

PROJECT REPORT

CSE- 332

Section: 12  
Project Name: Designing of ISA  
Date of Submission: 22nd October 2018

Submitted to: Khaleda Ali

Members:

|  |  |  |
| --- | --- | --- |
|  | Name | ID |
| 1. | Hossain Mohammad Mohit | 1421506042 |
| 2. | khorshed |  |
| 3. | Touhidul Raquib Nayeem | 1530941042 |

In this Project we are making a 10 bit mips processor with the help of an assembler which is capable of taking Assembly instruction of MIPS and is able to convert to its appropriate binary equivalent of 10 bits using RISC. We have created the assembler using python.

In this project we have used the instruction Set Architecture of MIPS and we have used R-type and I-type instruction formats but J-type instruction is not implemented.

We have broken the Instruction into two distinct type R-type and I-type.

R-Type Table: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Instruction Name | Opcode | Source Register 1 | Source Register2 | Destination  Register |
| add | 0000 | $t | $t | $t |
| sub | 0001 | $t | $t | $t |
| mul | 0010 | $t | $t | $t |
| div | 0011 | $t | $t | $t |
| and | 0100 | $t | $t | $t |
| or | 0101 | $t | $t | $t |
| not | 0110 | $t | $t | $t |

\*For R type, three operands will be used.

\*It can perform 7 different operations that are mentioned above.

\*It will perform register-register based instruction set.

I-Type Instructions:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

|  |  |  |  |
| --- | --- | --- | --- |
| Name of instructions | Opcode | Register Rs | Register Rt |
| addi | 0111 | $t | $t |
| Sw | 1000 | $t | $t |

\*For I type, three operands will be used.

\*It can perform 2 operations that are mentioned above.

\*It will perform register-memory based instruction set.

Register Table:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

|  |  |  |
| --- | --- | --- |
| Name of Register | Value of register in decimal | Binary |
| $zero | $0 | 00 |
| $t1 | $1 | 01 |
| $s1 | $2 | 10 |
| $s2 | $3 | 11 |

Our processor will compute 9 operations in total. This is for the simplicity of our processor since it is of 10 bits only.

How to use:

The user has to give some instructions to convert into machine codes in the text file in R type or I type. The system will convert valid MIPS instructions into machine language and generate those codes into output.

## Input File

The input file is located in the same folder named “mips”.

Limitation:

Our processor can not perform any J type operation.