CAD-to-DMC

User Manual

Manual Rev. 1.0b

By Galil Motion Control, Inc.

THIS PAGE LEFT BLANK INTENTIONALLY

Contents

THIS PAGE LEFT BLANK INTENTIONALLY

Chapter 1 Introduction

1.1 CAD-to-DMC Overview

The CAD-to-DMC software package generates a Galil motion program from a 2-dimensional CAD drawing. This allows you to draw complex motion paths and avoid the time consuming task of programming a list of coordinates.

The software reads in a standard DXF file and converts it into the proper Galil Digital Motion Controller (DMC) commands. The resulting DMC file can then be downloaded to the controller and executed as a motion program or combined with an existing program. The CAD-to-DMC translation process involves five steps:
1. Create a drawing of the motion using AutoCAD or similar drawing package that generates DXF files.
2. Add the necessary CAD-to-DMC commands to the drawing for translation.
3. Create a DXF file of the drawing.
4. Translate the DXF file using CAD-to-DMC.
5. Send the DMC file to the controller and execute.

1.2 Getting Started

1.2.1 System Requirements

Computer: PC operating under Windows
Additional Software:
In order to download and execute the translated DMC files, you will need to use a terminal emulator as provided with the COMMDISK and WSDK software packages.
Motion Control System:
- DMC series controller (2 axes minimum)
- Stepper or servo motors
- Power amplifiers
- Interconnect hardware (i.e. ICM-1100, ICM-1900, etc.)
- DMC User Manual

1.2.2 Installation
CD-ROM
1. Insert "Galil Software Products" disk into the CD-ROM drive.
2. Run CAD->Cadtodmc.exe from the CD-ROM.
3. Enter password for installation. This should have been provided when the software was purchased.
4. The InstallShield Wizard will guide you through the rest of the installation.

1.3 Basic Requirements

Figure 1 shows a sample CAD drawing which satisfies the minimum requirements for a successful CAD-to-DMC file conversion. Every CAD drawing must contain each of the following text directives:

Directive: *Description:*
SCALE=m,n   Relates encoder counts to user units for x and y axes
FIRST   Defines beginning of motion path
LAST   Defines end of motion path

(For full descriptions of these and other commands, please refer to Section 2.4 Command Reference.)
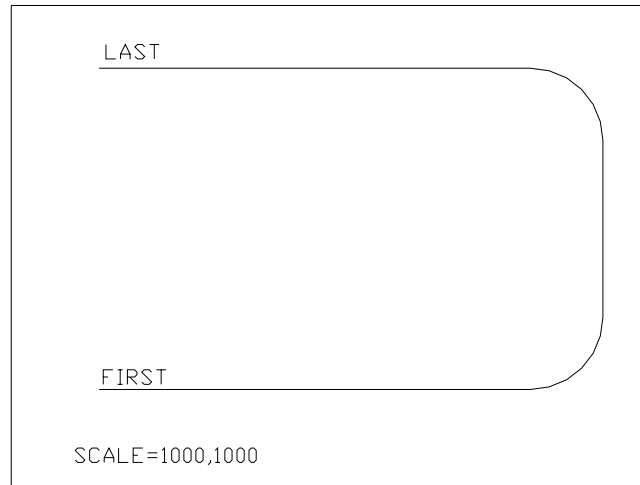

Figure 1: A valid CAD-to-DMC drawing

1.4 A Step-by-Step Example

This section takes you though the entire process of creating a motion control program from a CAD drawing. The example demonstrates the basic requirements of the CAD-to-DMC program, as well as more advanced features such as the use of macros.

Suppose you wish to apply adhesive to a part which is located on a stationary platform. The adhesive applicator moves with respect to the stationary part. The adhesive must be dispensed at a constant feed rate beginning at count coordinates (2000,4000). The dispenser should travel in the counter-clockwise direction around the desired path.

Step 1: Draw the Motion Path
Figure 2 shows the desired adhesive contour drawn in CAD. Dimensions are shown for demonstration purposes only and should not be included in any drawing to be translated.
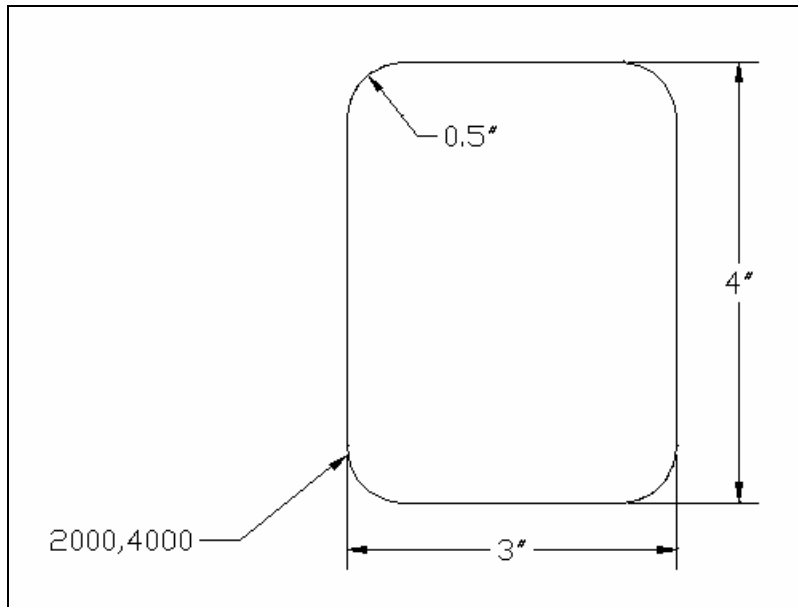
Figure 2: CAD drawing of desired motion path

Step 2: Specify FIRST, LAST and SCALE directives
FIRST and LAST are placed as shown in Figure 3. FIRST is placed adjacent to the point labeled
(2000,4000) in Figure 2. LAST is placed above FIRST and is also associated with the same point (for more
information on association, refer to Section 2.2 Association Principles). This causes the motion to follow a
counter-clockwise path.
SCALE is specified as 1000 encoder counts per inch for each axis.

Step 3: Specify mode of motion
Since a constant feed rate is desired, SMOOTH_MODE should be used. SMOOTH_MODE causes all
motion to be generated as a continuous, constant velocity path. SMOOTH_MODE is the default setting for
CAD-to-DMC and does not need to be explicitly specified in this case.

Step 4: Add macros for system initialization and ON/OFF control of dispenser
FIRST_MACRO is executed before the actual motion path takes place. In this application, the X and Y
axes are homed and then brought to coordinates (2000,4000) for the beginning of dispensing. Finally, a bit
is set which turns on the dispenser. LAST_MACRO is executed after the motion path has been completed.
Figure 3 shows the implementation of these two macros.

MOVING_MACRO OFF and MOVING_MACRO ON are used to turn the dispenser on and off based on
its location in the drawing. Figure 4 shows the use of the MOVING_MACRO's.

An alternate method for controlling the flow of adhesive is shown in Figure 5. LINETYPE macros are used
to associate a linetype with a series of DMC commands. In this example, a solid line becomes associated
with the SB1 command while a dashed line becomes associated with the CB1 command.
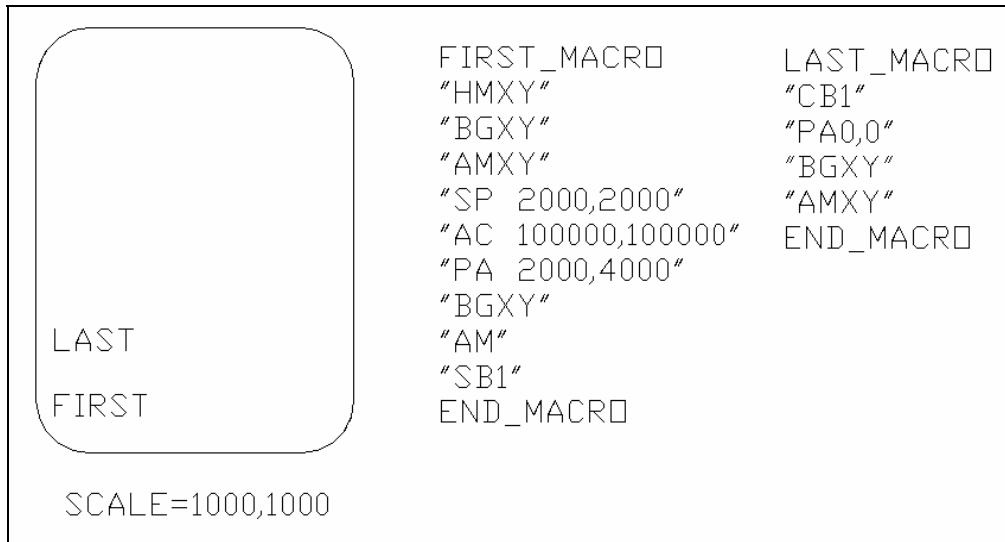
```
                    FIRST_MACRO           LAST_MACRO
                    "HMXY"                "CB1"
                    "BGXY"                "PA0,0"
                    "AMXY"                "BGXY"
                    "SP  2000,2000"       "AMXY"
                    "AC  100000,100000"   END_MACRO
                    "PA  2000,4000"
                    "BGXY"
   LAST             "AM"
                    "SB1"
   FIRST            END_MACRO


  SCALE=1000,1000
```

Figure 3: Placement of FIRST, LAST, SCALE and basic macros

```
     OFF       ON
                       FIRST_MACRO           LAST_MACRO
  ON          OFF      "HMXY"                "CB1"
                       "BGXY"                "PA0,0"
                       "AMXY"                "BGXY"
                       "SP  2000,2000"       "AMXY"
                       "AC  100000,100000"   END_MACRO
                       "PA  2000,4000"
     LAST              "BGXY"                MOVING_MACRO  OFF
                       "AM"                  "CB1"
  OFF                  "SB1"                 END_MACRO
     FIRST      ON     END_MACRO
     ON       OFF                            MOVING_MACRO  ON
                                             "SB1"
                       SCALE=1000,1000       END_MACRO
```
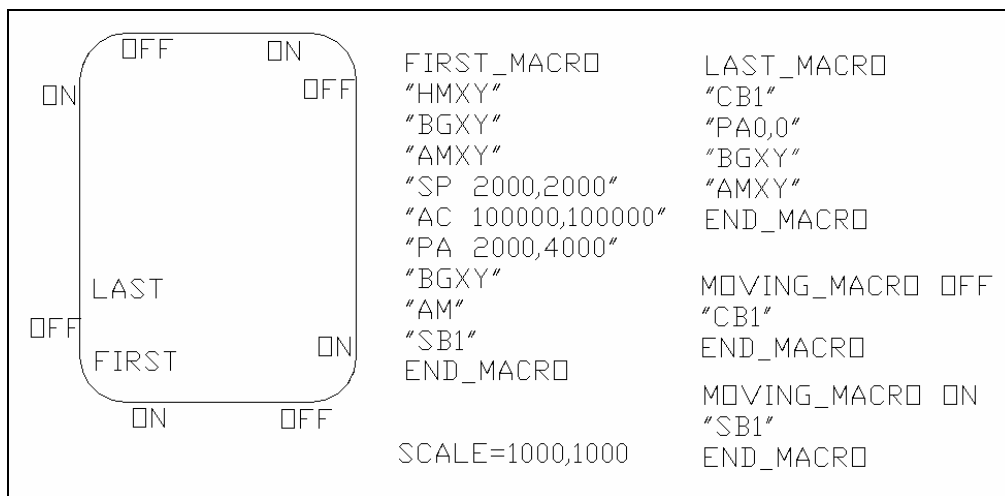
Figure 4: Final CAD drawing with dispenser control

```
        FIRST_MACRO        LAST_MACRO
        "HMXY"             "CB1"
        "BGXY"             "PA0,0"
        "AMXY"             "BGXY"
        "SP 2000,2000"     "AMXY"
        "AC 100000,100000" END_MACRO
        "PA 2000,4000"
LAST    "BGXY"             MOVING_MACRO LINETYPE
        "AM"               _____
        "SB1"              "CB1"
FIRST   END_MACRO          END_MACRO

                           MOVING_MACRO LINETYPE
                           _____
        SCALE=1000,1000    "SB1"
                           END_MACRO
```
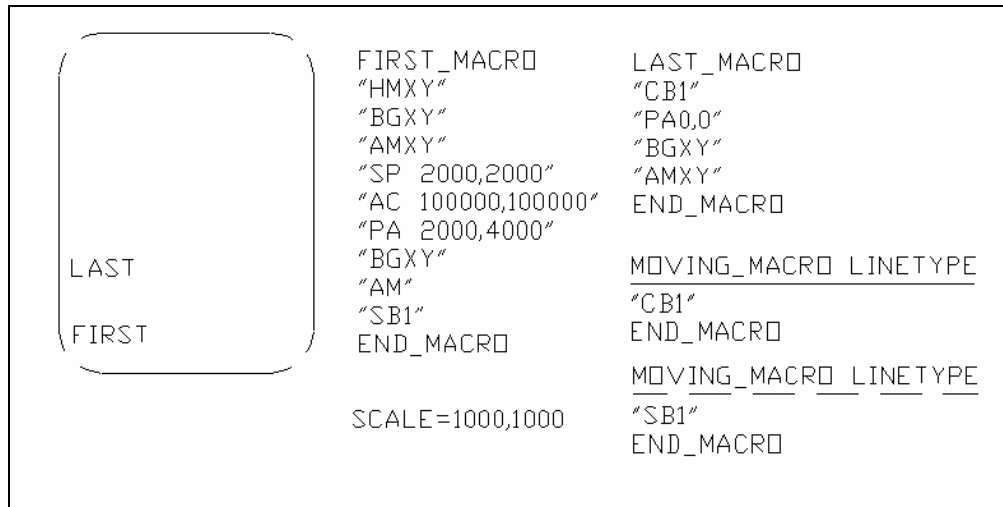
Figure 5: Dispenser control using LINETYPE macros

Step 5: Create a DXF file
With the drawing completed, export a DXF file using the appropriate commands for the CAD package you are using. With the DXF file saved, you can now exit the CAD program and open the CAD-to-DMC program.

Step 6: Translating from DXF to DMC
Select File->Open from within the CAD-to-DMC program. This will open a window similar to the one shown in Figure 6. Select the file you wish to open and click OK. If the CAD file was created correctly, the program will automatically translate it into DMC and create a new file of the same name as the DXF file with the SEN extension. In addition, the program will display a view of the complete motion profile as shown in Figure 7. Notice that an entry path (the diagonal line) shows up in the translated drawing even though it wasn't explicitly drawn in CAD. This is because the starting point for dispensing was specified to be 2000,4000.

The translated DMC file can be viewed by selecting View->DMC File and entering the appropriate file name. A portion of the translated code is shown in Figure 8.
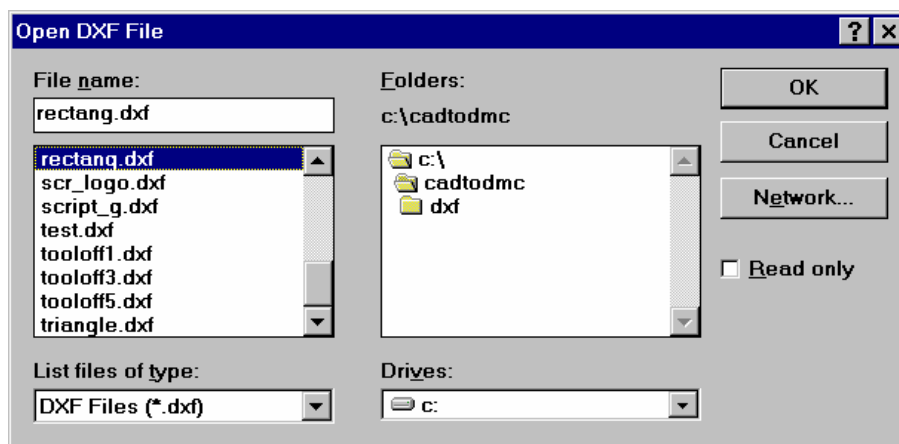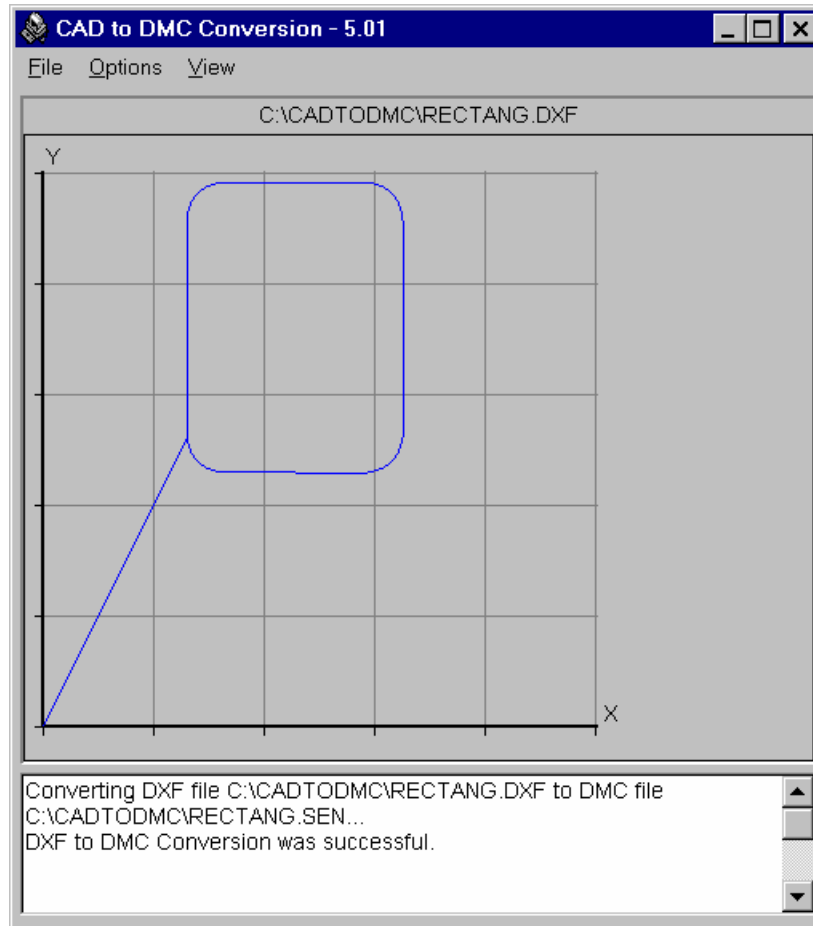

Figure 6: Opening a DXF file for translation

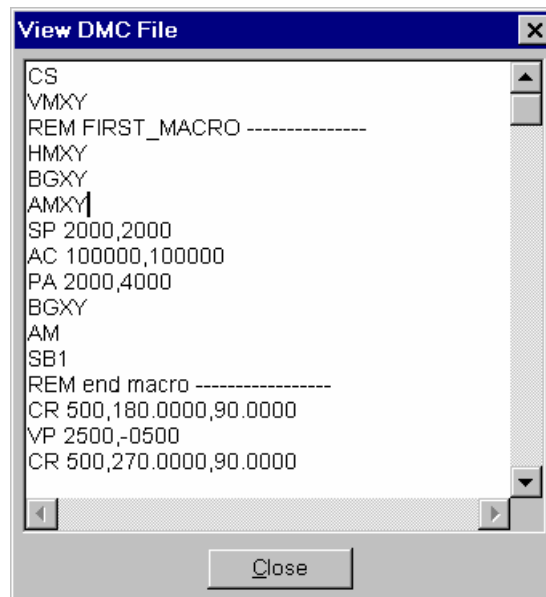**Figure 7:** Actual motion profile generated by CAD-to-DMC

Figure 8: Translated DMC code

Step 7: Download translated file and generate motion
After the DMC file has been created, you must download it to the controller and execute it. The simplest way to do this is to select File->Download File from the Galil terminal and select the appropriate file name. To execute the program, type "XQ" <Enter> at the terminal command line.

THIS PAGE LEFT BLANK INTENTIONALLY

Chapter 2 Reference Section

2.1 Geometry Requirements

All CAD-to-DMC motion is described as a sequence of arcs and lines performed with respect to the current controller location. Motions can only start and stop at the endpoints of these geometry elements. This precludes the use of circles and closed polygons, which have no endpoints. To get around this problem, break circles into at least two arcs (more if there are multiple start/stop points) and explode all polygons. Another geometry type which may cause problems is the ellipse. By default, ellipses are translated by CAD-to-DMC as a series of line segments. Depending on the application, the resulting profile may be undesirable. Exploding the ellipse will cause CAD-to-DMC to translate it as a series of arcs, creating a smoother motion path.

2.2 Association Principles

What is an association?
Associations, sometimes referred to as parent associations, are a means of relating drawing elements to one another. These relationships can be used to organize the drawing's information. Figure 9 shows some examples of associations. Associations are generally created by placing one element near another element. The elements need not be of the same type. For example, a text element can be associated with a line element. In fact, the CAD-to-DMC translator relies heavily on this type of association.
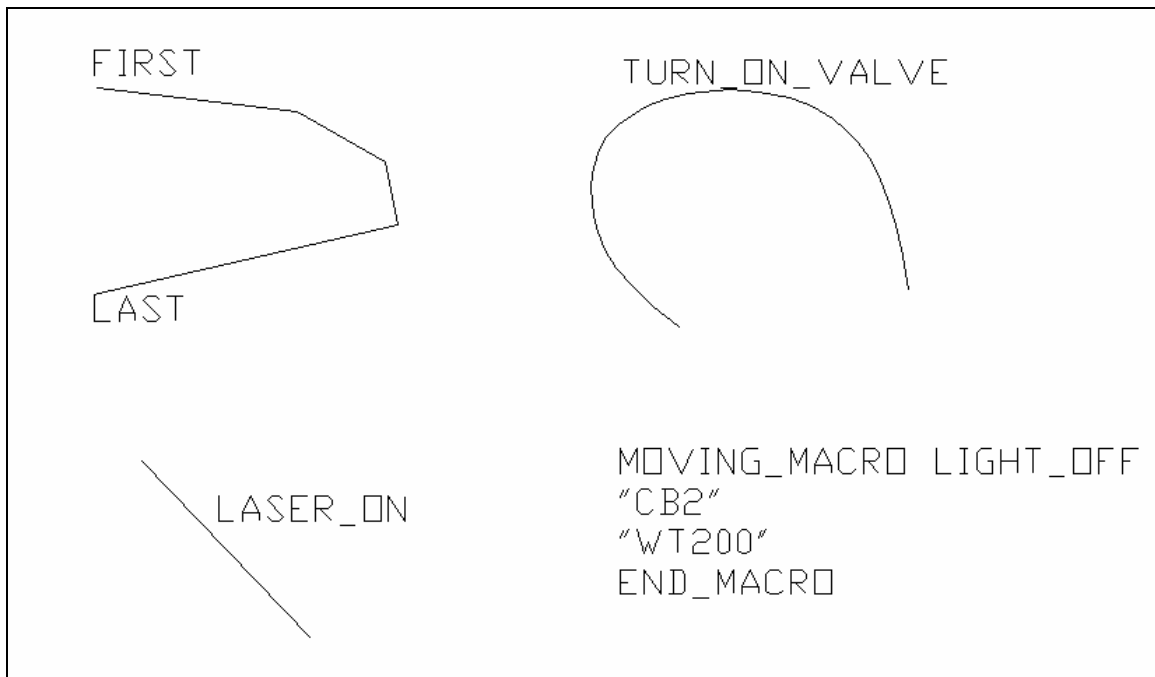

Figure 9: Association examples

Association Rules
Nearness
The position of a text element is the position of the lower left corner of the text when viewed as read. A text element becomes associated with the nearest geometry element within one text height of the text. The

---

distance between the two elements is measured differently for each geometric element type, as described in the following sections. An element which is near nothing is considered an "orphan" and will cause CAD-to-DMC to generate a warning message.

Lines

The distance between a text element and a line element is the minimum distance from the line to the text. Normally, this is the perpendicular distance between the line and the text (Figure 10, Case 1). However, if a perpendicular line cannot be drawn between the line and the location point of the text, the distance is measured from the text to the nearest endpoint of the line (Figure 10, Case 2).
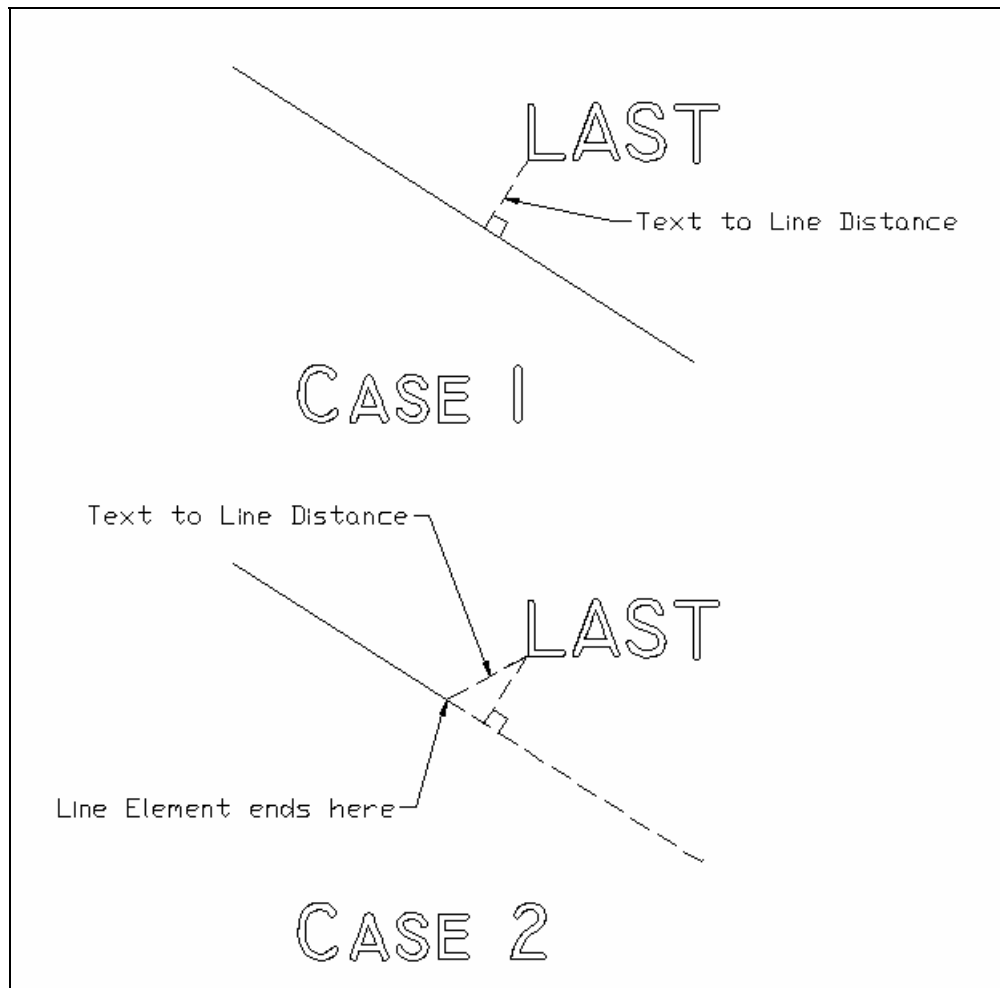
Figure 10: Measuring text-to-line distances

Polylines

The distance from a text element to a polyline element is the perpendicular distance to the closest line segment in the polyline. As with lines, if a perpendicular line cannot be drawn between the two elements, the distance is measured between the text element and the polyline endpoint. Although composed of lines, polylines remain distinct elements with only two endpoints.

Arcs

Usually the distance between an arc and a text element is measured as the difference between the arc radius and the distance from the arc center to the text (Figure 11, Case 1). If a radial line from the arc center to the

text element does not intersect the arc, the distance is measured between the text and the nearest arc endpoint (Figure 11, Case 2).
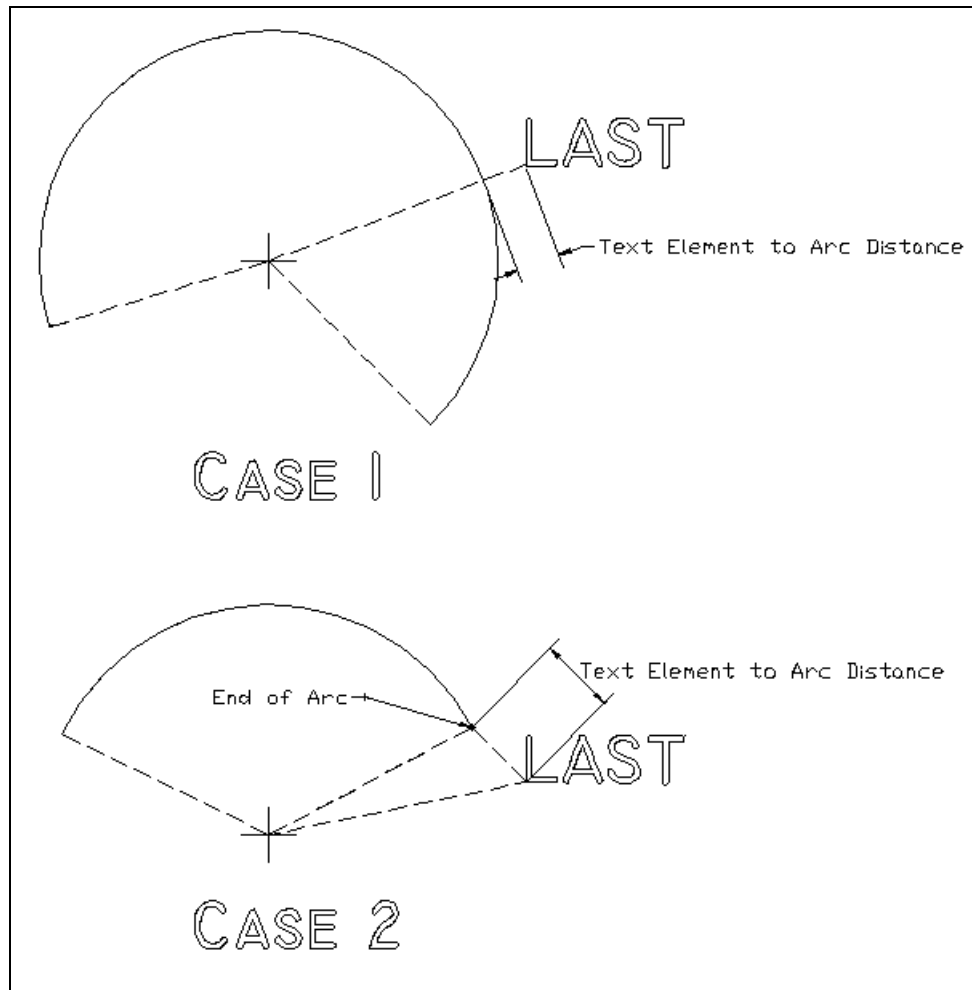

Figure 11: Measuring text-to-arc distances

Text
Some associations are just groups of text such as a macro definition. In these cases, the parent is usually a CAD-to-DMC command such as a macro heading and the children are additional parameters such as DMC commands. Text parents associate with the nearest text element within 1.2 character heights below the text with respect to drawing coordinates. Due to this requirement, bodies of text cannot be rotated. Each additional association child must also be within 1.2 text heights below its predecessor. Text associations are typically terminated with a CAD-to-DMC directive such as END_MACRO. They also terminate when another text element is not found within 3 text heights. If a text association ends for this reason, a warning is issued.

2.3 Macros

A macro is a way to package several commands together under one heading. When the macro is executed, all of the associated commands are executed as well. The body of the macro may contain DMC commands contained in quotes, other macro names, and certain CAD-to-DMC commands. All macros must end with the END_MACRO statement.

The CAD-to-DMC software does not perform any error checking on DMC commands issued through a macro; any errors that do exist will only be detected at runtime. It is the responsibility of the user to ensure that valid commands are sent.

Macros are declared using the macro commands listed below. In addition, Appendix A contains examples which use macros extensively.

FIRST_MACRO
PURPOSE:
      The FIRST_MACRO is used to perform operations before any motion takes place. This can be used to submit initialization information to the controller and prepare for the beginning of the process.

USAGE:
      FIRST_MACRO
      <command>
      <command>
      ………..
      <command>
      END_MACRO

      If FIRST_MACRO is present in a drawing, it will automatically execute prior to the beginning of motion. A user specified macro name is not required with this command so any word directly following FIRST_MACRO will generate a warning.

LAST_MACRO
PURPOSE:
      The LAST_MACROis used to send commands after the motion sequence is complete and all other commands have been sent. This is useful for shutting down the controller.

USAGE:
      LAST_MACRO
      <command>
      <command>
      ………..
      <command>
      END_MACRO

      If LAST_MACRO is present in a drawing, it will automatically execute after all other motion commands and macros have executed. Like FIRST_MACRO, LAST_MACRO requires no user name.

LINETYPE
PURPOSE:
      LINETYPE can be used in place of a user defined macro name to associate a macro with a particular linetype. For example, you could create a macro to increase cutting speed whenever a dashed line is encountered along the motion path. LINETYPE can be used with either MOVING_MACRO or STOPPING_MACRO.

USAGE:
      MOVING_MACRO (or STOPPING_MACRO) LINETYPE

      <command>
      <command>
      ………..
      <command>

END_MACRO

LINETYPE is substituted for the user-defined macro name. A line element of the linetype you would like to associate with the macro must be included directly below the macro declaration. If this linetype is missing, the macro will be ignored by CAD-to-DMC. When declared correctly, the macro will execute whenever the associated linetype is encountered during motion.

MOVING_MACRO
PURPOSE:
A MOVING_MACRO is used to perform operations during a motion profile. The command sequence is not executed until the appropriate location along the motion path has been reached.
USAGE:
MOVING_MACRO <name>
<command>
<command>
……….
<command>
END_MACRO

The word MOVING_MACRO is followed by a one word user-specified name which will refer to the entire macro. If a name is missing or if there are any words beyond the name a warning will be issued.

The macro is called by placing the macro name at the desired location along the motion path. The macro behavior will occur at the element endpoint nearest the name location along the motion path.

STOPPING_MACRO
PURPOSE:
A STOPPING_MACRO is used to perform operations between motion elements. Even if the program is currently in SMOOTH_MODE the motion will stop at the nearest element endpoint, the macro will be performed, and motion will continue.
USAGE:
STOPPING_MACRO <name>
<command>
<command>
……….
<command>
END_MACRO

The word STOPPING_MACRO is followed by a one word user-specified name which will refer to the entire macro. If a name is missing or if there are any words beyond the name a warning will be issued.

The macro is called by placing the macro name at the desired location along the motion path. The macro behavior will occur at the element endpoint nearest the name location along the motion path.

2.4 Command Descriptions

COMMENT_LINETYPE
PURPOSE:
> Sometimes it is useful in a drawing to include elements that are not related to the motion path but are helpful in relating overall part geometry to the user. The COMMENT_LINETYPE command allows such items to be included in the drawing without affecting motion translation.

USAGE:
> COMMENT_LINETYPE
>
> -------------------
>
> In a Macro   Yes
> Association   None
>
> COMMENT_LINETYPE is placed in a drawing with a segment of the linetype desired for comments placed directly below. This has the appearance of a legend on a map. The line must be placed within a character height of the text position of the command. Note that the comment linetype should not be used by the elements of the drawing representing motion or your drawing will be one large comment.

CORNER_ANGLE
PURPOSE:
> CORNER_ANGLE is used to specify the maximum allowable rotation angle between contiguous motion segments. If the angle is exceeded, motion will stop between the segments.

USAGE:
> CORNER_ANGLE=n
>   where n is an angle in degrees
> In a Macro   No
> Association   Drawing
>
> NOTE: CORNER_ANGLE must always be used in conjunction with CORNER_MODE.

EXAMPLE:
> Figure 12 shows the use of the CORNER_ANGLE command. With the angle specified, the motion will not stop at the transition between segments 2 and 3 because the angle of rotation is only 30 degrees. However, if the CORNER_ANGLE were specified to be 25 degrees, the motion would also stop at the transition between segments 2 and 3.
>
> NOTE: Angles in Figure 12 are shown for clarity and should not be included in a drawing to be translated.

CORNER_MODE
PURPOSE:
> CORNER_MODE is used to specify constant velocity motion unless a sharp transition is encountered. As long as the angle of rotation between consecutive segments is less than the specified CORNER_ANGLE, motion will be continuous. Any angle of rotation that exceeds the CORNER_ANGLE will cause the motion to stop at the transition between segments.

USAGE:
> In a Macro   No
> Association   Drawing
>
> NOTE: CORNER_MODE must always be used in conjunction with CORNER_ANGLE.

EXAMPLE:

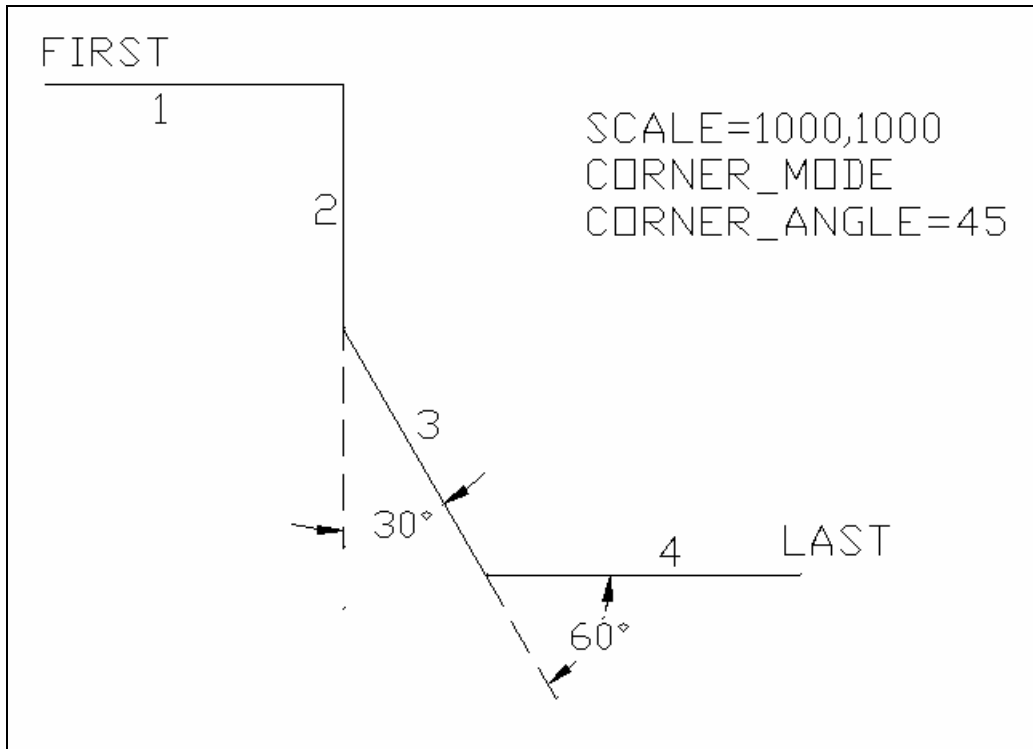Figure 12 shows the use of the CORNER_MODE command.

Figure 12: Use of the CORNER_ANGLE and CORNER_MODE commands
Note: Angles in Figure 12 are shown for clarity and should not be included in a drawing to be translated.

FIRST
PURPOSE:

FIRST indicates where the motion path begins in a drawing.

USAGE:

In a Macro   No
Association   Line Segment

The command FIRST is placed next to the endpoint of the line segment where motion is to begin. The first motion CAD-to-DMC performs will be from this endpoint to the opposite endpoint of the segment. This provides CAD-to-DMC with both initial position and initial direction information.

NOTE: This command is required in every CAD drawing you wish to convert.

LAST
PURPOSE:

LAST indicates where the motion path ends in the drawing.

USAGE:

In a Macro   No
Association   Line Segment

The command LAST is placed next to the endpoint of the line segment where motion is to end. The motion sequence will stop at the endpoint associated with LAST.

NOTE: This command is required in every CAD drawing you wish to convert.

REM
PURPOSE:
A line of text preceded by REM is considered a comment and is ignored by CAD-to-DMC. REM can be used to document drawings and provide other descriptive information for the user.
USAGE:
REM <comments>
In a Macro   Yes
Association   None

Note that REM must be followed by a space to work properly. A REM command only applies to the line of text in which it appears; if several lines of text appear in a drawing, each one must be preceded by REM. REM statements are allowed within the body of a macro as well as throughout the drawing.
EXAMPLES:
REM PROGRAM FOR TRIMMING FLASH
REM Author: John Doe, Date: 12/31/99

RESET_ORDER
PURPOSE:
RESET_ORDER is used to reset the Order Number count back to its initial value of 0. This is useful when creating a motion path containing repeating patterns. By invoking RESET_ORDER, it is possible to directly copy a section of a CAD drawing without changing the Order Numbers associated with the section.
USAGE:
In a Macro   Yes
Association   Line
EXAMPLE:
Figure 13 shows a repeating 'J' pattern which takes advantage of the RESET_ORDER command. In this example, the user-defined macro R is used to invoke the function. Alternatively, it would have been possible to omit the STOPPING_MACRO R and to substitute RESET_ORDER for R at the appropriate locations in the drawing.
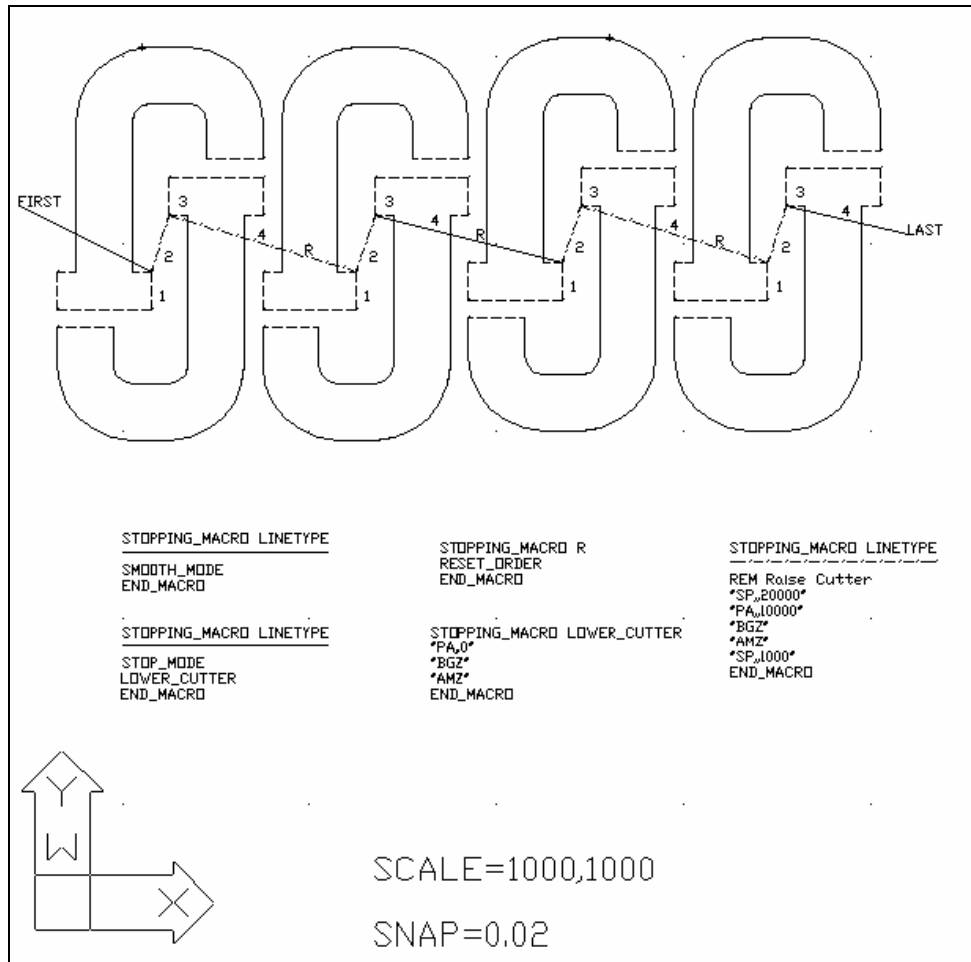
Figure 13: Use of the RESET_ORDER command

REVERSE_X, REVERSE_Y

PURPOSE:

The REVERSE_X and REVERSE_Y commands are used to mirror the interpretation of a drawing image. REVERSE_X inverts the direction of the X axis and REVERSE_Y inverts the direction of the Y axis.

USAGE:

In a Macro   No
Association   Drawing

The reflections only affect the motion path that is translated and not any DMC commands which are directly included within a macro. For example, "PR2000,2000" would not be affected by the REVERSE_X or REVERSE_Y commands. The reflection origin is with respect to the FIRST point in the drawing. Some systems may require the reverse commands to achieve the expected motion paths. For example, a stationary cutter operating on a moving XY table would need to have both axes reversed in order to have the cuts made as drawn.

SCALE

PURPOSE:

SCALE is used to relate encoder resolution to the user units used in a CAD drawing. It is usually a function of encoder resolution, lead screw pitch and gear ratios which connect the motor to the system.

USAGE:

SCALE= X,Y

where X and Y are the encoder counts per user unit for the X and Y axes, respectively

In a Macro   No
Association   Drawing

NOTE: This command is required in every CAD drawing you wish to convert.

EXAMPLE:

SCALE=1000,1000

SMOOTH_MODE

PURPOSE:

SMOOTH_MODE causes all motion to be represented as a single, continuous path with constant velocity through all segments.

USAGE:

In a Macro   Yes
Association   Line Segment or Drawing

SMOOTH_MODE can be used within a macro, along a motion trajectory, or all by itself. If found along a motion trajectory, the mode change will take effect after the endpoint of the associated element. If the command is not near a geometry element it will apply to the entire drawing. If the motion path contains sharp corners, large accelerations will occur at the transitions which will shock the system and increase position error and accelerate wear of system components. As a result, SMOOTH_MODE should only be used with motion paths containing smooth transitions.

SNAP

PURPOSE:

The SNAP command defines the maximum distance between elements that are considered touching and permits non-zero "contact" distances between elements.

USAGE:

SNAP=n

where n is a number expressed in drawing units of length

In a Macro   No
Association   Drawing

The SNAP command is an assignment statement which is associated with the drawing as a whole. If the SNAP command is not specified in a drawing, a default value of 0.1 will be used. A line segment cannot be shorter than the SNAP value specified. Otherwise, both ends of the segment and the endpoint of the next segment would all be within the SNAP distance and the program would be unable to determine which point is next. If this occurs, reduce the SNAP distance.

EXAMPLES:

SNAP=0.1
SNAP=10.5

STOP

PURPOSE:

STOP is used in a drawing which is operating in SMOOTH_MODE. STOP can be placed along a motion trajectory to suspend SMOOTH_MODE operation for the segment specified.

USAGE:

In a Macro   No

Association   Line Segment

STOP causes motion to stop momentarily at the nearest element endpoint. The coordinated move
will be completed and the system will be prepared to make another coordinated move sequence. It
is not necessary to use STOP from within STOP_MODE, as motion already stops at every
segment in this mode.

NOTE: If you would like to use STOP before performing a macro operation, use a
STOPPING_MACRO (see section 2.3 Macros).

STOP_MODE
PURPOSE:
STOP_MODE causes the program to treat each segment as a separate move. Motion will stop at
the endpoint of each segment in the sequence.
USAGE:
In a Macro   Yes
Association   Line Segment or Drawing

STOP_MODE can be used within a macro, along a motion trajectory, or all by itself. If found
along a motion trajectory, the mode change will take effect after the endpoint of the associated
element. If the command is not near a geometry element it will apply to the entire drawing.
STOP_MODE is recommended whenever there are sharp transitions along a motion path.

TOOL_RADIUS
PURPOSE:
TOOL_RADIUS is used to specify the radius of a tool bit for offset compensation of the motion
path.
USAGE:
TOOL_RADIUS=n
          where n is a number representing the tool radius in drawing units
In a Macro   No
Association   Drawing

NOTE: TOOL_RADIUS must always be used in conjunction with TOOL_SIDE.
EXAMPLE:
Figure 14 shows an application of the TOOL_RADIUS command.

TOOL_SIDE
PURPOSE:
TOOL_SIDE is used to specify on which side of the line TOOL_RADIUS correction is to be
performed.
USAGE:
In a Macro   No
Association   Line

The TOOL_SIDE directive is placed next to a line element on the side of the tool path that the
cutting tool is approaching from.

NOTE: TOOL_SIDE must always be used in conjunction with TOOL_RADIUS.
EXAMPLE:
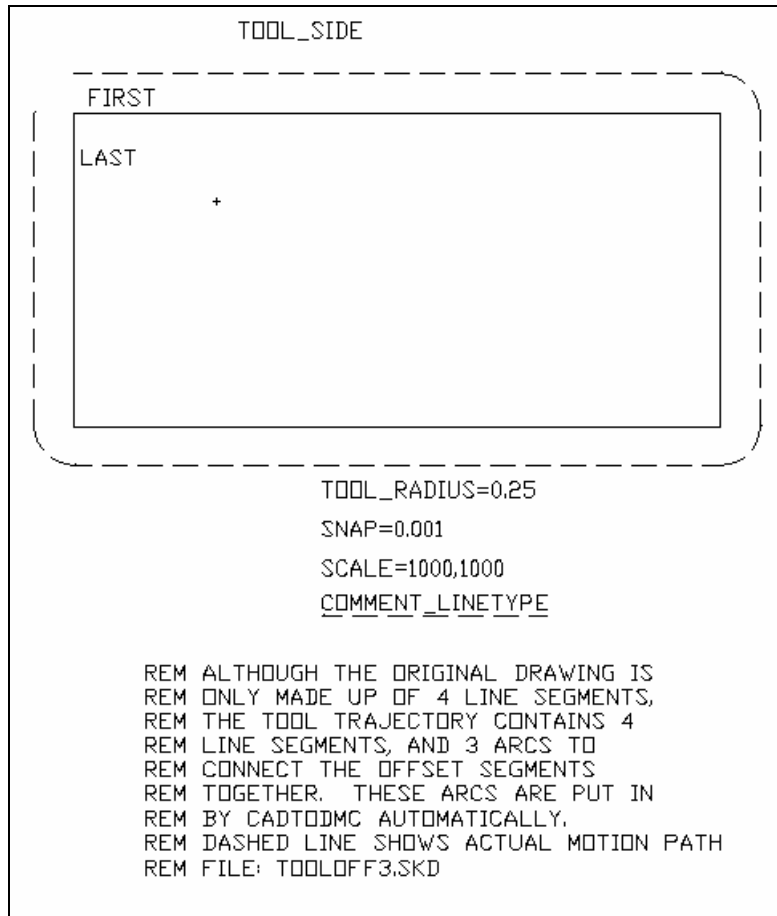Figure 14 shows an application of the TOOL_SIDE command.

Figure 14: Use of TOOL_RADIUS and TOOL_SIDE commands

2.5 Order Numbers

CAD drawings often contain a variety of branch points and line intersections. CAD-to-DMC requires sequencing information at each of these intersections in order to determine the proper motion path. This sequence information is provided through Order Numbers. Order Numbers can be used to resolve intersections along a path and to perform multiple cycles of sections of a drawing.

Order Numbers are text elements that are associated with line elements of a CAD drawing. When a number is placed next to a line element, it becomes associated with that element and can be used to resolve directions at intersections. CAD-to-DMC keeps track of the current Order Number and looks for the next highest number upon reaching an intersection. The motion path will always go in the direction of the next order number. CAD-to-DMC always begins counting with the number 0 so it is a good idea to use the number 1 at the first intersection of the drawing and increment by 1 at each subsequent intersection.

EXAMPLE:
Figure 15 shows a CAD drawing of a square with an entry line on the left and a departure line on the right. CAD-to-DMC begins following the path at FIRST, where the value of the order count is 0. Upon reaching the first intersection, it sees 2 to the right and 1 to the left. Since 1 is the next highest number, the motion path goes to the left. After tracing the top three sides of the square, CAD-to-DMC encounters another intersection where it turns right, in the direction of the 2. The path continues along the bottom side of the square until it encounters the first intersection again. At this point, the current count is 2 so CAD-to-DMC ignores the 1. However, a 3 is also present so it turns to the right, again tracing the top three sides of the

square. When the second intersection is again reached, CAD-to-DMC turns left towards the 4, completing the motion path.
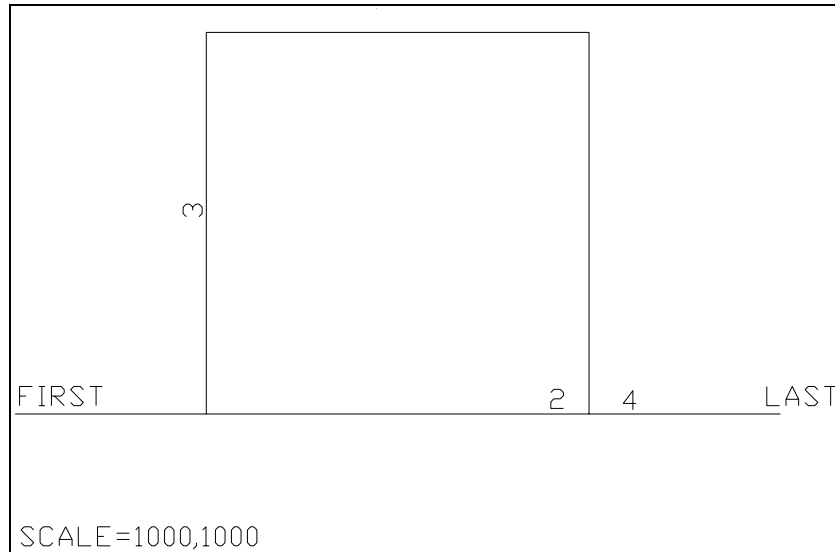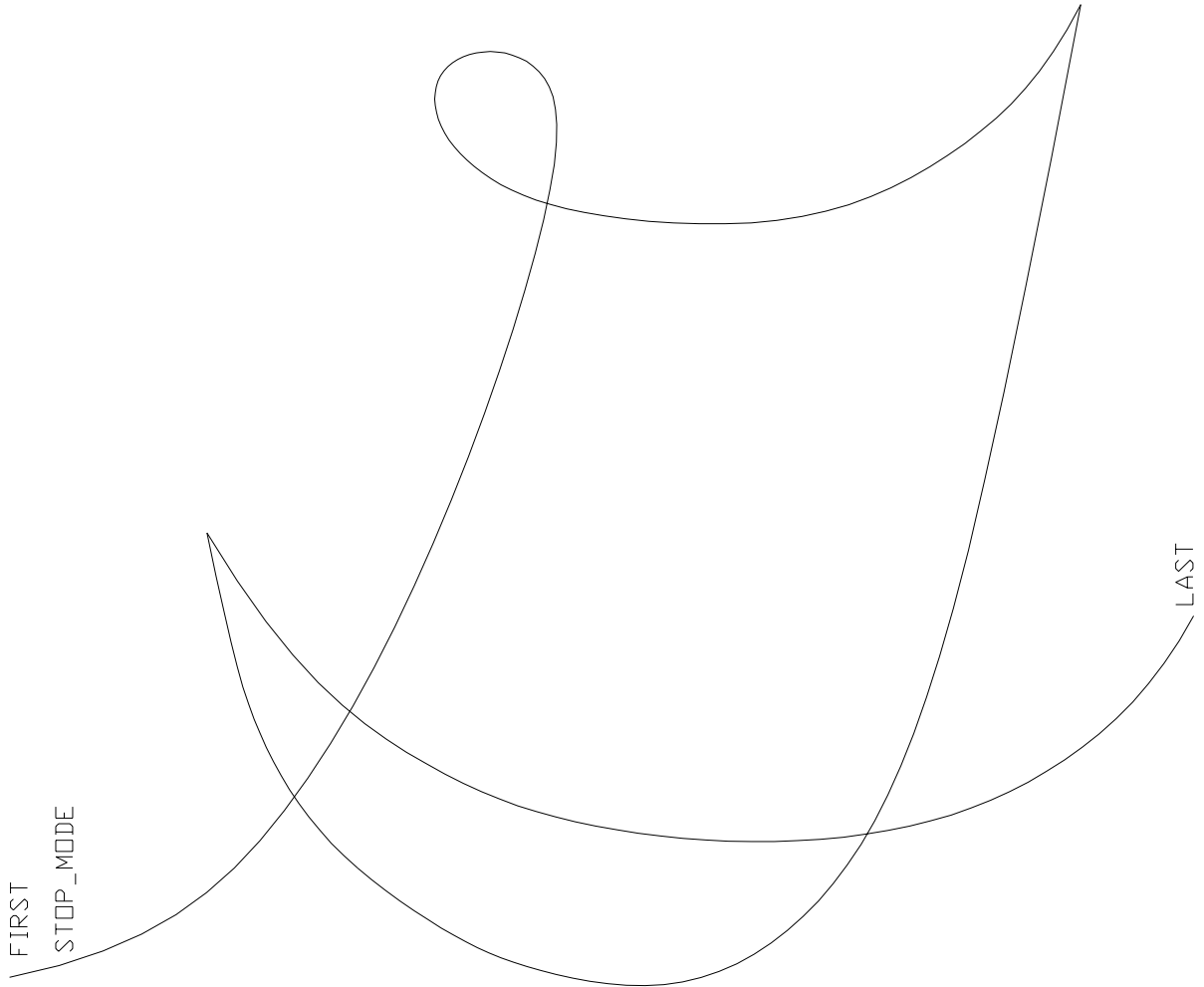

**Figure 15:** Use of Order Numbers

THIS PAGE LEFT BLANK INTENTIONALLY

Appendix A

Example Drawings

FIRST
STOP_MODE

LAST

SCALE=1000,1000

REM THIS DRAWING IS COMPOSED OF 3 AUTOSKETCH (TM) GENERATED
REM CURVES. THE FIRST DIRECTIVE IS USED AT THE BEGINNING OF
REM THE FIRST AND THE LAST DIRECTIVE IS USED AT THE END OF
REM THE THIRD. BY VARYING SETTINGS\CURVE\DRAWING-SEGMENTS
REM THE NUMBER OF LINE SEGMENTS THE G IS MADE OF CAN BE
REM VARIED. THE ONLY PENALTY FOR HAVING A LARGE NUMBER OF
REM SEGMENTS IS THE POSSIBILITY OF OVERRUNNING THE SEGMENT
REM LIMIT OF YOUR CONTROLLER. YOU WILL BE WARNED ABOUT THIS.
REM STOP_MODE IS USED TO PREVENT THE SHOCK THAT WOULD
REM OTHERWISE RESULT FROM AN ATTEMPTED CONSTANT SPEED
REM DIRECTION REVERSAL AT THE TOP AND LEFT SIDES OF THE G.

REM FILE: SCRIPT_G.SKD

4

1

STOP_MODE

3
2

FIRST
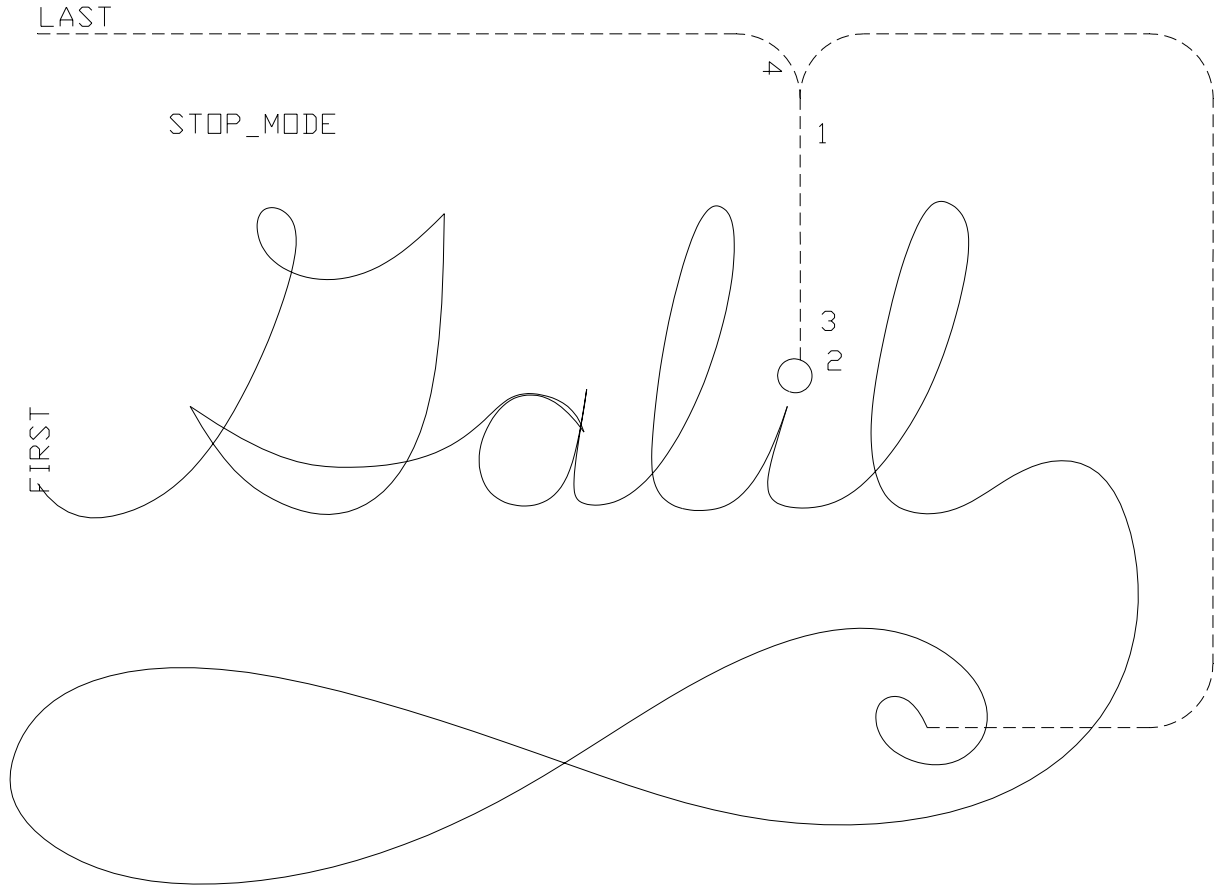
STOPPING_MACRO LINETYPE
_____
REM DON'T STOP DURING TRAVERSE
SMOOTH_MODE
REM RAISE ENGRAVER
"PA,,1000"
"BGZ"
"AMZ"
END_MACRO

SCALE=1000,1000

STOPPING_MACRO LINETYPE
_____
REM STOP AT CURVE ENDS
STOP_MODE
REM LOWER_ENGRAVER
"PA,,0"
"BGZ"
"AMZ"
END_MACRO

SNAP=0.02

REM THIS DRAWING MIGHT BE A MOTION DESCRIPTION FOR AN ENGRAVING
REM MACHINE. THE LINETYPE DIRECTIVES PERFORM TWO FUNCTIONS.
REM THE CUTTING HEAD IS RAISED AND LOWERED BY THE Z AXIS, AND
REM THE MOTION MODE IS CHANGED. THE CURVES OPERATE IN STOP_MODE
REM SO THAT LARGE ACCELERATIONS DO NOT OCCUR AT THE SHARP
REM CORNERS IN THE LETTERS. SMOOTH_MODE IS USED FOR THE
REM TRAVERSE BECAUSE THE LINES AND ARCS CONNECT TOGETHER
REM SMOOTHLY.

REM SCR_LOGO.SKD

FIRST

STOPPING_MACRO LINETYPE

"PA,,0"
"BGZ"
"AMZ"
END_MACRO

STOPPING_MACRO LINETYPE

"PA,,20000"
"BGZ"
"AMZ"
END_MACRO

LAST
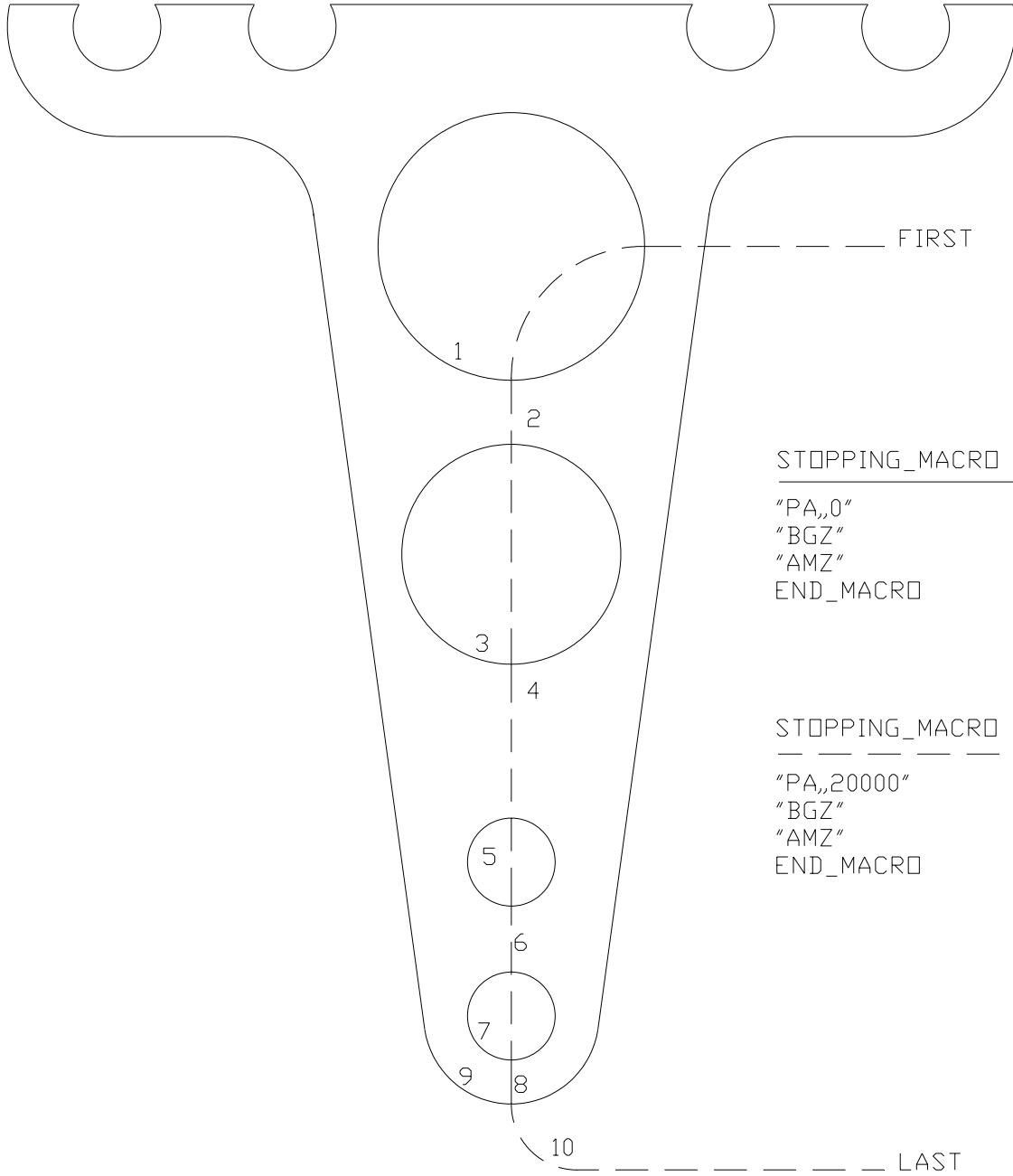
SCALE=1000,1000

REM   THIS DRAWING SHOWS THE USE OF STOPPING
REM   MACROS TO CONTROL THE HEIGHT OF A TOOL
REM   OVER THIS BRACKET OUTLINE.

REM   FILE: BRACKET.SKD

STOP

1   2

FIRST,LAST
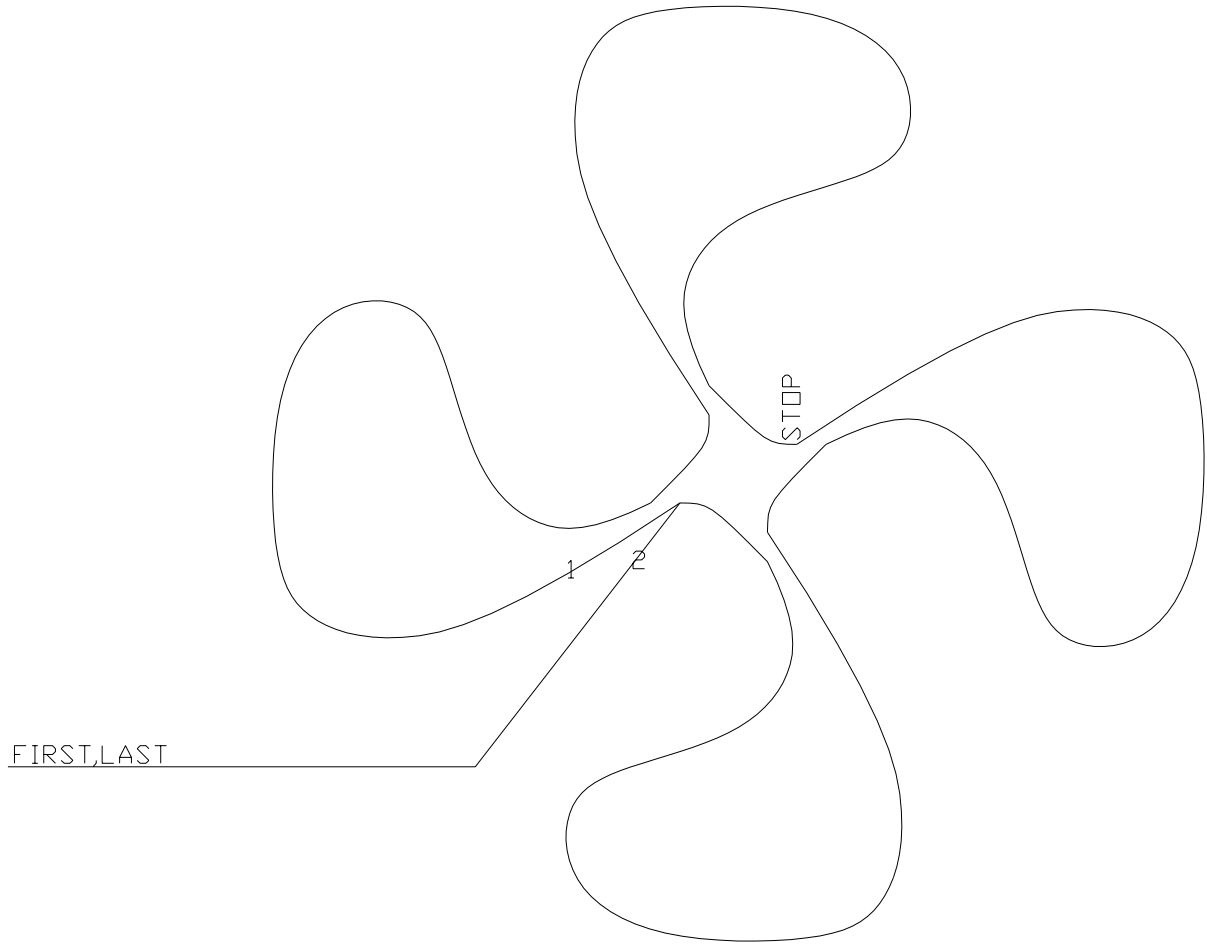
SCALE=1000,1000

REM THIS DRAWING WAS MADE WITH 8 CURVES AND
REM 2 LINES. THE ORDER NUMBERS ARE USED TO
REM DIRECT CADTODMC CLOCKWISE AROUND THE "FAN
REM BLADE". THE ORDER NUMBER "2" IS USED TO GET
REM THE TRAJECTORY OFF OF THE CURVES AND BACK
REM ON TO THE ENTRANCE\EXIT TRAJECTORY.

REM FANBLADE.SKD
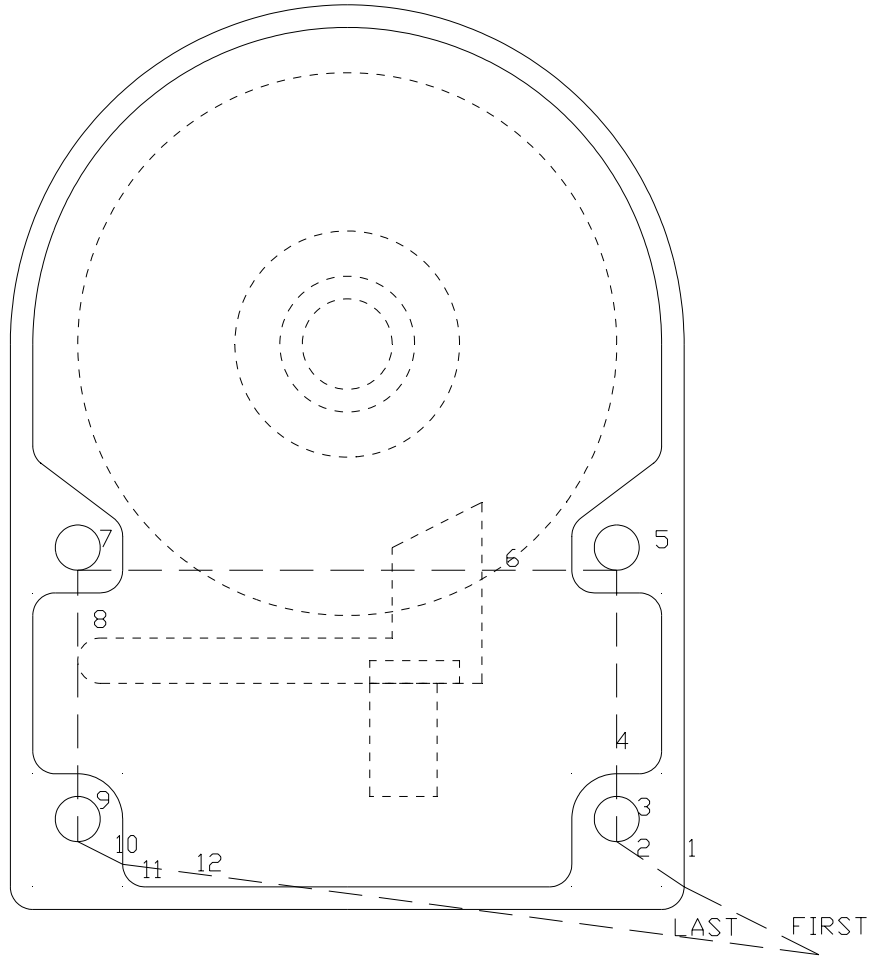
STOPPING_MACRO LINETYPE
— —— ——— — — — — — —
"PA,,10000"
"BGZ"
"AMZ"
"VS20000"
END_MACRO

STOPPING_MACRO LINETYPE
———————————————————————
"PA,,0"
"BGZ"
"AMZ"
"VS3000"
END_MACRO

SCALE=1000,1000

COMMENT_LINETYPE
- - - - - - - - - - - - - - - - -
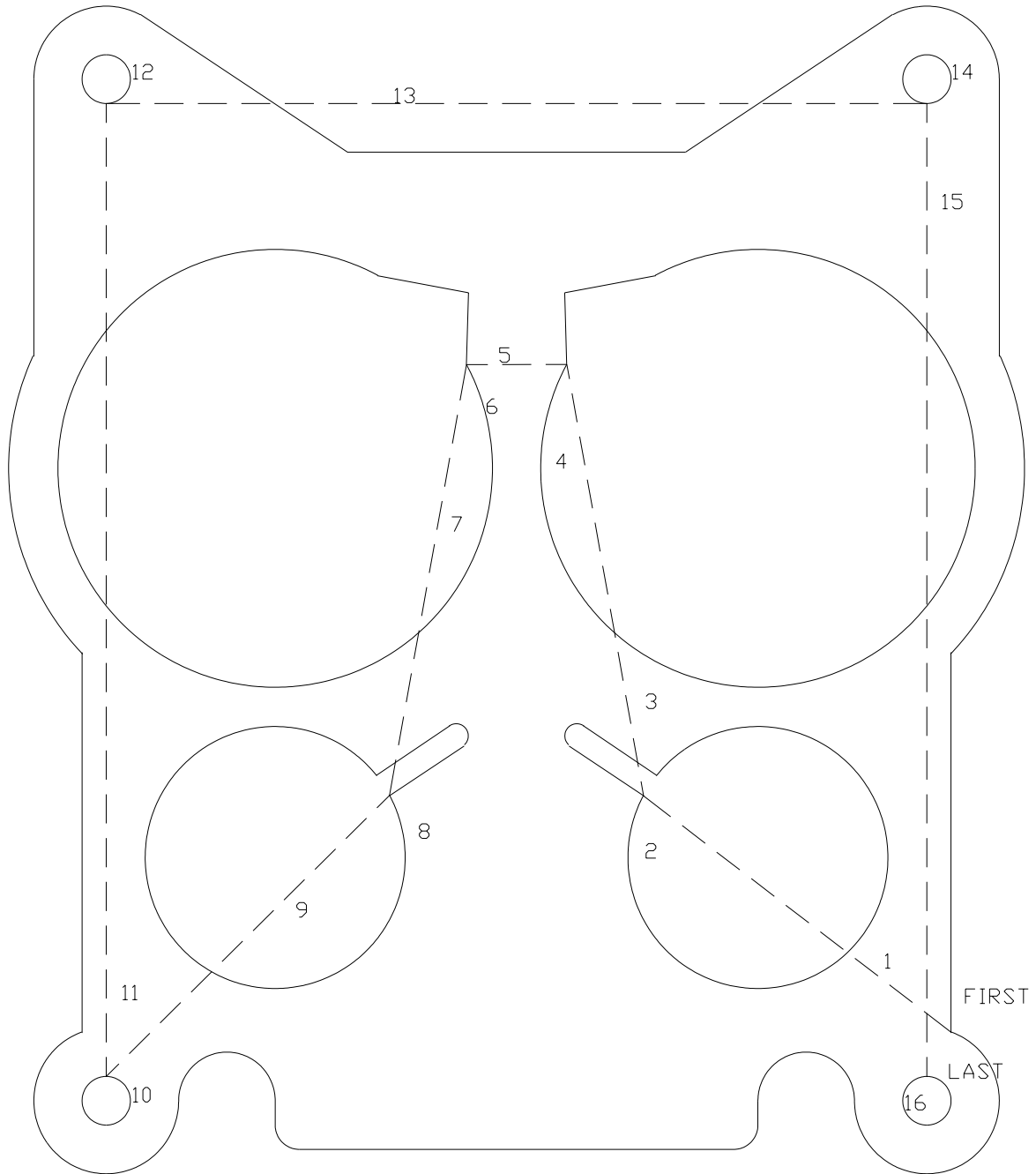
SNAP=0.05

REM POSSIBLE GASKET CUTTING CONTEXT. HERE THE
REM GASKET OUTLINE ON BOTH SIDES AND HOLES ARE
REM CUTOUT BY A PROGRAM WHICH PERFORMS HIGH
REM SPEED SKIPS ALONG DOTTED LINE SECTIONS.

REM FILE:GASKET_1

STOPPING_MACRO LINETYPE

— — —— — — — — —

"PA,,30000"
"BGZ"
"AMZ"
END_MACRO

STOPPING_MACRO LINETYPE
_____
"PA,,0"
"BGZ"
"AMZ"
END_MACRO

SCALE=1000,1000

SNAP=0.02

REM APPROXIMATION OF A CARBURETOR GASKET FOR A V8
REM ILLUSTRATES THE USE OF LINETYPE MACROS AND ORDER NUMBERS

REM FILE: CARBGASK.SKD

INDEX