

Simple 8-bit Assembler Simulator

- Simulator: schweigi.github.io/assembler-simulator
- Dokumentation: schweigi.github.io/assembler-simulator/instruction-set.html

Speicher

Register (6 Byte)

Speicher	Name	Bedeutung
1 Byte	A	beliebige Verwendung
1 Byte	B	beliebige Verwendung
1 Byte	C	beliebige Verwendung
1 Byte	D	beliebige Verwendung
1 Byte	IP: Instruction Pointer	enthält Adresse des nächsten Befehls
1 Byte	SP: Stack Pointer	enthält Adresse des Stacks

Flags (3 bit)

Speicher	Name	Bedeutung
1 bit	Z: Zero	TRUE, wenn Ergebnis des letzten Befehls 0
1 bit	C: Carry	TRUE, wenn letzter Befehl Übertrag erzeugte
1 bit	F: Failure	TRUE, wenn letzter Befehl ungültig

RAM (256 Byte)

Adresse	Bedeutung
0x00 - ...	Speicher für Programmcode
... - 0xE7	Stack
0xE8 - 0xFF	Output

Befehle

MOV

MOV A, B	Inhalt von B nach A kopieren
MOV A, 17	Wert 17 in Register A schreiben
MOV A, 0x11	Wert 17 in Register A schreiben
MOV A, [17]	Wert der Zelle 17 vom RAM Register A schreiben
MOV [17], A	Inhalt des Registers A in Zelle 17 in RAM kopieren
MOV [17], 25	Wert 25 in Zelle 17 in RAM schreiben

Stack

PUSH A Inhalt von A auf Stack kopieren
PUSH [17] Inhalt der Zelle 17 vom RAM auf Stack kopieren
POP A Inhalt des Stack in Register A kopieren

Rechnen

ADD A, B A += B
ADD A, [17] Inhalt der Speicherzelle 17 vom RAM zu Register A dazuzaddieren
SUB A, B A -= B
INC A A ++
DEC A A --
MUL B A *= B
DIV B A /= B

Vergleich

CMP A, B Setzt ZERO-Flag auf TRUE, wenn A==B

Sprünge

JMP 17 Sprung in RAM-Zelle 17
JZ 17 Sprung in RAM-Zelle 17, wenn ZERO-Flag TRUE
JNZ 17 Sprung in RAM-Zelle 17, wenn ZERO-Flag FALSE