|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Signal Name |  | |  | | | |  | | |  | | |  |
| RegDst | Rd: Instr[7:5] (I-format 1) | | Rd: Instr[4:2] (R-format) | | | | Rs:Instr[10:8] | | | R7 | | | Casex(Instr[15:11])  5’b010??: begin Wrt\_reg=Instr[7:5];  Reg\_En=1;  B\_select=sign\_Imm;  end  5’b101??:begin  Wrt\_reg=Instr[7:5];  Reg\_En=1;  B\_select=sign\_Imm;  end  5’b100??:begin  Wrt\_reg=(instr[12:11]==10)?Instr[10:8]:Instr[7:5];  Reg\_En=(instr[12:11]==00)?0:1;  MemEn=(^inst[12:11]==0)?1:0;  B\_select=sign\_Imm;  end  5’b110??:begin  Wrt\_reg=(instr[12:11]==00)?Instr[10:8]:Instr[4:2];  Reg\_En=1;  B\_select=Rt;  end  5’b111??:begin  Wrt\_reg=Instr[4:2];  Reg\_En=1;  B\_select=Rt;  end  5’b0011?:begin  Wrt\_reg=0x7;  Reg\_En=1;  B\_select=sign\_Imm;  end  Default: begin  Wrt\_reg=x;  Reg\_En=0;  B\_select=sign\_Imm;  End |
| 01000 | ADDI | 11001 | | BTR | | 10011 | | STU | 00110 | | JAL |
| 01001 | SUBI | 11011 | | ADD | | 11000 | | LBI | 00111 | | JALR |
| 01010 | XORI | 11011 | | SUB | | 10010 | | SLBI |  | | |
| 01011 | ANDNI | 11011 | | XOR | |  | | |  | | |
| 10100 | ROLI | 11011 | | ANDN | |
| 10101 | SLLI | 11010 | | ROL | |
| 10110 | RORI | 11010 | | SLL | |
| 10111 | SRLI | 11010 | | ROR | |
| 10000 | ST | 11010 | | SRL | |
| 10001 | LD | 11100 | | SEQ | |
|  | | 11101 | | SLT | |
| 11110 | | SLE | |
| 11111 | | SCO | |
| RegEn | None | | Writedata | | | |  | | | | | |
| 010XX | | | |
| 101XX | | | |
| 100xx(except for 10000 ST) | | | |
| 110xx | | | |
| 111xx | | | |
| 0011x | | | |
| MemEn | None | | Writedata | | | |  | | | | | |
| 10000 | ST | | |
| 10011 | STU | | |
| ALUOP | I-Format | | R-Format | | | |  | | | | | |
| 010XX | | 110xx | | | |
| 101xx | | 111xx | | | |
| 100xx | |  | | | |
| 11000(????) | |  | | | |
| WriteDataSelect | ALU Results | | Mem | | | Zero | | LT | | LTE | Overflow | | Casex(instr[15:11])  5’b010xx: begin  Write\_data=ALU\_result;  End  5’b101xx:begin  Write\_data=ALU\_result;  End  5’b10011:begin  Write\_data=ALU\_result;  End  \*\*\*  5’b110xx: begin  Write\_data=ALU\_result;  End  \*\*\*\*\*  5’b10001:begin  Write\_data=Mem;  End  5’b11100:begin  Write\_data=Zero;  End  5’b11101:begin  Write\_data=LT;  End  5’b11111:begin  Write\_data=LTE;  End  5’b00110:begin  Write\_data=PC+2;  End  5’b00111:begin  Write\_data=PC+2;  End  Default:  Write\_data=ALU\_Result | |
| 010xx | | 10001 | | | 11100 | | 11101 | | 11110 | 11111 | |
| 101xx | |
| 10011(STU) | | PC+2 | | |
| 110xx (except for 11000) | | 00110 | | |
| 00111 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| bits | Imm Sign extended | Imm Zero extended |  |
| 11  Instr[4:0] | 01000 (ADDI) | 01010(XORI) |  |
| 01001 (SUBI) | 01011(ANDNI) |
| 10000 (ST) |  |
| 10001 (LD) |  |
| 10011 (STU) |  |
| 8  Instr[7:0] | 01100 (BEQZ) | 10010(SLBI) |  |
| 01101 (BNEZ) |  |
| 01110(BLTZ) |  |
| 01111(BGEZ) |  |
| 00101(JR) |  |
| 00111 (JALR) |  |
| 5  Instr[10:0] | 00100 (J) |  |  |
| 00110 (JAL) |

|  |  |  |
| --- | --- | --- |
| HW opcode | ALUOpcode | Function |
| 100 (ADD) | 01000 | ADDI |
| 100 | 11011, 00 | ADD |
| 100 | 01001 | SUBI |
| 100 | 11011, 01 | SUB |
| 100 | 10000 | ST |
| 100 | 10001 | LD |
| 100 | 10011 | STU |
| 101 (OR) | 10010 | SLBI (OR) |
| 110 (XOR) | 01010 | XORI |
| 110 | 11011, 10 | XOR |
| 111 (AND) | 01011 | ANDNI |
| 111 | 11011, 11 | ANDN |
| 000 (ROL) | 10100 | ROLI |
| 000 | 11010, 00 | ROL |
| 000 | 10110 | RORI |
| 000 | 11010, 10 | ROR |
| 001 (SLL) | 10101 | SLLI |
| 001 | 11010, 01 | SLL |
| 011 (SRL) | 10111 | SRLI (logical) |
| 011 | 11010, 11 | SRL |
| 011 (sra) ???? |  |  |
| 011(sra) ????? |  |  |

|  |  |  |
| --- | --- | --- |
| New aluopcode (I-format) | Function | |
| instr[15:13]==010 | instr[12:11]==00 ||01 | ADD |
| Instr[12:11]==10 | XOR |
| instr[12:11]==11 | AND |
| Instr[15:13]==100 | Instr[12:11]!=10 | ADD |
| Instr[12:11]==10 | OR |
| Instr[15:13]==101 | Instr[12:11]==00 || 10 | ROL (&ROR) |
| Instr[12:11]==01 | SLL |
| Instr[12:11]==11 | SRL (logical) |

|  |  |  |
| --- | --- | --- |
| New aluopcode (R-format) | Function | |
| Instr[15:11]==11011 | Instr[1:0]==00 ||01 | ADD |
| Instr[1:0]==10 | XOR |
| Instr[1:0]==11 | AND |
| Instr[15:11]==11010 | Instr[1:0]==00 ||10 | ROL(&ROR) |
| Instr[1:0]==01 | SLL |
| Instr[1:0]==11 | SRL(logical) |

|  |  |  |
| --- | --- | --- |
| ALU Invert type | Function | |
| Negate | Instr[15:11]==01011 | ANDNI, ->invB=1, cin=0; |
| Instr[15:11]==11011 &&  Instr[1:0]==11 | ANDN, ->invB=1, cin=0; |
| 2’s complement | Instr[15:11]==01001 | SUBI, ->invA=1, cin=1; |
| Instr[15:11]=11011 &&  Instr[1:0]==01 | SUB , ->invA=1, cin=1; |

|  |  |  |
| --- | --- | --- |
| ALU Signed operation | Function | |
| ADDI | Instr[15:11]==01000 | Sign=1 |
| SUBI | Instr[15:11]==01001 |
| ST | Instr[15:11]==10000 |
| LD | Instr[15:11]==10001 |
| STU | Instr[15:11]==10011 |
| ADD | Instr[15:11]==11011 &&  Instr[1:0]==00 |
| SUB | Instr[15:11]==11011 &&  Instr[1:0]==01 |
| SEQ | Instr[15:11]==11100 |
| SLT | Instr[15:11]==11101 |
| SLE | Instr[15:11]==11110 |
| SCO | Instr[15:11]==11111 |

|  |  |  |
| --- | --- | --- |
| ALU Unsigned operation | Function | |
| XORI | Instr[15:11]==01010 | Sign=0 |
| ANDNI | Instr[15:11]==01011 |
| ROLI | Instr[15:11]==10100 |
| SLLI | Instr[15:11]==10101 |
| RORI | Instr[15:11]==10110 |
| SRLI | Instr[15:11]==10111 |
| XOR | Instr[15:11]==11011 &&  Instr[1:0]=10 |
| ANDN | Instr[15:11]==11011&&  Instr[1:0]=11 |
| ROL | Instr[15:11]==11010&&  Instr[1:0]=00 |
| SLL | Instr[15:11]==11010&&  Instr[1:0]=01 |
| ROR | Instr[15:11]==11010&&  Instr[1:0]=10 |
| SRL | Instr[15:11]==11010&&  Instr[1:0]=11 |