

# Clase 16 - Entradas y Salidas: Comunicación

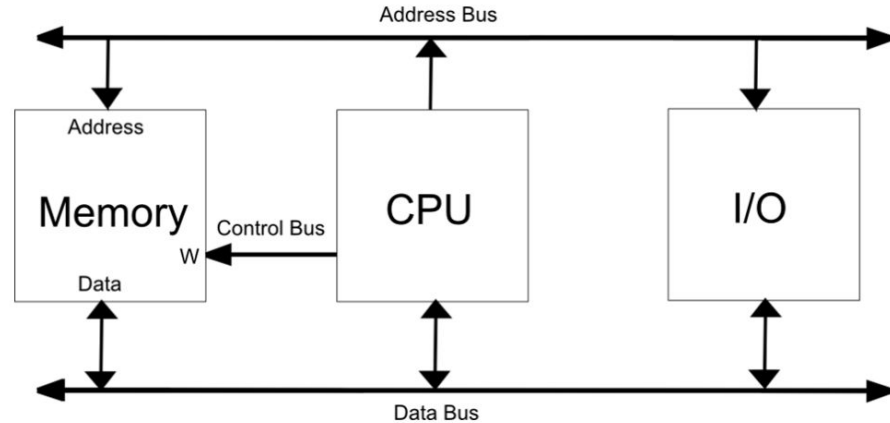
---

Profesor: **IIC2343 - Arquitectura de Computadores**  
- Felipe Valenzuela González  
Correo:  
frvalenzuela@alumni.uc.cl

# **Resumen de clase pasada**

# Arquitectura de Computadores: Dispositivos I/O

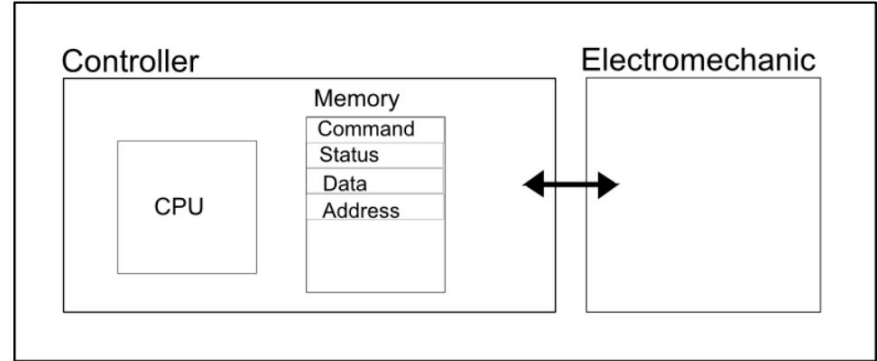
- Todos los dispositivos que no sean CPU o memoria, y se comuniquen con ellos, son llamados dispositivos de I/O



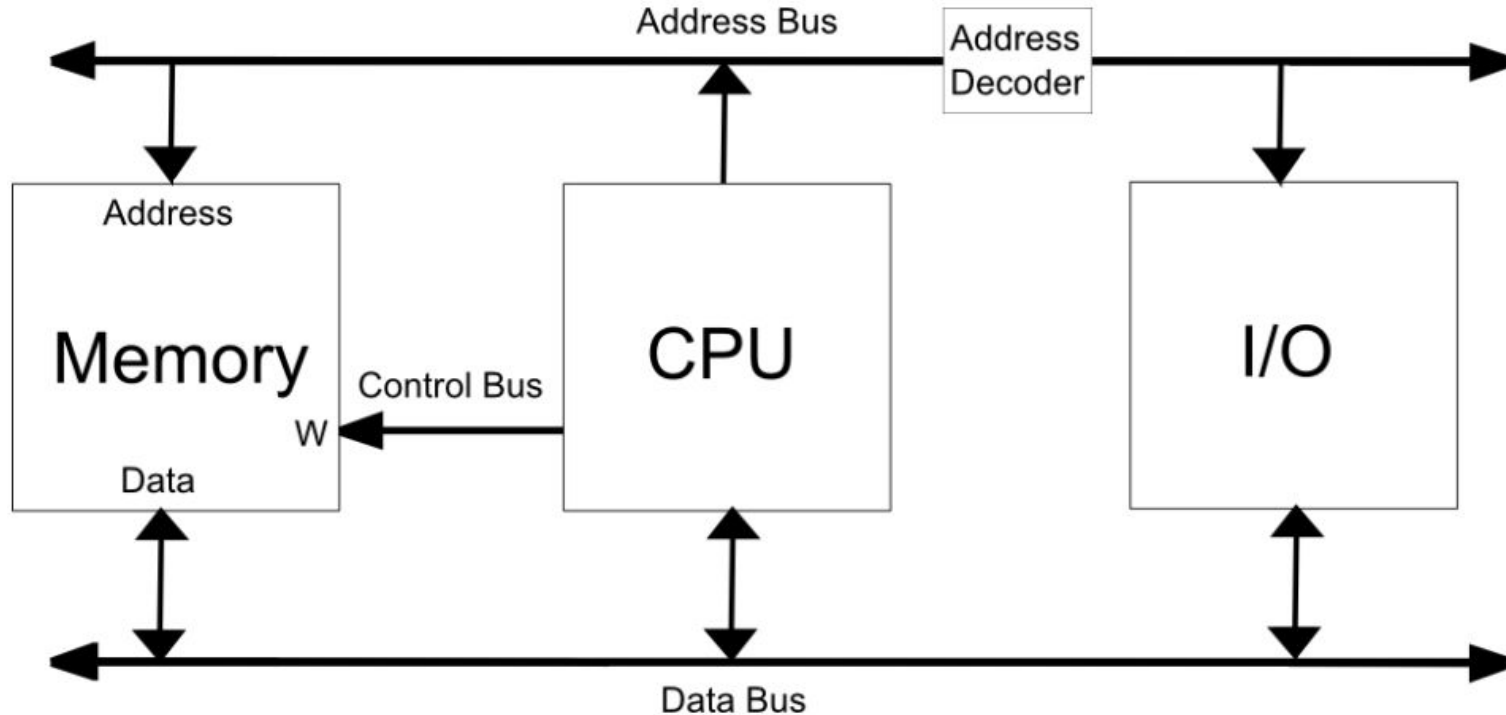
# Arquitectura de Computadores: Dispositivos I/O

- Un dispositivo de I/O tiene un controlador encargado de comunicarse con la CPU y la memoria, y de controlar la parte electromecánica. Todos los dispositivos que no sean CPU o memoria, y se comuniquen con ellos, son llamados dispositivos de I/O

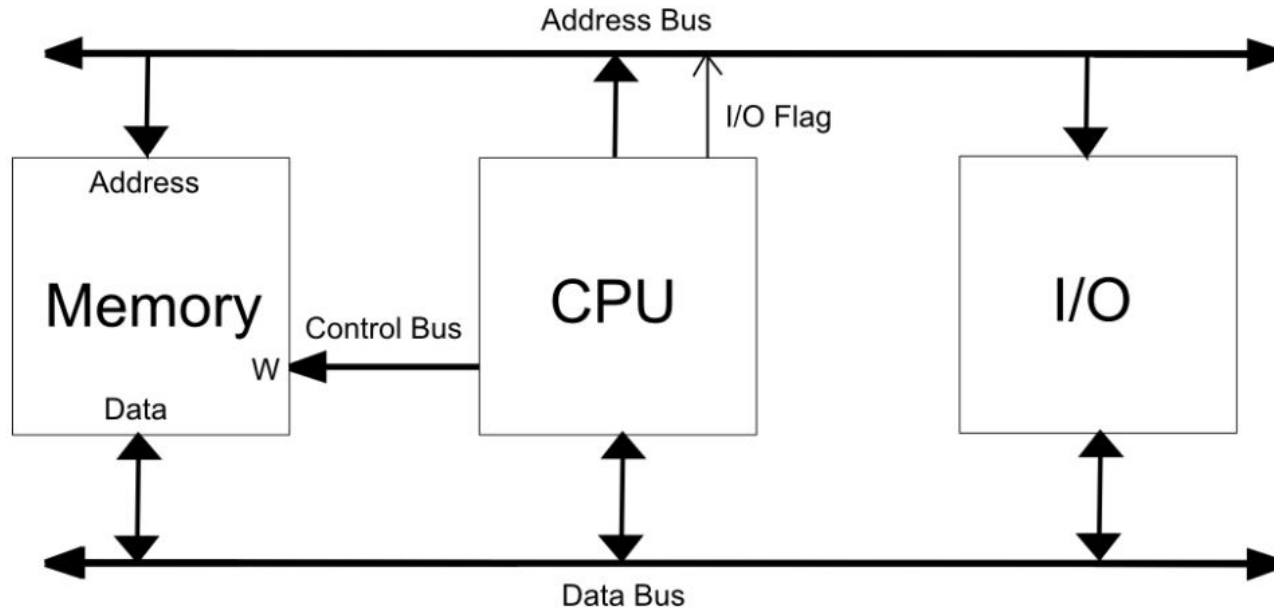
I/O



# Arquitectura de Computadores: Memory Map



# Arquitectura de Computadores: Port I/O



**¿Dudas?**

# Arquitectura de Computadores: Dispositivos I/O

Usaremos como analogía a un curso haciendo una guía de ejercicios:

- Alumnos (I/O) hacen guía de ejercicios, si alguien termina, el profesor quiere guardar la respuesta.
- Profesor (CPU) está corrigiendo pruebas.
- Pizarra o proyector (Memoria), donde se pueden ver los ejercicios y anotar las respuestas.
- Comandos: instrucciones de parte del profesor.
- Estado: alumnos trabajando, con dudas.
- Datos: preguntas y respuestas guía, dudas alumnos, respuestas profesor.
- Supuestos: alumnos no hablan entre ellos, se entregan varias guías durante la clase



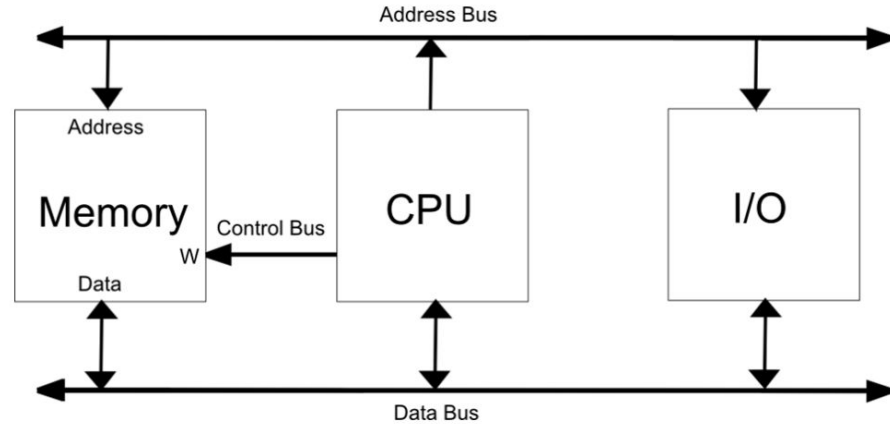
# Revisaremos tres modelos de interacción entre alumnos y profesor

Sin proyector ni pizarrón ni copias de las guías, alumnos tímidos:

## 1. Polling

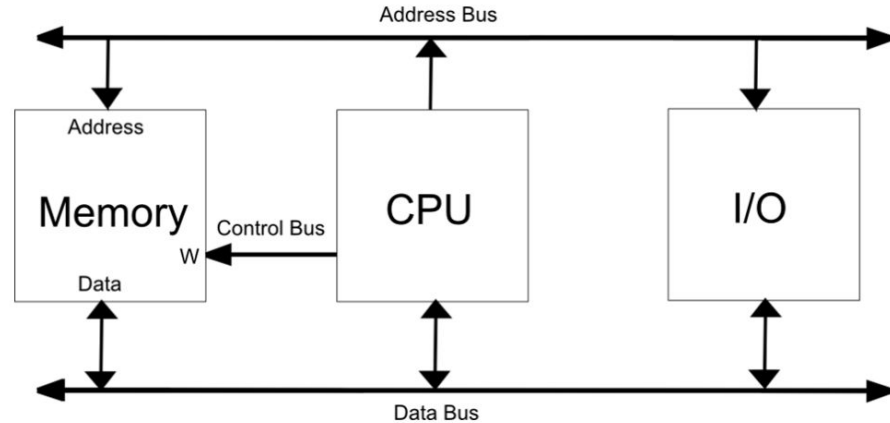
# Arquitectura de Computadores: Polling

- ¿Necesitamos cambio de hardware?



# Arquitectura de Computadores: Polling

- ¿Necesitamos cambio de hardware?
- **NO**



# Revisaremos tres modelos de interacción entre alumnos y profesor

Sin proyector ni pizarrón ni copias de las guías, alumnos tímidos:

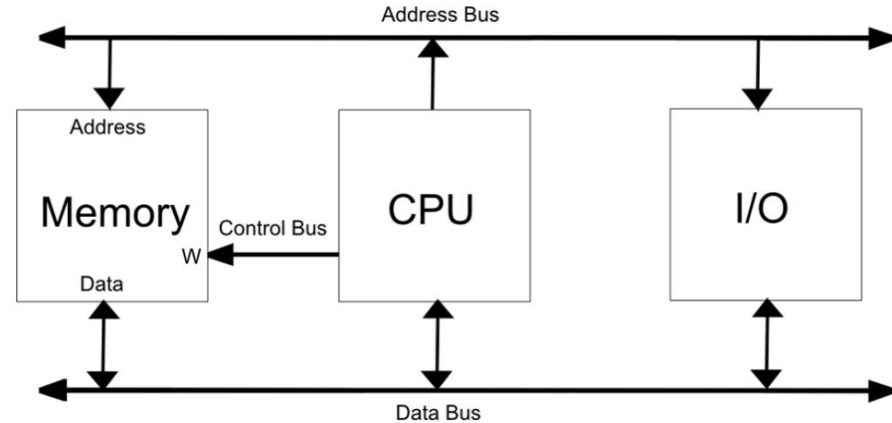
## 1. Polling

Sin proyector ni pizarrón ni copias de las guías, alumnos  
preguntones y participativos:

## 2. Interrupción

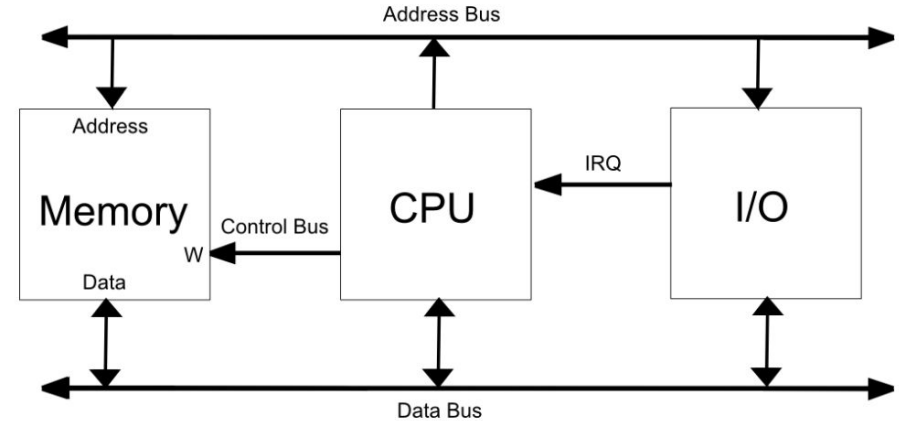
# Arquitectura de Computadores: Interrupción

- ¿Necesitamos cambio de hardware?



# Arquitectura de Computadores: Interrupción

- ¿Necesitamos cambio de hardware?
- SI, una señal de **solicitud de interrupción**



# Arquitectura de Computadores: Interrupción

1. Dispositivo solicita interrumpir, enviando señal IRQ
2. CPU termina de ejecutar la instrucción actual y guarda condition codes en el stack
3. CPU revisa si el flag de interrupciones está activo ( $IF = 1$ ). En caso contrario, saltar al paso 11.
4. CPU deshabilita atención de más interrupciones ( $IF=0$ )
5. CPU llama a la ISR asociada al dispositivo

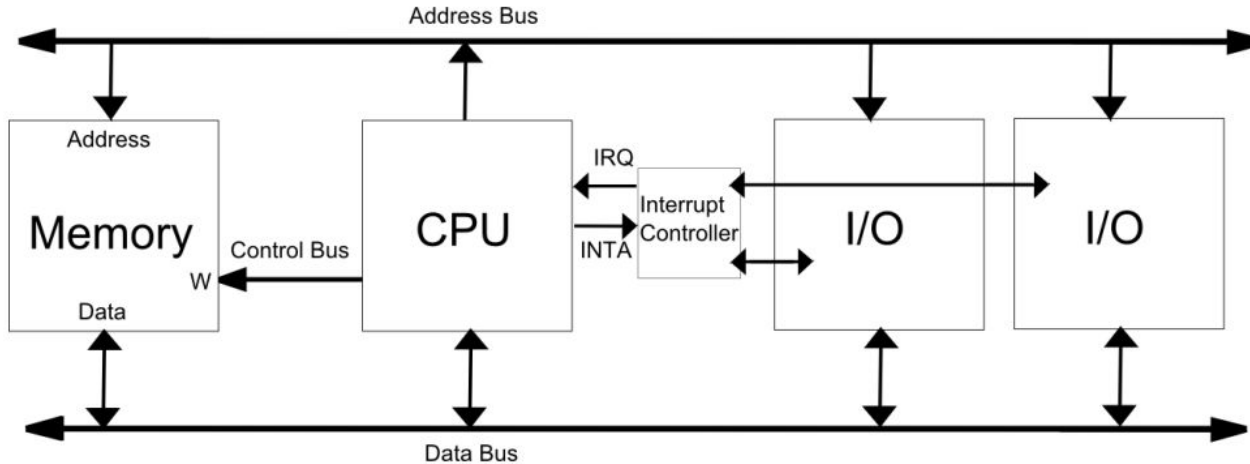
## Arquitectura de Computadores: Interrupción

6. ISR respalda estado actual de la CPU
7. ISR es ejecutada
8. ISR restaura estado de la CPU
9. ISR retorna
10. CPU habilita la atención de interrupciones ( $IF=1$ )
11. CPU recupera condition codes desde el stack



# Arquitectura de Computadores: Interrupción

- Al usar varios dispositivos, es necesario incorporar un controlador de interrupciones y una tabla de vectores de interrupción



# Arquitectura de Computadores: Interrupción

- En general, un controlador de interrupciones tendrá al menos los siguientes componentes:
- Registro de comandos y estado
- ***Interrupt Request Register***: Registro de interrupciones en espera de atención
- ***In-Service Register***: Registro de interrupciones en atención
- ***Interrupt Mask Register***: Registro de enmascaramiento de interrupciones
- Circuito para manejar prioridades de interrupciones

# Tipos de interrupción en la CPU

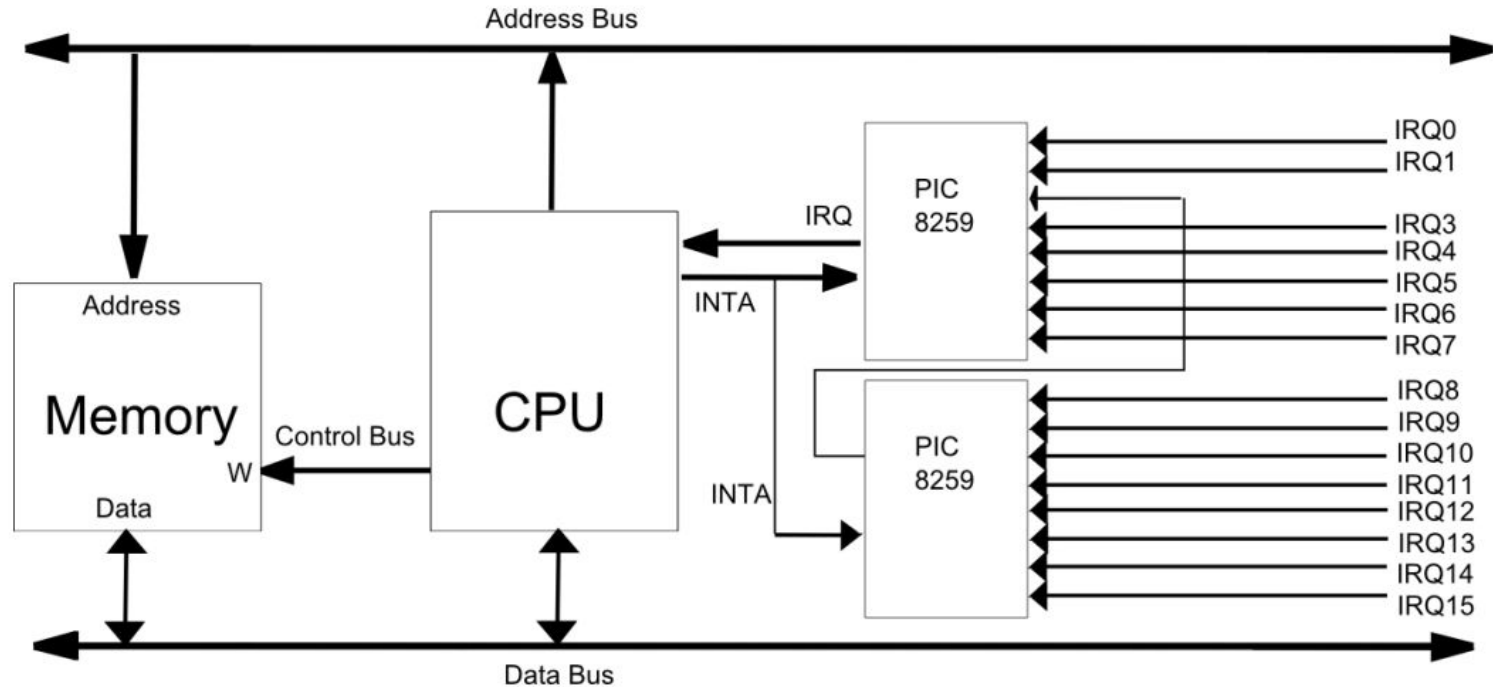
- **Excepciones:** Condiciones de error al ejecutar una instrucción. Estas son atendidas por la CPU (en particular, el sistema operativo) a través de **ISRs** específicas: *exception handler*

**Ejemplos:** división por cero, stack overflow.

- **Interrupciones de software (*traps*):** Son “llamadas al sistema operativo”, interrupciones explícitas en los programas para ceder el control al sistema operativo para que realice una acción

**Ejemplo:** `ecall`

## Arquitectura de Computadores: Interrupción Ejemplo



# Arquitectura de Computadores: Interrupción Ejemplo

IRQ	Dispositivo	Vector de interrupción
IRQ0	Timer del sistema	08
IRQ1	Puerto PS/2: Teclado	09
IRQ2	Conectada al PIC secundario	0A
IRQ3	Puerto serial	0B
IRQ4	Puerto serial	0C
IRQ5	Puerto paralelo	0D
IRQ6	Floppy disk	0E
IRQ7	Puerto paralelo	0F
IRQ8	Real time clock (RTC)	70
IRQ9-11	No tienen asociación estándar, libre uso.	71-73
IRQ12	Puerto PS/2: Mouse	74
IRQ13	Coprocador matemático	75
IRQ14	Controlador de disco 1	76
IRQ15	Controlador de disco 2	77

# Arquitectura de Computadores: Interrupción Ejemplo

Dirección del vector	Tipo	Función asociada
00-01	Excepción	Exception handlers
02	Excepción	Usada para errores críticos del sistema, no enmascarable
03-07	Excepción	
08	IRQ0	Timer del sistema
09	IRQ1	Puerto PS/2: Teclado
0A	IRQ2	Conectada al PIC secundario
0B	IRQ3	Puerto serial
0C	IRQ4	Puerto serial
0D	IRQ5	Puerto paralelo
0E	IRQ6	Floppy disk
0F	IRQ7	Puerto paralelo
10	Int. de Software	Funciones de video
11-6F	Int. de Software	Funciones varias
70	IRQ8	Real time clock (RTC)
71 - 73	IRQ9-11	No tienen asociación estándar, libre uso
74	IRQ12	Puerto PS/2: Mouse
75	IRQ13	Coprocesador matemático
76	IRQ14	Controlador de disco 1
77	IRQ15	Controlador de disco 2

**¿Dudas?**

# Revisaremos tres modelos de interacción entre alumnos y profesor

Sin proyector ni pizarrón ni copias de las guías, alumnos tímidos:

## 1. Polling

Sin proyector ni pizarrón ni copias de las guías, alumnos  
preguntones y participativos:

## 2. Interrupción

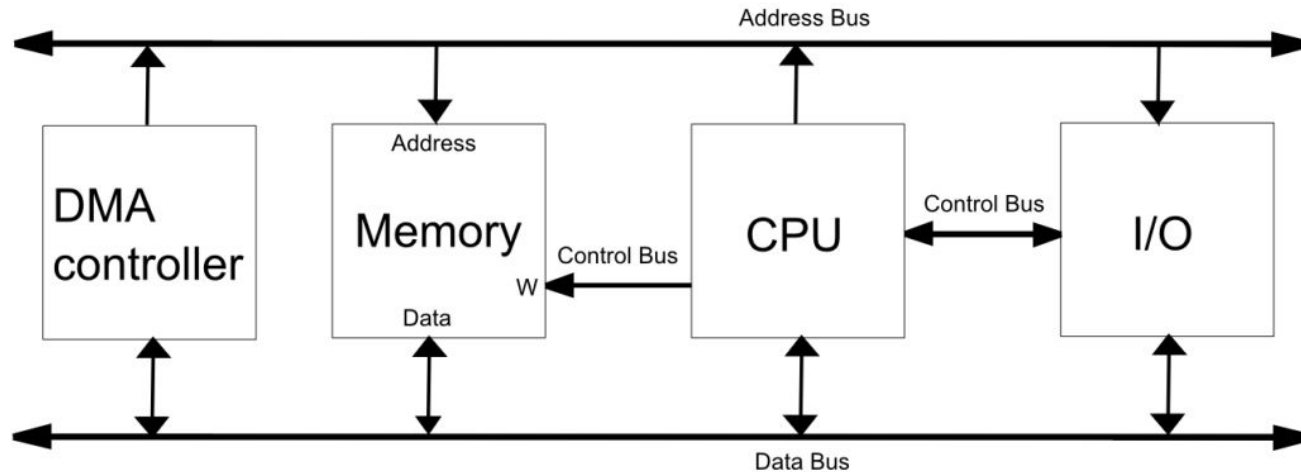
Pizarra interactiva, alumnos preguntones:

## 3. Interrupción con acceso directo a memoria



# Arquitectura de Computadores: DMA

- Para permitir que los I/O tengan acceso directo a memoria se utiliza un controlador DMA

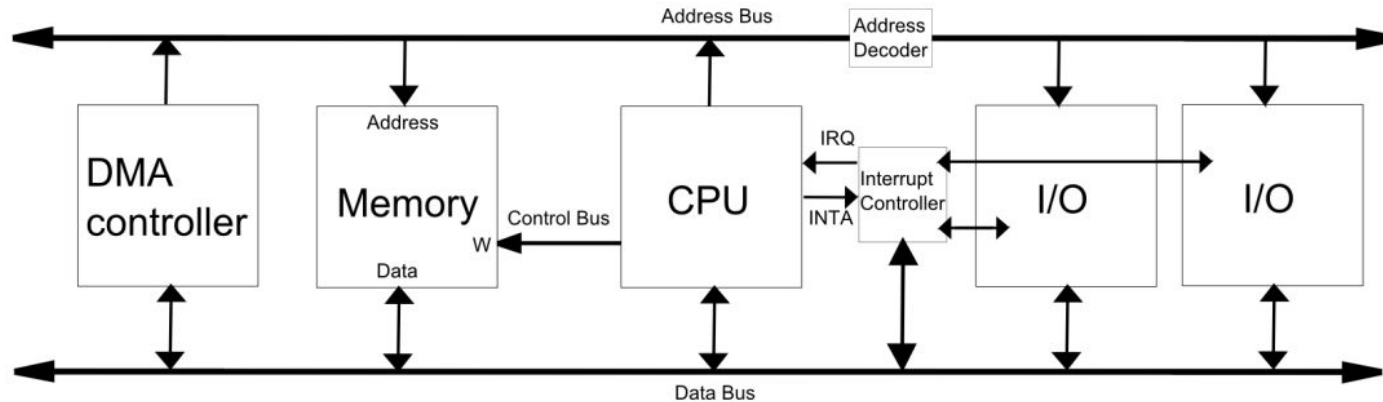


# Arquitectura de Computadores: DMA

- En general, un controlador de DMA tendrá al menos los siguientes componentes:
  - Registro de comandos y estado
  - Registros de dirección de origen y destino
  - Registro contador de palabras
  - Buffer de almacenamiento temporal
  - Unidad de control

# Arquitectura de Computadores: DMA e Interrupciones

- Arquitectura de un computador con memory mapped I/O, interrupciones y DMA



**¿Dudas?**

# Bibliografía

- Apuntes históricos. Hans Löbel, Alejandro Echeverría

## 8 - Comunicación de CPU y Memoria con IO

- D. Patterson, Computer Organization and Design RISC-V. Edition: The Hardware Software Interface. Morgan Kaufmann, 2020. Capítulo 6.9. Página 529.e1, 690 en PDF

⋮	🔗 <a href="#">Computer Organization and Design RISC-V edition.pdf</a>	✓	⋮
⋮	🔗 <a href="#">Tanenbaum, Andrew Stuart Austin, Todd Chandavarkar, B R Structured.pdf</a>	✓	⋮
⋮	Apuntes históricos	✓	⋮
⋮	🔗 <a href="#">01 - Representaciones Numéricas Parte 1 - Números Enteros.pdf</a>	✓	⋮
⋮	🔗 <a href="#">02 - Representaciones Numéricas Parte 2 - Números Racionales.pdf</a>	✓	⋮
⋮	🔗 <a href="#">03 - Operaciones Aritméticas y Lógicas.pdf</a>	✓	⋮
⋮	🔗 <a href="#">04 - Almacenamiento de datos.pdf</a>	✓	⋮
⋮	🔗 <a href="#">05 - Programabilidad.pdf</a>	✓	⋮
⋮	🔗 <a href="#">06 - Saltos y Subrutinas.pdf</a>	✓	⋮
⋮	🔗 <a href="#">07 - Arquitecturas de Computadores.pdf</a>	✓	⋮
⋮	🔗 <a href="#">08 - Comunicacion de CPU y Memoria con IO.pdf</a>	✓	⋮

# Bibliografía

- <https://github.com/IIC2343/Syllabus-antiores/tree/main/Enunciados>

The screenshot shows the GitHub interface for the repository `IIC2343 / Syllabus-antiores`. The left sidebar displays the file tree with the `Enunciados` directory selected. The main area shows the commit history for `Syllabus-antiores / Enunciados`, listing 15 commits by user `Geermy`. The commits include updates to previous exams, assessments, and the deletion of a specific file.

Name	Last commit message	Last commit date
...		
Enunciados 2019-1	feat: add more solutions to previous exams; add previous auxiliary cl...	2 years ago
Enunciados 2019-2	feat: add more solutions to previous exams; add previous auxiliary cl...	2 years ago
Enunciados 2020-1	feat: add more solutions to previous exams; add previous auxiliary cl...	2 years ago
Enunciados 2021-1	add enunciados antiguos	2 years ago
Enunciados 2021-2	feat: add assessments for past semester	2 years ago
Enunciados 2022-1	feat: add assessments for past semester	2 years ago
Enunciados 2022-2	feat: add more solutions to previous exams; add previous auxiliary cl...	2 years ago
Enunciados 2023-1	feat: add assessments for past semester	2 years ago
Enunciados 2023-2	add missing test solutions for last semester	last year
Enunciados 2024-1	remove unused readme	10 months ago
Enunciados 2024-2	Delete Enunciados/Enunciados 2024-2/a	4 months ago
Compilado_IIC2343.pdf	Agregando compilado de Felipe	5 years ago

# Clase 16 - Entradas y Salidas: Comunicación

---

Profesor: **IIC2343 - Arquitectura de Computadores**  
- Felipe Valenzuela González  
Correo:  
frvalenzuela@alumni.uc.cl