

INTERNATIONAL STANDARD

**ISO
4343**

Second edition
2000-10-15

Industrial automation systems — Numerical control of machines — NC processor output — Post processor commands

*Systèmes d'automatisation industrielle — Commande numérique des
machines — Informations de sortie des processeurs CN — Instructions
post-processeur*



Reference number
ISO 4343:2000(E)

AVA Documents
©ISO 2000
Document Sharing Hub

PDF disclaimer

This PDF file may contain embedded typefaces. In accordance with Adobe's licensing policy, this file may be printed or viewed but shall not be edited unless the typefaces which are embedded are licensed to and installed on the computer performing the editing. In downloading this file, parties accept therein the responsibility of not infringing Adobe's licensing policy. The ISO Central Secretariat accepts no liability in this area.

Adobe is a trademark of Adobe Systems Incorporated.

Details of the software products used to create this PDF file can be found in the General Info relative to the file; the PDF-creation parameters were optimized for printing. Every care has been taken to ensure that the file is suitable for use by ISO member bodies. In the unlikely event that a problem relating to it is found, please inform the Central Secretariat at the address given below.

© ISO 2000

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from either ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.ch
Web www.iso.ch

Printed in Switzerland

Contents

Foreword	viii
Introduction	ix
1 Scope	1
2 Normative references	1
3 Co-ordinate system	1
3.1 Part program reference	1
3.2 Machine program reference	1
3.3 Units of measure	2
4 General structure of post processor commands	2
4.1 NC processor	2
4.2 CLDATA	2
4.3 Post processor	3
5 General language	6
5.1 General comments	6
5.2 The ADAPTV command	8
5.3 The AIR command	9
5.4 The APPLY command	10
5.5 The AUXFUN command	11
5.6 The BREAK command	12
5.7 The CALSUB command	13
5.8 The CLAMP command	14
5.9 The CLDATA command	15
5.10 The COUPLE command	16
5.11 The CUTCOM command	17
5.12 The DEFSUB command	18
5.13 The DELAY command	19
5.14 The DISPLY command	20
5.15 The END command	21
5.16 The ENDSUB command	22
5.17 The FEDRAT command	23
5.18 The GOHOME command	26

5.19	The GOPARK command	27
5.20	The HOMEPT command	28
5.21	The INCLUD command	29
5.22	The INSERT command	30
5.23	The LEADER command	31
5.24	The LIMIT command	32
5.25	The LOCATE command	33
5.26	The LPRINT command	34
5.27	The MACHIN command	35
5.28	The MATERL command	36
5.29	The MCHFIN command	37
5.30	The MCHTOL command	38
5.31	The MODE command	39
5.32	The MOVETO command	43
5.33	The OPSKIP command	44
5.34	The OPSTOP command	45
5.35	The ORIGIN command	46
5.36	The PARKPT command	47
5.37	The PARTNO command	48
5.38	The PPFUN command	49
5.39	The PPRINT command	50
5.40	The PPTIME command	51
5.41	The PREFUN command	52
5.42	The RAPID command	53
5.43	The RESET command	54
5.44	The REWIND command	55
5.45	The SAFPOS command	56
5.46	The SEQNO command	57
5.47	The STOP command	58
5.48	The SYNCTR command	59
5.49	The TLLIFE command	61
5.50	The TMARK command	62
5.51	The TRANS command	63
6	Die sinking EDM language	64
6.1	General comments	64
6.2	The APPLY command	65
6.3	The CUTCOM command	66
6.4	The FLUSH command	69
6.5	The GENRTR command	72
6.6	The LOAD command	73
6.7	The OP command	74
6.8	The SELECT command	78
6.9	The SPINDL command	79
6.10	The TOOLNO command	83
6.11	The UNLOAD command	84

7	Flame cutting language	85
7.1	General comments	85
7.2	The APPLY command	86
7.3	The ASSIST command	87
7.4	The CLDIST command	88
7.5	The PIERCE command	89
7.6	The TORCH command	90
8	Grinding language	91
8.1	General comments	91
8.2	The APPLY command	92
8.3	The DRESS command	93
9	Laser-beam machining language	94
9.1	General comments	94
9.2	The APPLY command	95
9.3	The ASSIST command	96
9.4	The CLDIST command	97
9.5	The CYCLE command	98
9.6	The PIERCE command	101
10	Milling and drilling language	102
10.1	General comments	102
10.2	The APPLY command	103
10.3	The ARCSLP command	104
10.4	The CLAMP command	105
10.5	The CLEARP command	106
10.6	The COOLNT command	107
10.7	The CUTCOM command	108
10.8	The CYCLE command	111
10.9	The HEAD command	133
10.10	The INDPOS command	134
10.11	The LINTOL command	135
10.12	The LOAD command	136
10.13	The ORIGIN command	137
10.14	The RETRACT command	140
10.15	The ROTATE command	141
10.16	The SELECT command	142
10.17	The SPINDL command	143
10.18	The TOOLNO command	147
10.19	The UNLOAD command	148
11	Punching and forming language	149
11.1	General comments	149
11.2	The APPLY command	150
11.3	The CLAMP command	151

11.4	The CYCLE command	152
11.5	The LOAD command	163
11.6	The PIERCE command	164
11.7	The SELECT command	165
11.8	The TOOLNO command	166
11.9	The UNLOAD command	168
12	Turning language	169
12.1	General comments	169
12.2	The APPLY command	171
12.3	The BARFED command	172
12.4	The CATCHR command	173
12.5	The CHUCK command	174
12.6	The CLAMP command	176
12.7	The COOLNT command	179
12.8	The COUPLE command	180
12.9	The CUTCOM command	181
12.10	The DEFCON command	182
12.11	The LOAD command	183
12.12	The MODE command	184
12.13	The OP command	185
12.14	The PITCH command	195
12.15	The SAFETY command	196
12.16	The SELECT command	197
12.17	The SPINDL command	198
12.18	The STAN command	202
12.19	The STDYRS command	203
12.20	The TLSTCK command	204
12.21	The TOOLNO command	205
12.22	The TURRET command	206
12.23	The UNLOAD command	207
13	Wire EDM language	208
13.1	General comments	208
13.2	The APPLY command	209
13.3	The CLDIST command	210
13.4	The CUTCOM command	211
13.5	The CYCLE command	212
13.6	The FLUSH command	215
13.7	The GENRTR command	218
13.8	The LOAD command	219
13.9	The ORIGIN command	220
13.10	The SELECT command	221
13.11	The STAN command	222
13.12	The TOOLNO command	223
13.13	The UNLOAD command	224

14	Probing language	225
14.1	General comments	225
14.2	The APPLY command	226
14.3	The LOAD command	227
14.4	The MODE command	228
14.5	The PROBE command	231
14.6	The SELECT command	237
14.7	The TOOLNO command	238
14.8	The UNLOAD command	239
14.9	The VERIFY command	240
15	Drafting and plotting language	250
15.1	General comments	250
15.2	The DRAFT command	251
15.3	The LETTER command	252
15.4	The OV PLOT command	253
15.5	The PENDWN command	254
15.6	The PENUP command	255
15.7	The PPLOT command	256
	Annex A (normative) Rules used in the syntax definitions	258
	Annex B (normative) List of keywords	259

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 3.

Draft International Standards adopted by the technical committees are circulated to the member bodies for voting. Publication as an International Standard requires approval by at least 75% of the member bodies casting a vote.

Attention is drawn to the possibility that some of the elements of this International Standard may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights.

International Standard ISO 4343 was prepared by Technical Committee ISO/TC 184, *Industrial automation systems and integration*, Subcommittee SC 1, *Physical device control*.

This second edition cancels and replaces the first edition (ISO 4343:1978), which has been technically revised.

Annexes A and B form a normative part of this International Standard.

Introduction

The output of a general purpose numerical control processor is information used as input to a post processor. This information is called CLDATA, which was originally derived from “cutter location data.”

CLDATA provides a general language to pass manufacturing information from a numerical control processor to a post processor, where the general language is converted to the specific format required by the particular numerical control equipment. The logical and physical structure of CLDATA records are given in ISO 3592.

This International Standard defines a standard post processor vocabulary, in the context of command word and the parameters that can be associated with a command word. This vocabulary is encoded using the 2 000 class (“integer code type post processor command”) and 20 000 class (“literal type post processor command”) CLDATA records given in ISO 3592.

There is a one-to-one correspondence between the elements of the post processor vocabulary and the elements of the post processor command CLDATA records. The integer code numbers given in annex B of this International Standard are the code numbers that are used to represent keywords in the 2 000 class CLDATA records. The keyword names given in annex B of this International Standard are the names that are used to represent keywords in the 20 000 class CLDATA records.

Numerical control is applied to many types of machines, but the language defined in this International Standard has been developed primarily for numerically controlled machine tools – hence the words “tool” and “part” are used in the description of the language to indicate the working element and processed element respectively. Many of the vocabulary words are also derived from metal working terminology.

Industrial automation systems — Numerical control of machines — NC processor output — Post processor commands

1 Scope

This International Standard defines the elements of a set of post processor statements to be used in numerical control software. These statements are encoded on 2 000 class and 20 000 class CLDATA records or their equivalent.

Each processor using one of the ISO numerical control programming languages shall be capable of producing post processor command type CLDATA records as defined in this International Standard.

Each post processor shall be capable of using the post processor command type CLDATA records defined in this international Standard as input.

This International Standard does not prescribe

- the mechanism by which the statements are processed;
- the medium on which the input language statements are recorded;
- the medium and format of output machine control data;
- the order of statements within a part program.

2 Normative references

The following normative documents contain provisions which, through reference in this text, constitute provisions of this International Standard. For dated references, subsequent amendments to, or revisions of, any of these publications do not apply. However, parties to agreements based on this International Standard are encouraged to investigate the possibility of applying the most recent editions of the normative documents indicated below. For undated references, the latest edition of the normative document referred to applies. Members of ISO and IEC maintain registers of currently valid International Standards.

ISO 841¹⁾, *Industrial automation systems – Numerical control of machines – Coordinate system and motion nomenclature*.

ISO 3592:2000, *Industrial automation systems – Numerical control of machines – NC processor output – File structure and language format*.

ISO 4342:1985, *Numerical control of machines – NC processor input – Basic part program reference language*.

ISO 6983-1:1982, *Numerical control of machines – Program format and definition of address words – Part 1: Data*

format for positioning, line motion and contouring control systems.

3 Co-ordinate system

3.1 Part program reference

ISO 841 is the basis for defining the co-ordinate system of CLDATA.

In the CLDATA, the reference axes of the co-ordinate system are x, y and z. Co-ordinates refer to a reference point on a tool (usually the center of the tip) relative to the part co-ordinate system. CLDATA can define the following location and orientation components:

<i>x</i>	Dimension parallel to X
<i>y</i>	Dimension parallel to Y
<i>z</i>	Dimension parallel to Z
<i>i</i>	X axis component of the tool axis vector
<i>j</i>	Y axis component of the tool axis vector
<i>k</i>	Z axis component of the tool axis vector
<i>l</i>	X axis component of a secondary orientation vector
<i>m</i>	Y axis component of a secondary orientation vector
<i>n</i>	Z axis component of a secondary orientation vector

Unless otherwise specified, post processor command dimensional values refer to the CLDATA co-ordinate system. The following syntax is permitted as replacement for strings of numeric values representing CLDATA co-ordinates, CLDATA tool orientation vectors and CLDATA secondary orientation vectors.

```

XCOORD, x
YCOORD, y
ZCOORD, z
TLVEC, i, j, k
NORMAL, l, m, n
COORD, x, y, z [ , i, j, k [ , l, m, n ] ]

```

3.2 Machine program reference

ISO 841 is the basis for defining the standard configuration of machine axes.

On the machine, the reference axes of the co-ordinate system are x, y and z. Co-ordinates refer to a reference point on the machine (usually the center face of the tool holding mechanism) relative to the machine co-ordinate system. The origin and alignment of the CLDATA and machine reference systems coincide. Provision is made within this International Standard to define an alternate relationship

1) To be published. (Revision of ISO 841:1974)

between CLDATA and machine reference systems. The following machine axes are recognized:

- a angular dimension about the X axis
- b angular dimension about the Y axis
- c angular dimension about the Z axis
- p tertiary dimension parallel to the X axis
- q tertiary dimension parallel to the Y axis
- r tertiary dimension parallel to the Z axis
- u secondary dimension parallel to the X axis
- v secondary dimension parallel to the Y axis
- w secondary dimension parallel to the Z axis
- x primary dimension parallel to the X axis
- y primary dimension parallel to the Y axis
- z primary dimension parallel to the Z axis

The following syntax is permitted as replacement for strings of numeric values representing machine linear axis co-ordinates:

$$\left(\begin{matrix} \text{PAXIS} \\ \text{QAXIS} \\ \text{RAXIS} \\ \text{UAXIS} \\ \text{VAXIS} \\ \text{WAXIS} \\ \text{XAXIS} \\ \text{YAXIS} \\ \text{ZAXIS} \end{matrix} \right), a$$

The following syntax is permitted for numeric values representing machine rotary axis co-ordinates. The initial HEAD or TABLE keyword further qualifies the rotary axis when there are multiple axes providing rotation about a given axis.

$$\left(\left[\begin{matrix} \text{HEAD} \\ \text{TABLE} \end{matrix} \right], \begin{matrix} \text{AAXIS} \\ \text{BAXIS} \\ \text{CAXIS} \\ \\ \text{DAXIS} \\ \text{EAXIS} \end{matrix} \right), a$$

When specifying angles of planes, the positive direction is counterclockwise and the reference axis is as shown in table 1. The positive direction of angle is counterclockwise from the reference axis.

Table 1 - Reference axes

Plane	Reference axis
XY	x
YZ	y
ZX	z

3.3 Units of measure

Angles are expressed in degrees and decimal fractions of a degree.

The linear unit of measure of CLDATA is the millimeter. ISO 3592 has provision to define other units of measure. A change of units is modal and applies to subsequent co-ordinate data until changed again.

Unless otherwise specified, dimensional values appearing in post processor commands are in the same reference units as the CLDATA. Provision is made within this International Standard to define explicit units of measure for simple parameters on post processor commands. Table 2 lists the

preferred non dimensional keyword followed by the non preferred dimensional alternates. Only preferred keywords appear in syntax definitions. Non preferred keywords can be substituted, however, their use is not encouraged.

Table 2 – Dimensional keywords

Preferred keyword	
Non-preferred alternate	
PERMIN	CLDATA units per minute
IPM	inches per minute
MMPM	millimeters per minute
PERREV	CLDATA units per revolution of the spindle
IPR	inches per revolution of the spindle
MMPR	millimeters per revolution of the spindle
TPI	the reciprocal of IPR
MXPERM	maximum CLDATA units per minute
MAXIPM	maximum inches per minute
MXMMPM	maximum millimeters per minute
CSS	surface speed in CLDATA units per minute
SFM	surface speed in feet per minute
SMM	surface speed in meters per minute

Unless otherwise specified and providing that the capability exists, dimensional values appearing in the post processor generated machine program shall be in the same reference units as the CLDATA. Where this is impossible due to limitations of the machine, the post processor shall be responsible for converting the CLDATA units to the appropriate units supported by the machine. Provision is made within this International Standard to define explicit output units to be used for the entire machine program irrespective of the CLDATA units.

4 General structure of post processor commands

4.1 NC processor

ISO 4342 defines the syntax and semantics of a standard NC processor input language, including a limited set of post processor vocabulary.

Post processor command names are defined using a Major type post processor keyword. A comma separated parameter list, if present, is separated from the command name using a “/” delimiter.

Parameters can consist of any combination of numeric values, keywords and quote delimited strings, respecting the syntax of the particular post processor command.

4.2 CLDATA

ISO 3592 defines the CLDATA record format for post processor commands. Specifics concerning the command names and command parameters are defined within that International Standard.

2 000 class records carry post processor instructions and are formed of elements as follows:

- a) Element 1 (integer) is the CLDATA record sequence number.
- b) Element 2 (integer) is 2 000, which identifies the record as an integer code type post processor command.
- c) Element 3 (integer) is the integer code of the Major word identifying the post processor command.
- d) Element 4 onwards (various) is an optional list of post processor command parameters.

Keywords are represented in element 3 and onwards as integer codes.

20 000 class records also carry post processor instructions and are formed of elements as follows:

- a) Element 1 (integer) is the CLDATA record sequence number.
- b) Element 2 (integer) is 20 000, which identifies the record as a literal type post processor command.
- c) Element 3 (integer) is a code indicating whether the post processor command spans multiple records.

- d) Element 4 (keyword) is the text of the Major word identifying the post processor command.
- e) Element 5 onwards (various) is an optional list of post processor command parameters.

Keywords are represented in element 4 and onwards as character text.

4.3 Post processor

Each post processor shall support the basic set of commands defined in the general language section (see 5). Each post processor shall also support the additional set of commands defined in one or more machine family sections which follow. Each machine family section defines only those commands which are applicable to the machine type. The general language section and the applicable machine family language sections together provide the whole language for a machine family.

When a machine supports capabilities of multiple machine families defined within this International Standard, the APPLY command (see 5.4) shall be used to designate the machine family being manipulated at any given moment.

Table 3 lists a cross reference of commands and language sections.

Table 3 - Cross reference of commands and language sections

Command	General	Die EDM	Flame	Grinder	Laser	Milling	Punch	Turning	Wire EDM	Probe	Plotting
ADAPTV	5.2										
AIR	5.3										
APPLY	5.4	6.2	7.2	8.2	9.2	10.2	11.2	12.2	13.2	14.2	
ARCSLP						10.3					
ASSIST			7.3		9.3						
AUXFUN	5.5										
BARFED								12.3			
BREAK	5.6										
CALSUB	5.7										
CATCHR								12.4			
CHUCK								12.5			
CLAMP	5.8					10.4	11.3	12.6			
CLDATA	5.9										
CLDIST			7.4		9.4				13.3		
CLEARP						10.5					
COOLNT						10.6		12.7			
COUPLE	5.10							12.8			
CUTCOM	5.11	6.3				10.7		12.9	13.4		
CYCLE					9.5	10.8	11.4		13.5		
DEFCON								12.10			
DEFSUB	5.12										
DELAY	5.13										
DISPLY	5.14										
DRAFT											15.2
DRESS				8.3							
END	5.15										
ENDSUB	5.16										
FEDRAT	5.17										
FLUSH		6.4							13.6		
GENRTR		6.5							13.7		
GOHOME	5.18										
GOPARK	5.19										
HEAD						10.9					
HOMEPT	5.20										
INCLUD	5.21										
INDPOS						10.10					
INSERT	5.22										
LEADER	5.23										
LETTER											15.3
LIMIT	5.24										
LINTOL						10.11					
LOAD		6.6				10.12	11.5	12.11	13.8	14.3	
LOCATE	5.25										
LPRINT	5.26										
MACHIN	5.27										

Table 3 - Cross reference of commands and language sections (concluded)

Command	General	Die EDM	Flame	Grinder	Laser	Milling	Punch	Turning	Wire EDM	Probe	Plotting
MATERL	5.28										
MCHFIN	5.29										
MCHTOL	5.30										
MODE	5.31							12.12		14.4	
MOVETO	5.32										
OP		6.7						12.13			
OPSKIP	5.33										
OPSTOP	5.34										
ORIGIN	5.35					10.13			13.9		
OVPLOT											15.4
PARKPT	5.36										
PARTNO	5.37										
PENDWN											15.5
PENUP											15.6
PIERCE			7.5		9.6		11.6				
PITCH								12.14			
PPFUN	5.38										
PPLOT											15.7
PPRINT	5.39										
PPTIME	5.40										
PREFUN	5.41										
PROBE										14.5	
RAPID	5.42										
RESET	5.43										
RETRCT						10.14					
REWIND	5.44										
ROTATE						10.15					
SAFETY								12.15			
SAFPOS	5.45										
SELECT		6.8				10.16	11.7	12.16	13.10	14.6	
SEQNO	5.46										
SPINDL		6.9				10.17		12.17			
STAN								12.18	13.11		
STDYRS								12.19			
STOP	5.47										
SYNCTR	5.48										
TLLIFE	5.49										
TLSTCK								12.20			
TMARK	5.50										
TOOLNO		6.10				10.18	11.8	12.21	13.12	14.7	
TORCH			7.6								
TRANS	5.51										
TURRET								12.22			
UNLOAD		6.11				10.19	11.9	12.23	13.13	14.8	
VERIFY										14.9	

5 General language

5.1 General comments

5.1.1 General semantics

The general language section defines the language that is common to a number of machine families. The general language section and a machine family section together provide the whole language for the machine family. A post processor for the family shall provide recognition and resolution of the entire language set.

5.1.2 Sub-contents

For

- 1) the ADAPTV command, which controls the use of an adaptive controller, see 5.2;
- 2) the AIR command, which controls the supply of air, see 5.3;
- 3) the APPLY command, which defines the selection of machine family, see 5.4;
- 4) the AUXFUN command, which provides for the insertion of miscellaneous function (M) codes, see 5.5;
- 5) the BREAK command, which provides for segmentation of a machine program, see 5.6;
- 6) the CALSUB command, which activates a predefined or machine specific sub program, see 5.7;
- 7) the CLAMP command, which controls the application of axis clamps, see 5.8;
- 8) the CLDATA command, which controls the input of part program data, see 5.9;
- 9) the COUPLE command, which defines the association between part program axes and machine axes, see 5.10;
- 10) the CUTCOM command, which controls workpiece compensation, see 5.11;
- 11) the DEFSUB command, which indicates the start of a sub program definition, see 5.12;
- 12) the DELAY command, which temporarily halts the movement of axes on the machine, see 5.13;
- 13) the DISPLY command, which controls the display of operator messages at the machine console, see 5.14;
- 14) the END command, which designates the end of the machine program, see 5.15;
- 15) the ENDSUB command, which indicates the end of a sub program definition, see 5.16;
- 16) the FEDRAT command, which controls various functions related to feed velocity, see 5.17;
- 17) the GOHOME command, which moves the machine to the home position, see 5.18;
- 18) the GOPARK command, which moves the machine to the secondary home position, see 5.19;
- 19) the HOMEPT command, which defines the machine home position, see 5.20;
- 20) the INCLUD command, which specifies a source for additional part program data, see 5.21;
- 21) the INSERT command, which provides for the insertion of machine program data, see 5.22;
- 22) the LEADER command, which provides for the insertion of leader spacing, see 5.23;
- 23) the LIMIT command, which defines travel limits and guard zones, see 5.24;
- 24) the LOCATE command, which preloads machine reference system axes, see 5.25;
- 25) the LPRINT command, which provides for control of the post processor listing, see 5.26;
- 26) the MACHIN command, which defines the post processor, see 5.27;
- 27) the MATERL command, which defines the workpiece material, see 5.28;
- 28) the MCHFIN command, which defines the acceptable machining finish, see 5.29;
- 29) the MCHTOL command, which defines the acceptable machining tolerance, see 5.30;
- 30) the MODE command, which controls various modal settings, see 5.31;
- 31) the MOVETO command, which controls the movement of machine axes, see 5.32;
- 32) the OPSKIP command, which controls the insertion of block delete codes, see 5.33;
- 33) the OPSTOP command, which designates an optional stopping point in the machine program, see 5.34;
- 34) the ORIGIN command, which defines the relationship between part and machine reference systems, see 5.35;
- 35) the PARKPT command, which defines the secondary machine home position, see 5.36;
- 36) the PARTNO command, which provides a unique identification for the machine program, see 5.37;

- 37) the PPFUN command, which provides special commands or instructions for the post processor, see 5.38;
- 38) the PPRINT command, which provides messages in the output data, see 5.39;
- 39) the PPTIME command, which provides for modification of post processor calculated run time, see 5.40;
- 40) the PREFUN command, which provides for the insertion of preparatory function (G) codes, see 5.41;
- 41) the RAPID command, which specifies motion at rapid traverse velocity, see 5.42;
- 42) the RESET command, which designates a restart point in the machine program, see 5.43;
- 43) the REWIND command, which rewinds the control tape, see 5.44;
- 44) the SAFPOS command, which defines the working element change position, see 5.45;
- 45) the SEQNO command, which controls the numbering of machine program blocks, see 5.46;
- 46) the STOP command, which designates a required stopping point in the machine program, see 5.47;
- 47) the SYNCTR command, which controls the synchronization of multiple machine heads, see 5.48;
- 48) the TLLIFE command, which controls the use of tool life monitoring equipment, see 5.49;
- 49) the TMARK command, which provides for the insertion of rewind stop codes, see 5.50;
- 50) the TRANS command, which translates the part program co-ordinates, see 5.51.

5.1.3 Limitations

None.

5.2 The ADAPTV command

Controls the use of an adaptive controller.

ADAPTV / $\left(\begin{array}{c} \text{ON} \\ \text{OFF} \end{array} \right)$

5.2.1 Semantics

This command is used to engage and disengage the adaptive control unit, which is a control system that adjusts the response from conditions detected during the work.

ON (keyword) turns the adaptive control unit on and gives it control from that point in the machine program until it is turned off.

OFF (keyword) turns the adaptive control unit off and returns control to the machine program.

5.2.2 Examples

None.

5.2.3 Limitations

None.

5.3 The AIR command

Controls the supply of air.

**AIR / (ON
THRU) [, a]**

AIR / OFF

5.3.1 Semantics

This command controls the supply of air. The air device is operated in a modal fashion if a duration is not specified.

ON (keyword) specifies a general air device.

THRU (keyword) specifies an alternate air device.

a (real) indicates the duration in seconds that the air device is to be active. The duration shall be a non-zero positive value. If the duration is omitted, the air device shall operate continuously.

OFF (keyword) terminates the continuous operation of the air device.

5.3.2 Examples

The following command activates the air device for 6 seconds.

AIR/ON, 6

The following commands activate the air device during a tool change.

AIR/THRU
LOAD/TOOL, 1
AIR/OFF

5.3.3 Limitations

None.

5.4 The APPLY command

Defines the selection of machine family.

```
APPLY / ( DEDM  
          FLAME  
          GRIND  
          LASER  
          MILL  
          PROBE  
          PUNCH  
          TURN  
          WEDM )
```

5.4.1 Semantics

This command specifies which machine family to control when the application programming changes to another basic machine family within the same post processor. Subsequent part program data is interpreted using the language defined for the named machine family.

DEDM (keyword) specifies die sinking electrical discharge machining (EDM).

FLAME (keyword) specifies flame, plasma or water jet machining.

GRIND (keyword) specifies grinding.

LASER (keyword) specifies laser machining.

MILL (keyword) specifies milling or drilling.

PROBE (keyword) specifies probing or measuring.

PUNCH (keyword) specifies sheet metal forming.

TURN (keyword) specifies turning.

WEDM (keyword) specifies wire electrical discharge machining (EDM).

5.4.2 Examples

None.

5.4.3 Limitations

None.

5.5 The AUXFUN command

Provides for the insertion of miscellaneous function (M) codes.

AUXFUN / $a^{0:n}$ [, a] [, **NOW**]

5.5.1 Semantics

This command provides the facility to insert miscellaneous function (M) code numbers in the machine program.

a (real) specifies the miscellaneous function (M) code number to be output. If multiple codes are specified, they shall be inserted into the machine program as though each was specified in a separate AUXFUN command.

NOW (keyword) indicates that a break to a new block shall appear immediately following the output of each (M) code. The default action is to allow the (M) code to be combined with subsequent machine program data if possible.

5.5.2 Examples

The following command will output two (M) codes in the machine program, with each (M) code appearing in a separate block.

AUXFUN/5, 4, NOW

The following commands perform the identical function as the single command described above.

AUXFUN/5, NOW
AUXFUN/4, NOW

5.5.3 Limitations

This command produces machine program data which may not be portable.

5.6 The BREAK command

Provides for segmentation of a machine program.

$$\text{BREAK} \left[/ \left(\begin{array}{l} \text{LENGTH}, a \\ \text{TIME}, b \\ \text{NOW} \end{array} \right) \right]$$

5.6.1 Semantics

This command is used to segment a machine program. Segmentation can be immediate, or deferred based on machine program size or elapsed machine run time.

LENGTH, *a* (keyword,real) specifies the maximum machine program length in part program units before each new machine program segmentation.

TIME, *b* (keyword,real) specifies the machine run time duration in seconds before each new machine program segmentation.

NOW (keyword) specifies that machine program segmentation shall occur immediately.

A BREAK command without parameters shall be treated as a request for immediate segmentation.

5.6.2 Examples

None.

5.6.3 Limitations

None.

5.7 The CALSUB command

Activates a predefined or machine specific sub program.

CALSUB / $\left(\begin{smallmatrix} a \\ 'b' \end{smallmatrix} \right) \left(\begin{smallmatrix} \text{CLDATA} \\ \text{SYSTEM} \end{smallmatrix} \right) [, \text{ON}] [, \text{TIMES}, c] [, 'd']$

CALSUB / OFF

5.7.1 Semantics

This command controls the activation of a predefined or machine specific sub program.

a (real) identifies the sub program by a numeric identifier.

b (text) identifies the sub program by a textual identifier.

CLDATA (keyword) specifies that the sub program being called has been previously defined by a DEFSUB command (see 5.12).

SYSTEM (keyword) specifies that the sub program being called is a machine specific sub program which has been preloaded on the machine. Machine specific sub programs are not defined in the part program.

ON (keyword) specifies that the sub program shall be called once for each motion following in the part program until canceled by a CALSUB/OFF command. In the absence of this qualifier, the sub program is called immediately.

OFF (keyword) specifies that modal calling of the sub program shall be terminated.

TIMES,c (keyword,real) specifies the number of times to call the sub program. For modal operation, the sub program is called the specified number of times at each motion point in the part program. In the absence of this qualifier the sub program is called once.

d (text) specifies optional machine specific data to be appended to the end of each call sub program block. This provides a rudimentary mechanism for passing parameters to a sub program. The text is treated in the same manner as an INSERT command (see 5.22).

Predefined sub programs are those that have been defined earlier within the part program. A predefined sub program consists of a DEFSUB command naming the sub program, one or more part program instructions defining the body of the sub program, and a terminating ENDSUB (see 5.16) command.

The CALSUB command causes a copy of the part program data within the named sub program to be processed. On machines which do not support sub programs, the post processor shall substitute the part program data within the sub program definition at the point in the machine program where the CALSUB occurs.

The modal CALSUB operation causes the named sub program to be called at each motion point in the part program until canceled by a CALSUB/OFF command. When possible, the post processor shall combine the motion and the sub program call into a single block. When this is not possible, the motion shall be processed first, followed by the sub program call. This provides the capability to perform cyclic operations at a series of points.

5.7.2 Examples

The following example calls a sub program once for each point in a linear pattern.

```
CALSUB/9010,CLDATA,ON
GOTO/4,1,0
GOTO/4,2,0
GOTO/4,3,0
CALSUB/OFF
```

The following example calls a sub program, passing machine specific parameters.

```
CALSUB/8100,SYSTEM,'P01=4.5 P02=2'
```

5.7.3 Limitations

Machine specific parameters shall not be used when the machine does not support sub programs.

The machine specific formats of this command produce machine program data which may not be portable.

5.8 The CLAMP command

Controls the application of axis clamps.

$$\text{CLAMP} / \left(\begin{array}{c} \{mc-axis\} \\ \text{ALL} \end{array} \right) \left(\begin{array}{c} \text{ON} \\ \text{OFF} \\ \text{AUTO} \end{array} \right)$$

5.8.1 Semantics

This command controls a clamping operation.

{mc-axis} (keyword) specifies a machine axis by name (see 3.2 for a list of machine axis names and associated syntax).

ALL (keyword) specifies all machine axes.

ON (keyword) initiates a clamping operation.

OFF (keyword) terminates a clamping operation.

AUTO (keyword) specifies that the axis shall remain clamped by default. When motion is programmed for the axis, the clamps will be automatically removed before the motion commences and reapplied after the motion completes.

5.8.2 Examples

The following example clamps the rotary A axis table during each drilling motion in a cycle.

```
CYCLE/DRILL,DEPTH,4,PERREV,.012,CLEAR,.1,RETURN,.5
CLAMP/TABLE,AAXIS,AUTO
GOTO/0,0,8,0,0,1
GOTO/0,8,0,0,1,0
GOTO/0,0,-8,0,0,-1
GOTO/0,-8,0,0,-1,0
CLAMP/TABLE,AAXIS,OFF
CYCLE/OFF
```

5.8.3 Limitations

None.

5.9 The CLDATA command

Controls the input of part program data.

CLDATA / $\left(\begin{array}{c} \text{ON} \\ \text{OFF} \end{array} \right)$

5.9.1 Semantics

This command allows the part programmer to selectively control the part program data input to the manufacturing post processor. This command does not apply to a plotting post processor.

ON (keyword) specifies that all part program data following the command shall be processed by the manufacturing post processor.

OFF (keyword) specifies that all part program data following the command shall be ignored by the manufacturing post processor, with the exception of the CLDATA command.

5.9.2 Examples

None.

5.9.3 Limitations

None.

5.10 The COUPLE command

Defines the association between part program axes and machine axes.

$$\text{COUPLE} / \left(\begin{matrix} \text{XC00RD} \\ \text{YC00RD} \\ \text{ZC00RD} \end{matrix} \right), \{mc\text{-axis}\} \quad \theta:\pi \left[\begin{matrix} \left(\begin{matrix} \text{XC00RD} \\ \text{YC00RD} \\ \text{ZC00RD} \end{matrix} \right), \{mc\text{-axis}\} \end{matrix} \right]$$

5.10.1 Semantics

This command assigns an axis of motion from the part reference system to a specific axis on the machine. It is used to control machines which have one or more co-linear or co-rotary axes. The assignment is modal until changed.

[XYZ]COORD (keyword) specifies one of the three primary co-ordinate axes of the part reference system to be associated with the specified machine axis.

{mc-axis} (keyword) specifies a co-linear or co-rotary axis which will be used for subsequent motions (see 3.2 for a list of machine axis names and associated syntax). With co-linear axes, movement along the specified part reference axis will be resolved by moving the named co-linear axis. With co-rotary axes, rotation about the specified part reference axis will be resolved by rotating the named co-rotary axis.

5.10.2 Examples

The following command specifies that X motion in the part reference system is to be coupled to the Z machine axis, that Y motion in the part reference system is to be coupled to the X machine axis, and that Z motion in the part reference system is to be coupled to the Y machine axis. This is a common configuration for horizontal lathe programming.

```
COUPLE/XC00RD, ZAXIS, YC00RD, XAXIS, ZC00RD, YAXIS
```

The following command specifies that Y motion in the part reference system is to be coupled to the Z machine axis and that Z motion in the part reference system is to be coupled to the Y machine axis. This is a common configuration for vertical lathe programming.

```
COUPLE/YC00RD, ZAXIS, ZC00RD, YAXIS
```

The following command specifies that Z motion of the part reference system is to be coupled to the W machine axis (secondary Z axis).

```
COUPLE/ZC00RD, WAXIS
```

The following command resets the Z motion of the part reference system to the Z machine axis.

```
COUPLE/ZC00RD, ZAXIS
```

5.10.3 Limitations

Part/machine axis coupling shall be limited to the set of configurations defined by ISO 841.

5.11 The CUTCOM command

Controls workpiece compensation.

$$\text{CUTCOM / ON} \quad 1:n \left(\begin{pmatrix} \text{XCOORD} \\ \text{YCOORD} \\ \text{ZCOORD} \end{pmatrix}, a \right)$$

$$\text{CUTCOM / OFF} \quad 1:n \left(\begin{pmatrix} \text{XCOORD} \\ \text{YCOORD} \\ \text{ZCOORD} \end{pmatrix} \right)$$

CUTCOM / ON, COORD, a

CUTCOM / OFF, COORD

5.11.1 Semantics

This command controls the application and removal of workpiece compensation.

ON (keyword) specifies that workpiece compensation shall be applied.

OFF (keyword) removes workpiece compensation.

[XYZ]COORD (keyword) identifies a machine axis for which compensation is to be applied or removed.

COORD (keyword) identifies a general workpiece compensation to be applied to or removed from all axes.

a (real) identifies the offset register on the machine to use for workpiece compensation.

5.11.2 Examples

None.

5.11.3 Limitations

None.

NOTE

The XCOORD, YCOORD and ZCOORD keywords refer to machine axes, contrary to the general guidelines described in section 3.2. This axis specific form of workpiece compensation has been retained in this revision of the International Standard for the purpose of continuity.

5.12 The DEFSUB command

Indicates the start of a sub program definition.

DEFSUB / $\left(\begin{array}{c} a \\ 'b' \end{array} \right)$

5.12.1 Semantics

This command indicates the start of a predefined sub program.

a (real) identifies the sub program by a numeric identifier.

b (text) identifies the sub program by a textual identifier.

A predefined sub program consists of a DEFSUB command naming the sub program, one or more part program instructions defining the body of the sub program, and a terminating ENDSUB (see 5.16) command. A sub program is called by coding a CALSUB command (see 5.7) with a matching sub program identifier.

The sub program shall be stored by the post processor for subsequent processing by a CALSUB command. On machines which support sub programs, the storage of the sub program shall occur in the machine program in the appropriate format. On machines which do not support sub programs, the post processor shall defer the processing of the sub program until a CALSUB command is encountered.

5.12.2 Examples

Special consideration must be given for sub programs containing cyclic operations to be performed relative to the machine position when the sub program is activated. Sub programs used for this purpose should start and end as follows:

```
DEFSUB/9010
FROM/0,0,0
MODE/INCR
...
...
MODE/ABSOL
ENDSUB
```

The FROM statement defines a starting position for the tool corresponding to the tool position when the sub program is called. The co-ordinate used in the FROM command need not be (0,0,0). The MODE/INCR command specifies that subsequent tool positions shall be output as relative motions. The MODE/ABSOL command resets positioning to absolute mode before returning from the sub program. Absolute positioning eliminates the potential for position skew which would occur in incremental mode if the tool position at the end of the sub program is different from the starting position.

5.12.3 Limitations

None.

5.13 The DELAY command

Temporarily halts the movement of axes on the machine.

DELAY / $\left[\begin{array}{l} \text{DWELL,} \\ \text{REV} \end{array} \right] a$

5.13.1 Semantics

This command halts movement of machine axes for a specified duration, corresponding to the G04 definition in ISO 6983.

DWELL (keyword) specifies that the delay will be timed in seconds.

REV (keyword) specifies that the delay will be timed in revolutions of the spindle.

a (real) indicates the delay duration in the specified units. The duration shall be a non-zero positive value. If neither **DWELL** or **REV** are specified, then the duration defaults to seconds.

5.13.2 Examples

None.

5.13.3 Limitations

A delay duration in spindle revolutions is only valid for machines with spindles, and then only when the spindle is rotating.

5.14 The DISPLY command

Controls the display of operator messages at the machine console.

$$\text{DISPLY} / \left(\begin{array}{c} a \\ 'b' \\ \text{ON} \\ \text{OFF} \end{array} \right)$$

5.14.1 Semantics

This command controls the output of operator messages at the machine console. Messages can be specified either with the DISPLY command or with subsequent PPRINT commands (see 5.39).

a (real) specifies a console light or display pattern.

b (text) specifies a message to display at the console.

ON (keyword) specifies that subsequent PPRINT commands are to be displayed at the machine console.

OFF (keyword) specifies that subsequent PPRINT commands will no longer be displayed at the machine console.

5.14.2 Examples

The following example outputs a series of messages to the machine console.

```
DISPLY/'Message 1'  
DISPLY/'Message 2'  
DISPLY/'Message 3'
```

This next example performs the same operation.

```
DISPLY/ON  
PPRINT/'Message 1'  
PPRINT/'Message 2'  
PPRINT/'Message 3'  
DISPLY/OFF
```

5.14.3 Limitations

None.

5.15 The END command

Designates the end of the machine program.

END

5.15.1 Semantics

This command designates the end of the machine program, corresponding to the M02 definition in ISO 6983.

5.15.2 Examples

None.

5.15.3 Limitations

None.

5.16 The ENDSUB command

Indicates the end of a sub program definition.

ENDSUB

5.16.1 Semantics

This command indicates the end of a predefined sub program.

A predefined sub program consists of a DEFSUB command (see 5.12) naming the sub program, one or more part program instructions defining the body of the sub program, and a terminating ENDSUB command. A sub program is called by coding a CALSUB command (see 5.7) with a matching sub program identifier.

5.16.2 Examples

None.

5.16.3 Limitations

None.

5.17 The FEDRAT command

5.17.1 General comments

Controls one of feed movement velocity, or feed rate override inhibit.

5.17.1.1 Sub-contents

For

- 1) feed movement velocity specification, see 5.17.2;
- 2) feed rate override inhibit specification, see 5.17.3.

5.17.1.2 Limitations

None.

5.17.2 Feed movement velocity specification

FEDRAT / $\left[\begin{array}{c} \text{PERMIN,} \\ \text{PERREV} \\ \text{FPT} \end{array} \right] a \left[, \text{MXPERM}, b \right]$

5.17.2.1 Semantics

This command specifies the feed movement velocity and optionally establishes an upper limit to the velocity.

PERMIN (keyword) specifies that the velocity is measured in part program units per minute, corresponding to the G94 definition in ISO 6983.

PERREV (keyword) specifies that the velocity is measured in part program units per revolution of the spindle, corresponding to the G95 definition in ISO 6983.

FPT (keyword) specifies that the velocity is measured in part program units per tooth per revolution of the spindle. The number of teeth is defined using the FLUTES qualifier of the TOOLNO command. A single tooth (flute) shall be assumed if FLUTES have not been specified.

a (real) specifies the feed movement velocity in the specified or default units. The feed velocity shall be a non-zero positive value. The feed velocity units keyword remains in effect on subsequent FEDRAT commands until explicitly changed. The default feed velocity units at the start of a program is part program units per minute (PERMIN).

MXPERM,b (keyword,real) specifies an upper limit to the feed velocity measured in part program units per minute. The feed velocity limit shall be a non-zero positive value. This limit remains in effect until the next feed velocity specification command. If an upper limit is omitted, then feed limitation shall not occur.

5.17.2.2 Examples

The following example specifies a feed velocity of 0,2 part program units per revolution of the spindle, with an upper bound of 56,0 part program units per minute.

FEDRAT/PERREV, .2, MXPERM, 56

5.17.2.3 Limitations

None.

5.17.3 Feed rate override inhibit specification

**FEDRAT / LOCK, (ON
OFF)**

5.17.3.1 Semantics

This command controls the availability of a feed rate override capability at the machine.

LOCK,ON (keywords) specifies that a lock be placed on the programmed feed velocity, inhibiting any machine based override capability.

LOCK,OFF (keywords) removes the lock on the programmed feed velocity, allowing a machine based override capability.

5.17.3.2 Examples

None.

5.17.3.3 Limitations

None.

5.18 The GOHOME command

Moves the machine to the home position.

```
GOHOME [ / {mc-axis} 0:n[, {mc-axis} ] ]
```

5.18.1 Semantics

This command moves the machine at a rapid traverse rate to the home position of the machine, or if applicable, to the home position defined by the HOMEPT command (see 5.20).

{*mc-axis*} (keyword) specifies one or more machine axes to move to the home point (see 3.2 for a list of machine axis names and associated syntax). The order of the axes is not important as all named axes will be moved simultaneously. If GOHOME is specified without any machine axis names, all axes having a home position will be moved to the home point if required.

5.18.2 Examples

The following commands will first move the machine Z axis and machine W axis to their home positions, then move all remaining axes to the home position.

```
GOHOME/ZAXIS,WAXIS  
GOHOME
```

5.18.3 Limitations

None.

5.19 The GOPARK command

Moves the machine to the secondary home position.

GOPARK [/ {*mc-axis*} ^{0:n}[, {*mc-axis*}]]

5.19.1 Semantics

This command moves the machine at a rapid traverse rate to the secondary home position of the machine, or if applicable, to the home position defined by the PARKPT command (see 5.36). If a secondary home is not defined for the machine, the primary home point will be used (see 5.20).

{*mc-axis*} (keyword) specifies one or more machine axes to move to the secondary home point (see 3.2 for a list of machine axis names and associated syntax). If GOPARK is specified without any machine axis names, all axes having a secondary home position will be moved.

5.19.2 Examples

None.

5.19.3 Limitations

None.

5.20 The HOMEPT command

Defines the machine home position.

$$\text{HOMEPT} / \left(\begin{array}{c} \{mc\text{-axis}\} \\ \{cl\text{-axis}\} \end{array} \right), a \text{ } ^{0:n} \left[, \left(\begin{array}{c} \{mc\text{-axis}\} \\ \{cl\text{-axis}\} \end{array} \right), a \right]$$

HOMEPT / NOMORE

5.20.1 Semantics

This command defines the location of the machine home position used by the GOHOME command (see 5.18). The HOMEPT command is ignored if the machine has a fixed home position, is used for information purposes only if the machine has a reference home position, and is used to define a home position in all other cases. The default home position is the co-ordinates defined by the first FROM motion in the part program.

{mc-axis} (keyword) specifies that a home position is being defined for a machine axis (see 3.2 for a list of machine axis names and associated syntax).

{cl-axis} (keyword) specifies that a home position is being defined for a part reference axis (see 3.1 for a list of part program axis names and associated syntax).

a (real) defines a home position for either the named machine axis or for the named part co-ordinate axis.

NOMORE (keyword) cancels the last specified HOMEPT position and resets the machine home position to the co-ordinates defined by the first FROM motion in the part program.

Only those axes mentioned in a HOMEPT command are moved by a GOHOME command. All unreferenced axes remain unchanged.

5.20.2 Examples

None.

5.20.3 Limitations

None.

5.21 The INCLUDE command

Specifies a source for additional part program data.

INCLUDE / 'a'

5.21.1 Semantics

This command provides for including secondary part program data into the post processor input stream. This can be useful for loading standard information into the post processor, such as TOOLNO or MATERL specifications.

a (text) specifies the name of the file to be read as a secondary part program before continuing with the primary part program.

The INCLUDE command does not cause a permanent transfer of input to occur. When an end-of-file or FINI record is encountered in the secondary part program, processing shall continue with the instruction following the INCLUDE record in the primary part program.

Secondary files may contain INCLUDE commands.

5.21.2 Examples

None.

5.21.3 Limitations

An INCLUDE command shall not recursively reference a part program file.

5.22 The INSERT command

Provides for the insertion of machine program data.

INSERT / 'a' [,NEXT]

5.22.1 Semantics

This command provides the facility to insert data into the machine program.

a (text) specifies data to insert into the machine program.

NEXT (keyword) specifies that the data shall be appended to the next block. In the absence of this qualifier, the data shall be output immediately.

The post processor will provide sequence number information and end of block character data where appropriate.

5.22.2 Examples

None.

5.22.3 Limitations

This command produces machine program data which may not be portable.

5.23 The LEADER command

Provides for the insertion of leader spacing.

LEADER / [LENGTH,] a

5.23.1 Semantics

This command specifies the amount of leader spacing to insert into the control tape.

LENGTH,a (keyword,real) specifies the length of the leader in part reference units. The LENGTH keyword is optional and has no effect on the actions of the LEADER command.

5.23.2 Examples

None.

5.23.3 Limitations

None.

5.24 The LIMIT command

Defines travel limits and guard zones.

LIMIT / *a*, $\left(\begin{array}{c} \text{IN} \\ \text{OUT} \end{array} \right)^{1:n} (, \{mc\text{-axis}\}, b, c)$

LIMIT / *a*, $\left(\begin{array}{c} \text{ON} \\ \text{OFF} \end{array} \right)$

5.24.1 Semantics

This command defines a region of space that the working element (tool) must remain either entirely in (travel limit) or out (guard zone).

a (real) provides an integer identification of the travel limit zone that will be used by the post processor when reporting violations.

IN (keyword) specifies that the tool must remain within the zone being defined.

OUT (keyword) specifies that the tool must remain outside of the zone being defined.

{mc-axis} (keyword) specifies that travel limit checking is required for the named machine axis (see 3.2 for a list of machine axis names and associated syntax).

b,c (reals) specify the lower and upper travel limits for the named machine axis. Travel limits are measured in machine units for linear axes and degrees for rotary axes.

ON (keyword) enables limit checking for the specified travel zone. By default limit checking is enabled when a zone is defined.

OFF (keyword) disables limit checking for the specified travel zone.

5.24.2 Examples

The following example establishes a travel limit, identified as zone 1, which limits machine travel in the X axis to the range 0 through 20 and in the Y axis to the range -5 through 5.

LIMIT/1, IN, XAXIS, 0, 20, YAXIS, -5, 5

This next example establishes a guard zone, identified as zone 2, which restricts X axis travel in the range 15 through 30. The combined effect of zone 1 and zone 2 travel limits checking will limit the X axis to the range 0 through 15.

LIMIT/2, OUT, XAXIS, 15, 30

This last example removes the travel restriction imposed by zone 2 in the previous example.

LIMIT/2, OFF

5.24.3 Limitations

None.

5.25 The LOCATE command

Preloads machine reference system axes.

LOCATE / {*mc-axis*}, *a* ^{0:n}[, {*mc-axis*}, *a*]

LOCATE / OFF

5.25.1 Semantics

This command will preload the machine reference system axes to the specified values. A preload operation does not cause a motion to occur. Only those axes referenced in the LOCATE command will be preloaded, all other axes retain their current settings.

{*mc-axis*} (keyword) specifies that a preload operation is required for the named machine axis (see 3.2 for a list of machine axis names and associated syntax).

a (real) specifies the preload value for the named machine axis. Preload values are measured in machine units for linear axes and degrees for rotary axes.

OFF (keyword) resets the machine axes to the initial state, clearing all preload values.

5.25.2 Examples

The following example moves the machine secondary Z axis (W) to a position 20 machine units from the axis origin. The LOCATE command then resets the machine axis origin for this axis to that point.

```
MOVETO/WAXIS, 20
LOCATE/WAXIS, 0
```

This next example is typical of common programming practice to output a block at the start of the machine program resetting the machine axes to a required starting position. It is the machine operator's responsibility to insure that the machine is correctly positioning before starting the machine program.

```
PARTNO/'SAMPLE'
LOCATE/XAXIS, 50, YAXIS, 0, ZAXIS, 10
FROM/50, 0, 10
...
...
```

5.25.3 Limitations

None.

5.26 The LPRINT command

Provides for control of the post processor listing.

LPRINT / (**ON**
OFF)

5.26.1 Semantics

This command controls the output of information to the post processor listing.

ON (keyword) specifies that all relevant data shall be output to the post processor listing file.

OFF (keyword) suspends the output of data to the post processor listing file.

5.26.2 Examples

None.

5.26.3 Limitations

None.

5.27 The MACHIN command

Defines the post processor.

$$\text{MACHIN} / a, b \left[\begin{array}{l} \text{INCH} \\ \text{MM} \\ \text{CM} \\ \text{FEET} \end{array} \right]$$

5.27.1 Semantics

This command identifies the post processor for a specific machine and provides parameters for initialization.

a (text) specifies the name of the post processor. The name is an unquoted 1 to 6 character alphanumeric string.

b (real) further identifies the post processor using a unique integer number.

INCH (keyword) specifies that the machine program units shall be inches, corresponding to the G70 definition in ISO 6983.

MM (keyword) specifies that the machine program units shall be millimeters, corresponding to the G71 definition in ISO 6983.

CM (keyword) specifies that the machine program units shall be centimeters.

FEET (keyword) specifies that the machine program units shall be feet.

5.27.2 Examples

The following example requests machine program units of MM for the post processor identified as "TEST" number 7.

MACHIN/TEST, 7, MM

5.27.3 Limitations

None.

5.28 The MATERL command

Defines the workpiece material.

$$\text{MATERL} / [\text{TYPE}, a] [, \text{COND}, b] \text{ or } \left[\left(\begin{array}{c} \text{XDIM} \\ \text{YDIM} \\ \text{ZDIM} \\ \text{DIAMET} \end{array} \right), c \right]$$

5.28.1 Semantics

This command defines the workpiece material by type, hardness and size.

TYPE, *a* (keyword,real) specifies the material type by a machine specific code number.

COND, *b* (keyword,real) specifies the material condition with the brinell hardness number.

[XYZ]DIM, *c* (keyword,real) specifies the length of the workpiece when measured along the specified part reference axis. The dimension value is in part reference units.

DIAMET, *c* (keyword,real) specifies the diameter of the workpiece. The dimension value is in part reference units.

5.28.2 Examples

The following command defines a sheet thickness of 3 millimeters for a turret punch machine.

```
MATERL/ZDIM, 3
```

5.28.3 Limitations

At least one form of type, hardness or size information must be specified.

The use of machine specific material codes may not be portable.

5.29 The MCHFIN command

Defines the acceptable machining finish.

$$\text{MCHFIN} / \left(\begin{array}{c} a \\ \text{FINE} \\ \text{COARSE} \end{array} \right)$$

5.29.1 Semantics

This command defines the acceptable machine finish to be produced by the operations performed on the material.

a (real) specifies the required finish measured in RMS.

FINE (keyword) specifies that a fine finish is required.

COARSE (keyword) specifies that a coarse finish is allowable.

5.29.2 Examples

None.

5.29.3 Limitations

None.

5.30 The MCHTOL command

Defines the acceptable machining tolerance.

$$\text{MCHTOL} / \left(\begin{array}{c} a \\ \text{FINE} \\ \text{COARSE} \\ \text{TO}, b, \text{PAST}, c \end{array} \right)$$

5.30.1 Semantics

This command defines the maximum allowable error from true tool path for machine slide dynamics at discontinuities.

a (real) specifies the magnitude of allowable error measured in part reference units.

FINE (keyword) specifies that the allowable error shall be minimized.

COARSE (keyword) specifies that a normal amount of error can be tolerated.

TO, b (keyword, real) specifies the magnitude of allowable undershoot measured in part reference units.

PAST, c (keyword, real) specifies the magnitude of allowable overshoot measured in part reference units.

5.30.2 Examples

None.

5.30.3 Limitations

None.

5.31 The MODE command

5.31.1 General comments

Controls one of machine positioning mode, curve interpolation mode, or circular interpolation mode specifications.

5.31.1.1 Sub-contents

For

- 1) machine positioning mode specification, see 5.31.2;
- 2) curve interpolation mode specification, see 5.31.3;
- 3) circular interpolation mode specification, see 5.31.4.

5.31.1.2 Limitations

None.

5.31.2 Machine positioning mode specification

MODE / $\left(\begin{array}{c} \text{INCR} \\ \text{ABSOL} \end{array} \right)$

5.31.2.1 Semantics

This command defines the positioning mode to be used at the machine.

INCR (keyword) specifies that machine program data commanding movement of machine axes shall do so using an incremental format, corresponding to the G91 definition in ISO 6983.

ABSOL (keyword) specifies that machine program data commanding movement of machine axes shall do so using an absolute format, corresponding to the G90 definition in ISO 6983.

5.31.2.2 Examples

None.

5.31.2.3 Limitations

None.

NOTE

The choice of machine positioning mode is an output convention. It has no effect on the interpretation of co-ordinate data in the part program, which is always specified in absolute terms from the part reference system origin.