**Janus Specification**

**-** RISC architecture, load/store oriented

- 32-bit data, address and instruction width

- Supports up to 4Gb RAM

- 16 internal GP registers

- Supports addition, subtraction, AND, OR, XOR, NOT, shift left/right

**Instruction Set:**

Note: → Reg = Register

Imme. = Immediate value

DC = Don't care

|  |  |
| --- | --- |
| **Syntax:** | **Description:** |
| **LD** Reg1, Reg2 | Loads data from memory address stored in Reg2 into Reg1 |
| **ST** Reg1, Reg2 | Stores data from Reg1 into memory address stored in Reg2 |
| **ADD** Reg1, Reg2, Reg3 | Adds Reg1 and Reg2, stores value in Reg3 |
| **SUB** Reg1, Reg2, Reg3 | Subs Reg2 from Reg1, stores value in Reg3 |
| **AND** Reg1, Reg2, Reg3 | ANDs Reg1 and Reg2, stores value in Reg3 |
| **OR** Reg1, Reg2, Reg3 | ORs Reg1 and Reg2, stores value in Reg3 |
| **XOR** Reg1, Reg2, Reg3 | XORs Reg1 and Reg2, stores value in Reg3 |
| **SFR** Reg | Shifts Reg right 1 bit |
| **SFL** Reg | Shifts Reg left 1 bit |
| **MOV** Reg1, Reg2 | Moves Reg2 into Reg1 |
| **MVL** Reg, Imme. | Moves Imme. Into lower 16 bits of Reg, does not overwrite existing higher 16 bits |
| **MVH** Reg, Imme. | Moves Imme. Into higher 16 bits of Reg, does not overwrite the lower 16 bits |
| **CLR** Reg | Clears Reg |
| **JMP** Reg | Absolute jump to memory address stored in Reg |
| **JEZ** Reg1, Reg2 | Jumps to memory address stored in Reg2 if the value in Reg1 is equal to zero |
| **JGZ** Reg1, Reg2 | Jumps to memory address stored in Reg2 if the value in Reg1 is greater than zero |
| **JLZ** Reg1, Reg2 | Jumps to memory address stored in Reg2 if the value in Reg1 is less than zero |
| **HLT** | Halts the system |
| **NOP** | No operation |

**Addressing and Data:**

**Registers: →** 8-bit, #0 → #15 general purpose, #255 program counter

**Addresses:→** 32-bit, #0 → #65535

**Immediate values: →** Move command → 16-bit, signed -32768 → 32767, unsigned 0 → 65535

**Instruction Layout:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 8 | 8 | 8 | 8 |  |
| ID | R1 | R2 | R3 | → ADD, SUB, AND, OR, XOR |
| ID | R1 | R2 | DC | → LD, ST, JEZ, JGZ, JLZ, MOV |
| ID | R1 | Immediate | | → MVL, MVH |
| ID | R1 | DC | | → SFR, SFL, CLR, JMP |
| ID | DC | | | → HLT, NOP |

**ID Layout:**

- There will be four types of instructions: - 00→Jumps and misc.

- 01→Arithmetic

- 10→Logic

- 11→ Memory management

|  |  |
| --- | --- |
| 2 | 6 |
| Type | Binary assignment |

|  |  |
| --- | --- |
| **Instruction:** | **ID:** |
| NOP | 00-000000 |
| JMP | 00-000001 |
| JEZ | 00-000010 |
| JGZ | 00-000011 |
| JLZ | 00-000100 |
| HLT | 00-000101 |
|  |  |
| ADD | 01-000000 |
| SUB | 01-000001 |
|  |  |
| AND | 10-000000 |
| OR | 10-000001 |
| XOR | 10-000010 |
| SFL | 10-000011 |
| SFR | 10-000100 |
|  |  |
| LD | 11-000000 |
| ST | 11-000001 |
| MOV | 11-000010 |
| MVL | 11-000011 |
| MVH | 11-000100 |
| CLR | 11-000101 |

**External Connections/ IO:**

|  |  |  |
| --- | --- | --- |
| **Name (Abbrev.) - (Size in bits):** | **Direction:** | **Description:** |
| Clock – (clk) - (1) | Input | The system can take in a clock signal that it can distribute, divide, stop etc. |
| Reset – (rst) - (1) | Input | This is the master reset of the system, it wipes all internal registers to zero, clears the bus, ALU and program counter, turns off all control signals. |
| Data in bus – (dib) - (32) | Input | This is the 32-bit bus that carries data into the system. |
| Data out bus – (dob) - (32) | Output | This is the 32-bit bus that carries data from the system. |
| Address bus – (ab) - (32) | Output | This is the 32-bit bus that carries the addresses the system will use to the respective external devices. |
| Control bus – (cb) - (tbd) | Input/Output | This is the n-bit bus that carries all the control signals to the external devices, eg. R/W request, R/W ack etc. |
| Halt – (hlt) – (1) | Output | This is the halt signal that feeds to the external oscillator to halt the system. |

**Improvements:**

- Additional Arithmetic/Logic → Floating point, comparators, multiply, divide

**-** Give arithmetic/logic instructions immediate value capabilities

- Interrupt capabilities

- I2C/SPI/UART/USART

- PWM

- 2 → 4 Parallel I/O Ports

- Increased number of internal registers

- Special Function Registers → PSW, Interrupts, Null

- Timers

- Sleep modes

- Pipe-lining → Speculative

- Threading and scalability