**Project Documentation**

**Project name:** Designing a 12-bit Custom RISC-V Microprocessor.

**Course:** CSE332.

**Section:** 06.

**Semester:** Summer 2022.

**Group number:** 03.

**Name and id :**

Md. Faigul Azim 1921734042

Md. Jahid Hossain 1921214642

Zarrin Muskan 2014288042

Mohammad Mostaqeem Billah 2011277042

**Instruction Type :**

The instruction format we are going to use in this project is R-Type and I-Type. MIPS R-Type Instruction do not require a target address, immediate value, or branch displacement use an R-type coding format. This format has 6 fields for specifying of up to three registers, a shift amount and function. I – Type contains two source/destination registers and immediate value.

**Instruction format :**

R-Type:

|  |  |  |  |
| --- | --- | --- | --- |
| OP (3-bits) | RS (3-bits) | RT (3-bits) | RD (3-bits) |

I-Type:

|  |  |  |  |
| --- | --- | --- | --- |
| OP (3-bits) | RS (3-bits) | RT (3-bits) | Immediate (3-bits) |

We will be using four fields in the R-Type and I-Type format each field length is 3-bits.

The four fields that we will be using are:

Op = operation code.

rs = source register 1.

rt = source register 2.

rd = destination register

Or, Immediate.

**Operands and Operations :**

The instruction we will be using are arithmetic operations which will have 3 operands and 8 operations.

|  |  |  |
| --- | --- | --- |
| Number | code | type |
| 0 | 000 | add |
| 1 | 001 | sub |
| 2 | 010 | addi |
| 3 | 011 | subi |
| 4 | 100 | lw |
| 5 | 101 | sw |
| 6 | 110 | and |
| 7 | 111 | or |

1. ADD $1,$2,$3

ADD A B C [1+2=3]

RS RT Rd

1. SUB $1,$2,$3

SUB A B C [3-2=1]

RS RT RD

1. ADDi $2, $3, 5

ADDi A B C [2+5=7]

RS RT RD

1. SUBi $2, $3, 3

SUBi A B C [5-2=3]

RS RT RD

1. LW $2, $3, 2

LW shift load base

3 3 0

1. SW $2, $3, 2

SW shift store base

0 12(or, c) 0

1. OR $1,$2,$3

OR A B C

111 010 111; [010 OR 111 = 010]

1. AND $1,$2,$3

AND A B C

010 111 010; [010 AND 111 = 010]

Control Unit Table:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Operations** | **Opcode** | **RD/RT (Reg.Dst)** | **Reg. Write.En** | **ALU Src** | **AluOP 1** | **AluOP 0** | **C.in** | **B.Invert** | **lw\_enb** | **sw\_enb** | **RAM to Reg** |
| Add | 000 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| Sub | 001 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 |
| Addi | 010 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| Subi | 011 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 0 |
| Lw | 100 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 1 |
| Sw | 101 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 |
| And | 110 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Or | 111 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |