

Jaime Castillo Brandon Daniel

1791300 Grupo: 02

*Arquitectura Avanzada de Computadoras*

**Microprogramado**

*Tema 4*

El **microprograma** consta de instrucciones que especifican diferentes señales de control interno para la ejecución de microoperaciones de registro. Cada instrucción de máquina inicia una serie de microoperaciones en la memoria de control.

La **microinstrucción** especifica una o más microoperaciones para el sistema. Una secuencia de microinstrucciones constituye un microprograma.

* Una computadora con una unidad de control microprogramada tendra dos memorias separadas:
  + *Memoria principal*: El contenido de la memoria principal puede variar cuando se manipulan los datos y cada vez que se manipula el programa. El programa dentro de la memoria principal está formado por instrucciones de máquina y flujo de datos.
  + *Memoria de control*: La memoria de control contiene un microprograma fijo que el usuario ocasional no puede alterar.
* La **memoria de control** se considera como una ROM, dentro de la cual se almacena toda la información de forma permanente.
* **Registro de datos de control**: Contiene la microinstrucción mientras que la siguiente dirección se calcula y se lee en memoria.
* El **registro de líneas** de datos en ocasiones se denomina *registro de arquitectura paralela o paralelo****.***
* Cada instrucción de computadora *tiene su propia rutina de microprograma en la memoria de control* para generar las microoperaciones que ejecutan la instrucción.
* Los **microprogramas** que utilizan subrutinas deben tener una parte para almacenar la dirección de retorno durante una llamada a subrutina y restablecer la dirección durante un retorno de subrutina.
* Cuatro **instrucciones de computadora** de las 16 instrucciones posibles en la computadora:
  + *ADD*: Suma el contenido del operando que se encuentra en la dirección efectiva al contenido AC.
  + *BRANCH*: Provoca un brinco a la dirección efectiva si el operador AC es Negativo.
  + *STORE*: Transfiere el contenido de AC a la palabra de memoria específica por la dirección Efectiva.
  + *EXCHANGE*: Intercambia los datos entre AC y la palabra de memoria específica por la dirección efectiva.
* El propósito de un **secuenciador de microprograma** es presentar una dirección a la memoria de control con el fin de que pueda leerse y ejecutarse una microinstrucción.
* Formato de **microinstrucciones**:
  + *Micro Operaciones*
  + *Condición del bit de estado*
  + *Tipo de transferencia de control*
  + *Dirección de memoria*