

Beschreibung der Makrosprache (Anwendung in .TML-Dateien)

Die verwendete Sprache zur Programmierung benutzerdefinierter Funktionen ähnelt den Programmiersprachen C und Pascal.

Allgemeine Beschreibung

- Ein-Pass Compiler (Funktion kann nur vorher deklarierte benutzerdefinierte Funktionen aufrufen)
- Globale Variablen werden am Anfang des Programms zugewiesen
- Lokale Variablen werden am Anfang einer Funktion zugewiesen
- Parameterübergabe wird durch Übergabe einer Kopie ausgeführt
- Nur Variablen können als Parameter benutzt werden (Ausnahme bei Anwendung der ERROR- oder der PRINTMESSAGEBOX-Funktionen)
- Alle Variablentypen sind doppelt
- Funktionsüberladung ist nicht zugelassen
- Eine Reihe von mathematischen Funktionen und vordefinierten Funktionen sind verfügbar
- Variablen, Funktionsnamen, usw. unterscheiden Groß- und Kleinschreibung
- Alle Schlüsselwörter und vordefinierte Funktionen sind in Großbuchstaben
- Verschachtelte Funktionsanweisungen sind nicht zugelassen

Mathematische Funktionen

Folgende mathematische Funktionen sind verfügbar:

ABS (x)	Errechnet den absoluten Wert von x
ACOS (x)	Errechnet den Arcuskosinus von x im Bogenmaß
ASIN (x)	Errechnet den Arcussinus von x im Bogenmaß
ATAN (x)	Errechnet den Arcustangens von x im Bogenmaß
COS (x)	Errechnet den Kosinus von x im Bogenmaß
SIN (x)	Errechnet den Sinus von x im Bogenmaß
TAN (x)	Errechnet den Tangens von x im Bogenmaß
POW (x, y)	Errechnet die Potenz x hoch y. Größter Ausgabewert ist 2 ⁶⁴
LOG (x)	Errechnet den natürlichen Logarithmus von x
LOG10 (x)	Errechnet den dekadischen Logarithmus von x
SQRT (x)	Errechnet die Quadratwurzel von x
RADIANS (x)	Kehrt x ins Bogenmaß um

Parameterfunktionen

Parameter-Funktionen werden verwendet, um Werte in eine Reihe von vorherbestimmten Variablen außerhalb der benutzerdefinierten Funktionsdatei zu erhalten oder zu speichern. Sie reduzieren die Anzahl von Funktionen, welche für einfache Prozesse, wie Speicherung des Werts eines Zeichens (z. B. X, Y, Z, usw.) erstellt werden müssen. Diese externen Variablen werden nach jeder Zeile auf UNDEFINED zurückgesetzt.

SETA()	Setzt die externe Variable A
SETD()	Setzt die externe Variable D
SETH()	Setzt die externe Variable H
SETI()	Setzt die externe Variable I
SETJ()	Setzt die externe Variable J
SETK()	Setzt die externe Variable K
SETF()	Setzt die externe Variable F
SETO()	Setzt die externe Variable O
SETP()	Setzt die externe Variable P
SETQ()	Setzt die externe Variable Q
SETR()	Setzt die externe Variable R
SETS()	Setzt die externe Variable S
SETT()	Setzt die externe Variable T
SETU()	Setzt die externe Variable U
SETV()	Setzt die externe Variable V
SETW()	Setzt die externe Variable W
SETX()	Setzt die externe Variable X
SETY()	Setzt die externe Variable Y
SETZ()	Setzt die externe Variable Z
SETP1()	Setzt die externe Variable P1
SETP2()	Setzt die externe Variable P2
SETP3()	Setzt die externe Variable P3
SETP4()	Setzt die externe Variable P4
SETP5()	Setzt die externe Variable P5
SETP6()	Setzt die externe Variable P6
SETP7()	Setzt die externe Variable P7
GETA()	Übernimmt den Wert der externen Variable A
GETD()	Übernimmt den Wert der externen Variable D
GETH()	Übernimmt den Wert der externen Variable H
GETI()	Übernimmt den Wert der externen Variable I
GETJ()	Übernimmt den Wert der externen Variable J
GETK()	Übernimmt den Wert der externen Variable K
GETF()	Übernimmt den Wert der externen Variable F
GETO()	Übernimmt den Wert der externen Variable O
GETP()	Übernimmt den Wert der externen Variable P
GETQ()	Übernimmt den Wert der externen Variable Q
GETR()	Übernimmt den Wert der externen Variable R
GETS()	Übernimmt den Wert der externen Variable S
GETT()	Übernimmt den Wert der externen Variable T
GETU()	Übernimmt den Wert der externen Variable U
GETV()	Übernimmt den Wert der externen Variable V
GETW()	Übernimmt den Wert der externen Variable W
GETX()	Übernimmt den Wert der externen Variable X
GETY()	Übernimmt den Wert der externen Variable Y
GETZ()	Übernimmt den Wert der externen Variable Z
GETP1()	Übernimmt den Wert der externen Variable P1
GETP2()	Übernimmt den Wert der externen Variable P2
GETP3()	Übernimmt den Wert der externen Variable P3
GETP4()	Übernimmt den Wert der externen Variable P4
GETP5()	Übernimmt den Wert der externen Variable P5
GETP6()	Übernimmt den Wert der externen Variable P6
GETP7()	Übernimmt den Wert der externen Variable P7
GETCURRENTVALUE()	Übernimmt den Wert des aktuellen Zeichens (z.B. X11 erscheint als 11)

Maschinenfunktionen

Diese Maschinenfunktionen können in den benutzerdefinierten Funktionsdateien angegeben werden. Parameter müssen aus Variablen, im Gegensatz zu Zahlen oder Konstanten, bestehen (z. B. DWELL(12) wird einen Syntax-Fehler anzeigen). Zeichen können eingesetzt werden, um eine vorherbestimmte Maschinenfunktion ablaufen zu lassen, ohne dass der Benutzer dazu eigene Funktionsdateien erstellen muss. Die in diesem Fall verwendeten Parameter würden die selben sein, wie derjenige in der unten angegebenen Beschreibung (z. B. DWELL(D) verwendet die externe Variable D).

Name:	ABSOLUTEMODE()
Aktion:	Setzt das System auf absolute Koordinaten.
Parameter:	Keiner
Rücksprung:	1
Name:	ARCCCWTO(X,Y,Z,I,J,K)
Aktion:	Kreisinterpolation im Gegenuhrzeigersinn.
Parameter:	X - x Endpunkt-Koordinate oder x Endpunkt-Abstand Y - y Endpunkt-Koordinate oder y Endpunkt-Abstand Z - z Endpunkt-Koordinate oder z Endpunkt-Abstand I - x Abstand vom Startpunkt zum Mittelpunkt J - y Abstand vom Startpunkt zum Mittelpunkt K - z Abstand vom Startpunkt zum Mittelpunkt
Rücksprung:	1
Name:	ARCCCWTORADIUS(X,Y,Z,R)
Aktion:	Kreisinterpolation im Gegenuhrzeigersinn.
Parameter:	X - x Endpunkt-Koordinate oder x Endpunkt-Abstand Y - y Endpunkt-Koordinate oder y Endpunkt-Abstand Z - z Endpunkt-Koordinate oder z Endpunkt-Abstand R - Kreisbogen-Radius
Rücksprung:	1
Name:	ARCCWTO(X,Y,Z,I,J,K)
Aktion:	Kreisinterpolation im Uhrzeigersinn.
Parameter:	X - x Endpunkt-Koordinate oder x Endpunkt-Abstand Y - y Endpunkt-Koordinate oder y Endpunkt-Abstand Z - z Endpunkt-Koordinate oder z Endpunkt-Abstand I - x Abstand vom Startpunkt zum Mittelpunkt J - y Abstand vom Startpunkt zum Mittelpunkt K - z Abstand vom Startpunkt zum Mittelpunkt
Rücksprung:	1
Name:	ARCCWTORADIUS(X,Y,Z,R)
Aktion:	Kreisinterpolation im Uhrzeigersinn.
Parameter:	X - x Endpunkt-Koordinate oder x Endpunkt-Abstand Y - y Endpunkt-Koordinate oder y Endpunkt-Abstand Z - z Endpunkt-Koordinate oder z Endpunkt-Abstand R - Radius des Kreisbogens
Rücksprung:	1
Name:	CANCELCANNEDCYCLE()
Aktion:	Hebt einen vordefinierten festen Zyklus auf
Parameter:	Keiner
Rücksprung:	1

Name: CHANGETOOLOFFSET(P,R)
 Aktion: Ändert den Wert im Werkzeugspeicher für Werkzeuglänge und Radiuskorrektur.
 Parameter: P - Werkzeugspeicher-Nummer
 R - Absoluter oder inkrementeller Betrag im Werkzeugspeicher
 Rücksprung: 1

Name: CLAMPS1OFF()
 Aktion: Schaltet Klemmung 1 aus.
 Parameter: Keiner
 Rücksprung: 1

Name: CLAMPS1ON()
 Aktion: Schaltet Klemmung 1 ein.
 Parameter: Keiner
 Rücksprung: 1

Name: CLAMPS2OFF()
 Aktion: Schaltet Klemmung 2 aus.
 Parameter: Keiner
 Rücksprung: 1

Name: CLAMPS2ON()
 Aktion: Schaltet Klemmung 2 ein.
 Parameter: Keiner
 Rücksprung: 1

Name: COOLANT1OFF()
 Aktion: Schaltet Kühlmittel 1 aus.
 Parameter: Keiner
 Rücksprung: 1

Name: COOLANT1ON()
 Aktion: Schaltet Kühlmittel 1 ein.
 Parameter: Keiner
 Rücksprung: 1

Name: COOLANT2OFF()
 Aktion: Schaltet Kühlmittel 2 aus.
 Parameter: Keiner
 Rücksprung: 1

Name: COOLANT2ON()
 Aktion: Schaltet Kühlmittel 2 ein.
 Parameter: Keiner
 Rücksprung: 1

Name: CORNEROFFSETCIRCULARINTERPOLATION(X,Y)
 Aktion: Ermöglicht die Ausführung einer Kreisinterpolation mit der Werkzeugradiuskorrektur um eine Ecke herum.
 Parameter: Keiner
 Rücksprung: 1

Name: DWELL(D)
 Aktion: Setzt eine Verweilzeit.
 Parameter: D - Verweilzeit in Sekunden
 Rücksprung: 1

Name: INCH()
 Aktion: Setzt die Einheiten auf Zoll.
 Parameter: Keiner
 Rücksprung: 1

Name: INCREMENTALMODE()
 Aktion: Setzt das System auf inkrementelle Koordinaten.
 Parameter: Keiner
 Rücksprung: 1

Name: LENGTHCOMPMINUS(H)
 Aktion: Kompensiert die Werkzeuglänge in negativer Richtung mit einem Wert aus dem Werkzeugkorrekturspeicher.
 Parameter: H - Werkzeugkorrekturspeicher
 Rücksprung: 1

Name: LENGTHCOMPOFF()
 Aktion: Wählt die Längenkompensation ab.
 Parameter: Keiner
 Rücksprung: 1

Name: LENGTHCOMPPLUS(H)
 Aktion: Kompensiert die Werkzeuglänge in positiver Richtung mit einem Wert aus dem Werkzeugkorrekturspeicher.
 Parameter: H - Werkzeugkorrekturspeicher
 Rücksprung: 1

Name: LINEARABSTO(X,Y,Z)
 Aktion: Führt eine Linearbewegung auf die zugehörigen Koordinaten.
 Parameter: X - x Koordinate
 Y - y Koordinate
 Z - z Koordinate
 Rücksprung: 1

Name: LINEARTO(X,Y,Z)
 Aktion: Führt eine Linearbewegung auf die zugehörigen Koordinaten oder mit inkrementellen Abständen.
 Parameter: X - x Koordinate oder x Abstand
 Y - y Koordinate oder z Abstand
 Z - z Koordinate oder z Abstand
 Rücksprung: 1

Name: METRIC()
 Aktion: Setzt die Einheiten auf metrisch.
 Parameter: Keiner
 Rücksprung: 1

Name: OPTIONALPROGRAMSTOP()
 Aktion: Hält das Programm an, wenn der Schalter für den wahlweisen Halt aktiviert ist.
 Parameter: Keiner
 Rücksprung: 1

Name: ORIENTEDSPINDLESTOP()
 Aktion: Orientierter Spindelstopp.
 Parameter: Keiner
 Rücksprung: 1

Name: PRINTERERRORNUMBER(P1)
 Aktion: Zeigt die mit Parameter P1 bezeichnete Fehlermeldung an.
 Parameter: P1 - Fehler-Nummer
 Rücksprung: 1

Name: PROGRAMEND()
 Aktion: Beendet das Programm.
 Parameter: Keiner
 Rücksprung: 1

Name: PROGRAMENDRESET()
 Aktion: Beendet das Programm und setzt es auf Start zurück.
 Parameter: Keiner
 Rücksprung: 1

Name: PROGRAMSTOP ()
 Aktion: Hält das Programm an.
 Parameter: Keiner
 Rücksprung: 1

Name: QUEUETOOL(T)
 Aktion: Wählt ein Werkzeug zum Werkzeugwechsel vor.
 Parameter: T - Werkzeug-Nummer
 Rücksprung: 1

Name: RADIUSCOMPLEFT(D)
 Aktion: Kompensiert das Werkzeug zur linken Seite mit dem Wert aus der gewählten Werkzeugkorrektur-Nummer.
 Parameter: D - Werkzeugkorrektur-Nummer
 Rücksprung: 1

Name: RADIUSCOMPOFF()
 Aktion: Wählt die Werkzeugkorrektur ab.
 Parameter: Keiner
 Rücksprung: 1

Name: RADIUSCOMPRIGHT(D)
 Aktion: Kompensiert das Werkzeug zur rechten Seite mit dem Wert aus der gewählten Werkzeugkorrektur-Nummer.
 Parameter: D - Werkzeugkorrektur-Nummer
 Rücksprung: 1

Name: RAPIDABSTO(X,Y,Z)
 Aktion: Führt im Eilgang auf die zugehörigen Koordinaten.
 Parameter: X - x Koordinate
 Y - y Koordinate
 Z - z Koordinate
 Rücksprung: 1

Name: RAPIDTO(X,Y,Z)
 Aktion: Führt im Eilgang auf die zugehörigen Koordinaten oder mit inkrementellen Abständen.
 Parameter: X - x Koordinate oder x Abstand
 Y - y Koordinate oder y Abstand
 Z - z Koordinate oder z Abstand
 Rücksprung: 1

Name: REPOSITIONORIGIN(X,Y,Z)
 Aktion: Neupositionierung des Nullpunktes mit den XYZ-Koordinaten. Wenn eine Koordinate undefiniert ist, wird die aktuelle Koordinaten-Position genommen.
 Parameter: X - x Koordinate
 Y - y Koordinate
 Z - z Koordinate
 Rücksprung: 1

Name: RETURNTOREFERENCE(X,Y,Z)
 Aktion: Ermöglicht dem aktuellen Werkzeug über einen mit XYZ-Zwischenpunkt auf einen vordefinierten Referenzpunkt zu fahren. Wenn die XYZ-Koordinaten nicht definiert sind, dann fährt es direkt zum Referenzpunkt.
 Parameter: X - x Koordinate oder x Abstand
 Y - y Koordinate oder y Abstand
 Z - z Koordinate oder z Abstand
 Rücksprung: 1

Name: RETURNTOSECONDREFERENCE(X,Y,Z)
 Aktion: Ermöglicht dem aktuellen Werkzeug über einen mit XYZ-Zwischenpunkt auf einen vordefinierten zweiten Referenzpunkt zu fahren. Wenn die XYZ-Koordinaten nicht definiert sind, dann fährt es direkt zum Referenzpunkt.
 Parameter: X - x Koordinate oder x Abstand
 Y - y Koordinate oder y Abstand
 Z - z Koordinate oder z Abstand
 Rücksprung: 1

Name: SETFEEDPERREVOLUTION(F)
 Aktion: Bestimmt den Vorschubwert pro Umdrehung.
 Parameter: F - Vorschubwert pro Umdrehung
 Rücksprung: 1

Name: SETFEEDRATE(F)
 Aktion: Setzt Vorschubwert.
 Parameter: F - Vorschubwert
 Rücksprung: 1

Name: SETINITIALPLANEDEFAULT()
 Aktion: Bewirkt, dass das Werkzeug nach einer Bohrbearbeitung zur Anfangsebene zurückkehrt.
 Parameter: Keiner
 Rücksprung: 1

Name: SETXYPLANE()
 Aktion: Legt die XY-Ebene als Hauptbearbeitungsebene fest, um Kreisinterpolation und Werkzeugkorrektur auszuführen.
 Parameter: Keiner
 Rücksprung: 1

Name: SETXZPLANE()
 Aktion: Legt die XZ-Ebene als Hauptbearbeitungsebene fest, um Kreisinterpolation und Werkzeugkorrektur auszuführen.
 Parameter: Keiner
 Rücksprung: 1

Name: SETYZPLANE()
 Aktion: Legt die YZ-Ebene als Hauptbearbeitungsebene fest, um Kreisinterpolation und Werkzeugkorrektur auszuführen.
 Parameter: Keiner
 Rücksprung: 1

Name: SINGLEDIRECTIONPOSITIONING (X,Y,Z)
 Aktion: Genauhalt einer Einzelbewegung.
 Parameter: X - x Koordinate oder x Abstand
 Y - y Koordinate oder y Abstand
 Z - z Koordinate oder z Abstand
 Rücksprung: 1

Name: SPINDLEONCCW()
 Aktion: Schaltet die Spindel im Gegenuhrzeigersinn ein.
 Parameter: Keiner
 Rücksprung: 1

Name: SPINDLEONCW()
 Aktion: Schaltet die Spindel im Uhrzeigersinn ein.
 Parameter: Keiner
 Rücksprung: 1

Name: SPINDLESPEED (S)
 Aktion: Setzt die Spindeldrehzahl.
 Parameter: S - Drehzahl
 Rücksprung: 1

Name: SPINDLESTOP ()
 Aktion: Stoppt die Spindel.
 Parameter: Keiner
 Rücksprung: 1

Name: TOOLCHANGE ()
 Aktion: Wechselt das vorgewählte Werkzeug ein.
 Parameter: Keiner
 Rücksprung: 1

Name: WORKPIECECOORDINATESYSTEM(P3)
 Aktion: Wechselt zum mit P3 bezeichneten Arbeits-
 Koordinatensystem.
 Parameter: P3 - Koordinatensystem
 Rücksprung: 1

Name: HIGH SPEEDPECKDRILLINGCYCLE (Z,R,Q)
 Aktion: Vordefinierter fester Zyklus, der einen
 unterbrochenen Hochgeschwindigkeits-Bohrzyklus
 ausführt.
 Parameter: Z - z Tiefen Koordinate oder z Tiefen Abstand
 R - Rückzugs Koordinate oder Rückzugs Abstand
 Q - Zustelltiefe
 Rücksprung: 1

Name: DRILLINGCYCLE (X,Y,Z,R)
 Aktion: Vordefinierter fester Zyklus, der einen Bohrzyklus
 ausführt.
 Parameter: X - x Koordinate oder x Abstand
 Y - y Koordinate oder y Abstand
 Z - z Tiefen Koordinate oder z Tiefen Abstand
 R - Rückzugs Koordinate oder Rückzugs Abstand
 Rücksprung: 1

Name: SPOTDRILLINGCYCLE(X,Y,Z,R,P)
 Aktion: Vordefinierter fester Zyklus, der einen Senk-
 Bohrzyklus ausführt.
 Parameter: X - x Koordinate oder x Abstand
 Y - y Koordinate oder y Abstand
 Z - z Tiefen Koordinate oder z Tiefen Abstand
 R - Rückzugs Koordinate oder Rückzugs Abstand
 P - Verweilzeit in Sekunden
 Rücksprung: 1

Name: DEEPHOLEDRILLINGCYCLE(X,Y,Z,R,Q)
 Aktion: Vordefinierter fester Zyklus, der einen Tief-
 Bohrzyklus ausführt.
 Parameter: X - x Koordinate oder x Abstand
 Y - y Koordinate oder y Abstand
 Z - z Tiefen Koordinate oder z Tiefen Abstand
 R - Rückzugs Koordinate oder Rückzugs Abstand
 Q - Zustelltiefe
 Rücksprung: 1

Name: EQUALLEADTHREADCUTTING(X,Y,Z,F,Q)
 Aktion: Schneidet gleichförmige Gewindegänge.
 Parameter: X - x Koordinate oder x Abstand
 Y - y Koordinate oder y Abstand
 Z - z Tiefen Koordinate oder z Tiefen Abstand
 R - Rückzugs Koordinate oder Rückzugs Abstand
 Q - Zustelltiefe
 Rücksprung: 1

Maschinen-Eigenschaften Funktionen

Diese Funktionen werden in benutzerdefinierten Funktionsdateien verwendet. Sie benutzen keine Parameter und sind Ausgabe-Maschineneigenschaften.

GETABSOLUTEMODE()	Gibt 1 aus, bei absolutem Modus
GETLENGTHCOMPON()	Gibt 1 aus, wenn die Werkzeuglängen-Korrektur eingeschaltet ist
GETLENGTHCOMPUP()	Gibt 1 aus, wenn die positive Werkzeuglängen-Korrektur eingeschaltet ist
GETLENGTHCOMPREGISTER()	Gibt den Wert des Längenkorrektur-Speichers aus
GETRADIUSCOMPLEFT()	Gibt 1 aus, wenn die linke Radius-Korrektur eingeschaltet ist
GETRADIUSCOMPON()	Gibt 1 aus, wenn die Radius-Korrektur eingeschaltet ist
GETRADIUSCOMPREGISTER()	Gibt den Wert des Radiuskorrektur-Speichers aus
GETFEEDRATE()	Gibt den Vorschubwert aus
GETMAXFEEDRATE()	Gibt den Eilgangwert aus
GETCLAMP1ON()	Gibt 1 aus, bei Klemmung 1 Ein
GETCLAMP2ON()	Gibt 1 aus, bei Klemmung 2 Ein
GETSPINDLEON()	Gibt 1 aus, bei Spindel Ein
GETSPINDLECW()	Gibt 1 aus, wenn Spindel im Uhrzeigersinn dreht
GETSPINDLESPEED()	Gibt die Spindeldrehzahl aus
GETMAXSPINDLESPEED()	Gibt die Spindelhöchstdrehzahl aus
GETCOOLANT1ON()	Gibt 1 aus, bei Kühlmittel 1 Ein
GETCOOLANT2ON()	Gibt 1 aus, bei Kühlmittel 2 Ein
GETINCH()	Gibt 1 aus, wenn die Einheiten auf Zoll gesetzt sind
GETCURRENTX()	Gibt die Position auf der X-Achse aus
GETCURRENTY()	Gibt die Position auf der Y-Achse aus
GETCURRENTZ()	Gibt die Position auf der Z-Achse aus
GETRETRACTION()	Gibt den Rückzugswert bei unterbrochenen Zyklen aus
GETSHIFTDIRECTION()	In Feinbohrzyklen gebraucht. Gibt eine Zahl aus, die einer Abheberichtung entspricht
	1 - positive x Richtung
	2 - negative x Richtung
	3 - positive y Richtung
	4 - negative y Richtung

Gemischte Funktionen

Name:	BEGINSUBROUTINE(O)
Aktion:	Gibt ein Unterprogramm an.
Parameter:	O - Unterprogramm-Nummer
Rücksprung:	1
Name:	BLOCKNUMBER()
Aktion:	Keine Aktion
Parameter:	Keine
Rücksprung:	1

Name: CALLSUBROUTINE(P,L)
 Aktion: Ruft ein durch P benanntes Unterprogramm auf. L darf nicht negativ sein. Wiederholt das Unterprogramm L-mal.
 Z.B. Bei L = 0 wird das Unterprogramm einmal durchlaufen, bei L = 1 zweimal.
 Parameter: P - Unterprogramm-Nummer
 L - Anzahl der Wiederholungen.
 Rücksprung: 1

Name: ENDSUBROUTINE(P)
 Aktion: Kehrt zu der vorigen Stelle zurück, bevor das Unterprogramm aufgerufen wurde, wenn P nicht definiert ist. Bei definiertem P geht es zum mit P bezeichneten Satz.
 Parameter: P - Satznummer
 Rücksprung: 1

Name: NOTHING()
 Aktion: Keine Aktion
 Parameter: Keine
 Rücksprung: 1

Name: STARTLOOP(P1,P2,P3)
 Aktion: Führt Satz P1 bis einschließlich P2 aus. Die Schleife wird P3-mal ausgeführt. Wenn P3 Null oder weniger ist, dann werden die Zeilen P1 bis P2 übersprungen. Verschachtelte Schleifen sind möglich, wenn die inneren Schleifen nicht die Begrenzung der äußeren Schleifen übersteigen.
 Z.B. N10 ... STARTLOOP(10,60,3)
 N20 ... STARTLOOP(20,50,2)
 N30 ...
 N40 ...
 N50 ...
 N60 ...
 Parameter: P1 - Start-Satznummer der Schleife
 P2 - Ende-Satznummer der Schleife
 P3 - Anzahl der Durchläufe der Schleife
 Rücksprung: 1

Name: PRINTERROR("beliebiger Text")
 Aktion: Gibt eine Fehlermeldung aus und stoppt die Anwendung.
 Z.B. PRINTERROR("Fehlender X-Wert").
 Parameter: Angegebener Text
 Rücksprung: 1

Name: PRINTMESSAGEBOX("Beliebiger Text")
 Aktion: Gibt eine Meldung aus.
 Z.B. PRINTMESSAGEBOX("HELLO WORLD").
 Parameter: Angegebener Text
 Rücksprung: 1

Kommentare

Kommentare können auf zwei Arten in den Programmcode eingebettet werden:

- Kommentar eingeschlossen zwischen `/*` und `*/`
Z.B. `/* Dieser Code
ist ein
Kommentar */`
- Kommentar anfangen mit `//`. Der Kommentar endet am Zeilenende.
Z.B. `//Dies ist ein Kommentar`

Konstanten

Konstanten stehen zum Gebrauch in anwenderdefinierten Funktionsdateien zur Verfügung.

UNDEFINED	Wert einer Variablen, die zwar deklariert aber nicht definiert wurde.
PI	3.14159265359
TRUE	1
FALSE	0

Operatoren - Priorität

Die folgenden Operatoren werden nach höherer zu niedrigerer Priorität unterstützt und aufgelistet.

Symbol	Beschreibung	Assoziation
	Höchste Priorität	
<code>*</code> , <code>/</code>	Multiplikat., Division	Links nach rechts
<code>+</code> , <code>-</code>	Addition, Subtraktion	Links nach rechts
<code>!=</code>	Ungleich	
<code>==</code>	Gleich	
<code><</code>	Kleiner als	
<code><=</code>	Kleiner gleich	
<code>></code>	Größer als	
<code>>=</code>	Größer gleich	
AND	Boolesches und	Links nach rechts
OR	Boolesches oder	Links nach rechts
<code>:=</code>	Zuweisung	Rechts nach links
	Niedrigste Priorität	

Sprache - Definition

Programm :

PROGRAM globale Deklaration, Funktions-Deklarationsliste,
ENDOFFILE

Funktions-Deklarationsliste :

{ FUNCTION, Identifizierung (Parameterliste), lokale Deklaration,
Anweisungsliste, Rücksprung-Anweisung, ENDFUNCTION }

Globale Deklarierungen :

globale Identifizierungs-Liste
|

Lokale Deklarierungen :

lokale Identifizierungs-Liste
|

Funktions-Aufruf :

Identifikation (Parameterliste)
| vordefinierte Funktion (Parameterliste)
| PRINTMESSAGEBOX (" String ")
| PRINTERROR (" String ")

Parameterliste

Identifikations-Liste
|

Identifikations-Liste

Identifikation { , Identifikation }

Anweisungsliste :

{ Anweisung }

Anweisung :

While-Anweisung
| Zuordnungs-Anweisung
| bedingte Anweisung
| Funktionsaufruf

While-Anweisung :

WHILE (Ausdruck) DO Anweisungsliste ENDWHILE

Zuordnungs-Anweisung :

Identifikation := Ausdruck

Bedingte Anweisung :

IF (Ausdruck) THEN Anweisungsliste ENDIF

| IF (Ausdruck) THEN Anweisungsliste ELSE Anweisungsliste ENDIF

Rücksprung-Anweisung :

RETURN Ausdruck

Ausdruck :

Ausdruck2

| Ausdruck OR Ausdruck2

Ausdruck2 :

Ausdruck3

| Ausdruck2 AND Ausdruck3

Ausdruck3 :

Summe

| Summe Vergleichsoperator Summe

Summe :

Produkt

| Summe + Produkt

| Summe – Produkt

Produkt :

Element

| Produkt / Element

| Produkt * Element

Element :

Identifikation

| Zahl

| (Ausdruck)

| Funktions-Aufruf

Vergleichsoperatoren :

<

| <=

| >

| >=

| !=