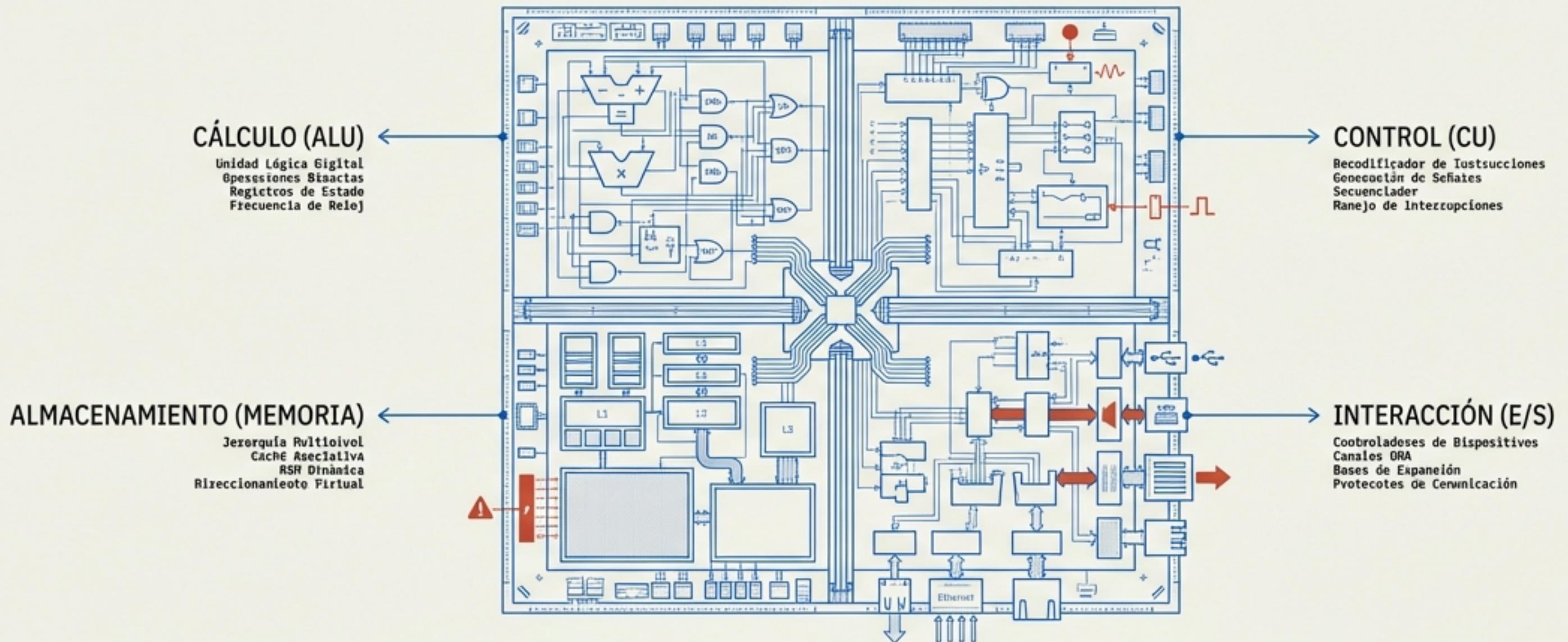


ARQUITECTURA DE COMPUTADORAS: DEL NÚCLEO A LA PERIFERIA

Una exploración integral de ALUs, Unidades de Control, Jerarquía de Memoria y Sistemas de E/S



Este material adopta un enfoque constructivo ('bottom-up') para desmitificar el hardware moderno.
El Recorrido:

1. Cálculo: La lógica digital pura dentro de la ALU.
2. Control: La toma de decisiones en la Unidad de Control.
3. Almacenamiento: La gestión de datos a través de la Jerarquía de Memoria.
4. Interacción: La comunicación con el mundo exterior mediante E/S.



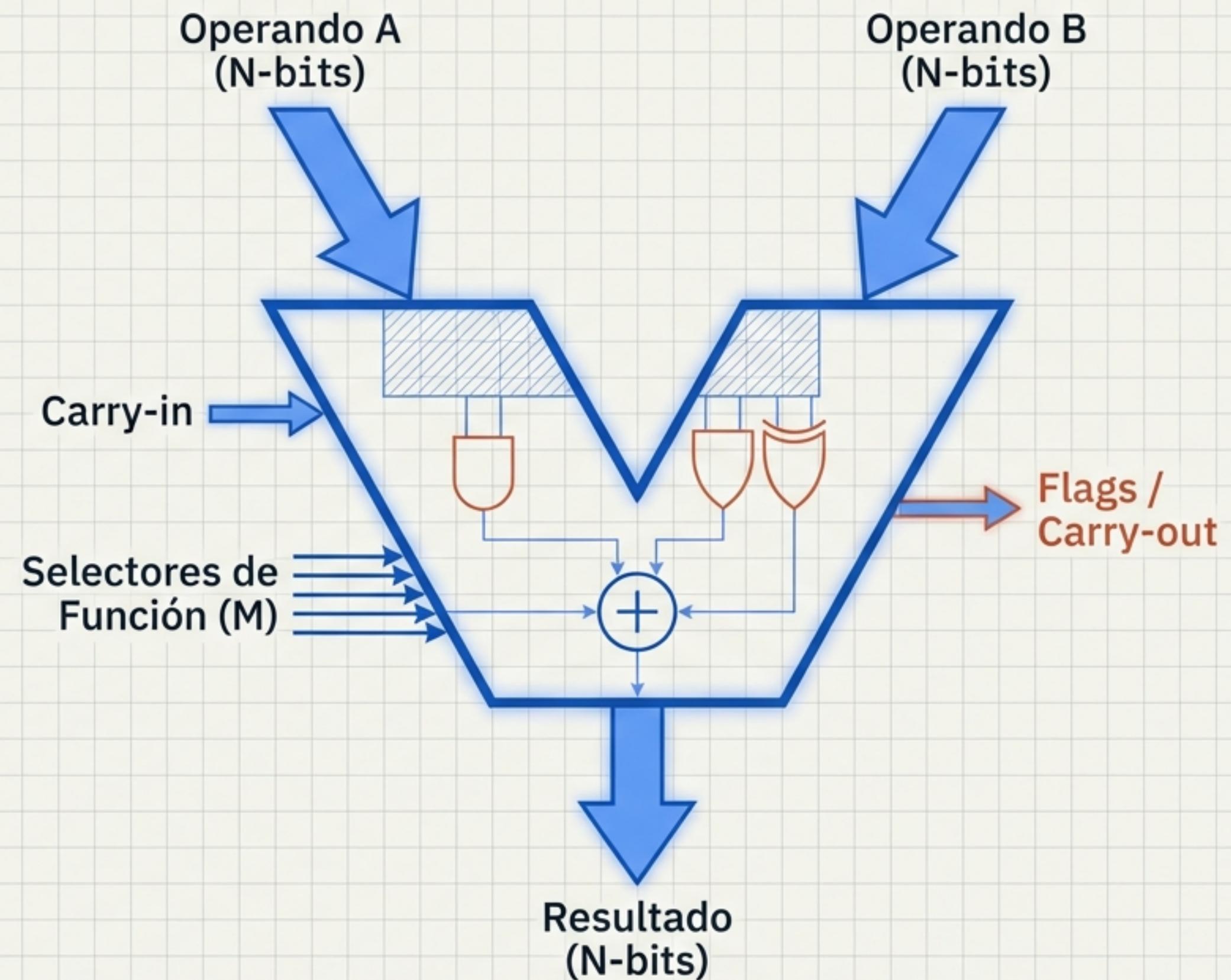
El Motor Matemático: La Unidad Aritmético Lógica (ALU)

Definición:

La ALU es el circuito combinacional central del microprocesador, encargado de ejecutar funciones aritméticas y lógicas.

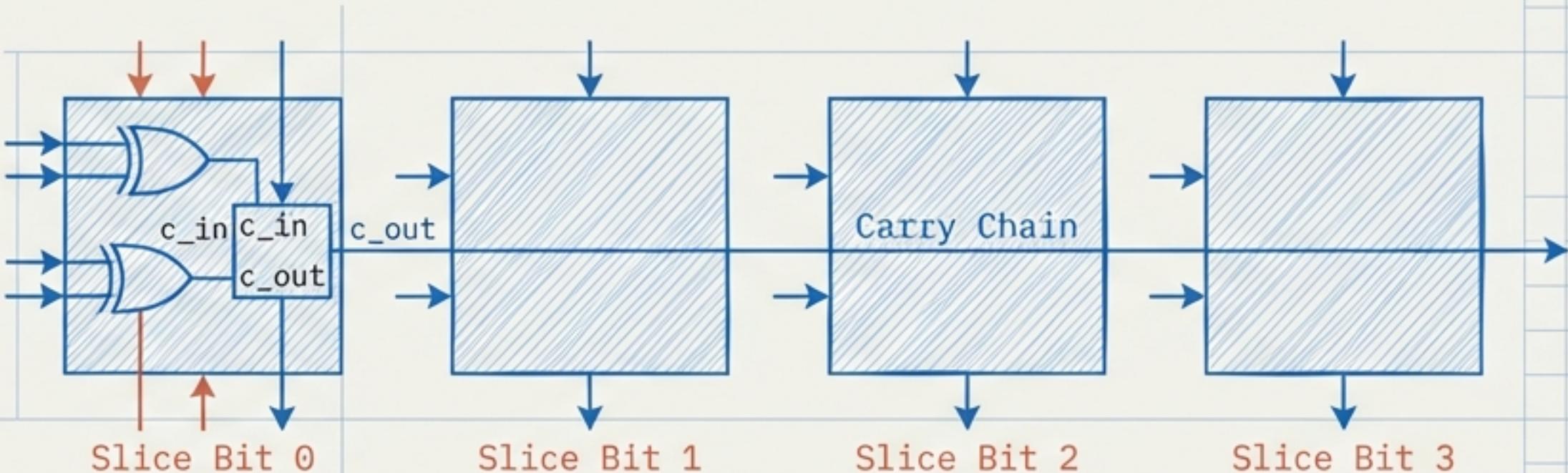
Funciones Típicas:

- Suma / Resta
- Multiplicación
- AND / OR / XOR / INV
- Desplazamiento (Shift)



Diseño Modular: La Metodología Bit-Slice

El Enfoque Bit-Slice: Dado que las ALUs operan sobre números binarios, el diseño se simplifica creando un bloque funcional para un solo bit y replicándolo N veces.



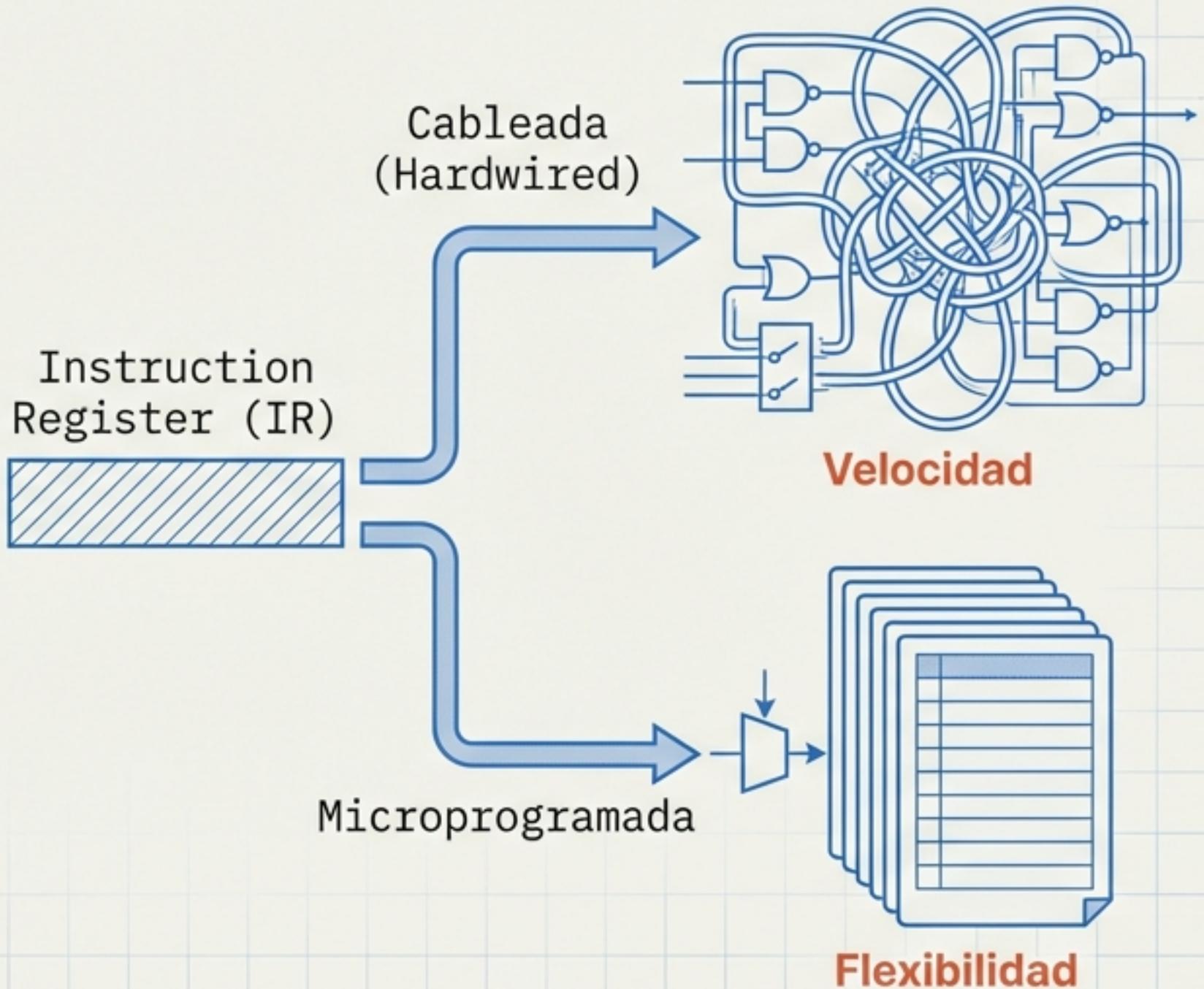
Ejemplo de OpCodes (Código de Operación)

OpCode	Descripción	Función
000	Suma	$F = A + B$
010	Resta	$F = A - B$
100	XOR bit a bit	$F = A \oplus B$
111	AND bit a bit	$F = A \wedge B$

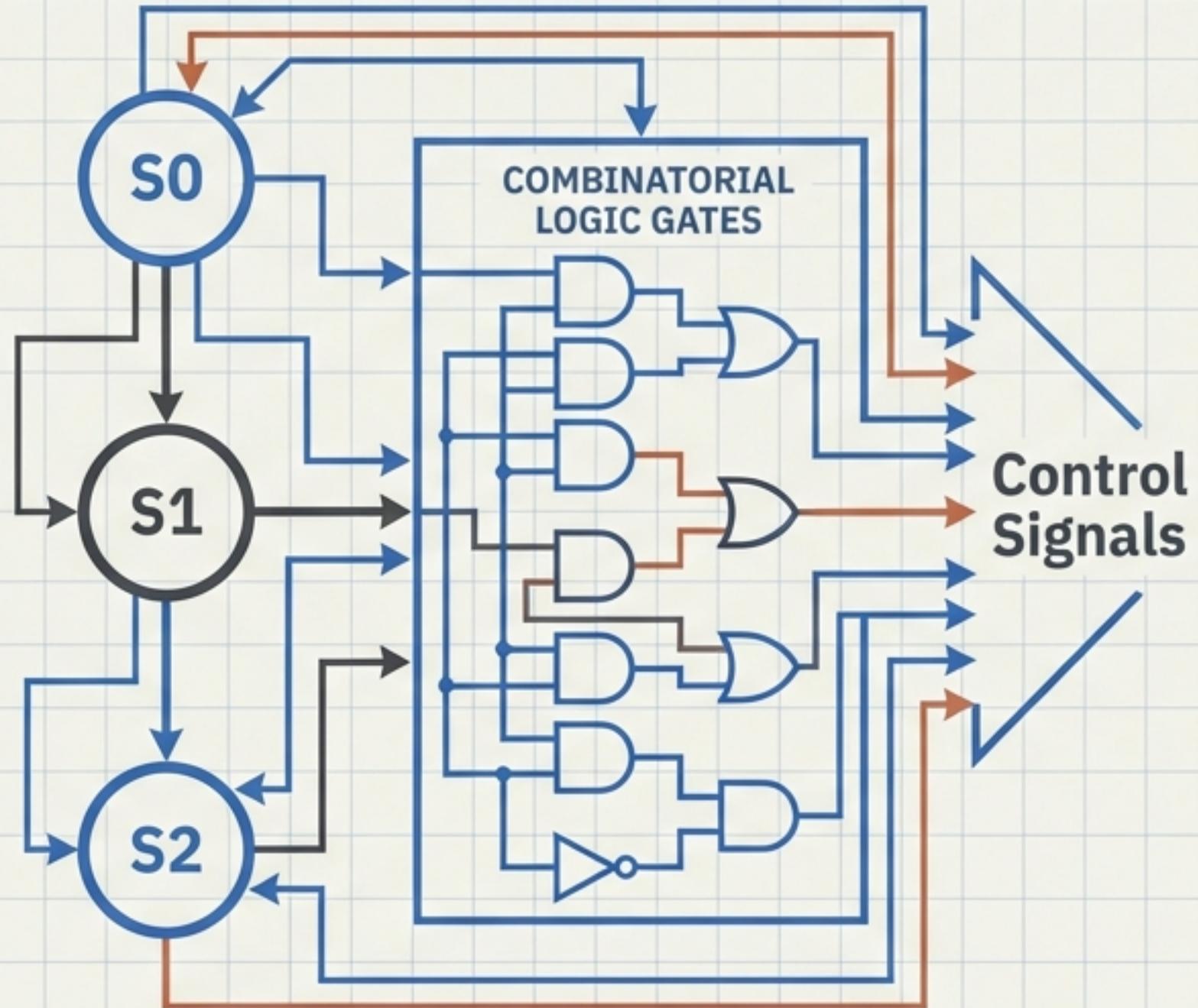
El Director de Orquesta: La Unidad de Control(CU)

****Misión Crítica:**** La fuerza bruta de la ALU es inútil sin dirección. La CU decodifica instrucciones y genera las señales de control precisas para mover datos y activar circuitos específicos.

El Gran Dilema de Diseño Existentes dos filosofías opuestas para construir este componente, definiendo la 'personalidad' del procesador.



Arquitectura Cableada: Velocidad Pura en Hardware



Mecanismo: Utiliza circuitos lógicos combinacionales fijos y máquinas de estado finito para generar señales basándose directamente en el OpCode.

Ventajas:

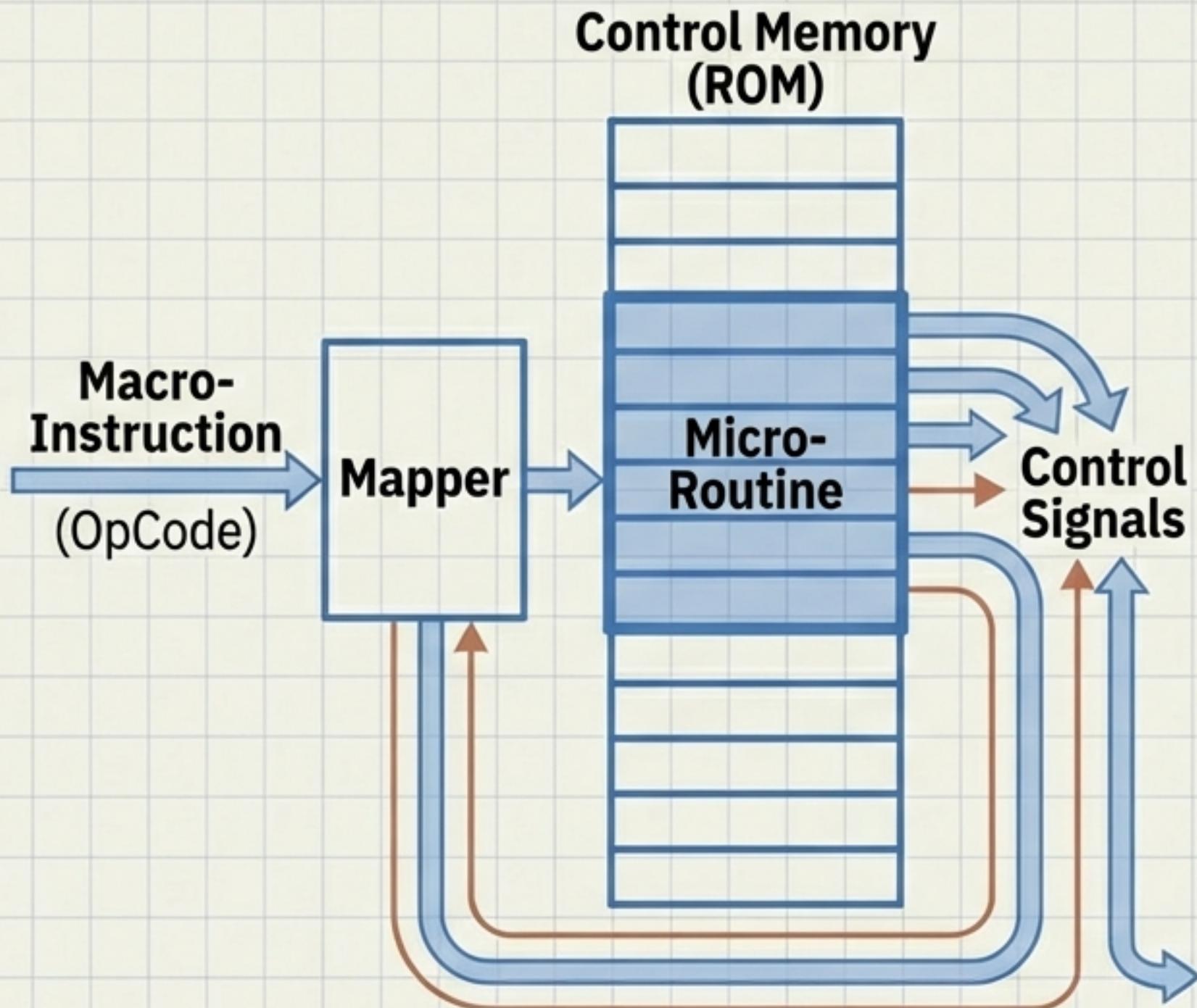
- Velocidad: Enfoque más rápido (sin latencia de fetch).
- Eficiencia: Ideal para RISC y procesadores ARM.

Desventajas:

- Rigidez: Difícil de modificar. Cambiar una instrucción requiere rediseñar el chip físico.
- Complejidad Geométrica: El circuito crece drásticamente con sets de instrucciones complejos.



Arquitectura Micropogramada: Flexibilidad vía Firmware



Mecanismo: Utiliza una 'memoria de control' interna que almacena microinstrucciones. Cada instrucción de máquina desencadena una secuencia de micro-pasos.

Ventajas:

- Adaptabilidad: Fácil de modificar actualizando el firmware.
- Potencia: Permite manejar arquitecturas complejas (CISC, x86 legacy, Mainframes).

Desventajas:

- Latencia: Ligeramente más lenta debido al "overhead" de búsqueda.
- Costo de Área: Requiere espacio físico extra para la ROM/RAM de control.



Batalla de Arquitecturas: Cableada vs. Microprogramada

Aspecto	Unidad de Control Cableada	Unidad de Control Microprogramada
Diseño	Circuitos combinacionales fijos	Secuencia de microinstrucciones en memoria
Modificabilidad	Difícil (requiere cambios de hardware)	Fácil (actualización de microprograma)
Velocidad	Más rápida	Generalmente más lenta (fetch overhead)
Complejidad	Alta en sets de instrucciones complejos	Fácil de implementar para sets complejos
Memoria	No requiere memoria de control	Requiere ROM/RAM para microcódigo
Casos de Uso	Procesadores RISC, Gaming, GPUs	Mainframes, CISC Legacy (Intel x86)

La Pirámide de Almacenamiento: Jerarquía de Memoria



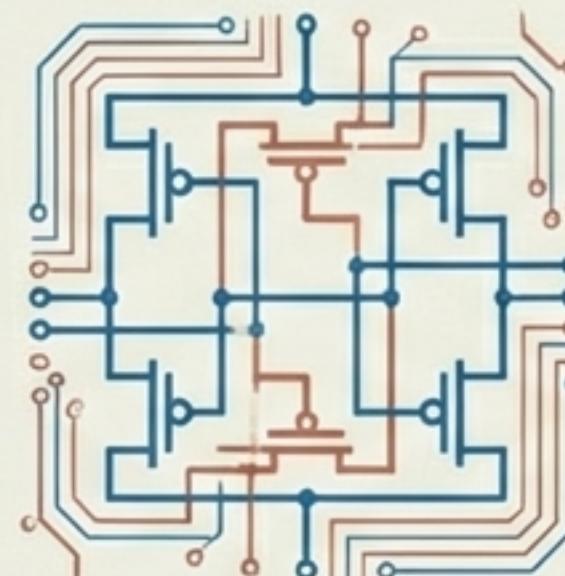
Principio Rector: Localidad de Referencia

- **Temporal**: Si usas un dato hoy, probablemente lo usarás pronto.
- **Espacial**: Si usas un dato, probablemente usarás a sus vecinos.

Tecnologías de Memoria Primaria: RAM y ROM

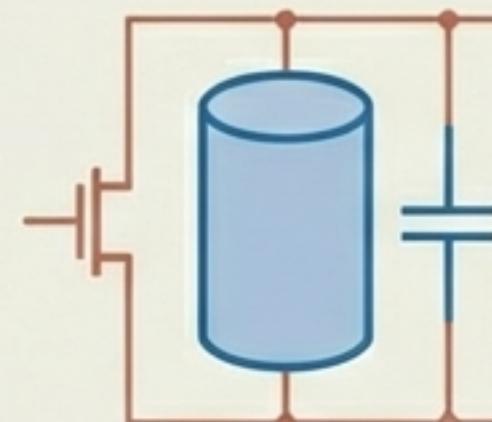
RAM (Random Access Memory)

Volátil - Datos de trabajo



SRAM (Estática)

Rápida, cara, usada en Caché. No requiere refresco.



DRAM (Dinámica)

Alta densidad, usada en RAM principal. Requiere refresco constante.

ROM (Read Only Memory)

No Volátil - Firmware/Bootstrap

PROM:

Programable una vez.

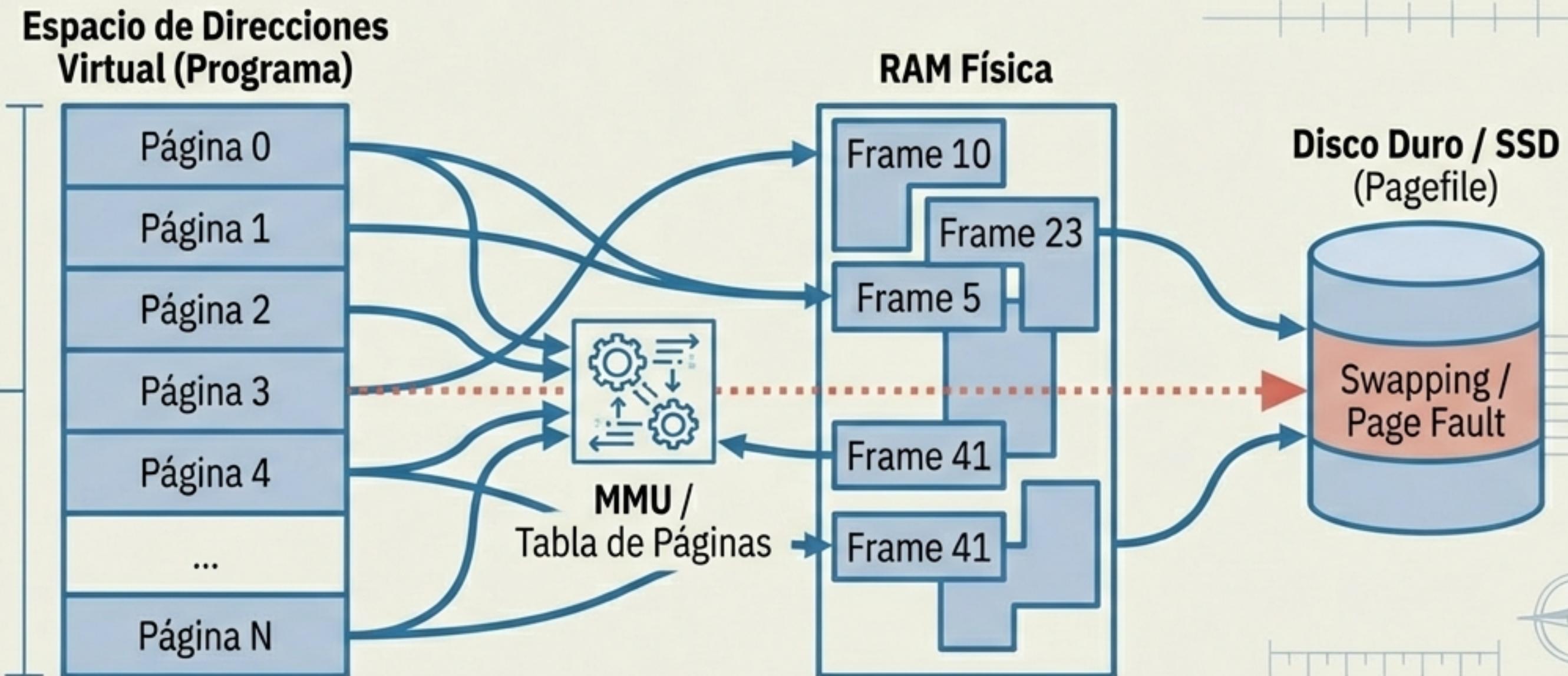
EPROM:

Borrable con luz UV.

EEPROM:

Borrable eléctricamente (byte a byte).

Memoria Virtual: La Ilusión de Capacidad Infinita

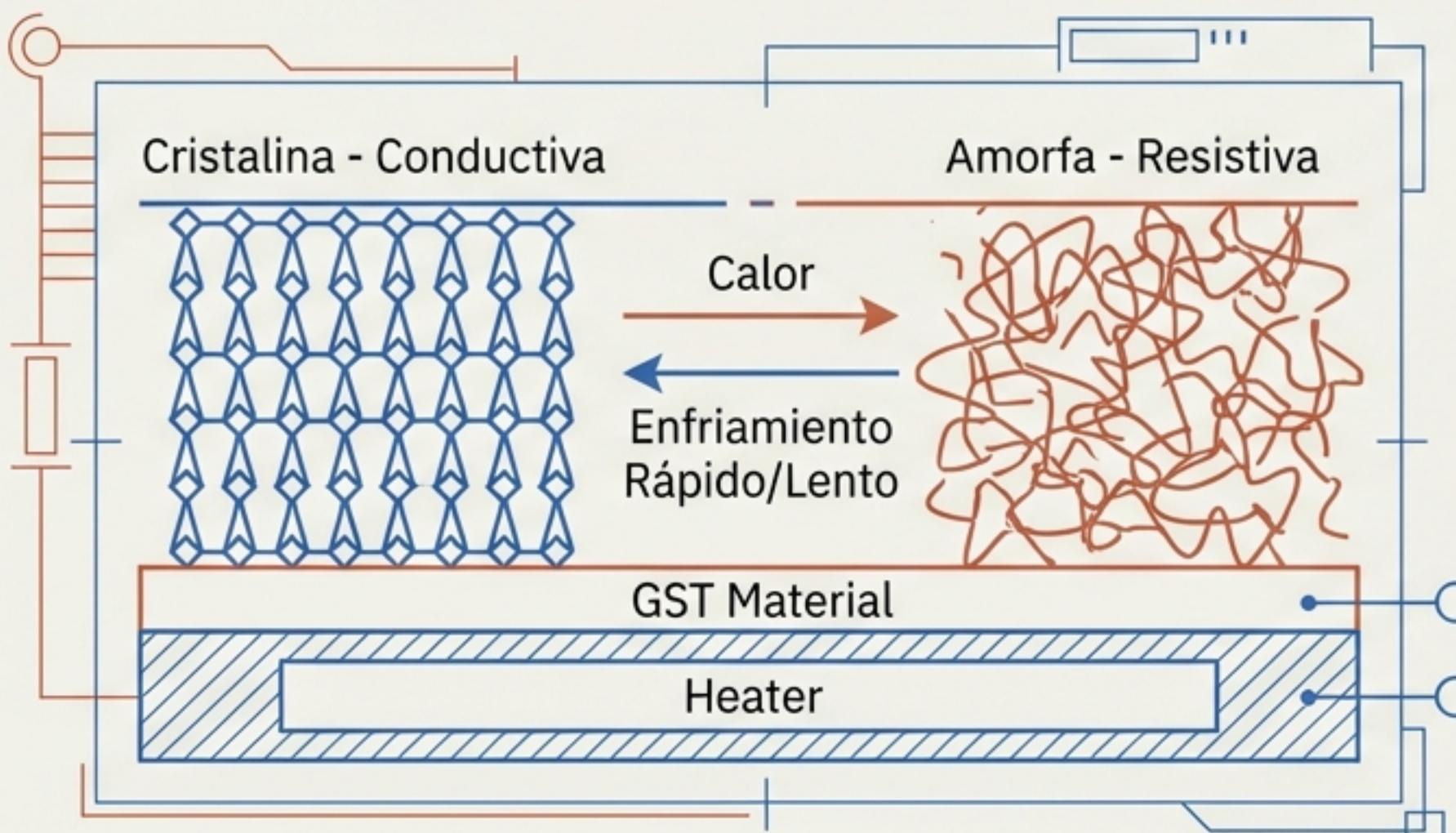


Concepto: Permite ejecutar programas más grandes que la memoria física usando el disco como extensión.
Mecanismo: Paganación por demanda. Solo los bloques activos residen en RAM; los inactivos se mueven al disco.

El Futuro: Tecnologías de Memoria No Volátil (NVM)

El Muro de la DRAM:

Límites de escalabilidad (<30nm) y alto consumo energético por refresco (hasta 47% de energía).



La Solución PCM (Phase Change Memory):

- **Material:** Vidrio de calcogenuro (GST) que cambia de fase con calor.
- **Ventaja:** No volátil (sin refresco), escalable, mayor densidad.
- **Ventaja:** No volátil (sin refresco), escalable, mayor densidad.
- **Desafío:** Latencia de lectura ~48% mayor que DRAM. Escritura costosa costosa energéticamente.

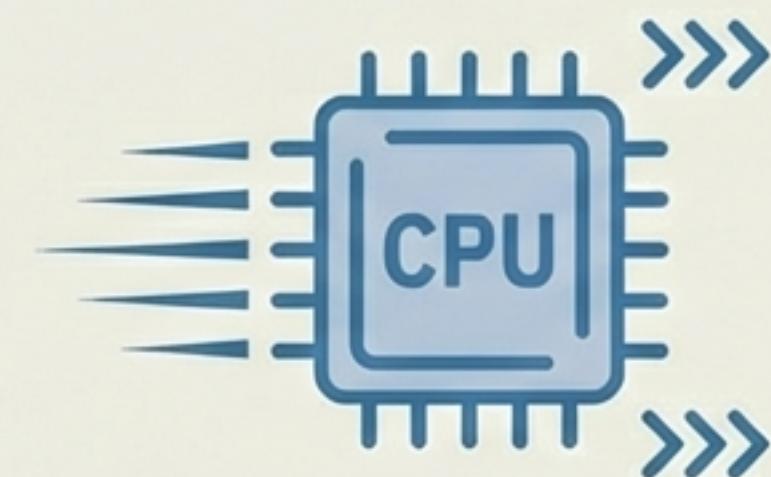


UPPSALA
UNIVERSITET



Universidad
Andina
del Cusco

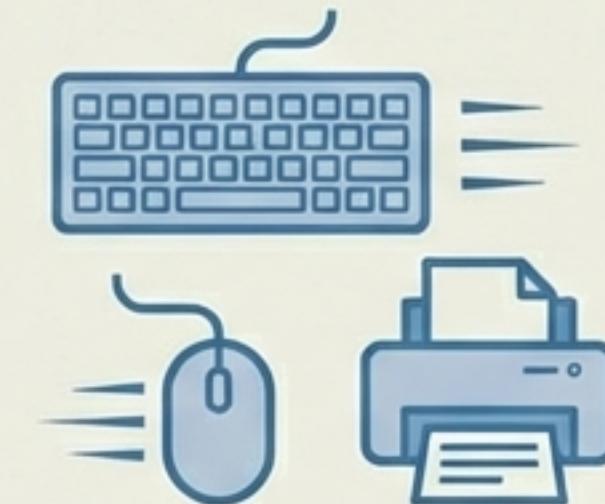
Comunicación con el Exterior: El Dilema de Entrada/Salida



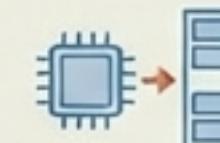
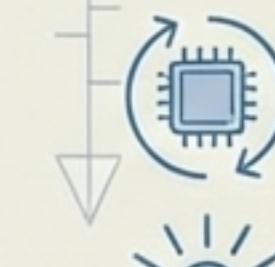
CPU @ GHz
(Miles de millones de ciclos/seg)

Speed Mismatch

! Synchronization Gap !



Periféricos Human-Scale
(Lentos)



1. **E/S Programada**: El CPU “pregunta” constantemente si hay datos.
(Ineficiente).
2. **E/S por Interrupciones**: El periférico “avisa” al CPU.
(Eficiente).
3. **DMA (Direct Memory Access)**: Transferencia directa a memoria sin molestar al CPU.

Eficiencia de E/S: Sondeo (Polling) vs. Interrupciones

Polling / Programada



Busy Wait (Desperdicio de CPU)

Interrupciones



Multitarea Real



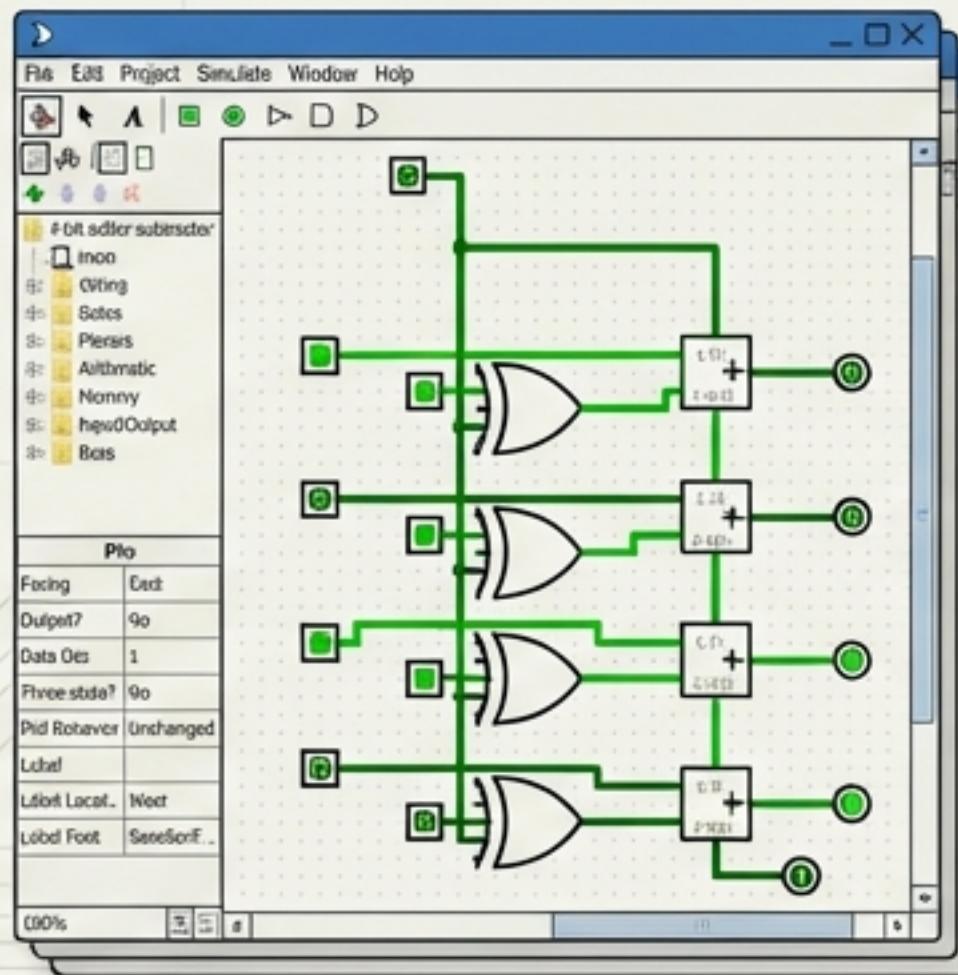
UPPSALA
UNIVERSITET



Universidad
Andina
del Cusco

Del Teórico al Práctico: Herramientas de Simulación

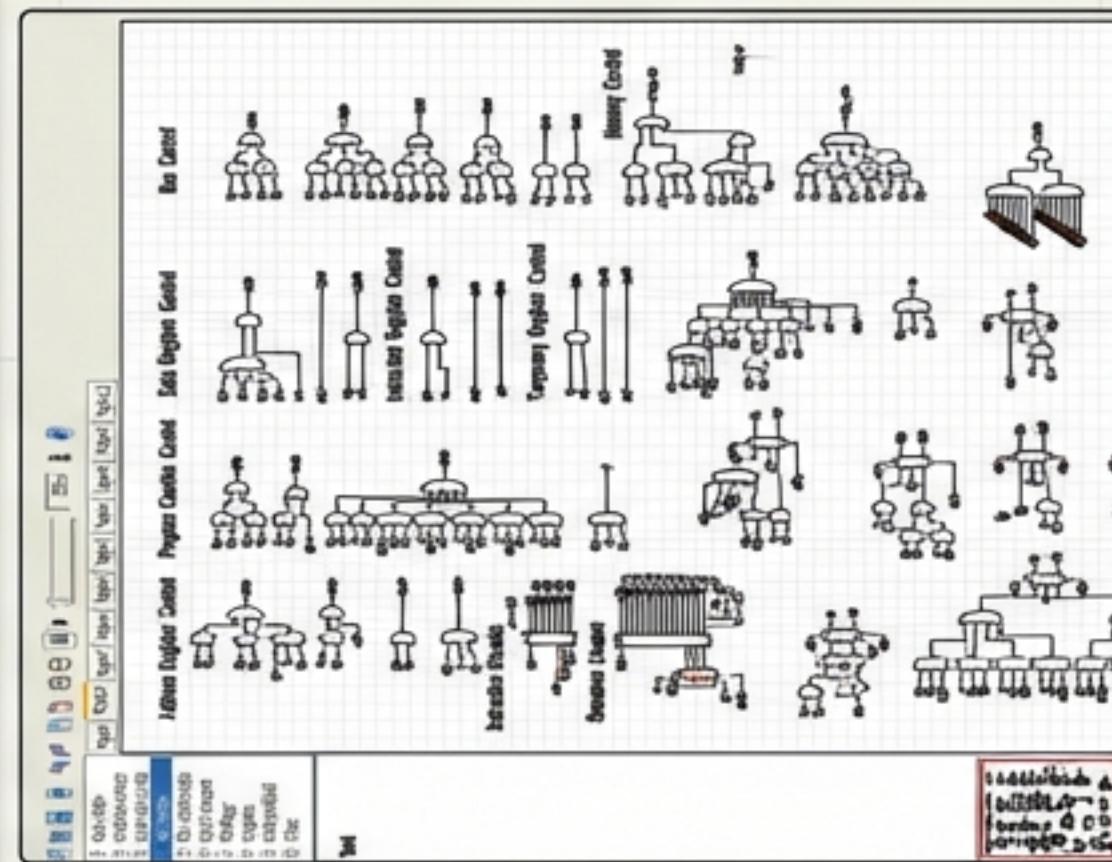
Logisim



Logisim

Nivel Puertas Lógicas
& Datapath

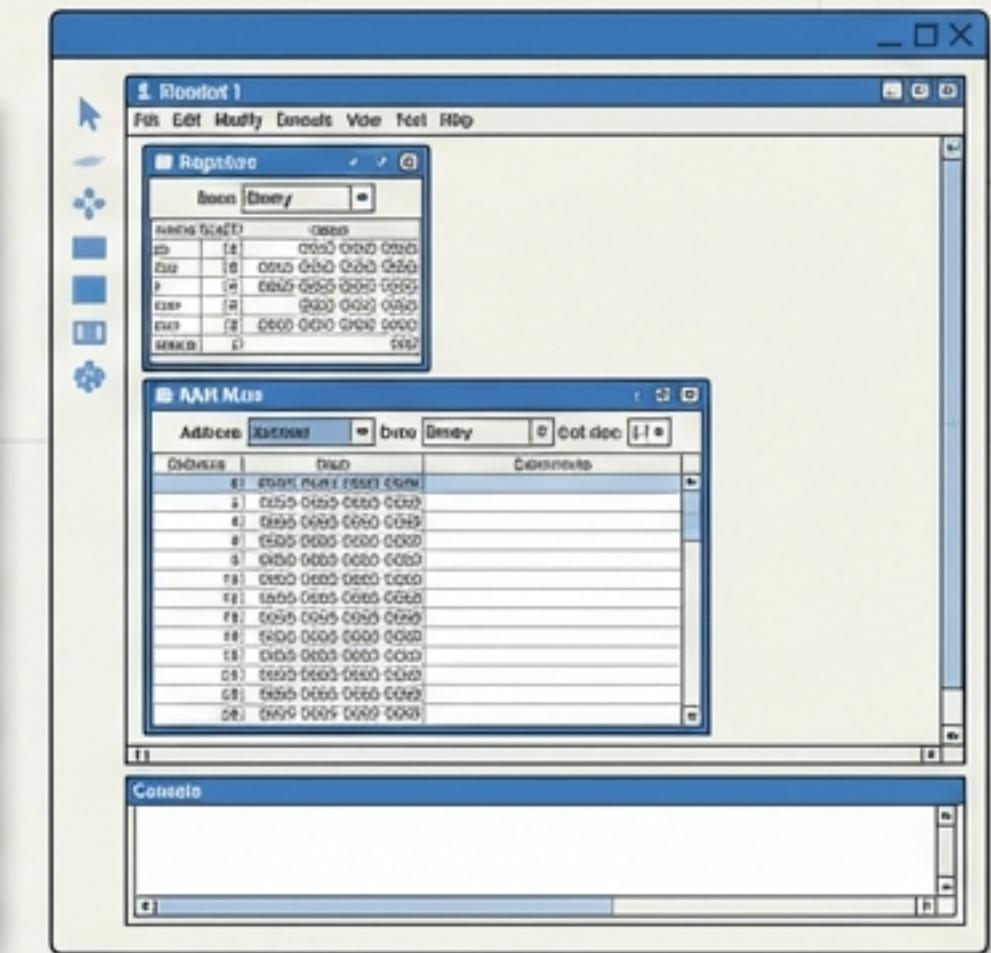
CEDAR Logic



CEDAR Logic

Nivel Transferencia
de Registros (RTL)

CPU Sim



CPU Sim

Nivel Microcódigo
& Ensamblador



La Orquesta Completa: Síntesis del Sistema

