Dokumentation HC1 16-Bit

Notations: A accumulator
M memory
PC program counter
aaaaa five bits for specifying a memory address
eeeee five bits for specifying a memory address > 1

8-Bit Instructions

Instruction	Encoding	Operation	Comment
LOAD A, address	000 aaaaa	A ← M[address]	Load A with content of memory location
STORE A, address	001 aaaaa	M[address] ← A	Store A into memory location
ADD A, address	010 aaaaa	A ← A + M[address]	Add A with M[address] and store result back into A
SUB A, address	011 aaaaa	A ← A - M[address]	Subtract A by M[address] and store result back into A
NAND A, address	100 eeeee	A ← not(A and M[address]	Perform bitwise logical NAND operation of A and M[address] and store result back into A
IN A	100 00000	A ← Input	Input to A
OUT A	100 00001	Output ← A	Output from A
JZ address	101 aaaaa	IF A == 0 THEN PC ← address	Jump to address if A is zero
JPOS address	110 aaaaa	IF A > 0 THEN PC ← address	Jump to address if A is a positive number
J address	111 aaaaa	PC ← address	Jump always to address

The upper 8 Bit of the instruction must be 0.
Example:
0000 0000 010 11111-- ADD

16-Bit Instructions

Instruction	Encoding	Operation	Comment
LOAD A, address	0001 aaaaaaaaaaaa	A ← M[address]	Load A with content of memory location
STORE A, address	0010 aaaaaaaa	M[address] ← A	Store A into memory location
ADD A, address	0011 aaaaaaaaaaaa	A ← A + M[address]	Add A with M[address] and store result back into A
SUB A, address	0100 aaaaaaaaaaaa	A ← A - M[address]	Subtract A by M[address] and store result back into A
NAND A, address	0101 aaaaaaaaaaaa	A ← not(A and M[address]	Perform bitwise logical NAND operation of A and M[address] and store result back into A
IN A	0110 000000000000	A ← Input	Input to A
OUT A	0110 000000000001	Output ← A	Output from A
JZ address	0111 aaaaaaaaaaaa	IF A == 0 THEN PC ← address	Jump to address if A is zero
JPOS address	1000 aaaaaaaaaaaa	IF A > 0 THEN PC ← address	Jump to address if A is a positive number
J address	1001 aaaaaaaaaaaa	PC ← address	Jump always to address
DIV A, address	1010 aaaaaaaaaaaa	A ← A / M[address]	Performs an integer division of A divided by M[address] and writes it back to A
MUL A, address	1011 aaaaaaaaaaaa	A ← A * M[address]	Performs a multiplication for up to 7*7 bits and writes it back to A
ADDI A, value	1100 bbbbbbbbbbbb	A ← A + bbbbbbbbbbbb	Adds an 12 bit integer to A and stores it in A and writes it back to A
SUBI A, value	1101 bbbbbbbbbbbb	A ← A - bbbbbbbbbbbb	Subtracts a 12 bit integer from A and writes it back to A
MOD A, address	1110 aaaaaaaaaaaa	A ← A % M[address]	Performs a modulo operation of A and M[address] and writes it back to A

Architektur of HC1 16-Bit (23.01.2018)