

# Befehlskatalog und Statusmeldungen der Steuerungen

## SMCU

## LPKF MotionSystem

Technische Beschreibung

---

LPKF Moiton & Control GmbH  
Mittelbergstraße 17  
D-98527 Suhl

Tel.           (++49 36 81) 89 24 0  
Fax.           (++49 36 81) 89 24 44  
E-Mail        [info@lpkf-mc.de](mailto:info@lpkf-mc.de)

# Inhaltsverzeichnis

<b>HPGL-Befehle</b>	<b>3</b>
<b>Zur Syntax der Befehle</b>	<b>3</b>
<b>Standardbefehle</b>	<b>4</b>
<b>Sonderbefehle</b>	<b>8</b>
<b>Direkt-Befehle</b>	<b>20</b>
<b>Grenzwerte von Eingabeparametern</b>	<b>21</b>
<b>Statusmeldungen der SMCU</b>	<b>22</b>
<b>Statusmeldungen des MotionSystems</b>	<b>23</b>
<b>Reaktion des Treiberprogramms auf bestimmte Statusmeldungen</b>	<b>25</b>

## HPGL-Befehle

HPGL ( Hewlett Packard Graphic Language ) ist ein weitverbreitetes Standardformat zur Ansteuerung von Plottern aller Art.

Eine als Datei abgelegte HPGL - Befehlsreihe kann grundsätzlich mit einem normalen Texteditor gelesen und verarbeitet werden. Damit lassen sich solche Dateien sehr leicht verändern und auch generieren.

Die LPKF-Steuerungen MotionSystem interpretieren die im folgenden beschriebenen HPGL-Befehle und setzt sie in die definierten Reaktionen um.

Die Implementierung der einzelnen Befehle ist von der Konfiguration der jeweiligen Steuerung abhängig.

Zur Syntax der Befehle

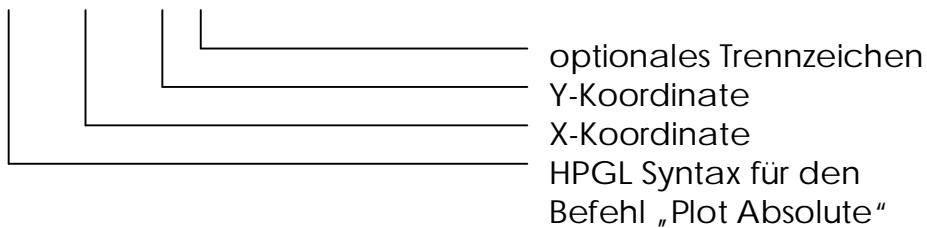
Symbol	Bedeutung
[...]	Der Inhalt kann beliebig oft wiederholt werden.
()	Zwischen diesen Klammern stehen die Parameter des Befehls
{}	Der Inhalt dieser Klammern ist optional und kann entfallen.

Die Steuerung erwartet zwischen den Parametern der Befehle ein Trennzeichen, das nicht Element der Zahlen und Großbuchstaben ist. Ein neuer Befehl darf ohne Trennzeichen auf einen Parameter folgen.

Das letzte Zeichen einer gesendeten Befehls-Datei muß ein Semikolon oder ein Carriage-Return - Zeichen (0D Hex) sein. Unbekannte Befehle werden von der Steuerung ignoriert.

Beispiel für die Schreibweise eines Befehls:

PA 1000,2000;



## Standardbefehle

### AA (x,y,a){;}

Name	Arc Absolute
Befehlskode	CMDT_ARC_ABSOLUTE
Beschreibung	Zeichnet um die absolute Koordinate (x,y) ausgehend von der aktuellen Position einen Kreisbogen mit dem Bogenwinkel a = [Grad]. Bei negativem Bogenwinkel wird im Uhrzeigersinn sonst entgegen dem Uhrzeigersinn gezeichnet.
Implementierung	SMCU: ja                      MotionSystem: ja

### AR (x,y,a){;}

Name	Arc Relative
Befehlskode	CMDT_ARC_RELATIVE
Beschreibung	Zeichnet um die relative Koordinate (x,y) ausgehend von der aktuellen Position einen Kreisbogen mit dem Bogenwinkel a = [Grad]. Bei negativem Bogenwinkel wird im Uhrzeigersinn sonst entgegen dem Uhrzeigersinn gezeichnet.
Implementierung	SMCU: ja                      MotionSystem: ja

### IN {;}

Name	Initialize
Befehlskode	CMDT_SYSTEM_RESET
Beschreibung	Versetzt die Steuerung in den gleichen Status wie nach dem Einschalten. Alle benutzerspezifischen Einstellungen gehen verloren.
Implementierung	SMCU: ja                      MotionSystem: ja

### IW (x0,y0,x1,y1){;}

Name	Input Window
Befehlskode	CMDT_INPUT_WINDOW
Beschreibung	Setzt den Bewegungsbereich der XY-Achse auf ein Fenster mit den in Schritten angegebenen Eck-Koordinaten (x0,y0) und (x1,y1).
Implementierung	SMCU: ja                      MotionSystem: ja

## OH {:}

Name	Output Hard Clip Limits
Befehlskode	CMDT_HARDCLIP_LIMITS
Beschreibung	<p>Die Steuerung ermittelt selbsttagig fur jede Achse den maximalen Bewegungs-bereich innerhalb der Endschalter und sendet die so ermittelten Koordinaten in Form eines ASCII-String:</p> <p>SMCU: "W Xmax, Ymax, Z0max&lt;cr&gt;" MotionSystem: "W Xmax, Ymax, Z0max...Znmax &lt;cr&gt;"</p> <p>an den PC. Die Koordinaten werden in Schritten angegeben. ACHTUNG: Das Bewegen der Achsen auf den Nullpunkt zuruck erfolgt mit der zuletzt definierten Geschwindigkeit. Sehr kleine Geschwindigkeiten fuhren zu sehr langen Befehlsabarbeitungszeiten.</p>
Implementierung	SMCU: ja                      MotionSystem: ja

## OH (n){:}

Name	Output Hard Clip Limits
Befehlskode	CMDT_HARDCLIP_LIMITS
Beschreibung	<p>Die Steuerung ermittelt in der durch den Parameter n definierten Achse den maximalen Bewegungsbereich innerhalb der Endschalter und sendet die so ermittelte Koordinate in Form des nachfolgend aufgefuhrten ASCII-Strings:</p> <p>SMCU: "W Achsemax&lt;cr&gt;", MotionSystem: "W Achsemax&lt;cr&gt;"</p> <p>an den PC. Die Koordinaten werden in Schritten angegeben. ACHTUNG: Das Bewegen der Achse auf den Nullpunkt zuruck erfolgt mit der zuletzt definierten Geschwindigkeit. Sehr kleine Geschwindigkeiten fuhren zu sehr langen Befehlsabarbeitungszeiten. Der Parameter n definiert die Achse. 0 = X-Achse 1 = Y-Achse 2 = Z0-Achse 3 = Z1-Achse 4 = Z2-Achse 5 = Z3-Achse</p>
Implementierung	SMCU: in Vorbereitung              MotionSystem: ja

## OS {;}

Name Output Status

Befehlskode CMDT\_OUTPUT\_STATUS

Beschreibung Die Steuerung sendet den Inhalt ihrer Statuszellen als ASCII-String ("Sxxx,yyy,zzz<cr>") an den PC.  
Dabei ist:  
xxx – Versionsnummer der Software (dreistellig)  
yyy – Erstellungsdatum der Version (fünf- oder sechsstellig)  
10696 – 1. Juni 1996  
300495 – 30. April 1995  
zzz – Kennung der Steuerung (sechsstellig)  
1xxxxx - SMCU  
2xxxxx - LPKF MotionSystem

Implementierung SMCU: ja MotionSystem: ja

## PA (x,y){;}

Name Plot Absolute

Befehlskode CMDT\_PLOT\_ABSOLUTE

Beschreibung Führt eine Bewegung der X- und Y-Achse von der aktuellen Position zu der in absoluten Koordinaten angegebenen Position (x,y) aus.

Implementierung SMCU: ja MotionSystem: ja

## PD{;}

Name Pen Down

Befehlskode CMDT\_PEN\_DOWN

Beschreibung Das Kommando senkt den Bohr-/Fräskopf der Maschine ab. Vor dem Absenken des Kopfes wird je nach Betriebsart des Maschine, Bohr- oder Fräsmodus, die Zeit td1 bzw. tm1 gewartet. Nach Ablauf dieser Zeit wird der Kopf abgesenkt und danach die Zeit td2 bzw. tm2 gewartet. Weiterhin verwendet die Steuerung nach dem Absenken des Kopfes die Geschwindigkeit, die durch das VSv; Kommando gesetzt wurde. Die Wartezeiten werden durch das !TD;- bzw. TM;-Kommando gesetzt. Das Kommando !CM; schaltet zwischen den beiden Arbeitsmodi, Bohren oder Fräsen, der Maschine um.

Implementierung SMCU: ja MotionSystem: nein

## PR (x,y){;}

Name Plot Relative

Befehlskode CMDT\_PLOT\_RELATIVE

Beschreibung Führt eine Bewegung der X- und Y-Achse von der aktuellen Position zu der in relativen Koordinaten angegebenen Position (x,y) aus.

Implementierung SMCU: ja MotionSystem: ja



## Sonderbefehle

Alle Sonderbefehle beginnen mit einem "!" Zeichen und entsprechen im weiteren der Syntax der HPGL-Standardbefehle.

### !AS (a,n){;}

Name	Acceleration Set
Befehlskode	CMDT_ACCELERATION_SET
Beschreibung	Definiert eine neue Beschleunigungskonstante. Für translatorische Achsen gilt die Einheit $a = [\text{mm/s}^2]$ und für rotatorische Achsen die Einheit $a = [\text{mrad/s}^2]$ . Der Parameter n bezeichnet die Bewegungsachse, welcher der Wert zugeordnet wird. 0 = X-Achse 1 = Y-Achse 2 = Z0-Achse 3 = Z1-Achse 4 = Z2-Achse 5 = Z3-Achse
Implementierung	SMCU: in Vorbereitung      MotionSystem: ja

### !AS (a){;}

Name	Acceleration Set
Befehlskode	CMDT_ACCELERATION_SET
Beschreibung	Definiert eine neue Beschleunigungskonstante. Für translatorische Achsen gilt die Einheit $a = [\text{mm/s}^2]$ und für rotatorische Achsen die Einheit $a = [\text{mrad/s}^2]$ .
Implementierung	SMCU: ja      MotionSystem: nein

### !CC {;}

Name	Close Channel
Befehlskode	CMDT_CLOSE_CHANNEL
Beschreibung	Schließt den mit !OC geöffneten Datenkanal
Implementierung	SMCU: ja      MotionSystem: ja

### !CC (n){;}

Name	Close Channel
Befehlskode	CMDT_CLOSE_CHANNEL
Beschreibung	Schließt den mit !OC n geöffneten Datenkanal
Implementierung	SMCU: in Vorbereitung      MotionSystem: ja

**!CI {;}**

Name	Close Interface
Befehlskode	CMDT_CLOSE_INTERFACE
Beschreibung	Schließt das mit !OI n geöffnete Interface.
Implementierung	SMCU: ja                      MotionSystem: ja

**!CM (n){;}**

Name	Change Mill- / Drill-Mode
Befehlskode	CMDT_MILL_DRILL_MODE
Beschreibung	Das Kommando wechselt die Wartezeiten für die PD;- / PU;- Kommandos der Maschine aus. n = 0 - Bohr-Mode (Drilling) n = 1 - Fräs-Mode (Milling)
Implementierung	SMCU: ja                      MotionSystem: nein

**!CS{;}**

Name	Continue after unusual machine status
Befehlskode	CMDT_CONTINUE_AFTER_UNSTATE
Beschreibung	Setzt das Abarbeiten von Kommandos, insbesondere von Bewegungs-kommandos, die durch einen speziellen Maschinenstatus unterbrochen wurden, fort. Dieser besondere Maschinenstatus kann z.B. durch externe Signale an einer IO-Karte ausgelöst worden sein.
Implementierung	SMCU: nein                      MotionSystem: ja

**!CT (n){;}**

Name	Command Counter
Befehlskode	CMDT_ENABLE_COMMAND_COUNTER
Beschreibung	Wechselt zwischen dem Echo-Mode und dem Non-Echo-Mode. Im Echo-Mode quittiert die Steuerung jedes ausgeführte Kommando mit der Meldung "C<cr>". Nach dem Einschalten ist der Echo-Mode aktiviert. n=0 – Echo Mode inaktiv n=1 – Echo Mode aktiv
Implementierung	SMCU: ja                      MotionSystem: ja

**IES (n){;}**

Name	Enable Stop
Befehlskode	CMDT_ENABLE_FAST_STOPP
Beschreibung	Freigeben und Sperren der externen Stop-Funktion. Nach dem Einschalten ist die externe Stop-Funktion immer gesperrt. n = 0 - Stop-Funktion sperren n = 1 - Stop-Funktion aktivieren
Implementierung	SMCU: ja                      MotionSystem: ja

**IEM (n){;}**

Name	External Motor On/Off
Befehlskode	CMDT_EXTERNAL_MOTOR_ON_OFF
Beschreibung	Schaltet die Bohr-/Frässpindel der Maschine an bzw. aus. n = 0 - Bohr-/Frässpindel aus n = 1 - Bohr-/Frässpindel an
Implementierung	SMCU: ja                      MotionSystem: ja

**IFP{;}**

Name	Power Up
Befehlskode	CMDT_POWER_UP
Beschreibung	Die Funktion schaltet die Halbierung des Phasenstroms in den Schrittmotorendstufen der Steuerung aus.
Implementierung	SMCU: ja                      MotionSystem: nein

**IGB{;}**

Name	Go Back after unusual machine status
Befehlskode	CMDT_GOBACK_AFTER_UNSTATE
Beschreibung	Bewegt alle Achsen des Antriebssystems auf den Startpunkt des letzten Bewegungskommandos zurück, wenn dieses Kommando durch einen speziellen Maschinenstatus unterbrochen wurde. Dieser besondere Maschinenstatus kann z.B. durch externe Signale an einer IO-Karte ausgelöst worden sein.
Implementierung	SMCU: nein                      MotionSystem: ja

**IHP{;}**

Name	Power Down
Befehlskode	CMDT_POWER_DOWN
Beschreibung	Die Funktion schaltet die Halbierung des Phasenstroms in den Schrittmotorendstufen der Steuerung ein. Dadurch wird die Verlustleistung und damit auch die Erwärmung der Schrittmotoren während ihres Stillstandes reduziert.
Implementierung	SMCU: ja                      MotionSystem: nein

**IIS {;}**

Name	Init System
Befehlskode	CMDT_INITIALIZE_SYSTEM
Beschreibung	Alle Achsen fahren nacheinander auf ihren Referenzpunkt und bleiben dort stehen.
Implementierung	SMCU: ja                      MotionSystem: ja

**IIS (n){;}**

Name	Init System
Befehlskode	CMDT_INITIALIZE_SYSTEM
Beschreibung	Die durch den Parameter n definierte Achse fährt auf ihren Referenzpunkt und bleibt dort stehen. Der Parameter n definiert die Achse. 0 = X-Achse 1 = Y-Achse 2 = Z0-Achse 3 = Z1-Achse 4 = Z2-Achse 5 = Z3-Achse
Implementierung	SMCU: in Vorbereitung      MotionSystem: ja

**IMA (x,y,z0,z1,z2,z3){;}**

Name	Move All Axes Absolute
Befehlskode	CMDT_MOVE_ALL_AXES_ABSOLUTE
Beschreibung	Führt eine Bewegung aller Achsen von der aktuellen Position zur dazu in absoluten Koordinaten angegebenen Position (x,y,z0,z1,z2,z3) aus.
Implementierung	SMCU: in Vorbereitung      MotionSystem: ja

### **!MR (x,y,z0,z1,z2,z3){;}**

Name	Move All Axes Relative
Befehlskode	CMDT_MOVE_ALL_AXES_RELATIVE
Beschreibung	Führt eine Bewegung aller Achsen von der aktuellen Position zur dazu in relativen Koordinaten angegebenen Position (x,y,z0,z1,z2,z3) aus.
Implementierung	SMCU: in Vorbereitung      MotionSystem: ja

### **!OA (n){;}**

Name	Output Actual Position																																										
Befehlskode	CMDT_OUTPUT_ACTUAL_POSITION																																										
Beschreibung	<p>Die Steuerung sendet die aktuelle Achs-Position ( Ist-Position am Meßsystem ). Der Parameter n bestimmt die Motorachse:</p> <table><tr><td>0</td><td>= X, Y, Z0, Z1, Z2, Z3 -Achse</td><td>/ mit externen Trigger</td></tr><tr><td>1</td><td>= X-Achse</td><td>/ mit externen Trigger</td></tr><tr><td>2</td><td>= Y-Achse</td><td>/ mit externen Trigger</td></tr><tr><td>3</td><td>= Z0-Achse</td><td>/ mit externen Trigger</td></tr><tr><td>4</td><td>= Z1-Achse</td><td>/ mit externen Trigger</td></tr><tr><td>5</td><td>= Z2-Achse</td><td>/ mit externen Trigger</td></tr><tr><td>6</td><td>= Z3-Achse</td><td>/ mit externen Trigger</td></tr><tr><td>100</td><td>= X, Y, Z0, Z1, Z2, Z3 -Achse</td><td></td></tr><tr><td>101</td><td>= X-Achse</td><td></td></tr><tr><td>102</td><td>= Y-Achse</td><td></td></tr><tr><td>103</td><td>= Z0-Achse</td><td></td></tr><tr><td>104</td><td>= Z1-Achse</td><td></td></tr><tr><td>105</td><td>= Z2-Achse</td><td></td></tr><tr><td>106</td><td>= Z3-Achse</td><td></td></tr></table> <p>Die Ist-Position bei Parameter n = 0..6 kann nur gelesen werden, wenn der Ist-Zähler zuvor getriggert wurde ( siehe technische Beschreibung der entsprechenden Zählerbaugruppe ). Die Ist-Position bei Parameter n = 100..106 wird direkt gelesen. Die Istposition wird in Form des nachfolgenden ASCII-Strings an den PC gesendet: "AXist, Yist, Z0ist, Z1ist, Z2ist, Z3ist&lt;cr&gt;".</p>	0	= X, Y, Z0, Z1, Z2, Z3 -Achse	/ mit externen Trigger	1	= X-Achse	/ mit externen Trigger	2	= Y-Achse	/ mit externen Trigger	3	= Z0-Achse	/ mit externen Trigger	4	= Z1-Achse	/ mit externen Trigger	5	= Z2-Achse	/ mit externen Trigger	6	= Z3-Achse	/ mit externen Trigger	100	= X, Y, Z0, Z1, Z2, Z3 -Achse		101	= X-Achse		102	= Y-Achse		103	= Z0-Achse		104	= Z1-Achse		105	= Z2-Achse		106	= Z3-Achse	
0	= X, Y, Z0, Z1, Z2, Z3 -Achse	/ mit externen Trigger																																									
1	= X-Achse	/ mit externen Trigger																																									
2	= Y-Achse	/ mit externen Trigger																																									
3	= Z0-Achse	/ mit externen Trigger																																									
4	= Z1-Achse	/ mit externen Trigger																																									
5	= Z2-Achse	/ mit externen Trigger																																									
6	= Z3-Achse	/ mit externen Trigger																																									
100	= X, Y, Z0, Z1, Z2, Z3 -Achse																																										
101	= X-Achse																																										
102	= Y-Achse																																										
103	= Z0-Achse																																										
104	= Z1-Achse																																										
105	= Z2-Achse																																										
106	= Z3-Achse																																										
Implementierung	SMCU: nein      MotionSystem: ja																																										

### **!OC {;}**

Name	Open Channel
Befehlskode	CMDT_OPEN_CHANNEL
Beschreibung	Öffnet den Datenkanal zur seriellen Schnittstelle 2. Das Kommando !CC; schließt den Datenkanal.
Implementierung	SMCU: ja      MotionSystem: ja

### !OC (n){;}

Name	Open Channel
Befehlskode	CMDT_OPEN_CHANNEL
Beschreibung	Öffnet den mit n spezifizierten Datenkanal. Das Kommando !CCn; schließt den Datenkanal.
Implementierung	SMCU: in Vorbereitung      MotionSystem: ja

### !OI (n){;}

Name	Open Interface
Befehlskode	CMDT_OPEN_INTERFACE
Beschreibung	Öffnet das mit n spezifizierte Interface. n = 1 Der Trackball bewegt die selektierte Achse wobei stets die neue Ist-Position der Achse im Zeitintervall von 1 Sekunde bereitgestellt wird. n = 2 Der Trackball bewegt die selektierte Achse. n = 3 Der Trackball liefert nur seine Positionsdaten ( z.B. als Cursorsteuerung für einen Monitor ) Das Kommando !CI schließt das Interface.
Implementierung	SMCU: ja      MotionSystem: ja

### !ON (n){;}

Name	Output Nominal Position
Befehlskode	CMDT_OUTPUT_NOMINAL_POSITION
Beschreibung	Die Steuerung sendet die nominale Position (Soll-Position numerisch ermittelt) der mit dem Parameter n adressierten Motorachse. Den Motorachsen sind folgende Adressen zugeordnet: 0 = X, Y, Z0, Z1, Z2, Z3 -Achse 1 = X-Achse 2 = Y-Achse 3 = Z0-Achse 4 = Z1-Achse 5 = Z2-Achse 6 = Z3-Achse Die Sollposition wird in Form des nachfolgenden Strings an den PC gesendet: SMCU: "PXsoll, Ysoll, Z0soll<cr>", MotionSystem: "PXsoll, Ysoll, Z0soll, Z1soll, Z2soll, Z3soll<cr>".
Implementierung	SMCU: ja      MotionSystem: ja

**!PH(a,b)<cr>(string of parameter)<cr>(string of parameter)<cr>...<cr>;**

Name Following vector

Befehlskode CMTD\_FOLLOWING\_VECTOR

Beschreibung Führt eine Bewegung der durch die Parameter a und b definierten Achsen von der aktuellen Position zu einer in Form mehrerer relativer Koordinatenpaare gegebenen Endposition in Form eines Folgevektor aus. Folgevektor bedeutet, die Bewegung ist kontinuierlich. Die Anzahl der Koordinatenstrings ist auf maximal 65 begrenzt. Ein Parametersatz ist wie folgt aufgebaut:

X Wert Y Wert I Wert J Wert F Wert O Wert <cr>, Wert - long int Zahl (32 Bit).

X - relative X-Koordinate des Endpunktes in Bezug auf die aktuelle Position  
Y - relative Y-Koordinate des Endpunktes in Bezug auf die aktuelle Position.

Zulässige Werte:

negative Koordinate - Bewegung in Richtung Nullpunkt

positive Koordinate - Bewegung in Richtung positiver Endschalter

Null - Endpunkt liegt auf derselben X- bzw. Y-Koordinate wie der Startpunkt der Bewegung

I - relative X-Koordinate des Kreismittelpunktes in Bezug auf die aktuelle Position.

J - relative Y-Koordinate des Kreismittelpunktes in Bezug auf die aktuelle Position.

Zulässige Werte:

negative Koordinate - Mittelpunkt liegt in Richtung des Nullpunktes

positive Koordinate - Mittelpunkt liegt in Richtung des positiven Endschalters

Null - Mittelpunkt liegt auf derselben X- bzw. Y-Koordinate wie der Startpunkt

F - relative Geschwindigkeit am Ende des Vektor bezogen auf die Anfangsgeschwindigkeit. Die Maßeinheit ist um/s. Zulässige Werte:

negative Geschwindigkeit - Bremsen

positive Geschwindigkeit - Beschleunigen

Null - Konstantlauf

O - Orientierung des Kreises, Zulässige Werte:

1 - mathematisch positiver Drehsinn, entgegen dem Uhrzeigersinn

-1 - mathematisch negativer Drehsinn, im Uhrzeigersinn

0 - Gerade

Die Parameter a und b definieren die Achse.

0 = X-Achse

1 = Y-Achse

2 = Z0-Achse

3 = Z1-Achse

4 = Z2-Achse

5 = Z3-Achse

Implementierung SMCU: nein MotionSystem: ja

### !RD (a){;}

Name	Read Port
Befehlskode	CMDT_READ_PORT
Beschreibung	Bewirkt das Lesen eines Daten-Port mit der Adresse a. Der Portinhalt wird in Form des nachfolgend angeführten Strings an den PC gesendet: SMCU: "Rxxxx<cr>, xxxx = Portinhalt als Dezimalzahl MotionSystem: "Rxxxx<cr>, xxxx = Portinhalt als Dezimalzahl
Implementierung	SMCU: in Vorbereitung      MotionSystem: ja

### !RC {;}

Name	Repeat command after unusual machine status
Befehlskode	CMDT_REPEAT_AFTER_UNSTATE
Beschreibung	Wiederholt das durch einen besonderen Maschinenstatus unterbrochene Kommando. Dieser spezielle Maschinenstatus kann z.B. durch externe Signale an einer IO-Baugruppe ausgelöst werden.
Implementierung	SMCU: nein                      MotionSystem: ja

### !RS (r,n){;}

Name	Resolution Set
Befehlskode	CMDT_RESOLUTION_SET
Beschreibung	Definiert eine neue Schrittweite r = [nm/Schritt] für Translationsachsen und r = [(1/1000)*Winkelsekunde/Schritt] bei Rotationsachsen. Der Parameter n definiert die entsprechende Bewegungsachse: 0 = X-Achse 1 = Y-Achse 2 = Z0-Achse 3 = Z1-Achse 4 = Z2-Achse 5 = Z3-Achse
Implementierung	SMCU: in Vorbereitung      MotionSystem: ja

### !RS (r){;}

Name	Resolution Set
Befehlskode	CMDT_RESOLUTION_SET
Beschreibung	Definiert eine neue Schrittweite r = [nm/Schritt] für Translationsachsen und r = [(1/1000)*Winkelsekunde/Schritt] bei Rotationsachsen.
Implementierung	SMCU: ja                      MotionSystem: nein

**ISR (min,max,n){;}**

Name Set Range

Befehlskode CMDT\_SET\_MOVING\_RANGE

Beschreibung Begrenzt den Bewegungsbereich der Achse n auf den mit <min,max> definierten Weg. Die Angabe des Bewegungsbereiches erfolgt in Schritten. Der Parameter n bezeichnet die entsprechende Bewegungsachse:  
0 = X-Achse  
1 = Y-Achse  
2 = Z0-Achse  
3 = Z1-Achse  
4 = Z2-Achse  
5 = Z3-Achse

Implementierung SMCU: in Vorbereitung MotionSystem: ja

**ITD (td1, td2, td3, td4){;}**

Name Times for Drilling

Befehlskode CMDT\_TIMES\_FOR\_DRILLING

Beschreibung Setzt die Wartezeiten für das PD;- bzw. PU;- Kommando für den Bohr-Mode der Maschine.  
td1 – Wartezeit, bevor der Kopf abgesenkt wird  
td2 – Wartezeit, nachdem der Kopf abgesenkt wurde  
td3 – Wartezeit, bevor der Kopf angehoben wird  
td4 – Wartezeit, nachdem der Kopf angehoben wurde

Implementierung SMCU: ja MotionSystem: nein

**ITM (tm1, tm2, tm3, tm4){;}**

Name Times for Milling

Befehlskode CMDT\_TIMES\_FOR\_MILLING

Beschreibung Setzt die Wartezeiten für das PD;- bzw. PU;- Kommando für den Fräs-Mode der Maschine.  
tm1 – Wartezeit, bevor der Kopf abgesenkt wird  
tm2 – Wartezeit, nachdem der Kopf abgesenkt wurde  
tm3 – Wartezeit, bevor der Kopf angehoben wird  
tm4 – Wartezeit, nachdem der Kopf angehoben wurde

Implementierung SMCU: ja MotionSystem: nein

**ITS (t){;}**

Name Time to Stabilize

Befehlskode CMDT\_TIME\_TO\_STABILIZE

Beschreibung Setzt die Beruhigungszeit t=[ms] zwischen den einzelnen Bewegungs-befehlen.

Implementierung SMCU: ja MotionSystem: ja

**ITW (t){;}**

Name	Time to Waite
Befehlskode	CMDT_TIME_TO_WAIT
Beschreibung	Fügt eine einmalige Wartezeit ein. Der folgende Befehl wird erst nach Ablauf der Wartezeit t=[ms] abgearbeitet.
Implementierung	SMCU: ja                      MotionSystem: ja

### !VZ (v,n){;}

Name	Velocity of Axis
Befehlskode	CMDT_VELOCITY_OF_AXIS
Beschreibung	Definiert die Geschwindigkeit der Bewegung der Achse n. Die Maßeinheit der Geschwindigkeit ist bei Translationsachsen [µm/s] und bei Rotationsachse [0,1 Grad/s]. Der Parameter n bezeichnet die Bewegungsachse: 0 = X-Achse 1 = Y-Achse 2 = Z0-Achse 3 = Z1-Achse 4 = Z2-Achse 5 = Z3-Achse
Implementierung	SMCU: in Vorbereitung      MotionSystem: ja

### !VU (v){;}

Name	Velocity Up
Befehlskode	CMDT_VELOCITY_UP
Beschreibung	Definiert die Bahngeschwindigkeit in der XY-Ebene wenn die Maschine mit angehobenem Kopf eine Zustellbewegung durchführt. Die Maßeinheit ist [µm/s].
Impementierung	SMCU: ja                      MotionSystem: nein

**!VZ (v){;}**

Name	Velocity of Axis
Befehlskode	CMDT_VELOCITY_Z_AXIS
Beschreibung	Definiert die Geschwindigkeit der Bewegung der Z-Achse. Die Maßeinheit der Geschwindigkeit ist bei Translationsachsen [ $\mu\text{m/s}$ ] und bei Rotationsachse [0,1 Grad/s].
Implementierung	SMCU: ja                      MotionSystem: nein

**!WR (a,d){;}**

Name	Write Port
Befehlskode	CMDT_WRITE_PORT
Beschreibung	Bewirkt eine Ausgabe des Datenwortes d auf die Portadresse a. Die oberen 4 Bit der Adresse charakterisieren die I/O-Bank der gewählten I/O-Baugruppe und die unteren 4 Bit bezeichnen das jeweilige I/O-Port auf der Baugruppe.
Implementierung	SMCU: ja                      MotionSystem: ja

**!ZA (k,n){;}**

Name	Move one axis absolute
Befehlskode	CMDT_ONE_AXIS_ABSOLUTE
Beschreibung	Bewegt die durch den Parameter n definierte Achse von der aktuellen Position zur dazu als absolute Koordinate angegebenen Position (k).
Implementierung	SMCU: nein                      MotionSystem: ja

**!ZA (k){;}**

Name	Move one axis absolute
Befehlskode	CMDT_Z_AXIS_ABSOLUTE
Beschreibung	Bewegt die Z-Achse der MAschine von der aktuellen Position zur dazu als absolute Koordinate angegebenen Position (k).
Implementierung	SMCU: ja                      MotionSystem: nein

**!ZR (k,n){;}**

Name	Move one axis relative
Befehlskode	CMDT_ONE_AXIS_RELATIVE
Beschreibung	Bewegt die durch den Parameter n definierte Achse von der aktuellen Position zur dazu als relative Koordinate angegebenen Position (k).
Implementierung	SMCU: nein                      MotionSystem: ja

## !ZR (k){}

Name Move one axis relative

Befehlskode CMDT\_Z\_AXIS\_RELATIVE

Beschreibung Bewegt die Z-Achse der Maschine von der aktuellen Position zur dazu als relative Koordinate angegebenen Position (k).

Implementierung SMCU: ja                      MotionSystem: nein

## Direkt-Befehle

Die Direkt-Befehle sind Sonderbefehle, die sofort nach ihrer Interpretation unter Umgehung des Befehl -Puffers abgearbeitet werden.

### !CB {;}

Name	Clear Buffer
Befehlskode	CMDT_DIRECT_CLEAR_BUFFER
Beschreibung	Löscht alle Befehle aus dem Befehlspeicher.
Implementierung	SMCU: ja                      MotionSystem: ja

### !GO{;}

Name	Go On
Befehlskode	CMDT_DIRECT_GO_ON
Beschreibung	Hebt das Stop-Kommando bzw. den durch einen speziellen Maschinenstatus verursachten Stop-Zustand auf und setzt die Befehlsabarbeitung mit dem nächsten aktuellen Befehl fort.
Implementierung	SMCU: ja                      MotionSystem: ja

### !RP {;}

Name	Repeat command
Befehlskode	CMDT_DIRECT_REPEAT_COMMAND
Beschreibung	Wiederholt den letzten abgearbeiteten Befehl.
Implementierung	SMCU: in Vorbereitung      MotionSystem: ja

### !ST {;}

Name	Stop
Befehlskode	CMDT_DIRECT_STOPP
Beschreibung	Unterbricht die Befehlsabarbeitung nach Beendigung des laufenden Befehls.
Implementierung	SMCU: ja                      MotionSystem: ja

## Grenzwerte von Eingabeparametern

Parameter	Wert	Maßeinheit
Minimale Schrittweite von Translationsachsen	100	nm
Maximale Schrittweite von Translationsachsen	1000000	nm
Minimale Schrittweite von Rotationsachsen	100	(1/1000)*Winkelsekunde
Maximale Schrittweite von Rotationsachsen	3600000	(1/1000)*Winkelsekunde
Minimale Translationsgeschwindigkeit	10	um/s
Minimale Rotationsgeschwindigkeit	1	0,1°/s
Minimale Beschleunigung für Linearachsen	10	mm/s <sup>2</sup>
Maximale Beschleunigung für Linearachsen	20000	mm/s <sup>2</sup>
Minimale Winkelbeschleunigung	1	mrad/s <sup>2</sup>
Maximale Winkelbeschleunigung	800000	mrad/s <sup>2</sup>

## Statusmeldungen der SMCU

Statusmeldung	Statusursache	Kode
E00<CR>	ESCC-Kanal A nicht bereit	0
E01<CR>	ESCC-Kanal B nicht bereit	1
E02<CR>	Fehler im internen Prozessor-RAM-Block 2	2
E03<CR>	Fehler im internen Prozessor-RAM-Block 1	3
E04<CR>	Fehler im externen Datenspeicher	4
E05<CR>	Fehler im externen Programmspeicher	5
E06<CR>	Einzelne Koordinate	6
E07<CR>	Parameter oder Koordinate wird vermißt	7
E08<CR>	Arbeitsbereich verlassen	8
E09<CR>	Endschalter ausgelöst	9
E0A<CR>	Schnellstop ausgelöst	10
E0B<CR>	Geschwindigkeit zu hoch	11
E0C<CR>	Geschwindigkeit zu klein	12
E0D<CR>	Unbekanntes Kommando	13
E0E<CR>	Empfänger-Paritäts-Fehler	14
E0F<CR>	Empfänger-Überlauf	15
E10<CR>	Empfänger-Framing-Fehler	16
E11<CR>	Beschleunigung zu hoch	17
E12<CR>	Beschleunigung zu klein	18
E13<CR>	Auflösung zu groß	19
E14<CR>	Auflösung zu klein	20
E15<CR>	Achse nicht vorhanden	21

## Statusmeldungen des MotionSystems

Statusmeldung	Statusursache	Kode
E100<CR>	ESCC-Kanal A nicht bereit	0
E101<CR>	ESCC-Kanal B nicht bereit	1
E102<CR>	Fehler im internen Prozessor-RAM-Block 2	2
E103<CR>	Fehler im internen Prozessor-RAM-Block 1	3
E104<CR>	Fehler im externen Datenspeicher	4
E105<CR>	Fehler im externen Programmspeicher	5
E106<CR>	Einzelne Koordinate	6
E107<CR>	Parameter oder Koordinate wird vermißt	7
E108<CR>	Arbeitsbereich verlassen	8
E109<CR>	Endschalter ausgelöst	9
E110<CR>	Schnellstop ausgelöst	10
E111<CR>	Geschwindigkeit zu hoch	11
E112<CR>	Geschwindigkeit zu klein	12
E113<CR>	Unbekanntes Kommando	13
E114<CR>	Empfänger-Paritäts-Fehler	14
E115<CR>	Empfänger-Überlauf	15
E116<CR>	Empfänger-Framing-Fehler	16
E117<CR>	Beschleunigung zu hoch	17
E118<CR>	Beschleunigung zu klein	18
E119<CR>	Auflösung zu groß	19
E120<CR>	Auflösung zu klein	20
E121<CR>	Achse nicht vorhanden	21
E122<CR>	Summen-Errorsignal aufgetreten	22
E123<CR>	Interface nicht vorhanden	23
E124<CR>	I/O-Modul nicht vorhanden	24
E125<CR>	Bewegungsbereich in - X-Richt. verlassen	25
E126<CR>	Bewegungsbereich in + X-Richt. verlassen	26
E127<CR>	Bewegungsbereich in - Y-Richt. verlassen	27
E128<CR>	Bewegungsbereich in + Y-Richt. verlassen	28
E129<CR>	Bewegungsbereich in - Z-Richt. verlassen	29
E130<CR>	Bewegungsbereich in + Z-Richt. verlassen	30
E131<CR>	Bewegungsbereich in - Z1-Richt. verlassen	31
E132<CR>	Bewegungsbereich in + Z1-Richt. verlassen	32
E133<CR>	Bewegungsbereich in - Z2-Richt. verlassen	33
E134<CR>	Bewegungsbereich in + Z2-Richt. verlassen	34
E135<CR>	Bewegungsbereich in - Z3-Richt. verlassen	35
E136<CR>	Bewegungsbereich in + Z3-Richt. verlassen	36
E137<CR>	Endschalter in - X-Richt. angesprochen	37
E138<CR>	Endschalter in + X-Richt. angesprochen	38
E139<CR>	Endschalter in - Y-Richt. angesprochen	39
E140<CR>	Endschalter in + Y-Richt. angesprochen	40
E141<CR>	Endschalter in - Z-Richt. angesprochen	41
E142<CR>	Endschalter in + Z-Richt. angesprochen	42
E143<CR>	Endschalter in - Z1-Richt. angesprochen	43
E144<CR>	Endschalter in + Z1-Richt. angesprochen	44
E145<CR>	Endschalter in - Z2-Richt. angesprochen	45
E146<CR>	Endschalter in + Z2-Richt. angesprochen	46
E147<CR>	Endschalter in - Z3-Richt. angesprochen	47
E148<CR>	Endschalter in + Z3-Richt. angesprochen	48
E149<CR>	Endstufenfehler in der X-Achse	49
E150<CR>	Endstufenfehler in der Y-Achse	50
E151<CR>	Endstufenfehler in der Z-Achse	51
E152<CR>	Endstufenfehler in der Z1-Achse	52
E153<CR>	Endstufenfehler in der Z2-Achse	53
E154<CR>	Endstufenfehler in der Z3-Achse	54

Statusmeldung	Statusursache	Kode
E155<CR>	Endschalterfehler in der X-Achse	55
E156<CR>	Endschalterfehler in der Y-Achse	56
E157<CR>	Endschalterfehler in der Z-Achse	57
E158<CR>	Endschalterfehler in der Z1-Achse	58
E159<CR>	Endschalterfehler in der Z2-Achse	59
E160<CR>	Endschalterfehler in der Z3-Achse	60
E161<CR>	Luftdruck zu gering	61
E162<CR>	Endstufenfehler	62
E163<CR>	Endschalterfehler	63
E164<CR>	Referenzpunkt in der X-Achse angesprochen	64
E165<CR>	Referenzpunkt in der Y-Achse angesprochen	65
E166<CR>	Referenzpunkt in der Z-Achse angesprochen	66
E167<CR>	Referenzpunkt in der Z1-Achse angesprochen	67
E168<CR>	Referenzpunkt in der Z2-Achse angesprochen	68
E169<CR>	Referenzpunkt in der Z3-Achse angesprochen	69
E170<CR>	Referenzpunkt angesprochen	70
E171<CR>	Messageinterrupt 0 ausgelöst	71
E172<CR>	Joystick nicht mit dem Interface verbunden	72
E173<CR>	Timeout-Fehler beim Kommandolesen	73
E174<CR>	Modul COUNT2, keine Achse selektiert	74
E175<CR>	Ein bereits gefundenes Modul liefert keine Identifikation mehr, Hardwarefehler	75
E176<CR>	Falsches Kommando nach einem Stop-Interrupt. Es wird das Direktkommando !GO erwartet.	76
E177<CR>	Falsches Kommando nach einem Limit-Interrupt. Es wird das Direktkommando !GO erwartet.	77
E178<CR>	Falsches Kommando nach einem Softframe-Fehler. Es wird das Direktkommando !GO erwartet.	78
E179<CR>	Falsches Kommando nach einem Message-Interrupt Es wird das Direktkommando !GO erwartet	79
E180<CR>	Messageinterrupt 1 ausgelöst	80
E181<CR>	Falsche Kommando nach Open Interface	81
E182<CR>	Folgevektor enthält zu viele Parameter	82
E183<CR>	Puffer für die Folgevektoren ist voll	83
E184<CR>	Referenzfahrt ist in der angegebenen Achse nicht erlaubt	84
E185<CR>	Kommandopuffer ist voll, Kommandos gehen verloren	85



