By Thomas Bonatti

On behalf of: Josh, Ben Effron, Jeff, Thomas

Jeff and Josh proposed Accumulator with 3 extra registers plus design, and we agreed to use that.

Agreed to use a git-hub repo

Ben proposed a pipeline design, but we decided to analyses our design goals before deciding.

DISCUSSION OF OPPERATIONS

Agreed to use a main register and a move command to save space in instructions by not addressing registers.

**Addressable Registers**

|  |  |  |  |
| --- | --- | --- | --- |
| **Register** | **Purpose** | **Register** | **Purpose** |
| $M | Main accumulator | $t0 | General use |
| $ra | Return address | $t1 | General use |
| $sp | Stack pointer | $s0 | General use |
| $at | Pseudo instructions | $s1 | Safe use |

**Non-Addressable Registers:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Register** | **Purpose** | **Register** | **Purpose** |
| PC | Points to current instruction. | ALUout |  |
|  |  |  |  |

**I-Type instructions (Immediate):**

I-type instructions will use the main accumulator register, and a 12 bit immediate. Most logic and arithmetic commands will use this.

|  |  |
| --- | --- |
| Op Code [4] | Immediate [12] |

**R-type instructions (Register):**

|  |  |  |  |
| --- | --- | --- | --- |
| Op code [4] | Func code [6] | Register [3] | Unused [3] |

**M-type (move):**

Copy and swap

|  |  |  |  |
| --- | --- | --- | --- |
| Op Code [4] | Function code [6 bits] | rd (destination registered) [3] | rs (source registered) [3] |

**IR-Type(Immediate and Register):**

Branches and load word store word

|  |  |  |
| --- | --- | --- |
| Op code [4] | Reg [3] | Immediate [9] |

|  |  |  |  |
| --- | --- | --- | --- |
| **Op Code** | **Operation** | **Op Code** | **Operation** |
| 0000 | Add | 1000 | li (load immediate) |
| 0001 | Beq | 1001 | Xor |
| 0010 | Bne | 1010 | Shift left |
| 0011 | J | 1011 | Srl |
| 0100 | Lw | 1100 | Sra |
| 0101 | Sw | 1101 | Jal |
| 0110 | Lui | 1110 | Jr |
| 0111 | or | 1111 | Use func code |

Move