****

**North South University**

Department of Electrical and Computer Engineering

CSE 332, Section 11, FALL 2018

Faculty: SMH2

Topic: Designing 16-bit MIPS CPU

Group 8 Member’s

Lubba Saha => 1712358642

Mushfiqur Rahman Chowdhury => 1711552642

Introduction

We have simulated a 16-bit MIPS CPU in Logisim software which has a data width of 16 bit and also, instruction format bits of 16-bits.

I/O Table

R-Type Format:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| OpCode | rs | rt | rd | func |
| 3 bits | 3 bits | 3 bits | 3 bits | 4 bits |

I-Type Format:

|  |  |  |  |
| --- | --- | --- | --- |
| OpCode | rs | rt | immediate |
| 3 bits | 3 bits | 3 bits | 7 bits |

J-Type Format:

|  |  |
| --- | --- |
| OpCode | Address |
| 3 bits | 13 bits |

Instruction to ROM Data

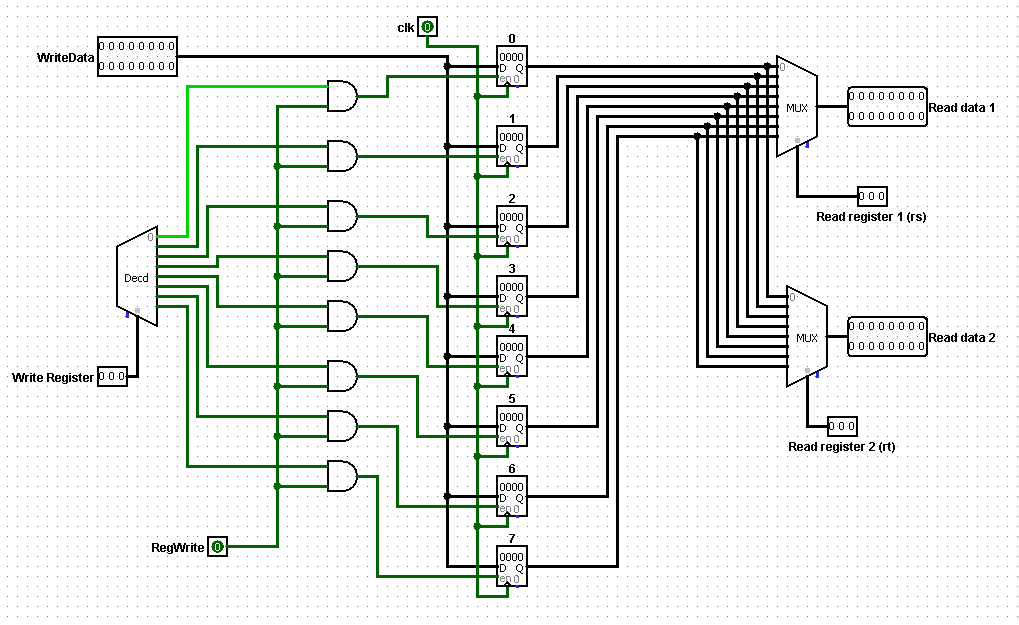
|  |  |
| --- | --- |
| Instruction | ROM Data |
| add | 0530 |
| sub | 0532 |
| lw | 2584 |
| sw | 4584 |
| beq | 6580 |
| jump | 8002 |

OPCode and func bits mapping

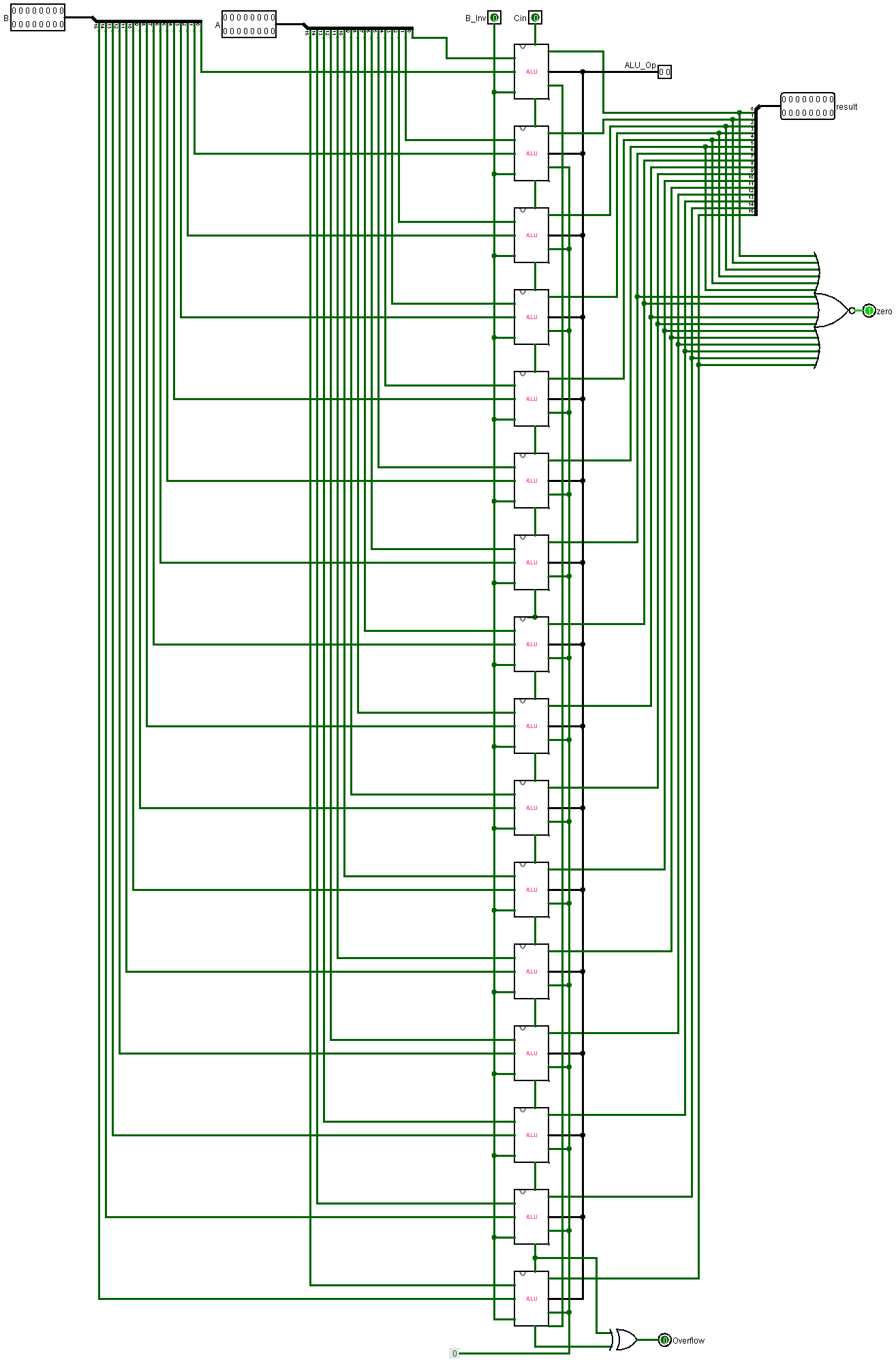
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| OPCode | Instruction-Type | ALU-Op | func | Instruction-Operation | Registered-ALU-Op | ALU-Control-Bits |
| 000 | R-Type | 10 | 0000 | ADD | ADD | 0010 |
| 000 | R-Type | 10 | 0010 | SUB | SUB | 0110 |
| 000 | R-Type | 10 | 0100 | AND | AND | 0000 |
| 000 | R-Type | 10 | 0101 | OR | OR | 0001 |
| 001 | I-Type | 00 | xxxx | Load Word | ADD | 0010 |
| 010 | I-Type | 00 | xxxx | Store Word | ADD | 0010 |
| 011 | I-Type | 01 | xxxx | Branch If Equal | SUB | 0110 |
| 100 | J-Type | xx | xxxx | Jump |  |  |

Components:

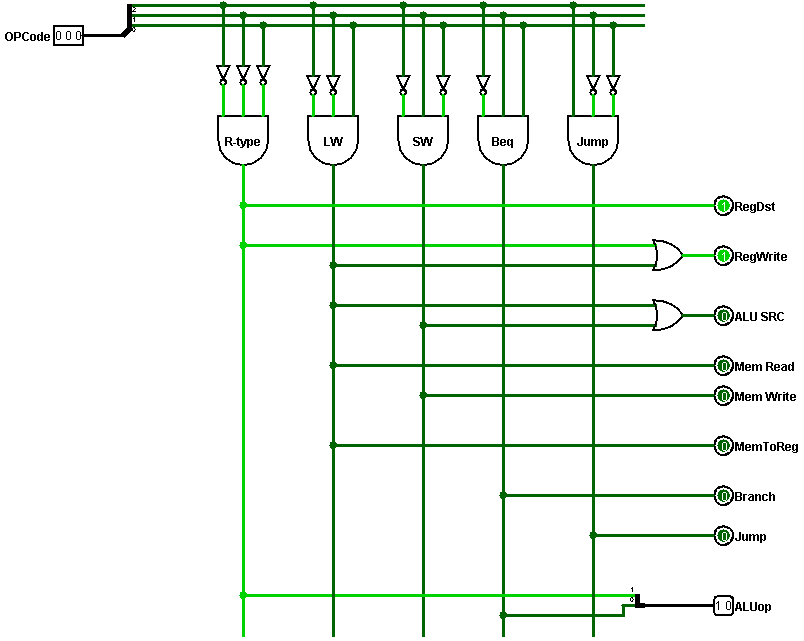
1. Register File containing 8 registers which are accessible using a 3-bit Write/Read bits. Each register is 16-bit width.



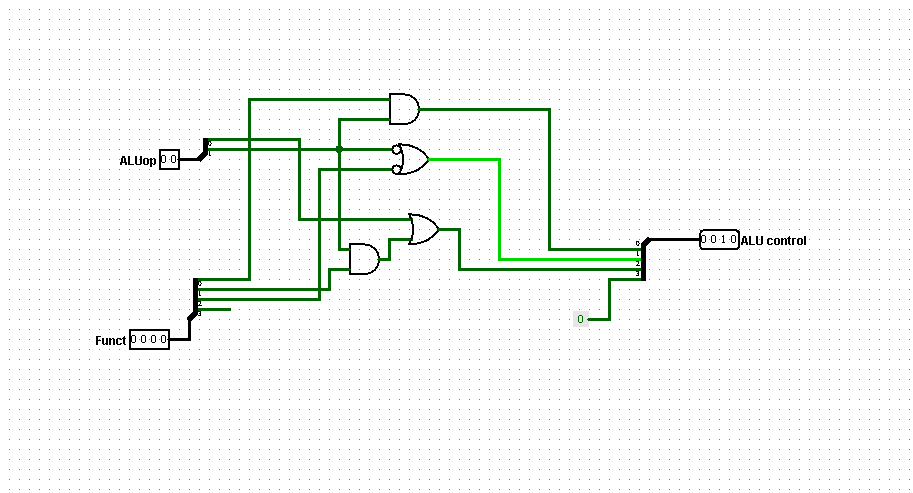
1. 16-bit ALU constructed from 16 1-bit ALU blocks which can detect zero result and overflow. It can perform AND, OR, ADD, SUB operations and has a 4x1 MUX selecting the ALU operations.



1. A control unit which controls parts of the CPU. The control unit takes the OPCode and decodes it to generate bits that enable parts of the CPU. It controls which component should be in use by which instruction.



1. An ALU control unit which takes ALU\_OP from the main control unit and takes the func bits of the instruction code to generate a 4-bit ALU control which specifies the ALU controls.



1. Final 16 bit MIPS CPU constructed from all the components mentioned above.

