TECHNICAL REPORT 6132

Published 1981-08-15

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION • МЕЖДУНАРОДНАЯ ОРГАНИЗАЦИЯ ПО СТАНДАРТИЗАЦИИ • ORGANISATION INTERNATIONALE DE NORMALISATION

Numerical control of machines — Operational command and data format

Commande numérique des machines — Format de données et de commandes

Technical Report 6132 was drawn up by Technical Committee ISO/TC 97, *Computers and information processing*, and approved by the majority of its members. It was decided to publish the document in the form of a Technical Report because it is not possible, in view of the present state of the art on the subject of numerical control systems, to draw up an International Standard which would be complete and precise and which would not be too restrictive in this rapidly developing field. It is intended as a guideline and will be reviewed and augmented periodically.

Contents

		Page
0	Introduction	2
1	Scope and field of application	2
1.1	Data types	2
1.2	Operational statements	2
1.3	Hardware configuration	2
1.4	Functional categories of commands	4
1.5	Machine programs	4
2	References	5
3	Format, symbols and conventions used in data entry	5
3.1	General information	5
3.2	Format for Operational Statements Commands	6
3.3	Conventions and symbols	6
4	Mode Selector Commands	7
4.1	General information	7
4.2	Mode Selector Commands	7
5	Universal Commands	7
5.1	General information	7
5.2	Universal Commands	7
6	Editor Commands	8
6.1	General information	8
6.2	Editor Commands	8
7	File Handler Commands	8
7.1	General information	8
7.2	Device mnemonics	8
7.3	File Handler Commands	9
8	Machine Control Commands	9
8.1	General information	9
8.2	Machine Control Commands	10
9	Keyboard	10
Annexes		
A	Programming examples and command summary	13
B	Character sets — LEVEL 1, 2 and 3	18

UDC 681.323 : 621.9-52 : 681.3.06 **Ref. No. ISO/TR 6132-1981 (E)**

Descriptors : data processing, numerical control, data layout, control procedures, high-level data link control, block formats, symbols.

Printed in Switzerland

Price based on 21 pages

ISO/TR 6132-1981 (E)

0 Introduction

This document is intended to provide guidelines for the design of a new set of command data for numerical control of machines. This new set is needed because of the rapid and widespread development of new techniques and new systems which may be termed "Advanced Systems". It is highly desirable that uncontrolled development of incompatible systems and simultaneous creations of different MAN-MACHINE LANGUAGES be avoided.

In order to take advantage of the flexibility and capability of current and future NC systems, it is necessary to be able to enter initialization and parameter data in addition to that included in the machine program formats specified in ISO 2539.

This Technical Report is intended to serve as a guide in the co-ordination of system design to promote uniformity in part programming and operating techniques for inputting extended machine set-up, initialization and/or operational parameter data. This format can be used either at a panel on the machine control equipment or at a central unit when DNC is used. The desirability of establishing uniformity and a degree of interchangeability of recorded input data for numerical control systems has long been recognized by equipment builders as well as users. Further, with extensive use of general purpose alphanumeric keyboards for entry of data by the operator, it is desirable to promote uniformity of manual entry formats between equipment to compensate for the lack of functional labelling of the system operator input devices.

In the past, this type of data was minimal and special entry devices such as functional pushbutton switches and digital entry switches (thumbwheel switches, for example) were used. The amount and variety of this type of data required make it desirable from economic and operational standpoints to enter this data via machine program input media such as punched tape or general purpose alphanumeric keyboards on the system.

1 Scope and field of application

1.1 Data types

The numerical control systems addressed by this Technical Report use a combination of two types of data :

1) Machine program data — Data formatted in accordance with ISO 2539 and coded in accordance with ISO 840.

2) Operational statements — Machine set-up, initialization and/or operational parameter data formatted in accordance with this Technical Report and coded in accordance with ISO 646.

Character subset :

0-8 0-9 0-10 0-13

characters of column 2 except 2-7

characters of column 3 except 3-11 and 3-15

characters of column 4 except 4-0

characters of column 5

characters 7-12 and 7-15

1.2 Operational statements

It is not expected that all NC systems will embody all the features and capabilities for which operational statements have been defined in this Technical Report. When an NC system has the capability to enter operational statements, the mnemonics and formats defined in this Technical Report shall be used. It is not intended that the functions defined herein shall limit the capabilities of NC systems; additional functions and operational statements may be defined and implemented by the control builder as required. It is also expected that in some instances, a control builder may, at his discretion, implement one or more of the functions defined herein exclusively with pushbutton switches or other conventional operator controls. If these functions are labelled in an abbreviated or mnemonic fashion, the mnemonics defined in this Technical Report shall be used.

1.3 Hardware configuration

This Technical Report deals with numerical control systems which may assume any of a wide variety of hardware configurations according to the purpose and capabilities required of the system. Figure 1 shows a numerical control system containing several types of

peripheral hardware. The primary concern of this Technical Report is the form of the data prepared by the part programmer or introduced by the operator into the numerical control system through one or more of these devices. In general, these data can be either "machine program data" or "operational statements". Binary data interchange in formats not conforming to this Technical Report may be utilized within or among various devices, at the convenience of the control builder, whenever such interchange is transparent to the user.

Figure 1 suggests some of the possible types of peripheral equipment that may be used with a numerical control system; it is not the intent of this Technical Report to specify or to limit the types of peripherals used in a given system. Other standards exist to define some of the physical interfaces implied by figure 1.

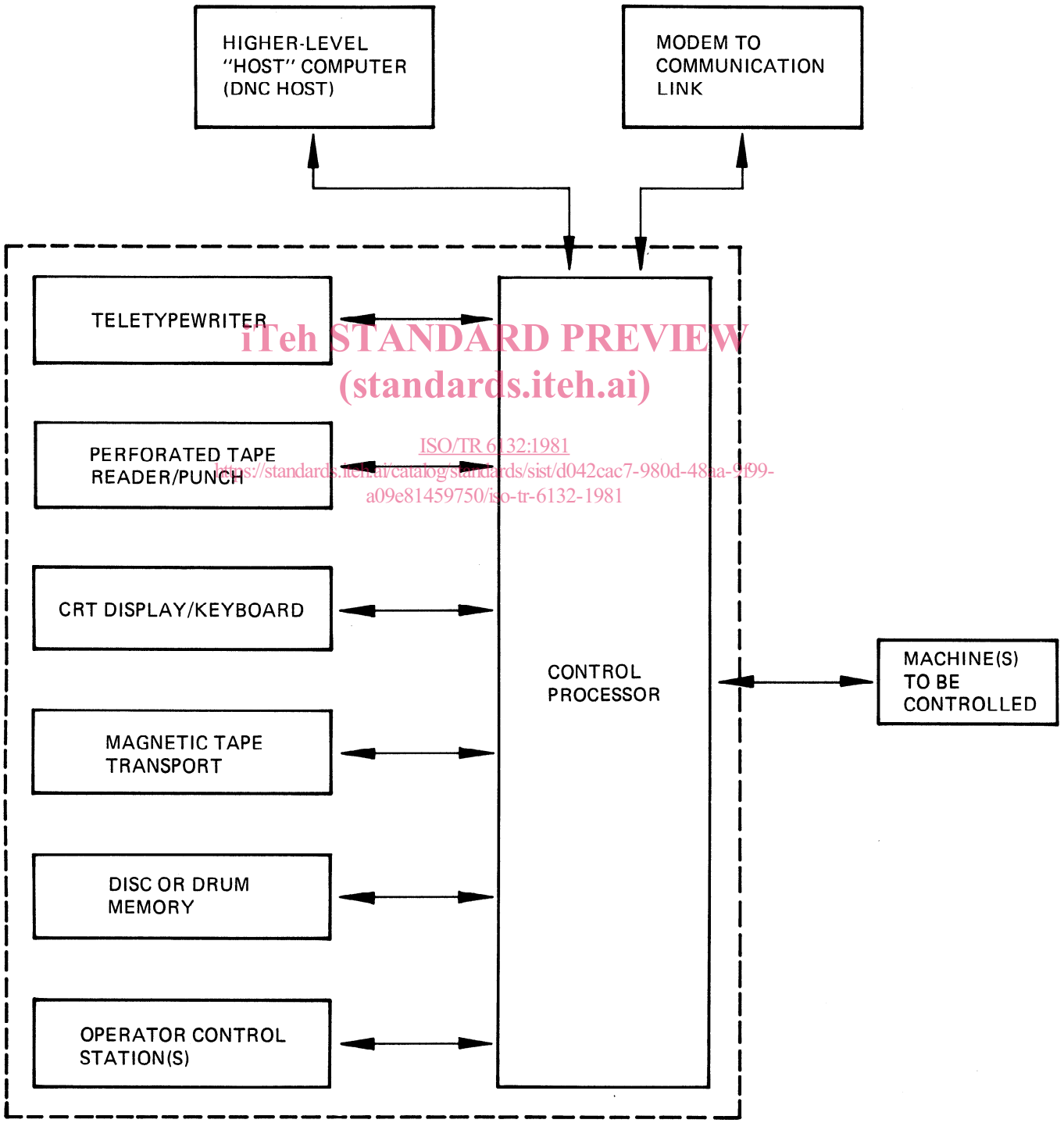

Figure 1 — Example of generalized hardware configuration of a numerical control system

1.4 Functional categories of commands

To illustrate the implementation of a system having a broad range of commands and with maximum utilization of an alphanumeric keyboard for command entry, a system having a structure organization according to figure 2 has been used.

Figure 2 shows several categories of commands that may be implemented. It is not the intent of this Technical Report to specifically require or to limit the commands to those shown in figure 2; the extent of the capabilities of any given numerical control system will depend upon the use to which the system is put. In some systems, diagnostic capabilities, management information system facilities and other specialized functions or categories might be provided in addition to those shown in figure 2.

A feature of the organization shown in figure 2 is the existence of a Mode Selector level in the system structure. An operator communicating with a numerical control system having this structure can gain access to any of the subordinate command categories from the Mode Selector level. Once he has entered the appropriate sub-level, he can perform any desired operation for which mnemonic commands have been defined in that command category. When he has completed his task, he can switch from the sub-level back to the Mode Selector level and, from there, enter any other desired sub-level. The primary function of the Mode Selector level in this illustration is to allow the operator to choose the desired type of system operation by means of keyboard commands. The control builder may wish to perform this mode selection in other ways, such as with a selector switch.

Note that the operations listed for each sub-level can involve the use of commands, initiated by the operator, to perform some function. Normally, in the course of issuing commands to the system, there would also be responses of the system back to the operator. These commands and responses would be transacted by means of a System Console which typically includes a data entry device and a display device and which may be a CRT/Keyboard terminal, a teletypewriter or other appropriate device.

Also shown in figure 2 is a category for Universal Commands. These commands are valid in all command modes including the Mode Selector level.

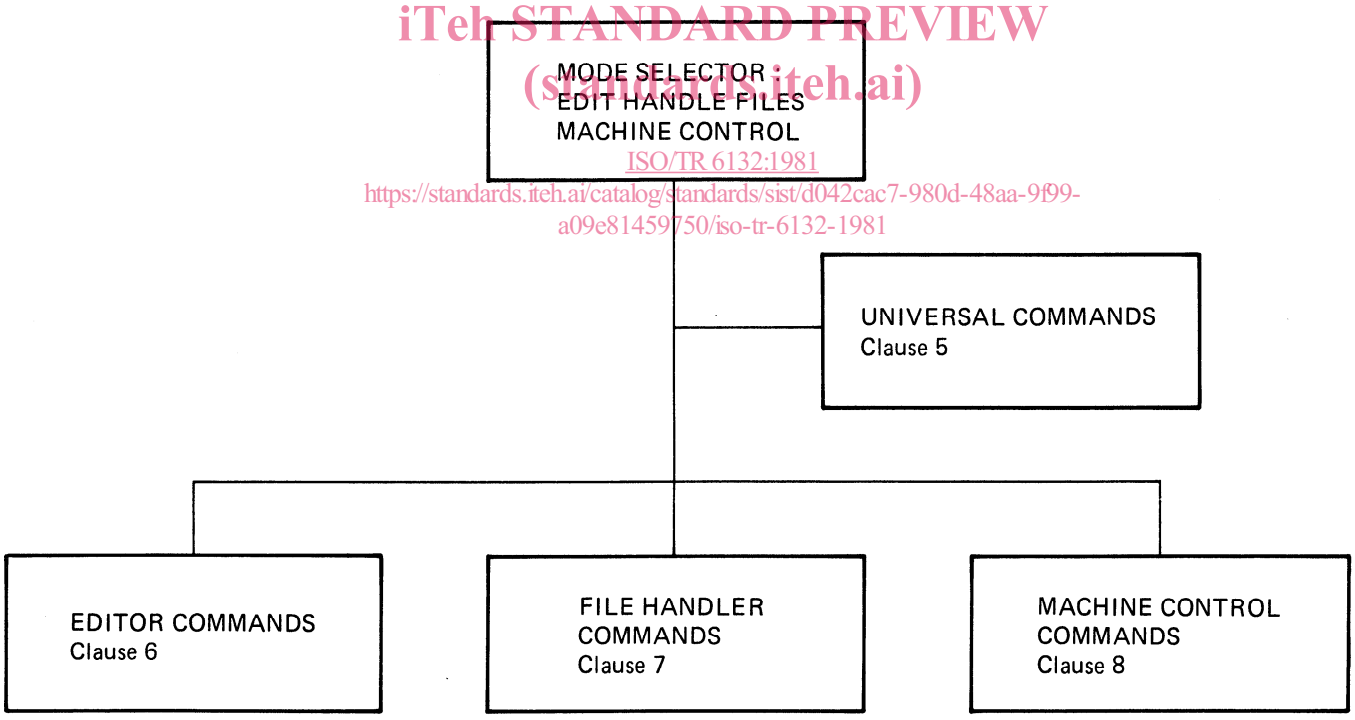

Figure 2 — Generalized functional command categories of a numerical control system

1.5 Machine programs

To facilitate the automatic operation of machines running under numerical control, it is desirable to provide the capability of combining some of the commands listed under the Machine Control sub-level with machine program data rather than restricting those commands to entry solely through the System Console.

In order to accomplish this, a structure for machine programs is established by this Technical Report, which can embody both "machine program data" and "operational statements". To allow the numerical control system to distinguish between the two types of data, Control Out and Control In codes are defined in accordance with ISO 2539 as part of the machine program structure.

Figure 3 shows a conceptualization of this structure. Note that a conventional machine program consisting of machine program data has embedded within it a sequence of operational statements set-up data. The open parenthesis, "(", is used to switch from the machine program data format to the operational statements format. The close parenthesis, ")", is used to return to machine program data format.

These format changes can occur throughout the machine program. It is recommended that sequence numbers as defined in ISO 2539 be used as line numbers. Sequence numbers should be assigned in ascending order.

It is strongly recommended that only persons with the necessary authority should be allowed to modify programs. When a switch is used for mode selection, operation should be by a key. When the mode is selected by push button or keyboard, passwords should be used with prompting from the system.

	Control Out (	Control In)
MACHINE PROGRAM DATA	OPERATIONAL STATEMENTS DATA	MACHINE PROGRAM DATA
Classical machine program data per ISO 2539 and ISO 840	Set-up data or operator messages per this Technical Report and ISO 646	Classical machine program data per ISO 2539 and ISO 840
	N42 X12487 Z76543 M01 NL N43 X34567 Z54321 NL	MACHINE PROGRAM DATA
	(MSL, 2*MSG, MACHINE NO 2 SELECTED)	OPERATIONAL STATEMENTS DATA
	N44 G01 X11268 NL N45 X25148 Y64318 F515 NL	MACHINE PROGRAM DATA

Figure 3 — Typical machine program data structure embodying both machine program and operational statements data

2 References

ISO 646, *7-bit coded character set for information processing interchange.*

ISO 840, *Numerical control of machines — 7-bit coded character set.*

ISO 2126, *Office machines — Basic arrangement for the alphanumeric section of keyboards operated with both hands.*

ISO 2539, *Numerical control of machines — Punched variable block format for contouring and contouring/positioning.*

ISO 3244, *Office machines and data processing equipment — Principles governing the positioning of control keys and keyboards.*

ISO 4342, *Numerical control processor input — Basic part program reference language.*[1]

ISO 6983/1, *Numerical control of machines — Word address program format and definition — Part 1 : Data format for positioning and contouring control systems.*[1]

ISO 6983/2, *Numerical control of machines — Word address program format and definition — Part 2 : Coding of preparatory functions, G, and miscellaneous functions, M.*[1]

3 Format, symbols and conventions used in data entry

3.1 General information

In the course of entering mnemonic commands from a keyboard, striking the wrong key may give rise to entry errors. The control builder should provide appropriate means to allow correction of and/or recovery from keyboard entry errors. Typically, functions would be defined to delete the previous character entered from the keyboard or to ignore the entire command just typed.

1) At present at the stage of draft.

Interactive "prompting" is recommended as an approach to operator/machine communication. Although not required by the commands defined by this Technical Report, such interactive response from the NC system assist in removing a burden from the operator and in reducing the probability of command errors. The control builder may elect, at his discretion, to lock out the keyboard of the System Console and/or to display error messages to the operator if commands are issued incorrectly or if, for any reason, the system is not ready to accept commands. The control builder may also elect to require the operator to enter a "password" to allow him access to certain sensitive commands.

It is also suggested that the control builder provide suitable interlocks for the machine motion system to inhibit motion (which could result unintentionally from an erroneous command) until the operator makes an active effort to initiate it (by pressing a Start button, for example).

3.2 Format for Operational Statements Commands

Operational Statements Commands shall be presented as operational statements whether entered manually or by other media. Each operational statement shall consist of a function mnemonic code followed, where appropriate, by one or more arguments qualifying the command.

Command function mnemonics are structured uniformly from three characters. The first and second characters will be alphabetic characters. The third will be an alphabetic or numeric character. Mnemonics beginning with the character U are permanently reserved for use by control builders and users.

Arguments contained within the operational statement shall be of such length and content as appropriate to the commands defined in the following sub-clauses. A word address format similar to that of ISO 2539 data is preferred in the presentation of arguments.

A delimiter character shall be used to separate the function mnemonic from argument(s). Additional delimiters may be required to separate arguments from one another or operational statements from each other.

3.3 Conventions and symbols

The operator initiates system responses to a command by entering the command and pressing a key, meaning "enter the command into the control system". In this Technical Report, the symbol ⌟ will be used to represent pressing the enter-key.

In the text that follows, certain command arguments are printed in italic. This convention is used to mean that the italicized element is variable and will be replaced (by the programmer or operator) by a specific entry according to the desired effect.

Symbol	Usage	Example
⌟	Represents pressing a key meaning "enter the command into the control system", causing termination of the command and initiation of the commanded function.	FIL ⌟
, (comma)	The comma is a delimiter which separates the command mnemonic from the argument which follows it and, when necessary, separates successive arguments of a command.	MIR, XIYO ⌟
n, *m*	Arguments representing line numbers within a file being edited, taking "lines" to be text separated by NEW LINE (line feed) code. *n* is the first line and *m* is the last line in the direction of normal program execution.	DEL, *n*, *m* ⌟
(2/2)"	A text delimiter required when using certain Editor Commands. No character need be printed.	
((open parenthesis)	Control Out ISO 2539. Used, when constructing machine programs employing both machine program and operational statements data, as the code to enter the operational statements data mode.	See figure 3
) (close parenthesis)	Control In ISO 2539. Code to return to the machine program data mode.	See figure 3
* (asterisk)	Delimiter to separate successive Commands within the operational statements data mode of machine programs employing both machine program and operational statements data.	See figure 3
NL	Represents the End of Block character (new line) for ISO 2539/840 programs.	See figure 3
< (less than)	Used in the operational statements mode to denote the beginning of a nested command.	See 7.3
> (greater than)	Used in the operational statements data mode to denote the end of a nested command.	See 7.3
/ (slash)	Delimiter used with File Handler Commands to separate *filenames* from *devicename*.	See 6.2

4 Mode selector commands

4.1 General information

The Mode Selector level is a portion of command structure that allows the operator to select the type of System operation. Its utilization is not compulsory and not functionally necessary, but is given as an option for organizing the operator procedure.

If utilized, it shall be in accordance with this Technical Report.

The Mode Selection, if any, may be performed :

a) by keyboard (Mode Selector commands table 1);

b) by separate push buttons;

c) by a switch.

The status of the System (Mode Selector level) should be clearly indicated to the operator.

4.2 Mode Selector Commands

Table 1 lists the mnemonic code, the format and the function of each Mode Selector Command. These commands correspond to the three command categories (operation modes) illustrated in this Technical Report.

Note that in several cases further interaction between the system and operator may be required to qualify the basic command.

Some NC systems may be configured with some or all of these sub-levels or additional sub-levels, or both, as required by the system. If additional sub-levels are provided by the control builder, appropriate commands shall be defined to access them.

5 Universal commands

5.1 General information

Universal commands are intended for general system control rather than for implementation of specific operational functions. The universal commands must be effective without selection of a particular mode.

5.2 Universal Commands

Table 2 lists the mnemonic code, the format and the function of each Universal Command.

Table 1 — Mode Selector Commands

Mnemonic	Format	Function
EDT	EDT, *oldfile*, *newfile* ⌟	Conditions system to enter Editor sub-level. *Oldfile* is altered if *newfile* is not defined
FIL	FIL ⌟	Conditions system to enter File Handler sub-level
MCH	MCH, *machine-name* ⌟	Conditions system to enter Machine Control sub-level. Specification of *machinename* optional
END	END	End of all the modes

Table 2 — Universal Commands

Mnemonic	Format	Function
DIS	DIS, *arguments* argument *m*, device name argument *n* . . . ⌟	DISPLAY on the device selected the parameters specified by the *arguments*. A list of arguments and the resulting displayed parameters, shall be specified by the control builder

6 Editor commands

6.1 General information

The Editor sub-level is a portion of the command structure by means of which the operator can change existing files. The system enters this category from the Mode Selector level through the command EDT. The System may request the operator to enter the name of the file to be edited.

6.2 Editor Commands

Table 3 lists the mnemonic code, the format and the function of each Editor Command. Where all of the arguments supported by the formats of the Editor Commands are not required, alternate forms may be used.

Table 3 — Editor Commands

Mnemonic	Format	Function
DEL	DEL,*n*,*m*, *text* ⌟	DELETE *text* in lines *n* through *m*
FND	FND,*n*,*m*, *text* ⌟	FIND *text* in lines *n* through *m*
INS	INS,*n*, *text* ⌟	INSERT *text* after line *n*
LST	LST,*n*,*m* ⌟	LIST lines *n* through *m*
RPL	RPL,*n*,*m*, *text* ⌟	Delete lines *n* through *m* and REPLACE with *text*

7 File handler commands

7.1 General information

The File Handler sub-level is a portion of the command structure by means of which the operator can create new files, list file directories, delete unwanted files from a storage device, transfer files between storage devices and copy files. The system enters this program from the Mode Selector level through the command FIL.

7.2 Device mnemonics

Since the File Handler is concerned not only with file names, but also device names, it shall be understood in the following format descriptions that whenever the terms *filename* or *newfilename* or *oldfilename* are used, reference shall also be made, if necessary (as defined by the control builder), to the device containing that file. The format to be used is *filename/devicename*. To allow the File Handler to distinguish between the file names and device names, a slash (/) delimiter is used. Standard device mnemonics are listed in table 4.

Table 4 — Standard Device Mnemonics

Mnemonic	Device
AM_n	Auxiliary Memory, *n*th device
DS_n	Disk storage unit, *n*th device
KB_n	Console keyboard
MMS	Main memory storage
MT_n	Magnetic tape unit, *n*th device
PP_n	Perforated tape punch, *n*th device
PR_n	Perforated tape reader, *n*th device
TT_n	Teleprinter or display terminal, *n*th device

7.3 File handler commands

Table 5 lists the mnemonic code, the format and the function of each File Handler command.

Table 5 — File Handler Commands

Mnemonic	Format	Function
APD	APD, *newfilename,* *oldfilename*$_1$, . . . *oldfilename*$_n$ ⌟	To create a new file by chaining (APPENDING) the contents of the old files in the order listed
CRE	CRE, *newfilename* ⌟	CREATE a new file named *newfilename*
DIR	DIR, *devicename* ⌟	Output the DIRECTORY of all file names contained on the device identified by *devicename*
DLF	DLF, *filename*	DELETE the FILE called *filename*
REN	REN, *oldfilename* *newfilename* ⌟	RENAME. Change the name of a file
XFR	XFR, *oldfile,* *newfile* ⌟	TRANSFER the contents of *oldfile* to *newfile*. The contents of *oldfile* are not affected

8 Machine control commands

8.1 General information

The Machine Control Commands defined in this clause are intended for two purposes : To enable the operator to enter machine control information through the System Console and to permit machine control information to be included with conventional machine program data in two-level machine programs, i.e. programs using operational statements data.

During its operation, the control system must often have information available to it which is impossible to program using the format of ISO 2539 alone.

This additional information consists of data such as mirror image conditions, the amount of cutter compensation, the amount of zero offset and similar functions that are unique to each job. Also, it is sometimes necessary that the operator be able to specify certain conditions of execution of the machine program such as one-block-at-a-time operation.

This clause defines the formats to be used in entering this information, whether it is through the System Console or by means of two-level machine programs.

8.1.1 Manual Operation from System Console

The system enters the Machine control sub-level through the command MCH. System dialogue ensues to request the operator to enter the identification number of the machine, in multiple-machine systems, to which the subsequent Machine Control commands are to be directed. Commands may be directed to another machine by entering the END code to return to the Mode Selector level.

8.1.2 Automatic Operation with Two-Level Machine Programs

When Machine Control Commands are embedded in machine programs, two-level operation is achieved by using the open parenthesis Control Out, "(", and close parenthesis Control In,")", codes to switch from and return to the ISO 2539 format.

When execution of a machine program begins, it is understood that the control system for the machine will interpret machine program data initially according to the format of ISO 2539. When an open parenthesis is detected, the control will interpret all successive data as Machine Control (set-up) Commands according to this Technical Report until a closed parenthesis is detected, whereupon interpretation will resume in ISO 2539 format.

Control commands may be entered as many times as desired throughout a machine program, switching each time to the Machine Control mode and returning to the ISO 2539 mode. Within the Machine Control mode, several Type 2 Machine Control Commands may be issued in succession if they are separated by an asterisk, "*", delimiter.

Under certain conditions it is also permissible to construct Machine Control Commands within other Machine Control Commands, by a technique known as "nesting".

The nested command is enclosed between the " < " and " > " characters to delimit it from the Machine Control Command in which it is embedded. This structure can be used with the CLS (call sub-program) or RPT (repeat) commands, each of which involves the definition of a sub-program in ISO 2539 format which itself can embody other Type 2 data, perhaps including other sub-programs, etc. It shall be the responsibility of the control builder to specify the extent to which Machine Control Commands can be nested.

Several of the Machine Control Commands are directly executable, that is, they result in a specified machine motion. The executable commands are CLS, GRZ, REF, and RPT. To ensure that these functions are not executed unexpectedly, execution shall be delayed until the (NL) following the Control in code, ")", read by the control system.

8.2 Machine Control Commands

Table 6 on the following pages lists the mnemonic code, format and the function of each Machine Control Command.

Note that formats have been defined generally for arguments of many of the commands. It is the responsibility of the control builder to specify these formats in detail according to their intended use. When Machine Control Commands are entered through the System Console, it is recommended that the control builder provide interactive prompting and comprehensive error messages wherever possible to reduce the possibility of error in data input.

Specification of the units (for example inch or metric) of Type 2 data is optional and may be indicated by a descriptive alpha character as a part of an argument.

9 Keyboard

The keyboard arrangement shall comply with ISO 2126 and ISO 3244. It is recommended that, within the limits imposed by these International Standards, the keyboards be arranged in the best way to facilitate their use by personnel not trained in typing.

Table 6 — Machine Control Commands

Mnemonic	Format	Function
ACP	ACP, *argument*$_1$ . . . *argument*$_n$	ADAPTIVE CONTROL PARAMETER specification. Format for arguments to be specified by control builder
AXO	AXO,*axis*$_1$ *offset*$_1$ . . . *axis*$_n$ *offset*$_n$	AXIS OFFSET. Specifies displacement of zero location of each axis. Modal axis-by-axis
AXI	AXI, *mach axis*$_1$ *input axis*$_1$, . . . *mach axis*$_n$ *input axis*$_n$	AXIS INTERCHANGE. Assigns input axis data to each machine axis. Modal axis-by-axis
BAK	BAK, *n*	BACK up in the program *n* blocks, executing motion commands in the reverse direction
BLD	BLD, 1 (enable) BLD, 0 (cancel)	BLOCK DELETE. Enables the Block Delete function or clears it. Modal
BLK	BLK, 1 (enable) BLK, 0 (cancel)	BLOCK by BLOCK. Enables machine program execution one block at a time. Each block is initiated by pressing the Cycle Start button
CLS	CLS, *n*, *r*, *filename*/*devicename*	CALL SUBPROGRAM. Execute subprogram *n* previously defined by a DFS command *r* times. Subprogram *n* might be found in file *filename* on device *devicename*. The only required parameter is *n*; the others may be used as individual applications dictate
CCP	CCP, *C no*$_1$ *delta*$_1$ . . . *C no*$_n$ *delta*$_n$	CUTTER COMPENSATION. Used to establish assignable Cutter Radius or diameter compensation values. Assigns a delta value to a specified location in a table of values. The table of values location to be identified by C. Radius delta values prefixed by R; diameter delta values prefixed by D
DFS	DFS, *n*, *subprogram*	*DEFINE SUBPROGRAM.* Define a subprogram and identify it as number *n* to be executed subsequently in the part program (as many times as desired) by one or more CLS commands. Modal for each *n*
FRL	FRL, *axis*$_1$ *ipmlimit*$_1$ . . . *axis*$_n$ *ipmlimit*$_n$	FEED RATE LIMIT. Specifies the maximum allowable programmed feed rate in inches per minute or mm per minute, as required. Modal axis-by-axis
FRO	FRO, *percent*	FEED RATE OVERRIDE. Specifies the feed rate override to apply to programmed feed rates (60 = 60 % of programmed feed rate). Optional specification of axis set with format FRO, *axis set*$_1$ *percent*$_1$ . . . *axis set*$_n$ *percent*$_n$
FXC	FXC, *fixture no.* *axis*$_1$ *dimension*$_1$ . . . *axis*$_n$ *dimension*$_n$, . . . *fixture no*$_m$ *axis*$_1$ *dimension*$_1$ . . . *axis*$_n$ *dimension*$_n$	FIXTURE COMPENSATION. For each fixture specified, fixture compensation dimensions are given. Modal by fixture
FST	FST, *format detail*	FORMAT STATEMENT. Specifies the format detail, per ISO 2539 of the input program. Modal
GRZ	GRZ, *axis*$_1$ . . . *axis*$_n$	GRID ZERO enables slide movement to the nearest grid on the axes specified (non-modal function)
REF	REF, *axis*$_1$ . . . *axis*$_n$	REFERENCE POSITION enables slide movement to the Reference Position on the axes specified. (Non-modal function)
JMA	JMA, 1 (enable) JMA, 0 (cancel)	JOG with MEMORY — ACTIVATE. Following this command, subsequent JOG operations are "remembered" by the control system so that at a later time the reverse motion path can be taken with a JMR command. JMA, 0 clears all moves stored under JMA, 1
JMR	JMR	JOG with MEMORY — RETURN. Enables the motions of JOG operations with JMA, 1 to be reversed so the machine position can be restored to its location before the JMA, 1 command. JMR clears moves stored under JMA, 1 after reverse execution
MCR	MCR, *axis*$_1$ argument$_1$ . . . *argument*$_n$ *axis*$_2$ *argument*$_1$, . . . *argument*$_n$ *axis*$_3$	MACHINE CORRECTION. Specified dimensional correction data for machine axes. Modal for each argument
MEX	MEX, *sourcefile*, *editfile*	MERGE AND EXECUTE. Run the machine program in *sourcefile* altered by the contents of *editfile*. MEX, 0 cancels the command
MIR	MIR, *axis*$_1$ *code*$_1$ . . . *axis*$_n$ *code*$_n$ (Example : X1 = reverse X0 = normal)	MIRROR IMAGE. Specifies which axes are to have reversed motion. Axes not specified execute normal motion. Modal axis-by-axis
MSG	MSG, *message*	Specifies a MESSAGE to be displayed
MSL	MSL, *machinecode*	MACHINE SELECT. This command directs Machine Control commands to a specified machine. Modal
OSP	OSP, 1 (enable) OSP, 0 (cancel)	OPTIONAL STOP. Enables the optional stop command or clears it. Modal
RPT	RPT, *r*, *subprogram*	REPEAT PROGRAM. Causes the operation sequence *subprogram* to be executed the number of times specified by *r* (repetitions)