



# ARQUITECTURA DE COMPUTADORAS.

ESQUEMA ORDENADOR.

2C – 2024.

FCyT – UADER – Oro Verde.

# OBJETIVOS

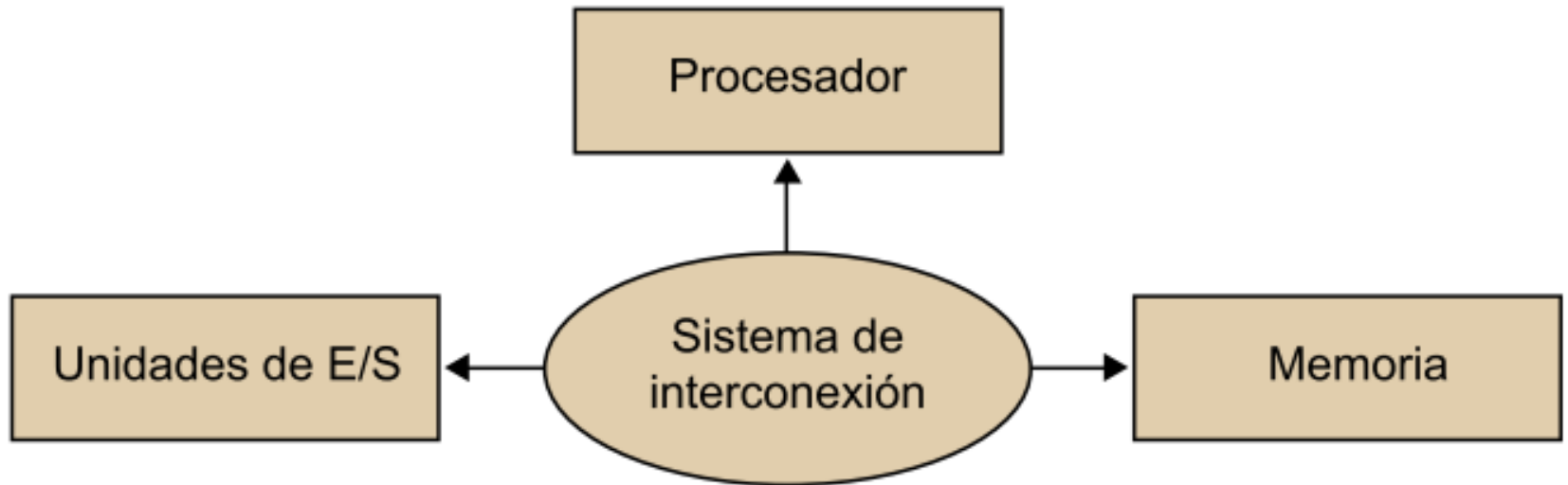
- Entender el concepto de computadora.
- Conocer los dos tipos de organizaciones de una computadora más habituales: Von Neumann y Harvard.
- Comprender el concepto de lo que es y hace un Esquema Ordenador.
- Identificar los elementos componentes del Esquema Ordenador.
- Conocer el uso y la importancia de los buses en el Esquema.
- Conocer la actividad que desarrolla el Esquema Ordenador
- Comenzar a manejar el vocabulario técnico asociado a la computación y el hardware específicamente.

# LA COMPUTADORA – REPASO.

Una computadora se puede definir como una máquina electrónica capaz de hacer las tareas siguientes:

- Aceptar información.
- Almacenarla.
- Procesarla según un conjunto de instrucciones.
- Producir y proporcionar unos resultados.

# LA COMPUTADORA – REPASO.



# CONCEPTOS BÁSICOS.

## Organización y arquitectura.

Arquitectura de computadoras: la arquitectura de computadoras se refiere al diseño conceptual y lógico de un sistema informático. Esto incluye las decisiones de alto nivel sobre la estructura y el funcionamiento general del sistema, como la organización de la memoria, el conjunto de instrucciones, las unidades de procesamiento y cómo se comunican entre sí. La arquitectura define las características clave del sistema y cómo se relacionan sus componentes para lograr un sistema informático.

# CONCEPTOS BÁSICOS.


## Organización y arquitectura.

Organización de Computadoras: La organización de computadoras se centra en cómo se implementa físicamente el diseño arquitectónico. Esto incluye aspectos como la interconexión de los componentes, el diseño de la unidad de control, la forma en que se accede y gestiona la memoria, y cómo se ejecutan las instrucciones en la unidad de procesamiento central (CPU). La organización se ocupa de la implementación concreta de los principios arquitectónicos en hardware.

# CONCEPTOS BÁSICOS.

## Organización y arquitectura.

En resumen, la arquitectura se centra en el diseño conceptual y lógico de un sistema informático, mientras que la organización se centra en cómo se implementa físicamente ese diseño en términos de hardware y circuitos.



Teniendo en cuenta esta diferencia, podemos tener computadores con una organización diferente, pero que comparten la misma arquitectura.

Por ejemplo, los microprocesadores Intel64 tienen una organización diferente de los microprocesadores AMD64, sin embargo, comparten una misma arquitectura (excepto ciertas diferencias), la arquitectura que se denomina x86-64.



# TIPOS DE ARQUITECTURAS

- La arquitectura Von Neumann.
- La arquitectura Harvard.

# TIPOS DE ARQUITECTURAS

Se puede decir que la mayoría de los computadores actuales utilizan la arquitectura Von Neumann, o una arquitectura Von Neumann modificada, ya que a medida que las computadoras han evolucionado se le ha añadido a esta, características procedentes de la arquitectura Harvard.

# ARQUITECTURA VON NEUMANN

## ARQUITECTURA VON NEUMANN



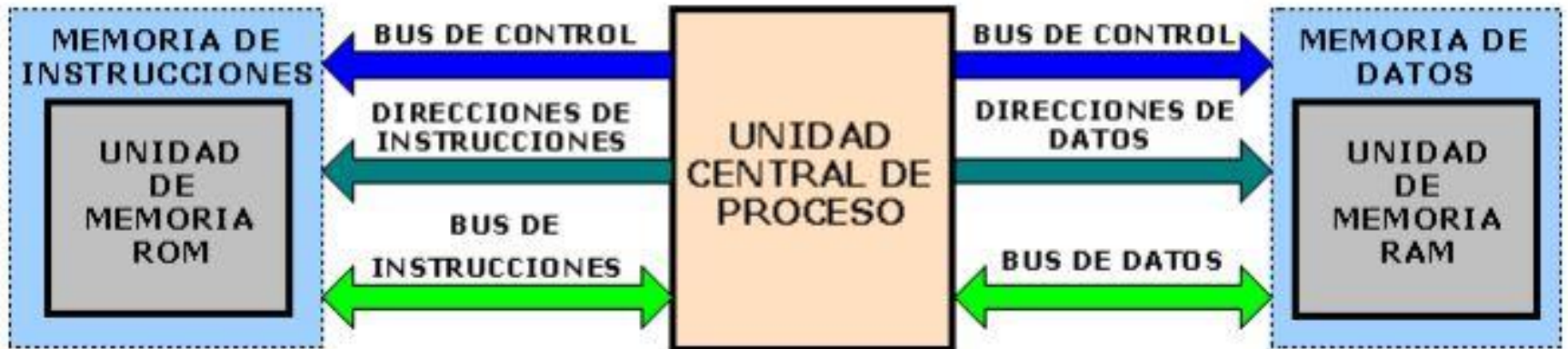
# ARQUITECTURA VON NEUMANN

La arquitectura Von Neumann se basa en tres propiedades:

- Hay un único espacio de memoria de lectura y escritura, que contiene las instrucciones y los datos necesarios.
- El contenido de la memoria es accesible por posición, Independientemente de que se acceda a datos o a instrucciones.
- La ejecución de las instrucciones se produce de manera secuencial.

# ARQUITECTURA HARVARD

## ARQUITECTURA HARVARD



# ARQUITECTURA HARVARD

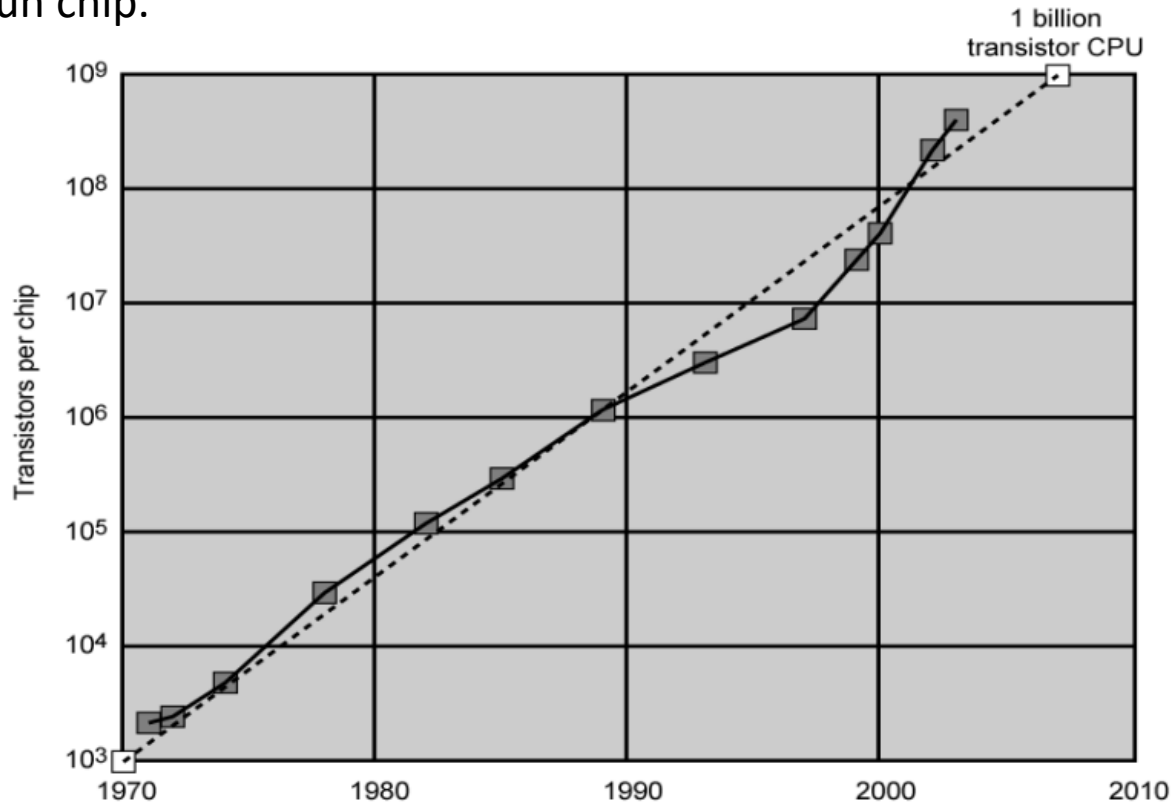
La organización del computador según el modelo Harvard, básicamente, se distingue del modelo Von Neumann por la división de la memoria en una memoria de instrucciones y una memoria de datos, de manera que el procesador puede acceder separada y simultáneamente a las dos memorias.

# ARQUITECTURA HARVARD

La arquitectura Harvard no se utiliza habitualmente en computadoras de propósito general, sino que se utiliza en computadoras para aplicaciones específicas.

# LEY DE MOORE.

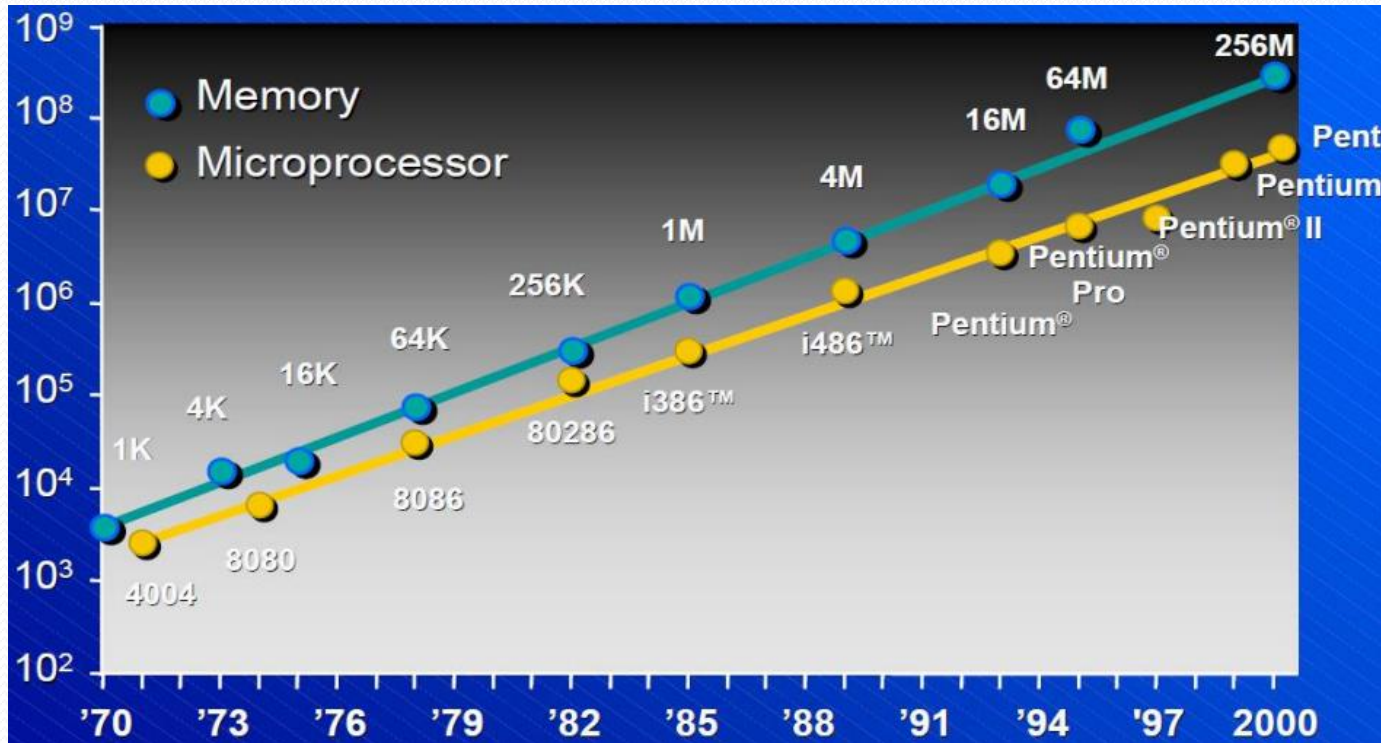
La Ley de Moore es una observación empírica que fue formulada por Gordon Moore, co-fundador de Intel, en 1965. Esta ley establece que el número de transistores en un circuito integrado (como un microprocesador) tiende a duplicarse aproximadamente cada dos años, lo que conduce a un aumento exponencial en la densidad de transistores en un chip.





# LEY DE MOORE.

La Ley de Moore es una observación empírica que fue formulada por Gordon Moore, co-fundador de Intel, en 1965. Esta ley establece que el número de transistores en un circuito integrado (como un microprocesador) tiende a duplicarse aproximadamente cada dos años, lo que conduce a un aumento exponencial en la densidad de transistores en un chip.



# LEY DE MOORE.

Algunos puntos clave sobre la Ley de Moore son:

- Crecimiento Exponencial: La Ley de Moore implica un crecimiento exponencial en la cantidad de transistores en los circuitos integrados con el tiempo.
- Mayor Potencia de Procesamiento: El aumento en la densidad de transistores permite la creación de microprocesadores más potentes y eficientes, lo que se traduce en un mejor rendimiento y capacidad de procesamiento.
- Reducción del Tamaño: A medida que el número de transistores en un chip aumenta, los transistores individuales se vuelven más pequeños. Esto ha llevado a la miniaturización de los dispositivos electrónicos.
- Disminución del Costo por Transistor: La Ley de Moore ha resultado en una disminución constante en el costo por transistor, lo que ha permitido que las tecnologías electrónicas sean más asequibles.

# AUMENTO EN EL GRADO DE INTEGRACIÓN.

La evolución del tamaño de los transistores en los microprocesadores ha seguido una tendencia de miniaturización constante a lo largo de los años, conocida como Ley de Moore.

**Intel 4004 (1971):** Los transistores en el Intel 4004 tenían un tamaño de aproximadamente **10 micrómetros ( $\mu\text{m}$ )**.

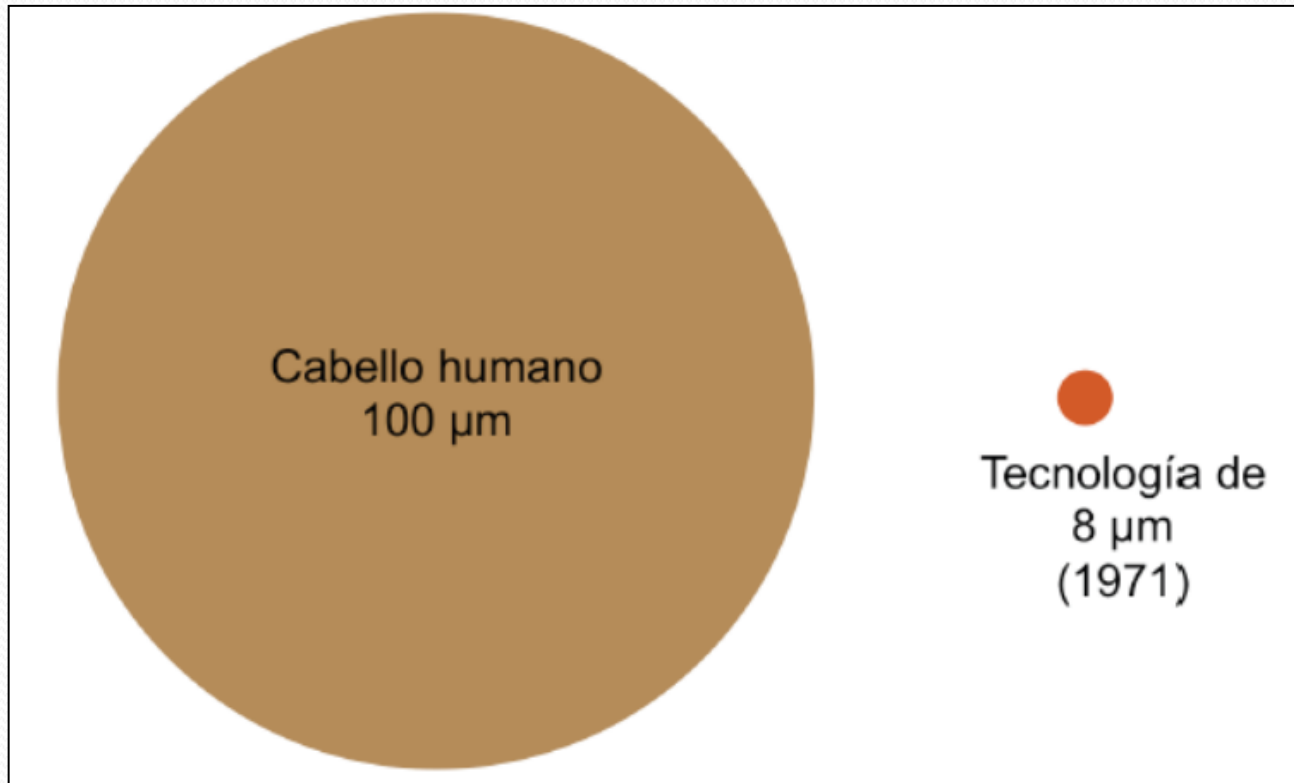
**Años 80 y 90:** La miniaturización de los transistores continuó, con tamaños que se redujeron a alrededor de **1 micrómetro ( $\mu\text{m}$ )** en algunos casos a finales de los años 80 y principios de los 90.

**Años 2000:** Los tamaños de los transistores se redujeron aún más, alcanzando aproximadamente **0.18 micrómetros (180 nm)** en procesadores como el Intel Pentium 4 y el AMD Athlon XP.

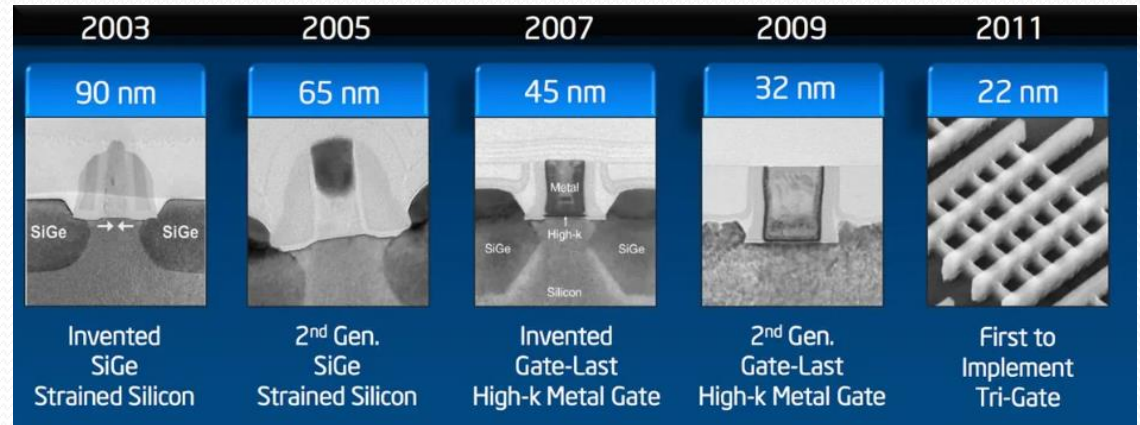
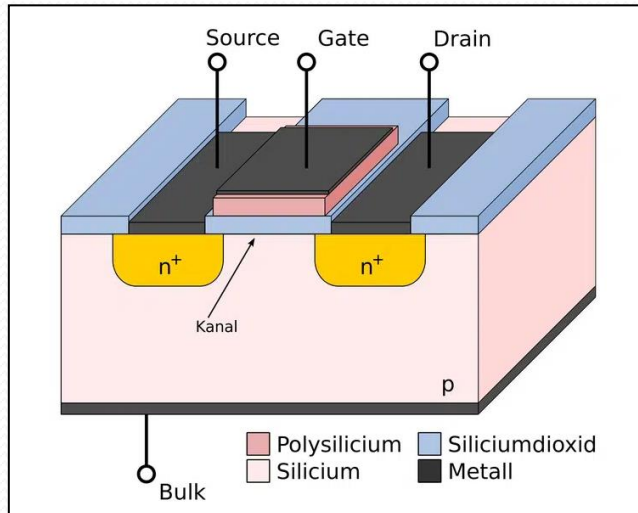
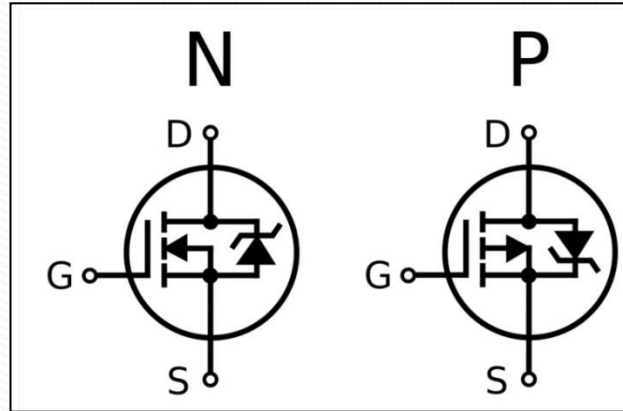
**Década de 2010:** La tecnología de proceso avanzó hacia los 45 nanómetros (0.045 micrómetros) y luego a los **32 nm**, permitiendo una mayor densidad de transistores y un rendimiento mejorado en procesadores como los Intel Core de segunda y tercera generación.

**Década de 2020:** La industria continuó avanzando hacia tamaños más pequeños, llegando a los **7 nm** en procesadores como el AMD Ryzen 5000 y el Apple M1.

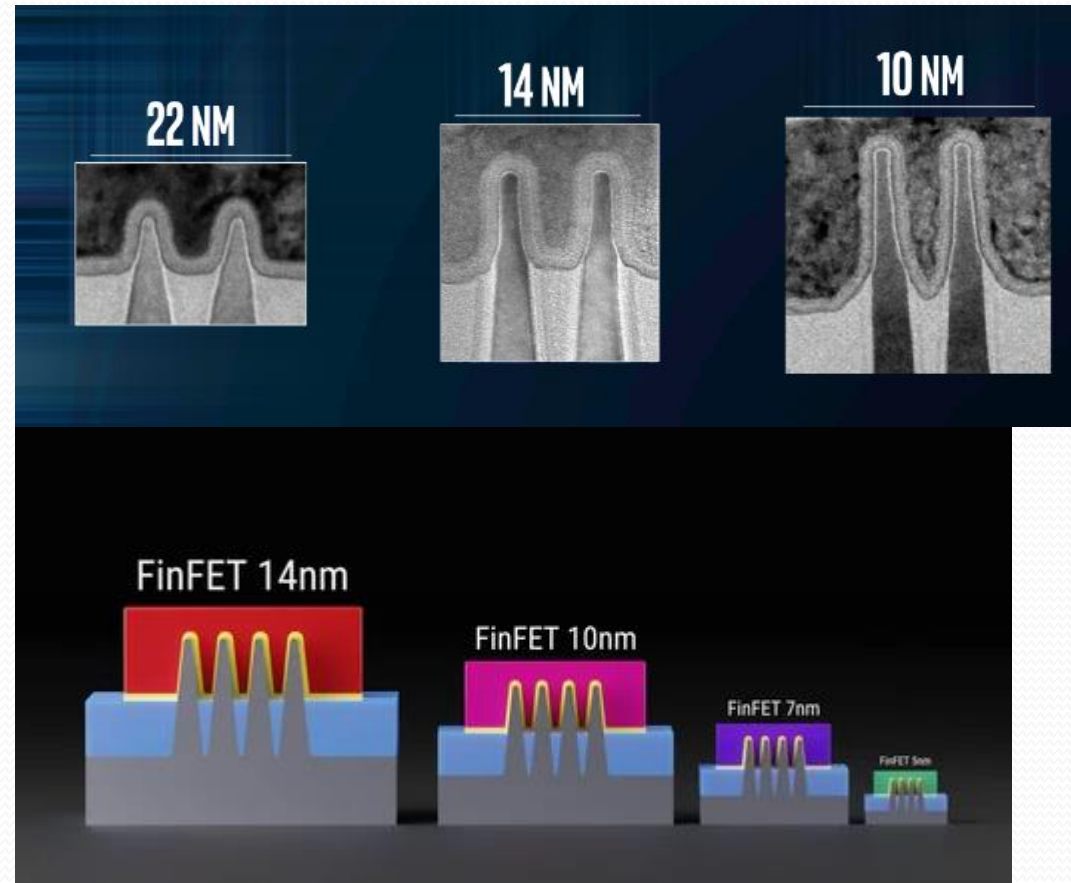
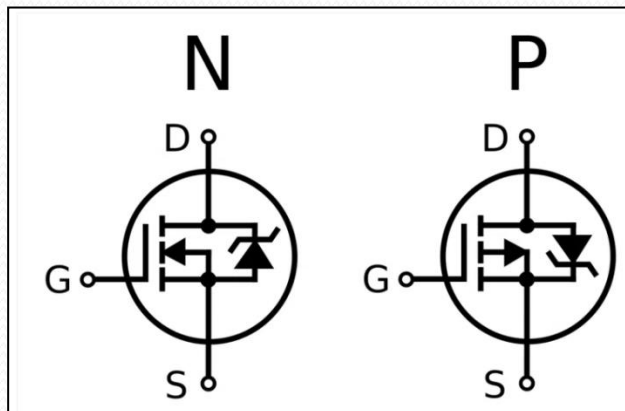
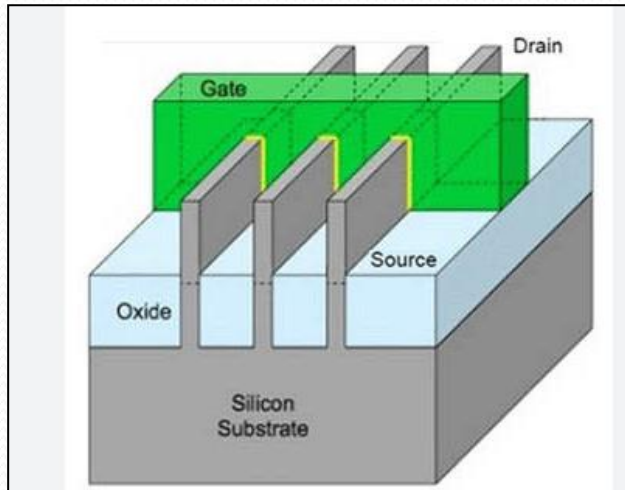
# AUMENTO EN EL GRADO DE INTEGRACIÓN. ¿DE QUE TAMAÑOS HABLAMOS?



# AUMENTO EN EL GRADO DE INTEGRACIÓN. ¿DE QUE TAMAÑOS HABLAMOS?

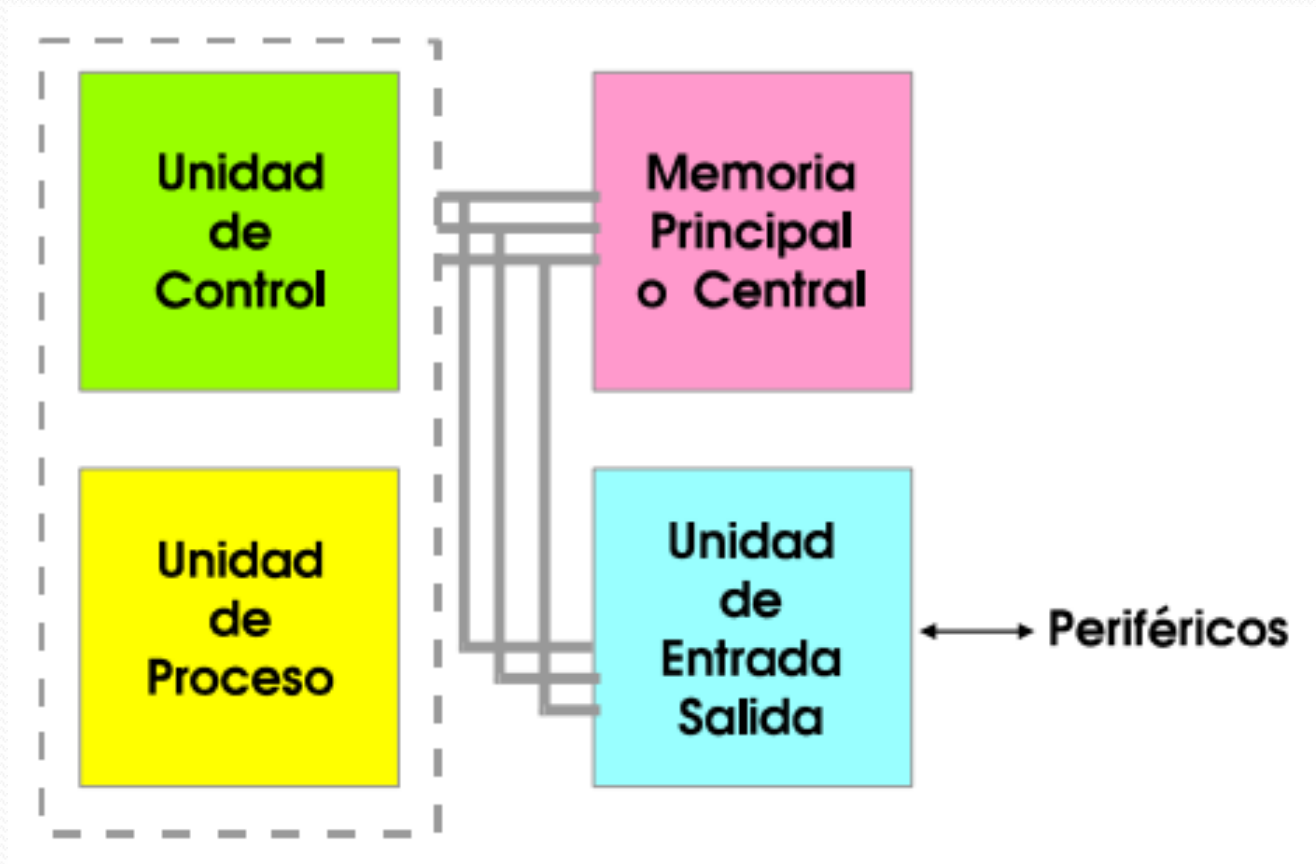


# AUMENTO EN EL GRADO DE INTEGRACIÓN. ¿DE QUE TAMAÑOS HABLAMOS?

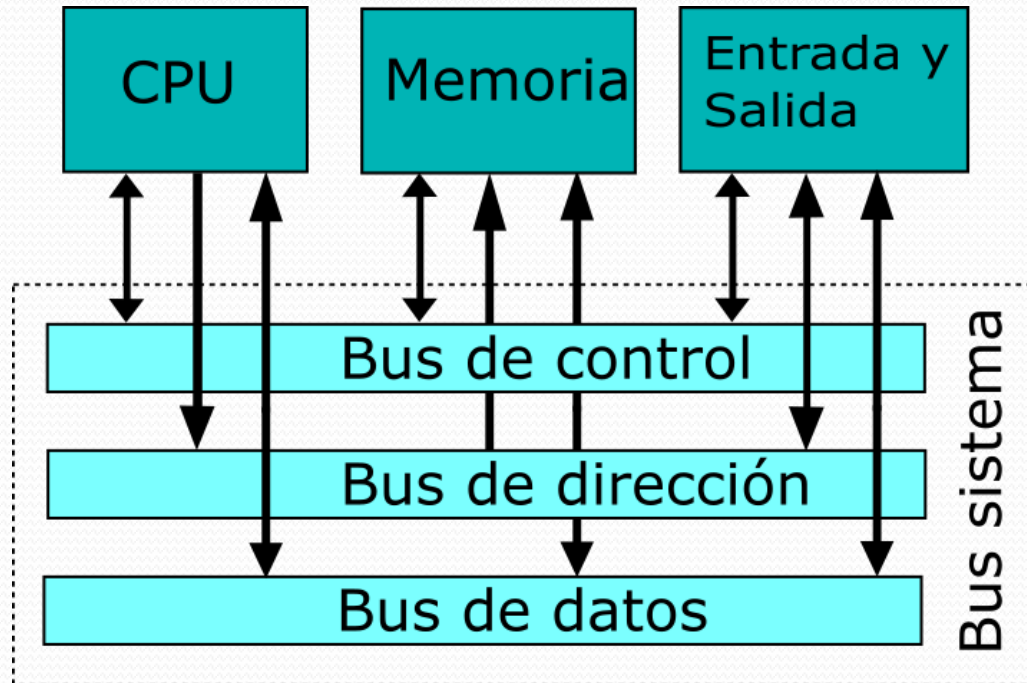




# ESQUEMA ORDENADOR



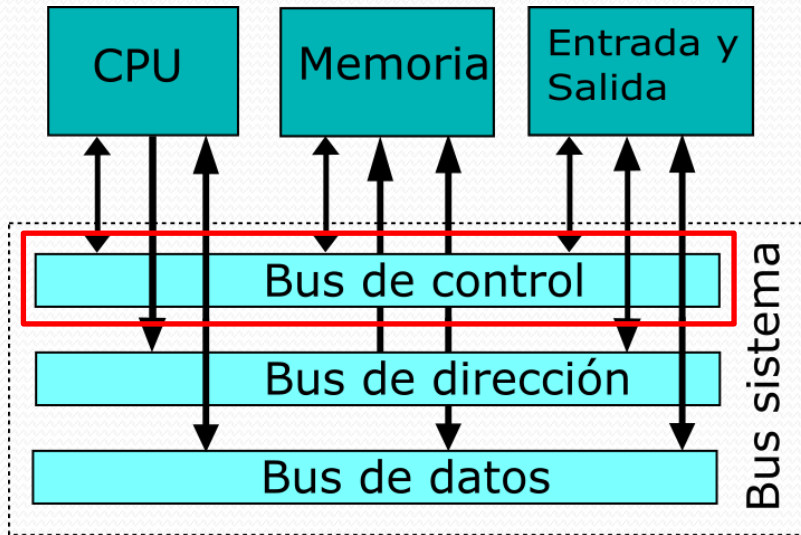
# BUSES.



Los buses son un “conjunto de cables” que transfieren información binaria, NO TIENEN CAPACIDAD PARA AMACENAR LA INFORMACION SÓLO LA TRANSMITEN. Su nombre; datos, control o dirección; se establece a partir del tipo de información que transmiten.



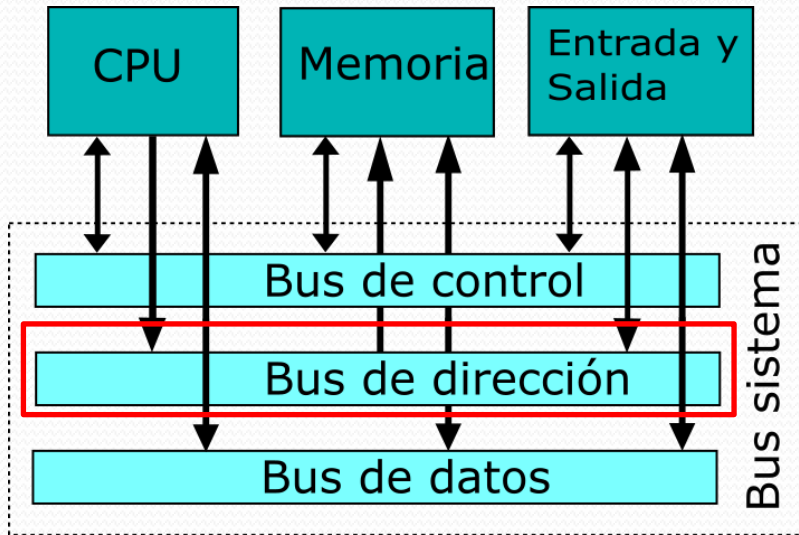
# BUSES.



Transporta las señales de control y de estado, indicando la dirección de la transferencia de datos, controlando la temporización de eventos durante la transferencia, transmitiendo las señales de interrupción, etc.

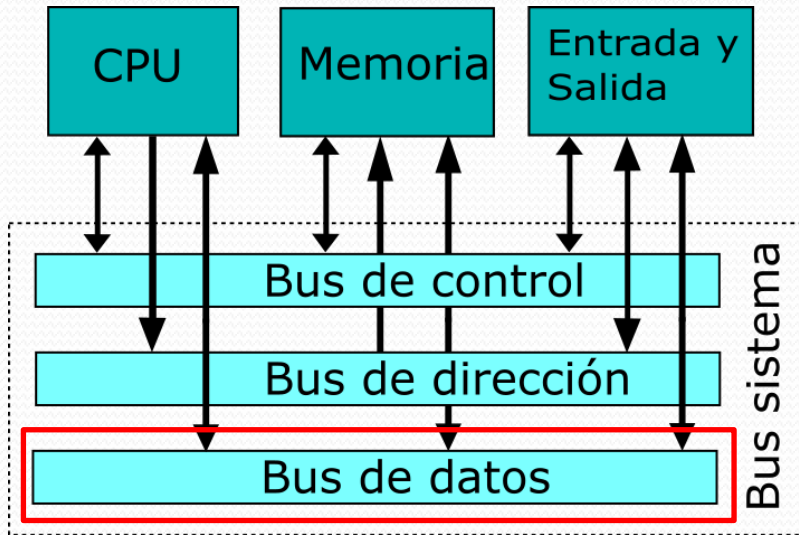
En definitiva, son las señales para controlar y sincronizar todos los componentes. Las señales de control parten desde la unidad de control hacia el resto de elementos, y las de estado parten del resto de los elementos hacia la unidad de control.

# BUSES.



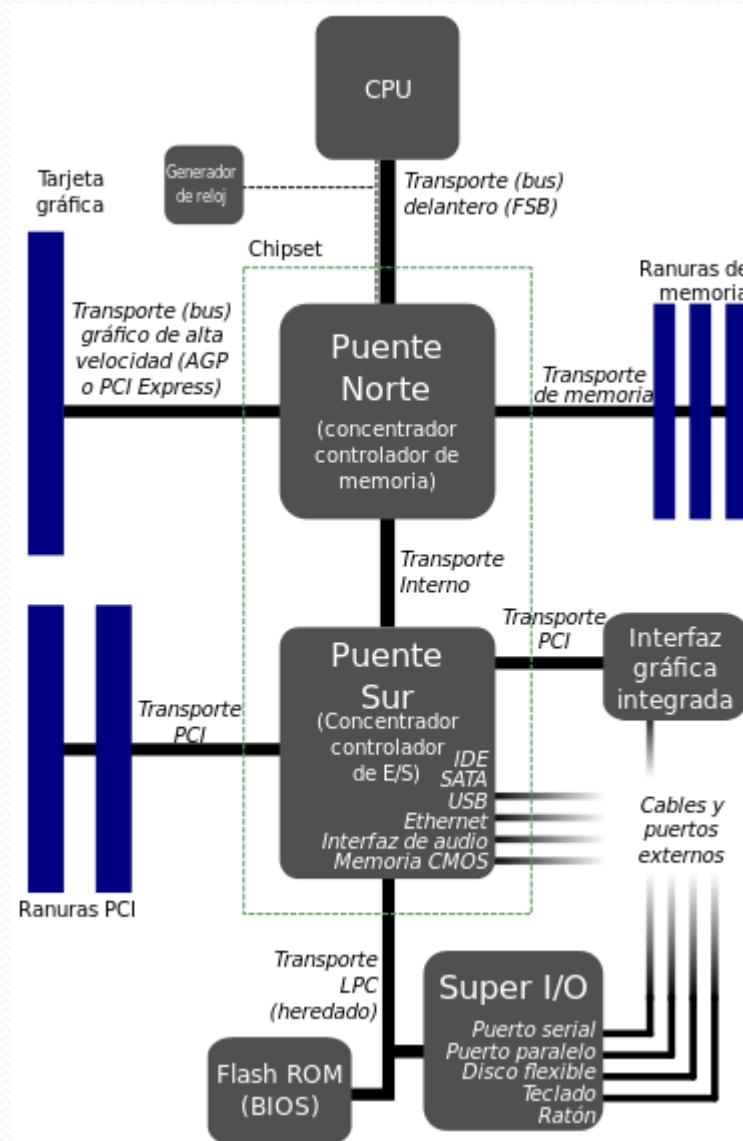
Transporta la dirección de la posición de memoria o del periférico que interviene en el tráfico de información (de dónde procede el dato o a dónde se dirige). Permite la comunicación entre el procesador y las celdas de la memoria RAM.

# BUSES.



Transporta los datos que se transfieren entre unidades. El número de líneas (y por tanto el número de bits) que utilice el bus de datos en un ordenador determina el tamaño de su palabra, es decir el tamaño del dato que se puede transmitir. Es bidireccional, es decir, los mismos hilos o cables se utilizan para transmitir información hacia dentro o hacia fuera de una unidad en instantes diferentes.

# ARQUITECTURA.



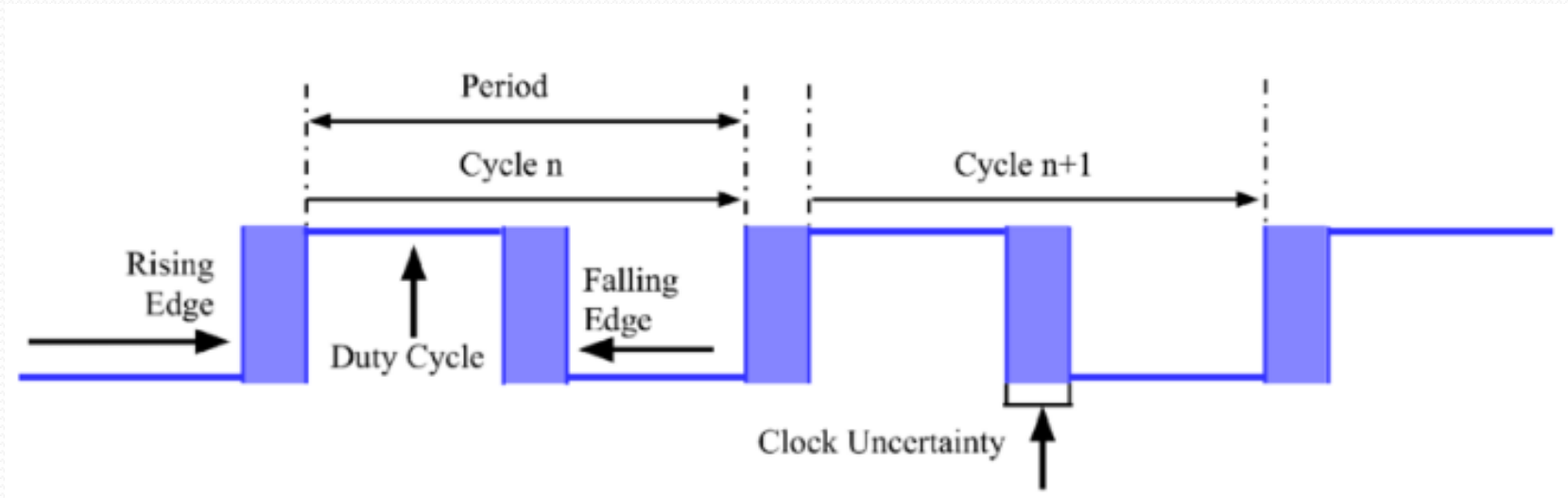
# EL RELOJ

Su función es la de marcar la sincronización de trabajo de todo el Esquema Ordenador. Logra que los dispositivos internos del Esquema Ordenador trabajen con un “ritmo” marcado por el mismo.

# EL RELOJ

- Este elemento está directamente asociado al microprocesador.
- La unidad de medida de la frecuencia del reloj es el Hertz.
- La mayoría de los dispositivos internos electrónicos del Esquema Ordenador trabajan con una frecuencia de reloj, pero fundamentalmente lo hace el microprocesador. Sin embargo no todas estas frecuencias son iguales lo que nos dice sencillamente que no todos los dispositivos que constituyen el Esquema Ordenador pueden trabajar a la misma velocidad y es precisamente por esta razón que debe existir una sincronización de actividad y trabajo entre ellos.

# EL RELOJ



# EL PROGRAMA

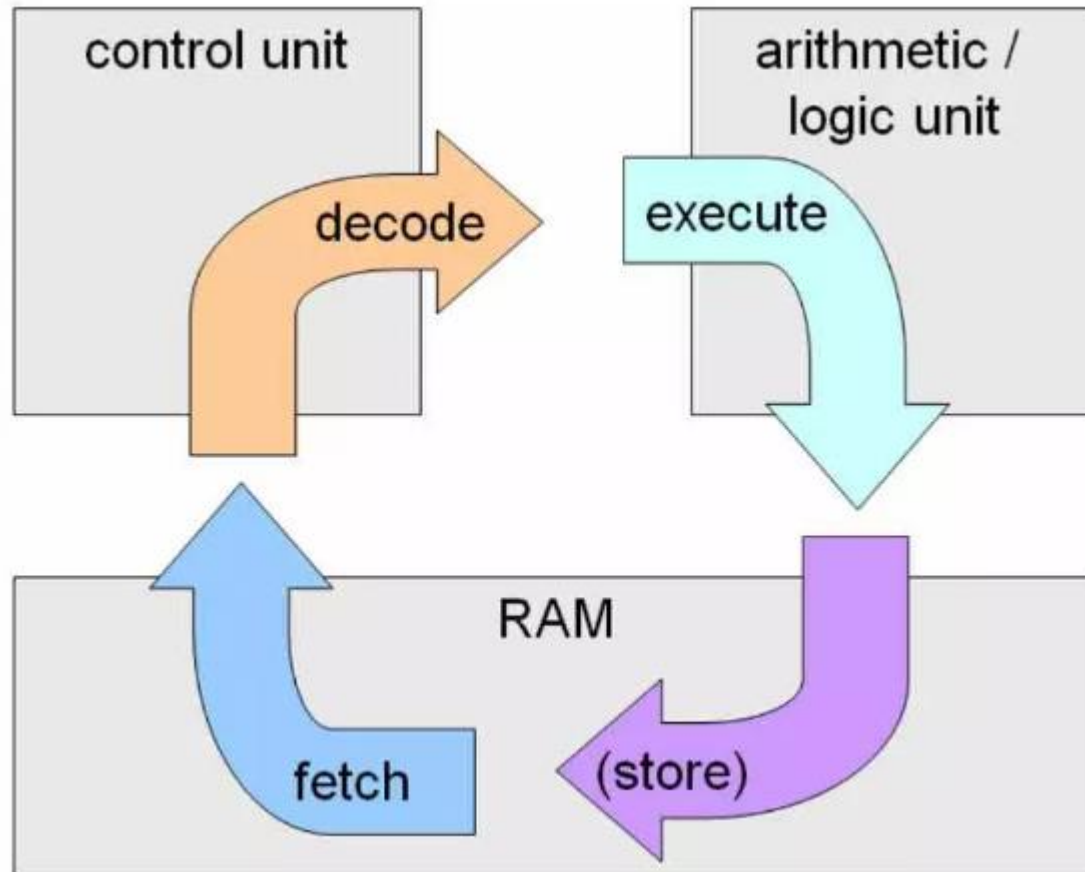
Cómo ya sabemos un programa es un algoritmo constituido por un conjunto de instrucciones que se ejecutan de manera secuencial o por el orden determinado por el propio algoritmo.

En consecuencia, para ejecutarlo, esta máquina que es el Esquema Ordenador siempre realiza la misma tarea que es:

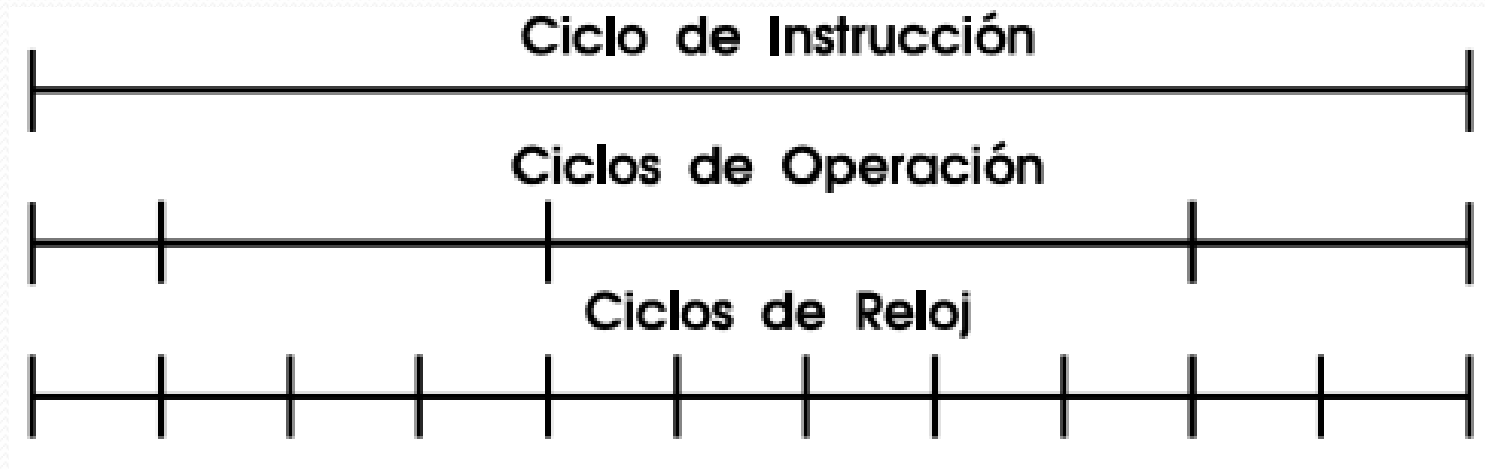
## Lee – Interpreta - Ejecuta



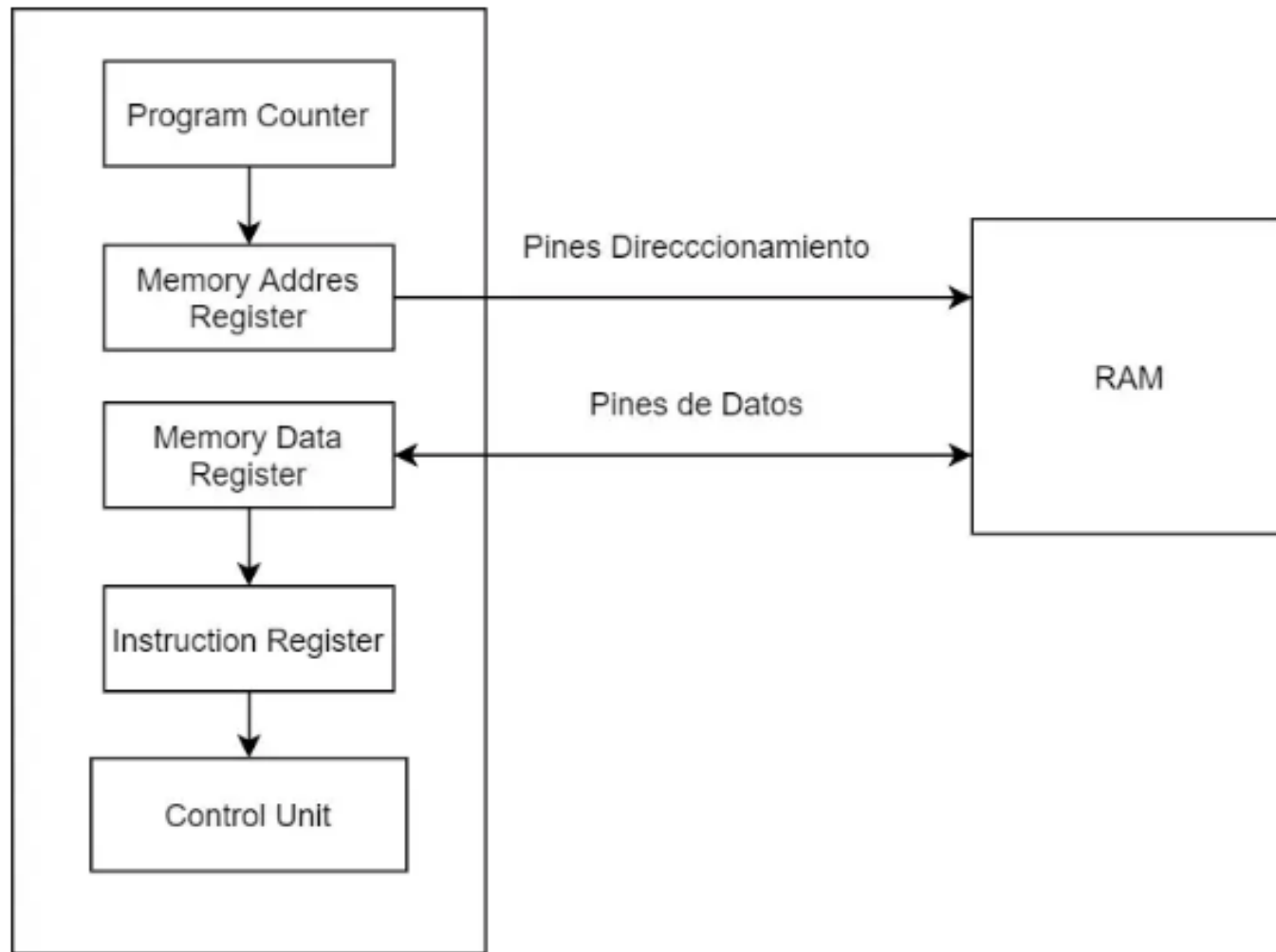
# EL CICLO DE INSTRUCCIÓN



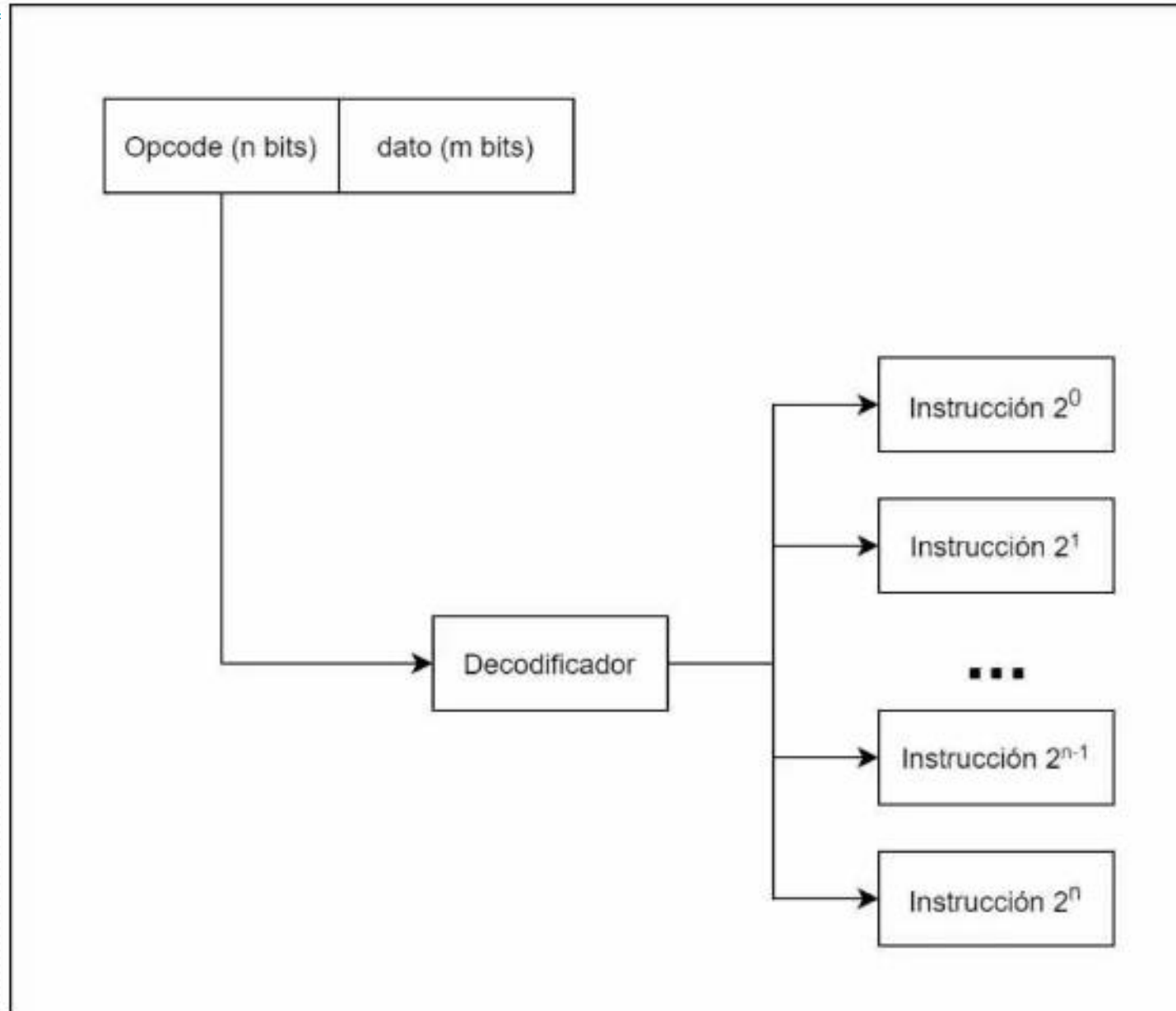
# EL CICLO DE INSTRUCCIÓN



## Primera etapa del ciclo de instrucción: Fetch



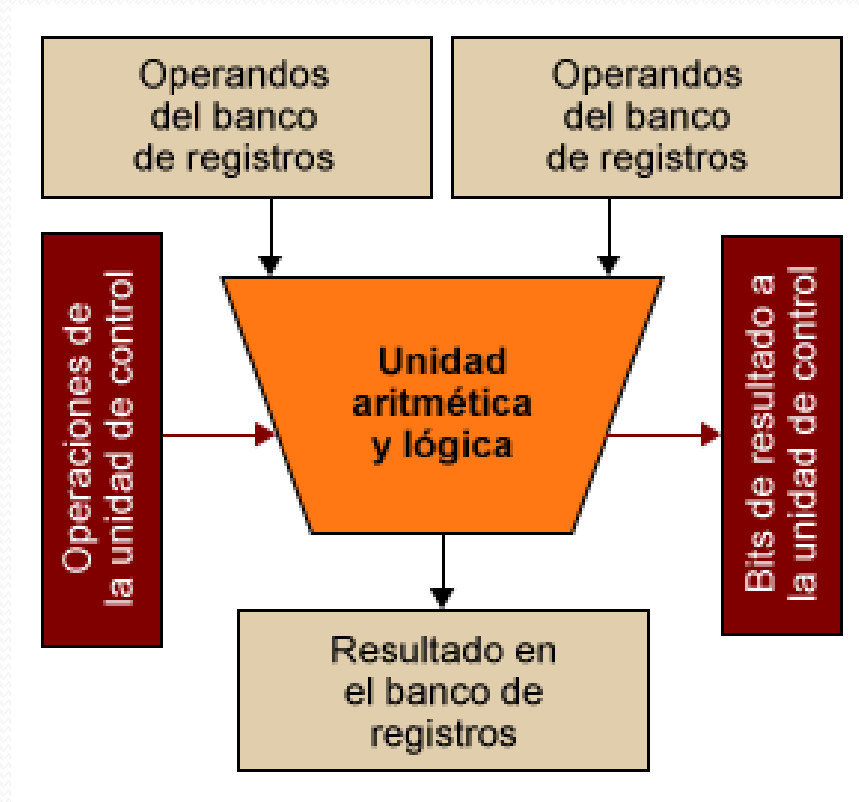
## La segunda etapa: Decode



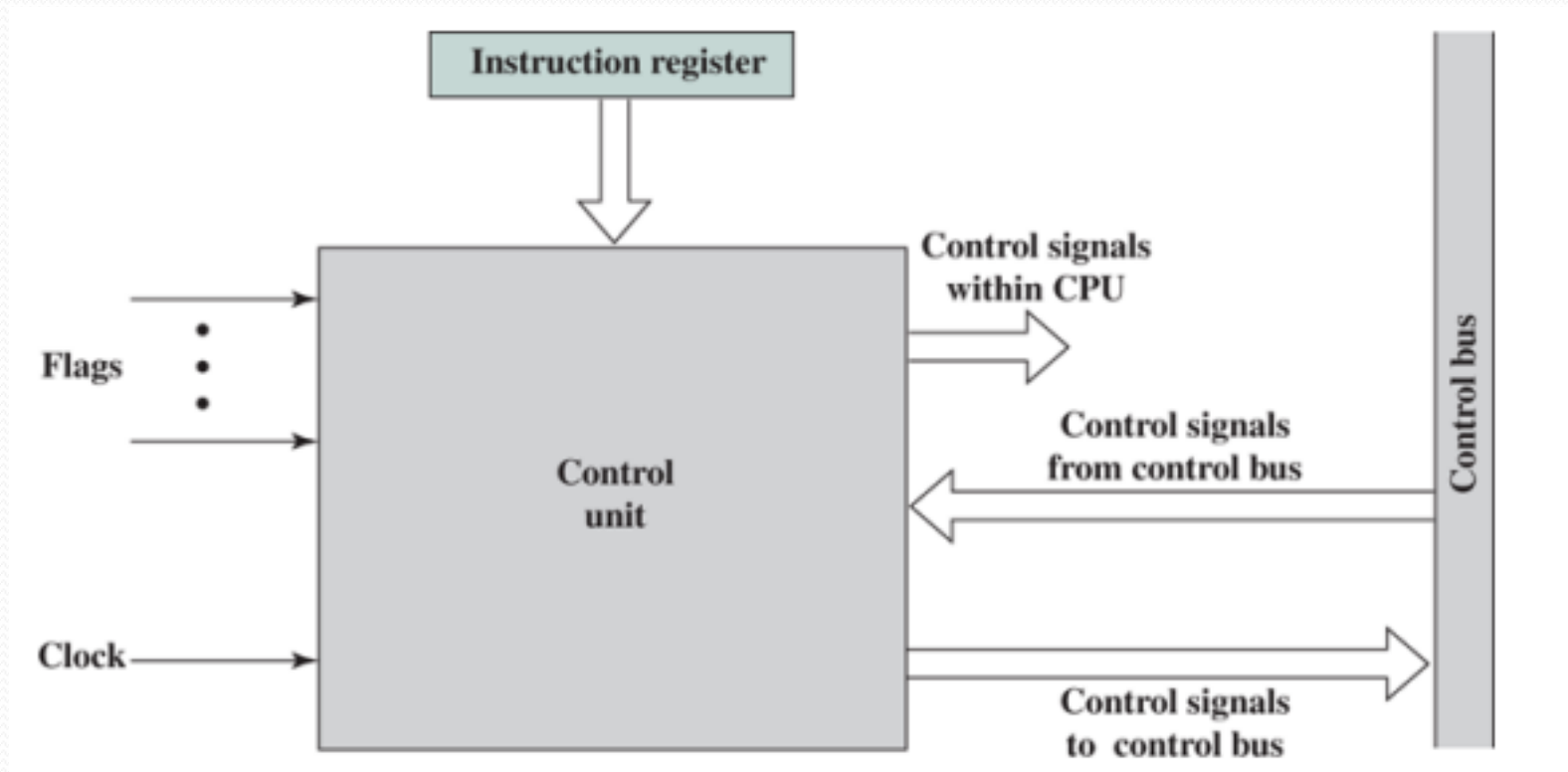
## Tercera etapa: Execute

- Instrucciones de movimiento de bits: En el cual se manipula el orden de los bits que contienen el dato.
- Instrucciones aritméticas: Donde se realizan operaciones matemáticas y también lógicas, estas se solucionan en las llamadas ALU o unidades aritmético-lógicas.
- Instrucciones de salto: En la que se cambia la siguiente el valor del contador de programa, lo que permite utilizar el código de manera recursiva.
- Instrucciones a memoria: Son con las que el procesador lee y escribe la información de la memoria del sistema.

# ALU

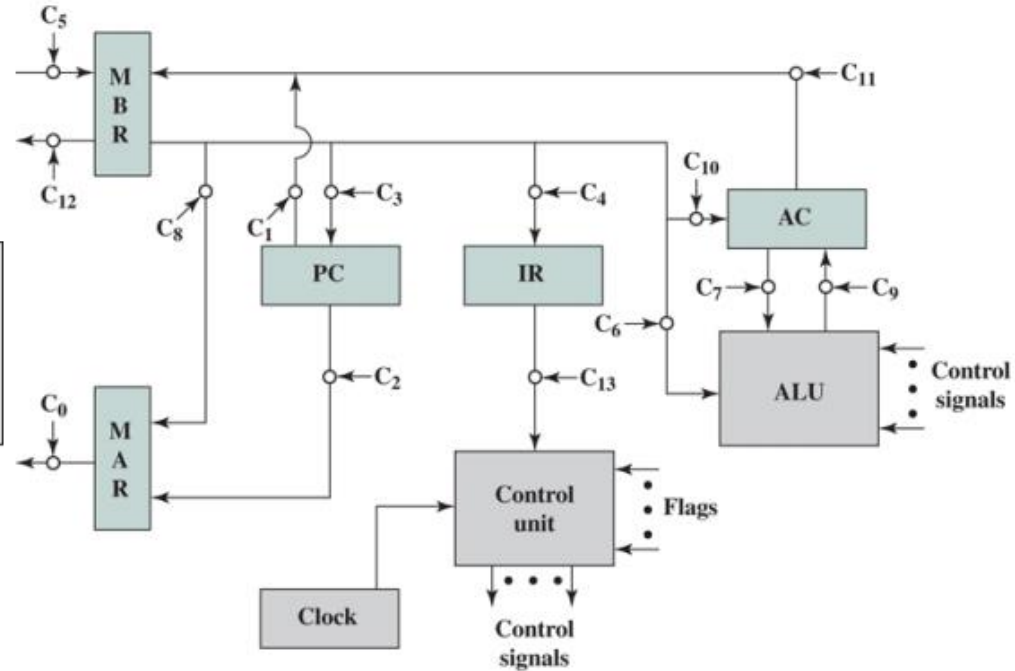
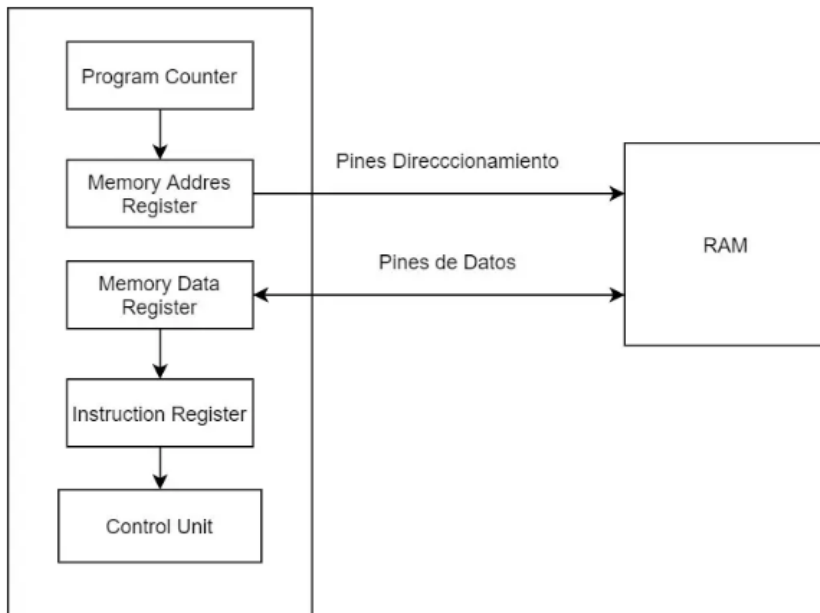


# UNIDAD DE CONTROL



# UNIDAD DE CONTROL

## Primera etapa del ciclo de instrucción: Fetch





# UNIDAD DE CONTROL (HARDWIRED)

Una unidad de control cableada es un componente clave en una CPU que se encarga de coordinar y controlar las operaciones de ejecución de instrucciones.

## Decodificación de Instrucciones:

En una unidad de control cableada, el circuito de control está diseñado con una serie de compuertas lógicas y circuitos combinacionales que están cableados de manera fija para decodificar las instrucciones. Cada instrucción tiene un código de operación (opcode) que se utiliza para determinar las operaciones de control necesarias para ejecutar la instrucción.

## Generación de Señales de Control:

El circuito de control cableado genera señales de control específicas en función de la instrucción actual. Estas señales activan o desactivan los componentes de la CPU, como la ALU, los registros y los buses, de acuerdo con los requisitos de la instrucción.

## Secuencia de Ejecución:

A medida que se ejecuta una instrucción, la unidad de control cableada emite una secuencia predefinida de señales de control en función del opcode de la instrucción. Estas señales de control dirigen las operaciones de la CPU en el orden correcto para completar la ejecución de la instrucción.

# UNIDAD DE CONTROL (HARDWIRED)

## Comportamiento Fijo:

A diferencia de la microprogramación, donde las secuencias de control son programables y se almacenan en una memoria de control, en una unidad de control cableada, las secuencias de control están cableadas de manera permanente. Esto significa que el comportamiento de la unidad de control está determinado por el diseño físico de los circuitos y no se puede cambiar sin modificar el hardware.

## Eficiencia y Rendimiento:

Las unidades de control cableadas tienden a ser más rápidas y eficientes en términos de rendimiento, ya que las señales de control se generan directamente a través de circuitos lógicos dedicados en lugar de ser accedidas desde una memoria.

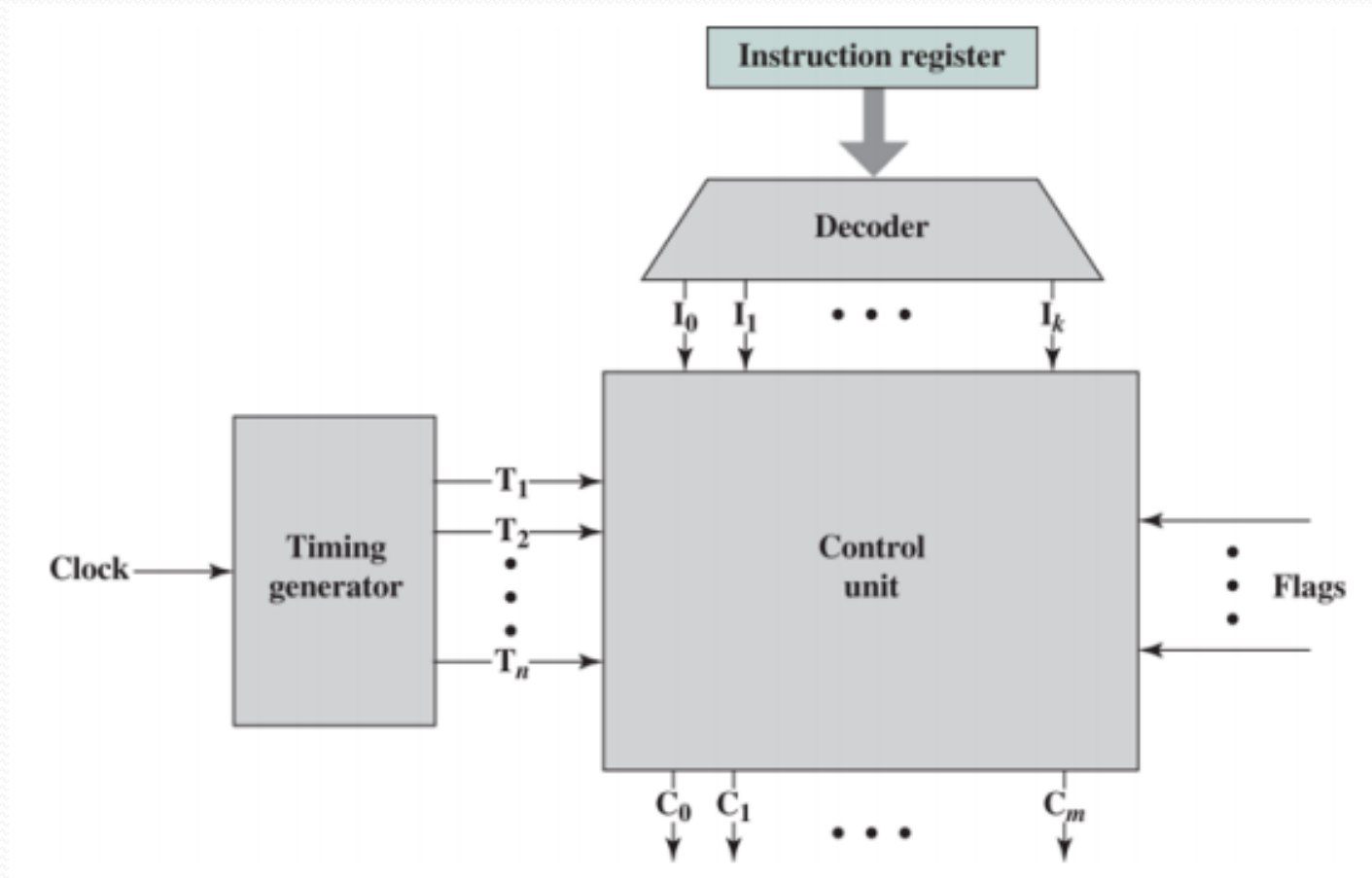
## Limitaciones y Flexibilidad:

Aunque las unidades de control cableadas son eficientes, pueden ser menos flexibles en términos de actualizaciones y cambios. Cualquier modificación en las instrucciones o en el comportamiento de control requeriría cambios físicos en el diseño del procesador.

# UNIDAD DE CONTROL (HARDWIRED)

Una unidad de control cableada utiliza circuitos lógicos fijos y cableados para decodificar las instrucciones y generar señales de control que dirigen las operaciones de la CPU. Aunque son eficientes, carecen de la flexibilidad de actualización que ofrecen las unidades de control microprogramadas.

# UNIDAD DE CONTROL (HARDWIRED)



# UNIDAD DE CONTROL (μPROGRAMADA)

Una unidad de control microprogramada es una parte esencial de una CPU que se encarga de coordinar y controlar las operaciones de ejecución de instrucciones.

## Microprogramación:

En lugar de implementar la lógica de control directamente en hardware físico, la microprogramación utiliza un enfoque más flexible y programable. Se define un conjunto de microinstrucciones que controlan las operaciones de la CPU. Estas microinstrucciones están almacenadas en una memoria llamada memoria de control o ROM de control.

## Secuencia de Instrucciones:

Cuando una instrucción se ejecuta, se descompone en una secuencia de microinstrucciones. Cada microinstrucción representa una operación elemental que la CPU debe realizar para ejecutar la instrucción original.

# UNIDAD DE CONTROL (μPROGRAMADA)

## Secuencia de Ejecución:

La unidad de control microprogramada selecciona y ejecuta las microinstrucciones en una secuencia específica para completar la ejecución de la instrucción original. Cada microinstrucción controla los componentes internos de la CPU, como la ALU, los registros y los buses.

## Decodificación y Control:

La unidad de control microprogramada interpreta el código de operación (opcode) de la instrucción actual para determinar la secuencia de microinstrucciones necesarias. Esta decodificación se realiza consultando una tabla de control almacenada en la memoria de control.

## Flexibilidad y Actualización:

Una de las ventajas clave de la microprogramación es su capacidad para actualizar y modificar el comportamiento de la CPU sin cambiar su hardware físico. Esto permite realizar cambios y mejoras en las operaciones de control sin tener que rediseñar todo el procesador.

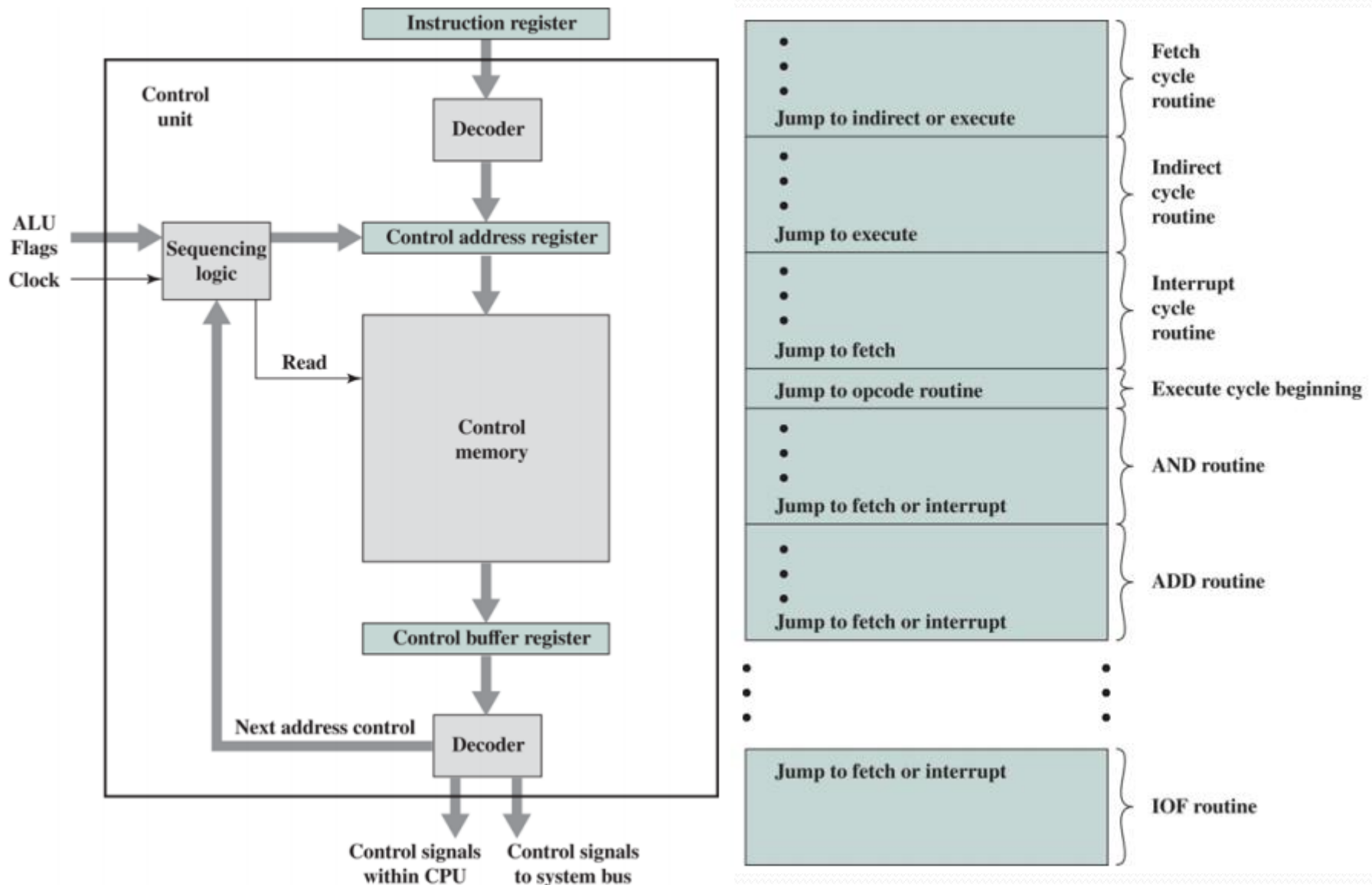
## Ciclo de Instrucción:

En cada ciclo de instrucción, la unidad de control microprogramada busca la siguiente microinstrucción en la secuencia y la ejecuta. Una vez que se completan todas las microinstrucciones necesarias para una instrucción, la CPU está lista para ejecutar la siguiente instrucción.

# UNIDAD DE CONTROL ( $\mu$ PROGRAMADA)

Una unidad de control microprogramada utiliza una secuencia de microinstrucciones almacenadas en memoria para controlar y coordinar las operaciones de la CPU durante la ejecución de instrucciones. Este enfoque proporciona flexibilidad y programabilidad, lo que permite actualizar y modificar el comportamiento de la CPU de manera más eficiente.

# UNIDAD DE CONTROL (μPROGRAMADA)





# UNIDAD DE CONTROL (μPROGRAMADA)

