**Assignment 4 MIPS -> Binary -> Hexadecimal**

Derrick Palma

Unique ID: n01363905\_6065dea943cf2

slt $a3, $a0, $a1 = 000000 10110 10111 11001 00000 001111 = 2D7C80F

and $a3, $s3, $a1 = 00000000111101111100100000010110 = F7C816

srl $t9, $v1, 27 = 00000000011000001010111011010000 = 60AED0

div $t1, $t7 = 00000001101100110000000000010101 = 1B30015

xori $t3, $s7, 11 = 00110101111010110000000000001011 = 35EB000B

slti $s3, $t1, 15 = 00110000111011010000000000001111 = 30ED000F

xor $s1, $a3, $t5 = 00000011001100010010100000010111 = 3312817

sll $t2, $s7, 1 = 00000001011000000111000001001110 = 160704E

mult $a2, $s7 = 00000011000010110000000000010010 = 30B0012

lh $s4, 15($s3) = 01100001000001110000000000001111 = 6107000F

sub $s4, $a0, $s1 = 00000010110001010100000000011000 = 2C54018

lw $t6, 2($t8) = 01010110010101000000000000000010 = 56540002

mflo $t6 = 00000010010000000000000000001100 = 240000C

ori $s2, $s4, 3 = 01011000110010000000000000000011 = 58C80003

sra $t9, $t2, 16 = 00000001110000001010110000010011 = 1C0AC13

lb $t5, 7($a3) = 01010010001110010000000000000111 = 52390007

addi $s4, $a1, 13 = 01000001000101110000000000001101 = 4117000D

sb $t1, 13($t8) = 01011101101101000000000000001101 = 5DB4000D

or $s0, $s2, $v0 = 00000000110000100010000000001101 = C2200D

mfhi $s1 = 00000000101000000000000000010001 = A00011

Table

Description automatically generated

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Registers** | |  | **Instructions** | | | | |  | **Instruction Formats** | | | | | |
| **Number** | **Name** | **Name** | **Type** | **Opcode/** **Function** | **Format** | **Notes** | **R-type** | | | | | |
| 0 | $zero | ADD | R | 20 | add rd, rs, rt |  | **Opcode** | **rs** | **rt** | **rd** | **Shift** | **Function** |
| 1 | $at | ADDI | I | 16 | addi rs, rt, i |  | 6 bits | 5 bits | 5 bits | 5 bits | 5 bits | 6 bits |
| 2 | $v0 | AND | R | 22 | and rd, rs, rt |  |  | | | | | |
| 3 | $v1 | ANDI | I | 11 | andi rs, rt, i |  |
| 4 | $s0 | BEQ | I | 17 | beq rs, rt, i |  | **I-type** | | | | | |
| 5 | $s1 | BNE | I | 18 | bne rs, rt, i |  | **Opcode** | **rs** | **rt** | **Immediate/Offset** | | |
| 6 | $s2 | DIV | R | 21 | div rs, rt | rd = 0 | 6 bits | 5 bits | 5 bits | 16 bits | | |
| 7 | $s3 | J | J | 15 | j address |  |  | | | | | |
| 8 | $s4 | JAL | J | 19 | jal address |  |
| 9 | $s5 | JR | R | 25 | jr rs | rt = 0, rd = 0 | **J-type** | | | | | |
| 10 | $s6 | LB | I | 20 | lb rs, i (rt) |  | **Opcode** | **Target Address** | | | | |
| 11 | $s7 | LH | I | 24 | lh rs, i (rt) |  | 6 bits | 26 bits | | | | |
| 12 | $t0 | LW | I | 21 | lw rs, i (rt) |  |  | | | | | |
| 13 | $t1 | MFHI | R | 17 | mfhi rs | rt = 0, rd = 0 |
| 14 | $t2 | MFLO | R | 12 | mflo rs | rt = 0, rd = 0 |
| 15 | $t3 | MULT | R | 18 | mult rs, rt | rd = 0 |
| 16 | $t4 | OR | R | 13 | or rd, rs, rt |  |
| 17 | $t5 | ORI | I | 22 | ori rs, rt, i |  |
| 18 | $t6 | SB | I | 23 | sb rs, i (rt) |  |
| 19 | $t7 | SH | I | 25 | sh rs, i (rt) |  |
| 20 | $t8 | SLL | R | 14 | sll rd, rs, shift | rt = 0 |
| 21 | $t9 | SLT | R | 15 | slt rd, rs, rt |  |
| 22 | $a0 | SLTI | I | 12 | slti rs, rt, i |  |
| 23 | $a1 | SRA | R | 19 | sra rd, rs, shift | rt = 0 |
| 24 | $a2 | SRL | R | 16 | srl rd, rs, shift | rt = 0 |
| 25 | $a3 | SUB | R | 24 | sub rd, rs, rt |  |
| 26 | $k0 | SW | I | 14 | sw rs, i (rt) |  |
| 27 | $k1 | XOR | R | 23 | xor rd, rs, rt |  |
| 28 | $gp | XORI | I | 13 | xori rs, rt, i |  |
| 29 | $sp |  | | | | |
| 30 | $fp |
| 31 | $ra |