

ARQUITECTURA DE COMPUTADORES

2º Grado en Informática

Página 1 de 2

Práctica 2

Control microprogramado en SiCoMe

Ejercicio 1: (1 punto)

A partir del repertorio básico y la lógica de control de bifurcación proporcionados en la práctica 1, extraer la microprogramación, utilizando la hoja de Excel que se facilita en la práctica 2, las instrucciones del repertorio básico.

- Ciclo de Fetch y HALT
- JMP y JMPI
- CSR
- ISZ y ISZ_R

Indicar en cada ciclo qué micro operación(es) se está(n) realizando, la condición de habilitación si la hubiera, y la operación de la lógica de control de bifurcación (incrementa | bifurca | carga) y sus posibles condiciones.

Ejercicio 2: (2 puntos)

Añadir al repertorio expandido la siguiente instrucción:

- DEC *dir*. Reduce en 1 el contenido de la dirección *dir*

El contenido del acumulador debe quedar tal como estaba cuando se llamó a la instrucción

Ejercicio 3: (2 puntos)

Expandir el repertorio básico incluyendo las siguientes instrucciones, guardar como *repertorio_expandido*:

- PUSH *dir*. Almacena el contenido de la dirección *dir* en la pila.
- PUSHI *dir*: Almacena el contenido de la dirección almacenada en *dir* en la pila (indirecta).
- POP *dir*: Almacena el último valor de la pila en la dirección *dir*.
- POPI *dir*: Almacena el último valor de la pila en la dirección almacenada en la dirección *dir* (indirecta).

Ejercicio 4: (2.5 puntos)

Añadir al repertorio expandido las siguiente instrucciones:

- ADDV11 val. Suma el valor inmediato codificado en la instrucción en los 11 bits menos significativos de la propia instrucción al contenido del acumulador.
- SUBV11 val. Resta el valor inmediato codificado en la instrucción en los 11 bits menos significativos de la propia instrucción al contenido del acumulador.

En ambas instrucciones, los valores pasados como argumento se interpretarán como números positivos.

Ejercicio 5: (2.5 puntos)

Añadir al repertorio extendido la siguiente instrucción:

- JEG11 v11. Salta la siguiente instrucción si el valor contenido en el acumulador es menor o igual que el valor inmediato codificado en los 11 bits menos significativos de la propia instrucción.

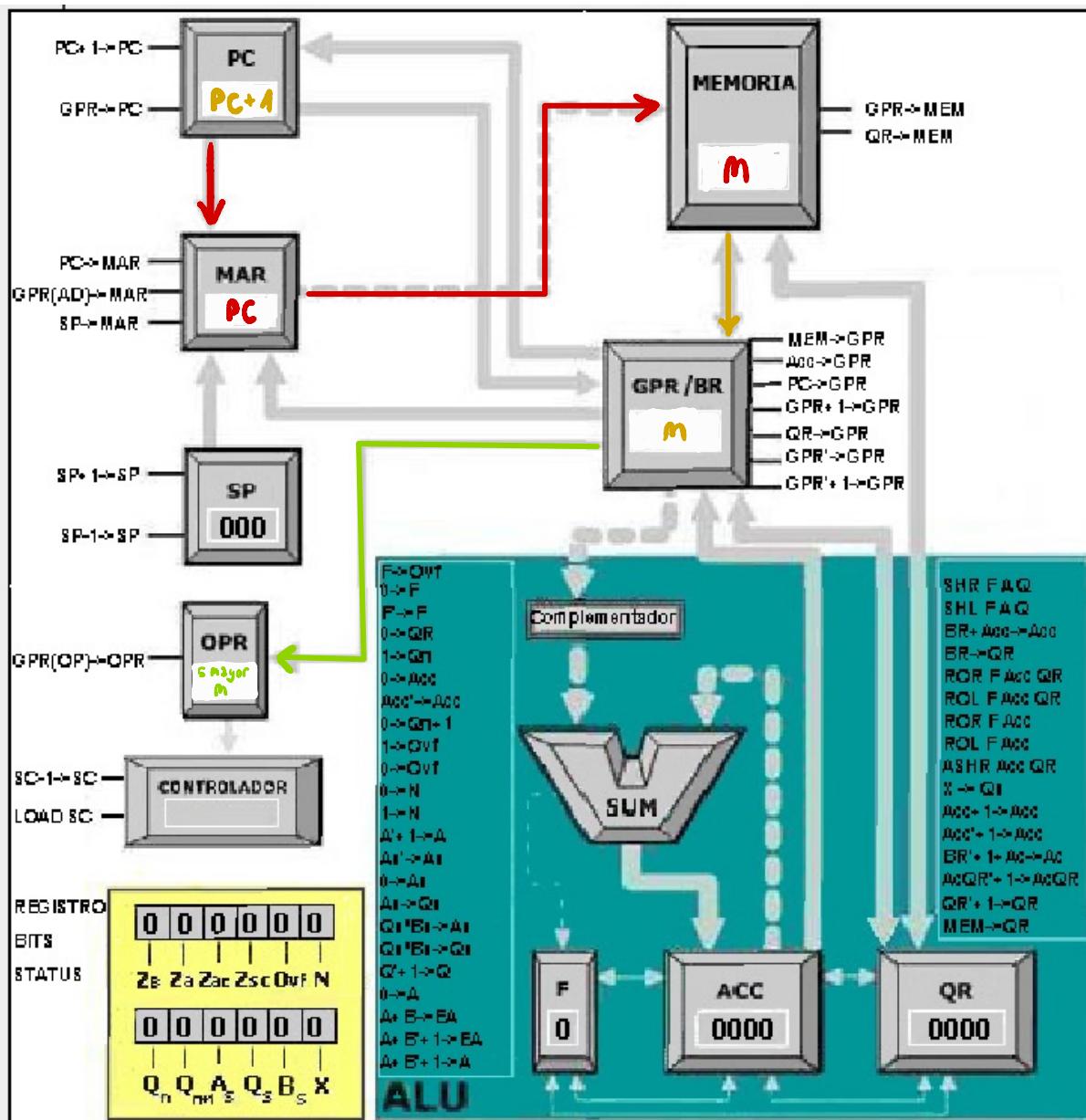
Ejercicio 1

C.B.

PC → MAR || Habilita || Incrementa ●

PC+1 → PC || M → GPR || Habilita || Incrementa ●

GPR(OP) → OPR || Habilita || Carga ●

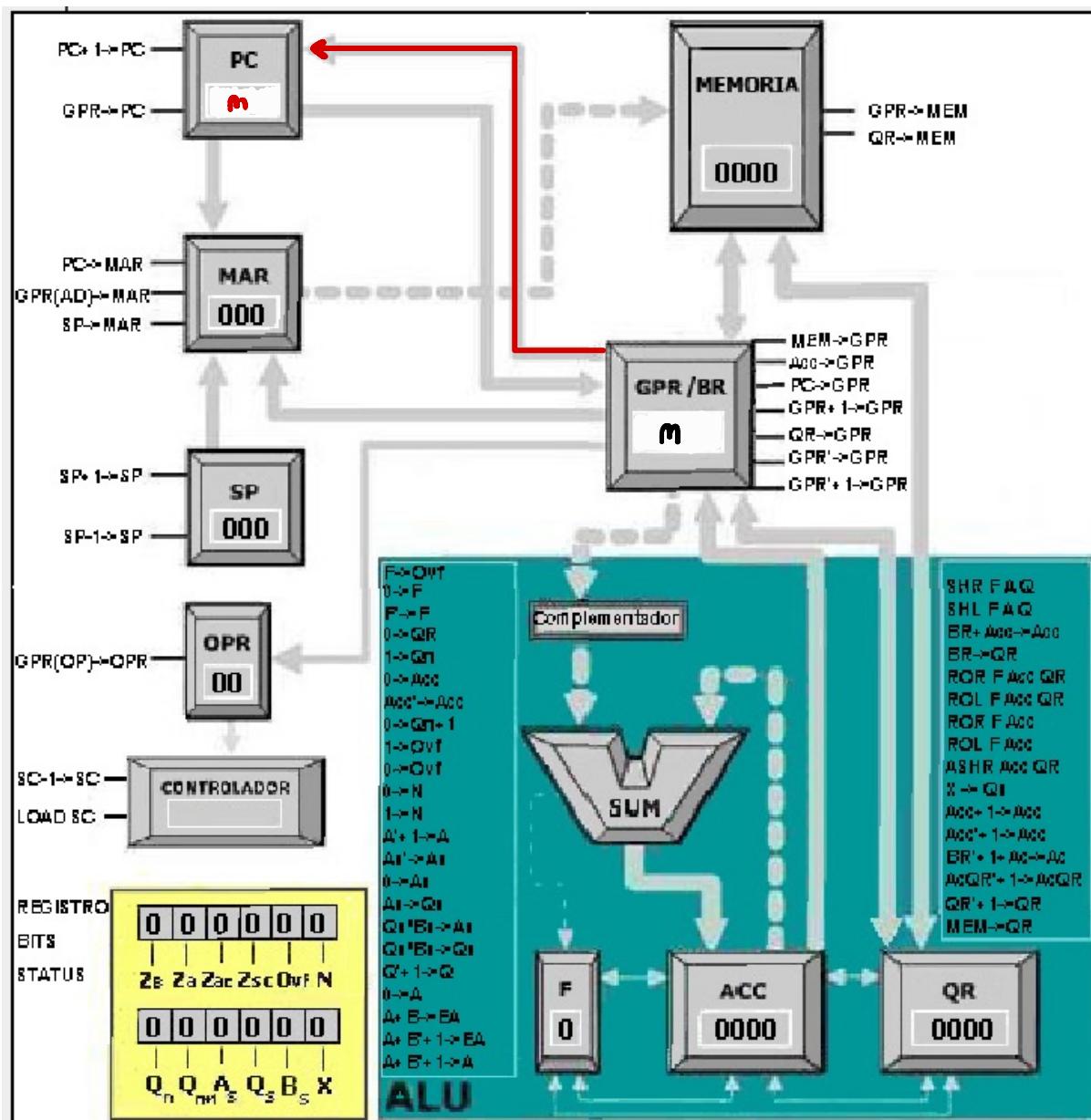


HALT

Nada

JMP m

GPR > PC || Habilita || Bifurca ●

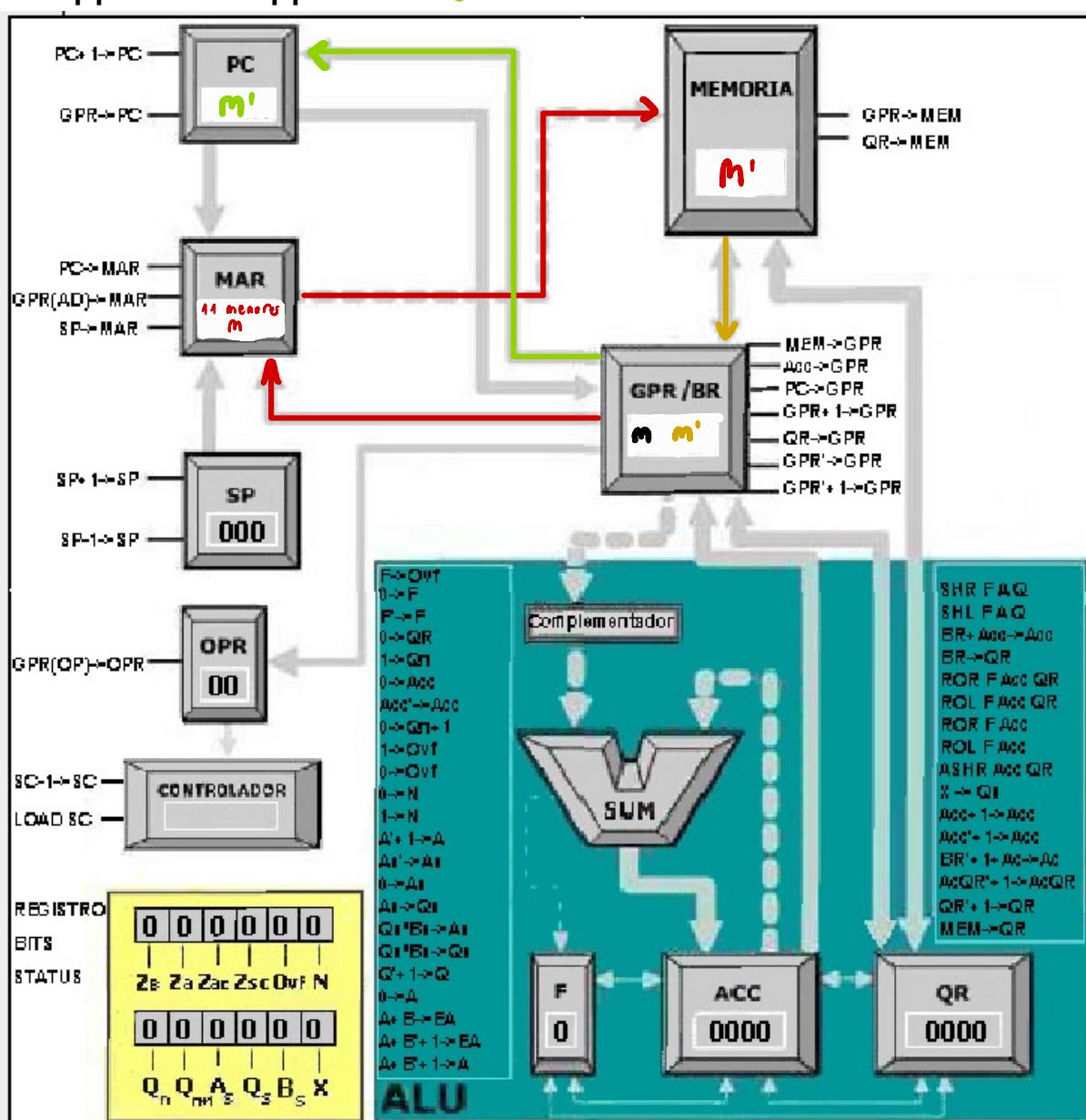


JMPI m

GPR(AD) -> MAR || Habilita || Incrementa ●

M -> GPR || Habilita || Incrementa ○

GPR -> PC || Habilita || Bifurca ●



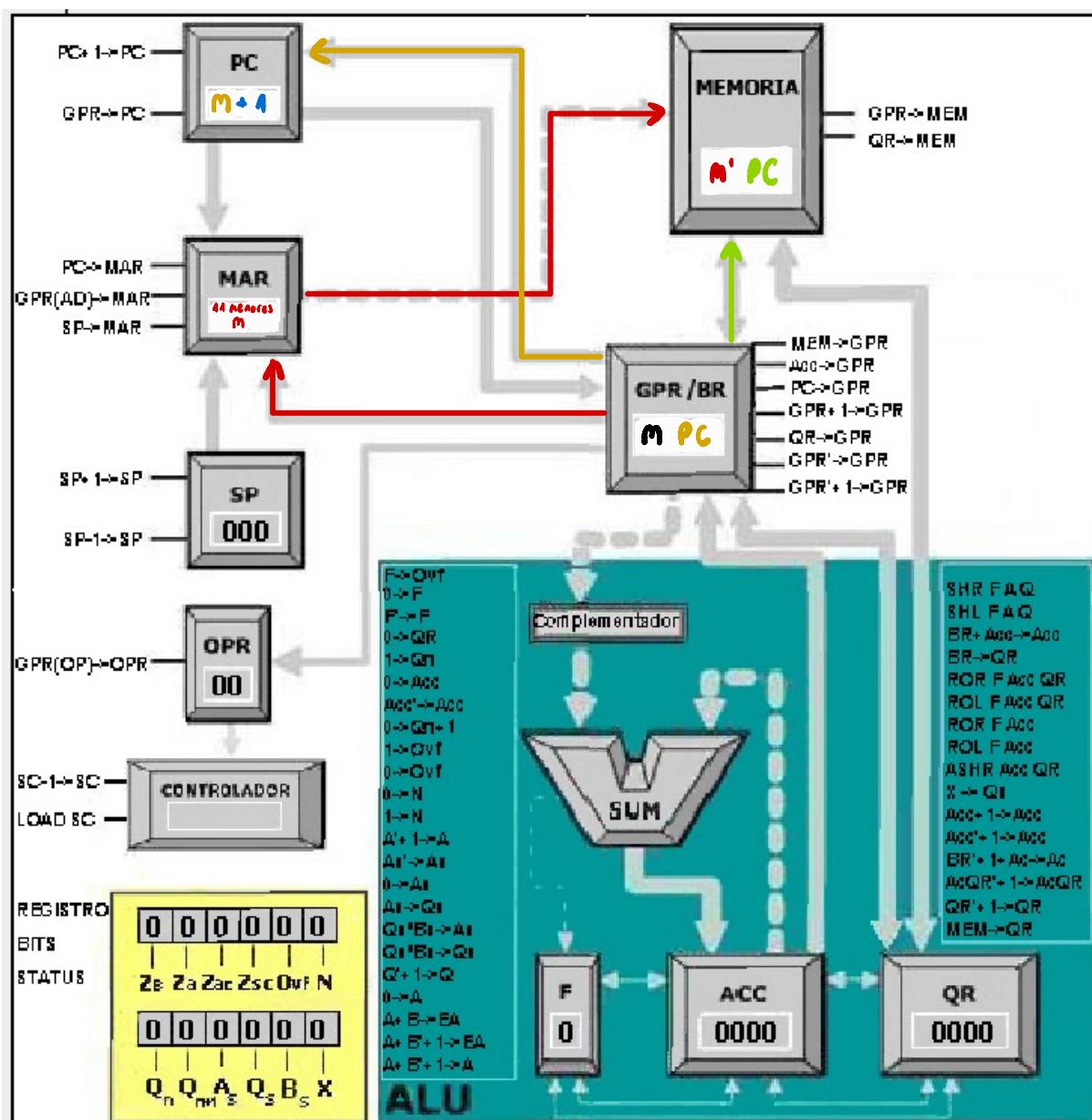
CSR m

GPR(AD) -> MAR || Habilita || Incrementa ●

GPR -> PC || PC -> GPR || Habilita || Incrementa

GPR -> M || Habilida || Incrementa

PC+1 -> PC || Habilita || Bifurca



ISZ m

GPR(AD) -> MAR || Habilita || Incrementa ●

M -> GPR || Habilita || Incrementa ●

GPR+1 -> GPR || Habilita || Incrementa ●

GPR -> M || Habilita || Incrementa ●

Habilita || Zb = 0 Bifurca, Zb = 1 Incrementa

PC+1 -> PC || Habilita || Bifurca ●

ISZ_R m

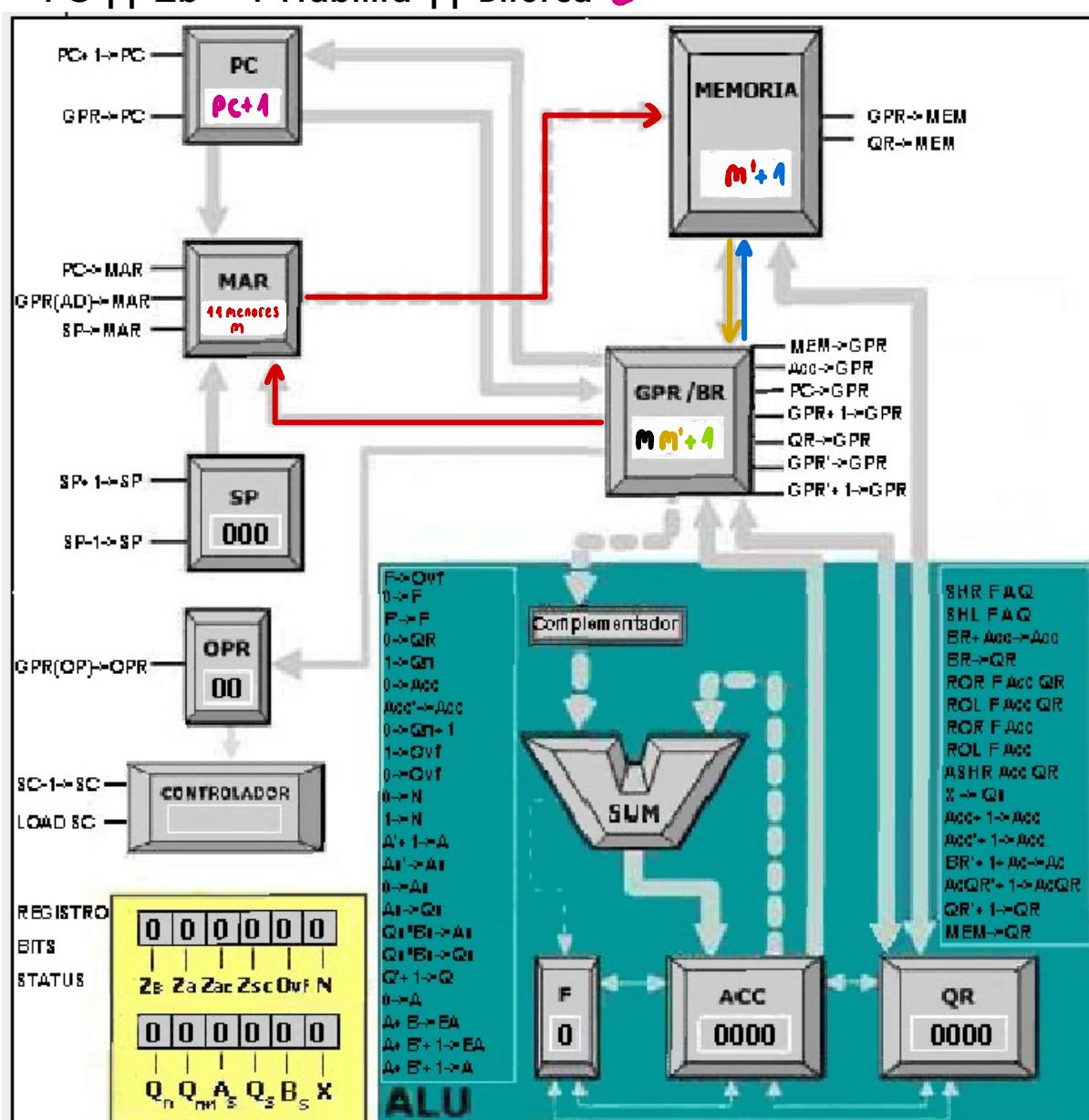
GPR(AD) -> MAR || Habilita || Incrementa ●

M -> GPR || Habilita || Incrementa ●

GPR+1 -> GPR || Habilita || Incrementa ●

GPR -> M || Habilita || Incrementa ●

PC+1 -> PC || Zb = 1 Habilita || Bifurca ●



Ejercicio 2

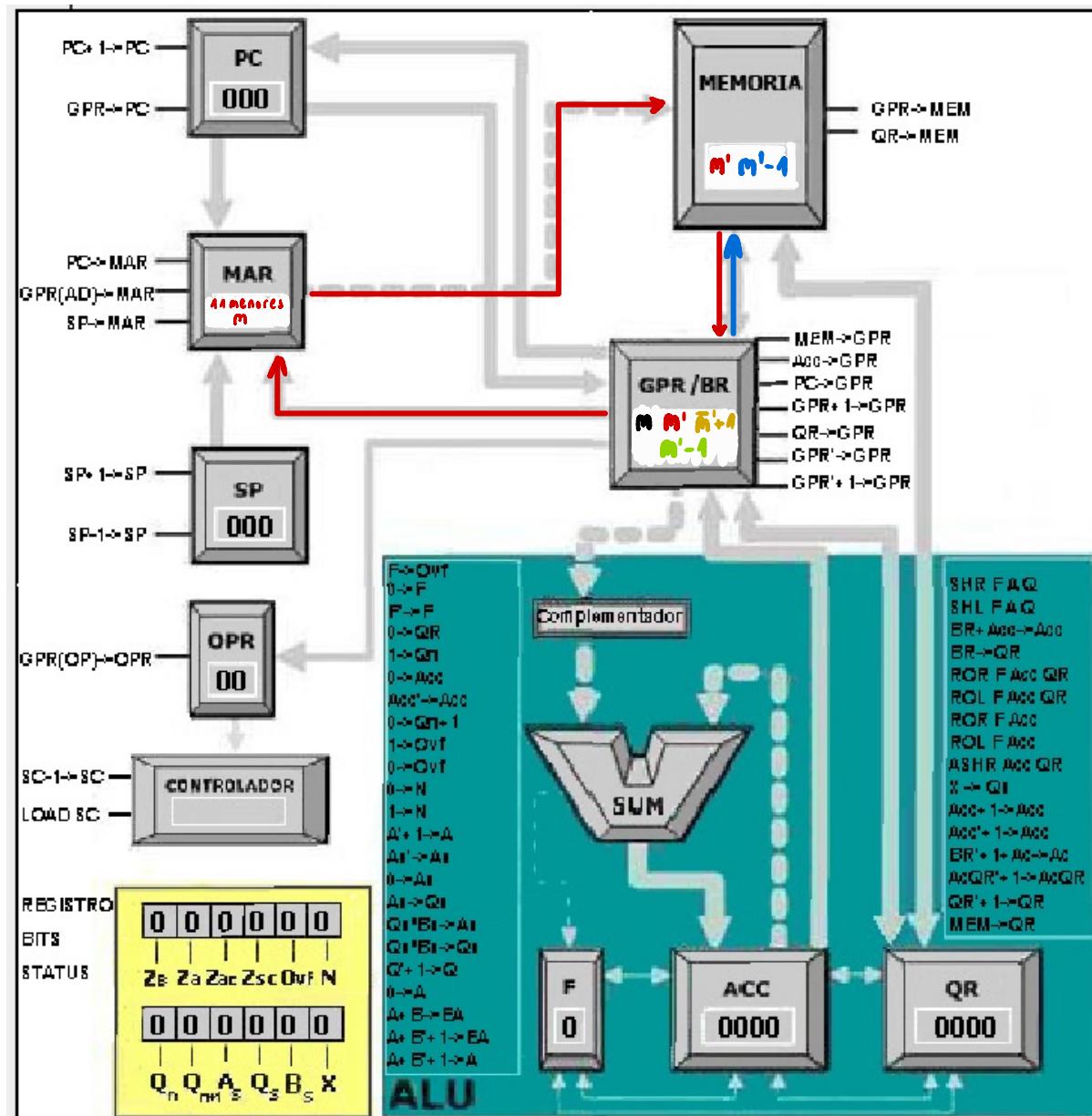
DEC m

GPR(AD) \rightarrow MAR || M \rightarrow GPR || Habilita || Incrementa ●

menosGPR + 1 \rightarrow GPR || Habilita || Incrementa ○

menosGPR \rightarrow GPR || Habilita || Incrementa ○

GPR \rightarrow M || Habilita || Bifurca ○



Ejercicio 3

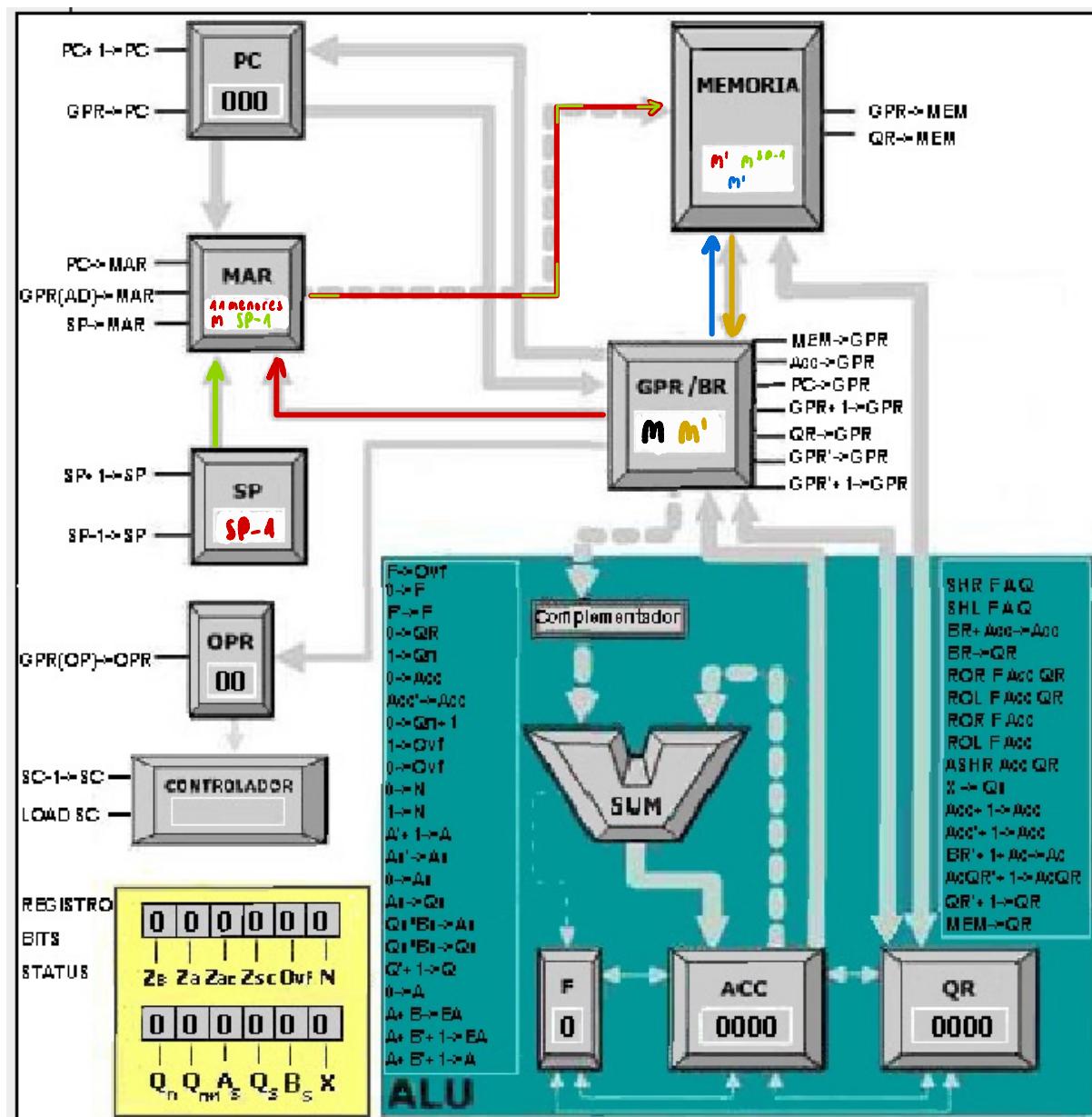
PUSH m

SP - 1 → SP || GPR(AD) → MAR || Habilita || Incrementa ●

M → GPR || Habilita || Incrementa ●

SP → MAR || Habilita || Incrementa ●

GPR → M || Habilita || Bifurca ●



PUSHI m

SP - 1 → SP || GPR(AD) → MAR || Habilita || Incrementa ●

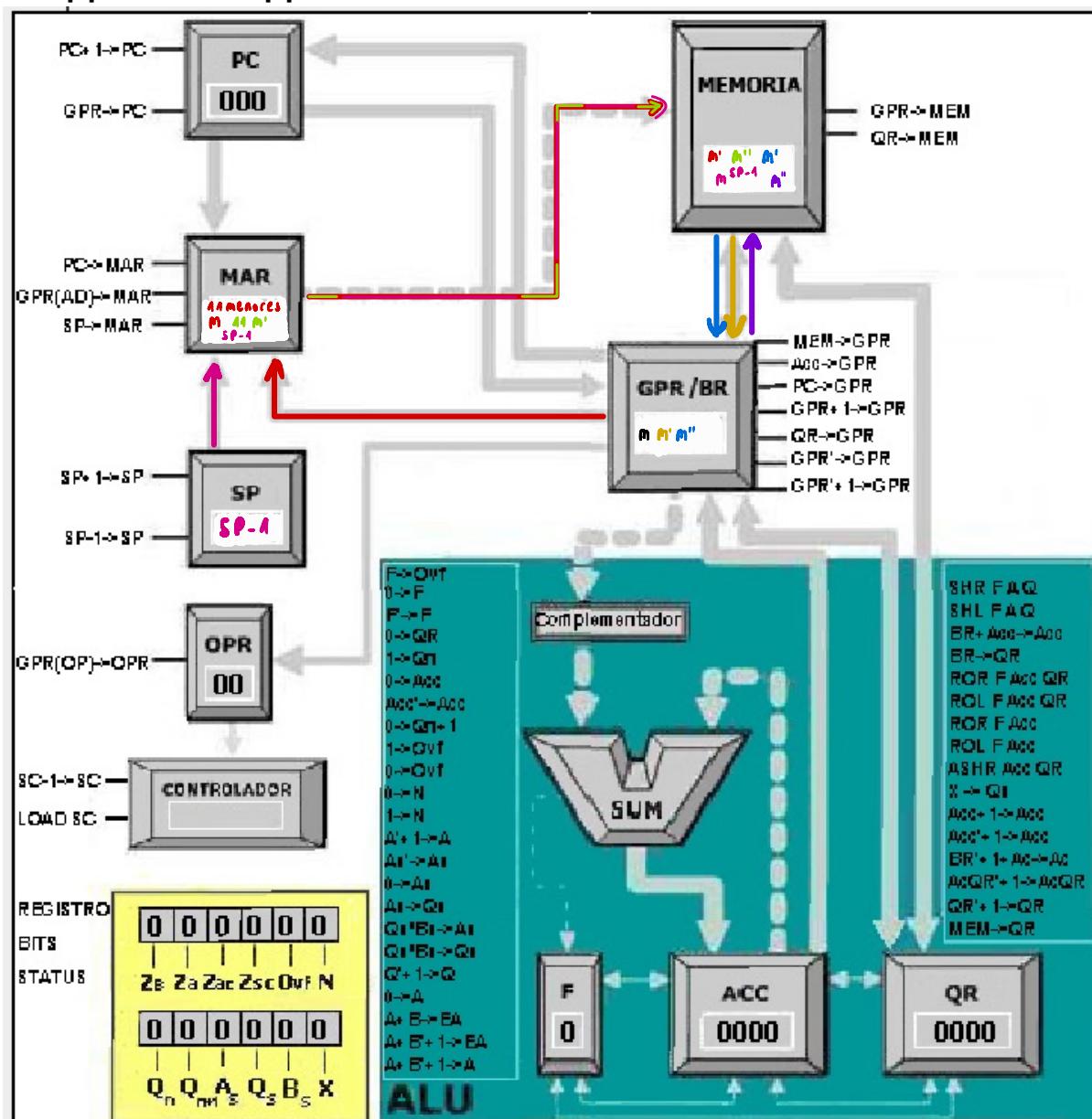
M → GPR || Habilita || Incrementa ●

GPR(AD) → MAR || Habilita || Incrementa ●

M → GPR || Habilita || Incrementa ●

SP → MAR || Habilita || Incrementa ●

GPR → M || Habilita || Bifurca ●



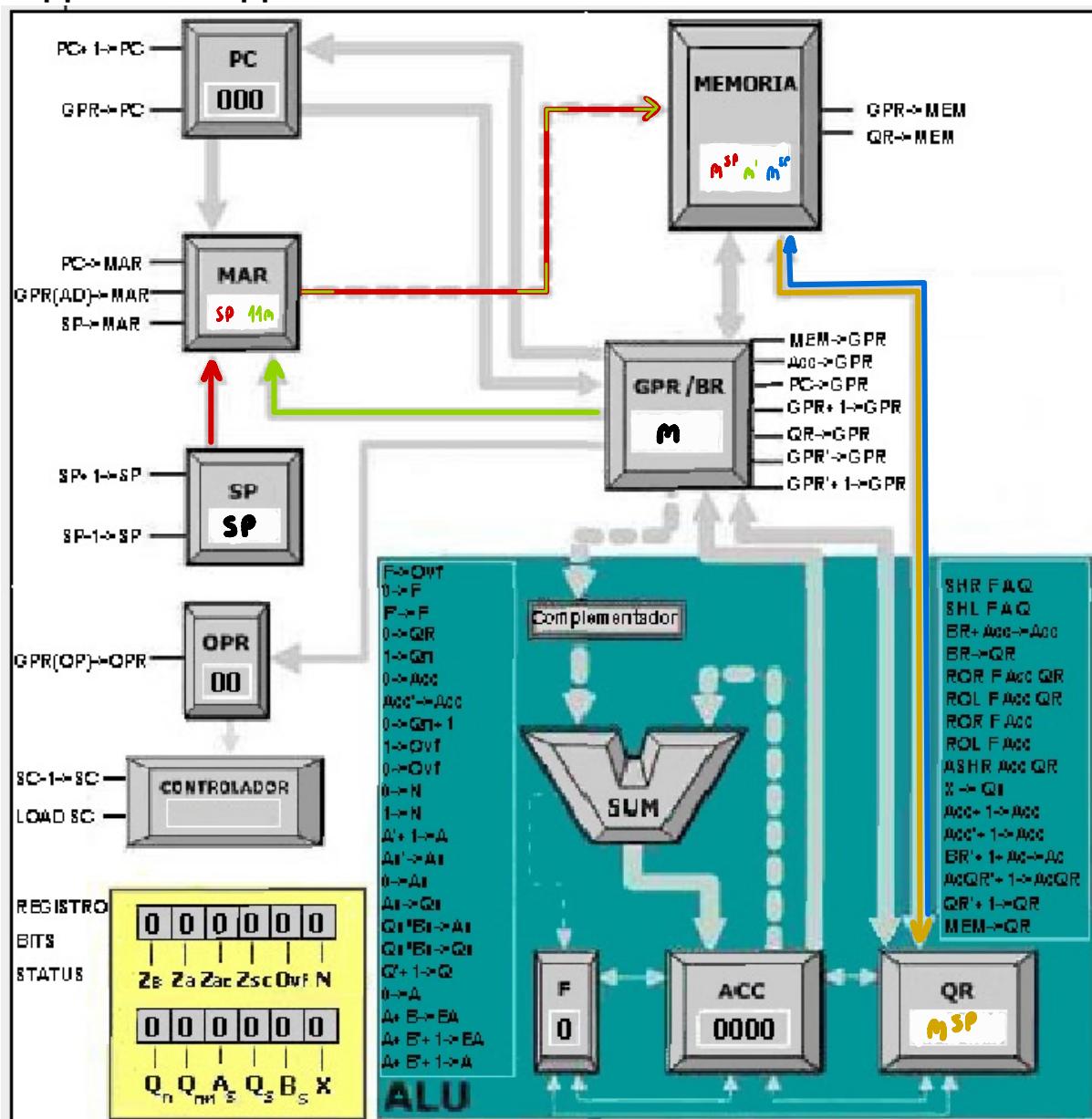
POP m

SP → MAR || Habilita || Incrementa ●

M → QR || Habilita || Incrementa ●

GPR(AD) → MAR || Habilita || Incrementa ●

QR → M || Habilita || Bifurca ●



POPI m

SP → MAR || Habilita || Incrementa ●

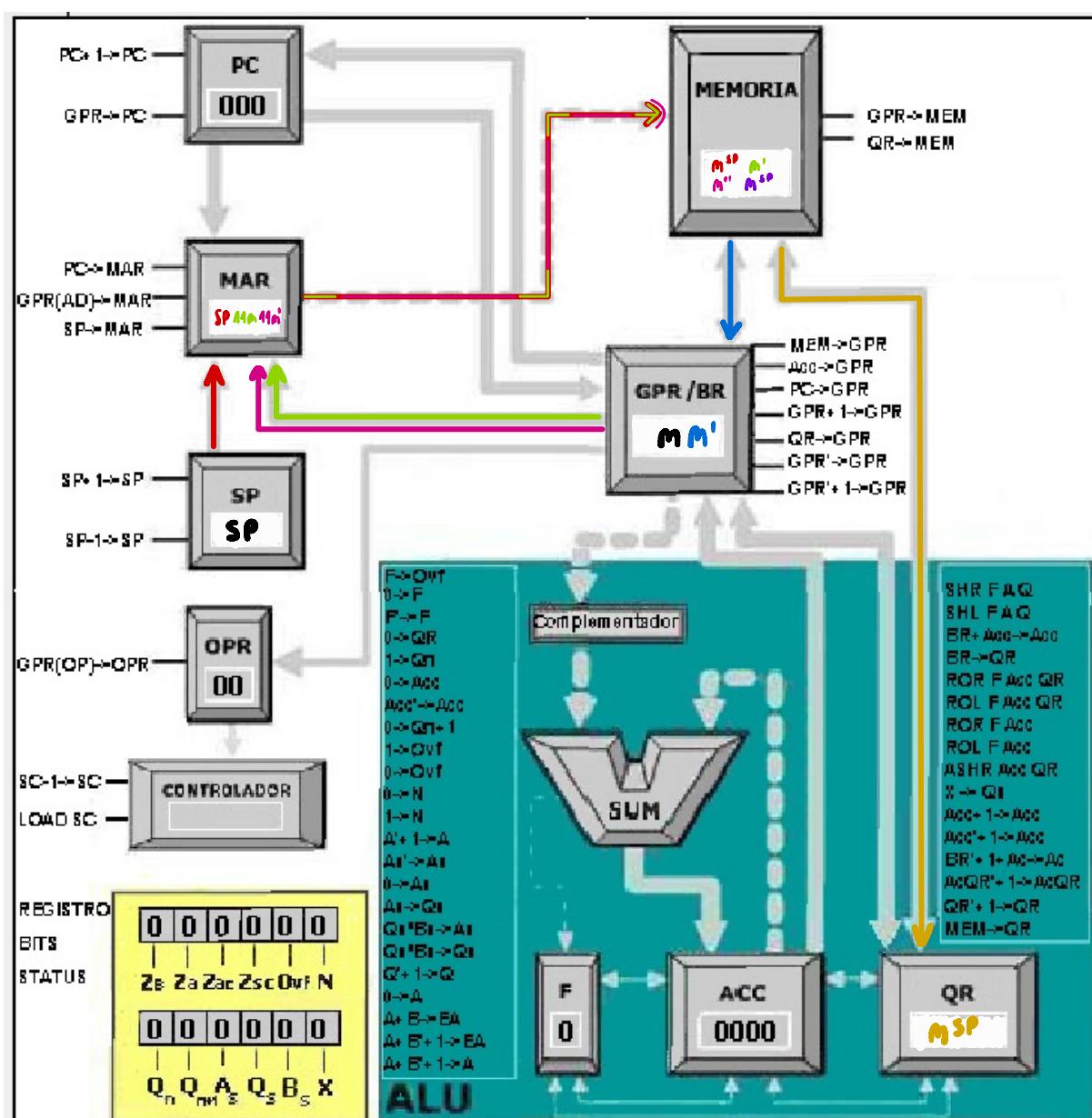
M → QR || Habilita || Incrementa ●

GPR(AD) → MAR || Habilita || Incrementa ●

M → GPR || Habilita || Incrementa ●

GPR(AD) → MAR || Habilita || Incrementa ●

QR → M || Habilita || Bifurca ●



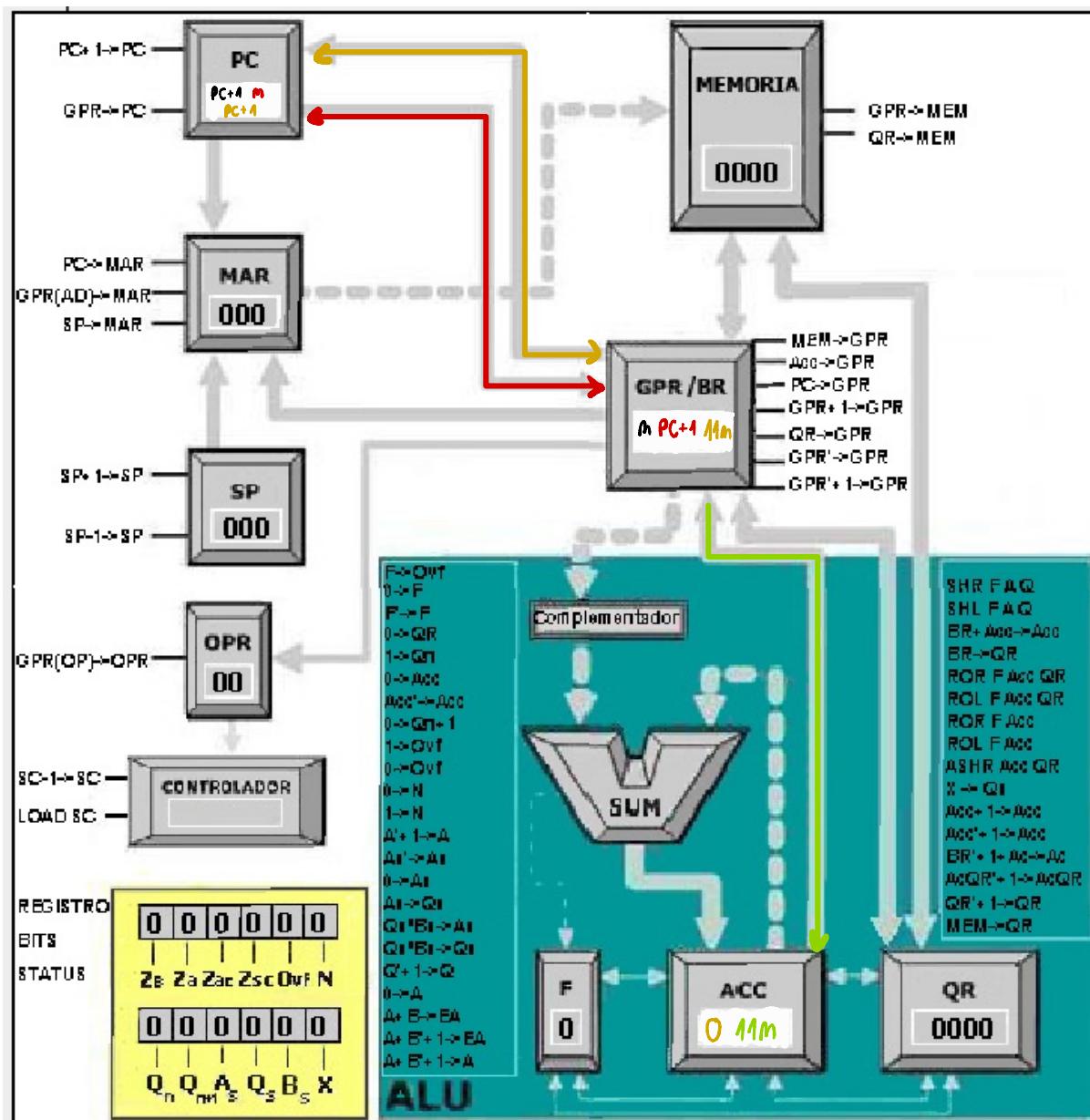
Ejercicio 4

ADDV11 val

GPR -> PC || PC -> GPR || Habilita || Incrementa ●

GPR -> PC || PC -> GPR || Habilita || Incrementa ○

GPR + ACC -> ACC || Habilita || Bifurca ●

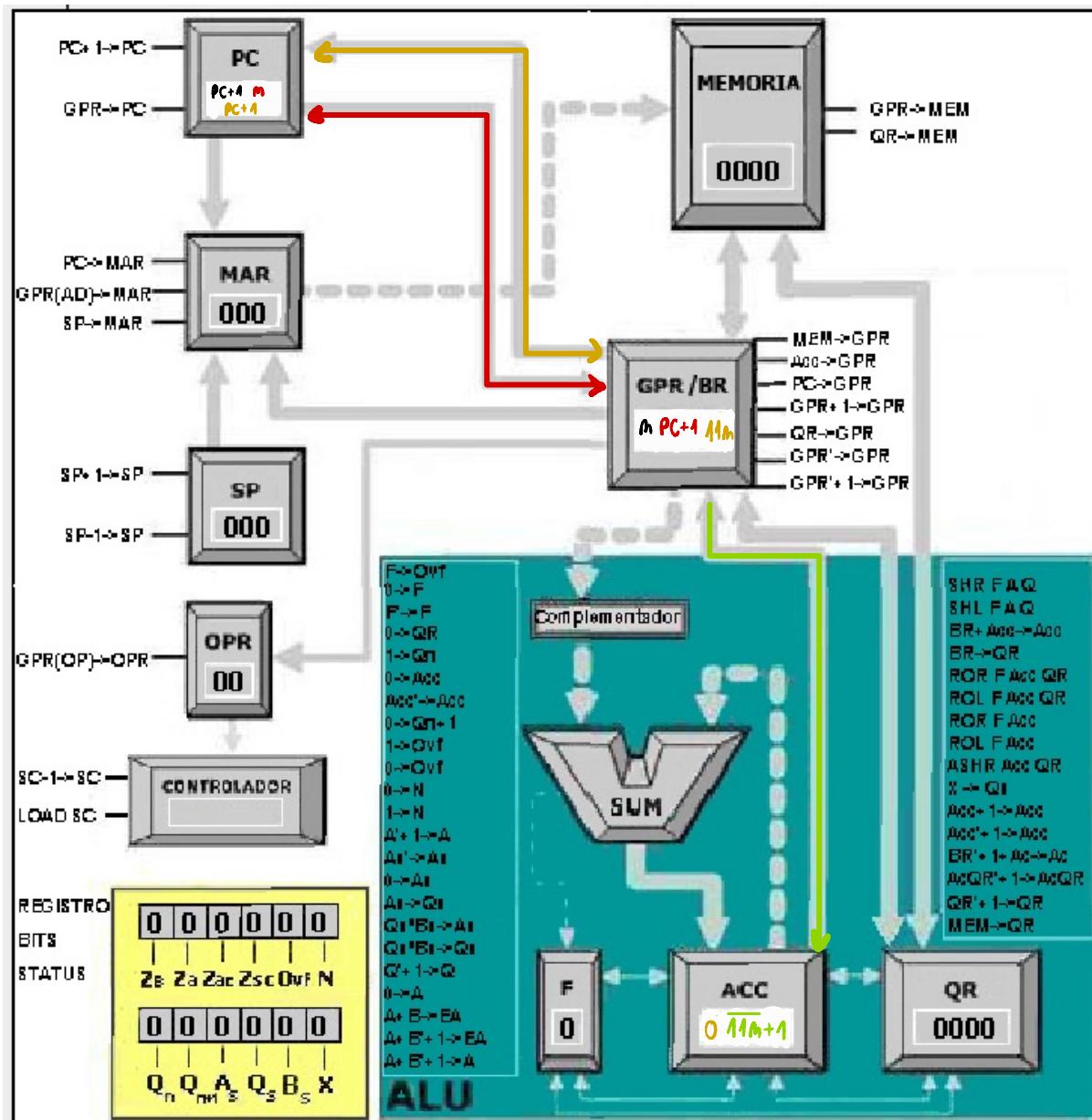


SUBV11 val

GPR -> PC || PC -> GPR || Habilita || Incrementa ●

GPR -> PC || PC -> GPR || Habilita || Incrementa ●

menosGPR + ACC -> ACC || Habilita || Bifurca ●



Ejercicio 5

JEG11 v11

GPR -> QR || SP -1 -> SP || Habilita || Incrementa ●

ACC -> GPR || 0 -> F || Habilita || Incrementa ●

SP -> MAR || 0 -> ACC || Habilita || Incrementa ●

GPR -> M || Habilita || Incrementa ●

QR -> GPR || Habilita || Incrementa ●

GPR + ACC -> ACC || LOAD SC -> 5 || Habilita || Incrementa ●

ROL F ACC || SC - 1 -> SC || Habilita || Incrementa ●

Habilita || Zsc = 0 Bifurca (58), Zsc = 1 Incrementa

LOAD SC -> 5 || Habilita || Incrementa ●

ASHR ACC QR || SC - 1 -> SC || Habilita || Incrementa ●

M -> GPR || Zsc = 1 Habilita || Zsc = 0 Bifurca (5B), Zsc = 1 Incrementa ●

menosGPR + 1 + ACC -> ACC || Habilita || Incrementa ●

PC + 1 -> PC || As = 0 Habilita || Incrementa ●

0 -> ACC || Habilita || Incrementa ●

GPR + ACC -> ACC || Habilita || Bifurca ●

