

[Kurzübersicht der Befehle] Genauere Erläuterung im Anschluss

[Maschinensteuerung]

NEU	Neuanfang
NEU S	Neu schnell
LB	Lösche Bild
WA	Werkzeug aus
WE	Werkzeug ein
WB n	Werkzeugbreite n Pixel
WF r g b	Werkzeugfarbe RGB
HF r g b	Hintergrundfarbe RGB
FL x y	Fülle ab x,y mit WF
DS n	Drehsinn rechts / links

[Programmsteuerung]

BT n	Befehlstakt in Hundertstel Sekunden
VZ n	Verzögerung, vz 0 ist schnellstens
[xxx]	Kommentare
ENG	Steuerprogramm ins Englische übersetzen
DEU	Steuerprogramm ins Deutsche übersetzen
GB	in Großbuchstaben wandeln
KB	in Kleinbuchstaben wandeln
WH n	Wiederhole n-mal
---	Wiederholen Ende (3x Minus)
PR name	Aufruf eines Unterprogrammes
+ name	Beginn eines Unterprogrammes mit Name name
+++	Ende eines Unterprogrammes
VA v n	Variable v auf Wert n setzen (v = 1..100)
VU v n	Variable v um Wert n ändern
V_v	Schreibweise einer Variablen, z.B. V_3
WT n	Warte n Zentelsekunden (0,1s)

[Absolute Bewegung]

FA x y	Fahre an Position X Y
FAX x	Fahre an Position X
FAY y	Fahre an Position Y
RI w	Richtung
KR r	Kreis (1 Parameter)
KR rx ry	Ellipse (2 Parameter)
KR r v b	Kreisbogen (3 Parameter)
KR rx ry v b	Ellipsenbogen (4 Parameter)
SA x y	Setze auf Position X Y
SAX x	Setze auf Position X
SAY y	Setze auf Position Y

[Relative Bewegung]

FU m n	Fahre um m Schritte in der X-Achse, n in der Y-Achse
FUX n	Fahre um n Schritte in der X-Achse
FUY n	Fahre um n Schritte in der Y-Achse
DL w	Drehe nach links um w Grad
DR w	Drehe nach rechts um w Grad
VW n	Vorwärts um n Schritte in Richtung RI
RW n	Rückwärts um n Schritte in Richtung RI
BR r g	Bogen nach Rechts mit Radius r um g Grad
BL r g	Bogen nach Links mit Radius r um g Grad
SU m n	Setze um m Schritte in X-, um n in Y-Achse
SUX n	Setze um n Schritte in X-Achse
SUY n	Setze um n Schritte in Y-Achse

[Allgemein]

Mit diesem kleinen Simulationsprogramm wird die **2D-Programmsteuerung** einer Werkzeugmaschine nachgestellt.

Die Arbeitsfläche ist 851 x 601 Einheiten groß. Es gibt zwei Achsen: X (0...850) und Y (0...600). Nullpunkt ist links oben! Eine solche Steuerung könnte man zum Trennschweißen, Leiterplattenbohren oder Schildergravieren einsetzen. Mit "PlotMachine" kann man "nur" auf die Arbeitsfläche Linien zeichnen. "PlotMachine" ist aber kein Malprogramm! Jede Linie muss programmiert werden.

Folgende Möglichkeiten sind dabei hilfreich:

- Während des Programmablaufs wird der aktuelle Befehl hervorgehoben und gefärbt:
 - grün bedeutet: es sollte alles soweit in Ordnung sein
 - rot bedeutet: in der betreffenden Zeile sind Fehler (meist Syntax)
- Nach dem Ablauf eines Programmes sind alle Kommentare fett formatiert, das verbessert die Übersichtlichkeit.

- Alle unbenutzten Zeilen sind ebenfalls fett formatiert, das soll die Fehlersuche erleichtern.
- Der Programmcode wird in Wiederholschleifen eingerückt.
- Fehlerhafte Zeilen werden im Fehlerfenster angezeigt
- Wenn sich das Fehlerfenster rot färbt, ist die Maschine außerhalb ihres Arbeitsbereiches.
- Ein Doppelklick auf eine Zeilennummer im Fehlerfenster setzt den Cursor in die fehlerhafte Zeile.

"PlotMachine" soll spielerisch an planvolles, genaues, analytisches und strukturiertes Arbeiten heranzuführen und trotzdem Spaß machen. Bei der Suche nach einem solchen Programm - es sollte natürlich noch kostenlos sein und problemlos in der Schule einsetzbar sein - bin ich nicht fündig geworden, also habe ich es selbst geschrieben (Delphi (TM) 5).

Wenn dieses Programm im Unterricht eingesetzt werden sollte, empfehle ich, diese Hilfedatei zu sperren (umbenennen, verschieben oder löschen) und den Befehlssatz schrittweise einzuführen. Erfahrungsgemäß macht es Einsteigern Schwierigkeiten, zwischen absoluter und relativer Wegeprogrammierung zu unterscheiden. Da aber beide Methoden Vorteile haben, sollte man auch beide verstehen und sinnvoll einsetzen können.

Es ist nicht das Ziel, einen komfortablen Befehlssatz zu bieten. Die Aufgabe besteht eben auch darin, mit diesen grundsätzlichen Befehlen auszukommen, um sein Vorhaben zu realisieren.

Das Programm wurde unter Windows (TM) 98SE und Windows (TM) XP erfolgreich getestet. Es ist eine Bildschirmauflösung von mindestens 1024x768 nötig.

Es gibt zwei Betriebsarten: Direkt-Mode und Programm-Mode

[Direkt-Mode]

Im Direkt-Mode wird jeder Befehl sofort ausgeführt. Damit kann man experimentieren und die Steuerung kennenlernen. Man kann Schritt für Schritt zum Ergebnis kommen.

[Programm-Mode]

Im Programm-Mode können beliebig viele Befehle eingegeben und anschließend abgearbeitet werden. Je Zeile darf ein Befehl geschrieben werden. Bei einer Leerzeile im Programmtext wird die Programmausführung beendet. Man kann also nach einer Leerzeile Hinweise o.ä. hinterlassen oder das Programm für Tests hier stoppen. Programme können gespeichert und geladen werden. Im Programmfenster kann man alle üblichen Editor-Befehle (Strg-C = Kopieren, Strg-V = Einfügen, Str-X = Ausschneiden...) nutzen.

Während der Programmabarbeitung dürfen keine Tastatureingaben gemacht werden, da der Focus wegen der mitlaufenden Markierung auf dem Editor liegt (Abbruch beim Versuch). Natürlich kann man so auch ein laufendes Programm beabsichtigt abbrechen - am besten mit der Taste [Esc], damit sich am Programmtext nichts ändert.

Zwischen Direkt-Mode und Programm-Mode kann jederzeit gewechselt werden, alle Einstellungen bleiben dabei erhalten. Um nichts dem Zufall zu überlassen, sollte man am Anfang eines Programmes alle Einstellungen festlegen, wenn einem die Standardeinstellungen nicht gefallen.

Folgende Befehle stehen zur Verfügung:

[Allgemeine Maschinenbefehle]

NEU	Neuanfang , Bild gelöscht (X=0, Y=0, RI 90, WA, WB 3, WF gelb, HF blau, BT 100, VZ 3, DS R)
NEU S	Neuanfang schnell , Bild gelöscht (wie NEU, aber BT=1, VZ=0)
LB	Lösche Bild
WA	Werkzeug aus
WE	Werkzeug ein
WB n	Werkzeugbreite (n = 1...850)
WF r g b	Werkzeugfarbe R G B, (R,G,B = 0...255) siehe Beispielprogramm "farben.prg"
HF r g b	Hintergrundfarbe R G B wie bei WF. Achtung: Wechsel der Hintergrundfarbe löscht Bildschirm !!!
FL x y	Fülle von Position x,y ausgehend bis an die "Kanten" mit Farbe WF

DS n Drehsinn links oder rechts, wichtig für das Zeichnen von Kreisbögen

[Befehle zur Programmsteuerung]

BT n **B**efehlstakt in 1/100 Sekunden (n=1...2000, also 0,01 ... 20s) um besser beobachten/erklären zu können.
VZ n **V**erzögerung (n=0.....50), um das Zeichnen der Linien verfolgen zu können, muss der Computer gebremst werden.
n = 1...50 zwingt zu einer zusätzlichen (!) Pause von n Millisekunden nach jedem gezeichneten Pixel. Sonderfall ist n=0, dabei wird jeder Befehl unverzögert ausgeführt. Mit BT kann dafür gesorgt werden, dass man die Abarbeitung des Programmes dennoch Schritt für Schritt verfolgen kann.

[xxx] Kommentare zur besseren Übersicht sind in Klammern möglich, auch in <>, (), {}, # oder //. Damit der Kommentar nicht als Fehler erkannt wird, genügt es, am Anfang jeder Kommentarzeile ein solches Zeichen zu notieren.

ENG Übertägt Programmtext in **englische** Schreibweise
DEU Übertägt Programmtext in **deutsche** Schreibweise
GB Wandelt Programmtext in **Großbuchstaben**
KB Wandelt Programmtext in einheitliche **Kleinbuchstaben**
(Leerzeile) beendet Programmausführung, anschließend sind also Notizen o.ä. möglich.

WH n Wiederhole die folgenden Programmzeilen n mal
--- Ende des zu wiederholenden Programmbereiches (3x Minus)

PR name Aufruf eines Unterprogrammes mit dem Namen name
+ name Beginn eines Unterprogrammes mit Name name, die Unterprogramme müssen am Ende des Programmtexes stehen und vor dem ersten Unterprogramm muss eine Leerzeile sein

+++ Ende eines Unterprogrammes

VA v n **V**ariable v **auf** Wert n setzen (v = 1..100)
VU v n **V**ariable v **um** Wert n ändern
V_v Schreibweise einer Variablen, z.B. V_3, Variable können als Parameter angegeben werden
WT n **W**arte n Zehntelsekunden

[Absolute Bewegungen]

FA x y **F**ahre **a**n Position X Y
FAX x **F**ahre **a**n Position X, Y bleibt unverändert, also horizontale Bewegung
FAY y **F**ahre **a**n Position Y, X bleibt unverändert, also vertikale Bewegung
RI w **R**ichtung (w = 0...360) 0 = nach oben, 90 nach rechts, 180 nach unten, ...
KR r **K**reis um aktuelle Position (eventuell erst SA/SU x y) mit Radius r, KR 0 zeichnet einen Punkt mit aktueller WB
KR rx ry **K**reis um aktuelle Position (eventuell erst SA/SU x y) mit Radius rx (Richtung der X-Achse) und ry (Richtung Y-Achse)
KR r v b **K**reisbogen an aktueller Position (eventuell erst SA/SU x y) von v bis b mit Radius r (v und b entsprechen Festlegung bei RI)
KR rx ry v b **E**llipsenbogen, Prinzip wie bei Kreisbogen
SA x y **S**etze **a**n Position x y, dabei wird nie gezeichnet (wichtig für Kreisbewegungen)
SAX x **S**etze **a**n X-Wert x, Y bleibt unverändert, also horizontale Bewegung, dabei wird nie gezeichnet
SAY y **S**etze **a**n Y-Wert y, X bleibt unverändert, also vertikale Bewegung, dabei wird nie gezeichnet

[Relative Bewegungen]

FU m n **F**ahre **u**m m Schritte in der X-Achse, n Schritte in der Y-Achse (m und n können auch negativ sein)
FUX n **F**ahre **u**m n Schritte in der X-Achse, Y bleibt unverändert, also horizontale Bewegung
FUY n **F**ahre **u**m n Schritte in der Y-Achse, X bleibt unverändert, also vertikale Bewegung
DL w **D**rehe nach **l**inks um w Grad (w = 0 360)
DR w **D**rehe nach **r**echts um w Grad (w = 0 360)
VW n **V**orw_ärts um n Schritte in Richtung RI
RW n **R**ückw_ärts um n Schritte in Richtung RI
BR r g **B**ogen **R**echts um g Grad mit Radius r
BL r g **B**ogen **L**inks um g Grad mit Radius r

SU m n Setze um m Schritte in der X-Achse, n Schritte in der Y-Achse, dabei wird nie gezeichnet
SUX n Setze um n Schritte in der X-Achse, dabei wird nie gezeichnet
SUY n Setze um n Schritte in der Y-Achse, dabei wird nie gezeichnet (m und n können auch negativ sein)

[Beispielprogramme]

Am einfachsten ist es, man sieht sich ersteinmal ein paar Beispielprogramme an. Wer ein geniales Steuerprogramm geschrieben hat, kann es an mich mailen. Ich würde es dann vielleicht zu den Beispielprogrammen packen. Vergiss also Name, Alter und dein Land nicht am Ende des Programmes!

[Tipps]

Alle Befehle können klein und/oder groß geschrieben werden. "Plotmaschine" versteht jederzeit beide Sprachen (deutsch, englisch). Wenn sinnvoll, dann sind auch negative Parameter möglich.

Folgende Befehle sind in ihrer Wirkung gleich (nur Beispiele):

DL 90	= DR -90	
DL 90	= DR 270	Man sollte immer den logischsten Befehl nutzen!
VW 15	= RW -15	RW -20 ist nicht gerade sinnvoll, aber möglich.
RI 0	= RI 360	
RI 380	= RI 20	

Alle Zahlen können auch Hexadezimal angegeben werden. Dazu ein \$ voranstellen (\$25 = 37).

[Eastereggs]

Ja, gibt es.

[Hinweise / Copyright / Disclaimer]

Jede Veränderung der Software (gilt für plotmachine_xx.exe und Hilfetexte) ist nicht gestattet. Das Urheberrecht verbleibt beim Autor.

Das Programm ist Freeware und darf nur kostenlos weitergegeben werden. Es ist selbstverständlich erlaubt, bei der Weitergabe der Software zusätzliche Beispielprogramme mitzugeben.

Der Autor übernimmt keine Haftung für durch die Anwendung der Software eintretende Schäden jeglicher Art.

Die Software beinhaltet keine Adware, Spyware oder andere Dinge dieser Art.

Die Installation beschränkt sich auf das Entpacken der Zipdatei. Es sind keine Administratorrechte erforderlich. Es erfolgen keine Einträge in die Registry. Das Programm kann ohne Vorkehrungen wieder gelöscht werden.

Ich bin für jeden Tipp (Fehler, Verbesserungsvorschläge) dankbar. eMail und Homepage des Autors sind über die Programmoberfläche zu finden.

Thomas Kraft - Chemnitz / Sachsen / Deutschland

p.s.: Dieser Hilfetext lässt sich markieren und in die Zwischenablage übernehmen. Er kann dann mit einem beliebigen Textprogramm gedruckt werden. Er liegt auch im Ordner \help als rtf-Datei.