

Clase 10 - Arquitectura de Computadores

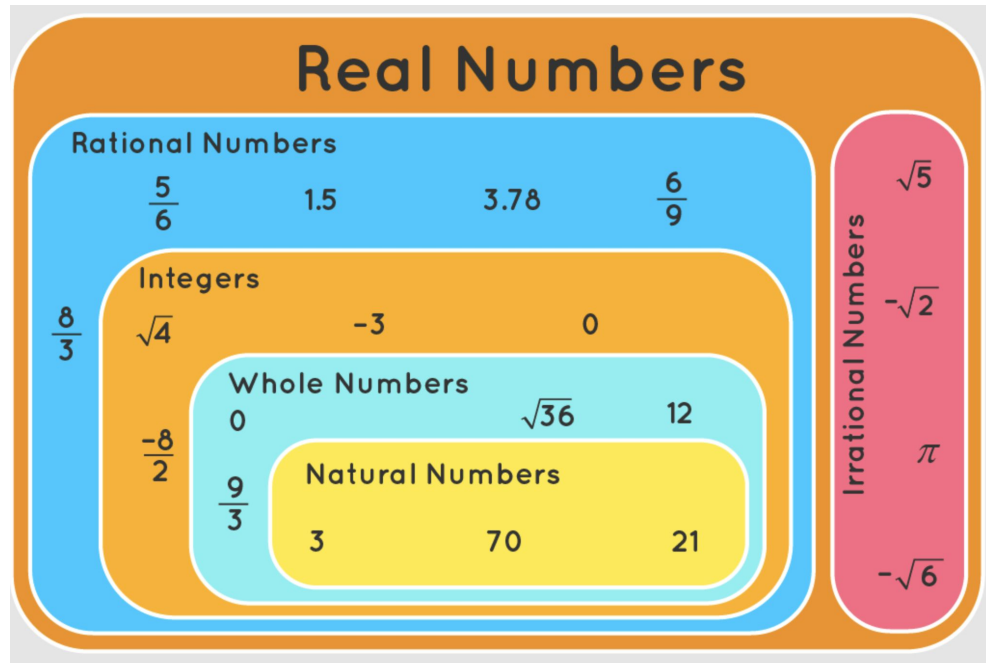


Profesor: **IIC2343 - Arquitectura de Computadores**
- Felipe Valenzuela González
Correo:
frvalenzuela@alumni.uc.cl

Resumen de la clase pasada

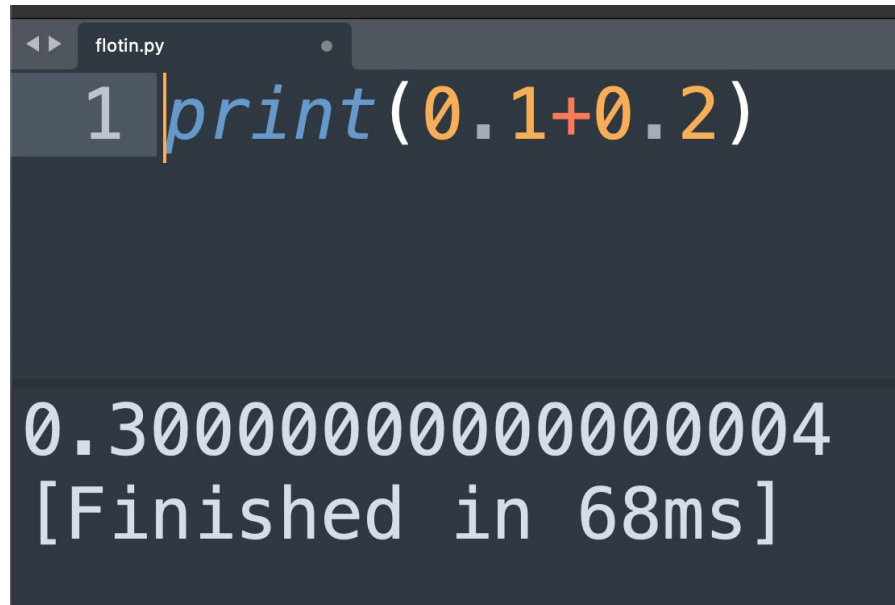
Reales: Imposible

- Un computador tiene memoria y espacio **finito**
- Por lo mismo es **el imposible representar números irracionales**
- Sin embargo, los números **racionales** son posibles



Racionales: Error de redondeo de punto flotante

- Un computador no puede representar el valor exacto de 0.1
- Al operar con el decimal 0.1, no toda operación dará el valor esperado
- Esto es causado por la limitante física para representar números de periodicidad infinita en base binaria



A screenshot of a Python terminal window titled 'flotin.py'. The first line of code is `1 print(0.1+0.2)`. The output displayed is `0.30000000000000004` followed by `[Finished in 68ms]`. The code and output are shown in a light blue font on a dark background.

```
flotin.py
1 print(0.1+0.2)
0.30000000000000004
[Finished in 68ms]
```

Racionales: Punto fijo

- Se divide de forma **fija** en tres partes
 1. Signo
 2. Parte Entera (t)
 3. Parte Fractional (f)
- Simple y rápido de ocupar
- Rango **muy** pequeño

Ejemplo 8 bits:

10,111

0	010	1110
signo	t	f

Racionales: Punto flotante

- Representación que permite “**mover**” la coma (flota)
- Se compone de **exponente** y **significante**
- **Pérdida de precisión**

Ejemplo 8 bits:

$10,111 \rightarrow 1,01 \text{ }^{001}$

0	101	0	001
<hr/>			
signo s	s	signo e	e

Racionales: Single Precision Floating Point - IEEE754

- Basada en notación científica **normalizada**
- Exponente **desfasado** en 127
- Significante **normalizado**
- Tiene un **gran** rango
- Tiene **valores reservados**

1 bit	8 bits	23 bits
signo	significante	exponente
		significante

Valores reservados:

+0=	00000000	000000000000000000000000
-0=	10000000	000000000000000000000000
+∞=	01111111	000000000000000000000000
-∞=	11111111	000000000000000000000000
NaN=	01111111	xxxxxxxxxxxxxxxxxxxxxxxxxxxx

Racionales: Single Precision Floating Point - IEEE754

- Un número de punto flotante, en estándar IEEE754, se ve así:

$$N = (-1)^{\text{signo}} \times 1.\text{significante} \times 10^{(\text{exponente} - 127)_b}$$

Ejemplo:

$$0.00101b = (1.01 * 10^{-11})b$$

$$\text{exponente} - 127 = -3 \Rightarrow \text{exponente} = 124 = (01111100)$$

$$\text{Según IEEE754: } 0.00101b = 001111100010000000000000000000$$

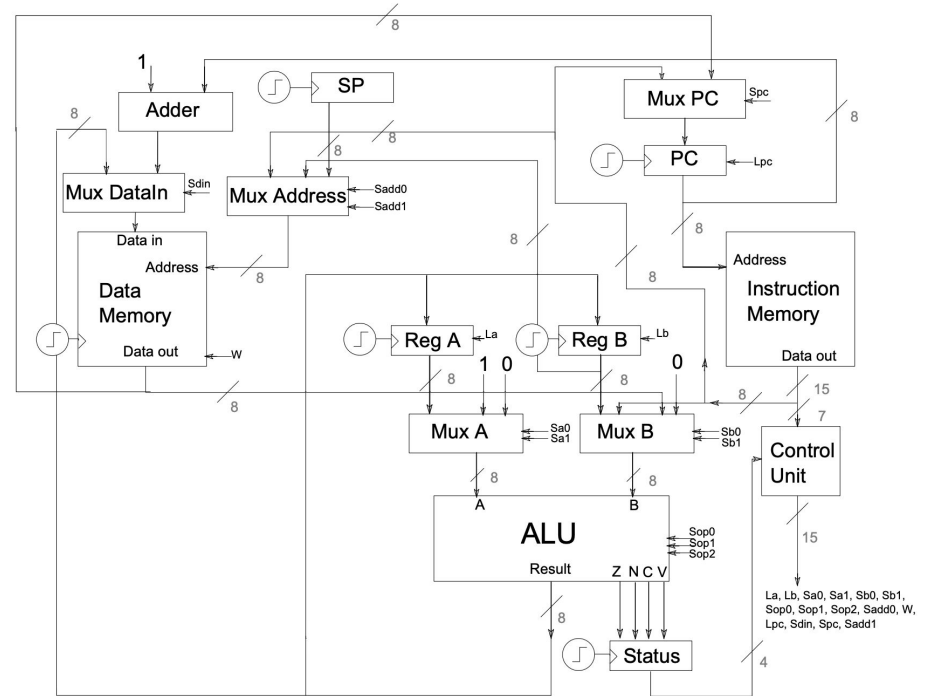
¿Dudas?

Introducción del curso: ✓

- Un computador lo definimos como una **máquina programable que ejecuta programas.** ✓
- Para programar necesitamos:
 - Datos: números (enteros, racionales) , texto, imágenes, etc ✓
 - Operaciones: suma, resta, multiplicación, división, etc ✓
 - Variables: simples, arreglos ✓
 - Control de flujo: comparaciones, manejo de ciclos ✓
- ¡COMPLETAMOS LA INTRODUCCIÓN! **Ahora comienza el curso...**

Arquitectura de Computadores: Computador Básico

- Posee **registros** y unidades de ejecución y control
- Además de hacer cálculos, puede realizar operaciones de **control de flujo**
- Provee modularidad básica, al dar soporte para **subrutinas**



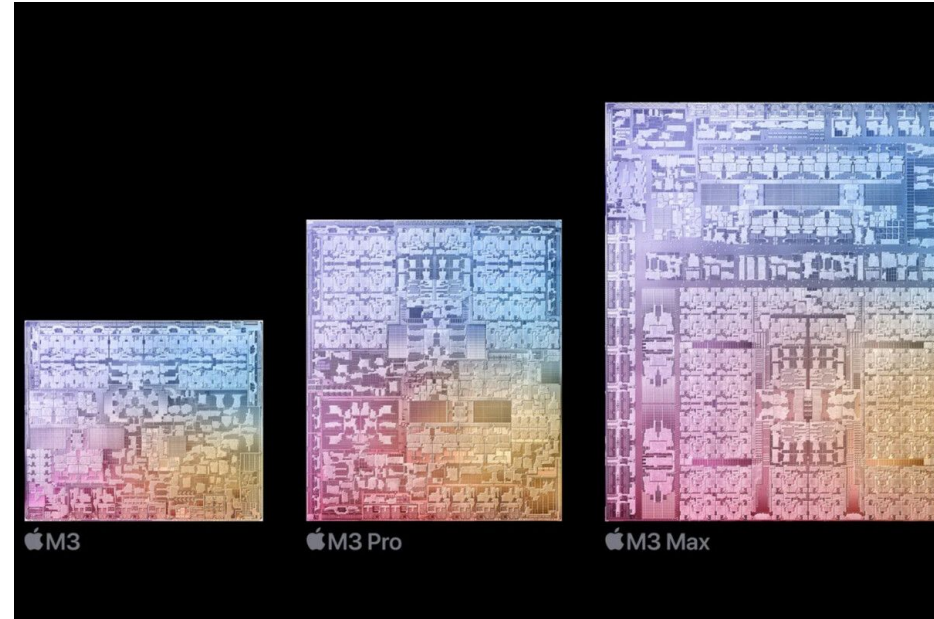
Arquitectura de Computadores: Otras alternativas

- Distintos computadores pueden diferir en el conjunto de **funcionalidades básicas y fundamentales**
- Por otro lado, existen computadores que son programados de la misma manera (ej. AMD-Intel), pero su construcción interna es distinta
- Decisiones en cuanto a cantidad de registros, tamaño de buses, memorias, instrucciones, definen la **arquitectura de un computador**



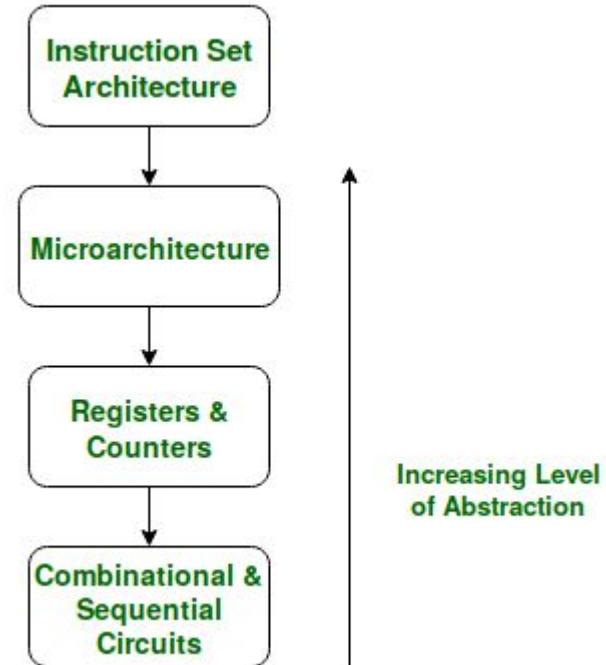
Arquitectura de Computadores: Microarquitectura

- Se refiere a los distintos **componentes de hardware** que están presentes en el computador
- Se compone básicamente de:
 - Registros
 - Unidades de control
 - Condition Codes
 - Memorias (Datos, Stack, etc)



Arquitectura de Computadores: Instruction Set Architecture (ISA)

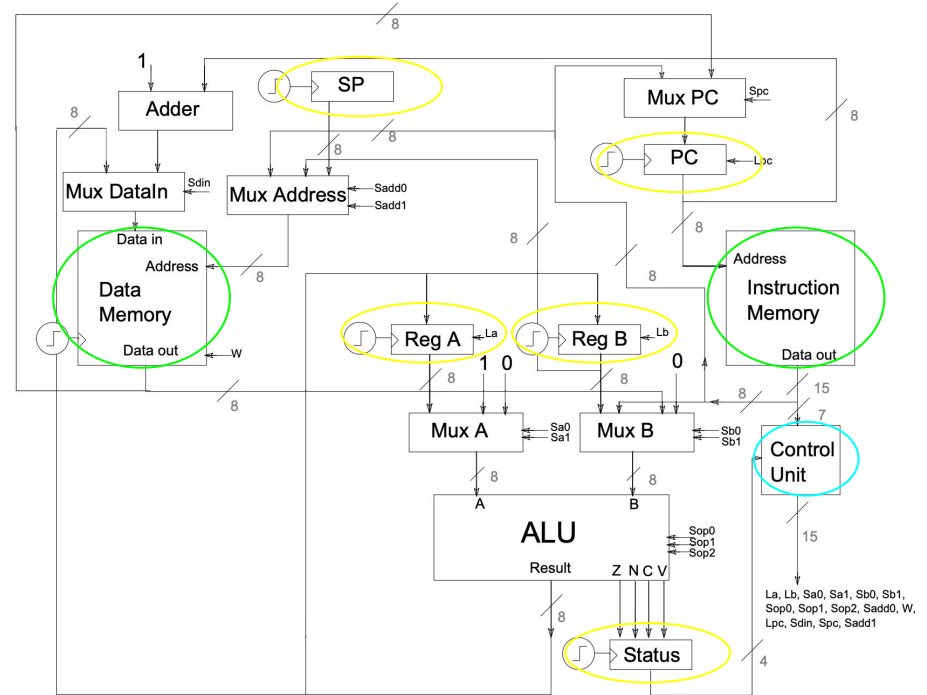
- Se refiere a tipo, formato, características de las instrucciones soportadas por **el computador**
- Especifica **cómo** escribir los programas en el computador
- Se compone básicamente de:
 - Tipos de instrucciones
 - Tipos de datos
 - Modos de direccionamiento de memoria
 - Formato de instrucción
 - Manejo del stack
 - Palabras por instrucción
 - Ciclos por instrucción



¿Dudas?

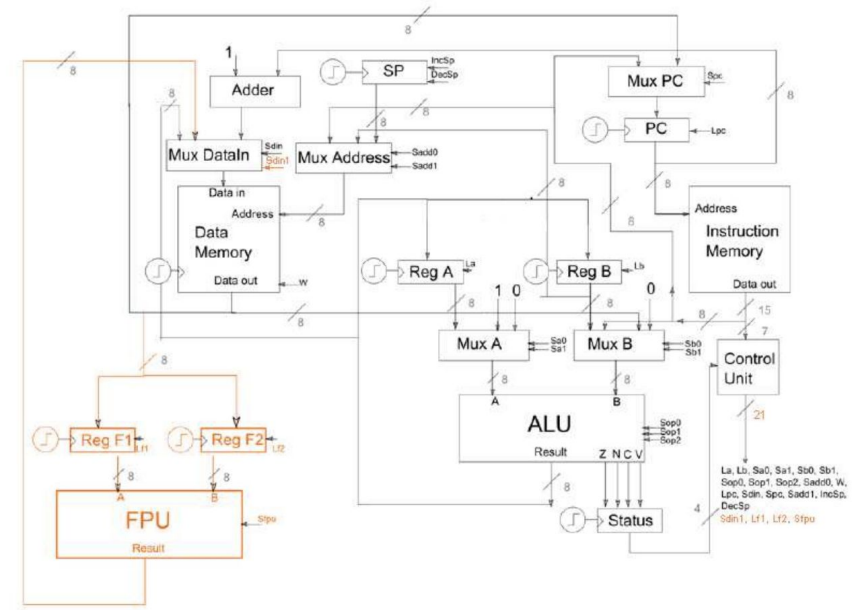
Arquitectura de Computadores: Microarquitectura

- Podemos separar lo que es **memorias** de lo que es utilizado para procesar información o también llamado **procesador**
- Así mismo podemos hacer separación, dentro del procesador, lo que son **registros**, **unidad de ejecución** y **unidad de control**



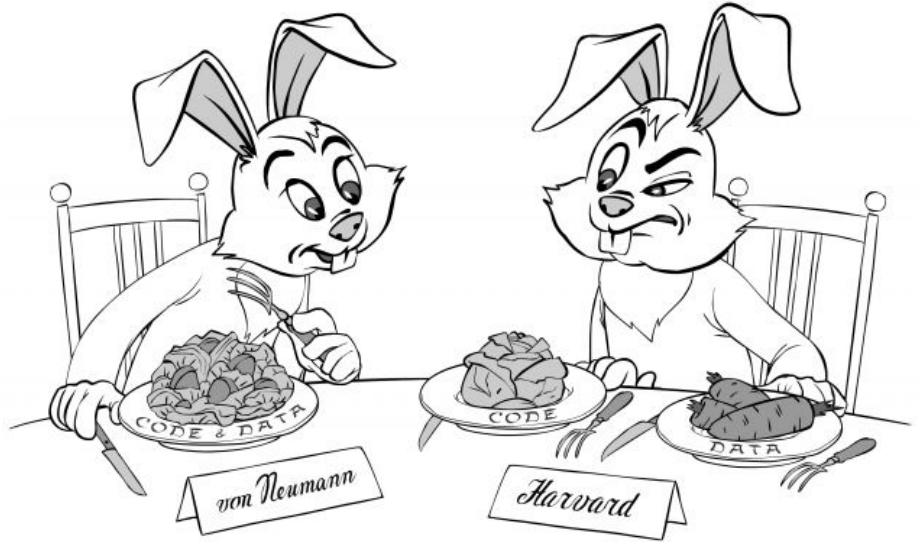
Arquitectura de Computadores: Microarquitectura - modificaciones

- Dentro de la misma arquitectura podemos hacer distintas modificaciones tales como:
 - Aumentar más registros
 - Aumentar tamaños
 - Separar memoria de datos y de pila
 - Agregar más unidades de ejecución



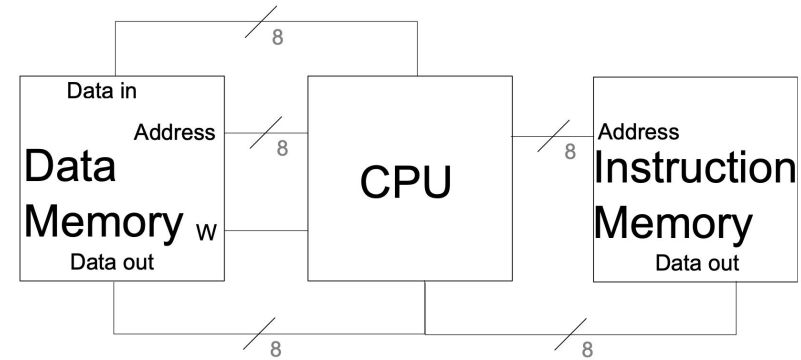
Arquitectura de Computadores: Microarquitectura - Paradigmas

- **Arquitectura Harvard:** Memoria de datos e instrucciones independientes
- **Arquitectura Von Neumann:** Memoria única que comparte datos e instrucciones. Permite escribir datos como si fueran instrucciones



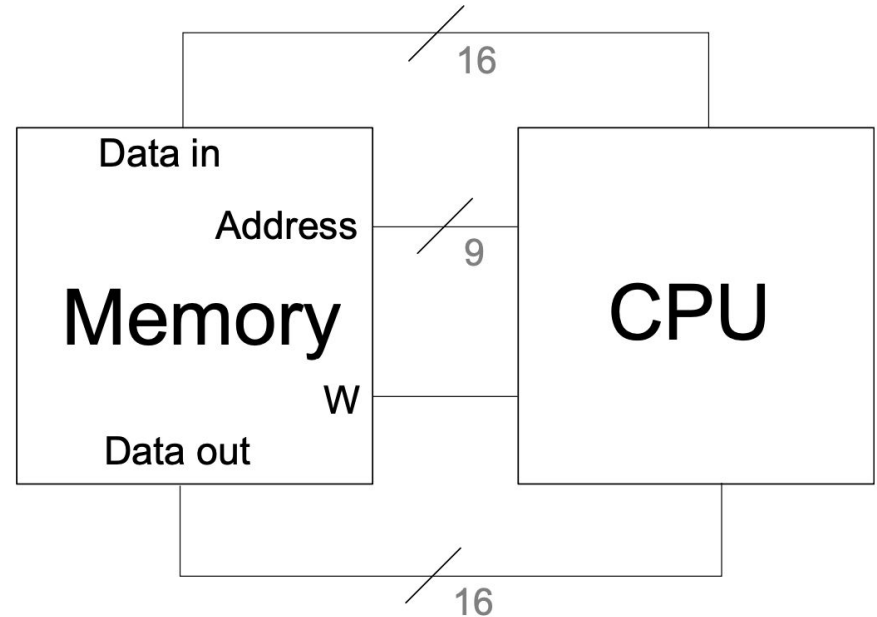
Arquitectura de Computadores: Harvard

- El bus de datos funciona de forma **bidireccional** entre la memoria de datos y la CPU
- Los datos de la memoria de instrucciones van en una **única** dirección a la CPU



Arquitectura de Computadores: Von Neumann

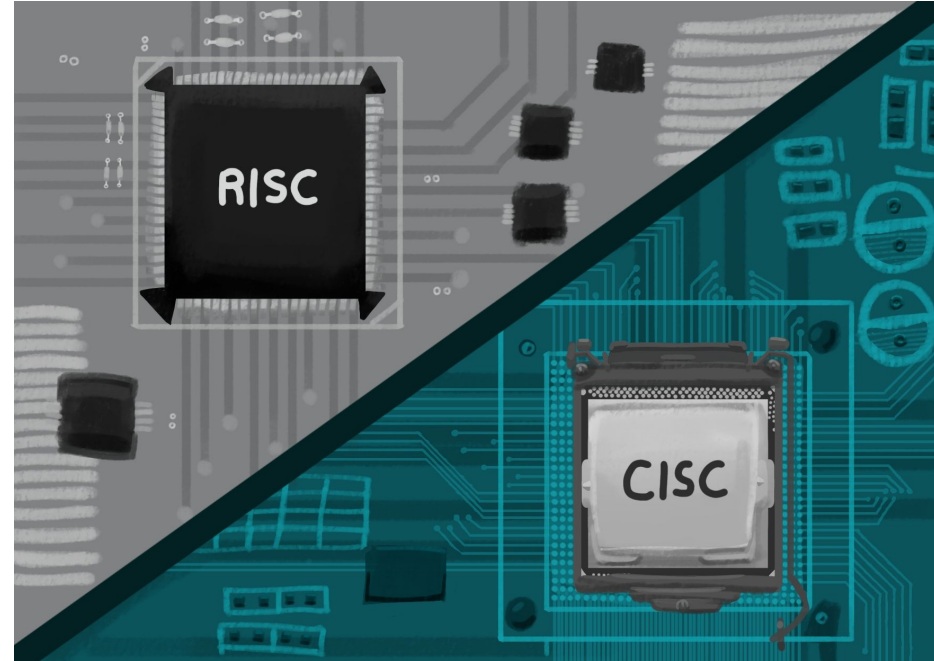
- El bus de datos funciona de forma **bidireccional** entre la memoria de datos y la CPU
- Las instrucciones se tratan como si fueran **datos**, por lo que recorren las mismas conexiones que las variables almacenadas



¿Dudas?

Arquitectura de Computadores: ISA - Paradigmas

- **RISC: Reduced Instruction Set Computer.** Instrucciones pequeña y simples. Su diseño permite simplificar el hardware, poniendo énfasis en el software.
- **CISC: Complex Instruction Set Computer.** Muchas instrucciones y con complejidad alta. Énfasis en un hardware más complejo para poder ejecutarlas.

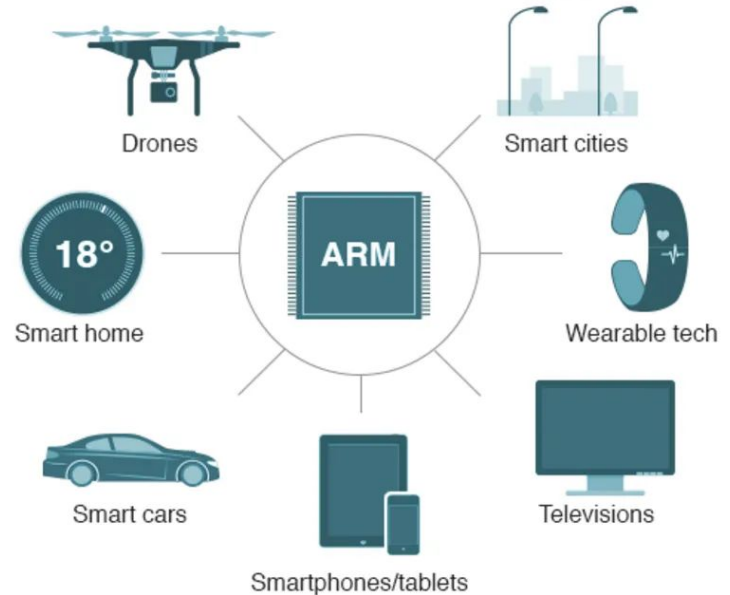


Arquitectura de Computadores: ISA - Computador Básico

- **Tipos de instrucciones:** cargas, aritméticas, saltos, etc
- **Tipos de datos:** Números binarios con y sin signo
- **Modos de direccionamiento de memoria:** Directo e indirecto por registro
- **Manejo del stack:** General
- **Formato de instrucción:** Mixto (0, 1 o 2 argumentos)
- **Palabras por instrucción:** 1 (salvo por RET y POP)
- **Ciclos por instrucción:** 1 (salvo por RET y POP)
- **RISC**

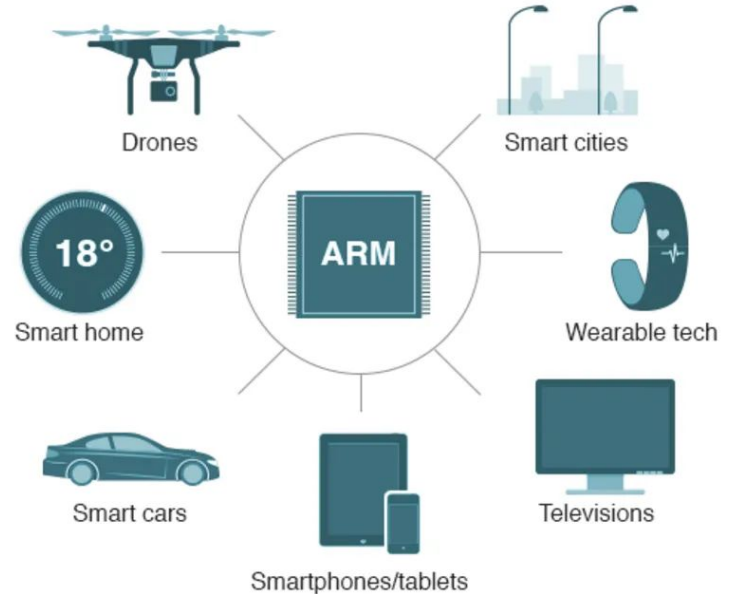
Arquitectura de Computadores: ISA - Industria

- **ARM:** Empresa que se dedica al diseño de ISAs de bajo consumo energético (RISC). No fabrican CPUs, **venden diseños de ISA y microarquitecturas**. Así, otras empresas fabrican CPUs que corren bajo esta misma ISA



Arquitectura de Computadores: ISA - Industria - ARM

- **ARM:** Empresa que se dedica al diseño de ISAs de bajo consumo energético (RISC). No fabrican CPUs, **venden diseños de ISA y microarquitecturas**. Así, otras empresas fabrican CPUs que corren bajo esta misma ISA



Arquitectura de Computadores: ISA - Industria - RISC-V

- **RISC-V** es una arquitectura de instrucciones (ISA) **libre**, basada en el diseño RISC, que optimiza el número de instrucciones
- A diferencia de otras ISAs, RISC-V **es abierta y no requiere regalías**, lo que permite a cualquiera diseñar, fabricar y vender chips y software
- Aunque no es la primera ISA abierta, destaca por su versatilidad y aplicación en una amplia variedad de dispositivos

Intel x86 32-bit



~1300 instructions
3 instructions / month
182 hours to read

ARM 32-bit



~500 instructions
79 hours to read

RISC-V RV32I



40 instructions
6 hours to read

¿Dudas?

Clase 10 - Arquitectura de Computadores

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