

Erfahrungsbericht – Programmierbare Taschenrechner

Dirk Stieglitz, Axel Kloth

Haben programmierbare Taschenrechner Fehler?

Bei Durchsicht der einschlägigen Literatur zu den programmierbaren Taschenrechnern fällt die Häufigkeit der Stellungnahmen zu den Rechnern TI-58/59 zunehmend auf. Eine intensive Auseinandersetzung mit diesen Veröffentlichungen und eigene Programmiererfahrung lassen es sinnvoll erscheinen, offensichtliche Rechenschwächen, aber auch die Vorteile dieser beiden Rechner zusammenfassend und ergänzend darzustellen. Wir fanden heraus, daß die Rechner noch einige Funktionen aufweisen, die sicherlich nicht zu den in der Bedienungsanleitung beschriebenen Funktionen gehören.

Da die im folgenden detailliert beschriebenen Rechneroperationen nicht nur zu uneinsichtigen und undefinierbaren „Rechenergebnissen“, sondern im Extremfall sogar zu einem unbeabsichtigten Stromverbrauch führten, erscheint es uns angebracht, ausführlich unsere bisherigen Ergebnisse darzustellen.

Der Ausgangspunkt für alle Überlegungen waren die HIRE-Befehle. Mancher Anwender der Rechner mag beim Programmieren auf sie gestoßen sein. Für den, der sie noch nicht bemerkt hat, sind sie hier noch einmal aufgeführt: 21 und 26 2nd, 31 LRN, 41 SST, 46 Ins, 51 BST, 56 Del und 82 HIR.

Der letzte HIRE-Befehl, nämlich 82 HIR, ist der am meisten verwendete Befehl dieser Art. Wie bekommt man diesen Befehl nun in den Programmspeicher? Man betätigt einen Kombi-Befehl, damit der Rechner die folgende zweistellige Zahl als einen einzigen Befehl anerkennt. Nun gibt man 82 ein, geht mit BST zum Kombi-Befehl zurück und löscht diesen mit Del, so daß 82 HIR im Programmspeicher übrigbleibt. Die Anweisung ist nach dem gleichen Muster einzugeben. Sogar ein internes Programm beinhaltet diesen Befehl.

Wir wollen diesen Befehl etwas genauer untersuchen. Er erschließt 8 weitere Speicher mit voller Speicherarithmetik. Diese Befehle sind aber nur im Programm verwendbar und nicht über die Tastatur zu adressieren. Um diesen Befehl benutzen zu können, benötigt man den Befehl 82 HIR und eine zweite Anweisung, die dem Rechner mitteilt, welche Operation in welchem Register auszuführen ist. Dieser zweite Befehl ergibt eine zweistellige Zahl, wie z.B. bei der üblichen Speicheradressierung, XY.

Die folgende Tabelle erklärt dies:

X	Y	Funktion
0	1-8	STO
1	1-8	RCL
2	1-8	Nop
3	1-8	SUM
4	1-8	Prd
5	1-8	INV SUM
6-9	1-8	INV Prd

in Stackregister 1-8

Da der Rechner in diesen Registern die Zwischenergebnisse speichert, muß beachtet werden, daß keine Zahlen überschrieben werden. Diese Register werden Stackregister genannt und sind bei einigen anderen Rechnerarten direkt abrufbar.

Unbekannt, mit einigen Ausnahmen, ist bisher die Wirkung von X0 und X9 für alle X = 0 bis 9. Bekannt ist die Codekombination 82 96, sie bewirkt, daß z.B. die Zahl 0,05 in 500 umgewandelt wird, es gilt also allgemein $a \cdot 10^{-x}$ wird zu $a \cdot 10^x$ [1].

Da man bisher nur zweistellige Adressen XY benutzen kann, stellt sich die Frage, ob nicht auch dreistellige Befehle XYZ möglich sind. Dies ist leider nicht der Fall. Es wird als

82 HIR OX YZ

anerkannt, da es nicht möglich ist, eine dreistellige Zahl in einem Programmspeicherschnitt zu speichern.

Diese Codes veranlassen den Rechner, die vorhergehende Zahl in das Register X abzuspeichern und dann den Befehl YZ auszuführen. Der Befehl YZ entspricht dann einem Code einer Taste auf dem Tastenfeld.

Nachstehend ein kleines Programm zur Überprüfung der Wirksamkeit der HIRE-Befehle. Dieses Programm kann benutzt werden, wenn man wissen möchte, welche Zahlen in den Stackregistern gespeichert sind [2].

```
000 76 LBL
001 11 A
002 82 HIR
003 11 11
004 99 PRT
005 82 HIR
006 12 12
007 99 PRT
008 82 HIR
009 13 13
010 99 PRT
011 82 HIR
012 14 14
013 99 PRT
014 82 HIR
015 15 15
016 99 PRT
017 82 HIR
018 16 16
019 99 PRT
020 82 HIR
021 17 17
022 99 PRT
023 82 HIR
024 18 18
025 99 PRT
026 98 ADV
027 91 R/S
```

Bei Anwendung ohne Drucker
können die PRT-Befehle durch
PAUSE ersetzt werden

Eine sehr nützliche Weiterentwicklung davon ist das nächste Programm zum Redigieren falscher Eingaben, z. B. in der Statistik [3].

```
000 76 LBL
001 11 A
002 01 1
003 82 HIR
004 57 57
005 82 HIR
006 17 17
007 82 HIR
008 78 78
009 32 x  $\frac{1}{t}$ 
010 82 HIR
011 18 18
012 22 INV
013 78  $\Sigma +$ 
014 91 R/S
```

Dieses Programm hat die Aufgabe, eine Fehleingabe ohne Probleme rückgängig zu machen. (Vorsicht jedoch bei $x = 0!$)

Die Tastenfolge OP 16, STO 00, Pgm 01, A, Pgm 0, + hat auf die Rechner verheerende Folgen [4]. Denn wie eingangs erwähnt, verbrauchen die Rechner Energie, obwohl die Anzeige total verschwindet und der Rechner auf keinerlei Tastendrücke mehr reagiert.

Es sind sicherlich noch andere Funktionen vorhanden, die den Rechner „überfordern“, denn außer den HIRE-Befehlen lassen sich noch andere, hier nicht aufgeführte Operationen, kaum nützlich in einem Programm verwenden. Aber warum sollen sie nicht doch Anwendung in einem Programm finden?

Schade ist auch, daß die Freude an diesem kleinen Wunderwerk der Elektronik durch solche Fehler wie mechanische Anfälligkeit der Bauteile geschmälert wird. Die Auskünfte eines Händlers ließen erkennen, daß ca. 58% der Rechner [5] während der Garantiezeit unter anderem wegen Prellen der Tasten zurück gesandt werden mußten.

Außerdem seien die Magnetkarten nicht kompatibel, was uns allerdings nicht aufgefallen ist, von TI aber bestätigt wurde [6].

Abschließend kann wohl gesagt werden, daß die Rechner, wenn sich ihre Mängel beseitigen ließen, überaus intelligente und fortschrittliche Geräte sind. Die Fehler machen es einem allerdings in manchen Situationen sehr schwer, korrekt und logisch einwandfrei zu programmieren.

Literatur

- [1] Chip (Vogel-Verlag, Würzburg) 9/79, Leserbrief.
- [2] Gespro (Gespro GmbH, Koblenz) 2/79.
- [3] PPX (Gespro GmbH, Koblenz) 4/79.
- [4] Chip 3/80, Leserbrief.
- [5] Chip 12/79, Leserbrief.
- [6] Funkschau (Franzis-Verlag, München) 12/79.