrupción	SI	*
#RESET	E	Entrada de Reset (salto de CPU a FFFF:0000)	SI	SI

20.07.23

(12)

Descripción del 8086 (II)

Nombre	DIR	Función	Min.	Max.
#TEST	E	Sincronización del programa con el Coprocesador u otro hardware externo	SI	SI
#LOCK	S	Se activa mediante instrucción LOCK. Indica que la CPU no cede los buses	NO	SI
CLK	E	Entrada de reloj de la CPU	SI	SI
VCC	E	Alimentación de la CPU	SI	SI
GND	E	Masa de la CPU	SI	SI
MN#/MX	E	Control del Modo Mínimo y del Máximo	SI	SI
M#/IO	S	Indica si acceso a bus es a Memoria o a E/S	SI	*
#RD	S	Indica orden de lectura a Memoria o a E/S	SI	SI
#WR	S	Indica orden de escritura a Memoria o a E/S	SI	*
HOLD	E	Peticion de cesion del bus.	SI	NO
HOLDA	S	Reconocimiento de cesion del bus	SI	NO
#RQ#/GT0	E/S	Dos canales de petición/reconocimiento de cesion de bus	NO	SI
#RQ#/GT1	E/S		NO	SI

20.07.23

(12)

Descripción del 8086 (III)

Nombre	DIR	Función	Min.	Max.
QS0, QS1	S	Indican el estado de la cola de instrucciones del Coprocesador. También se encuentran en este grupo (relacionado con el Coprocesador) la entrada TEST# de la CPU, relacionada con la instrucción WAIT.	NO	SI
#S0, #S1, #S2	S	Señales de Estado para el Controlador de Bus 8288. Indican el tipo de Ciclo de Bus	NO	SI

S3	S4	Registro de segmento utilizado
0	0	ES
0	1	SS
1	0	CS (o ninguno)
1	1	DS

- * La señal es generada por el Controlador de Bus 8288 a partir de las señales de estado (S0#, S1#, S2#)
- (1) En el modo Máximo el 8288 genera DEN, activa a nivel alto

20.07.23

(12) **Modos de Funcionamiento del 8086: Mínimo y Máximo**

- **Modo Mínimo:**
 - Pequeños sistemas con una CPU.
 - Señales de control sencillas, generadas directamente por la CPU.
 - No pueden funcionar CPU y Coprocesador juntos.
- **Modo Máximo:**
 - Las señales de control de los buses se generan con una pastilla auxiliar: Controlador de Bus 8288
 - Pueden funcionar varias CPUs si se incorpora un circuito adicional, el Árbitro de Bus 8289, siguiendo un estándar de INTEL llamado MULTIBUS
 - Se puede incorporar un Coprocesador matemático
 - Permite implementar sistemas más completos, con capacidades de control y gestión mayores de los sistemas operativos (SO)

(12) **Generador de Reloj 8284 (I)**

- Los sistemas basados en el 8086 necesitan un circuito auxiliar (8284) como Generador de Reloj para poder disponer de varias señales de control además del patrón de reloj para la CPU

(12) **Generador de Reloj 8284 (II)**

- **F/C#:** señal de control de la entrada **EFI** (Entrada de Frecuencia Externa). Permite configurar al 8284 para que funcione en modo esclavo, sin **XTAL** propio, aplicando una señal de reloj procedente de la salida **OSC** de otro 8284, en el terminal **EFI**. Para ello es necesario que **F/C#** esté a "1".
- La salida **RESET** del 8284 se conecta a la CPU y aunque se activa al poner a "0" la entrada **#RES** del mismo circuito, cumple con las características (sincronización, duración) impuestas por el fabricante de la CPU para esta señal que hace saltar al procesador a la dirección **FFFF:0000** donde se encuentra la rutina de inicio del sistema.
- La señal **READY** generada por el 8284 cumple los cánones de temporización que requiere la CPU (8086) e incluye la lógica necesaria para utilizar dos posibles entradas de la señal **READY**: **RDY1** habilitada por **#AEN1** y **RDY2** habilitada por **#AEN2**.
- Las señales **CSYNC** y **#ASYNC** proporcionan diversas posibilidades de sincronización en sistemas con varios procesadores.
- **PCLK** y **OSC** son señales de reloj que se disponen con otras finalidades (otros posibles circuitos hardware que puedan necesitarlas y que formen parte del computador).

(12) **Protocolo de Gestión de Buses**

(12) **Organización de la Memoria (I)**

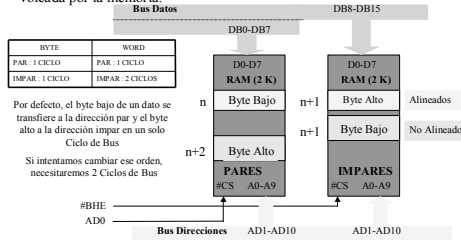
- El 8086 es una CPU de **16 bits** pero también puede manejar datos de **8 bits**.
 - Ello implica que la memoria del sistema sea diseñada adecuadamente para que pueda ser accedida a nivel de byte.
 - Por ello, es necesario construir **2 bancos de memoria** (o considerar que el Mapa de Memoria se divide en 2 partes, cada una de ellas de 512 Kb).
 - Ambos bancos se direccionan simultáneamente utilizando las señales **A1...A19** del Bus de Direcciones.
 - La señal **A0** del Bus de Direcciones se utiliza para habilitar (CS) el banco que contiene las direcciones PARES (Byte BAJO del Bus de Datos, es decir, D0...D7)
 - La señal **#BHE** de la CPU se utilizará para habilitar (CS) el banco de direcciones IMPARES (Byte ALTO del Bus de Datos, D8...D15).
- **Un dato tipo byte** puede estar en una dirección par o impar, consumiendo **sólo un Ciclo de Bus** su lectura o escritura, ya que la memoria está organizada en bytes, numerados de forma correlativa desde 00000H a FFFFFH.).

(12) **Organización de la Memoria (II)**

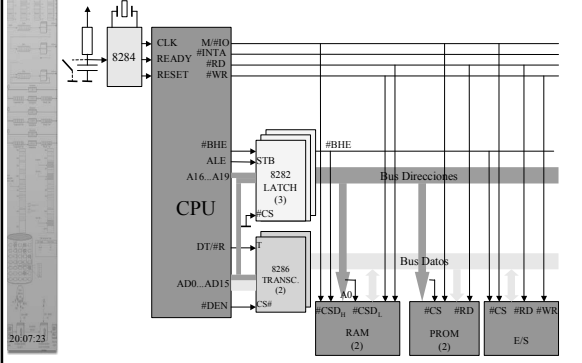
- Un dato tipo word puede ocupar el **par dirección par (byte bajo)-dirección impar (byte alto)** y se dice que está **alineado**.
 - Cuando un dato tipo word está alineado, al acceder a una dirección par para su lectura o escritura, la CPU activa tanto **A0** (que se pone a "0" al ser dirección par) como **#BHE** (se activa automáticamente, poniéndose a "0"). De ese modo se accede a las **2 direcciones de forma simultánea, en un sólo Ciclo de Bus**.
- Puede ocupar el **par dirección impar (byte bajo)-dirección par (byte alto)** y se dice que está **no alineado**, sin ninguna restricción.
 - Cuando un word no está alineado, su lectura o escritura conlleva **2 Ciclos de Bus** en lugar de uno. En el primer ciclo se accede al byte bajo (byte impar) activando **#BHE** y en el segundo ciclo se accede al byte alto (byte par) con **A0** a "0", al incrementarse la dirección

(12) Organización de la Memoria (III)

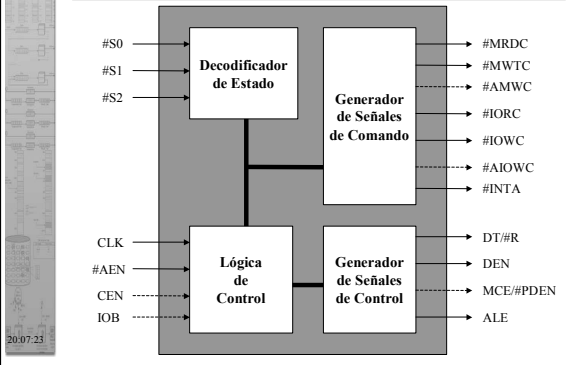
- El uso de A0 y BHE# es fundamental en las operaciones de escritura para asegurar que los datos se escriben en las direcciones adecuadas. En los casos de sólo lectura (ej. Memoria ROM) no es necesario que formen parte de la Lógica de Selección que activa su CS (chip select), ya que la CPU ignora la parte del Bus de Datos por donde no espera ninguna información, aunque haya sido volcada por la memoria.



(12) Ejemplo de Sistema basado en el 8086 en Modo Mínimo



(12) Controlador de Bus (8288) (I)



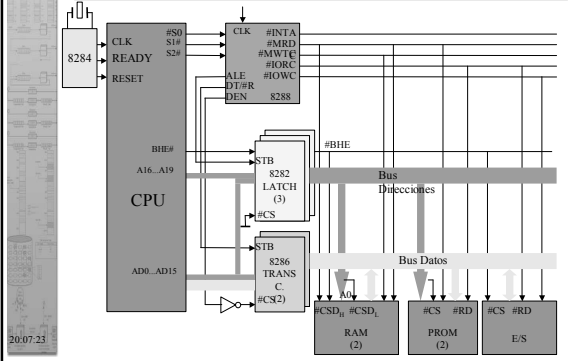
(12) Controlador de Bus (8288) (II)

- Para que el 8288 funcione, CEN a "1" y AEN# a "0" (habilitan las señales de comando).
- AMWC# (Comando de Escritura en Memoria Avanzado) y AIOWC# (Comando de Escritura en dispositivo E/S Avanzado) son semejantes a MWTC# y IOWC# pero con una temporización diferente.
- MRDC# comando que indica una lectura en memoria
- MWRC# comando que indica una escritura en memoria
- IORC# comando que indica una lectura en dispositivo de E/S
- IOWC# comando que indica una escritura en dispositivo de E/S
- PDEN# es una señal de control equivalente a DEN pero para periféricos
- DT/R#, DEN, ALE son las mismas que en el 8086

(12) Información de las Señales de Estado (Status)

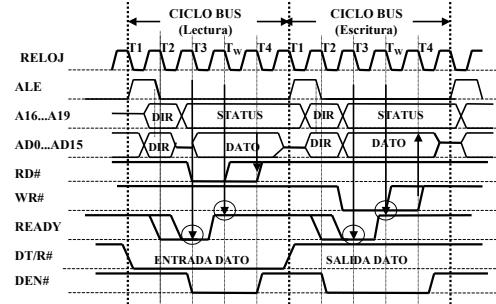
#S2	#S1	#S0	Tipo de ciclo del bus
0	0	0	Reconocimiento de Interrupción
0	0	1	Lectura de un puerto de E/S
0	1	0	Escritura de un puerto de E/S
0	1	1	Estado de Parada (HALT)
1	0	0	Fetch de un Código de Operación
1	0	1	Lectura de Memoria
1	1	0	Escritura en Memoria
1	1	1	No se accede al bus

(12) Ejemplo de Sistema basado en el 8086 en Modo Máximo



(12) Ciclo de Bus (I)

- Secuencia de Estados que ocurren en los accesos de la CPU a Memoria o a E/S



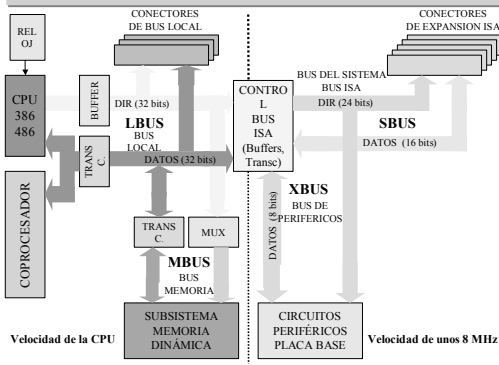
20.07.23

(12) Ciclo de Bus (II)

- Si la señal READY no es un "1" al comienzo de T3 o de cada T_w se inserta (por parte de la CPU, de forma automática) un nuevo Estado de Espera (T_w).
- La señal READY debe activarse con un circuito exterior, relacionado con la Lógica de Selección.
 - Este circuito, en el caso del PC es un contador cuyo valor inicial de carga depende de una variable del SETUP.
 - El contador prolonga un número de ciclos determinado el tiempo que la señal READY se mantendrá a "0".

20.07.23

(12) Sistema de Buses del PC-AT



20.07.23

Velocidad de la CPU

Velocidad de unos 8 MHz

(12) Buses Internos (I)

- Los elementos de la arquitectura PC están conectados a diferentes buses:
 - Bus Local (LBUS)
 - Formado por los buses de direcciones, datos y control de la CPU (misma anchura y velocidad). Al depender de la CPU, ha evolucionado con ella (desde el 8086 a los nuevos Pentium IV).
 - Bus de Memoria (MBUS)
 - Conectado al LBUS a través de transceptores y multiplexores (bus multiplexado para direccionar filas y columnas de las memorias). Soporta la conexión de la memoria RAM dinámica del sistema (DRAM).
 - Bus de Periféricos (XBUS)
 - Con un bus de datos de 8 bits. Todos los circuitos básicos del PC se conectan a él: PICs, DMAs, temporizadores, UART y P. Paralelo (al principio, luego se conectaron al SBUS a través de un Slot de Expansión), etc. En los AT, el bus de datos es de 16 bits y los dispositivos se conectan al Byte Bajo o al Byte Alto en función de si la dirección es Par o Impar respectivamente.

20.07.23

(12) Buses Internos (II)

- Bus del Sistema (SBUS)
 - separado del Bus Local mediante buffers (o latches) y transceptores.
 - Inicialmente, en los primeros PC/XT/AT tenía las mismas características de anchura y velocidad que el Bus Local.
 - Soluciona los problemas ocasionados por el aumento de la frecuencia de la CPU y del tamaño del bus de datos de 16 a 32 bits (386) (Compatibilidad).
 - A él se conectan el resto de los elementos que constituyen el PC.
 - En él se encuentran los Slots de Expansión (tarjeta gráfica, tarjeta de sonido, tarjetas de comunicaciones, controladoras de discos duros y disquetes, puertos serie y paralelo, etc.).
 - Se le conoce como Bus ISA (16 bits de datos y 24 de direcciones).

20.07.23

(12) Buses Internos (III)

- Las características del mismo son:
 - Velocidad en torno a los 8 MHz de los primeros PC-AT independientemente de la velocidad de la CPU, mediante la Máquina de Estados del Bus AT o Controlador de Bus ISA que genera Ciclos de Bus adecuados.
 - Se incluyeron multiplexores (como en el XBUS) y señales de control (BS16#) para adaptar el bus de datos de 16 bits a los nuevos buses de 32 bits (386) y de 64 bits (Pentium), convirtiendo las transferencias de datos de 32 y de 64 bits en varios ciclos de acceso de 16 bits. Se volvió a perder velocidad.
 - El bus MCA de IBM y el EISA de 32 bits no llegaron a asentarse comercialmente, aunque fueron intentos de mejora.

20.07.23

(12) **Bus ISA (Industry Standard Architecture) (I)**

GND	B1	A1	vO_CH_CK#	D1	C1	SBHE
RESET_DRV	B2	A2	SD7	D2	C2	LA23
+5 V	B3	A3	SD6	D3	C3	LA22
DRQ2	B4	A4	SD5	D4	C4	LA21
+5 V	B5	A5	SD4	D5	C5	LA20
MEMR#	B6	A6	SD3	D6	C6	LA19
-12 V	B7	A7	SD2	D7	C7	LA18
OVS	B8	A8	SD1	D8	C8	LA17
+12 V	B9	A9	SD0	D9	C9	MEMR#
GND	B10	A10	IO_CH_RDY	D10	C10	MEMW#
SMEMW#	B11	A11	AEN	D11	C11	SD9
SMEMR#	B12	A12	SA19	D12	C12	SD10
ROW#	B13	A13	SA18	D13	C13	SD11
IOR#	B14	A14	SA17	D14	C14	SD12
DACK3#	B15	A15	SA16	D15	C15	SD13
DRQ3#	B16	A16	SA15	D16	C16	SD14
DACK1#	B17	A17	SA14	D17	C17	SD15
DRQ1	B18	A18	SA13	D18	C18	SD15
REFRESH#	B19	A19	SA12			
CLK	B20	A20	SA11			
DRQ7	B21	A21	SA10			
RO6	B22	A22	SA9			
RO5	B23	A23	SA8			
RO4	B24	A24	SA7			
RO3	B25	A25	SA6			
DACK2#	B26	A26	SA5			
T/C	B27	A27	SA4			
BALE	B28	A28	SA3			
+5 V	B29	A29	SA2			
OSC	B30	A30	SA1			
GND	B31	A31	SA0			

MEM_CS16#
IO_CS16#
MEMR#
MEMW#
SBHE
LA23
LA22
LA21
LA20
LA19
LA18
LA17
MEMR#
MEMW#
SD9
SD10
SD11
SD12
SD13
SD14
SD15

Ampliación a 16 bits del AT

Conector original del PC/XT

(12) **Bus ISA (Industry Standard Architecture) (II)**

- SA0...SA19
 - Bus de direcciones para memoria y E/S (CPU o DMA)
- LA17...LA23
 - Completan el bus de direcciones del AT (capturarlas con latches)
- SD0...SD7
 - Bus de datos de 8 bits del PC/XT
- SD8...SD15
 - Ampliación del bus de datos a 16 bits (AT)
- BCLK, CLK
 - Señal de reloj del bus (unos 8 MHz)
- OSC
 - Señal de reloj de 14.31818 MHz

(12) **Bus ISA (Industry Standard Architecture) (III)**

- #IOR, #IOW
 - Órdenes de lectura y de escritura en el mapa de E/S
- #SMEMR, #SMEMW
 - Órdenes de lectura y de escritura en memoria si la dirección está por debajo de 1 MB
- #MEMR, #MEMW
 - Órdenes de lectura y de escritura en memoria para cualquier dirección (AT).
- DRQ1, DRQ2, DRQ3
 - Señales de petición de DMA en el PC/XT (3 canales) del controlador 8237.
- DRQ0, DRQ5, DRQ6, DRQ7
 - En el AT, el canal 0 para refresco de memoria y 3 canales más con un nuevo controlador DMA 8237.

(12) **Bus ISA (Industry Standard Architecture) (IV)**

- #REFRESH
 - Para refrescar la memoria DRAM. Sustituye al canal 0 del DMA que realizaba esa función en los primeros AT.
- DACK1, DACK2, DACK3
 - Señales de reconocimiento de petición de DMA en el PC/XT (3 canales).
- DACK0, DACK5, DACK6, DACK7
 - En el AT, reconocimiento de DMA de los canales añadidos.
- AEN
 - Validación de la dirección
- #SBHE
 - Indica una transferencia por el byte alto del Bus de Datos.
- #MASTER
 - Entrada para que un dispositivo exterior solicite el control de los buses.

(12) **Bus ISA (Industry Standard Architecture) (V)**

- #MEM_CS16
 - Entrada a través de la cual un dispositivo exterior indica que va a realizar una transferencia con la memoria de 16 bits
- #IO_CS16
 - Entrada a través de la cual un dispositivo exterior indica que va a realizar una transferencia con un disp. De E/S de 16 bits.
- OWS
 - Indica al sistema que un ciclo de acceso al bus debe realizarse con cero estados de espera (accesos de 16 bits) o con un mínimo de 2 estados de espera (ciclos de 8 bits).
- T/#C
 - Señal #EOP del contador de DMA que cuando se pone a 1 indica que el contador ha finalizado la cuenta. Común para todos los canales.

(12) **Bus ISA (Industry Standard Architecture) (VI)**

- RESET_DRV
 - Señal Reset del sistema que sirve para inicializar los circuitos de las tarjetas.
- IO_CH_RDY
 - Se usa para insertar estados de espera en los ciclos de bus desde el exterior. A 1 indica ejecución normal, sin ciclos de espera.
- #IO_CH_CK
 - Señal para indicar a la placa base que se ha producido un error de paridad o algún error irrecuperable, provocando cuando se activa a 0 una interrupción NMI.
- BALE
 - Su flanco de bajada sirve para cargar (latch) direcciones presentes en el bus. SA0 a SA19 son cargadas en latches internos de la placa base pero LA17 a LA23 deben ser cargadas en latches externos.

(12) **Bus ISA (Industry Standard Architecture) (VII)**

- El **Controlador de Bus ISA** es una máquina de estados que se coordina con la CPU y que hace que el Bus ISA se comporte como el bus (Local) de un 286 a 8 MHz.
- Están previstos **4 tipos de ciclos**, dependiendo de un acceso a memoria o a E/S y de accesos de 8 o 16 bits. Además, se puede distinguir entre **Ciclo Normal** y **Ciclo OWS**. Estos ciclos no son el Ciclo de Bus de la CPU.
- Tiene verdaderas limitaciones cuando se trata de transferir grandes volúmenes de datos.

Tipo de acceso	Tamaño del dato	Ciclo normal		Ciclo OWS	
		Ciclos	Velocidad	Ciclos	Velocidad
Memoria (R/W)	16 bits	3 (2+1w)	5.33 MB/s	2(2+0w)	8.0 MB/s
	8 bits	(2+5w)	1.33 MB/s	3(2+1w)	2.66 MB/s
E/S (R/W)	16 bits	3(2+1w)	5.33 MB/s	3(2+1w)	5.33 MB/s
	8 bits	(2+5w)	1.33 MB/s	3(2+1w)	2.66 MB/s

Los estados de espera se insertan automáticamente ya que así se ha diseñado, y aunque pueden anularse en algunos casos, hay otros en los que no y reducen la velocidad de transferencia.

(12) **Bus VL (VESA Local Bus)**

- VESA (Video Equipment Standard Association) es un consorcio de fabricantes que deciden paliar las deficiencias de velocidad de transferencia del bus ISA para las nuevas tarjetas de vídeo que estaban apareciendo en el mercado. Aparecen en los 386 y 486.
- El bus ISA con su velocidad de transferencia de 2.5 Mbytes/s a 8 bits no podía soportar las necesidades de las nuevas controladoras VGA.
- Se añadió un conector suplementario (tipo Mico-Channel) alineado con algún conector ISA, que establecía conexiones directas con los buses de la CPU, sin buffers ni circuitería adicional ya que sólo se necesitaba conectar una tarjeta de vídeo.
- Su velocidad podía alcanzar 66 MBps (33 MHz del Bus Local a 32 bits, es decir 2 bytes).
- Su gran ventaja fue lo barato de su implementación y sus desventajas, el cuidado que hay que tener al conectar directamente a la CPU y la dependencia con la misma (33 MHz).

(12) **Bus PCI (I) (Peripheral Component Interconnect)**

- Diseño original de Intel (1.992), actualmente cedido a un consorcio llamado SIG (Special Interest Group) (24 o más fabricantes importantes).
- Intenta paliar los inconvenientes del Bus ISA ante el fracaso del Bus MCA y del EISA.
- Al igual que existía un circuito adaptador del Bus ISA (Máquina de Estados del Bus AT), ahora existe un conjunto o juego de circuitos de Intel (i82420) que permite incorporar el bus PCI a un sistema ISA:
- CDC (i82424TX) Cache/Dram Controller. Hace de puente entre CPU y Bus PCI, incluye controladores de DRAM y de caché de segundo nivel.
- DPU (i82423TX) Data Path Unit. Puente de datos entre CPU y PCI.
- SIO (i82378IB) System I/O. Puente entre los buses PCI e ISA. Incorpora los elementos clásicos de ISA: PICs, DMAs, Timers, etc.

(12) **Bus PCI (II) (Peripheral Component Interconnect)**

- Características del Bus PCI:
- Independencia de la CPU
- Frecuencia de funcionamiento: 33 MHz
- Ancho del bus: 32 o 64 bits
- Velocidad de transferencia: 132 MBps a 32 bits y 264 MBps a 64 bits
- Alimentación: Placa base a 3.3 v o 5 v, tarjetas a 3.3 v, 5 v. o ambas.
- Conector: tipo Micro-Channel de 32 y 64 bits.
- Reloj: cualquier frecuencia entre 0 y 33 MHz
- Bus: Direcciones y datos multiplexados en 32 terminales. Posible expansión a 64 bits.
- Direccionamiento: soporta hasta 255 dispositivos por cada bus PCI.

(12) **Bus PCI (III) (Peripheral Component Interconnect)**

- Más características del Bus PCI:
- Transferencias
 - Siempre en modo burst, de longitud negociada entre Maestro y Esclavo.
 - Comprobación de integridad mediante verificación de paridad en direcciones, datos y comandos.
- Interrupciones
 - Cuatro entradas de petición de interrupción: #INTA, #INTB, #INTC, #INTD que son sensibles a nivel y pueden ser compartidas.
- Arbitración de bus
 - Centralizada.
 - Las tarjetas capaces de ser Master (Initiator) manejan las señales #REQ y #GT.
- Autoconfiguración
 - La norma especifica con detalle el registro de configuración de cada dispositivo PCI.
- Señales de error
 - #PERR que indica error de paridad en los datos
 - #SERR que indica error de paridad de dirección, dato especial y errores catastróficos.

