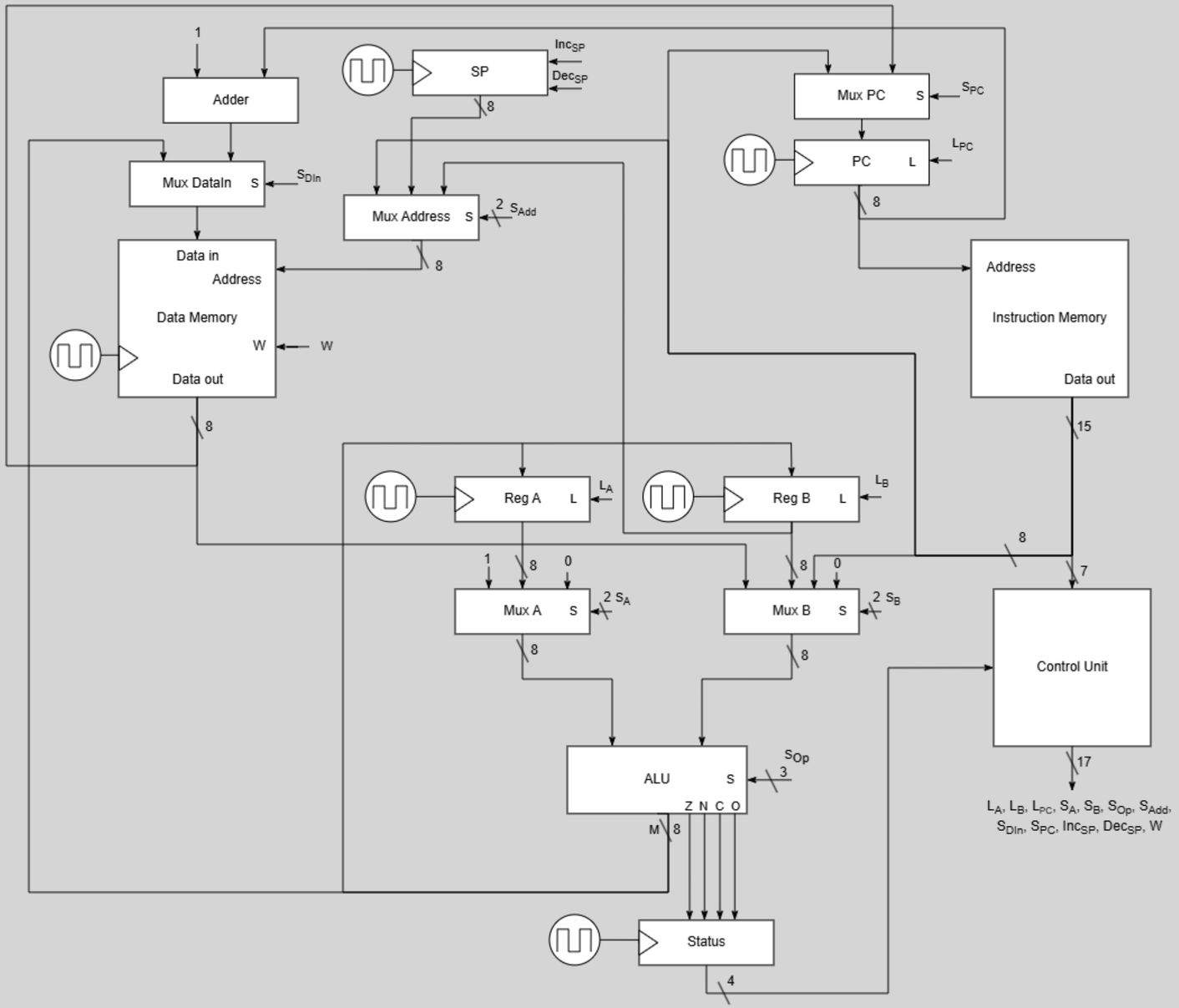
SUB RUTINAS



PONTIFICIA
UNIVERSIDAD
CATÓLICA
DE CHILE

Repaso Ayudantía - Carlos Canales y Nicolás Romo - IIC2343

¿DÓNDE ESTAMOS?



IDEA DE UNA SUBRUTINA

Idea guía: **Saltar** a un bloque de código, **ejecutarlo** y **volver** al lugar correcto

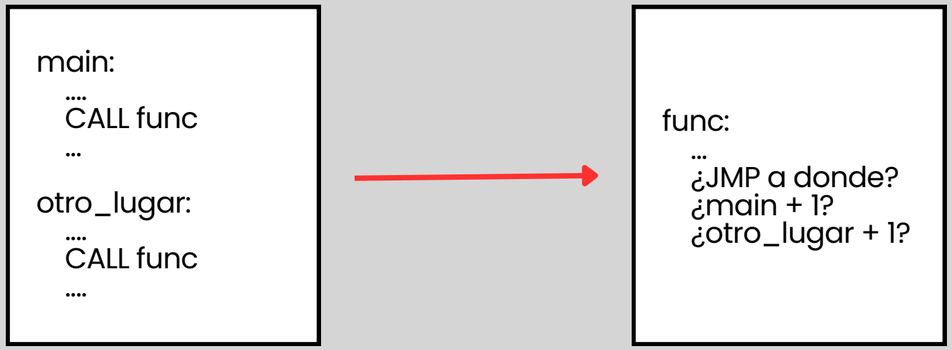


¿QUÉ PROBLEMA NOS QUEDA

Con saltos es **posible** ejecutar la secuencia

0 → 1 → 2 → 4 → 5 → 3

¿Por qué saltos no bastan?



LLegar a la funcion es **fácil**, básta con saltar al label.

Lo **difícil** es saber a dónde **volver**.

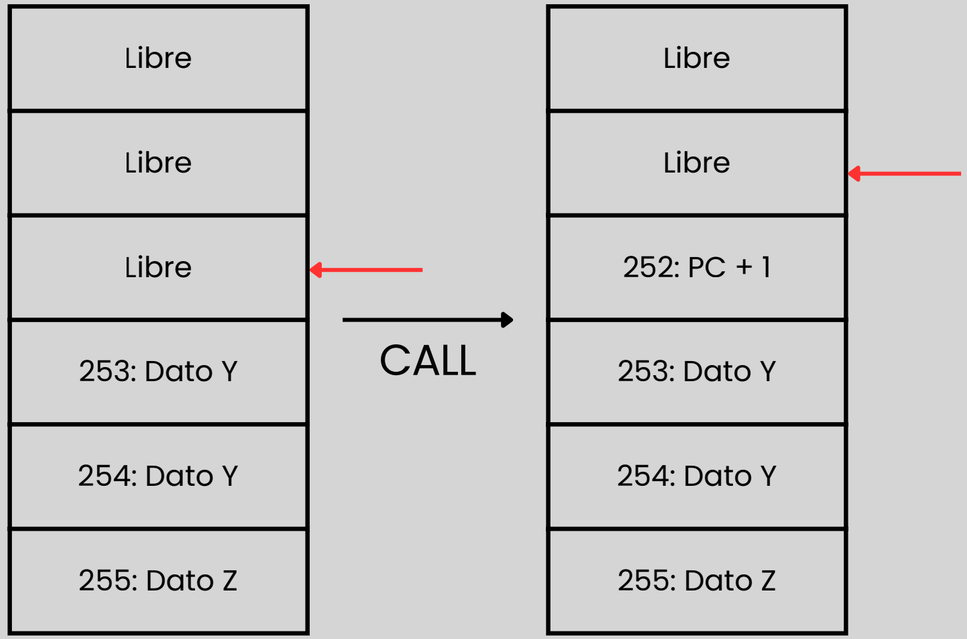
CALL

No es sólo un salto, además guarda la dirección de retorno

CALL func

Hace **3** cosas

1. Mem[SP] = PC+1
2. SP = SP-1
3. Carga PC con la dirección label (Salta)



Recuerden: Dura **1 ciclo**

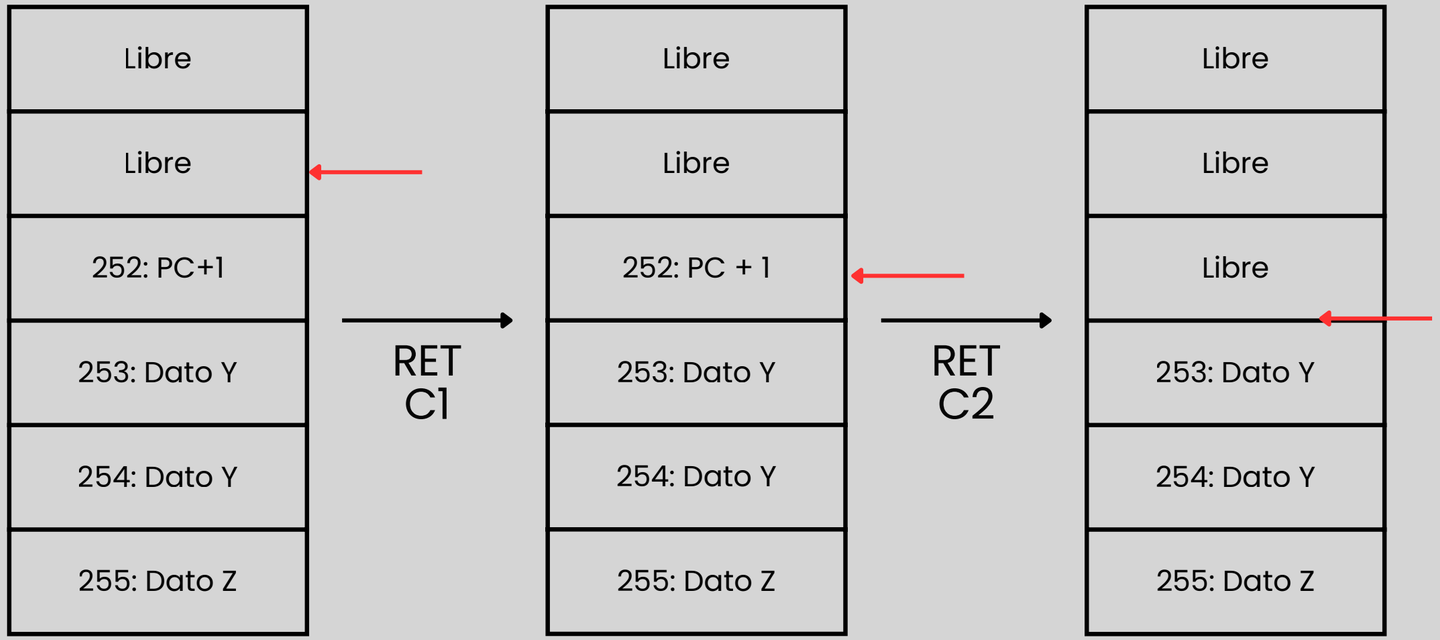
RET

No adivina, lee el stack

RET

Hace **3** cosas

1. SP++
2. PC = Mem[SP]
3. Lpc = 1



Recuerden: Dura **2 ciclos**

PUSH Y POP

NO cambian el PC, sirven para **guardar** y **recuperar** datos

PUSH A

Hace 2 cosas

1. Mem[SP] = A
2. SP = SP-1

Recuerden: Dura **1 ciclo**

POP A

Hace 2 cosas

1. SP++
2. A=Mem[SP]

Recuerden: Dura **2 ciclos**

El orden es MUY importante, ¿Por qué?

UNA TABLA IMPORTANTE

Instrucción	Condición	Operación
CALL	Dir	Mem[SP] = PC+1; SP--; PC = Dir
RET	-	SP++
		PC=Mem[SP]
PUSH	Reg	Mem[SP] = Reg; SP--
POP	Reg	SP++
		Reg=Mem[SP]

Regla mental:
CALL/RET controlan **flujo**,
PUSH/POP controlan **datos**