

Reader-Kanal. Dieser Kanal kann mit Funktion 3 angesprochen werden. Beim mc-CP/M-Computer ist dies der Eingabekanal der SIO B. Im Monitor, vor dem CP/M-Start, muß der Kanal mit AR = P auch logisch zugewiesen worden sein, sonst werden die Zeichen fälschlich von der Konsole erwartet.

Ein ankommendes Zeichen gelangt in das Register A. Es wird solange gewartet, bis wirklich ein Zeichen angekommen ist.

Beispiel: Warten auf das Zeichen „A“

```
WARTE: MVI C,3
        CALL 5
        CPI 'A'
        JNZ WARTE
```

Bei der Eingabe vom Reader wird kein Echo erzeugt.

## Funktion 4: Punch-Ausgabe

Damit kann der Punch-Kanal angesprochen werden. Beim mc-CP/M-Computer ist dies die Ausgabe auf SIO B, wenn im Monitor vor dem Start von CP/M der Befehl AP = P gegeben wurde, um den Kanal auch logisch zuzuordnen. Das Zeichen muß an BDOS in Register E übergeben werden. Die Ausgabe erfolgt transparent, das heißt, es werden die Zeichen genauso übertragen wie sie im Register E stehen (die Konsol-Ausgabe mit der Funktion 2 z. B. ist dagegen nicht transparent, denn das Zeichen CTRL-I (9) wird als Sequenz von Leerzeichen ausgegeben).

Beispiel: Ausgabe des Zeichens „B“.

```
MVI C,4
MVI E,'B'
CALL 5
```

## Funktion 5: Drucker-Ausgabe

Mit Funktion 5 kann ein Zeichen, das in Register E steht, an den Drucker gegeben werden. Beim mc-CP/M-Computer ist dies der Kanal SIO B, wenn zuvor im Monitor AL = L eingestellt wurde. Die Ausgabe ist ebenfalls transparent.

## Funktion 6: Direkte Konsolen-Ein-/Ausgabe

In vielen früheren Anwenderprogrammen für CP/M finden sich seltsame Konstruktionen, die versuchen die Anfangsadresse des BIOS zu finden, um dann

Code zu modifizieren, um so direkt irgendwohin zu gelangen. Dadurch sollte die Ein-/Ausgabe des BIOS verwertbar gemacht werden, da dort die I/O-Geräte direkt zugänglich sind. Ab CP/M, Version 2.0, gibt es die Funktion 6, mit der die Konsol-Vektoren direkt angesprungen werden können. Dazu erhält das Register E einen Parameter. Ist der Wert des Registers mit E = FF eingestellt, so wird eine Eingabe verlangt, ist der Wert in E ungleich FF, so soll eine Ausgabe dieses Wertes in E erfolgen. Bei einer Eingabe erscheint nach dem Aufruf im Register A entweder das Zeichen oder, falls keines da war, der Code 00. Damit wartet also die Routine nicht so lange, bis ein Zeichen eingegeben wurde, sondern dies muß man, wenn gewünscht, als Anwender selbst im Programm so einrichten.

Beispiel: Ausgabe eines Zeichens.

```
MVI C,6
MVI E,'Z'
CALL 5
```

Beispiel: Eingabe eines Zeichens.

```
WARTE: MVI C,6
        MVI E,OFFH
        CALL 5
        CPI 0
        JZ WARTE
; In A steht nun das Zeichen
```

Die Routine ist wegen der Parameterkonvention leider nicht ganz transparent. So kann das Zeichen FF nicht ausgegeben werden und das Zeichen 00 darf nicht als Eingabewert vorkommen.

## Funktion 7: I/O-Byte holen

Bei CP/M ist auf Adresse 3 ein I/O-Byte definiert, das vom BIOS ausgewertet werden kann. Mit dieser Funktion wird der Wert dieser Speicherzelle in das Register A geholt.

Im mc-CP/M-Computer wird dieses Byte im BIOS nicht ausgewertet, da der Monitor selbst über ein solches Byte verfügt.

Beispiel: Holen des I/O-Bytes

```
MVI C,7
CALL 5
; in Register A steht der Wert
```

## Funktion 8: Setzen des I/O-Bytes

Mit dieser Funktion wird der Inhalt der Zelle 3 verändert. Der neue Wert muß in Register E stehen. Mancher Leser wird

sich fragen, warum es hier nicht einfacher ist, den Wert direkt in die Speicherzelle zu schreiben oder, wie im vorherigen Fall, einfach daraus zu lesen. Dies hängt mit der Philosophie von CP/M zusammen. Der BDOS-Aufruf ist eine klare Schnittstelle. Es gibt nämlich auch CP/M-Systeme, bei denen die TPA nicht bei 100 H anfängt. Der BDOS-Vektor und auch das I/O-Byte liegen dann nicht an den normalen Stellen. Programme die direkt Speicherzellen modifizieren, können dann nur mit Vorsicht an die neue Umgebung angepaßt werden. Ein BDOS-Aufruf dagegen läßt sich leicht umstellen, da nur eine einzige Adresse (neben der Programmstartadresse) geändert werden muß.

Beispiel: Setzen auf 0.

```
MVI C,8
MVI E,0
CALL 5
```

Beim mc-CP/M-Computer wird das I/O-Byte nicht ausgewertet. Ein Programm, das dies verlangt, läuft also auf dem mc-CP/M-Computer nicht wie vorgesehen; dies ließe sich aber leicht ändern.

## Funktion 9: Ausgabe eines Strings

Mit Funktion 9 kann eine Zeichenkette auf der Konsole ausgegeben werden. Die Anfangs-Adresse der Zeichenkette wird im Registerpaar DE übergeben. Das Ende des Strings wird durch das Zeichen „\$“ angezeigt. Tabulatoren werden expandiert.

Beispiel: Ausgabe eines Textes.

```
MVI C,9          25 03
LXI D,TEXT      11 00 11
CALL 5          00 00 00
...
```

TEXT: DB 'MC-Computer\$'

Auch Zeichen wie CR, LF dürfen im Text vorkommen. Nur das Zeichen „\$“ bestimmt das Ende der Sequenz.

## Funktion 10: Eingabe des Konsolen-Puffers

Sehr komfortabel ist die Funktion 10. Damit ist die Eingabe einer ganzen Textzeile möglich. Im Registerpaar DE wird die Startadresse des Textpuffers angegeben. Das erste Byte in diesem Puffer gibt an, wieviele Zeichen maximal eingelesen werden dürfen. Der Bereich liegt zwischen 1 und 255 (der Wert 0 bedeutet „kein Zeichen“ und ist sinnlos). Nach