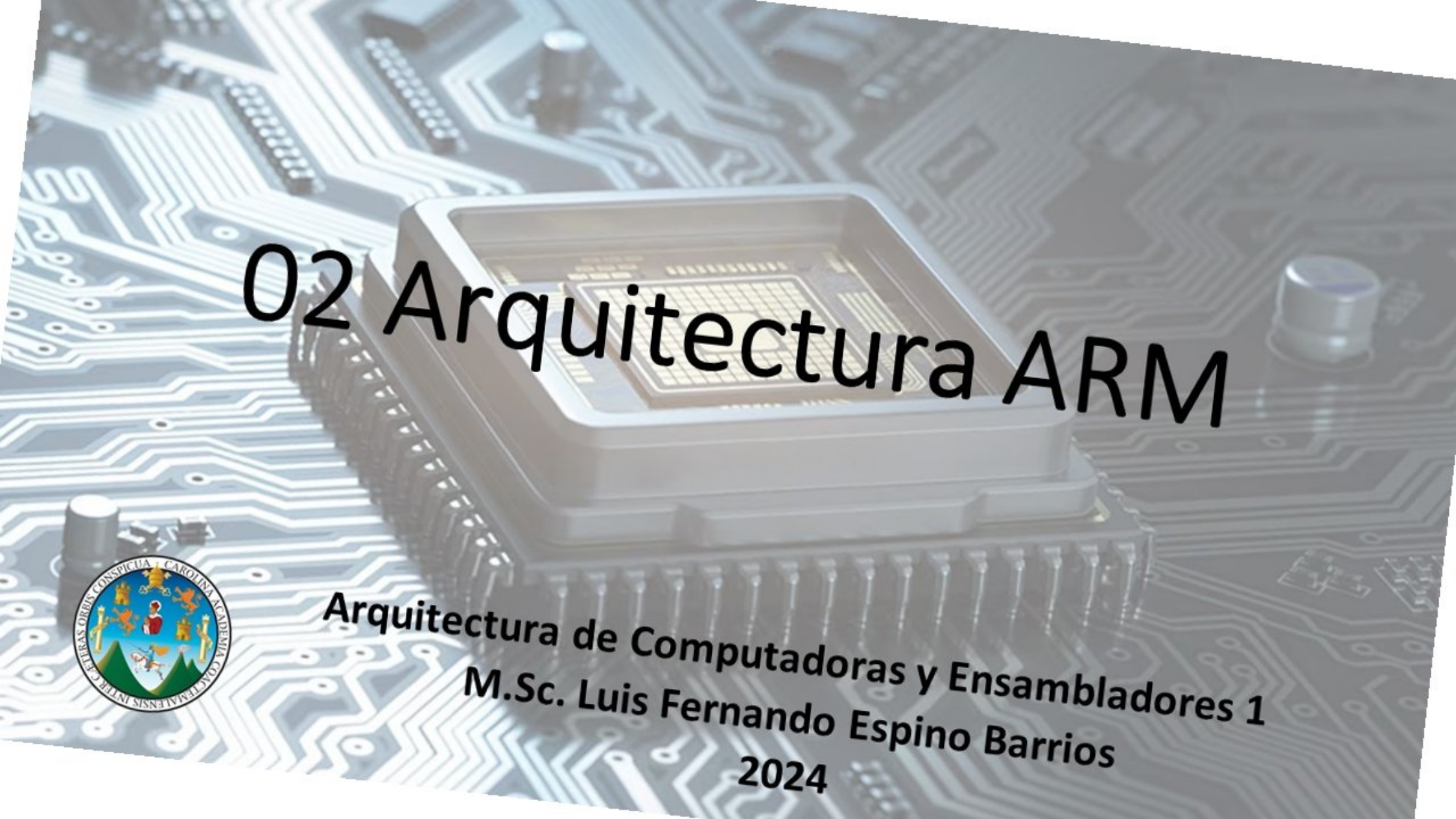


02 Arquitectura ARM

Arquitectura de Computadoras y Ensambladores 1

M.Sc. Luis Fernando Espino Barrios

2024

Arquitectura ARM

- Es una familia de arquitecturas basada en la arquitectura RISC.
- Su nombre es un acrónimo de Advanced RISC Machine, aunque inicialmente la A era por la empresa Acorn.
- La primera versión ARMv1 se lanzó en 1985.
- La última versión ARMv9 se lanzó en 2021.
- De ARM1 a ARM7 son de 32 bits, y la ARM8 y ARM9 son de 64 bits.

ARMv1

ARMv9.2 Snapdragon 8 Gen 4

Arquitectura ARM

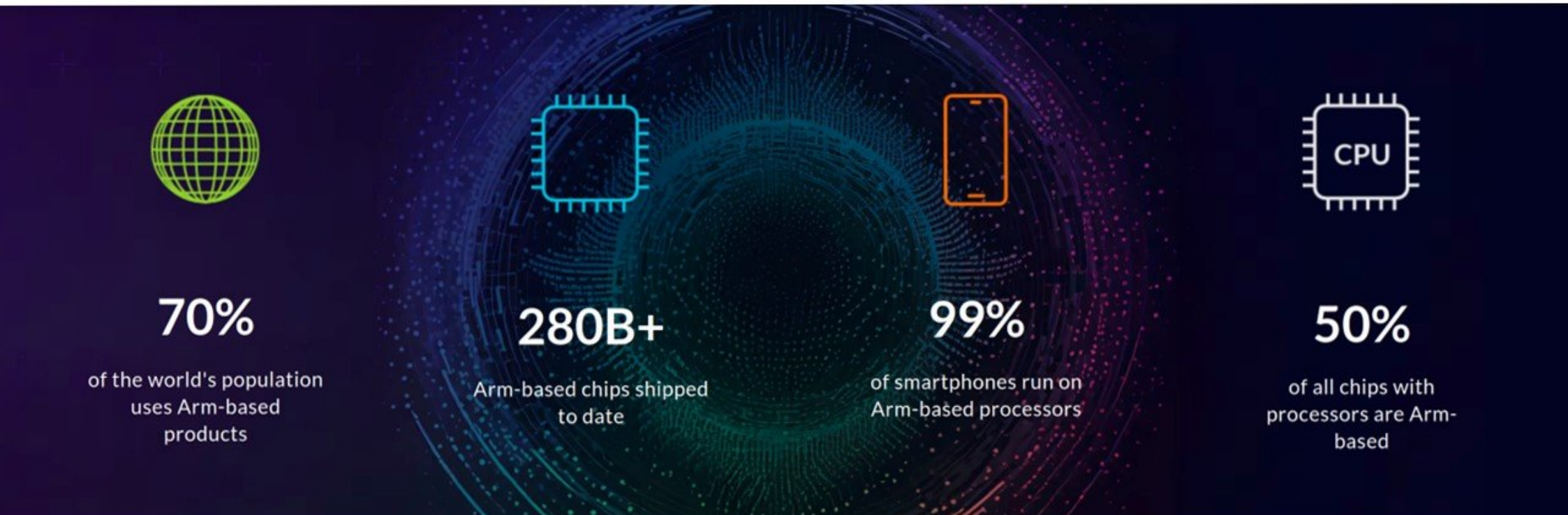
- Dentro de sus características están:
  - El bajo costo,
  - El bajo consumo energético, y
  - La baja generación de calor.

- Estas características han hecho ideal su arquitectura para dispositivos móviles y embebidos.

Arquitectura ARM

- ARM es una arquitectura propietaria, a diferencia de RISC-V que es de código abierto.

- Recientemente MIPS, otra arquitectura RISC que se introdujo en 1985, publicó que el desarrollo de esta arquitectura ha finalizado y que harán una transición a RISC-V.

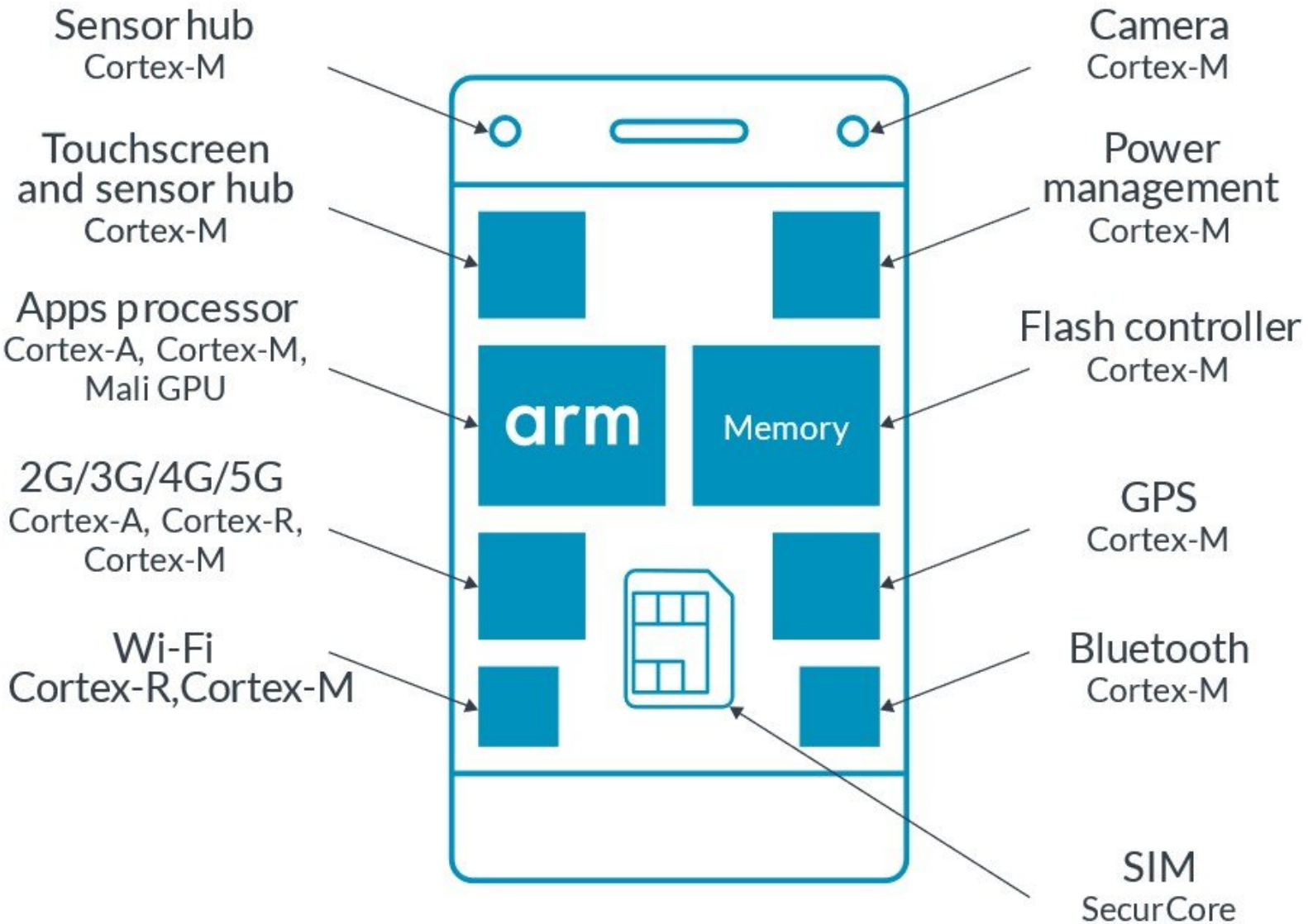

CPU
70%
of the world's population uses Arm-based products
280B+
Arm-based chips shipped to date
99%
of smartphones run on Arm-based processors
50%
of all chips with processors are Arm-based

Componentes de la arquitectura ARM

- Conjunto de instrucciones
- Conjunto de registros
- Modelo de excepción
- Modelo de memoria
- Modelo de debug

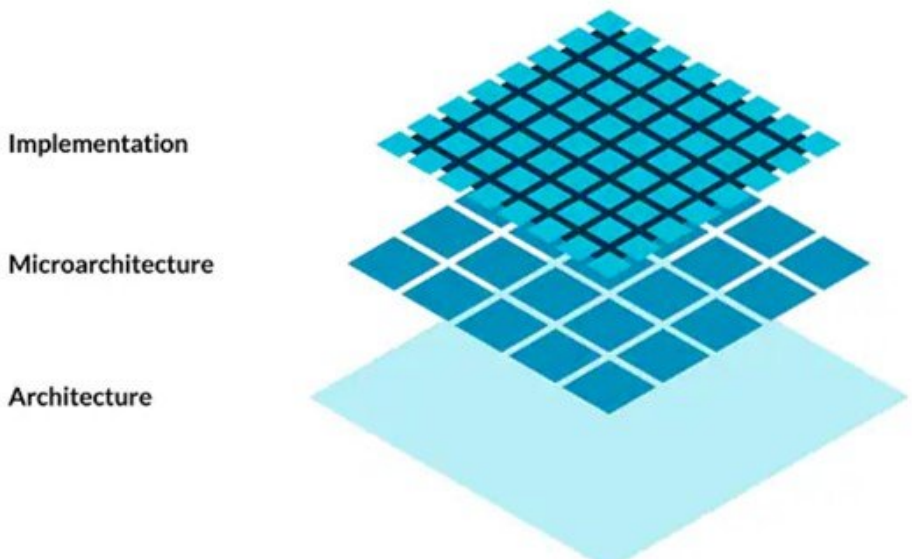
Perfiles de arquitectura ARM

- A (Applications): de alto rendimiento y diseñado para dispositivos con sistemas operativos.
- R (Real-Time): cubre necesidades de tiempo real y diseñado para sistemas embebidos.
- M (Microcontroller): pequeño con alta eficiencia energética y diseñado para IoT

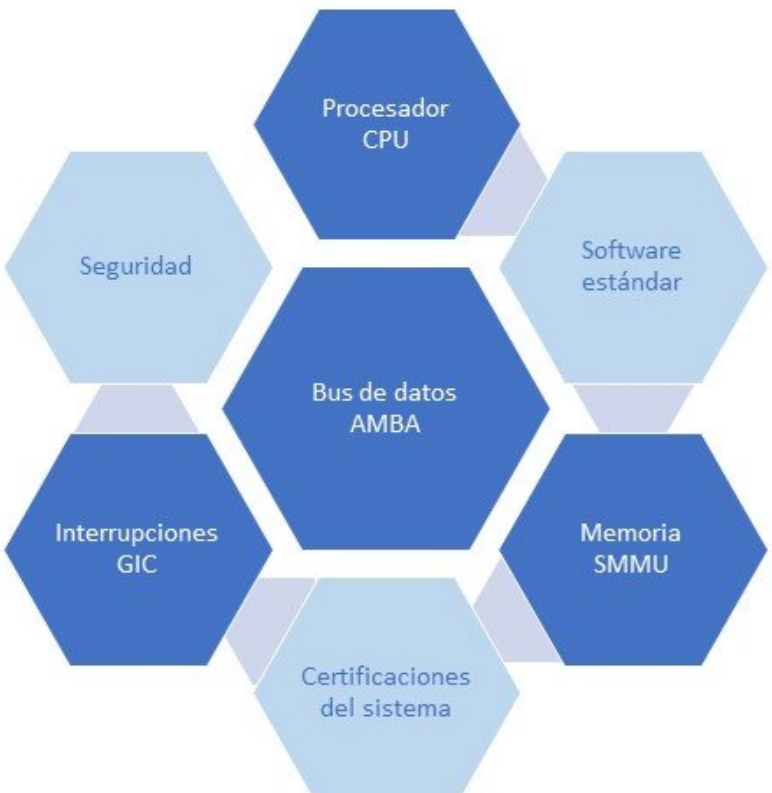

Sensor hub
Cortex-M
Touchscreen and sensor hub
Cortex-M
Apps processor
Cortex-A, Cortex-M, Mali GPU
2G/3G/4G/5G
Cortex-A, Cortex-R, Cortex-M
Wi-Fi
Cortex-R,Cortex-M
Camera
Cortex-M
Power management
Cortex-M
Flash controller
Cortex-M
GPS
Cortex-M
Bluetooth
Cortex-M
SIM
SecurCore
arm
Memory

Arquitectura del CPU

- La arquitectura del CPU de Arm está implementada por microarquitecturas para entregar software compatible según energía y rendimiento.
- La arquitectura del CPU incluye un conjunto de instrucciones y los modelos de excepción y memoria.
- La microarquitectura determina como la implementación se basa en el diseño del procesador.



Arquitectura del sistema



Microprocesadores ARM

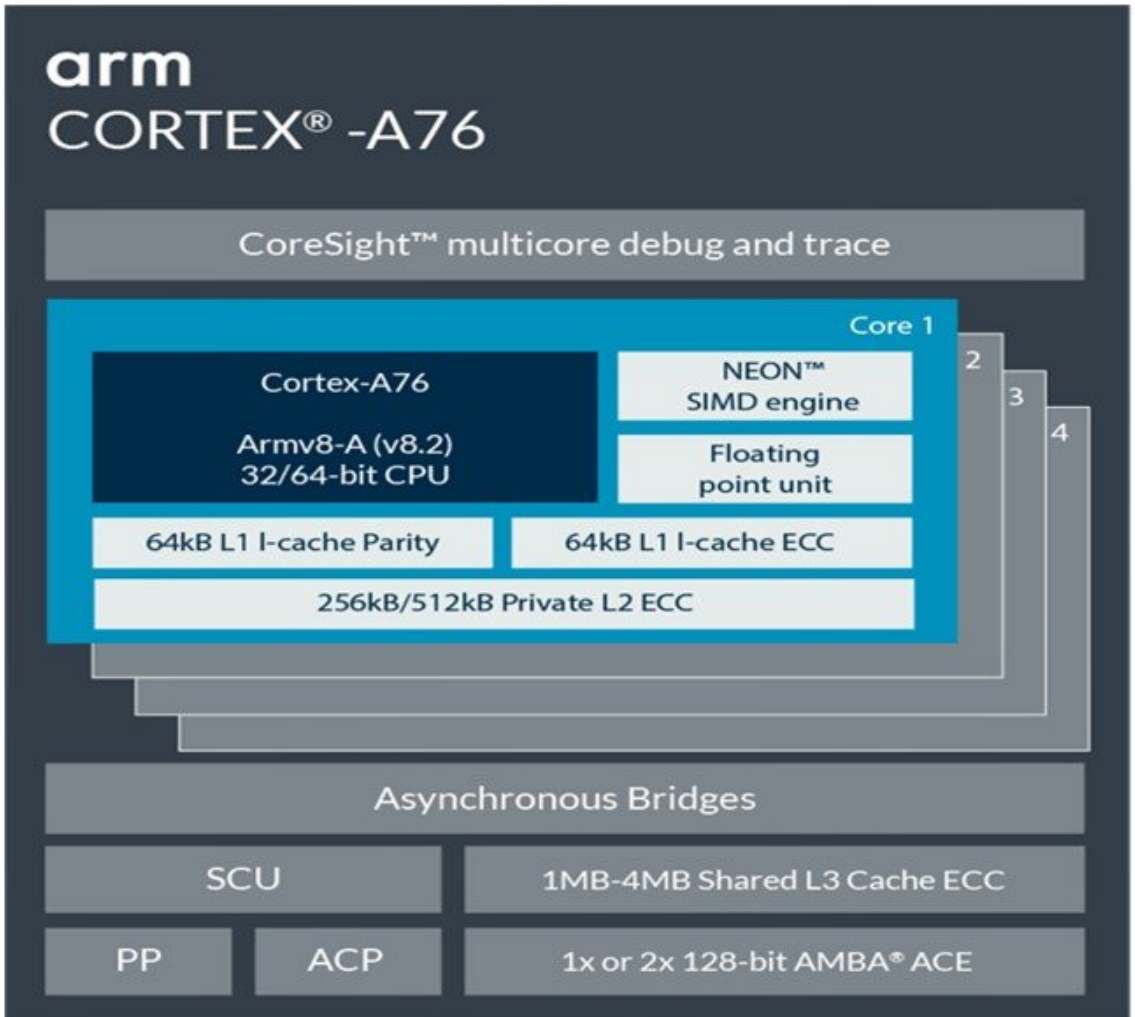
- Según la versión de arquitectura implementada se construye el o los microprocesadores.
- Para ARMv7A corresponden los microprocesadores Cortex-A del 5 a 17

  lanzados en 2011.
- Para Raspberry Pi 2B

  se utiliza el Cortex-A7.

Cortex-A76

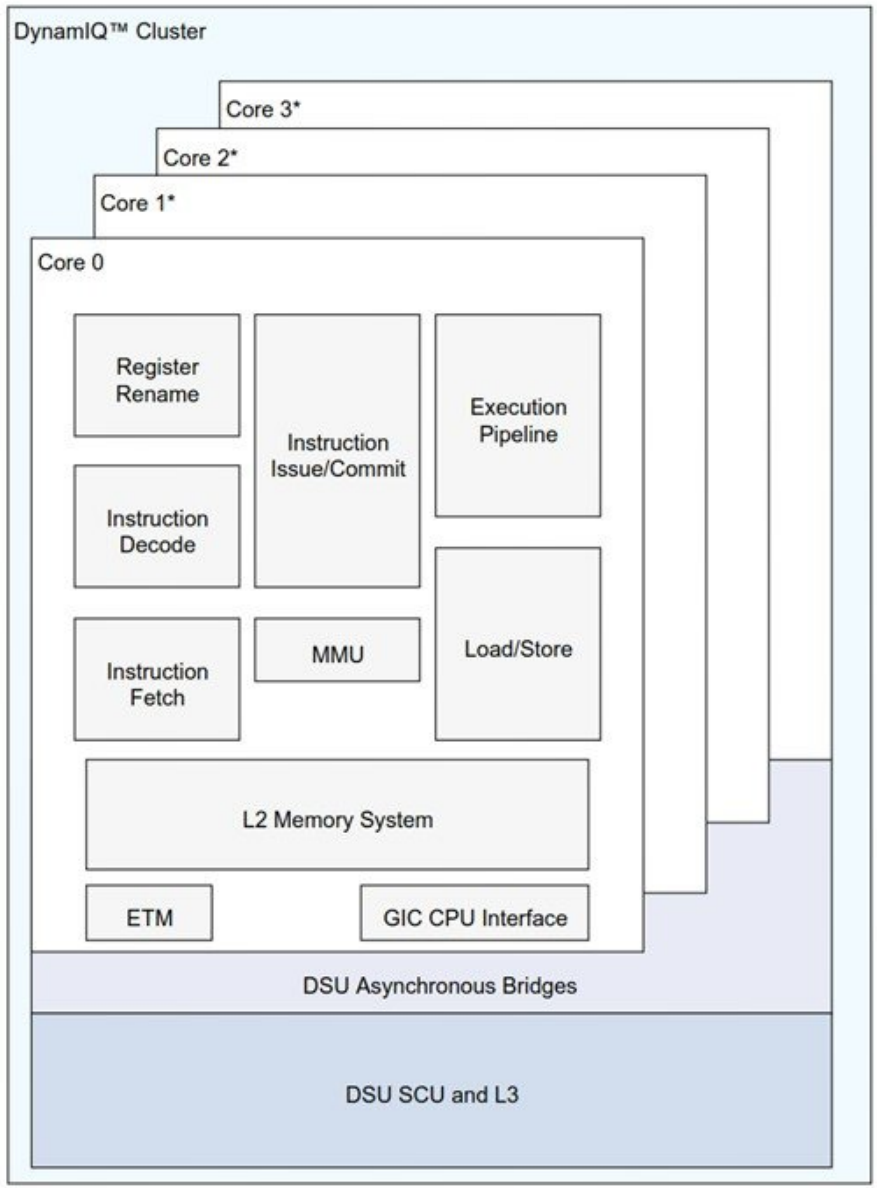
- El Cortex-A76 es un procesador segunda generación de alto rendimiento y bajo consumo .
- Implementa la arquitectura ARMv8-A.
- Normalmente está compuesto por 4 núcleos manejados por la unidad de clúster DSU.
- Tiene dos niveles de memoria caché L1 y L2.


arm
CORTEX® -A76
CoreSight™ multicore debug and trace
Core 1
2
3
4
Cortex-A76
Armv8-A (v8.2)
32/64-bit CPU
NEON™
SIMD engine
Floating
point unit
64kB L1 I-cache Parity
64kB L1 I-cache ECC
256kB/512kB Private L2 ECC
Asynchronous Bridges
SCU
1MB-4MB Shared L3 Cache ECC
PP
ACP
1x or 2x 128-bit AMBA® ACE

Componentes del Cortex-A76

- Fecth de instrucciones
- Decode de instrucciones
- Renombre de registros
- Control de instrucciones
- Pipeline
- Memoria L1
- Memoria L2

The following figure is an overview of the Cortex-A76 core.



Mejoras en el Cortex-A76

- En comparación con su antecesor el A75, ARM asegura que en cuanto a la eficiencia energética mejor en un 40%.
- En cuanto a la mejora en ML/IA en 4 veces;
- En cuanto al rendimiento del procesador en un 35%.

Arquitectura ARMv8-A

- Es la arquitectura ARM que soporta 64 bits AARCH64 y utiliza el perfil de aplicaciones.
- Se basa en los conceptos del estado de ejecución, en el conjunto de instrucciones, en los registros y en el depurador.
- A diferencia de los otros perfiles maneja memoria virtual.
- Maneja el conjunto de instrucciones llamado A64.

Tipos de datos

- Maneja cinco tipos de datos:
  - byte de 8 bits,
  - halfword de 16 bits,
  - word de 32 bits,
  - doubleword de 64 bits y
  - quadword de 128 bits.
- En cuanto a la precisión de punto flotante maneja de precisión a la mitad, precisión simple y precisión doble.

Registros de la unidad entera

- 31 registros de propósito general, Rn, que van del R0 al R30. Pueden ser accedidos según el tamaño requerido de 32 bit con Wn; o de 64 bits con Xn.
- Un registro de 64 bits, SP, dedicado al Stack Pointer. El segmento menos significativo de 32 bits de dicho registro se puede acceder mediante WSP.
- Un registro de 64 bits, PC, dedicado al Program Counter que mantiene la dirección de la actual instrucción.

Registros de la unidad vectorial

- La arquitectura en la unidad vectorial tiene 32 registros para punto flotante, Vn, que van del V0 al V31. Pueden ser accedidos según el tamaño requerido:
  - Q0 a Q31, de 128 bits.
  - D0 a D31, de 64 bits.
  - S0 a S31, de 32 bits.
  - H0 a H31, de 16 bits.
  - B0 a B31, de 8 bits.

Modos de excepción

- Conocido antes como modos de operación, en esta versión se redujo a 4 niveles de excepción:
  - EL0: modo usuario.
  - EL1: modo kernel.
  - EL2: modo hipervisor.
  - EL3: monitor de seguridad.

Estados del proceso

- Es una abstracción de la información del estado del proceso.
- Maneja las banderas de condición, los bits de máscara de excepciones y los controles de habilitación de modo SME.

Banderas

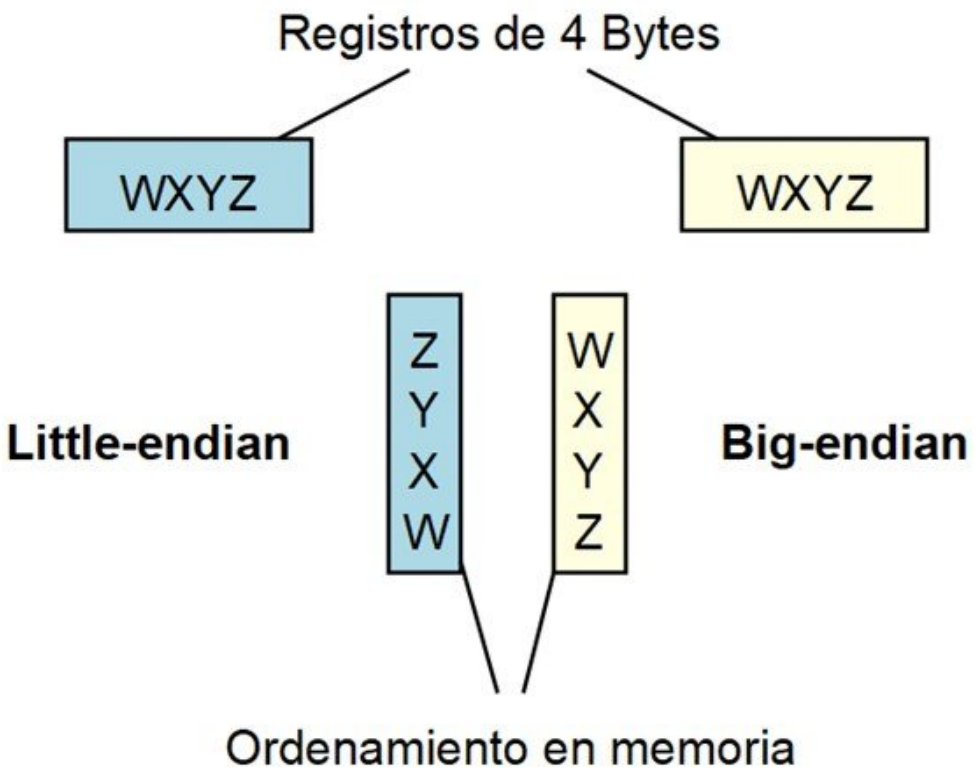
- N: bandera de condición negativa, si el resultado de una instrucción es el complemento a 2, entonces tendrá 1 si el resultado es negativo y 0 si el resultado es positivo o cero.
- Z: bandera de condición cero, tendrá el valor 1 si el resultado de una instrucción es cero y 0 si el resultado es diferente a cero.
- C: bandera de condición de corrimiento, tendrá el valor 1 si la instrucción resulta en un corrimiento y 0 si el resultado no lo tiene.
- V: bandera de condición de desborde, tendrá el valor 1 si la condición se desbordó y 0 si no lo hizo.

Bit de máscara de escepción

- Están relacionados con los modos de operación, los cuales tendrán un valor de 1 si hay excepción o 0 si no la hay, entre estos bits se tienen:
  - D: es el bit de máscara de excepción de depuración.
  - A: es el bit de máscara de interrupción por error del sistema.
  - I: es el bit de máscara de interrupción por IRQ.
  - F: es el bit de máscara de interrupción por FIQ.

Endianness

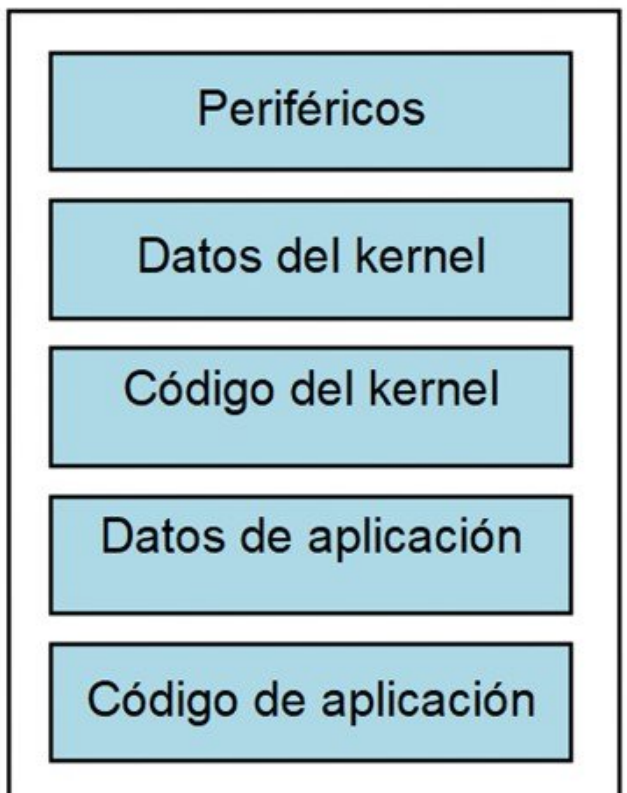
- Es el orden en que los bytes de un word o dobleword son transmitidos por una canal o accedidos mediante una dirección.
- Se divide en big-endian y little-endian, nombres inspirados de la novela de los viajes de Gulliver.
- ARM utiliza tanto en instrucciones como en direcciones little-endian, aunque se puede trabajar con big-endian.


Registros de 4 Bytes
WXYZ
WXYZ
Little-endian
Z
Y
X
W
W
X
Y
Z
Big-endian
Ordenamiento en memoria

Modelo de memoria

- Es la forma en que se organiza y define cómo se comporta la memoria.
- Provee la estructura y conjunto de reglas para el direccionamiento, regiones.
- Aunque en la documentación técnica hay detalle acerca del mapeo de la memoria incluso a nivel de rango de direcciones, este varía dependiendo del dispositivo a utilizar y la cantidad de memoria que tenga el dispositivo.

A manera general se dispone de un sistema simple de regiones de la memoria (la direcciones de memoria inician en la parte inferior y finalizan en la parte superior):



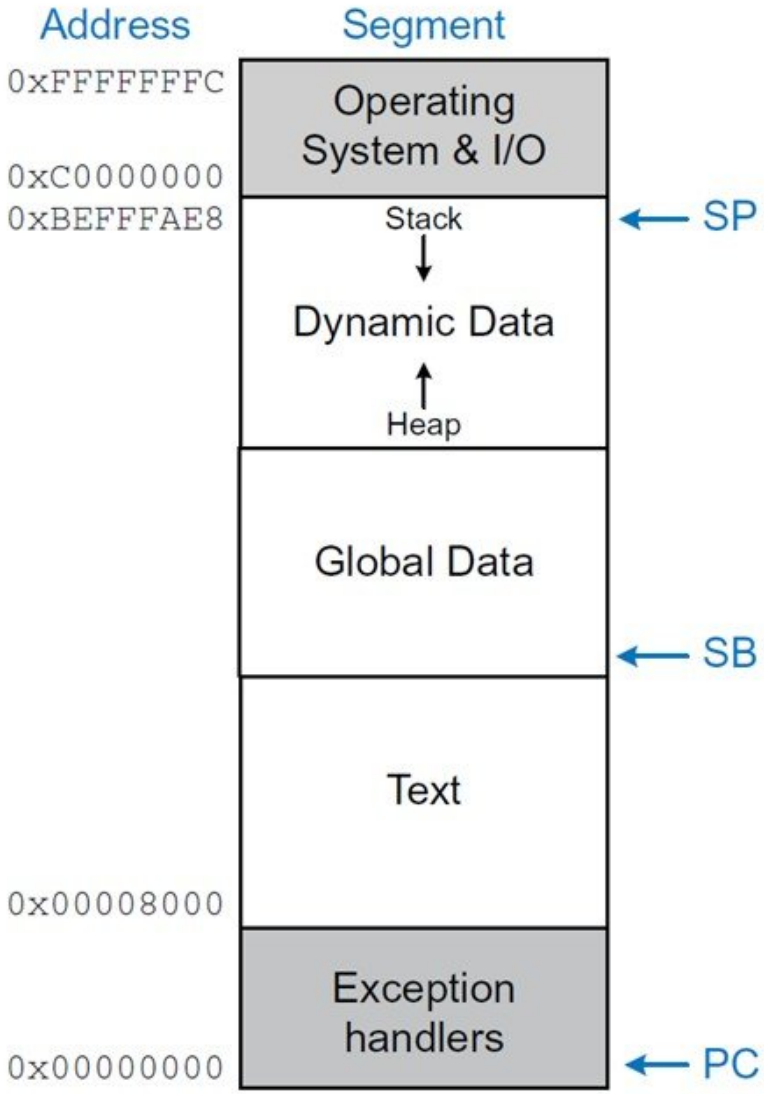


Figure 6.30 Example ARM memory map

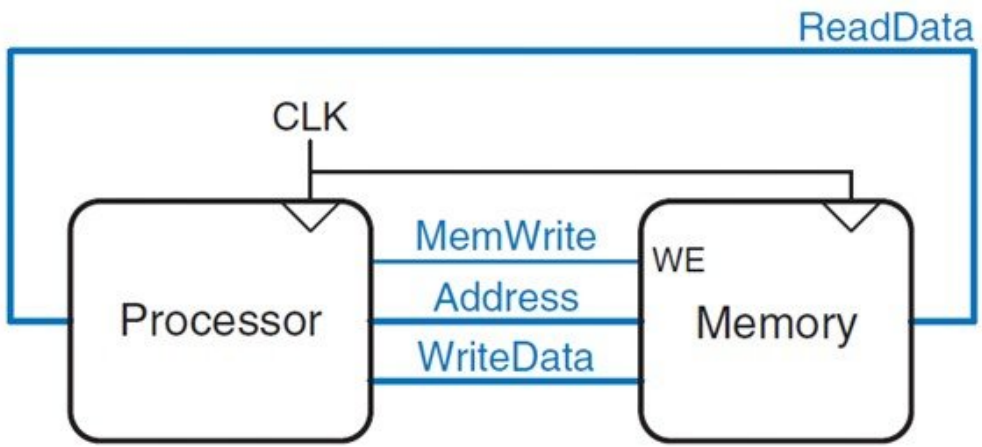


Figure 8.1 The memory interface

Memoria en ARM

- También se utiliza el direccionamiento del byte más significante (MSB) y menos significante (LSB).

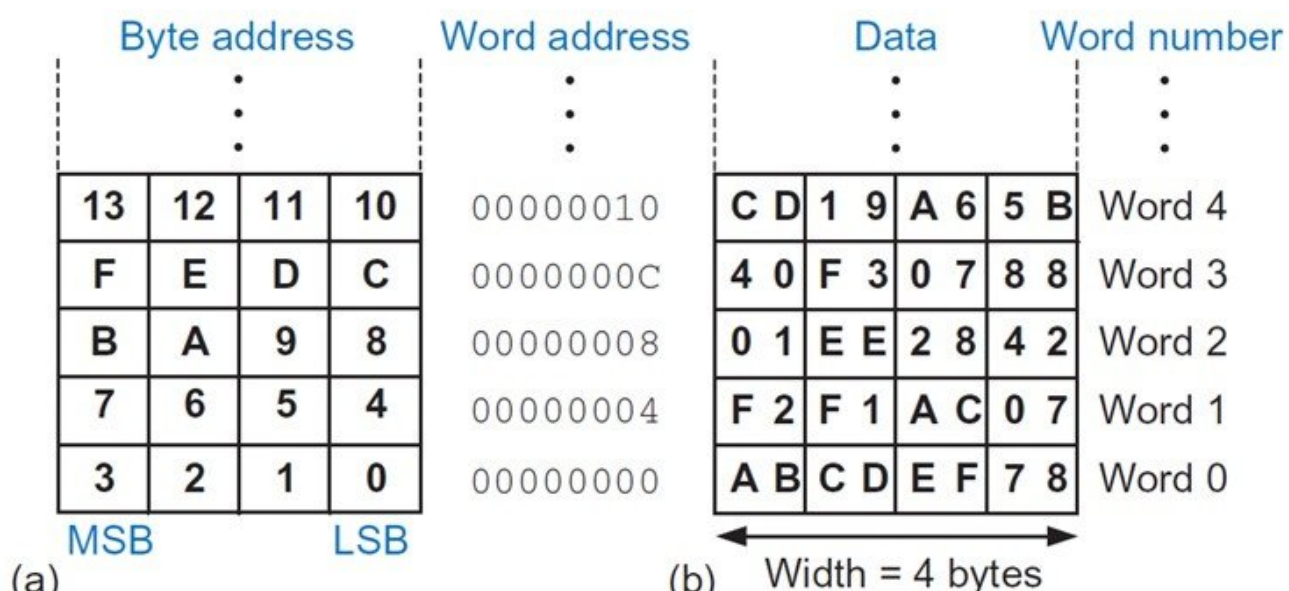


Figure 6.1 **ARM byte-addressable memory showing: (a) byte address and (b) data**

Modos de direccionamiento de memoria

- Para el acceso de memoria están permitidos los siguientes modos de direccionamiento:
  - Direccionamiento con offset: es un direccionamiento con corrimiento sin alterar el registro base. Su sintaxis es: [Rn, offset]
  - Direccionamiento pre-indexed: es un direccionamiento con corrimiento alterando antes el registro. Su sintaxis es: [Rn, offset]!
  - Direccionamiento post-index: es un direccionamiento con corrimiento alterando después el registro. Su sintaxis es: [Rn], offset
- El offset puede ser una constante inmediata, un registro, o un registro desplazado.

Conjunto de instrucciones A64

- Procesamiento de datos con inmediatos (aritméticos, lógicos y desplazamiento)
- Procesamiento de datos con registros (aritméticos, lógicos y desplazamiento)
- Carga/almacenamiento e intercambio
- Ramificación incondicional y condicional
- Procesamiento de datos con números punto flotante
- Generación de excepciones, retorno y depuración
- Procesamiento de datos con vectores
- Procesamiento de datos con matrices

Microarquitecturas

- Se utilizan para conectar la lógica con la arquitectura.

- En ARM hay tres microarquitecturas:
  - Single-cycle: ejecuta una instrucción en un ciclo.
  - Multicycle: ejecuta instrucciones en varios ciclos.
  - Pipelined: ejecuta instrucciones de manera simultánea.

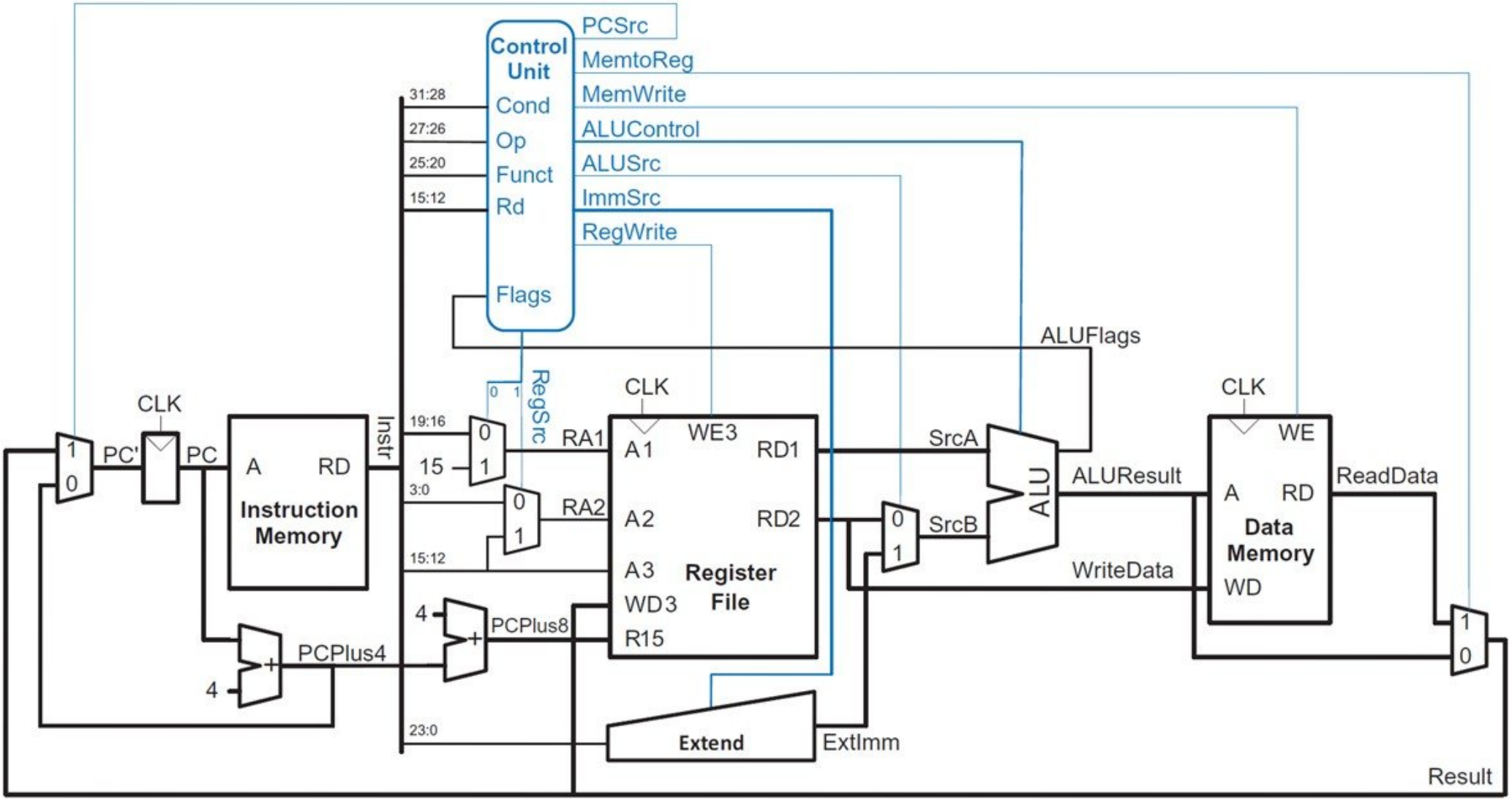


Figure 7.13 Complete single-cycle processor

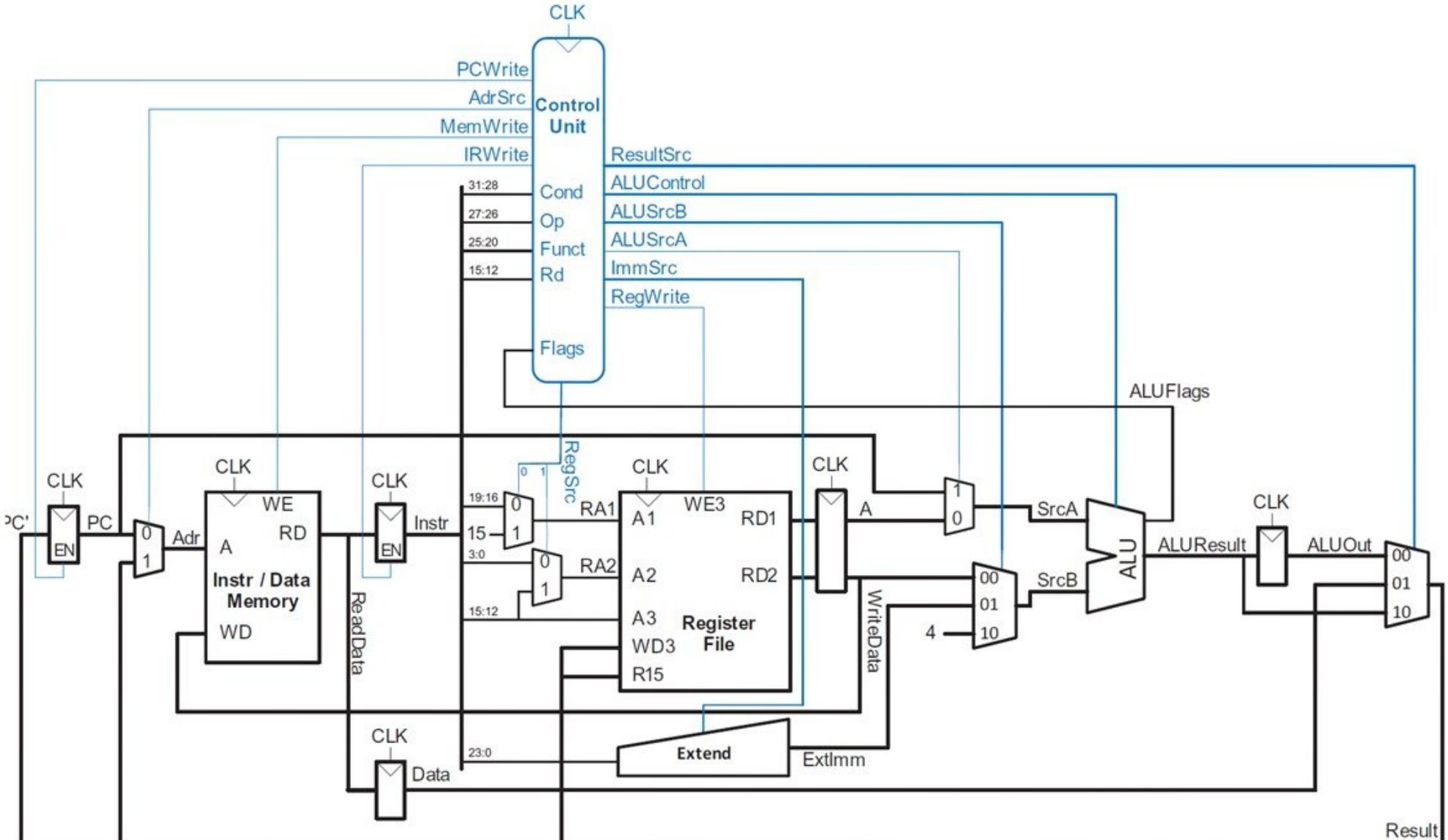


Figure 7.30 Complete multicycle processor

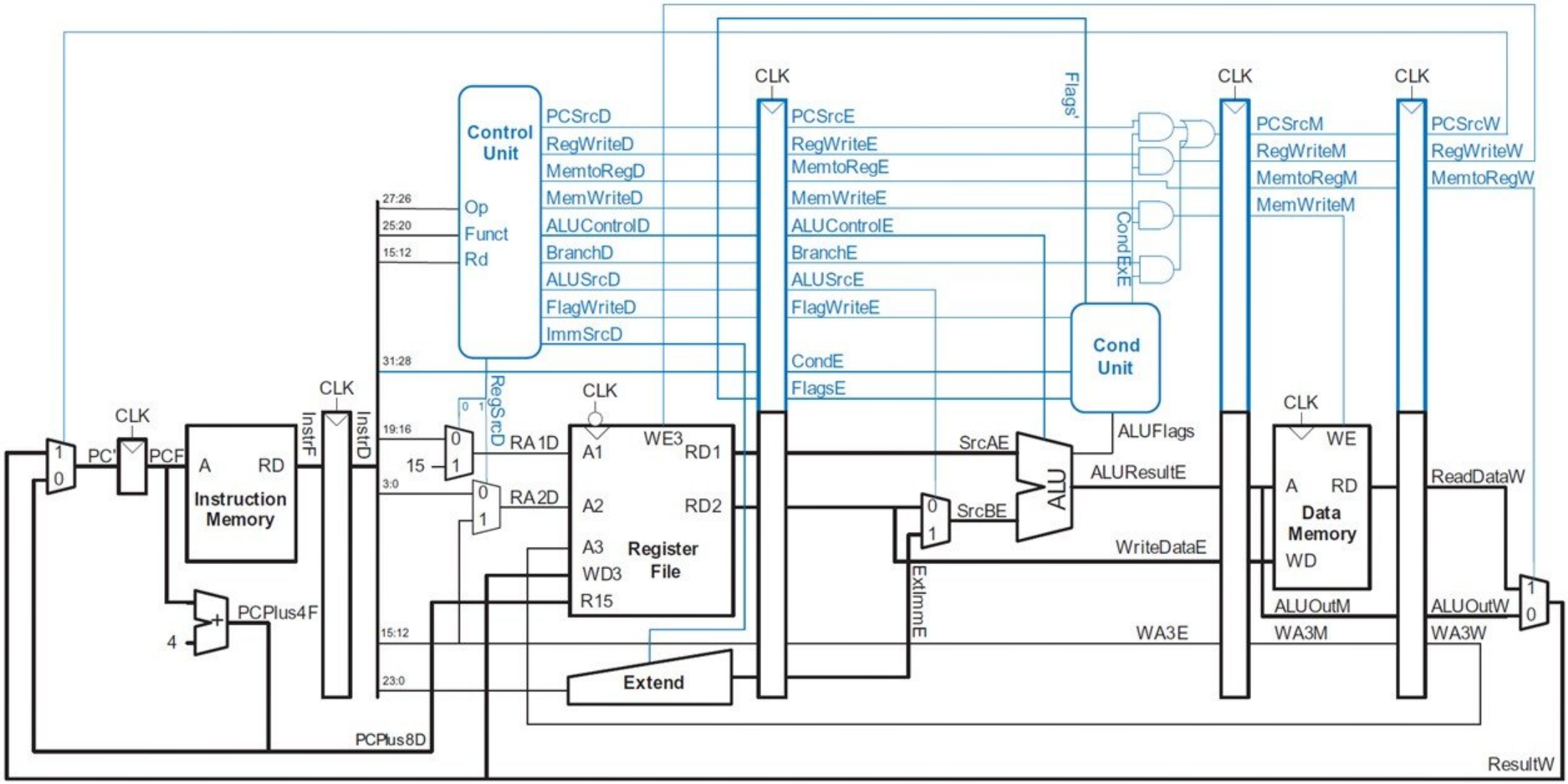


Figure 7.47 Pipelined processor with control

Bibliografía

Elahi, A. (2022). Fundamentals of Computer Architecture and ARM Assembly Language. Springer.

Harris, S. & Harris, D. (2016). Digital Design and Computer Architecture: ARM Edition. Elsiever Inc.

Mazidi, M. & Otros. (2013). ARM Assembly Language: Programming & Architecture.

Smith, S. (2019). Raspberry Pi Assembly Language Programming: ARM Processor Coding. Apress.