**Computer Architecture CS322 Lab 7 Report**

**Name : Chandrawanshi Mangesh Shivaji**

**Roll Number : 1801CS16**

**Date : 31/10/2020**

Task 1: Using blocks given blocks and other glue logic, implement single cycle RISC processor Instructions (given 32-bit reg file and 32-bit, memory blocks, ALU).

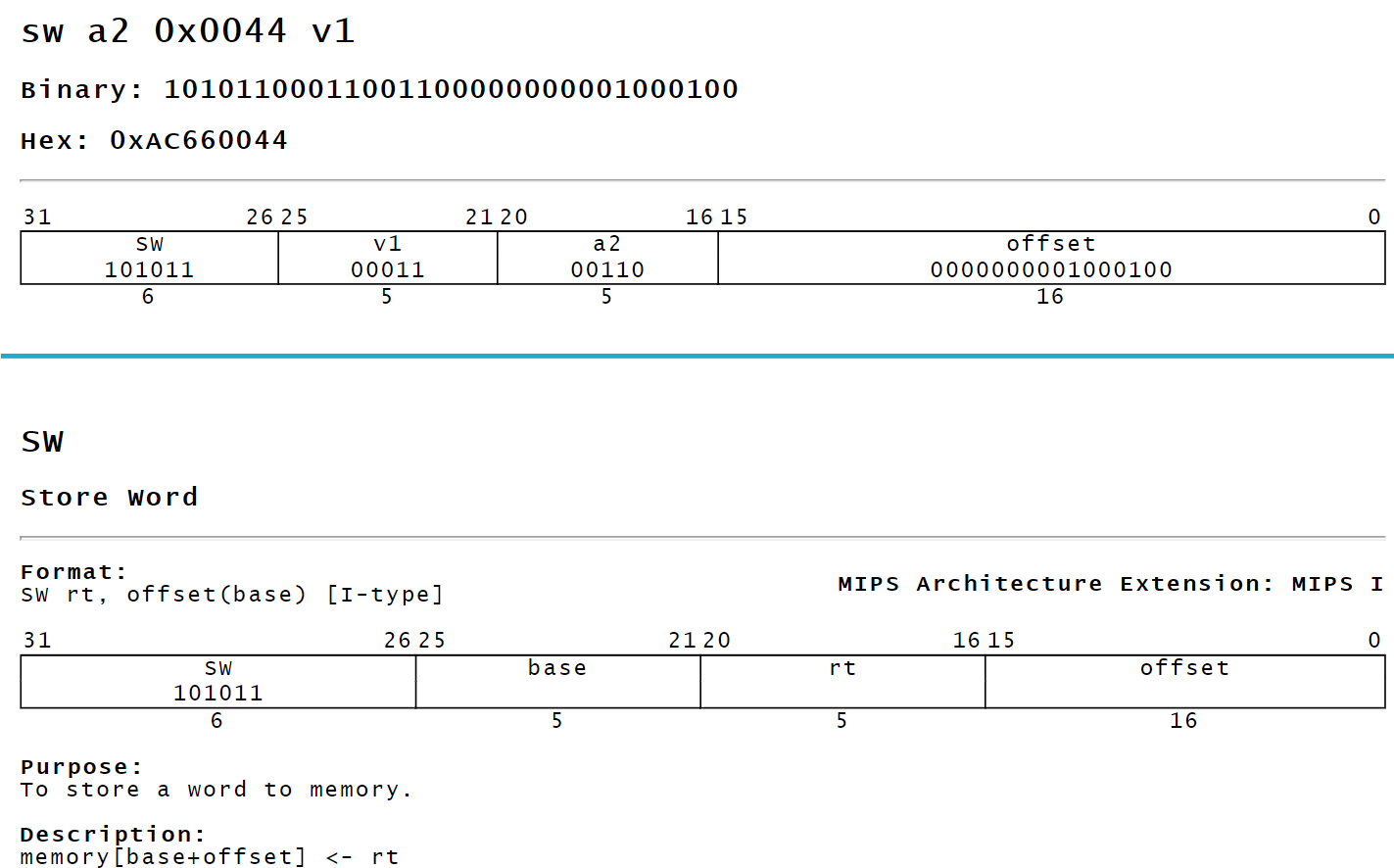
1. **sw $6, 68($3)**

**Encoding of the Instruction: -**

$6 => $a2

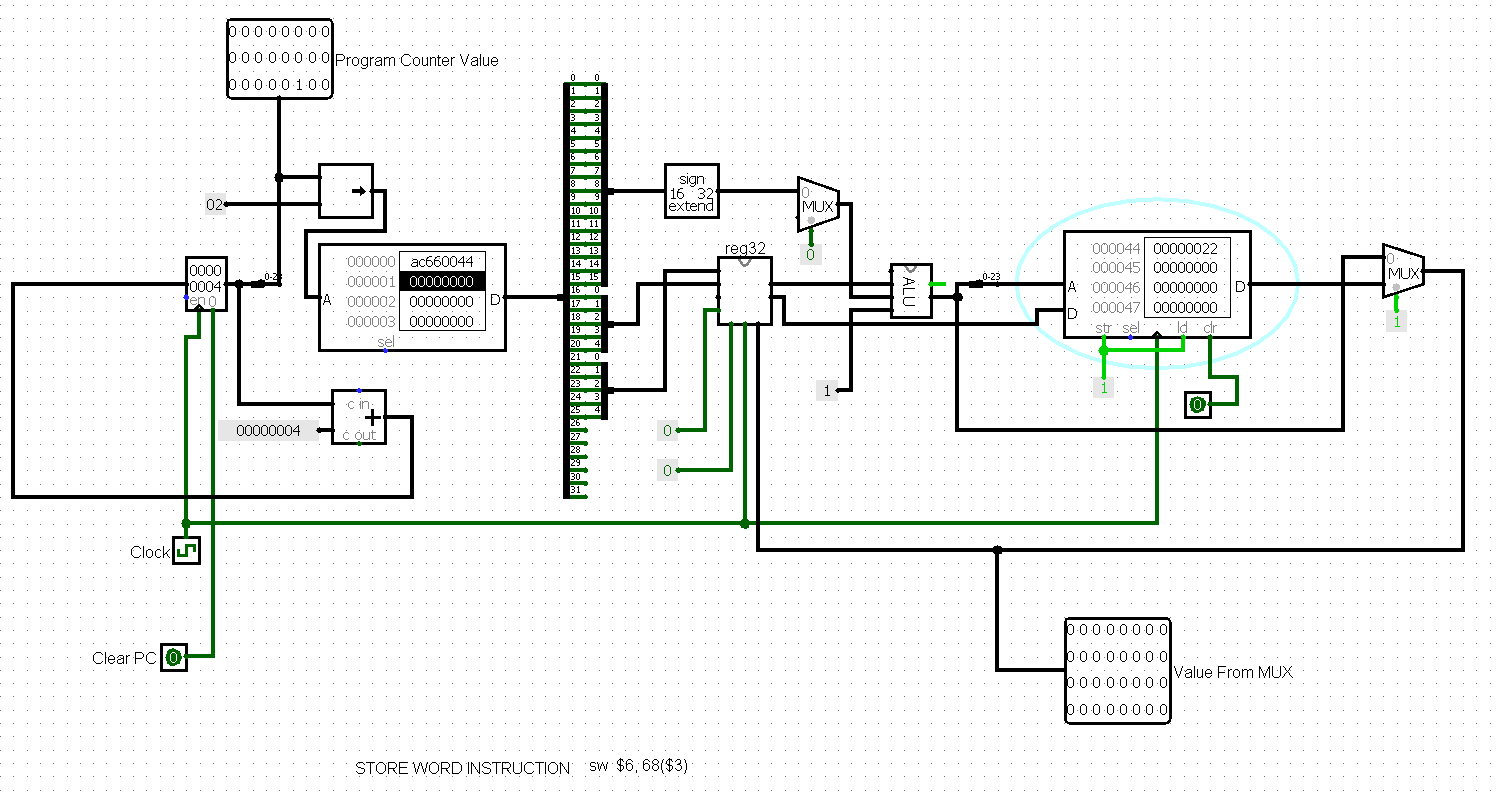
68 => 0x0044

$3 => $v1



**Used Hex Encoding: 0xAC660044**

**Logisim Implementation**:



Stores value at given memory location (here, M[$3+68]) from source register (here, $6).

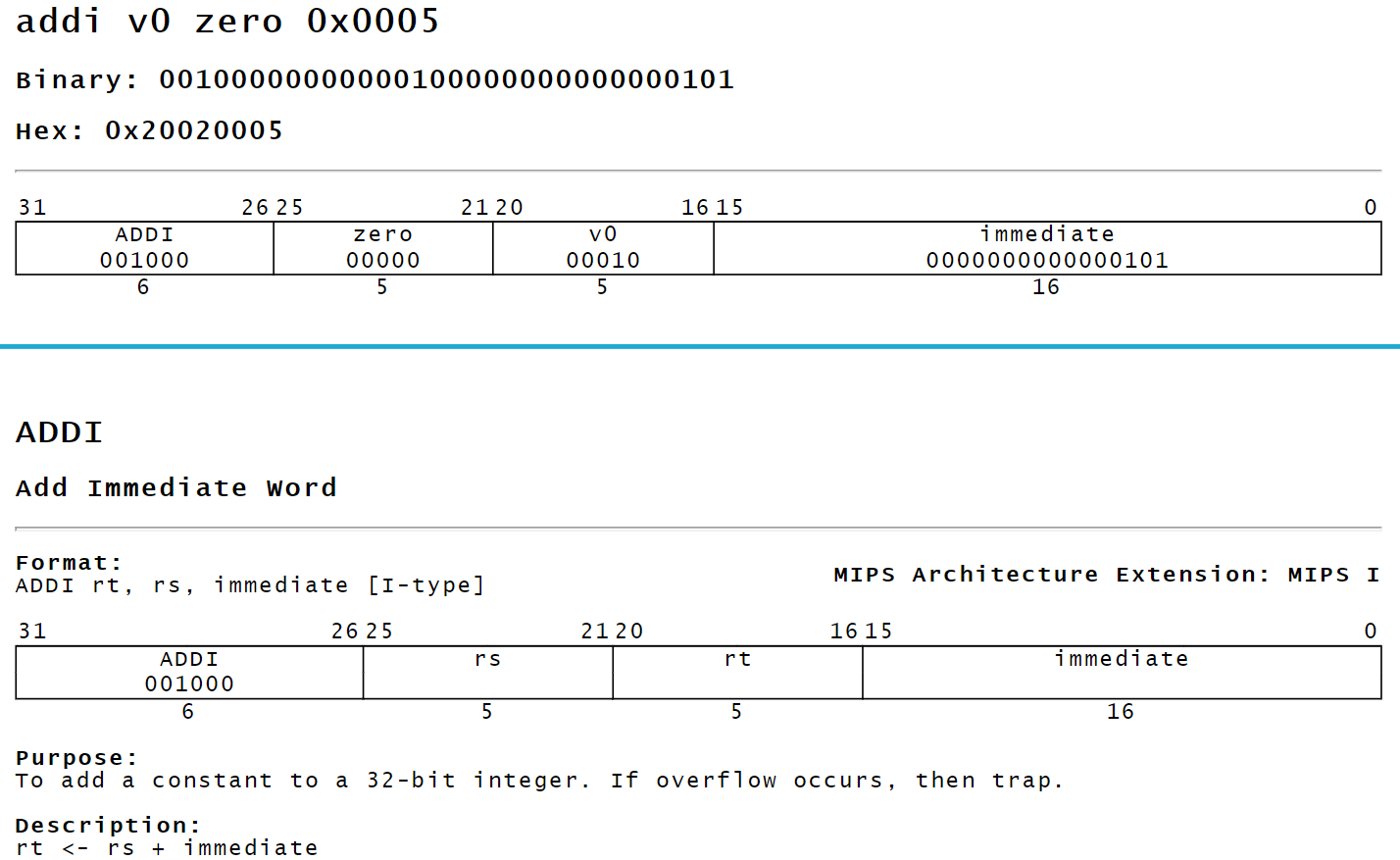
**(b) addi $2, $0, 5**

**Encoding of the Instruction: -**

$2 => $v0

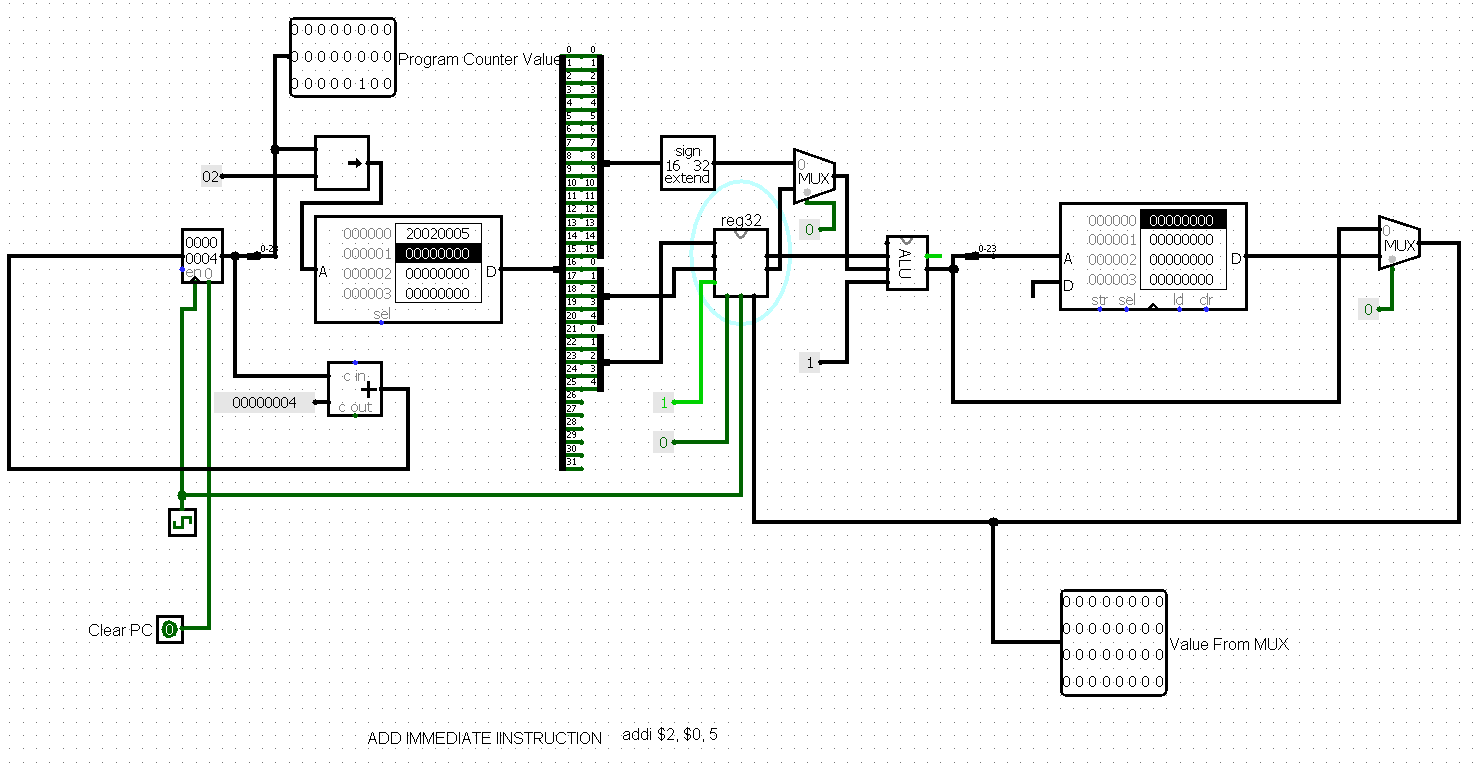
5 => 0x0005

$0 => $zero



**Used Hex Encoding: 0x20020005**

**Logisim Implementation:**



Adds immediate value (here,5) to source register (here, $0) and then stores the sum back to target register (here, $2)

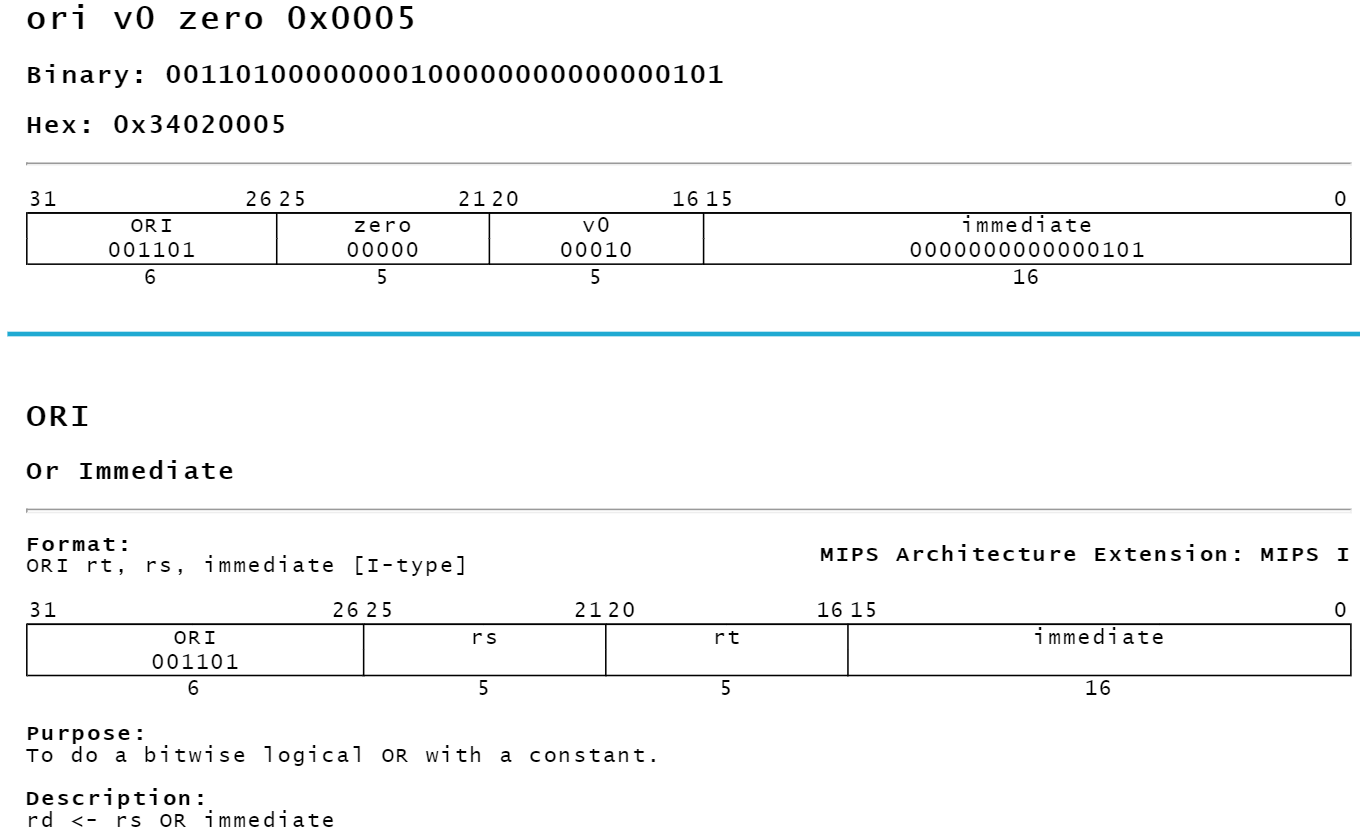
**(c) ori $2, $0, 5**

**Encoding of the Instruction: -**

$2 => $v0

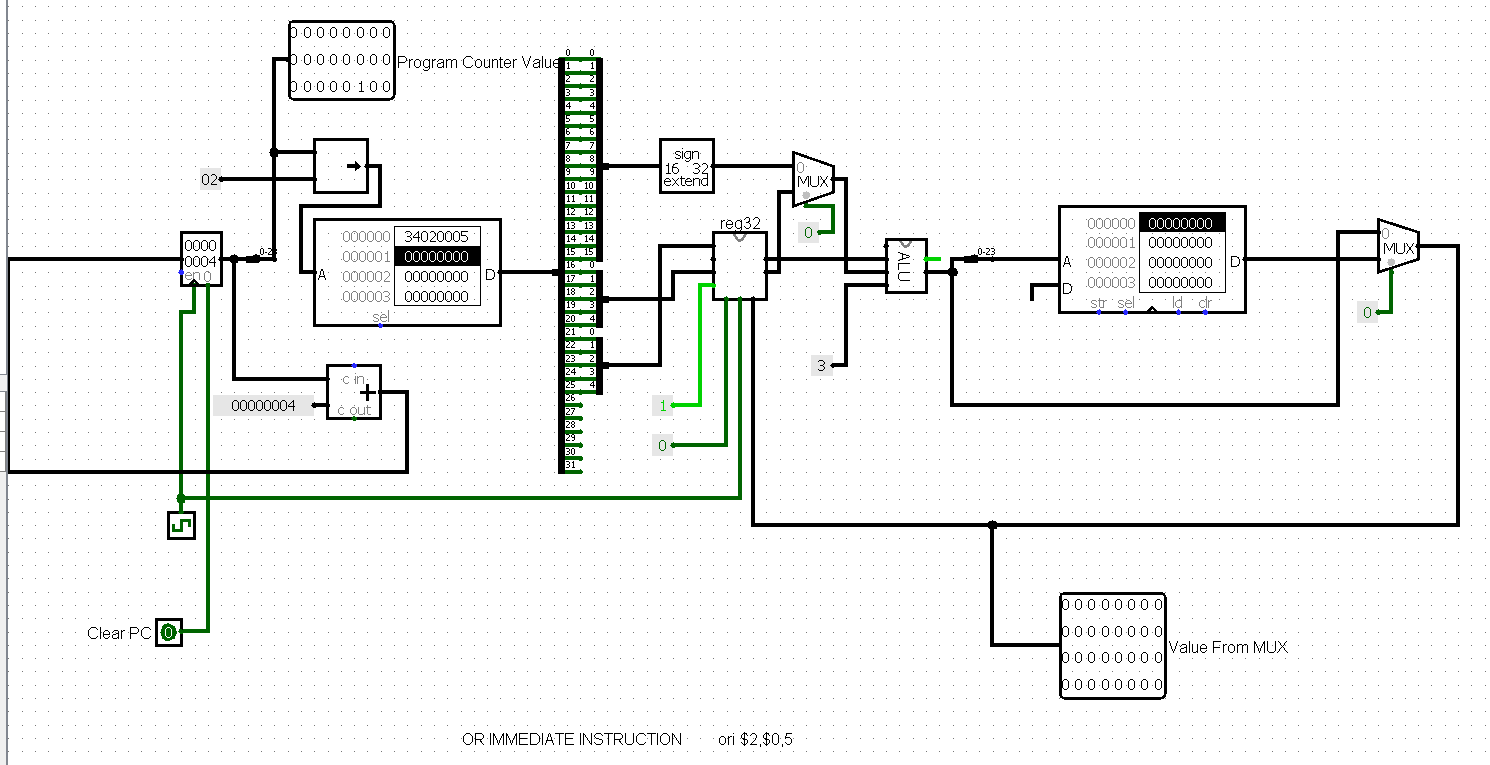
5 => 0x0005

$0 => $zero



**Used Hex Encoding: 0x34020005**

**Logisim Implementation:**



Performs or with immediate value (here,5) for source register (here, $0) and then stores the result back to target register (here, $2)