Addenum

Two-byte instructions		One-byte instructions	
		NOP	0x00
		END	0x02
MOV A, address	0x11	MOV A, [AP]	0x14
MOV AP, address	0x13	MOV A, [SP+]	0x1C
MOV A, const	0x19	MOV AP, [SP+]	0x1E
MOV AP, const	0x1B		
MOV address, A	0x21	MOV [-SP], A	0x2C
MOV address, AP	0x23	MOV [-SP], AP	0x2E
ADD A, address	0x31	ADD A, [AP]	0x34
ADD A, const	0x39		
ADD AP,const	0x3B		
SUB A, address	0x41	SUB A, [AP]	0x44
SUB A, const	0x49		
SUB AP, const	0x4B		
		NOT	0x50
OR address	0x61	OR [AP]	0x64
OR const	0x69		
AND address	0x71	AND [AP]	0x74
AND const	0x79		
XOR address	0x81	XOR [AP]	0x84
XOR const	0x89		
		SHR	0x90
CALL procedure	0xC1	RET	0xB0
		IN	0xD0
		OUT	0xE0
JMP label	0xA1		
JZ label	0xA5		
JC label	0xA9		

Note: in case of two-byte instructions, second byte is address or constant. So `address`, `const`, `label` and `procedure` will be compiled into appropriate binary number as second byte.