

Instruction-set of the SIMPLE processor

OpCode	Operanden	Instructie	Operatie
2	RXY	load R,XY	register[R]:=XY immediate load
1	RXY	load R,[XY]	register[R]:=memory[XY] direct load
3	RXY	store R,[XY]	memory[XY]:=register[R] direct store
D	ORS	load R,[S]	register[R]:=memory[register[S]] indirect load
E	ORS	store R,[S]	memory[register[s]]:=register[R] indirect store
4	ORS	move S,R	register[S]:=register[R]
5	RST	addi R,S,T	register[R]:=register[S]+register[T] integer add (twos complement representation)
6	RST	addf R,S,T	register[R]:=register[S]+register[T] floating-point add
7	RST	or R,S,T	register[R]:=register[S] OR register[T] bitgewijze OR
8	RST	and R,S,T	register[R]:=register[S] AND register[T] bitwise AND
9	RST	xor R,S,T	register[R]:=register[S] XOR register[T] bitwise eXclusieve OR
A	R0X	ror R,X	register[R]:=register[R] ROR X ROtate Right register R met X bits
B	RXY	jmpEQ R=R0,XY	PC:=XY, if register[R]=register[R0]
F	RXY	jmpLE R<=R0,XY	PC:=XY, if register[R]<=register[R0] (twos complement representation)
B	0XY	jmp XY	PC:=XY
C	000	halt	stop program

Some directives (pseude-instructions):

org *adr* next code starts at address *adr*.
db *data_1, data_2, ..., data_n* puts data directly into the memory