



**UNIVERSIDAD TECNOLÓGICA NACIONAL**  
INSTITUTO NACIONAL SUPERIOR DEL PROFESORADO TÉCNICO

# Procesadores CISC - RISC

Prof. : Juan .C. Capia

Sistemas de Computación  
Comisión 2.602

# Procesadores: Arquitectura de Set de Instrucciones

- 1.Arquitectura x86 vs Sistemas Embebidos vs Arquitectura ARM
- 2.Arquitectura del conjunto (Set) de instrucciones del procesador CISC y RISC -- x86 y ARM
  - 1.Set de instrucciones funciones y características
  - 2.X86 vs ARM: tipos de datos
  - 3.X86 vs ARM: tipos de operaciones
  - 4.X86 vs ARM: modos de direccionamiento
  - 5.Little Endian vs Big Endian
- 3.RISC vs CISC – Conclusiones
- 4.Procesadores - Conclusiones generales

# Procesadores CISC y RISC

# Comparativas arquitecturas

Arquitectura X86	Sistemas embebidos	Arquitectura ARM
<p>Arquitectura con conjuntos de instrucciones complejas (CISC) Ha evolucionado para seguir compatible con las versiones anteriores Todos los cambios han implicado adicciones Al conjunto de instrucciones sin sustracciones El 8086 de 1978 se introdujo con una velocidad de reloj de 5 MHz y tenía 29000 transistores un Core i7 Edge 4960 x de 6 núcleos a 4 GHz con una velocidad de 800 y tiene 1860 millones de transistores</p>	<p>Electrónica y Software dentro de un producto en contraposición a una computadora de uso general Entre los ejemplos figuran los teléfonos celulares las cámaras digitales las videocámaras las calculadoras los hornos microondas los sistemas de seguridad para el hogar las lavadoras los sistemas de iluminación los termostatos las impresoras diversos sistemas en los automóviles las raquetas de tenis los cepillos de dientes y numerosos tipos de sensores y actuadores en sistemas automatizados Otro ejemplo claro es el uso en internet de las cosas (IoT)</p>	<p>Su nombre Acorn RISC Machine (ARM)Es una familia de microprocesadores y microcontroladores basados en conjunto de instrucciones reducidas (RISC) Los chips ARM son procesadores de alta velocidad conocidos por su pequeño tamaño de matriz y sus bajos requerimientos de energía Los chips ARM son los procesadores de los populares dispositivos iPod o iPhone de Apple, y también se utilizan en casi todos los teléfonos inteligentes AndroidIncluso en placas de video cómo Nvidia</p>

## 2 .Arquitectura del Set de Instrucciones de: CISC y RISC < x86 y ARM

Sus partes

# Procesador: Partes internas

A la hora de Definir la arquitectura de set de instrucciones debemos preguntarnos:

¿Qué tipos de datos pueden manejar "nativamente"?

¿Cómo se almacenan?

¿Cómo se acceden?

¿Qué operaciones (instrucciones) puede ejecutar?

¿Cómo se codifican estas operaciones?

La respuesta a estas preguntas se llama Instruction Set Architecture también conocida como ISA

Cuando estamos estudiando una ISA, interesa entender su comportamiento

# Procesador: Partes Internas

Cantidad total de instrucciones disponibles

Arquitectura del conjunto set de instrucciones

RISC: Reduced Instruction Set Computer

CISC: Complex Instruction Set Computer

Longitud de las instrucciones ("ancho")

Cantidad de memoria que un programa requiere ("largo x ancho")

Tipos de datos

Entero (8 16 32 bis; complemento a 2)

Punto flotante. Punto fijo

¿BCD, ASCII?

Almacenamiento: Big Endian y Little Endian

# X86 vs ARM: Tipos de datos



# Tipos de datos

ARQUITECTURA X86	ARQUITECTURA ARM
El x86 puede manejar tipos de datos de 8 (byte), 16 (palabra), 32 (palabra doble), 64 (palabra cuádruple) y 128 (palabra doble cuádruple) bits de longitud	Los procesadores ARM soportan tipos de datos de 8 (byte), 16 (media palabra) y 32 (palabra) bits de longitud
Enteros con signo: Complemento a 2 de 16, 32 o 64 bits	Enteros con signo: Complemento a 2 de 16, 32 y 64 bits
Punto flotante: IEEE754	Punto flotante: IEEE754 no todos los procesadores ARM tienen soporte para ahorrar energía. En dichos procesadores se debe implementar por software
Orden de almacenamiento de datos: Little Endian	Orden de almacenamiento de datos: Little y Big Endian puede seleccionarse mediante configuración independientemente del sistema operativo

# Instrucciones

¿Cómo son las instrucciones en bajo nivel?

Secuencia de bits divididas en campos

Para no usar secuencias de bits se utiliza una representación simbólica

CODIGOOPERACION	REFERENCIA A OPERANDO	REFERENCIA A OPERANDO
-----------------	-----------------------	-----------------------

Los operandos pueden ser:

- Memoria principal Registros
- Inmediato
- Entrada / Salida

El número del código de operaciones (CODOP) varía mucho en una arquitectura a la otra, Pero poseen los mismos tipos generales. Las operaciones son: transferencia de datos, aritmética, lógico, conversión, Entrada / Salida, control del sistema y transferencia de control

# Tipos de operaciones

ARQUITECTURA X86	ARQUITECTURA ARM
Instrucciones de longitud variable	Instrucciones de longitud fija
Instrucciones de procesamiento de datos: instrucciones lógicas, suma, resta, test y comparadores	Instrucciones Load / Store: solo posee estas dos para acceder a memoria.
Instrucciones CALL / RETURN	Instrucciones aritméticas y lógicas solo sobre registros
Instrucciones de manejo de memoria: utilizadas solamente por el S.O	Instrucciones de Multiplicación
Flags de Estados y Códigos de condicionales	Instrucciones de Sumas y Restas en paralelo
Instrucciones SIMD: Instrucciones MMX para tareas multimedia. Permite ejecutar la misma instrucción sobre múltiples datos	Instrucciones de acceso a Registros a Estados
	Códigos de Condicionales

# Tipos de operaciones

ARQUITECTURA X86	ARQUITECTURA ARM
<p>Imaginemos que para este ejemplo tenemos la instrucción "MULT". Cuando se ejecuta la instrucción está carga los dos operandos en registros separados La ejecución de la instrucción concluye con solo la siguiente instrucción:</p> <pre>MOV AX, DS: [0052h] MULT DS :[0023h], AX</pre> <p>MULT es una "instrucción compleja" que opera directamente en la memoria y no requiere indicaciones explícitas por el programador para realizar la carga de los datos es un lenguaje de alto nivel seria: <math>A = A * B</math> .</p> <p>Una ventaja es que el compilador tiene poco trabajo que hacer al traducir la instrucción a un lenguaje de alto nivel</p>	<p>El código apropiado para esta ejecución es:</p> <pre>LOAD AX, DS:[0023h] LOAD BX, DS:[0052h] PROD AX, BX STORE DS:[0023h], AX</pre> <p>Parece menos eficiente ya que requiere de más instrucciones para ejecutarse, más memoria para almacenarse y el compilador deberá trabajar más para realizar la traducción a partir de un lenguaje de alto nivel</p> <p>Sin embargo, tiene ventajas al ejecutar cada instrucción en un ciclo de reloj. En tiempo de ejecución sería muy similar al de CISC</p> <p>Requiere de menos transistores en el Hardware que las instrucciones complejas haciendo más barata su construcción</p>

En esta sección se examinan las técnicas o modos de direccionamiento más comunes:

Inmediato,  
Directo,  
Indirecto,  
por Registros,  
por Desplazamiento o Pila

# Tipos de Direccionamiento

ARQUITECTURA X86	AQUITECTURA ARM
<p>Existen 6 registros de segmento .</p> <p>Cada registro de segmento contiene la dirección de inicio de los segmentos correspondientes. Además, hay dos registros que pueden utilizarse para construir una dirección: el registro de base y el registro de índice .</p> <p>Dónde la dirección se conforma como Segmento + Desplazamiento.</p> <p>El resto de los modos de direccionamiento hacen referencia a las ubicaciones en la memoria. La ubicación de la memoria debe especificarse en términos del segmento que contiene la ubicación y el desplazamiento desde el principio del segmento.</p>	<p>Utiliza un conjunto simple y relativamente sencillo de modos de direccionamiento.</p> <p>Estos modos se califican de la forma más conveniente con respecto al tipo de instrucción .</p> <p>Direccionamiento por LOAD / STORE: las instrucciones de carga y almacenamiento son las únicas instrucciones que hacen referencia a la memoria. Indirectamente a través de un registro base + desplazamiento .</p> <p>Direccionamiento de instrucciones de procesamiento de datos .</p> <p>Instrucciones de bifurcaciones.</p> <p>Direccionamiento múltiple de LOAD / STORE</p>

# Little Endian vs Big Endian



# Tipos de almacenamiento

LITTLE ENDIAN	BIG ENDIAN
<p>El byte es una posición de memoria menor, es menos significativo. En general, Intel y computadoras CISC</p> <p>Ejemplo: ¿Como se almacena 0x1234 (16bits) en 8 bits?</p> <p>M[0] = 0 x 34 M[1] = 0 x 12</p>	<p>EL byte es una posición de memoria mayor, es el más significativo. En general, Motorola, Computadoras RISC y redes.</p> <p>Ejemplo: ¿Como se almacena 0 x 1234 (16bits) en 8 bits?</p> <p>M[0] = 0 x 12 M[1] = 0 x 34</p>

# RISC vs CISC - Conclusiones

¿Podemos compararlas?

# Conclusiones

La evolución de los procesadores CISC iba hacia la incorporación de cada vez más instrucciones, más modos de direccionamiento, registros más especializados, etc.

La aparición de RISC y el auge que ha tenido con la aparición de ARM llevo a intento de comprar el tamaño de los programas y la velocidad de ejecución de los programas

Hay varios problemas al intentar tales comparaciones, ya que no hay un par de máquinas RISC y CISC comparables en cuanto:

- Al costo del ciclo de vida

- Nivel de tecnología

- Complejidad de las compuertas

- Sofisticación del compilador

- Soporte del sistema operativo, etc.

# Conclusiones

**No existe un grupo de programas de prueba definitivo**

**El rendimiento varía con el programa. Es difícil separar los efectos de Hardware de los efectos debido a la habilidad en la escritura del compilador**

**En los últimos años, la controversia entre el RISC y el CISC ha disminuido en gran medida. Debido a una convergencia gradual de las tecnologías**

**A medida que la densidad de los chips y la velocidad de los equipos en bruto aumentan, los sistemas RISC se han vuelto más complejos**

**Al mismo tiempo, es una fuerza por exprimir al máximo el rendimiento, los diseños de CISC sean centrado en cuestiones tradicionalmente asociadas con RISC, como un mayor número de registros de uso general y un mayor énfasis en el diseño de tuberías de instrucción**

# Conclusiones-Procesadores

Hasta el momento en el curso hablamos de dos tipos de Arquitecturas:

Arquitectura de la computadora:

Von Neumann

Harvard

Arquitectura de Set de Instrucciones del Procesador:

CISC

RISC

Es común que esto genere confusión, Una es como está conformada internamente toda la computadora y la segunda refiere a como está conformado internamente el procesador...

Una arquitectura de computadora puede tener un procesador con cualquier arquitectura de set de instrucciones.

A continuación, vemos algunos ejemplos...

# Intel Core i9 – 9 Generación

Cache :16 MB

Cantidad de núcleos de CPU 8

Tipos de memoria RAM soportados  
DDR4

Tamaño de memoria RAM soportada  
:128gb

Frecuencia mínima de Reloj :36 GHz

Frecuencia Máxima de reloj 5Ghz

Procesador grafico :Intel UHD Graphics  
360

Cantidad de hilos de CPU:16

Año de lanzamiento 2018 .



# Arduino UNO

- Microcontrolador :Atmel ATmega329p
- Flash MEMORY 32KB
- SPRAM 2 KB ( AT mega 328P)
- EEPROM 1KB ( AT mega 328P)
- Clock speed 16 Mhz
- Operating Voltage : 5v
- Digital I/O Pins 14
- PW Digital i/o Pins 6
- Analog Input Pins 6
- Weight 25g



# Raspeberry Pi 4

- Broadcom BCM2711 .Quad core Cortex-AR72 (ARMV8) 64 BIT Soc @1.5Hz
- 2GB ,4GB OR 8GB LPDDR4 3200 SDRAM 8depending on model
- 2.4 Ghz and 5.0 Ghz IEEE 802.11 ac Wireless Bluetooth 5.0 BLE
- Gigabit Ethernet
- Rasperry Pi standart 40 pin GPIO header
- 2xmicro HDMI ports (up to 4kp60 supported)
- OpenGL ES 3.0 graphics.





# Jetson Nano

- GPU :128 –CORE Nvidia Maxwell arquitectura –based GPU
- CPU:Quad –core ARM A57
- Video :4k @30 fps(H264. h265) encode and decode
- Camera MIPS CSI-2 DPHY 12X and x1
- Memory 4gb 64 bits LPDDR4 25.6 25 GIGABITS POR SECOND
- OS Support LINUX for TEGRA
- Module size 70mm x 45mm
- Developer Kits :100mm x 80mm



# iPhone 11

Pantalla: 5.88", 1125 x 2436 pixeles

Procesador: Apple A Bionic (ARM v8)

RAM: 4GB

Almacenamiento: 64GB // 256GB // 512GB

Expansión: sin microSD

Camara: Triple, 12MP + 12MP

Bateria: 3190 mAh

OS: iOS 13

Perfil: 8.1 mm

Peso: 188g



# Motorola Edge

Procesador: Qualcomm Snapdragon 765 → kyro 475Prime (ARM Cortex – A76) a 2.3 Ghz  
GPU: Adreno 620  
Memoria RAM: 6 GB  
Almacenamiento: 128 GB expandible con microSD  
Sistema Operativo: Android 10 con My UX de Motorola  
Principal: 64 mpx f/1.8 y Frontal 25 mpx f/2.0  
Bateria: 4,500 mAh con carga rápida de 18W  
Dimensiones y peso: 161.64 x71.1 x 9.29 mm y 188 gramos  
Conectividad: USB – C, 5G NR Sub – 6GHz, 4G LTE, Bluetooth 5.1, NFC, WIFI 6, GPS

