



Unidad 1 / Escenario 1

Lectura fundamental

# Introducción a la arquitectura del computador

## Contenido

- 1 ¿Qué es la arquitectura del computador?
- 2 ¿Por qué aprender arquitectura del computador?
- 3 Arquitectura en el contexto de la Ingeniería de Sistemas
- 4 Arquitectura como Interfaz Hardware-Software
- 5 Introducción al Proyecto

**Palabras clave:** arquitectura del computador, conjunto de instrucciones, microarquitectura, procesador.

## 1. ¿Qué es la Arquitectura del Computador?

La palabra arquitectura de acuerdo con la Real Academia de la Lengua Española es el arte de diseñar edificios para un propósito particular (Real Academia de la Lengua Española, 2017). Un arquitecto desarrolla la capacidad de determinar qué requiere una construcción, de acuerdo con el uso que se le quiere dar. Por ejemplo, son diferentes los requerimientos y diseño de una construcción que será destinada como vivienda de una familia numerosa a aquella destinada a las oficinas de un banco.

De otra parte, el computador se define como un artefacto en el que se encuentran elementos interconectados como procesadores, memorias, buses de datos y buses de control, capaces de aplicar un set de operaciones aritméticas y lógicas sobre un conjunto de datos para transformarlos, almacenarlos, o tratarlos con el propósito de resolver problemas de diversas índoles (Real Academia de la Lengua Española, 2017).

De acuerdo con lo anterior, la arquitectura del computador se define como: “la ciencia y el arte de diseñar, seleccionar e interconectar componentes de hardware, y el de diseñar interfaces hardware/software para crear sistemas de cómputo que cumplan ciertas metas funcionales, de desempeño, de consumo de energía, de costo, entre otros” (Patt, 2001). Claramente, la arquitectura del computador no se refiere al diseño de edificios. Sin embargo, sí toma de la definición de “arquitectura” el hecho de diseñar una estructura y la interconexión de componentes según las especificaciones de uso esperado, en el caso de la arquitectura del computador del dispositivo de cómputo.

En síntesis...

La arquitectura del computador no es sólo una ciencia, es también un arte.



## 2. ¿Por qué Aprender Arquitectura del Computador?

El diseño y construcción de computadores, así como la comprensión de cómo funcionan y se interconectan sus componentes de hardware, es parte fundamental de la Ingeniería de Sistemas. En este proceso de diseño es necesario considerar dispositivos y plataformas de hardware que se adapten a las aplicaciones requeridas teniendo en cuenta los requerimientos de almacenamiento, capacidad de procesamiento, costo, tamaño, tiempos, entre otros (Ortega & Anguita, 2012).

### Cómo mejorar...



Cuando piensas una aplicación móvil, ¿Qué requerimiento de hardware es más importante, la capacidad de almacenamiento o el peso (en gramos) del dispositivo?

Comprender cuáles son las características del hardware y cómo se realiza la comunicación entre el software y el hardware pueden contribuir a obtener mejores diseños de sistemas computacionales y programas de software. Por ejemplo, conocer la forma en que se organiza la memoria del computador y en que se asignan los recursos de hardware durante la ejecución de un programa puede contribuir con la creación de programas más eficientes.

Adicionalmente, los dispositivos computacionales son la base de la ingeniería de sistemas en Colombia. En este sentido, entender su arquitectura es de gran importancia para comprender usos actuales y tendencias. En el futuro, ¿cómo serán los computadores?, ¿qué partes tendrán?, ¿qué usos tendrán?, ¿cómo se programarán?, ¿qué se podrá hacer con ellos?, ¿qué limitaciones tendrán? (García Sánchez, Expósito, García, Pérez, & Pérez, n.d.).

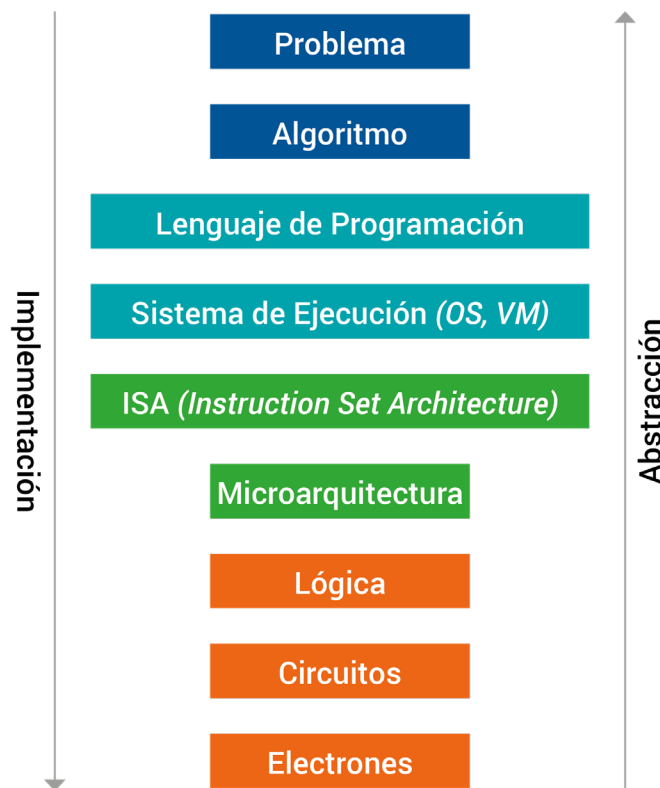
### Cómo mejorar...



En el futuro, ¿cómo serán los computadores?, ¿qué partes tendrán?, ¿qué usos se les darán?, ¿cómo se programarán?, ¿qué se podrá hacer con ellos?, ¿qué limitaciones tendrán?

### 3. Arquitectura en el contexto de la Ingeniería de Sistemas

En el esquema de la Figura 1, se presentan los niveles de transformación a través de los cuales se utilizan sistemas computacionales para resolver un problema.



**Figura 1. Niveles de transformación del proceso de cómputo para resolver un problema con un sistema computacional**

Fuente: elaboración propia

¿Cuáles son los niveles de transformación que se utilizan en un sistema de cómputo para resolver un problema? El nivel inicial corresponde a la abstracción de la situación problema, escribiéndola en un **lenguaje natural**, como el español o el inglés. Una vez abstraído el problema, se debe estructurar en un **algoritmo**. El paso siguiente corresponde a la escritura del algoritmo en un **lenguaje de programación** estructurado, de alto nivel, siguiendo ciertas reglas sintácticas de acuerdo con el lenguaje seleccionado. Si se siguen las reglas de sintaxis adecuadamente, el programa se puede compilar, lo que implica que un compilador lo traduzca de alto a bajo nivel. El lenguaje de bajo nivel se llama también ensamblador, y corresponde a la escritura de un **conjunto de instrucciones (ISA)**. **La microarquitectura** es el siguiente paso. Ésta corresponde a la interconexión de elementos de hardware, como la memoria o la unidad lógico-aritmética (ALU por sus siglas en inglés), que en conjunto pueden ejecutar cada una de las instrucciones del conjunto de instrucciones, ISA.

## ¿Sabía que...?



Un algoritmo es un conjunto de pasos simples ordenados de forma secuencial que permiten hallar la solución a un problema.

Cada uno de los elementos que constituyen la microarquitectura es construido usando elementos de **lógica digital**: compuertas AND, OR y NOT, conectadas en estructuras combinacionales y secuenciales. Las compuertas básicas son las unidades fundamentales de la lógica digital que se construyen usando transistores que conmutan y se ensamblan en **circuitos**. En esencia, los transistores conmutan gracias a sus propiedades de semiconductores, los cuales permiten o no el flujo de **electrones**.

## En síntesis...

Los electrones fluyen controladamente por los transistores. Éstos se configuran en circuitos lógicos que se ensamblan en una microarquitectura que incluye componentes como memorias o el procesador; capaces de ejecutar una serie de instrucciones definidas en el ISA. El ISA conforma instrucciones en bajo nivel que ensamblan un lenguaje de alto nivel, que a su vez, implementa un algoritmo que busca resolver un problema.



La ingeniería de sistemas se desarrolla en torno a los siguientes niveles: problema, algoritmo, lenguaje de programación, compilación, ISA y microarquitectura. La ingeniería electrónica se encuentra en los niveles: ISA, microarquitectura, lógica, circuitos y electrones. La arquitectura del computador estudia justo los dos niveles que comparten ambas ingenierías: ISA y microarquitectura. Es por esto que se dice que la arquitectura está en la interfaz software/hardware y permite que un programa pueda ser ejecutado por un conjunto de elementos de hardware.

Desde la perspectiva de la ingeniería de sistemas, diseñar un sistema computacional se orienta al software respondiendo a la pregunta: ¿cómo se puede diseñar un programa traducible a lógica digital? Desde la ingeniería de electrónica, la perspectiva se enfoca en el hardware. De manera que diseñar un sistema computacional corresponde a responder la pregunta: ¿cómo se puede diseñar un sistema que compute información interconectando compuertas lógicas? En arquitectura del computador, la pregunta resulta de la combinación de las anteriores y es: ¿cómo se puede diseñar un computador que satisfaga las metas de diseño del sistema?

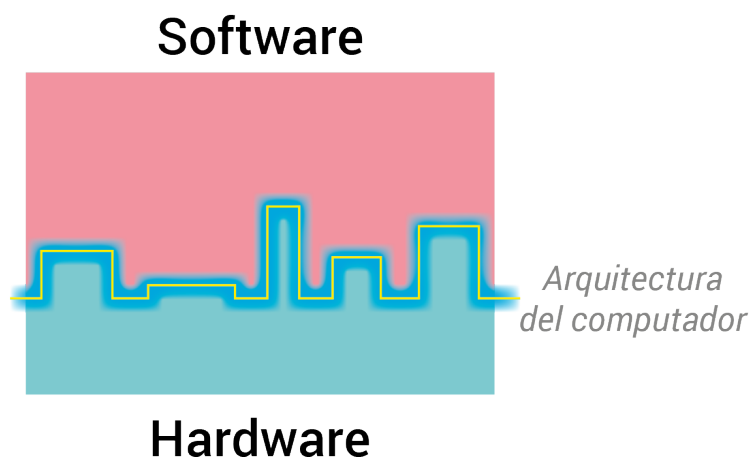
## En síntesis...

La arquitectura del computador se ocupa de responder la pregunta:  
¿cómo se puede diseñar un computador que satisfaga las metas  
de diseño del sistema?



### 4. Arquitectura como Interfaz Hardware-Software

Como se presentó en la sección anterior, la arquitectura es la frontera difusa entre el hardware y el software. En la Figura 2 se representa el punto medio entre lo físico (hardware) y lo lógico o intangible (software) como aquello que permite que un conjunto de líneas de código sea interpretable por compuertas lógicas interconectadas en circuitos.



**Figura 2. La arquitectura es el punto medio entre el hardware y el software**

Fuente: elaboración propia

En esta transición entre hardware y software existen dos niveles que son de los que se encarga principalmente la arquitectura del computador: **ISA** (Instruction Set Architecture) que corresponde al conjunto de instrucciones que el hardware sabe implementar y la **microarquitectura** que corresponde a los bloques de hardware que se interconectan para ejecutar dichas instrucciones.

### Conjunto de Instrucciones (ISA)

Corresponde a la parte de la arquitectura que se estudia desde la programación. Dentro del ISA (Instruction Set Architecture) se incluyen las siguientes definiciones (García Sánchez, Expósito, García, Pérez, & Pérez, n.d.):

- **Instrucciones:** cada una descrita en términos de su nombre, código binario único que la representa (como si fuera su cédula), parámetros que recibe y un ejemplo de cómo se usa.
- **Registros:** corresponde a cuántos se usan y cómo es su codificación; qué representa cada uno, qué numeración tienen y cómo se relacionan con las instrucciones. Los registros son espacios de memoria pequeños y cercanos al procesador, que son los que almacenan los operandos y los resultados.
- **Formato de instrucciones:** en esta parte se especifica cómo se codifica cada instrucción según su tipo. Hay instrucciones que son aritméticas, otras que se refieren al almacenamiento y carga de información en o desde la memoria principal, otras asociadas a condicionales, entre otras. Cada tipo de instrucción se codifica en un formato diferente. Sin embargo, instrucciones de un mismo tipo tienden a tener la misma codificación para facilitar su interpretación a nivel de hardware o microarquitectura.
- **Modos de direccionamiento:** Se refieren a cómo se accede la información que va a ser operada: ¿Es un valor constante? ¿Es un valor variable almacenado en los registros? ¿Es un valor variable almacenado en la memoria RAM? ¿Es una dirección? ¿Es una dirección de la memoria de programa?
- **Condiciones de excepción:** corresponde a cuándo cambia el flujo de programa o qué interrupciones existen y cómo se manejan.

Existen principalmente dos tipos de conjuntos de instrucciones. El primero, como el usado en Intel 80x86, en el que la memoria principal (RAM) puede ser accedida desde cualquier tipo de instrucción. El segundo, que será el trabajado en este curso, es usado en procesadores ARM y MIPS, y sólo permite que la memoria sea accedida por operaciones de carga y almacenamiento: tipo *Load/*

*Store*. Los conjuntos de instrucciones actuales son, en su mayoría, del tipo *Load/Store*. Estos incluyen registros de propósito general que son una porción mucho más pequeña de memoria RAM accesibles por operaciones aritmético-lógicas y de salto, y que pueden contener valores de la memoria principal sólo si se han cargado previamente a través de una instrucción *load*. Así mismo, los registros pueden contener valores a escribir en la memoria RAM, proceso que se realiza a través de operaciones *store*.

## Microarquitectura

La microarquitectura corresponde a los elementos de hardware que interconectados a través de buses de datos y control son capaces de ejecutar el conjunto de instrucciones definidos en el ISA del computador. Los elementos de hardware principales son:

- **Memoria de programa:** en ésta se almacena el código binario que corresponde a cada una de las instrucciones del algoritmo que debe ejecutar el computador.
- **Memoria de Datos:** esta memoria sólo almacena datos, información que corresponde a los operandos o resultados que se van dando en el procesamiento del algoritmo que se está ejecutando.
- **Registros:** se utilizan en las arquitecturas del tipo *load/store*. Son una pequeña memoria cercana al procesador donde cada espacio puede ser de propósito general o específico. Los de propósito general almacenan información que es operada o resulta directamente de la unidad lógico aritmética o contienen apuntadores a la memoria principal. Los de propósito específico, corresponden a apuntadores especiales como el apuntador de pila, registros destinados a los parámetros de funciones o apuntadores para el retorno de funciones.
- **Unidad de Control:** Es como el maestro de orquesta del computador. Se encarga de: definir qué elemento trabaja en cada momento, decodificar la instrucción, enviar a cada parte los parámetros que requiere según la instrucción que está siendo operada, entre otras.
- **Unidad Lógico-Aritmética (ALU):** Contiene bloques de hardware para ejecutar instrucciones lógicas (AND, OR, NOT, XOR, etc) y aritméticas (SUMA, RESTA, MULTIPLICACIÓN, DIVISIÓN). Recibe un código de operación (OpCode) que corresponde a qué operación se desea ejecutar y, adicionalmente, dos entradas que son las que se operan. Además, tiene una salida correspondiente al resultado de la operación y algunas banderas de salida que indican aspectos del resultado. Estas banderas son: cero (cuando el resultado da cero), negativo (cuando el resultado es menor que cero), *overflow* (cuando el resultado es de mayor tamaño que el espacio disponible para tal fin).



- **Buses de control:** Es el que conecta la Unidad de control con las entradas de control de cada uno de los elementos del hardware (memoria de datos, ALU, Unidad de entrada salida, entre otros).
- **Buses de datos:** a través del cuales fluye la información: operandos, resultados, información almacenada en la memoria de programa, entre otros.
- **Unidad de entrada salida:** es la que permite la comunicación del computador con periféricos de entrada/salida, para facilitar la interacción con el usuario.

¿Sabía qué...?



Una de las grandes innovaciones en la computación es que los programas (algoritmos) se manejan como datos en el computador?

## 5. Introducción al Proyecto

En este curso desarrollaremos un sencillo procesador que sea capaz de completar una tarea: determinar si un número es primo o no. Este proyecto se realizará sobre Logisim. (<https://sourceforge.net/projects/circuit/>).

De un lado, se diseñará el conjunto de instrucciones, al menos las necesarias para poder cumplir con el problema especificado. El diseño del conjunto de instrucciones incluirá el detalle de cada instrucción, la descripción de la cantidad de bits sobre la que se manejará la arquitectura, los tipos de direccionamiento, los formatos de instrucción y otras consideraciones que sean necesarias precisar para documentar completamente el diseño de su procesador. Adicionalmente, incluirá la microarquitectura simulada en Logisim (<https://sourceforge.net/projects/circuit/>) para ejecutar las instrucciones del ISA diseñado para su procesador.

# Referencias

García Sánchez, J. D., Expósito, S. D., García, J., Pérez, O. A., & Pérez, J. M. (n.d.). *Fundamentos del Diseño de Computadores - Arquitectura del Computador*. Obtenido de Fundamentos del Diseño de Computadores - Arquitectura del Computador: <http://ocw.uc3m.es/ingenieria-informatica/arquitectura-de-computadores/materiales/es-m1-01-fund-ocw.pdf>

Ortega, J., & Anguita, M. (2012). Arquitectura de Computadores en seis créditos ECTS. *Enseñanza y Aprendizaje de Ingeniería de Computadores*, 13-25.

Patt, Y. (2001). Requirements, bottlenecks, and good fortune: agents for microprocessor evolution. *Proceedings of the IEEE*, 1553 - 1559.

Real Academia de la Lengua Española. (2017). *Diccionario de la Real Academia de la Lengua Española*. Obtenido de Diccionario de la Real Academia de la Lengua Española: <http://dle.rae.es/?id=3dyUvi4>

Real Academia de la Lengua Española. (2017). *Diccionario de la Real Academia de la Lengua Española*. Obtenido de Diccionario de la Real Academia de la Lengua Española: <http://dle.rae.es/?id=A4hIGQC>

## INFORMACIÓN TÉCNICA



FACULTAD DE  
**INGENIERÍA, DISEÑO  
E INNOVACIÓN**

**Módulo:** Arquitectura del Computador

**Unidad 1:** Introducción a la arquitectura del computador

**Escenario 1:** Introducción a la arquitectura del computador

**Autor:** Laura Juliana Cortés Rico

**Asesor Pedagógico:** Edwin Alcides Mojica Quintero

**Diseñador Gráfico:** Carlos Montoya

**Asistente:** Leidy Alejandra Morales Eslava

*Este material pertenece al Politécnico Grancolombiano. Por ende, es de uso exclusivo de las Instituciones adscritas a la Red Ilumino. Prohibida su reproducción total o parcial.*