

Opcode	Instruction	Registers	Bits			
0000	Register to Register	$R_A \rightarrow R_Y$	Op (4)	R_A (4)	X (4)	R_Y (4)
0001	Immediate to Register	$I \rightarrow R_Y$	Op (4)	X (8)		R_Y (4)
			Immediate (16)			
0010	Register to Memory	$R_A \rightarrow M_{RB}$	Op (4)	R_A (4)	R_B (4)	X (4)
0011	Memory to Register	$M_{RA} \rightarrow R_Y$	Op (4)	R_A (4)	X (4)	R_Y (4)
0100	Bitwise OR	$R_A \parallel R_B \rightarrow R_Y$	Op (4)	R_A (4)	R_B (4)	R_Y (4)
0101	Bitwise NOT	$\sim R_A \rightarrow R_Y$	Op (4)	R_A (4)	X (4)	R_Y (4)
0110	Bitwise AND	$R_A \&\& R_B \rightarrow R_Y$	Op (4)	R_A (4)	R_B (4)	R_Y (4)
0111	Bitwise XOR	$R_A \wedge R_B \rightarrow R_Y$	Op (4)	R_A (4)	R_B (4)	R_Y (4)
1000	Add	$R_A + R_B \rightarrow R_Y$	Op (4)	R_A (4)	R_B (4)	R_Y (4)
1001	Subtract	$R_A - R_B \rightarrow R_Y$	Op (4)	R_A (4)	R_B (4)	R_Y (4)
1010	Logical NOT	$!R_A \rightarrow R_Y$	Op (4)	R_A (4)	X (4)	R_Y (4)
1011	Logical Shift Right	$R_A \gg 1 \rightarrow R_Y$	Op (4)	R_A (4)	X (4)	R_Y (4)
1100	Push* (Details below)	$(R_A \text{ or } PC) \rightarrow$ Top stack	Op (4)	R_A (4)	Op ₂ (1)	X (7)
1101	Pop** (Details below)	Top stack $\rightarrow (R_A$ or PC)	Op (4)	R_A (4)	Op ₂ (1)	X (3) R_Y (4)
1110	Unconditional Jump	$R_A \rightarrow PC$	Op (4)	R_A (4)	X (8)	
1111	Jump Compare*** (Details below)	If $(R_B \text{ ?? } 0)$: $R_A \rightarrow PC$	Op (4)	R_A (4)	R_B (4)	Flags (4)

*Push:

Op ₂	Behavior
0	R _A → Top stack
1	PC → Top stack

**Pop:

Op ₂	Behavior
0	Top stack → R _A
1	Top stack → PC

***Jump compare:

Flags	Comparison
0000	N/A (Instruction functions as a NOP)
0001	R _B == 0
001X	R _B != 0
01XX	R _B < 0
1XXX	R _B > 0