

# Core

## 1.1 Introducción

La complejidad del procesador determina si se clasifican como:

- **Microprocesadores:** Tienen mínima cantidad de memoria y comunicación con otros componentes del chip.
- **Microcontroladores:** Poseen mayor cantidad de memoria y además comunicación con los periféricos integrados a nivel de SoC.

Los procesadores incrustados son definidos a partir de su **arquitectura**. Se diferencian a partir del set de instrucciones.

## 1.2 Arquitectura ISA

Se utiliza como referencia del curso. ISA: Instruction Set Architecture. Posee los siguientes aspectos:

1. **Operaciones:** son las instrucciones del procesador que se pueden ejecutar en esta arquitectura.
2. **Operandos:** Los datos que son aceptados. La manera en la que se accesan depende de la arquitectura.
3. **Almacenamiento:** La manera en la que se maneja la memoria
4. **Modos de direccionamiento:** Manera en la que se traen los datos para ser procesados.
5. **Manejo de interrupciones:** Es la atención de eventos que el procesador puede hacer.

Estas características llegan a influir en el desempeño, tiempo de ejecución, funcionalidades disponibles y costos.

### 1.2.1. Operaciones

También se les refiere como instrucciones. Son las funciones ejecutadas sobre los datos e incluyen por lo general cálculos matemáticos, movimiento de datos de la memoria a registros, saltos condicionales e incondicionales, operaciones de entrada y salida, y operaciones de conmutación de texto.

El formato de las operaciones se define por medio de una numeración de bits conocida como Código de Operación (OpCode).

### 1.2.2 Operandos

Se definen los tipos y formatos específicos para la arquitectura. Se definen los operandos según la cantidad de bits que utiliza la arquitectura. Por ejemplo, 1 byte (8 bits) o 1 word (32 bits). También se establece la forma cómo se soportan: binario, decimal, etc.

### 1.2.3. Almacenamiento

Es la arquitectura en particular de cómo se maneja la memoria.

#### 1.2.3.1. Organización de memoria:

La memoria es un arreglo en el espacio de almacenamiento, donde se guarda operaciones, operandos y otros. Los índices de este arreglo son las localizaciones de la memoria y se conocen como **direcciones de memoria**. El rango virtual de direcciones disponibles es conocido como **espacio direccionable**. El direccionamiento puede ser por ejemplo lineal o segmentado.

Existen dos maneras de almacenar la información:

- **Big Endian:** El byte más significativo es almacenado primero.
- **Little Endian:** El byte menos significativo es almacenado primero.

#### 1.2.3.2. Conjunto de registros:

Un registro es una memoria de fácil acceso, utilizado para almacenar operandos que luego son utilizados. El conjunto de registros de un procesador es conocido como **Register Set** o **Modelo de Programación**. En el caso específico del S12, tiene 4 registros de 16 bits y 3 registros de 8 bits.

### 1.2.4. Modos de direccionamiento

Los modos de direccionamiento definen la forma en que un procesador accede a los **operandos** en el espacio de almacenamiento. Las dos arquitecturas según los modos de direccionamiento más comunes son:

- **Arquitectura Load/Storage:** sólo permite operaciones para procesar datos en los registros y no en el espacio de memoria.
- **Arquitectura Register/Memory:** permite que las operaciones sean procesadas tanto en registros como en la memoria. Esta es la utilizada en el Freescale S12.

### 1.2.5. Manejo de interrupciones y excepciones.

Las interrupciones son mecanismos que detienen el flujo normal del programa con el objetivo de ejecutar otro segmento de código en respuesta a algún **evento**. El ISA define cualquier tipo de soporte de hardware que se tenga para las interrupciones y define la forma como estas excepciones son procesadas en la arquitectura.

### 1.2.6. Clasificación de los modelos ISA.

Los modelos ISA tienen las siguientes clasificaciones:

#### 1.2.6.1. Modelo ISA de Aplicación Específica

También conocidos como Application-Specific Instruction Set Processor (ASIP). Se utilizan en aplicaciones como aceleradores de hardware para procesamiento de señales, video coding, TV video controllers. Se dividen en los siguientes modos:

- **Controller:** No requieren una ejecución compleja en la manipulación de datos, tales como procesadores de video y audio que funcionan como esclavos en un sistema de televisión.

- **Datapath:** Implementado en procesadores cuyo propósito es ejecutar repetidamente un cálculo fijo sobre diferentes conjuntos de datos. Por ejemplo los procesadores DSP.
- **FSMD (Finite State Machine with Datapath):** es implementado para aplicaciones que no requiere una compleja manipulación de datos y deben ejecutar repetidamente cálculos fijos sobre diferentes grupos de datos. Entre estos se encuentran las ASICs, PLD y FPGAs.
- **ISA JVM (Java Virtual Machine):** están implementados en un sistema incrustado por medio de hardware como el aj-80 y aj-100. Poseen Java de forma nativa.

#### 1.2.6.2. Modelo ISA de Propósito General

Fueron desarrollados con una arquitectura “abierta” que permite a los desarrolladores una implementación de programas y soluciones a partir de ellas. Se dividen en dos categorías:

- **Complex Instruction Set Computing (CISC):** Tienen un extenso set de instrucciones, muchos modos de direccionamiento y posee instrucciones complejas (que utilizan múltiples ciclos de reloj).
- **Reduced Instruction Set Computing (RISC):** Tiene un set de instrucciones limitado y posee pocos modos de direccionamiento. La mayoría de sus operaciones son rápidas.

#### 1.2.6.3. Modelo ISA de Paralelismo a Nivel de Instrucción

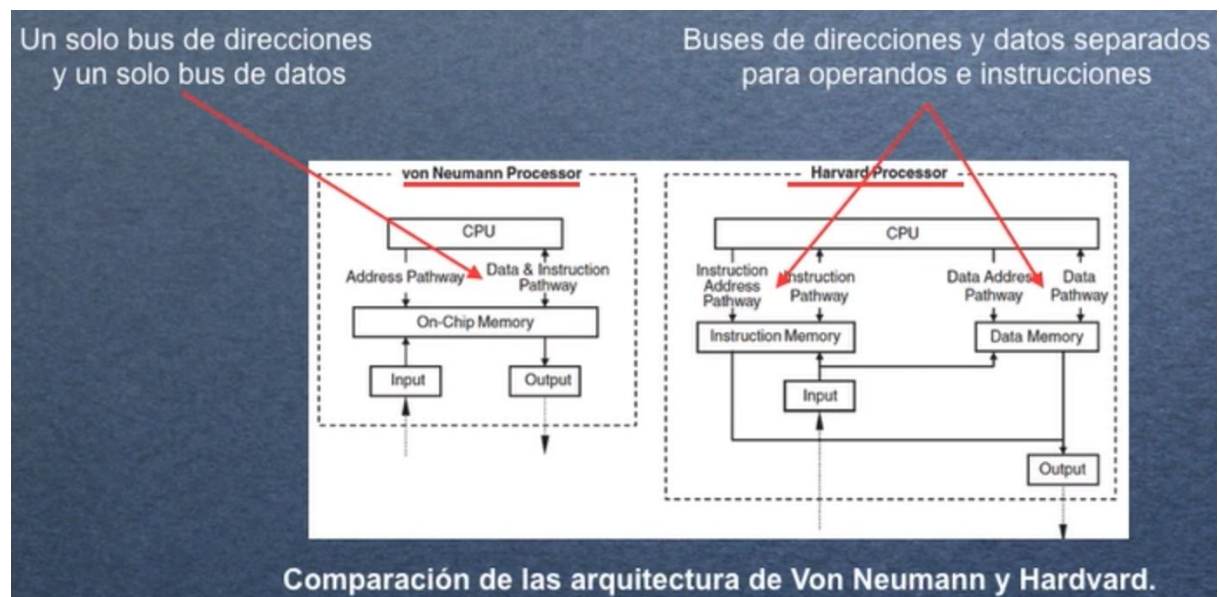
Posee la robustez del modelo tipo CISC, con la rapidez del modelo tipo RISC. Entre estas se pueden encontrar los siguientes modelos:

- **SIMD (Single Instruction, Multiple Data):** está diseñado para procesar una instrucción simultáneamente con múltiples componentes de datos que requieren que la misma acción sea ejecutada sobre ellos. Por ejemplo un controlador en una impresora.
- **Máquina Superescalar:** Procesa múltiples instrucciones simultáneamente dentro de un mismo ciclo de reloj mediante la implementación de componentes de hardware funcionales dentro del procesador. También requiere de un planificador de código que analice la secuencia de las ejecuciones.
- **VLIW (Very Long Instruction Word Computing):** posee instrucciones más largas con múltiples operaciones. Estas operaciones se parten y son procesadas en paralelo. Esta tarea no es resuelta por un planificador de código, sino que el compilador debe encargarse de esto.

### 1.3. Diseño interno del procesador

El ISA define qué puede hacer un procesador, y a su vez el hardware interno ha sido dispuesto para satisfacer los requerimientos del ISA. El **modelo de Von Neumann** define los componentes fundamentales en un sistema incrustado que serían: un CPU, memoria, componentes I/O y buses.

### 1.3.1. Modelo de Von Neumann y Harvard



### 1.3.2. CPU (CORE)

Es la unidad de procesamiento. Es el responsable de ejecutar los ciclos de búsqueda de instrucciones (fetch), decodificar la instrucción y ejecutarla. Estos pasos son conocidos como el **Three-Stage Pipeline**. El core está internamente construido por:

- **Unidad aritmético-lógica (ALU):** implementa las operaciones del ISA.
- **Registros:** El tipo de memoria de acceso rápido. Pueden ser de almacenamiento temporal de la CPU o acciones directas de la CPU. Deben de tener un tamaño definido (8,16,32 bits). Existen de propósito general como los acumuladores, o de propósito específico como las banderas.
- **Unidad de Control:** Administra el ciclo completo de búsqueda y decodificación de las instrucciones. Posee una máquina de estados que genera las señales y temporización para el Three-Stage Pipeline. También, define la función a ejecutar en la ALU y selecciona las señales de escritura para las unidades funcionales del CPU.
- **Buses internos:** Interconectan la ALU, la CU y los registros. Se dividen según la función que cumplen en: bus de datos, bus de direcciones y bus de control.

### 1.3.3. Memoria

Existe una forma de organización de la memoria, conocida como **jerarquía de memoria**. Esta implementación depende del modelo ISA utilizado.

#### 1.3.3.1. Memoria ROM

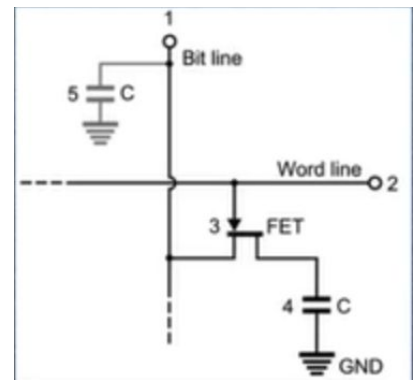
Puede estar integrada en el procesador. Contiene las instrucciones necesarias aún cuando el sistema no tiene potencia. Por esta razón debe ser no volátil, y por lo general es de solo lectura. Existen las siguientes tecnologías:

- **MROM**
- **EPROM**
- **EEPROM**
- **Flash EEPROM**

### 1.3.3.2. Memoria RAM

Es memoria volátil, es decir, cuando se suprime la energía eléctrica se borra su información. Se divide en Static RAM (SRAM) y Dynamic RAM (DRAM).

Las RAM dinámicas borran el dato después de cada lectura, por lo que es necesario volver a escribir si se requiere nuevamente el dato. Esto lleva a una mayor simplicidad de diseño, más densidad y menor costo. Por esta razón necesita un Controlador de Memoria que esté refrescando los contenidos de lectura y reescribiendo, en un ciclo regular.



### 1.3.3.3. Administración de Memoria

Se utilizan direcciones lógicas virtuales para representar las direcciones físicas. Para esto utiliza administradores de memoria que resuelven en direccionamiento. Pueden ser de dos tipos:

- **Controlador de Memoria (MEMC):** se utilizan en interfaces como caché, SRAM, DRAM. Se encargan de sincronizar los accesos a memoria y verificar la integridad de los datos a ser transferidos.
- **Unidades de Administración de memoria (MMU):** se utilizan para traducir la dirección lógica en una dirección física a través de **mapeo de memoria**. También maneja la seguridad de la memoria, el control del caché, y la arbitración de los buses.

### 1.3.3.4. Organización de Memoria

Incluye tanto la jerarquía de memoria, como su organización interna. Esto se refiere a las posiciones de memoria que pueden o no ser utilizadas y cómo son accedidas por diferentes partes del sistema. En el caso del S12, se parte la memoria de manera que una porción almacena las instrucciones y la otra almacena los datos. Se utiliza un arreglo bidimensional llamado **mapa de memoria** para definir qué direcciones están ocupadas por cuáles componentes.

### 1.3.4. Procesamiento de Entrada/Salida

Estos componentes son los encargados de mover la información desde y hacia el procesador desde otros componentes. Estos componentes son conocidos como periféricos, y pueden ser tanto físicos, como etapas de conexión de datos; por ejemplo, UART para conexión a Internet.