

T4.elcaminodedatosylaunidaddecon...



Anónimo



Arquitectura de Computadores



2º Grado en Ingeniería Informática



**Escuela Politécnica Superior de Córdoba
Universidad de Córdoba**



[Accede al documento original](#)



Escuela de
Organización
Industrial

Contigo que evolucionas.
Contigo que lideras. Contigo que transformas.

**Esto es EOI.
Mismo propósito,
nueva energía.**



Descubre más aquí



EOI Escuela de
Organización
Industrial

Importante

Puedo eliminar la publi de este documento con 1 coin

¿Cómo consigo coins? → Plan Turbo: barato
→ Planes pro: más coins

perdo
espacio



T4. EL CAMINO DE DATOS Y LA UNIDAD DE CONTROL

Dibujar el esquema de una unidad de control microprogramada y explicar brevemente cada uno de los blowues

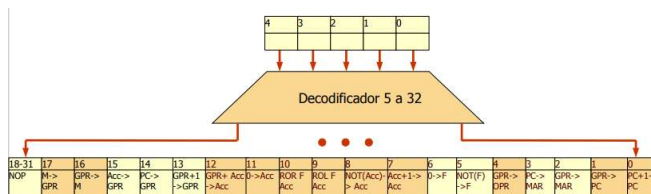
wtf algoritmos de reemplazo??

El formato de las microinstrucciones:

- Formato no codificado

| | | | | | | | | | | | | | | | | | |
|--------|--------|----------|---------|------------|--------------|-----------|-----------|--------------|------------|------|-----------|----------|---------|----------|---------|----------|---|
| 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| M->GPR | GPR->M | Acc->GPR | PC->GPR | GPR+1->GPR | GPR+Acc->Acc | ROR F Acc | ROL F Acc | NOT(Acc) Acc | Acc+1->Acc | D->F | NOT(F) >F | GPR->DPR | PC->MAR | GPR->MAR | GPR->PC | PC+1->PC | |

- Formato completamente codificado



- Formato codificado por trozos
- Formato con solapamientos
- Nanoprogramación

Tipos de microprogramación

Microprogramación horizontal

(Excel)

- Formato no codificado.
- Microinstrucciones muy largas.
- Alto grado de paralelismo.
- Alto consumo de memoria de control.
- Más rápidas.

Microprogramación vertical

(Como si en el excel pusieran decodificadores)

- Formato codificado. (Por trozos o solapamiento)
- Microinstrucciones cortas
- Bajo paralelismo.
- Menor consumo de memoria de control.

- Menos rápidas.

La nanoprogramación es un tipo de microprogramación vertical cuyo objetivo es reducir la memoria de control

Microprogramación

Secuencia dentro de un microprograma:

Tras la ejecución de cada microinstrucción se tiene que determinar qué nueva microinstrucción se ha de ejecutar.

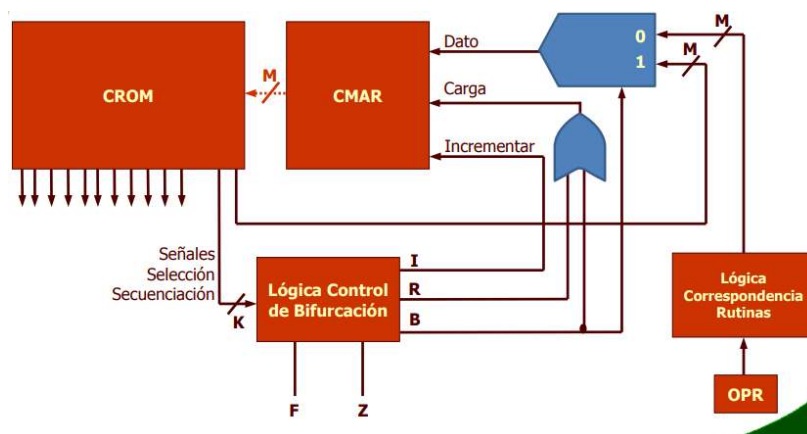
Hay 3 posibilidades:

1. Pasar a ejecutar la microinstrucción que se encuentra en la dirección CROM consecutiva (Incremento)
2. Pasar a ejecutar la microinstrucción que se encuentre en una dirección CROM especificada explícitamente (Bifurcación)
3. Pasar a ejecutar la primera microinstrucción asociada al ciclo de ejecución de una instrucción determinada (Carga de Rutina)

Además, la determinación de la siguiente microinstrucción puede ser:

Incondicional o Condicional. (Es condicional cuando se tienen en cuenta F o Zsc...)

El secuenciamiento se consigue variando la dirección contenida en el registro de dirección de acceso a la CROM (CMAR: Control Memory Access Register)

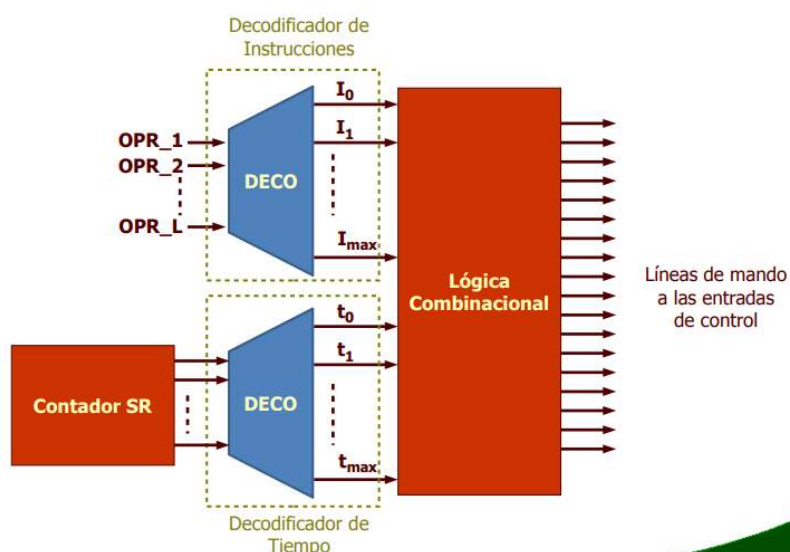


rep basico:

| S2 | S1 | S0 | Z | F | I | B | R | Comentario |
|----|----|----|---|---|---|---|---|----------------------------|
| 0 | 0 | 0 | X | X | 1 | 0 | 0 | Incremento |
| 0 | 0 | 1 | X | X | 0 | 1 | 0 | Bifurcación incondicional |
| 0 | 1 | 0 | 0 | X | 0 | 1 | 0 | Salta si Z=0 |
| 0 | 1 | 0 | 1 | X | 1 | 0 | 0 | Incrementa si Z=1 |
| 0 | 1 | 1 | X | 0 | 1 | 0 | 0 | Incrementa si F=0 |
| 0 | 1 | 1 | X | 1 | 0 | 1 | 0 | Salta si F=1 |
| 1 | X | X | X | X | 0 | 0 | 1 | Carga Rutina incondicional |

Cableado

Sicome (ult pract de mierda que hice desde sicome pq sino no cargaba)



En el diseño cableado mediante decodificadores de tiempo e instrucciones, la secuenciación se consigue modificando el contador de tiempo SR que tiene cuenta ascendente, carga paralela y puesta a 0.

Cableado mediante registros de Desplazamiento.

También llamada mediante “trenes de biestables”.

Implementación sencilla.

Alta flexibilidad.

Facilidad de análisis.

Bastante rápida.

Minimiza la necesidad de lógica adicional.

Alto uso de biestables (muy por encima del óptimo)

Cableado mediante Circuito Secuencial Síncrono.

Tedioso y rígido.

Difícil de analizar.

Difícil de detectar errores.

Garantiza la ejecución más rápida de todos los métodos.

Minimiza el número de biestables.

Microprogramado mediante ROM de Control.

Muy fácil de diseñar.

Extremadamente flexible.

Importante

Puedo eliminar la publi de este documento con 1 coin

¿Cómo consigo coins? → Plan Turbo: barato
→ Planes pro: más coins

pierdo
espacio



Fácil de analizar.

Es el mecanismo más lento.

Costosa en circuitería (ROM, Registros, Lógica combinatorial, etc).

Cableado mediante Decodificadores de Tiempo e Instrucción.

Sencillo de diseñar.

Bastante flexible.

Fácil de analizar.

Tiene una ejecución bastante rápida.

Bastante lógica combinatorial.

Necesito
concentración

ali ali ooh
esto con 1 coin me
lo quito yo...

WUOLAH