MZ-80B - intern Nr. 8

Die Tastatur des MZ-80B wird, wie alle Ein/Ausgabeteile, durch Software gesteuert. Für die Tastatur ist die Z-80A-PIO verantwortlich. Die Tasten der Tastatur sind in einer Matrix angeordnet, die aus 12 Reihen und 8 Spalten besteht. An fast allen Kreuzungen von Spalten und Reihen befindet sich eine Taste, die diese Leitungen verbindet oder trennt. Über die Z-80A-PIO und einen Decoder werden nun vom Monitor die Reiheleitungen nacheinander auf Low-Pegel gelegt und die Spaltenleitungen eingelesen. Damit wird festgestellt, welche Tasten gedrückt wurden. Aus deren Spalten- und Reihenposition wird ihr ASCII-Code ermittelt. Dazu wird eine Tabelle im Monitor, die der in der Bedienungsanleitung auf Seite 54 abgedruckten entspricht, verwendet. Im Monitor SB-1510 liegt diese Tabelle zwischen $0D48 und $0DF7, im Monitor SB-1511 zwischen $0FD7 und $1052. Im Monitorlisting sind die einzelnen Reihen mit S0 bis S10 gekennzeichnet.

Da die Tasten aber bis zu drei verschiedene Zeichen liefern, wird noch eine weitere Unterteilung in Gruppen vorgenommen. Gruppe 1 enthält die Tasten ohne Umschaltfunktion, wie Cursor- und Funktionstasten, Zehnerblock, Leertaste und BREAK-, TAB- und CR-Taste. Gruppe 2 enthält die Tasten mit SHIFT-Funktion, dies sind alle Zifferntasten in der oberen Tastenreihe, <=>, <:>, <;>, <ß>, <Ä>, <Ü>, <HOME> und <DEL>. Zur dritten Gruppe schließlich gehören die mit SHIFT- und GRPH-Funktion, also alle Tasten mit weißen Grafikaufdruck sowie ausnahmsweise <^> und <ä>. Neben diesen drei Gruppen gibt es noch eine Spezialgruppe für die Umschalttasten SHIFT, SFTLOCK, GRPH und RVS. Im Monitorlisting ist erst Gruppe 1 (KTBL/KTBLS), dann Gruppe 2 für die SHIFT-Zeichen (SMALL/KTBLG) und zum Schluß Gruppe 3 für die GRPH-Zeichen (GRAPH) abgedruckt. Zur Erzeugung der Negativzeichen ist keine Tabelle vorhanden, sie werden erzeugt, indem zum ASCII-Code des normalen Zeichen 128 (=$80) addiert wird. Siehe auch Bild 1.

Man kann nun die Zeichen, die bestimmte Tasten liefern, ganz einfach ändern, indem man in die zu der Taste gehörenden Speicherplätze der Tabelle andere ASCII-Codes schreibt. So kann man mit den in Bild 2 gezeigten POKE-Befehlen Y und Z vertauschen, um einer deutschen Tastatur etwas näher zu kommen.

Nun zum CHANGE-Befehl. Dieser Befehl tauscht in der besprochenen Tabelle Klein- und Großbuchstaben gegeneinander aus (A bis Z). Es handelt sich dabei um die ASCII-Codes $41 bis $5A bzw. $61 bis $7A, das sind jeweils 26 (=$1A) Zeichen. Die CHANGE-Routine liegt bei BASIC SB-5510 ab $2BB9 und bei BASIC SB-6510 ab $2D7C. Wer sich diese Routine mit einem Disassembler einmal genauer ansieht, wird feststellen, daß hier eigentlich gar nichts ausgetauscht wird, sondern nur das 6. Bit der ASCII-Codes der Zeichen A-Z gesetzt oder gelöscht wird. Das ist nämlich der einzige Unterschied zwischen den ASCII-Codes von Groß- und Kleinbuchstaben. Wie Bild 3 zeigt, enthält diese Routine die Anfangsadressen der Großbuchstaben und der Kleinbuchstaben und deren Anzahl. Diese Werte lassen sich natürlich ändern. Damit läßt sich aber nicht erreichen, daß die Umlaute in den CHANGE-Befehl einbezogen werden, da diese Zeichen wild verstreut in der Tabelle liegen. Es ist zwar möglich, diese Zeichen auf einen Bereich zu konzentrieren und sie dann in den CHANGE-Befehl einzubinden, dies hat aber den Nachteil, daß auch Tastenkappen vertauscht werden müssen. Darauf sollte man aber verzichten, weil die Änderung des CHANGE-Befehls sich ja nur auf die BASIC-Interpreter auswirkt. Bei PASCAL, CP/M, FDOS, usw. bleibt aber alles beim alten, was zur Folge hätte, das die gleiche Taste bei verschiedenen Programmen verschiedene Zeichen liefert.

Ein anderer Weg scheint sinnvoller. Und zwar werden die Umlaute einfach durch POKE-Befehle in der SHIFT-Funktion umgekehrt, im Klartext, die Zeichen werden in der Tabelle sozusagen von Hand ausgetauscht. Damit bleiben die Umlaute zwar auf den Klammer- und Strichtasten, wo sie laut DIN nicht hingehören, liefern aber auf Wunsch auch ohne <SHIFT> Kleinbuchstaben und mit <SHIFT> Großbuchstaben. Außerdem bleibt die Kompatibilität gewahrt, was wohl sehr wichtig ist. Die entsprechenden POKEs zeigt Bild 4.

Bild 1: Beispiele zur Tastaturdecodierung

Die Taste U liegt auf der Kreuzung der 6. Reihen- und 5. Spaltenleitung (vergl. S. 54 der Bedienungsanleitung). Das Zeichen U (ASCII $55) findet man deshalb im Monitorlisting auf S. 77 bei 56 an 6. Stelle (weil die erste Stelle Spalte 0 entspricht). Das kleine u (ASCII $75) liegt entsprechend bei 568 an 6. Stelle, und das zur U-Taste gehörende Grafikzeichen liegt bei 566 an 6. Stelle.

Bild 2: Tausch von Y und Z

```
POKE $0D81,$5A: POKE $0D82,$59: POKE $0DB9,$7A: POKE $0DBA,$79
```

Alten Zustand wiederherstellen:

```
POKE $0D81,$59: POKE $0D82,$5A: POKE $0DB9,$79: POKE $0DBA,$7A
```

Bild 3: Adressen im CHANGE-Befehl

BASIC SB-5510	BASIC SB-6510	Inhalt
$2BBB/$2BBC	$2D7E/$2D7F	Anfangsadresse des Speicherbereichs, der normalerweise die Großbuchstaben enthält
$2BBE/$2BBF	$2D81/$2D82	Anfangsadresse des Speicherbereichs, der normalerweise die Kleinbuchstaben enthält
$2BCF	$2D92	Länge (= Anzahl der Buchstaben) jedes oben genannten Bereiches

Bild 4: Umkehrung der SHIFT-Funktion für die Umlaute incl. ß

```
POKE $0D96,$7B: POKE $0D84,$7C: POKE $0D98,$7D: POKE $0D83,$7E:
POKE $0DCE,$5B: POKE $0DBC,$5C: POKE $0DD0,$5D: POKE $0DBB,$5E
```

Alten Zustand wiederherstellen:

```
POKE $0D96,$5B: POKE $0D84,$5C: POKE $0D98,$5D: POKE $0D83,$5E:
POKE $0DCE,$7B: POKE $0DBC,$7C: POKE $0DD0,$7D: POKE $0DBB,$7E
```

* * *