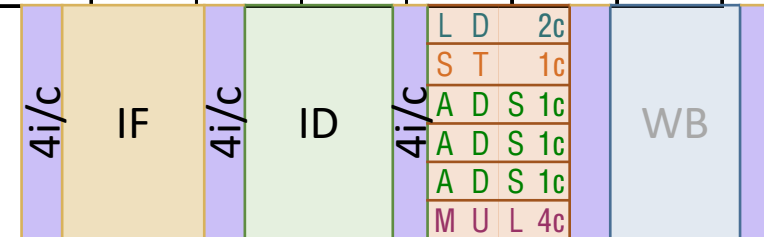


# Ejercicio 1. Orden en el qué se emiten las instrucciones. Tabla de ciclos

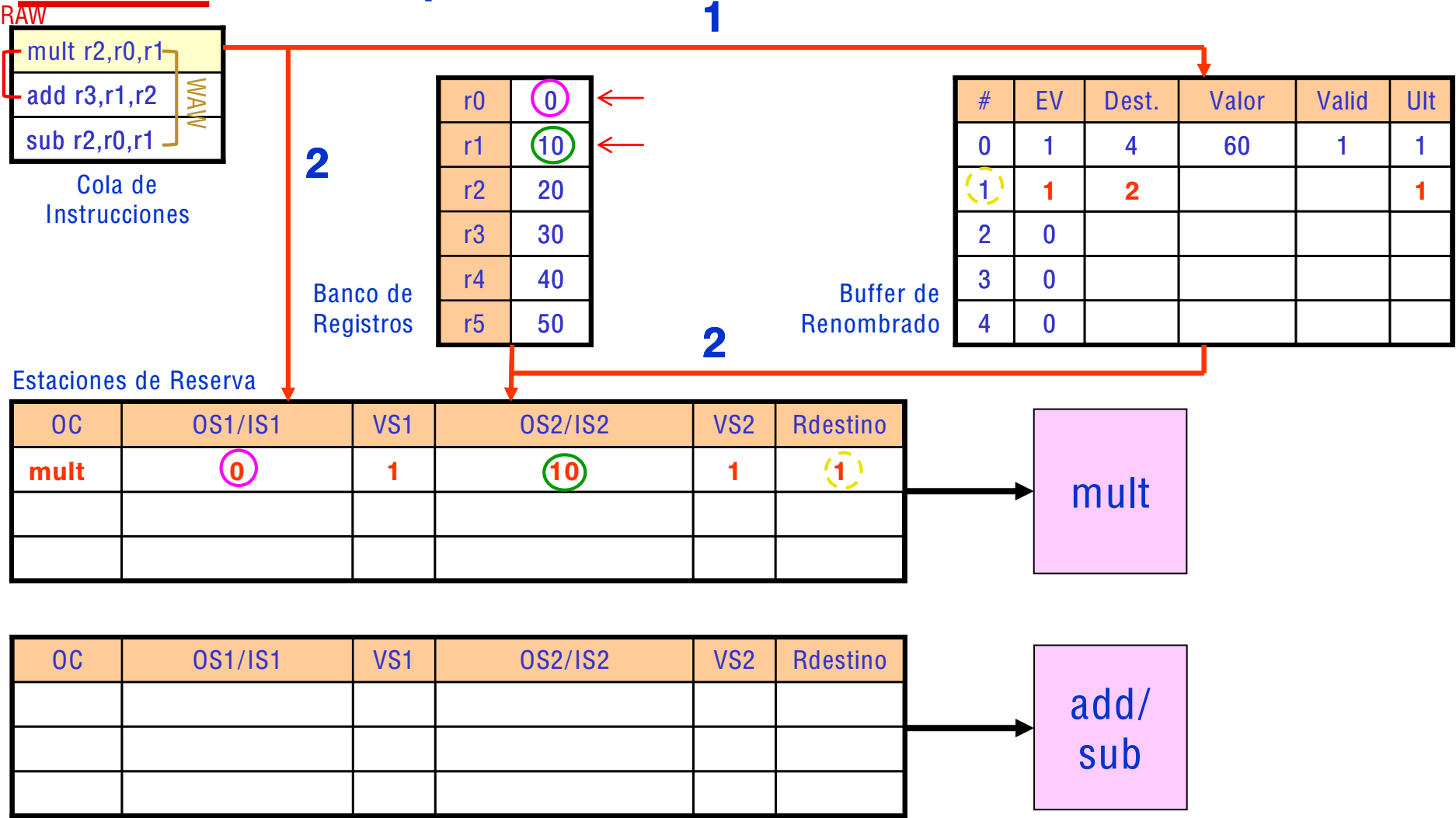
	Ciclos													
INSTRUCCIÓN	1	2	3	4	5	6	7	8	9	10	11	12	13	14
lw r1, 0x1ac														
lw r2, 0xc1f														
add r3, r0, r0														
mul r4, r2, r1														
add r3, r3, r4														
add r5, r0, 0x1ac														
add r6, r0, 0xc1f														
sub r5, r5, #4														
sub r6, r6, #4														
sw (r5), r3														
sw (r6), r4														

- Unidades: 1 LD (2 ciclos), 1 ST (1 ciclo), 3 ADD/SUB (1), 1 MUL (4).
- Emisión **ordenada**.
- Suponemos unidades funcionales **no segmentadas**. Suponemos válido el contenido de un registro la primera vez que se usa en el código.
- WAW y WAR resueltos por **renombrado** (los colores representan renombrado y destacan los RAW)
- Las instrucciones decodificadas, que estarán en la ventana de instrucciones, se pueden emitir si tienen los **operandos disponibles** y hay **unidad disponible para ejecutarlas**.



# Bits de Válido (VS1 y VS2) en la Ventana de instr./estaciones de reserva

## Emisión de la multiplicación



# Ejercicio 1a. Captación (*Instruction Fetch-IF*)

Ciclos

OK	INSTRUCCIÓN	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	lw r1, 0x1ac	IF													
1	lw r2, 0xc1f	IF													
11	add r3, r0, r0	IF													
00	mul r4, r2, r1	IF													
00	add r3, r3, r4		IF												
11	add r5, r0, 0x1ac		IF												
11	add r6, r0, 0xc1f		IF												
01	sub r5, r5, #4		IF												
01	sub r6, r6, #4			IF											
00	sw (r5), r3			IF											
00	sw (r6), r4			IF											

- Las instrucciones se captan y decodifican en el orden del programa
- Ciclo 1: al terminar este ciclo estarán captadas las 4 primeras instrucciones. En *gris* se muestra lo que aún no ha ocurrido.
- Primera columna: bits OK o *válido* (VS1 y VS2) de la ventana de instrucciones para los dos operandos de entrada. En *gris* se muestran los bits de las instrucciones que aún no están en la ventana de instrucciones.

# Ejercicio 1a. Decodificación (*Instruction Decode –ID*)

		Ciclos													
OK	INSTRUCCIÓN	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	lw r1, 0x1ac	IF	ID												
1	lw r2, 0xc1f	IF	ID												
11	add r3, r0, r0	IF	ID												
00	mul r4, r2, r1	IF	ID												
00	add r3, r3, r4		IF	ID											
11	add r5, r0, 0x1ac		IF	ID											
11	add r6, r0, 0xc1f		IF	ID											
01	sub r5, r5, #4		IF	ID											
01	sub r6, r6, #4			IF	ID										
00	sw (r5), r3			IF	ID										
00	sw (r6), r4			IF	ID										

- Las instrucciones se captan y decodifican en el orden del programa
- Ciclo 2: al terminar este ciclo se habrán decodificado las 4 primeras instrucciones (estarán entonces en la ventana de instrucciones) y captado las 4 siguientes.

# Ejercicio 1a. Decodificación (*Instruction Decode –ID*)



		Ciclos													
OK	INSTRUCCIÓN	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	lw r1, 0x1ac	IF	ID												
1	lw r2, 0xc1f	IF	ID												
11	add r3, r0, r0	IF	ID												
00	mul r4, r2, r1	IF	ID												
00	add r3, r3, r4		IF	ID											
11	add r5, r0, 0x1ac		IF	ID											
11	add r6, r0, 0xc1f		IF	ID											
01	sub r5, r5, #4		IF	ID											
01	sub r6, r6, #4			IF	ID										
00	sw (r5), r3			IF	ID										
00	sw (r6), r4			IF	ID										

➤ Ciclo 3:

- Las 4 primeras instrucciones ya están en la ventana de instruc.
- Se puede emitir a ejecución en este ciclo sólo la primera instrucción
- Se podrían emitir las tres instrucciones con operandos disponibles si hubiera 2 unidades de LD, pero sólo hay 1 (dependencia estructural).
- Se podría emitir también la tercera instrucción si la emisión fuese desordenada.

4i/c

IF

4i/c

ID

4i/c

L	D	2c
S	T	1c
A	D	S 1c
A	D	S 1c
A	D	S 1c
M	U	L 4c

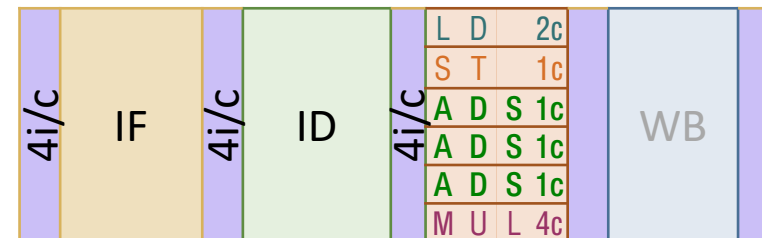
WB

# Ejercicio 1a. Ejecución (*Execution* - Ex)

		Ciclos													
INSTRUCCIÓN		1	2	3	4	5	6	7	8	9	10	11	12	13	14
1 lw r1, 0x1ac		IF	ID	EX	EX										
1 lw r2, 0xc1f		IF	ID												
11 add r3, r0, r0		IF	ID	EX											
00 mul r4, r2, r1		IF	ID												
00 add r3, r3, r4			IF	ID											
11 add r5, r0, 0x1ac			IF	ID	EX										
11 add r6, r0, 0xc1f			IF	ID	EX										
01 sub r5, r5, #4			IF	ID											
01 sub r6, r6, #4				IF	ID										
00 sw (r5), r3				IF	ID										
00 sw (r6), r4				IF	ID										

➤ Ciclo 3: Con emisión desordenada se podría haber emitido también add r3, r0, r0

➤ Ciclo 4: Con emisión desordenada se podrían emitir add r5, r0, 0x1ac y add r6, r0, 0xc1f

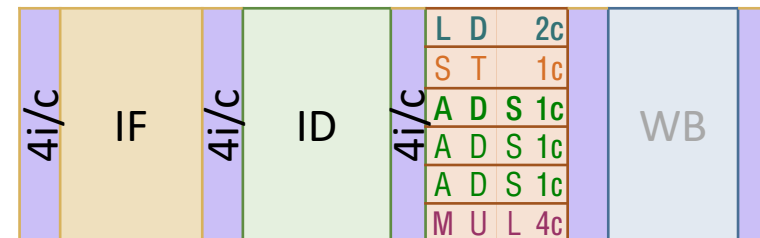


# Ejercicio 1a. Ejecución

		Ciclos													
INSTRUCCIÓN		1	2	3	4	5	6	7	8	9	10	11	12	13	14
1 lw r1, 0x1ac		IF	ID	EX	EX										
1 lw r2, 0xc1f		IF	ID												
11 add r3, r0, r0		IF	ID												
01 mul r4, r2, r1		IF	ID												
00 add r3, r3, r4			IF	ID											
11 add r5, r0, 0x1ac			IF	ID											
11 add r6, r0, 0xc1f			IF	ID											
01 sub r5, r5, #4			IF	ID											
01 sub r6, r6, #4				IF	ID										
00 sw (r5), r3				IF	ID										
00 sw (r6), r4				IF	ID										

## ➤ Ciclo 5:

- Ya está disponible el resultado de lw r1, 0x1ac, que es el 2º operando de mul r4, r2, r1
- Se pueden emitir lw r2, 0xc1f y add r3, r0, r0 (ambas tienen los operandos disponibles y unidad disponible)



# Ejercicio 1a. Ejecución

		Ciclos													
INSTRUCCIÓN		1	2	3	4	5	6	7	8	9	10	11	12	13	14
1 lw r1, 0x1ac		IF	ID	EX	EX										
1 lw r2, 0xc1f		IF	ID			EX	EX								
11 add r3, r0, r0		IF	ID			EX									
01 mul r4, r2, r1		IF	ID												
10 add r3, r3, r4			IF	ID											
11 add r5, r0, 0x1ac			IF	ID											
11 add r6, r0, 0xc1f			IF	ID											
01 sub r5, r5, #4			IF	ID											
01 sub r6, r6, #4				IF	ID										
00 sw (r5), r3				IF	ID										
00 sw (r6), r4				IF	ID										

## ➤ Ciclo 6:

- Ya está disponible el resultado de add r3, r0, r0, que es el 1<sup>er</sup> operando de add r3, r3, r4
- No se puede emitir ninguna instrucción (las que hay después, en el orden del programa, a la última emitida no tiene los dos operandos disponibles)

➤ EX rojo: si la emisión hubiera sido desordenada se hubiera emitido antes. EX verde: si no hubiera habido dependencia estructural se hubiera emitido antes.

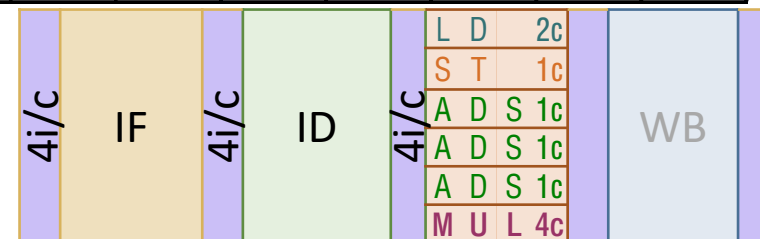


# Ejercicio 1a. Ejecución

		Ciclos													
INSTRUCCIÓN		1	2	3	4	5	6	7	8	9	10	11	12	13	14
1 lw r1, 0x1ac		IF	ID	EX	EX										
1 lw r2, 0xc1f		IF	ID			EX	EX								
11 add r3, r0, r0		IF	ID			EX									
11 mul r4, r2, r1		IF	ID												
10 add r3, r3, r4			IF	ID											
11 add r5, r0, 0x1ac			IF	ID											
11 add r6, r0, 0xc1f			IF	ID											
01 sub r5, r5, #4			IF	ID											
01 sub r6, r6, #4				IF	ID										
00 sw (r5), r3				IF	ID										
00 sw (r6), r4				IF	ID										

## ➤ Ciclo 7:

- Ya está disponible el resultado de lw r2, 0xc1f, que es el 1<sup>er</sup> operando de mul r4, r2, r1 (ya tiene todos los operandos disponibles)
- Se puede emitir mul r4, r2, r1 (tiene todos los operandos disponibles y unidad disponible)



# Ejercicio 1a. Ejecución

		Ciclos													
INSTRUCCIÓN		1	2	3	4	5	6	7	8	9	10	11	12	13	14
1 lw r1, 0x1ac		IF	ID	EX	EX										
1 lw r2, 0xc1f		IF	ID			EX	EX								
11 add r3, r0, r0		IF	ID			EX									
11 mul r4, r2, r1		IF	ID					EX	EX	EX	EX				
10 add r3, r3, r4			IF	ID											
11 add r5, r0, 0x1ac			IF	ID											
11 add r6, r0, 0xc1f			IF	ID											
01 sub r5, r5, #4			IF	ID											
01 sub r6, r6, #4				IF	ID										
00 sw (r5), r3				IF	ID										
00 sw (r6), r4				IF	ID										

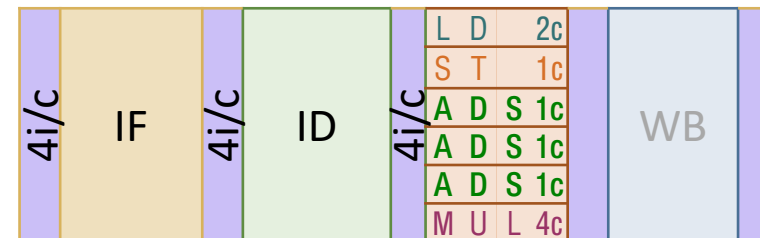
- Unidades: 1 LD (2 ciclos), 1 ST (1 ciclo), 3 ADD/SUB (1), 1 MUL (4). Emisión ordenada.
- Las flechas notan dependencias RAW (ver colores de los registros)
- Ciclos 8-10: No hay instrucciones que se puedan emitir porque la siguiente instrucción, en el orden del programa, a la última emitida necesita que acabe la multiplicación para tener todos sus operandos.

# Ejercicio 1a. Ejecución

		Ciclos													
INSTRUCCIÓN		1	2	3	4	5	6	7	8	9	10	11	12	13	14
1 lw r1, 0x1ac		IF	ID	EX	EX										
1 lw r2, 0xc1f		IF	ID			EX	EX								
11 add r3, r0, r0		IF	ID			EX									
11 mul r4, r2, r1		IF	ID					EX	EX	EX	EX				
11 add r3, r3, r4			IF	ID											
11 add r5, r0, 0x1ac			IF	ID											
11 add r6, r0, 0xc1f			IF	ID											
01 sub r5, r5, #4			IF	ID											
01 sub r6, r6, #4				IF	ID										
00 sw (r5), r3				IF	ID										
01 sw (r6), r4				IF	ID										

## ➤ Ciclo 11:

- Ya está disponible el resultado de mul r4, r2, r1, que es el 2º operando add r3, r3, r4
- Se pueden emitir add r3, r3, r4, add r5, r0, 0x1ac y add r6, r0, 0xc1f (tiene todos los operandos disponibles y, como hay 3 unidades ADD/SUB, también tienen unidades disponibles)



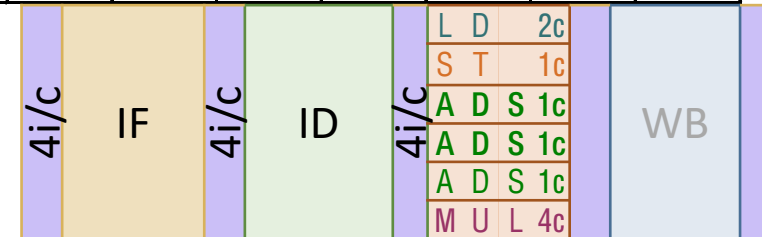
# Ejercicio 1a. Ejecución

		Ciclos													
INSTRUCCIÓN		1	2	3	4	5	6	7	8	9	10	11	12	13	14
1 lw r1, 0x1ac		IF	ID	EX	EX										
1 lw r2, 0xc1f		IF	ID			EX	EX								
11 add r3, r0, r0		IF	ID			EX									
11 mul r4, r2, r1		IF	ID					EX	EX	EX	EX				
11 add r3, r3, r4			IF	ID								EX			
11 add r5, r0, 0x1ac			IF	ID								EX			
11 add r6, r0, 0xc1f			IF	ID								EX			
11 sub r5, r5, #4			IF	ID											
11 sub r6, r6, #4				IF	ID										
01 sw (r5), r3				IF	ID										
01 sw (r6), r4				IF	ID										

## ➤ Ciclo 12:

- Ya están disponibles los resultados de add r3, r3, r4, add r5, r0, 0x1ac y add r6, r0, 0xc1f.
- Se pueden emitir sub r5, r5, #4 y sub r6, r6, #4

(tiene todos los operandos disponibles y unidades disponibles porque hay 3 unidades ADD/SUB)



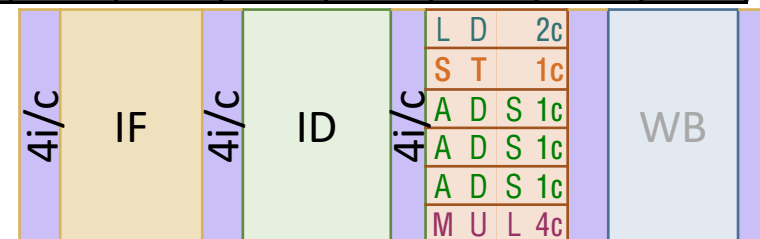
➤ EX rojo: si la emisión hubiera sido desordenada se hubiera emitido antes. EX verde: si no hubiera habido dependencia estructural se hubiera emitido antes.

# Ejercicio 1a. Ejecución

		Ciclos													
INSTRUCCIÓN		1	2	3	4	5	6	7	8	9	10	11	12	13	14
1 lw r1, 0x1ac		IF	ID	EX	EX										
1 lw r2, 0xc1f		IF	ID			EX	EX								
11 add r3, r0, r0		IF	ID			EX									
11 mul r4, r2, r1		IF	ID				EX	EX	EX	EX					
11 add r3, r3, r4			IF	ID							EX				
11 add r5, r0, 0x1ac			IF	ID								EX			
11 add r6, r0, 0xc1f			IF	ID								EX			
11 sub r5, r5, #4			IF	ID									EX		
11 sub r6, r6, #4				IF	ID								EX		
11 sw (r5), r3				IF	ID										
11 sw (r6), r4				IF	ID										

## ➤ Ciclo 13:

- Ya están disponibles los resultados de **sub r5, r5, #4** y **sub r6, r6, #4** (**sw (r5), r3** y **sw (r6), r4** tienen todos los operandos disponibles).
- De las dos instrucciones con operandos disponibles se puede emitir sólo la primera, **sw (r5), r3** porque hay sólo 1 unidad ST.

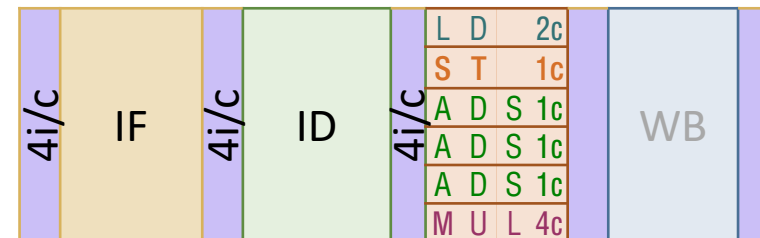


# Ejercicio 1a. Ejecución

		Ciclos													
INSTRUCCIÓN		1	2	3	4	5	6	7	8	9	10	11	12	13	14
1 lw r1, 0x1ac		IF	ID	EX	EX										
1 lw r2, 0xc1f		IF	ID			EX	EX								
11 add r3, r0, r0		IF	ID			EX									
11 mul r4, r2, r1		IF	ID					EX	EX	EX	EX				
11 add r3, r3, r4			IF	ID								EX			
11 add r5, r0, 0x1ac			IF	ID								EX			
11 add r6, r0, 0xc1f			IF	ID								EX			
11 sub r5, r5, #4			IF	ID									EX		
11 sub r6, r6, #4				IF	ID								EX		
11 sw (r5), r3				IF	ID									EX	
11 sw (r6), r4				IF	ID										EX

## ➤ Ciclo 14:

- Ya está disponible el resultado de sw (r5), r3.
- Se puede emitir sw (r6), r4 (porque ya tiene unidad disponible).



# Ejercicio 1a. Ejecución

		Ciclos													
INSTRUCCIÓN		1	2	3	4	5	6	7	8	9	10	11	12	13	14
1 lw r1, 0x1ac		IF	ID	EX	EX										
1 lw r2, 0xc1f		IF	ID			EX	EX								
11 add r3, r0, r0		IF	ID			EX									
11 mul r4, r2, r1		IF	ID				EX	EX	EX	EX					
11 add r3, r3, r4			IF	ID							EX				
11 add r5, r0, 0x1ac			IF	ID							EX				
11 add r6, r0, 0xc1f			IF	ID							EX				
11 sub r5, r5, #4			IF	ID								EX			
11 sub r6, r6, #4				IF	ID								EX		
11 sw (r5), r3				IF	ID									EX	
11 sw (r6), r4				IF	ID										EX

- EX rojo: si la emisión hubiera sido desordenada se hubiera emitido antes. EX en verde: si no hubiera habido dependencia estructural se hubiera emitido antes.
- Las instrucciones están ejecutándose desde el ciclo 3 al ciclo 14.
- Las instrucciones comienzan a procesarse en el ciclo 1, terminarán de procesarse cuando todas hayan modificado los registros de la arquitectura (no se muestra la finalización del procesamiento).

# Ejercicio 1b. Captación (IF) y Decodificación (ID)

Ciclos

INSTRUCCIÓN	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1 lw r1, 0x1ac	IF	ID												
1 lw r2, 0xc1f	IF	ID												
11 add r3, r0, r0	IF	ID												
00 mul r4, r2, r1	IF	ID												
00 add r3, r3, r4		IF	ID											
11 add r5, r0, 0x1ac		IF	ID											
11 add r6, r0, 0xc1f		IF	ID											
01 sub r5, r5, #4		IF	ID											
01 sub r6, r6, #4			IF	ID										
00 sw (r5), r3			IF	ID										
00 sw (r6), r4			IF	ID										

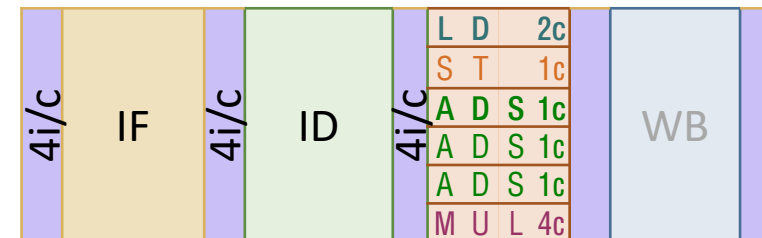
➤ Unidades: 1 LD (2 ciclos), 1 ST (1 ciclo), 3 ADD/SUB (1), 1 MUL (4).

➤ Emisión desordenada.

➤ Suponemos unidades funcionales **no segmentadas**. Suponemos válido el contenido de un registro la primera vez que se usa en el código

➤ WAW y WAR resueltos por **renombrado** (los colores representan renombrado y destacan los RAW)

➤ Las instrucciones decodificadas, que estarán en la ventana de instrucciones, se pueden emitir si tienen los **operandos disponibles** y hay **unidad disponible para ejecutarlas**.

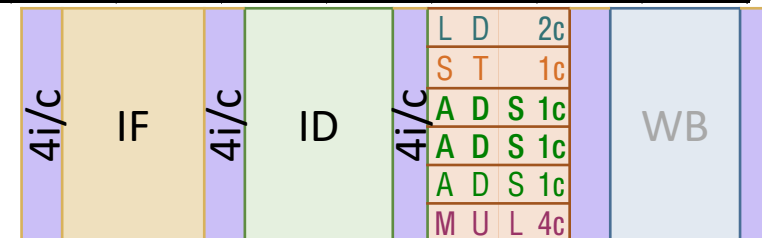




# Ejercicio 1b. Captación (IF) y Decodificación (ID)

Ciclos

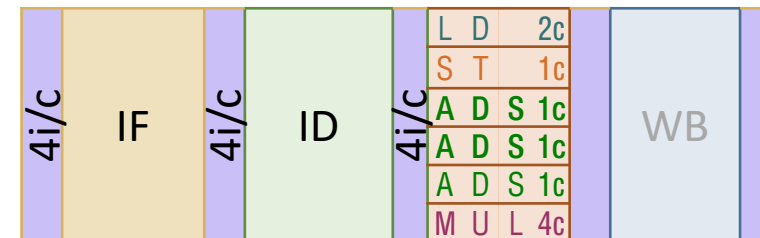
INSTRUCCIÓN	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1 lw r1, 0x1ac	IF	ID	EX	EX										
1 lw r2, 0xc1f	IF	ID												
11 add r3, r0, r0	IF	ID	EX											
00 mul r4, r2, r1	IF	ID												
10 add r3, r3, r4		IF	ID											
11 add r5, r0, 0x1ac		IF	ID											
11 add r6, r0, 0xc1f		IF	ID											
01 sub r5, r5, #4		IF	ID											
01 sub r6, r6, #4			IF	ID										
00 sw (r5), r3			IF	ID										
00 sw (r6), r4			IF	ID										



# Ejercicio 1b. Ejecución (*Execution* - Ex)

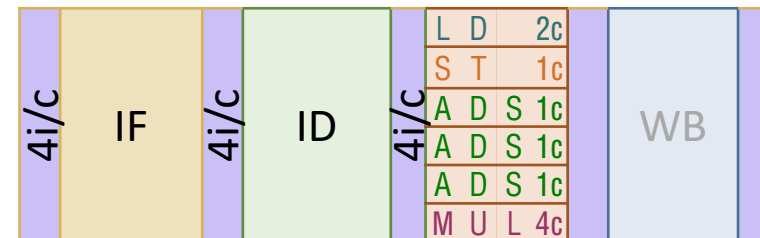
Ciclos

INSTRUCCIÓN	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1 lw r1, 0x1ac	IF	ID	EX	EX										
1 lw r2, 0xc1f	IF	ID												
11 add r3, r0, r0	IF	ID	EX											
01 mul r4, r2, r1	IF	ID												
10 add r3, r3, r4		IF	ID											
11 add r5, r0, 0x1ac		IF	ID	EX										
11 add r6, r0, 0xc1f		IF	ID	EX										
11 sub r5, r5, #4		IF	ID											
11 sub r6, r6, #4			IF	ID										
00 sw (r5), r3			IF	ID										
00 sw (r6), r4			IF	ID										



# Ejercicio 1b. Ejecución

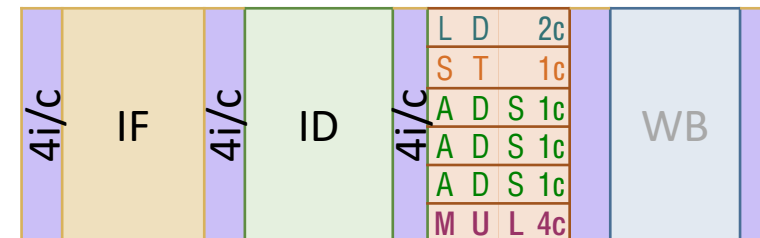
		Ciclos											
INSTRUCCIÓN		1	2	3	4	5	6	7	8	9	10	11	12
1	lw r1, 0x1ac	IF	ID	EX	EX								
1	lw r2, 0xc1f	IF	ID			EX	EX						
11	add r3, r0, r0	IF	ID	EX									
01	mul r4, r2, r1	IF	ID										
10	add r3, r3, r4		IF	ID									
11	add r5, r0, 0x1ac		IF	ID	EX								
11	add r6, r0, 0xc1f		IF	ID	EX								
11	sub r5, r5, #4		IF	ID		EX							
11	sub r6, r6, #4			IF	ID	EX							
10	sw (r5), r3			IF	ID								
10	sw (r6), r4			IF	ID								



# Ejercicio 1b. Ejecución

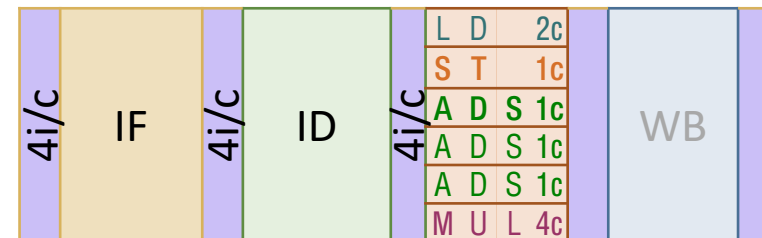
Ciclos

INSTRUCCIÓN	1	2	3	4	5	6	7	8	9	10	11	12
1 lw r1, 0x1ac	IF	ID	EX	EX								
1 lw r2, 0xc1f	IF	ID			EX	EX						
11 add r3, r0, r0	IF	ID	EX									
11 mul r4, r2, r1	IF	ID										
10 add r3, r3, r4		IF	ID									
11 add r5, r0, 0x1ac		IF	ID	EX								
11 add r6, r0, 0xc1f		IF	ID	EX								
11 sub r5, r5, #4		IF	ID		EX							
11 sub r6, r6, #4			IF	ID	EX							
10 sw (r5), r3			IF	ID								
10 sw (r6), r4			IF	ID								



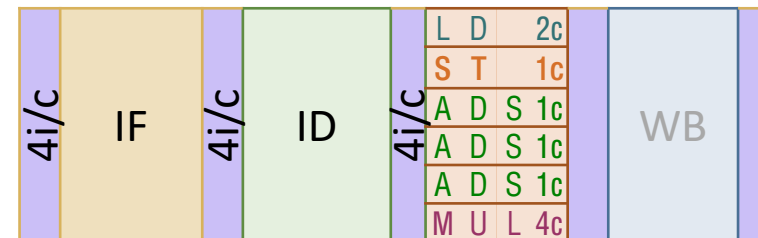
# Ejercicio 1b. Ejecución

		Ciclos											
INSTRUCCIÓN		1	2	3	4	5	6	7	8	9	10	11	12
1	lw r1, 0x1ac	IF	ID	EX	EX								
1	lw r2, 0xc1f	IF	ID			EX	EX						
11	add r3, r0, r0	IF	ID	EX									
11	mul r4, r2, r1	IF	ID					EX	EX	EX	EX		
11	add r3, r3, r4		IF	ID									
11	add r5, r0, 0x1ac		IF	ID	EX								
11	add r6, r0, 0xc1f		IF	ID	EX								
11	sub r5, r5, #4		IF	ID		EX							
11	sub r6, r6, #4			IF	ID	EX							
10	sw (r5), r3			IF	ID								
11	sw (r6), r4			IF	ID								



# Ejercicio 1b. Ejecución

		Ciclos											
INSTRUCCIÓN		1	2	3	4	5	6	7	8	9	10	11	12
1	lw r1, 0x1ac	IF	ID	EX	EX								
1	lw r2, 0xc1f	IF	ID			EX	EX						
11	add r3, r0, r0	IF	ID	EX									
11	mul r4, r2, r1	IF	ID					EX	EX	EX	EX		
11	add r3, r3, r4		IF	ID								EX	
11	add r5, r0, 0x1ac		IF	ID	EX								
11	add r6, r0, 0xc1f		IF	ID	EX								
11	sub r5, r5, #4		IF	ID		EX							
11	sub r6, r6, #4			IF	ID	EX							
11	sw (r5), r3			IF	ID								
11	sw (r6), r4			IF	ID							EX	



# Ejercicio 1b. Ejecución

		Ciclos											
INSTRUCCIÓN		1	2	3	4	5	6	7	8	9	10	11	12
1 lw r1, 0x1ac		IF	ID	EX	EX								
1 lw r2, 0xc1f		IF	ID			EX	EX						
11 add r3, r0, r0		IF	ID	EX									
11 mul r4, r2, r1		IF	ID					EX	EX	EX	EX		
11 add r3, r3, r4			IF	ID								EX	
11 add r5, r0, 0x1ac			IF	ID	EX								
11 add r6, r0, 0xc1f			IF	ID	EX								
11 sub r5, r5, #4			IF	ID		EX							
11 sub r6, r6, #4				IF	ID	EX							
11 sw (r5), r3				IF	ID								EX
11 sw (r6), r4				IF	ID							EX	

➤ Las instrucciones están ejecutándose desde el ciclo 3 al ciclo 12 (en lugar del 14).

¿Qué diferencia habría si las unidades estuvieran segmentadas?

¿Cuánto tiempo tardaría el procesamiento de esta secuencia de instrucciones si se usara ROB y la etapa WB pudiera retirar del ROB 4 instrucciones por ciclo?

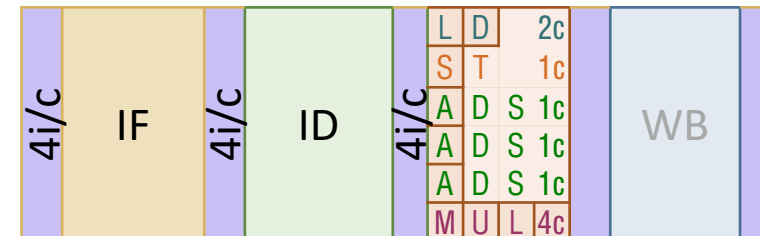
# Ejercicio 1b. Ejecución suponiendo unidades segmentadas

Ciclos

INSTRUCCIÓN	1	2	3	4	5	6	7	8	9	10	11	12
1 lw r1, 0x1ac	IF	ID	EX	EX								
1 lw r2, 0xc1f	IF	ID		EX	EX							
11 add r3, r0, r0	IF	ID	EX									
11 mul r4, r2, r1	IF	ID				EX	EX	EX	EX			
11 add r3, r3, r4		IF	ID							EX		
11 add r5, r0, 0x1ac		IF	ID	EX								
11 add r6, r0, 0xc1f		IF	ID	EX								
11 sub r5, r5, #4		IF	ID		EX							
11 sub r6, r6, #4			IF	ID	EX							
11 sw (r5), r3			IF	ID							EX	
11 sw (r6), r4			IF	ID						EX		

¿Qué diferencia habría si las unidades estuvieran segmentadas?

Las instrucciones estarían ejecutándose desde el ciclo 3 al ciclo 11 (en lugar del 12).



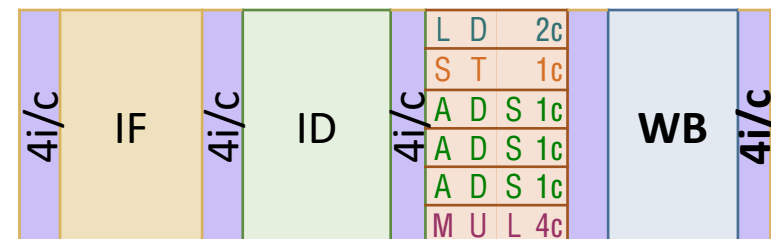


# Ejercicio 1b. Ejecución incluyendo la última etapa de WB

		Ciclos												
INSTRUCCIÓN		1	2	3	4	5	6	7	8	9	10	11	12	13
1 lw r1, 0x1ac		IF	ID	EX	EX	WB								
1 lw r2, 0xc1f		IF	ID			EX	EX	WB						
11 add r3, r0, r0		IF	ID	EX				WB						
11 mul r4, r2, r1		IF	ID				EX	EX	EX	EX	WB			
11 add r3, r3, r4			IF	ID							EX	WB		
11 add r5, r0, 0x1ac			IF	ID	EX							WB		
11 add r6, r0, 0xc1f			IF	ID	EX							WB		
11 sub r5, r5, #4			IF	ID		EX						WB		
11 sub r6, r6, #4				IF	ID	EX							WB	
11 sw (r5), r3				IF	ID							EX	WB	
11 sw (r6), r4				IF	ID							EX		WB

¿Cuánto tiempo tardaría el procesamiento de esta secuencia de instrucciones si se usara ROB y la etapa WB pudiera retirar del ROB 4 instrucciones por ciclo?

**13 ciclos** (suponemos que el almacenamiento en el ROB no consume ningún ciclo)

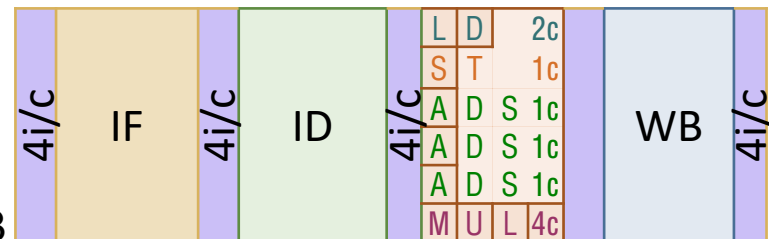


# Ejercicio 1b. Ejecución con unidades segmentadas y la última etapa de WB

		Ciclos											
INSTRUCCIÓN		1	2	3	4	5	6	7	8	9	10	11	12
1 lw r1, 0x1ac		IF	ID	EX	EX	WB							
1 lw r2, 0xc1f		IF	ID		EX	EX	WB						
11 add r3, r0, r0		IF	ID	EX			WB						
11 mul r4, r2, r1		IF	ID				EX	EX	EX	EX	WB		
11 add r3, r3, r4			IF	ID							EX	WB	
11 add r5, r0, 0x1ac			IF	ID	EX							WB	
11 add r6, r0, 0xc1f			IF	ID	EX							WB	
11 sub r5, r5, #4			IF	ID		EX						WB	
11 sub r6, r6, #4				IF	ID	EX							WB
11 sw (r5), r3				IF	ID							EX	WB
11 sw (r6), r4				IF	ID						EX		WB

¿Cuánto tiempo tardaría el procesamiento de esta secuencia de instrucciones si las unidades estuvieran segmentadas y se usara ROB y la etapa WB pudiera retirar del ROB 4 instrucciones por ciclo?

**12 ciclos** (suponemos que el almacenamiento en el ROB no consume ningún ciclo)



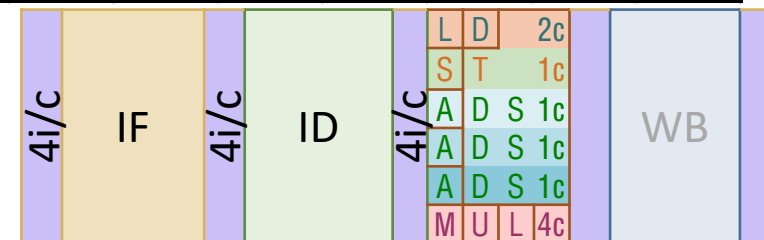
# Ejercicio 1c. Con ER distribuidas

## Captación (IF) y Decodificación (ID)

Ciclos

INSTRUCCIÓN	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1 lw <b>r1</b> , 0x1ac	IF	ID												
1 lw <b>r2</b> , 0xc1f	IF	ID												
11 add <b>r3</b> , r0, r0	IF	ID												
00 mul <b>r4</b> , <b>r2</b> , <b>r1</b>	IF	ID												
00 add <b>r3</b> , <b>r3</b> , <b>r4</b>		IF	ID											
11 add <b>r5</b> , r0, 0x1ac		IF	ID											
11 add <b>r6</b> , r0, 0xc1f		IF	ID											
01 sub <b>r5</b> , <b>r5</b> , #4		IF	ID											
01 sub <b>r6</b> , <b>r6</b> , #4			IF	ID										
00 sw ( <b>r5</b> ), <b>r3</b>			IF	ID										
00 sw ( <b>r6</b> ), <b>r4</b>			IF	ID										

- Unidades: 1 LD (2 ciclos), 1 ST (1 ciclo), 3 ADD/SUB (1), 1 MUL (4).
- Estación de reserva con 3 líneas para cada UF (en la 1ª columna se notan con colores las ER. Envío desde ER **ordenada**).
- Suponemos unidades funcionales **segmentadas**. Suponemos válido el contenido de un registro la primera vez que se usa



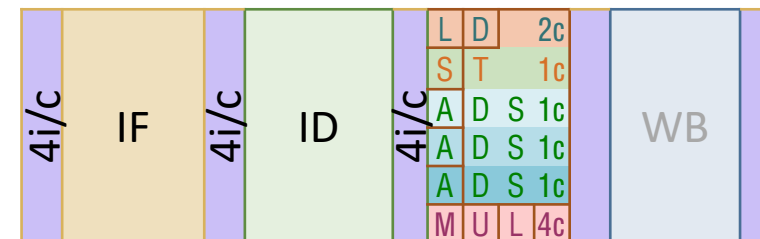
- WAW y WAR resueltos por **renombrado** (los colores en registros representan renombrado y destacan los RAW)
- Las instrucciones decodificadas, que estarán en las ER, se pueden emitir si tienen los **operandos disponibles** y **no hay intr. sin emitir en su ER que están antes en el orden del programa**

# Ejercicio 1c. Ejecución suponiendo unidades segmentadas

Ciclos

INSTRUCCIÓN	1	2	3	4	5	6	7	8	9	10	11	12	13
1 lw r1, 0x1ac	IF	ID	EX	EX									
1 lw r2, 0xc1f	IF	ID		EX	EX								
11 add r3, r0, r0 (1)	IF	ID	EX										
11 mul r4, r2, r1	IF	ID				EX	EX	EX	EX				
11 add r3, r3, r4 (2)		IF	ID							EX			
11 add r5, r0, 0x1ac (3)		IF	ID	EX									
11 add r6, r0, 0xc1f (1)		IF	ID	EX									
11 sub r5, r5, #4 (2)		IF	ID								EX		
11 sub r6, r6, #4 (3)			IF	ID	EX								
11 sw (r5), r3			IF	ID								EX	
11 sw (r6), r4			IF	ID									EX

Se han repartido las instrucciones add y sub entre las 3 ER usando **turno rotatorio**. Se pueden enviar a ejecutar a la vez instrucciones en distintas ER. El uso de las estaciones de reserva con envío ordenado hace (a) que la 1ª instrucción sub no se pueda enviar hasta haber enviado la 2ª instrucción add porque están en la misma estación de reserva y (b) que la 2ª instrucción sw no pueda ejecutarse antes que la primera.

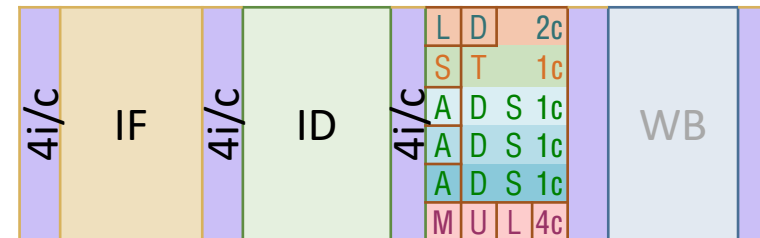


# Ejercicio 1c. Ejecución suponiendo unidades segmentadas

Ciclos

INSTRUCCIÓN	1	2	3	4	5	6	7	8	9	10	11	12
1 lw r1, 0x1ac	IF	ID	EX	EX								
1 lw r2, 0xc1f	IF	ID		EX	EX							
11 add r3, r0, r0 (1)	IF	ID	EX									
11 mul r4, r2, r1	IF	ID				EX	EX	EX	EX			
11 add r3, r3, r4 (2)		IF	ID							EX		
11 add r5, r0, 0x1ac (3)		IF	ID	EX								
11 add r6, r0, 0xc1f (1)		IF	ID	EX								
11 sub r5, r5, #4 (3)		IF	ID		EX							
11 sub r6, r6, #4 (1)			IF	ID	EX							
11 sw (r5), r3			IF	ID							EX	
11 sw (r6), r4			IF	ID								EX

Se han repartido las instrucciones add y sub entre las 3 ER usando **turno rotatorio, ocupación y disponibilidad de operandos en las instrucciones que se emiten a la vez a la misma unidad** (si una instr. emitida a la vez tiene operandos disponibles se sabe que estará poco en la estación). Con este reparto, la ejecución de todas las instr. excepto la última, coincide con la del caso con estación centralizada 1b. *Se pueden enviar a ejecutar a la vez instrucciones en distintas ER.* El uso de las estaciones de reserva con envío ordenado hace que la 2ª instrucción sw no pueda ejecutarse antes que la 1ª.



# Ejercicio 2. Tabla de ciclos. Emisión desordenada

Ciclos

INSTRUCCIÓN	1	2	3	4	5	6	7	8	9	10	11	12	13	14
lw r3, 0x10a														
addi r2, r0, #128														
add r1, r0, 0x0a														
lw r4, 0(r1)														
lw r5, -8(r1)														
mult r6, r5, r3														
add r5, r6, r3														
add r6, r4, r3														
sw 0(r1), r6														
sw -8(r1), r5														
sub r2, r2, #16														

# Ejercicio 2b. Captación y decodificación

Ciclos

INSTRUCCIÓN	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
lw r3, 0x10a	IF																	
addi r2, r0, #128	IF																	
add r1, r0, 0x0a	IF																	
lw r4, 0(r1)	IF																	
lw r5, -8(r1)		IF																
mult r6, r5, r3		IF																
add r5, r6, r3		IF																
add r6, r4, r3		IF																
sw 0(r1), r6			IF															
sw -8(r1), r5			IF															
sub r2, r2, #16			IF															

# Ejercicio 2b. Captación y decodificación

Ciclos

INSTRUCCIÓN	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1 lw r3, 0x10a	IF	ID																
11 addi r2, r0, #128	IF	ID																
11 add r1, r0, 0x0a	IF		ID															
0 lw r4, 0(r1)	IF		ID															
0 lw r5, -8(r1)		IF		ID														
0 mult r6, r5, r3		IF		ID														
00 add r5, r6, r3		IF			ID													
00 add r6, r4, r3		IF			ID													
00 sw 0(r1), r6			IF			ID												
00 sw -8(r1), r5			IF			ID												
01 sub r2, r2, #16			IF				ID											



# Ejercicio 2b. Ejecución

Ciclos

INSTRUCCIÓN	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1 lw r3, 0x10a	IF	ID	EX	EX														
11 addi r2, r0, #128	IF	ID	EX															
11 add r1, r0, 0x0a	IF		ID															
0 lw r4, 0(r1)	IF		ID															
0 lw r5, -8(r1)		IF		ID														
0 mult r6, r5, r3		IF		ID														
00 add r5, r6, r3		IF			ID													
00 add r6, r4, r3		IF			ID													
00 sw 0(r1), r6			IF			ID												
00 sw -8(r1), r5			IF			ID												
11 sub r2, r2, #16			IF				ID											

# Ejercicio 2b. Ejecución

Ciclos

INSTRUCCIÓN	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1 lw r3, 0x10a	IF	ID	EX	EX														
11 addi r2, r0, #128	IF	ID	EX															
11 add r1, r0, 0x0a	IF		ID	EX														
1 lw r4, 0(r1)	IF		ID															
1 lw r5, -8(r1)		IF		ID														
01 mult r6, r5, r3		IF		ID														
01 add r5, r6, r3		IF			ID													
01 add r6, r4, r3		IF			ID													
10 sw 0(r1), r6			IF			ID												
10 sw -8(r1), r5			IF			ID												
11 sub r2, r2, #16			IF				ID											

# Ejercicio 2b. Ejecución

Ciclos

INSTRUCCIÓN	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1 lw r3, 0x10a	IF	ID	EX	EX														
11 addi r2, r0, #128	IF	ID	EX															
11 add r1, r0, 0x0a	IF		ID	EX														
1 lw r4, 0(r1)	IF		ID		EX	EX												
1 lw r5, -8(r1)		IF		ID														
01 mult r6, r5, r3		IF		ID														
01 add r5, r6, r3		IF			ID													
01 add r6, r4, r3		IF			ID													
10 sw 0(r1), r6			IF			ID												
10 sw -8(r1), r5			IF			ID												
11 sub r2, r2, #16			IF				ID											

# Ejercicio 2b. Ejecución

Ciclos

INSTRUCCIÓN	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1 lw r3, 0x10a	IF	ID	EX	EX														
11 addi r2, r0, #128	IF	ID	EX															
11 add r1, r0, 0x0a	IF		ID	EX														
1 lw r4, 0(r1)	IF		ID		EX	EX												
1 lw r5, -8(r1)		IF		ID														
01 mult r6, r5, r3		IF		ID														
01 add r5, r6, r3		IF			ID													
11 add r6, r4, r3		IF			ID													
10 sw 0(r1), r6			IF			ID												
10 sw -8(r1), r5			IF			ID												
11 sub r2, r2, #16			IF				ID											

# Ejercicio 2b. Ejecución

Ciclos

INSTRUCCIÓN	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1 lw r3, 0x10a	IF	ID	EX	EX														
11 addi r2, r0, #128	IF	ID	EX															
11 add r1, r0, 0x0a	IF		ID	EX														
1 lw r4, 0(r1)	IF		ID		EX	EX												
1 lw r5, -8(r1)		IF		ID			EX	EX										
01 mult r6, r5, r3		IF		ID														
01 add r5, r6, r3		IF			ID													
11 add r6, r4, r3		IF			ID		EX											
11 sw 0(r1), r6			IF			ID												
10 sw -8(r1), r5			IF			ID												
11 sub r2, r2, #16			IF				ID											

# Ejercicio 2b. Ejecución

Ciclos

INSTRUCCIÓN	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1 lw r3, 0x10a	IF	ID	EX	EX														
11 addi r2, r0, #128	IF	ID	EX															
11 add r1, r0, 0x0a	IF		ID	EX														
1 lw r4, 0(r1)	IF		ID		EX	EX												
1 lw r5, -8(r1)		IF		ID			EX	EX										
11 mult r6, r5, r3		IF		ID														
01 add r5, r6, r3		IF			ID													
11 add r6, r4, r3		IF			ID		EX											
11 sw 0(r1), r6			IF			ID		EX										
10 sw -8(r1), r5			IF			ID												
11 sub r2, r2, #16			IF				ID	EX										

# Ejercicio 2b. Ejecución

Ciclos

INSTRUCCIÓN	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1 lw r3, 0x10a	IF	ID	EX	EX														
11 addi r2, r0, #128	IF	ID	EX															
11 add r1, r0, 0x0a	IF		ID	EX														
1 lw r4, 0(r1)	IF		ID		EX	EX												
1 lw r5, -8(r1)		IF		ID			EX	EX										
11 mult r6, r5, r3		IF		ID					EX	EX	EX	EX	EX	EX				
11 add r5, r6, r3		IF			ID													
11 add r6, r4, r3		IF			ID		EX											
11 sw 0(r1), r6			IF			ID		EX										
10 sw -8(r1), r5			IF			ID												
11 sub r2, r2, #16			IF				ID	EX										

# Ejercicio 2b. Ejecución

Ciclos

INSTRUCCIÓN	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1 lw r3, 0x10a	IF	ID	EX	EX														
11 addi r2, r0, #128	IF	ID	EX															
11 add r1, r0, 0x0a	IF		ID	EX														
1 lw r4, 0(r1)	IF		ID		EX	EX												
1 lw r5, -8(r1)		IF		ID			EX	EX										
11 mult r6, r5, r3		IF		ID				EX	EX	EX	EX	EX	EX	EX				
11 add r5, r6, r3		IF			ID													
11 add r6, r4, r3		IF			ID		EX											
11 sw 0(r1), r6			IF			ID		EX										
11 sw -8(r1), r5			IF			ID												
11 sub r2, r2, #16			IF				ID	EX										



# Ejercicio 2b. Ejecución

Ciclos

INSTRUCCIÓN	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1 lw r3, 0x10a	IF	ID	EX	EX														
11 addi r2, r0, #128	IF	ID	EX															
11 add r1, r0, 0x0a	IF		ID	EX														
1 lw r4, 0(r1)	IF		ID		EX	EX												
1 lw r5, -8(r1)		IF		ID			EX	EX										
11 mult r6, r5, r3		IF		ID				EX	EX	EX	EX	EX	EX	EX				
11 add r5, r6, r3		IF			ID										EX			
11 add r6, r4, r3		IF			ID		EX											
11 sw 0(r1), r6			IF			ID		EX										
11 sw -8(r1), r5			IF			ID										EX		
11 sub r2, r2, #16			IF				ID	EX										

# Ejercicio 2b. ROB

Ciclos

INSTRUCCIÓN	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
lw r3, 0x10a	IF	ID	EX	EX	RB													
addi r2, r0, #128	IF	ID	EX	RB														
add r1, r0, 0x0a	IF		ID	EX	RB													
lw r4, 0(r1)	IF		ID		EX	EX	RB											
lw r5, -8(r1)		IF		ID			EX	EX	RB									
mult r6, r5, r3		IF		ID				EX	EX	EX	EX	EX	EX	EX	RB			
add r5, r6, r3		IF			ID										EX	RB		
add r6, r4, r3		IF			ID		EX	RB										
sw 0(r1), r6			IF			ID		EX	RB									
sw -8(r1), r5			IF			ID										EX	RB	
sub r2, r2, #16			IF				ID	EX		RB								

En el ciclo 9 no se puede escribir en el ROB el resultado de las tres instrucciones que terminan

# Ejercicio 2b. WB

Ciclos

INSTRUCCIÓN	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
lw r3, 0x10a	IF	ID	EX	EX	RB	WB												
addi r2, r0, #128	IF	ID	EX	RB		WB												
add r1, r0, 0x0a	IF		ID	EX	RB	WB												
lw r4, 0(r1)	IF		ID		EX	EX	RB	WB										
lw r5, -8(r1)		IF		ID			EX	EX	RB	WB								
mult r6, r5, r3		IF		ID				EX	EX	EX	EX	EX	EX	EX	RB	WB		
add r5, r6, r3		IF			ID										EX	RB	WB	
add r6, r4, r3		IF			ID		EX	RB									WB	
sw 0(r1), r6			IF			ID		EX	RB								WB	
sw -8(r1), r5			IF			ID										EX	RB	WB
sub r2, r2, #16			IF				ID	EX		RB								WB