CSCI 260 Notes

Ralph Vente Nick Szewczak Anton Goretsky

https://github.com/rvente/CSCI-260-Notes

Contents

2019 02 13

This section provides diagrams of the binary codes implemented in MIPS. Each diagram is an illustration of what value is stored in memory and how the instructions are stored in memory and they physical layout of their arguments.

R-type instructions

```
op rd, rs, rt

---------------------------------------
|        | rs | rt | rd | shamt | code |
---------------------------------------

lw rt, offset(rs)
sw rt, offset(rs)
addi rt, rs, imm

---------------------------------------
|        | rs | rt |    offset         |
---------------------------------------
|  6     | 5  | 5  |      16           |
---------------------------------------
```

`beq`: Branch on equal

000000	rs	rt	offset
6	5	5	16

What is actually stored is the word offset to the next instruction.

`j`: Jump

000000	target
6	26

target = concat(PC[31:28],target * 4)

pg 851 appendix

Note: Don't forget to add 1 in twos complement notation.