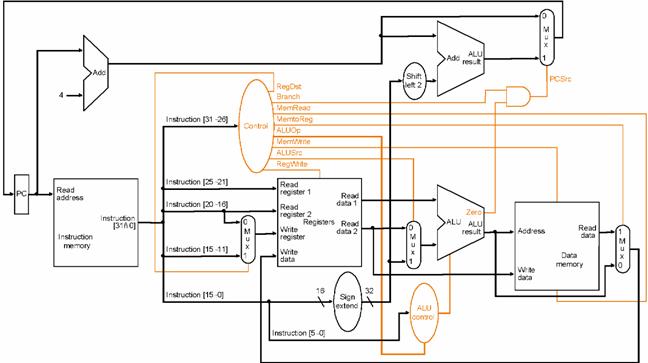
|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Operation | OPcode | RegDst | ALUSrc | Mem2Reg | MemRead | MemWrite | RegWrite | PCSrc | Pop | Push | ALUOp |
| ADD(0) | 000000 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 00000(0) |
| SUB | 000100 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 00100(4) |
| AND | 011000 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 11000(24) |
| OR | 011110 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 11110(30) |
| XOR | 010110 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 10110(22) |
| NOT | 010101 | 0 | X | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 10101(21) |
| SLA | 001000 | 1 | X | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 01000(8) |
| SRA | 001010 | 1 | X | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 01010(10) |
| SRL | 001011 | 1 | X | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 01011(11) |
| ADDI | 000001 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 00000(0) |
| SUBI | 000101 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 00100(4) |
| ANDI | 111000 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 11000(24) |
| ORI | 111110 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 11110(30) |
| XORI | 110110 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 10110(22) |
| LD | 100000 | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 00000(0) |
| ST | 100001 | 0 | 1 | x | 0 | 1 | 0 | 0 | 0 | 0 | 00000(0) |
| BR | 100100 | X | X | X | 0 | 0 | 0 | 1 | 0 | 0 | X |
| BMI | 100101 | x | 0 | x | 0 | 0 | 0 | depends | 0 | 0 | 00100(4) |
| BPL | 100110 | x | 0 | x | 0 | 0 | 0 | depends | 0 | 0 | 00000(0) |
| BZ | 100111 | x | 0 | x | 0 | 0 | 0 | depends | 0 | 0 | 00000(0) |
| CALL | 101010 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 00000(0) |
| RET | 101011 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 00000(0) |
| MOVE | 101110 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 00000(0) |
| MOVEI | 101111 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 00000(0) |
| HALT | 101100 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 00000(0) |
| NO OP | 101101 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 00000(0) |



* Along with this one additional register is kept to facilitate call and return function.