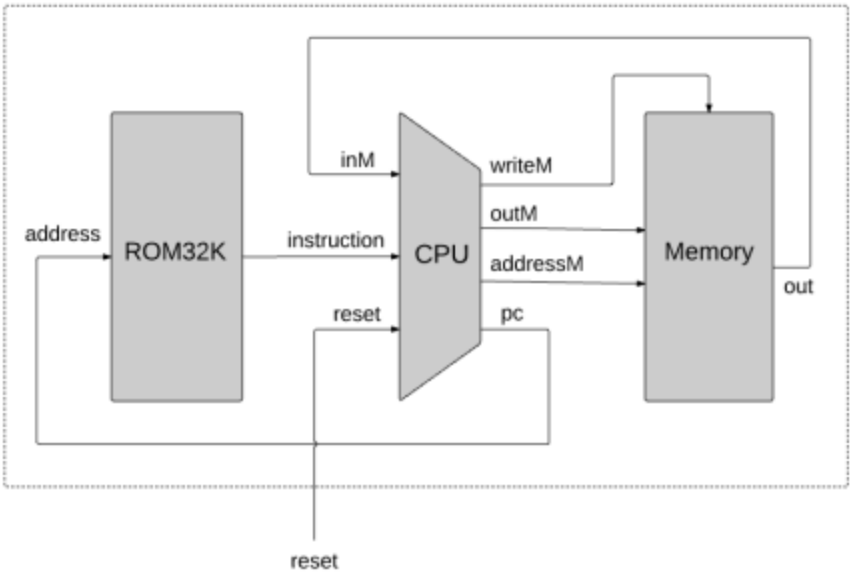

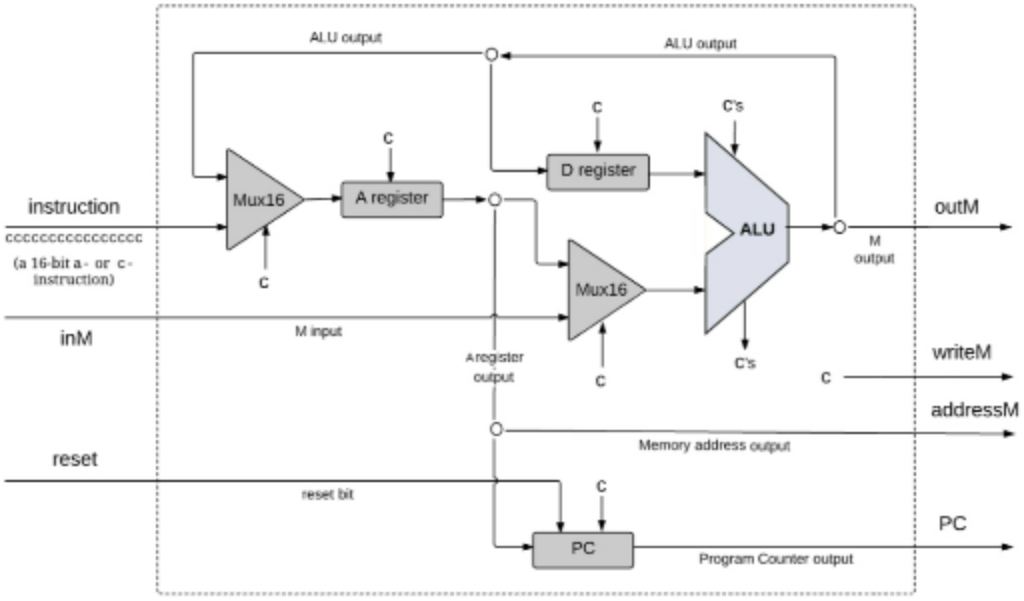

16-bit Instruction Format: **ixxac$_1$c$_2$c$_3$c$_4$c$_5$c$_6$d$_1$d$_2$d$_3$j$_1$j$_2$j$_3$**

i determines an A-Instruction or a C-Instruction
i = 0 is an A-instruction (Sets the address. The remaining fifteen bits will instead be used to select the address)
i = 1 is a C-Instruction

x is blank and doesn't matter if the instruction is a C-Instruction, these are used as address bits for A-Instructions

a determines whether the Y-Input to the ALU is the value of the current address (**a**=0) or the value stored in memory at the current address (**a**=1)

The values at **c** determine the options for the ALU

$\mathbf{c_1}$ is zx
$\mathbf{c_2}$ is nx
$\mathbf{c_3}$ is zy
$\mathbf{c_4}$ is ny
$\mathbf{c_5}$ is f
$\mathbf{c_6}$ is no

The values for **d** determine where the result will be saved (A, M, or D)

$\mathbf{d_1}$ is A
$\mathbf{d_2}$ is D
$\mathbf{d_3}$ is M

The values for **j** determine the conditions for a jump to execute

$\mathbf{j_1}$ is when less than zero
$\mathbf{j_2}$ is when equal to zero
$\mathbf{j_3}$ is when greater than zero

So the values for the **j** bits for the command "JGE" (*comp* is ≥ 0) would be 011

(when a=0) ***comp*** *mnemonic*	c1	c2	c3	c4	c5	c6	(when a=1) ***comp*** *mnemonic*
0	1	0	1	0	1	0	
1	1	1	1	1	1	1	
-1	1	1	1	0	1	0	
D	0	0	1	1	0	0	
A	1	1	0	0	0	0	M
!D	0	0	1	1	0	1	
!A	1	1	0	0	0	1	!M
-D	0	0	1	1	1	1	
-A	1	1	0	0	1	1	-M
D+1	0	1	1	1	1	1	
A+1	1	1	0	1	1	1	M+1
D-1	0	0	1	1	1	0	
A-1	1	1	0	0	1	0	M-1
D+A	0	0	0	0	1	0	D+M
D-A	0	1	0	0	1	1	D-M
A-D	0	0	0	1	1	1	M-D
D&A	0	0	0	0	0	0	D&M
D|A	0	1	0	1	0	1	D|M