

Pauta Auxiliar 6

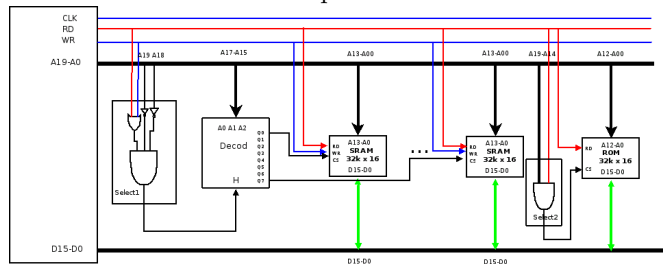
Rodrigo Cánovas

1. Problema 1

- a) Se agrega la siguiente ($1M=2^{20}$):

A_{19}	A_{18}	A_{17}	A_{16}	A_{15}	A_{14}	$A_{13} - A_{00}$	
1	1	1	1	1	*	*	$[1M - 16kb, 1M[$

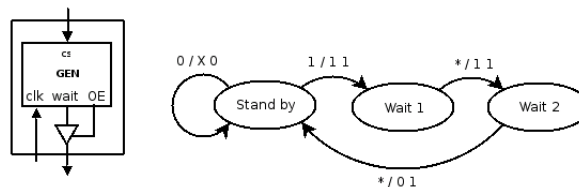
Lo nuevo será un selector que dirá cuando deberemos activar la ROM



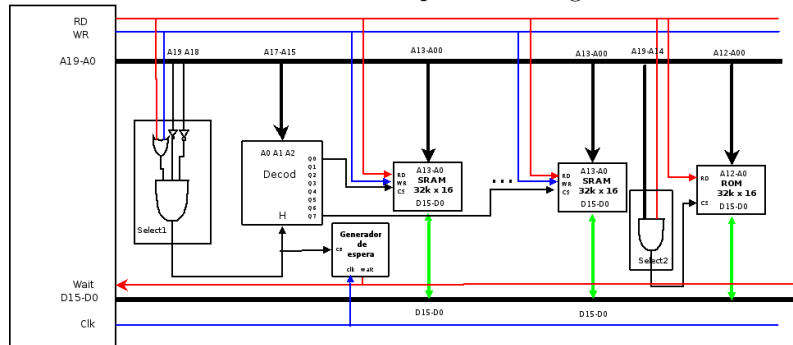
- b) Debemos agregar un generador de espera:

Generador de espera

Gen: cs/wait oe(on execution)



Entonces finalmente uniendo todo quedaria lo siguiente:



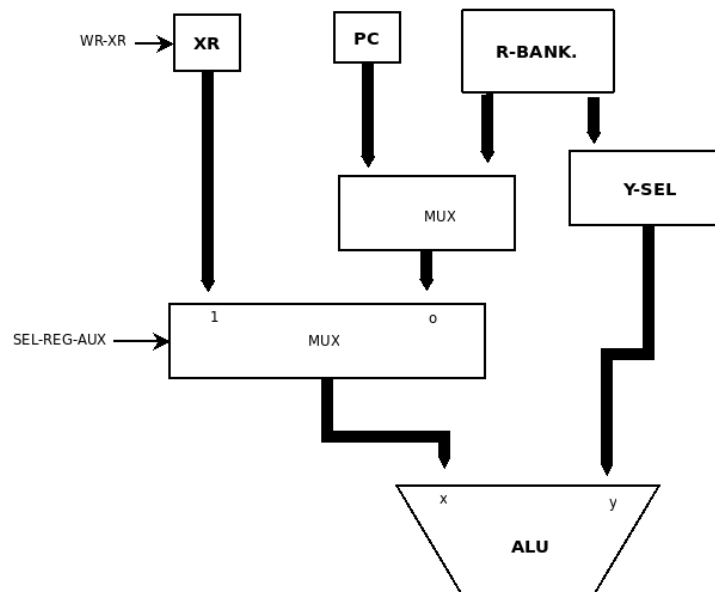
2. Problema 2

a) La instrucción SWAP no es soportado por M32 por dos razones:

- Esta instrucción toma más de dos ciclos del reloj. (los de M32 no toman más 2)
- No existe una unidad capaz de almacenar un registro auxiliar sin pisar información relevante del programa.

b) Crearemos un componente XR que sera un registro auxiliar que funciona igual que PC o AR (lo haremos funcionar como un LATCH) y agregaremos un multiplexor a M32. Además agregaremos las siguientes señales de control:

- wr-xr : controlara si se escribe en XR o no.
- sel-regaux: servira para que el multiplexor sepa cuando usar la información de XR.



c) Consiste en 4 ciclos:

Execute 1	
$AR \leftarrow reg1 + \{imm reg2\}$ A31-A2 \leftarrow AR goto Execute 2	SEL-REG, OR-Y-SEL \leftarrow @Inst, OP-ALU \leftarrow @ADD, WR-SR, WR-AR, OP-ABI \leftarrow @w, EN-A
Execute 2	
XR $\leftarrow M^w[AR]$ goto Execute 3	RD, WR-XR, SEL-REG-AUX, SEL-D, OP-DBI \leftarrow @LDW, EN-A, OP-ABI \leftarrow @W
Execute 3	
$M^w[AR] \leftarrow regd$ goto Execute 4	RD-Dest, OP-Y-SEL \leftarrow @0, OP-ALU \leftarrow @OR, EN-A, OP-ABI \leftarrow @w, OP-DBI \leftarrow @stw, EN-D, WR
Execute 4	
regd \leftarrow XR	OP-Y-SEL \leftarrow @0, OP-ALU \leftarrow @OP, WR-RD, SEL-REG-AUX