

PROBLEMAS

1 En el diseño de la estructura de un computador se consideran los siguientes tiempos:

- lectura o escritura de cada registro: 1 unidad de tiempo (ut).
- operación elemental de mayor duración: 25 ut.
- acceso a Memoria de Control: 20 ut.
- secuenciador de microprograma: 6 ut.

Determine el mínimo tiempo del ciclo de reloj del computador, para cada una de las siguientes posibilidades de su Unidad de Control:

- a) cableada.
- b) microprogramada.

2 Sea un computador con Unidad de Control μ programada cuyo tiempo de ciclo es de 10 ut. La memoria es asíncrona y activa una señal *READY* cuando acaba el acceso que se le ha solicitado. Microprograme a nivel RT la instrucción de una palabra POP [--.Ri], indicando qué microinstrucciones pertenecen a cada una de las fases de ejecución de la instrucción incluyendo el fetch. Suponga que el puntero de pila crece en direcciones decrecientes de memoria y apunta al primer hueco en pila. Indique el tiempo medio de ejecución de dicha instrucción, suponiendo que el número medio de ciclos necesario para cada acceso a memoria es de 3 ciclos.

3 Indique, justificando su respuesta, si las siguientes afirmaciones son verdaderas o falsas:

- a) La Unidad de Control necesita como entrada el registro de estado para poder ordenar la ejecución de las instrucciones de comparación.
- b) El registro de instrucción es un registro de propósito específico que contiene la dirección de la siguiente instrucción a ejecutar.
- c) Las únicas instrucciones en las que alguna de sus fases de ejecución conllevan un acceso a memoria son las instrucciones load y store.
- d) El registro puntero de pila es un registro de propósito general que puede contener tanto direcciones como datos.

4 Sea un computador cuyo tiempo de acceso a memoria es de 24 ut, y cuya operación elemental más larga dura 16 ut.

a) Desglose en operaciones elementales a nivel RT la instrucción de dos palabras: ADD #5[.3++], /dir, indicando qué operaciones se solaparán en el mismo ciclo de reloj, e incluyendo el fetch de la siguiente instrucción.

b) Indique el tiempo total de ejecución de dicha instrucción suponiendo que:

1.- La unidad de control sea cableada

2.- La unidad de control sea microprogramada, y los retardos de algunos dispositivos son:

- Multiplexores: 1 ut
- Secuenciador de UC: 14 ut
- Acceso a Memoria de Control: 18 ut
- Decodificador: 5 ut

5 Sea un computador cuyo tiempo de ciclo es de 10 ut y el tiempo de acceso a memoria es de 22 ut. Microprograme a nivel RT la instrucción de dos palabras que se indica a continuación, indicando qué microinstrucciones pertenecen a cada una de las fases de ejecución de la instrucción: PUSH /1000, e incluyendo el fetch de la siguiente instrucción. Suponga que el puntero de pila crece en direcciones decrecientes de memoria y apunta a la última palabra introducida en pila. Indique el tiempo total de ejecución de dicha instrucción.

6 Responda razonadamente si son ciertas o no las siguientes afirmaciones sobre la unidad de control

- 1.- El tiempo de ciclo debe estar determinado para que dé tiempo a ejecutar la operación elemental más larga posible, incluidos los accesos a memoria y los ciclos de bus.
- 2.- La codificación de todos los campos de la microinstrucción de la unidad de control es esencial para el mejor rendimiento del procesador.
- 3.- Hoy en día el tipo de unidad de control más utilizada es la microprogramada por ser la más flexible y tener una estructura común en la misma familia de procesadores.

7 Sea un computador con una unidad de control microprogramada, con 4 registros temporales (TMP1 a TMP4) y con un microcontador para poder realizar microbucles. La memoria principal es asíncrona, activa una señal READY cuando acaba el acceso solicitado, y es direccionable a nivel de byte. Se quiere dotar a este computador de instrucciones múltiples que operan con operandos vectoriales. Entre ellas, la instrucción de una palabra PUSHM #n, [.Ri]. Esta instrucción introduce en pila #n palabras de memoria apuntadas por el registro .Ri, cuyo valor no queda modificado tras la instrucción. El puntero de pila crece hacia direcciones decrecientes de memoria y apunta al último dato introducido en pila.

Microprograme a nivel RT la instrucción PUSHM #n, [.Ri], especificando cómo se resolvería:

- el incremento de direcciones de memoria si el direccionamiento es a nivel de byte.
- la carga del dato inmediato #n en el microcontador.

8 Sea la estructura del procesador de la figura 1.

A través de los buses de datos y de direcciones se conecta con una memoria asíncrona que activa una señal *READY* cuando ha acabado la operación solicitada y cuyo tiempo medio de acceso es de 110 ut.

a) Añada en la figura las señales de control que estime necesarias.

b) Calcule el tiempo de ciclo de reloj para este procesador en el caso de que la unidad de control sea microprogramada. Los retardos de algunos dispositivos del computador son:

- Acceso al banco de registros (BR): 8 ut
- Operación ALU: 54 ut
- Acceso a registro: 4 ut
- Acceso a Memoria de Control: 38 ut
- Multiplexores: 1 ut
- Decodificador: 5 ut
- Triestado: 1 ut

c) ¿Qué modos de direccionamiento podría admitir esta estructura del procesador? ¿Sería posible el direccionamiento relativo a registro base? Proponga algún cambio para mejorar el rendimiento con este modo de direccionamiento y explique cómo se lograría dicha mejora.

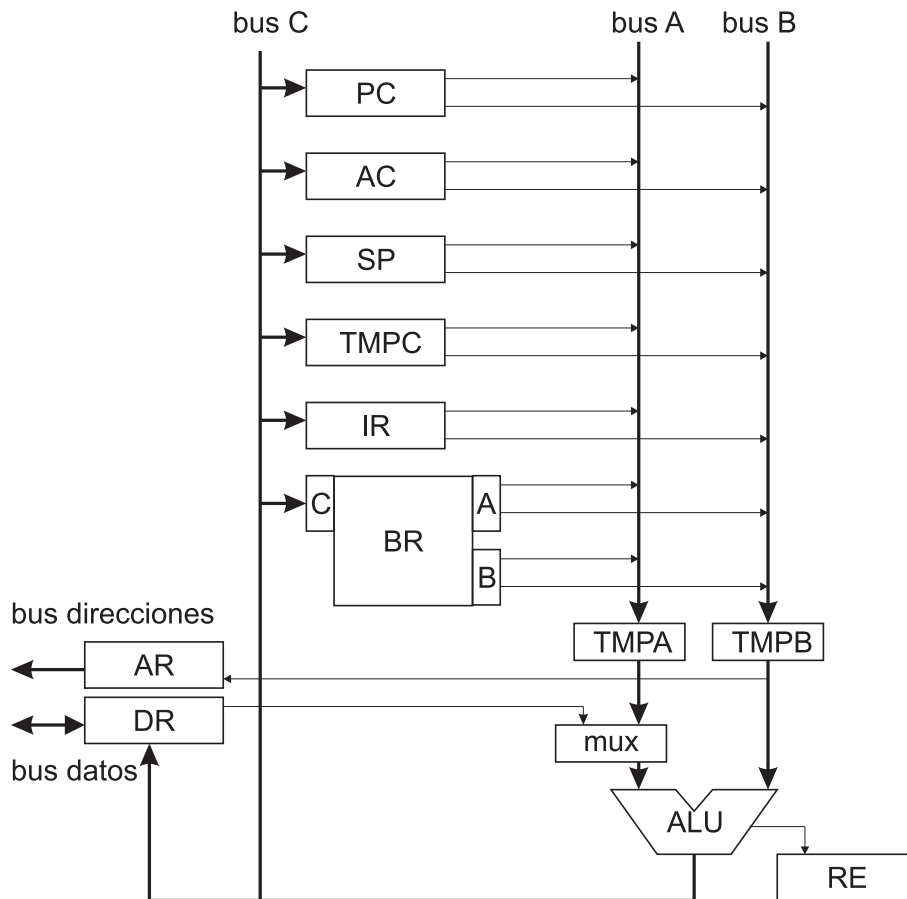


Figura 1. Estructura de la CPU

9 La figura 2 muestra la estructura de la CPU de un computador con palabras y direcciones de 32 bits, cuya unidad de control es microprogramada. El banco de registros (BR) contiene 25 registros generales (R1 a R25) y una puerta (A) de lectura/escritura. El registro de estado es de 6 bits. Ninguno de los registros de este computador permite autoincremento o autodecremento. La memoria es asíncrona, lo que se indica mediante la señal WAIT. Las operaciones que realiza la ALU aparecen en el recuadro inferior de la figura.

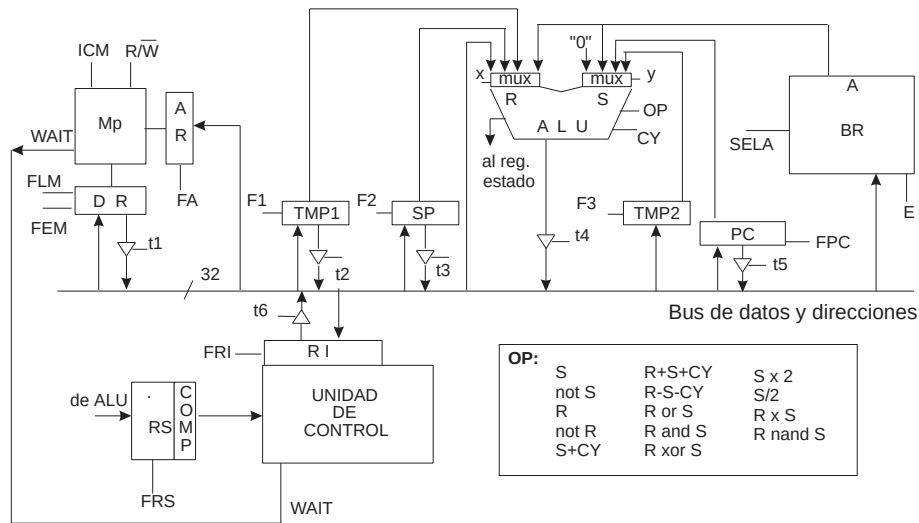


Figura 2. Estructura de la CPU y operaciones de la ALU

1. Realice, a nivel RT, el microprograma de la instrucción de dos palabras `ADD_INT .R1, .R2, /Dir` cuyo formato se muestra en la figura 3:

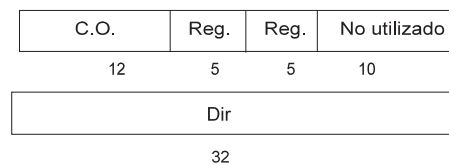


Figura 3. Formato de la instrucción

Esta instrucción suma el contenido de dos registros del banco de registros e intercambia el resultado con el contenido de una dirección de memoria. El intercambio sólo se produce si el resultado de la suma es distinto de cero, en cuyo caso, se almacena en la dirección de memoria especificada en la segunda palabra de la instrucción y el contenido de ésta se carga en ambos registros.

2. De acuerdo con las características del computador representado en la figura 1 y, teniendo en cuenta, la siguiente información:
- El juego de instrucciones consta de 200 instrucciones.
 - El tamaño medio de cada microprograma es de 30 palabras de control.
 - Se reservan 144 posiciones para la microprogramación de microsubrutinas.

Diseñe el formato de las microinstrucciones y calcule el tamaño mínimo de la memoria de control de este computador.

- *TMP1 y TMP2: Registros temporales, transparentes al usuario*
- *BR: banco de registros con 12 registros generales (R1 a R12) y una puerta de lectura/escritura (A)*
- *AR: Registro de direcciones de memoria*
- *DR: Registro de datos de memoria*
- *RI: Registro de instrucción*
- *PC: Contador de programa*
- *SP: Puntero de pila, que apunta a la última dirección ocupada de la pila*
- *RS: Registro de estado, de 8 bits*

Diagrama de un procesador de datos de 32 bits. El diagrama muestra una arquitectura de procesamiento de datos con un bus de 32 bits. Los componentes principales incluyen:

- Multiplicador (Mp) y Divisor (DR):** Reciben FLM y FEM. El Mp produce WAIT y el DR produce FEM.
- Registro de Aritmético (AR) y Registro de Estado (FA):** El AR recibe WAIT y produce FA. El FA recibe FA y produce WAIT.
- Registro de Instrucciones (RI):** Recibe FRI y produce WAIT.
- Registro de Estado (RS) y Registro de Control (COMP):** Reciben de ALU y FRS. El RS produce FRS y el COMP produce WAIT.
- Registro de Instrucciones (RI):** Recibe FRI y produce WAIT.
- Registro de Estado (RS) y Registro de Control (COMP):** Reciben de ALU y FRS. El RS produce FRS y el COMP produce WAIT.
- Registro de Instrucciones (RI):** Recibe FRI y produce WAIT.
- Registro de Estado (RS) y Registro de Control (COMP):** Reciben de ALU y FRS. El RS produce FRS y el COMP produce WAIT.

El bus de datos y direcciones (32 bits) conecta todos los componentes. El ALU (Arithmetic Logic Unit) recibe datos de los registros y el bus, y produce resultados que se almacenan en los registros. El PC (Program Counter) recibe el resultado del ALU y el bus, y produce el siguiente PC. El BR (Branch Register) recibe el resultado del ALU y el bus, y produce el siguiente PC. El diagrama también muestra un controlador de flujo de datos (FPC) que recibe el resultado del ALU y el bus, y produce el siguiente PC. El diagrama incluye una tabla de operaciones (OP) que define las operaciones de aritmética y lógica que se pueden realizar con los registros.

OP:		
S	R+S+CY	S x 2
not S	R-S-CY	S/2
R	R or S	R x S
not R	R and S	R nand S
S+CY	R xor S	

Figura 4. Estructura de la CPU y operaciones de la ALU

- *Tiempo de lectura/escritura del banco de registros: 3 ns.*
- *Tiempo de la operación más lenta de la ALU: 10 ns.*
- *Tiempo de respuesta de una puerta triestado: 1 ns.*
- *Tiempo de lectura/escritura de los registros transparentes o específicos: 2 ns.*
- *Tiempo de acceso a memoria de control: 20 ns.*
- *Tiempo del secuenciador del microprograma: 3 ns.*

Determine el mínimo periodo de reloj que debe utilizarse en este computador, considerando despreciables los tiempos de lectura/escritura de los registros internos de la unidad de control.

2. Microprograme a nivel RT (transferencia entre registros) la siguiente instrucción de una palabra, incluyendo la fase de fetch:

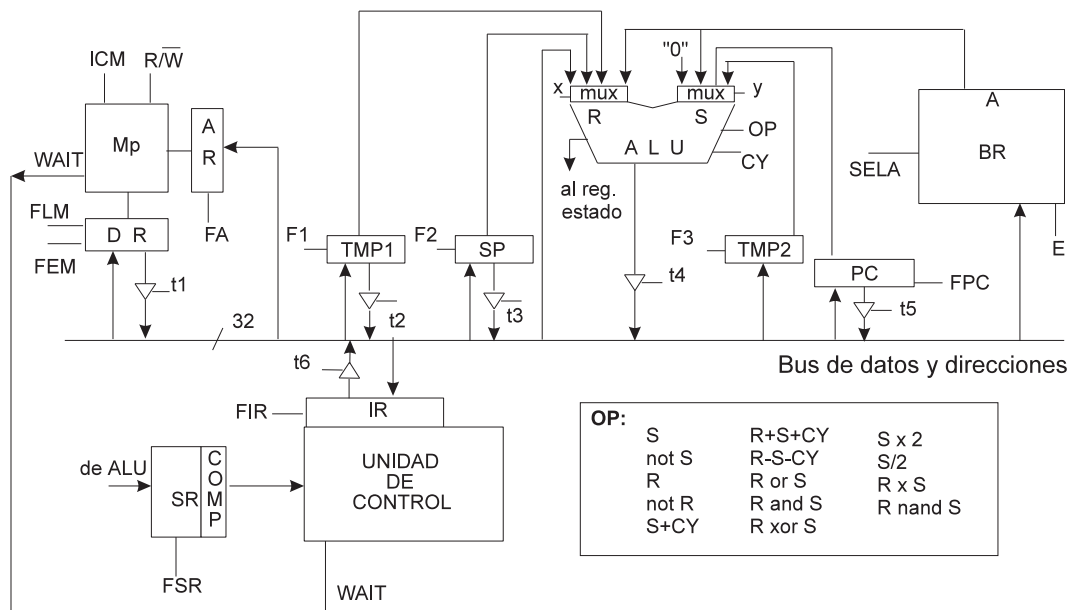
CALL_ADDC .R4, [.R5 ++], \$ DESP

Esta instrucción, cuyo código de operación es de 12 bits, realiza la suma de los dos primeros operandos, almacenando el resultado en el primero y produce una llamada a subrutina si el resultado de la suma produce acarreo. La dirección de salto a subrutina, se especifica en el tercer operando con direccionamiento relativo a contador de programa, siendo DESP el correspondiente desplazamiento. Considere que existen en la máquina los mecanismos automáticos adecuados para hacer la extensión de signo del desplazamiento, cuando se carga en un registro.

3. Determine el rango de direcciones en el que puede situarse la subrutina, teniendo en cuenta que la instrucción del apartado 2) está almacenada en la dirección de memoria $H'0FFFFFFF$ y que el desplazamiento está representado en complemento a 2. Justifique su respuesta.

- TMP1 y TMP2 son registros temporales transparentes al usuario.
- BR es un banco de registros compuesto por 18 registros generales (R1 a R18).
- AR y DR son, respectivamente, el registro de direcciones de memoria y el registro de datos de memoria.
- IR es el registro de instrucción, PC el contador de programa y SR el registro de estado.

Las operaciones que puede realizar la ALU están indicadas en el recuadro inferior de la figura 5.



a) Teniendo en cuenta las especificaciones anteriores, realice los microprogramas a nivel RT (transferencia entre registros) de las instrucciones de una palabra `PUSH [Ri++]` y `POP [--Ri]` y de la microsubrutina de fetch.

- Lectura o escritura del banco de registros: 3 ns.
- Lectura o escritura de un registro: 2 ns.
- ALU y multiplexores: 14 ns.
- Buffer triestado: 1 ns.
- Secuenciador: 5 ns.
- Memoria de control: 10 ns.

12 La figura 6 muestra esquemáticamente la estructura de la CPU de un computador con direccionamiento a nivel de byte, cuyas palabras y direcciones son de 32 bits. La CPU tiene un bus interno de direcciones y otro bus interno de datos; un registro acumulador AC; registros temporales y transparentes al usuario T1, T2, T3 y T4; banco de registros BR con otros tres registros transparentes más (T10 a T12), además de 14 registros generales (R1 a R14), y un registro especial R0, que siempre contiene un 0. AR es el registro de direcciones de memoria, DR es el de datos de memoria, IR es el registro de instrucción, y SR es el registro de estado. La CPU dispone además del PC (contador de programa) y del SP (puntero de pila), en principio sin posibilidad de autoincremento ni autodecremento. La unidad de control es microprogramada, y dispone de un secuenciador con microcontador para microbucles, así como la posibilidad de realizar microsubrutinas.

Las operaciones que es capaz de realizar la ALU de este computador aparecen en el recuadro de la figura 6, y su memoria funciona de forma asíncrona, dependiendo de la activación de una señal Ready.

El juego de instrucciones de este computador tiene, entre otras, la instrucción de una palabra *ORM #n, [.Ri], [.Rj]*, que realiza la operación OR del contenido de los dos vectores de *n* elementos apuntados por *Ri* y *Rj*, dejando el resultado a partir de *Ri*. Al finalizar la ejecución de esta instrucción, los registros *Ri* y *Rj* contendrán la dirección de comienzo de los vectores.

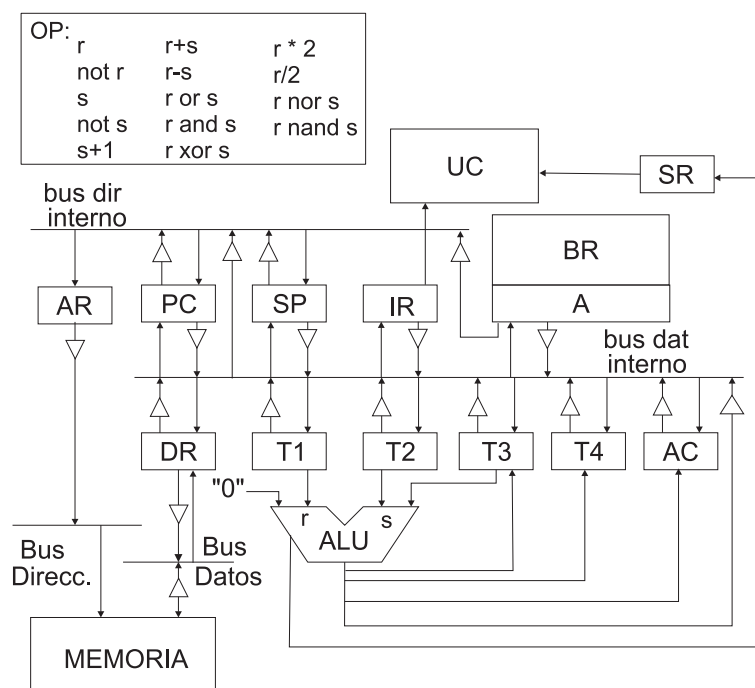


Figura 6. Estructura de la CPU y operaciones de la ALU

a) Teniendo en cuenta que el direccionamiento es a nivel de byte, indique posible/s mejora/s en el diseño de la estructura que mejore el rendimiento. Explique y argumente su respuesta.

b) Modifique la estructura de la CPU de la figura para que el rendimiento sea mejor en la instrucción descrita en el enunciado, describiendo las mejoras de sus propuestas.

c) Microprograme a nivel RT (transferencia entre registros) la instrucción del enunciado (incluyendo la microrrutina de fetch) haciendo uso de las modificaciones introducidas en el apartado a).

d) Suponga que en el juego de instrucciones existe también la siguiente instrucción de una palabra *ORCMPZM #n, [.Ri], [.Rj], [.Rk]*

Esta instrucción es similar a la anterior *ORM*, ya que también realiza la operación OR del contenido de los dos vectores de *n* elementos apuntados por *Ri* y *Rj*, dejando el resultado a partir de *Ri*. Pero al mismo tiempo, va elaborando un nuevo vector en memoria a partir de la dirección apuntada por *Rk*. En este otro vector incluye un 0 en la posición de memoria correspondiente cuando la operación OR da como resultado un 0, y en caso contrario, almacena un 1.

Indique **a grandes rasgos** cómo se podría perfilar la microprogramación de la instrucción *ORCMPZM*, partiendo de un código similar al del apartado c), modificándolo. Por ejemplo, añadiendo llamadas a microsubrutinas. Indique dónde, y en qué consistirían éstas, y si habría que cuidar información guardada en registros temporales de alguna forma, cómo se podría salvaguardar, etc.

13 Sea la CPU cuyo esquema simplificado aparece en la figura 7, en la que se muestran las rutas de datos, puertas triestado, y las señales de control/estado. El ancho de palabra es de 32 bits, así como todos los registros, rutas de datos y direcciones. El direccionamiento es a nivel de byte.

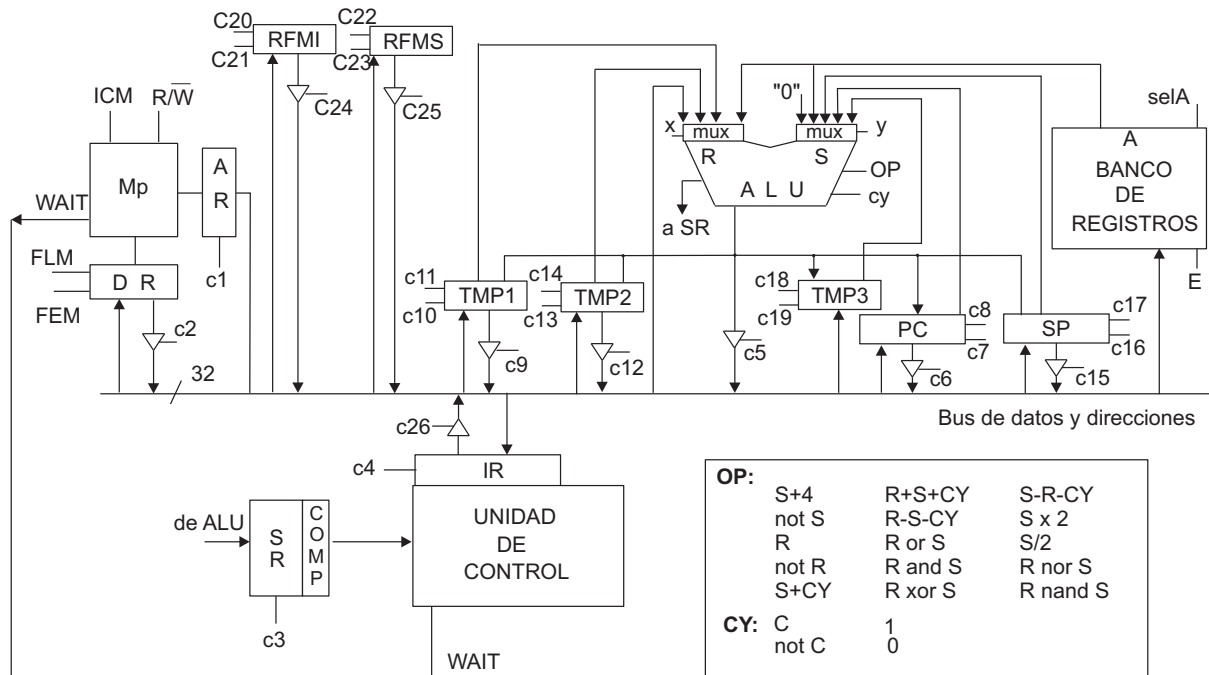


Figura 7

Los registros TMP1 a TMP3 son transparentes al usuario, y el banco de registros contiene dos registros transparentes más (TMP4 y TMP5), y 14 registros de propósito general (R0, que siempre contiene 0, y R1 a R13). La ALU puede hacer las operaciones indicadas en la tabla OP de la figura.

Esta CPU tiene dos modos de operación: modo supervisor y modo usuario. El registro SR es el registro de estado, y contiene, entre otros, el campo S (tendrá un 1 si opera en modo supervisor, y un 0 si opera en modo usuario).

Algunas operaciones de este computador están catalogadas como privilegiadas. En concreto, en modo usuario, la zona de memoria a la que se puede acceder tanto en lectura como en escritura está limitada por dos registros frontera, que apuntan a las direcciones mínima y máxima de memoria que el usuario puede direccionar (Registro Frontera de Memoria Inferior, RFMI, y Registro Frontera de Memoria Superior, RFMS). Cualquier intento de lectura o escritura en memoria fuera de este rango por parte de un usuario causará respectivamente una excepción de privilegio en lectura o una excepción de privilegio en escritura en memoria. En modo supervisor, se puede acceder a cualquier posición de memoria.

Considerando los siguientes tiempos:

- Tiempo de lectura/escritura de registros: 1 ut.
- Tiempo de retardo de triestados: 0,5 ut.
- Tiempo de retardo de decodificador: 2 ut.
- Tiempo de lectura/escritura de banco de registros: 2 ut.
- Tiempo de retardo de la ALU: 25 ut.
- Tiempo de acceso a Memoria de Control: 15 ut.
- Tiempo de retardo del secuenciador de microprograma: 8 ut.
- Tiempo de retardo de multiplexor: 2 ut.
- Tiempo medio de acceso a memoria principal: 55 ut.

a) Determine el mínimo tiempo de ciclo de reloj que debe utilizarse en este computador para los siguientes casos:

1. Que la unidad de control sea cableada.
2. Que la unidad de control sea microprogramada.

b) Microprograme una microsubrutina **comprobar_dir** que compruebe si la dirección en la que se va a operar en memoria es privilegiada o no. Indique y decida cuáles son y dónde se dejan tanto el parámetro de entrada como el de salida en esta subrutina.

c) Se quiere incorporar al repertorio de instrucciones de esta CPU la instrucción **privilegiada de dos palabras** LD .SR, /Dir. Se pide:

1. Modifique la estructura de esta CPU para que se pueda ejecutar dicha instrucción.
2. Escriba a nivel de transferencia entre registros, el microprograma de la instrucción LD .SR, /Dir teniendo en cuenta que el computador tiene unidad de control microprogramada. Utilice la microsubrutina **comprobar_dir** si cree que le hiciera falta. Tenga en cuenta que esta instrucción puede causar violación de privilegio por código de operación por tratarse de una instrucción privilegiada (ya que pudiera escribir en el registro de estado SR). Incluya la microrutina de fetch.
3. Teniendo en cuenta sus respuestas al resto de este apartado, calcule el tiempo de ejecución de la instrucción anterior suponiendo que se ejecuta con SR.S=1.

14 Sea la CPU cuyo esquema simplificado aparece en la figura 8, en la que se muestran las rutas de datos, puertas triestado, y las señales de control/estado. El ancho de palabra es de 32 bits, así como todos los registros, rutas de datos y direcciones. El direccionamiento es a nivel de byte.

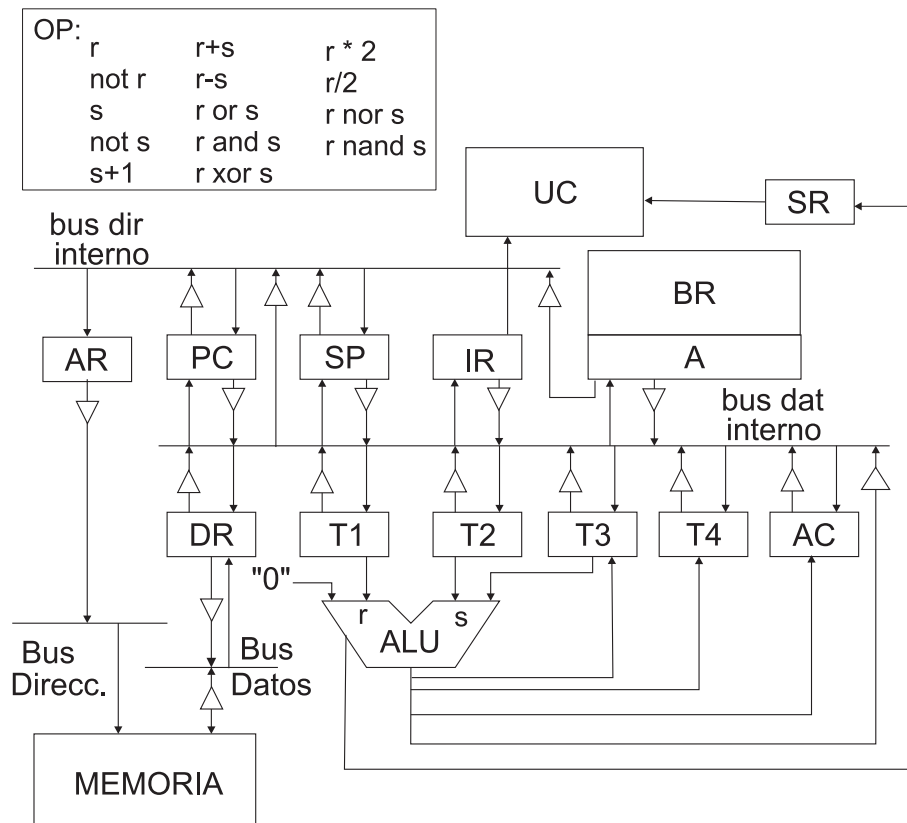


Figura 8

Los registros T1 a T4 son transparentes al usuario, el AC es un registro Acumulador, PC es el Contador de Programa, SP es el Puntero de Pila y el banco de registros contiene dos registros transparentes más (T5 y T6), y 26 registros de propósito general (R0, que siempre contiene 0, y R1 a R25). La ALU puede hacer las operaciones indicadas en la tabla OP de la figura.

La Unidad de Control es microprogramada, y el tiempo de ciclo de reloj es de 100 ut. El tiempo medio de acceso a memoria principal: 150 ut.

a) En el repertorio de instrucciones se quiere incluir la llamada a subrutina condicional:

CALLEQ .Ac, .Ri, [.Rj]

Esta instrucción salta a la subrutina almacenada a partir de [.Rj] en el caso de que el contenido del Acumulador y el registro .Ri sean iguales. Si no, continúa secuencialmente.

1. Microprogramme a nivel RT (Transferencia entre Registros) la instrucción CALLEQ. Incluya el microcódigo del ciclo de fetch.
2. ¿Cuánto tarda en ejecutarse esta instrucción (incluido el fetch), suponiendo que la probabilidad de salto es del 80 %? (tenga en cuenta su respuesta del apartado anterior)

b) Se quiere modificar el diseño previo de esta CPU para que tenga dos modos de operación: modo supervisor y modo usuario, de forma que, se pueda controlar que haya instrucciones y ciertas operaciones no permitidas en modo usuario, por ser privilegiadas. En concreto, acceder a memoria fuera del rango permitido sería una de estas operaciones privilegiadas que sólo podría hacerse en modo supervisor. Cualquier intento de lectura o escritura en memoria fuera de este rango por parte de un usuario causará respectivamente una excepción de privilegio en lectura o en escritura en memoria.

1. Modifique la estructura de esta CPU para que se puedan controlar las excepciones de acceso a zona privilegiada de memoria. Indique cuál es el mecanismo para recordar cuál es el modo de ejecución actual, y cómo se podría modificar este modo. ¿Sería esta modificación una operación privilegiada, o una operación válida al usuario?
2. Indique qué modificaciones habría que hacer en el microprograma que propuso de la instrucción `CALLEQ .Ac, .Ri, [.Rj]` para que compruebe posibles excepciones de acceso a zona privilegiada de memoria. ¿Cómo afectarían estas modificaciones al tiempo de ejecución de esta instrucción?

15 Sea la estructura del procesador de la figura 9, que dispone de unidad de control cableada. La memoria de este computador tiene un tiempo de acceso de dos ciclos de reloj. Se quiere dotar a este procesador de la instrucción de dos palabras `OR .R1, .R2, /dir`. El destino se deja en el registro R1.

Desglose en operaciones elementales a nivel RT la instrucción `OR .R1, .R2, /dir`, indicando qué operaciones se solaparán en el mismo ciclo de reloj, e incluyendo el fetch de la siguiente instrucción.

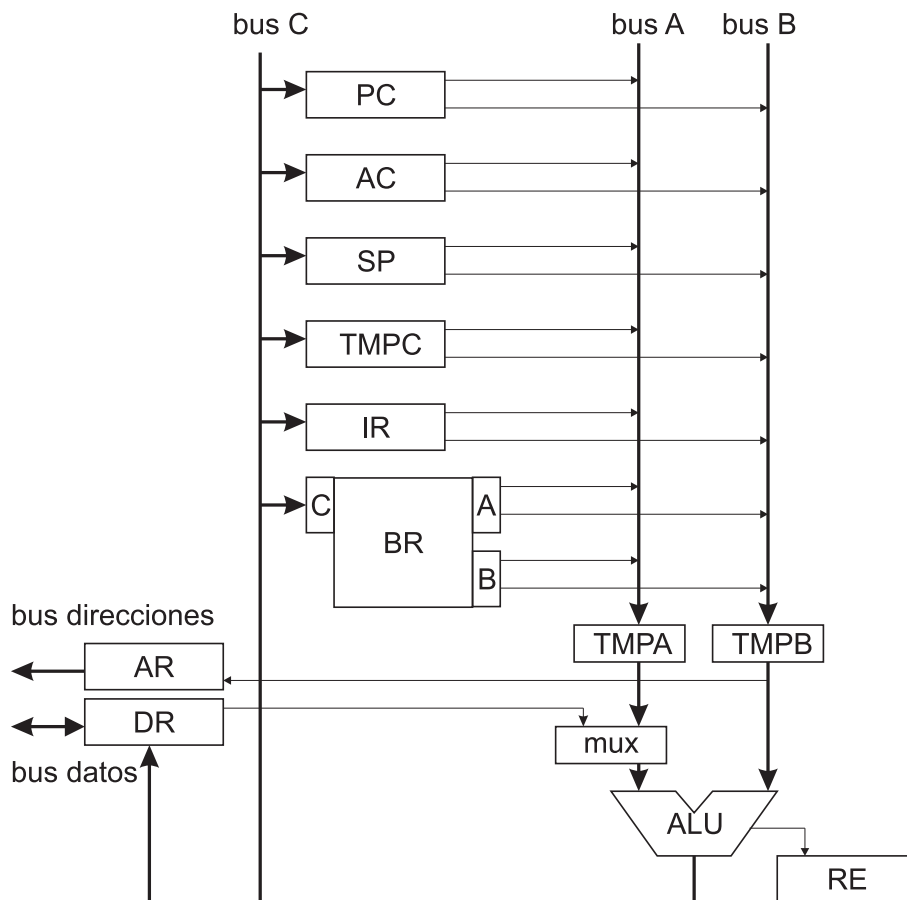


Figura 9. Estructura de la CPU

16 En la figura 10 se muestra el esquema de un computador de 64 bits con Unidad de control microprogramada y direccionamiento a nivel de byte. Los accesos a memoria tienen una duración media de 2 ciclos de reloj. La pila crece hacia direcciones decrecientes y el puntero de pila (SP) apunta a la primera posición vacía de la cima de la pila. Los incrementos o decrementos de los registros, se realizan a través de la ALU.

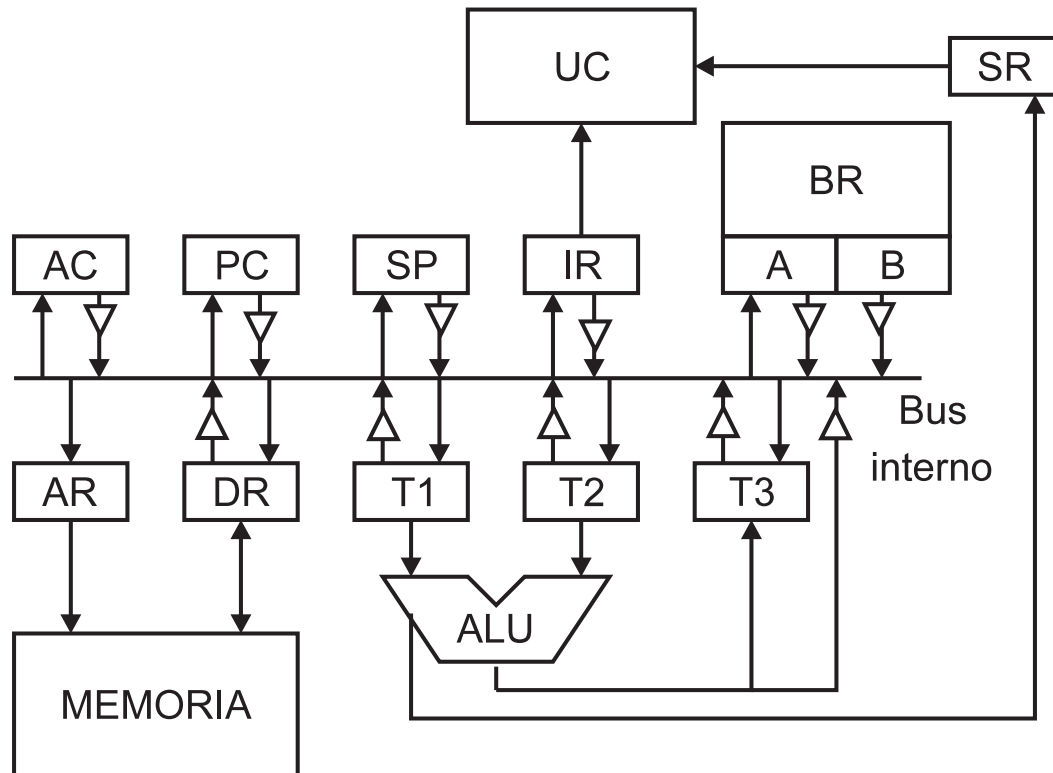


Figura 10. Estructura de la CPU

a) Realice, a nivel RT (transferencia entre registros), el microprograma de la microsubrutina de fetch

b) Realice a nivel RT el microprograma de la fase de ejecución de la instrucción de una palabra, perteneciente al juego de instrucciones de este computador: `CALLNZ #desp[++.R2]`. Esta instrucción realiza un salto a subrutina condicional si NZ.

c) De acuerdo con los microprogramas de los apartados anteriores y considerando una frecuencia de reloj de 500 MHz, determine el tiempo medio que tarda en ejecutarse la instrucción del apartado b.

17 En la siguiente figura 11 se muestra el esquema de un computador con palabras y direcciones de 64 bits y direccionamiento a nivel de palabra, que incluye dos buses internos al procesador (datos y direcciones), cuatro registros transparentes (T1, T2, T3 y T4) y un registro acumulador (AC). El Banco de registros tiene un puerto de entrada y dos de salida. Los accesos a memoria tienen una duración de 2 ciclos de reloj. El incremento o decremento de cualquier registro, se realiza a través de la ALU, cuyas operaciones se muestran en el recuadro superior.

Nota: Por simplificación en la figura, no se han detallado en ella las señales de control.

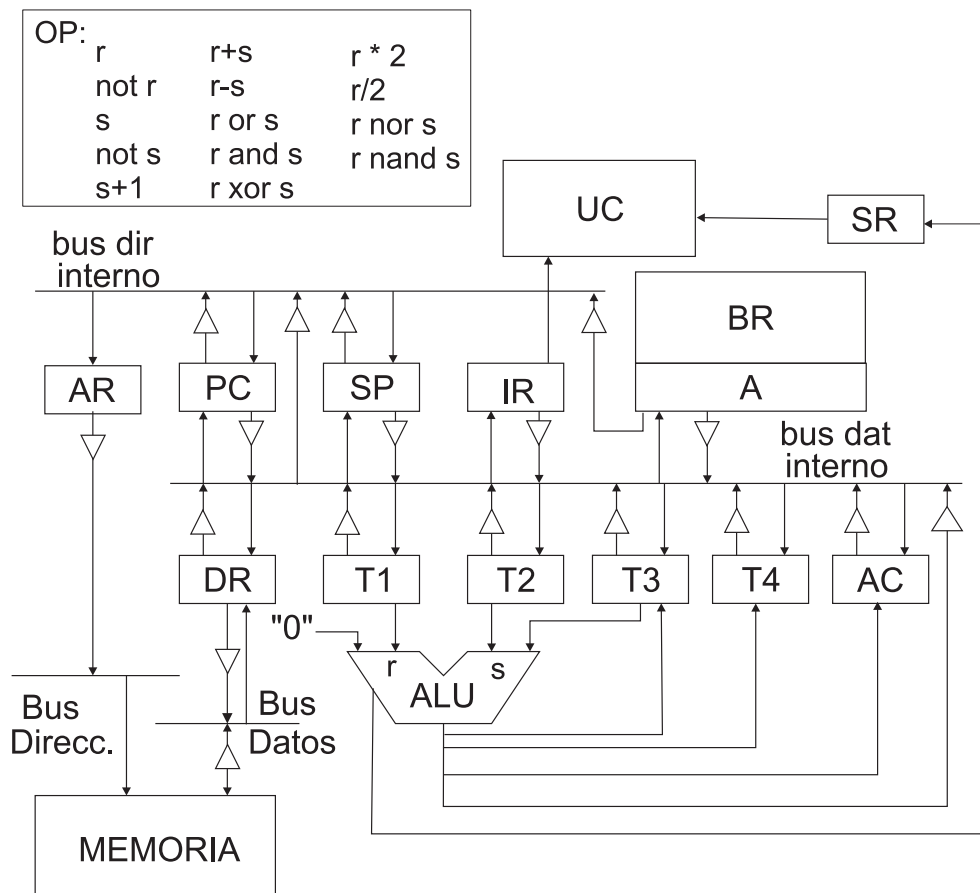


Figura 11.

a) Considerando que la Unidad de Control es **cableada**, detalle a nivel RT (transferencia entre registros) las operaciones elementales que se producirán en cada ciclo de reloj, durante la fase de fetch de una instrucción.

b) Si el computador trabaja a una frecuencia de 400 MHz, determine el tiempo que tarda en ejecutarse la fase de fetch especificada en el apartado anterior.

c) Teniendo en cuenta ahora, que la Unidad de Control es **microprogramada**, realice (a nivel RT) el microprograma de la fase de ejecución de la instrucción de una palabra: `XMEM_NC .R2, #desp[.R3]` Esta instrucción realiza el intercambio entre un registro y una posición de memoria, si no está activado el acarreo.

18 Sea la CPU cuyo esquema simplificado aparece en la figura 12. La ALU, todos los registros, rutas de datos y de direcciones son de 32 bits.

PC: Reg. contador de programa SP: Reg. puntero de pila
 AR: Reg. de direcciones DR: Reg. de datos
 IR: Reg. de instrucciones SR: Reg. de estado
 Y, Z: registros transparentes AC: es el único registro de propósito general

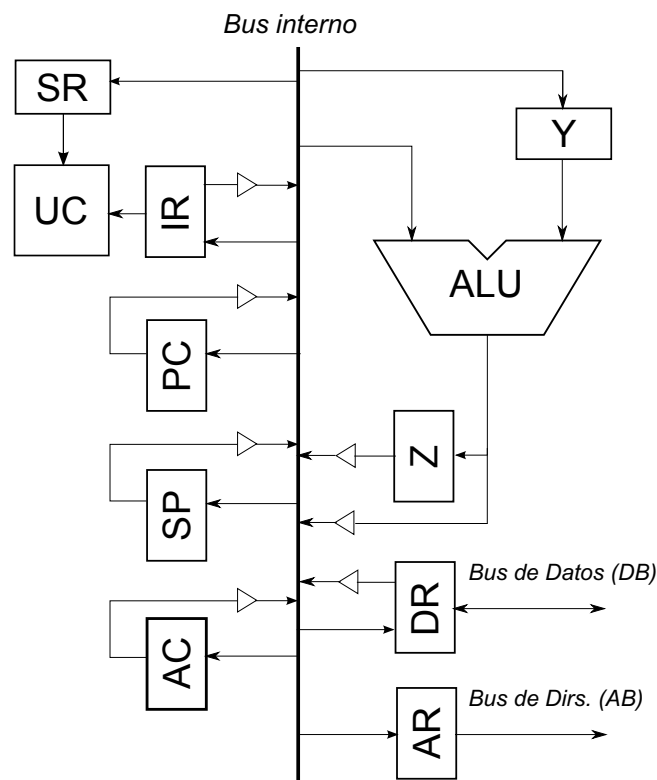


Figura 12. Estructura de la CPU

a) Suponiendo que:

- las instrucciones ocupan todas una sola palabra (incluso las que usan un direccionamiento directo a memoria, /Dir, e inmediato, #valor).
- todas la microoperaciones duran un ciclo de reloj, incluidas las de acceso a memoria.
- el tiempo de ciclo se ha establecido es 100 ns.
- la ALU realiza las operaciones habituales (suma, resta, incremento, decremento, etc.)
- la pila crece hacia direcciones decrecientes y SP apunta a la primera dirección libre.

a.1) Realice la descomposición en una secuencia de microoperaciones, indicando claramente las acciones que se realizan en cada ciclo de reloj, para:

1) el **fetch** (común a todas las instrucciones, ya que todas ocupan una palabra.)

2) las siguientes seis instrucciones (señale con “fetch” la secuencia anterior, supuesta al principio de cada instrucción). En cada caso, indique claramente el ciclo (de haberlo) en que se deba actualizar el registro de estado.

1) LD .AC, /Dir

2) ST .AC, /Dir

- 3) `ADD .AC, #valor`
- 4) `BR /Dir`
- 5) `CALL /Dir`
- 6) `RET`

a.2) Según el apartado anterior, indique en cada caso el número total de ciclos –incluido el fetch– que tardaría en ejecutarse cada instrucción y su equivalente en tiempo. Calcule asimismo el número medio de ciclos y tiempo medio, supuesta una distribución uniforme de las seis instrucciones.

b) En una segunda fase se ha decidido que las microoperaciones que impliquen acceso a memoria ocupen dos ciclos de reloj y el tiempo de ciclo se baje a 60 ns. Para esta nueva situación, se pide:

b.1) Realice las modificaciones que estime necesarias en la descomposición en microoperaciones del apartado a).

b.2) Vuelva a calcular el número de ciclos y el tiempo de ejecución de cada una de las instrucciones y sus valores medios.

b.3) Comente los resultados obtenidos ahora en comparación con los que se obtuvieron anteriormente.