

CAMM-GL II

Programmer's Manual

ROLAND DG CORPORATION

CAUTION

1. Unauthorized copying or transferral, in whole or in part, of this manual is prohibited.
2. The contents of this operation manual and the specifications of this product are subject to change without notice.
3. The operation manual and the product have been prepared and tested as much as possible. If you find any misprint or error, please inform us.
4. We cannot in any way assume any responsibility whatsoever with regard to whatever consequences that may happen subsequent to the making of changes or alterations to this product. We also cannot in any way assume responsibility for whatever may result when this product is operated, or with regard to whatever results from making use of any explanatory documentation.

Contents

Introduction

Part 1 General Description of the CAMM-GLII

1.1 INSTRUCTION SYSTEM	1-2
1.2 A BRIEF LOOK AT THE mode1 INSTRUCTIONS	1-3
1.2.1 Syntax of mode1 Instructions	1-3
1.2.2 BASIC Syntax	1-5
1.3 A BRIEF LOOK AT THE mode2 INSTRUCTIONS	1-6
1.3.1 mode2 Instructions	1-6
1.3.2 Syntax of mode2 Instructions	1-6
1.3.3 BASIC Syntax	1-8
1.3.4 Instruction Overview	1-9
1.4 mode1 and mode2 COMMON INSTRUCTIONS	1-18

Part 2 mode1 Instructions

H	2-2	Q	2-14
D	2-3	N	2-15
M	2-4	C	2-16
I	2-5	E	2-17
R	2-6	A	2-18
L	2-7	G	2-19
B	2-9	K	2-20
X	2-11	T	2-22
P	2-12	^	2-24
S	2-13		

Part 3 mode2 Instructions

AA	3-2	OI	3-33
AR	3-3	OO	3-34
CA	3-4	OP	3-35
CC	3-5	OS	3-36
CI	3-6	OW	3-37
CP	3-7	PA	3-38
CS	3-9	PD	3-39
DF	3-10	PR	3-40
DI	3-11	PT	3-41
DR	3-13	PU	3-42
DT	3-14	RA	3-43
EA	3-15	RR	3-44
ER	3-16	SA	3-45
ES	3-17	SC	3-46
EW	3-18	SI	3-47
FT	3-20	SL	3-48
IM	3-21	SM	3-49
IN	3-22	SR	3-50
IP	3-23	SS	3-51
IW	3-24	TL	3-52
LB	3-25	UC	3-53
LT	3-26	VS	3-55
OA	3-28	WD	3-56
OC	3-29	WG	3-57
OE	3-30	XT	3-58
OF	3-31	YT	3-59
OH	3-32		

Part 4 Common Use Instructions

!MC	4-2	!VZ	4-5
!NR	4-3		
!PZ	4-4		

Introduction

• The Content of the Manual

- * The manual provides a description of the graphic language CAMM-GL II loaded in computer-output devices of Roland DG Corp.
It explains the instructions of CAMM-GL II independent of special devices and technology.
- * The instruction explanations in the manual do not provide any parameter range because each device has its own specifiable parameter range varying from one model to another. For information about the parameter range, refer to Operation Manual for each device.

• How to Read the Manual (About symbols)

- * The manual provides programming examples written in the format of BASIC language. If you want to change the format, please do it according to the computer and the programming software used.
 - * Control characters are marked as shown in <Example 1>
<Example 1> ESC (1B₍₁₆₎) or ESC
(The "1B₍₁₆₎" shows that the number is hexadecimal.)
 - * Character codes are marked as shown in <Example 2>
<Example 2> 00₍₁₆₎ or CHR\$(Character code)
(The "00₍₁₆₎" shows that the number is hexadecimal.)
-

Chapter 1

General Description of the CAMM-GLII

1.1 INSTRUCTION SYSTEM

The CAMM-2 is designed to employ the CAMM-GL II instruction system.

CAMM-GL II system is compatible with Roland DG Corporation plotters, cutting machines and modeling machines, and consists of two types of instructions sets, which are:

- **mode1:** Includes two types: one-letter instructions, and three-letter instructions headed by ! which are in common with mode2 instructions.
These instructions function independently from one another, thus making for easy engraving.
mode2 instructions can be used by calling them.
Consideration has been made for to ensure compatibility with DXY-GL for plotters from Roland DG Corp. and with CAMM-GL III for the CAMM-1 Series of cutting machines (mode1), as well as with CAMM-GL I for the CAMM-3 Series of compact modeling machines (mode1).
- **mode2:** Includes two types: two-letter instructions, and three-letter instructions headed by ! which are in common with mode1.
mode2 is systematically organized to provide for the efficient operation of the CAMM-2. All instructions are very flexible, and allow for a great deal of precision when making settings for the tasks your CAMM-2 is to perform.
Consideration has been made for to ensure compatibility with RD-GL I for plotters from Roland DG Corp. and with CAMM-GL III for the CAMM-1 Series of cutting machines (mode2), as well as with CAMM-GL I for the CAMM-3 Series of compact modeling

1.2 A BRIEF LOOK AT THE *mode1* INSTRUCTIONS

1.2.1 Syntax of mode1 Instructions

A mode1 instruction can be divided into the following four parts:

- **Instruction**
- **Parameter(s)**
- **Delimiter**
- **Terminator**

The syntax of a mode1 instruction is as shown in the following example.

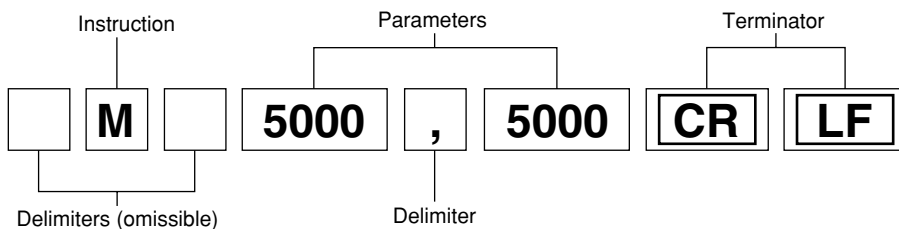


Figure 1-1

• **Instruction**

mode1 instructions consist of one letter, or three letters beginning with `!`, and provide the CAMM-2 with the information it needs to operate. The example in Figure 1-1 denotes that the tool is to be moved to the coordinates given after the instruction.

• **Parameter**

Parameters supply the necessary coordinate values, characters, or other data that the instruction requires. The example in Figure 1-1 denotes that the tool is to be moved to the coordinates 5000, 5000.

The meaning of the parameters will vary depending on the instruction. There are three types of instructions: those that have no parameters, those that require parameters, and those that carry parameters which can be omitted.

The permissible range for a parameter varies depending on the instruction. The `+` sign can be omitted from numerical values. If a parameter is omitted with an instruction that permits this, the default values will be used. Certain instructions require that dummy parameters be supplied, even though they have no meaning.

• **Delimiter**

Delimiters are used as separators between an instruction and a parameter, as well as between successive parameters.

If an instruction is given without delimiters separating its parameters, the CAMM-2 will not be able to recognize the individual parameters, when two or more parameters are run together. To prevent this, you need to supply delimiters.

As the delimiter, you can use either a space or a comma.

Chapter1

You can omit the delimiter between an instruction and the first parameter since the CAMM-2 will understand it, but you cannot omit delimiters between two or more parameters. Thus, instructions could be written in any of the following ways:

```
M 5000 5000
M5000,5000
M5000 5000
```

Throughout this manual, however, the following format is shown in the interest of simplicity.

```
M5000,5000
```

• Terminator

The terminator indicates the end of an instruction. It can be omitted only when another instruction immediately follows the first instruction.

The terminator consists of the **CR** (0D₍₁₆₎) and **LF** (0A₍₁₆₎) characters.

In mode2, you alternately can use a semicolon, ";" as the terminator.

NOTE

- Be careful with the signs of parameters. If the location of the sign is improper, an error will be produced.

Example

Data content	Initially interpreted as:	After error is cleared, treated as:
[M- □ 300,200] -->	[M] [0] [300] [200] -->	[M] [0] [300]
[M-300, + □ 200] -->	[M] [-300] [0] [200] -->	[M] [-300] [0]

*The "□" character indicates a space.

Since the CAMM-2 interprets space as a delimiter, it displays an error message because there appear to be three parameters. It then only accepts the first two parameters.

- If no terminator is sent, the CAMM-2 may fail to operate. For example, if you send the data below, the unit cannot determine whether the final D 100,200 is D 100,200 or D 100,2008 until the last data is received. It continues waiting until the last data or terminator has been received.

```
M 0,0 CRLF PD 1000,1000 CRLF M 500,700 CRLF D 100,200
```

The terminator conveys to the CAMM-2 that a series of data is complete.

- Sometimes there is no need to specifically include a terminator. If, for example, you type in line 1 below using IBM BASIC, the data in 2 will automatically be sent from the parallel port (printer terminal). This applies only for model1.

```
1. LPRINT "M1000,1000"
2. [M] [1] [0] [0] [0] [,] [1] [0] [0] [0] CRLF
```

1.2.2 BASIC Syntax

The example in Figure 1-1 shows how data can be sent to the CAMM-2. It could be written in BASIC in either of the two ways, depending on whether a Parallel or Serial configuration is used.

```
OPEN "LPT1:" AS #1                      (Parallel Configuration)
PRINT #1, "M5000,5000"
or
LPRINT "M5000,5000"
```

```
OPEN "COM1:9600,N,8,1" AS #1           (Serial Configuration)
PRINT #1, "M5000,5000"
```

For a Serial configuration, you need to have all protocol items (baud rate, number of data bits and stop bits, and parity checking) matched on both computer and CAMM-2; and need to open the RS-232C port beforehand with an "OPEN" statement.

You can also assign parameters to variables. For example, instead of this

```
10 OPEN "LPT1:" AS #1
20 PRINT #1, "M1000,2000"
```

you could assign variables as follows:

```
10 OPEN "LPT1:" AS #1
20 X=1000,Y=2000
30 PRINT #1, "M";X;" ";Y
```

You can also rewrite line 30 as follows:

30 PRINT #1, "M";X;" ";Y	Use space instead of comma
30 PRINT #1, "M";X,Y	Separate variables with comma
30 PRINT #1, "M"X,Y	Omit semicolon between instructions and variable

Moreover, you can write label parameters as follows:

```
20 B$="ABCDabcd"
30 PRINT #1, "P";B$

20 X= 1000,Y=2000
30 PRINT #1, "P";X;Y
```

The syntax may vary depending on the model of computer and programming software used. Refer to the operation manuals for your computer and software.

1.3 A BRIEF LOOK AT THE *mode2* INSTRUCTIONS

1.3.1 mode2 Instructions

mode2 instructions can be divided into the following six categories.

- | | |
|-----------------------------------|--|
| • INITIALIZATION Instructions | Set defaults, etc. |
| • COORDINATE CONTROL Instructions | Set scaling points, window, etc. |
| • TOOL CONTROL Instructions | Control tool speed, moves, engraving, etc. |
| • ENHANCED Instructions | Provide a variety of additional engraving features |
| • LABELING Instructions | Used to engrave characters |
| • OUTPUT Instructions | Request output of data from the CAMM-2 |

For actual engraving, the above instructions are combined to form a program.

1.3.2 Syntax of mode2 Instructions

Each mode2 instruction can be divided into the following four parts.

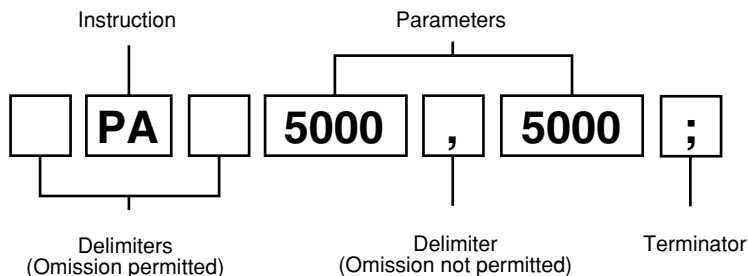


Figure 1-2

• Instruction

mode2 instructions provide the CAMM-2 with the settings it needs to operate. There are two types of instructions: two-letter instructions and three-letter instructions that begin with !. They can be in either lower case or upper case letters. The example in Figure 1-2 denotes that the tool is to be moved to the coordinates specified by the parameters.

• Parameters

Parameters supply the necessary coordinate values, characters, or other data that the instruction requires. The example in Figure 1-2 denotes that the tool is to be moved to the coordinates 5000, 5000. There are three types of instructions:

- Instructions without parameters
- Instructions requiring parameters
- Instructions with omissible parameters

Depending on the instruction, the meaning and range of the parameters will vary. You can omit the + sign from numeric parameters. If a parameter is omitted with an instruction that permits this, the default values will be used.

• Delimiter

Delimiters are used as separators between an instruction and a parameter, as well as between successive parameters. For the delimiter, you can use either space or a comma.

You can omit the delimiter between an instruction and the first parameter, but you cannot omit delimiters between two or more parameters. Thus, instructions could be written in any of the following ways:

```
PA 5000 5000;
PA5000,5000;
PA5000 5000;
```

Throughout this manual, however, the following format is shown in the interest of simplicity.

```
PA5000,5000;
```

• Terminator

A terminator indicates the end of an instruction. It can be omitted only when another instruction immediately follows the first instruction. The terminator consists of a semicolon, “;”.

However, since the LB and WD instructions will interpret a semicolon as a character, you need to use ETX (03₍₁₆₎) as a label terminator.

NOTE

- Be careful with the signs of parameters. If the location of the sign is improper, an error will be produced.

Example

Data content	Initially interpreted as:	After error is cleared, treated as:
[PA- □300,200] --->	[PA] [0] [300] [200] --->	[PA] [0] [300]
[PA-300,+ □200] --->	[PA] [-300] [0] [200] --->	[PA] [-300] [0]

*The "□" character indicates a space.

Since the CAMM-2 interprets space as a delimiter, it displays an error message because there appear to be three parameters. It then only accepts the first two parameters.

- If no terminator is sent, the CAMM-2 may fail to operate. For example, if you send the data below, the unit cannot determine whether the final PD 100,200 is PD 100,200 or PD 100,2008 until the last data is received. It continues waiting until the last data or terminator has been received.

```
PU0,0;PD1000,1000;PU500,700;PD100,200
```

The terminator conveys to the CAMM-2 that a series of data is complete.

In mode2, you simply use a semicolon ";" as a terminator.

1.3.3 BASIC Syntax

The example in Figure 1-2 shows the syntax of data sent to the CAMM-2. It could be written in BASIC in either of the two ways, depending on whether a Parallel or Serial configuration is used.

```

OPEN "LPT1:" AS #1                      (Parallel Configuration)
PRINT #1, "PA5000,5000;"

or

LPRINT "PA5000,5000;"

OPEN "COM1:9600,N,8,1" AS #1            (Serial Configuration)
PRINT #1, "PA5000,5000;"

```

For a Serial configuration, you need to have all protocol items (baud rate, number of data bits and stop bits, and parity checking) matched on both computer and CAMM-2; and need to open the RS-232C port beforehand with an "OPEN" statement.

You can also assign parameters to variables. For example, instead of this:

```

10 OPEN "LPT1:" AS #1
20 PRINT #1, "PA1000,2000;"

```

You could assign variables as follows:

```

10 OPEN "LPT1:" AS #1
20 X=1000,Y=2000
30 PRINT #1"PA";X;" ";Y;" ";

```

You can also rewrite line 30 as follows:

```

30 PRINT #1, "PA";X;" ";Y;" ";          Use space instead of comma
30 PRINT #1, "PA";X,Y;" ";              Separate variables with comma
30 PRINT #1, "PA"X,Y;" ";               Omit semicolon between variables

```

Moreover, you can write label parameters in the following ways.

```

20 B$="ABCDabcd"
30 PRINT #1, "LB";B$;CHR$(3)

20 X=1000,Y=2000
30 PRINT #1, "LB";X;Y;CHR$(3)

```

The syntax may vary depending on the model of computer and programming software used. Refer to the operation manuals for your computer and software.

1.3.4 Instruction Overview

1. INITIALIZATION Instructions

The DF and IN instructions cause any settings that have been modified (feed rate, character size, etc.) to be reverted to their default values. The IN instruction places the CAMM-2 in the same status it is in when power is first turned on. It is usually good practice to execute these instructions before beginning to send data.

Default instruction	DF
Initialize instruction	IN
Input mask instructions	IM

2. COORDINATE CONTROL Instructions

These instructions are employed when wishing to make settings pertaining to coordinate unit, origin, and engraving area.

2-1 SCALING

The CAMM-2 can operate in accord with three coordinate systems: a machine coordinate system, a work coordinate system, and a user coordinate system. "Scaling" refers to the establishment of such user coordinates.

By scaling you can set, as well as compress or expand, the coordinate units.

The coordinate parameters of an instruction follow the machine coordinate system until you scale. After that, the CAMM-2 performs the engraving in accord with the user coordinate system.

• Machine Coordinate System

Machine coordinates are a system of coordinates determined mechanically for the CAMM-2. The origin point is fixed at a certain point on the CAMM-2 (this point varies from one model to another).

The length of one unit in the machine coordinate system is fixed. For example, if one unit is 0.01 mm, then to obtain 1mm you should specify 100 units.

• Work Coordinate System

The work coordinate system has an origin point which can be moved as needed compared with the fixed origin point of the machine coordinate system. In the work coordinate system, the origin point for the X-Y axes is called the "home position," and the Z-axis origin is called "Z0." The origin point for work coordinates is determined by the specifying these two points.

After power is turned on the work coordinate system origin point is set at the same position as the machine coordinate system. Coordinate system units are also the same as the machine coordinate system.

• User Coordinate System

This coordinate system allows the user to set, relative to the machine coordinate system, any desired coordinate origin and coordinate units. Once user coordinates have been set, all subsequent coordinate parameters will comply with the user coordinate system.

If you specify coordinates beyond the maximum parameter range, an error occurs. When setting user coordinates, you should ensure that the engraving area does not extend beyond the permissible parameter range.

- **Setting Scaling Points P1 and P2**

Scaling points P1 and P2 form the reference points for scaling (or setting user coordinates). P1 and P2 are set at the desired points, and scaling is performed by executing an SC instruction which establishes the coordinate values of each point. Remember that setting P1 and P2 is not enough to complete the scaling. The IP instruction is used to set P1 and P2.

For example, if you send the following data with the IP instruction:

```
"IP100,150,4000,2500;"
```

P1 is set to (100,150) and P2 is set to (4000,2500). Here, the coordinate value parameters must be in work coordinates. Although the user coordinate system may be in effect at this point, such as when re-scaling, these parameter values need to be in work coordinates.

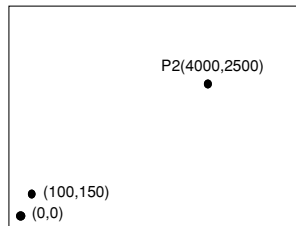


Figure 1-3

If you set P1 only, P2 will automatically be set at the same distance as P1. For example, if you send the following data after executing the above example and set P1 and P2,

```
"IP0,0;"
```

P1 will be set to (0,0) and P2 will be set to (3900,2350).

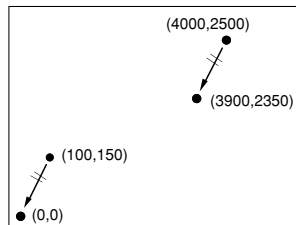


Figure 1-4

- **Scaling using the SC Instruction**

The SC instruction determines the user coordinate values of P1 and P2, after they have been set at the desired points. For example, if you send the following data,

```
"IP100,150,4000,2500;"
```

```
"SC0,1,0,1;"
```

P1, at work coordinates (100,150), is changed to user coordinates (0,0); and P2, at work coordinates (4000,2500), is changed to user coordinates (1,1). (Note that the order of parameters for the SC instructions differs from that with the IP instruction.) Of course, user coordinates are also set outside P1 and P2. The coordinate values for engraving performed thereafter will comply with this user coordinate system.

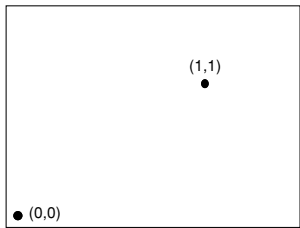


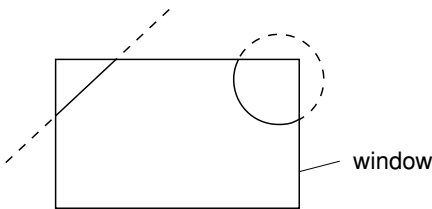
Figure 1-5

In this manner, through combining the IP and SC instructions to set the required coordinates for any desired position, the programming burden can be reduced considerably.

Input P1 & P2 instruction	IP
Scaling instruction	SC

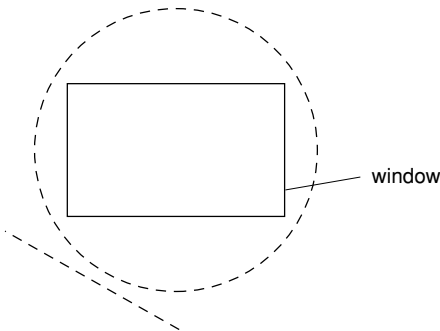
2-2 Window

The engraving area determined by the IW instruction is referred to as a window. Instructions to engrave outside the window do not produce an error, but any portions which extend outside the window will not be engraved. The default value for the window is equivalent to the maximum engraving area.



Portions represented by the dotted lines lie outside the window, and will not be engraved.

Figure 1-6



If an instruction directs that engraving is to be performed completely outside the window, nothing will be engraved.

Figure 1-7

Input window instruction	IW
--------------------------	----

3 TOOL CONTROL Instructions

3-1 Move Tool

If the required engraving is relatively simple, tool instructions alone may be sufficient. A simple example is given in [EXAMPLE 1-1] below, which engraves a straight line from (1000,2000) to (4000,8000).

[EXAMPLE 1-1]

```
10 OPEN "LPT1:" AS #1
20 PRINT #1, "PA;"
30 PRINT #1, "PU1000,2000;"
40 PRINT #1, "PD4000,8000;"
```

(Explanation)

Line 20 Specifies absolute coordinates.
 Line 30 Moves to (1000,2000) with the tool up.
 Line 40 Moves to (4000,8000) with the tool down. (Straight line engraving)

• Absolute Coordinates vs. Relative Coordinates

There are two ways of specifying coordinates: as absolute coordinates, which express the coordinates as a distance from the origin (0,0); or as relative coordinates, which express the coordinates as the distance from the current tool position. For example, [EXAMPLE 1-1] specifies the use of absolute coordinates on line 20. If relative coordinates were to be used, you would rewrite line 40 as follows:

```
40 PRINT #1, "PR;PD3000,6000;"
```

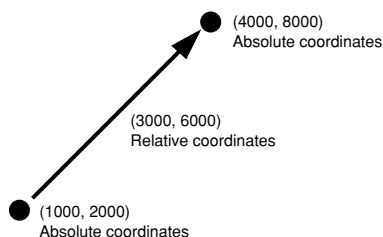


Figure 1-8

Line 40 specifies, by means of the PR instruction, that relative coordinates are to be employed. Thus, subsequent coordinates need to be specified as relative coordinates. The tool position on line 30 is at (1000,2000). This means that to engrave up to absolute coordinates (4000,8000), you would send the relative coordinates (3000,6000), which is the distance from the current tool position. To return relative coordinates to absolute coordinates, the PA instruction must be sent.

The only instructions in mode2 that result in a change between absolute or relative coordinates are PA and PR. Instructions other than PA and PR do not affect the coordinates. The coordinates are absolute when the machine is switched on.

Engrave absolute instruction	PA
Engrave relative instruction	PR
Tool up instruction	PU
Tool down instruction	PD

3-2 Set Tool Function

The velocity select instruction is used to set the tool speed.

Velocity select instruction	VS
-----------------------------	----

4 ENHANCED ENGRAVING Instructions

4-1 Instructions that Enhance Engraving

Although complicated graphics can be engraved using tool control instructions only, mode2 provides numerous instructions that are more convenient to use for complicated graphics. For example, when engraving a rectangle, you could send four engraving line instructions to create the individual sides of the rectangle. But you could also give a single edge rectangle instruction, EA, and obtain the same result. The availability of such instructions makes programming easier.

Some of these instructions perform engraving, whereas others do not actually engrave anything, but establish the various settings needed for engraving. One example is the RA instruction, which is for filling in a rectangle. The filling pattern used by this instruction is set with the FT instruction. This means that you must first specify the pattern with the FT instruction (which makes no cut), and then send the RA instruction to perform the engraving.

Edge rectangle absolute instruction	EA
Edge rectangle relative instruction	ER
Shade rectangle absolute instruction	RA
Shade rectangle relative instruction	RR
Circle instruction	CI
Arc absolute instruction	AA
Arc relative instruction	AR
Edge wedge instruction	EW
Shade wedge instruction	WG
Fill type instruction	FT
Tool diameter instruction	PT
Line type instruction	LT
Tick length instruction	TL
X-tick instruction	XT
Y-tick instruction	YT

5 LABELING Instructions

5-1 Engraving and Making Settings for Labels

The LB instruction is used for label engraving. When you send the LB instruction, the CAMM-2 enters the label mode, and starts engraving the character string that was supplied after the LB instruction.

For example, if you send the following data

```
PRINT #1, "LBABCDabcd" ; CHR$( &H03 )
```

the CAMM-2 starts engraving "ABCDabcd" from the current tool position. The ETX (03₍₁₆₎) above is the label terminator, and is supplied at the end of the label string to clear the label mode. If you omit the label terminator, the CAMM-2 interprets subsequent instructions as part of the label string. To change the label terminator, use the DT instruction.

The character sets used by the LB instruction are those specified by the CS, SS, CA and SA instructions. The codes of engravable characters range from 21₍₁₆₎ through 7E₍₁₆₎. Otherwise, among the characters that cannot be engraved (the control characters from 00₍₁₆₎ through 20₍₁₆₎) the following have a special role in engraving. All other control characters are ignored.

ETX (03 ₍₁₆₎)	: terminates label mode
BS (08 ₍₁₆₎)	: moves one full character to the left
HT (09 ₍₁₆₎)	: moves half of a character to the left
LF (0A ₍₁₆₎)	: goes down one line
VT (0B ₍₁₆₎)	: goes up one line
CR (0D ₍₁₆₎)	: moves to start of line
SO (0E ₍₁₆₎)	: selects alternate character set
SI (0F ₍₁₆₎)	: selects standard character set

You can also set the character size (SI, SR), engraving direction (DI, DR), character slant (SL), etc. with respect to characters that are to be engraved.

Label instruction	LB
Absolute character size instruction	SI
Relative character size instruction	SR
Absolute direction instruction	DI
Relative direction instruction	DR
Character slant instruction	SL
Extra space instruction	ES
User defined character instruction	UC
Symbol mode instruction	SM
Define label terminator instruction	DT
Character chord angle instruction	CC
Character engrave instruction	CP

5-2 Character Sets

You can use the character sets offered by the CAMM-2 to engrave text. With different character sets, the same character codes connected used to engrave different characters. This makes it possible to engrave character fonts in languages other than English.

Specify two character sets beforehand as the standard and alternate character sets. Specify the standard set with the CS instruction, and the alternate set with the CA instruction. Specify character set numbers with the parameters that follow the instructions. The SS and SA instructions are used to select character sets you use. The SS instruction (SS;) or SI (0F₍₁₆₎) is used to select the standard set, and the SA instruction (SA;) or SO (0E₍₁₆₎) is used to select the alternate set.

This is summarized in the following schematic diagram.

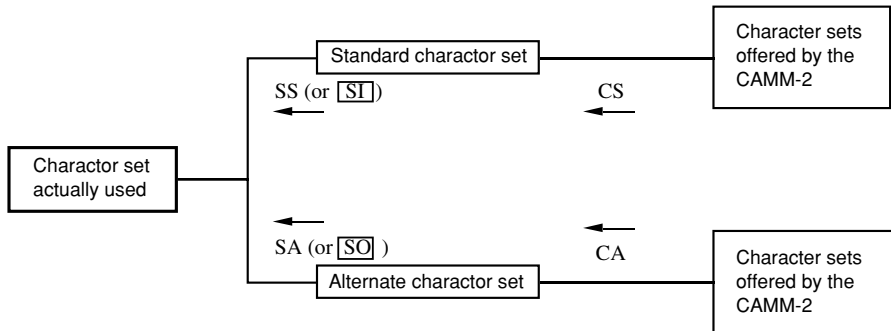


Figure 1-9

This means that if you specify different character sets for the standard and alternate sets at the beginning of a program, you can then change between these character sets using the SS and SA instructions. If you include **[SI]** (0F₍₁₆₎) or **[SO]** (0E₍₁₆₎) in the parameters for an LB instruction, two types of characters can be mixed and engraved using only one LB instruction.

A simple program for such mixed character engraving appears below.

[EXAMPLE 1-2]

```

10 OPEN "LPT1:" AS #1
20 PRINT #1, "CS0;"
30 PRINT #1, "CA7;"
40 PRINT #1, "LB";CHR$(&H0F);"ABC";CHR$(&H0E);"ABC";CHR$(&H03)
50 END
  
```

(Explanation)

- Line 20 Specifies character set 0 as the standard set.
- Line 30 Specifies character set 7 as the alternate set.
- Line 40 Engraves "ABC" in both standard and alternate sets.

In the alternate character set, "ABC" correspond to"êôû". Therefore the CAMM-2 engraves as follows:

ABCêôû

Standard character set instruction	CS
Select standard set instruction	SS
Alternate character set instruction	CA
Select alternate set instruction	SA

1.3.6 Fonts for Character Sets

One or all of the following fonts are available for each of the character sets offered by the CAMM-2.

- **Fixed-space vector font**
- **Variable-space arc font**
- **Fixed-space arc font**

Fixed-space font:

Character widths of all characters are equal. It is convenient since the length of a character string can be precalculated.

Variable-space font:

Character widths are variable to balance the spacing of a character string for a better look when engraved. This is like proportional pitch font.

Vector font:

The curved portions of a character are expressed in continuation with many short straight lines. The shape of a character may be too simple for some purposes, but this is optimal for high speed engraving.

Arc font:

The curved portions of a character are expressed with smooth curve lines. Although the engraving speed is relatively slower than that of the vector font, much better looking characters are engraved.

< Example for Fixed-space vector font >

1234567890
ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrstuvwxyz

< Example for Variable-space arc font >

1234567890
ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrstuvwxyz

< Example for Fixed-space arc font >

1234567890
ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrstuvwxyz

6 OUTPUT Instructions

When an instruction requesting output is sent, the CAMM-2 will ready the data for output to the computer. Such data includes the coordinate values of the current tool position, the tool up/down status at that position, error codes, etc.

The output instruction is effective only when the CAMM-2 is connected in serial. It then can output its status data to the computer.

Output instructions are used when such status data from the CAMM-2 is needed for engraving. A sample program is given in [EXAMPLE 1-3] below.

[EXAMPLE 1-3]

```
100 OPEN "COM1:9600,N,8,1" AS #1
110 PRINT #1,"OP;"
120 INPUT #1,P1X,P1Y,P2X,P2Y
130 PRINT P1X,P1Y,P2X,P2Y
140 END
```

(Explanation)

- Line 100 Opens the RS-232C port.
- Line 110 Sends the OP instruction, requesting output of the coordinate values of scaling points P1 and P2.
- Line 120 Reads the coordinate values for each point.
- Line 130 Displays the coordinate values on the computer.

Output actual point instruction	OA
Output commanded position instruction	OC
Output P1 & P2 instruction	OP
Output window instruction	OW
Output hard-clip limits instruction	OH
Output factor instruction	OF
Output identification instruction	OI
Output option parameter instruction	OO
Output error instruction	OE
Output status instruction	OS

1.4 mode1 and mode2 COMMON INSTRUCTIONS

Common instructions for mode1 and mode2 have 3 characters each, and start with an exclamation mark. These instructions are a mode1 format but can also be used with the mode2 format.

The terminator attached to the mode1 and mode2 common instructions differs depending on whether mode1 or mode2 is being used. If used with the mode 1 format the terminators are CR (0D₍₁₆₎) and LF (0A₍₁₆₎). If used with the mode2 format the semicolon mark ; is needed.

The mode1 and mode2 common instructions are listed below.

Spindle Motor Control	!MC
Not Ready	!NR
Set Z1 and Z2	!PZ
Z Axis Rapid Traverse	!VZ

Chapter 2

mode1 Instructions

H

Home

FORMAT

H

PARAMETER

None

EXPLANATION

The H instruction moves the tool to the work coordinate origin (0,0) with tool up. In addition, it releases an error if an error occurs in CAMM-2.

EXAMPLE

```
10 '*** H Instruction ***  
20 OPEN "LPT1:" AS #1  
30 PRINT #1,"H"  
40 END
```

D

Draw

FORMAT

D x1,y1,x2,y2,...,xn,yn

PARAMETERS

xn : Absolute X-axis coordinate

yn : Absolute Y-axis coordinate

RANGE

Varies between model

Varies between model

EXPLANATION

The "D" instruction engraves a straight line from the current tool position to specified coordinates with tool down in the order of (x1, y1) (x2, y2) ... (xn, yn).

All the coordinates are absolute coordinates, and any number of pair coordinates can be specified as (X-axis coordinate, Y-axis coordinate). Any decimal fractions are rounded up or down to the nearest integer, and the + sign can be omitted.

If the parameters are within the range, but out of the engraving area, the tool stops at the boundary of the engraving area and goes to tool up status. In other words the tool only engraves within the effective engraving area. The tool will not carry out engraving outside of the effective area.

An error occurs if no parameters, parameters out of the range, or only one parameter are keyed in. If an odd number of parameters are specified, they are executed one by one from the beginning. However, an error occurs in the last parameter.

EXAMPLE

```
10 '*** D Instruction ***
20 OPEN "LPT1:" AS #1
30 PRINT #1,"M1000,1000"
40 PRINT #1,"D1000,2000,2000,2000,2000,1000,1000,1000"
50 PRINT #1,"H"
60 END
```

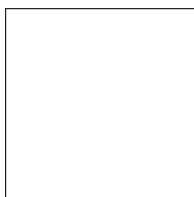


Figure D-1

M**Move****FORMAT**

M x1,y1,x2,y2, . . . ,xn,yn

PARAMETERS

xn : Absolute X-axis coordinate

yn : Absolute Y-axis coordinate

RANGE

Varies between model

Varies between model

EXPLANATION

The “M” instruction moves the tool (with tool up) from the current tool position to specified coordinates in the order of (x1, y1) (x2, y2) ... (xn, yn).

All the coordinates are absolute coordinates. Any decimal fractions are rounded up or down to the nearest integer, and the + sign can be omitted.

When the parameters are within the range, but out of the effective engraving area, the tool does not move. However, when the tool reenters the effective engraving area, it starts moving again.

If only one parameter is keyed in, an error occurs. In addition if an odd number of parameters are specified, they are executed one by one from the beginning. However in the last parameter, if the parameters exceed the allowable range, an error occurs.

EXAMPLE

```
10 '*** M Instruction ***
20 OPEN "LPT1:" AS #1
30 PRINT #1,"M0,1000,1000,1000,1000,0,0,0"
40 PRINT #1,"H"
50 END
```

I

Relative Draw

FORMAT

I $\Delta x1, \Delta y1, \Delta x2, \Delta y2, \dots, \Delta xn, \Delta yn$

PARAMETERS

Δxn : Relative X-axis coordinate

Δyn : Relative Y-axis coordinate

RANGE

Varies between model

Varies between model

EXPLANATION

The "I" instruction engraves a straight line from the current tool position to the coordinates specified by a variation ($\Delta x1$) in X and a variation ($\Delta y1$) in Y. In turn, it starts engraving another straight line up to the coordinates specified by a variation ($\Delta x2, \Delta y2$) in X and Y with respect to the last point at which the former line has been terminated.

All the coordinates are expressed in the value of relative coordinate, and any number of pair coordinates can be specified as (variation in X-axis coordinate, variation in Y-axis coordinate).

Any decimal fractions are rounded up or down to the nearest integer, and the + sign can be omitted. However, an error is generated if the specified coordinate is an absolute coordinate value outside the parameter range. If an odd number of parameters are specified, they are executed one by one from the beginning. However, an error occurs in the last parameter.

EXAMPLE

```
10 '*** I Instruction ***
20 OPEN "LPT1:" AS #1
30 PRINT #1, "M1000,1000"
40 PRINT #1, "I0,1000,1000,0,0,-1000,-1000,0"
50 PRINT #1, "H"
60 END
```

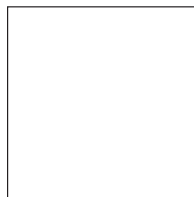


Figure I-1

R**Relative Move****FORMAT**

R $\Delta x1, \Delta y1, \Delta x2, \Delta y2, \dots, \Delta xn, \Delta yn$

PARAMETERS

Δxn : Relative X-axis coordinate Varies between model

Δyn : Relative Y-axis coordinate Varies between model

RANGE**EXPLANATION**

The “R” instruction moves the tool (with tool up) from the current tool position to the coordinates specified by a variation ($\Delta x1$) in X and a variation ($\Delta y1$) in Y. In turn, it starts moving the tool to the coordinates specified by a variation ($\Delta x2, \Delta y2$) in X and Y with respect to the last point at which the former line has been terminated. All the coordinates are relative coordinates.

Any parameters below the decimal point are rounded off, and the + sign can be omitted.

However, an error is generated if the specified coordinate is an absolute coordinate value outside the parameter range.

If an odd number of parameters are specified, they are executed one by one from the beginning. However, an error occurs in the last parameter.

EXAMPLE

```
10 '*** R Instruction ***
20 OPEN "LPT1:" AS #1
30 PRINT #1, "R0,1000,1000,0,0,-1000,-1000,0"
40 PRINT #1, "H"
50 END
```

L

Line Type

FORMAT

L p

PARAMETER

p : line pattern

RANGE

-5 — +5 (Default: 0)

EXPLANATION

The “L” instruction specifies the type of an engraving line.

When the straight line engraving instructions “D” and “T”, the circle engraving instructions “C”, “E” and “G” instructions, the division line engraving instruction “K” and the hatching instruction “T” are used, line type and pitch become those that will be specified by the “L” and “B” instruction, respectively.

Depending on the parameter p, the following 11 types of lines can be selected.

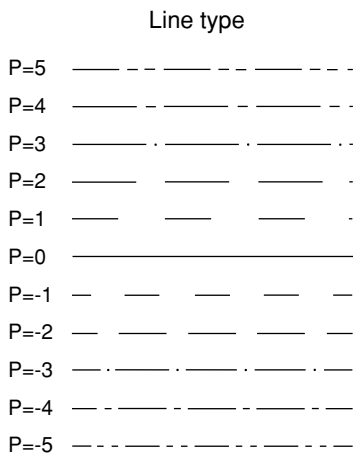


Figure L-1

The startpoint and endpoint of a line vary depending on the positive and negative values of the parameter p.

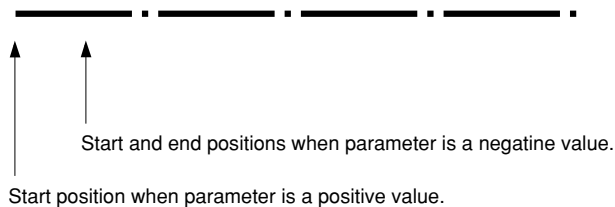


Figure L-2

For example, if you specify a positive value ($p=3$) for a rectangle with an alternate long and short dash line, the start point of that line varies depending on the lengths of lines as illustrated in Example 1. But if you specify a negative value ($p=-3$), you can align the start and end points so they are symmetrically matched to the lengths of lines, as illustrated in Example 2. In this case, the engraving pitch may change slightly.

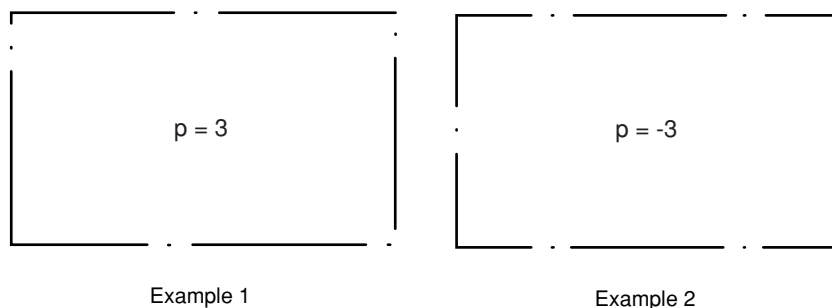


Figure L-3

Any parameter value beyond the -5 — $+5$ range results in error if specified, and the previous value is imposed. The default value ($p=0$) is a solid line.

Once specified using this instruction, the line type remains in effect until you turn off the power or re-specify it with an L or ^LT instruction (mode2).

EXAMPLE

```

100 '*** L INSTRUCTION ***
110 OPEN "LPT1:" AS #1
120 P=-5
130 FOR I=0 TO 10
140   PRINT #1, "M1500, ";800+600*I
150   PRINT #1, "L";P
160   PRINT #1, "S8PP=";P
170   P=P+1
180   PRINT #1, "I5000,0"
190 NEXT I
200 PRINT #1, "H"
210 END

```


B

Line Scale

FORMAT

B l

PARAMETER

l : pitch length

RANGE

Varies between model (Default: 80)

EXPLANATION

The “B” instruction specifies the pitch of broken line, dashed line and two-dot chain line specified by the “L” instruction.

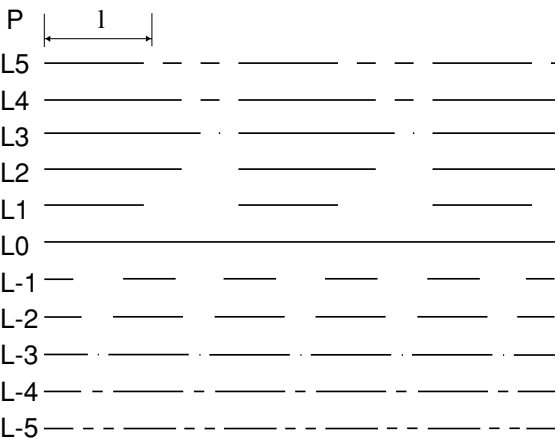


Figure B-1

If a parameter greater than the length of the line to be engraved is specified, CAMM-2 will engrave a solid line.

When a negative parameter is specified with the “L” instruction, the pitch is adjusted so that an integer number of patterns enter the engraving interval. Therefore, the pitch will be slightly varied from the one specified. The pitch that has been specified once using this instruction will remain in effect until it is reset using the “B” or “^LT” instruction (mode2), or the power switch is turned off.

EXAMPLE

```
100 '*** B Instruction ***
110 OPEN "LPT1:" AS #1
120 L=500
130 FOR I=0 TO 4
140   PRINT #1,"M1500,";1500+1000*I
150   PRINT #1,"L2"
160   PRINT #1,"B";L
170   PRINT #1,"S8PL=";L
180   L=L+500
190   PRINT #1,"I5000,0"
200 NEXT I
210 PRINT #1,"H"
220 END
```

L= 2500 _____

L= 2000 _____

L= 1500 _____

L= 1000 _____

L= 500 _ _ _ _ _

Figure B-2

X**Axis****FORMAT**

X p,q,r

PARAMETERS

p : coordinate axis

q : tick interval

r : number

RANGE

0,1

Varies between model

Varies between model

EXPLANATION

The "X" instruction engraves coordinate axes and ticks parallel to X-axis or Y-axis from the current tool position.

If 0 is selected for the parameter p (p=0), Y-coordinate axis is specified. If 1 is selected (p=1), X-axis coordinate is specified. If parameters other than 0 or 1 are specified, an error occurs in this instruction.

The parameter q specifies the spacing of ticks. If a negative value is specified, CAMM-2 engraves coordinate axes in the negative direction (downward from the current tool position for p=1 and to left side from the current tool position for p=0).

The parameter r specifies the number of times the ticks are engraved.

Any decimal fractions for the parameter r are discarded.

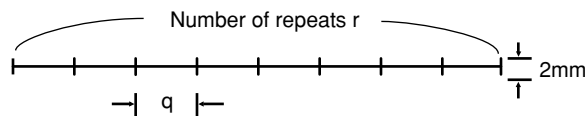


Figure X-1

EXAMPLE

```

100 '*** X Instruction ***
110 OPEN "LPT1:" AS #1
120 FOR N=0 TO 1
130   FOR I=-300 TO 300 STEP 600
140     PRINT #1,"M5000,4000"
150     PRINT #1,"X";N;"",",",I;"10"
160   NEXT I
170 NEXT N
180 PRINT #1,"H"
190 END

```

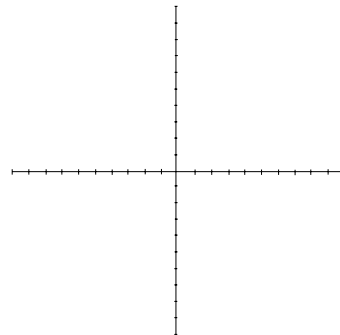


Figure X-2

P**Print****FORMAT**

P clc2...cn

PARAMETER

cn : Character string

EXPLANATION

The “P” instruction engraves a character, symbol or the contents of a variable followed by “P”.

Characters remain in effect until $\boxed{\text{CR}}$ ($0D_{(16)}$) is sent. The size of a character or symbol to be engraved can be changed using the “S” instruction. CAMM-2 engraves from the lower-left corner of a character taken as the current the tool position and moves tool one-full character to the right after execution.

Any part out of the engraving area cannot be engraved. (The part within the engraving area is engraved.)

The characters to be actually engraved are those in the character set that have been selected before the engraving.

EXAMPLE

```
100 '*** P Instruction ***  
110 OPEN "LPT1:" AS #1  
120 PRINT #1, "M1500,3500"  
130 PRINT #1, "S35"; "PP Instruction"  
140 PRINT #1, "H"  
150 END
```

S

Alpha Scale

FORMAT

S n

PARAMETER

n : Character size

RANGE

Varies between model (Default: 3)

EXPLANATION

The "S" instruction sets the size of character specified by the "P" and "N" instructions.

Parameters outside the allowable range are ignored and return to the previously indicated value. The size and spacing depend on the parameters and are as described below.

Character height : $(n + 1) \times 0.8$ mm
 Character width : $(n + 1) \times 0.4$ mm
 Character spacing : $(n + 1) \times 0.08$ mm
 Line spacing : $(n + 1) \times 0.64$ mm

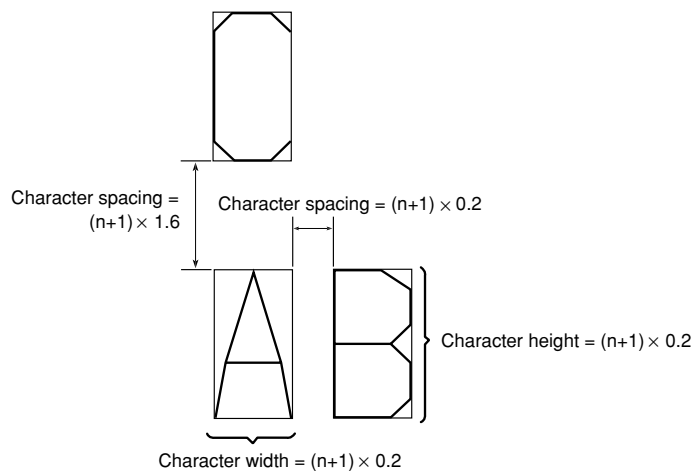


Figure S-1

EXAMPLE

```
100 '*** S Instruction ***
110 OPEN "LPT1:" AS #1
120 PRINT #1,"M700,1500"
130 FOR I=5 TO 25 STEP 5
140   PRINT #1,"S";I
150   PRINT #1,"PS";I
160 NEXT I
170 PRINT #1,"S3"
180 PRINT #1,"H"
190 END
```

s 5 S 10 S 15 S 20 S 25

Figure S-2

Q

Alpha Rotate

FORMAT

Q n

PARAMETER

n : Rotation angle

RANGE

0— 3 (Default: 0)

EXPLANATION

The “Q” instruction specifies the rotation angle of characters by the “P” and “N” instructions. Once set, the rotation angle remain in effect until it is reset again. Any decimal fractions are rounded up or down to the nearest integer. Any parameter out of the range is ignored and reset to the previous setting. Depending on the parameters, the rotation angle is changed as described in Figure Q-1.

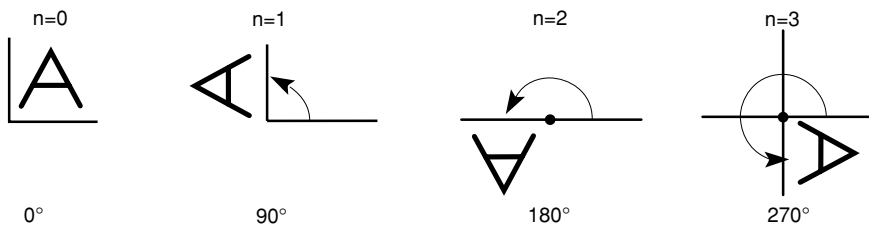


Figure Q-1

EXAMPLE

```

100 '*** Q Instruction ***
110 OPEN "LPT1:" AS #1
120 PRINT #1,"S10"
130 FOR I=0 TO 3
140   PRINT #1,"M5000,4000"
150   PRINT #1,"Q";I
160   PRINT #1,"P  Roland"
170 NEXT I
180 PRINT #1,"Q0"
190 PRINT #1,"H"
200 END
  
```

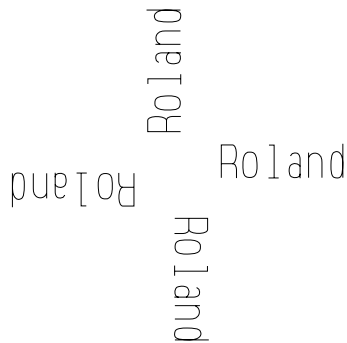


Figure Q-2

N

Mark

FORMAT

N n

PARAMETER

n : Special symbol number

RANGE

1— 15

EXPLANATION

The “N” instruction engraves special symbols described in Table N-1 from the current tool position taken as the center point. After engraving, the tool does not move to the position of a next character as with the “P” instruction. The symbol size can be changed using the “S” instruction.
















n	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Special Symbol															

Table N-1

Any part out of the engraving area cannot be engraved. (The part within the engraving area is engraved.)

EXAMPLE

```
100 '*** N Instruction ***
110 OPEN "LPT1:" AS #1
120 PRINT #1,"S25":PRINT #1,"M500,3000"
130 FOR I=0 TO 15
140   PRINT #1,"N";I:PRINT #1,"R600,0"
150 NEXT I
160 PRINT #1,"S3"
170 PRINT #1,"H"
180 END
```



Figure N-1

C**Circle****FORMAT**

C x,y,r,θ1,θ2(,θd)

PARAMETERS

x, y : Center coordinate
 r : Radius
 θ1 : Start angle
 θ2 : Completion angle
 θd : Resolution

RANGE

Varies between model
 Varies between model
 Varies between model
 Varies between model
 Varies between model (Default: 5°)

EXPLANATION

The “C” instruction engraves an arc determined by the radius r from the start angle θ1 to end angle θ2 which takes user coordinates (x,y) as the center point, using angle resolution θd.

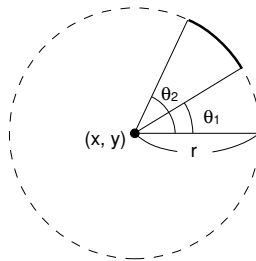


Figure C-1

The parameters x, y and r are expressed in user coordinates and the parameters θ1, θ2 and θd are expressed in degrees (°). They are specified as described in Figure C-1. The + signs are omissible. When θ1<θ2, the engraving direction is counterclockwise.

When θ1>θ2, the engraving direction is clockwise.

Since an arc is engraved with many short straight lines, its angle resolution is expressed in θd. θd is compensated for the value that divides the center angle (θ2-θ1) into even angles.

In addition, polygons can be engraved using θd. For example, if θd is 60° with respect to a circle (360°), a hexagon will be engraved. If θd is 45°, an octagon will be engraved.

EXAMPLE

```
100 '*** C Instruction ***
110 OPEN "LPT1:" AS #1
120 PRINT #1,"C5000,4000,3000,0,360"
130 PRINT #1,"C5000,4000,2000,360,0"
140 PRINT #1,"H"
150 END
```

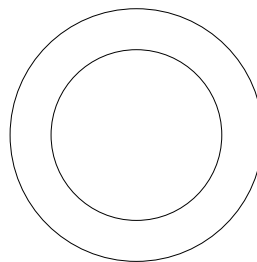


Figure C-2

E**Relative Circle****FORMAT**

E r, θ_1 , θ_2 (, θ_d)

PARAMETERS

r : Radius
 θ_1 : Start angle
 θ_2 : Completion angle
 θ_d : Resolution

RANGE

Varies between model
 Varies between model
 Varies between model
 Varies between model (Default: 5°)

EXPLANATION

Like the “C” and “G” instructions, the “E” instruction engraves a circle or arc.

However, it engraves an arc determined by the radius r in such a manner that the current tool position becomes the start angle θ_1 , up to the end angle θ_2 using the angle resolution θ_d .

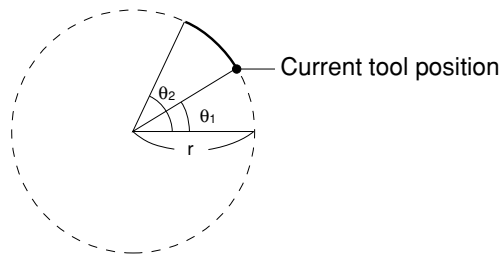


Figure E-1

Since the current tool position becomes the start point of a circle or arc, it is useful when engraving a circle or arc in succession.

The parameter r is expressed in user coordinates, and the parameters θ_1 , θ_2 and θ_d are expressed in degrees (°). They are specified as described in Figure E-1. The + signs can be omitted. When $\theta_1 < \theta_2$, the engraving direction is counterclockwise. When $\theta_1 > \theta_2$, the engraving direction is clockwise.

Since an arc is engraved with many short straight lines, its angle resolution is expressed in θ_d . θ_d is compensated for the value that divides the center angle ($\theta_2 - \theta_1$) into even angles.

In addition, polygons can be engraved using θ_d . For example, if θ_d is 60° with respect to a circle (360° arc), a hexagon will be engraved. If θ_d is 45°, an octagon will be engraved.

EXAMPLE

```
100 '*** E Instruction ***
110 OPEN "LPT1:" AS #1
120 PRINT #1, "M5000,4000"
130 FOR I=0 TO 330 STEP 30
140   PRINT #1, "E1800, "; I; ", "; 360+I
150 NEXT I
160 PRINT #1, "H"
170 END
```

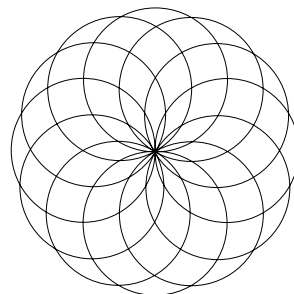


Figure E-2

A**Circle Center****FORMAT**

A x,y

PARAMETERS

x : Center X coordinate

y : Center Y coordinate

RANGE

Varies between model

Varies between model (Default: x=0, y=0)

EXPLANATION

The “A” instruction establishes the center of a circle or arc to be engraved by the “G” and “K” instructions. The center coordinates (x,y) remain in effect until they are reset in user coordinates. The + sign can be omitted. If parameters out of the range are specified, they are ignored, and the previous coordinate values will remain in effect. If coordinates out of the engraving area are specified, no engraving may be performed even when the “G” and “K” instructions are used.

G**A+Circle****FORMAT**

G r, θ_1 , θ_2 (, θ_d)

PARAMETERS

r : Radius

θ_1 : Start angle

θ_2 : Completion angle

θ_d : Resolution

RANGE

Varies between model

Varies between model

Varies between model

Varies between model (Default: 5°)

EXPLANATION

The “G” instruction engraves an arc determined by radius r from the start angle θ_1 to end angle θ_2 , centered at the point specified by the “A” instruction, using the angle resolution θ_d .

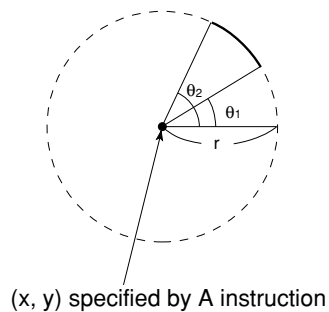


Figure G-1

Remember that at this point, the “A” instruction must have been executed.

If the “A” instruction has not been executed, the “G” instruction engraves circles or arcs based on the user’s reference points.

The parameter r is expressed in user coordinates, and the parameters θ_1 , θ_2 and θ_d are expressed in degrees (°). They are specified as described in Figure G-1. The + signs can be omitted. When $\theta_1 < \theta_2$, the engraving direction is counterclockwise. When $\theta_1 > \theta_2$, the engraving direction is clockwise.

Since an arc is engraved with many short straight lines, its angle resolution is expressed in θ_d . θ_d is compensated for the value that divides the center angle ($\theta_2 - \theta_1$) into even angles.

In addition, polygons can be engraved using θ_d . For example, if θ_d is 60° with respect to a circle (360° arc), a hexagon will be engraved. If θ_d is 45°, an octagon will be engraved.

EXAMPLE

```
100 '*** G Instruction ***
110 OPEN "LPT1:" AS #1
120 PRINT #1, "A5000,4000"
130 PRINT #1, "G2000,0,360"
140 PRINT #1, "H"
150 END
```

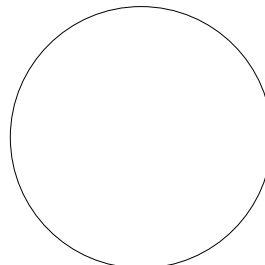


Figure G-2

K**A+%****FORMAT**

K n,l1,l2

PARAMETERS

n : Angle of segment line

l1 : Length from center to end of segment line

l2 : Length from center to beginning of segment line

RANGE

Varies between model

Varies between model

Varies between model

EXPLANATION

The “K” instruction engraves a division line or leader for a circle or arc that has been engraved by the “A” and “G” instructions. Remember that at this point, the “A” instruction must have been executed.

The parameter n can specify the angle of a division line, taking top of a circle (90°) as 0% and one turn of a circle (360°) as 100% . The + signs can be omitted. If the parameter n is positive or negative, n% angle is determined with respect to the clockwise or counterclockwise direction, respectively.

The parameter l1 specifies the distance of a division line’s or leader’s endpoint from the centerpoint of a circle. The parameter l2 specifies the distance of a division line’s or leader’s startpoint from the centerpoint of a circle. That is, the length of a division line or leader is determined by $(l1 - l2)$.

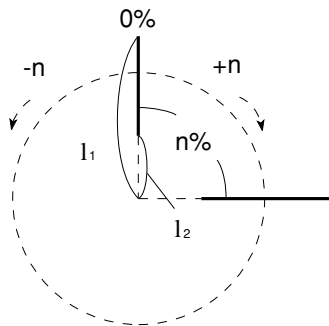


Figure K-1

If $(l1 = \text{circle's radius})$ and $(l2 = 0)$ are specified, a division line is engraved. And if $(l1 > \text{circle's radius})$ and $(0 < l2 < \text{circle's radius})$ are specified, a leader is engraved from the inside to the outside of a circle.

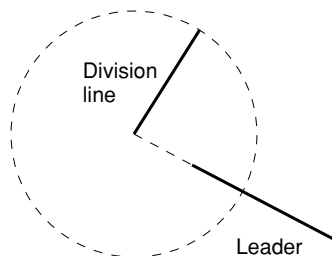


Figure K-2

EXAMPLE

```
100 '*** K Instruction ***
110 OPEN "LPT1:" AS #1
120 PRINT #1,"A5000,4000"
130 PRINT #1,"G2000,0,360"
140 A=0
150 FOR X=1 TO 5
160   READ B
170   A=A+B
180   PRINT #1,"K";A;"",2000,0"
190 NEXT X
200 PRINT #1,"H"
210 DATA 0,35,25,20,15,5
220 END
```

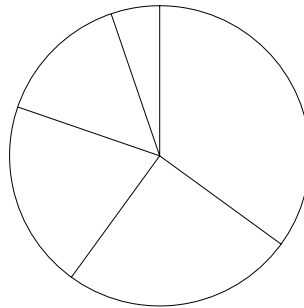


Figure K-3

T

Hatching

FORMAT

T n, x, y, d, t

PARAMETERS

n	: Hatching pattern	RANGE	0 — 3
x, y	: Length of rectangle side		Varies between model
d	: Intervals between hatching lines		Varies between model
t	: Hatching angle		1 — 4

EXPLANATION

The “T” instruction engraves and hatches a rectangle from the current tool position taken as the start point. The parameter n selects a hatching pattern and rectangle type as shown below. Any decimal fractions are rounded up or down to the nearest integer.

n=1 : hatching only
 n=2 : rectangle engraving only
 n=3 : hatching and rectangle engraving
 n=0 : same as n=3

The parameters x, y specify the lengths of a line in X-axis and Y-axis directions.

The parameter d specifies the spacing of hatching. Even in the case of a rectangle only (n=2), this parameter must be used as a dummy.

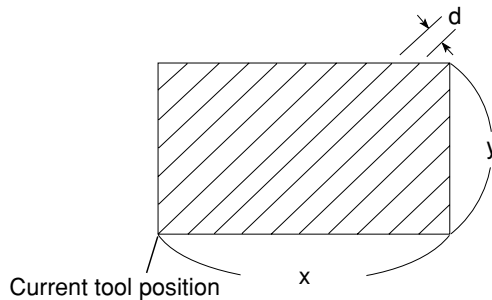


Figure T-1

The parameter t specifies the angle of hatching. Values 1 through 4 correspond to four available hatching angles (0°, 45°, 90°, 135°) respectively. Any values below the decimal point are rounded off. If a parameter out of range is specified, it is ignored if n = 2, but otherwise an error is generated. Even in the case of a rectangle only, this parameter must be used as a dummy.

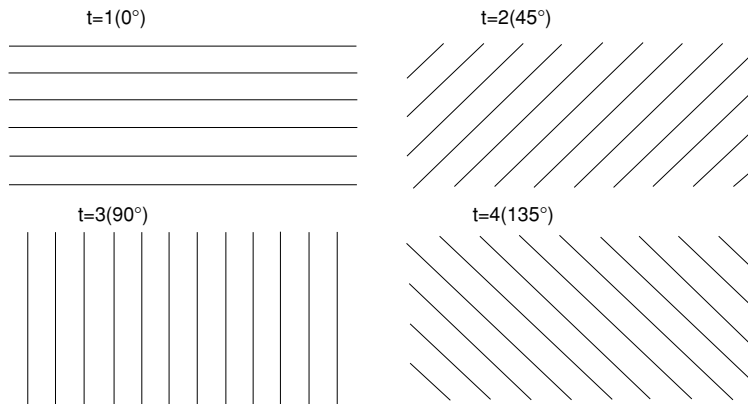


Figure T-2

EXAMPLE

```

100 '*** T Instruction ***
110 OPEN "LPT1:" AS #1
120 PRINT #1,"M1000,1000"
130 PRINT #1,"T3,8000,5657,500,2"
140 PRINT #1,"H"
150 END

```

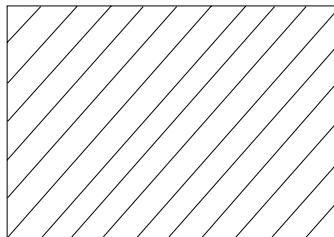


Figure T-3

^

Call mode2

FORMAT

^ [mode2 instruction] [parameter] ..., [parameter] [terminator(;)]

PARAMETER RANGE

[mode2 instruction] parameter range

EXPLANATION

When model has been entered from the control panel, the “^” instruction invokes a mode2 instruction. This is accomplished only by keying in ^ at the head of a mode2 instruction. mode1 and mode2 can be used together in one program.

EXAMPLE

```
100 '*** ^ Instruction ***
110 OPEN "LPT1:" AS #1
120 PRINT #1, "M1000,4000"
130 PRINT #1, "^SL0.4;"
140 PRINT #1, "S50"
150 PRINT #1, "PRoland"
160 PRINT #1, "^SL;"
170 PRINT #1, "S3"
180 PRINT #1, "H"
190 END
```



Figure ^-1

Chapter 3

mode2 Instructions

AA**Arc Absolute****FORMAT**

AA x,y, θ_c (, θ_d) [Terminator]

PARAMETERS

x, y : center coordinates

θ_c : center angle

θ_d : chord tolerance

RANGE

Varies between model

Varies between model

Varies between model (Default: 5°)

EXPLANATION

Taking a point specified by absolute coordinates (x, y) as an arc with the current tool position as the startpoint, only by the angle specified by θ_c using the angle resolution θ_d .

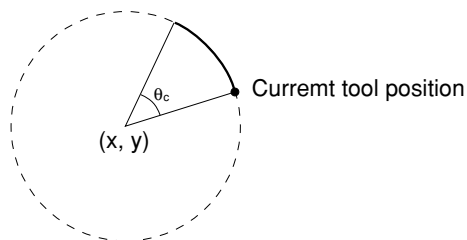


Figure AA-1

CAMM-2 operates in the tool state immediately before the "AA" instruction is sent. (It does not engrave with tool up.)

Since an arc is engraved with many short straight lines, its angle resolution is expressed in θ_d . θ_d is compensated for the value that divides the center angle θ_c into even angles. If 0° is set to θ_d , CAMM-2 will engrave the smoothest arc. If θ_d is omitted, the default value 5° will be set.

In addition, polygons can be engraved using θ_d . For example, if θ_d is 60° with respect to a circle (360° arc), a hexagon will be engraved. If θ_d is 45°, an octagon will be engraved.

EXAMPLE

```
10 \ *** AA Instruction ***
20 OPEN "LPT1:" AS #1
30 PRINT #1,"PA;PU7000,6000;"
40 FOR I=1 TO 3
50   READ A:D=360/A
60   PRINT #1,"PD;AA5000,4000,360,";D;";"
70 NEXT I
80 PRINT #1,"PU100000,100000;"
90 DATA 8,5,3
100 END
```

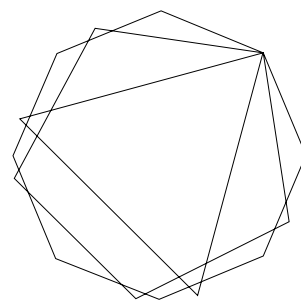


Figure AA-2

AR**Arc Relative****FORMAT**

AR $\Delta x, \Delta y, \theta c$ (, θd) [Terminator]

PARAMETERS

$\Delta x, \Delta y$: relative center coordinates
 θc : center angle
 θd : chord tolerance

RANGE

Varies between model
 Varies between model
 Varies between model (Default: 5°)

EXPLANATION

Taking a point specified by relative coordinates ($\Delta x, \Delta y$) as the centerpoint, the "AR" instruction engraves an arc with the current tool position as the startpoint, only by the angle specified by θc using the angle resolution θd .

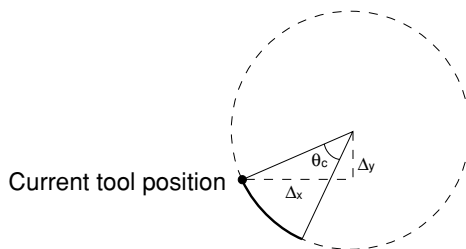


Figure AR-1

CAMM-2 operates in the tool state immediately before the "AA" instruction is sent. (It does not engrave with tool up.)

Since an arc is engraved with many short straight lines, its angle resolution is expressed in θd . θd is compensated for the value that divides the center angle θc in even angles. If 0° is set to θd , CAMM-2 will engrave the smoothest arc. If θd is omitted, the default value 5° will be set.

EXAMPLE

```
10 \ *** AR Instruction ***
20 OPEN "LPT1:" AS #1
30 FOR DY=-3000 TO -1000 STEP 1000
40   PRINT #1,"PA;PU5000,7000;"
50   PRINT #1,"PD;AR0,";DY;"",360,10;"
60 NEXT DY
70 PRINT #1,"PU100000,100000;"
80 END
```

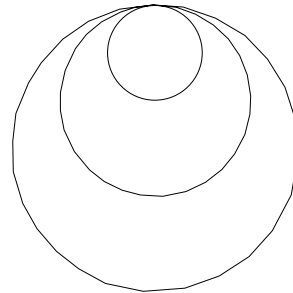


Figure AR-2

CA**Alternate Character Set****FORMAT**

CA n[Terminator]

CA[Terminator]

PARAMETER

n : character set number

RANGE

Varies between model (Default: Varies between model)

EXPLANATION

This instruction designates one of the character sets available with the CAMM-2 as the alternate character set.

The parameter n is a character set number, and any values below the decimal point are rounded off. Any parameters out of range causes an error, and the “CA” instruction is ignored. No parameter is specified, then sets the default value.

To engrave a character set specified by the “CA” instruction, an alternate character set should have been selected by the “SA” instruction or **SO** (0E₍₁₆₎).

CC**Character Cord Angle****FORMAT**CC θ_c [Terminator]

CC [Terminator]

PARAMETER θ_c : center angle**RANGE**

Varies between model (Default: 5°)

EXPLANATION

This specifies the resolution (smoothness) of the CAMM-2's arc font characters.

Arc fonts are express the curved portions of a character with many short straight lines. The "CC" instruction specifies the smoothness of that character by specifying the center angle for the true curved portions corresponding to straight lines. The center angle can be made small, but this lengthens the engraving time. The center angle remains constant, but the lengths of straight lines vary because the radii of each arc also vary.

The maximum value of center angle is 45°. The sign of parameter can be ignored.

EXAMPLE

```

10 \ *** CC Instruction ***
20 OPEN "LPT1:" AS #1
30 PRINT #1,"IN;PA;PU1500,3000;"
40 PRINT #1,"SI3,4.5;CS10;"
50 PRINT #1,"CC45;LBC";CHR$(&H03)
60 PRINT #1,"CC30;LBC";CHR$(&H03)
70 PRINT #1,"CC15;LBC";CHR$(&H03)
80 PRINT #1,"CC5;LBC";CHR$(&H03)
90 PRINT #1,"PU100000,100000;"
100 END

```

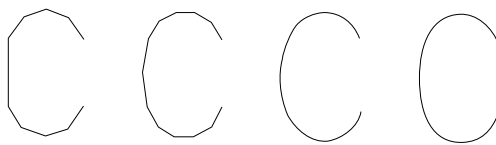


Figure CC-1

CI**Circle****FORMAT**

CI r(,θd) [Terminator]

PARAMETERS

r : radius

θd : chord tolerance

RANGE

Varies between model

Varies between model (Default: 5°)

EXPLANATION

The “CI” instruction engraves a circle determined by the radius r, which takes the current tool position as the centerpoint, counterclockwise using the angle resolution θd.

The parameter r is expressed in user coordinates, and the angle resolution is expressed in degrees (°). The + sign can be omitted.

Since a circle is engraved with many short straight lines, its angle resolution is expressed in θd. θd is compensated for the value that divides the center angle (360°) into even angles. If 0° is set to θd, CAMM-2 will engrave smoothest circle. If θd is omitted, the default value 5° will be set.

In addition, polygons can be engraved using θd. For example, if θd is 60°, a hexagon will be engraved. If θd is 45°, an octagon will be engraved.

EXAMPLE

```
100 ' **** CI Instruction ****
110 OPEN "LPT1:" AS #1
120 PRINT #1,"IN;"
130 PRINT #1,"PA;PU5000,4000;"
140 FOR R=800 TO 2400 STEP 800
150   PRINT #1,"CI";R;";"
160 NEXT R
170 PRINT #1,"PU100000,100000;"
180 END
```

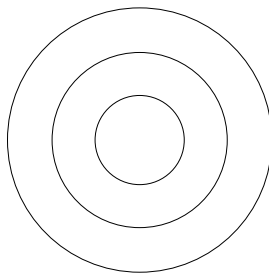


Figure CI-1

CP

Character Engrave

FORMAT

CP nx,ny [Terminator]
CP [Terminator]

PARAMETERS

nx : number of characters in X-axis direction
ny : number of characters in Y-axis direction

RANGE

Varies between model
Varies between model

EXPLANATION

The “CP” instruction moves the tool according to the specified number of characters in the character cell and the current tool (up/down) status.

A character cell is one full-character area, including spacing between characters. For the relation between the character cell and the character size, refer to the paragraphs under “SI” Instruction.

The “CP” instruction moves the tool to the right or upward if the number of characters in parameter is positive, and to the left or downward if the number of characters is negative. However, these directions vary depending on the directions of character engraving as described in Figure CP-1. In addition, even if the parameter is the same, the greater the character size, the greater the tool moving distance.

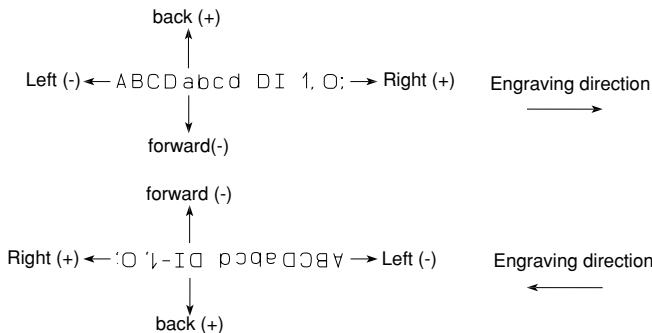


Figure CP-1

A “CP” instruction without parameters moves the tool one full-character down from the tool position set before character engraving. (This is the same as when **CR** (OD₍₁₆₎) and **LF** (OA₍₁₆₎) are executed.)

The “CP” instruction is much more convenient than the “PA” and “PR” instructions in the case of character plotting, because it specifies the moving distance of the tool using the number of characters.

EXAMPLE

```
100 ' *** CP Instruction ****
110 OPEN "LPT1:" AS #1
120 PRINT #1,"IN;PA1500,6500;"
130 PRINT #1,"SI1,1.5;"
140 PRINT #1,"LBCP INSTRUCTION";CHR$(&H03)
150 PRINT #1,"CP-7,-2;PR0,0;"
160 PRINT #1,"LBTHE";CHR$(&H03)
170 PRINT #1,"CP;LBCHARACTER";CHR$(&H03)
180 PRINT #1,"CP;LBENGRAVE";CHR$(&H03)
190 PRINT #1,"CP;LBCPINSTRUCTION,CP";CHR$(&H03)
200 PRINT #1,"PU100000,100000;"
210 END
```

CP INSTRUCTION

THE
CHARACTER
ENGRAVE
CPINSTRUCTION, CP

Figure CP-2

CS

Standard Character Set

FORMAT

CS n[Terminator]

CS[Terminator]

PARAMETER

n : character set number

RANGE

Varies between model (Default: Varies between model)

EXPLANATION

This instruction designates one of the character sets available with the CAMM-2 as the standard character set.

The parameter n is character set number, and any values below the decimal point are rounded off. Any parameters out of the range cause an error and the “CS” instruction is ignored. No parameter is specified, then sets the default value.

To engrave a character set specified by the “CS” instruction, a standard character set should have been selected by the “SS” instruction or SI (0F₍₁₆₎) beforehand.

DF**Default****FORMAT**

DF [Terminator]

PARAMETER

None

EXPLANATION

The “DF” instruction defaults the size, slant and so forth of characters without changing the scaling points P1 and P2, etc.

Function	Equivalent instruction	Content
Coordinate mode	PA;	Absolute coordinates
Character engraving direction	DR;	Horizontal to right
Type of engraving line	LT;	Solid line
1 pitch length of engraving line	LT;	4% of line diagonal to P1,P2
Window	IW;	Work coordinates
Character size	SR;	Width: 0.75% of (P2x-P1x) Height: 1.5% of (P2y-P1y)
Symbol mode	SM;	OFF
Tick length	TL;	X-axis : 0.5% of (P2x-P1x) Y-axis : 0.5% of (P2y-P1y)
Standard character set	CS;	Varies between model
Alternate character set	CA;	Varies between model
Character set	SS;	Standard character set
Character slant	SL;	0°
Scaling	SC;	Not scaled
Tool feed rate	VS;	Varies between model
Label terminator	DT";CHR\$(&H03);";	ETX (CHR\$(&H03))
Resolution of arc font character	CC;	5°
Character and line spacing	ES;	0,0
Error mask value	IM;	223
Hatching pattern	FT;	Bidirectional filling
Hatching spacing	FT;	1% of line diagonal to P1,P2
Hatching angle	FT;	Horizontal
Tool size	PT;	0.75 x 0.4 mm
Z-axis tool feed rate	!VZ;	Varies between model
Tool up/down positions	!PZ;	Varies between model
Spindle motor rotation	!MC;	ON

Table DF-1

DI

Absolute Direction

FORMAT

DI run,rise[Terminator]
DI[Terminator]

PARAMETERS

run : X-axis direction vector
rise : Y-axis direction vector

RANGE

Varies between model (Default: 1)
Varies between model (Default: 0)

EXPLANATION

The character direction that is to be specified by the “DI” instruction is the absolute direction. That is, the direction is determined independently of the scaling points P1 and P2.

The relation between the two parameters (run and rise specifying the engraving direction) and the engraving direction is as described in Figure DI-1.

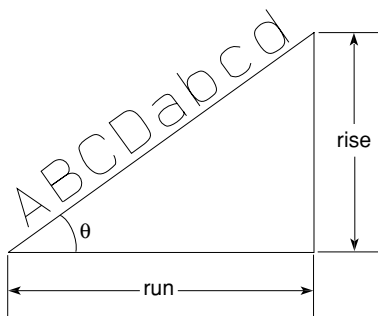


Figure DI-1

	run	rise
0°	1	0
45°	1	1
90°	0	1
135°	-1	1
180°	-1	0
225°	-1	-1
270°	0	-1
315°	1	-1

Table DI-1

When rise=0, a character is engraved in the horizontal direction. When run=0, a character is engraved in the vertical direction. However, if both run and rise are set with 0, an error occurs.

When you already know the angle θ of the engraving direction you desire, you just key in “DI”;cos θ ;;“;sin θ ;;“ and execute.

A “DI” instruction without parameters is defaulted to “DI1,0;” (horizontal direction). If there is only one parameter, error occurs and the instruction is ignored. If three or more parameters are keyed in, an error occurs but the first two parameters will remain in effect.

If the parameters are outside the allowable range or if the run and rise are 0 then error occurs and that instruction is ignored.

The character direction that has been set by the “DI” instruction remains in effect until executing other “DI” and “DR” instructions or defaulting it using the “IN” and “DF” instruction.

EXAMPLE

```

100 ' **** DI Instruction ****
110 OPEN "LPT1:" AS #1
120 PRINT #1,"IN;PA4500,4500;SI0.65,1.3;"
130 PRINT #1,"DI1,1;LB  -- 45deg";CHR$(&H0D);CHR$(&H03)
140 PRINT #1,"DI0,1;LB  -- 90deg";CHR$(&H0D);CHR$(&H03)
150 PRINT #1,"DI-1,1;LB  --135deg";CHR$(&H0D);CHR$(&H03)
160 PRINT #1,"DI-1,0;LB  --180deg";CHR$(&H0D);CHR$(&H03)
170 PRINT #1,"DI-1,-1;LB  --225deg";CHR$(&H0D);CHR$(&H03)
180 PRINT #1,"DI0,-1;LB  --270deg";CHR$(&H0D);CHR$(&H03)
190 PRINT #1,"DI1,-1;LB  --315deg";CHR$(&H0D);CHR$(&H03)
200 PRINT #1,"DI;LB  --DI 0deg";CHR$(&H03)
210 PRINT #1,"PU100000,100000;"
220 END

```

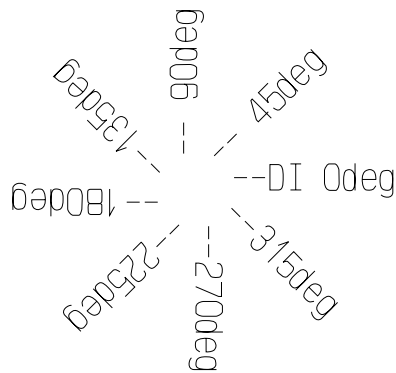


Figure DI-2

DR

Relative Direction

FORMAT

DR run,rise[Terminator]
DR[Terminator]

PARAMETERS

run : X-axis direction vector
rise : Y-axis direction vector

RANGE

Varies between model (Default: 1)
Varies between model (Default: 0)

EXPLANATION

The parameter run is specified in percentage (%) with respect to the X-axis coordinate difference between the scaling points P1 and P2, that is, $(P2x - P1x)$. The parameter rise is specified in percentage (%) with respect to the Y-axis coordinate difference between the scaling points P1 and P2, that is, $(P2y - P1y)$. The relation between run and rise is as described in Figure DR-1. Remember that the engraving direction varies depending on the positions of P1 and P2.

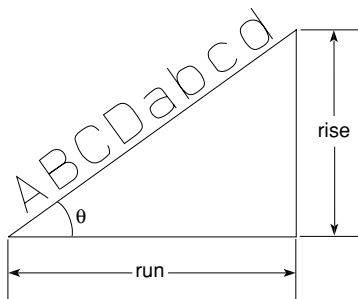


Figure DR-1

rise=0 sets the horizontal direction. run=0 sets the vertical direction. However, if 0 is set to both run and rise, an error occurs.

A “DR” instruction without parameters is defaulted to “DR1,0;” (horizontal direction). However if only one parameter is keyed in an error occurs and the instruction is ignored. If three or more parameters are keyed in an error occurs but the first parameter will remain in effect.

As described in Figure DR-2, parameter signs are determined depending on the directions a character is engraved when the current tool position is determined as the origin.

If the parameters are outside the allowable range or if the run and rise are 0 then an error occurs and that instruction is ignored.

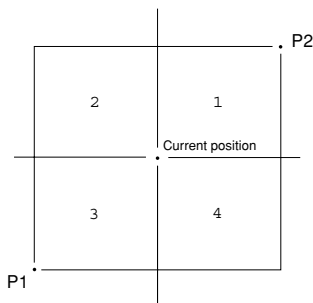


Figure DR-2

	run	rise
1	+	+
2	-	+
3	-	-
4	+	-

Table DR-1

DT**Define Label Terminator****FORMAT**

DT t [Terminator]

DT [Terminator]

PARAMETERt : label terminator (Default: **ETX** (03₍₁₆₎))**EXPLANATION**

If the default label terminator **ETX** (03₍₁₆₎) cannot be used, the “DT” instruction defines another label terminator.

The “DT” instruction is able to define all characters (symbol) other than the null character **NULL** (00₍₁₆₎) in ASCII. However, in the case of ASCII code, the true function of that character is subject to execution. In the case of characters that can be engraved, they are also subjected to engraving. That is, if the “DT” instruction defines **LF** (0A₍₁₆₎) as the label terminator, it will implement line feed (moves one-full line down) as well as release the character engraving mode. And if the “DT” instruction defines **Z** as the label terminator, it will engrave a “Z” at the end of a character string as well as release the character engraving mode. It is therefore, recommended that the user specifies meaningless control characters.

Even when executing a “DT” instruction without parameter, the label terminator does not default to **ETX** (03₍₁₆₎). Since “DT”; defines a symbol “;” as the label terminator, the user should execute “DT”;CHR\$(&H03);” using the “DF” or “IN” instruction when he or she wants to default that label terminator to **ETX** (03₍₁₆₎).

EA**Edge Rectangle Absolute****FORMAT**

EA x,y[Terminator]

PARAMETERS

x, y : absolute XY coordinate of opposite angle of rectangle

RANGE

Varies between model

EXPLANATION

As described in Figure EA-1, taking the current tool position as the start point, the “EA” instruction engraves a rectangle diagonal to the coordinate values of X- and Y-axes.

An “EA” instruction without parameters is ignored.

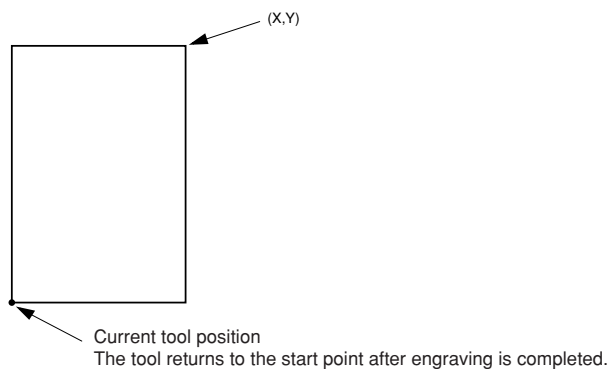


Figure EA-1

The direction and position of rectangle engraving vary depending on the parameter values with respect to the tool position as described Figure EA-2.

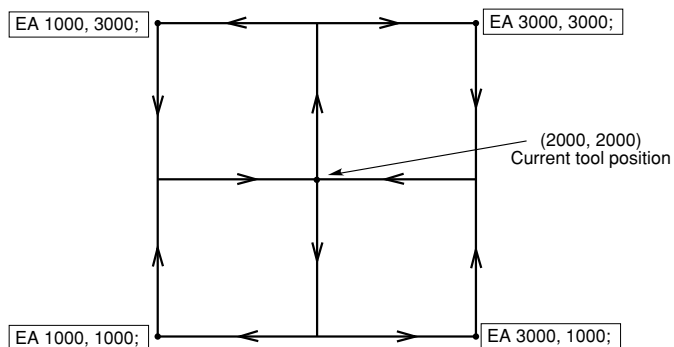


Figure EA-2

ER**Edge Rectangle relative****FORMAT**ER $\Delta x, \Delta y$ [Terminator]**PARAMETERS** $\Delta x, \Delta y$: relative XY coordinate of opposite angle of rectangle**RANGE**

Varies between model

EXPLANATION

As described in Figure ER-1, taking the current tool position as the start point, the “ER” instruction engraves a rectangle diagonal to a point determined by the relative distances of X- and Y-axes directions. After engraving, the tool moves back to the position from which engraving was started.

An “ER” instruction without parameters is ignored.

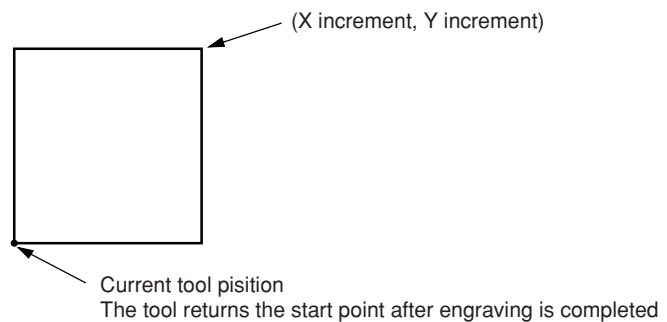
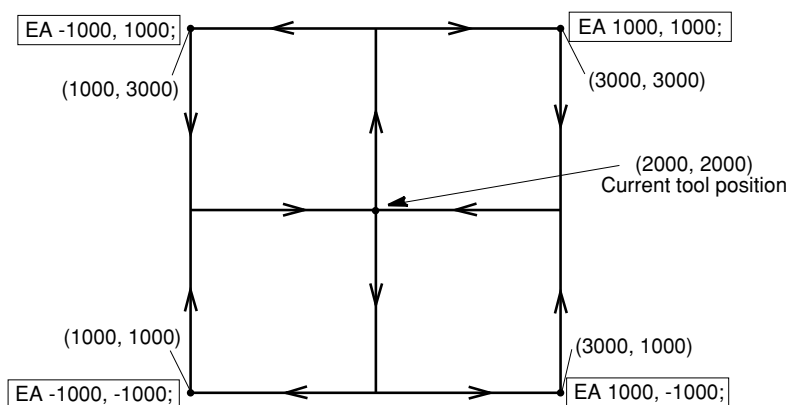


Figure ER-1

The direction and position of rectangle engraving vary depending on the parameter values with respect to the tool position as described below.



* () indicates absolute coordinates

Figure ER-2

ES

Extra Space

FORMAT

ES w(,h) [Terminator]

ES [Terminator]

PARAMETERS

w : character spacing

h : line spacing

RANGE

Varies between model (Default value : 0)

Varies between model (Default value : 0)

EXPLANATION

The “ES” instruction specifies the spacing between characters (character spacing), as well as the spacing between lines (line spacing).

The character spacing refers to gaps between characters that stand side by side (horizontal direction). The line spacing refers to gaps between lines that stand one behind the other (vertical direction). Assuming the width and height of one-full constant width font character is 1, the “ES” instruction specifies character and line spacing by multiplying that size.

If specifying a positive parameter, the character and line spacing gets larger and characters are futher apart. If specifying a negative parameter, on the other hands, characters are closer to each other. When that parameter is -1, the characters are superimposed.

The character and line spacing follows the character size that was determined by the “SI” and “SR” instructions.

EXAMPLE

```
100 ' *** IM Instruction ***
110 OPEN "LPT1:" AS #1
120 PRINT #1,"IN;PA700,6000;SI0.6,0.8;"
130 PRINT #1,"ES;LBES; sets";CHR$(&H03)
140 PRINT #1,"CP;LBthese spaces and lines.";CHR$(&H03)
150 PRINT #1,"PA700,4000;"
160 PRINT #1,"ES-0.2,-0.25;LBES-0.2,-0.25; sets";CHR$(&H03)
170 PRINT #1,"CP;LBthese spaces and lines.";CHR$(&H03)
180 PRINT #1,"PA700,2000;"
190 PRINT #1,"ES0.18,0.25;LBES0.18,0.25; sets";CHR$(&H03)
200 PRINT #1,"CP;LBthese spaces and lines.";CHR$(&H03)
210 PRINT #1,"PU100000,100000;"
220 END
```

ES; sets
these spaces and lines.

ES-0.2,-0.25; sets
these spaces and lines.

ES0.18,0.25; sets
these spaces and lines.

Figure ES-1

EW

Edge Wedge

FORMAT

EW r,θ1,θc(,θd) [Terminator]

PARAMETERS

r : radius
θ1 : start angle
θc : center angle
θd : chord tolerance

RANGE

Varies between model
Varies between model
Varies between model
Varies between model (Default: 5°)

EXPLANATION

The “EW” instruction engraves an arc wedge determined by the radius r of a circle from the start angle θ1 to the center angle θc of that circle using the angle resolution θd.

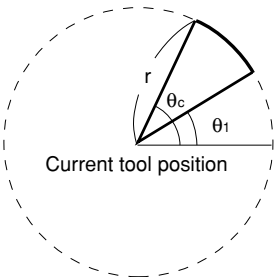


Figure EW-1

The parameter r is expressed in user coordinates, and the parameters θ1, θc and θd are expressed in degrees (%). The + signs can be omitted. An “EW” instruction without parameters is ignored. After engraving an arc wedge, the tool goes back to the position from which engraving was started. Remember that the position of a reference point varies depending on the positive and negative radius values. Also, how the angles from the reference point are determined, varies according to the positive or negative values of the start and center angle parameters. θd (angle resolution) is the parameter that will specify the smoothness of arc portions. This parameter is specified in degrees (°). If specifying 0° to θd, the “EW” instruction engraves the smoothest circle. The engraving direction varies depending on the parameter signs as follows.

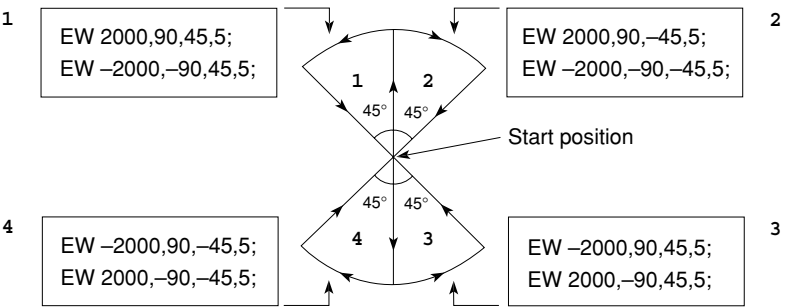


Figure EW-2

EXAMPLE

```
100 ` *** EW Instruction ***
110 OPEN "LPT1:" AS #1
120 PRINT #1,"IN;PA;PU5000,4000;"
130 FOR I=1 TO 4
140   READ A$
150   PRINT #1,"EW";A$
160 NEXT I
170 DATA "-3500,-90,45,5","-3500,-90,-45,5"
180 DATA "3500,-90,-45,5","3500,-90,45,5"
190 PRINT #1,"PU100000,100000;"
200 END
```

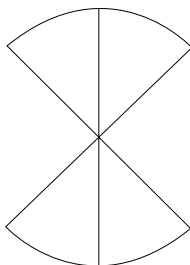


Figure EW-3

FT

Fill Type

FORMAT

FT n(,d(,θ) [Terminator]
FT [Terminator]

PARAMETERS

n : pattern
d : spacing
θ : angle

RANGE

1 — 5 (Default: 1)
Varies between model (Default: $(P2x - P1x) \times 0.01$)
Varies between model (Default: 0°)

EXPLANATION

The “FT” instruction specifies hatching patterns set by the “RA”, “RR” and “WG” instructions.

[n : pattern]

Specifiable patterns are available the following four types. Any parameters below the decimal point are rounded off.

- 1 : bi-directional filling
- 2 : un-directional filling
- 3 : hatching (with parameter d)
- 4 : cross hatching (with parameter d)
- 5 : ignored

1 and 2 are almost the same filling, but 1 engraves back and forth, while 2 engraves in a single direction. Pattern 4 carries out the same hatching as pattern 3, however it has another hatching rotated a further 90 degrees. If no pattern is specified the parameter defaults to 1.

[d: spacing]

The parameter is specified with the parallel spacing of hatching or cross hatching in the units of user coordinates. When n is 1 or 2 (filling), this parameter d is ignored and the value specified by the “PT” instruction will be used.

If not specifying d, the d specified by the “FT” instruction immediately before this is established. If d is not specified at the beginning of a program, it will be defaulted (which is 1% of the distance between the scaling points P1 and P2).

If 0 is set to d, this parameter is ignored, and the value that has been specified by the “PT” instruction at this point will be applied.

[θ : angle]

Hatching can be specified in 45° increments by the line slant at hatching. The specification of 0, 90 and 45 hatches horizontally, vertically and at 45° direction with their respective lines. If there is no angle specification or the specification is of a value other than a multiple of 45, the angle specified by the “FT” instruction immediately before this is applied. And if that value is used at the beginning of a program, 0 angle will be established.

IM

Input Mask

FORMAT

IM e[Terminator]

IM[Terminator]

PARAMETER

e : error mask value

RANGE

0 — 255 (Default: 223)

EXPLANATION

The “IM” instruction determines whether CAMM-2 shows up the error and outputs it to the computer if an error occurs in CAMM-2.

The value of error mask is the sum of error mask bit values (error numbers in the table IM-1) from which the user wants to know. If an error occurs in the bits established, the CAMM-2 shows up the error.

For example the error mask value 223 (=128+64+16+8+4+2+1) will cause an error to show up on the display when an error other than Error Number 6 occurs. Since Error Numbers 4, 7 and 8 have not been used, the value of error mask bits 23 excluding those Numbers is the same as the default value 223.

Error mask	Error No.	Meaning
1	1	Unrecognizable instruction being executed
2	2	Incorrect number of parameters
4	3	Out-of-range parameter being used
8	4	Unused
16	5	Unusable character set being specified
32	6	Coordinates being overflowed
64	7	Unused
128	8	Unused

Table IM-1

The value of error mask bits is set to 223 at default or initialization. In addition, an “IM” instruction without parameter or an “IM” instruction with parameter out of the range defaults the value of error mask bits to 223.

IN

Initialize

FORMAT

IN[Terminator]

PARAMETER

None

EXPLANATION

In addition to implementing the same default as the “DF” instruction does, the “IN” instruction sets as follows.

- (1) moves tool up (PU;).
- (2) initializes the scaling points P1 and P2 (IP;).
- (3) clears an error and sets the 3rd bit of the status byte.

IP

Input P1 & P2

FORMAT

IP P1x,P1y(,P2x,P2y) [Terminator]

IP[Terminator]

PARAMETERS

P1x, P1y : XY coordinate of P1

P2x, P2y : XY coordinate of P2

RANGE

Varies between values

Varies between values

EXPLANATION

The “IP” instructions sets the scaling points P1 and P2.

The coordinates of P1 and P2 are specified in the values of work coordinates. Any values below the decimal point are rounded off. The initial values will not be changed if the work origin is changed. That is, if the work origin is changed, P2 will be outside the engraving area.

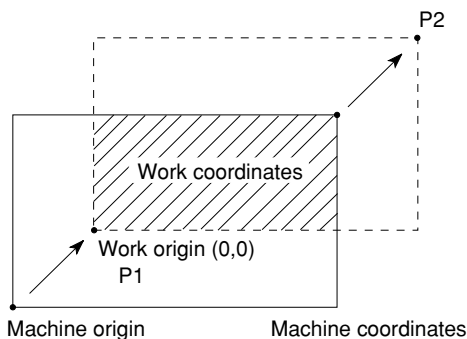


Figure IP-1

P1 and P2 have a role in determining a position that attributes the work coordinates of CAMM-2 to user coordinates with the arbitrary scale the user intends to use. The size of user coordinates is specified by the “SC” instruction.

When P2 is omitted in the “IP” instruction, the position of P2 automatically moves the same distance and in the same direction as P1. That is, the values of $(P2x - P1x)$ and $(P2y - P1y)$ will be equal before and after executing the “IP” instruction in which P2 was omitted.

IW**Input Window****FORMAT**

IW LLx,LLy,URx,URy [Terminator]
 IW [Terminator]

PARAMETERS

LLx, LLy : XY coordinate at lower-left corner of window
 URx, URy : XY coordinate at upper-right corner of window

RANGE

Varies between model
 Varies between model

EXPLANATION

The “IW” instruction sets an engraving area (called a window).

The four parameters are interpreted as work coordinates and become the X- and Y-axes coordinates at the two lower-left and two upper-right corners of a window, respectively. If $LLx > URx$, or if $LLy > URy$, the “IW” instruction always replaces them with each other to make $LLx < URx$ and $LLy < URy$.

The initial values and ranges of the parameters are the maximum engraving area, but they vary depending on the settings of work coordinates. The establishment of parameters beyond this area does not produce an error, and part outside the maximum engraving area will be set.

The “IW” instruction is used to establish another engraving area when the work is smaller than the existing engraving area, or when the user wants to engrave only part of many engravings on a piece of work.

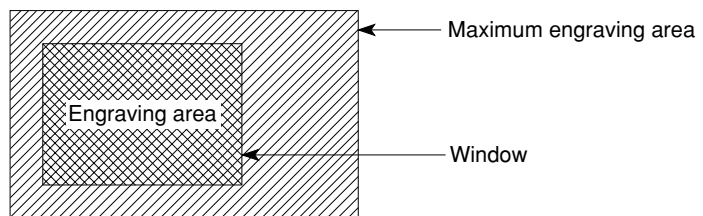


Figure IW-1

LB

Label

FORMAT

LB c1c2...cn [Label terminator]

PARAMETER

cn : character

EXPLANATION

The “LB” instruction enters the character engraving mode and engraves a character string until this character string receives the label terminator defined by the “DT” instruction.

In the case when the “DT” instruction has been omitted, the label terminator is the default terminator ETX ($03_{(16)}$). Remember that if the user forgets to enter the label terminator at the end of a character string when using the “LB” instruction, this instruction interprets all instructions behind that character string, as a character string if any.

With regard to the relation of relative positions between the current tool position and the character to be engraved, the current tool position is at the lower-left corner of the character to be engraved. Therefore, the user should have moved the tool to the desired position using the “PA”, “PR” or “CP” instruction before execution of the “LB” instruction. Any part that extrudes from the engraving area cannot be engraved.

LT

Line Type

FORMAT

LT n(,l) [Terminator]
LT [Terminator]

PARAMETERS

n : pattern number
l : l pitch length

RANGE

-128 — +127 (Default: solid line)
Varies between values (Default: $(P2-P1) \times 0.04$)

EXPLANATION

The “LT” instruction specifies the types of lines to be engraved.
A “LT” instruction without parameters sets solid line. When parameters are entered, the types of lines are determined by pattern numbers as described below.

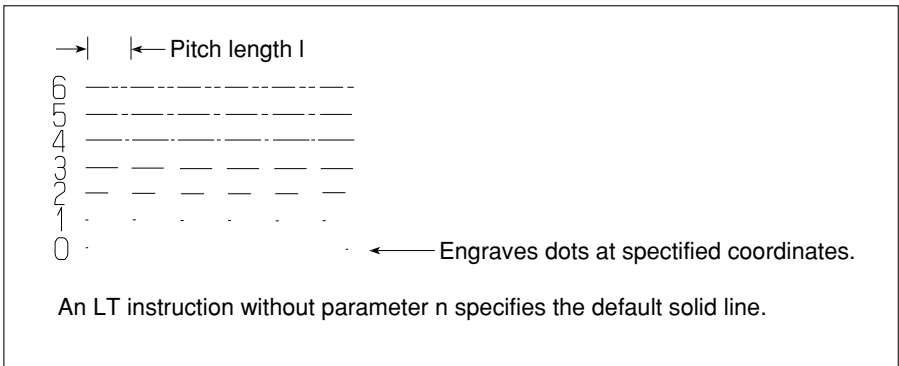


Figure LT-1

Any pattern numbers below decimal point are rounded up and down to the nearest integer. If a pattern number between +7 and +127 is specified, that instruction is ignored. Any pattern number between -128 and -1 sets solid line. Other than these, an error occurs with any other instruction.
So long as a pattern number between +1 and +127 is specified, a parameter for 1 pitch’s length is in effect. This parameter expresses the length of 1 pitch in percentage (%) of the diagonal line to the scaling points P1 and P2. If this parameter is omitted, the default value of 4% is established.

EXAMPLE

```

10 '*** LT Instruction ***
20 OPEN "LPT1:" AS #1
30 PRINT #1,"IN;PA;PU1500,1000;SI0.5,0.8;LBNo parameter";CHR$(&H03)
40 PRINT #1,"LT;PD;PR3000,0;"
50 P=0
60 FOR I=0 TO 6
70     PRINT #1,"PA;PU1500,";1800+800*I;";"
80     PRINT #1,"LT";P;";"
90     PRINT #1,"SI0.7,1.2;LB";P;CHR$(&H03)
100    P=P+1
110    PRINT #1,"PD;PR5000,0;"
120 NEXT I
130 PRINT #1,"PU100000,100000;"
140 END

```

```

6  -----
5  -----
4  -----
3  -----
2  -----
1  -----
0  -----
No parameter_____

```

Figure LT-2

OA**Output Actual Point****FORMAT**

OA[Terminator]

PARAMETER

None

EXPLANATION

Enables output of the work coordinates for the current tool position.

Upon reception of the OA instruction, the following three values are outputted in ASCII code.

X, Y, T [TERM]

X and Y are integer coordinate values in work coordinates, and T denotes the current tool status. Its value will be 0 for tool-up and 1 for tool-down.

[TERM] is the output terminator of RS-232C.

Some models show up the current tool position (work coordinates) and tool state on its display.

EXAMPLE

```
10 ' *** OA Instruction ***
20 OPEN "COM1:9600,N,8,1" AS #1
30 PRINT #1,"OA;"
40 INPUT #1,X,Y,T
50 PRINT X,Y,T
60 END
```

OC**Output Commanded Position****FORMAT**

OC[Terminator]

PARAMETER

None

EXPLANATION

Enables output of the user coordinates for the current tool position.

Upon reception of the OC instruction, the following three values are outputted in ASCII code.

X, Y, T [TERM]

X and Y are the coordinate values in work coordinates or in user coordinates when scaling has been entered. They are values up to the fourth decimal point. T denotes the tool state. It is 0 for tool up and 1 for tool down. [TERM] is the output terminator of RS-232C.

Some models show up the current tool position (user coordinates) and tool state on its display.

EXAMPLE

```
10 ' *** OC Instruction ***
20 OPEN "COM1:9600,N,8,1" AS #1
30 PRINT #1,"OC;"
40 INPUT #1,X,Y,T
50 PRINT X,Y,T
60 END
```

OE**Output Error****FORMAT**

OE [Terminator]

PARAMETER

None

EXPLANATION

Upon reception of the OE instruction, the CAMM-2 outputs of its error codes in the ASCII format shown below. At the same time, it clears any error.

Error code [TERM]

[TERM] is the output terminator of the RS-232C.
Some models show up the error code on its display.

If errors occur, the first error takes priority.
Error codes are defined as in the Table OE-1.

Error Code	Meaning
0	No error
1	Unrecognizable instruction being executed
2	Incorrect number of parameters
3	Out-of-range parameters being used
4	Unused
5	Unusable character set being specified
6	Coordinates being overflowed
7	Unused
8	Unused

Table OE-1

EXAMPLE

```

10 ' *** OE Instruction ***
20 OPEN "COM1:9600,N,8,1" AS #1
30 PRINT #1,"OE;"
40 INPUT #1,E
50 PRINT E
60 END

```

OF**Output Factor**For serial
connection
only**FORMAT**

OF [Terminator]

PARAMETER

None

EXPLANATION

A machine unit value corresponding to each 1 mm is output to the computer. For instance when the output is as shown below,

100,100 [TERM]

In this model it indicates that 100 machine units per 1 mm exist on both the X-axis and Y-axis.

[TERM] is the output terminator of RS-232C.

Some models show up the number of units in the X- and Y-axes directions per 1 mm on its display.

EXAMPLE

```
10 ' *** OF Instruction ***
20 OPEN "COM1:9600,N,8,1" AS #1
30 PRINT #1,"OF;"
40 INPUT #1,X,Y
50 PRINT X,Y
60 END
```

OH**Output Hard-Clip Limits****FORMAT**

OH [Terminator]

PARAMETER

None

EXPLANATION

Upon reception of the OH instruction, the CAMM-2 outputs the integer coordinates for LL and UR in ASCII code as follows:

LLx, LLy, URx, URy [TERM]

(LLx, LLy) are the coordinates at the lower-left corner of the maximum engraving area, and (URx, URy) are the coordinates at the upper-right corner of the maximum engraving area.

[TERM] is the output terminator of RS-232C.

Some models show up the maximum engraving area on its display.

The maximum engraving area varies depending on the changes of work coordinates.

The contents of output by this instruction are not affected by the "IW" instruction.

EXAMPLE

```
10 ' *** OH Instruction ***
20 OPEN "COM1:9600,N,8,1" AS #1
30 PRINT #1, "OH;"
40 INPUT #1, LLX, LLY, URX, URY
50 PRINT LLX, LLY, URX, URY
60 END
```


OI**Output Identification**For serial
connection
only**FORMAT**

OI [Terminator]

PARAMETER

None

EXPLANATION

The OI instruction identifies the model of the unit currently connected to the computer.

Upon reception of the OI instruction, the CAMM-2 outputs the following from its RS-232C connector:

code [TERM]

The "code" is in ASCII code. For instance for the PNC-2000, the code is PNC-2000.

[TERM] is the output terminator of RS-232C.

Some models show up the names of models on its display.

EXAMPLE

```
10 ' *** OI Instruction ***
20 OPEN "COM1:9600,N,8,1" AS #1
30 PRINT #1,"OI;"
40 INPUT #1,N$
50 PRINT N$
60 END
```

OO

Output Option Parameter

**FORMAT**

OO [Terminator]

PARAMETER

None

EXPLANATION

In this instruction, the 8 integers, separated by commas, are output to the computer.

The integer value output varies between models.

The integer value output is called the optional parameter and includes information related to the CAMM-2 hardware structure/CAMM-GL II.

(For more detailed information on the integral value, refer to the instruction manual that corresponds to the model.)

[TERM] is the output terminator of RS-232C.

Some models show up the option parameter on its display.

EXAMPLE

```
10 ' *** OO Instruction ***
20 OPEN "COM1:9600,N,8,1" AS #1
30 PRINT #1, "OO;"
40 INPUT #1, O1, O2, O3, O4, O5, O6, O7, O8
50 PRINT O1, O2, O3, O4, O5, O6, O7, O8
60 END
```

OP**Output P1 & P2**For serial
connection
only**FORMAT**

OP [Terminator]

PARAMETER

None

EXPLANATION

Outputs the coordinates of scaling points P1 and P2.

Upon reception of the OP instruction, the CAMM-2 outputs the following four coordinates in ASCII code.

P1x, P1y, P2x, P2y [TERM]

These coordinate values are integers, in work coordinate units.

[TERM] is the output terminator of RS-232C.

Some models show up the coordinates of the scaling points P1 and P2 on its display.

Once the CAMM-2 has completed the output, bit 1 of the status byte is cleared. (See OS Instruction.)

EXAMPLE

```
10 ' *** OP Instruction ***
20 OPEN "COM1:9600,N,8,1" AS #1
30 PRINT #1, "OP;"
40 INPUT #1, P1X, P1Y, P2X, P2Y
50 PRINT P1X, P1Y, P2X, P2Y
60 END
```

FORMAT

OS [Terminator]

PARAMETER

None

EXPLANATION

Results in output of the value of the status byte, which provides data on the status of the CAMM-2. This instruction is useful for debugging programs. No parameters are necessary.

The CAMM-2 maintains data indicating its internal status. This data consists of eight bits, from 0 through 7, which together form the status byte. Upon reception of the OS instruction, the CAMM-2 converts the status byte's value to decimal, from 0 through 255, then readies the value for output in integers, in the following ASCII form.

Value of status byte in decimal [TERM]

[TERM] is the output terminator of RS-232C.

Each bit of the status byte is defined as follows:

Bit value	Bit No.	Meaning
1	0	Tool down
2	1	Scaling point P1 or P2 was changed (Cleared by OP instruction)
4	2	Unused (always 0)
8	3	Initialized (Cleared by OS instruction)
16	4	Ready to receive data (always 1)
32	5	Error has occurred (Cleared by OE instruction)
64	6	Unused (always 0)
128	7	Unused (always 0)

Table OS-1

When the machine is switched on, the status byte will be 24 in decimal. This is because bits 3 and 4 of the status byte have been set (they have been initialized and made ready to receive data). That is, 24 (=8+16).

After the OS instruction has executed, bit 3 of the status byte is cleared.

Some models show up the value of the status byte on its display.

EXAMPLE

```
10 ' *** OS Instruction ***
20 OPEN "COM1:9600,N,8,1" AS #1
30 PRINT #1,"OS;"
40 INPUT #1,S
50 PRINT S
60 END
```

OW**Output Window****FORMAT**

OW[Terminator]

PARAMETER

None

EXPLANATION

Upon reception of the OW instruction, the CAMM-2 produces the coordinate values for the lower-left and upper-right corners of the window that has been created by the IW instruction, and outputs them in ASCII code.

The coordinate values will be in work coordinates if scaling was not in effect when the IW instruction was executed, and in user coordinates if it was. The order in which these integer coordinate values will be output by the CAMM-2 is as follows:

LLx, LLy, URx, URy [TERM]

[TERM] is the output terminator of RS-232C.

Some models show up the work coordinates at the lower-left and upper-right corners of an engraving area (window) on its display.

EXAMPLE

```
10 ' *** OW Instruction ***
20 OPEN "COM1:9600,N,8,1" AS #1
30 PRINT #1, "OW;"
40 INPUT #1, LLx, LLy, URx, URy
50 PRINT LLx, LLy, URx, URy
60 END
```

PA

Engrave Absolute

FORMAT

PA x1,y1 (,x2,y2, . . . ,xn,yn) [Terminator]
PA [Terminator]

PARAMETERS

xn, yn : absolute XY coordinate

RANGE

Varies between model

EXPLANATION

The “PA” instruction sets absolute coordinates or moves the tool in absolute coordinates.

A “PA” instruction without parameters is to set absolute coordinates. Absolute coordinates are to express a position in the distance from the position of work coordinates (0,0). By this, the parameters of the “PD” or “PU” instruction are interpreted as they are expressed in absolute coordinates.

A “PA” instruction with parameters sets absolute coordinates and moves the tool to the position specified by those parameters. The tool state (up/down) remains the same as that immediately before receiving the “PA” instruction.

This means that the tool only moves without engraving if it is up and engraves if it is down.

The parameters x and y are two in a pair, and any number of parameters can be keyed in. If an odd number of parameters is keyed in, they are interpreted as X- and Y-axes coordinates in order and move the tool. However, an error occurs in the last parameter. (Even if an error occurs, further engraving instructions will be executed correctly.)

EXAMPLE

```
10 `*** PA Instruction ***
20 OPEN "LPT1:" AS #1
30 PRINT #1,"IN;"
40 PRINT #1,"PA1000,2000;"
50 PRINT #1,"PD1000,6000,5000,6000,5000,2000,1000,2000;"
60 PRINT #1,"PU6000,2000;"
70 PRINT #1,"PA;PD6000,6000,9000,6000,9000,2000,6000,2000;"
80 PRINT #1,"PU100000,100000;"
90 END
```

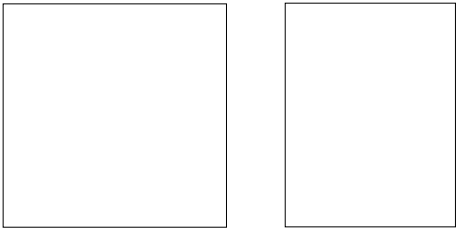


Figure PA-1

PD

Tool Down

FORMAT

PD [Terminator]

PD x1,y1 (,x2,y2, . . . ,xn,yn) [Terminator]

PARAMETERS

xn, yn : XY coordinate

RANGE

Varies between model

EXPLANATION

The “PD” instruction lowers the tool or moves it while it is down.

A “PD” instruction without parameters only moves the tool down. If there are parameters, the tool in the down position moves to a point specified by those parameters. That point varies depending on the absolute and relative coordinates. For example, that point is in absolute coordinates if the “PA” instruction was input and in relative coordinates if the “PR” instruction was input before tool movement.

Parameters are set in absolute coordinates at power-on, or when CAMM-2 is defaulted by the “DF” instruction or initialized by the “IN” instruction. Because of this, if using the “PD” instruction directly without inputting the “PA” and “PR” instructions, the parameters are interpreted as absolute coordinates.

If a point is specified outside a window, the tool moves to the boundary of that window and goes up.

The parameters X and Y are two in a pair, and any number of parameters can be keyed in. If an odd number of parameters is keyed in, they are interpreted as X- and Y-axes coordinates and move the tool. However, an error occurs in the last parameter.

PR**Engrave Relative****FORMAT**

PR $\Delta x1, \Delta y1 (, \Delta x2, \Delta y2, \dots, \Delta xn, \Delta yn)$ [Terminator]
 PR [Terminator]

PARAMETERS

$\Delta xn, \Delta yn$: relative XY coordinate

RANGE

Varies between model

EXPLANATION

The “PR” instruction sets relative coordinates or moves the tool in relative coordinates. A “PR” instruction with parameters is to set relative coordinates. Relative coordinates are to express a position in the distance from the current tool position. By this, the parameters of the “PD” and “PU” instructions after this are interpreted as they are expressed in relative coordinates.

A “PR” instruction with parameters sets relative coordinates and moves the tool only as far as specified by those parameters. The tool state (up/down) remains the same as immediately before receiving the “PR” instruction. This means that the tool only moves without engraving if it is up and engraves if it is down.

The parameters Δx and Δy are two in a pair, and any number of parameters can be keyed in. If an odd number of parameters is keyed in, they are interpreted as $\Delta x, \Delta y$ in order and move the tool. However, an error occurs in the last parameter.

Remember that since the state after power-on or execution of the “IN” or “DF” instruction is the same as before executing the “PR” instruction, the user must execute the “PR” instruction when he or she wants to enter the values of X- and Y-axes in the relative coordinate system.

The following EXAMPLE shows a rectangle engraved by the “PR” instruction, which is the same as that shown in EXAMPLE of “PA” Instruction.

Each point denotes $\Delta x, \Delta y$ in the relative coordinate system. Compare them to the relative coordinate values in parenthesis ().

EXAMPLE

```
10 ' *****PR" Instruction *****
20 OPEN "LPT1:" AS #1
30 PRINT #1, "IN;"
40 PRINT #1, "PA1000,2000;PD;PR0,4000,4000,0,0,-4000,-4000,0;"
50 PRINT #1, "PU6000,0;PD0,4000,3000,0,0,-4000,-3000,0;"
60 PRINT #1, "PA;PU100000,100000;"
70 END
```

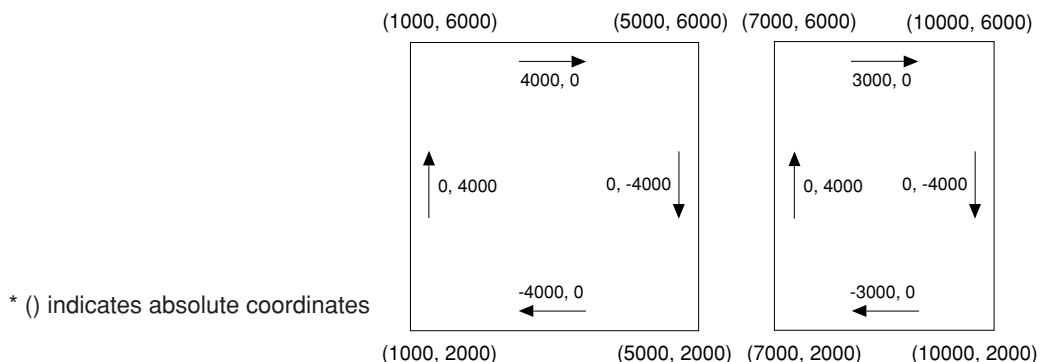


Figure PR-1

PT**Tool Diameter****FORMAT**

PT d[Terminator]

PT[Terminator]

PARAMETER

d : tool width (diameter)

RANGE

0—15 (Default: 0.75)

EXPLANATION

The “PT” instruction specifies the diameter of the tool.

When the hatching spacing (filling) specified by the “FT” instruction depends on the size of the tool, the value specified by the “PT” instruction will be in effect.

The actual tool size (unit : mm) is equivalent to the parameter $\times 0.4$. The default value is 0.75 and so the actual calculation is $0.75 \times 0.4 = 0.3$ mm.

EXAMPLE

```
10 ' **** PT Instruction ****
20 OPEN "LPT1:" AS #1
30 PRINT #1,"FT1,0,45;"
40 PRINT #1,"PT4;"
50 PRINT #1,"PA;PU2000,2000;"
60 PRINT #1,"RA6000,5000;"
70 PRINT #1,"PU100000,100000;"
80 END
```

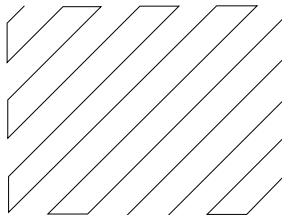


Figure PT-1

PU

Tool Up

FORMAT

PU [Terminator]
PU x1,y1 (,x2,y2, . . . ,xn,yn) [Terminator]

PARAMETERS

xn , yn : XY coordinate

RANGE

Varies between model

EXPLANATION

The “PU” instruction raises the tool or moves it while it is up.
A “PU” instruction without parameters only raises the tool. If there are parameters, the tool moves to the point indicated by those parameters. However, that point varies depending on the absolute and relative coordinates. For example, that point is in absolute coordinates if the “PA” instruction was input and in relative coordinates if the “PR” instruction was input before that tool movement.
Since the parameters of that point are set in absolute coordinates at power-on or when defaulted by the “DF” instruction or initialized by the “IN” instruction, they are interpreted as absolute coordinates if using the “PU” instruction directly without inputting the “PA” and “PR” instructions.
If a point is specified outside a window, the tool moves to the boundary of that window and stops there.
The parameters X and Y are two in a pair, and any number of parameters can be keyed in. If an odd number of parameters is keyed in, they are interpreted as X,Y in order and move the tool. However, an error occurs in the last parameter.

RA**Shade Rectangle Absolute****FORMAT**

RA x,y[Terminator]

PARAMETERS

x, y : absolute XY coordinate of opposite angle of rectangle

RANGE

Varies between model

EXPLANATION

As described in Figure RA-1, taking the current tool position as the start point, the "RA" instruction hatches a rectangle diagonal to the position of X- and Y-axes coordinate specified.

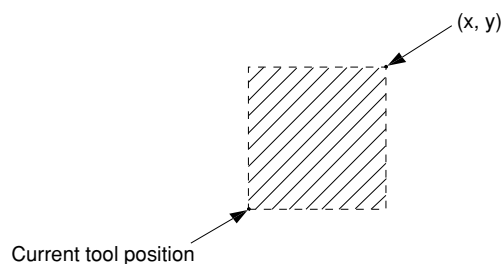


Figure RA-1

This hatching is performed in accordance with the settings (hatching pattern, spacing, angle) of the currently effective "FT" and "PT" instructions.

After completion of the hatching, the tool returns to the startpoint.

EXAMPLE

```
10 '*** RA Instruction ***
20 OPEN "LPT1:" AS #1
30 PRINT #1,"FT3,200,45;"
40 PRINT #1,"PA;PU2000,2000;"
50 PRINT #1,"RA6000,5000;"
60 PRINT #1,"PU100000,100000;"
70 END
```

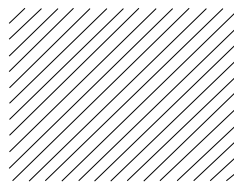


Figure RA-2

RR**Shade Rectangle Relative****FORMAT**RR $\Delta x, \Delta y$ [Terminator]**PARAMETERS** $\Delta x, \Delta y$: relative XY coordinate of opposite angle of rectangle**RANGE**

Varies between model

EXPLANATION

As described in Figure RR-1, taking the current tool position as the start point, the "RR" instruction hatches a rectangle diagonal to the position determined by the relative distances of X- and Y-axes directions specified.

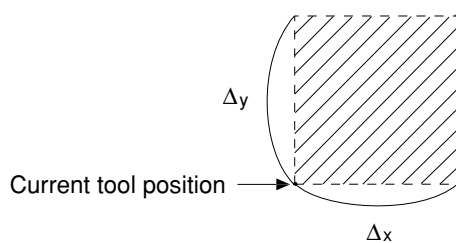


Figure RR-1

This hatching is performed in accordance with the settings (hatching pattern, spacing, angle) of the currently effective "FT" and "PT" instructions.

After completion of the hatching, the tool returns to the startpoint.

EXAMPLE

```

10 '*** RR Instruction ***
20 OPEN "LPT1:" AS #1
30 PRINT #1, "FT4,300,90;"
40 PRINT #1, "PA;PU2000,2000;"
50 PRINT #1, "RA4000,3000;"
60 PRINT #1, "PU100000,100000;"
70 END

```

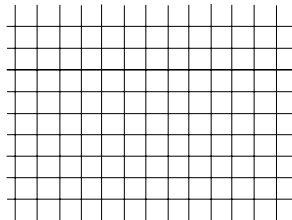


Figure RR-2

SA**Select Alternate Set****FORMAT**

SA [Terminator]

PARAMETER

None

EXPLANATION

The “SA” instruction specifies that CAMM-2 uses the alternate character set specified by the “CA” instruction. This alternate character set remains in effect unless a standard character set is selected by the “SS” instruction or **SI** (0F₍₁₆₎), or that alternate character set is defaulted by the “DF” or “IN” instruction. In character engraving mode, **SO** (0E₍₁₆₎) can be used to select an alternate character set.

SC**Scaling****FORMAT**

SC Xmin,Xmax,Ymin,Ymax[Terminator]
SC[Terminator]

PARAMETERS

Xmin : user X-axis coordinate of P1
Xmax : user X-axis coordinate of P2
Ymin : user Y-axis coordinate of P1
Ymax : user Y-axis coordinate of P2

RANGE

Varies between model
Varies between model
Varies between model
Varies between model

EXPLANATION

The “SC” instruction sets the user coordinates of the scaling points P1 and P2.

The “SC” instruction sets user coordinates so that the position of P1 becomes Xmin, Ymin and the position of P2 becomes Xmax, Ymax. The parameters are real numbers, and an error occurs when Xmax=Xmin or Ymax=Ymin.

User coordinates established in this manner remain in effect until they are scaled again using the “SC” instruction or returned to work coordinates using an “SC” instruction without parameters. All the parameters of an engraving instruction during this period are interpreted as user coordinates.

An “SC” instruction without parameters releases scaling, and the parameters of an engraving instruction after this become work coordinates.

SI**Absolute Character Size****FORMAT**

SI w,h[Terminator]

SI [Terminator]

PARAMETERS

w : character width

h : character height

RANGE

Varies between model (Default: 0.38)

Varies between model (Default: 0.53)

EXPLANATION

The “SI” instruction specifies the size of a character (including width and height).

The actual character size (cm) is equivalent to one in which parameters are multiplied by 0.4.

Figure SI-1 shows the relation between width, height and character cell.

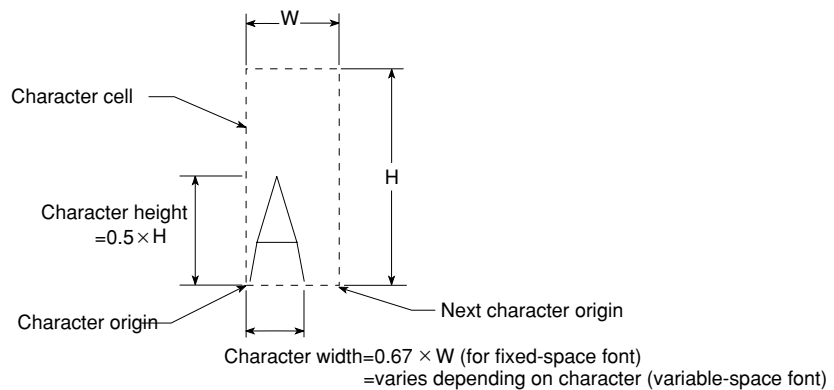


Figure SI-1

Once character size is set, it will not be changed even if the “IP”, “SC” and “IW” instructions are executed. If only one or more than three parameters are keyed in, an error occurs. However, the character size will not be changed.

SL**Character Slant****FORMAT**SL tan θ [Terminator]

SL[Terminator]

PARAMETERtan θ : character slant**RANGE**

Varies between model (Default: 0)

EXPLANATION

The “SL” instruction specifies the slant of characters.

The parameter is specified in the value of tan θ from a perpendicular as described in Figure SL-1. If there are more than two parameters, parameters other than the first are ignored. An “SL” instruction without parameter defaults characters to no slant.

The slant of characters established by the “SL” instruction remains in effect until the “IN”, “DR” or another “SL” instruction is executed.

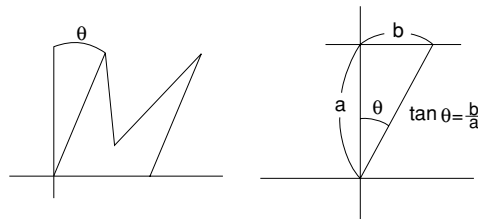


Figure SL-1

SM**Symbol Mode****FORMAT**

SM s[Terminator]

SM[Terminator]

PARAMETER

s : character or symbol

RANGE

Varies between model (Default: Clears symbol mode)

EXPLANATION

The “SM” instruction engraves a specified character or symbol at a point specified by the “PA”, “PR” or “PD” instruction taking that point as the center point.

The parameter must be limited in use for one character or symbol that can be engraved. Once a parameter is specified, it remains in effect until it is specified again or the symbol mode is released. If the “SM” instruction is used to specify another character in symbol mode, it cannot this mode, but only the previous character will be changed to a new one. To release the symbol mode, execute an “SM” instruction without parameter (“SM;”). Remember that in this case, the “IN” and “DF” instructions cannot release the symbol mode.

Note that the character to be engraved is affected by instructions that specify size (SI, SR), slant (SL) and engraving direction (DI, DR).

Semicolons (;) (3B₍₁₆₎) are interpreted as terminator so they cannot be specified as symbols or characters.

EXAMPLE

```

100 \ *** SM Instruction ***
110 OPEN "LPT1:" AS #1
120 PRINT #1,"IN;PA1000,5000;"
130 PRINT #1,"PD1000,500,1000,7500,9000,7500,9000,500,1000,500;PU;"
140 FOR I=1 TO 3
150   READ S$
160   PRINT #1,"S10.7,1.2;SM";S$;" ";
170   X=2000
180   FOR L=1 TO 4
190     READ Y
200     PRINT #1,"PA";X;" ";Y;" ";PD;"
210     X=X+2000
220   NEXT L
230   PRINT #1,"PU;"
240 NEXT I
250 PRINT #1,"SM;PA;PU100000,100000;"
260 END
270 DATA *,1000,2500,6000,6900
280 DATA R,6300,1700,1300,3000
290 DATA G,4800,400,3000,1500

```

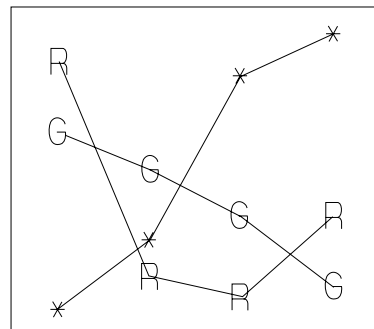


Figure SM-1

SR

Relative Character Size

FORMAT

SR w,h[Terminator]
SR[Terminator]

PARAMETERS

w : character width
h : character height

RANGE

Varies between model (Default: 0.75% of (P2x-P1x))
Varies between model (Default: 1.5% of (P2y-P1y))

EXPLANATION

The “SR” instruction specifies the size (width and height) of a character in proportion to the spacing of scaling points P1 and P2.

The formulas below are used to make specification in percentages using the spacing between the X coordinates and between the Y coordinates for scaling points P1 and P2.

$$\begin{aligned}\text{Character width} &= (P2x - P1x) \times (\text{width parameter } w / 100) \\ \text{Character height} &= (P2y - P1y) \times (\text{height parameter } h / 100)\end{aligned}$$

If only one parameter is keyed in, an error occurs. And the instruction is ignored.
If three or more of parameters are keyed in, an error occurs. However, the first two parameters will remain in effect.

SS**Select Standard Set****FORMAT**

SS [Terminator]

PARAMETER

None

EXPLANATION

The “SS” instruction specifies that CAMM-2 uses the standard character set specified by the “CS” instruction. Once a standard character set is specified, it remains in effect unless an alternate character set is selected using the “SA” instruction or $\boxed{\text{SO}}$ ($0F_{(16)}$), or that standard character set is defaulted using the “DF” or “IN” instruction. In character engraving mode, $\boxed{\text{SI}}$ ($0F_{(16)}$) can be used to select a standard character set.

TL

Tick Length

FORMAT

TL lp(,lm) [Terminator]
TL [Terminator]

PARAMETERS

lp : tick length in positive direction
lm : tick length in negative direction

RANGE

-128 — +128 (%) (Default: 0.5% of (P2y-P1y))
-128 — +128 (%) (Default: 0.5% of (P2x-P1x))

EXPLANATION

The “TL” instruction engraves ticks on coordinate axes.
Ticks in the positive direction are those in the upper direction of X-axis and in the right-hand direction of Y-axis. Ticks in the negative direction are those in the lower direction of X-axis and in the left-hand direction of Y-axis.

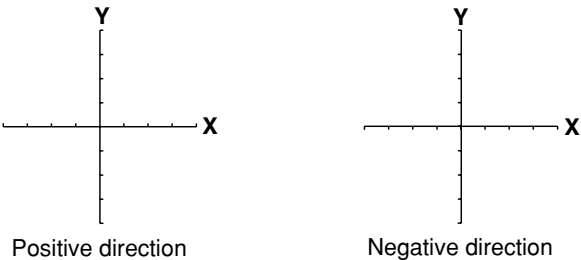


Figure TL-1

In the “XT” instruction, the tick length is the percentage (%) between the Y-axis coordinates of scaling points P1 and P2, that is, (P2y–P1y). In the “YT” instruction, it is the percentage (%) between the X-axis coordinates of scaling points P1 and P2, that is, (P2x–P1x). If the parameter is 0, no ticks will enter. If the parameter is 100, ticks of the lengths in X- and Y-axes directions between P1 and P2 will enter. It would be better to use the parameter 100 for entering separations into a table rather than entering ticks.
The tick length is the ratio of the lengths in X- and Y-axes directions established by P1 and P2. Because of this, remember that if an engraving area set by P1 and P2 is not a square, the tick lengths in X- and Y-axes directions vary from one another.
The tick length will be automatically set to 5% of the lengths in X- and Y-axes directions between P1 and P2 if the parameters are initialized by the “DF” or “IN” instruction, or if a “TL” instruction without parameters is executed. If only the tick length in the positive direction is specified, the tick length in the negative direction is set to 0.
The “TL” instruction remains in effect until another “TL” instruction or the “IN” or “DF” instruction is executed.
In addition, remember that if parameters out of the range are used, an error occurs after execution of the “XT” and “YT” instructions.

UC

User Defined Character

FORMAT

UC (c,) $\Delta x1, \Delta y1, (c,) \dots (\Delta xn, \Delta yn,) \dots$ [Terminator]
UC [Terminator]

PARAMETERS

c : tool control value
 Δxn : units of movement in X-axis direction
 Δyn : units of movement in Y-axis direction

RANGE

Varies between model
 $-99 < \Delta xn < +99$
 $-99 < \Delta yn < +99$

EXPLANATION

The “UC” instruction defines an arbitrary character and engraves. There are three types of parameters: tool control value, number of moving units to X-axis direction and number of moving units in Y-axis direction. The tool control value c is to specify tool up and down; it raises the tool when it is below -99 and lowers the tool when it is over 99 .

To engrave a character, at least one tool down must be specified in the “UC” instruction because the tool always goes up when that instruction is executed. In addition, the tool goes up after completion of the “UC” instruction, moves to the lower-left corner of the next character cell to the right and returns to the up/down state before execution of that instruction.

The number of moving units in X- and Y-axes directions specifies how much the tool moves horizontally or vertically in the number of grids. For this, any number between over -99 and below 99 or decimal fractions can be used.

The grid unit that shows the degree of motion of the tool corresponds to the character cell that is divided into horizontal $6 \times$ vertical 16 . A normal character is engraved in the dotted (horizontal $4 \times$ vertical 8) area as described in Figure UC-1. If a character is defined so that it enters that area when desired, a normal character will be engraved. This means that any size of the character can be defined. However, after engraving is performed by the “UC” instruction, the tool moves one-full character cell from the startpoint of engraving and stops there in the up position, like a normal character.

Therefore, if the character defined is not accommodated in one character cell, the next character will be superimposed on that character unless the tool is moved using the “PA”, “PR” or “CP” instruction.

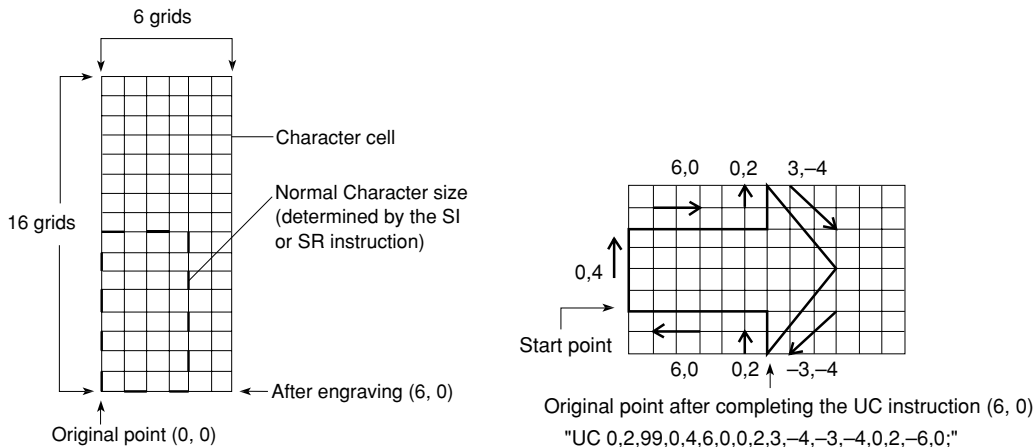


Figure UC-1

EXAMPLE shows that an arrow is engraved, as an example using the “UC” instruction. Since the character origin after execution of the “UC” instruction is (6,0), another character, if engraved without doing anything, will be superimposed on the point of that arrow. To prevent this, “CP1,0;” should be executed to move the character origin one-full character to make (12,0).

In addition, a “UC” instruction without parameters moves the tool to the first character’s position of the “LB” instruction previously executed.

EXAMPLE

```
100 \ *** UC Instruction ***
110 OPEN "LPT1:" AS #1
120 PRINT #1,"IN;PA3000,2500;"
130 PRINT #1,"SI15,9;UC0,2,99,0,4,6,0,0,2,3,-4,-3,-4,0,2,-6,0;CP1,0;"
140 PRINT #1,"PU100000,100000;"
150 END
```

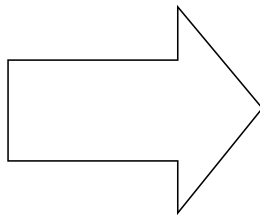


Figure UC-2

VS

Velocity Select

FORMAT

VS v[Terminator]

VS [Terminator]

PARAMETERS

v : tool speed

RANGE

Varies between model (Default: Varies between model)

EXPLANATION

The “VS” instruction specifies the tool feed velocity in X- and Y-axes directions.

A VS instruction without parameter is set to the initial value.

Integers are valid for parameter v.

Z axis velocity is set with the !VZ instruction. Refer to the !VZ instruction items for setting the Z axis velocity.

Some models in the CAMM-2 series allow you to set the feed rate manually. On such models, the feed rate set most recently (either manually or with this instruction) takes priority.

WD**Write to Display****FORMAT**

WD c1c2...cn[Label terminator]

WD[Label terminator]

PARAMETER

cn: characters

EXPLANATION

The “WD” instruction shows up an arbitrary character string up to the label terminator defined by the “DT” instruction on the display.

If the “DT” instruction is omitted, the label terminator is the default ETX ($03_{(16)}$).

The number of characters that can be displayed is limited by the number of lines in the display, and the number of characters per line. Characters appear from the left side of the top line of the display. When the top line is full, display continues on the next line. Characters beyond the maximum number that can be displayed are ignored.

A “WD” instruction without parameter, (“WD”;CHR\$(&H03)), restores the display before that instruction was executed.

EXAMPLE

```
100 ` *** WD Instruction ***
110 OPEN "COM1:9600,N,8,1" AS #1
120 PRINT #1,"IN;PA2000,2000;"
130 PRINT #1,"WD ^ CIRCLE v RECT < TRIANG > *END*";CHR$(&H03)
140 END
```

^ CIRCLE v RECT < TRIANG > *END*

* For a display with 2 lines by 20 characters

Figure WD-1

WG**Shade Wedge****FORMAT**

WG r, θ_1 , θ_c (, θ_d) [Terminator]

PARAMETERS

r : radius

θ_1 : start point

θ_c : center angle

θ_d : chord tolerance

RANGE

Varies between model

Varies between model

Varies between model

Varies between model (Default: 5°)

EXPLANATION

The “WG” instruction hatches an arc wedge from the current tool position taken as the center point.

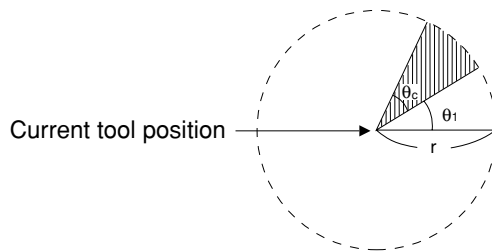


Figure WG-1

Hatching pattern, spacing and angle follow the settings established by the “FT” and “PT” instructions. Remember that the position of a reference point varies depending on the positive and negative values of radius r. In addition, the positive and negative values of start angle and center angle will determine how those angles are assumed from the reference point.

θ_d (angle resolution) is the parameter that specifies the smoothness of an arc portion.

The value of this parameter is given in degrees (°). θ_d is in effect from 0° through 180°. Other than these, θ_d is set to 0° if it is below 0° and to 180° if it is over 180°. In addition, θ_d is compensated for the integer value that divides θ_c into even angles.

After completion of the hatching, the tool returns to the start point.

EXAMPLE

```

100 \ *** WG Instruction ***
110 OPEN "LPT1:" AS #1
120 PRINT #1, "IN;PA;PU5000,4000;"
130 PRINT #1, "FT3,150,0;"
140 PRINT #1, "WG3000,90,45;"
150 PRINT #1, "FT3,150,90;"
160 PRINT #1, "WG-3000,90,-45;"
170 PRINT #1, "FT3,150,135;"
180 PRINT #1, "WG-3000,90,45;"
190 PRINT #1, "PU100000,100000;"
200 END

```

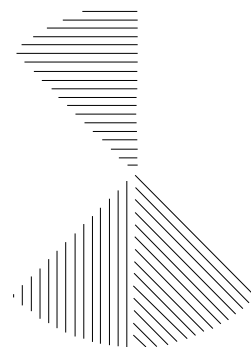


Figure WG-2

XT**X-Tick****FORMAT**

XT [Terminator]

PARAMETER

None

EXPLANATION

The “XT” instruction engraves only one tick of the length specified by the “TL” instruction on X-axis in reference with the current tool position.

The “XT” instruction engraves a tick on X-axis in reference with the current tool position when the tool is either up or down. Although the tick length can be changed using the “TL” instruction, the default value of the tick length is 0.5% of (P2y–P1y) in both positive and negative directions.

EXAMPLE 1

Marks ticks while engraving X axis.

```
100 ' *** XT Instruction No.1 ***
110 OPEN "LPT1:" AS #1
120 PRINT #1,"IN;PA1000,4000;PD;XT;"
130 FOR I=1 TO 10
140   PRINT #1,"PR800,0;XT;"
150 NEXT I
160 PRINT #1,"PU100000,100000;"
170 END
```



Figure XT-1

EXAMPLE 2

Engraves the same thing as [Example 1], except it engraves the X axis first, then engraves the ticks.

```
100 ' *** XT Instruction NO.2 ***
110 OPEN "LPT1:" AS #1
120 PRINT #1,"IN;PA1000,4000;PD;"
130 PRINT #1,"PR8000,0;XT;PU;"
140 FOR I=1 TO 10
150   PRINT #1,"PR-800,0;XT;"
160 NEXT I
170 PRINT #1,"PU100000,100000;"
180 END
```

YT

Y-Tick

FORMAT

YT[Terminator]

PARAMETER

None

EXPLANATION

The “YT” instruction engraves only one tick of the length specified by the “TL” instruction on Y-axis in reference with the current tool position.

The “YT” instruction engraves a tick on Y-axis in reference with the current tool position when the tool is either up or down. Although the tick length can be changed using the “TL” instruction, the default value of the tick length is 0.5% of (P2x–P1x) in the positive and negative directions.

- MEMO -

Chapter 4

Common Use Instructions

!MC

Motor Control

FORMAT

!MC n[Terminator]

!MC[Terminator]

PARAMETER

n: motor ON/OFF switching

RANGE

Varies between model (Default: motor ON)

EXPLANATION

The “!MC” instruction controls the rotation of the spindle motor.

The spindle motor does not activate if the parameter n is other than 0 or omitted.

After this, if an engraving instruction is sent, the spindle motor starts activating. However, as far as n = 0, it does not activate even if an engraving instruction is sent.

Some models in the CAMM-2 series allow manual control of spindle motor rotation. If the rotation of the spindle motor has been switched off manually (by setting the motor switch or DIP switch etc.), the motor will not rotate even when an !MC instruction is given.

!NR**Not Ready****FORMAT****!NR** [Terminator]**PARAMETER**

None

EXPLANATION

Sets the CAMM-2 in pause status.

In this state the CAMM-2 will not operate even if the next instruction is sent. When the pause cancel key is pressed and pause is canceled, engraving will begin with the next instruction following the !NR instruction.

EXAMPLE

```
100 `*** !NR Instruction ***
110 OPEN "LPT1:" AS #1
120 PRINT #1,"!NR;"
130 CLS:CLS:PRINT"If you are ready, press CAMM-2's CONT key."
140 PRINT #1,"PU1000,4000;"
150 PRINT #1,"SI13,5;"
160 PRINT #1,"LB Roland";CHR$(&H03)
170 PRINT #1,"PU100000,100000;"
180 GOTO 120
190 END
```

!PZ**Set Z1 & Z2****FORMAT**

`!PZ z1(,z2) [Terminator]`

PARAMETERS

`z1` : cutting depth (Z1 point)

`z2` : tool up position (Z2 point)

RANGE

Varies between model (Default: none)

Varies between model (Default: none)

EXPLANATION

Sets Z1 (cutting depth) and Z2 (tool up position).

Sets the Z1 and Z2 points in the work coordinate system.

Some models in the CAMM-2 series allow the Z1 and Z2 points to be set manually, but the setting made most recently takes priority.

EXAMPLE

```
10 ' *** !PZ Instruction ***  
20 OPEN "LPT1" AS #1  
30 PRINT #1, "!PZ-10,200;"  
40 END
```


!VZ**Drill Down Velocity Set****FORMAT**

`!VZ vz [Terminator]`

`!VZ [Terminator]`

PARAMETER

vz: feed rate (Z axis)

RANGE

Varies between model (Default: Varies between model)

EXPLANATION

Sets only the Z axis tool feed rate.

Parameter vz is in units of mm/sec. Some models in the CAMM-2 series allow you to manually set the Z-axis feed rate, but the setting made most recently takes priority.

EXAMPLE

```
100 ' *** !VZ Instruction ***
110 OPEN "LPT1" AS #1
120 PRINT #1, "IN; !VZ2;"
130 END
```

- MEMO -

CAMM-GL II

