

# Representación de instrucciones

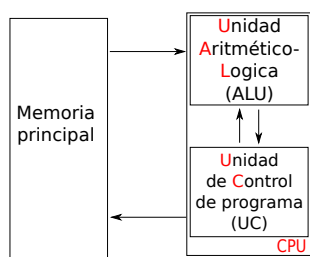
Organización de computadoras 2014

Universidad Nacional de Quilmes

Los detalles que se explican en este apunte pueden no parecer importantes a un programador que esté utilizando lenguajes de alto nivel como Gobstones o Java, donde no son *visibles* las características de la arquitectura.

Un punto de encuentro en que el diseñador del computador y el programador pueden ver la misma máquina es el repertorio de instrucciones. Desde el punto de vista del diseñador, el conjunto de instrucciones máquina constituye la especificación o requisitos funcionales del procesador: implementar el procesador es una tarea que, en buena parte, implica implementar el repertorio de instrucciones máquina. Desde el punto de vista del programador, quien elige programar en lenguaje ensamblador, se hace consciente de la estructura de registros y de memoria, de los tipos de datos que acepta directamente la máquina y del funcionamiento de la ALU.

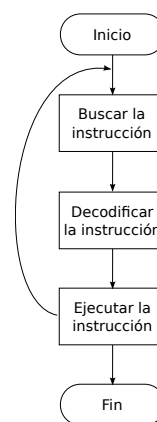
La descripción del repertorio de instrucciones máquina de un computador es un paso más hacia la explicación del procesador de la computadora.



## 1. Características de las instrucciones máquina

El funcionamiento del procesador está determinado por las instrucciones que ejecuta. Estas instrucciones se denominan instrucciones máquina o instrucciones del computador. Al conjunto de ins-

trucciones distintas que puede ejecutar el procesador se denomina **repertorio de instrucciones**.



## Elementos de una instrucción máquina

Cada instrucción debe contener la información que necesita el procesador para su ejecución. Dichos elementos son:

- **Código de operación:** especifica la operación a realizar (suma, resta, etc). La operación se indica mediante un código binario denominado código de operación, o de manera abreviada, **codop**.
- **Referencia a los operandos origen:** La operación puede implicar a uno o mas operandos origen, es decir operandos que son entradas para la operación.
- **Referencia al resultado:** La operación puede producir un resultado y entonces puede ser necesario indicar donde se almacenará.
- **Referencia a la siguiente instrucción:** Indica al procesador de donde captar la siguiente ins-

trucción tras completarse la ejecución de la instrucción actual

La siguiente instrucción a captar está en memoria principal o, en el caso de un sistema de memoria virtual, bien en memoria principal o en memoria secundaria (disco). En la mayoría de los casos, la siguiente instrucción a captar sigue inmediatamente a la instrucción en ejecución y en tales casos no hay referencia explícita a la siguiente instrucción. Cuando sea necesaria la referencia explícita debe suministrarse la dirección de memoria principal o de memoria virtual.

Los operandos y el resultado pueden estar en alguna de las tres áreas:

- **Memoria principal o virtual:** Como en las referencias a instrucciones siguientes, debe indicarse la dirección de memoria principal o memoria virtual.
- **Registro del procesador:** Salvo raras excepciones, un procesador contiene uno o mas registros que pueden ser referenciados por instrucciones maquina. Si solo existe un registro, la referencia a él puede ser implícita. Si existe mas de uno, cada registro tendra asignado un número único y la instrucción debe contener el número de registro deseado.
- **Dispositivo de entrada/salida:** La instrucción debe especificar el modulo y dispositivo de Entrada/Salida para la operación. En el caso de E/S asignadas en memoria, se dará otra dirección de memoria principal o virtual.

## Representación de las instrucciones

Dentro del computador, cada instrucción se representa por una secuencia de bits. La instrucción está dividida en campos correspondientes a los elementos constitutivos de la misma. La siguiente figura muestra un ejemplo sencillo de **formato de instrucción**. En la mayoría de los repertorios de instrucciones se emplea mas de un formato.

Durante su ejecución, la instrucción se escribe en un registro de instrucción (IR) del procesador y este debe ser capaz de extraer los datos de los distintos campos de la instrucción para realizar la operación requerida.

Es difícil, tanto para los programadores como para los lectores de un libro de texto, manejar las **representaciones binarias** de las instrucciones máquina. Por ello, es una práctica común utilizar **representaciones simbólicas** de las instrucciones máquina.

Los **codops** se representan mediante abreviaturas, denominadas *nemotécnicos*, que indican la operación en cuestión. Ejemplos usuales son:

ADD	Sumar
SUB	Restar
MUL	Multiplicar
DIV	Dividir
LOAD	Cargar datos desde memoria
STORE	Almacenar datos en memoria

Los operandos también suelen representarse simbólicamente. Por ejemplo, la instrucción **ADD R,Y** puede significar sumar el valor contenido en la posición de datos Y al contenido del registro R. En este ejemplo, Y hace referencia a la dirección de una posición de memoria, y R a un registro particular. Observe que la operación se realiza con el contenido de la posición, no con su dirección.

Es posible pues escribir un programa en lenguaje máquina en forma simbólica. Cada codop simbólico tiene una representación binaria fija, y el programador especifica la posición de cada operando simbólico. Por ejemplo, el programador podría comenzar con una lista de definiciones:

X=513  
Y=514

y así sucesivamente. Un sencillo programa aceptaría como entrada esta información simbólica, convertiría los codops y referencias a operandos a su forma binaria y construiría las instrucciones máquina binarias.

Es raro encontrar ya programadores en lenguaje máquina. La mayoría de los programas actuales se escriben en un lenguaje de alto nivel o, en ausencia del mismo, en lenguaje ensamblador.

## Fuente

Extraído de *Organización y Arquitectura de Computadores*, William Stallings - 7ta edición. Capítulo 10mo.