

Industrielle Automation

Rechnergestütztes Konstruieren

Format zum Austausch geometrischer Informationen

DIN
66 301

Industrial automation; computer aided design; format for the exchange of geometrical informations

1 Anwendungsbereich und Zweck

Diese Norm schafft die Grundlage für einen Datenaustausch zwischen verschiedenen rechnergestützten Entwurfs- und Fertigungssystemen. Dabei ist die vorliegende Norm als ein erster Schritt von sehr viel umfassenderen Festlegungen anzusehen, die dringend erforderlich sind und die in der nächsten Zukunft Zug um Zug erarbeitet werden sollen.

Die vorliegende Norm legt die Syntax und Semantik eines Formats für den Austausch geometrischer Informationen, hauptsächlich für den Bereich von Freiformflächen, fest. Die Syntax ist formal definiert. Die Semantik ist durch geeignete Wahl von Variablennamen, Erläuterungen, Abbildungen und Beispielen erläutert.

2 Begriffe

Begriffe siehe DIN 44 300, DIN 44 302, DIN 66 257 und DIN ISO 7498 (z. Z. Entwurf).

3 Datenformate

3.1 Datei-Struktur

3.1.1 Die Gesamtmenge der übertragenen Daten, beginnend mit Anfangskennung und schließend mit Endkennung, wird als eine Datei bezeichnet. Die Datei besteht aus Sätzen (Zeilen) zu je 80 Zeichen des 7-Bit-Code nach DIN 66 003.

Die Darstellung auf Datenträgern erfolgt nach den Normen der Reihe DIN 66 004. Bei Verwendung von 8-Bit-Zeichen ist das höchstwertige Bit 0.

3.1.2 Die Sätze enthalten in den Spalten 73 bis 80 rechtsbündig eine Folgenummer. Die Nummerung ist aufsteigend, aber nicht notwendig lückenlos.

3.1.3 Die logischen Bestandteile der Datei sind geometrische Elemente, Strukturierungen sowie Kommentare. Sie beginnen jeweils auf einer neuen Zeile.

3.2 Sprachelemente

Die Elemente haben folgenden Aufbau:

Name = Befehlswort
 oder Name = Befehlswort/Elementparameter

Je Element ist nur ein Name und ein Befehlswort zulässig. Name, Gleichheitszeichen (=) und Befehlswort müssen in einer Zeile stehen.

Als Parameter sind nur Zahlen erlaubt.

3.2.1 Zahlen

Ganze Zahlen (INTEGER) sind in der Form $\pm k_1 \dots k_n$ anzugeben, wobei k_i Ziffern zwischen 0 und 9 sind. Negativen Zahlen muß das Minuszeichen, positiven Zahlen kann das Pluszeichen vorangestellt werden. Vorzeichenlose Zahlen sind positiv. Es dürfen nicht mehrere Vorzeichen hintereinander verwendet werden. Der Betrag der Zahl ist kleiner als 2^{31} .

Reelle Zahlen (Gleitpunktzahlen, REAL) müssen in einer der folgenden Formen dargestellt werden.

a) $\pm x_1 \dots x_n \cdot y_1 \dots y_m$

$n \geq 0, m \geq 0$

$1 \leq (n + m) \leq \text{realim}$ oder

b) $\pm x_1 \dots x_n \cdot y_1 \dots y_m E \pm z_1 \dots z_e$

$n \geq 0, m \geq 0, 1 \leq e \leq 2, e$ ganzzahlig

$1 \leq (n + m) \leq \text{realim}$

x, y, z sind Ziffern zwischen 0 und 9

realim ist die maximale Anzahl an Ziffern, die in einer reellen Zahl auftreten darf, sie beträgt hier 16.

Der Dezimalpunkt muß immer vorhanden sein. Das Zeichen „E“ kann in der Exponentialschreibweise durch das Zeichen „D“ ersetzt werden.

3.2.2 Befehlsworte

Befehlsworte kennzeichnen den Elementtyp. Es sind nur die in den Abschnitten 4 und 5 näher beschriebenen Befehlsworte zulässig.

3.2.3 Trennzeichen und Sonderzeichen

Als Trennzeichen dienen die Sonderzeichen „=“ (Gleichheitszeichen), „/“ (Schrägstrich) und „.“ (Komma).

Das Gleichheitszeichen trennt den Namen vom Befehlswort.

Der Schrägstrich trennt das Befehlswort von den Elementparametern.

Trennzeichen zwischen den einzelnen Parametern ist das Komma.

3.2.4 Namen

Die Namen der Elemente bestehen aus Großbuchstaben (A bis Z) und den Ziffern (0 bis 9). Das erste Zeichen muß ein Buchstabe sein. Der Name besteht aus 1 bis 8 Zeichen. Es gibt keine reservierten Zeichenkombinationen. Die Namen müssen innerhalb einer Datei eindeutig sein.

Fortsetzung Seite 2 bis 9

Normenausschuß Maschinenbau (NAM) im DIN Deutsches Institut für Normung e. V.
 Normenausschuß Informationsverarbeitungssysteme (NI) im DIN

Jede Art der Vervielfältigung, auch auszugsweise, nur mit Genehmigung des DIN Deutsches Institut für Normung e. V., Berlin, gestattet.

3.3 Regeln für das freie Format

Die Darstellung erfolgt in einem freien Format, wobei folgende Regeln zu beachten sind:

3.3.1 Leerzeichen vor und nach den Trennzeichen sowie vor dem Namen werden beim Interpretieren ignoriert.

3.3.2 Ganze Zahlen, reelle Zahlen, Namen und Befehlsörter müssen stets in der Zeile enden, in der sie beginnen, einschließlich des evtl. folgenden Trennzeichens.

3.3.3 Trennzeichen zwischen den einzelnen Parametern ist das Komma.

3.3.4 Eine mögliche Folgezeile wird daran erkannt, daß das letzte von einem Leerzeichen unterschiedliche Zeichen in Spalte 1 bis 72 der vorhergehenden Zeile ein Komma ist.

3.3.5 Kommentarzeilen besitzen keine Folgezeilen, jede Kommentarzeile muß am Anfang durch „\$\$“ als eine Kommentarzeile gekennzeichnet werden.

3.3.6 Die Namen der Befehlsörter dürfen nur mit Großbuchstaben geschrieben werden.

3.4 Notation für die Beschreibung des Sprachaufbaus

3.4.1 Großbuchstaben sind unverändert zu übernehmen.

Beispiel:

POINT sind die folgenden Zeichen nach DIN 66 003: P, O, I, N und T.

3.4.2 Wörter, die mit Kleinbuchstaben beginnen, bezeichnen eine Variable. Sie ist durch ihre aus dem jeweiligen Problem gegebene Zeichenkette oder ihren Zahlenwert zu ersetzen.

Beispiel:

name ist durch den Namen zu ersetzen, der im konkreten Fall gemeint ist, etwa MAX oder MORITZ. vx ist im konkreten Fall durch einen Zahlenwert wie etwa 1401.54 zu ersetzen.

3.4.3 Ziffern sind unverändert zu übernehmen, wenn sie alleine stehen (wie etwa -60 oder 29.03) oder wenn ihnen ein oder mehrere Großbuchstaben vorausgehen (z. B. MAX1 oder KAR07 oder P27).

3.4.4 Die Schreibweise „(n) * [...]“ bedeutet, daß der Inhalt der Klammer so oft zu wiederholen ist, wie n (natürliche Zahl) angibt. Wiederholungen werden jeweils durch ein Komma getrennt.

Beispiel:

(2) * [A, B] bedeutet: A, B, A, B

3.4.5 Stehen in den eckigen Klammern Variable (Kleinbuchstaben), dann kann ihr Wert bei jeder Wiederholung verschieden sein.

Beispiel:

(4) * [x] kann bedeuten: 124., 48.345, -12.47E34, .78

3.4.6 Vor der eckigen Wiederholungsklammer können auch Variable (Kleinbuchstaben) stehen. Die Anzahl der Wiederholungen entspricht dann dem Zahlenwert der Variablen (positiv Integer).

Beispiel:

(n) * [x, y, z] $1 \leq n \leq (2^{31}) - 1$

3.4.7 Vor der eckigen Wiederholungsklammer können auch Ausdrücke mit Variablen stehen. Diese werden durch „(“ und „)“ geklammert. Die Anzahl der Wiederholungen entspricht dann dem aktuellen Wert des geklammerten Ausdruckes.

Beispiel:

((iordu · iordv) * [kx])

3.4.8 Die Wiederholungsklammern können geschachtelt auftreten.

Beispiel:

(n) * [x, y, z] entspricht vollständig

(n) * [(3) * [koord]]

4 Geometrische Elemente

4.1 Typen

Folgende geometrischen Elemente können definiert werden:

Typ	Befehlswort	siehe Abschnitt
Punkt	POINT	4.5
Punktfolge	PSET	4.6
Punkt-Vektor-Folge (Master Dimension)	MDI	4.7
Kurve	CURVE	4.8
Fläche	SURF	4.9

4.2 Maßeinheiten

Es sind nur Millimeter (mm) als Maßeinheiten zugelassen.

4.3 Koordinatensystem

Ein rechtshändiges, rechtwinkliges Koordinatensystem mit den Achsen x, y und z nach DIN 66 217 ist zu verwenden.

4.4 Sprachaufbau

Der allgemeine Sprachaufbau für die geometrischen Elemente sieht folgendermaßen aus:

name = befehlswort / parameter

Die geometrischen Elemente lassen nur die im Abschnitt 4.1 aufgeführten Befehlswörter zu.

4.5 Punkt

Syntax: name = POINT / x, y, z

Parameter: x, y, z: real Punktkoordinaten

Gültige Beispiele: P01 = POINT / 10.5, -200., +23E12

P02 = POINT / 2.12, 34., -47.231

Ungültig: P03 = POINT / 12, 34.23, -85.42
Integerwert als Punktkoordinate

P04 = POINT / -23.42, 12.53
Eine Punktkoordinate fehlt.

4.6 Punktfolge

Syntax: name = PSET / n, (n) * [x, y, z]

Parameter: n: integer / x, y, z: real

n Anzahl der Punkte
x, y, z Punktkoordinaten

Gültige PSET = PSET / 1, 12., -12.82, .6287E-12
Beispiele: PS01 = PSET / 2, 0., 12.5, 12.9, -3.1, 4.2, 5.1

Ungültig: PS02 = PSET / 2, 0., 12.1, 3.4, 1.9, 3.3, 5.98
Anzahl der Punkte ist als Gleitpunktzahl angegeben.
PS03 = PSET / 3, 0., 12.1, 3.4, 2.9, 3.3, 5.98
Anzahl der Punkte stimmt nicht mit der folgenden Anzahl der Punktkoordinaten überein.

Anmerkung: PSET kann eingesetzt werden zur Übertragung von Polygonzügen oder Kurven, die durch Punktdaten definiert sind

4.7 Punkt-Vektor-Folge

Syntax: name = MDI / n, (n) * [x, y, z, vx, vy, vz]

Parameter: n: integer / x, y, z, vx, vy, vz: real

n Anzahl der Sechstupel
x, y, z Punktkoordinaten
vx, vy, vz Vektorkomponenten

Gültiges

Beispiel: M1 = MDI / 2, -11.47, 41.12, 42.67, 1.34, 9.2, 1., 1.933, 3.12, 5.34, 4.13, .91, .32

Ungültig: M2 = MDI / 1, 2.27, 48.18, 5.85, .43, .23, .45, Letztes Komma überschüssig.

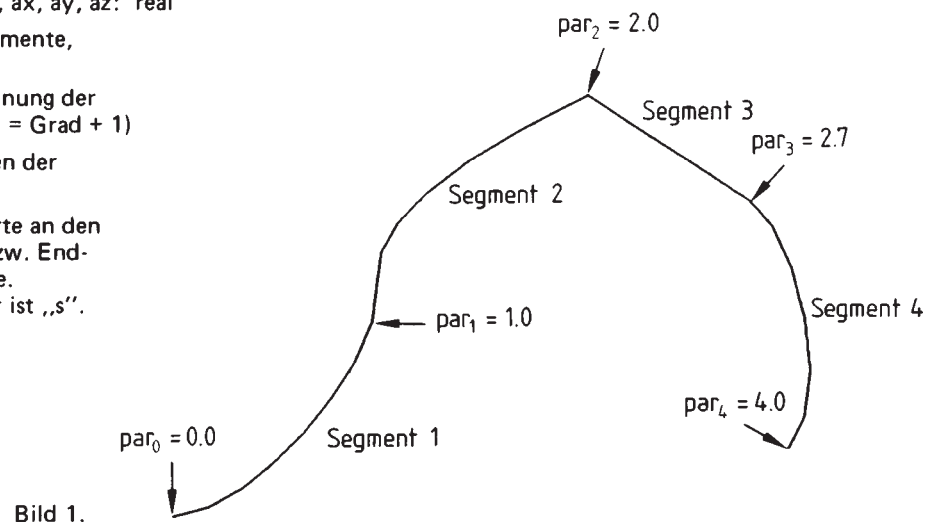
Anmerkung: MDI kann eingesetzt werden, um NC-Punktfolgen mit den dazugehörigen Normalenvektoren nach DIN 66 246 Teil 1 zu übertragen. Der Normalenvektor muß nicht auf eine normierte Länge festgelegt sein.

4.8 Kurve

Syntax: name = CURVE / n, (n + 1) * [par],
(n) * [iord, (iord) * [ax],
(iord) * [ay],
(iord) * [az]]

Parameter: n, iord: integer, par, ax, ay, az: real

n Anzahl der Kurvensegmente, die eine Kurve bilden
iord Jeweilige Polynomordnung der Kurvensegmente (iord = Grad + 1)
ax, ay, az Jeweilige Koeffizienten der Kurvensegmente
par Globale Parameterwerte an den jeweiligen Anfangs- bzw. Endpunkten der Segmente. Der globale Parameter ist „s“.



Beispiel: Numerische Werte zu Bild 1

SP1 = CURVE / 4, 0., 1., 2., 2.7, 4.,

5,	0.0000000,	63.8891754,	- 51.4652710,	77.0936432,
	- 48.5175476,	0.0000000,	6.4475098,	56.9390259,
	- 23.5717163,	3.1851807,	30.0000000,	0.0000000,
	- 60.0000000,	40.0000000,	- 10.0000000,	
3,				
	41.0000000,	- 0.9153137,	41.9153137,	43.0000000,
	31.1755676,	1.8244324,	0.0000000,	- 20.0000000,
	15.0000000,			
2,				
	82.0000000,	29.0000000,	76.0000000,	- 21.0000000,
	- 5.0000000,	- 5.0000000,		
4,				
	111.0000000,	26.5623550,	- 10.8701935,	- 10.6921616,
	55.0000000,	- 19.2350578,	- 77.3120193,	49.5470734,
	- 10.0000000,	30.0000000,	- 30.0000000,	10.0000000,