

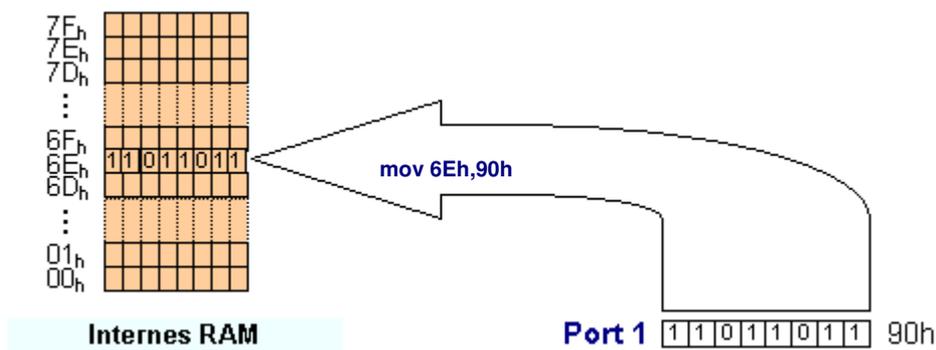
 Friedrich-Ebert-Schule Esslingen	MIKROCONTROLLER	Name:
2.3.4	Befehlsaufbau	Datum:

Einzelne Programmanweisungen oder Operationen (=Maschinenbefehle) werden im Mikrocomputer als Binärwerte im Programmspeicher abgelegt. Man unterscheidet zwischen Operationen zum Datentransport, Arithmetikoperationen, logischen Operationen und Programmsteueroperationen. Je nach Art und Verwendung benötigt ein vollständiger Maschinenbefehl 1, 2 oder 3 Byte Speicherplatz. Das erste Byte enthält dabei immer den **Operationscode** (kurz: Opcode). Im Befehlsumfang eines 8-Bit μC gibt es maximal $2^8 = 256$ Opcodes.

Jeder **Befehlszyklus** beginnt immer mit dem Lesen und Dekodieren des Opcodes, anhand dessen das Steuerwerk die weiteren Befehlsschritte steuert. Bei 2- oder 3-Byte-Befehlen veranlasst das Steuerwerk, dass weitere Bytes aus dem Programmspeicher gelesen werden. Diese werden als Operanden bezeichnet und enthalten bei einem Transportbefehl zum Beispiel Quell- und Zieladresse für die zu transferierenden Daten.

Beispiel: Ein 8-Bit Eingangsbyte an Port 1 (SFR-Adresse 90h) des Controllers soll in die Speicherstelle mit der Adresse 6Eh im internen RAM geschrieben werden. Der Assembler-Befehl hierzu lautet:

`mov 6Eh,90h ; 6Eh = Zieladresse, 90h = Quelladresse`



Übersetzt in den binären Maschinencode des Controllers lautet dieser Befehl:

10000101 10010000 01101110
Opcode **Quelle** **Ziel**

Hier steht zunächst der Quell- und anschließend der Zieloperand. In der mnemonischen Darstellung gilt die umgekehrte Reihenfolge.

Zur Anzeige werden die binären Maschinencodes in einen Hexcode umgewandelt. Für das Beispiel ergibt sich folgender Hexcode:

85 90 6E

Das Speichermapping zeigt die Anordnung der drei Befehlsbytes im Programmspeicher.

Der Befehl wird während der Programmlaufzeit in folgenden **Schritten** umgesetzt:

1. Aus dem Programmspeicher wird zunächst der **Opcode** gelesen (Opcode-Fetch).
2. Der Befehlsdeko­der erkennt den Befehl **mov adr,adr** (Opcode 85h). Es werden 2 weitere Befehlsbytes benötigt.
3. Das Steuerwerk holt die Quelladresse für den mov-Befehl aus dem Programmspeicher.
4. Die Zieladresse der Daten wird gelesen.
5. Der mov-Befehl wird ausgeführt.

Der Programmzähler (**PC**) zeigt nach der Befehlsausführung auf den Opcode des nachfolgenden Befehls.

Programmspeicher

Adressen (hex)	Daten (hex)	
0C78	...	} mov 6Eh,90h
0C77	... (nächster Opcode)	
0C76	6E (Zieladresse)	
0C75	90 (Quelladresse)	
0C74	85 (Opcode)	
0C73	...	
0C72	...	