Laufwerke

CP/M verwaltet bis zu 16 Laufwerke, die mit den Buchstaben A…P angesprochen werden. Die Laufwerksbuchstaben müssen nicht in alphabetischer Reihenfolge vorliegen; sie können beliebig vergeben werden. So gibt es häufiger das Laufwerk M: für eine RAM-Floppy.

Ein Laufwerk A: sollte aber immer vorhanden sein.

Sektoren

Aus CP/M-Sicht besteht die Diskette aus Spuren 0..t-1 und einzelnen logischen Sektoren (Records a 128 Byte) 0..SPT-1.

Das BDOS berechnet aus Block- und Recordnummer des FCB zuerst eine absolute logische Sektornummer der Diskette. Mit der Anzahl von logischen Sektoren pro Spur (SPT, siehe DPB) wird daraus die Spur- und logische Recordnummer innerhalb einer Spur bestimmt.

Aus physikalischer Sicht besteht eine Diskette aus Seiten, Spuren, physikalischen Sektoren.

Bei 8-Zoll-Disketten ist das Format fest vorgegeben: die physikalischen Sektoren sind 128 Byte groß, die Sektornummern laufen von 1..26.

Auf Mini- und Microdisketten sind die physikalischen Sektoren größer als 128 Byte. Häufig sind sie 512 Byte oder 1KByte groß. Physikalischen Sektoren zählen meist von 1..n. Die Sektornummern können beim Formatieren frei vergeben werden, sie wird auf der Diskette am Anfang eines jeden Sektors gespeichert. Die Sektoren können also gleich mit Versatz auf der Diskette angelegt werden, so dass durch die Pause, die beim Verarbeiten der gelesenen Daten besteht und in dieser Zeit sich die Diskette auch weitergedreht hat, dann die hochgezählte nächste Sektornummer unter dem Lesekopf des Laufwerks angekommen ist und sofort gelesen werden kann. (Sektorversatz beim Formatieren; im BIOS muss kein Sektorversatz beachtet werden).

Interleave/Skew

Bei 8-Zoll-Disketten ist dagegen ein Software-Sektorversatz üblich:

(orig. J. Plate, von mir korrigiert)

Beim Lesen von Diskette oder beim Schreiben auf Diskette tritt ein weiteres Problem auf. Angenommen, Sie wollen ein Programm mit 2 KByte Länge abspeichern, dann belegt das Programm 16 Sektoren. Das Betriebssystem ist aber nicht in der Lage, die 16 Sektoren in einem Zuge hintereinander zu lesen oder zu schreiben. Es ist ja allerhand zu berechnen und zu prüfen und bis z.B. Sektor 1 gelesen und geprüft ist, hat sich die Diskette weitergedreht und es befindet sich schon Sektor 3 oder Sektor 4 unter dem Lesekopf. Es muss also für jeden eine ganze Umdrehung der Platte abgewartet werden, was die Zugriffszeit drastisch anhebt.

Um diesen Nachteil zu vermeiden, kann ein Trick angewendet werden: Die Sektoren werden nicht

mehr hintereinander beschrieben oder gelesen, sondern es werden andere dazwischengeschoben.

Beispiel:
Alte Anordnung:
12 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18
Neue Anordnung:
1 10 6 15 2 11 7 16 3 12 8 17 4 13 9 18 5 14

Nun hat sich der Abstand zwischen den Sektoren erhöht, und damit kann innerhalb einer Umdrehung mehr als nur ein Sektor gelesen werden, wenn die Rechenzeit kleiner als der Zeitabstand zwischen den jetzt logisch aufeinanderfolgenden Sektoren ist. Es gibt nun mehrere Möglichkeiten, eine solche Anordnung zu erreichen. Zum einen könnte mit der verwendeten Floppy durch spezielle Formatierung eine solche Numerierung erreicht werden. In CP/M aber gibt es im BIOS eine Übersetzungstabelle, die jedem ankommenden logischen Sektor des CP/M-Systems einen physikalischen Sektor zuweist, wie ihn das Floppy-System versteht. Eine solche Tabelle könnte wie folgt aussehen:

1 5 9 13 17 3 7 11 15 2 6 10 14 18 4 8 12 16

Nehmen wir einmal an, wir wollten zunächst Sektor 0 schreiben oder lesen (logischer Sektor). Dann nehmen wir den ersten Eintrag unserer Tabelle und erhalten den Sektor 1 als physikalischen Sektor. Nun wollen wir den nächsten logischen Sektor schreiben oder lesen, also Sektor 1. In der Tabelle steht an der zweiten Stelle der Wert 5. Wir greifen damit also in Wirklichkeit auf den Sektor 5 (selbe Spur) auf der Diskette zu. Zwischen Sektor 1 und Sektor 5 liegt der Abstand 3 Sektoren, es bleibt also Rechenzeit zwischen den Sektoren 0 und 1 (logische Sektoren), da wir in Wirklichkeit die Sektoren 1 und 5 verwendet haben. Mit den anderen Sektoren verhält sich dies ganz analog. Diese Tabelle wird auch Sektorsprungtabelle genannt. Der Abstand der Sektoren ist der sogenannte Interleaving-Faktor.

In unserem Beispiel war es der Wert 4. Aus dem Wert 4 läßt sich die Tabelle eindeutig aufbauen. Dazu wird bei Sektor 1 begonnen. Dann wird 4 addiert, und es ergibt sich als zweiter Eingang der Wert 5; dann nochmals, und es ergibt sich 9, dann 13, dann 17 - und was nun? 21 gibt es nicht, also minus 18 rechnen, damit ergibt sich Jetzt geht es weiter mit 7, 11, 15 und dann 19, 19 - 18 ergibt 1, aber den Sektor 1 gab es schon in der Tabelle. Nun wird nach dem nächsten nicht in der Tabelle schon vorhandenen Wert gesucht und es ergibt sich der Sektor 1. Dann wird wieder fortgefahren, bis schließlich alle Sektor-Zuordnungen ermittelt sind. Zur Konstruktion dieser Tabelle (wie auch der anderen Tabellen) gibt es zu CP/M die Macrobibliothek DISKDEF.LIB, siehe dazu diskdeflib.

Der BIOS-Funktion SECTRAN werden logische Recordnummer (innerhalb einer Spur) und die Adresse der Sektorversatztabelle übergeben (s.u.). Die BIOS-Funktion muss daraus den physikalischen Sektor berechnen:

aus A Skeletal CBIOS:

```
                    SECTRAN:
                        ;TRANSLATE THE SECTOR GIVEN BY BC USING THE
                        ;TRANSLATE TABLE GIVEN BY DE
   4BA7 EB              XCHG            ;HL=.TRANS
   4BA8 09              DAD    B        ;HL=.TRANS (SECTOR)
   4BA9 6E              MOV    L, M        ;L=TRANS (SECTOR)
   4BAA 2600            MVI    H, 0        ;HL=TRANS (SECTOR)
   4BAC C9              RET            ;WITH VALUE IN HL
```

Die Macrobibliothek DISKDEF.LIB erzeugt bei weniger als 256 Sektoren/Track eine Byte-Liste, bei mehr als 256 Sektoren würde eine Liste mit 16-Bit-Werten erzeugt. Die Bios-Funktion SECTRAN muss dann auch darauf ausgelegt sein.

Überschrift

Die Sektor-Verschränkungs-Tabelle (XLT) dient zur Umrechnung von logischen zu physikalischen Sektornummern einer Spur. Die Länge dieser Tabelle entspricht der Anzahl logischer Sektoren die im DPB (s.u.) definiert sind.

Das BDOS berechnet aus Block- und Recordnummer des FCB eine absolute logische Sektornummer der Diskette.

Mit der Anzahl von logischen Sektoren pro Spur (siehe DPB) kann daraus die Spur- und Sektornummer bestimmt werden.

BDOS22

```
SEEK:
        ...
        CALL    SETTRKF    ;track set up
        ...
        LD    HL,(TRANV)   ; XLT aus DBH
        EX    DE,HL    ;BC=sector#, DE=.tran
        CALL    SECTRAN    ;HL = tran(sector)
        LD    C,L
        LD    B,H    ;BC = tran(sector)
        JP    SETSECF    ;sector selected
```

From:
https://hc-ddr.hucki.net/wiki/ - **Homecomputer DDR**

Permanent link:
https://hc-ddr.hucki.net/wiki/doku.php/cpm/write_a_bios/disketten?rev=1745479215

Last update: **2025/04/24 07:20**