Michael Konz

6502-Rechenroutinen

Will man einem Mikrocomputer das Rechnen beibringen, so stellen die vier Grundrechenarten bekanntlich die größten Probleme dar. Die folgenden Routinen befassen sich daher auch nur mit Addieren und Subtrahieren, Multiplizieren und Dividieren.

Das nachfolgend beschriebene Programm wurde für den AIM-65 geschrieben, der gewissermaßen als Nachfolger des KIM bezeichnet werden kann. Natürlich laufen die Routinen auch auf diesem, wenn geringfügige Änderungen, auf die später noch eingegangen werden wird, vorgenommen werden. Eine alphanumerische Ein- und Ausgabemöglichkeit wäre jedoch nützlich, wenn auch nicht unbedingt erforderlich.

Die Genauigkeit der Routinen läßt sich durchaus mit der guter Taschenrechner vergleichen. In den meisten Fällen liegt sie sogar höher. Es können Zahlen im Bereich von $1 \cdot 10^{-99}$ bis $9{,}999999999 \cdot 10^{99}$ verarbeitet werden. Intern wird mit einer zwölfstelligen Mantisse gearbeitet. Diese Stellen können bei entsprechender Parametereingabe auch alle angezeigt werden. Dennoch ist solch eine Programmierung nicht zu empfehlen, will man nicht auf Schutzstellen verzichten.

Bevor die einzelnen Routinen nun besprochen werden, soll eine Auflistung aller dem Anwender zur Verfügung stehenden Unterprogramme gegeben werden.

0200
Der Register-Pointer wird abhängig vom Inhalt der beiden Indexregister gesetzt.
0233
Ein Register wird in ein anderes kopiert. Angabe von Ursprungs- und Zielregister erfolgt durch die beiden Indexregister der CPU.
0240
Das x-Register wird in das y-Register kopiert.
0246
Das y-Register wird in das x-Register kopiert.
0259
Das x-Register wird mit dem y-Register vertauscht.
0269
Das durch das Indexregister X angegebene Register wird gelöscht.
027C
Exponent und Vorzeichen des durch den Index X angegebenen Registers werden gelöscht.
028D
Der Exponent im x-Register wird um 1 erhöht. Tritt ein Überlauf auf, so erfolgt eine Fehlermeldung und Rückkehr zum Monitor.
02AB
Der Exponent im x-Register wird um 1 erniedrigt; sonst wie 028D.
02BB
Die Mantisse im x-Register wird um 1 nach links verschoben, was einer Multiplikation mit 10 gleichkommt. Ein dabei auftretender Übertrag bleibt unberücksichtigt.
02C7
Die Zahl im x-Register wird in die Exponentialform gebracht.
02ED
Die Exponenten der Zahlen in x und y werden gleichgesetzt.
0354
Die Mantisse von x wird um 1 nach rechts verschoben, was einer Division durch zehn gleichkommt. Die letzte Ziffer geht verloren, der Exponent bleibt unverändert.
0369
Wie 0354, jedoch für das y-Register.
037E
Der Inhalt des x-Registers wird ausgegeben.
0437
Die Inhalte des x- und des y-Registers werden verglichen, das Ergebnis des Vergleiches an der Zero-Page-Adresse F2 gespeichert.
044C
Wie 0437, es werden aber nur die absoluten Beträge verglichen.
0464
Wie 0437, es werden jedoch nur die Mantissen verglichen.
04B8
Die Inhalte von x und y werden addiert, das Ergebnis im x-Register gespeichert.
0518
Die Inhalte von x und y werden subtrahiert (x–y), das Ergebnis im x-Register gespeichert.
0523
Eingabe einer Zahl in das x-Register.
05CF
Die Inhalte von x und y werden multipliziert, das Ergebnis im x-Register gespeichert.
069A
Der Inhalt von x wird durch den Inhalt von y dividiert, das Ergebnis in x gespeichert.
0766
Ausgabe des Textes „UNVERSTAENDLICHE EINGABE".
0768
Ausgabe eines beliebigen alphanumerischen Textes.
077A
Ausgabe des Textes „RECHENKAPAZITAET UEBERSCHRITTEN" und Rückkehr zum Monitor.

Um Verwirrungen zu vermeiden, muß zunächst einmal auf die Register eingegangen werden. Die von den Routinen benutzten Register dienen zur Aufnahme der Daten und belelgen je 16 Byte im Speicher. Sie dürfen nicht mit den beiden Indexregistern der CPU verwechselt werden.

In der Zero-Page stehen insgesamt 15 Register zur Verfügung. Davon werden neun von den Routinen belegt, die übrigen kann der Programmierer zur eigenen Datenspeicherung verwenden. Die beiden ersten Register erhielten den Namen x und y. Sie sind an den Adressen 0000-000F und 0010-001F zu finden. Weiterhin kann der Anwender ab einer von ihm bestimmbaren Page beliebig viele zusätzliche Register festlegen. Dies geschieht durch Schreiben der Zero-Page-Adresse F4. Der Wert 0A bezeichnet dann den Speicherraum ab der Adresse 0A00. Dieser Parameter ist vor dem ersten Aufruf einer Transfer-Routine einzugeben. Natürlich muß dabei darauf geachtet werden, daß die Daten nicht das Programm zerstören,

```
0200 8A 20 18 02 85 FB 20 2D 02 85 FA 98 AA 20 18 02
0210 85 FD 20 2D 02 85 FC 60 29 F0 C9 F0 D0 04 A9 00
0220 F0 0A 8A 4A 4A 4A 4A D8 18 65 F4 F8 60 8A 0A 0A
0230 0A 0A 60 20 00 02 A0 0F B1 FA 91 FC 88 10 F9 60
0240 A2 F0 A0 F1 30 ED A2 F1 A0 F0 30 E7 A2 F0 A0 F2
0250 20 33 02 A2 F1 A0 F3 30 DA A2 F1 A0 FD 20 33 02
0260 20 40 02 A2 FD A0 F0 30 CA 20 00 02 A0 0F A9 00
0270 91 FA 88 10 FB A9 01 A0 0D 91 FA 60 20 00 02 A0
0280 0C A9 00 91 FA C8 C8 91 FA C8 91 FA 60 F8 A5 0E
0290 30 0D A5 0F 18 69 01 85 0F 90 03 4C 7A 07 60 A5
02A0 0F 38 E9 01 85 0F D0 F6 85 0E 60 F8 A5 0F D0 05
02B0 E6 0F C6 0E 60 24 0E 30 DB 10 E6 A2 00 B5 01 95
02C0 00 E8 E0 0B 90 F7 60 F8 A0 00 A5 00 F0 0F A5 0D
02D0 C9 01 D0 01 60 C6 0D 20 8D 02 B8 50 F1 20 BB 02
02E0 20 AB 02 C8 C0 0C 90 E2 A2 F0 4C 7C 02 A5 0E C5
02F0 1E F0 1F 24 0E 10 0D A5 0F F0 44 20 54 03 20 8D
0300 02 B8 50 F3 A5 1F D0 04 E6 1E 10 2A 20 62 03 B8
0310 50 F2 A5 0F C5 1F D0 01 60 24 0E 10 17 90 0C 20
0320 4B 03 20 54 03 20 8D 02 B8 50 F4 20 4B 03 20 62
0330 03 B8 50 F7 90 09 20 4B 03 20 77 03 B8 50 F7 20
0340 4B 03 20 54 03 20 8D 02 B8 50 F4 A5 0F C5 1F D0
0350 02 68 68 60 A2 0A B5 00 95 01 CA 10 F9 A9 00 85
0360 00 60 A5 1F 38 E9 01 85 1F A2 0A B5 10 95 11 CA
0370 10 F9 A9 00 85 10 60 A5 1F 18 69 01 90 E9 20 44
0380 EB A9 03 85 F0 A9 00 85 F1 A2 F0 A0 FD 20 33 02
0390 A5 F1 30 23 A5 0F C9 10 B0 1D 24 0E 30 0C A5 0F
03A0 F0 15 E6 0D 20 AB 02 B8 50 F4 A5 0F F0 09 20 54
03B0 03 20 8D 02 B8 50 F3 D8 18 A5 0D 65 F0 F8 C9 0C
03C0 90 02 A9 0B AA 86 F0 B5 00 C9 05 90 24 CA EA 18
03D0 B5 00 69 01 48 0A 0A 0A 0A 68 29 0F 95 00 90 11
03E0 CA 10 EC E6 0D 20 54 03 E6 00 A5 F1 10 03 20 C7
03F0 02 A2 00 A5 0C 10 04 A9 2D 10 02 A9 20 20 7A E9
0400 B5 00 20 51 EA E8 E4 F0 B0 08 E4 0D D0 F2 A9 2E
0410 10 EB A5 0F F0 1A 20 3E E8 A9 45 20 7A E9 A5 0E
0420 10 04 A9 2D 10 02 A9 2B 20 7A E9 A5 0F 20 46 EA
0430 A2 FD A0 F0 4C 33 02 A5 0C C5 1C 90 07 F0 09 A9
0440 80 85 F2 60 A9 40 10 F9 A5 0C 30 37 A5 0E C5 1E
0450 90 F2 F0 02 B0 E9 A5 0E 30 1F A5 0F 30 1B 90 DF
0460 F0 02 B0 E0 A2 00 B5 00 D5 10 90 D3 F0 02 B0 D4
0470 E8 E0 0C 90 F1 A9 00 F0 C8 A5 0F C5 1F 90 C5 F0
0480 E3 B0 BC A5 0E C5 1E 90 B6 F0 02 B0 B7 A5 0E 30
0490 0A A5 0F C5 1F 90 AD F0 0C B0 A4 A5 0F C5 1F 90
04A0 9E F0 02 B0 9F A2 00 B5 00 D5 10 90 97 F0 02 B0
04B0 8E E8 E0 0C 90 F1 B0 BD 20 ED 02 A5 0C C5 1C D0
04C0 21 A2 0B 18 B5 00 75 10 48 0A 0A 0A 0A 68 29 0F
04D0 95 00 CA 10 EF 90 07 20 54 03 E6 0D E6 00 20 C7
04E0 02 60 20 64 04 A5 0C 10 08 24 F2 70 1F E6 0C 10
04F0 06 24 F2 70 17 C6 0C 38 A2 0B B5 10 F5 00 29 0F
0500 95 00 CA 10 F5 B0 D7 20 54 03 F0 D2 38 A2 0B 20
0510 B4 05 EA CA 10 F9 30 ED 20 ED 02 A5 0C C5 1C D0
0520 A0 F0 BF A2 0F A9 00 95 00 CA 10 FB 20 F0 E9 4E
0530 11 A4 A0 00 20 5F E9 C8 C9 0D D0 F8 AE 15 A4 9D
0540 38 A4 0E 11 A4 A2 00 A0 00 B9 38 A4 C9 0D F0 2F
0550 C9 2D D0 05 C6 0C C8 10 F0 C9 2E D0 08 86 0D A5
0560 0D F0 17 D0 F1 C9 45 F0 23 20 7D EA B0 0C C9 0A
0570 B0 08 95 00 E8 C8 E0 0B 90 CF 20 66 07 50 A4 A5
0580 0D D0 02 86 0D 20 C7 02 4C 7E 03 EA C8 B9 38 A4
0590 C9 2D D0 06 C6 0E C8 B9 38 A4 C9 3A B0 DC 20 84
05A0 EA B0 D7 C8 B9 38 A4 C9 3A B0 CF 20 84 EA B0 CA
05B0 85 0F 90 CB B5 00 F5 10 29 0F 95 00 60 A5 FF C6
05C0 FE 30 09 C6 FE 30 07 18 65 FF 90 F7 A9 00 60 20
05D0 4C 02 A9 00 85 FA 20 6C 02 A9 10 85 FA 20 6C 02
05E0 A2 00 D8 A9 0B 86 F8 38 E5 F8 A8 F8 B5 30 F0 50
05F0 85 FF A2 0B B9 20 00 85 FE 20 BD 05 48 29 0F 18
0600 75 10 48 0A 0A 0A 0A 68 29 0F 95 10 68 08 CA 10
0610 12 48 20 69 03 68 4A 4A 4A 4A 28 65 10 85 10 E6
0620 1D 10 0C 4A 4A 4A 4A 28 75 10 95 10 88 10 C5 20
0630 59 02 20 C7 02 20 59 02 20 B8 04 A2 F1 20 69 02
0640 A6 F8 E8 E0 0C 90 9B 20 4D 06 4C C7 02 A5 0F F0
0650 04 C6 0F E6 0D A5 2C C5 3C F0 02 C6 0C A5 2E C5
0660 3E D0 0F 85 0E 18 A5 2F 65 3F 90 03 4C 7A 07 85
0670 0F 60 A5 2F C5 3F 24 2E 10 12 90 08 38 E5 3F 85
0680 0F C6 0E 60 A5 3F 38 E5 2F 85 0F 60 90 06 20 7C
0690 06 E6 0E 60 20 84 06 C6 0E 60 A2 F0 A0 F4 20 33
06A0 02 A2 F1 A0 F5 20 33 02 A2 F1 20 7C 02 A2 F0 20
06B0 69 02 20 37 04 A5 F2 D0 0B A2 FF 9A A2 BE 20 68
06C0 07 4C 82 E1 A2 F4 A0 F0 20 33 02 A2 F0 20 7C 02
06D0 A2 F0 A0 F7 20 33 02 A2 F6 20 69 02 A2 00 86 F7
06E0 86 F9 20 18 05 A5 0C 30 04 E6 F7 10 F5 A5 F7 A6
06F0 F9 95 60 E8 E0 0C B0 3F 86 F9 48 20 46 02 20 C7
0700 02 A2 F1 20 69 02 68 85 10 20 CF 05 20 40 02 A2
0710 F7 A0 F0 20 33 02 20 18 05 A2 F5 A0 F1 20 33 02
0720 A2 F1 20 7C 02 E6 0D 20 C7 02 A2 F0 A0 F7 20 33
0730 02 A2 00 86 F7 F0 AB A2 F6 A0 F0 20 33 02 20 44
0740 07 4C C7 02 A2 F4 A0 F2 20 33 02 A2 F5 A0 F3 20
0750 33 02 A5 2C C5 3C F0 02 C6 0C A5 2E C5 3E D0 03
0760 4C 72 06 4C 63 06 A2 85 20 F0 E9 BD 00 07 C9 02
0770 F0 06 20 7A E9 E8 D0 F3 B8 60 A2 9E 20 68 07 A2
0780 FF 9A 4C 82 E1 55 4E 56 45 52 53 54 41 45 4E 44
0790 4C 49 43 48 45 20 45 49 4E 47 41 42 45 02 52 45
07A0 43 48 45 4E 4B 41 50 41 5A 49 54 41 45 54 20 55
07B0 45 42 45 52 53 43 48 52 49 54 54 45 4E 02 52 45
07C0 53 55 4C 54 41 54 20 55 4E 45 4E 44 4C 49 43 48
07D0 02 03
```

Bild 1. Dieses hexadezimale Listing enthält alle AIM-65-Rechenroutinen

das sich an den Speicherplätzen 0200-07D0 befindet.

Innerhalb der Register sind die Daten folgendermaßen angeordnet:
XXX0-XXXB: 12stellige Mantisse

XXXC: Vorzeichen der Mantisse (00: positiv, FF: negativ)
XXXD: Stellung des Dezimalpunktes
XXXE: Vorzeichen des Exponenten
XXXF: Exponent

Nun zu den einzelnen Routinen. Zum Transfer eines Registers in ein anderes wird die Routine ab der Adresse 0233 verwendet. Das Ursprungsregister wird durch eine hexadezimale Ziffer angegeben, die vor dem Aufruf der Routine im Indexregister X gespeichert werden muß. Das Zielregister wird genauso im Index Y gespeichert. Dabei bezeichnen die Zahlen FX Register in der Zero-Page. F0 ist das x-Register, F1 das y-Register. Alle anderen Register beziehen sich auf die Angabe in F4. Ist dieser Parameter z. B. 0A, dann bezeichnet das Register 00 die 16 Byte an den Adressen 0A00...0A0F; 01 ist bei 0A10...0A1F zu finden usw.

Zu den Registern wäre noch zu sagen, daß das Register FF, das in der Zero-Page an den Adressen FO...FF stehen würde, nicht benutzt werden kann, da die FX-Adressen für den Programmablauf wichtige Parameter enthalten.

Für die Routinen 0269 und 027C gilt das oben Gesagte für die Adressierung der Register. Es wird jedoch nur ein Wert im Index X benötigt.

Die Routine 0200 schließlich dient dazu, den Pointer auf die angegeben Register zu setzten. Dieser Pointer befindet sich an den Adressen FA und FB bzw. FC und FD. Normalerweise wird diese Routine nicht benötigt, da eine Transferroutine ihren Aufruf beinhaltet.

Die Routine 02C7 gehört zu den wichtigsten überhaupt. Sie sorgt dafür, daß alle Daten nach jeder Eingabe und nach jeder Rechenoperation immer in Exponentialform vorliegen:
$1{,}23456789012 \cdot 10^{99}$

Die Routine 02ED ist nicht weniger wichtig. Sie gleicht die Exponenten im x- und y-Register aneinander an. Dazu ein Beispiel:

Im x-Register steht die Zahl $1{,}23 \cdot 10^{08}$,
im y-Register die Zahl $5{,}67 \cdot 10^{05}$.

Nach Aufruf der Routine steht die Zahl im x-Register unverändert, im y-Register steht jedoch $0{,}00567 \cdot 10^{08}$. Es ist sofort zu sehen, daß der Wert derselbe ist wie zuvor.

Zur Ausgabe von Daten stehen zwei Formate zur Verfügung. Wird die Routine 037E aufgerufen, so erfolgt die Anzeige mit Festkomma, das auf drei Stellen gesetzt ist, soll die Anzeige im Ex-

```
0800 A2 FC A0 F0 20 33 02 A2 F9 A0 F1 20 33 02 20 CF
0810 05 A2 F0 A0 FC 20 33 02 A2 F0 20 69 02 A2 FA A0
0820 F1 4C 33 02 20 C7 02 20 59 02 20 CF 05 A2 F0 A0
0830 FA 20 33 02 A9 00 85 AC 20 40 02 A2 FC A0 F0 20
0840 33 02 20 9A 06 20 40 02 A2 F8 A0 F0 20 33 02 20
0850 B8 04 A2 F0 A0 F8 20 33 02 4C 00 08 20 23 05 A2
0860 F0 A0 F8 20 33 02 20 40 02 20 CF 05 A2 F0 A0 F9
0870 20 33 02 A2 FA 20 69 02 E6 A0 A2 F8 A0 FC 20 33
0880 02 A9 00 85 F6 20 00 08 20 99 08 C6 0C 20 24 08
0890 20 99 08 20 24 08 4C 88 08 A6 F6 E0 15 90 0F 68
08A0 68 A2 F8 A0 F0 20 33 02 20 F0 E9 4C 7E 03 A0 00
08B0 BD C3 08 99 00 00 E8 C8 C0 03 90 F4 A9 02 85 0F
08C0 86 F6 60 00 00 06 00 02 00 00 04 02 00 07 02 01
08D0 01 00 01 05 06 02 01 00 20 5C 08 20 5F E9 4C D8
08E0 08
```

Bild 2. Berechnung des Sinus, Startadresse ist 08D8

ponentialformat erfolgen, so müssen zuvor die Zero-Page-Adressen F0 und F1 beschrieben werden. Normalerweise stehen hier die Werte 03 und 00. Für das Exponentialformat muß der Wert von F1 zu FF geändert werden. An F0 muß dann die Zahl der Nachkommastellen geschrieben werden. Die Startadresse lautet dann 0389. In beiden Fällen findet eine automatische 5/4-Rundung statt. Vorsicht: Beim Aufruf der Routine wird das Register FD überschrieben.

Die nächsten drei Routinen sind eigentlich eine einzige mit drei verschiedenen Startadressen. Wie bereits gesagt, dienen sie zum Vergleich des x- und des y-Registers. Das Resultat des Vergleiches hat folgende Form:

Ist x kleiner als y, dann wird Bit 7 von F2 gesetzt.

Ist x größer als y, dann wird Bit 6 von F2 gesetzt.

Ist x gleich y, dann wird F2 mit dem Wert 00 geschrieben.

Das Ergebnis des Vergleiches kann also leicht mit der Instruktion BIT F2 abgefragt werden, gefolgt von BMI, BPL, BVC oder BVS, beziehungsweise von der Befehlsfolge LDA F2; BEQ/BNE.

Die Routine 04B8 ist die erste wirkliche Rechenroutine. 0523 dient zur Übernahme von Daten ins x-Register. Bei der Eingabe ist folgendes zu beachten: Das Vorzeichen ist immer vor der Zahl einzugeben (Im Gegensatz zu den meisten Taschenrechnern). Die Exponenteneingabe wird mit einem E begonnen. Der Exponent muß immer zweistellig eingegeben werden. Die Eingabe –123E-8 würde zu einer Fehlermeldung führen. Richtig muß es heißen: –123E-08.

Eine fehlerhafte Eingabe kann durch Betätigen der DEL-Taste rückgängig gemacht werden, solange nicht RETURN gedrückt wurde. Die zulässige Stellenanzahl kann durch den Wert an der Adresse 0577 bestimmt werden. Eine Stellenzahl von 10 ist vorgesehen. Mehr Ziffern führen nach RETURN zu einer Fehlermeldung. Zu beachten ist, daß nach einer Fehlermeldung die Eingabe automatisch wiederholt wird, bis eine korrekte Eingabe erfolgt.

Sehr nützlich ist die Routine 0768, die dem Anwender erlaubt, eigene Erläuterungen zu Rechenoperationen ausgeben zu lassen. Es stehen hier noch 47 Byte für beliebige ASCII-Zeichen zur Verfügung. Diese werden ab der Adresse 07D1 bis maximal 07FF gespeichert. Jeder Text wird mit CNTRL B (Code 02) beendet. Vor jedem Aufruf muß die Startadresse für den Text im Indexregister X gespeichert werden, d. h. das niederwertige Adreßbyte. Werden die drei Fehlermeldungen der Routinen vom Anwender abgekürzt, so steht ein größerer Raum für eigene Texte zur Verfügung. Dazu die Adressen:

06BD: Fehlermeldung bei Division durch 0

0767: Fehlermeldung bei fehlerhafter Eingabe

077B: Fehlermeldung bei Überschreiten der Rechenkapazität

Der Textbuffer beginnt bei der Adresse 0785.

Soviel zu den Routinen. Ein kleines Beispiel soll zeigen, wie aus ihnen komplexe Funktionen zusammengesetzt werden können. *Bild 2* zeigt ein Programm zur Berechnung des Sinus. Die Eingabe erfolgt im Bogenmaß, Startadresse ist 08D8.

Wie schon gesagt, sind einige Änderungen erforderlich, soll das Programm auf einem anderen System laufen. Die größten Schwierigkeiten dürften dabei bei der Eingabe-Routine auftreten, da der Display-Buffer des AIM als Zwischenspeicher benutzt wird. Nur so ist die Benutzung der DELETE-Funktion für Korrekturen möglich. Der Anwender muß hier ggf. nach einer gleichwertigen Lösung suchen oder auf die Korrekturmöglichkeit verzichten.

Ferner finden folgende Routinen des AIM-Monitors Verwendung:

E97A

Ausgabe eines ASCII-Zeichens im Akku.

EA51

Ausgabe der vier niederwertigen Bits des Akku als ASCII-Zeichen.

EA46

Wie EA51, jedoch werden zuvor die vier höherwertigen Bits ausgegeben.

EA7D

Ein ASCII-Zeichen im Akku wird in eine hexadezimale Ziffer von 0-F verwandelt.

EA84

Wie EA7D, jedoch werden bei zweimaligem Aufruf zwei ASCII-Zeichen in ein Byte gepackt. In beiden Fällen wird das Carry-Flag gesetzt, wenn es sich bei dem Zeichen um keine hexadezimale Ziffer handelt.

E9F0

Ausgabe einer CRLF-Folge bzw. Löschen des Display.

E95F

Eingabe eines ASCII-Zeichens von der Tastatur in den Akku.

Abschließend noch einige Worte zur Rechengeschwindigkeit. Hier die Werte für die verschiedenen Operationen:

Addition: 0,5...3,5 ms
Subtraktion: 0,5...3,5 ms
Multiplikation: 4,5...60 ms
Division: 140...180 ms

Die Werte gelten für eine Taktfrequenz von 1 MHz. Diese Geschwindigkeiten erscheinen recht langsam, vergleicht man sie mit denen guter programmierbarer Taschenrechner. Die Überlegenheit des Mikrocomputers zeigt sich erst in langen Programmen, da zwischen der Ausführung zweier Operationen nur wenig Mikrosekunden vergehen. Dadurch wird die relativ geringe Rechengeschwindigkeit mehr als ausgeglichen.

Als letztes sollte noch gesagt werden, daß, wenn der Anwender innerhalb seines eigenen Programmes den Dezimalmodus mit dem Befehl CLD aufhebt, vor Aufruf der nächsten Rechenroutine unbedingt der gegenteilige Befehl, nämlich SED, folgen muß, sollen die Routinen einwandfrei arbeiten.