

ROLL: 1801CS08

The screenshot displays a logic simulator interface with a complex digital circuit. The circuit includes several multiplexers (MUX), an ALU (Arithmetic Logic Unit), and various registers. Key components and their states are as follows:

- Registers:**
 - A register at the top left contains the value `00000000`.
 - A register in the middle left contains the value `00000000`.
 - A register in the middle right contains the value `00000000`.
 - A register at the bottom right contains the value `00000000`.
- ALU:** The ALU is located in the center of the circuit, with inputs from registers and other components. It is currently performing an addition operation, resulting in the value `00000000`.
- MUX (Multiplexers):** There are several MUX components throughout the circuit, selecting between different data paths. One MUX is labeled "MUX" and another is labeled "MUX".
- Logic Gates:** The circuit includes various logic gates, including AND, OR, and NOT gates, which are used to perform logical operations on the data.
- Control Signals:** The circuit is controlled by several signals, including "Clear PC" (set to 0), "sign", and "ZP-11".

The circuit is interconnected with numerous logic gates and data paths, illustrating a detailed hardware implementation of a digital system.

Logisim: Hex Editor

File	Edit	Project	Simulate	Window	Help
000000	00000000	2003000c	20677ff7	00e22025	00642824 00a42620 10a7000a 0064202a
000010	20020001	ac020054	00000000	00000000	00000000 00000000 00000000 00000000
000020	00000000	00000000	00000000	00000000	00000000 00000000 00000000 00000000
000030	00000000	00000000	00000000	00000000	00000000 00000000 00000000 00000000
000040	00000000	00000000	00000000	00000000	00000000 00000000 00000000 00000000
000050	00000000	00000000	00000000	00000000	00000000 00000000 00000000 00000000
000060	00000000	00000000	00000000	00000000	00000000 00000000 00000000 00000000

The screenshot shows the Logisim Hex Editor interface. The menu bar includes File, Edit, Project, Simulate, Window, and Help. The main area displays a memory dump starting at address 000000. Address 000001 is selected, indicated by a blue highlight. The hex values for each byte are as follows:

Address	Hex Value
000000	000000
000001	000000
000002	000000
000003	000000
000004	000000
000005	000000
000006	000000
000007	00000007
000008	000000
000009	000000
00000A	000000
00000B	000000
00000C	000000
00000D	000000
00000E	000000
00000F	000000

2. Program to add 5 numbers.

Initially 5 numbers are stored in registers no 1 to 5.

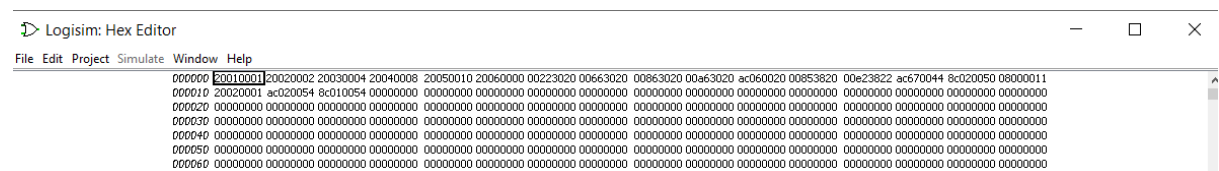
\$1 = 34, \$2 = 21, \$3 = 15, \$4 = 45, \$5 = 12, Result of addition \$6 = 0 initially

All numbers are in hexadecimal form. Final Result \$6 = C1(Hex)

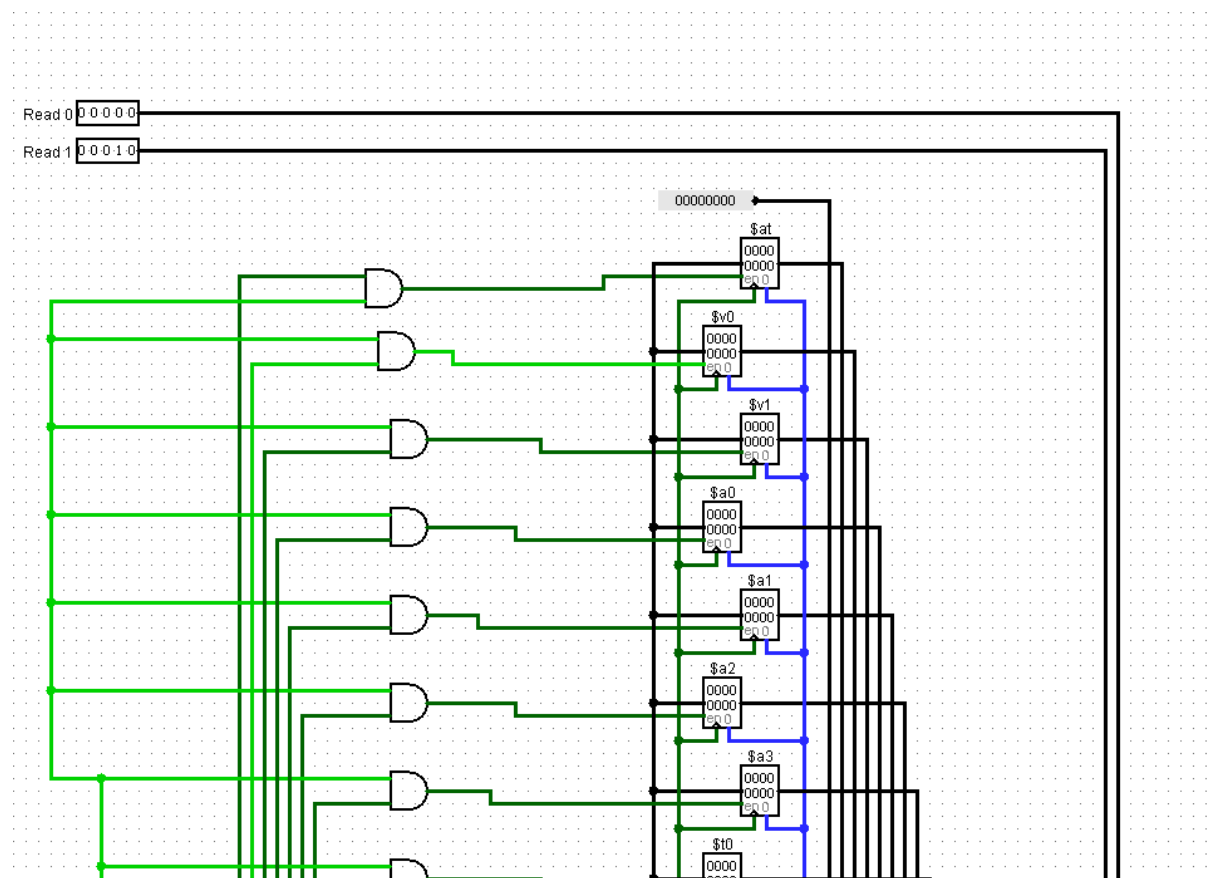
52+33+21+69+18=193=C1(Hex)

MIPS code	Machine Code
addi \$1, \$0, 52	0x20010034
addi \$2, \$0, 33	0x20020021
addi \$3, \$0, 21	0x20030015
addi \$4, \$0, 69	0x20040045
addi \$5, \$0, 18	0x20050012
addi \$6, \$0, 0	0x20060000
add \$6, \$1, \$2	0x00223020
add \$6, \$3, \$6	0x00663020
add \$6, \$4, \$6	0x00863020
add \$6, \$5, \$6	0x00A63020
sw \$6, 32(\$0)	0xAC060020

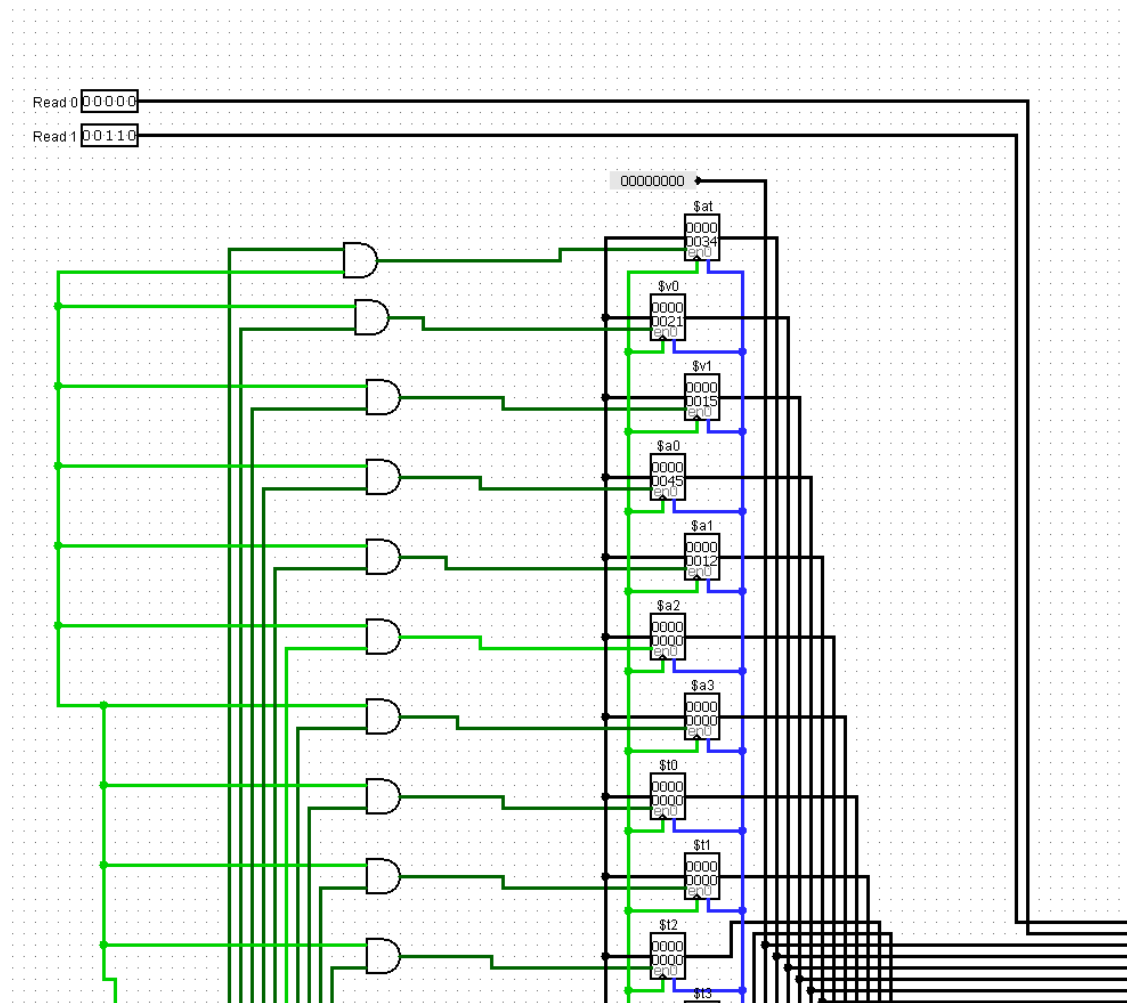
ROM loaded with Machine Code



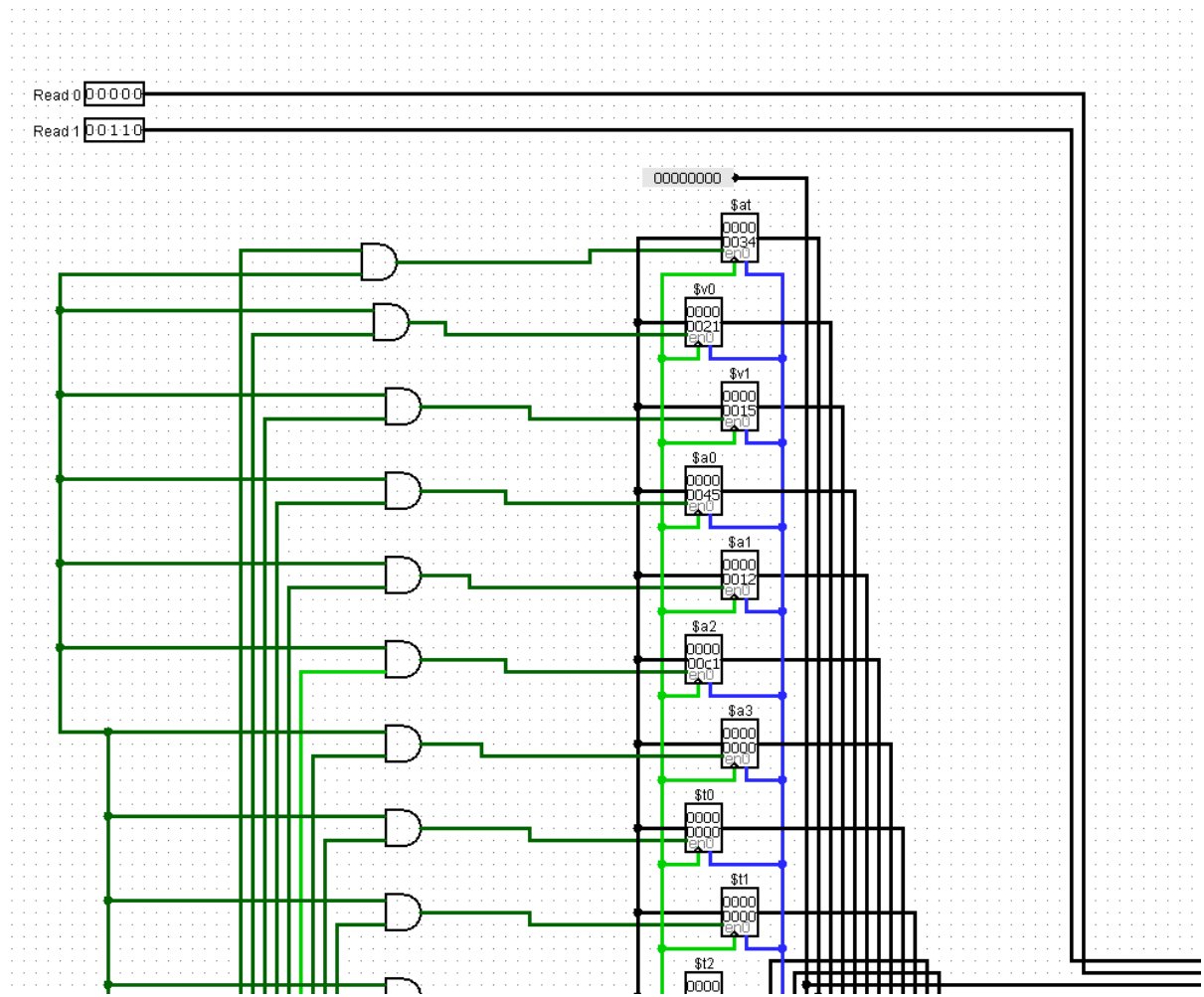
Before running instructions



Before Addition after storage in registers



After addition \$6=C1



Memory location 32 or 20(Hex) has stored sum = 193=C1 in Hex

