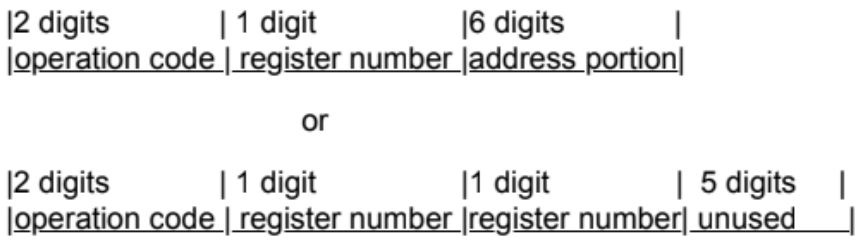
The SANSKAR'S ASSEMBLER is a decimal computer with 1,000,000 words of memory. Each word consists of 9 decimal digits. There are an additional 10 words of memory called registers. These are on chip memory. The machine language instructions for the assembler are of the following form.



A machine language program is a sequence of machine language instructions stored in memory. The computer runs a machine language program by executing machine instructions stored in successive words of memory. The SANSKAR'S assembler machine assumes that the first instruction to be executed will be at location 100. The following are the machine language instructions for the assembler.

NAME	OP. CODE	MEANING
ADD	01	Reg <-- c(Reg) + c(ADDR) (The contents of the register specified in the instruction and of the memory location specified by the address portion of the instruction are added together. The result is placed in the register.):
SUBTRACT	02	Reg <-- c(Reg) - c(ADDR)
MULTIPLY	03	Reg <-- c(Reg) * c(ADDR)
DIVIDE	04	Reg <-- c(Reg) / c(ADDR)
LOAD	05	Reg <-- c(ADDR)
STORE	06	ADDR <-- c(Reg)
ADD REG	07	REG1 <--c(REG1) + c(REG2) (Note: this is the second instruction format where two registers are specified.)
SUB REG	08	REG1 <--c(REG1) - c(REG2)
MULT REG	09	REG1 <--c(REG1) * c(REG2)

DIV REG	10	REG1 <--c(REG1) / c(REG2)
READ	11	A line is read in and the number found there is recorded in the specified memory address. The register value is ignored.
WRITE	12	c(ADDR) is displayed The register value is ignored.
BRANCH	13	go to ADDR for next instruction. The register value is ignored.
BRANCH MINUS	14	go to ADDR if c(Reg) < 0
BRANCH ZERO	15	go to ADDR if c(Reg) = 0
BRANCH POSITIVE	16	go to ADDR if c(Reg) > 0
HALT	17	terminate execution. The register value and address are ignored.