



TEMA VII

MICROPROGRAMACIÓN



7 Microprogramación

7.1 Modelo original de Wilkes

7.2 Estructura de una unidad de control microprogramada

7.2.1 Conceptos básicos

7.2.2 Elementos de una unidad de control microprogramada

7.2.3 Secuenciamiento de las microinstrucciones

7.2.4 Organización de la memoria de control

7.2.5 Ejecución de las microinstrucciones

7.3 Representación de los microprogramas

7.3.1 Ejemplo: Desarrollo de un sencillo microprograma

7.4 Ejemplo de diseño microprogramado: multiplicador binario

7.5 Unidad de control microprogramada de un computador

7.5.1 Estructura de SIMPLE2

7.5.2 Formato de la microinstrucción de *ucmP*

7.5.3 Fases en la ejecución de una microinstrucción

7.5.4 Inicialización de SIMPLE2

7.5.5 Búsqueda y decodificación de las instrucciones máquina

7.5.6 Ejemplo: Microprogramación de un repertorio de instrucciones máquina de SIMPLE2

7.1 Modelo original de Wilkes

- Se dice que un procesador está microprogramado cuando las informaciones generadas por la unidad de control se almacenan en una memoria (memoria de control)
 - Cada posición de la memoria de control contiene microinstrucciones
 - Un microprograma es una serie ordenada de microinstrucciones
- Cada instrucción máquina:
 - Un conjunto de microórdenes que especifican las transferencias de información entre los diferentes componentes de la parte operacional
 - Mediante la activación de los puntos de control, controlamos el flujo de datos
- Microprogramación:
 - Método sistemático para diseñar la unidad de control de cualquier sistema digital

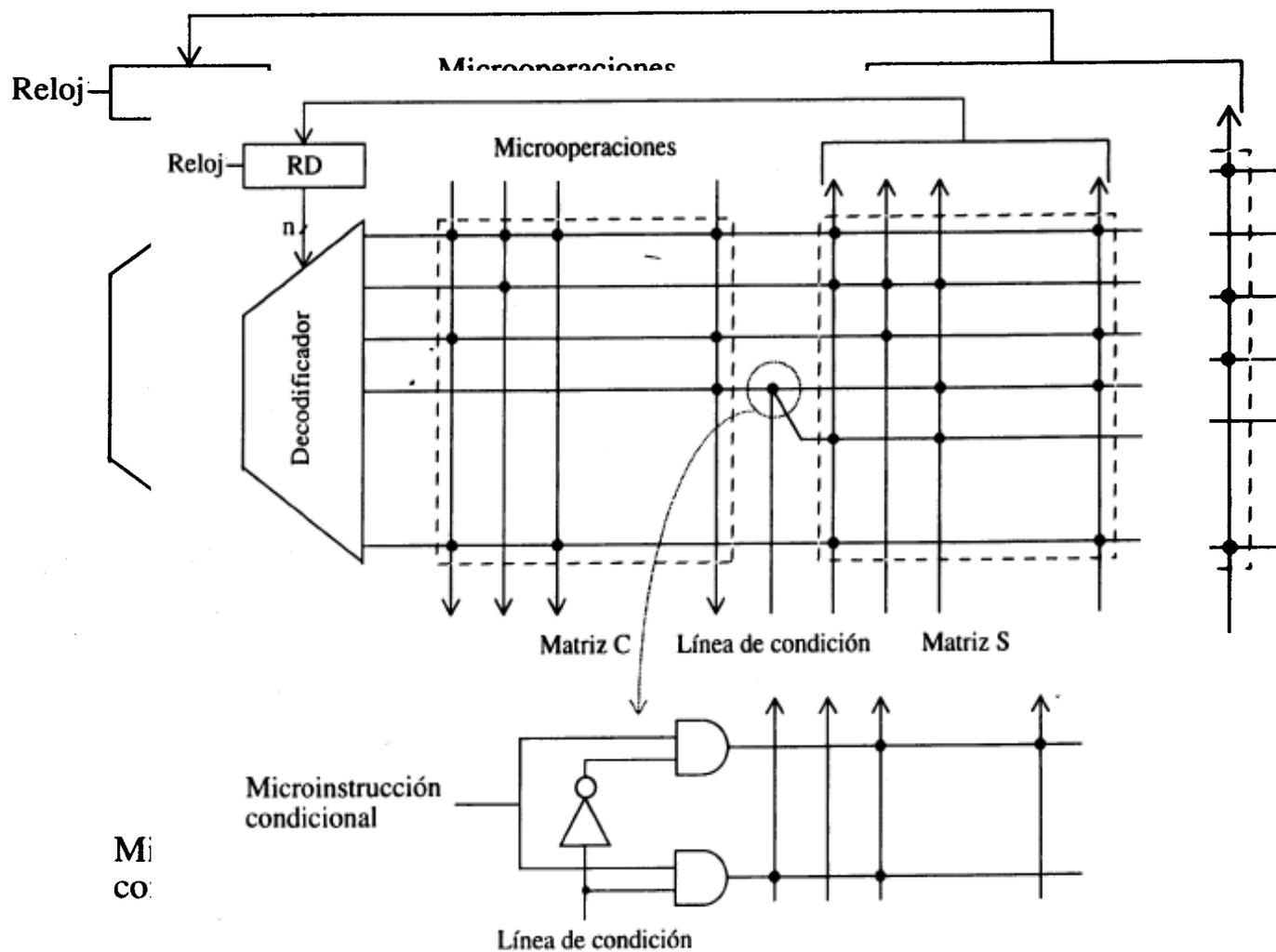


Figura 6-12 Modelo original propuesto por Wilkes

Figura 6-12 Modelo original propuesto por Wilkes

7.2 Estructura de una unidad de control microprogramada

■ 7.2.1 Conceptos básicos

□ Tareas de la U.C. μ -programada

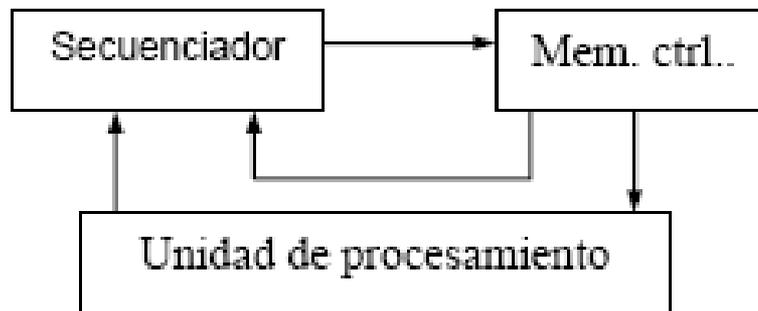
- Secuenciamiento de las μ -instrucciones
 - Obtener la próxima microinstrucción
- Ejecución de las μ -instrucción
 - Generar las señales de control

□ Una microinstrucción

- Es un conjunto de micro-órdenes que se pueden ejecutar de forma simultánea y que está contenida en una palabra de la memoria de control

7.2.2 Elementos de una unidad de control microprogramada

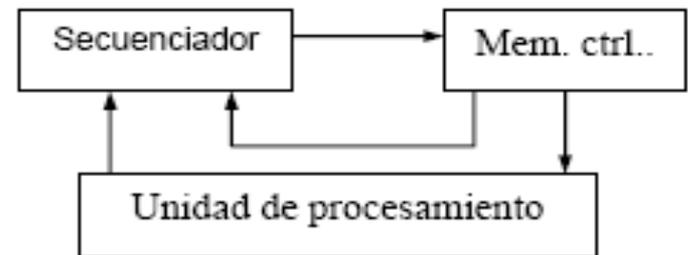
- Memoria de control
 - Contiene el micro programa
 - Micro ordenes
 - Siguiente micro orden
- Unidad de secuenciamiento
 - Contiene la dirección de la micro orden actual
 - Calcula la siguiente micro orden



7.2.2 Elementos de una unidad de control microprogramada

■ **Funcionamiento:**

- 1.- La instrucción entra en IR y tras decodificarse carga en RDC la dirección de la 1ª μ -instrucción.
- 2.- RDC apunta a la memoria de control que saca el dato a RMC
- 3.- RMC contiene 3 campos
 - Señales de control al bus del sistema.
 - Señales de control internas a CPU
 - Próxima dirección de μ -instrucción
- 4.- El secuenciador carga la próxima instrucción en RDC y continúa la secuencia.
 - Opciones:
 - Bif. a siguiente instrucción:
 - $RDC \leftarrow RDC+1$
 - Bifurcación: $RDC \leftarrow RMC$ [dirección]
 - $RDC \leftarrow \text{Función (IR[cod. Oper.]}$



el control

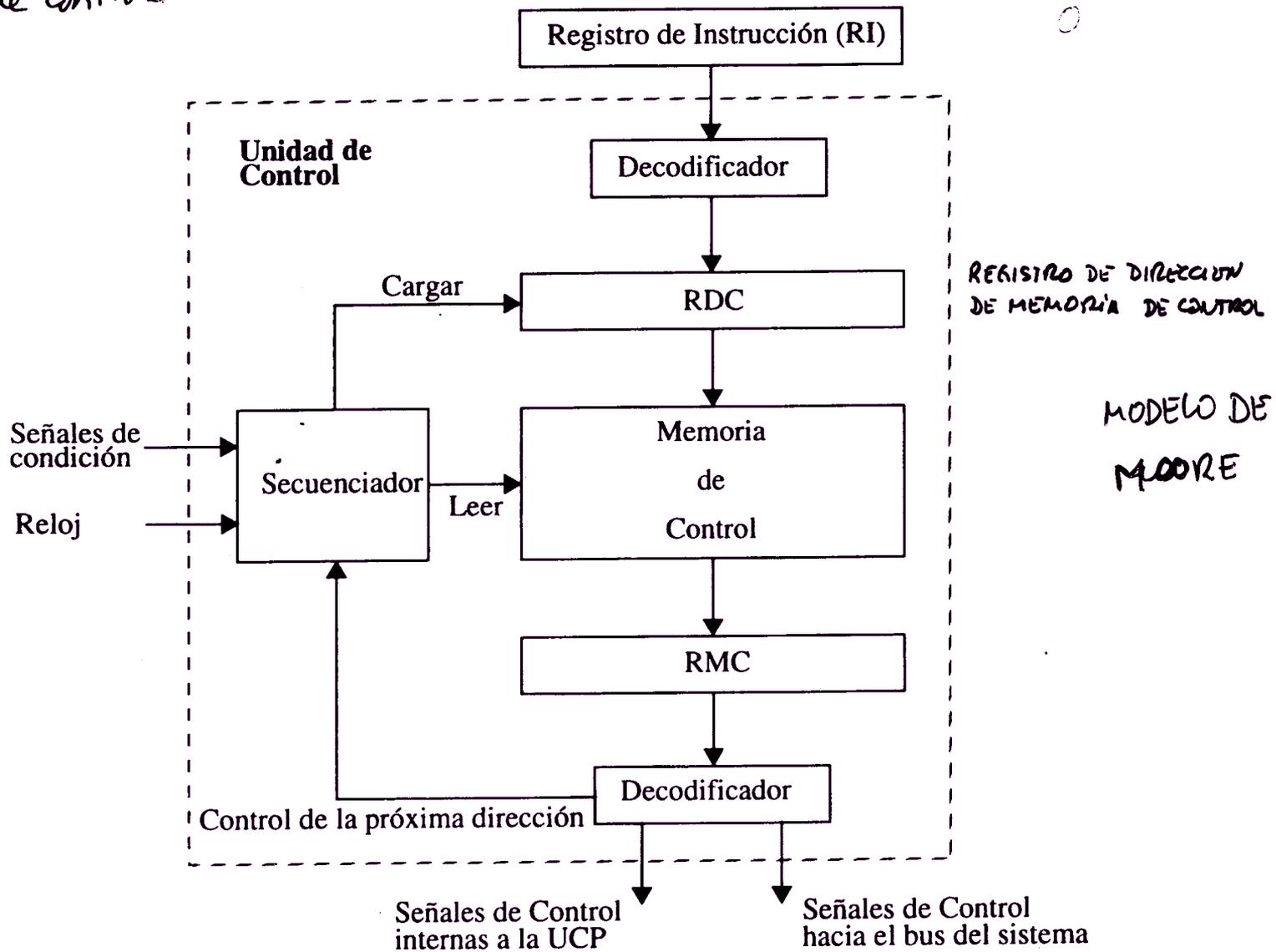
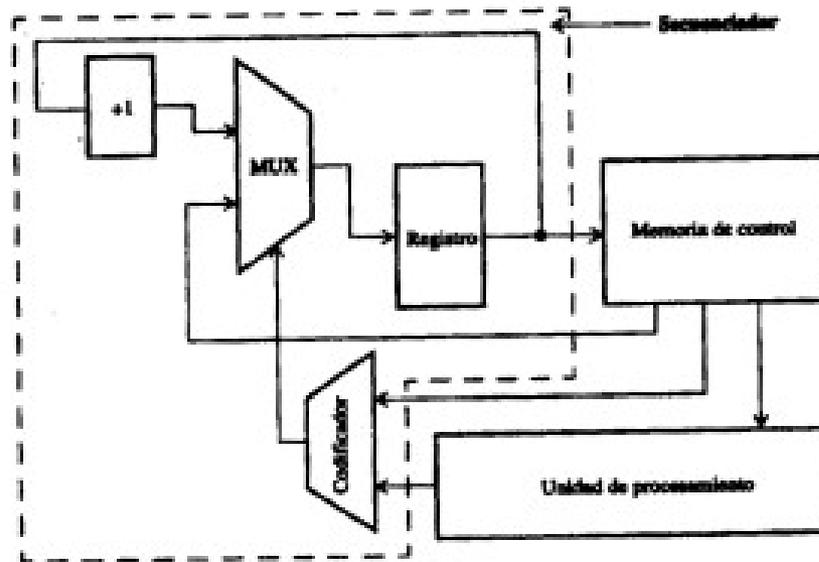


Figura 6-17 Funcionamiento de una Unidad de Control microprogramada

7.2.3 Secuenciamiento de las microinstrucciones

- Se encarga de secuenciar la ejecución de las μ -instrucciones
- Saltos en un secuenciador
 - Salto condicional
 - Incremento de la dirección actual



Secuenciador que permite el salto condicional entre microinstrucciones

Secuenciación

- Para determinar la siguiente microinstrucción:
 - Implícito
 - Diferencia mediante un campo si se trata de una μ -instrucción de control o de salto.
 - Direccionamiento explícito
 - Especificado en un campo separado la dirección de salto

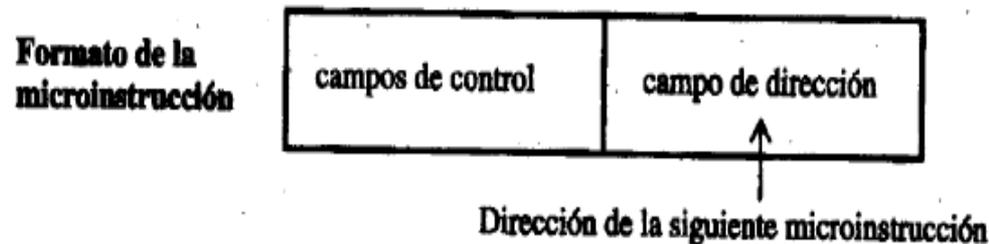
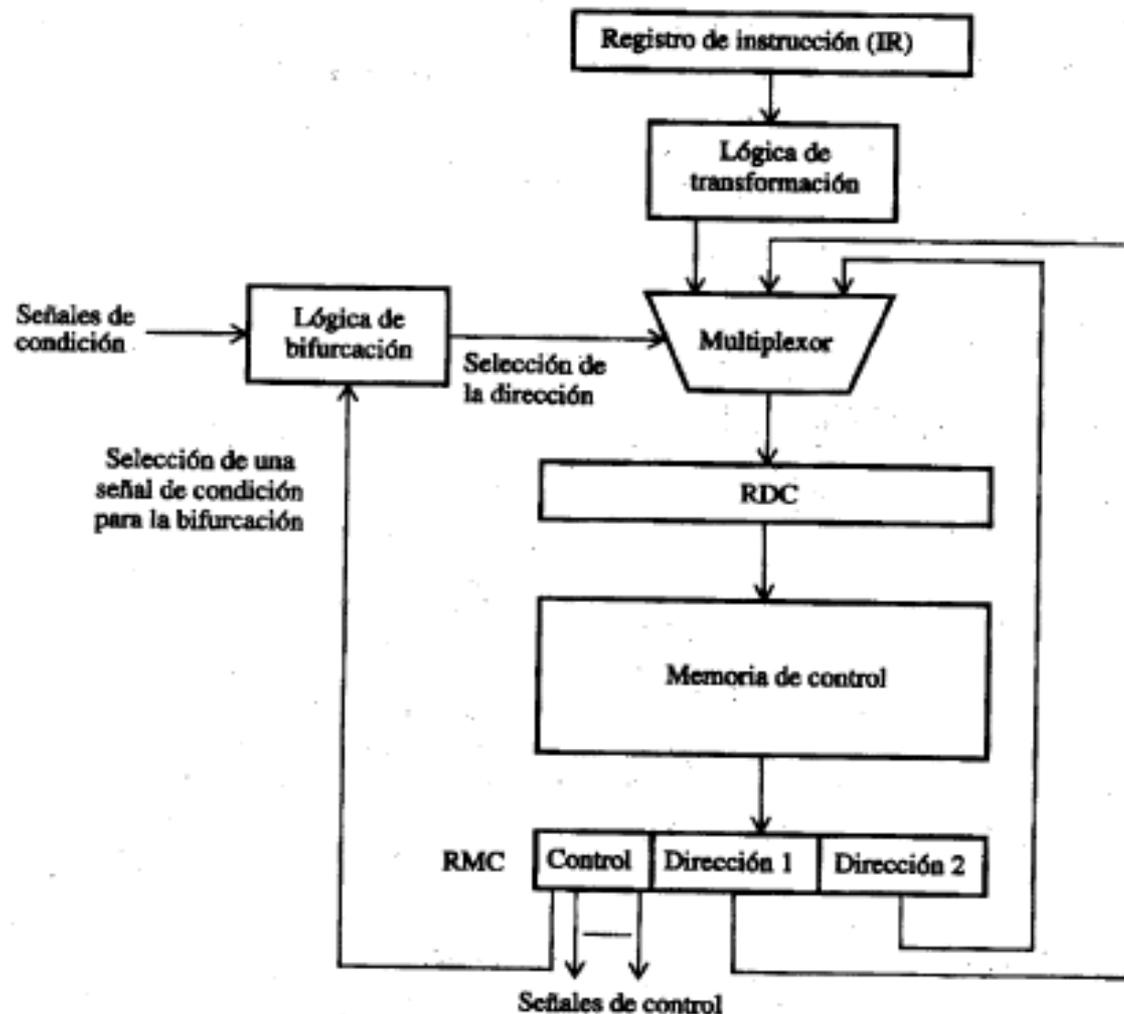


Figura 7.7: Direccionamiento explícito

Direccionamiento explícito con dos direcciones por micro-instrucción



Direccionamiento explícito con una dirección por micro-instrucción

una sola dirección

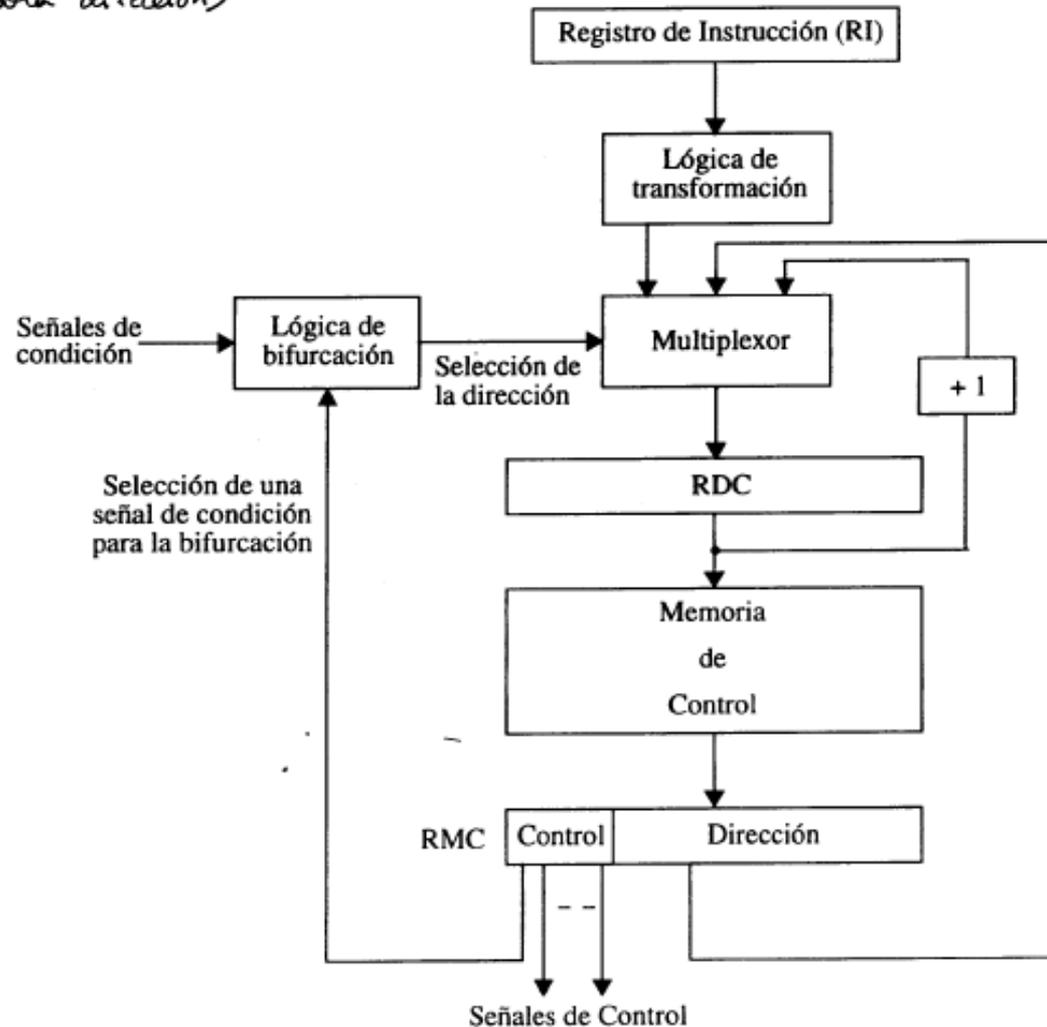
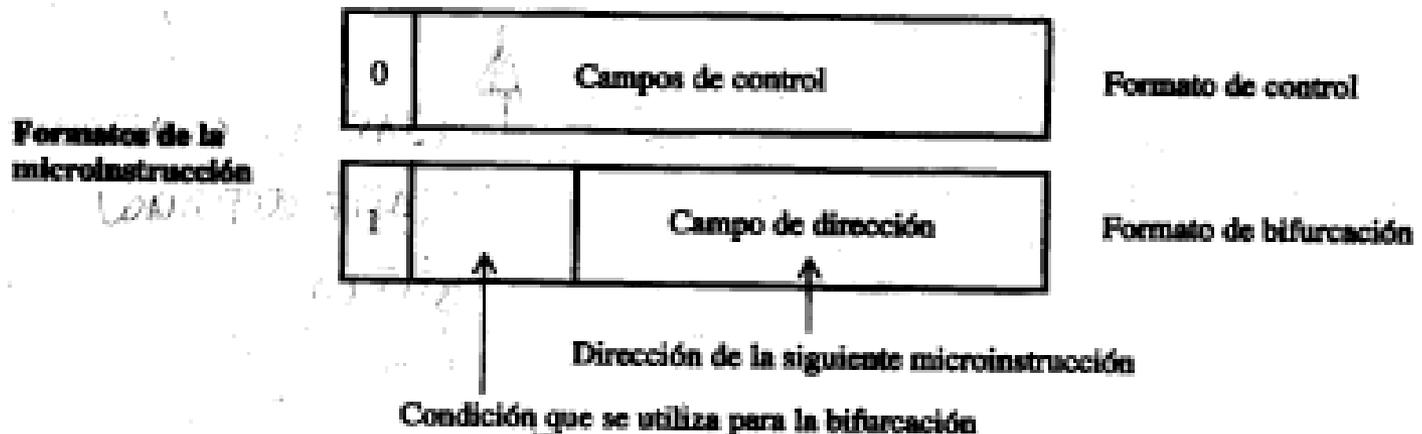
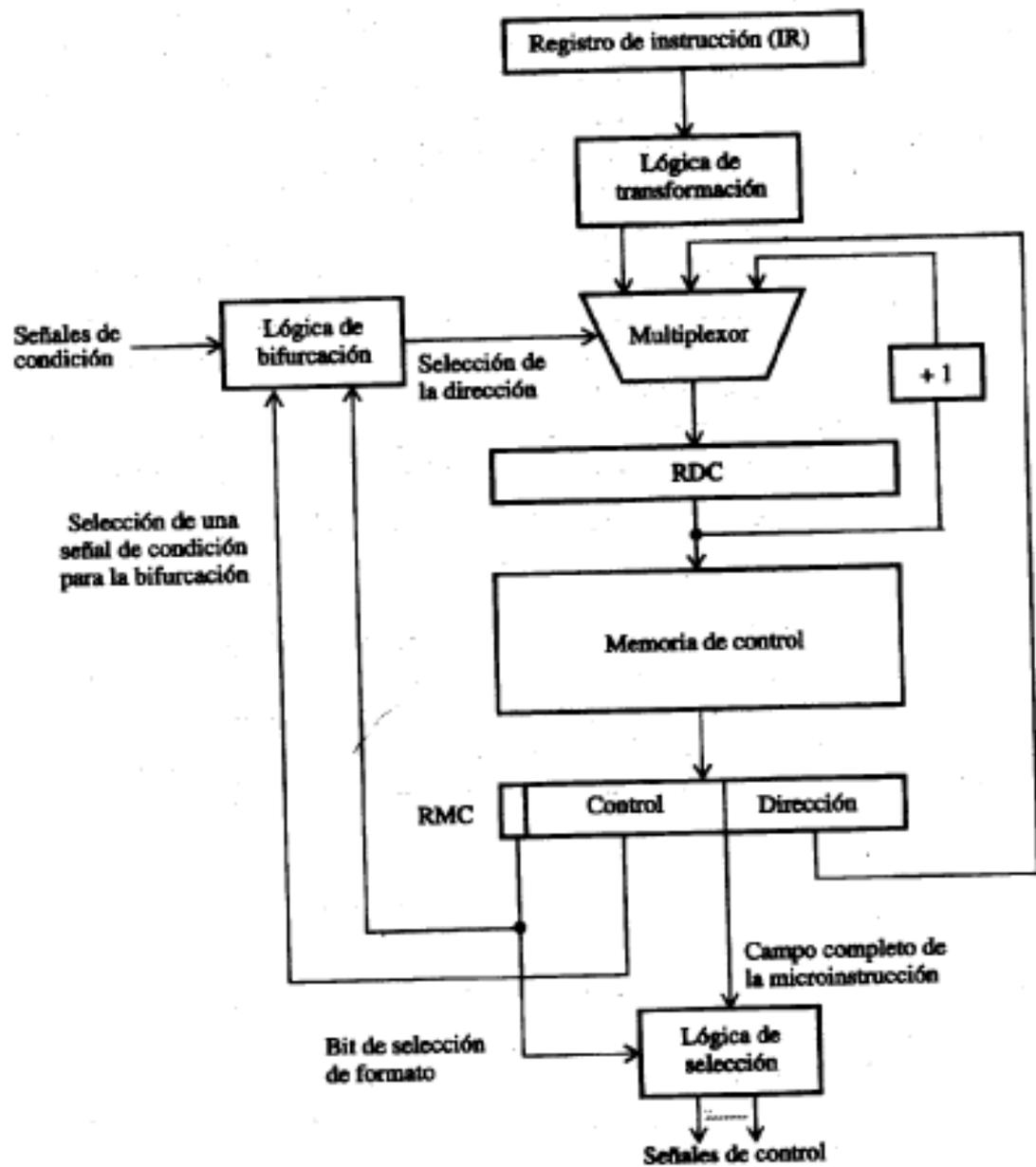


Figura 6-20 Unidad de Control microprogramada con una dirección por microinstrucción

Direccionamiento implícito

- Reduce el tamaño de las microinstrucciones
- Requiere dos tipos (clases) de microinstrucciones
 - De control
 - De bifurcación
 - Reduce el tamaño de las microinstrucciones
 - Si cumple la condición bifurca a la dirección especificada
 - Si no cumple la condición, o es una instrucción de control, sigue en la siguiente instrucción





7.2.4 Organización de la memoria de control

- Una micro-instrucción tiene dos partes:
 - Bits de la dirección
 - Bits correspondientes a las señales de control
 - Vamos a ver como podemos representar los bits de las señales de control
 - Dos opciones
 - Microprogramación horizontal
 - Microprogramación vertical
- Microprogramación horizontal
 - Las señales de control se representan Sin codificar
 - Un bit por cada señal de control que exista en la unidad de control
 - La mayoría estarán a cero (solo activa las que afecten a esa micro-instrucción)
 - Tamaño muy grande

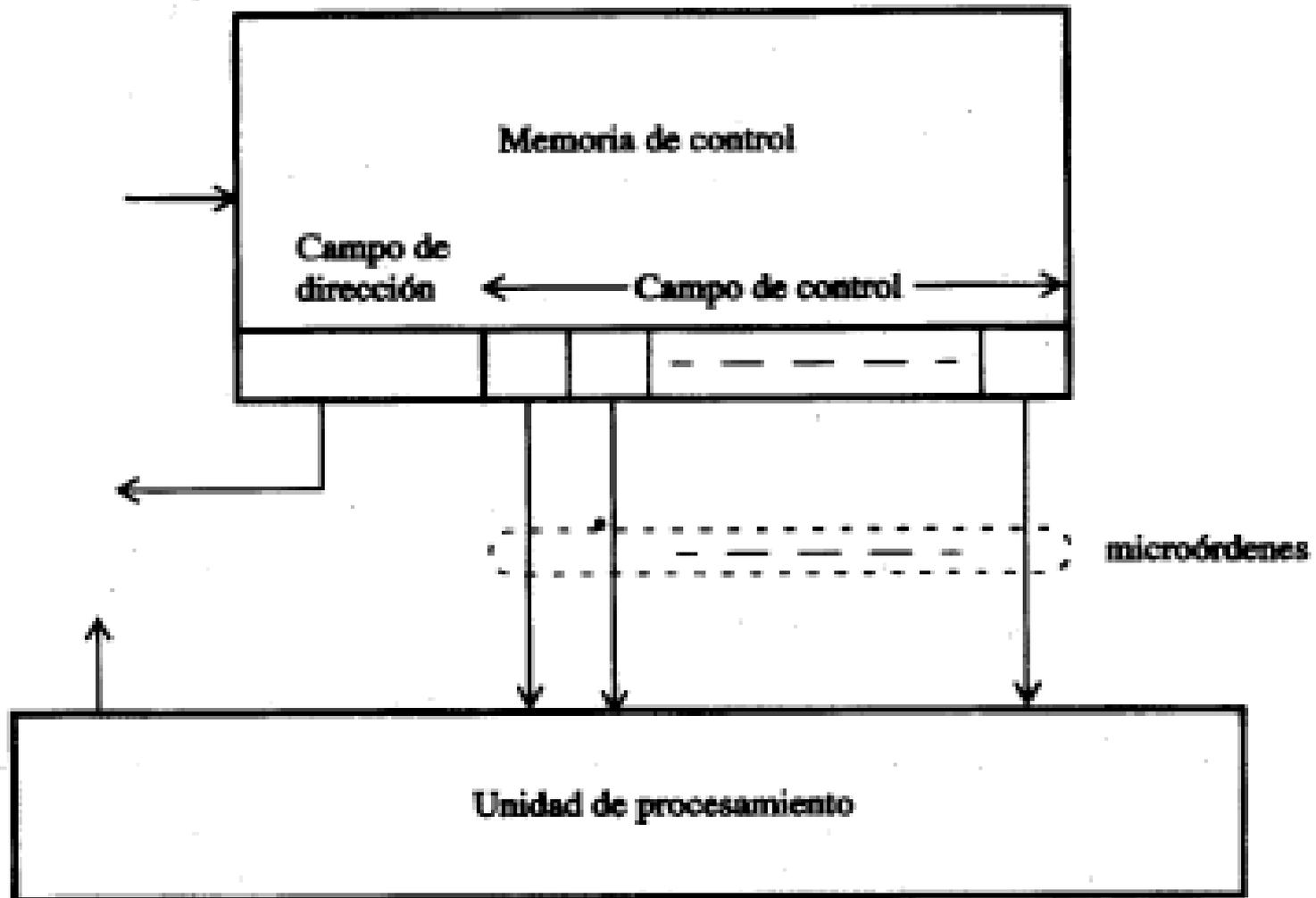


Figura 7.12: Formato horizontal de una microinstrucción

Microprogramación vertical

- Se codifican las señales (se numeran y se identifican por su número)
- El campo de control está dividido en subcampos
- Cada uno de los subcampos controla operadores excluyentes entre sí (no se pueden producir en el mismo instante)
 - Los subcampos de control contiene la identificación de la señal de control
 - Con subcampos de j bits se pueden especificar $2^j - 1$ señales. Una codificación está reservada para cuando ninguna está activa
- El formato vertical es más lento porque precisa la decodificación de las señales

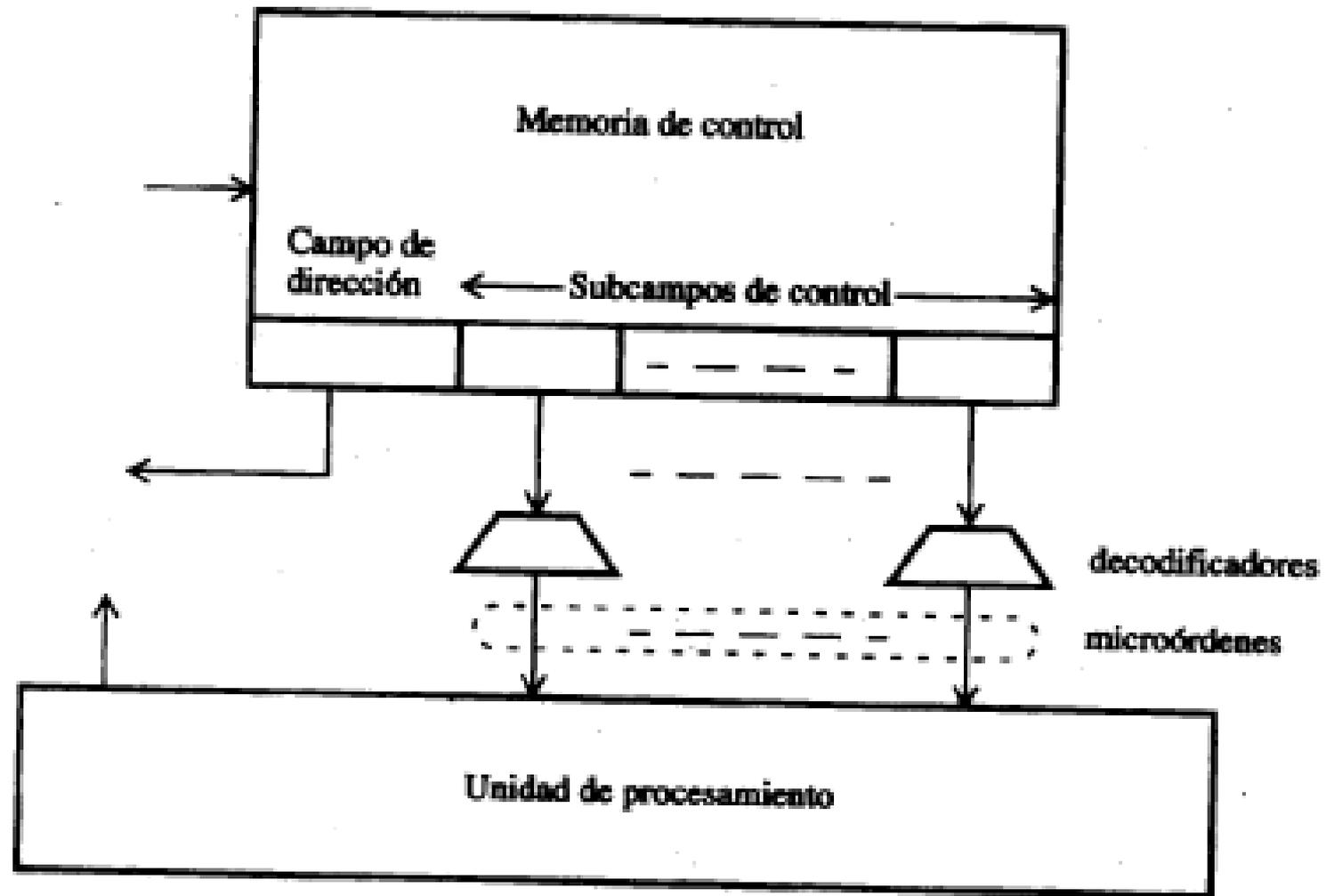


Figura 7.13: Formato vertical de una microinstrucción codificada por subcampos

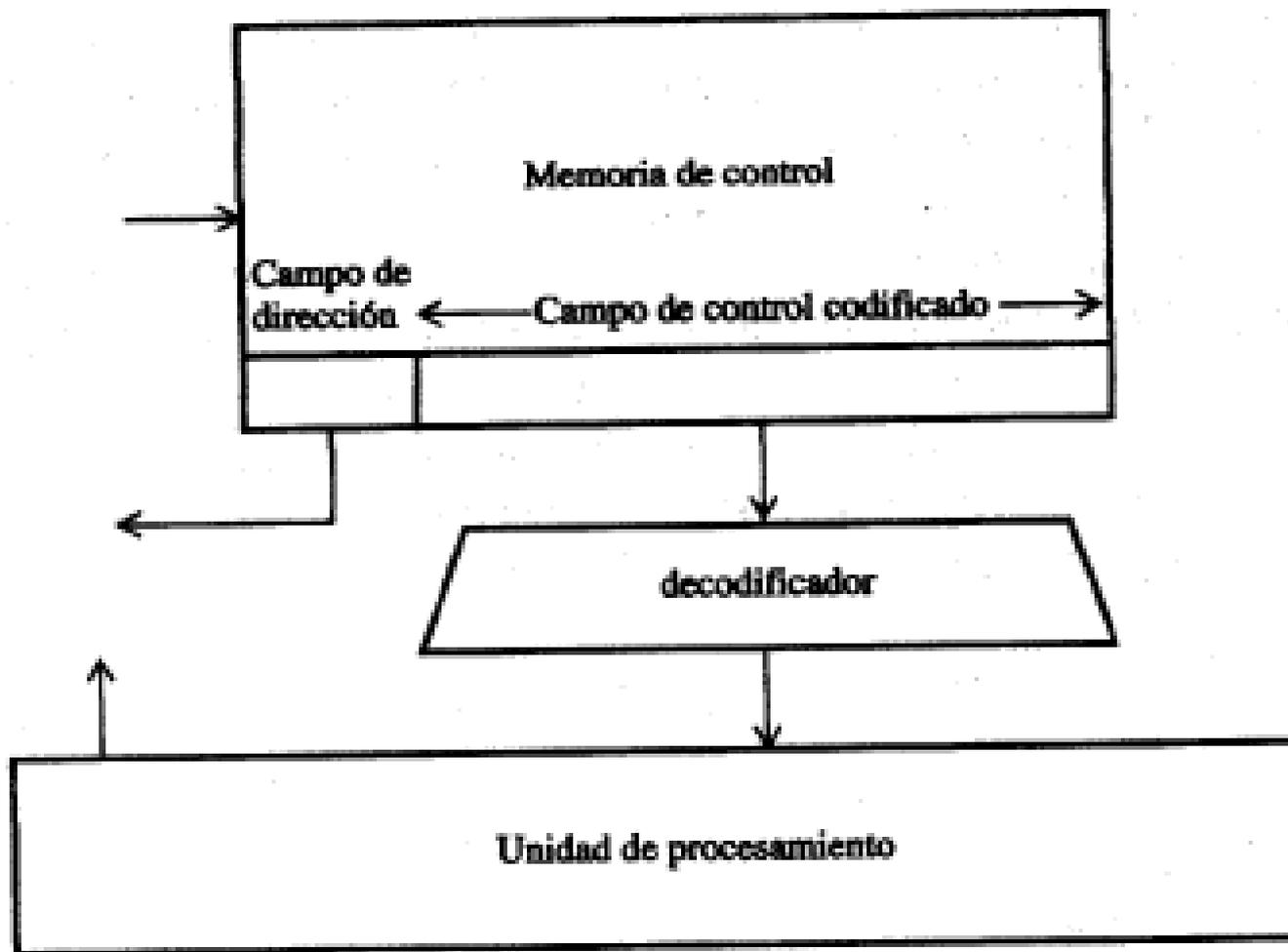


Figura 7.14: Formato vertical de una microinstrucción totalmente codificada

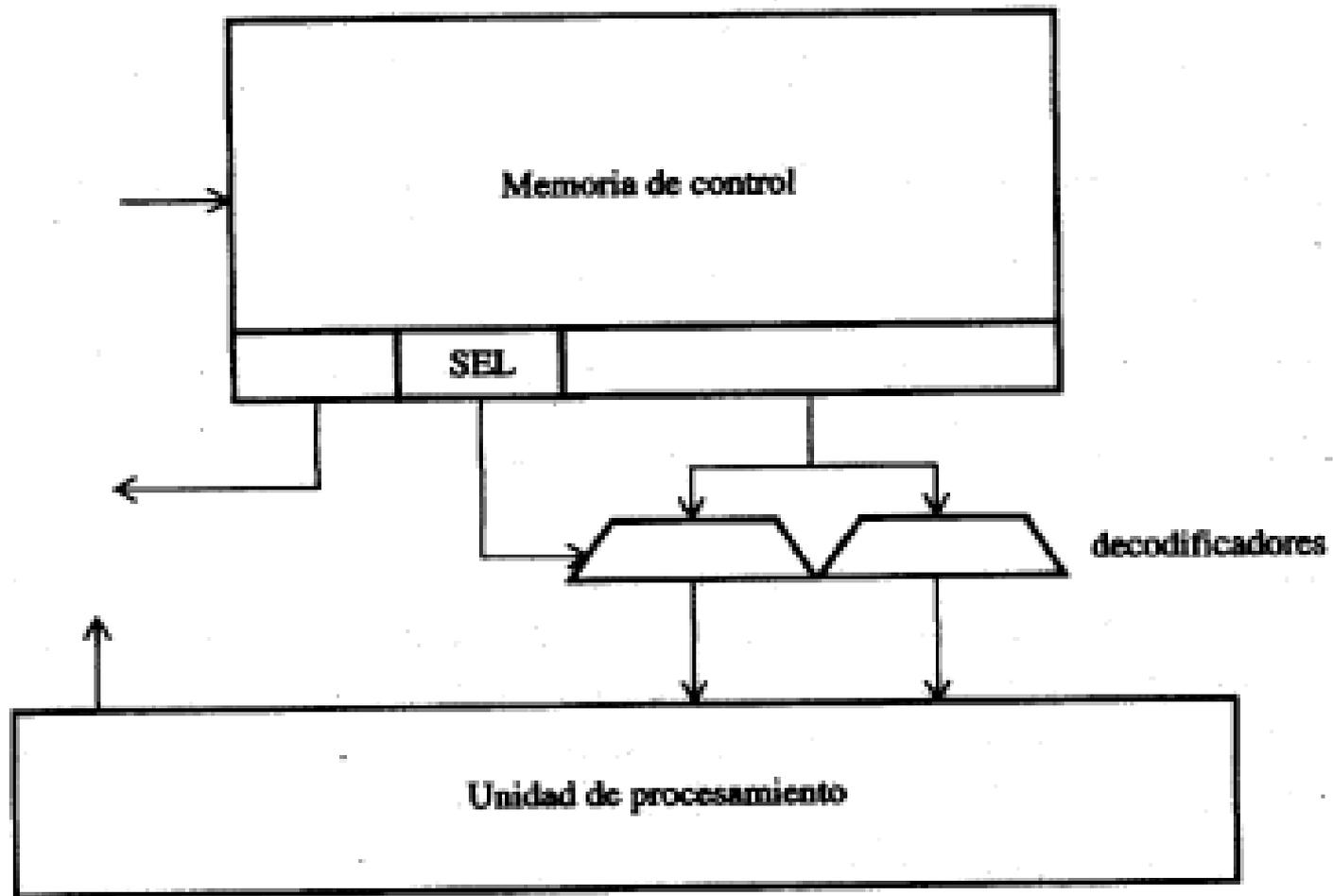


Figura 7.15: Formato vertical con campo de selección del tipo de microinstrucción

AVISO

- ***Mucho cuidado porque en el caso del formato vertical, al codificar las señales siempre hay que tener en cuenta una codificación de más, la correspondiente a que todas las señales de control estén desactivadas.***
- ***El caso del formato horizontal, esto no se considera porque cuando se precisa que no esté activada ninguna de las señales, basta con ponerlas todas a 0.***

7.2.5 Ejecución de las microinstrucciones

■ Fases

1. Búsqueda de la μ -instrucción
2. Decodificación de los campos de la μ -instrucción
3. Ejecución de las μ -operaciones
4. Cálculo de la dirección de la próxima μ -instrucción

■ Tipos de ejecuciones de las microinstrucciones

Monofásicas

- Las μ -operaciones se pueden ejecutar al mismo tiempo

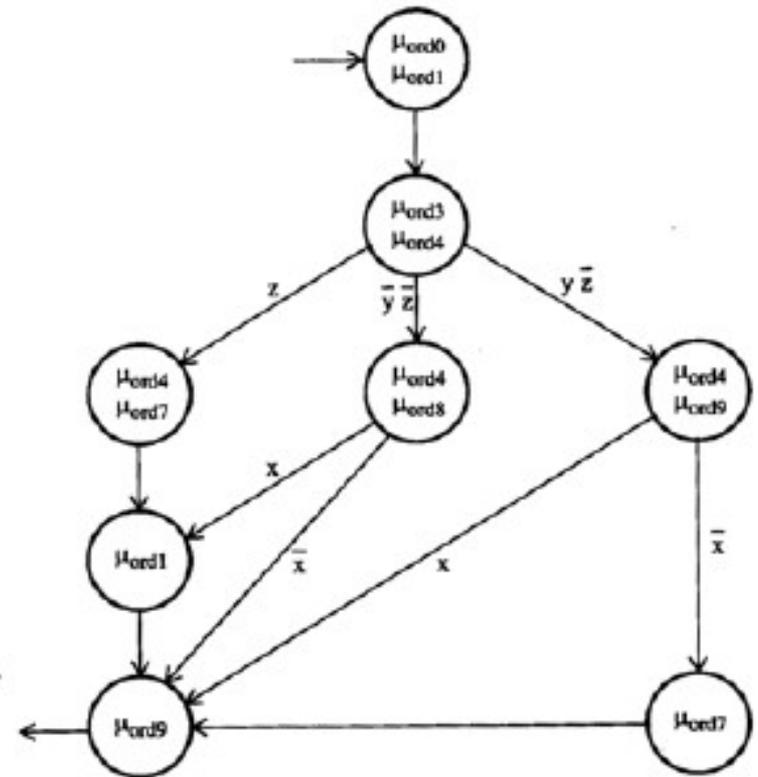
Polifásicas

- Los campos se utilizan de forma escalonada en el tiempo de ejecución

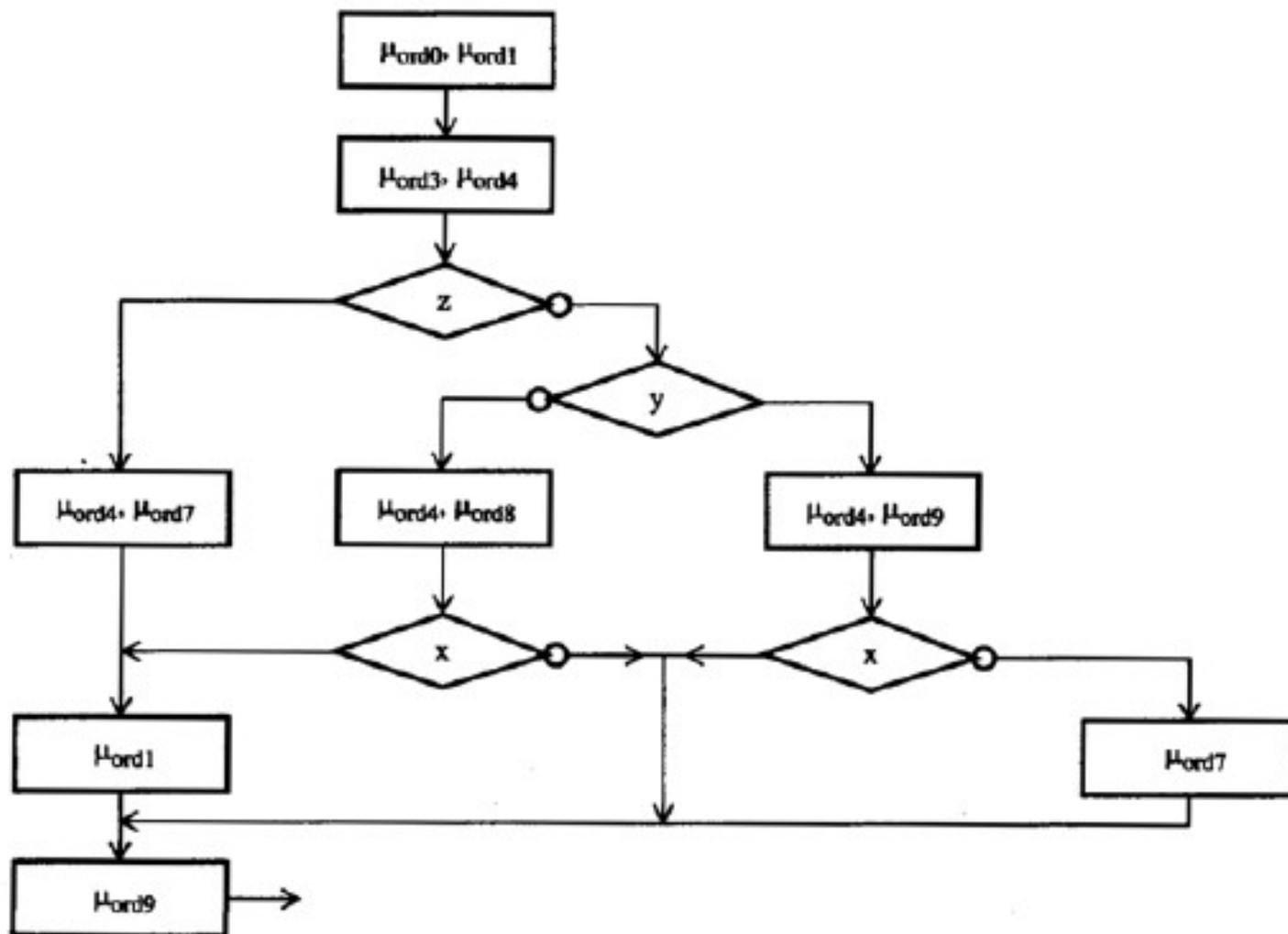
7.3 Representación de los microprogramas

■ Grafos orientado

- Los Nodos representan
 - Las μ -operaciones ejecutadas por la unidad de procesamiento
- Las Líneas representan
 - Las Secuencias realizada por la Unidad de Control



Representación de un microprograma como un grafo orientado



Representación de un microprograma como un organigrama o diagrama de flujo

7.3.1 Ejemplo:

Desarrollo de un sencillo microprograma

- Diseño de la Unidad de Control de un procesador sencillo con las siguientes especificaciones:
- Elementos:
 - $M[MAR]$ → memoria RAM de 256×8 bits
 - T → Registro de 8 bits
 - R1 → Registro de 8 bits
 - MAR → Registro de dirección de Memoria de 8 bits
 - DECR (-1) → operador de Decremento en 1
 - INCR (+1) → operador de Incremento en 1
 - SUM → Sumador

Microórdenes de la ruta de datos		
Microorden	Significado	Acción
$\mu\text{ord}0$	Decrementa T	$T \leftarrow T-1$
$\mu\text{ord}1$	Transfiere T+1 a MAR	$\text{MAR} \leftarrow T+1$
$\mu\text{ord}2$	Transfiere T a MAR	$\text{MAR} \leftarrow T$
$\mu\text{ord}3$	Suma R1 a M[MAR]	$M[\text{MAR}] \leftarrow M[\text{MAR}]+R1$
$\mu\text{ord}4$	Transfiere M[MAR] a R1	$R1 \leftarrow M[\text{MAR}]$

Microórdenes para el secuenciamiento		
Microorden	Significado	Acción
$\mu\text{ord}5$	Secuencia normal	$\text{RDC} \leftarrow \text{RDC}+1$
$\mu\text{ord}6$	Salto condicional a $\mu l(x)$	$\text{RDC} \leftarrow x$

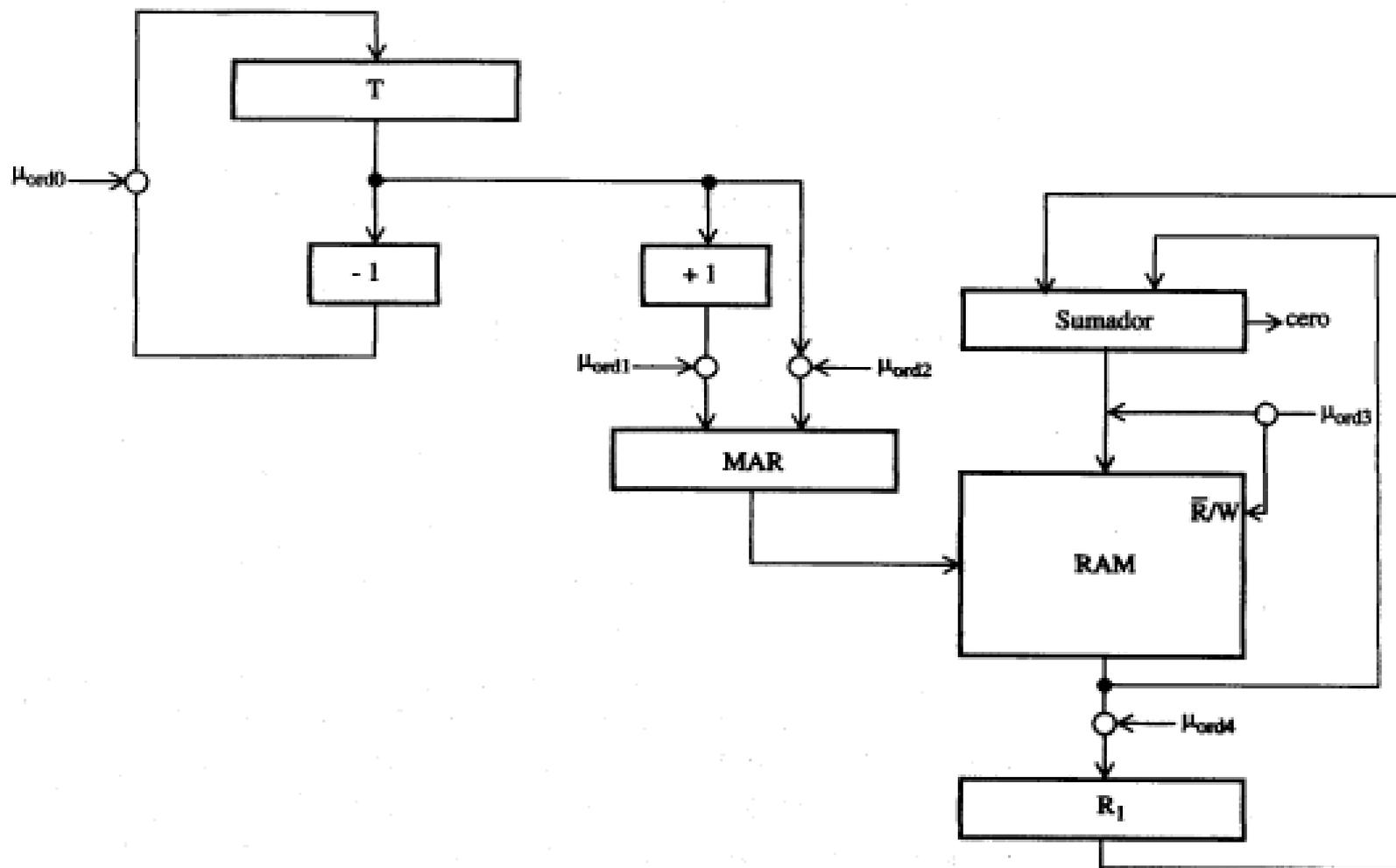


Figura 7.21: Ruta de datos del procesador del ejemplo 7.3.1

Objetivo del microprograma

- $T \leftarrow T-1$
 - $M[T] \leftarrow M[T] + M[T+1]$
 - **If** $M(T) = 0$ **then** $RDC \leftarrow x$ **else** $RDC \leftarrow RDC+1$
- Unidad de control

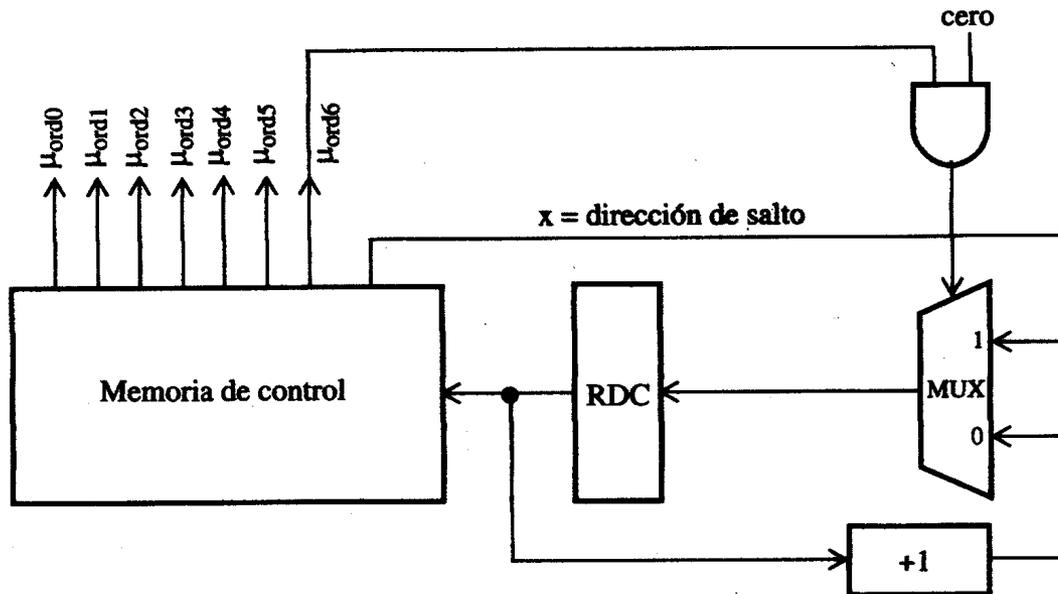


Figura 7.22: Unidad de control del procesador del ejemplo 7.3.1

Primera solución: Formato horizontal

- Cada señal de control, tiene su propio bit en la memoria :
 - Mayor anchura de la memoria **15 bits**

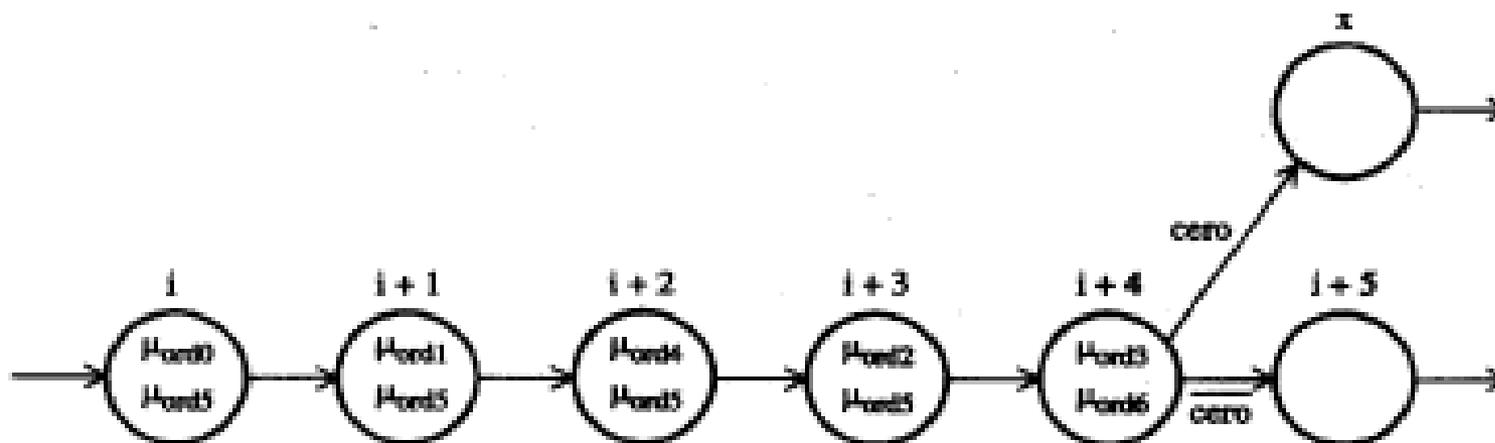


Figura 7.23: Grafo orientado del microprograma para la primera solución del ejemplo 7.3.1

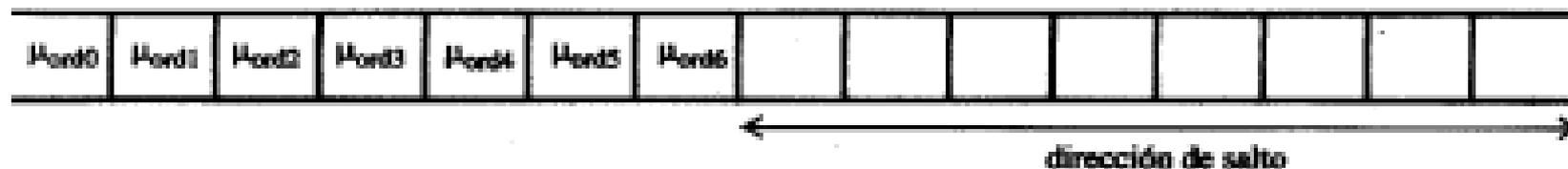


Figura 7.24: Formato de la microinstrucción para la primera solución del ejemplo 7.3.1

En la Tabla 7.3 se muestra el código binario del microprograma correspondiente a esta solución que está compuesto de 5 palabras de 15 bits.

Microprograma	Codificación binaria														
	Pos0	Pos1	Pos2	Pos3	Pos4	Pos5	Pos6	dirección de salto							
$\mu l(i)$	1	0	0	0	0	1	0	-	-	-	-	-	-	-	-
$\mu l(i+1)$	0	1	0	0	0	1	0	-	-	-	-	-	-	-	-
$\mu l(i+2)$	0	0	0	0	1	1	0	-	-	-	-	-	-	-	-
$\mu l(i+3)$	0	0	1	0	0	1	0	-	-	-	-	-	-	-	-
$\mu l(i+4)$	0	0	0	1	0	0	1								

Tabla 7.3: Código binario del microprograma para la primera solución del ejemplo 7.3.1

Segunda solución: Formato vertical

- El campo de control está dividido en subcampos
- Cada subcampo controla un conjunto de operadores
 - Estos operadores son excluyentes entre sí (no se pueden producir en el mismo instante)
 - Está codificado indicando la señal a controlar
 - Se precisa un decodificador por subcampo

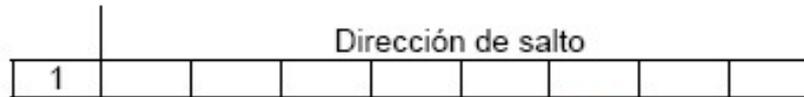
Tercera solución: Direccionamiento implícito microprogramación vertical

Bit diferenciador

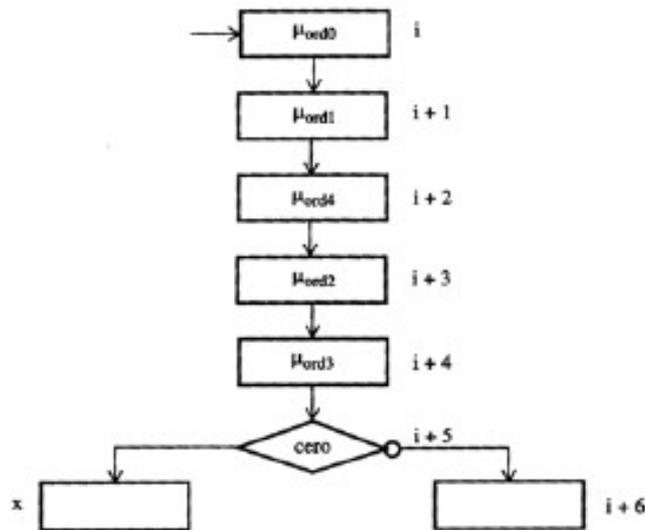
Formato de ejecución



Formato de test



8 bits pq la mem es de 256*8



Organigrama equivalente al grafo orientado

Código binario						Microprograma				
						OP	μ1	μ2	μ4	
-	-	-	-	-	0	0	0	0	0	μ(i)
-	-	-	-	-	1	0	0	0	0	μ(i+1)
-	-	-	-	-	0	0	1	0	0	μ(i+2)
-	-	-	-	-	0	1	0	0	0	μ(i+3)
-	-	-	-	-	1	1	0	0	0	μ(i+4)
					x				1	μ(i+5)

Código binario del microprograma

Microprograma	Codificación binaria									
	OP	μ_1	μ_2	μ_3						
$\mu I(i)$	0	0	0	0	-	-	-	-	-	
$\mu I(i+1)$	0	0	0	1	-	-	-	-	-	
$\mu I(i+2)$	0	1	0	0	-	-	-	-	-	
$\mu I(i+3)$	0	0	1	0						
$\mu I(i+4)$	0	0	1	1	-	-	-	-	-	
$\mu I(i+4)$	1								x	

Tabla 7.6: Código binario del microprograma para la segunda solución del ejemplo 7.3.1

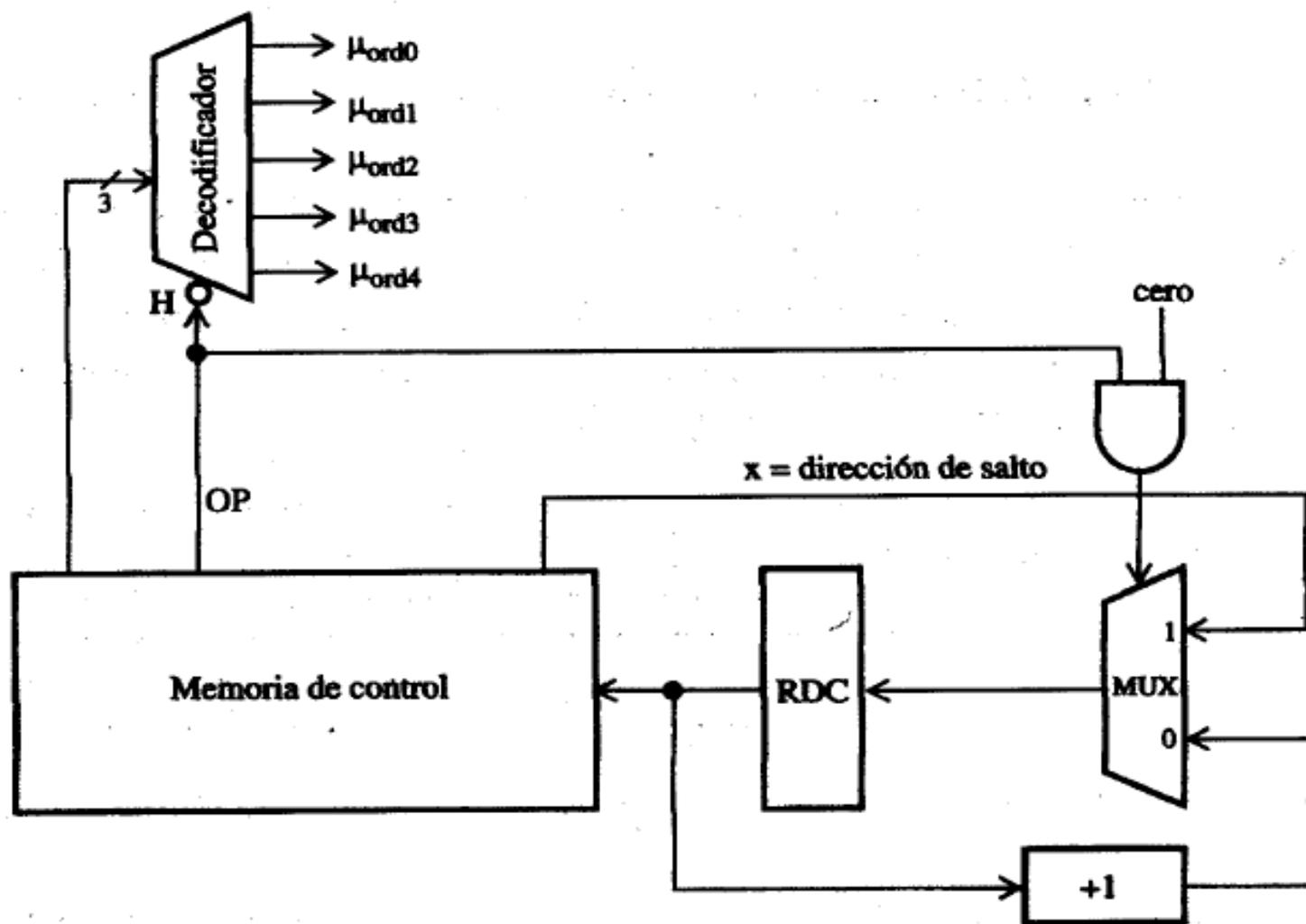


Figura 7.29: Esquema de la unidad de control para la tercera solución del ejemplo 7.3.1

Cuarta solución

- Se basa en reordenar la secuencia de μ -ordenes para realizar el mismo tiempo las que no tengan influencia unas sobre otras.

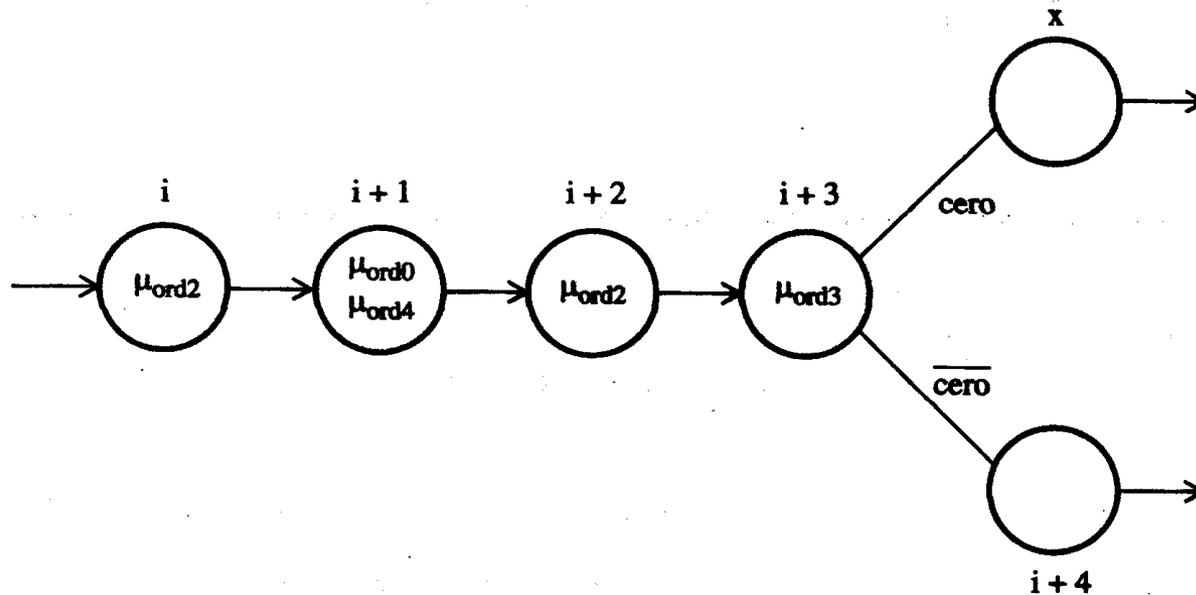


Figura 7.30: Grafo orientado del microprograma para la cuarta solución del ejemplo 7.3.1

7.4 Ejemplo de diseño micro-programado: multiplicador binario

- 1) Inicializar y Borrar_C en el estado S_{00}
- 2) Sumar_Cargar en el estado S_{10}
- 3) Borrar_C y Desplazar_Dec en el estado S_2
- 4) Ninguna señal de control activa en los estados S_0 y S_1

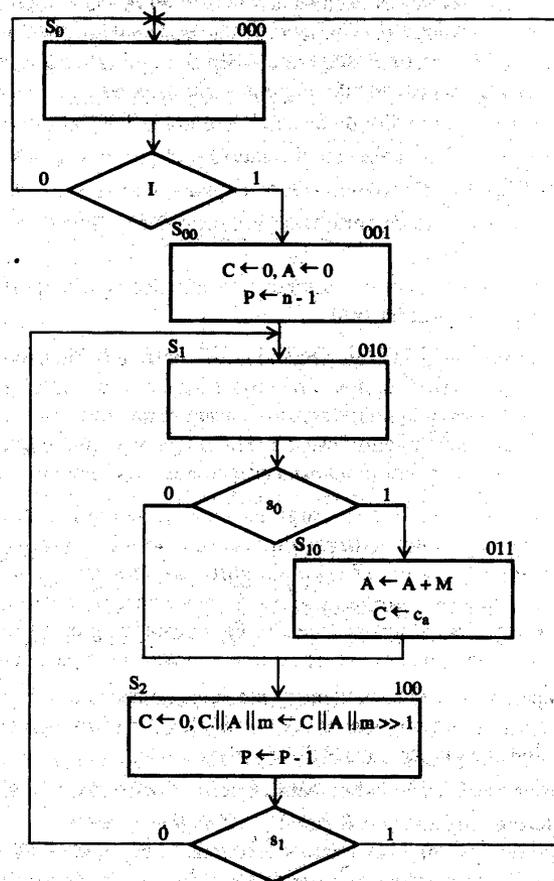


Figura 7.31: Diagrama ASM para el diseño de la unidad de control microprogramada del multiplicador

Borrar_C. Estas señales de control y las transferencias de registros asociadas se muestran en la Tabla 7.7.

Señal de control	Transferencia de registro	Estados en los que la señal está activa	Bit de posición en la μI	Nombre simbólico
Inicializar	$A \leftarrow 0, P \leftarrow n - 1$	S_{00}	0	IN
Sumar_Cargar	$A \leftarrow A + M, C \leftarrow c_a$	S_{10}	1	SC
Borrar_C	$C \leftarrow 0$	S_{00}, S_2	2	BC
Desplazar_Dec	$C \parallel A \parallel_m \leftarrow C \parallel A \parallel_m \gg 1, P \leftarrow P - 1$	S_2	3	DD

Tabla 7.7: Señales de control y transferencias de registros asociadas para el multiplicador binario

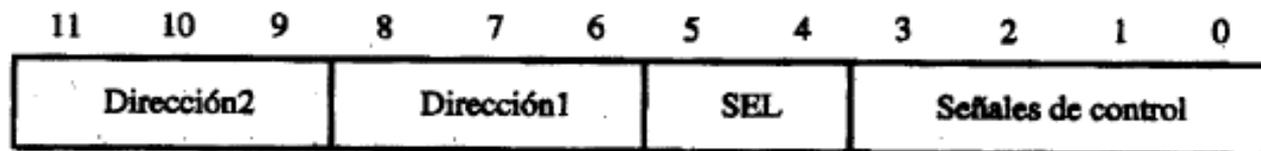


Figura 7.32: Formato de la microinstrucción

SEL		Secuenciamiento de microórdenes
Nombre simbólico	Código binario	
SIG	00	RDC ← Dirección1
DI	01	\bar{I} RDC ← Dirección1 I RDC ← Dirección2
Ds ₀	10	\bar{s}_0 RDC ← Dirección1 s ₀ RDC ← Dirección2
Ds ₁	11	\bar{s}_1 RDC ← Dirección1 s ₁ RDC ← Dirección2

Tabla 7.8: Definición del campo de selección SEL para el secuenciamiento de la unidad de control del multiplicador

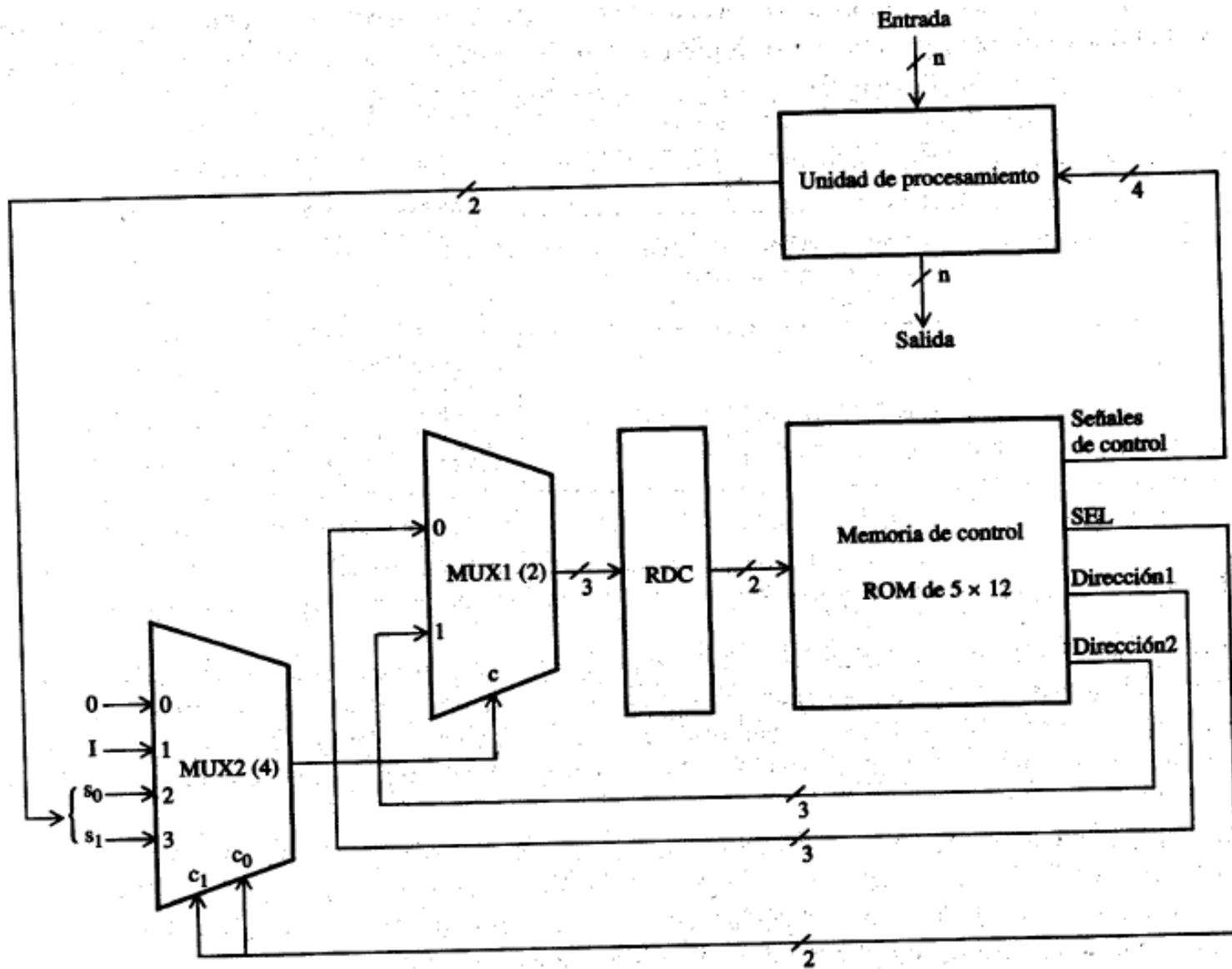


Figura 7.33: Unidad de control microprogramada del multiplicador

Dirección	Sentencias de transferencia simbólica
S_0	$I: RDC \leftarrow S_{00}, \bar{I}: RDC \leftarrow S_0$
S_{00}	$C \leftarrow 0, A \leftarrow 0, P \leftarrow n - 1, RDC \leftarrow S_1$
S_1	$s_0: RDC \leftarrow S_{10}, \bar{s}_0: RDC \leftarrow S_2$
S_{10}	$A \leftarrow A + M, C \leftarrow c_a, RDC \leftarrow S_2$
S_2	$C \leftarrow 0, C \parallel A \parallel_m \leftarrow C \parallel A \parallel_m >> 1, s_1: RDC \leftarrow S_0, \bar{s}_1: RDC \leftarrow S_1, P \leftarrow P - 1$

Tabla 7.9: Microprograma del multiplicador en la notación de transferencia de registro

Dirección	Dirección2	Dirección1	Selección (SEL)	Señales de control
S_0	S_{00}	S_0	DI	Ninguna
S_{00}	---	S_1	SIG	IN, BC
S_1	S_{10}	S_2	Ds_0	Ninguna
S_{10}	---	S_2	SIG	SC
S_2	S_0	S_1	Ds_1	BC, DD

Tabla 7.10: Microprograma simbólico del multiplicador

Dirección	Dirección2	Dirección1	Selección (SEL)	Señales de control
000	001	000	01	0000
001	000	010	00	0101
010	011	100	10	0000
011	000	100	00	0010
100	000	010	11	1100

Tabla 7.11: Código binario del microprograma del multiplicador



7.5 Unidad de control microprogramada de un computador

- Leer

- Una unidad de control micro-programada con direccionamiento implícito tiene una memoria de control con 24 bits de longitud de palabra.
- Si las microinstrucciones emplean 15 bits para los campos de control y el tamaño máximo de la memoria de control de esta Unidad de Control microprogramada es de 2^{20} palabras
- ¿Cuántas condiciones como máximo se pueden evaluar en el caso de microinstrucciones de bifurcación?
 - A) 3
 - B) 16
 - C) 4
 - D) Ninguna de las anteriores.

SOLUCIÓN

- [Ver sección 6-2 del libro de teoría.]
- En el caso de direccionamiento implícito, se definen dos formatos de instrucciones diferenciadas por el primer bit.
- En caso de ser 0 tienen formato de control y se usan 15 bits adicionales para el control, es decir, se usan 16 bits y quedan 8 libres sin usar.
- En el caso de formato de bifurcación (primer bit a 1) se debe evaluar la condición que sea necesaria. Como el número de palabras de 2^{20} , 20 es número de bits necesarios para el campo dirección.
- Por lo tanto, el número de bits utilizados para codificar el número máximo de condiciones de bifurcación será: $24-20-1=3$ bits. Siendo por lo tanto 8 el número máximo de condiciones.
- Respuesta: D

- 
- El tamaño máximo de la memoria de control de un computador es de 2^{20} palabras.
 - La parte de control del formato de una microinstrucción (de la unidad de control microprogramada) de este computador emplea 8 bits para seleccionar independientemente las microoperaciones que se llevan a cabo.
 - Esta unidad de control tiene un campo para indicar el tipo de bifurcación a realizar (condicional, incondicional o no bifurcar) y un campo para seleccionar una única señal de condición de las 32 señales disponibles.
 - ¿Cuál es el tamaño de las microinstrucciones en bits?
 - A) 20
 - B) 35
 - C) 52
 - D) Ninguna de las anteriores

SOLUCIÓN

- [Ver problema 7.2 del libro de problemas].
- Como el tamaño máximo de la memoria de control es de 20^{20} palabras, se necesitan reservar 20 bits para la dirección.
- Si la microinstrucción tiene X bits de longitud, de los cuales 8 bits se emplean para seleccionar las microoperaciones,
- 2 bits para codificar el tipo de bifurcación y
- 5 ($2^5=32$) para seleccionar una única señal de condición,
- $X=20+8+2+5=35$ De esta forma, el tamaño de las microinstrucciones es de 35 bits.
- Respuesta: B (35)

- Un computador usa un formato de microinstrucción mixto, parte horizontal y parte vertical.
- La parte con formato horizontal de codificación tiene una longitud de k bits y
- La parte con formato vertical de codificación posee m campos codificados de n bits cada uno.
- ¿Cuál es el máximo número de señales de control que pueden usarse en este computador?
 - A) $k + n \times 2^m$
 - B) $k + n^m$
 - C) $k + n \times (2^m - 1)$
 - D) Ninguna de las anteriores.

Solución

- La organización de la memoria de control se explica en el Apartado 7.2.4 del texto de teoría.
- La parte con *formato horizontal* de codificación tiene una longitud de k bits, por tanto pueden usarse k señales de control.
- La parte con *formato vertical* de codificación posee m campos codificados de n bits cada uno.
- Cada campo de n bits codifica $2^n - 1$ señales.
- Así pues, el número máximo de señales de control que puede usarse en este computador es $k + m \times (2^n - 1)$.
- Respuesta:
 - D (Ninguna de las anteriores)

- En una Unidad de Control microprogramada con formato de microinstrucciones *vertical*,
- Un subcampo que deba especificar 16 señales de control
- Habrá de tener una anchura mínima de:

- A) 4 bits.
- B) 5 bits.
- C) 16 bits.
- D) Ninguna de las anteriores

■ SOLUCIÓN

- Un subcampo codificado de j bits puede especificar a lo sumo $2^j - 1$ señales de control.
- Para gobernar 16 señales de control no es suficiente con 4 bits ya que $2^4 - 1 = 15 < 16$, pero si son suficientes 5 bits: $2^5 - 1 = 31$
- Respuesta correcta:
 - B(5 bits)

- Un computador usa el formato vertical de codificación de instrucciones para parte de las señales de control y el formato horizontal para k señales de control.
- El formato vertical posee n campos codificados de m bits cada uno.
- ¿Cuál es el máximo número de señales de control que pueden usarse en este computador?
 - A) $k + nx2^m$.
 - B) $k + n^m$.
 - C) $k + nx(2^m-1)$.
 - D) Ninguna de las anteriores.
- **SOLUCION**
 - Respuesta correcta: C ($k + nx(2^m-1)$).

- Un computador microprogramado tiene un total de 132 señales de control.
- De ellas, un grupo de 16 son mutuamente excluyentes entre sí y otro grupo de 30 son mutuamente excluyentes entre sí.
- Indique si las siguientes afirmaciones son verdaderas:
- I. Utilizando formato vertical, el tamaño de los subcampos codificados sería de 4 y 5 bits, respectivamente.
- II. Al existir señales mutuamente excluyentes no puede utilizarse el formato vertical de microinstrucciones.
 - A) I: sí, II: sí.
 - B) I: sí, II: no.
 - C) I: no, II: sí.
 - D) I: no, II: no.
- **SOLUCIÓN**
 - La afirmación I es falsa ya que un subcampo codificado de i bits puede codificar a lo sumo $2^i - 1$ señales de control diferentes. Por este motivo harían falta dos subcampos de 5 bits cada uno.
 - La afirmación II es falsa ya que el formato vertical de microinstrucciones saca partido precisamente del hecho de que existan señales de control mutuamente excluyentes.
 - Respuesta correcta:
 - D (I: no; II: no).

- Una Unidad de Control microprogramada con direccionamiento explícito con dos direcciones por microinstrucción, tiene una memoria de control con 35 bits de longitud de palabra.
- Si las microinstrucciones emplean 15 bits para los campos de control, el número máximo de palabras de la memoria de control de esta Unidad de Control microprogramada es de:
 - A) 2^{10} palabras
 - B) 2^{20} palabras
 - C) 2^{17} palabras
 - D) Ninguna de las anteriores

■ SOLUCIÓN

- [Ver apartado 7.2.3 del texto base de teoría y los problemas 7-1 y 7-2 del libro de problemas].
- De los 35 bits, como 15 son para los campos de control, quedan 20 bits para las dos direcciones, 10 para cada una de ellas.
- Por tanto, el tamaño máximo de la memoria de control es de 2^{10} palabras.
- Respuesta:
 - A (2^{10} palabras)