

Adressierungsarten

Wie das Wort schon sagt behandeln Adressierungsarten unterschiedliche Arten von Adressierungen im CPU-System. Es beschreibt auf welche Operanden sich ein Maschinenbefehl bezieht.

Maschinenbefehle sind die Art von Befehlen, die die CPU ausführen kann. Diese Befehle können entweder direkt in Assembler, der hardwarenächsten, aber auch am aufwendigsten zu schreibenden Programmiersprache (nach Maschinencode) geschrieben werden, oder man benutzt eine höhere Programmiersprache (z.B. Python oder Java) und der Compiler wandelt diesen Code automatisch in Assembler-Code um.

Der Vorteil der Assembler Programmierung besteht dabei darin dass derjenige der den Code schreibt direkten Einfluss darauf nehmen kann, welche Adressierungsart verwendet wird. Bei der wesentlich komfortableren Programmierung in einer höheren Programmiersprache verliert der Programmierer diese Kontrolle, da der Compiler den Code in Assembler umwandelt.

Dabei kann man grob in zwei Arten der Adressierung unterteilen. Bei der einen wird der Befehl direkt an den Hauptspeicher des CPUs übermittelt, damit dieser den Befehl verarbeiten kann. Bei der zweiten Option wird zuerst nur eine Speicheradresse übergeben, welche dann vom Prozessor abgerufen wird.

Oben wurden die zwei Groben Unterschiede beschrieben, nun werden wir einmal die häufigsten Adressierungsarten aufzählen.

Registeradressierung

Wie der Name auch hier bereits sagt, liegt hier der Befehl in einem der Register des Prozessorsystems. Nun kann es nur noch sein, dass der Befehl entweder bereits im Richtigen Register steht, der Befehl in einem festgelegten Register zu finden ist oder das Register über einen im Registerfeld eingetragenen Code gefunden werden muss.

Unmittelbare Adressierung

Hierbei sind die Operanden nicht direkt im Befehl enthalten, sondern folgen in dem dem Speicher des Befehl direkt folgenden Speicherraum.

Direkte Adressierung

Dies ist die Adressierungsart, bei der nicht die Operanden, sondern lediglich eine vollständige Speicheradresse angegeben wird, die direkt zu den Operanden führt.

Indirekte Adressierung

Bei der indirekten Adressierung hat man im Gegensatz zu der direkten Adressierung nicht die vollständige Speicheradresse vorliegen, sondern nur eine Basisadresse, zu der Zahlen hinzuaddiert werden, die in einem anderen Register liegen, oder aus den Zahlen in einem anderen Register generiert werden. Dies ist nützlich, wenn man zur Entwicklungszeit noch nicht weiß, welche Zahlen in dem Register stehen werden.

Indizierte Adressierung

Bei der indizierten Adressierung wird, ähnlich wie bei der indirekten Indizierung, zu einer Basisadresse ein Indexregister addiert und das Ergebnis daraus entspricht dann der letztendlichen Adresse. Dabei gibt es mehrere Arten, einen zu addierenden Wert festzulegen.