

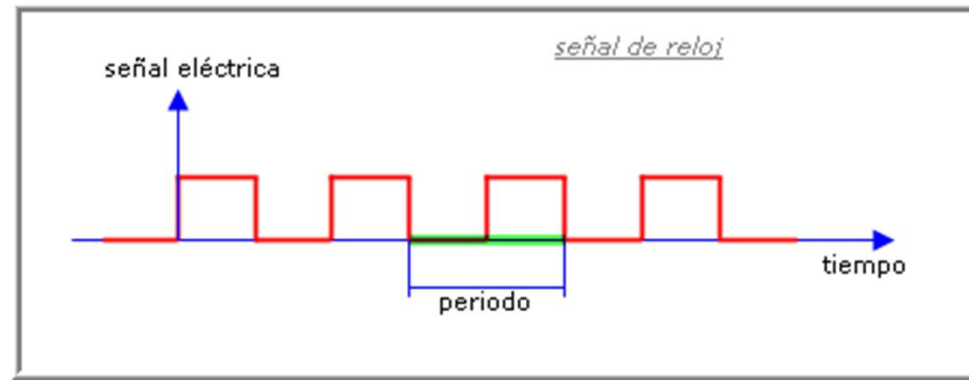


2.2. Unidad de control

- Controla la ejecución de las instrucciones de los programas.
- Consta de:
 - **CP** y **RI**, se pueden considerar como componentes de la UC.
 - **Reloj**,
 - Proporciona una sucesión de impulsos eléctricos o ciclos
 - Intervalos constantes que marcan los instantes en que comienzan los pasos de cada instrucción.
 - Circuito **Decodificador**, que se encarga de
 - Extraer el código de operación de la instrucción del RI
 - Lo analiza
 - Informar al secuenciador de los pasos para ejecutar la instrucción.
 - Circuito **Secuenciador** que
 - Genera órdenes muy elementales (microórdenes) que, sincronizadas por los impulsos de reloj, van ejecutando poco a poco la instrucción cargada en el RI.
 - Estas microórdenes las comunica al resto de los componentes del ordenador a través del bus de control.

3.2. Unidad de control

RELOJ

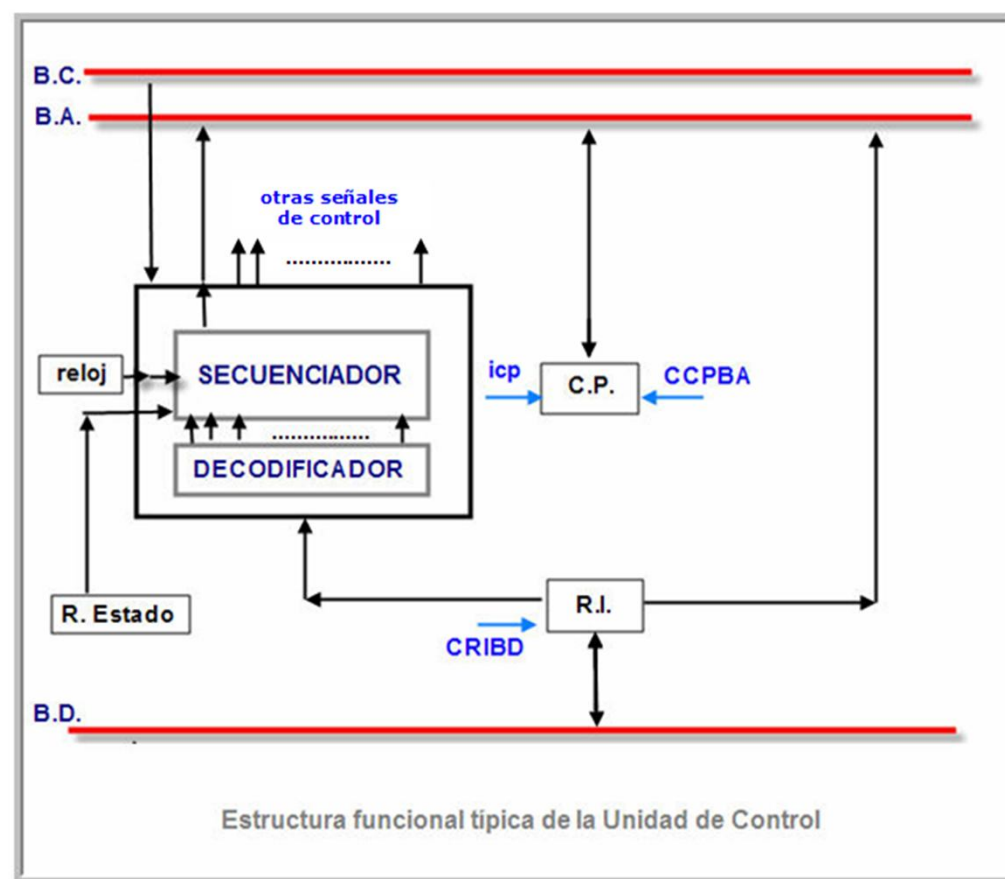


El reloj es un circuito oscilador que genera autónomamente (es decir, sin necesidad de ninguna entrada) **una señal en forma de pulsos** como la que aparece en la figura.

Proporciona una sucesión de impulsos eléctricos o **ciclos** a intervalos constantes (frecuencia constante), que **marcan los instantes en que han de comenzar** los distintos pasos de que consta cada instrucción



3.2. Unidad de control ...



Componentes internos de la UC.

Actividad

- ¿Dónde se encuentran ubicados los siguientes elementos?. Relaciona cada elemento de la columna de la izquierda con la descripción o descripciones correspondientes de la columna de la derecha.

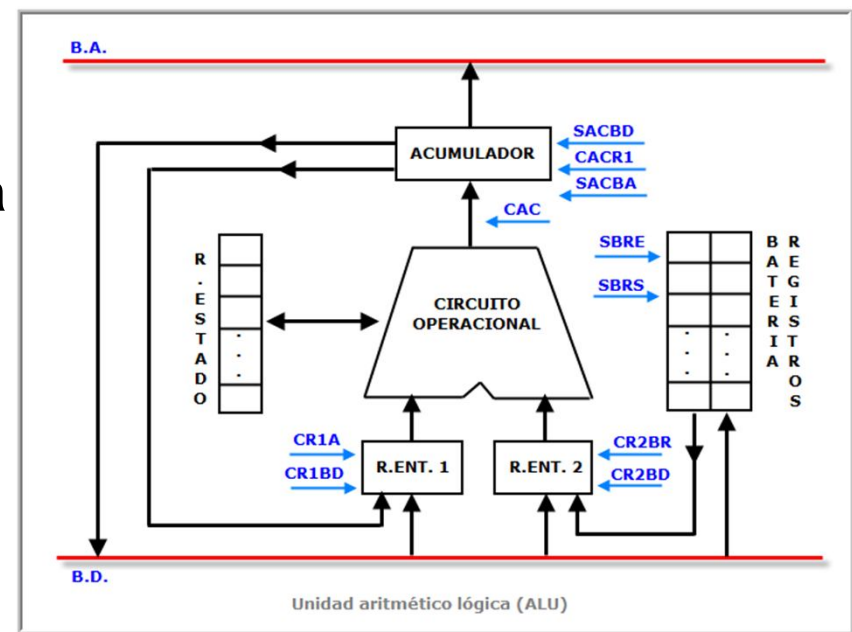


ELEMENTO	DESCRIPCIÓN
Decodificador	Proporciona una sucesión de ciclos
Contador de programa	Ubica la dirección de memoria de la siguiente instrucción a ejecutar
Registro de instrucción	Obtiene y analiza el código de operación de la instrucción del registro de instrucción
Reloj	Genera órdenes muy elementales o microórdenes
Secuenciador	Contiene la instrucción que se está ejecutando.
	Informa al secuenciador de los pasos para la ejecución de la instrucción.
	Crea una sucesión de impulsos eléctricos constantes que marcan el instante de comienzo de cada instrucción



3.3. Unidad Arimético-Lógica

- Su función es realizar operaciones:
 - **Aritméticas**: Suma, resta, multiplicación división, etc
 - **Lógicas**: NOT, AND, OR, XOR, comparaciones, etc
 - **Otras**: desplazamientos de bits a dcha o izda, etc
- En función del tipo de operación (unaria o binaria), utilizara 1 o 2 registros de entrada para los operandos
- El resultado de la operación lo almacena en el Acumulador.
- Algunas operaciones también modificaran algunos bits del FLAGS.





3.3. Unidad Arimético-Lógica

- Con la evolución de las CPUs, se integró en la ALU la **Unidad de coma Flotante (FPU- Floating Point Unit)**, también llamado coprocesador matemático, que realiza operaciones aritméticas con n^0 reales en coma flotante mucho mas rápido que la ALU tradicional.



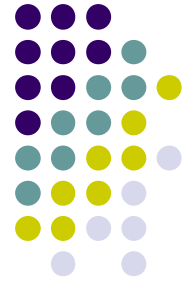
Actividad

- Describe detalladamente los pasos que se realizan en la ALU para calcular la suma de dos valores teniendo en cuenta que el resultado es una dirección



4. Memoria principal

- Está compuesta por dos tipos de memoria:
 - **ROM (Read Only Memory):**
 - Memoria de solo lectura
 - No volátil
 - Contiene el software básico (BIOS) para poder cargar el SS.OO desde los periféricos de E/S a la RAM.
 - El SS.OO a su vez se encargará de cargar los programas y datos de usuario en la RAM desde la E/S.
 - **RAM (Random Access Memory):**
 - Memoria de acceso aleatorio.
 - Volátil
 - Almacena tanto los programas (conjunto de instrucciones) como los datos que manejan estos programas **en ejecución**.



4. Memoria principal...

- La memoria está compuesta de celdas, casillas, o posiciones de memoria de un determinado número de bits. Normalmente un byte. **Los bytes se agrupan en palabras.**
- Cada celda de memoria tiene:
 - **Una dirección:** número que identifica unívocamente a la casilla de memoria. Se utilizan direcciones para **palabras**
 - Si la 1ª dirección es la 0 y la palabra es 64 bits (8 bytes), la siguiente dirección de palabra es la 8.
 - **Un contenido:** Instrucción o dato que se almacena en la celda en cada momento
- Esta ligada a las unidades más rápidas del ordenador (UC y ALU)



4. Memoria Principal

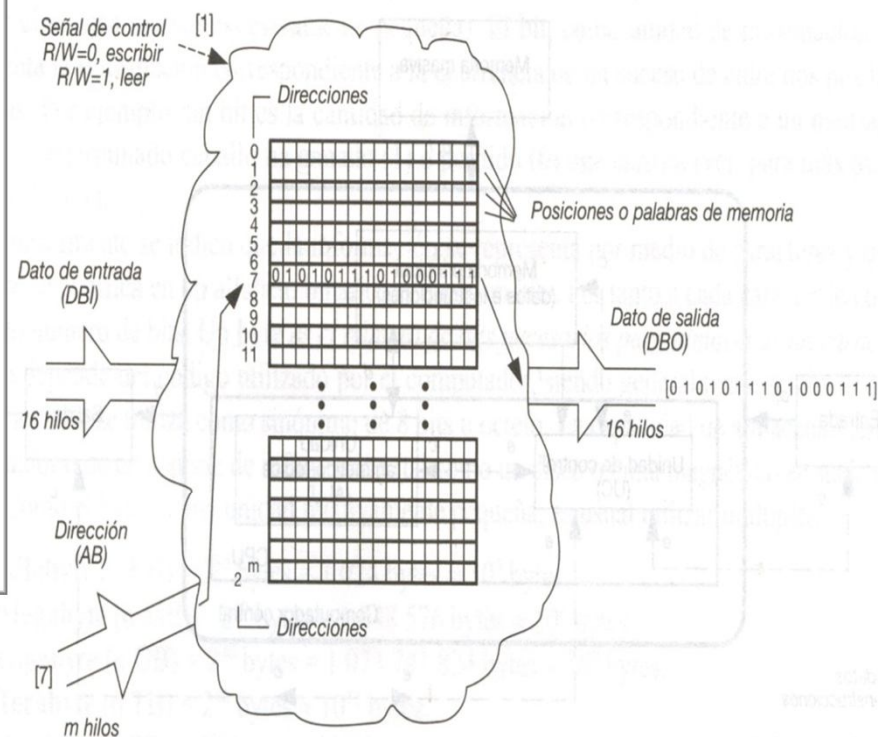
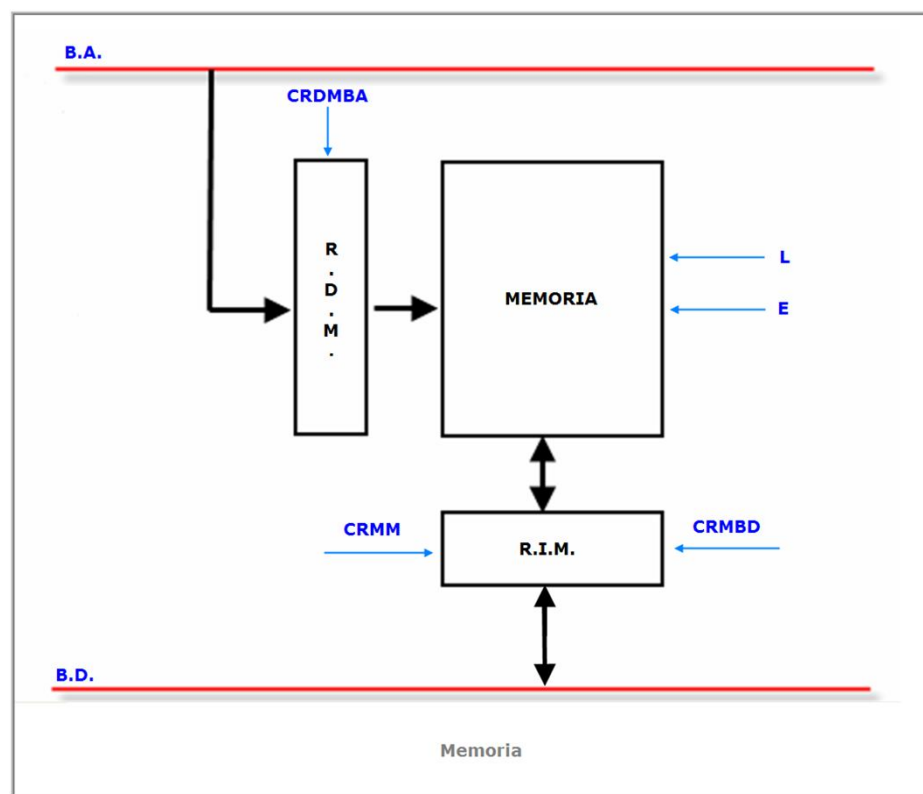


Diagrama simplificado de una memoria principal.

Actividad.

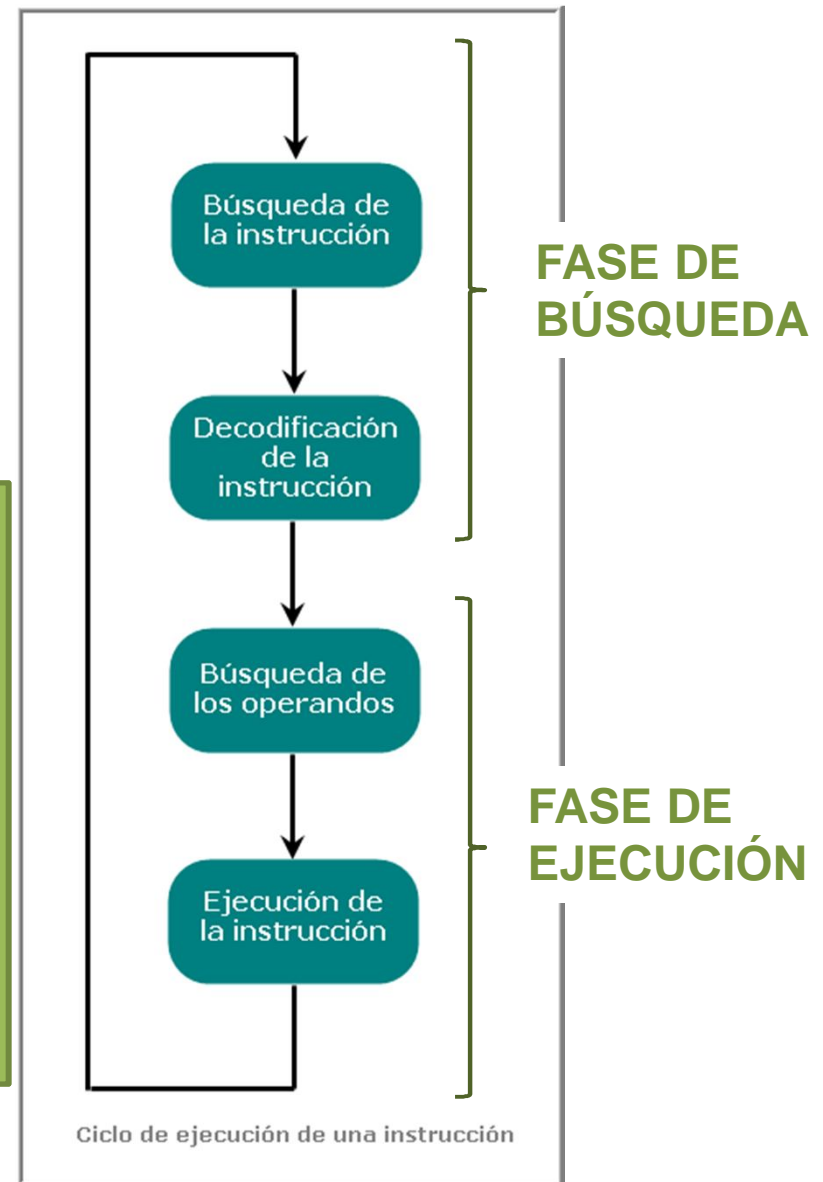
- Escribe detalladamente los pasos para escribir en una posición de memoria



5. Ejecución de una instrucción

CICLO DE INSTRUCCIÓN

La ejecución de una instrucción siempre **conlleva realizar la misma secuencia de pasos** independientemente del repertorio de instrucciones específicos que posea el procesador y de los campos y modos de direccionamiento que se hayan definido. Cada uno de **estos pasos puede necesitar un número diferente de ciclos de reloj** dependiendo de su propia complejidad y de los recursos que la CPU tenga para su realización.



SUMAR 033 992 993 (sumar los contenidos de las posiciones de memoria 033 y 992, almacenando el resultado en la posición 993).

- Código de operación (CO): **SUMAR**.
- Dirección del primer sumando (OP1): **033**.
- Dirección del segundo sumando (OP2): **992**.
- Dirección del resultado (OP3): **993**.

EJEMPLO

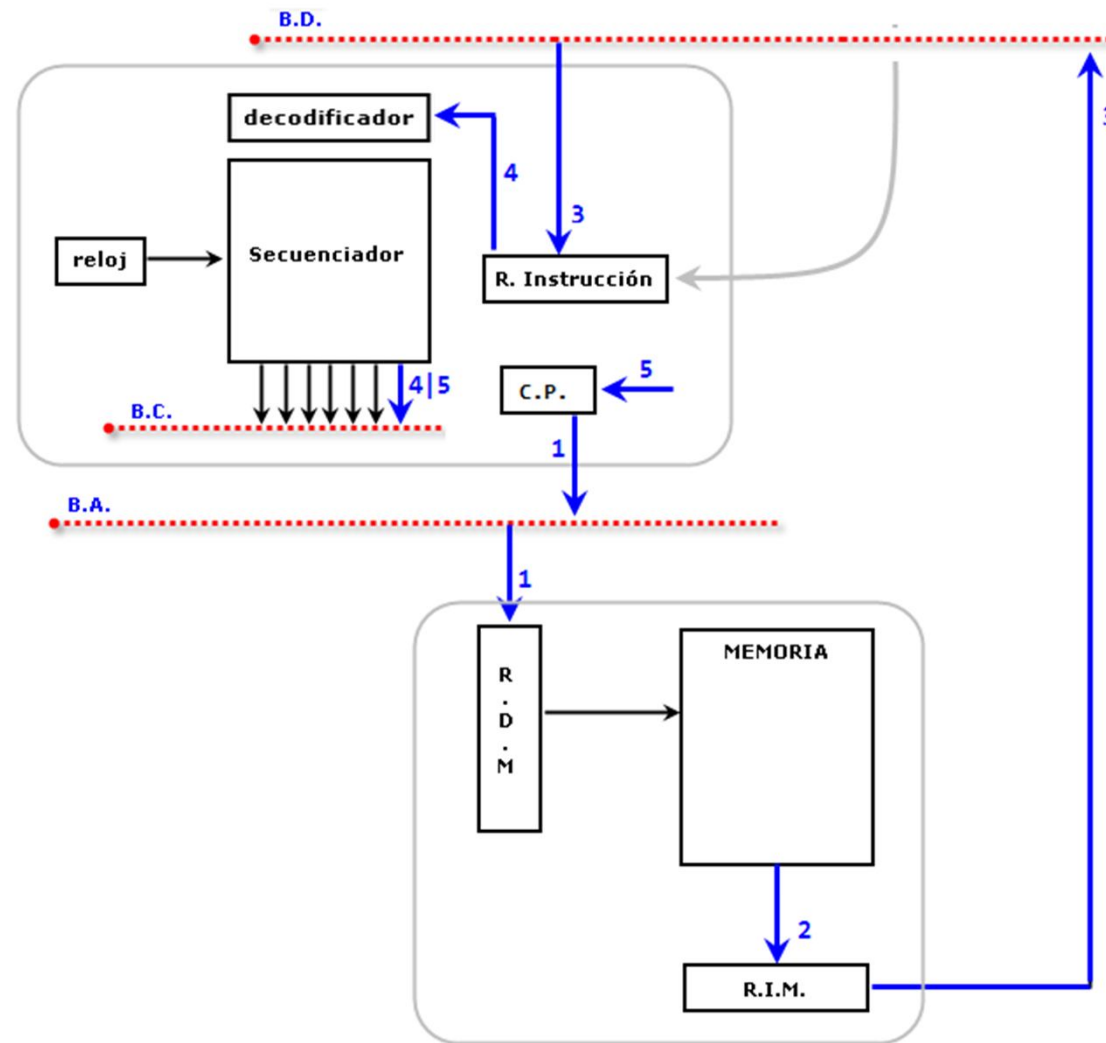
En la **fase de búsqueda** de la instrucción se realizan los siguientes pasos:

1. La unidad de control envía una microorden para que el contenido del registro contador de programa que contiene la dirección de la siguiente instrucción, sea transferido al registro de dirección de memoria.
2. La posición de memoria que figura en el registro de dirección de memoria es utilizada por la memoria para transferir su contenido al registro de intercambio de memoria.
3. Se transfiere la instrucción desde el registro de intercambio de memoria al registro de instrucción.
4. Posteriormente el decodificador procede a interpretar la instrucción que acaba de llegar al registro de instrucción, en este caso SUMAR, quedando dispuesto para la activación del circuito sumador de la ALU e informando al secuenciador.
5. El registro contador de programa se autoincrementa con un valor 1 (o n en el caso de que sea ésta la longitud de la palabra), de tal forma que quede apuntado a la siguiente instrucción situada consecutivamente en memoria. Si la instrucción en ejecución es de ruptura de secuencia, el contador de programa se cargará con la dirección que corresponda.

FASE DE BÚSQUEDA

EJEMPLO FASE DE BÚSQUEDA DE UNA INSTRUCCIÓN DE TRES OPERANDOS

cod. instrucción	dirección Oper1	dirección Oper2	dirección Resultado
------------------	-----------------	-----------------	---------------------



La **fase de ejecución** se realiza en los siguientes pasos, pero hay que tener en cuenta que si la instrucción no tuviese operandos, los pasos del 1 al 6 y el paso 8 se omitirían:

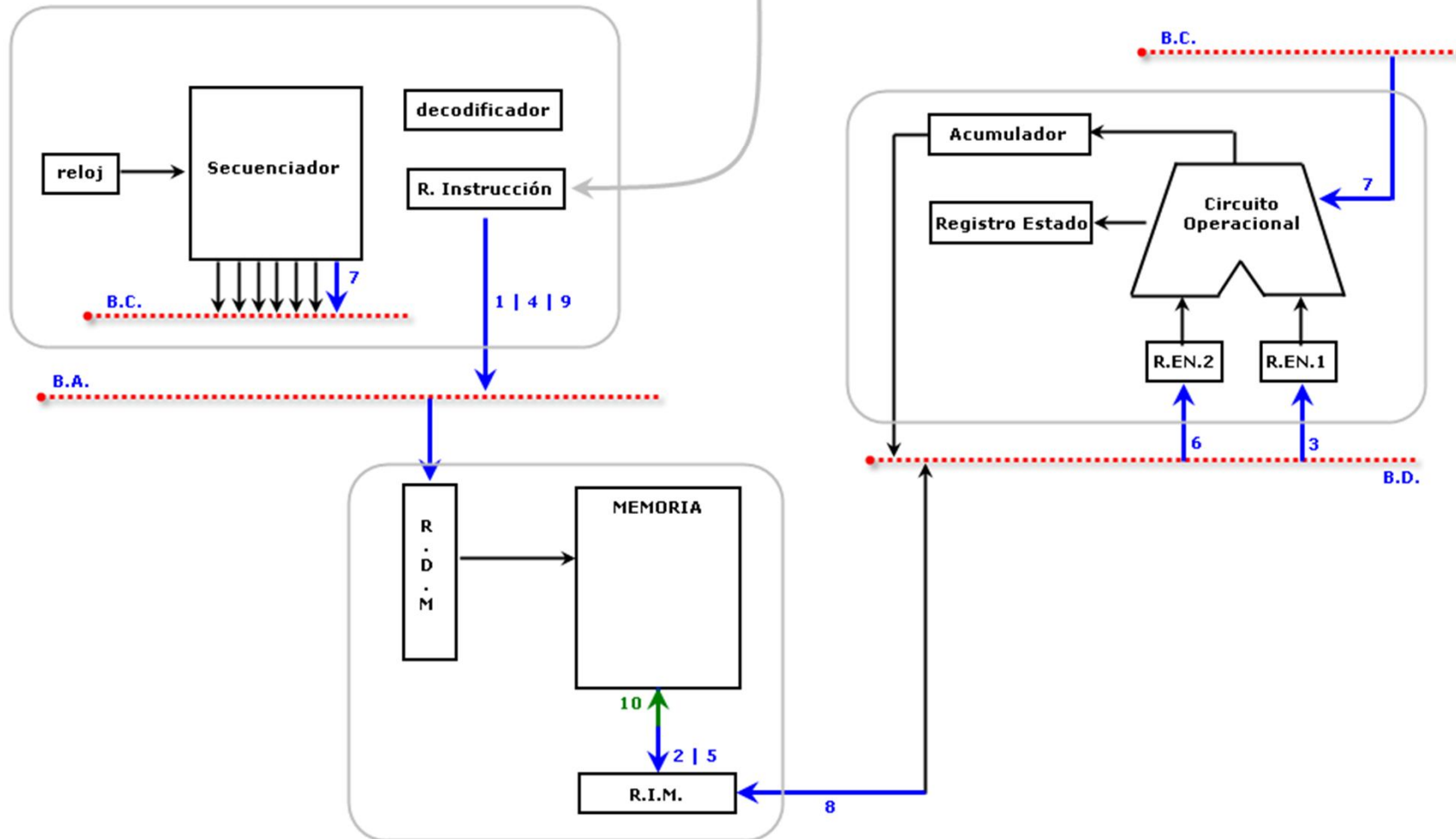
1. Se transfiere la dirección del primer operando desde el registro de instrucción al registro de dirección de memoria.
2. La memoria extrae dicho dato depositándolo en el registro de intercambio de memoria.
3. Se lleva este operando desde este RIM al registro de entrada 1 de la ALU.
4. Se transfiere la dirección del segundo operando desde el registro de instrucción al registro de memoria.
5. La memoria dicho dato y lo deposita en el registro de intercambio de memoria.
6. Se lleva este operando desde este registro al registro de entrada 2 de la ALU.
7. El secuenciador envía una micro-orden a la ALU para que se ejecute la operación. El resultado de la operación queda almacenado en el acumulador.
8. Este resultado se envía desde el acumulador al registro de intercambio de memoria.
9. Se transfiere desde el registro de instrucción al registro de dirección de memoria la dirección donde ha de almacenarse el resultado en la memoria.
10. Se transfiere el resultado desde el registro de intercambio de memoria a la dirección de memoria indicada en el registro de dirección de memoria.

CONTINUACIÓN

FASE DE EJECUCIÓN

EJEMPLO FASE DE EJECUCIÓN DE UNA INSTRUCCIÓN DE TRES OPERANDOS

cod. instrucción	dirección Oper1	dirección Oper2	dirección Resultado
------------------	-----------------	-----------------	---------------------



Actividad.

- Escribe detalladamente todos los pasos que se producen en la **fase de búsqueda y en la fase de ejecución** de la siguiente instrucción:
- **AND 042 827 828** (realizar la operación lógica and de las posiciones de memoria 042 y 827, almacenando el resultado en la posición 828).

Actividad.

- En el aula virtual Tema 2 → Práctica 1

Actividad.

- En el aula virtual Tema 2 → Ejercicios 2