Adressierung

Mit der Einführung des i80386 standen 32-Bit-Register zur Verfügung. Mit einem 32-Bit-Zeiger kann man 23^2 = 4GB Speicher adressieren. Jede Adresse wird durch einen einzigen 32-Bit-Wert beschrieben, den 32-Bit-**Offset**.

Adressierungsarten

Unmittelbare Adressierung

Bei der unmittelbaren Adressierung (immediate adressing) steht der Quelloperand unmittelbar im Befehl. Er wird bei Übersetzung fest in den Maschinencode eingebunden und folgt unmittelbar auf den Befehlscode.

Beispiele:

```
MOV BX,9                ; 9 wird unmittelbar ins Register bx geschrieben
BYTE_VALUE  DB  150     ; Ein Byte-Wert (BYTE_VALUE) wird zum Wert 150
definiert
ADD  BYTE_VALUE, 65     ; Es wird unmittelbar der Wert 65 addiert
MOV  AX, 45H            ; 45H wird unmittelbar ins Register AX geshrieben
```

Registeradressierung

Der Registeradressierung sind Quelle und Ziel interne Register des Prozessors.

Beispiele:

MOV EAX, EBX

Direkte Speicheradressierung

Bei der direkten Speicheradressierung wird der Offset des adressierten Speicherplatzes direkt angegeben und liegt nach der Assemblierung fest. Die Angabe des Offsets kann als konstante Zahl oder (besser) als Variablenname erfolgen. Der Variablenname kann – muss aber nicht – in eckigen Klammern eingeschlossen sein. Es können auch Feldelemente direkt adressiert werden, indem nach dem Feldnamen ein Pluszeichen und eine Konstante folgen.

Beispiele:

```
mov ax, Zaehler1 ;Direkte Adressierung ohne eckige Klammern
```

```
mov [bigcount],ecx ;Direkte Adressierung mit eckigen Klammern
mov ecx, [Feld+2] ;Direkte Adr. von Feld + 2 Byte
```

From:
https://info-bw.de/ -

Permanent link:
https://info-bw.de/faecher:informatik:oberstufe:techinf:assembler:adressierung:start?rev=1632158920

Last update: **20.09.2021 17:28**