

**Name** **CHKOUT**  
**Zweck** Kanal zum Ausgang definieren  
**Adresse** \$FFC9 dez. 65481  
**Vorbereitungen** OPEN  
 log. Filenummer ins X-Register  
 0,3,5,7  
**Fehler** 4  
**Stapelbedarf**  
**Register** Akku, X-Register

Mit dieser Routine kann jedes File, das zuvor durch OPEN spezifiziert worden ist, zum Ausgabe-File erklärt werden. Natürlich muß dann das derart angesprochene Gerät auch ein Ausgabe-gerät sein. Andernfalls ergibt sich ein Fehler. Bevor man Daten über einen Kanal senden will, muß CHKOUT durchgeführt werden. Wenn die mittels OPEN übergebene Geräteadresse größer als 3 ist, sendet diese Routine automatisch auch ein LISTEN-Kommando an das Ausgabe-gerät. LISTEN setzt dann zum Beispiel den Drucker in Empfangsbereitschaft. Die Durchführung von CHKOUT ist einfach (vorausgesetzt, man hat vorher OPEN aufgerufen): In das X-Register wird die logische Filenummer geschrieben und dann per JSR FFC3 CHKOUT angesteuert.

Nun zur anderen Vorbereitung von FFD2, zu OPEN:

**Name** **OPEN**  
**Zweck** Öffnen eines logischen Files  
**Adresse** \$FFC0 dez. 65472  
**Vorbereitungen** SETLFS,SETNAM  
**Fehler** 1,2,4,5,6  
**Register** Akku, X- und Y-Register

Die Routine OPEN an sich anzusprechen, ist relativ einfach. Es genügt ein JSR FFC0. Zuvor allerdings — der Rattenschwanz wird länger — muß mit SETNAM der Filename und mit SETLFS die logische Filenummer, die Geräteadresse und eventuell eine Sekundäradresse festgelegt sein. Erst danach kann das File geöffnet werden durch OPEN. Also sehen wir uns noch SETLFS und SETNAM an:

**Name** **SETLFS**  
**Zweck** Spezifikationen eines logischen Files  
**Adresse** \$FFBA dez. 65466  
**Vorbereitungen** logische Filenummer in Akku  
 Gerätnummer ins X-Register  
 Sekundäradresse ins Y-Register  
**Fehler** keine  
**Stapelbedarf**  
**Register** keine

SETLFS legt für die anderen Kern-Routinen logische Filenummer, Gerätnummer und Sekundäradresse fest. Die logische Filenummer ist dabei eine Schlüsselzahl, die in eine durch OPEN angelegte File-Tabelle weist. Die Gerätnummer kann zwischen 0 und 31 liegen, dabei sind folgende Zuordnungen vorgesehen:

- 0 Tastatur
- 1 Datensette
- 2 RS232C-Kanal
- 3 Bildschirm

Gerätenummern ab 4 beziehen sich automatisch auf Geräte am seriellen Bus. Dabei gilt im allgemeinen:

- 4 Drucker
- 8 Diskettenstation

Die Sekundäradresse ist eine Kommandonummer, die für das jeweils angesprochene Gerät spezifisch ist, zum Beispiel 10 bewirkt beim Drucker Commodore 1526, daß das Gerät in die Grundstellung geht (siehe jeweiliges Handbuch). Will man keine Sekundäradresse verwenden, dann muß FF ins Y-Register ge-

**Stapelbedarf** 2  
**Register** Akku, X- und Y-Register

Vor der Eröffnung eines Files mittels OPEN muß diese Routine den Filenamen festlegen. Dazu schreibt man in den Akku die Länge des Namens und in die Register X, Y die Startadresse (LSB ins X-Register, MSB ins Y-Register) der Namenstabelle. Der Ort dieser Tabelle ist frei wählbar. Wird kein Filename gewünscht, dann gibt man dem Akku die Länge 0 an. X- und Y-Register sind in dem Fall ohne Bedeutung.

Damit — sollte man meinen — hätten wir nun alle Bedingungen erfüllt, FFD2 zur Ausgabe auf den Drucker zu bewegen. Leider ist das noch nicht der Fall: FFD2 schließt nämlich das File und den Ausgabekanal nicht. Das kann — wenn man's nicht beachtet — zu Fehlern oder zur weiteren Ansprache des Druckers führen, auch wenn die gar nicht mehr erwünscht ist. Deswegen sollten noch zwei Kern-

Die Wirkung ist enorm: Mit einem Schlag werden alle Kanäle freigeräumt. Eingangskanälen wird ein UNTALK (dem Gerät wird gesagt: Halt den Mund), Ausgabekanälen ein UNLISTEN (das bedeutet soviel wie: Hör nicht mehr zu) übermittelt. Der Ausgangszustand stellt sich wieder her: Tastatur als Eingabe-Bildschirm als Ausgabe-gerät.

Die endgültig letzte Routine für diesmal ist CLOSE:

**Name** **CLOSE**  
**Zweck** Schließen logischer Files  
**Adresse** \$FFC3 dez. 65475  
**Vorbereitungen** logische Filenummer in Akku  
**Fehler** 0  
**Stapelbedarf** 2  
**Register** Akku, X- und Y-Register

Wenn für ein File alle Ein- und Ausgabeoperationen beendet sind, kann es — nach Einschreiben der Filenummer in den Ak-

Nummer	Text	Bedeutung
0	BREAK	Während des Programms wurde die RUN/STOP-Taste gedrückt
1	TOO MANY FILES	Man kann maximal 10 offene Files einrichten
2	FILE OPEN	Ein bereits geöffnetes File wird nochmals geöffnet
3	FILE NOT OPEN	Auf ein noch nicht geöffnetes File sollte zugegriffen werden
4	FILE NOT FOUND	Das geforderte File ist nicht verfügbar
5	DEVICE NOT PRESENT	Das angesprochene Gerät zeigt keine Reaktion
6	NOT INPUT FILE	Aus einem Schreibfile kann nicht gelesen werden
7	NOT OUTPUT FILE	In ein Lesefile kann nicht geschrieben werden
8	MISSING FILE NAME	Bei Operationen, die einen Filenamen erfordern, fehlt dieser
9	ILLEGAL DEVICE NUMBER	Das versuchte Kommando ist beim angesprochenen Gerät nicht möglich

Tabelle 3. Fehlernummern und ihre Bedeutung Die Nummern findet man bei gesetztem Carry im Akku

schrieben werden. Der Aufruf von SETLFS geschieht also in folgender Weise: In den Akku lädt man die gewünschte logische Filenummer, ins X-Register die Geräteadresse und ins Y-Register FF oder aber die Sekundäradresse. Danach erfolgt der Sprung mit JSR FFBA.

Schließlich noch zu SETNAM:

**Name** **SETNAM**  
**Zweck** Filenamen festlegen  
**Adresse** FFBD dez. 65469  
**Vorbereitungen** Namenslänge in den Akku  
 LSB des Namens-textes in X-Register  
 MSB des Namens-textes in Y-Register  
**Fehler** keine

Routinen angehängt werden, von denen die eine (CLRCHN) alle Ein- und Ausgabekanäle wieder in den Ausgangszustand zurückführt, und die andere (CLOSE) das File ordnungsgemäß schließt:

**Name** **CLRCHN**  
**Zweck** Ein- und Ausgabekanäle in Ausgangszustand bringen  
**Adresse** \$FFCC dez. 65484  
**Vorbereitungen** keine  
**Fehler** keine  
**Stapelbedarf** 9  
**Register** Akku, X-Register

Der Aufruf von CLRCHN erfolgt einfach durch JSR FFCC.

ku — mittels CLOSE ordnungsgemäß geschlossen werden. Der Eintrag in der Filetabelle wird auf diese Weise gelöscht.

So, jetzt sind wir soweit, daß wir die Textausgabe auf dem Drucker programmieren können. Bild 3 faßt die einzelnen Schritte nochmal zusammen.

Und hier das Programm dazu. Wir verwenden die im anderen Beispiel schon aufgebaute Text-tabelle weiter. Zunächst also SETNAM:

```
601C LDA #00
601E JSR FFBD
6021 BCS 6053
```

Wenn ein Fehler aufgetreten ist, findet man ein gesetztes Carry-Bit. In dem Fall wird verzweigt zu einem BRK-Kommando (was die Anwesenheit eines Mo-