

Wahl des Anzeigeformat

Dieser Taschenrechner bietet eine Vielzahl von Anzeigeformaten zur Darstellung von Zahlen und ALPHA-Zeichen. Sie können das Format, in dem die Zahlen dargestellt werden, selbst bestimmen.

Die interne Darstellung aller Zahlen ist allerdings unabhängig von der Wahl des Anzeigeformats; die Zahlen bestehen aus einer 10-stelligen Mantisse und einem zweistelligem Exponenten zur Basis 10.

Sie können auch wählen, ob Vorkommastellen und Nachkommastellen durch ein Komma (wie im europäischen Sprachraum üblich) getrennt werden, oder wie in Amerika üblich durch einen Punkt. Das gewünschte Format können Sie über die Zahlenanzeigeflags einstellen.

Siehe auch

[Festkommaformat](#)

[Wissenschaftliches Anzeigeformat](#)

[Technisches Anzeigeformat](#)

[Zahlenanzeigesteuerflags](#)

FIX Festkommaformat

Im Festkommaformat werden alle Zahlen mit einer exakt definierten Anzahl von Stellen hinter dem Komma angezeigt. Der Exponent wird nicht dargestellt; Wenn eine Darstellung der Zahl in diesem Format nicht möglich ist, zum Beispiel wenn die Zahl größer als $1e10$ ist wechselt die Darstellung in das wissenschaftliche Anzeigeformat über.

Das Format wird mit den Tasten 

FIX, gefolgt von der Zahl der gewünschten Nachkommastellen 0 bis 9 gewählt:

Tastenfolge	Anzeige	
2.24136	2.24136_	<i>Die Zahl</i>
 FIX	FIX _	<i>In der Anzeige erscheint die Funktion FIX. Der Rechner erwartet eine zweistellige Eingabe.</i>
2	FIX 2	<i>Wenn Sie die Taste kurz gedrückt halten erscheint die Funktion ...</i>
	2.24	<i>.. und anschließend wird die Zahl auf zwei Stellen hinter dem Komma gerundet angezeigt.</i>
 FIX 0	2.	<i>Die Zahl wird jetzt ohne Nachkommastellen angezeigt.</i>
 FIX 9	2.241360000	<i>Die Zahl mit 9 Nachkommastellen. Die Zahl wird mit nachfolgenden Nullstellen ergänzt, um alle neun Nachkommastellen anzuzeigen.</i>
 FIX 4	2.2414	<i>Die Zahl erscheint in der Anzeige aufgerundet.</i>

Siehe auch

[Wissenschaftliches Anzeigeformat](#)

[Technisches Anzeigeformat](#)

[Zahlenanzeigesteuerflags](#)

SCI Wissenschaftliches Anzeigeformat

Das wissenschaftliche Anzeigeformat entspricht dem Anzeigeformat für Zahlen in Exponent-Darstellung. Es wird immer eine Ziffer vor dem Komma angezeigt und eine wählbare Anzahl Stellen hinter dem Komma; Diese sind jedoch auf maximal sieben Stellen begrenzt, weil für die Darstellung des Exponenten immer drei Stellen benötigt werden.

Dieses Format wird mit den Tastenfunktionen 

SCI und der Anzahl anzuzeigender Nachkommastellen gewählt:

Tastenfolge	Anzeige	
299792500	299792500_	<i>Lichtgeschwindigkeit im Vakuum.</i>
 SCI	SCI_	<i>Die Funktion und das Unterstreichungszeichen.</i>
3	2.99808	<i>Bedeutet 2.998e08. Die Anzeige wird an der dritten Nachkommastelle aufgerundet.</i>
 SCI 0	3.08	<i>Bedeutet 3e08.</i>

Siehe auch

[Festkommaformat](#)

[Technisches Anzeigeformat](#)

[Zahlenanzeigesteuerflags](#)

ENG Technisches Anzeigeformat

Das technische Anzeigeformat ähnelt dem wissenschaftlichen Anzeigeformat, mit dem Unterschied, daß der Exponent zur Basis 10 ein Vielfaches von 3 ist. Diese Anzeige ist besonders in den Bereichen nützlich, wenn Eingabewerte und Resultate in Vielfachen von 1000 dargestellt werden sollen.

Dieses Format wird mit den Tastenfunktionen 

ENG und der Anzahl anzuzeigender Nachkommastellen gewählt:

Tastenfolge	Anzeige	
7.8379 EEEX 3	7.8379 3_	Die Zahl.
 ENG	ENG _	Die Funktion und das Unterstreichungszeichen.
3	78.38 03	Technisches Anzeigeformat. Die Zahl erscheint auf zusätzliche 3 Stellen hinter der ersten Stelle gerundet.
 ENG 4	78.379 03	Die Anzeige wird auf insgesamt 5 Stellen gerundet.
 ENG 0	80. 03	Die Anzeige wird auf genau eine Stelle gerundet. Eine zusätzliche Null wird vor dem Komma noch angefügt damit eine Darstellung in diesem Format möglich ist.

Beachten Sie, daß in diesem Zahlenformat eine Rundung, wie im letzten Beispiel, auch vor dem Komma auftreten kann. Im technischen Format wird der Dezimalpunkt immer so verschoben, daß der Exponent immer ein Vielfaches von 3 ist. Wird die letzte Zahl zum Beispiel durch 10 dividiert, verschiebt sich der Dezimalpunkt um eine Stelle nach links und der Exponent bleibt unverändert:

Tastenfolge	Anzeige	
 ENG 3	78.38 03	Die Zahl.
10 	7.838 03	Der Dezimalpunkt wird verschoben.
100 	783.8 03	Der Dezimalpunkt wird verschoben.

Siehe auch

Festkommaformat

Wissenschaftliches Anzeigeformat

Zahlenanzeigesteuerflags

Programmkorrektur

Auf dem Tastenfeld des Taschenrechners befinden sich eine Reihe von Funktionen, die das Überarbeiten ihrer Programme einfach gestalten.

CLP	<u>Löschen eines Programms.</u>
←	<u>Löschen einer Programmzeile.</u>
DEL	<u>Löschen mehrerer Programmzeilen.</u>
SST	<u>Einzelschritt vor.</u>
BST	<u>Einzelschritt zurück.</u>
GTO ◊	<u>Sprung zu Programmzeile oder einer ALPHA-Marke.</u>
GTO ◊ ◊	<u>Sprung an Ende eines Programmes.</u>
SIZE	<u>Festlegen der Speicherplatzzuteilung</u>
PACK	<u>Packen des Programmspeichers.</u>

CLP Löschen von Programmen

Sie können jedes Program, daß Sie in dem Rechner gespeichert haben, mittels der **CLP**-Fktion löschen, indem Sie zusätzlich den Programmnamen angeben.

CLP löscht sämtliche Befehle eines Programms inklusive der Programmarke und der **END**-Anweisung. Aus diesem Grund sollten Sie Ihren Programmen ein **END** anfügen, da sonst versehentlich weitere Programme gelöscht werden könnten. Wenn Sie

CLP ohne Angabe eines Funktionsnamens ausführen (drücken Sie dazu

CLP

ALPHA

ALPHA), wird das Programm gelöscht, auf dem der Rechner positioniert ist.

Wenn das Programm oder eine ALPHA-Marke innerhalb des Programms einer Taste zur Ausführung zugewiesen war, so werden diese Zuordnungen aufgehoben.

SST Einzelschritt vorwärts

Im PRGM-Modus springt der Rechner um eine Zeile im Programmspeicher vor.

Im Normal- oder im USER-Modus führt **SST** die in die augenblickliche Zeile gespeicherte Anweisung aus und springt dann zur folgenden Anweisung. Wenn Sie **SST** für kurze Zeit gedrückt halten, erscheint die augenblickliche Zeile in der Anzeige. Wie alle anderen Funktionen kann die Ausführung des Einzelschrittes abgebrochen werden, wenn Sie die Taste solange gedrückt halten, bis **NULL** in der Anzeige erscheint. Bei der Verwendung von **CATALOG** wird um einen Eintrag vorgeschaltet.

BS1 Einzelschritt zurück

Im PRGM-, Normal- und USER-Modus springt der Rechner um eine Zeile im Programmspeicher zurück. Wenn Sie **SS1** für kurze Zeit gedrückt halten, erscheint diese Zeile anschließend in der Anzeige. Im Gegensatz zu

SS1 werden mit

BS1 keine Anweisungen ausgeführt.

Bei der Verwendung von **CATALOG** wird um einen Eintrag zurückgeschaltet.

-Funktion

Im PRGM-Modus werden alphanumerische Eingaben oder ganze Zeilen, die sich im Programmspeicher befinden, gelöscht.

DEL Löschen einer Programmzeile

Im PRGM-Modus werden alphanumerische Eingaben oder ganze Zeilen, die sich im Programmspeicher befinden, gelöscht.

GTO

•

Wenn eine dreistellige Zahl als Eingabe folgt, springt der Rechner auf die angegebene Programmzeile des augenblicklichen Programms. Ist die Eingabe eine ALPHA-Marke, springt der Rechner auf diese Marke.

Die **GTO**

•

• - Funktion

Wenn Sie

GTO

•

• drücken, wird die

PACK-Anweisung ausgeführt, und der Rechner im Anschluß an das Ende des Programmbereichs gesetzt.

Im PRGM-Modus erscheint in der Anzeige **00 REG nn**, wobei nn die Anzahl der freien Register im Programmspeicher angibt.

Ferner wird überprüft, ob das letzte im Programmspeicher stehende Programm mit einer **END**-Anweisung abgeschlossen ist. Wenn diese Anweisung fehlt, fügt die

GTO

•

•-Anweisung eine

END-Anweisung ein. Auf diese Weise kann der Rechner den Programmspeicher automatisch verwalten.

SIZE Festlegen der Speicherplatzzuteilung

Mit der **SIZE**-Anweisung können Sie die Anzahl der Speicherregister für die Daten festlegen.

Wenn sie mit der Eingabe einer dreistelligen Zahl die Anzahl der Daten-Speicherregister angeben, werden dem Programmspeicher automatisch die verbleibenden freien Register zugewiesen. Wenn Sie versuchen, den Speicherplatz für die Daten zu vergrößern, aber für die Anweisung nicht genügend freier Programmspeicherplatz zur Verfügung steht, erscheinen die Meldungen **PACKING** und **TRY AGAIN** in der Anzeige. Wenn Sie jetzt **SIZE** nochmals ausführen, und wieder **TRY AGAIN** in der Anzeige erscheint, bedeutet dies, daß eine Neuordnung des Speicherplatzes ohne vorheriges Löschen von Programmschritten nicht möglich ist.

Wenn Sie den Speicherplatz für die für Daten verkleinern, geht die Information aus den nicht mehr zugeteilten Registern verloren.

Wenn Sie den Versuch unternehmen, ein Register anzusprechen, das bei gegebener Speicherplatzzuordnung nicht vorhanden ist, erscheint die Meldung **NONEXISTENT** in der Anzeige.

Die **PACK** - Funktion

Während Sie ein Programm editieren, fügt der ttCalc zusätzlich freie Zeilen in Ihr Programm ein. Diese freien Zeilen bleiben jedoch unsichtbar. Zweck der freien Zeilen ist, daß beim Löschen und Einfügen von Anweisungen der Rechner so schnell wie möglich auf Ihre Instruktionen reagiert.

Es existieren verschiedene Möglichkeiten diesen freien Zeilen wieder zu entfernen. Der Vorgang wird mit *Packen* bezeichnet. Unter folgenden Bedingungen packt der Rechner automatisch den Programmspeicher:

- Jedesmal, wenn **CLP** ausgeführt wird.
- Bei jedem Versuch zusätzlich Zeilen einzufügen, wenn nicht genügend Platz vorhanden ist. Der Rechner meldet sich anschließend mit **TRAY AGAIN**, worauf Sie die Zeile noch einmal eingeben sollten.
- Wenn Sie **CLP**
CLP
•
• drücken.
- Bei einer Tastenzuordnung mittels **ASN**, wenn nicht genügend Programmspeicher vorhanden ist, um die Zuordnung auszuführen.

Sie können jederzeit den Programmspeicher packen, indem sie die **CLP** - Funktion ausführen.

Meldungen und Fehler

Im folgenden sind alle Meldungen und Fehler aufgelistet, die im Display des ttCalc erscheinen können. Sie können die Fehlermeldung löschen, in dem Sie die Back-Space-Taste auf der PC-Tastatur drücken oder **CLP** ausführen.

Anzeige	Bedeutung
ALPHA DATA	Der Rechner versuchte, eine numerische Operation wie Addition oder Subtraktion mit nicht-numerischen Daten oder einem ALPHA-String durchzuführen.
DATA ERROR	Der ttCalc versuchte eine unerlaubte Operation auszuführen. Diese Operationen sind: CLP , wenn $x=0$. y^x , wenn $y=0$. y^x , wenn $y<0$ und x nicht ganzzahlig. √x , wenn $x<0$. 1/x , wenn $x=0$. LOG , wenn $x≤0$. LN , wenn $x≤0$. LN1+X , wenn $x≤-1$. COS⁻¹ , wenn $ x >1$. SIN⁻¹ , wenn $ x >1$. MEAN , wenn $n=0$. %CH , wenn $y=0$. OCT , wenn $ x >1073741823$ (dezimal). OCT , wenn x nicht ganzzahlig. DEC , wenn x 8 oder 9 enthält. DEC , wenn x nicht ganzzahlig ist. FACT , wenn $x<0$ oder x nicht ganzzahlig. STO CLP , wenn $x=0$. CLP CLP X , wenn $ x ≥10$. CLP CLP X , wenn $ x ≥10$. CLP CLP X , wenn $ x ≥10$. STONE CLP X , wenn $ x ≥10$.
MEMORY LOST	Der Langzeitspeicher des Rechners ist gelöscht. Ungewollt kann dies zum Beispiel geschehen, wenn die Datei Hauptspeicherdatei bei der Initialisierung der Applikation ttCalc fehlerhaft ist oder nicht vorhanden ist.
NONEXISTENT	Der Rechner versucht ein Register anzusprechen das nicht existiert oder momentan nicht als Speicherregister zugewiesen ist. Oder es wurde versucht, eine nicht existierende Funktion mit ASN oder XEQ anzusprechen. Oder es wurde versucht, mit ASN , CLP oder XEQ eine nicht existierende ALPHA-Marke anzusprechen.
NULL	Das Drücken der Taste wurde dadurch ungültig, daß die Taste länger als etwa eine Sekunde

gedrückt blieb.

- PRIVATE** Es wurde versucht ein geschütztes Programm einzulesen.
- OUT OF RANGE** Eine Zahl überschreitet den darstellbaren Bereich. Eine Bereichsüberschreitung tritt auf, wenn $|x_i| > 9.999999999999999e99$ ist.
FACT, wenn $x > 69$.
10^x, wenn $x \geq 100$.
e^x, wenn $x \geq 230.2585093$.
SDEV, wenn bei der Berechnung der Standardabweichung S_x , S_y für x , y eine Division durch Null erfolgt oder unter der Wurzel ein negativer Term steht.
 $(S_x = \sqrt{M/n(n-1)} ; S_y = \sqrt{N/n(n-1)})$.
 $(M = n\sum x^2 - (\sum x)^2 ; N = n\sum y^2 - (\sum y)^2)$.
- PACKING** Der Programmspeicherinhalt wird im Moment gepackt.
- TRY AGAIN** Im Rahmen einer PACK-Operation ist die letzte eingetastete Befehlsfolge zu wiederholen. Dies kann ein **CLP**, **XEQ** oder **CLP CLP CLP**-Befehl sein.
Oder der Versuch einen Befehl in ein Programm einzufügen.
- YES** Die Antwort auf die Frage nach dem Flag-Zustand, wenn das Ergebnis der Operation *wahr* ist.
Möglicherweise auch das Ergebnis eines Vergleichs zwischen dem X-Register und 0 (oder dem Y-Register), wenn das Ergebnis *wahr* ist.
- NO** Die Antwort auf die Frage nach dem Flag-Zustand, wenn das Ergebnis der Operation *falsch* ist.
Möglicherweise auch das Ergebnis eines Vergleichs zwischen dem X-Register und 0 (oder dem Y-Register), wenn das Ergebnis *falsch* ist.
- RAM** Es wurde versucht, ein RAM-Programm mit **COPY** zu kopieren.
- ROM** Es wurde versucht, ein ROM-Programm zu verändern.

Flags

Die Flags sind ein wichtiges Hilfsmittel bei der Programmierung des Taschenrechners ttCalc. Die Flags nehmen innerhalb des Speichers einen 7-Byte großen Bereich ein. Da es sich bei diesen Flags um Speicherbits handelt, die gesetzt (binär 1) oder gelöscht (binär 0) sein können, existieren 56 verschiedene Anwender- und Systemflags.

Flag-Funktionen

Anwenderflags:

- Anwenderflags für spezielle Zwecke
- Bereichsfehlerflag
- Dateneingabeflags
- Druckersteuerungsflag
- Fehlerignorierflag
- Flags zur allgemeinen Verwendung
- Flag für automatischen Programmablauf
- Steuerungsflag für akustische Signale
- User-Modus-Flag
- Zahlenanzeige-Steuer-Flags

Systemflags:

- Alpha-Modus-Flag
- Anzeigeformat-Flags
- Batterie-Kontroll-Flag
- Dateneingabe-Systemflag
- Drucker-Anwesenheits-Flag
- Ein/Ausgabe-Flag
- Einschaltungsdauerflags
- Katalogflag
- Meldungsflag-Flag
- Pause-Flag
- CLP Peripherie-Flags
- CLP Prgm-Modus-Flag
- CLP Shift-Flag
- CLP SST-Flag
- CLP Stellenzahl-Flags
- CLP Tasten-Teilfolge-Flag
- CLP Winkel-Modus-Flags

System-Flags (Flags 30 bis 55)

Die Systemflags werden intern für verschiedenste Zwecke vom Rechner verwendet. Daher ist ein Setzen und Löschen dieser Flags vom Anwender nicht erlaubt. Sie können den Zustand der Systemflags jedoch mit den Tastenfunktionen **FS?** und **FC?** überprüfen.

Anwender-Flags (Flags 00 bis 29)

Die 30 Anwender-Flags können von Ihnen beliebig gelöscht, gesetzt und geprüft werden. Die Verwendung der Systemflags (Flags 30 bis 55) ist dagegen nur eingeschränkt möglich.

Flags 0 .. 10

Flags 10 .. 20

Flag 11

Flag 21

Flags 22 und 23

Flag 24

Flag 25

Flag 26

Flag 27

Flags 28 und 29

Flag 30

Flags 31 .. 35

Flags 36 .. 39

Flags 40 und 41

Flags 42 und 43

Flag 44

Flag 45

Flag 46

Flag 47

Flag 48

Flag 49

Flag 50

Flag 51

Flag 52

Flag 53

Flag 54

Flag 55

Flag-Funktionen

Ihnen stehen sechs Funktionen zum Testen, Setzen und Löschen von Flags zur Verfügung:

- SF** Flag setzen.
- CF** Flag löschen.
- FS?** Flag prüfen, ob es gesetzt ist.
- FC?** Flag prüfen, ob es gelöscht ist.
- FS?C** Flag prüfen, ob es gesetzt ist, und anschließend löschen.
- FC?C** Flag prüfen, ob es gelöscht ist, und anschließend löschen.

Wenn sie zum Beispiel die **SF**-Funktionen ausführen, werden Sie zur Eingabe der gewünschten Flag-Nummer von 00 bis 55 aufgefordert:

Tastenfolge	Anzeige	
CLP SF	SF _ _	<i>Aufforderung zur Eingabe der Flag-Nummer.</i>
00	0.0000	<i>Flag 00 ist gesetzt und im Anzeigefeld erscheint der Indikator.</i>

Sie können die Flag-Prüffunktionen **FS?**,

FC?,

FS?C und

FC?C zur Abfrage des aktuellen Zustands der Flags verwenden. Die Ausführung dieser Funktionen ist vom Systemzustand des Rechners abhängig:

- Wenn im Programmlauf die Bedingung erfüllt ist, setzt der Rechner die Ausführung mit der nächsten Programmzeile fort. Ist die Bedingung nicht erfüllt, wird die nächste Programmzeile übersprungen.
- Wird eine dieser Funktionen über das Tastenfeld ausgeführt erscheint die Antwort auf die Prüffunktion in der Anzeige. Wenn die Bedingung erfüllt ist, erscheint **YES**, ist sie nicht erfüllt, erscheint **NO**.

Siehe auch

Anwenderflags

Systemflags

Dateneingabe-Flags (Flag 22 und 23)

Diese beiden Flags dienen als Indikatoren für die Dateneingabe über das Tastenfeld:

- Flag 22 wird als Indikator für die Eingabe numerischer Daten verwendet. Wenn sie Zahlen über das Tastenfeld eingeben, wird Flag 22 automatisch vom Rechner gesetzt.
- Flag 23 wird als Indikator für die Eingabe alphanumerischer Daten verwendet. Wenn sie ALPHA-Daten über das Tastenfeld eingeben, wird Flag 23 automatisch vom Rechner gesetzt.

Die Flags 22 und 23 werden beim Einschalten des Rechners gelöscht.

Flag für automatischen Programmlauf (Flag 11)

Diese Flag ist eines der Anwenderflags für spezielle Zwecke. Seine Verwendung ist die Steuerung des Programmlaufs beim Einschalten des Rechners.

Wird bei gesetztem Flag 11 der Rechner aus- und wieder eingeschaltet, dann wird automatisch mit der Ausführung des Programms begonnen, auf das der Rechner gesetzt ist. Damit Sie ein solches Verhalten bemerken können, wird in diesem Fall vor Ausführungsbeginn ein kurzer Signalton ausgegeben.

Wie alle Anwenderflags für spezielle Zwecke wird auch dieses Flag beim Einschalten des Rechners gelöscht.

Anwenderflags für spezielle Zwecke (Flags 11 bis 20)

Diese Flags stehen ebenfalls unter Ihrer Kontrolle, haben jedoch eine zusätzliche besondere Funktion. Diese Flags können von Ihnen gesetzt, gelöscht und mit jeder beliebigen Prüffunktion abgefragt werden. Unter gewissen Bedingungen kann der Zustand dieser Flags auch vom Rechner verändert werden.

Alle 10 Anwenderflags für spezielle Zwecke werden beim Einschalten des Rechners gelöscht.

Flags zur allgemeinen Verwendung (Flags 00 bis 10)

Diese Flags stehen ohne jede Einschränkung unter Ihrer Kontrolle. Sie können gesetzt, gelöscht und mit jeder beliebigen Prüffunktion abgefragt werden. Der aktuelle Status der Flags 00 bis 04 kann zudem jederzeit in der Anzeige des Rechners verifiziert werden.

Der Zustand dieser Flags wird im Langzeitspeicher des Rechner aufrechterhalten.

Drucker-Steuerungsflag (Flag 21)

Diese Flag wird zur Zeit nicht, verwendet, da eine Simulation des Druckers HP-82143 in dieser Version nicht realisiert ist. Der Rechner verhält sich immer so, als wäre kein Drucker angeschlossen.

Flag 21 wird beim Einschalten des Rechners gelöscht.

Fehler-Ignorierflag (Flag 24)

Dieses Flag steuert die Reaktion des Rechners bei der Ausführung einer verbotenen Operation.

Normalerweise bricht der Rechner die Ausführung ab, wenn eine Anweisung nicht korrekt durchgeführt werden kann, wie zum Beispiel eine Division durch Null. Dann erscheint entweder die Fehlermeldung **DATA ERROR** oder **OUT OF RANGE**, wenn es sich um einen Bereichsfehler handelt, in der Anzeige. Diese verbotene Operation wird jedoch ignoriert, wenn Flag 25 gesetzt war. Das Programm wird fortgesetzt, ohne die entsprechende Anweisung auszuführen.

Das Flag 25 wird automatisch bei jedem Versuch eine verbotene Operation auszuführen gelöscht. Es ist daher sinnvoll dieses Flag unmittelbar vor der Anweisung zu setzen, bei der eine Fehler auftreten kann. Es ist auch denkbar das Fehler-Ignorierflag auch nach einer verdächtigen Funktion abzufragen. Dies bietet Ihnen die Möglichkeit, eine Programmunterbrechung aufgrund fehlerhaften Daten zu vermeiden.

Flag 25 wird beim Einschalten des Rechners automatisch gelöscht.

Siehe auch

[Bereichsfehler-Ignorierflag](#)

[Meldungen und Fehler](#)

Bereichsfehler-Ignorierflag (Flag 24)

Dieses Flags steuert die Reaktion des Rechners bei einer arithmetischen Bereichsüberschreitung.

Normalerweise wird, wenn diese Bereichsüberschreitung auftritt, sofort die Fehlermeldung **OUT OF RANGE** ausgegeben, ohne die fehlerhafte Funktion auszuführen. Wenn dies während eines Programmlaufs geschieht, wird auch die Ausführung des Programms gestoppt. Das Flag 24 ermöglicht es Ihnen diesen Overflow-Error zu ignorieren.

Wenn Flag 24 gesetzt ist, wird bei einer Bereichsüberschreitung $\pm 9.999999999e99$ in das angesprochene Register eingetragen und die Ausführung fortgesetzt. Beachten Sie, daß das Flag 24 nach einer solchen Operation nicht gelöscht wird. Bereichsfehler treten auf, wenn das Ergebnis einer arithmetischen Operation $\pm 9.999999999e99$ überschreitet. Bereichsunterschreitungen, Zahlen kleiner als $\pm 9.999999999e-99$, lösen keinen Fehler aus. In diesem Fall wird das Ergebnis gleich Null gesetzt.

Flag 24 wird beim Einschalten des Rechners automatisch gelöscht.

Siehe auch

[Fehler-Ignorierflag](#)

[Meldungen und Fehler](#)

Steuerungsflag für das akustische Signal (Flag 26)

Wenn dieses Flag gesetzt ist, erzeugen die Funktionen **CLP** und **BEEP** Töne. Es dieses Flag dagegen gelöscht, lassen sich die akustischen Signale nicht auslösen.

Zur Ausgabe der akustischen Signale sollten Sie eine Soundkarte (z.B Soundblaster) installieren. Eine Ausgabe ist jedoch mit Hilfe des SPEAKER-Drivers von Microsoft™ möglich.

Flag 26 wird beim Einschalten des Rechners automatisch gesetzt.

Zahlenanzeige-Steuerflags (Flag 28 und 29)

Die Flags 28 und 29 bestimmen die Darstellung der Zahlen im Anzeigefeld des Rechners.

- Das Dezimalpunktflag (Flag 28) steuert das Dezimalzeichen und das Trennzeichen bei der Zahlendarstellung. Unter Trennzeichen wird verstanden, daß die Vorkommastellen bei großen Zahlen in Gruppen zu je drei Ziffern aufgeteilt werden. In Europa ist das Dezimalzeichen das Komma und das Trennzeichen der Punkt. In den USA ist dies genau umgekehrt. Das Setzen und Löschen dieses Flags ermöglicht es Ihnen, jenes Trenn- und Dezimalzeichen zu verwenden, mit dem Sie am meisten vertraut sind.
- Das Zifferngruppierungsflag (Flag 29) steuert die Verwendung des Trennzeichens bei der Zahlendarstellung. Ist dieses Bit gesetzt, werden die Gruppen durch das Trennzeichen bei der Zahlendarstellung getrennt, sonst erscheinen die Ziffern direkt aufeinanderfolgend.

Als Beispiel erscheint je nach Systemzustand die Zahl 1234567.89 folgendermaßen im Anzeigefeld:

Flag 28	Flag 29	Anzeige
<i>gesetzt</i>	<i>gesetzt</i>	1,234,567.89
<i>gesetzt</i>	<i>gelöscht</i>	1234567.89
<i>gelöscht</i>	<i>gesetzt</i>	1.234.567,89
<i>gelöscht</i>	<i>gelöscht</i>	1234567,89

Der Zustand dieser Flags wird im Langzeitspeicher des Rechner aufrechterhalten.

User-Modus-Flag (Flag 27)

Wenn dieses Flag gesetzt ist, befindet sich der Rechner im USER-Modus, in dem die Tastenzuordnungen aktiv sind. Ist dieses Bit gelöscht, ist der USER-Modus ausgeschaltet.

Sie können dieses Flag 27 setzen, löschen und abfragen wie ein allgemeines Anwenderflag.

Das Setzen und löschen entspricht dabei exakt dem Drücken der **USER**-Taste.

Der Zustand dieses Flags wird im Langzeitspeicher des Rechner aufrechterhalten.

Katalog-Flag (Flag 30)

Dieses Flag wird intern zur Durchführung der Katalogfunktion verwendet. Außerdem wird es für den Anwender dauernd auf einen gelöschten Zustand geprüft.

Peripherie-Flags (Flags 31 bis 35)

Diese Flags werden vom HP-41C™ intern für Funktionen peripherer Erweiterungseinheiten verwendet.

Das Windows-Emulationsprogramm ttCalc verwendet diese Flags nicht.

Stellenzahl-Flags (Flags 36 bis 39)

Die Kombination dieser vier Flags wird zur Festlegung der angezeigten Dezimalstellen im

CLP,

CLP und

CLP- Anzeigeformat verwendet. Die Stellenzahl ist in diesen Bits binär abgespeichert.

Der Zustand dieses Flags wird im Langzeitspeicher des Rechner aufrechterhalten.

Anzeigeformat-Flags (Flags 40 und 41)

Mit diesen Flags wird das Anzeigeformat der Zahlen in der Anzeige gesteuert:

- Wenn beide Speicherbits gelöscht sind befindet sich der Rechner im **CLP**-Format.
- Wenn Flag 41 gesetzt und Flag 40 gelöscht ist, ist der Rechner im **CLP**-Format.
- Wenn Flag 41 gelöscht und Flag 40 gesetzt ist der Rechner im **CLP**-Format.
- Wenn Flag 41 gesetzt und Flag 40 gesetzt ist, ist der Rechner im **CLP**-Format.

Der Zustand dieses Flags wird im Langzeitspeicher des Rechner aufrechterhalten.

Siehe auch

[Stellenzahlflags](#)

[Zahlenanzeigeflags](#)

[Festkommaformat](#)

[Wissenschaftliches Anzeigeformat](#)

[Technisches Anzeigeformat](#)

[Zahlenanzeigesteuerflags](#)

Winkel-Modus-Flags (Flags 42 und 43)

Mit diesen Flags ist der augenblickliche Winkelmodus des Rechners definiert.

- Wenn beide Speicherbits gelöscht sind, befindet sich der Rechner im Bogenmaß-Modus.
- Wenn Flag 42 gesetzt und Flag 43 gelöscht ist, ist der Rechner im Neugrad-Modus.
- Wenn Flag 43 gesetzt und Flag 42 gelöscht ist der Rechner im Altgrad-Modus.

Der Zustand dieses Flags wird im Langzeitspeicher des Rechner aufrechterhalten.

Siehe auch

[Winkel-Modus](#)

Einschaltungsdauer-Flag (Flag 44)

Dieses Flag steuert im HP-41C™, ob der Rechner ständig eingeschaltet bleibt oder nicht. Ist dieses Flag gesetzt, bleibt der HP-41C™ dauernd eingeschaltet, sonst schaltet sich der Rechner automatisch nach 10 Minuten ab.

Das Windows-Emulationsprogramm ttCalc verwendet dieses Flag nicht.

Dateneingabe-Flag (Flag 45)

Dieses Flag wird intern bei der Dateneingabe verwendet. Außerdem wird es für den Anwender dauernd auf einen gelöschten Zustand geprüft.

Tastenteilfolge-Flag (Flag 46)

Dieses Flag wird intern bei der Durchführung von Funktionen verwendet, die eine ein- oder mehrstellige Parametereingabe erwarten. Außerdem wird es für den Anwender dauernd auf einen gelöschten Zustand geprüft.

Shift-Flag (Flag 47)

Dieses Flag wird ttCalc-intern bei der Ausführung der Umschalttaste **CLP** verwendet. Wenn das Flag 47 gesetzt ist, erscheint der SHIFT-Status Indikator in der Anzeige des Rechners.

Alpha-Modus-Flag (Flag 48)

Dieses Flag wird intern zur Steuerung des ALPHA-Modus verwendet. Solange diese Flag gesetzt ist, befindet sich der Rechner im ALPHA-Modus.

Batterie-Kontrollflag (Flag 49)

Das Windows-Emulationsprogramm verwendet dieses Flag nicht.

Meldungs-Flag (Flag 50)

Wenn dieses Flag gesetzt ist, enthält die Anzeige einer Fehlermeldung. Sonst erscheint die übliche Anzeige, je nach Betriebszustand.

SST-Flag (Flag 51)

Dieses Flag wird intern zur Durchführung eines Einzelschrittes bei der Programmausführung verwendet. Außerdem wird es für den Anwender dauernd auf einen gelöschten Zustand geprüft.

Prgm-Modus-Flag (Flag 52)

Dieses Flag wird intern zur Steuerung des PRGM-Modus verwendet. Es ist gesetzt, wenn Sie Programme editieren. Sonst ist es gelöscht, auch bei der Ausführung eines Programmes. Außerdem wird es für den Anwender dauernd auf einen gelöschten Zustand überprüft.

Ein/Ausgabe-Flag (Flag 53)

Das Windows-Emulationsprogramm verwendet dieses Flag nicht.

Pause-Flag (Flag 54)

Wenn dieses Flag gesetzt ist, dann ist im Ablauf eines Anwenderprogramms eine Pausenanweisung **PSE** wirksam. Solange keine PSE-Anweisung wirksam ist, ist dieses Bit Null.

Drucker-Anwesenheits-Flag (Flag 55)

Das Windows-Emulationsprogramm verwendet dieses Flag nicht.

ttCalc Hilfe-Index

Einführung

[Was ist ttCalc ?](#)

[Einführung in die UPN-Logik](#)

[Literatur](#)

Menübefehle

[Datei](#)

[Bearbeiten](#)

[?](#)

Funktionsübersicht

[Allgemeine mathematische Funktionen](#)

[Akustische Funktionen](#)

[Ausführung der Standardfunktionen](#)

[Flags](#)

[Indirekte Operationen](#)

[Logarithmen und Exponentialfunktionen](#)

[Meldungen und Fehler](#)

[Programmierung](#)

[Speicherregister](#)

[Sonderfunktion der oberen zwei Tastenreihen](#)

[Sonstige Funktionen](#)

[Stacklift Bedingungen](#)

[Statistikfunktionen](#)

[Tastenfeld](#)

[Trigonometrische Funktionen](#)

[Vertauschungsfunktionen](#)

[Verzeichnisse](#)

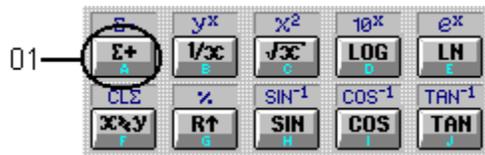
[Zahleneingabe](#)

[Zahlenumwandlungen](#)

Sonderfunktion der oberen zwei Tastenreihen

Die oberen zwei Tastenreihen stehen bei Funktionen, die in der Anzeige eine ein- oder mehrparametrische Eingabe erwarten, in Beziehung zu den Zahlen 1 bis 10. Aufgrund dieser Beziehung können Sie mit einem einzigen Tastendruck zweiziffrige Labels, Adressen oder Funktionsparameter eingeben.

Wenn Sie beispielsweise **CLP** und **Σ+** eintasten, wird dies als **CLP** 01 interpretiert, da die Taste **Σ+** der Zahl 01 entspricht.



Wenn Funktionen mit zweiziffrigen Adressen oder Parametern gedrückt werden, genügt es die der Zahl entsprechende Taste zu drücken:

- CLP** **SIN** **CLP** 08
- =
- LBL** **CLP** **LBL** 02
- =
- CLP** **x*y** **CLP** 06
- =
- RCL** **Σ+** **CLP** 01
- =

Verzeichnisse

Analog zum HP-41C™ existieren auch bei diesem Rechner drei Funktionsverzeichnisse. Wenn Sie die Katalog-Funktion ausführen, gibt der Rechner das entsprechende Verzeichnis aus, indem er am Anfang des Verzeichnisses beginnt und jede Funktionsbezeichnung anzeigt.

- Das Programmverzeichnis enthält alle Funktionen und Programme die Sie geschrieben und gespeichert haben. Sie können dieses Verzeichnis mit **[CLP] 1** ansprechen. Die Einträge sind entsprechend ihrer Position im Programmspeicher angeordnet. Bei der Ausgabe dieses Verzeichnisses wird der Rechner im Programmspeicher dort positioniert, wo das Programm steht, dessen Funktionsname gerade in der Anzeige erscheint. **[CLP] 1** gibt nur die ALPHA-Marken und die END-Anweisungen aus, davon ausgenommen sind die lokalen Marken A bis J und a bis e. Wenn ein Programm nur lokale Marken oder gar keine Marken enthält, aber mit einer END-Anweisung abgeschlossen wurde, wird nur diese END-Anweisung im Listing von CATALOG 1 erscheinen.
- Das Verzeichnis mit dem Sie alle Funktionen der Erweiterungseinheiten beim HP-41C ansprechen können, kann mit **[CLP] 2** angesehen werden. Die Erweiterungseinheiten sind in dieser Version von ttCalc noch nicht implementiert. Mit dieser Anweisung wird daher nur das Standardverzeichnis angezeigt.
- Die **[CLP] 1**-Anweisung listet alle Standardfunktionen des Rechners auf. Die Liste wird alphabetisch geordnet angezeigt. Wenn Sie nähere Informationen zur gerade angezeigten Funktion in diesem Verzeichnis wünschen, drücken sie die F1-Taste. Dadurch wird die Verzeichnisausgabe, falls nötig, gestoppt und ein Hilfe-Fenster mit dem zugehörigen Kontext geöffnet. Zu einigen 'komplizierteren' Funktionen existieren mehrere Hilfefenster, in diesem Fall erscheint zuerst ein Fenster mit der Auflistung aller Themenbereiche.

Sie brauchen eine Verzeichnisliste nicht bis zum Ende auszugeben. Die Ausgabe kann jederzeit mit der **[R/S]**-Taste gestoppt werden. Sie können dann entweder mit **[CLP]** (Einzelschritt zurück) oder

[CLP] (Einzelschritt vorwärts) die gewünschte Position suchen. Sie können aber auch mit **[R/S]** die Ausgabe wieder fortsetzen.

Nur mit der **[R/S]** Taste kann die Ausgabe des Verzeichnisses angehalten oder fortgesetzt werden, alle anderen Tastenpositionen bewirken eine Verlangsamung der Ausgabe.

Vertauschungsfunktionen

Beim ttCalc existiert die $\boxed{X\leftrightarrow}$ Funktion zum Austausch der Register-Inhalte eines beliebigen Registers mit X:

- Um X mit einem anderen Stackregister oder LASTX auszutauschen, führen Sie folgende Tastenkombination aus:

$\boxed{X\leftrightarrow}$

\boxed{CLP} und L, Y, Z, oder T (je nach Register).

- Um X mit einem numerischen Register von 00 bis 99 auszutauschen, drücken Sie $\boxed{X\leftrightarrow}$ gefolgt von einer zweistelligen Registeradresse. Drücken Sie z.B dazu:

$\boxed{X\leftrightarrow}$

\boxed{CLP} X

\boxed{CLP}

\boxed{COS}

\boxed{CLP}

\boxed{TAN}

\boxed{CLP} nn

Die Vertauschung der Register-Inhalte von X und Y erfolgt am einfachsten mit $\boxed{X\leftrightarrow Y}$.

Statistikfunktionen

Der ttCalc stellt Ihnen verschiedene statistische Funktionen wie Standardabweichung und Mittelwert zur Verfügung. Im einzelnen stehen Ihnen folgende Funktionen zur Verfügung:

Berechnung des Mittelwertes

Berechnung der Standardabweichung

Definieren des Statistikblockes

Eingabe von Statistikdaten

Korrigieren falsch eingegebener Daten

Löschen der Statistik-Daten

$\text{CL}\Sigma$ Löschen der Statistikdaten

Diese Funktion löscht den Statistik-Datenblock den Sie mit ΣREG definiert haben. Bevor Sie Berechnungen mit der $\Sigma+$ -Funktion durchführen, sollten Sie dabei verwendeten Daten-Register mit dieser Funktion auf Null zurücksetzen.

Eingabe von Statistikdaten

Wenn Sie die Σ^+ -Funktion ausführen, werden gleichzeitig mehrere Summen und Produkte der Inhalte im X und Y-Register berechnet. Um diese Funktionen für die verschiedenen statistischen Funktionen verfügbar zu halten, werden sie automatisch in einem aus sechs Registern bestehenden Datenblock gespeichert. Diesen Block definieren Sie mit der ΣREG -Funktion. Bevor Sie Berechnungen mit der Σ^+ -Funktion durchführen, sollten Sie den Statistikblock mit der $\text{CL}\Sigma$ -Funktion löschen.

Folgende Schritte sind zur Eingabe der Daten notwendig:

- Tasten Sie den y-Wert ein.
- Drücken Sie **ENTER**.
- Tasten Sie die x-Wert ein.
- Führen Sie Σ^+ aus.

Anschließend geschieht im einzelnen folgendes:

- Die Zahl im X-Register wird zu dem Inhalt des ersten Statistik-Registers addiert.
- Das Quadrat der Zahl im X-Register wird zu dem Inhalt des zweiten Statistik-Registers addiert.
- Die Zahl im Y-Register wird zu dem Inhalt des dritten Statistik-Registers addiert.
- Das Quadrat der Zahl im Y-Register wird zu dem Inhalt des vierten Statistik-Registers addiert.
- Die Zahl im X-Register wird mit der Zahl im Y-Register multipliziert und das Produkt zu dem Inhalt des ersten Statistik-Registers addiert.
- Der Inhalt des sechsten Statistik-Registers wird um eins erhöht. Nachdem der Rechner diese Schritte durchgeführt hat wird der Inhalt dieses Registers in die Anzeige und in das X-Register geschrieben.

Siehe auch

[Korrigieren falsch eingegebener Daten](#)

[Berechnung des Mittelwertes](#)

[Berechnung der Standardabweichung](#)

[Definieren des Statistikblockes](#)

Korrigieren falsch eingegebener Statistikdaten

Wenn Sie einen falschen Wert eingegeben haben und $\Sigma+$ noch nicht ausgeführt haben, drücken Sie statt dessen

CLX oder
 CLP .

Wenn einer der Werte geändert werden soll, oder eine der Zahl mit CLP fehlerhaft eingegeben wurde, können Sie diesen Fehler wie folgt rückgängig machen:

- Tasten Sie das fehlerhafte Datenpaar in das X- und Y-Register ein.
- Drücken Sie CLP
- $\Sigma-$. Damit werden die Werte aus den verschiedenen Summen entfernt.
- Falls benötigt tasten Sie das korrekte Wertepaar ein und führen Sie CLP

Siehe auch

[Eingabe von Statistikdaten](#)

[Berechnung des Mittelwertes](#)

[Berechnung der Standardabweichung](#)

[Definieren des Statistikblockes](#)

Standardabweichung

Mit der **CLP**-Funktion wird die Stichproben- Standardabweichung (als Maß für die Streuung um den Mittelwert) der in den Statistik-Registern summierten Werte berechnet.

Folgendes geschieht, wenn Sie **CLP** ausführen:

- Mit den Daten der Statistikregister die Σx , Σx^2 und **n** enthalten, wird die Standardabweichung der x-Werte berechnet. Das Ergebnis erscheint im X-Register.
- Mit den Daten der Statistikregister die Σy , Σy^2 und **n** enthalten, wird die Standardabweichung der y-Werte berechnet. Das Ergebnis erscheint nach Ausführung im Y-Register. Drücken sie **CLP** um den Wert in die Anzeige zu bringen.

Die einfachste Methode, die für die **CLP**-Funktion benötigten Werte zu summieren ist die **CLP**-Funktion.

Siehe auch

[Eingabe von Statistikdaten](#)

[Korrigieren falsch eingegebener Daten](#)

[Berechnung des Mittelwertes](#)

[Definieren des Statistikblockes](#)

Mittelwert

Mit der **CLP**-Funktion wird das arithmetische Mittel der in den Statistik-Registern summierten x- und y-Werte berechnet.

Folgendes geschieht, wenn Sie **CLP** ausführen:

- Mit den Daten des ersten und letzten Statistikregisters wird der Mittelwert der x-Werte berechnet; dieser erscheint im X-Register.
- Mit den Daten des zweiten und letzten Statistikregisters wird der Mittelwert der y-Werte berechnet. Das Ergebnis befindet sich nach der Ausführung im Y-Register. Mit **CLP** wird dieser Wert für eine eventuelle Verwendung sichtbar gemacht.

Die einfachste Methode, die für die **CLP**-Funktion benötigten Werte zu summieren ist die **CLP**-Funktion.

Siehe auch

[Eingabe von Statistikdaten](#)

[Korrigieren falsch eingegebener Daten](#)

[Berechnung der Standardabweichung](#)

[Definieren des Statistikblockes](#)

Definieren des Statistikblocks

Der Statistikblock dient zur Aufnahme der Statistikregister und ist damit Teil der Speicher-Register. Dieser Block wird mit der Σ REG-Funktion festgelegt.

Wenn Sie Σ REG ausführen, werden sie mit Σ REG __ aufgefordert eine zweistellige Registeradresse einzugeben. Die von Ihnen eingegebene Adresse definiert den Anfang des aus sechs Registern bestehenden Registerblocks R_s . Wenn Sie keinen Statistikblock auf diese Weise definiert haben, werden automatisch die Register R_{11} bis R_{16} verwendet. Die Statistik-Register haben damit folgende Bedeutung:

R_s	Σx
R_{s+1}	Σx^2
R_{s+2}	Σy
R_{s+3}	Σy^2
R_{s+4}	Σxy
R_{s+5}	n

Siehe auch

[Eingabe von Statistikdaten](#)

[Korrigieren falsch eingegebener Daten](#)

[Berechnung des Mittelwertes](#)

[Berechnung der Standardabweichung](#)

Logarithmen und Exponentialfunktionen

Neben dem dekadischen und natürlichen Logarithmus stehen ihnen beim ttCalc auch die entsprechenden Umkehrfunktionen (Exponentialfunktionen) zur Verfügung. Die folgende Auflistung zeigt die verfügbaren Funktionen. Zu beachten ist, daß die Anweisungen, je nachdem ob sie über die Tastatur oder die Anzeige ausgeführt werden, eine unterschiedliche Form haben:

- Natürliche Exponentialfunktion **CLP** oder **E+X** Berechnet die x-te Potenz von e . x entspricht dem Wert im X-register und e ist die Eulersche Konstante.
- Natürlicher Logarithmus **CLP** Berechnet den Logarithmus der Zahl im X-Register zur Basis e .
- Dekadische Exponentialfunktion **CLP** oder **10+X** Berechnet die x-te Potenz von 10. x entspricht dem Wert im X-register.
- Dekadischer Logarithmus **CLP** Berechnet den Logarithmus der Zahl im X-Register zur Basis 10.
- Natürliche Exponentialfunktion (für Argumente nahe eins) **E+X-1** Berechnet $(e^x) - 1$, wobei x eine Zahl sehr nahe Null ist.
- Natürlicher Logarithmus (für Argumente nahe eins) **CLP** Berechnet $\ln(1+x)$, wobei x eine Zahl sehr nahe Null ist.
- Exponentialfunktion **CLP** oder **Y+X** Berechnet die x-te Potenz von y . Dabei entsprechen x und y den Werten im X und Y-Register,

Trigonometrische Funktionen (Übersicht)

Addition und Subtraktion von Zeiten und Winkeln

Koordinatentransformationen

Trigonometrische Funktionen

Umwandlung von Stunden, Minuten, Sekunden

Umwandlung zwischen Grad und Bogenmaß

Winkel-Modus

Zahlenumwandlungen

Der ttCal verfügt über zwei Funktionen zur Transformation von Zahlen zwischen der Basis zur Zahl Zehn und Acht:

- Die **CLP** - Funktion wandelt die Octalzahl x im X-Register in die äquivalente Dezimalzahl um. Wenn x größer als 1,073,741,823 (dezimal) oder nicht ganzzahlig ist erscheint die Meldung **DATA ERROR** in der Anzeige.
- Die **CLP** - Funktion wandelt die Dezimalzahl x im X-Register in die ihr entsprechende Octalzahl um. Wenn x nicht ganzzahlig ist oder die Ziffern 8 und 9 enthält erscheint die Meldung **DATA ERROR** in der Anzeige. 777,777,777 ist die größte Octalzahl, die umgewandelt werden kann.

Akustische Funktionen

Der ttCalc verfügt über zwei Funktionen, die Akustische Signale erzeugen können:

- Die **CLP** - Funktion erzeugt eine Folge akustischer Töne.
- Wenn Sie **CLP** und eine Zahl zwischen 0 und 9 drücken, wird nur ein einziges Signal ausgegeben. Durch den eingetasteten Parameter wird die Tonhöhe des Signals bestimmt; je größer der Parameter, desto höher ist die Frequenz des erzeugten Tons.

Zur Anwendung dieser Funktionen sollten Sie eine Soundkarte (z.B Soundblaster) besitzen. Eine Ausgabe ist jedoch auch mit Hilfe des SPEAKER-Divers von Microsoft™ möglich. Zur Ausgabe werden keine WAVE-Dateien benötigt, da intern eine Berechnung der Frequenzen durchgeführt wird.

Koordinatentransformationen

Zur Umwandlung zwischen Polarkoordinaten und rechtwinkligen Koordinaten stehen Ihnen zwei Funktionen zur Verfügung. Der Winkel θ ist in Abhängigkeit von der Wahl des Winkel-Modus entweder in Altgrad, Neugrad oder im Bogenmaß anzugeben.

Um die rechtwinkligen (x, y) -Koordinaten in die Polarkoordinaten (r, θ) umzuwandeln:

- Tasten Sie die y-Koordinate ein.
- Drücken Sie **ENTER↑**.
- Tasten Sie die x-Koordinate ein.
- Führen Sie **R·P** aus.

Im X-Register steht der Betrag r , im Y-Register der Winkel θ . Um θ anzuzeigen, drücken Sie **CLP**.

Um umgekehrt die Polarkoordinaten (r, θ) in die rechtwinkligen (x, y) -Koordinaten umzuwandeln:

- Tasten Sie den Wert für den Winkel θ ein.
- Drücken Sie **ENTER↑**.
- Tasten Sie den Betrag r ein.
- Führen Sie **P·R** aus.

In der Anzeige und im X-Register steht anschließend die x-Koordinate, die y-Koordinate entsprechend im Y-Register; zur Anzeige können Sie **CLP** verwenden.

Die Funktionen **R·P**

P·R verändern die Stackinhalte von T und Z nicht. Der ursprüngliche Inhalte von den X- und Y-Registern werden mit den neuen Koordinaten überschrieben.

Addition und Subtraktion von Zeiten und Winkeln

Winkel und Zeiten, die in dezimaler Form vorliegen können mit den Funktionen $\boxed{+}$ und $\boxed{-}$ addiert und subtrahiert werden.

Liegen die Ausgangsgrößen dagegen in der Form "Stunden, Minuten und Sekunden" vor, können sie die Funktionen $\boxed{HMS+}$ und $\boxed{HMS-}$ für diese arithmetischen Operationen verwenden.

Analog dazu können Winkel in Format Grad°, Minuten´ Sekunden" mit diesen Funktionen addiert und Subtrahiert werden.

Umwandlung von Stunden, Minuten, Sekunden

Die Umwandlung zwischen Stunden in dezimaler Form und dem Format "Stunden, Minuten und Sekunden" ist mit folgenden Tastenfunktionen möglich:

- **HMS** führt die Umwandlung von Stunden in dezimaler Form in das Format "Stunden, Minuten und Sekunden" durch.
- **HR** führt die Umwandlung von Stunden im Format "Stunden, Minuten und Sekunden" in die dezimaler Form durch.

Diese Funktion kann auch dazu verwendet Winkel im in dezimalen Grad in die Form Grad°, Minuten´ Sekunden" umzuwandeln.

Umwandlung zwischen Grad und Bogenmaß

Die Umwandlung zwischen Altgrad und Bogenmaß (Rad) ist mit den folgenden Tastenfunktionen möglich:

- **D-R** führt die Umwandlung von einem Winkel im Altgrad zu einem Winkel im Bogenmaß durch.
- **R-D** führt die Umwandlung von einem Winkel im Bogenmaß zu einem Winkel in Altgrad durch.

Siehe auch

[Winkel-Modus](#)

Winkel-Modus

Wenn sie mit dem ttCalc trigonometrische Funktionen verwenden, können Winkelargumente wahlweise in Altgrad, Neugrad oder im Bogenmaß ausgedrückt werden.

- Um den Bogenmaß-Modus zu wählen, führen Sie die **RAD** -Funktion aus.
- Um den Neugrad-Modus zu wählen, führen Sie die **GRAD** -Funktion aus.
- Um den Altgrad-Modus zu wählen, führen Sie die **DEG** -Funktion aus.

Im Display erscheint je nach gewähltem Modus ein Status-Indikator.

Trigonometrische Funktionen

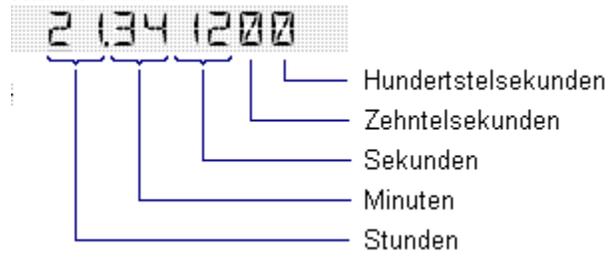
Der ttCalc verfügt über 6 trigonometrische Funktionen. Nachstehen werden die Funktionen erst für eine Ausführung über die Tastatur wie auch für die Ausführung über die Anzeige aufgelistet.

- sinus: **CLP**
- arcus sinus: **CLP** oder **ASIN**
- cosinus: **CLP**
- arcus cosinus: **CLP** oder **ACOS**
- tangens: **CLP**
- arcus tangens: **TAN⁻¹** oder **ATAN**

Siehe auch

[Winkel-Modus](#)

HMS-Format:



$$360 \text{ Altgrad} = 2\pi \text{ (RAD)} = 400 \text{ Neugrad}$$

Ausführung der Standardfunktionen

Der HP-41C™ Taschenrechner von Hewlett-Packard besitzt ca. 130 Funktionen, von denen die meisten durch das ttCalc-Programm simuliert werden.

Trotz der Vielzahl der Funktionen, werden Sie feststellen, daß alle Funktionen leicht auszuführen sind:

- Wenn Sie eine Funktionstaste drücken, und gleich loslassen, führt der Rechner die Funktion sofort aus.
- Wenn Sie die Funktionstaste nur kurz gedrückt halten, erscheint die Funktionsbezeichnung in der Anzeige. Die Funktion wird nach dem Loslassen der Taste ausgeführt.
- Wenn Sie die Funktionstaste länger als etwa eine Sekunde gedrückt halten, erscheint erst die Funktionsbezeichnung und dann **NULL** in der Anzeige. Die Funktion wird in diesem Fall nicht ausgeführt.

68 dieser Funktionen können unmittelbar durch das Drücken der Funktionstaste ausgeführt werden. Weitere 34 Funktionen stehen Ihnen nur beim ttCalc direkt oberhalb dieser Tasten, durch ein Textfeld markiert, zur Verfügung. Diese Funktionen können ebenfalls mit einem einzigen Mausklick ausgeführt werden. Die restlichen Funktionen sind auf unterschiedliche Art aufrufbar:

- Über die Anzeige: Sie drücken **CLP** und geben dann im ALPHA-Modus die Funktionsbezeichnung ein. Ein weiteres Drücken der ALPHA-Taste bewirkt anschließend die Ausführung der Funktion.
- Durch eine Tastenzuordnung im USER-Modus: Wenn Sie die Funktion mittels der **CLP** - Funktion einer Tastenposition zuordnen kann die Funktion durch einen einzigen Mausklick ausgeführt werden.
- Über die PC-Tastatur.

Sie können sich jederzeit das Funktionsverzeichnis des Rechners mit der Anweisung **CLP** **CLP** 3 ausgeben lassen.

Siehe auch

[Ausführung über die Anzeige](#)

[Tastenzuordnungen](#)

Tastenzuordnungen

Der USER-Modus gibt Ihnen die Möglichkeit, Funktionen dem Tastenfeld so zuzuordnen, wie es Ihnen beliebt. Die geschieht mit Hilfe der **[CLP]** - Funktion. Mit **[CLP]** geben sie eine Funktionsbezeichnung und die gewünschte Tastenposition an. Wenn eine Funktion einmal einer bestimmten Tastenposition zugeordnet ist, können Sie diese Funktion ausführen, indem Sie in den USER-Modus schalten und die neu zugeordnete Taste drücken. Folgende Positionen können nicht zugeordnet werden:

[CLP] ,

[ON] ,

[CLP] ,

[PRGM] , und

[CLP]. Eine Zuordnung von Zahlen und Buchstaben ist nicht möglich.

Es gibt 68 Tasten-Positionen, die neu zugeordnet werden können. Folgende Schritte müssen für eine neue Zuweisung getätigt werden:

- Drücken Sie **[CLP]**
- **[CLP]**. Der Rechner verlangt mit **ASN __** die Eingabe der Funktionsbezeichnung.
- Drücken Sie **[CLP]**, um in den ALPHA-Modus zu schalten.
- Geben Sie die gewünschte Funktionsbezeichnung ein.
- Drücken Sie **[CLP]**, um in den Normal-Modus zurückzukehren.
- Drücken Sie die Taste (oder **[CLP]** und die Taste), der Sie die Funktion zuordnen wollen.

Wenn Sie die Taste für kurze Zeit gedrückt halten, erscheint die Funktionsbezeichnung und der Tasten-Code der neuzugeordneten Funktion.

Der Tasten-Code kennzeichnet die Position mittels einer Spalten- und Zeilennummer. Tasten-Codes für umgeschaltete Tasten führen ein "-" als Präfix. Die Position -35 entspricht demnach dem Funktions-Code **[CLP]** (umgeschaltete **[CLP]**-Taste).

Siehe auch

[Löschen von Zuordnungen](#)

[Verwendung von Zuordnungen](#)

Verwendung von Zuordnungen

Jede Funktion, die Sie einer neuen Tastenposition zugeordnet haben, kann verwendet werden, wenn der Rechner in den USER-Modus geschaltet wird. Sobald Sie **CLP** drücken, werden alle Funktionen, die Sie dem Tastenfeld zugeordnet haben, wirksam.

Siehe auch

[Zuordnungen](#)

[Löschen von Zuordnungen](#)

Löschen von Zuordnungen

Wenn Sie die Zuordnung einer Funktion zu einer Taste im USER-Modus rückgängig machen wollen, drücken Sie **CLP**

CLP

CLP

CLP und die entsprechende Taste.

Siehe auch

[Zuordnungen](#)

[Verwendung von Zuordnungen](#)

Ausführung einer Funktion über die Anzeige

Beispiel: Berechnung der Fakultät von 6. **CLP** ist eine der Funktionen die nicht über die Standardtastatur aufrufbar sind. Erst tasten Sie die Zahl 6 ein und drücken dann **CLP**. Der Rechner zeigt dann XEQ __ in der Anzeige:

Tastenfolge	Anzeige	
6	6_	Die Zahl.
XEQ	XEQ __	Der ttCalc verlangt die zur Ausführung erforderliche 2-stellige Eingabe.

Jetzt müssen Sie noch die Bezeichnung der ausführenden Funktion in die Anzeige schreiben. Zuerst verlangt der Rechner mit __ (zwei Unterstreichungszeichen) eine zweistellige numerische Marke. Sobald Sie jedoch **CLP** drücken, wechselt die Anzeige zu einem einzigen Unterstreichungszeichen. Geben Sie nun die gewünschten Zeichen ein. Fehlerhaft eingegebene Zeichen können mit der **CLP** - Taste korrigiert werden.

Tastenfolge	Anzeige	
ALPHA	XEQ _	Schaltet den Rechner in den ALPHA-Modus.
FACT	XEQ FACT_	Eingabe der F, A, C, und T Zeichen.
ALPHA	720.0000	Die Funktion in der Anzeige wird ausgeführt, sobald der ALPHA-Modus beendet ist. Das Ergebnis steht im angezeigten X-Register.

Beachten Sie, daß der Inhalt des ALPHA-Registers nicht verändert wird wenn Sie eine Funktion über die **CLP** - Anweisung aufrufen. Es folgt ein Beispiel für eine Funktion, die eine Eingabe erfordert, **CLP** 6:

Tastenfolge	Anzeige	
XEQ	XEQ __	Der ttCalc verlangt die zur Ausführung erforderliche 2-stellige Eingabe.
ALPHA	XEQ _	Schaltet den Rechner in den ALPHA-Modus.
FIX	XEQ FIX_	Eingabe der F, A, C, und T Zeichen.
ALPHA	FIX _	Sie werden zur Eingabe der gewünschten Stellenzahl aufgefordert.
6	720.0000	Die Funktion CLP wird ausgeführt.

Jede Funktion, die eine Eingabe erfordert, wird ausgeführt, wenn Sie die zuletzt geforderte Stelle angeben. **CLP** benötigt eine Stelle, wird daher sofort ausgeführt wenn eine Stelle eingegeben wird. Andere Anweisungen benötigen mehrere Stellen und werden aufgerufen, wenn die zuletztbenötigte Stelle eingegeben wird.

Sonstige Funktionen

ADV	<u>Papiervorschub</u>
ON	<u>Stromeinschaltung</u>
OFF	<u>Stromausschaltung</u>
PRGM	<u>PRGM-Modus</u>
CLP	<u>ALPHA-Modus</u>
CLP	<u>USER-Modus</u>

Allgemeine mathematische Funktionen

ABS	<u>Absolutwert</u>
FRC	<u>Dezimalteil einer Zahl</u>
CLP	<u>Fakultät</u>
INT	<u>Ganzzahliger Anteil einer Zahl</u>
π	<u>Kreiszahl Pi</u>
MOD	<u>Modulo-Function (Rest)</u>
$\%$	<u>Prozent</u>
CLP	<u>Prozentuale Unterschiede</u>
\sqrt{x}	<u>Quadratwurzel</u>
x^2	<u>Quadrieren</u>
CLP	<u>Reziprokwert</u>
RND	<u>Rundung einer Zahl</u>
SIGN	<u>Vorzeichen von X</u>
CHS	<u>Vorzeichenwechsel einer Zahl</u>

CLP **Papieranschub**

Die Funktion führt beim Taschenrechner von Hewlett-Packard einen Zeilenvorschub beim Drucker aus, wenn dieser angeschlossen ist.

Die Funktion wird beim ttCalc nicht simuliert.

CLP Kopieren eines ROM-Programms

Die ROM-Routinen des HP-41CV™ von Hewlett-Packard sind in dieser Version von ttCalc nicht implementiert. Daher führt **CLP**-Funktion im ttCalc-Programm keine Operation aus.

CLP Stromeinschaltung

Die **CLP**-Funktion wird bei dem ttCalc nicht simuliert.

Die **CLP**-Taste führt zum Beenden der Applikation.

OFF **Stromausschaltung**

Die Funktion führt beim Taschenrechner von Hewlett-Packard zum Ausschalten der Anzeige.
Die Funktion wird beim ttCalc nicht simuliert.

Die **CLP** Modus-Taste

Mit der **CLP**-Taste ermöglicht eine individuelle Konfiguration des Rechner Tastenfeldes.

In diesem Modus werden die Funktionen wirksam, die Sie mittels der Zuordnungsfunktion **CLP** dem Tastenfeld zugeordnet haben. Um zu signalisieren, daß sich der Rechner in diesem Modus befindet, erscheint der USER-Indikator in der Anzeige. Um den USER-Modus wieder zu beenden, drücken Sie einfach ein zweites mal **CLP**. Dann befindet sich der Rechner wieder im Normal-Modus, und alle Tasten wieder ihre normale Funktion. Die im Normal-Modus wirksamen Funktionen sind auf der Tastenoberfläche und oberhalb dieser Taste angegeben.

Die **[CLP]** Modus-Taste

Mit dieser Taste wird der Programm-Modus ein- und ausgeschaltet. Wenn sich der Rechner im PRGM-Modus befindet werden Tastenfolgen als Befehle innerhalb des Programm gespeichert.

[CLP] kann nicht Bestandteil eines Programms sein.

Die **CLP** Modus-Taste

Die **CLP**-Taste ist eine interessante Eigenschaft des Rechners, die die Verwendung von alphanumerischen Zeichen und einigen Sonderzeichen zulässt.

Wenn Sie den ALPHA-Modus mit **CLP** einschalten, werden die Tastenfunktionen wirksam, die sich auf der abgeschrägten Vorder-Seite befinden. Zusätzlich erscheint der ALPHA-Indikator in der Anzeige. Sie können den ALPHA-Modus wieder beenden, indem Sie nochmals **CLP** drücken.

Mit den **AON** - und

AOFF - Funktionen wird der ALPHA-Modus ein- und ausgeschaltet. Als programmierbare Anweisungen haben diese Anweisungen dieselbe Funktion wie

CLP.

CHS Vorzeichenwechsel (Change Sign)

Drücken Sie zur Eingabe einer negativen Zahl zuerst die Zifferntasten für die Zahl selber und anschließend **CHS**. Die Zahl wird jetzt mit einem vorangestellten Minuszeichen "-" dargestellt.

Tastenfolge	Anzeige	
2.54	2.54_	<i>Die Zahl.</i>
CHS	-2.54_	<i>Das Vorzeichen wechselt.</i>
CHS	2.54_	<i>Das Vorzeichen wechselt erneut.</i>

Um das Vorzeichen des Exponenten einer Zahl zu ändern, müssen Sie **CHS** unmittelbar nach Eingabe des Exponenten (d.h. vor einer Operation, die die Zifferneingabe beendet) drücken. Sobald die Zifferneingabe beendet ist bewirkt **CHS** ein Vorzeichenwechsel der Mantisse und nicht des Exponenten.

RND Rundung einer Zahl

Mit der Änderung des Anzeigeformatates (**CLP**),
CLP , oder

CLP) wird die Genauigkeit der internen Darstellung der Zahlen nicht beeinflusst. Alle Zahlenwerte sind dort mit zehn Stellen in der Mantisse und einem zweistelligen Exponenten gespeichert. Es ist daher unbedeutend wieviele Ziffern angezeigt werden. Wenn sie jedoch die

RND Funktion ausführen, übernimmt der Rechner die Zahl in der Form, wie sie in der Anzeige erscheint.

Die Rundung der Rydbergkonstante auf 2 Nachkommastellen ergibt im **CLP** Format:

Tastenfolge	Anzeige	
1.0973731	1.0973731_	<i>Die Mantisse im Register X.</i>
CLP 7	1.0973731 7_	<i>Die Rydbergkonstante.</i>
CLP 2	1.10 07	<i>Das Anzeigeformat. Der interne Wert bleibt mit voller Genauigkeit erhalten.</i>
CLP	XEQ _ _	<i>Der ttCalc verlangt die zur Ausführung erforderliche 2-stellige Eingabe.</i>
CLP RND CLP	1.10 07	<i>Ausführung der RND - Funktion.</i>
CLP CLP 6	1,100000 07	<i>Das CLP 6 Anzeigeformat demonstriert die Rundung</i>

Siehe auch

Festkommaformat

Wissenschaftliches Anzeigeformat

Technisches Anzeigeformat

Zahlenanzeigesteuerflags

CLP Absolutwert

Diese Funktion berechnet den Absolutwert oder Betrag der Zahl, die im X-Register steht. Um beispielsweise den Absolutwert der Zahl -3 zu berechnen, müssen Sie folgende Anweisungen ausführen :

Tastenfolge	Anzeige	
3 CHS	-3_	
CLP	XEQ __	
CLP ABS	3.0000	<i>abs(-3)</i>
CLP		

INT Ganzzahliger Anteil einer Zahl

CLP Dezimalteil einer Zahl

Die Anweisung **INT** dient dazu den ganzzahligen Anteil einer Zahl im X-Register zu bestimmen (integer = ganzzahlig). Um beispielsweise von der Zahl 9876.5432 den Dezimalteil abzuschneiden:

Tastenfolge	Anzeige	
9876.5432	9876.5432	
CLP	XEQ __	
CLP INT CLP	9876.0000	<i>Es verbleibt nur der ganzzahlige Anteil</i>

Wenn sie **INT** ausführen, geht der Dezimalteil der Zahl verloren. Sie können allerdings die vollständige Zahl dem LAST X - Register entnehmen.

Um nur den Dezimalteil der Zahl im X-Register zu erhalten, führen Sie **CLP** (gebrochener Anteil) aus. Um beispielsweise nur den Dezimalteil der zuvor verwendeten Zahl zu erhalten:

Tastenfolge	Anzeige	
CLP LASTX	9876.5432	<i>Ruft den ursprünglichen Wert der Zahl in das X-Register zurück</i>
CLP	XEQ __	
CLP FRC	9876.0000	<i>Es verbleibt nur der Dezimalteil der Zahl</i>
CLP		

Wenn sie **CLP** ausführen, geht der ganzzahlige Anteil der Zahl verloren. Sie können allerdings die vollständige Zahl ebenfalls dem LAST X - Register entnehmen.

CLP Modulo-Funktion (Rest)

Bei einer **CLP** Anweisung bestimmt der Rechner y modulo x . Wenn Sie zwei Zahlen in die X- und Y-Register eingeben, wird y durch x geteilt und der Rest der Division im X-Register gespeichert.

Zum Beispiel ergibt 127 modulo 13:

Tastenfolge	Anzeige	
127 ENTER	128.0000	<i>Der y-Wert.</i>
13	13	<i>Der x-Wert.</i>
CLP	XEQ _ _	
CLP MOD	10.0000	<i>Das Ergebnis steht in X.</i>
CLP		

Soll y modulo x bei $x=0$ ermittelt werden, ist das Ergebnis y .

CLP Reziprokwert

Diese Funktion berechnet den Reziprowert der Zahl aus dem X-Register. Als Beispiel ist der Reziprokwert von 1.7588028×10^{11} (das Ladung/Masse- Verhältnis eines Elektrons):

Tastenfolge	Anzeige	
1.7588028	1.7588028	<i>Die Zahl.</i>
CLP 11	11	
CLP	5.56857	<i>- Der Reziprokwert.</i>
	12	

CLP Fakultät

Diese Funktion kann zu Berechnungen in der Wahrscheinlichkeitstheorie verwendet werden. Als Beispiel sei die Wahrscheinlichkeit angeführt im Lotto einen Volltreffer zu erzielen, das heißt 6 richtige aus 49 Zahlen anzukreuzen. Die allgemeine Formel für eine Kombination von n Elementen zur k-ten Klasse lautet : $C(n,k) = n! / [k! (n-k)!]$.

Tastenfolge	Anzeige	
49 CLP	6.0828	<i>Facultät von 49.</i>
	62	
43 CLP	6.0415	<i>Facultät von 43 (49-6).</i>
	52	
6 CLP	720.0000	<i>Facultät von 6.</i>
CLP CLP	13,983,816.00	<i>Kombination C(49,6). Dies entspricht der Anzahl der verschiedenen Möglichkeiten einen Tippzettel auszufüllen.</i>
CLP	7.1511	<i>- Die Wahrscheinlichkeit für 6 Richtige im Lotto</i>
	08	

Wird die **CLP** - Funktion auf Zahlen größer als 69 angewendet, erscheint in der Anzeige **OUT OF RANGE**.

CLP

SQRT Quadratwurzel

Die Anweisung **CLP** dient dazu die Quadratwurzel einer Zahl im X-Register zu bestimmen. Auf der Tastatur ist die Funktion als

CLP bezeichnet. Wenn Sie dagegen die Funktion über die Anzeige ausführen, oder einer Taste zuordnen heißt die Funktionsbezeichnung

SQRT.

Die Berechnung der Quadratwurzel von 16 (mit **CLP**) und anschließend der Wurzel des Ergebnisses (mit

SQRT) ist damit:

Tastenfolge	Anzeige
16	<i>16_</i>
CLP	<i>4.0000</i>
CLP	<i>XEQ _ _</i>
CLP SQRT	
CLP	<i>2.0000</i>

CLP

X² Quadrieren

Um eine Zahl im X-Register ins Quadrat zu erheben, führen Sie die **CLP** - Funktion aus. Auf der Tastatur ist die Funktion als

CLP bezeichnet. Wenn Sie dagegen die Funktion über die Anzeige ausführen, oder einer Taste zuordnen heißt die Funktionsbezeichnung

X².

Die Berechnung der Quadrate von 16 (mit **CLP**) und anschließend des Quadrates des Ergebnisses (mit

X²) ist damit:

Tastenfolge	Anzeige	
29 CLP CLP	729.0000	Quadrieren über die Tastatur.
CLP	XEQ _ _	Quadrieren über die Anzeige.
CLP X CLP ^ CLP 2		Der Hochpfeil (^) ist im ALPHA-Modus die umgeschaltete Funktion der N-Taste (ENTER!).
CLP	5431,441.0000	

CLP

PI Verwendung der Kreiszahl PI

Der Wert der Kreiskonstante ist im ttCalc auf 10 Stellen genau festgelegt (3.141592654).

Ausführung über die Tastatur: **CLP**

CLP

Ausführung über die Anzeige: **CLP**

PI oder

CLP

CLP PI

CLP

CLP Prozent

Die **CLP** - Funktion ist eine Anweisung von zwei Variablen. Wenn Sie berechnen wollen, wie groß ein gegebener Prozentsatz ist:

- Tasten Sie die Grundzahl ein.
- Drücken Sie **CLP**.
- Tasten Sie die Prozentzahl ein.
- Drücken Sie **CLP**.

Wenn Sie **CLP** ausführen, ändert sich der Stackinhalt wie folgt:

	Vorher	CLP	Nachher	
T	<i>4.0000</i>		<i>4.0000</i>	
Z	<i>3.0000</i>		<i>3.0000</i>	
X	<i>500.0000</i>	→	<i>500.0000</i>	Grundzahl.
Y	<i>94.0000</i>	→	<i>470.0000</i>	Ergebnis.

Beachten Sie, daß die Prozentzahl im X-Register überschrieben wird, während die Grundzahl im Y-Register erhalten bleibt.

CLP Berechnung prozentualer Unterschiede

Die **CLP** - Funktion (prozentuale Änderung oder Differenz) dient zur Berechnung prozentualer Unterschiede und ist eine Funktion von zwei Variablen. Dieser wird berechnet mit: $\%CH = [(x - y) * 100] / y$. Zur Ausführung verfahren Sie wie folgt:

- Geben Sie die Basiszahl ein.
- Drücken Sie **CLP**.
- Geben Sie die zweite Zahl ein.
- Führen Sie die **CLP** - Funktion über die Anzeige oder die Tastatur aus.

Der Stackinhalt verhält sich bei dieser Funktion analog zur Prozentfunktion.

CLP Vorzeichen von X

Mit dieser Funktion wird abhängig vom Wert im X-Register eine 0,-1 oder 1 in das X-Register geschrieben.

- Wenn der Wert in X aus ALPHA-Zeichen besteht, wird mit **CLP** eine 0 nach X geschrieben.
- Wenn der Wert in X kleiner als Null (negativ) ist , wird mit **CLP** eine -1 nach X geschrieben.
- Wenn der Wert in X größer oder gleich Null ist , wird mit **CLP** eine 1 nach X geschrieben.

Der ursprüngliche Wert von X bleibt im LASTX - Register erhalten.

Alpha-Register

Wenn sich der Rechner im ALPHA-Modus befindet erscheint nicht die Zahl, die im X-Register steht, in der Anzeige, sondern die eingetasteten alphanumerischen Zeichen. Die Eingabe von Zeichen in diesem Modus beeinflusst den Stack nicht.

Das ALPHA-Register kann bis zu 24 beliebige Zeichen aufnehmen. Sobald das 24. Zeichen eingetastet ist, ertönt der PC-Lautsprecher. Dies gilt als Warnung, daß das am weitesten links stehende Zeichen bei der nächsten Zeicheneingabe verloren geht.

Bei der Eingabe eines Strings, der aufgrund seiner Länge nicht ganz in der Anzeige dargestellt werden kann, werden die verbleibenden Zeichen nach und nach, automatisch durch die Anzeige verschoben.

Mit der **APPEND** - Funktion (

CLP K im ALPHA-Modus) können sie weitere Zeichen an eine bestehende Zeichenkette im ALPHA-Register anfügen. Wenn das Unterstreichungszeichen nicht sichtbar ist, geht dagegen bei der Eingabe eines neuen Zeichens der alte Inhalt des ALPHA-Registers verloren.

CLA Löschen des Alpha-Registers

Der Inhalt des ALPHA-Registers wird gelöscht, in dem Sie im ALPHA-Modus **CLP** **CLA** drücken. Der Inhalt der Stackregister wird durch diese Funktion nicht beeinflusst. Wenn Sie dagegen diese Tastenkombination im Normal-Modus des Rechners verwenden wird der Inhalt des X-Registers gelöscht.

CLP Löschen des X-Registers

Der Inhalt des X-Registers wird gelöscht, in dem Sie im Normal-Modus **CLP** drücken. Wenn die Eingabe der Zahl bereits abgeschlossen ist (in der Anzeige ist dann kein Unterstreichungszeichen mehr sichtbar), können sie auch die Korrekturtaste **CLP** zur Ausführung der Funktion verwenden.
Diese Funktion verändert auch die Stacklift-Bedingungen des Rechners.

CLP Korrektur der Eingabe

Die **CLP**-Taste ermöglicht es Ihnen bei einer fehlerhaften Eingabe, die zuletzt getätigte Eingabe rückgängig zu machen. Sie können jederzeit entweder falsch eingegeben Ziffern, Alpha-Zeichen oder Ergebniswerte von vorausgehenden Berechnungen mit dieser Funktion löschen.

Folgt auf das Unterstreichungszeichen auf eine Zeichenkette oder eine Ziffernfolge in der Anzeige, kann mit **CLP** der Inhalt der Anzeige Zeichen für Zeichen gelöscht werden. Fehlt das Unterstreichungszeichen dagegen, wird dieser Funktion die Anzeige gelöscht.

Im PRGM-Modus wird die Zeile, die sich in der Anzeige befindet, aus dem Programmspeicher gelöscht.

Wenn im Display eine Fehlermeldung angezeigt, wird löscht **CLP** nur diese Fehlermeldung. Der ursprüngliche Inhalt des X-Registers oder des Alpha-Registers geht nicht verloren.

Umordnen der Stackinhalte

Die Funktionen **R↓** und

R↑ ermöglichen es Ihnen, die Inhalte aller vier Stackregister in die Anzeige zu bringen, oder für folgende Berechnungen die Stackregister umzuordnen.

- **R↓** führt eine zyklische Vertauschung nach unten durch. Der ursprüngliche Inhalt des X-Registers steht im Anschluß an diese Funktion im T-Register. Alle anderen Register werden um ein Register angehoben, so daß in der Anzeige der ursprüngliche Inhalt des Y-Registers erscheint. Wenn Sie diese Funktion über die Anzeige ausführen, lautet die Funktionsbezeichnung

RDN.

- Analog dazu führt **R↑** eine zyklische Vertauschung nach oben durch. Der ursprüngliche Inhalt des T-Registers steht im Anschluß an diese Funktion im X-Register und damit in der Anzeige. Alle anderen Register werden um ein Register erniedrigt, so daß sich der Inhalt des X-Registers anschließend im Y-Register befindet.

Durch viermaliges Ausführen dieser Funktionen können Sie den Inhalt aller vier Register betrachten, und der Stackinhalt befindet sich wieder im ursprünglichen Zustand.

Austausch von X- und Y-Register

Die **CLP**-Funktion vertauscht den Inhalt des X- Registers mit dem Inhalt des Y-Registers, ohne daß diese Funktion einen Einfluß auf die weiteren Stackregister T und Z hat. Sie können den Inhalt des X-Register auch mit vielen anderen Registern mit der Vertauschungsfunktion **CLP** austauschen.

Die **LASTX**-Funktion

Mit dieser Funktion kopieren Sie den Inhalt des L-Registers in das X-Register.

Vor der Ausführung der meisten Funktionen (dazu zählen zum Beispiel **CLP**,

CLP,

CLP, oder

CLP) wird der ursprüngliche Inhalt des X-Registers in das L-Register kopiert.

Sie können jetzt die **LASTX**-Funktion dazu verwenden, um ein versehentliches Drücken einer Taste zu korrigieren.

Das LASTX-Register kann auch für Berechnungen verwendet werden, die Zwischenergebnisse oder Zahlen öfter als einmal benötigen, in dem Sie das Zwischenergebnis aus dem L-Register kopieren.

Löschen des Rechenregistertapels

Mit **CLST** wird der Inhalt des automatischen Rechenregister-Stapels gelöscht. Im einzelnen werden die Inhalte der X,Y,Z, und T-Register gleich Null gesetzt.

Die **CLP** - Taste

Die **CLP** -Funktion dient bei der UPN-Logik dazu, die Zifferneingabe für eine Zahl abzuschließen. Erst nach der Ausführung dieser Funktion ist die Eingabe einer weiteren Zahl möglich.

Abgesehen von dieser Eigenschaft wird mit **CLP** auch die eingegebene Zahl im Rechenregister-Stapel nach oben verschoben. Im einzelnen findet folgendes statt, wenn sie diese Funktion eintasten:

Tastenfolge	Anzeige	
4 ENTER†	4.0000	
3 ENTER†	3.0000	Die ursprüngliche Zahl im X-Register (4) wird in das Y-Register verschoben.
2 ENTER†	2.0000	Die neue Zahl im X-Register (3) wird in das Y-Register verschoben.
1 ENTER†	1.0000	Die neue Zahl im X-Register (1) wird in das Y-Register verschoben.
987.3	987.3_	Eintasten einer weiteren Zahl.
CLP	987.3000	Schließt die Eingabe der Zahl 987.3 ab.

Bei der Eingabe der Zahl 987.3 hat sich der Stackinhalt wie folgt verändert:

	Vorher CLP	Nachher
T	4.0000	3.0000
Z	3.0000	2.0000
Y	2.0000	1.0000
X	1.0000	987.3_

Mit der Ausführung von **CLP** wird die eingetaste Zahl auch in das Y-Register kopiert: Die Inhalte der Y- und Z-Register werden entsprechend um eine Position nach oben verschoben. Der Inhalt des T-Registers geht verloren:

	Vorher CLP	Nachher
T	3.0000	2.0000
Z	2.0000	1.0000
Y	1.0000	987.3000
X	987.3_ →	987.3000

Im Anschluß an das Drücken der **CLP** -Taste ist das X-Register für die Eingabe einer neuen Zahl vorbereitet, die den alten Wert im X-Register überschreibt. Wenn Sie jetzt die Zahl 573.04 eintasten, ohne im Anschluß **CLP** zu drücken, ändert sich der Inhalt wie folgt:

	Vorher	573.04	Nachher
T	2.0000		2.0000
Z	1.0000		1.0000

Y	987.3000	987.3000
X	987.3000	573.04_

Beachten Sie, daß die Inhalte der Stackregister nicht verschoben werden, wenn die Eingabe einer Zahl unmittelbar auf **CLP** oder **CLP** folgt. Die meisten übrigen Funktionen bereiten dagegen einen Stacklift vor.

Indirekte Operationen

Die meisten Funktionen, die eine ein- oder mehrstellige Eingabe als Tastenanweisung erwarten, erlauben auch die Möglichkeit diese Funktion als indirekte Operation auszuführen. Jedes Stack- und Datenspeicher-Register kann für indirekte Operationen verwendet werden. Sie können indirekte Operationen ausführen, in dem Sie nach der Funktionstaste die Umschalttaste **CLP** und die Registeradresse angeben. Die Funktion verwendet dann den Inhalt des angegebenen Registers als Adresse.

- Die indirekte Adressierung kann auf jedes der Primär- oder der erweiterten Speicherregister angewendet werden, die augenblicklich dem Datenspeicher zugeordnet sind.
- Alle Datenspeicher oberhalb von Register 99 können nur mit der indirekten Adressierung angesprochen werden.
- Wenn die direkte Adresse kein Stackregister ist und nicht im Bereich des aktuellen Datenblocks liegt, erscheint die Fehlermeldung **NONEXISTENT** in der Anzeige.
- In allen Fällen wird nur der ganzzahlige absolute Anteil, der im Register gespeicherten Zahl, als Adresse für die indirekte Adressierung verwendet.

Siehe auch

[Indirektes Speichern und Zurückrufen von Daten](#)

[Indirektes Speichern und Zurückrufen von Alpha-Ketten](#)

[Indirekte Steuerung von Funktionen \(Beispielprogramm\)](#)

[Indirekte Steuerung von Programmverzweigungen und Unterprogrammen](#)

[Liste aller indirekten Funktionen](#)

[Meldungen und Fehler](#)

Indirektes Speichern und Zurückrufen von Daten

Daten können unter Verwendung eines Speicherregisters in einem beliebigen Stack- oder Daten-Speicherregister indirekt gespeichert und wieder zurückgerufen werden. Sie müssen dazu die Funktion **CLP** oder

CLP ausführen, und dann **CLP** und die indirekte Adresse eingeben.

Um beispielsweise die Zahl 2.54 in R₁₀ abzuspeichern, wobei R₀₂ als indirektes Adressregister Verwendung findet, müssen Sie folgendes eintasten:

Tastenfolge	Anzeige	
10 STO 02	10.0000_	Speichern der gewünschten Registeradresse (R ₀₂) in das indirekte Adreßregister (R ₁₀).
2.54	2.54_	Die Zahl.
STO CLP	STO IND _ _	Die Funktion in der Anzeige wird ausgeführt, sobald der ALPHA-Modus beendet ist. Das Ergebnis steht im angezeigten X-Register.
02	2.5400	Die Zahl 2.54 steht jetzt im Register 10.

Analog dazu können Sie das Adreßregister R₀₅ verwenden, um den Inhalt von R₁₀ zurückzurufen:

Tastenfolge	Anzeige	
10 STO 05	10.0000_	Speichern der gewünschten Registeradresse (R ₀₂) in das indirekte Adreßregister (R ₁₀).
RCL CLP	RCL IND _ _	Der Rechner verlangt die zur Ausführung benötigte indirekte Adresse.
05	2.5400	Die Zahl 2.54 steht jetzt wieder im X-Register und in der Anzeige.

Register-Arithmetik wird mit dem Inhalt indirekt adressierter Register wird mit den folgenden Funktionen ausgeführt:

CLP
CLP
CLP nn. **CLP**
CLP
CLP nn. **CLP**
CLP
CLP nn. **CLP**
CLP

CLP nn.

Unter Verwendung von Register R₁₁ indirektem Adreßregister können Sie den Inhalt von Register R₁₀ mit 5280 multiplizieren, indem Sie folgendes durchführen:

Tastenfolge	Anzeige	
10 STO 11	10.0000	
5280	5,280_	
STO x CLP	ST × IND _ _	<i>Der Rechner verlangt die zur Ausführung benötigte indirekte Adresse.</i>
11	5,280.0000	<i>5280 wird mit der Zahl in Register 10 multipliziert.</i>
STO 10	13,411.2000	<i>Das Ergebnis.</i>

Indirektes Speichern und Zurückrufen von Alpha-Ketten

Die Funktionen **ASTO** und

ARCL können wie

CLP und

CLP in Verbindung mit indirekter Adressierung verwendet werden.

Speichern Sie zur Ausführung zuerst die gewünschte Adresse im entsprechenden indirekten Adreßregister; führen Sie die Funktion aus, und geben Sie **CLP** und das indirekte Adreßregister an.

Speichern Sie beispielsweise die Zeichenkette **WASSER** nach R₀₈, wobei Sie R₀₀ als indirektes Adressregister verwenden:

Tastenfolge	Anzeige	
8 STO 00	8.0000	
CLP WATER	WASSER_	
CLP ASTO	ASTO IND __	
00	WASSER	Die Kette WASSER steht jetzt im Register 08.
CLP CLA		

Analog dazu können Sie die Kette mittels indirekter Adressierung zurückrufen:

Tastenfolge	Anzeige	
CLP ARCL CLP	ARCL IND __	
00	WASSER	Die Zeichenkette WASSER aus Register 08 wird in das Alpha-Register zurückgerufen.
CLP CLA		Löschen des Alpha-Registers.
CLP	8.0000	Rückkehr in den Normal-Modus.

Indirekte Steuerung von Funktionen

Dieses Beispiel zeigt die Verwendung der indirekten Adressierung anhand der **CLP**-Funktion. Das Programm zählt von 0 bis 9 und steuert die erste Schleife mit **ISG** und zählt dann wieder bis 0 zurück, woei die **DSE**-Anweisung verwendet wird.

Tastenfolge	Anzeige	
PRGM		
CLP GTO ▢ ▢	00 REG 46	
CLP LBL CLP		
SONG CLP	01 LBL"SONG	
.009	02 .009_	
STO 01	03 STO 01	Die erste Schleifensteuerzahl wird im Register 01 gespeichert.
9	04 9_	
STO 02	05 STO 02	Die zweite Schleifensteuerzahl wird im Register 02 gespeichert.
CLP LBL 01	06 LBL 01	Der Anfang der Schleife.
CLP CLP TONE		
CLP		
CLP 01	07 TONE IND 01	TONE verwendet R ₀₁ als indirekte Adresse und um die Tonhöhe zu steuern.
CLP ISG 01	08 ISG 01	Addiert 1 zu der Schleifensteuerzahl in R ₀₁ . Die Zahl wird geprüft, ob sie kleiner als 9 ist. Wenn nicht wird die nächste Zeile übersprungen, sonst wird die Schleife wiederholt.
CLP GTO 01	09 GTO 01	Springt zum Schleifenbeginn.
CLP LBL 02	08 LBL 02	Der Anfang der zweiten Schleife.
TONE CLP 02	09 TONE IND 02	TONE verwendet R ₀₂ als indirekte Adresse und um die Tonhöhe zu steuern.
CLP CLP DSE		
CLP 02	09 DSE 02	Subtrahiert 1 von der Zahl in R ₀₂ . Die Zahl wird geprüft, ob sie kleiner oder gleich 0 ist. Wenn nicht wird die nächste Zeile übersprungen, sonst wird die Schleife wiederholt.
CLP GTO 01	09 GTO 02	Springt zum Beginn der zweiten Schleife.
CLP GTO ▢ ▢	00 REG 42	

Wenn Sie jetzt das Programm ausführen, werden Sie feststellen, daß die akustischen Töne zu Beginn tief sind, dann immer höher werden und anschließend die Tonhöhe wieder abfällt.

Tastenfolge Anzeige

PRGM	0.0000
CLP CLP SONG	
CLP	0.0000

Indirekte Steuerung von Programmverzweigungen und Unterprogrammen

Die Technik der indirekten Adressierung bei der Verwendung von Speicherregister können Sie in ähnlicher Weise verwenden, um ganze Tastenfolgen, Unterprogramme und komplette Programme indirekt auszuführen.

Um ein Unterprogramm mit einer ALPHA-Marke oder einer numerischen Marke indirekt zu adressieren, verwenden Sie **CLP**

CLP nn (indirekter Sprung) innerhalb des Programms. Wenn während der Programmausführung die Anweisung **GTO IND nn** erreicht wird, sucht der Rechner den Programmspeicher erst bis zum Ende des Programms und dann von Anfang des Programms bis zur Ausgangsstellung nach der im indirekten Adreßregister angegebenen Marke ab. Wenn die Marke nicht gefunden wird, oder diese Marke nicht zulässig ist, wie z.B. eine Marke größer als 99, erscheint **NONEXISTENT** in der Anzeige. Lokale Marken (A,B, ... J und a,b, ... e) können nicht indirekt adressiert werden.

Unterprogramme und Routinen außerhalb des augenblicklichen Programms werden mit **CLP** **CLP** nn (indirekte Programmausführung) indirekt adressiert. Wenn während der Programmausführung die Anweisung **XEQ IND nn** erreicht wird, verzweigt der Rechner zu der ALPHA-Marke oder der numerischen Marke, die in dem indirekten Adreßregister enthalten ist. Dieses so adressierte Programm wird als Unterprogramm ausgeführt. Nach Beendigung des Unterprogramms wird das Hauptprogramm fortgesetzt. Beachten Sie, daß nur Programme die im Hauptspeicher stehen, auf diese Weise ausgeführt werden können. Die

Standardfunktionen können mit **CLP**

CLP nicht ausgeführt werden.

Liste aller indirekten Funktionen

Folgende Funktionen können bei der indirekten Adressierung verwendet werden:

CLP CLP nn	Speichern.
CLP CLP CLP nn	Speichern mit addieren.
ST+ CLP nn	
CLP CLP CLP nn	Speichern mit subtrahieren.
ST- CLP nn	
CLP CLP CLP nn	Speichern mit multiplizieren.
ST× CLP nn	
CLP CLP CLP nn	Speichern mit dividieren.
ST÷ CLP nn	
CLP CLP nn	Zurückrufen.
CLP CLP nn	Alpha-Ketten speichern.
CLP CLP nn	Alpha-Ketten zurückrufen.
VIEW CLP nn	Anzeige eines Register-Inhalts.
CLP CLP nn	Programmverzweigung.
CLP CLP nn	Unterprogrammausführung.
CLP CLP nn	FIX-Anzeigeform.
CLP CLP nn	SCI-Anzeigeform.
CLP CLP nn	ENG-Anzeigeform.
DSE CLP nn	Schleifensteuerung mit Dekrement.
CLP CLP nn	Schleifensteuerung mit Inkrement.
CLP CLP nn	Tonhöhe.
CLP CLP nn	Definition des Statistik-Blocks.
CLP CLP nn	Flag setzen.
CLP CLP nn	Flag löschen.
CLP CLP nn	Flag-Abfrage ob gesetzt.
CLP CLP nn	Flag-Abfrage ob gelöscht.
CLP CLP nn	Flag-Abfrage ob gesetzt und löschen.
CLP CLP nn	Flag-Abfrage ob gelöscht und löschen.
CLP CLP nn	Austauschs von X mit beliebigem Register.
CLP CLP nn	Katalog-Funktion

Programmierung

Ein Programm besteht aus einer Folge von Tastendrücken, die Sie im Fall einer manuellen Lösung des Problems auch vom Tastenfeld aus drücken würden. Der Rechner speichert diese Tastenfolge, und führt diese Tastenfolgen aus, wenn Sie das Programm starten. Wegen der Eigenschaften dieses Rechners können selbstgeschriebene Programme wie jede andere Standardfunktionen behandelt werden.

Der Rechner ttCalc besitzt folgende programmtechnische Merkmale:

- Maximale Anzahl an Programmzeilen: 2240
- Frei definierbare Tasten für lokale Marken: 15.
- Beliebig viele alphanumerische Marken.
- Bis zu 127 numerische Marken innerhalb eines Programms.
- Programmdurchsicht durch zeilenweises Durchlaufen (Vorwärts/Rückwärts).
- Einfügen und Löschen von Zeilen und ganzen Programmen.
- Vertgleichsoperationen: 10
- Flags: 56
- Bedingte und unbedingte Programmsprünge.
- Unterprogrammebenen: 6
- Schleifensteuerung
- Indirekte Steuerung von Datenspeicherung, Datenrückruf und Programmausführung.
- Speicherregisterarithmetik.
- USER-Modus und ALPHA-Modus.
- Automatische Programm-Speicherverwaltung.

usw.....

Siehe auch

[Eingabe eines Programms](#)

[Ausführung eines Programms](#)

[Korrekturmöglichkeiten](#)

[Lokale Marken](#)

[Nicht-Programmierbare Funktionen](#)

[Programmierung mit ALPHA-Ketten](#)

[Programmschleifen](#)

[Programmspeicher](#)

[Unbedingte Sprünge](#)

[Unterbrechen der Programmausführung](#)

[Unterprogramme](#)

Vergleichsoperationen

Vergleichsoperationen

Der Rechner verfügt über zehn Vergleichsoperationen:

X=Y?	Prüft, ob die Inhalte von X- und Y-Register gleich sind.
X≠Y?	Prüft, ob die Inhalte von X- und Y-Register ungleich sind.
X>Y?	Prüft, ob die Zahl im X-Register größer als die Zahl im Y-Register ist.
X<Y?	Prüft, ob die Zahl im X-Register kleiner als die Zahl im Y-Register ist.
X≤Y?	Prüft, ob die Zahl im X-Register kleiner als die oder gleich der Zahl im Y-Register ist.
X=0?	Prüft, ob die Zahl im X-Register gleich Null ist.
X≠0?	Prüft, ob die Zahl im X-Register ungleich Null ist.
X>0?	Prüft, ob die Zahl im X- Register größer als Null ist.
X<0?	Prüft, ob die Zahl im X- Register kleiner als Null ist.
X≤0?	Prüft, ob die Zahl im X- Register kleiner als oder gleich Null ist.

Die beiden ersten Vergleichsoperationen **X=Y?** und **X≠Y?** können auch auf ALPHA-Strings angewendet werden. Zwei ALPHA-Ketten sind genau dann gleich, wenn sowohl die Länge der Strings, als auch deren Inhalt gleich sind. Die restlichen Funktionen erzeugen, die Fehlermeldung **ALPHA DATA**, wenn einer der beiden Operanden keine Zahl ist.

Die Wirkungsweise der Vergleichsoperationen ist vom Systemzustand des Rechners abhängig:

- Wenn im Programmlauf die Bedingung erfüllt ist, setzt der Rechner die Ausführung mit der nächsten Programmzeile fort. Ist die Bedingung nicht erfüllt, wird die nächste Programmzeile übersprungen.
- Wird eine dieser Funktionen über das Tastenfeld ausgeführt erscheint die Antwort auf die Prüffunktion in der Anzeige. Wenn die Bedingung erfüllt ist, erscheint **YES**, ist sie nicht erfüllt, erscheint **NO**.

Die auf den Vergleichsbefehl folgende Programmzeile kann eine beliebige Programmanweisung enthalten. In der Regel steht an dieser Stelle eine **CLP**-Anweisung. Auf diese Weise wird die Programmausführung, wenn die Bedingung erfüllt ist, an einer anderen Stelle des Programms fortgesetzt.

Programmschleifen

Der Rechner verfügt über zwei Befehle mit denen Programmschleifen einfach gestaltet werden können. Diese Funktionen sind **CLP** (Inkrement und Sprung, wenn größer) und **CLP** (Dekrement und Sprung, wenn gleich).

Beide Funktionen verwenden eine Zahl, dessen Format bei der Steuerung der Programmschleifen interpretiert wird. Die Zahl, die in einem Speicherregister oder im Stack stehen kann, hat folgende Gestalt:

IIIII.FFFCC

- Der IIIII-Anteil entspricht dem Anfangswert der Laufvariablen, mit der gezählt wird, wie oft eine Schleife durchlaufen wird. Der IIIII-Anteil kann eine beliebige ein- bis fünfstellige Zahl sein.
- Der FFF-Anteil entspricht dem Endwert der Laufvariablen, bei der der Rechner aufhört zu zählen. Der FFF-Anteil muß als vorzeichenlose dreistellige Zahl angegeben werden. Wenn kein FFF-Anteil angegeben ist, hört der Rechner mit Null auf zu zählen.
- Der CC-Anteil der Zahl entspricht dem Inkrement, mit dem gezählt wird. Bei Ausführung der **CLP**- oder **CLP**-Funktion wird der IIIII-Wert um den CC-Wert erhöht oder erniedrigt. Wenn kein CC-Anteil angegeben ist, nimmt der Rechner an, daß um eins inkrementiert bzw. dekrementiert werden soll. Der CC-Wert muß als vorzeichenlose zweistellige Zahl angegeben werden.

Die **CLP**-Funktion:

Jedesmal, wenn **CLP** ausgeführt wird, erhöht sich der Wert der Laufvariablen IIIII um den Wert CC. Anschließend wird geprüft, ob IIIII größer als FFF ist. Wenn dies der Fall ist, überspringt der Rechner im Falle einer Programmausführung die nächste Programmzeile.

Die **CLP**-Funktion:

Jedesmal, wenn **CLP** ausgeführt wird, erniedrigt sich der Wert der Laufvariablen IIIII um den Wert CC. Anschließend wird geprüft, ob IIIII kleiner als oder gleich FFF ist. Wenn dies der Fall ist, überspringt der Rechner im Falle einer Programmausführung die nächste Programmzeile.

Siehe auch

[Eingabe eines Programms](#)

[Ausführung eines Programms](#)

[Korrekturmöglichkeiten](#)

[Lokale Marken](#)

[Nicht-Programmierbare Funktionen](#)

[Programmierung mit ALPHA-Ketten](#)

[Programmspeicher](#)

[Unbedingte Sprünge](#)

[Unterbrechen der Programmausführung](#)

[Unterprogramme](#)

[Vergleichsoperationen](#)

Unbedingte Sprünge

Ähnlich, wie Sie mit **CLP**

CLP eine Programmzeile oder eine ALPHA-Marke den Rechner auf eine beliebige Stelle des Programmspeicher setzen können, kann der

CLP-Befehl gefolgt von einer numerischen Marke oder einer ALPHA-Marke als Bestandteil eines Programms zur Ausführung einer Programmverzweigung an eine beliebige Stelle verwendet werden. Programmverzweigungen, die auf alle Fälle stattfinden, nennt man unbedingte Sprünge.

Die Anweisung **CLP** 01 innerhalb eines Programmlaufs bewirkt zum Beispiel folgendes:

- Der Rechner sucht sequentiell nach der ersten numerischen Marke **CLP** 01 innerhalb des augenblicklichen Programms. Wird diese Marke nicht gefunden, bricht der Rechner die Ausführung des Programms mit der Fehlermeldung **NONEXISTENT** ab (Drücken Sie die Korrektur-Taste **CLP**, um den Fehler zu beheben). Existiert die Marke wird die Ausführung mit dieser Marke fortgesetzt.
- Der Rechner führt die Suche nach lokalen Marken nur das erste Mal aus. Bei jeder folgenden Interpretation des Sprungbefehls ist der Suchvorgang nicht mehr erforderlich. Sobald nämlich die Suche nach einer numerischen Marke erfolgreich verlaufen ist, weiß der Rechner wo sich die zugehörige Marke befindet, und verzweigt unmittelbar. Diese Eigenschaft ist bei üblichen Programmiersprachen (PASCAL, FORTRAN etc.) als Kompilierung bekannt.

Siehe auch

[Eingabe eines Programms](#)

[Ausführung eines Programms](#)

[Korrekturmöglichkeiten](#)

[Lokale Marken](#)

[Nicht-Programmierbare Funktionen](#)

[Programmierung mit ALPHA-Ketten](#)

[Programmspeicher](#)

[Unterbrechen der Programmausführung](#)

[Unterprogramme](#)

[Vergleichsoperationen](#)

Unterbrechen der Programmausführung

Während eines Programmlauf ist es häufig notwendig das Programm zu unterbrechen, um entweder Daten einzugeben oder Ergebnisse in der Anzeige zu betrachten, bevor das Programm fortgesetzt wird. Sie haben dazu verschiedene Möglichkeiten zur Auswahl:

- Verwendung von **CLP** und **STOP**:

Die **STOP**-Funktion kann entweder durch das Drücken der **CLP**-Taste oder durch **CLP** mit nachfolgender Funktionsbezeichnung STOP als Programmanweisung eingegeben werden. Wird

STOP in einem Programm ausgeführt, hält das Programm in der der **STOP**-Anweisung folgenden Zeile an folgendes: Wenn ein Programm abläuft, wird die Programmausführung unterbrochen, ansonsten wird mit **CLP** ein angehaltenes Programm mit der aktuellen Zeilennummer fortgesetzt.

- Verwendung von **CLP**:

Die **CLP**-Anweisung hält, als Bestandteil eines Programms die Programmausführung für etwa eine Sekunde an. Während dieser Zeit wird der Inhalt des X-Register angezeigt. Es können mehrere

CLP-Anweisung aufeinander folgen um die Betrachtungszeit zu verlängern. Während eines Programmlaufs sind nur die Tasten

CLP und

CLP wirksam. Wenn jedoch eine

CLP-Anweisung ausgeführt wird, wird das ganze Tastenfeld wirksam.

Solange Sie während einer **CLP**-Ausführung Tasten zur Dateneingabe drücken wird die

CLP-Anweisung wiederholt ausgeführt (oder bis die Dateneingabe beendet ist). Die Tasten zur Daten-Eingabe sind

CLP,

CLP,

CLP,

CLP, 0 bis 9,

CLP,

EEX und alle ALPHA-Zeichen.

Wird dagegen irgendeine andere Taste, die keine Dateneingabe einbezieht, während einer **CLP**-Ausführung gedrückt, wird die Pause beendet und das Programm angehalten. Die der Taste entsprechende Anweisung wird ausgeführt. Mit **CLP** können Sie das Programm fortsetzen.

Siehe auch

[Eingabe eines Programms](#)

[Ausführung eines Programms](#)

[Korrekturmöglichkeiten](#)

[Lokale Marken](#)

[Nicht-Programmierbare Funktionen](#)

[Programmierung mit ALPHA-Ketten](#)

[Programmschleifen](#)

[Programmspeicher](#)

Unbedingte Sprünge
Unterprogramme
Vergleichsoperationen

Eingabe eines Programms

Angenommen Sie wollen die Flächeninhalt eines Kreises berechnen. Die Formel dazu lautet $A = \pi r^2$. Wenn Sie Berechnung manuell über das Tastenfeld durchführen, werden Sie zuerst den Radius r eingeben und diesen Wert mit **CLP**

CLP quadrieren. Als nächstes müssen Sie den Wert der Kreiskonstante π mit

CLP

CLP in die Anzeige rufen. Abschließend drücken Sie

CLP, um den quadrierten Radius mit π zu multiplizieren.

Ein Programm unterscheidet sich nicht wesentlich von einer Folge von Tastendrücker, die Sie betätigen würden, um ein Problem manuell zu lösen. Daher sind für die Erstellung eines Programms, das die Berechnung der Fläche eines beliebigen Kreises durchführt, die gleichen Anweisungen zu verwenden, die Sie sonst zur Berechnung per Hand durchgeführt hätten.

Daher wird die Tastenfolge **CLP**

CLP,

CLP

CLP und

CLP auch Bestandteil dieses Programms sein.

Darüberhinaus wird das Programm noch zwei weitere Operationen enthalten: **CLP** und **CLP**.

- Anfang eines Programms: **CLP**-Anweisung.
Der Anfang eines Programms sollte durch eine ALPHA-Zeichenkette gekennzeichnet werden. Die Verwendung dieser Programmmarken ermöglicht einen Überblick der im Programmspeicher enthaltenen Programme und vereinfacht ihre Anwendung.
- Ende eines Programms: **CLP**-Anweisung.
Durch die **CLP**-Anweisung wird das Ende des Programms definiert. Wenn der Rechner während eines Programmlaufs auf diese Anweisung trifft, wird die Ausführung beendet.

Wenn sich der Rechner im PRGM-Modus befindet, werden die Operationen und Funktionen, die normalerweise beim Drücken der entsprechenden Taste ausgeführt werden, nicht ausgeführt. Statt dessen werden diese in den Programmspeicher geschrieben und können später ausgeführt werden. Bis auf einige nicht-programmierbare Funktionen können alle Operationen als Anweisung in den Programmspeicher eingegeben werden.

Funktionen, die auf dem Tastenfeld vorhanden sind, können einfach durch das Drücken der entsprechenden Taste eingegeben werden. Funktionen, die nicht über das Tastenfeld aufrufbar sind, werden entweder erst einer Taste zugeordnet, die dann im USER-Modus gedrückt wird, oder können mit **CLP** und der Angabe des Funktionsnamens eingegeben werden. Die Vorgehensweise ist damit diesselbe, wie bei der manuellen Ausführung einer Funktion.

Um ein neues Programm einzugeben sollten Sie, falls Sie mit der Materie noch nicht vertraut sind, folgendermaßen vorgehen:

- Drücken Sie **CLP**, um den Rechner in den Programm-Modus zu schalten.
- Drücken Sie **CLP**

CLP

CLP

CLP, um den Rechner auf einen freien Teil des Programmspeichers zu setzen.

Damit lautet das Programm zur Berchnung einer beliebigen Kreisfläche :

Tastenfolge	Anzeige	
CLP	00 REG 46	Schalten Sie den Rechner in den PRGM-Modus (Die Anzeige kann je nach Rechnerkonfiguration von dieser Darstellung abweichen).
CLP CLP CLP CLP	00 REG 46	Das Programm kann jetzt in den Rechner eingetastet werden.
CLP CLP	01 LBL __	Eingabe der globalen Marke. Die in der Anzeige links stehenden Ziffern geben die Programmzeile an.
CLP CIRCLE CLP	01 LBL "CIRCLE	Drücken Sie die Alpha-Taste, und C,I,R,C,L und E, sowie die Alpha-Taste, um die Eingabe der Marke abzuschließen.
CLP 	03 X2	Es folgt die eigentliche Berechnung der Kreisfläche.
CLP 	04 PI	
CLP	05 x	Die zweite Schleifensteuerzahl wird im Register 02 gespeichert.
CLP CLP CLP CLP	00 REG 46	Diese Tastenfolge schließt das Programm ab. Es wird eine END -Anweisung in Zeile 5 angefügt, und die Anzahl der verbleibenden freien Register angezeigt.
CLP	0.0000	Der Programm-Modus wird beendet, und der Rechner zeigt wieder die Zahl im X-Register an.

Das Programm zur Berechnung des Flächeninhaltes eines Kreises ist damit im Programmspeicher eingegeben.

Siehe auch

[Ausführung eines Programms](#)
[Korrekturmöglichkeiten](#)
[Lokale Marken](#)
[Nicht-Programmierbare Funktionen](#)
[Programmierung mit ALPHA-Ketten](#)
[Programmschleifen](#)
[Programmspeicher](#)
[Unbedingte Sprünge](#)
[Unterbrechen der Programmausführung](#)
[Unterprogramme](#)
[Vergleichsoperationen](#)

Unterprogramme

Unabhängig von der Programmiersprache, sei es nun eine Hochsprache wie Pascal oder eine Tasten-Code Sprache wie bei diesem Rechner, kommt es häufig vor, daß sich bestimmte Befehle und Anweisungen wiederholen. Die Technik, solche Befehle zusammenzufassen um Programmspeicher zusammenzufassen und zugleich die Übersichtlichkeit des Programms zu erhöhen, besteht in der Ausführung von Unterprogrammen.

Bei diesem Rechner können Sie durch die Verwendung des {bmc xeq.bmp}-Befehl mit ALPHA-Marken und numerische Marken gekennzeichnete Unterprogramme aufrufen. In einem Programm wird die Ausführung mit der in der {bmc xeq.bmp}-Anweisung angegebene Programmarke fortgesetzt. Trifft der Interpreter auf ein {bmc end.bmp} oder {bmc rtn.bmp}, wird die Ausführung wieder im Hauptprogramm fortgesetzt. Dort wird dann der Befehl ausgeführt, der der {bmc xeq.bmp}-Anweisung unmittelbar folgt. Im Gegensatz dazu setzt eine {bmc gto.bmp}-Anweisung lediglich die Programmausführung mit der in der angegebenen Marke fort, ein Rücksprung in das Hauptprogramm erfolgt in diesem Fall nicht.

Weitere Details:

- Numerische Marken und lokale ALPHA-Marken (A, ... J und a, ... e) werden für Programme innerhalb eines Programmblocks verwendet. Der Rechner sucht den Programmspeicher nach diesen Programmarken nur innerhalb des augenblicklichen Programms ab: Dies geschieht so, daß ab der augenblicklichen Position der Programmspeicher bis zum ersten **CLP** nach numerischen Marken oder lokalen ALPHA-Marken untersucht wird. Anschließend wird auch noch der Bereich vom Programmanfang bis zur Ausgangsposition durchgecheckt. Ist jetzt die entsprechende Marke immer noch nicht gefunden erscheint die Fehlermeldung **NONEXISTENT** in der Anzeige. Unterprogramme werden normalerweise mit **RTN** abgeschlossen, damit ein gesamtes Programm übersichtshalber genau eine Anfangsmarke und nur eine **CLP**-Anweisung besitzt. Befindet sich das Unterprogramm am Ende eines Programms, kann es auch mit einem **CLP** abgeschlossen werden.
- ALPHA-Marken werden im allgemeinen für Programme und Unterprogramme verwendet, die sich außerhalb anderer Programme befinden. Der Rechner sucht den gesamten Programmspeicher nach ALPHA-Marken ab. Der Suchvorgang beginnt mit der letzten ALPHA-Marke, wonach aufsteigend alle ALPHA-Marken im Programmspeicher abgesucht werden. Ist die Marke nicht vorhanden erscheint **NONEXISTENT** in der Anzeige.
- Der Rechner kann sich eine maximal die Rücksprungadressen von 6 Unterprogrammen speichern. Sobald die 7. Unterprogrammebene angesprochen wird, geht die 1. Rücksprungadresse verloren. Das Programm wird damit nicht mehr zum Hauptprogramm zurückkehren können.
- Der Rechner kann sich eine maximal die Rücksprungadressen von 6 Unterprogrammen speichern. Sobald die 7. Unterprogrammebene angesprochen wird, geht die 1. Rücksprungadresse verloren. Das Programm wird damit nicht mehr zum Hauptprogramm zurückkehren können.

Beispiel: Eine quadratische Gleichung der Form ax^2+bx+c hat die beiden Lösungen $X_{1,2} = \frac{-b \pm \sqrt{b^2-4ac}}{2a}$. Da sich die beiden Lösungen nur im einen Vorzeichen unterscheiden, kann für den Teil der identisch ist ein Unterprogramm verwendet werden:

Tastenfolge

Anzeige

CLP		
CLP CLP CLP	00 REG 46	
CLP		
CLP CLP CLP		
QROOT CLP	01 LBL"QROOT	
CLP	02"a?	
CLP a? CLP		
CLP CLP		
PROMPT CLP	03 PROMPT	Textausgabe und Programmstop zur Dateneingabe.
CLP 01	04 STO 01	
CLP	05"b?	
CLP b? CLP		
CLP CLP		
PROMPT CLP	06 PROMPT	Textausgabe und Programmstop zur Dateneingabe.
CLP 01	07 STO 02	
CLP	08"c?	
CLP c? CLP		
CLP CLP		
PROMPT CLP	09 PROMPT	Textausgabe und Programmstop zur Dateneingabe.
CLP 01	10 STO 03	
CLP 01	11 XEQ 01	Hier beginnt die Routine zur Berechnung der ersten Lösung.
□	12 -	
CLP 01	13 RCL 01	
2	14 2_	
CLP	15 ×	
÷	16 /	
CLP CLP PSE		
CLP	17 PSE	Pause und Anzeige der ersten Lösung.
CLP 01	18 XEQ 01	Hier beginnt die Routine zur Berechnung des zweiten Lösungswegs.
+	19 +	
CLP 01	20 RCL 01	
2	21 2_	
CLP	22 ×	
÷	23 /	
CLP CLP PSE		
CLP	24 PSE	Pause und Anzeige der zweiten Lösung.
R/S	25 RTN	Hier hält das Programm endgültig an.

CLP CLP 01	26 LBL 01	Anfang des Unterprogramms. Diese befindet sich innerhalb des Hauptprogramms.
CLP 02	27 RCL 02	
CHS	28 CHS	
CLP 02	29 RCL 02	
x²	30 X ²	
CLP 01	31 RCL 02	
CLP 03	32 RCL 03	
CLP	33 ×	
4	34 4_	
CLP	35 ×	
-	36 -	
CLP	37 SQRT	Ende des Unterprogramms
CLP CLP CLP	00 REG 38	Abschließen des Unterprogramms und Einfügen eines END -Befehls.
CLP		

Die Lösung der quadratischen Gleichung $x^2+x-6=0$ ergibt sich mit diesem Programm:

Tastenfolge	Anzeige	
CLP	0.0000	Beendet den PRGM-Modus
CLP CLP	a?	
QROOT CLP		
1 R/S	b?	
1 R/S	c?	
6 CHS 1 R/S	-3.0000	1. Lösung
	2.0000	2. Lösung

Siehe auch

- [Eingabe eines Programms](#)
- [Ausführung eines Programms](#)
- [Korrekturmöglichkeiten](#)
- [Lokale Marken](#)
- [Nicht-Programmierbare Funktionen](#)
- [Programmierung mit ALPHA-Ketten](#)
- [Programmschleifen](#)
- [Programmspeicher](#)
- [Unbedingte Sprünge](#)
- [Unterbrechen der Programmausführung](#)
- [Vergleichsoperationen](#)

Textausgabe mit Ergebniswerten

Manchmal ist es nützlich, numerische Ergebnisse mit einem entsprechenden Text zu versehen. Dies kann folgendermaßen durchgeführt werden:

- Tasten Sie die ALPHA-Kette in eine Programmzeile ein.
- Fügen Sie das Ergebnis mit **CLP** der ALPHA-Kette an.
- Laden Sie den Inhalt des ALPHA-Registers mit **AVIEW** in die Anzeige.

Das folgende Beispiel verlangt mit einer Textausgabe eine Zahl, führt einen Programmstopp aus und berechnet den dekadischen Logarithmus der Zahl (Modifikation des Beispiels aus der Textausgabe mit der PROMPT-Funktion). Das Ergebnis wird zusammen mit einem Text versehen:

Tastenfolge	Anzeige	
CLP		
CLP CLP CLP CLP	00 REG 46	
CLP CLP CLP LOGTXT CLP	01 LBL"LOGTXT	Der Programmname LOGTXT.
CLP NUMBER? CLP	02"NUMBER?	Laden des ALPHA-Registers mit NUMBER?.
CLP CLP PROMPT CLP	03 PROMPT	Zeigt das ALPHA-Register an führt einen Programmstopp zur Dateneingabe aus.
LOG	04 LOG	
CLP LOG=	05"LOG=	Laden des ALPHA-Registers mit LOG=.
CLP CLP CLP X	06 ARCL X	Mit dieser Anweisung wird das Ergebnis im X-Register dem Text im ALPHA-Register angefügt.
CLP AVIEW	07 AVIEW	Mit dieser Anweisung wird der Inhalt des ALPHA-Register ausgegeben.
CLP CLP CLP CLP	00 REG 41	Abschließen des Programms und Einfügen eines END -Befehls.

Der dekadische Logarithmus von 12 läßt sich somit bestimmen.

Tastenfolge	Anzeige	
CLP	0.0000	Beendet den PRGM-Modus.
CLP CLP LOGTXT CLP	NUMBER?	Die Textausgabe.
12	12_	Eingabe der Zahl
R/S	LOG=1.0792	Die Textausgabe und das Ergebnis.

CLP **CLX** 0.0000

Textausgabe mit der PROMPT-Funktion

Die einfachste Art der Textausgabe innerhalb eines Programms erfolgt mit der **PROMPT**-Funktion. Die

PROMPT-Anweisung zeigt den Inhalt des ALPHA-Registers an und verursacht eine Programmunterbrechung.

Das folgende Beispiel verlangt mit einer Textausgabe eine Zahl, führt einen Programmstopp aus und berechnet den dekadischen Logarithmus der Zahl:

Tastenfolge	Anzeige	
CLP		
CLP CLP CLP CLP	00 REG 46	
CLP CLP CLP CLOG CLP	01 LBL"CLOG	Der Programmname CLOG.
CLP CLP NUMBER? CLP	02"NUMBER?	Laden des ALPHA-Registers mit NUMBER?.
CLP CLP PROMPT CLP	03 PROMPT	Zeigt das ALPHA-Register an führt einen Programmstopp zur Dateneingabe aus.
LOG	04 LOG	
CLP CLP CLP CLP	00 REG 46	Abschließen des Programms und Einfügen eines CLP -Befehls.

Der dekadische Logarithmus von 8 läßt sich somit bestimmen:

Tastenfolge	Anzeige	
CLP	0.0000	Beendet den PRGM-Modus.
CLP CLP CLOG CLP	NUMBER?	Die Textausgabe.
8	8_	Eingabe der Zahl
R/S	0.9031	Der Logarithmus von 8.
CLP CLX	0.0000	

Die Textausgabe in einem Programm kann auch mit **AVIEW** und **CLP** ausgeführt werden. Ähnlich der **PROMPT**-Funktion bewirkt **AVIEW** ein Anzeige des ALPHA-Registers und **CLP** die Programmunterbrechung.

Verwendung der ASHF-Funktion

Die **ASHF**-Funktion verschiebt den Inhalt des ALPHA-Registers um sechs Zeichen nach links. Diese Zeichen gehen verloren.

Da maximal sechs Zeichen innerhalb eines Registers abgespeichert werden können, kann diese Funktion dazu verwendet werden, lange ALPHA-Ketten aufzutrennen und in mehrere Datenregister zu laden.

Löschen der Anzeige

Die **CLD**-Funktion löscht als Bestandteil eines Programms während der Ausführung dieses Programms die Anzeige. Der vorher angezeigte Text verschwindet und das -Symbol kehrt in die Anzeige zurück.

Wenn Sie den Inhalt des ALLPHA-Registers löschen möchten, können Sie den Befehl **CLP** als Programmzeile einfügen.

Programmierung mit Alpha-Ketten

Die Fähigkeiten Strings zu verarbeiten, läßt sich bei diesem Rechner am wirkungsvollsten in Programmen ausnutzen. ALPHA-Ketten können zur Bezeichnung der Eingabe, als Zustandsbeschreibung eines Programms und zur Bezeichnung von Ausgangswerten verwendet werden.

Sie können ALPHA-Ketten auf unterschiedliche Weise in Ihren Programmen verwenden:

- Sie können beispielsweise einen String zum Bestandteil eines Programms machen und diesen String während der Ausführung mit **AVIEW** anzeigen. Die in der Programmzeile stehende ALPHA-Kette wird in das ALPHA-Register geladen, und kann mit **AVIEW** in die Anzeige gerufen werden. Während das Programm abläuft, bleibt die ALPHA-Kette in der Anzeige, bis das Programm die Anzeige löscht, oder Sie eine neue ALPHA-Kette in die Anzeige laden.
- Jedesmal, wenn ein ALPHA-Kette in die Anzeige gerufen wird, ersetzt dieser String das \rightarrow -Symbol. Wenn das Programm die Anzeige löscht oder das Programm unterbrochen wird, kehrt das \rightarrow -Symbol in die Anzeige zurück. Der PRGM-Indikator wird während des Programmlaufs von diesen Ein/Ausgaberoutinen nicht beeinflußt, und erscheint immer in der Anzeige. Die maximale Länge einer ALPHA-Kette in einer Programmzeile beträgt 15 Zeichen. Sie können jedoch mit **CLP** Ketten von einer Länge bis zu 24 Zeichen zusammenstellen. Die, mit dem **CLP** Befehl in die Programmzeile eingetasteten Zeichen, werden während des Programmlaufs an den Inhalt des Alpha-Register angehängt.

Siehe auch

[Löschen der Anzeige.](#)

[Textausgabe mit der PROMPT-Funktion.](#)

[Textausgabe mit Ergebniswerten.](#)

[Verwendung der ASHF-Funktion.](#)

Lokale Marken

Lokale Marken werden genauso eingetastet wie normale ALPHA-Marken, erfüllen jedoch eine Sonderfunktion. Diese 15 fünfzehn Marken sind **[CLP]** A bis

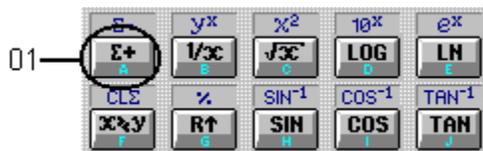
[CLP] und

[CLP] a bis

[CLP] e:

- Lokale Marken werden bei der Funktion **[CLP]** 1 nicht angezeigt.
- Wenn sich der Rechner im USER-Modus befindet und Sie eine Taste der oberen zwei Reihen (oder **[CLP]** und eine Taste der obersten Reihe) drücken, sucht der Rechner nach der entsprechenden lokalen Marke innerhalb des augenblicklichen Programms. Wenn die lokale Marke nicht gefunden wird, führt der Rechner die Standard-Funktion dieser Taste aus, ansonsten, wenn die lokale Marke gefunden wird, startet die Programmausführung mit dieser Marke.

Wenn Sie beispielsweise **[CLP]** im USER-Modus drücken, sucht der Rechner die globale Marke **[CLP]** A im augenblicklichen Programm. Enthält das Programm diese lokale Marke nicht wird die **[CLP]**-Funktion ausgeführt.



Wenn die Marke **[CLP]** A im augenblicklichen Programm existiert, wird die Ausführung mit dieser Marke gestartet. Die Verwendung von lokalen Marken in dieser Weise setzt voraus, daß der Rechner in einem Teil des Programmspeichers steht, der die lokale Marke enthält.

Wenn irgendeine Funktion der oberen Tastenposition zur Ausführung im USER-Modus zugeordnet ist, wird die Suche nach einer lokalen Marke für diese Taste nicht ausgeführt.

Siehe auch

[Eingabe eines Programms](#)

[Ausführung eines Programms](#)

[Korrekturmöglichkeiten](#)

[Nicht-Programmierbare Funktionen](#)

[Programmierung mit ALPHA-Ketten](#)

[Programmschleifen](#)

[Programmspeicher](#)

[Unbedingte Sprünge](#)

[Unterbrechen der Programmausführung](#)

[Unterprogramme](#)

[Vergleichsoperationen](#)

Ausführung eines Programms

Programme, die eine ALPHA-Marke enthalten können entweder mit Hilfe der **CLP**-Taste oder nach Zuordnung und Drücken der Taste im USER-Modus ausgeführt werden. Die Ausführung im USER-Modus benötigt dabei weniger Tastendrucke und Zeit.

Zwei Indikatoren in der Anzeige des Rechners vermitteln die Ausführung eines Programms:

- Wenn das Programm abläuft, erscheint ein \rightarrow in der Anzeige. Jedesmal, wenn eine Programmmarke erreicht wird, verschiebt sich das \rightarrow um eine Position nach rechts. Wenn das \rightarrow die am weiteste recht Position erreicht hat, springt das \rightarrow an die Anfangsposition in der Anzeige zurück.
- Als zusätzlicher Hinweis leuchtet der PRGM-Indikator in der Anzeige auf, wenn ein Programm gestartet wird. Sobald das Programm beendet wird, erlischt der Indikator.
- Nach der Ausführung eines **CLP** oder **CLP** erscheint das \rightarrow nicht., der PRGM-Indikator wird jedoch weiterhin angezeigt.

Das Programmbeispiel zur Berechnung eines beliebigen Kreisinhaltes kann auf zwei unterschiedliche Art und Weisen gestartet werden:

Tastenfolge	Anzeige	
14	14_	Der Radius in m.
CLP	XEQ _ _	Der Rechner verlangt die zur Ausführung erforderliche Eingabe.
CLP CIRCLE		
CLP	615.7522	Das Ergebnis in m ² .
.55	.55_	Der zweite Radius in m.
CLP	XEQ _ _	Der Rechner verlangt die zur Ausführung erforderliche Eingabe.
CLP CIRCLE		
CLP	0,9503	Das Ergebnis in m ² .

Wesentlich schneller kann das Beispiel gestartet werden, wenn Sie **CIRCLE** einer Taste zuordnen:

Tastenfolge	Anzeige	
CLP	ASN _ _	
ASN		
CLP ASN CLP	ASN CIRCLE _	Der Rechner verlangt die für die Zuordnung erforderliche Tastenposition.
LN	SN CIRCLE 15	Die Funktion CIRCLE wird der Taste in Zeile 1 Spalte 5 (LN) zugeordnet. Sie können den zugewiesenen Tasten-Code sehen, wenn Sie die Taste kurz gedrückt halten.
CLX	0.9503	Schaltet den Rechner in den USER-Modus. Alle Funktionen, die Sie der Tastatur zugewiesen haben, werden wirksam. Die Zahl entstammt dem vorangehenden Beispiel.

10.7

CIRCLE (**LN**)

359.6809

Da jetzt **CIRCLE** der **LN**-Taste zugeordnet ist, wird **CIRCLE** ausgeführt, wenn Sie **CLP** im USER-Modus drücken.

.439

CIRCLE (**LN**)

"**CIRCLE**

Wenn Sie die **LN**-Taste kurz gedrückt halten, erscheint die zu **CIRCLE**-zugehörige Funktionsbezeichnung in der Anzeige.

CLP **CLX**

0.0000

CLX

0.0000

Beendet den USER-Modus.

Wie das Beispiel zeigt können Sie von Ihnen erstellte Funktionen wie jede Standard-Funktion des Rechners ausführen und dabei die Belegung des Tastenfeldes selbst bestimmen. Indem Sie Ihre Programme und Funktionen einfach den entsprechenden Tastenfunktionen zuordnen können Sie den Rechner völlig nach Ihrem Wunsch gestalten.

Ein Anwenderprogramm kann jedoch im Gegensatz zu den Standardfunktionen, die mehrmals verschiedenen Tasten zugeordnet werden können, nur einer einzigen Taste zugeordnet werden. Es trifft in diesem Fall die zuletzt vorgenommene Tastenposition zu.

Die Funktion der im USER-Modus-zugeordneten Tasten bleiben solange erhalten, bis Sie entweder das entsprechende Programm aus dem Programmspeicher löschen oder der Taste erneut eine Funktion zuordnen. Beispielsweise bleibt **CIRCLE** der **CLP**-Taste solange zugeordnet, bis Sie **CIRCLE**, oder die zugehörige Marke, aus dem Programmspeicher entfernen oder der **CLP**-Taste eine andere Funktion zuordnen.

Siehe auch

[Eingabe eines Programms](#)

[Korrekturmöglichkeiten](#)

[Lokale Marken](#)

[Nicht-Programmierbare Funktionen](#)

[Programmierung mit ALPHA-Ketten](#)

[Programmschleifen](#)

[Programmspeicher](#)

[Unbedingte Sprünge](#)

[Unterbrechen der Programmausführung](#)

[Unterprogramme](#)

[Vergleichsoperationen](#)

Programmspeicher

Der Rechner ist so konzipiert, daß Sie sich über die Struktur des Programmspeichers keine Gedanken zu machen brauchen. Sie geben lediglich Ihre Anweisungen ein - der Rechner verwaltet automatisch den Programmspeicher.

Eine Anweisung oder Programmzeile besteht aus einer Tastenfolge, die eine vollständige Operation in einem Programm bildet. Jede vollständige Anweisung wird mit einer Zeilennummer versehen. Diese Nummer erscheint beim Eintasten eines Programms in der Anzeige. Auf Grund der Speichergröße eines Registers zu 7 Byte können bis zu 7 Anweisungen in einem Register gespeichert werden.

Eine Anweisung besteht aus einer einzelnen Funktion und allen erforderlichen Eingaben, die die Operation vervollständigen. Zahlen und ALPHA-Strings innerhalb eines Programms werden als einzelne Anweisungen behandelt und belegen daher nur eine Programmzeile.

CLP,

CLP 6 und

CLP 3 sind Beispiele weiterer Anweisungen.

CLP alleine ist eine vollständige Anweisung, da sie nur eine Operation ausführt und keiner weiteren Anweisung bedarf. Dagegen sind

CLP und

CLP keine vollständigen Anweisungen. Diese Funktionen benötigen eine Eingabe des Parameters zur Vollständigkeit.

Nichtprogrammierbare Funktionen

Folgende Operationen können nicht als Anweisungen innerhalb eines Programms verwendet werden. Die Liste besteht im wesentlichen aus den Funktionen die eine Korrektur und ein Editieren der Programme erlauben.

CLP	<u>Löschen eines Programms.</u>
CLP	<u>Löschen einer Programmzeile.</u>
CLP	<u>Löschen mehrerer Programmzeilen.</u>
CLP	<u>Einzelschritt vor.</u>
CLP	<u>Einzelschritt zurück.</u>
CLP CLP	<u>Sprung zu Programmzeile oder einer ALPHA-Marke.</u>
CLP CLP	<u>Sprung an Ende eines Programmes.</u>
CLP	<u>Anzahl der zugewiesenen Datenspeicherregister.</u>
CLP	<u>Packen des Programmspeichers.</u>
CLP	<u>Tastenzordnungen</u>
CLP	<u>User-Modus-Taste</u>
CLP	<u>Programm-Modus-Taste</u>
CLP	<u>Alpha-Modus-Taste</u>
CLP	<u>Katalog-funktion</u>

Stacklift Bedingungen

Abgesehen von den für die Zifferneingabe benötigten Tasten (**CLP**),

CLP,

CLP,

CLP,

CLP und

CLP) beenden alle Funktionstasten die Zifferneingabe. Der Rechner erkennt, daß eine Ziffer, die nach Betätigung einer dieser Funktionstasten eingegeben wird, Bestandteil einer neuen Zahl ist. Mit dieser neuen Zahl wird die im X-Register stehende Zahl überschrieben.

Abhängig von der jeweiligen Funktion kann im Stack eine Verschiebung der Zahlen zu höheren Register stattfinden. so daß der Inhalt des des X-Registers in das Y-Register geschrieben wird, bevor die neue Zahl in das X-Register eingetastet wird.

Die ALPHA-Eingabe wird durch alle Funktionstasten außer **CLP**

CLP beendet. Soll nach Beenden der ALPHA-Zeichen-Eingabe der ALPHA-String verlängert werden, ist lediglich

CLP

CLP zu drücken.

Die Funktionen lassen sich bezüglich ihrer Wirkung auf den Stack in drei Gruppen einteilen. Die meisten Funktionen bereiten einen Stack-Lift vor, einige unterdrücken ihn und andere wieder verhalten sich neutral.

- Wenn Sie im unmittelbaren Anschluß an **Stacklift vorbereitende Funktionen** eine Zahl eintasten, werden die Inhalte der Register im Stack angehoben und die eingegeben Zahl erscheint in der Anzeige.
- Wenn Sie im unmittelbaren Anschluß an **Stacklift-unterdrückende Funktionen** eine Zahl eintasten, werden die Inhalte des Stacks nicht angehoben. Der Inhalt des X-Registers wird damit zu Beginn der Eingabe einer neuen Zahl nicht in das Y-Register kopiert. Stack-Lift unterdrückende Funktionen sind:

CLP,

CLP,

CLP und

CLP.

- **Stacklift neutrale Funktionen** verändern den Status des Stack-Lifts nicht, so daß es von der unmittelbar zuvor eingegebenen Funktion abhängig ist, ob der Stack angehoben wird oder nicht. Wenn Sie beispielsweise **CLP** und

CLP eingeben, wird der Stack nicht verändert, Geben Sie jedoch

CLP nach

CLP ein, dann wird der Stack verändert, da

CLP eine Stack-Lift vorbereitende Funktion ist. Die

CLP hat keinen Einfluß auf den Stack-Lift. Die Stacklift neutralen Funktionen sind:

CLP,

CLP,

CLP

CLP und

CLP.

Speicherregister-Operationen

Analog zum HP41-CV™ von Hewlett-Packard verfügt dieses Simulationsprogramm ttCalc über 319 Datenspeicher-Register. Eine größere Anzahl von Hauptspeicher-Registern ließ sich nicht realisieren, da sonst die Kompatibilität mit dem HP41-CV™ verlorengehe.

Die Speicherregister werden sowohl für den Programmspeicher und die einzelnen Programmschritte, als auch für die Datenspeicher und die Tastenzuordnungen verwendet. Sie können dabei selber bestimmen, wieviel Platz dem Programmspeicher zugewiesen wird. Wenn Sie den Rechner das erste mal verwenden, stehen Ihnen 100 Primär-Speicherregister, 173 zusätzliche Speicherregister und 46 Register für den Programmspeicher zur Verfügung.

Die Speicherregister dienen zum Speichern und Laden von numerischen Daten und Strings. Diese Speicherregister sind unabhängig von dem Rechenregister-Stapel und dem LAST X-Register. Die Informationen in diesen Daten-Registern werden beim Beenden des Programms automatisch gesichert, und stehen Ihnen beim nächsten Start der Applikation zur Verfügung.

Ihnen stehen folgende Funktionen bezüglich der Speicherregister zu Verfügung:

Anzeigen von Registerinhalten

Festlegen der Speicherplatzzuteilung

Löschen der Speicherregister

Speichern von ALPHA-Ketten

Speichern von Daten

Speicherregister-Arithmetik

Zurückrufen von ALPHA-Ketten

Zurückrufen von Daten

Speicherregisterarithmetik

Für die Ausführung arithmetischer Grundrechnungen in den Rechenregistern drücken Sie zuerst **CLP**, dann die entsprechende Arithmetiktaste und schließlich die Zifferntaste zur Bezeichnung der gewünschten Registeradresse.

Operation	Ergebnis
CLP CLP 01	Die Zahl im X-Register wird zu dem Inhalt von Register R ₀₁ addiert. Die Summe wird in R ₀₁ gespeichert.
CLP CLP 02	Die Zahl im X-Register wird von dem Inhalt von Register R ₀₂ subtrahiert. Die Differenz wird in R ₀₂ gespeichert.
CLP CLP 03	Die Zahl im X-Register wird mit dem Inhalt von Register R ₀₃ multipliziert. Das Produkt in R ₀₃ gespeichert.
CLP CLP 04	Der Inhalt von Register R ₀₄ wird durch die Zahl im X-Register dividiert. Der Quotient wird in R ₀₄ gespeichert.

Bei dieser Form der Arithmetik steht das Ergebnis der Rechnungen im entsprechenden Speicherregister. Der Inhalt der Stackregister bleibt unverändert, solange keines der Stackregister als Zieladresse angegeben ist.

Löschen der Speicherregister

Für das Löschen der Speicherregister haben Sie folgende Auswahlmöglichkeiten:

- Wollen Sie ein bestimmtes Register löschen, ersetzen Sie einfach dessen Inhalt durch Null. Um beispielsweise R₁₂ zu löschen drücken Sie **CLP** **12**.
 - Um mit einem Schritt den Inhalt sämtlicher Stackregister zu löschen, führen Sie **CLP** durch. Das LASTX-Register wird durch diese Funktion nicht beeinflusst.
 - Um mit einem Schritt alle Daten-Speicherregister zu löschen, führen Sie **CLRG** durch. Dadurch werden die Inhalte sämtlicher Daten-Speicherregister durch Null ersetzt. **CLRG** hat keinen Einfluß auf den Programmspeicher oder den automatischen Rechenregister- Stapel.

Anzeige von Registerinhalten

Wenn sich der Rechner im Normal-, USER- oder im ALPHA-Modus befindet, kann der Inhalt jedes Speicherregisters betrachtet werden, ohne daß der Stack dadurch beeinflußt wird.

Sie müssen dazu **CLP**

CLP und die Registeradresse drücken. Die Stackregister und das LASTX-Register kann auf die gleiche Art und Weise angezeigt werden. Tasten Sie den Dezimalpunkt

CLP gefolgt von X,Y,Z,T oder L ein, sobald der Rechner mit den Unterstreichungszeichen eine Eingabe anfordert.

Wenn Sie die **CLP**-Taste im ALPHA-Modus verwenden, wird die

CLP-Funktion ausgeführt. Mit

CLP wird lediglich der Inhalt des ALPHA-Registers in die Anzeige geschrieben.

Speichern von Alpha-Ketten

Strings bzw. ALPHA-Ketten bestehen aus einem oder mehreren aufeinanderfolgenden Alpha-Zeichen. Diese Strings, die Sie in das ALPHA-Register eingeben, können Sie in beliebigen Registern abspeichern und wieder zurückrufen.

Mit **CLP** werden die sechs am weitesten links in der Anzeige stehenden Zeichen in das angegebene Register abgespeichert. Mit einer weiteren Funktion können auch Strings mit einer Länge von mehr als sechs Zeichen gespeichert werden. Die **ASHF**-Anweisung verschiebt den Inhalt des ALPHA-Registers um sechs Zeichen nach links. Durch ein wiederholtes Anwenden von **CLP** und **ASHF** kann der gesamte Inhalt des Alpha-Registers auf verschiedene Einzelregister aufgeteilt werden. Ein Zurückrufen des Inhaltes ist mit der **CLP**-Funktion möglich.

Der Inhalt des Stacks wird durch die Funktionen **CLP**, **ASHF**, **CLP** nicht beeinflusst.

Um den Inhalt des Alpha-Registers in eines der Primär-Speicherregister zu kopieren, müssen Sie folgendermaßen verfahren:

- Drücken Sie im ALPHA-Modus **CLP** **CLP** (drücken Sie dazu die Tasten **CLP** **CLP**). Der Rechner verlangt mit **ASTO __** die Registeradresse als Parameter.
- Drücken Sie die entsprechenden Zifferntasten der Registeradresse (00 bis 99). Da der Rechner die Eingabe der Registeradresse erwartet, ist es nicht notwendig für die Zifferneingabe den ALPHA-Modus zu verlassen.

Zurückrufen von Alpha-Ketten

Das Zurückrufen der ALPHA-Daten aus einem Primär-Speicherregister in das X-Registers funktioniert analog zum Speichern der Daten:

- Drücken Sie **CLP**
- **CLP**. Der Rechner verlangt mit **ARCL__** die Registeradresse als Parameter.
- Geben Sie die gewünschte Registeradresse ein (00 bis 99).

Die **CLP**-Funktion fügt die zurückgerufene Zeichenkette an das Ende des Inhaltes des ALPHA-Registers an. Dabei werden keine Zeichen überschrieben.

Eine weitere Funktion hat die **CLP**-Anweisung, wenn der Inhalt des zu ladenden Registers keine Zeichenkette, sondern eine Zahl ist. In diesem Fall wird die Zahl in Form von alphanumerischen Zeichen in das ALPHA-Register geladen. Der geladene String entspricht dabei der Form, wie Sie durch das Zahlenformat mit

CLP,

CLP oder

CLP festgelegt wird. Zahlen, die einen Exponenten enthalten, werden mit einem Präfix E vor dem Exponenten angezeigt.

Speichern von Daten

Um den Inhalt des X-Registers in eines der Primär-Speicherregister zu kopieren, müssen Sie folgendermaßen verfahren:

- Drücken Sie **CLP**. Der Rechner verlangt mit **STO __** die Registeradresse als Parameter.
- Drücken Sie die entsprechenden Zifferntasten der Registeradresse (00 bis 99). Die Adresse muß aus zwei Ziffern bestehen. Die Operation wird ausgeführt, wenn Sie die zweite Ziffer angegeben haben.

Sie können auch den Inhalt des X-Registers in ein weiteres Stack-Register oder das LASTX-Register kopieren:

- Drücken Sie **CLP**.
- **CLP**. Der Rechner verlangt jetzt mit **STO ST _** den Namen des Registers.
- Drücken Sie nun die der Adresse entsprechende Taste (X,Y,Z,T oder L). Es ist in diesem Fall nicht nötig in den ALPHA-Modus umzuschalten.

Zurückrufen von Daten

Das Zurückrufen der Daten aus einem Primär-Speicherregister in das X-Registers funktioniert analog zum Speichern der Daten:

- Drücken Sie **CLP**. Der Rechner verlangt mit **RCL _ _** die Registeradresse als Parameter.
- Drücken Sie die entsprechenden Zifferntasten der Registeradresse (00 bis 99). Die Adresse muß aus zwei Ziffern bestehen. Die Operation wird ausgeführt, wenn Sie die zweite Ziffer angegeben haben.

Sie können auch den Inhalt eines beliebigen Stack-Registers oder des LASTX-Registers in die Anzeige und das X-Register laden:

- Drücken Sie **CLP**.
- **CLP**. Der Rechner verlangt jetzt mit **RCL ST _** den Namen des Registers.
- Drücken Sie nun die der Adresse entsprechende Taste (X,Y,Z,T oder L). Es ist in diesem Fall nicht nötig in den ALPHA-Modus umzuschalten.

Was ist ttCalc ?

Dieses Windows Programm ttCalc stellt eine Simulation des Taschenrechner HP-41CV von Hewlett-Packard dar. Mit Ausnahme der Drucker-Funktionen und ROM-Funktionen werden alle Funktionen des Taschenrechners HP-41CV simuliert.

Im besitzt der Rechner folgende Merkmale:

- Dreidimensionale Oberfläche der Tasten.
- Die 12-Stellige Flüssigkristall-Anzeige des HP-41CV wird simuliert.
- Die Status-Indikatoren geben jederzeit einen Überblick über den Systemzustand des Rechners.
- Drei verschiedene Anzeigemodi für Zahlen.
- Die Fehler-Meldungen sind identisch zum HP-41CV.
- Die Eingabelogik entspricht der umgekehrten polnischen Notation (UPN). Mit einem automatischen Rechenregister-Stapel einer Fehlerbehebung und einer Stack-Verwaltung.
- Die meisten der 130 verschiedenen Standardfunktionen sind integriert. Die Ausführung ist entweder über die Tastatur oder die Anzeige möglich.
- Volle Programmierbarkeit mit der Möglichkeit Programme auch schrittweise ablaufen zu lassen.
- Bis zu 320 Datenregister und ca. 2240 Programmzeilen.
- Alle Tasten des Taschenrechner sind vom Anwender frei mit Funktionen oder Programmen belegbar.
- Steuerung des Rechners über 56 verschiedene Flags.

Daneben bietet der Rechner noch folgende Merkmale:

- Automatisches Abspeichern der Daten, Programme und Tastenbelegungen analog zum Permanentspeicher des Hewlett-Packard-Rechners.
- Simulation die Piezo-Lautsprechers des HP-41CV über den PC-Lautsprecher oder eine beliebige Multimedia-Karte.
- Alle Funktionen und Programme können mit der Maus oder über die PC-Tastatur ausgeführt werden.
- Alle Funktionen und Programme können einer beliebigen Taste der PC-Tastatur zugeordnet werden.
- Analog zum Uhr von Microsoft können Sie den Rechner immer im Vordergrund sichtbar lassen.
- Clipboard-Funktionen.

Einführung in die UPN-Logik

UPN ist die Abkürzung von umgekehrter polnische Notation. Dies ist ein Eingabesystem für algebraische Rechnungen, die keine "="-Taste benötigen. Die UPN-Logik hat Vorteile gegenüber dem üblicheren algebraischen Operations-System (AOS-Rechner), da im allgemeinen weniger Tastendrucke benötigt und Zwischenergebnisse automatisch angezeigt werden. Durch die automatische Speicherung von bis zu vier Zwischenergebnissen auf einem Stack wird auch der Gebrauch von Klammern überflüssig.

Die meisten Standardfunktionen des Rechners beziehen sich entweder auf einen oder auf zwei Zahlenwerte (Ausnahmen von dieser Regel bilden die Statistik-Funktionen **CLP**

CLP und

CLP

CLP). Die Zahlenwerte für die Ausführung dieser Funktionen werden auf dem automatischen Rechenregister-Stapel, genauer im X-Register und eventuell im Y-Register erwartet.

Um die Berechnung von Funktionen mit einer Variable durchzuführen, Tasten Sie die Zahl ein und drücken Sie die entsprechende Funktionstaste. Der UPN-Rechner verhält sich in diesem Fall wie die meisten AOS-Taschenrechner, die ebenfalls zuerst die Eingabe des Arguments, der Zahl, erwarten und erst anschließend die Eingabe der Funktionsbezeichnung zulassen.

Beispiele dazu sind:

$$\begin{aligned} 1/25 &= 0.0400 && (25 \text{ **1/x**}) \\ \log 8.45 &= 0.9269 && (8.45 \text{ **CLP**}) \\ 10^3 &= 100 && (3 \text{ **CLP**}) \\ &&& (\text{**10^x**}) \\ 71^2 &= 5,041.0000 && (71 \text{ **CLP**}) \\ &&& (\text{**CLP**}) \end{aligned}$$

Für die Funktionen mit zwei Variable müssen zwei Zahlen vorhanden sein, um die Operation durchzuführen. Beispiel für diese Funktionen sind die arithmetischen Grundrechenarten **CLP**,

CLP,

CLP und

CLP.

Tasten Sie in diesem Fall die erste Zahl ein und drücken Sie die **CLP**, um die eingegebene Zahl von der nachfolgenden zu trennen. Tasten Sie anschließend die zweite Zahl ein und drücken Sie die entsprechende Funktionstaste. In diesem Fall unterscheidet sich die UPN-Eingabe von der AOS-Eingabe. Bei AOS-Rechner wird die Funktionsbezeichnung zwischen den Funktionsargumenten erwartet, und danach ist noch zusätzlich die "="-Taste zu drücken.

Beispiele dazu sind:

$$\begin{aligned} 15+5 &= 20.0000 && (15 \text{ **CLP** } 5 \\ &&& (\text{**CLP**}) \\ 15-5 &= 10.0000 && (15 \text{ **CLP** } 5 \\ &&& (\text{**CLP**}) \\ 15 \times 5 &= 75.0000 && (15 \text{ **CLP** } 5 \\ &&& (\text{**CLP**}) \\ 15 \div 5 &= 3.0000 && (15 \text{ **CLP** } 5 \\ &&& (\text{**CLP**}) \\ 1.5^{12} &= 129.7463 && (1.5 \text{ **CLP** } 12 \\ &&& (\text{**CLP**}) \\ &&& (\text{**y^x**}) \end{aligned}$$

Auch bei längeren Berechnungen (Kettenrechnungen) wird mit der UPN-Logik jeweils nur eine Operation auf einmal durchgeführt und das Zwischenergebnis sofort angezeigt. Der automatische Rechenregisterstapel, auch Stack genannt, speichert dabei selbstständig bis zu vier Zwischenergebnisse. Dabei müssen Sie bei der Berechnung so vorgehen, wie Sie es auf dem Papier gewohnt sind. Bei der Aufgabe $(17-5) \times 4$ zum Beispiel wird zuerst der innere Klammerausdruck berechnet, und anschließend die Summe mit 4 multipliziert.

Beispiele dazu sind:

Aufgabe	Tastenfolge	Anzeige	
$(5+11) \div 8$	5 CLP	5.0000	
	11 CLP	16.0000	Zwischenergebnis.
	8 CLP	2.0000	Endergebnis.
$(23 \times 16) \div 12$	23 CLP	23.0000	
	16 CLP	138.0000	Zwischenergebnis.
	12 CLP	11.5000	Endergebnis.
$(9+17-4+23) \div 4$	9 CLP	9.0000	
	17 CLP	26.0000	
	4 CLP	22.0000	
	23 CLP	45.0000	Zwischenergebnis.
	4 CLP	11.2500	Endergebnis.
$(6+5) \times (9-3)$	6 CLP	6.0000	
	5 CLP	11.0000	Zwischenergebnis.
	9 CLP	9.0000	
	3 CLP	6.0000	Zwischenergebnis.
	CLP	66.0000	Endergebnis.
$(\text{LOG}(16.3-2)) \div 1.1$	16.3 CLP	16.3000	
	2 CLP	14.3000	Zwischenergebnis.
	CLP	1.1553	Zwischenergebnis.
	1.1 CLP	1.0503	Endergebnis.

Die **CLP** Taste

Das Drücken dieser Taste entspricht dem Ausführen des Kommandos ALT-F4 über die Tastatur ihres PC's. Folglich schaltet diese Taste den Rechner aus. Alle Änderungen, die Sie vom Start der Applikation bis zu Ihrem beenden vorgenommen haben, werden automatisch gesichert und anschließend wird das Programm beendet.

Tastatur

Der ttCalc-Taschenrechner ist in erster Linie auf eine Bedienung per Maus ausgelegt. Trotzdem können Sie auch nur mit fünf verschiedenen Tasten alle Funktionen des Rechners ausführen. Wählen Sie mit den Cursor-Tasten die gewünschte Taste des Rechners aus, anschließend entspricht das Drücken der Enter-Taste auf der PC-Taste exakt dem Mausklick.

Einige weitere elementare Funktionen sind ebenfalls auf der Tastatur fest implementiert: Zum Beispiel können sie die Ziffern 0 bis 9 und den Dezimalpunkt wie gewohnt über die Tastatur eingeben. Die Funktionen + , - , * , und / arbeiten ebenfalls wie Sie es gewohnt sind.

Einige der unten aufgelisteten und fest installierten Funktionen sind nicht ausführbar, wenn sich der Taschenrechner im ALPHA-Modus befindet. Im ALPHA-Modus können Zeichen über die Tastatur wesentlich einfacher und schneller als über das Tastenfeld des Rechner eingegeben werden.

Überblick:

- CLP** Das Drücken der PAGE-DOWN-Taste entspricht der SST-Funktion.
- CLP** Das Drücken der PAGE-UP-Taste entspricht der BST-Funktion.
- CLP** Das Drücken von CTRL+A entspricht dem Mausklick auf die ALPHA-Taste.
- CLP** Das Drücken von CTRL+P entspricht dem Mausklick auf die PRGM-Taste.
- CLP** Das Drücken von CTRL+S entspricht dem Mausklick auf die SHIFT-Taste.
- CLP** Das Drücken von CTRL+U entspricht dem Mausklick auf die USERTaste.
- CLP** Das Drücken von CTRL+R entspricht der R/S-Funktion.
- CLP** Das Drücken <- (Back-Space) entspricht dem Drücken der Korrekturtaste
- CLP** .
- CLP** Die Enter-Taste (auf dem erweiterten Tastenfeld) entspricht der ENTER^ -Funktion.
- CLP** Die N-Taste entspricht der CHS-Funktion.
- CLP** Die E-Taste entspricht der EEX-Funktion.
- CLP** Die Zeicheneingabe von '!' bewirkt die Ausführung der FACT-Funktion.
- CLP** Ein Zeicheneingabe von '%' bewirkt die Ausführung der %-Funktion.
- CLP**
- CLP**
- CLP** Ein Zeicheneingabe von +,-,*,/ bewirkt die Ausführung der entsprechenden Funktion.
- CLP** Wenn sie sich das Funktionsverzeichnis mit
- CLP** 1 betrachten, bewirkt ein Druck auf die F1-Taste die Anzeige einer kontext-sensitiven Hilfe zur gerade angezeigten Funktion.

Ferner haben Sie die Möglichkeit eine beliebige Funktion aus dem Standard-Funktionsverzeichnis oder ein Programm aus dem Hauptspeicher einer Tastenkombination aus [CTRL][SHIFT] und [A..Z] beziehungsweise [F1..F24] zuzuweisen. Wählen Sie dazu im Menü Datei den Menüpunkt Funktion zuordnen.

Das Tastenfeld

Die obersten vier Tasten des Rechners besitzen eine spezielle Bedeutung. Mit diesen Tasten wird der Systemzustand des Rechners bestimmt; d.h. ob im Augenblick numerische Eingaben und arithmetische Berechnungen möglich sind, ob ein Programm ediert wird, oder das alphanumerische Tastenfeld wirksam ist.

CLP Die ALPHA-Modus-Taste.

CLP Die ON-Taste.

CLP Die PRGM-Modus-Taste.

CLP Die USER-Modus-Taste.

Mit jeder Taste des Tastenfeldes können mehrere unterschiedliche Funktionen ausgeführt werden. Die jeweils wirksame Funktion hängt vom Systemzustand des Rechners ab. Befindet sich der Rechner nicht im PRGM-, ALPHA-, oder USER-Modus können alle Funktionen aufgerufen werden, deren Symbol auf der Tastenoberseite oder oberhalb der Taste steht.



Zur Ausführung der Funktion, deren Symbol oberhalb der Taste steht, existieren zwei Möglichkeiten. Eine Variante besteht darin, zuerst die Umschalttaste **CLP** und anschließend die Funktionstaste selber zu drücken. Sie können aber auch die Funktion ausführen, indem Sie das Textfeld oberhalb der Taste mit dem Mauszeiger anklicken. Sie können immer feststellen, ob Sie die **CLP**-Taste gedrückt haben. Der SHIFT-Indikator erscheint jedesmal, wenn Sie **CLP** gedrückt haben. Der Indikator geht wieder aus, sobald Sie die ungeschaltete Funktion ausgeführt haben, oder nochmals **CLP** drücken.

Zur Ausführung der Funktion, deren Symbol auf der Oberseite der Taste abgebildet ist, drücken Sie einfach die Funktionstaste.

Das hellblaue Zeichen, auf der abgeschrägten vorderen Seite der Taste ist nur im ALPHA-Modus verfügbar. Ein Druck der Taste bewirkt im ALPHA-Modus die Eingabe dieses Zeichens.

Eintasten von Zahlen

Zahlen werden im eingetastet, indem Sie die Zifferntasten in der Reihenfolge drücken, wie Sie die Zahl auf einem Blatt Papier notieren würden. Der Dezimalpunkt ist, falls er Bestandteil der Zahl ist, an der entsprechenden Stelle einzutasten (wenn er hinter der letzten Stelle steht ist die Eingabe des Dezimalpunktes überflüssig). Mit einem Unterstreichungszeichen wird jeweils die Eingabe der nächsten Ziffer verlangt. Die Zahl 30.6593 wird beispielsweise wie folgt eingetastet:

Tastenfolge	Anzeige	
30.6593	30.6593_	<i>Die Zahl 30.6543 steht in der Anzeige</i>

Zahlen, die im ALPHA-Modus eingegeben werden, sind nur ALPHA-Zeichen und können nicht in Rechenoperationen verwendet werden. Die Tasten **CLP**

CLP 4

CLP ergeben zum Beispiel das Zeichen 4, mit der keine arithmetischen Berechnungen durchgeführt werden können.

Eingabe negativer Zahlen

Drücken Sie zur Eingabe einer negativen Zahl zuerst die Zifferntasten für die positive Zahl und anschließend **CLP** Die Zahl wird jetzt mit einem vorangestellten Minuszeichen dargestellt. Wenn Sie ein zweites Mal

CLP drücken wandelt sich das negative Vorzeichen der Zahl wieder in ein positives Vorzeichen.

Tastenfolge	Anzeige
CLP	-30.6593_
CLP	30.6593_

Eingabe des Zehnerexponenten

Drücken Sie zur Eingabe des Exponenten zuerst die **CLP**-Taste und dann die dem Exponenten entsprechenden Zifferntasten. Auch in diesem Fall wird mit einem Unterstreichungszeichen die Eingabe der Ziffern für den Exponenten erwartet. Mit der **CLP** wird nach dem Drücken der

CLP-Taste das Vorzeichen des Zehnerexponenten verändert.

Um zum Beispiel die elektrische Elementarladung eines Elektrons (1.6021892e-19 Coulomb) einzugeben, können sie folgendermaßen vorgehen:

Tastenfolge	Anzeige	
CLP	0.0000	
1.6021892	1.6021892_	<i>Eingabe der Mantisse.</i>
CLP	1.6021892	<i>Der Rechner verlangt jetzt die Eingabe des Exponenten.</i>
	-	
19	1.6021892	<i>Eingabe des Zehnerexponenten</i>
	19	
CLP	1.6021892 -	<i>Die Ladung eines Elektrons in Coulomb.</i>
	19	

Die **CLP**-Korrekturtaste

Wenn Sie einzelne Ziffern, den Dezimalpunkt mit **CLP**, den Exponenten mit **CLP** oder das Vorzeichen eines Exponenten mit **CLP** fehlerhaft eingegeben haben, muß die Zahl im allgemeinen nicht komplett neu eingetastet werden.

Im Normalmodus können Sie mit **CLP** einzelne Ziffern oder auch den Dezimalpunkt der Zahl löschen und korrigieren. Wenn das Unterstreichungszeichen in der Anzeige nicht sichtbar ist, hat die **CLP**-Taste dieselbe Wirkung wie **CLP** (mit einer Ausnahme, wenn Sie nämlich gerade die zweite Ziffer des Exponenten eingetastet haben erscheint kein Unterstreichungszeichen in der Anzeige, eine Korrektur der letzten Eingabe ist dennoch möglich).

Datei-Menübefehle

Neu

Öffnen

Speichern

Speichern unter

Funktion zuordnen

Optionen

Beenden

Neu

Dieses Kommando löscht alle Programme, die sich im Programmspeicher des Rechners befinden, alle Datenregister und alle Register des automatischen Rechenregister-Stapel. Die Funktionszuordnungen zu den 'Buttons' des Rechners werden ebenfalls gelöscht, dagegen bleiben die Funktionszuordnungen, die sie mit Menübefehl Funktion zuordnen einer Taste auf der PC-Tastatur zugeordnet haben, gültig.

In der Anzeige erscheint die Meldung **MEMORY LOST**. Wenn Sie anschließend **CLP** drücken, wird wieder der Inhalt des X-Registers angezeigt.

Öffnen

Mit diesem Kommando können Sie Programme, Daten und Tastenzuordnungen, die Sie mit den Menü-Befehlen Speichern und Speichern unter gesichert haben, wieder einlesen.

Bevor die Standard-Dialog Box zum Öffnen der Dateien am Bildschirm erscheint, können Sie bestimmen, ob die alten Daten gespeichert werden sollen.

Das Öffnen einer ttCalc-Datendatei (im allgemeinen besitzen diese Files die Endung *.mem) hat keinen Einfluß auf die Tastenzuordnungen, die sie mit dem Menübefehl Funktion zuordnen vorgenommen haben.

Speichern

Mit diesem Kommando werden die Programme, Daten und Tastenzuordnungen, die sich im Hauptspeicher des Rechners befinden, als binäre Datei abgespeichert.

Die Datei wird unter dem Dateinamen abgespeichert, den Sie vorher mit dem Befehl Öffnen oder dem Befehl Speichern unter festgelegt haben.

Speichern unter

Mit diesem Kommando werden die Programme, Daten und Tastenzuordnungen, die sich im Hauptspeicher des Rechners befinden, als binäre Datei unter einem neuen Dateinamen abgespeichert.

Optionen

Dieses Kommando ist in der augenblicklichen Version von ttCalc noch nicht realisiert, und zukünftigen Versionen vorbehalten.

Geplant ist hier in Zukunft unter anderem eine individuellere Anpassung des Rechners an das Windows-System. Ferner hoffe ich noch in Zukunft das Outfit des Rechners individueller zu gestalten.

Möglich wären z.B. eigene neue frei positionierbare und belegbare Buttons. Eine Hexadezimal-Octal-Binär und Dezimal Umwandlung. Ganz zu schweigen von Time-Modul, Extended-Function-Modul, Math-Modul und

Funktionen zuordnen

Verwenden Sie diesen Befehl, um eine Standard-Funktion des Rechners oder ein Programm einer Tastenkombination zuzuordnen. Einige Tastenkombinationen sind mit vordefinierten Funktionen belegt. Welche können Sie unter Tastatur nachsehen.

Dialogfeldoptionen

Zuzuordnende Funktion

Wählen Sie hier die Funktionsbezeichnung aus einer Liste aus, oder geben Sie den Programmnamen ein. Verwenden Sie dabei keine Leerzeichen.

Tastenkombination definieren

Sie können eine Tastenkombination definieren, die die spätere Ausführung der Funktion erleichtert.

STRG+ Fügt der Tastenkombination die Taste STRG hinzu.

UMSCHALT+ Fügt der Tastenkombination die UMSCHALTTASTE hinzu.

Taste Fügt der Tastenkombination eine Buchstabentaste, Funktionstaste oder eine andere verfügbare Taste hinzu.

Aktuelle Belegung

Zeigt die aktuelle Funktionsbelegung für die ausgewählte Tastenkombination an.

Beschreibung

Geben Sie eine kurze Beschreibung der Funktion ein.

OK

Klicken Sie diesen Schalter an, um die Zuordnung in Ihrer Datei zu speichern und zum Rechner ttCalc zurückzukehren. Die Zuordnungen stehen Ihnen bei einem erneuten Aufruf des Rechners wieder zur Verfügung.

Zuordnen

Klicken Sie diesen Schalter an, um die Zuordnung einer Funktion zu einer Taste vorzunehmen.

Löschen

Klicken Sie diesen Schalter an, um die Zuordnung einer Funktion zu einer Taste zu löschen.

Beenden

Mit diesem Kommando wird die ttCalc-Applikation beendet. Alle Tastenzuweisungen, Daten und Programme, die sich im Hauptspeicher des Rechners befinden werden automatisch unter dem Dateinamen gesichert, den Sie vorher mit dem Befehl Öffnen oder dem Befehl Speichern unter festgelegt haben.

Beim Nächsten Start von ttCalc wird diese Datei automatisch wieder eingelesen.

Bearbeiten-Menübefehle

Kopieren

Einfügen

Immer im Vordergrund

Immer im Vordergrund

Mit diesem Kommando können Sie bestimmen, ob der Rechner immer sichtbar bleiben soll, selbst wenn Sie eine andere Applikation aktivieren. Dadurch können Sie zum Beispiel während eines Programmlaufs Zwischenergebnisse beobachten.

Einfügen

Verwenden Sie dieses Kommando um die Zahl, die sich in Textform in der Zwischenablage befindet, in das X-Register des Rechners zu kopieren. Die Zahl erscheint anschließend in der Anzeige, die ursprünglichen Inhalte des X-, Y-, und Z-Registers werden um ein Register nach oben verschoben, der Inhalt des T-Registers geht verloren.

Kopieren

Verwenden Sie dieses Kommando um die Zahl in der Anzeige des Rechners in die Zwischenablage zu kopieren. Wenn sich der Rechner im PRGM-Modus befindet wird, das augenblickliche Programm in die Zwischenablage kopiert.

Durch das Kopieren des Textes oder Grafiken in die Zwischenablage wird der dort zuvor gespeicherte Inhalt überschrieben.

?-Menübefehle

[Index](#)

[Tastatur](#)

[Suchen](#)

[Information](#)

Mit diesem Kommando öffnen Sie diese Hilfe-Datei.

Mit diesem Kommando können Sie nach bestimmten Schlüsselwörtern in dieser Hilfe-Datei suchen.

Information und Fragen



Wenn Sie Anregungen oder Fragen haben, Fehler im Programm ttCalc entdeckt haben, oder einfach das Programm gut finden, können Sie sich mit dem Autor unter der unten aufgeführten Adresse schriftlich in Verbindung setzen. Vergessen Sie bitte nicht Ihre Anschrift und eventuell Ihre Email-Adresse mit anzugeben. Telefonisch können Sie den Autor unter der Rufnummer (0941) 54938 erreichen.

Stefan Seiwert
Graf-Spee-Straße 1
93053 Regensburg
Deutschland

Literatur

Wenn Sie nähere Informationen über den HP-41CV von Hewlett-Packard und damit auch das Simulationsprogramm benötigen, können Sie in den folgenden Publikationen Details über diese Rechner erfahren. Dies sind im wesentlichen auch die Literaturstellen, die zur Programmierung des ttCalc-Programms herangezogen wurden.

- J.S.Dearing: Tricks, Tips und Routinen für Taschenrechner der Serie HP-41. Deutsche Ausgabe von Heinz Dalkowski. ISBN 3-88538-801-4 (1984). Helderemann Verlag.
- K.Jarett: Synthetisches Programmieren auf dem HP-41 - leicht gemacht. Deutsche Ausgabe von Heinz Dalkowski. ISBN 3-88538-802-2 (1984). Helderemann Verlag.
- K.Jarett: Erweiterte Funktionen des HP-41 - leicht gemacht. Deutsche Ausgabe von Heinz Dalkowski. ISBN 3-88538-803-0 (1984). Helderemann Verlag.
- W.C.Wickes: Synthetisches Programmieren auf dem HP-41C/CV. Deutsche Ausgabe von Heinz Dalkowski. ISBN 3-88538-800-6 (1983). Helderemann Verlag.
- B.Dieckmann: Statistik mit den Taschenrechnern HP15C und HP-41CV. ISBN 3-486-27741-3 (1984). Oldenbourg Verlag.

