

Návrh strojového kódu

Prvý bajt	Druhý bajt	Assemblerový zápis	Význam
00		NOP	Neurob nič
01	n	LDA n	Prenes bajt z adresy n do registra A
02	n	STA n	Prenes bajt z registra A na adresu n
03	n	ADD n	Pripočítaj k A hodnotu z adresy n Nastav bit Z: keď je výsledok nula tak 1, inak 0
04	n	SUB n	Odčítaj od A hodnotu z adresy n, Nastav bit Z: keď je výsledok nula tak 1, inak 0
05	adr	JMP adr	Skoč na adresu adr
06	n	CMP n	Nastav bit G: keď je A neznamienkovo väčšie než obsah adresy n tak 1, inak 0 Nastav bit Z: keď je A rovné obsahu adresy n tak 1, inak 0
07	adr	JNE adr	Ak je bit Z v reg. F rovný 0, tak skoč na adresu adr
08	adr	JG adr	Ak je bit G v reg. F rovný 1, tak skoč na adresu adr
FF		HALT	Zastav procesor
x		DATA x	Pseudoinštrukcia - ulož do výsledného programu číslo x

Prvý program v strojovom kóde

Adresa	Strojový kód	Assemblerový zápis
00	01 05	LDA 5
02	03 06	ADD 6
04	FF	HALT
05	02	DATA 2
06	03	DATA 3

Symbolické vyjadrenie adries odbremení programátora od zisťovania ich číselnej hodnoty:

LDA X
ADD Y
HALT
X: DATA 2
Y: DATA 3

Čo bude v registri A keď procesor zastaví?

Assemblerový jazyk sa nazýva aj Jazyk symbolických adries