

ISA description:

The ISA has 6 encoding types of instructions. The description of the types is given later.

Opcode	Instruction	Semantics Syntax	Type
10000	Addition	Performs $reg3 = reg1 + reg2$. add reg1 reg2 reg3 If the computation overflows, then the overflow flag is set	A
10001	Subtraction	Performs $reg3 = reg1 - reg2$. sub reg1 reg2 reg3 In case $reg2 > reg1$, 0 is written to reg3 and overflow the flag is set.	A
10010	Move Immediate	Performs $reg1 = \$Imm$ mov reg1 \$Imm where Imm is an 8 bit value.	B
10011	Move Register	Performs $reg2 = reg1$ mov reg1 reg2	C
10100	Load	Loads data from mem_addr into reg1. ld reg1 mem_addr	D
10101	Store	Stores data from reg1 to mem_addr. st reg1 mem_addr	D
10110	Multiply	Performs $reg3 = reg1 \times reg2$. mul reg1 reg2 reg3 If the computation overflows, then the overflow flag is set.	A

10111	Divide	Performs reg3/reg4. div reg3 reg4 Stores the quotient in R0 and the remainder in R1.	C
11000	Right Shift	Right shifts reg1 by rs reg1 \$Imm \$Imm, where \$Imm is an 8 bit value.	
11001	Left Shift	Left shifts reg1 by ls reg1 \$Imm \$Imm, where \$Imm is an 8 bit value.	B
11010	Exclusive OR Performs bitwise	XOR of reg1 and reg2. Stores the result in reg3. xor reg1 reg2 reg3	A

11011	Or	Performs bitwise OR or reg1 reg2 reg3 of reg1 and reg2. Stores the result in reg3.	A
11100	And	Performs bitwise AND and reg1 reg2 reg3 of reg1 and reg2. Stores the result in reg3.	A
11101	Invert	Performs bitwise NOT not reg1 reg2 of reg1. Stores the result in reg2.	C

11110	Compare	Compares reg1 and reg2 and sets up the FLAGS register. cmp reg1 reg2	C
-------	---------	---	---

11111	Unconditional Jump	Jumps to mem_addr, jmp mem_addr where mem_addr is a memory address.	E
01100	Jump If Less Than	Jump to mem_addr if jlt mem_addr the less than flag is set (less than flag = 1), where mem_addr is a memory address.	E
01101	Jump If Greater Than	Jump to mem_addr if the greater than flag is set (greater than flag = 1), where mem_addr is a memory address. jgt mem_addr	E

01111	Jump If Equal	Jump to mem_addr if the equal flag is je mem_addr set (equal flag = 1), where mem_addr is a memory address.	E
-------	---------------	--	---

01010	Halt	Stops the machine from executing until reset hlt	F
-------	------	---	---

where reg(x) denotes register, mem_addr is a memory address (must be an 8-bit binary number), and **imm** denotes a constant value (must be an 8-bit binary number). The ISA has 7 general purpose registers and 1 flag register. The ISA supports an address size of **8 bits**, which is **double byte addressable**. Therefore, each address fetches two bytes of data. This results in a total address space of 512 bytes. **This ISA only supports whole number arithmetic**. If the subtraction results in a negative number; for example “3 - 4”, the reg value will be set to 0 and overflow bit will be set. All the representations of the number are hence unsigned. The registers in assembly are named as R0, R1, R2, ... , R6 and FLAGS. Each register is 16 bits.