|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | PC ctrl | Main ctrl | | | | ALU ctrl | | | | | | Main ctrl | | | | | | | |
| Instruction | opcode | PC src | Ext op | Selection1 | Src1 | RegWrite | Zero | Carry | Overflow | positive | negative | ALU op | Selection2 | Memory data | Wb data | Memory read | Memory write | OPERAND2 | Mode | Data selector |
| AND | 0000 | 0 | X | 0 | 1 | 1 | X | X | X | X | X | AND | 1 | X | 0 | 0 | 0 | 1 | X | X |
| ADD | 0001 | 0 | X | 0 | 1 | 1 | X | X | X | X | X | ADD | 1 | X | 0 | 0 | 0 | 1 | X | X |
| SUB | 0010 | 0 | X | 0 | 1 | 1 | X | X | X | X | X | SUB | 1 | X | 0 | 0 | 0 | 1 | X | X |
| ADDI | 0011 | 0 | 1 | X | 1 | 1 | X | X | X | X | X | ADD | 1 | X | 0 | 0 | 0 | 0 | X | X |
| ANDI | 0100 | 0 | 0 | X | 1 | 1 | X | X | X | X | X | AND | 1 | X | 0 | 0 | 0 | 0 | X | X |
| LW | 0101 | 0 | 1 | X | 1 | 1 | X | X | X | X | X | ADD | 1 | X | 1 | 1 | 0 | 0 | X | 0 |
| LBU | 0110 | 0 | 1 | X | 1 | 1 | X | X | X | X | X | ADD | 1 | X | 1 | 1 | 0 | 0 | 0 | 1 |
| LBS | 0110 | 0 | 1 | X | 1 | 1 | X | X | X | X | X | ADD | 1 | X | 1 | 1 | 0 | 0 | 1 | 1 |
| SW | 0111 | 0 | 1 | X | 1 | 0 | X | X | X | X | X | ADD | 1 | 1 | X | 0 | 1 | 0 | X | X |
| BGT | 1000 | 1 | 1 | 1 | 1 | 0 | 0 | X | 0 | 1 | 0 | SUB | X | X | X | 0 | 0 | X | X | X |
| BGTZ | 1000 | 1 | 1 | 1 | 0 | 0 | 0 | X | X | 1 | 0 | SUB | X | X | X | 0 | 0 | X | X | X |
| BLT | 1001 | 1 | 1 | 1 | 1 | 0 | X | X | 0 | 0 | 1 | SUB | X | X | X | 0 | 0 | X | X | X |
| BLTZ | 1001 | 1 | 1 | 1 | 0 | 0 | X | X | X | 0 | 1 | SUB | X | X | X | 0 | 0 | X | X | X |
| BEQ | 1010 | 1 | 1 | 1 | 1 | 0 | 1 | X | X | X | X | SUB | X | X | X | 0 | 0 | X | X | X |
| BEQZ | 1010 | 1 | 1 | 1 | 0 | 0 | 1 | X | X | X | X | SUB | X | X | X | 0 | 0 | X | X | X |
| BNE | 1011 | 1 | 1 | 1 | 1 | 0 | 0 | X | X | X | X | SUB | X | X | X | 0 | 0 | X | X | X |
| BNEZ | 1011 | 1 | 1 | 1 | 0 | 0 | 0 | X | X | X | X | SUB | X | X | X | 0 | 0 | X | X | X |
| JUMP | 1100 | 2 | X | X | X | 0 | X | X | X | X | X | X | X | X | X | 0 | 0 | X | X | X |
| CALL | 1101 | 2 | X | X | X | 0 | X | X | X | X | X | X | X | X | X | 0 | 0 | X | X | X |
| RET | 1110 | 3 | X | 2 | X | 0 | X | X | X | X | X | X | X | X | X | 0 | 0 | X | X | X |
| SV | 1111 | 0 | X | X | 1 | 0 | X | X | X | X | X | X | 0 | 0 | X | 0 | 1 | X | X | X |

Sara Awayssa-1211642

Hala Hammad-1210606

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Signal | PC src | Ext op | Selection 1 | Src1 | RegWrite | Zero | Carry | Overflow | Positive | Negative | ALU op | Selection2 | Memory data | Wb data | Memory read | Memory write | Operand2 | Mode | Data selector |
| Description | Used to choose the suitable value of next PC. | Used to select if the required extension is zero or signed extension. | Used to choose the second operand which will appear at the 2nd read bus. | Used to select the first operand which enters the ALU. | A flag if there is a required data to write back it to the register file. | A flag resulted from ALU to indicate if the result is 0. | A flag resulted from ALU to indicate if the result contains a carry. | A flag resulted from ALU to indicate if there is an overflow. | A flag resulted from ALU to indicate if the result is positive. | A flag resulted from ALU to indicate if the result is negative. | Used to determine the used ALU unit. | Used to select the required data memory address. | Used to select the data in to be written to the data memory is store instructions. | Used to select the required data to be written back to the register file. | Used to indicate if the data memory will be read. (Load) | Used to indicate if the data memory will be written. (store) | Used to choose the second operand which enters the ALU. | Used to determine the type of extension to the loaded byte. (0 or sign) | Used to select the data out in load operation (word or extended byte) |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Signal | Ext op | Selection 1 | Src1 | RegWrite | Zero | Carry | Overflow | Positive | Negative | ALU op | Selection2 | Memory data | Wb data | Memory read | Memory write | Operand2 | Mode | Data selector |
| Boolean equation | ~ANDI. |  | Used to select the first operand which enters the ALU. | A flag if there is a required data to write back it to the register file. | A flag resulted from ALU to indicate if the result is 0. | A flag resulted from ALU to indicate if the result contains a carry. | A flag resulted from ALU to indicate if there is an overflow. | A flag resulted from ALU to indicate if the result is positive. | A flag resulted from ALU to indicate if the result is negative. | Used to determine the used ALU unit. | Used to select the required data memory address. | Used to select the data in to be written to the data memory is store instructions. | Used to select the required data to be written back to the register file. | Used to indicate if the data memory will be read. (Load) | Used to indicate if the data memory will be written. (store) | Used to choose the second operand which enters the ALU. | Used to determine the type of extension to the loaded byte. (0 or sign) | Used to select the data out in load operation (word or extended byte) |