# A Post-Processor Interface For CADAM NC and Dyna Milling Machines

by

Harinder Singh Oberoi

Thesis submitted to the Faculty of the

Virginia Polytechnic Institute and State University

in partial fulfillment of the requirements for the degree of

Master of Science

in

Mechanical Engineering

APPROVED:

Dr. A. Myklebust, Chairman

Dr. M. P. Deisenroth

Dr. C. F. Reinholtz

May 29, 1987

Blacksburg, Virginia

**A Post-Processor Interface For CADAM NC and**

**Dyna Milling Machines**

by

Harinder Singh Oberoi

Dr. A. Myklebust, Chairman

Mechanical Engineering

(ABSTRACT)

This thesis introduces a program which serves as a post-processor interface between CADAM NC and a DYNA milling machine. This program processes the CLDATA file that is the output of the IBM System/370 APT-AC processor. The input to the APT-AC processor is an APT file, generated by passing CADAM NC data through the CADAM/APT interface. The CLDATA file is post-processed and instructions in DYNA language, required to machine the part on the DYNA milling machine, are created. This is achieved through different subroutines that have been written for different types of statements in the CLDATA file. For each type of statement in the CLDATA file, a different subroutine is accessed which processes that statement according to the requirements of the DYNA milling machine and writes the corresponding statement in DYNA language to a file. The post-processor then reads the different DYNA statements from different files and creates the complete part program in DYNA language required to machine the part.

*"He who would prove all life, leaves it empty,*
*To know the way of everything is to be left with*
*the geometry of things and with the substance of*
*nothing,*
*To reduce the world to an equation is to leave it*
*without head or feet"*

*Leopoldo Alas Y Urena,*
*Spanish Novelist*

# Acknowledgements

This thesis is dedicated to my parents and my brother, without whose love, support, and encouragement, I would not have accomplished what I have.

To Dr. Arvid Myklebust, my advisor and teacher, my sincere thanks and deepest gratitude for everything. The experience of working with him shall remain as one of my cherished memories.

I am grateful to Dr. M. P. Deisenroth for his advice and his willingness to dedicate his time and efforts towards my research. I also thank him for having served on my committee.

I thank Dr. C. F. Reinholtz for having served on my committee and devoted his time to advise me.

This thesis would not have seen its completion without the efforts of Roy Vickers and Mitch Keil. Roy for having brought the APT Interface to a working condition and Mitch for his friendly advice and willingness to help at all times. Their efforts are sincerely appreciated.

Amongst my friends I would like to thank Ashit Gandhi, Jayaram Sankar, V. Arun, and Sandi Pennington for their constant support and advice. I am also grateful to Rajbir Singh Parmar and Chandra for their friendly and valuable advice.

I also thank my friend Amod Wagh for having stood by me throughout my life.

Finally I would like to thank Almighty GOD for everything.

# Table of Contents

# List of Illustrations

# Chapter 1

# Introduction

Current manufacturing technologies and concepts trace their roots back to the beginning of the industrial revolution. In today's competitive world, manufacturing industries have realized the necessity for improved production quality and the need has evolved for refinement of existing machine tools. Up to the beginning of 1940, all machine tools were under manual control. In the early 1940's John Parsons conceived a method of using punched cards for machine tool control [1]. Thus originated the concept of Numerical Control (NC) of machine tools. Numerical Control may be broadly defined as a form of programmable automation in which a machine is controlled by numbers. A programming language tailored specifically to meet NC needs was developed and named Automatically Programmed Tools (APT).

The greatest revolution in manufacturing technology was brought about by the emergence of the digital computer. In the early phase of computer technology, the potential for graphical data processing was realized and soon computers began to be used in design and drafting, giving birth to Computer-Aided Design (CAD). As computers began to be used in a manufacturing environment, Computer-Aided Manufacturing (CAM) evolved. With more advancement in computer-aided design, it was found that interactive graphics increased design and drafting productivity. At the same time, computers began to be used more widely in numerical control part programming. Interactive graphics began to be used in the part programming process leading to the development of CAD/CAM systems. The need was soon realized for the integration of CAD and CAM. Today, an area under constant development is the creation of the necessary link between CAD and CAM.

The use of interactive graphics in NC part programming is a good example of the integration of computer-aided design and computer-aided manufacturing. The programming procedure is carried out on the graphics terminal of a CAD/CAM system. Using the same geometric data which defined the part in the computer-aided design process, the programmer constructs the tool path using high level commands to the system. In most cases the tool path is automatically generated by the software of the CAD/CAM system. The output resulting from this procedure is a listing of the APT file or the actual CLFILE (cutter location file) which can be post-processed to generate the NC instructions. These NC instructions are then down loaded to the machine for machining the part.

# *Objectives*

The principle objective of this study is to integrate a CAD system with a CAM machine. After designing the part on a CAD system, the user should be able to generate NC instructions for machining the part and these instructions should then be processed to generate a CLDATA file, which is post processed to generate the specific instructions to be downloaded to the machine controller.

The process of producing a part from concept to design to its manufacturing or machining is shown in Figure 1. The user starts by designing the part on the CADAM system. The tool path for machining the part is then created by using the CADAM NC function. The NC data along with the geometric data of the part is then passed through the CADAM/APT interface to generate an APT file for machining the part. This APT file is then processed by the APT/AC processor to generate the cutter location data (CLDATA) file for machining the part.

The program CADDYN introduced in this thesis, uses the CLDATA file as its input and post processes it to generate specific NC instructions for the DYNA milling machine. The output of the program CADDYN is a file that contains the instuctions in DYNA language for machining the part. This file is down loaded on to a disk from which it can be down loaded to the machine controller through a personal computer, for machining the part.

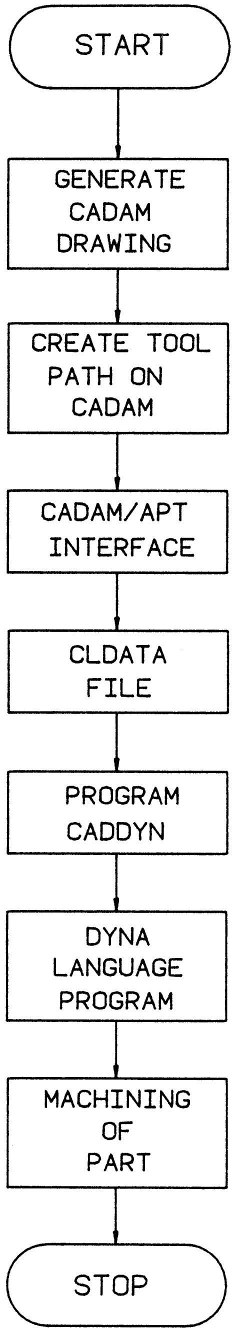


**Figure 1. Flowchart for CAD to CAM process**

## *Scope of Study*

This thesis introduces a program named CADDYN that has been created to write the NC instructions for a milling machine. The program has been written in FORTRAN 77 and is implemented on a IBM-4341 operating under VM/CMS and serves as an interface between CADAM NC and the DYNA milling machine. CADAM is the CAD/CAM System used to generate the tool path and IBM system /370 APT-AC is the software used for generating the CLFILE for this tool path.

# Chapter 2

# Literature Review

With the recent developments in computer technology and the emergence of CAD/CAM systems, the integration of CAD and CAM has become an important issue in manufacturing industries. Computer graphics used in NC part programming is a prime example of the integration of CAD and CAM. Continued research in both computer graphics and NC part programming has resulted in advancement of their original concepts. Today's research concentrates on integrating computer graphics into the manufacturing environment.

Krouse [2], while discussing the importance of integrating CAD and CAM, points out that it would, in the long run lead to the concept of 'engineering without paper'. Nilson [3] asserts that integrating CAD/CAM projects in an industry would lead to great reductions in lead time and cost, whereas Wilson [4] believes that with the integration of CAD and CAM, there is improvement in internal operating efficiency and external marketing effectiveness of a company. As an example, Hannam and Miller [5] have developed a linkage between CAD and CAM for jig fixture design in a company and have found that it reduces lead times and cost, improving productivity to a good extent.

Numerical control (NC) part programming has come a long way from manual to computer assisted part programming. The use of computers in the part programming process led to the development of the programming language Automatically Programmed Tools (APT) for numerical control. APT was first developed at MIT under a joint program with the US Air Force. Twinstra [6] traces the history of APT and its evolution to the new APT-like languages. Today APT has become a standard NC processing language [7]. After its initial conception at MIT, most of the research in APT was conducted at the IIT Research institute [8] and an encyclopedia on APT was published by the same institute [9] which has been used in this thesis. Kendall [10] has developed a method of teaching APT to students at the university level. Recently APT has been further developed, and the latest version is APT IV, which is being further enhanced by the Sculptured Surface Project of CAM-I, Inc., (Computer-aided Manufacturing International, Inc.) in Arlington, Texas [11].

The output of any NC processor is a CLFILE (cutter location file). This CLFILE is the only link between NC language processors and post processors. CLFILE generated by different processors have different formats and word structures and converting CLDATA from one format to another is not simple. Subramanian [12] has developed software that reads CLFILE in six different formats, interprets their contents, edits the data and outputs the results in any one of the six different formats. CLFILE is the vital link between a CAD system and a CAM machine. A cutter location file cannot be directly input to a machine, it has to be post-processed and converted to a format acceptable to the machine controller. Today many commercial post-processors are being marketed.

Continued research is being carried out in interfacing CAD drawings with the actual manufacture of parts. Zapomueel [13] has written software called MAPROS for the integration of NC drawing and manufacture of mechanical workpieces. Kishi, Nagai and Hatta [14] have developed OKISURF which is a general purpose NC language system for machining sculptured surfaces. UNILEX developed by Gossling and Warner [15] is an interface language between the computer programs which define the geometry of the sculptured three-dimensional surfaces and the post-processors which generate NC instructions for the machine tools. Chan [16] has developed an interface ROMAPT between a bounded geometric modeller ROMULUS and an unbounded NC processor APT.

At an instructional level research is being carried out at various universities to develop programs to provide a low-cost link between CAD and CAM. NCFILE is one such program written by Brooks and Riley [17]. This program has been developed on a Terak minicomputer using MINN/DRAFT (an educational CAD System). Some researchers are concentrating on small portions of the total integration process. One such area is the generation of the NC tool path on a CAD system. Kuttner, Majcher and Snedecor [18] at Ford motor company have done work in the area of automation of tool path generation. Bobrow [19] has developed an algorithm for generating NC tool path directly from constructive solid geometry part representations.

A new approach towards integrating CAD and CAM through generative process planning using artificial intelligence has been put forth by Phillips and Mouleeswaran [20]. Integration of CAD and CAM has come a long way since its initial perception. Post-processors remain a vital link in this process. It is hoped that the program CADDYN which serves as a post-processor interface between CADAM NC and DYNA Milling Machine will help in clarifying the concepts behind this integration process.

# Chapter 3

# Support Software

## *CADAM*

CADAM is a CAD/CAM System developed by Lockheed Corporation [21] and very widely used for computer-aided design and manufacturing. The computer-aided design portion of the CADAM system consists of a design/drafting module along with various aids for design analysis. The computer-aided manufacturing (CAM) portion of the CADAM system automatically generates NC data.

The CADAM system uses a central data base for storing geometric data. This data can be used, both in design and manufacturing. Geometric construction is based on descriptive geometry and the graphics display is equivalent to a drafting board with viewing transformation capability to get the desired resolution. In addition 3-D wireframes and Solid Models can be created.

Among the various capabilities of CADAM are:

- The facility for 3D mesh generation for construction of finite element models.
- Facility for storage of standard symbols and other user defined symbols.
- Ability to obtain hardcopy drawings through plotter output.
- English, metric or dual dimensions can be used.
- Design changes made to components which are used in various subassemblies can be rapidly incorporated into applicable drawings.
- 3-D Wireframe modeling
- Solid modeling

The CADAM system provides the user with the capability to generate the part program for a particular drawing, and this process is aided by the animation of the tool path on the graphics display [22]. This facility is available on the CADAM system under the function N/C. Traditionally the part programming process starts with the selection of the tool for the particular machining operation. The programmer uses the parts geometric description (created on CADAM) to generate the cutter path by selecting the various geometric elements. Once the cutter pass is created, it can be replayed for verification. The CADAM system animates the entire pass on the graphics display, showing the location of the cutter and displaying the X, Y, and Z coordinates. After verification, the various cutter passes are combined to form a program using the CADAM 'POST' menu item .

The CADAM system APT interface module consists of the necessary programs that interface CADAM with the IBM System/370 APT-AC processor. The cutter path along with other APT auxiliary control statements is supplied by CADAM to APT through the CADAM/APT interface to generate an APT file for the cutter path. The IBM System/370 APT-AC processor uses this APT file, together with the CADAM system drawing to produce a CLFILE (cutter location file) adhering to ANSI X3J7 standard. This CLFILE is then post-processed to generate the completed part program for machining the part.

## *IBM System/370 APT-AC Processor*

The System/370 Automatically Programmed Tool - Advanced Contouring (APT-AC) is a numerical control (NC) processor. IBM has been involved with the development and maintenance of NC processors for more than twenty years [23]. Its initial product System/360 AUTOSPOT allowed two-dimensional point to point machining operations only. System/360 APT was then developed for three-dimensional machining operations and the latest System/370 APT-AC is their most advanced processor.

The APT-AC processor accepts as input user-oriented language describing the NC operations to be performed. This language is based on the APT language and contains four types of statements that can be used in the part program [24]:

- Geometry statements - define the part geometry.
- Motion statements - define the motion of the cutter with respect to the part geometry.
- Passive and Dynamic statements - specify the machining conditions.

- Control statements - control the processing modes of the APT-AC processor.

The APT-AC system structure consists of the following major sections:

- The control system - it performs the input-output activities and checks the flow of the part program through the various sections of the APT-AC processor.
- The translation section - it translates the input file into an intermediate file called PROFIL which then becomes the input to the other sections of processing.
- The calculation section - it uses the PROFIL as input and calculates the successive cutter positions that represent the machined part.
- The edit section - it performs the necessary calculations to transform, print or plot the CLDATA.
- The post-processor section - it consists of two parts, a post-processor dispatching section and individual post-processors. The dispatching section, using the information from the input file, sends the CLDATA file to the individual post-processor which then generates the machine tool-readable instructions.

# Chapter 4

# Manufacturing Equipment

## *DYNA 2400 Milling Machine*

The manufacturing equipment used here for the purpose of integrating a CAD/CAM system (CADAM) to a CAM machine is the DYNA 2400 CNC milling machine.

The DYNA 2400 is a light (290 lb) bench top, four axes, vertical CNC milling machine. The machine requires a single phase power source of 120 VAC, 60 Hz. The power required is 500 watts [26]. The machine is generally operated from the controller keyboard. The range of the spindle speed is 0-10,000 RPM and can be achieved in three steps by a change in the position of the spindle drive belt. The spindle can be turned ON and OFF either manually or by an instruction in the program. The entire spindle head can be lowered, elevated, and rotated on the head post.

The fourth axis on the Dyna machine is an optional rotational table that can be mounted horizontally or vertically. For machining operations, the X and Y tables, and the spindle head (Z axis) move, but for conveinence, it is assumed that the tool tip moves (since it is the actual cutting point). The three axes X, Y, and Z are mutually perpendicular to each other. Their intersection point is defined as the zero coordinate and is expressed as (0,0,0). The zero coordinate point can be placed at different locations as required by machining conditions. Accordingly there are three positions of the zero coordinate:

- Home zero - It is the position when the X, Y, and Z axes are physically at their limit switches.
- Reference zero - It is the position which is programmed into the machine and to which all programmed functions are referenced.
- Local zero - It is specified by the user within the program and is used mainly to specify programming and to avoid unnecessary calculations of coordinate points.

The maximum travel in the three axes are:

X = 6.2 inches (157.48 mm)

Y = 5.0 inches (127 mm)

Z = 4.0 inches (101.6 mm)

The DYNA milling machine allows the user to specify the feedrate in X, Y, and Z direction. The feedrate specified should lie between 0.1 and 32.0 inches per minute. If the feedrate is not specified, a default value of 8.0 inches per minute (203.2 mm/min) is assumed.

The DYNA machine has four operating modes:

- The Manual Mode - It is used for tool calibration and for manual operation of the machine.
- The Line No Mode - It is used for review of programs stored in the controller of the machine and for editing the programs.
- The Program Enter Mode - It is used for entering a program into the controller of the machine.
- The Program Run Mode - It is used for operating the machine under program control.

The DYNA machine can be programmed with the help of prompts. This is done by displaying the prompt on the controller display. The user must respond to the prompt. If there is no response from the user, the controller will halt and will not go to the next step or accept any other entry.

The DYNA machine has an RS 232-C interface which allows the user to interface the controller with a external computer or peripheral communication link that can be set up to 50 feet away. This facility allows the user to download a program from the computer to the controller and then execute it, up load a program from the controller to the computer, and incrementally operate the machine (line by line download and execute). For programming the DYNA machine also has its own language DYNALAN which consists of a few very powerful instructions which are designed to simplify machine move programming. Along with this there are some functions which have been preprogrammed into the basic machine controller. Some of these functions include mill, frame, rectangular frame, rectangular pocket, circular pocket, arcframe, and drill.

# Chapter 5

# CLDATA and DYNA Language

The output of any general purpose numerical control processor is a file called CLDATA (cutter location file). This file is used as an input to a post processor which generates specific instructions to machine the part on a particular machine.

## *CLDATA Structure*

The CLDATA file basically contains the X, Y, and Z coordinates of the center of the cutter as it moves along the path to machine the part. The coordinates are printed along with other statements which are specific to the machine being used and hence are known as machine statements. These statements are cutter, feedrate, spindle on, spindle off, coolant on, coolant off, etc. The CLDATA file also contains GOTO statements which identify the surface along which the cutter has to move.

In the CLDATA file the cutter statement contains information regarding the diameter of the cutter, the length of the cutter, and the corner radius of the cutter. The feedrate statement contains information on the feedrate in the X, Y, and Z directions. The spindle on statement in the CLDATA file switches the spindle on and also contains information on the speed of the spindle.

## *DYNA Language*

The DYNA language consists of a few instructions which are designed to simplify machine move programming. Tool movement is specified by GO REL and GO ABS instructions. The GO REL instruction applies to any X, Y, Z, radius (r), or angle (a) moves. This instruction moves the tool from its present position to a new position which is stated relative to the tool's present position. The GO ABS instruction moves the tool from its present position to a new position which is relative either to the local zero coordinate or the reference zero coordinate. In addition, a description of the cut, internal or external can also be specified.

The DYNA language also contains tool diameter, feedrate, spindle on, spindle off, and dwell (which causes a delay of a specific period in the program) statements.

## *Conversion of CLDATA to DYNA Language*

The CLDATA file cannot be directly input to a machine controller because of its structure. An additional step - post processing is required to adapt the CLDATA output to the particular machine control unit. This is done by a program called the Post Processor.

The primary functions of the post processor are to:

- convert CLDATA to the machine coordinate system.
- ensure that the physical limits of the machine are not exceeded, e.g. range, feed rate.
- eliminate reader limitations.
- calculate cutter compensation information.
- generate error diagnostics when necessary.

The program CADDYN developed in this thesis serves as a post processor for the DYNA milling machine. It reads in the CLDATA for a particular part and converts it into a program in DYNA language.

As an example, the APT file, the CLDATA file and the corresponding program in DYNA language to machine a triangle are shown. Figure 2. shows the cutter path for machining the triangle.

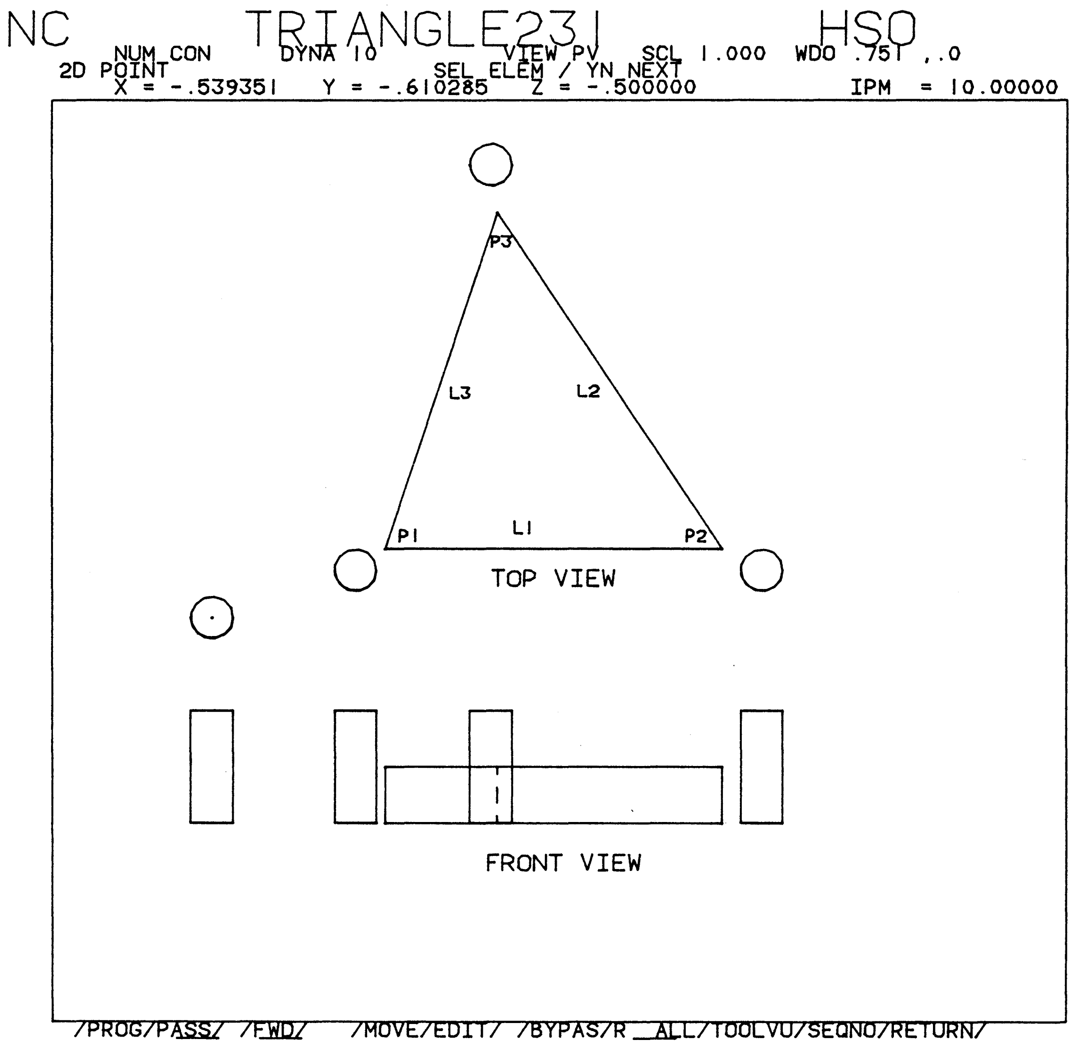


**Figure 2. Cutter path for machining triangle**

# *APT File for machining Triangle*

```
PARTNO              TRIANGLE
SETPT  =  POINT/0,0,0
P1     =  POINT/1.0,0.0,0.0
P2     =  POINT/4.0,0.0,0.0
P3     =  POINT/2.0,3.0,0.0
P4     =  POINT/1.0,1.0,0.0
P5     =  POINT/0.5,0.5,-0.1
P6     =  POINT/0.0,0.0,-0.1
P7     =  POINT/0.5,-0.5,-0.1
PL1    =  PLANE/P5,P6,P7
L1     =  LINE/P1,P2
L2     =  LINE/P2,P3
L3     =  LINE/P3,P1
L4     =  LINE/P4,P1
          CLPRNT
          CUTTER/0.375
          FEDRAT/10.0
          FROM/SETPT
          SPINDL/ON
          GO/TO,L1,TO,PL1,TO,L4
          GORGT/L1,PAST,L2
          GOLFT/L2,PAST,L3
          GOLFT/L3,PAST,L1
          GOTO/SETPT
          SPINDL/OFF
          FINI
```

# *CLDATA for Machining a Triangle*

```
0001 PARTNO/  TRIANGLE                                  00002
0016 CUTTER/    0.3750                                  00004
0017 FEDRAT/   10.0000                                  00006
0018   FROM/     SETPT                                  00008
                 0.0        0.0        0.0
0019 SPINDL/        ON                                  00010
0020   GOTO/        L1                                  00012
                 0.8125     0.1875    -0.1000
0021   GOTO/        L1                                  00014
                 4.1003     0.1875    -0.1000
0022   GOTO/        L2                                  00016
                 1.9434     3.4230    -0.1000
0023   GOTO/        L3                                  00018
                 0.7399    -0.1875    -0.1000
0024   GOTO/     SETPT                                  00020
                 0.0        0.0        0.0
0025 SPINDL/       OFF                                  00022
0026 ***** FINI *****                                   00024
```

# *DYNA Program to Machine a Triangle*

```
001 START INS 10
002  TD= 0.3750
003 FR XYZ=  10.0000
004 SETUP > ZCXYU
005 GO X  0.8125
```

```
006 GO Y  0.1875
007 SPINDLE ON
008 GO Z -0.1000
009 GO X  4.1003
010 GO X  1.9434
011    Y  3.4230
012 GO X  0.7399
013    Y -0.1875
014 GO Z  0.0000
015 GO X  0.0000
016 GO Y  0.0000
017 SPINDLE OFF
018 END
```

# Chapter 6

# Program Structure

This chapter presents a detailed description of the program CADDYN which serves as a post processor interface between CADAM NC and the DYNA milling machine. This program has been written in FORTRAN 77 and a listing is included in the Appendix B. A flowchart showing the structure of the program is shown in Fig. 3.

The output of the APT/AC processor is a file stored with filename and filetype, H1 and H1 respectively. This file is a listing file of the APT statements and the CLDATA for the part. This listing file is read in as input by a program CONVERT written in Fortran 77, which separates the APT listing and the CLDATA. The listing for the program CONVERT is included in the Appendix B. This program keeps reading the file H1 till it reaches the portion that has the CLDATA. It then starts writing this CLDATA to a file with filename and filetype, CADDYN and IN respectively. This file serves as input to the program CADDYN.

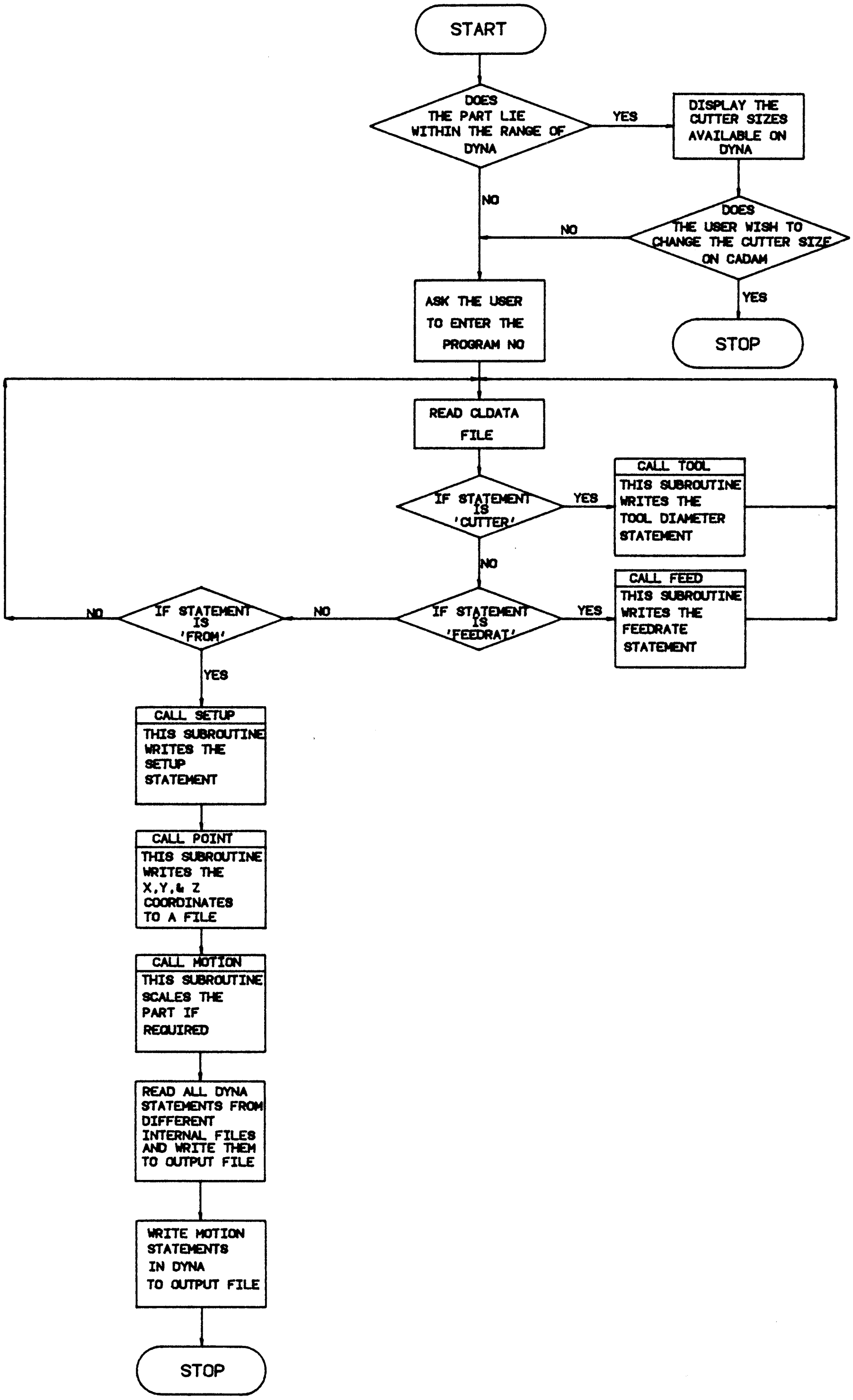


**Figure 3. Flowchart for Program CADDYN**

## *Program CADDYN*

The program CADDYN reads the CLDATA file as input. Before reading the input file, the range of the DYNA milling machine in all 3 directions X, Y, and Z is displayed. The user is then asked whether the part he wishes to machine lies within these ranges. If the answer is yes, the cutter sizes available on DYNA are displayed and the user is asked whether he wishes to go back and change the cutter size when creating the tool path on CADAM, if the answer is yes the program is terminated and if the answer is no the user is prompted to enter the program number. If the part does not lie within the range of DYNA's 3 axes the user is prompted to enter the program number, as the first line of any program to be down loaded to the DYNA machine has to be the program number line. Each record of the input file is read into a character string of eighty characters and this string is then analyzed to determine the type of statement and accordingly the subroutines are called. Every time a different type of statement is read, a different value for the counter KCOUNT is set to identify the type of statement. If the statement is a CUTTER statement, the TOOL subroutine is called, if the statement is a FEEDRATE statement, the FEED subroutine is called, and if the statement is a FROM statement, the SETUP subroutine is called.

After the SETUP subroutine, the subroutine POINT is called which reads in the CLDATA file till it reaches the FINI statement. All the above subroutines write different type of statements to different internal files. After this control is returned to the main program, where all the internal files created by the various subroutines are rewound. Then the main subroutine MOTION is called. This subroutine checks whether the part is to be scaled down so that it can machined on DYNA. If the part lies within the range of DYNA's 3 axes, the scale value is taken as 1.0 and the X, Y, and Z coordinate data is written into an array and passed to the main program. If the part does not lie within the range of DYNA's 3 axes, the scale value is determined, and the cutter size as well as the X, Y, and Z coordinate data are scaled down and written to an internal file and an array correspondingly. Control is then returned to the main program and the internal files created by the MOTION subroutine are rewound.

Following this the machining instructions in DYNA language are written to an output file. The output file has filename and filetype, CADDYN and OUT respectively. First, for every statement the value of the counter KCOUNT is checked to determine the type of statement that had been read. Accordingly a record is read from the internal file created for that particular type of statement, and this is then written to the output file. So the initial statements like TOOL DIAMETER, SETUP, FEEDRATE, SPINDLE ON, SPINDLE OFF are read from the internal file and written to the output file. After this the motion statements are written, for this the X, Y, and Z coordinate array passed by the MOTION subroutine is used. For the first motion, when the cutter moves from its load point to the point where it has to begin cutting, the motion is along one axis, so the instructions in DYNA are written for single axis motion for this path.

Next the X, Y, and Z values for every cutter position are compared with the values of the previous cutter position, to determine whether the motion is one axis, two axes, or three axes motion. According to this DYNA instructions for that particular movement of the cutter are written. Thus finally the program contains the instructions for the complete path of the cutter. If, in the middle of the path, the cutter is to be changed or the feedrate is to be changed, or the spindle is to be switched on or off, flags KFLAG1, KFLAG2, KFLAG3, and KFLAG4 are set to signal that these statements occur in the middle of the motion statements and accordingly the corresponding internal files are read and the information written to the output file. Each statement in the output file is preceeded by the statement number as required by the DYNA machine controller. In the end, the output file has the complete part program in DYNA language required to machine the part.

## *Subroutine TOOL*

The TOOL subroutine is called when the type of statement in the CLDATA file is a CUTTER statement. This subroutine is called every time a cutter is changed and a new cutter is to be installed on the machine. This subroutine takes the cutter statement from the input file and converts it to the tool diameter statement in DYNA language. This tool diameter statement is then written to an internal file which has unit number 13. The cutter statement in the CLDATA file contains information on the diameter of the cutter, the corner radius of the cutter, and the length of the cutter. This subroutine reads only the diameter and ignores the rest of the information as in DYNA only the tool diameter needs to be specified. A flowchart for subroutine TOOL is shown in Fig. 4.

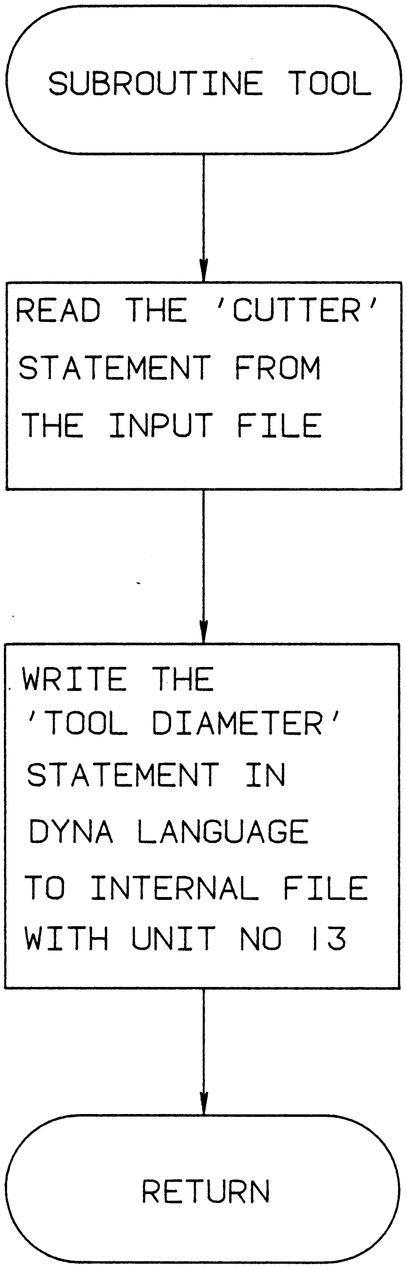


**Figure 4. Flowchart for Subroutine TOOL**

## *Subroutine FEED*

The FEED subroutine is called when the type of statement in the CLDATA file is a feedrate statement. This subroutine takes the feedrate statement from the input file and checks whether the feedrate lies in the range of feedrates that can be specified for the DYNA milling machine. This range is from 0.1 to 32.0 inches per minute (2.54 to 812.8 mm/min). If the feedrate lies outside this range, this subroutine prompts the user to specify a feedrate within the range and then reads the input value and writes the feedrate statement in DYNA language to an internal file which has unit number 14. This subroutine is called everytime the feedrate is changed in the program. Although when creating the tool path on CADAM, the feedrate can be specified different in all three directions and even when milling the part on DYNA the same can be achieved, this subroutine assumes that the feedrate specified is the same in all three directions. The user should specify a constant feedrate in all three directions, although the facility for changing the feedrate whenever desired is provided in the program. A flowchart for subroutine FEED is shown in Fig. 5.

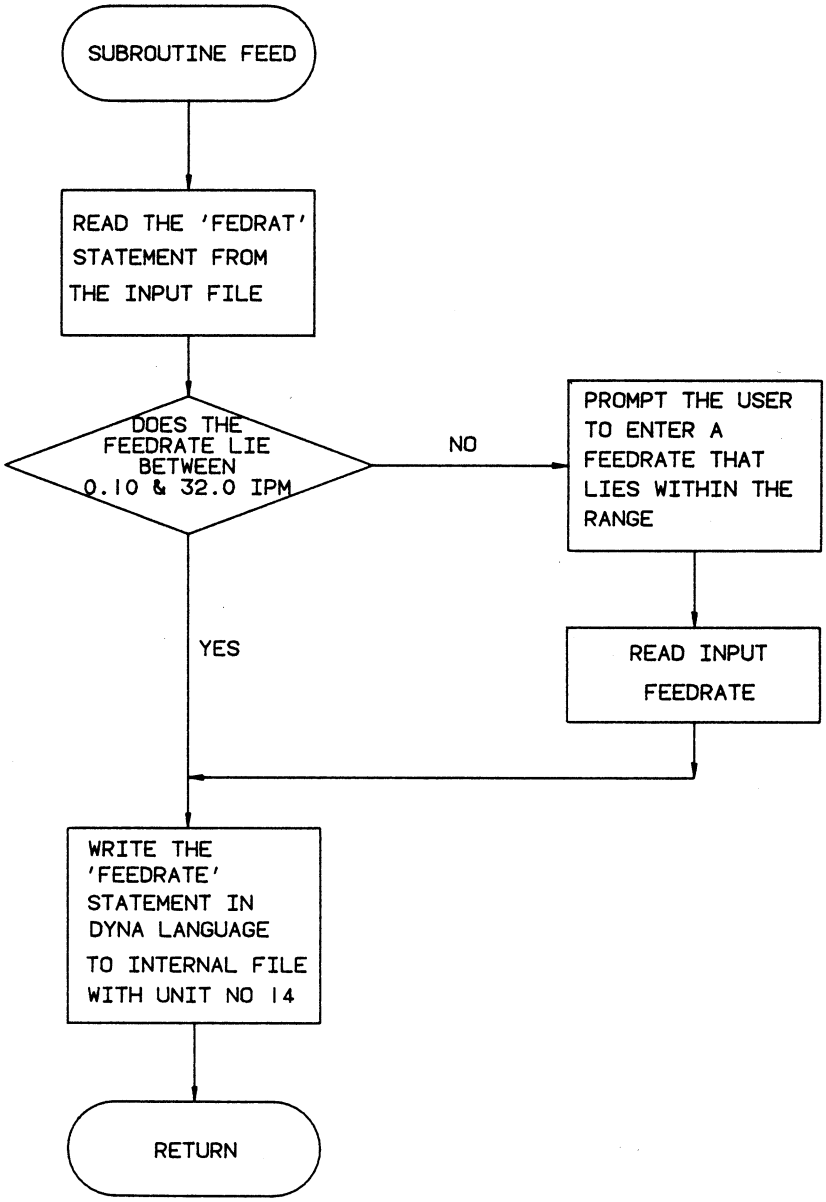


**Figure 5. Flowchart for Subroutine FEED**

## *Subroutine SPINON*

The SPINON subroutine is called when the type of statement in the CLDATA file is a SPINDLE ON statement. This subroutine is called every time the spindle is switched on in the program. It writes the 'spindle on' instruction in DYNA language to an internal file which has unit number 15. There is a dwell period of 4 seconds every time the spindle is switched on. This is provided for the spindle to achieve the specified speed to begin machining. Although the spindle speed can be specfied while creating the tool path on CADAM, on the DYNA machine there is no facility for adjusting the speed in the program. It has to be done manually. So, even if the speed of the spindle is specified, this subroutine ignores it and writes only a spindle on statement. After the spindle on statement, the dwell statement is written to incorporate a dwell period. A flowchart for subroutine SPINON is shown in Fig. 6.

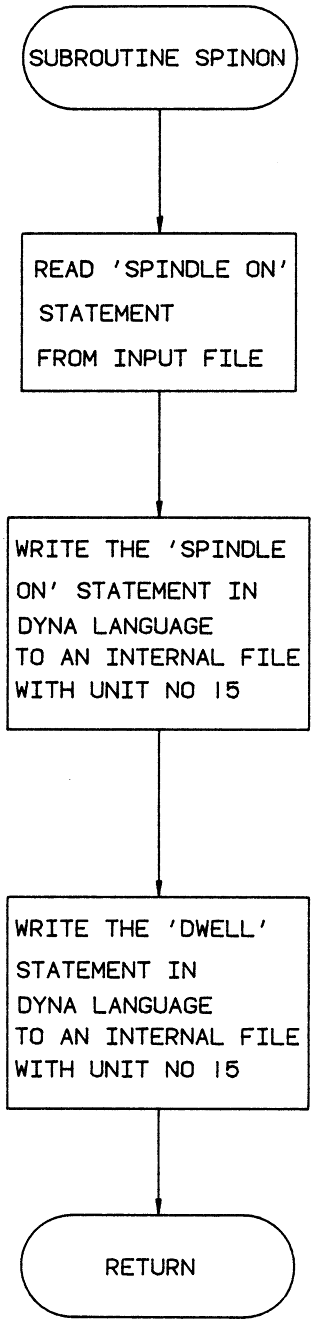


**Figure 6. Flowchart for Subroutine SPINON**

## *Subroutine SPIOFF*

The SPIOFF subroutine is called when the type of statement in the CLDATA file is a SPINDLE OFF statement. This subroutine is called every time the spindle is switched off in the program. It writes the spindle off instruction in DYNA language to an internal file which has unit number 16. The SPINDLE OFF statement is also written when the tool is being changed. The tool change statement in DYNA does not switch the spindle off, so the SPINDLE OFF statement is inserted after every tool change statement. A flowchart for subroutine SPIOFF is shown in Fig. 7.

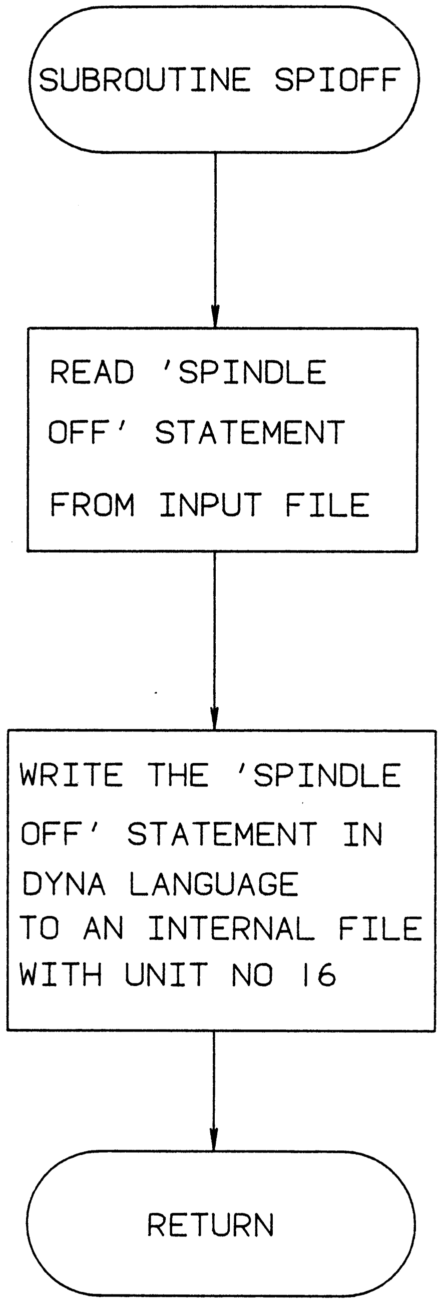


**Figure 7. Flowchart for Subroutine SPIOFF**

## *Subroutine SETUP*

The SETUP subroutine is called when the type of statement in the CLDATA file is a FROM statement. This subroutine takes the FROM statement in input file which contains coordinates of load point and converts it to the setup statement in the DYNA language. The setup statement allows the user to manually set the reference zero on the DYNA milling machine. The setup statement is written to an internal file which has unit number 17. In the subroutine MOTION the setup point is fixed as the reference zero from which all dimensions are referenced. So the initial choice of the setup point while creating the tool path on CADAM is very important and it is advised that the setup point should be the origin of the three axes, for producing good results. A Flowchart for subroutine SETUP is shown in Fig. 8.

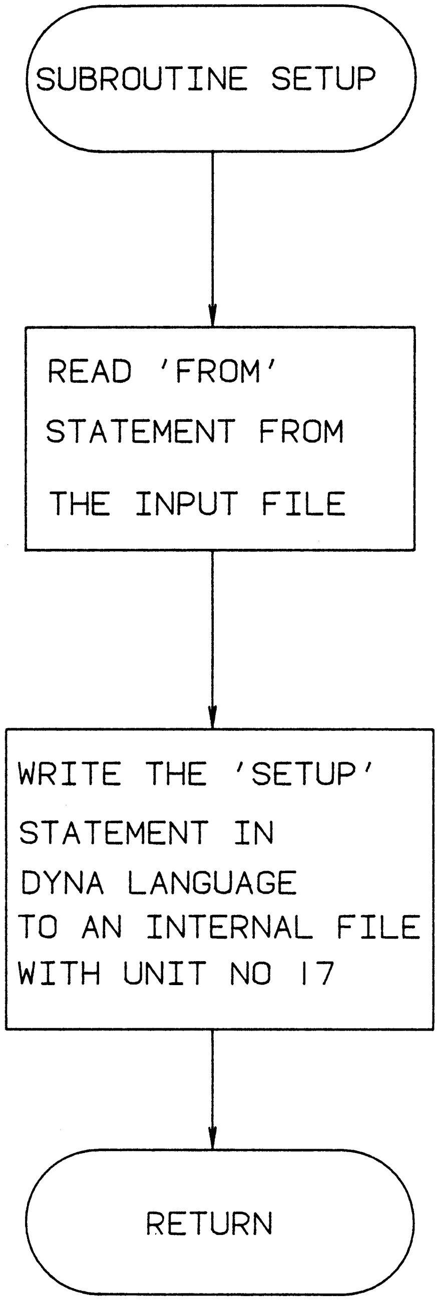


**Figure 8. Flowchart for Subroutine SETUP**

## *Subroutine POINT*

The POINT subroutine is called immediately after the SETUP subroutine. The subroutine POINT reads the input from the CLDATA file and separates the X, Y, and Z coordinate data from the other input statements. The POINT subroutine takes the X, Y, and Z coordinate data and writes it to an internal file which has unit number 18. This subroutine also checks for CUTTER, SPINDLE ON, and SPINDLE OFF statements in the CLDATA file. This subroutine reads in other statements like MCHTOL, END, LEADER, and STOP, but since these are not required in the DYNA program, no information regarding these statements is stored. Whenever it reads a statement other than an X, Y, and Z coordinate statement a flag is set. These flags KFLAG1, KFLAG2, KFLAG3, and KFLAG4 are used by the main program later on to determine the type of statement. This subroutine also determines the number of cutters used in the program and passes this information to the MOTION subroutine. A flowchart for subroutine POINT is shown in Fig. 9.

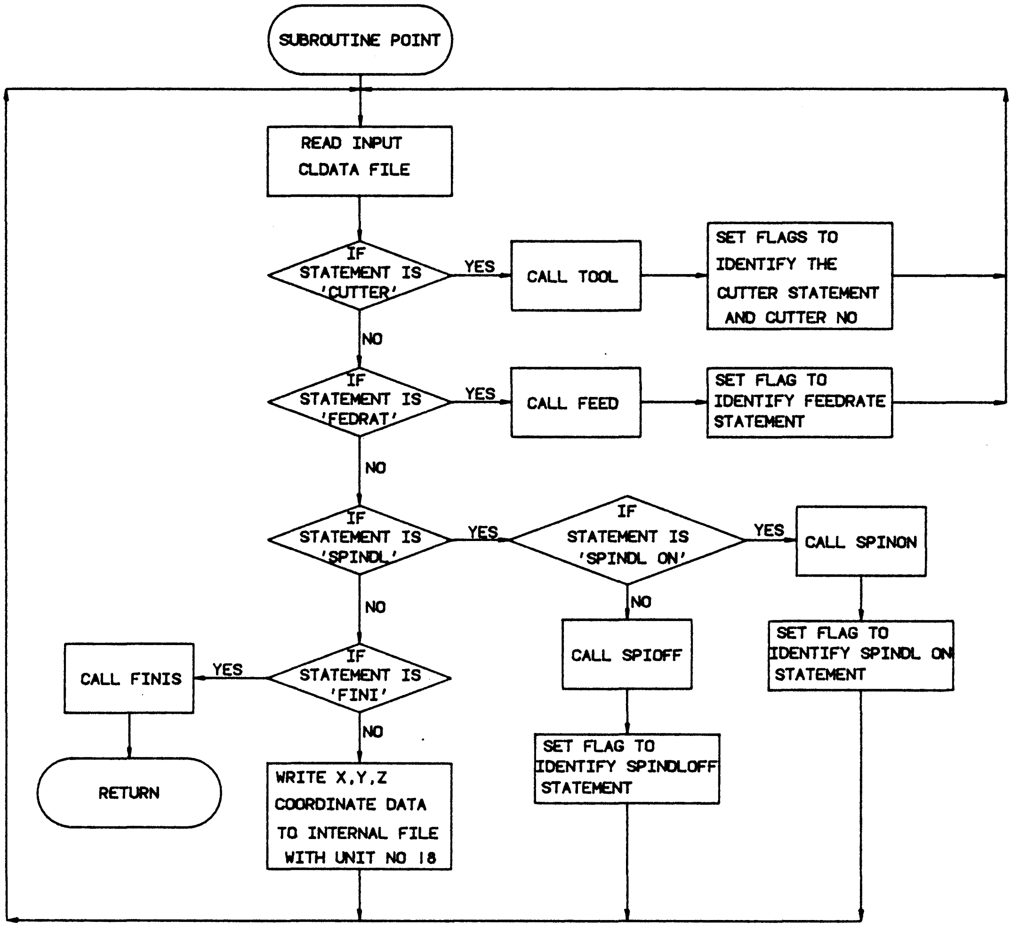


Figure 9. Flowchart of Subroutine POINT

## *Subroutine MOTION*

The MOTION subroutine is the main subroutine that works with the X, Y, and Z data. In this subroutine the X, Y, and Z data is read in from the internal file created by the POINT subroutine. First the origin is shifted to the setup point and all the coordinate data is transformed with respect to this origin. After this, the minimum and maximum values of X, Y, and Z coordinates are determined and the range of travel in the X, Y, and Z direction is calculated. If these ranges are greater than the ranges of DYNA in the three directions, the approximate scale value in all three directions is calculated. The maximum of all three approximate scale values is then taken and used to determine the final value of the scale for the part. After this the tool diameters are read in from the internal file created by the TOOL subroutine. If the part lies within the range of the DYNA machine no further calculations are done and control is returned to the main program.

If the part is to be scaled down, the first tool diameter is taken and divided by the approximate scale factor to determine the scaled cutter size. At this point the CUTTER subroutine is called which checks whether the scaled cutter size is available on DYNA. If the size lies between two cutter sizes available on DYNA, the minimum of the two is taken and the user at this point is asked whether he wishes to use this cutter. If the answer is yes, the new data is generated according to this cutter. If the answer is no and the user still wants to use the original scaled cutter then he is reminded that the data will be generated according to that cutter, but the part will have to be machined using a cutter smaller than that to maintain the geometry of the part. The final scale value is then determined from the cutter that will be used and the X, Y, and Z coordinate values are scaled down accordingly.

If the program contains more than one cutter, then the scale value is determined with respect to the first cutter and all the other cutters are scaled down with respect to this scale value. In one part program a maximum of eight cutters are allowed . The final scale value is displayed and the user is asked whether he wishes to use this scale, if not the user is asked to enter a new scale and the cutters and the X, Y, and Z coordinate data are scaled down according to this scale.

This subroutine finally passes the X, Y, and Z data in an array form to the main program which uses it to write the final output file that contains the machining instructions in DYNA language. A flowchart for subroutine MOTION is shown in Fig. 10.

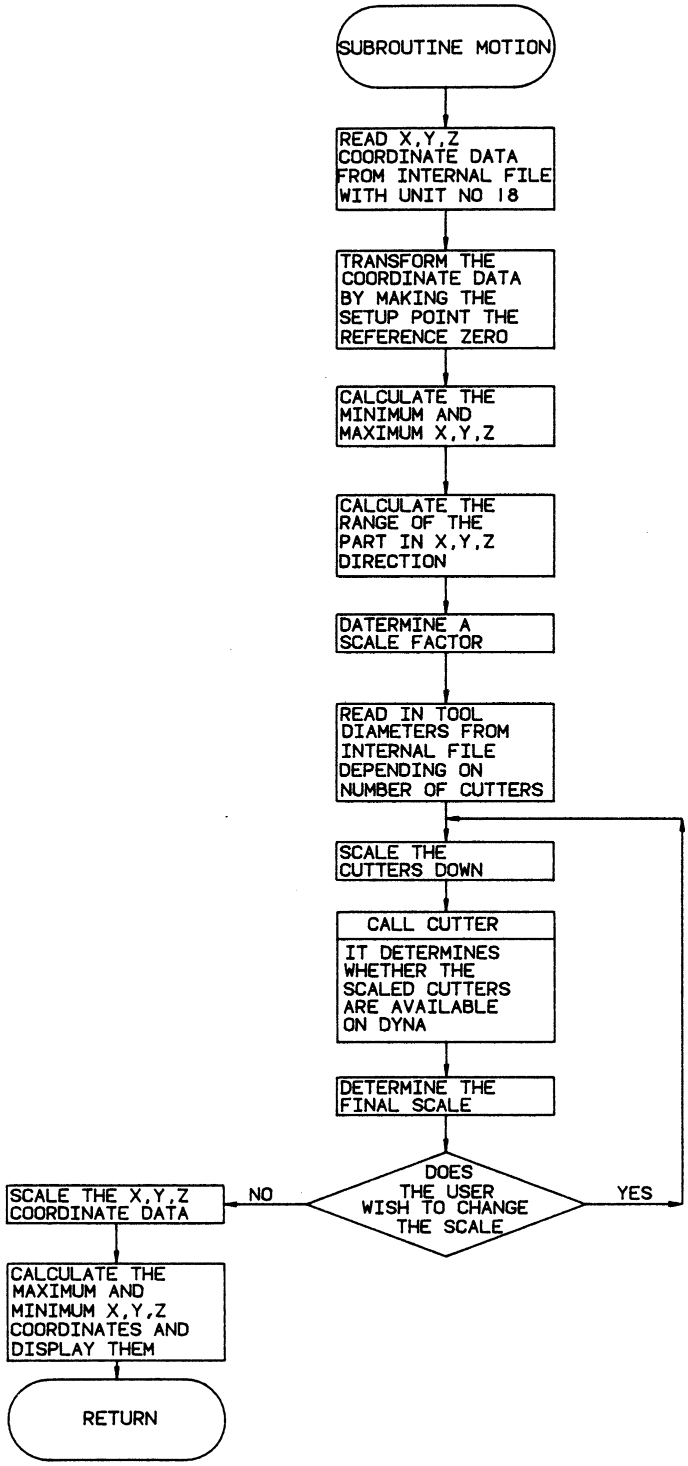


**Figure 10. Flowchart for Subroutine MOTION**

## *Subroutine CUTTER*

The CUTTER subroutine takes the scaled value of the cutter size and checks whether the size is available on DYNA. If the cutter is available on DYNA then the user is prompted with a message saying that the cutter is available on DYNA. If the size lies between two cutter sizes available on DYNA then the smaller of the two is taken and the user is prompted for acceptance. If the answer is no, the user is informed that the data will be generated according to the orginal scaled cutter, but when machining the part the user will have to use a cutter smaller than the scaled size. If the answer is yes, the data is generated according to the cutter available on DYNA. This subroutine does this for eight different cutters. A flowchart for subroutine CUTTER is shown in Fig. 11.

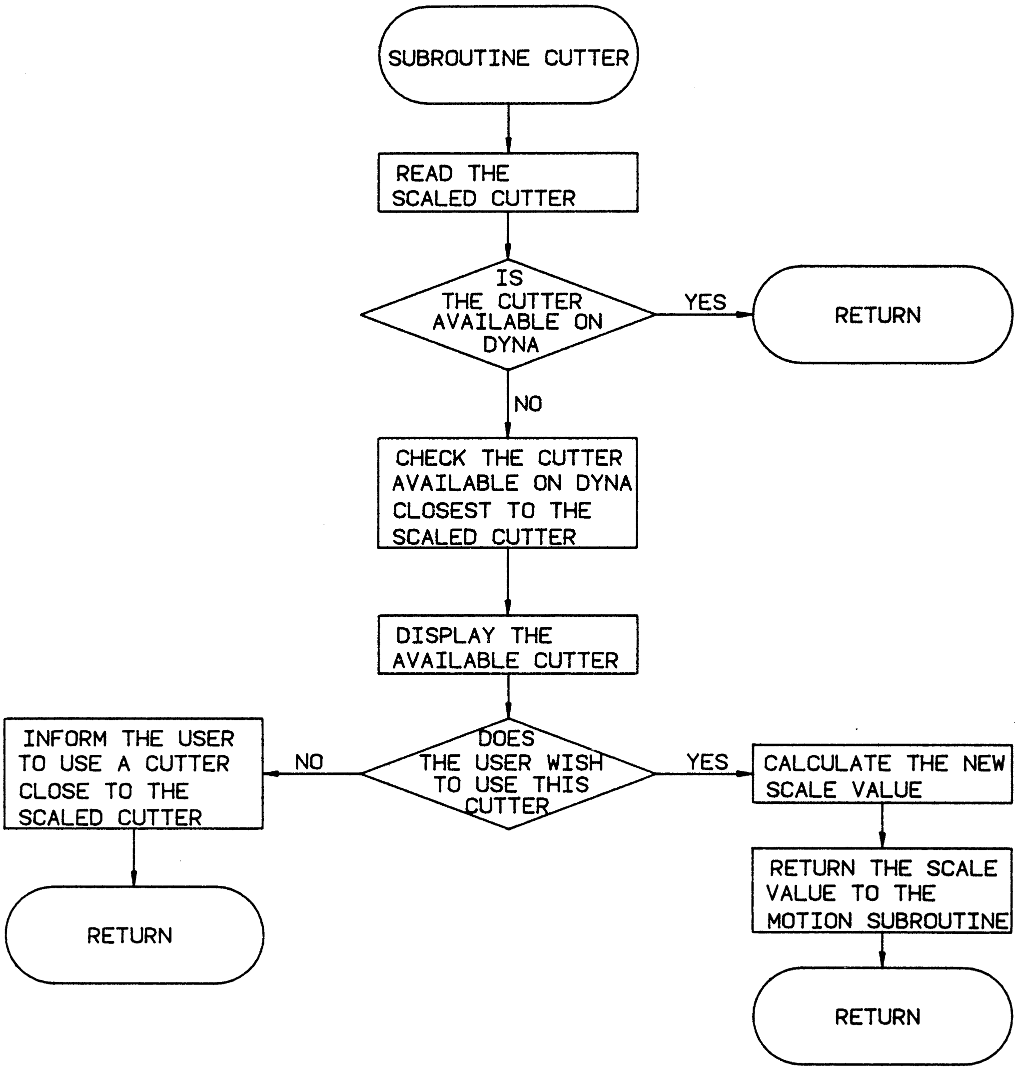


**Figure 11. Flowchart for Subroutine CUTTER**

## *Subroutine FINIS*

This subroutine is called when the statement in the CLDATA file is a FINI statement. This subroutine takes the FINI statement from the input file and converts it to the END statement for the DYNA milling machine and writes it to an internal file which has unit number 20. A flowchart for subroutine FINIS is shown in Fig. 12.

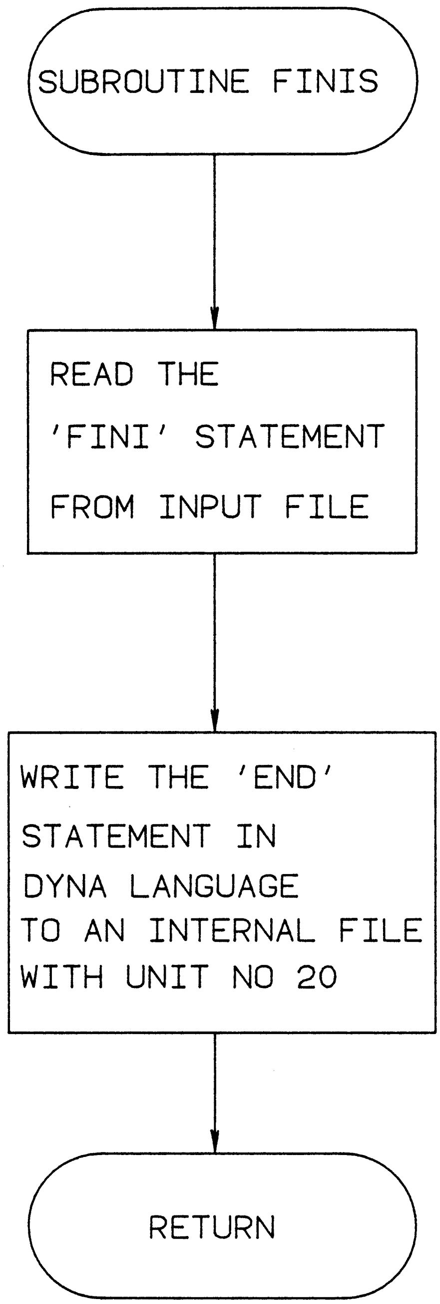


**Figure 12. Flowchart for Subroutine FINIS**

# Chapter 7

# Program Modification

The program CADDYN was developed without using any of the subroutines preprogrammed into the DYNA machine controller. For milling circular surfaces, the program CADDYN writes out the complete point data from the CLDATA file. The advantage of this is that, all three axes can be controlled simultaneously, as for a circle projected on to a sloping floor. The size of the output file for such an operation is very large to be downloaded to the machine controller. This may become a problem since the controller does not accept more than 900 lines of code for a particular part program.

The solution to this problem was achieved by modifying the program CADDYN and writing a new program MODDYN. The program MODDYN is different from program CADDYN in only one respect, an extra subroutine CIRCLE has been written which uses the circle routine preprogrammed into the DYNA machine controller. This reduces the size of the output file considerably and helps in the machining of complex surfaces. The program CADDYN still remains the original work in this thesis and has a clear advantage over the program MODDYN in 3 axes milling. A listing of the program MODDYN is included in the appendices.

## *Subroutine CIRCLE*

The DYNA machine controller has a routine preprogrammed into its memory, for milling circular surfaces. The input to this routine is in form of the center of the circle and the angle of the arc to be machined. The Subroutine CIRCLE written in the program MODDYN reads the circular surface data from the CLDATA file and writes the necessary statements in DYNA language for input to the circle routine of the DYNA machine.

The CIRCLE subroutine reads the data from the input file and calculates the angle of the arc to be machined. It then writes the center of the circle and the angle of the arc to different internal files from where they are read and written to the output file. The point data is not written and hence the size of the output file is decreased. In other aspects the program MODDYN is similar to the program CADDYN. A flowchart for subroutine CIRCLE is shown in Fig. 13.

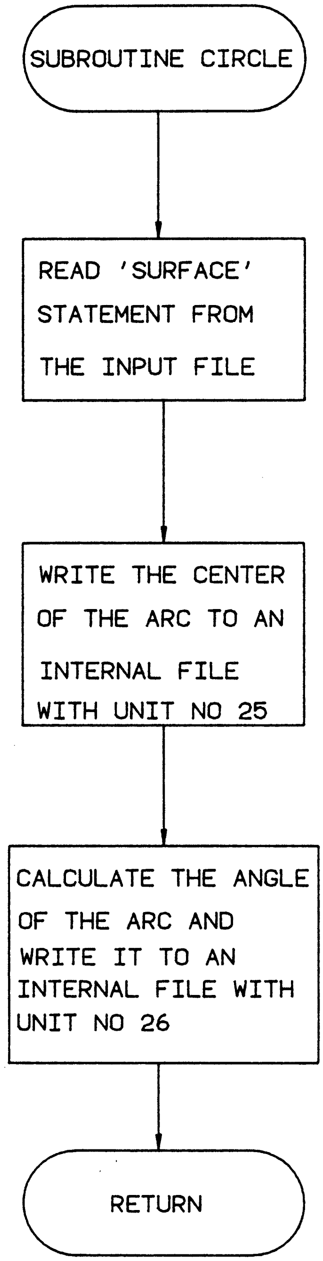


**Figure 13. Flowchart for Subroutine CIRCLE**

# Chapter 8

# Conclusions and Recommendations

The interface between a CAD system and a CAM machine was sucessfully developed in this study. The program CADDYN which was developed accepts the CLDATA file as input and converts it to NC instructions for the DYNA milling machine. The program CADDYN post-processes different types of statements as described in the program structure chapter.

During the course of this study and while developing the program, the aim was to develop a post-processor to serve as an interface between a CAD system and a CAM machine. This was primarily to understand the integration process between CAD and CAM, as well as to link CADAM NC to the DYNA milling machine. Two orthographic views of a part were drawn on CADAM and then NC instructions were created and post-processed to generate the DYNA program to machine the part. The parts were machined using wax as the raw material.

Future work should be directed in modifying the program CADDYN to accomodate more numerical control functions, as well as to use the advanced NC capabilities of the CADAM system. Some of this include machine tolerance, coolant on, coolant off, and transformation of cuts specifications. At present the feedrate specified has to be constant in the directions X, Y, and Z. Future work could concentrate on specifying different feedrates in all three directions. The spindle on statement could also be modified to include the speed specification, if a way to adjust the speed of the spindle on DYNA through the program can be developed. Another area of potential future work is the transformation of cuts from one point to another as it would help in machining pockets using CADAM NC and its advanced pocket routines.

# References

1. Groover, M. P., and Zimmers, E. W., *CAD/CAM: Computer-Aided Design and Manufacturing,* Prentice Hall Inc., Englewood Cliffs, NJ, p134.

2. Krouse, J. K., "Engineering without paper," *High Technology,* March, 1986.

3. Nilson, E. N., "Integrating CAD and CAM - Future Directions," *CAD/CAM: Meeting todays productivity challenge,* CASA of SME, Dearbon, Michigan, pp 115-135.

4. Wilson, J. F., "Integrated Manufacturing and the Central Role of CADCAM," *Effective CADCAM 1985,* ImechE conference publications 1985, London, pp 99-106.

5. Hannam, R. G., and Miller, A. S., "Linking Design and Manufacturing Through CADCAM - a case study on jig fixture design," *Effective CADCAM 1985,* ImechE conference publications 1985, London, pp 117-124.

6. Twinstra, P. A., "History of APT," *Technical Paper,* Society of Manufacturing Engineers, Dearborn, Michigan, 1975.

7. American National Standard Institute Inc., *Programming Language APT,* January, 1977.

8. IIT Research Institute, *APT Part Programming,* Mcgraw-Hill Book Company, 1967.

9. IIT Research Institute, *APT Encyclopedia,* Univac Division of Sperry Rand Corporation, 1963.

10. Kendall, L. A., "A Computer Paced Learning Module APT (Automatic Programmed Tools)," *Proceedings of the second international computer engineering conferences,* San Diego, California, August 15-19, 1982, vol 4, pp 269-272.

11. Kochan, D., "CAM Developments in Computer-Integrated Manufacturing," IFIP-State of the Art Reports, Springer-Verlag, October, 1985.

12. Subramanian, M. L., "CLFILE Manipulator," *Proceedings of the second international computer engineering conferences,* San Diego, California, August 15-19, 1982, vol 1, pp 179-184.

13. Zapomueel, H., "MAPROS Preprocessing - an Economical Solution for the Integration of NC Drawing and Manufacture of Mechanical Workpieces," *Proceedings*

*of the 3rd international IFIP/IFAC conference on programming languages for machine tools, PROLOMAT 76,* Stirling, Scotland, June, 1976, pp 81-90.

14. Kishi, H., Nagai, K., and Hatta, T., "The Application of OKISURF," *Proceedings of the 3rd international IFIP/IFAC conference on programming languages for machine tools, PROLOMAT 76,* Stirling, Scotland, June, 1976, pp 153-166.

15. Gossling, T. H., and Warner, K. N., "UNILEX - an Interface Between Geometric Data and Numerical Control," *Proceedings of the 3rd international IFIP/IFAC conference on programming languages for machine tools, PROLOMAT 76,* Stirling, Scotland, June, 1976, pp 449-460.

16. Chan, B. T. F., "ROMAPT: A New Link between CAD and CAM," *Computer-Aided Design,* Butterworths, September, 1982, vol 14, no 5, pp 261-266.

17. Brooks, B., and Riley, D. R., "NCFILE: Drafting to N/C Milling for Instruction," *Proceedings of the 1984 international computers in engineering conference and exhibit,* Las Vegas, Nevada, August 12-15, 1984, pp 73-77.

18. Kuttner, B. C., Majcher, D. S., and Snedecor, P. B., "Systematic Processing: an Approach to Fully Automatic NC Tool Path Generation," *AUTOFACT'85, Conference Proceedings,* Detroit, Michigan, November 4-7, 1985, pp 18.7-19.1.

19. Bobrow, J. E., "NC Machine Tool Path Generation from CSG Part Representations," *Computer-Aided Design,* Butterworths, March, 1985, vol 17, no 2.

20. Phillips, R. H., and Mouleeswaran, C. B., "An Artificial Intelligence Approach to Integrating CAD and CAM Through Generative Process Planning," *Proceedings of the 1984 international computers in engineering conference and exhibit,* Las Vegas, Nevada, August 12-15, 1984, pp 459-463.

21. CADAM Inc., *Computer Graphics Augmented Design and Manufacturing System (CADAM) General Information Manual,* IBM, G320-6667-0, January, 1982.

22. CADAM Inc., *Introduction to CADAM Numerical Control Programming,* IBM, December, 1982.

23. IBM, *System/370 Automatically Programmed Tool - Advanced Contouring Numerical Control Processor, General Information,* GH20-1423-3, September, 1986.

24. IBM, *System/370 Automatically Programmed Tool - Advanced Contouring Numerical Control Processor, Program Reference Manual,* SH20-1414-2, Julyl, 1985.

25. IBM, *System/370 Automatically Programmed Tool - Advanced Contouring Numerical Control Processor, Operations guide,* SH20-1413-3, July, 1985.

26. DYNA Electronics Inc., *DM2400/ 2200 Programming Manual,* Santa Clara, CA.

# Appendix A

# EXAMPLE

The following pages present an example part, a bellcrank, which was drawn on CADAM, and for which the cutter path was also generated on CADAM. A listing of the CLDATA file and the output file is also included. Fig. 14 shows the orthographic views of the part and Fig. 15 shows the cutter path to machine the part. Fig. 16 shows a photograph of the final part.

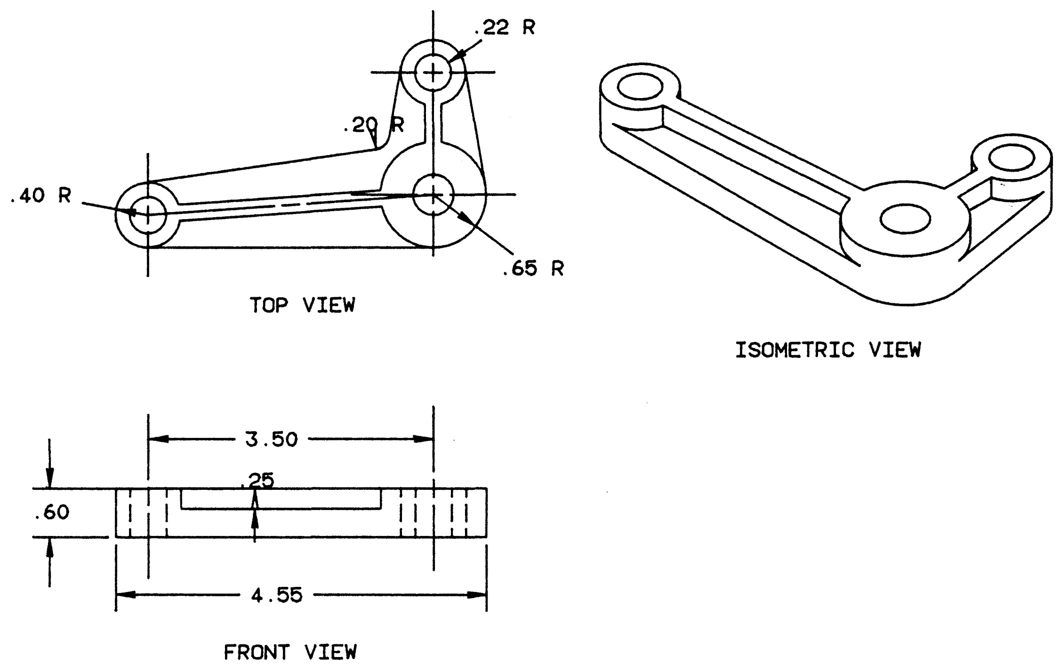


Figure 14. Orthographic Views of the Example Part

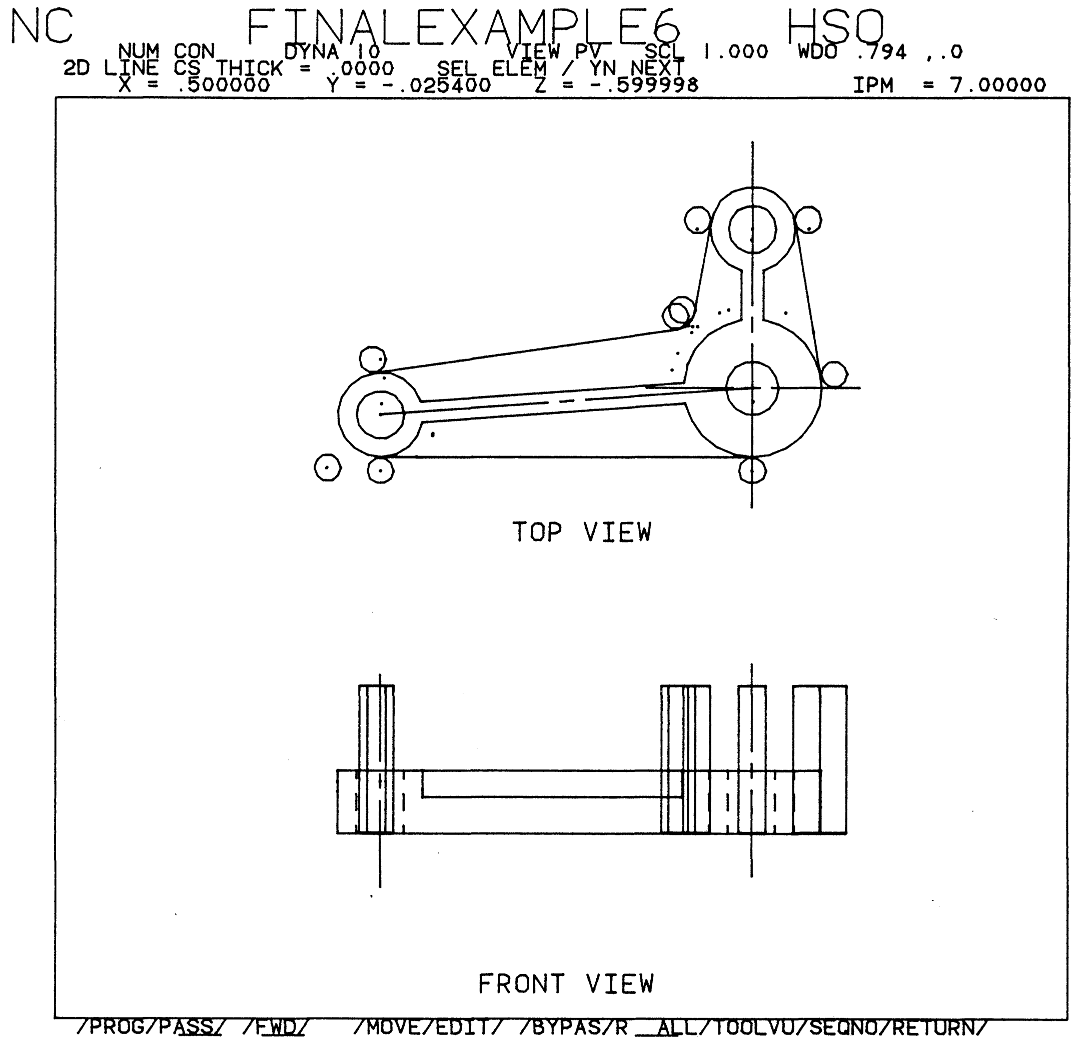


Figure 15. Cutter Path for the Example Part

Figure 16. Final Machined Part

# CLDATA FILE

```
1                                                                                PAGE 0001
 IBM S/370 APT-AC N/C PROGRAM  VERSION=1.41.000     DATE=05/22/87 TIME=12:10:23
0
 ***** DREAD ENCOUNTERED CSF ERROR *****
 ***** DREAD ENCOUNTERED CSF ERROR *****
 ***** DREAD ENCOUNTERED CSF ERROR *****
                                             ... BEGIN TRANSLATION PHASE... (SECTION 1)
0
 0001 PARTNO  FINALEXAMPLE6   HSO
 0002 PPRINT  HSO     NO NAME       NC          87142   121009
 0003 PPRINT  --NO MACHINE SPECIFIED
 0004 CALL/GRAPT,SEC1, 101, 0,       1
        CADAM-APT INTERFACE *** R20.1.1  ***
 0005   FINI


    NO DIAGNOSTICS ELICITED DURING TRANSLATION PHASE
     5 N/C SOURCE RECORDS (SYSIN)


        SECTION 1 ELAPSED CPU TIME IN MIN/SEC IS 0000/00.3766
        SECTION 2 ELAPSED CPU TIME IN MIN/SEC IS 0000/00.4366
1                                                                                PAGE 0002
 ....SECTION 3....
0
 ISN                                                               LABEL  REC M     CARD
 0001 PARTNO/                                                            00002
        FINALEXAMPLE6   HSO
 0002 PPRINT/                                                            00004
        HSO      NO NAME         NC          87142    121009
 0003 PPRINT/                                                            00006
        --NO MACHINE SPECIFIED
 0003  INTOL/     0.0010     0.0010     0.0010     0.0010                00007
 0003 OUTTOL/     0.0010     0.0010     0.0010     0.0010                00008
 0007 SEQNO /     0.0                                                    00010 GRAPHICS
 0009 PPRINT/                                                            00012 GRAPHICS
                                                                            120
      787142
 0009  UNITS/INCHES                                                      00013 GRAPHICS
 0012 PPRINT/                                                            00015 GRAPHICS
      **** CUTTER DIMENSIONS-DIAM = .2500      CORNER RAD =.0100
 0012 CUTTER/     0.2500     0.0100                                      00016 GRAPHICS
 0012 MCHTOL/     0.0010     0.0010                                      00018 GRAPHICS
 0012   FROM/      POINT                                                 00019 GRAPHICS
                  0.0        0.0        1.0000
 0014 FEDRAT/     7.0000                                                 00021 GRAPHICS
 0014   GOTO/     SRFMIL                                                 00023 GRAPHICS
                  4.0000    -0.0254    -0.6000
 0016 SPINDL/        ON                                                  00025 GRAPHICS
 0016   GOTO/                                                            00027 GRAPHICS
                  4.0499    -0.0234    -0.6000
 0016                                                                    00028 GRAPHICS
 0016                                                                    00029 GRAPHICS
 0016 SURFACE                           CIRCLE     DS(     ON)           00031 GRAPHICS
                  4.0000     0.7500     0.0
                  0.0        0.0        1.0000     0.7750
 0016   GOTO/                                                            00032 GRAPHICS
```

```
                4.1038     -0.0190    -0.6000
                4.2098      0.0029    -0.6000
                4.3116      0.0393    -0.6000
                4.4074      0.0895    -0.6000
                4.4952      0.1526    -0.6000
                4.5735      0.2272    -0.6000
                4.6406      0.3120    -0.6000
                4.6953      0.4053    -0.6000
                4.7364      0.5054    -0.6000
                4.7633      0.6101    -0.6000
                4.7753      0.7176    -0.6000
                4.7723      0.8257    -0.6000
                4.7642      0.8792    -0.6000
 0017   GOTO/   SRFMIL                                              00034 GRAPHICS
                4.5177      2.3375    -0.6000
 0018   GOTO/                                                       00036 GRAPHICS
                4.5099      2.3750    -0.6000
 0018                                                               00037 GRAPHICS
 0018                                                               00038 GRAPHICS
1                                                                       PAGE 0003
 0018 SURFACE                          CIRCLE     DS(     ON)       00040 GRAPHICS
                4.0000      2.2500     0.0
                0.0         0.0        1.0000      0.5250
 0018   GOTO/                                                       00041 GRAPHICS
                4.4983      2.4183    -0.6000
                4.4625      2.5006    -0.6000
                4.4131      2.5756    -0.6000
                4.3517      2.6411    -0.6000
                4.2801      2.6952    -0.6000
                4.2003      2.7364    -0.6000
                4.1147      2.7633    -0.6000
                4.0258      2.7754    -0.6000
                3.9360      2.7721    -0.6000
                3.8482      2.7536    -0.6000
                3.7648      2.7205    -0.6000
                3.6882      2.6736    -0.6000
                3.6207      2.6145    -0.6000
                3.5643      2.5447    -0.6000
                3.5205      2.4663    -0.6000
                3.4907      2.3816    -0.6000
                3.4823      2.3375    -0.6000
 0019   GOTO/   SRFMIL                                              00043 GRAPHICS
                3.3392      1.4907    -0.6000
 0020   GOTO/                                                       00045 GRAPHICS
                3.3355      1.4768    -0.6000
 0020                                                               00046 GRAPHICS
 0020                                                               00047 GRAPHICS
 0020 SURFACE                          CIRCLE     DS(     ON)       00049 GRAPHICS
                3.2653      1.5032     0.0
                0.0         0.0        1.0000      0.0750
 0020   GOTO/                                                       00050 GRAPHICS
                3.3305      1.4643    -0.6000
                3.3129      1.4440    -0.6000
                3.2892      1.4311    -0.6000
                3.2759      1.4290    -0.6000
 0021   GOTO/   SRFMIL                                              00052 GRAPHICS
                0.4254      1.0197    -0.6000
 0022   GOTO/                                                       00054 GRAPHICS
                0.3877      1.0129    -0.6000
 0022                                                               00055 GRAPHICS
 0022                                                               00056 GRAPHICS
 0022 SURFACE                          CIRCLE     DS(     ON)       00058 GRAPHICS
                0.5000      0.5000     0.0
                0.0         0.0        1.0000      0.5250
 0022   GOTO/                                                       00059 GRAPHICS
                0.3438      1.0023    -0.6000
                0.2600      0.9681    -0.6000
```

```
                  0.1834      0.9201     -0.6000
                  0.1162      0.8596     -0.6000
                  0.0602      0.7886     -0.6000
                  0.0173      0.7090     -0.6000
                 -0.0114      0.6232     -0.6000
                 -0.0249      0.5338     -0.6000
                 -0.0229      0.4434     -0.6000
1                                                                      PAGE 0004
                 -0.0055      0.3547     -0.6000
                  0.0268      0.2703     -0.6000
                  0.0732      0.1926     -0.6000
                  0.1321      0.1240     -0.6000
                  0.2019      0.0666     -0.6000
                  0.2806      0.0220     -0.6000
                  0.3657     -0.0086     -0.6000
                  0.4548     -0.0241     -0.6000
                  0.5000     -0.0250     -0.6000
 0023    GOTO/    SRFMIL                                           00061 GRAPHICS
                  4.0000     -0.0250     -0.6000
 0024    GOTO/     POINT                                           00063 GRAPHICS
                  4.0000     -0.0250      1.0000
 0025    GOTO/    SRFMIL                                           00065 GRAPHICS
                  4.0000      0.6246     -0.6000
 0026    GOTO/                                                     00067 GRAPHICS
                  3.9813      0.6264     -0.6000
 0026                                                              00068 GRAPHICS
 0026                                                              00069 GRAPHICS
 0026 SURFACE                             CIRCLE      DS(     ON)  00071 GRAPHICS
                  4.0000      0.7500      0.0
                  0.0         0.0         1.0000      0.1250
 0026    GOTO/                                                     00072 GRAPHICS
                  3.9606      0.6303     -0.6000
                  3.9234      0.6499     -0.6000
                  3.8947      0.6808     -0.6000
                  3.8778      0.7193     -0.6000
                  3.8745      0.7613     -0.6000
                  3.8852      0.8020     -0.6000
                  3.9087      0.8369     -0.6000
                  3.9424      0.8621     -0.6000
                  3.9826      0.8748     -0.6000
                  4.0036      0.8750     -0.6000
 0026 SURFACE                             CIRCLE      DS(     ON)  00074 GRAPHICS
                  4.0000      0.7500      0.0
                  0.0         0.0         1.0000      0.1250
 0026    GOTO/                                                     00075 GRAPHICS
                  4.0247      0.8736     -0.6000
                  4.0640      0.8586     -0.6000
                  4.0962      0.8315     -0.6000
                  4.1176      0.7952     -0.6000
                  4.1260      0.7540     -0.6000
                  4.1203      0.7123     -0.6000
                  4.1011      0.6748     -0.6000
                  4.0707      0.6457     -0.6000
                  4.0324      0.6282     -0.6000
                  4.0115      0.6255     -0.6000
 0027    GOTO/     POINT                                           00077 GRAPHICS
                  4.0115      0.6250      1.0000
 0028    GOTO/    SRFMIL                                           00079 GRAPHICS
                  4.0000      2.1496     -0.6000
 0029    GOTO/                                                     00081 GRAPHICS
                  4.0167      2.1514     -0.6000
 0029                                                              00082 GRAPHICS
1                                                                      PAGE 0005
 0029                                                              00083 GRAPHICS
 0029 SURFACE                             CIRCLE      DS(     ON)  00085 GRAPHICS
                  4.0000      2.2500      0.0
                  0.0         0.0         1.0000      0.1000
```

Patron,

Page 61 is missing and not available via ILL.

```
                      3.2759     0.4727    -0.2500
0040  GO---/          SRFMIL                                          00112 GRAPHICS
                      3.2852     0.4495    -0.2500
                      3.2952     0.4266    -0.2500
                      3.3125     0.3914    -0.2500
                      3.3316     0.3570    -0.2500
                      3.3523     0.3237    -0.2500
                      3.3747     0.2915    -0.2500
                      3.3987     0.2605    -0.2500
                      3.4242     0.2307    -0.2500
                      3.4512     0.2022    -0.2500
                      3.4796     0.1752    -0.2500
                      3.5093     0.1496    -0.2500
                      3.5403     0.1255    -0.2500
                      3.5725     0.1031    -0.2500
                      3.6058     0.0823    -0.2500
                      3.6400     0.0632    -0.2500
                      3.6752     0.0459    -0.2500
                      3.7113     0.0304    -0.2500
                      3.7480     0.0167    -0.2500
                      3.7854     0.0049    -0.2500
                      3.8234    -0.0050    -0.2500
                      3.8618    -0.0130    -0.2500
                      3.9006    -0.0190    -0.2500
                      3.9361    -0.0228    -0.2500
                      3.9718    -0.0249    -0.2500
                      3.9859    -0.0253    -0.2500
                      4.0000    -0.0254    -0.2500
0077 TRACUT/          VU  PV                                          00114 GRAPHICS
                      1.0000     0.0        0.0        0.0
                      0.0        1.0000     0.0        0.0
                      0.0        0.0        1.0000     0.0
0078    GOTO/          POINT                                          00116 GRAPHICS
                      4.0000    -0.0250     1.0000
0079    GOTO/         SRFMIL                                          00118 GRAPHICS
                      3.2438     0.9216    -0.2500
0080    GOTO/                                                         00120 GRAPHICS
                      3.2568     0.9696    -0.2500
1                                                                          PAGE 0007
0080                                                                  00121 GRAPHICS
0080                                                                  00122 GRAPHICS
0080 SURFACE                                CIRCLE     DS(     ON)    00124 GRAPHICS
                      4.0000     0.7500     0.0
                      0.0        0.0        1.0000     0.7750
0080    GOTO/                                                         00125 GRAPHICS
                      3.2732     1.0219    -0.2500
                      3.3187     1.1215    -0.2500
                      3.3778     1.2138    -0.2500
                      3.4493     1.2968    -0.2500
                      3.5318     1.3689    -0.2500
                      3.6236     1.4286    -0.2500
                      3.7230     1.4749    -0.2500
                      3.7751     1.4917    -0.2500
0081    GOTO/          POINT                                          00127 GRAPHICS
                      3.7750     1.4920     1.0000
0082    GOTO/         SRFMIL                                          00129 GRAPHICS
                      4.2250     1.4920    -0.2500
0083    GOTO/                                                         00131 GRAPHICS
                      4.2722     1.4756    -0.2500
0083                                                                  00132 GRAPHICS
0083                                                                  00133 GRAPHICS
0083 SURFACE                                CIRCLE     DS(     ON)    00135 GRAPHICS
                      4.0000     0.7500     0.0
                      0.0        0.0        1.0000     0.7750
0083    GOTO/                                                         00136 GRAPHICS
                      4.3208     1.4566    -0.2500
                      4.4128     1.4071    -0.2500
```

```
                4.4973     1.3457    -0.2500
                4.5727     1.2736    -0.2500
                4.6378     1.1920    -0.2500
                4.6914     1.1023    -0.2500
                4.7324     1.0063    -0.2500
                4.7602     0.9057    -0.2500
                4.7742     0.8022    -0.2500
                4.7750     0.7500    -0.2500
 0084    GOTO/   POINT                                            00138 GRAPHICS
                4.7754     0.7500     1.0000
 0085    GOTO/  SRFMIL                                            00140 GRAPHICS
                4.2250     1.7751    -0.2500
 0086    GOTO/                                                    00142 GRAPHICS
                4.2590     1.7933    -0.2500
 0086                                                             00143 GRAPHICS
 0086                                                             00144 GRAPHICS
 0086 SURFACE                         CIRCLE     DS(     ON)      00146 GRAPHICS
                4.0000     2.2500     0.0
                0.0        0.0        1.0000     0.5250
 0086    GOTO/                                                    00147 GRAPHICS
                4.2932     1.8133    -0.2500
                4.3555     1.8623    -0.2500
                4.4096     1.9200    -0.2500
                4.4545     1.9853    -0.2500
                4.4891     2.0565    -0.2500
                4.5126     2.1321    -0.2500
1                                                                      PAGE 0008
                4.5245     2.2104    -0.2500
                4.5250     2.2500    -0.2500
 0087    GOTO/   POINT                                            00149 GRAPHICS
                4.5254     2.2500     1.0000
 0088    GOTO/  SRFMIL                                            00151 GRAPHICS
                3.7750     1.7752    -0.2500
 0089    GOTO/                                                    00153 GRAPHICS
                3.7408     1.7935    -0.2500
 0089                                                             00154 GRAPHICS
 0089                                                             00155 GRAPHICS
 0089 SURFACE                         CIRCLE     DS(     ON)      00157 GRAPHICS
                4.0000     2.2500     0.0
                0.0        0.0        1.0000     0.5250
 0089    GOTO/                                                    00158 GRAPHICS
                3.7066     1.8134    -0.2500
                3.6444     1.8624    -0.2500
                3.5903     1.9201    -0.2500
                3.5454     1.9854    -0.2500
                3.5109     2.0566    -0.2500
                3.4874     2.1322    -0.2500
                3.4755     2.2104    -0.2500
                3.4750     2.2500    -0.2500
 0090    GOTO/   POINT                                            00160 GRAPHICS
                3.4746     2.2500     1.0000
 0091    GOTO/  SRFMIL                                            00162 GRAPHICS
                0.9576     0.7583    -0.2500
 0092    GOTO/                                                    00164 GRAPHICS
                0.9369     0.7911    -0.2500
 0092                                                             00165 GRAPHICS
 0092                                                             00166 GRAPHICS
 0092 SURFACE                         CIRCLE     DS(     ON)      00168 GRAPHICS
                0.5000     0.5000     0.0
                0.0        0.0        1.0000     0.5250
 0092    GOTO/                                                    00169 GRAPHICS
                0.9124     0.8265    -0.2500
                0.8537     0.8894    -0.2500
                0.7854     0.9418    -0.2500
                0.7095     0.9825    -0.2500
                0.6280     1.0102    -0.2500
                0.5430     1.0242    -0.2500
```

```
                 0.5000    1.0250    -0.2500
0093    GOTO/     POINT                                   00171 GRAPHICS
                 0.5000    1.0254     1.0000
0094    GOTO/    SRFMIL                                   00173 GRAPHICS
                 0.9896    0.3094    -0.2500
0096 TRACUT/     VU  FV                                   00175 GRAPHICS
                 1.0000    0.0        0.0      0.0
                 0.0       0.0       -1.0000   0.0
                 0.0       1.0000     0.0      0.0
0100    GOTO/    SRFMIL                                   00177 GRAPHICS
                 0.9896    0.3094    -0.2500
0100   GO---/    SRFMIL                                   00178 GRAPHICS
                 0.9822    0.2915    -0.2500
                 0.9742    0.2738    -0.2500
1                                                              PAGE 0009
                 0.9594    0.2451    -0.2500
                 0.9429    0.2174    -0.2500
                 0.9248    0.1908    -0.2500
                 0.9050    0.1653    -0.2500
                 0.8837    0.1411    -0.2500
                 0.8610    0.1183    -0.2500
                 0.8369    0.0969    -0.2500
                 0.8116    0.0770    -0.2500
                 0.7851    0.0586    -0.2500
                 0.7575    0.0420    -0.2500
                 0.7289    0.0271    -0.2500
                 0.6995    0.0139    -0.2500
                 0.6693    0.0026    -0.2500
                 0.6385   -0.0068    -0.2500
                 0.6071   -0.0144    -0.2500
                 0.5754   -0.0200    -0.2500
                 0.5485   -0.0232    -0.2500
                 0.5214   -0.0250    -0.2500
                 0.5107   -0.0253    -0.2500
                 0.5000   -0.0254    -0.2500
0131 TRACUT/     VU  PV                                   00180 GRAPHICS
                 1.0000    0.0        0.0      0.0
                 0.0       1.0000     0.0      0.0
                 0.0       0.0        1.0000   0.0
0132    GOTO/     POINT                                   00182 GRAPHICS
                 0.5000   -0.0250     1.0000
0133    GOTO/    SRFMIL                                   00184 GRAPHICS
                 0.7803    0.9444    -0.2500
0134    GOTO/    SRFMIL                                   00186 GRAPHICS
                 3.4891    1.3333    -0.2500
0135    GOTO/     POINT                                   00188 GRAPHICS
                 3.4891    1.3333     1.0000
0136    GOTO/    SRFMIL                                   00190 GRAPHICS
                 0.9655    0.7436    -0.2500
0137    GOTO/    SRFMIL                                   00192 GRAPHICS
                 3.2976    1.0785    -0.2500
0138    GOTO/     POINT                                   00194 GRAPHICS
                 3.2976    1.0785     1.0000
0139    GOTO/    SRFMIL                                   00196 GRAPHICS
                 0.9575    0.7583    -0.2500
0140    GOTO/    SRFMIL                                   00198 GRAPHICS
                 3.2438    0.9216    -0.2500
0141    GOTO/     POINT                                   00200 GRAPHICS
                 3.2438    0.9216     1.0000
0142    GOTO/    SRFMIL                                   00202 GRAPHICS
                 3.5772    0.1000    -0.2500
0143    GOTO/    SRFMIL                                   00204 GRAPHICS
                 0.8407    0.1000    -0.2500
0144    GOTO/     POINT                                   00206 GRAPHICS
                 0.8407    0.1000     1.0000
0145    GOTO/    SRFMIL                                   00208 GRAPHICS
                 3.3514    0.3250    -0.2500
```

```
 0146     GOTO/     SRFMIL                                                    00210 GRAPHICS
                    0.9954      0.3250     -0.2500
1                                                                                PAGE 0010
 0147     GOTO/      POINT                                                    00212 GRAPHICS
                    0.9954      0.3250      1.0000
 0148     GOTO/     SRFMIL                                                    00214 GRAPHICS
                    3.2759      0.4727     -0.2500
 0149     GOTO/     SRFMIL                                                    00216 GRAPHICS
                    0.9896      0.3094     -0.2500
 0150     GOTO/      POINT                                                    00218 GRAPHICS
                    0.9896      0.3094      1.0000
 0151     GOTO/     SRFMIL                                                    00220 GRAPHICS
                    3.5488      1.9807     -0.2500
 0152     GOTO/     SRFMIL                                                    00222 GRAPHICS
                    3.4296      1.2752     -0.2500
 0153     GOTO/      POINT                                                    00224 GRAPHICS
                    3.4296      1.2752      1.0000
 0154     GOTO/     SRFMIL                                                    00226 GRAPHICS
                    3.7449      1.7907     -0.2500
 0155     GOTO/     SRFMIL                                                    00228 GRAPHICS
                    3.6890      1.4603     -0.2500
 0156     GOTO/      POINT                                                    00230 GRAPHICS
                    3.6890      1.4603      1.0000
 0157     GOTO/     SRFMIL                                                    00232 GRAPHICS
                    3.7750      1.7752     -0.2500
 0158     GOTO/     SRFMIL                                                    00234 GRAPHICS
                    3.7750      1.4920     -0.2500
 0159     GOTO/      POINT                                                    00236 GRAPHICS
                    3.7750      1.4920      1.0000
 0160     GOTO/     SRFMIL                                                    00238 GRAPHICS
                    4.4512      1.9808     -0.2500
 0161     GOTO/     SRFMIL                                                    00240 GRAPHICS
                    4.5704      1.2752     -0.2500
 0162     GOTO/      POINT                                                    00242 GRAPHICS
                    4.5704      1.2752      1.0000
 0163     GOTO/     SRFMIL                                                    00244 GRAPHICS
                    4.2551      1.7906     -0.2500
 0164     GOTO/     SRFMIL                                                    00246 GRAPHICS
                    4.3110      1.4603     -0.2500
 0165     GOTO/      POINT                                                    00248 GRAPHICS
                    4.3110      1.4603      1.0000
 0166     GOTO/     SRFMIL                                                    00250 GRAPHICS
                    4.2250      1.7752     -0.2500
 0167     GOTO/     SRFMIL                                                    00252 GRAPHICS
                    4.2250      1.4920     -0.2500
 0168     GOTO/      POINT                                                    00254 GRAPHICS
                    0.0         0.0        -0.2500
 0170 SPINDL/          OFF                                                    00256 GRAPHICS
 0171 STOP                                                                    00258 GRAPHICS
 0171 LEADER/   10.0000                                                       00259 GRAPHICS
 0171 END                                                                     00261 GRAPHICS
 0005 ***** FINI *****                                                        00263
 ....END OF SECTION 3....
        SECTION 3 ELAPSED CPU TIME IN MIN/SEC IS 0000/00.6099

       TOTAL PART PROGRAM CPU TIME IN MIN/SEC IS 0000/01.4233
              **** END OF APT PROCESSING ****
```

# *OUTPUT FILE*

```
001 START INS 10
002  TD= 0.2500
003 SETUP > ZCXYU
004 FR XYZ=   7.0000
005 GO X  4.0000
006 GO Y -0.0254
007 SPINDLE ON
008 DWELL 04
009 GO Z -0.6000
010 GO X  4.0499
011    Y -0.0234
012   ZERO AT
013   X  4.0000
014   Y  0.7500
015  GR a    91.9087
016  > REF COODS
017 GO X  4.5177
018    Y  2.3375
019 GO X  4.5099
020    Y  2.3750
021   ZERO AT
022   X  4.0000
023   Y  2.2500
024  GR a   151.7444
025  > REF COODS
026 GO X  3.3392
027    Y  1.4907
028 GO X  3.3355
029    Y  1.4768
030   ZERO AT
031   X  3.2653
032   Y  1.5032
033  GR a   -51.0486
034  > REF COODS
035 GO X  0.4254
036    Y  1.0197
037 GO X  0.3877
038    Y  1.0129
039   ZERO AT
040   X  0.5000
041   Y  0.5000
042  GR a   162.7259
043  > REF COODS
044 GO X  4.0000
045 GO Z  1.0000
046 GO X  4.0000
047 GO Y  0.6246
048 GO Z -0.6000
049 GO X  3.9813
050    Y  0.6264
051   ZERO AT
052   X  4.0000
053   Y  0.7500
054  GR a  -360.0000
055  > REF COODS
056 GO Z  1.0000
057 GO X  4.0000
058 GO Y  2.1496
059 GO Z -0.6000
060 GO X  4.0167
061    Y  2.1514
062   ZERO AT
063   X  4.0000
```

```
064   Y  2.2500
065  GR a    334.1042
066   > REF COODS
067 GO Z  1.0000
068 GO X  0.5000
069 GO Y  0.6004
070 GO Z -0.6000
071 GO X  0.4833
072     Y  0.5986
073   ZERO AT
074   X  0.5000
075   Y  0.5000
076  GR a    334.1042
077   > REF COODS
078 GO Z  1.0000
079 GO X  3.2759
080 GO Y  0.4727
081 GO Z -0.2500
082 GO X  3.2852
083     Y  0.4495
084 GO X  3.2952
085     Y  0.4266
086 GO X  3.3125
087     Y  0.3914
088 GO X  3.3316
089     Y  0.3570
090 GO X  3.3523
091     Y  0.3237
092 GO X  3.3747
093     Y  0.2915
094 GO X  3.3987
095     Y  0.2605
096 GO X  3.4242
097     Y  0.2307
098 GO X  3.4512
099     Y  0.2022
100 GO X  3.4796
101     Y  0.1752
102 GO X  3.5093
103     Y  0.1496
104 GO X  3.5403
105     Y  0.1255
106 GO X  3.5725
107     Y  0.1031
108 GO X  3.6058
109     Y  0.0823
110 GO X  3.6400
111     Y  0.0632
112 GO X  3.6752
113     Y  0.0459
114 GO X  3.7113
115     Y  0.0304
116 GO X  3.7480
117     Y  0.0167
118 GO X  3.7854
119     Y  0.0049
120 GO X  3.8234
121     Y -0.0050
122 GO X  3.8618
123     Y -0.0130
124 GO X  3.9006
125     Y -0.0190
126 GO X  3.9361
127     Y -0.0228
128 GO X  3.9718
129     Y -0.0249
130 GO X  3.9859
```

```
   Y -0.0253
GO X  4.0000
   Y -0.0254
GO Z  1.0000
GO X  3.2438
GO Y  0.9216
GO Z -0.2500
GO X  3.2568
   Y  0.9696
  ZERO AT
  X  4.0000
  Y  0.7500
 GR a    -52.6204
 > REF COODS
GO Z  1.0000
GO X  4.2250
GO Y  1.4920
GO Z -0.2500
GO X  4.2722
   Y  1.4756
  ZERO AT
  X  4.0000
  Y  0.7500
 GR a    -65.5816
 > REF COODS
GO Z  1.0000
GO X  4.2250
GO Y  1.7751
GO Z -0.2500
GO X  4.2590
   Y  1.7933
  ZERO AT
  X  4.0000
  Y  2.2500
 GR a     56.1226
 > REF COODS
GO Z  1.0000
GO X  3.7750
GO Y  1.7752
GO Z -0.2500
GO X  3.7408
   Y  1.7935
  ZERO AT
  X  4.0000
  Y  2.2500
 GR a    -56.0984
 > REF COODS
GO Z  1.0000
GO X  0.9576
GO Y  0.7583
GO Z -0.2500
GO X  0.9369
   Y  0.7911
  ZERO AT
  X  0.5000
  Y  0.5000
 GR a     51.6311
 > REF COODS
GO Z  1.0000
GO X  0.9896
GO Y  0.3094
GO Z -0.2500
GO X  0.9822
   Y  0.2915
GO X  0.9742
   Y  0.2738
GO X  0.9594
```

```
198     Y  0.2451
199 GO X  0.9429
200     Y  0.2174
201 GO X  0.9248
202     Y  0.1908
203 GO X  0.9050
204     Y  0.1653
205 GO X  0.8837
206     Y  0.1411
207 GO X  0.8610
208     Y  0.1183
209 GO X  0.8369
210     Y  0.0969
211 GO X  0.8116
212     Y  0.0770
213 GO X  0.7851
214     Y  0.0586
215 GO X  0.7575
216     Y  0.0420
217 GO X  0.7289
218     Y  0.0271
219 GO X  0.6995
220     Y  0.0139
221 GO X  0.6693
222     Y  0.0026
223 GO X  0.6385
224     Y -0.0068
225 GO X  0.6071
226     Y -0.0144
227 GO X  0.5754
228     Y -0.0200
229 GO X  0.5485
230     Y -0.0232
231 GO X  0.5214
232     Y -0.0250
233 GO X  0.5107
234     Y -0.0253
235 GO X  0.5000
236     Y -0.0254
237 GO Z  1.0000
238 GO X  0.7803
239 GO Y  0.9444
240 GO Z -0.2500
241 GO X  3.4891
242     Y  1.3333
243 GO Z  1.0000
244 GO X  0.9655
245 GO Y  0.7436
246 GO Z -0.2500
247 GO X  3.2976
248     Y  1.0785
249 GO Z  1.0000
250 GO X  0.9575
251 GO Y  0.7583
252 GO Z -0.2500
253 GO X  3.2438
254     Y  0.9216
255 GO Z  1.0000
256 GO X  3.5772
257 GO Y  0.1000
258 GO Z -0.2500
259 GO X  0.8407
260 GO Z  1.0000
261 GO X  3.3514
262 GO Y  0.3250
263 GO Z -0.2500
264 GO X  0.9954
```

```
265 GO Z  1.0000
266 GO X  3.2759
267 GO Y  0.4727
268 GO Z -0.2500
269 GO X  0.9896
270    Y  0.3094
271 GO Z  1.0000
272 GO X  3.5488
273 GO Y  1.9807
274 GO Z -0.2500
275 GO X  3.4296
276    Y  1.2752
277 GO Z  1.0000
278 GO X  3.7449
279 GO Y  1.7907
280 GO Z -0.2500
281 GO X  3.6890
282    Y  1.4603
283 GO Z  1.0000
284 GO X  3.7750
285 GO Y  1.7752
286 GO Z -0.2500
287 GO Y  1.4920
288 GO Z  1.0000
289 GO X  4.4512
290 GO Y  1.9808
291 GO Z -0.2500
292 GO X  4.5704
293    Y  1.2752
294 GO Z  1.0000
295 GO X  4.2551
296 GO Y  1.7906
297 GO Z -0.2500
298 GO X  4.3110
299    Y  1.4603
300 GO Z  1.0000
301 GO X  4.2250
302 GO Y  1.7752
303 GO Z -0.2500
304 GO Y  1.4920
305 GO Z  1.0000
306 GO X  0.0000
307 GO Y  0.0000
308 SPINDLE OFF
309 END
```

# Appendix B
# Program Listing

The following pages present the Fortran code for the post-processor interface between CADAM NC and DYNA Milling Machine.

```
************************************************************************
      PROGRAM CONVER
************************************************************************
C     THIS PROGRAM CONVERTS THE OUTPUT OF THE APT-AC PROCESSOR TO A FORM
C     THAT CAN BE INPUT TO THE DYNA PROGRAM
************************************************************************
C     WRITTEN BY: HARINDER SINGH OBEROI
************************************************************************

      CHARACTER*81 INFILE, INFIL
      INTEGER IN, OUT, I, J

      IN=7
      OUT=9
      DO 100 I = 1, 10000
      READ(IN,10) INFILE
   10 FORMAT(A81)
      IF(INFILE(6:14) .EQ. 'SECTION 3') THEN
      GOTO 1000
      ENDIF
  100 CONTINUE

 1000 DO 300 J = 1,10
      READ(IN,20) INFIL
   20 FORMAT(A81)
      IF(INFIL(7:12) .EQ. 'CUTTER') THEN
      WRITE(OUT,30) INFIL(7:81)
      GO TO 310
      ENDIF
  300 CONTINUE
  310 DO 320 J=1,10000
      READ(IN,20) INFIL
      IF(INFIL(13:16) .EQ. 'FINI') THEN
      GOTO 1100
      ENDIF
      WRITE(OUT,30) INFIL(7:81)
   30 FORMAT(A75)
  320 CONTINUE
 1100 WRITE(OUT,40) INFIL(13:16)
   40 FORMAT(A4)
      END
```

# *Program CADDYN*

```
************************************************************************
      PROGRAM  CADDYN
************************************************************************
C     THIS PROGRAM INTERFACES THE CADAM NC II FUNCTION TO THE DYNAMYTE
C     MILLING MACHINE
************************************************************************
C     WRITTEN BY: HARINDER SINGH OBEROI
************************************************************************
C     N=NUMBER OF POINTS
C     KCOUNT=FLAG TO IDENTIFY THE TYPE OF STATEMENT
C     KFLAG=FLAG TO DETERMINE THE NUMBER OF CUTTERS
C     KFLAG1=FLAG TO IDENTIFY CUTTER STATEMENT
C     KFLAG2=FLAG TO IDENTIFY FEEDRATE STATEMENT
C     KFLAG3=FLAG TO IDENTIFY SPINDLE ON STATEMENT
C     KFLAG4=FLAG TO IDENTIFY SPINDLE OFF STATEMENT
************************************************************************
      REAL TD
      DIMENSION X(5000), Y(5000), Z(5000)
      INTEGER N, IN, OUT, KCOUNT(100), KFLAG, M, J, KFLAG1(0:5000), K
      INTEGER KFLAG2(0:5000),KFLAG3(0:5000),KFLAG4(0:5000)
      CHARACTER*80 INFILE, IMP, TEMP, DUMMY
      CHARACTER*25 BLANK, NUB*3, ANS*1, ANS1*1
      CHARACTER*20 FINAL
      DATA BLANK/'                         '/
************************************************************************
      IN = 9
      OUT = 12
      WRITE(6,*)'****************************************'
      WRITE(6,*)'THE RANGE OF THE DYNA MILLING MACHINE IS'
      WRITE(6,*)'X = 5.5 INCHES'
      WRITE(6,*)'Y = 4.5 INCHES'
      WRITE(6,*)'Z = 4.0 INCHES'
      WRITE(6,*)'****************************************'
      WRITE(6,*)'****************************************'
      WRITE(6,*)'DOES THE PART TO BE MACHINED LIE'
      WRITE(6,*)'WITHIN THESE PHYSICAL LIMITS'
      WRITE(6,*)'ENTER (Y/N)'
      WRITE(6,*)'****************************************'
      READ(5,7) ANS
    7 FORMAT(A1)

      IF(ANS .EQ. 'Y') THEN
      WRITE(6,*)'****************************************'
      WRITE(6,*)'THE CUTTER SIZES AVAILABLE ON DYNA ARE'
      WRITE(6,*)'0.03125, 0.0625, 0.09375, 0.1875'
      WRITE(6,*)'0.25, 0.3125, 0.375, 0.5'
      WRITE(6,*)'****************************************'
      WRITE(6,*)'THE CUTTER SPECIFIED SHOULD BE EITHER'
      WRITE(6,*)'ONE OF THESE OR CLOSE TO ONE OF THESE'
      WRITE(6,*)'SIZES'
      WRITE(6,*)'****************************************'
      WRITE(6,*)'IF THE CUTTER SIZE SPECIFIED IS'
      WRITE(6,*)'NOT ONE OF THESE'
      WRITE(6,*)'DO YOU WISH TO GO BACK AND CHANGE THE'
      WRITE(6,*)'CUTTER SIZE WHILE GENERATING THE'
      WRITE(6,*)'TOOL PATH  ENTER (Y/N)'
      WRITE(6,*)'****************************************'
      READ(5,8) ANS1
    8 FORMAT(A1)

      IF(ANS1 .EQ. 'Y') THEN
      GO TO 6000
      ENDIF
```

```
      IF(ANS1 .EQ. 'N') THEN
      GO TO 6001
      ENDIF

      ENDIF

      IF(ANS .EQ. 'N') THEN
      GO TO 6001
      ENDIF
***********************************************************************
 6001 WRITE(6,*)'****************************************'
      WRITE (6,*) 'PLEASE ENTER THE PROGRAM NUMBER'
      WRITE (6,*) 'IT SHOULD LIE BETWEEN 00 AND 99'
      WRITE(6,*)'****************************************'
      READ (5,*) N
      WRITE (OUT,9)'001 START INS', N
    9 FORMAT (A13,1X,I2)
***********************************************************************
C     THIS IS THE MAIN PROGRAM WHICH READS FROM THE INPUT FILE AND CALLS
C     THE VARIOUS SUBROUTINES ASSOCIATED WITH EACH STATEMENT READ
***********************************************************************
      M=0

 1000 DO 100 I = 1,100

      READ (IN,10) INFILE
   10 FORMAT(A80)

      IF(INFILE(1:6) .EQ. 'CUTTER') THEN
      KCOUNT(I)=1
      CALL TOOL(INFILE)
      M=M+1
      ENDIF

      IF(INFILE(1:6) .EQ. 'MCHTOL') THEN
      M=M+1
      ENDIF

      IF(INFILE(1:6) .EQ. 'FEDRAT') THEN
      KCOUNT(I)=2
      CALL FEED(INFILE)
      M=M+1
      ENDIF

      IF(INFILE(1:25) .EQ. BLANK) THEN
      GO TO 1000
      ENDIF


      IF(INFILE(3:6) .EQ. 'FROM') THEN
      KCOUNT(I)=3
      M=M+1
      CALL SETUP(INFILE)
      CALL POINT(KFLAG,KFLAG1,KFLAG2,KFLAG3,KFLAG4,N)
      GO TO 110
      ENDIF

  100 CONTINUE
  110 CONTINUE
***********************************************************************
C     THIS PART OF THE PROGRAM REWINDS ALL THE INTERNAL FILES CREATED
C     CALLS THE MOTION SUBROUTINE
***********************************************************************
      REWIND(13)
      REWIND(14)
      REWIND(15)
      REWIND(16)
```

```
      REWIND(17)
      REWIND(18)
      CALL MOTION(KFLAG,KFLAG1,N,X,Y,Z)
      REWIND(19)
      REWIND(20)
***********************************************************************
C     THIS PART OF THE PROGRAM WRITES THE TOOL DIAMETER, FEEDRATE, AND
C     SPINDLE ON AND SPINDLE OFF STATEMENT TO A INTERNAL FILE
***********************************************************************
C     TEMP=TEMPORARY CHARACTER STRING
***********************************************************************
      DO 120 I=1,M

      IF(KCOUNT(I) .EQ. 1) THEN
      READ(19,11) TEMP
      WRITE(23,12) TEMP
      ENDIF

      IF(KCOUNT(I) .EQ. 2) THEN
      READ(14,11) TEMP
      WRITE(23,12) TEMP
      ENDIF

      IF(KCOUNT(I) .EQ. 3) THEN
      READ(17,11) TEMP
      WRITE(23,12) TEMP
      ENDIF

   11 FORMAT(A80)
   12 FORMAT(A80)
  120 CONTINUE
***********************************************************************
C     THIS PART OF THE PROGRAM WRITES THE MOTION STATEMENTS TO AN
C     INTERNAL FILE
***********************************************************************
***********************************************************************
      READ(14,13) TEMP
   13 FORMAT(A80)
      WRITE(23,13) TEMP
      WRITE(23,14)'GO X',X(2)
   14 FORMAT(A4,1X,F7.4)
      WRITE(23,15)'GO Y',Y(2)
   15 FORMAT(A4,1X,F7.4)
      WRITE(23,16)'SPINDLE ON'
   16 FORMAT(A10)
      WRITE(23,17)'DWELL 04'
   17 FORMAT(A8)
      WRITE(23,18)'GO Z',Z(2)
   18 FORMAT(A4,1X,F7.4)
***********************************************************************

      DO 130 I=3,N-1

      DO 3020 J=1,4

      IF(KFLAG1(I) .EQ. J) THEN
      READ(19,19) TEMP
   19 FORMAT(A80)
      WRITE(23,20) TEMP
   20 FORMAT(A80)
      READ(19,19) TEMP
      WRITE(23,20) TEMP
      READ(19,19) TEMP
      WRITE(23,20) TEMP
      READ(19,19) TEMP
      WRITE(23,20) TEMP
      ENDIF
```

```
      IF(KFLAG2(I) .EQ. J) THEN
      READ(14,19) TEMP
      WRITE(23,20) TEMP
      ENDIF

      IF(KFLAG3(I) .EQ. J) THEN
      READ(15,21) TEMP
   21 FORMAT(A80)
      WRITE(23,22) TEMP
   22 FORMAT(A80)
      READ(15,21) TEMP
      WRITE(23,22) TEMP
      ENDIF

      IF(KFLAG4(I) .EQ. J) THEN
      READ(16,23) TEMP
   23 FORMAT(A80)
      WRITE(23,24) TEMP
   24 FORMAT(A80)
      ENDIF

 3020 CONTINUE

      IF(Z(I) .GT. 0.5) THEN
      WRITE(23,25)'GO Z', Z(I)
      GO TO 130
      ENDIF

      IF(Z(I-1) .GT. 0.5) THEN
      WRITE(23,25)'GO X', X(I)
      WRITE(23,25)'GO Y', Y(I)
      WRITE(23,25)'GO Z', Z(I)
      GO TO 130
      ENDIF

      IF(X(I) .EQ. X(I-1) .AND. Y(I) .EQ. Y(I-1) .AND.
     1    Z(I) .EQ. Z(I-1)) THEN
      GO TO 130
      ENDIF

      IF(X(I) .EQ. X(I-1) .AND. Y(I) .EQ. Y(I-1)) THEN
      WRITE(23,25)'GO Z',Z(I)
   25 FORMAT(A4,1X,F7.4)
      GO TO 130
      ENDIF

      IF(X(I) .EQ. X(I-1) .AND. Z(I) .EQ. Z(I-1)) THEN
      WRITE(23,26)'GO Y',Y(I)
   26 FORMAT(A4,1X,F7.4)
      GO TO 130
      ENDIF

      IF(Y(I) .EQ. Y(I-1) .AND. Z(I) .EQ. Z(I-1)) THEN
      WRITE(23,27)'GO X',X(I)
   27 FORMAT(A4,1X,F7.4)
      GO TO 130
      ENDIF

      IF(X(I) .EQ. X(I-1)) THEN
      WRITE(23,31)'GO Y',Y(I)
   31 FORMAT(A4,1X,F7.4)
      WRITE(23,32)'   Z',Z(I)
   32 FORMAT(A4,1X,F7.4)
      GO TO 130
      ENDIF
```

```
      IF(Y(I) .EQ. Y(I-1)) THEN
      WRITE(23,33)'GO X',X(I)
   33 FORMAT(A4,1X,F7.4)
      WRITE(23,34)'   Z',Z(I)
   34 FORMAT(A4,1X,F7.4)
      GO TO 130
      ENDIF

      IF(Z(I) .EQ. Z(I-1)) THEN
      WRITE(23,35)'GO X',X(I)
   35 FORMAT(A4,1X,F7.4)
      WRITE(23,36)'   Y',Y(I)
   36 FORMAT(A4,1X,F7.4)
      GO TO 130
      ENDIF

      WRITE(23,37)'GO X',X(I)
   37 FORMAT(A4,1X,F7.4)
      WRITE(23,38)'   Y',Y(I)
   38 FORMAT(A4,1X,F7.4)
      WRITE(23,39)'   Z',Z(I)
   39 FORMAT(A4,1X,F7.4)

  130 CONTINUE
************************************************************************
      WRITE(23,40)'GO Z',Z(1)
   40 FORMAT(A4,1X,F7.4)
      WRITE(23,41)'GO X',X(N)
   41 FORMAT(A4,1X,F7.4)
      WRITE(23,42)'GO Y',Y(N)
   42 FORMAT(A4,1X,F7.4)
      WRITE(23,'(A11)')'SPINDLE OFF'
************************************************************************
      READ(20,43) TEMP
   43 FORMAT(A80)
      WRITE(23,44) TEMP
   44 FORMAT(A80)
************************************************************************
C     THIS PART OF THE PROGRAM DETERMINES THE LENGTH OF THE MOTION
C     STATEMENT FILE
************************************************************************
      REWIND(23)
      K=1

      DO 140 I=1,2000

      READ(23,45) DUMMY
   45 FORMAT(A80)

      IF(DUMMY(1:3) .EQ. 'END') THEN
      GO TO 150
      ENDIF

      K=K+1
  140 CONTINUE
************************************************************************
C     THIS PART OF THE PROGRAM READS THE FINAL INTERNAL FILE, WRITES
C     STATEMENT NUMBERS IN FRONT OF EACH STATEMENT AND WRITES IT TO
C     THE FINAL OUTPUT FILE
************************************************************************
  150 REWIND(23)
      J=2

      DO 160 I=1,K

      READ(23,46) FINAL
   46 FORMAT(A20)
```

```
      IF(J .LT. 10) THEN
      WRITE(NUB,'(2H00,I1)')J
      ENDIF

      IF(J .GT. 9 .AND. J .LT. 100) THEN
      WRITE(NUB,'(1H0,I2)')J
      ENDIF

      IF(J .GE. 100 .AND. J .LT. 1000) THEN
      WRITE(NUB,'(I3)')J
      ENDIF

      WRITE(OUT,47) NUB,FINAL
   47 FORMAT(A3,1X,A20)
      J=J+1

  160 CONTINUE

 6000 END
*************************************************************************
      SUBROUTINE TOOL(IMP)
*************************************************************************
C     THIS SUBROUTINE CONVERTS THE CUTTER STATEMENT FROM THE INPUT FILE
C     TO THE TOOL DIAMETER STATEMENT IN DYNA
*************************************************************************
C     DIAM=TOOL DIAMETER READ FROM THE INPUT FILE
*************************************************************************
      CHARACTER*80 IMP
      REAL TD, DIAM

      READ(IMP,48) DIAM
   48 FORMAT(8X,F10.5)
      WRITE(13,49)'TD=', DIAM
   49 FORMAT(A4,1X,F6.4)
      RETURN
      END
*************************************************************************
      SUBROUTINE FEED(IMP)
*************************************************************************
C    THIS SUBROUTINE TAKES THE FEEDRATE STATEMENT FROM THE INPUT FILE
C    AND CHECKS WHETHER IT LIES IN THE RANGE OF FEEDRATES THAT CAN BE
C    SPECIFIED FOR THE DYNA MILLING MACHINE AND WRITES THE CORRESPONDING
C    DYNA FEEDRATE STATEMENT
*************************************************************************
C     FER=FEEDRATE READ FROM THE INPUT FILE
*************************************************************************
      CHARACTER*80 IMP
      REAL FER

      READ(IMP,50) FER
   50 FORMAT(8X,F10.5)

      IF(FER .GT. 0.10 .AND. FER .LT. 32.0) THEN
      GO TO 170
      ENDIF

 3030 WRITE(6,*)'THE FEEDRATE'
      WRITE(6,'(8X,F10.5)') FER
      WRITE(6,*)'CHOSEN IS OUT OF THE RANGE FOR THE DYNA'
      WRITE(6,*)'MILLING MACHINE, PLEASE ENTER A FEEDRATE WHICH LIES'
      WRITE(6,*)'IN THE RANGE OF 0.10 TO 32.0 IPM'
      READ(5,*) FER

      IF(FER .GT. 0.10 .AND. FER .LT. 32.0) THEN
      GO TO 180
      ENDIF
```

```
      GO TO 3030

  170 WRITE(14,51)'FR XYZ=', FER
   51 FORMAT(A7,1X,F8.4)
      GO TO 190
  180 WRITE(14,52)'FR XYZ=', FER
   52 FORMAT(A7,1X,F8.4)
  190 RETURN
      END
************************************************************************
      SUBROUTINE SPINON
************************************************************************
C     THIS SUBROUTINE TAKES THE SPINDLE ON STATEMENT FROM THE INPUT FILE
C     AND CONVERTS IT TO THE SPINDLE ON STATEMENT FOR THE DYNA MILLING
C     MACHINE
************************************************************************
      CHARACTER*10 BLOCK
      DATA BLOCK/'SPINDLE ON'/
      WRITE(15,53) BLOCK
   53 FORMAT(A10)
      WRITE(15,54)'DWELL 04'
   54 FORMAT(A8)
      RETURN
      END
************************************************************************
      SUBROUTINE SPIOFF
************************************************************************
C     THIS SUBROUTINE TAKES THE SPINDLE OFF STATEMENT FROM THE INPUT
C     FILE AND CONVERTS IT TO THE SPINDLE OFF STATEMENT FOR THE DYNA
C     MILLING MACHINE
************************************************************************
      CHARACTER*11 BLOCK
      DATA BLOCK/'SPINDLE OFF'/
      WRITE(16,54) BLOCK
   54 FORMAT(A11)
      RETURN
      END
************************************************************************
      SUBROUTINE SETUP(IMP)
************************************************************************
C     THIS SUBROUTINE TAKES THE FROM STATEMENT IN THE INPUT FILE AND
C     CONVERTS IT TO THE SETUP STATEMENT FOR THE DYNA MILLING MACHINE
************************************************************************
      CHARACTER*80 IMP,DUMMY
      INTEGER OUT,IN
      IN=9
      OUT=12
      WRITE(17,55)'SETUP > ZCXYU'
   55 FORMAT(A13)
      RETURN
      END
************************************************************************
      SUBROUTINE POINT(KFLAG,KFLAG1,KFLAG2,KFLAG3,KFLAG4,N)
************************************************************************
C     THIS SUBROUTINE TAKES THE X, Y, AND Z COORDINATES AND WRITES THEM
C     TO A FILE
************************************************************************
C     KFLAG=FLAG TO DETERMINE THE NUMBER OF CUTTERS
C     KFLAG1=FLAG TO IDENTIFY CUTTER STATEMENT
C     KFLAG2=FLAG TO IDENTIFY FEEDRATE STATEMENT
C     KFLAG3=FLAG TO IDENTIFY SPINDLE ON STATEMENT
C     KFLAG4=FLAG TO IDENTIFY SPINDLE OFF STATEMENT
C     N=NUMBER OF POINTS
************************************************************************
      CHARACTER*80 INFIL
      CHARACTER*25 BLANK
```

```
      INTEGER N, M, KFLAG, IN, KFLAG1(0:5000),KFLAG2(0:5000)
      INTEGER KFLAG3(0:5000),KFLAG4(0:5000)
      REAL A,B,C
      N=0
      IN=9
      KFLAG=0
**************************************************************************
      DATA BLANK/'                                      '/
      M=1

 2001 DO 210 I=1,5000

      READ(IN,56) INFIL
   56 FORMAT(A80)

      IF(INFIL(1:25) .EQ. BLANK) THEN
      GO TO 2001
      ENDIF

      IF(INFIL(3:6) .EQ. 'GOTO') THEN
      GO TO 2001
      ENDIF

      IF(INFIL(2:3) .EQ. 'GO') THEN
      GO TO 2001
      ENDIF

      IF(INFIL(1:4) .EQ. 'STOP') THEN
      GO TO 2001
      ENDIF

      IF(INFIL(1:6) .EQ. 'LEADER') THEN
      GO TO 2001
      ENDIF

      IF(INFIL(1:3) .EQ. 'END') THEN
      GO TO 2001
      ENDIF

      IF(INFIL(3:6) .EQ. 'FROM') THEN
      GO TO 2001
      ENDIF

      IF(INFIL(1:7) .EQ. 'SURFACE') THEN
      CALL CIRCLE(INFIL)
      GO TO 2001
      ENDIF

      DO 3000 J=1,4

      IF(INFIL(1:6) .EQ. 'CUTTER') THEN
      CALL TOOL(INFIL)
      KFLAG1(N)=M
      KFLAG=KFLAG+1
      GO TO 200
      ENDIF

      IF(INFIL(1:6) .EQ. 'FEDRAT') THEN
      CALL FEED(INFIL)
      KFLAG2(N)=M
      GO TO 200
      ENDIF

      IF(INFIL(1:6) .EQ. 'SPINDL') THEN
      IF(INFIL(16:17) .EQ. 'ON') THEN
      CALL SPINON
      KFLAG3(N)=M
```

```
      GO TO 200
      ELSE
      CALL SPIOFF
      KFLAG4(N)=M
      GO TO 200
      ENDIF
      ENDIF

      GO TO 3001
 3000 CONTINUE

  200 M=M+1
      GO TO 2001

 3001 IF(INFIL(1:4) .EQ. 'FINI') THEN
      CALL FINIS(INFIL)
      GO TO 220
      ENDIF

      READ(INFIL,57) A, B, C
   57 FORMAT(10X,F8.4,3X,F8.4,3X,F8.4)
      WRITE(18,58) A, B, C
   58 FORMAT(10X,F8.4,3X,F8.4,3X,F8.4)
      N=N+1
      M=1

  210 CONTINUE
  220 RETURN
      END
************************************************************************
      SUBROUTINE CIRCLE(IMP)
************************************************************************
C     THIS SUBROUTINE TAKES THE CIRCULAR SURFACE DATA AND WRITES IT TO
C     THE OUTPUT FILE
************************************************************************
      CHARACTER* 80 IMP, INFI
      CHARACTER* 25 BLANK
      INTEGER IN, OUT
      REAL A,B,C,D,E,F
      DATA BLANK/'                         '/
      IN=9
      OUT=12
      READ(IN,59) INFI
   59 FORMAT(A80)
      IF (INFI(1:25) .EQ. 'BLANK') THEN
      READ(IN,60) A,B,C
      READ(IN,60) D,E,F
   60 FORMAT(10X,F7.4,4X,F7.4,4X,F7.4)
      ELSE
      READ(IN,60) D,E,F
      ENDIF
      RETURN
      END
************************************************************************
      SUBROUTINE MOTION(KFLAG,KFLAG1,N,X,Y,Z)
************************************************************************
C     THIS SUBROUTINE TAKES THE GOTO STATEMENT FROM THE INPUT FILE AND
C     CONVERTS IT TO THE MOTION STATEMENT FOR THE DYNA MILLING MACHINE
************************************************************************
C     SCALX=SCALE FACTOR IN THE X DIRECTION
C     SCALY=SCALE FACTOR IN THE Y DIRECTION
C     SCALZ=SCALE FACTOR IN THE Z DIRECTION
C     DRANX=RANGE OF DYNA IN THE X DIRECTION
C     DRANY=RANGE OF DYNA IN THE Y DIRECTION
C     DRANZ=RANGE OF DYNA IN THE Z DIRECTION
C     TD1 TO TD8= EIGHT CUTTERS AVAILABLE ON DYNA MACHINE
C     FSCAL=FINAL SCALE FACTOR TO BE USED FOR SCALING THE PART DOWN
```

```
C     DIAM1 TO DIAM8= TOOL DIAMETERS READ FROM THE INPUT FILE
*********************************************************************
      DIMENSION A(5000),B(5000),C(5000)
      DIMENSION X(5000),Y(5000),Z(5000)
      INTEGER INT,N,J,KFLAG,KFLAG2
      REAL SCALX,SCALY,SCALZ,DRANX,DRANY,DRANZ,SCAL1,FSCAL
      REAL TD1,TD2,TD3,TD4,TD5,TD6,TD7,TD8,SDIAM
      REAL DIAM1, DIAM2, DIAM3, DIAM4
      REAL DIAM5, DIAM6, DIAM7, DIAM8
      REAL XRANGE,YRANGE,ZRANGE,FER,XZERO,YZERO
      DIMENSION Y1(10000)
      REAL XMIN,YMIN,XMAX,YMAX,ZMAX,ZMIN
      CHARACTER*1 ANS

      DRANX=5.5
      DRANY=4.5
      DRANZ=4.0
      TD1=0.03125
      TD2=0.0625
      TD3=0.09375
      TD4=0.1875
      TD5=0.25
      TD6=0.3125
      TD7=0.375
      TD8=0.5
*********************************************************************
C     THIS PORTION OF THE SUBROUTINE READS IN THE X, Y, AND Z VALUES
C     FROM A FILE
*********************************************************************
      DO 230 I=1,N

      READ(18,61) X(I),Y(I),Z(I)
   61 FORMAT(10X,F8.4,3X,F8.4,3X,F8.4)

  230 CONTINUE
*********************************************************************
C     THIS PORTION OF THE SUBROUTINE PERFORMS THE COORDINATE
C     TRANSFORMATIONS BY MAKING THE SETUP X, Y, AND Z VALUES ZERO
*********************************************************************
      XZERO=X(1)
      YZERO=Y(1)

      DO 240 I=1,N

      X(I)=X(I)-XZERO
      Y(I)=Y(I)-YZERO

  240 CONTINUE
*********************************************************************
C     THIS PORTION OF THE SUBROUTINE CALCULATES THE MINIMUM X, Y, AND Z
C     VALUES
*********************************************************************
      XMIN=X(1)
      YMIN=Y(1)
      ZMIN=Z(1)

      DO 250 I=2,N

      IF(X(I) .LT. XMIN) THEN
      XMIN=X(I)
      ENDIF

      IF(Y(I) .LT. YMIN) THEN
      YMIN=Y(I)
      ENDIF

      IF(Z(I) .LT. ZMIN) THEN
```

```
      ZMIN=Z(I)
      ENDIF

  250 CONTINUE
************************************************************************
C     THIS PORTION OF THE SUBROUTINE CALCULATES THE MAXIMUM X, Y, AND Z
C     VALUES
************************************************************************
      XMAX=X(1)
      YMAX=Y(1)
      ZMAX=Z(1)

      DO 260 I=2,N

      IF(X(I) .GT. XMAX) THEN
      XMAX=X(I)
      ENDIF

      IF(Y(I) .GT. YMAX) THEN
      YMAX=Y(I)
      ENDIF

      IF(Z(I) .GT. ZMAX) THEN
      ZMAX=Z(I)
      ENDIF

  260 CONTINUE
************************************************************************
C     THIS PORTION OF THE SUBROUTINE CALCULATES THE RANGE IN THE X, Y,
C     Z DIRECTION
************************************************************************
      XRANGE=XMAX-XMIN
      YRANGE=YMAX-YMIN
      ZRANGE=ZMAX-ZMIN
************************************************************************
C     THIS PORTION OF THE SUBROUTINE CALCULATES THE APPROXIMATE SCALE
C     VALUE IN THE X, Y, AND Z DIRECTION
************************************************************************
      IF(XRANGE .GT. DRANX) THEN
      SCALX=XRANGE/DRANX
      ELSE
      SCALX=1.0
      ENDIF

      IF(YRANGE .GT. DRANY) THEN
      SCALY=YRANGE/DRANY
      ELSE
      SCALY=1.0
      ENDIF

      IF(ZRANGE .GT. DRANZ) THEN
      SCALZ=ZRANGE/DRANZ
      ELSE
      SCALZ=1.0
      ENDIF
************************************************************************
C     THIS PORTION OF THE SUBROUTINE CALCULATES THE MAXIMUM OF THE
C     APPROXIMATE SCALE VALUES IN THE X, Y, AND Z DIRECTION
************************************************************************
      IF(SCALX .GE. SCALY) THEN
      SCAL1=SCALX
      ENDIF

      IF(SCALY .GE. SCALX) THEN
      SCAL1=SCALY
      ENDIF
```

```
      IF(SCAL1 .LE. SCALZ) THEN
      SCAL1=SCALZ
      ENDIF
************************************************************************
C     THIS PORTION OF THE SUBROUTINE READS IN THE TOOL DIAMETERS FROM A
C     INTERNAL FILE
************************************************************************
      READ(13,62) DIAM1
      IF(KFLAG .GT. 0) THEN
      READ(13,62) DIAM2
      ENDIF

      IF(KFLAG .GT. 1) THEN
      READ(13,62) DIAM3
      ENDIF

      IF(KFLAG .GT. 2) THEN
      READ(13,62) DIAM4
      ENDIF

      IF(KFLAG .GT. 3) THEN
      READ(13,62) DIAM5
      ENDIF

      IF(KFLAG .GT. 4) THEN
      READ(13,62) DIAM6
      ENDIF

      IF(KFLAG .GT. 5) THEN
      READ(13,62) DIAM7
      ENDIF

      IF(KFLAG .GT. 6) THEN
      READ(13,62) DIAM8
      ENDIF

   62 FORMAT(5X,F6.4)
************************************************************************
      IF(SCAL1 .EQ. 1.0) THEN
      WRITE(6,*)'*****************************************'
      WRITE(6,*)'THE PART IS WITHIN THE RANGE OF DYNA '
      WRITE(6,*)'SO IT DOES NOT HAVE TO BE SCALED DOWN'
      WRITE(6,*)'*****************************************'
      GO TO 270
      ENDIF
      GO TO 280
************************************************************************
C     THIS PORTION OF THE SUBROUTINE WRITES OUT THE TOOL DIAMETERS IF
C     THE PART IS NOT TO BE SCALED DOWN
************************************************************************
  270 WRITE(19,63)'TD=', DIAM1

      IF(KFLAG .GT.0) THEN
      WRITE(19,64)'TOOL 2'
      WRITE(19,65)'SPINDLE OFF'
      WRITE(19,63)'TD=', DIAM2
      WRITE(19,66)'SPINDLE ON'
      WRITE(19,103)'DWELL 04'
      ENDIF

      IF(KFLAG .GT.1) THEN
      WRITE(19,64)'TOOL 3'
      WRITE(19,65)'SPINDLE OFF'
      WRITE(19,63)'TD=', DIAM3
      WRITE(19,66)'SPINDLE ON'
      WRITE(19,103)'DWELL 04'
      ENDIF
```

```
      IF(KFLAG .GT.2) THEN
      WRITE(19,64)'TOOL 4'
      WRITE(19,65)'SPINDLE OFF'
      WRITE(19,63)'TD=', DIAM4
      WRITE(19,66)'SPINDLE ON'
      WRITE(19,103)'DWELL 04'
      ENDIF

      IF(KFLAG .GT.3) THEN
      WRITE(19,64)'TOOL 5'
      WRITE(19,65)'SPINDLE OFF'
      WRITE(19,63)'TD=', DIAM5
      WRITE(19,66)'SPINDLE ON'
      WRITE(19,103)'DWELL 04'
      ENDIF

      IF(KFLAG .GT.4) THEN
      WRITE(19,64)'TOOL 6'
      WRITE(19,65)'SPINDLE OFF'
      WRITE(19,63)'TD=', DIAM6
      WRITE(19,66)'SPINDLE ON'
      WRITE(19,103)'DWELL 04'
      ENDIF

      IF(KFLAG .GT.5) THEN
      WRITE(19,64)'TOOL 7'
      WRITE(19,65)'SPINDLE OFF'
      WRITE(19,63)'TD=', DIAM7
      WRITE(19,66)'SPINDLE ON'
      WRITE(19,103)'DWELL 04'
      ENDIF

      IF(KFLAG .GT.6) THEN
      WRITE(19,64)'TOOL 8'
      WRITE(19,65)'SPINDLE OFF'
      WRITE(19,63)'TD=', DIAM8
      WRITE(19,66)'SPINDLE ON'
      WRITE(19,103)'DWELL 04'
      ENDIF

   63 FORMAT(A4,1X,F6.4)
   64 FORMAT(A6)
   65 FORMAT(A11)
   66 FORMAT(A10)
  103 FORMAT(A8)

      FSCAL=1.0
      GO TO 310
************************************************************************
C     THIS PORTION OF THE SUBROUTINE CALCULATES THE SCALE VALUE AND
C     CORRESPONDINGLY SCALES THE TOOL DIAMETER
************************************************************************
  280 WRITE(6,*)'******************************************'
      WRITE(6,*)'THE PART SHOULD BE SCALED DOWN'
      WRITE(6,67)SCAL1
   67 FORMAT(1X,F6.4)
      WRITE(6,*)'TIMES, SO THAT IT CAN BE MACHINED ON DYNA'
      WRITE(6,*)'******************************************'
************************************************************************
      DO 290 I=1,5

      KFLAG2=1
      CALL CUTTER(DIAM1,SCAL1,KFLAG2,SDIAM)
      SCAL1=DIAM1/SDIAM

      IF(KFLAG .GT. 0) THEN
```

```
      KFLAG2=2
      CALL CUTTER(DIAM2,SCAL1,KFLAG2,SDIAM)
      ENDIF

      IF(KFLAG .GT. 1) THEN
      KFLAG2=3
      CALL CUTTER(DIAM3,SCAL1,KFLAG2,SDIAM)
      ENDIF

      IF(KFLAG .GT. 2) THEN
      KFLAG2=4
      CALL CUTTER(DIAM4,SCAL1,KFLAG2,SDIAM)
      ENDIF

      IF(KFLAG .GT. 3) THEN
      KFLAG2=5
      CALL CUTTER(DIAM5,SCAL1,KFLAG2,SDIAM)
      ENDIF

      IF(KFLAG .GT. 4) THEN
      KFLAG2=6
      CALL CUTTER(DIAM6,SCAL1,KFLAG2,SDIAM)
      ENDIF

      IF(KFLAG .GT. 5) THEN
      KFLAG2=7
      CALL CUTTER(DIAM7,SCAL1,KFLAG2,SDIAM)
      ENDIF

      IF(KFLAG .GT. 6) THEN
      KFLAG2=8
      CALL CUTTER(DIAM8,SCAL1,KFLAG2,SDIAM)
      ENDIF
*************************************************************************
      WRITE(6,*)'THE SCALE IS '
      WRITE(6,68)SCAL1
   68 FORMAT(1X,F6.4)
      WRITE(6,*)'DO YOU WISH TO USE THE SAME SCALE? (Y/N)'
      READ(5,69) ANS
   69 FORMAT(A1)

      IF(ANS .EQ. 'Y') THEN
      GO TO 300
      ENDIF

      IF(ANS .EQ. 'N') THEN
      WRITE(6,*)'PLEASE ENTER THE SCALE YOU WISH TO USE'
      READ(5,70) SCAL1
   70 FORMAT(F6.4)
      ENDIF

  290 CONTINUE
*************************************************************************
C     THIS PORTION OF THE SUBROUTINE WRITES THE TOOL DIAMETERS TO AN
C     INTERNAL FILE
*************************************************************************
  300 FSCAL=SCAL1
      FDIAM=DIAM1/SCAL1
      WRITE(19,71)'TD=',FDIAM

      IF(KFLAG .GT. 0) THEN
      FDIAM=DIAM2/SCAL1
      WRITE(19,72)'TOOL 2'
      WRITE(19,73)'SPINDLE OFF'
      WRITE(19,71)'TD=',FDIAM
      WRITE(19,74)'SPINDLE ON'
      WRITE(19,104)'DWELL 04'
```

```
      ENDIF

      IF(KFLAG .GT. 1) THEN
      FDIAM=DIAM3/SCAL1
      WRITE(19,72)'TOOL 3'
      WRITE(19,73)'SPINDLE OFF'
      WRITE(19,71)'TD=',FDIAM
      WRITE(19,74)'SPINDLE ON'
      WRITE(19,104)'DWELL 04'
      ENDIF

      IF(KFLAG .GT. 2) THEN
      FDIAM=DIAM4/SCAL1
      WRITE(19,72)'TOOL 4'
      WRITE(19,73)'SPINDLE OFF'
      WRITE(19,71)'TD=',FDIAM
      WRITE(19,74)'SPINDLE ON'
      WRITE(19,104)'DWELL 04'
      ENDIF

      IF(KFLAG .GT. 3) THEN
      FDIAM=DIAM5/SCAL1
      WRITE(19,72)'TOOL 5'
      WRITE(19,73)'SPINDLE OFF'
      WRITE(19,71)'TD=',FDIAM
      WRITE(19,74)'SPINDLE ON'
      WRITE(19,104)'DWELL 04'
      ENDIF

      IF(KFLAG .GT. 4) THEN
      FDIAM=DIAM6/SCAL1
      WRITE(19,72)'TOOL 6'
      WRITE(19,73)'SPINDLE OFF'
      WRITE(19,71)'TD=',FDIAM
      WRITE(19,74)'SPINDLE ON'
      WRITE(19,104)'DWELL 04'
      ENDIF

      IF(KFLAG .GT. 5) THEN
      FDIAM=DIAM7/SCAL1
      WRITE(19,72)'TOOL 7'
      WRITE(19,73)'SPINDLE OFF'
      WRITE(19,71)'TD=',FDIAM
      WRITE(19,74)'SPINDLE ON'
      WRITE(19,104)'DWELL 04'
      ENDIF

      IF(KFLAG .GT. 6) THEN
      FDIAM=DIAM8/SCAL1
      WRITE(19,72)'TOOL 8'
      WRITE(19,73)'SPINDLE OFF'
      WRITE(19,71)'TD=',FDIAM
      WRITE(19,74)'SPINDLE ON'
      WRITE(19,104)'DWELL 04'
      ENDIF

   71 FORMAT(A4,1X,F6.4)
   72 FORMAT(A6)
   73 FORMAT(A11)
   74 FORMAT(A10)
  104 FORMAT(A8)
************************************************************************
C     THIS PORTION OF THE SUBROUTINE CALCULATES THE NEW X, Y, AND Z
C     COORDINATE DATA
************************************************************************
  310 DO 320 I=1,N
```

```
      X(I)=X(I)/FSCAL
      Y(I)=Y(I)/FSCAL
      Z(I)=Z(I)/FSCAL

  320 CONTINUE
************************************************************************
C     THIS PORTION OF THE SUBROUTINE CALCULATES THE MAXIMUM VALUES OF THE
C     NEW X, Y, AND Z COORDINATES AND INFORMS THE USER OF THE
C     POSITION OF THE SETUP POINT
************************************************************************
      XMAX=X(1)
      YMAX=Y(1)
      ZMAX=Z(1)

      DO 330 I=2,N

      IF(X(I) .GT. XMAX) THEN
      XMAX=X(I)
      ENDIF

      IF(Y(I) .GT. YMAX) THEN
      YMAX=Y(I)
      ENDIF

      IF(Z(I) .GT. ZMAX) THEN
      ZMAX=Z(I)
      ENDIF

  330 CONTINUE
      WRITE(6,*)'*******************************************'
      WRITE(6,*)'THE MAXIMUM TRAVEL IN X DIRECTION IS'
      WRITE(6,105) 'XMAX=',XMAX,'INCHES'
      WRITE(6,*)'*******************************************'
      WRITE(6,*)'THE MAXIMUM TRAVEL IN THE Y DIRECTION IS'
      WRITE(6,105) 'YMAX=',YMAX,'INCHES'
      WRITE(6,*)'*******************************************'
      WRITE(6,*)'THE MAXIMUM TRAVEL IN THE Z DIRECTION IS'
      WRITE(6,105) 'ZMAX=',ZMAX,'INCHES'
      WRITE(6,*)'*********************************************'
  105 FORMAT(1X,A5,1X,F8.4,1X,A6)
      XMIN=X(1)
      YMIN=Y(1)
      ZMIN=Z(1)

      DO 340 I=2,N

      IF(X(I) .LT. XMIN) THEN
      XMIN=X(I)
      ENDIF

      IF(Y(I) .LT. YMIN) THEN
      YMIN=Y(I)
      ENDIF

      IF(Z(I) .LT. ZMIN) THEN
      ZMIN=Z(I)
      ENDIF

  340 CONTINUE

      WRITE(6,*)'THE MINIMUM X, Y, AND Z COORDINATES ARE'
      WRITE(6,106) 'XMIN=',XMIN,'INCHES'
      WRITE(6,106) 'YMIN=',YMIN,'INCHES'
      WRITE(6,106) 'ZMIN=',ZMIN,'INCHES'
  106 FORMAT(1X,A5,1X,F8.4,1X,A6)
      RETURN
```

```
      END
************************************************************************
      SUBROUTINE CUTTER(TDIAM,SCAL1,KFLAG2,SDIAM)
************************************************************************
C     THIS SUBROUTINE TAKES THE VARIOUS DIFFERENT CUTTERS, SCALES THEM
C     DOWN AND CHECKS WHETHER THE SCALED CUTTERS ARE AVAILABLE ON DYNA
************************************************************************
      REAL TDIAM, SCAL1, SDIAM
      INTEGER KFLAG2
      CHARACTER*1 ANS
      REAL TD1,TD2,TD3,TD4,TD5,TD6,TD7,TD8


      TD1=0.03125
      TD2=0.0625
      TD3=0.09375
      TD4=0.1875
      TD5=0.25
      TD6=0.3125
      TD7=0.375
      TD8=0.5

      WRITE(6,*)'****************************************'

      IF(KFLAG2 .EQ. 1) THEN
      WRITE(6,*)'THE FIRST CUTTER IS SCALED DOWN TO A SIZE'
      ENDIF

      IF(KFLAG2 .EQ. 2) THEN
      WRITE(6,*)'THE SECOND CUTTER IS SCALED DOWN TO A SIZE'
      ENDIF

      IF(KFLAG2 .EQ. 3) THEN
      WRITE(6,*)'THE THIRD CUTTER IS SCALED DOWN TO A SIZE'
      ENDIF

      IF(KFLAG2 .EQ. 4) THEN
      WRITE(6,*)'THE FOURTH CUTTER IS SCALED DOWN TO A SIZE'
      ENDIF

      IF(KFLAG2 .EQ. 5) THEN
      WRITE(6,*)'THE FIFTH CUTTER IS SCALED DOWN TO A SIZE'
      ENDIF

      IF(KFLAG2 .EQ. 6) THEN
      WRITE(6,*)'THE SIXTH CUTTER IS SCALED DOWN TO A SIZE'
      ENDIF

      IF(KFLAG2 .EQ. 7) THEN
      WRITE(6,*)'THE SEVENTH CUTTER IS SCALED DOWN TO A SIZE'
      ENDIF

      IF(KFLAG2 .EQ. 8) THEN
      WRITE(6,*)'THE EIGHTH CUTTER IS SCALED DOWN TO A SIZE'
      ENDIF

      SDIAM=TDIAM/SCAL1
      WRITE(6,75) SDIAM
   75 FORMAT(1X,F6.4)

      IF(SDIAM .EQ. TD1) THEN
      WRITE(6,*)'WHICH IS AVAILABLE ON DYNA'
      GO TO 350
      ENDIF

      IF(SDIAM .EQ. TD2) THEN
      WRITE(6,*)'WHICH IS AVAILABLE ON DYNA'
```

```
      GO TO 350
      ENDIF

      IF(SDIAM .EQ. TD3) THEN
      WRITE(6,*)'WHICH IS AVAILABLE ON DYNA'
      GO TO 350
      ENDIF

      IF(SDIAM .EQ. TD4) THEN
      WRITE(6,*)'WHICH IS AVAILABLE ON DYNA'
      GO TO 350
      ENDIF

      IF(SDIAM .EQ. TD5) THEN
      WRITE(6,*)'WHICH IS AVAILABLE ON DYNA'
      GO TO 350
      ENDIF

      IF(SDIAM .EQ.TD6) THEN
      WRITE(6,*)'WHICH IS AVAILBLE ON DYNA'
      GO TO 350
      ENDIF

      IF(SDIAM .EQ. TD7) THEN
      WRITE(6,*)'WHICH IS AVAILBLE ON DYNA'
      GO TO 350
      ENDIF

      IF(SDIAM .EQ. TD8) THEN
      WRITE(6,*)'WHICH IS AVAILABLE ON DYNA'
      GO TO 350
      ENDIF

      IF(SDIAM .LT. TD1) THEN
      WRITE(6,*)'WHICH IS CLOSE TO'
      WRITE(6,76) TD1
      WRITE(6,*)'AVAILABLE ON DYNA'
      WRITE(6,*)'DO YOU WISH TO USE THIS CUTTER? (Y/N)'
      READ(5,77) ANS

      IF(ANS .EQ. 'Y') THEN
      SDIAM=TD1
      ENDIF

      IF(ANS .EQ. 'N') THEN
      WRITE(6,*)'THE DATA WILL BE GENERATED ACCORDING TO'
      WRITE(6,*)'THE ORIGINAL SCALED CUTTER, BUT SINCE THIS'
      WRITE(6,*)'CUTTER IS NOT AVAILABLE ON DYNA, YOU WILL'
      WRITE(6,*)'HAVE TO MACHINE THE PART WITH A CUTTER'
      WRITE(6,*)'CLOSE TO THIS SIZE'
      ENDIF

      GO TO 350
      ENDIF

      IF(SDIAM .GT. TD1 .AND. SDIAM .LT. TD2) THEN
      WRITE(6,*)'WHICH IS CLOSE TO'
      WRITE(6,76) TD1
   76 FORMAT(1X,F6.4)
      WRITE(6,*)'AVAILABLE ON DYNA'
      WRITE(6,*)'DO YOU WISH TO USE THIS CUTTER? (Y/N)'
      READ(5,77) ANS

      IF(ANS .EQ. 'Y') THEN
      SDIAM=TD1
      ENDIF
```

```
IF(ANS .EQ. 'N') THEN
WRITE(6,*)'THE DATA WILL BE GENERATED ACCORDING TO'
WRITE(6,*)'THE ORIGINAL SCALED CUTTER, BUT SINCE THIS'
WRITE(6,*)'CUTTER IS NOT AVAILABLE ON DYNA, YOU WILL'
WRITE(6,*)'HAVE TO MACHINE THE PART WITH A CUTTER'
WRITE(6,*)'CLOSE TO THIS SIZE'
ENDIF

GO TO 350
ENDIF

IF(SDIAM .GT. TD2 .AND. SDIAM .LT. TD3) THEN
WRITE(6,*)'WHICH IS CLOSE TO'
WRITE(6,76) TD2
WRITE(6,*)'AVAILABLE ON DYNA'
WRITE(6,*)'DO YOU WISH TO USE THIS CUTTER? (Y/N)'
READ(5,77) ANS

IF(ANS .EQ. 'Y') THEN
SDIAM=TD2
ENDIF

IF(ANS .EQ. 'N') THEN
WRITE(6,*)'THE DATA WILL BE GENERATED ACCORDING TO'
WRITE(6,*)'THE ORIGINAL SCALED CUTTER, BUT SINCE THIS'
WRITE(6,*)'CUTTER IS NOT AVAILABLE ON DYNA, YOU WILL'
WRITE(6,*)'HAVE TO MACHINE THE PART WITH A CUTTER'
WRITE(6,*)'CLOSE TO THIS SIZE'
ENDIF

GO TO 350
ENDIF

IF(SDIAM .GT. TD3 .AND. SDIAM .LT. TD4) THEN
WRITE(6,*)'WHICH IS CLOSE TO'
WRITE(6,76) TD3
WRITE(6,*)'AVAILABLE ON DYNA'
WRITE(6,*)'DO YOU WISH TO USE THIS CUTTER? (Y/N)'
READ(5,77) ANS

IF(ANS .EQ. 'Y') THEN
SDIAM=TD3
ENDIF

IF(ANS .EQ. 'N') THEN
WRITE(6,*)'THE DATA WILL BE GENERATED ACCORDING TO'
WRITE(6,*)'THE ORIGINAL SCALED CUTTER, BUT SINCE THIS'
WRITE(6,*)'CUTTER IS NOT AVAILABLE ON DYNA, YOU WILL'
WRITE(6,*)'HAVE TO MACHINE THE PART WITH A CUTTER'
WRITE(6,*)'CLOSE TO THIS SIZE'
ENDIF

GO TO 350
ENDIF

IF(SDIAM .GT. TD4 .AND. SDIAM .LT. TD5) THEN
WRITE(6,*)'WHICH IS CLOSE TO'
WRITE(6,76) TD4
WRITE(6,*)'AVAILABLE ON DYNA'
WRITE(6,*)'DO YOU WISH TO USE THIS CUTTER? (Y/N)'
READ(5,77) ANS

IF(ANS .EQ. 'Y') THEN
SDIAM=TD4
ENDIF

IF(ANS .EQ. 'N') THEN
```

```
WRITE(6,*)'THE DATA WILL BE GENERATED ACCORDING TO'
WRITE(6,*)'THE ORIGINAL SCALED CUTTER, BUT SINCE THIS'
WRITE(6,*)'CUTTER IS NOT AVAILABLE ON DYNA, YOU WILL'
WRITE(6,*)'HAVE TO MACHINE THE PART WITH A CUTTER'
WRITE(6,*)'CLOSE TO THIS SIZE'
ENDIF

GO TO 350
ENDIF

IF(SDIAM .GT. TD5 .AND. SDIAM .LT. TD6) THEN
WRITE(6,*)'WHICH IS CLOSE TO'
WRITE(6,76) TD5
WRITE(6,*)'AVAILABLE ON DYNA'
WRITE(6,*)'DO YOU WISH TO USE THIS CUTTER? (Y/N)'
READ(5,77) ANS

IF(ANS .EQ. 'Y') THEN
SDIAM=TD5
ENDIF

IF(ANS .EQ. 'N') THEN
WRITE(6,*)'THE DATA WILL BE GENERATED ACCORDING TO'
WRITE(6,*)'THE ORIGINAL SCALED CUTTER, BUT SINCE THIS'
WRITE(6,*)'CUTTER IS NOT AVAILABLE ON DYNA, YOU WILL'
WRITE(6,*)'HAVE TO MACHINE THE PART WITH A CUTTER'
WRITE(6,*)'CLOSE TO THIS SIZE'
ENDIF

GO TO 350
ENDIF

IF(SDIAM .GT. TD6 .AND. SDIAM .LT. TD7) THEN
WRITE(6,*)'WHICH IS CLOSE TO'
WRITE(6,76) TD6
WRITE(6,*)'AVAILABLE ON DYNA'
WRITE(6,*)'DO YOU WISH TO USE THIS CUTTER? (Y/N)'
READ(5,77) ANS

IF(ANS .EQ. 'Y') THEN
SDIAM=TD6
ENDIF

IF(ANS .EQ. 'N') THEN
WRITE(6,*)'THE DATA WILL BE GENERATED ACCORDING TO'
WRITE(6,*)'THE ORIGINAL SCALED CUTTER, BUT SINCE THIS'
WRITE(6,*)'CUTTER IS NOT AVAILABLE ON DYNA, YOU WILL'
WRITE(6,*)'HAVE TO MACHINE THE PART WITH A CUTTER'
WRITE(6,*)'CLOSE TO THIS SIZE'
ENDIF

GO TO 350
ENDIF

IF(SDIAM .GT. TD7 .AND. SDIAM .LT. TD8) THEN
WRITE(6,*)'WHICH IS CLOSE TO'
WRITE(6,76) TD7
WRITE(6,*)'AVAILABLE ON DYNA'
WRITE(6,*)'DO YOU WISH TO USE THIS CUTTER? (Y/N)'
READ(5,77) ANS

IF(ANS .EQ. 'Y') THEN
SDIAM=TD7
ENDIF

IF(ANS .EQ. 'N') THEN
WRITE(6,*)'THE DATA WILL BE GENERATED ACCORDING TO'
```

```
      WRITE(6,*)'THE ORIGINAL SCALED CUTTER, BUT SINCE THIS'
      WRITE(6,*)'CUTTER IS NOT AVAILABLE ON DYNA, YOU WILL'
      WRITE(6,*)'HAVE TO MACHINE THE PART WITH A CUTTER'
      WRITE(6,*)'CLOSE TO THIS SIZE'
      ENDIF

      GO TO 350
      ENDIF

      IF(SDIAM .GT. TD8) THEN
      WRITE(6,*)'WHICH IS CLOSE TO'
      WRITE(6,76) TD8
      WRITE(6,*)'AVAILABLE ON DYNA'
      WRITE(6,*)'DO YOU WISH TO USE THIS CUTTER? (Y/N)'
      READ(5,77) ANS

      IF(ANS .EQ. 'Y') THEN
      SDIAM=TD8
      ENDIF

      IF(ANS .EQ. 'N') THEN
      WRITE(6,*)'THE DATA WILL BE GENERATED ACCORDING TO'
      WRITE(6,*)'THE ORIGINAL SCALED CUTTER, BUT SINCE THIS'
      WRITE(6,*)'CUTTER IS NOT AVAILABLE ON DYNA, YOU WILL'
      WRITE(6,*)'HAVE TO MACHINE THE PART WITH A CUTTER'
      WRITE(6,*)'CLOSE TO THIS SIZE'
      ENDIF

      GO TO 350
      ENDIF

   77 FORMAT(A1)
  350 WRITE(6,*)'*******************************************'
      RETURN
      END
************************************************************************
      SUBROUTINE FINIS(IMP)
************************************************************************
C     THIS SUBROUTINE TAKES THE FINI STATEMENT FROM THE INPUT FILE AND
C     CONVERTS IT TO THE END STATEMENT FOR THE DYNA MILLING MACHINE
************************************************************************
      CHARACTER*80 IMP
      INTEGER OUT
      OUT=12
      WRITE(20,78)'END'
   78 FORMAT(A3)
      RETURN
      END
```

# *Program MODDYN*

```
************************************************************************
      PROGRAM  MODDYN
************************************************************************
C     THIS PROGRAM INTERFACES THE CADAM NC II FUNCTION TO THE DYNAMYTE
C     MILLING MACHINE USING THE CIRCLE SUBROUTINE PREPROGRAMMED INTO
C     THE DYNA MACHINE CONTROLLER
************************************************************************
C     WRITTEN BY: HARINDER SINGH OBEROI
************************************************************************
C     N=NUMBER OF POINTS
C     KCOUNT=FLAG TO IDENTIFY THE TYPE OF STATEMENT
C     KFLAG=FLAG TO DETERMINE THE NUMBER OF CUTTERS
C     KFLAG1=FLAG TO IDENTIFY CUTTER STATEMENT
C     KFLAG2=FLAG TO IDENTIFY FEEDRATE STATEMENT
C     KFLAG3=FLAG TO IDENTIFY SPINDLE ON STATEMENT
C     KFLAG4=FLAG TO IDENTIFY SPINDLE OFF STATEMENT
C     KFLAG5=FLAG TO IDENTIFY SURFACE STATEMENT
************************************************************************
      REAL TD, ANG
      DIMENSION X(5000), Y(5000), Z(5000)
      INTEGER N, IN, OUT, KCOUNT(100), KFLAG, M, J, KFLAG1(0:5000), K
      INTEGER KFLAG2(0:5000),KFLAG3(0:5000),KFLAG4(0:5000),L,D(100)
      INTEGER KFLAG5(0:5000),P, Q, R, FLG
      REAL XC,YC,ZC
      CHARACTER*80 INFILE, IMP, TEMP, DUMMY
      CHARACTER*25 BLANK, NUB*3, ANS*1, ANS1*1
      CHARACTER*20 FINAL
      DATA BLANK/'                         '/
************************************************************************
      IN = 9
      OUT = 12
      WRITE(6,*)'****************************************'
      WRITE(6,*)'THE RANGE OF THE DYNA MILLING MACHINE IS'
      WRITE(6,*)'X = 5.5 INCHES'
      WRITE(6,*)'Y = 4.5 INCHES'
      WRITE(6,*)'Z = 4.0 INCHES'
      WRITE(6,*)'****************************************'
      WRITE(6,*)'****************************************'
      WRITE(6,*)'DOES THE PART TO BE MACHINED LIE'
      WRITE(6,*)'WITHIN THESE PHYSICAL LIMITS'
      WRITE(6,*)'ENTER (Y/N)'
      WRITE(6,*)'****************************************'
      READ(5,7) ANS
    7 FORMAT(A1)

      IF(ANS .EQ. 'Y') THEN
      WRITE(6,*)'***************************************'
      WRITE(6,*)'THE CUTTER SIZES AVAILABLE ON DYNA ARE'
      WRITE(6,*)'0.03125, 0.0625, 0.09375, 0.1875'
      WRITE(6,*)'0.25, 0.3125, 0.375, 0.5'
      WRITE(6,*)'***************************************'
      WRITE(6,*)'THE CUTTER SPECIFIED SHOULD BE EITHER'
      WRITE(6,*)'ONE OF THESE OR CLOSE TO ONE OF THESE'
      WRITE(6,*)'SIZES'
      WRITE(6,*)'***************************************'
      WRITE(6,*)'IF THE CUTTER SIZE SPECIFIED IS'
      WRITE(6,*)'NOT ONE OF THESE'
      WRITE(6,*)'DO YOU WISH TO GO BACK AND CHANGE THE'
      WRITE(6,*)'CUTTER SIZE WHILE GENERATING THE'
      WRITE(6,*)'TOOL PATH  ENTER (Y/N)'
      WRITE(6,*)'***************************************'
      READ(5,8) ANS1
    8 FORMAT(A1)
```

```
      IF(ANS1 .EQ. 'Y') THEN
      GO TO 6000
      ENDIF
      IF(ANS1 .EQ. 'N') THEN
      GO TO 6001
      ENDIF

      ENDIF

      IF(ANS .EQ. 'N') THEN
      GO TO 6001
      ENDIF
************************************************************************
 6001 WRITE(6,*)'****************************************'
      WRITE (6,*) 'PLEASE ENTER THE PROGRAM NUMBER'
      WRITE (6,*) 'IT SHOULD LIE BETWEEN 00 AND 99'
      WRITE(6,*)'****************************************'
      READ (5,*) N
      WRITE (OUT,9)'001 START INS', N
    9 FORMAT (A13,1X,I2)
************************************************************************
C     THIS IS THE MAIN PROGRAM WHICH READS FROM THE INPUT FILE AND CALLS
C     THE VARIOUS SUBROUTINES ASSOCIATED WITH EACH STATEMENT READ
************************************************************************
      M=0

 1000 DO 100 I = 1,100

      READ (IN,10) INFILE
   10 FORMAT(A80)

      IF(INFILE(1:6) .EQ. 'CUTTER') THEN
      KCOUNT(I)=1
      CALL TOOL(INFILE)
      M=M+1
      ENDIF

      IF(INFILE(1:6) .EQ. 'MCHTOL') THEN
      M=M+1
      ENDIF

      IF(INFILE(1:6) .EQ. 'FEDRAT') THEN
      KCOUNT(I)=2
      CALL FEED(INFILE)
      M=M+1
      ENDIF

      IF(INFILE(1:25) .EQ. BLANK) THEN
      GO TO 1000
      ENDIF


      IF(INFILE(3:6) .EQ. 'FROM') THEN
      KCOUNT(I)=3
      M=M+1
      CALL SETUP(INFILE)
      CALL POINT(KFLAG,KFLAG1,KFLAG2,KFLAG3,KFLAG4,KFLAG5,D,N,L)
      GO TO 110
      ENDIF

  100 CONTINUE
  110 CONTINUE
************************************************************************
C     THIS PART OF THE PROGRAM REWINDS ALL THE INTERNAL FILES CREATED
C     CALLS THE MOTION SUBROUTINE
************************************************************************
```

```
      REWIND(13)
      REWIND(14)
      REWIND(15)
      REWIND(16)
      REWIND(17)
      REWIND(18)
      REWIND(25)
      CALL MOTION(KFLAG,KFLAG1,L,N,X,Y,Z)
      REWIND(19)
      REWIND(20)
      REWIND(26)
      REWIND(28)
************************************************************************
C     THIS PART OF THE PROGRAM WRITES THE TOOL DIAMETER, FEEDRATE, AND
C     SPINDLE ON AND SPINDLE OFF STATEMENT TO A INTERNAL FILE
************************************************************************
C     TEMP=TEMPORARY CHARACTER STRING
************************************************************************
      DO 120 I=1,M

      IF(KCOUNT(I) .EQ. 1) THEN
      READ(19,11) TEMP
      WRITE(23,12) TEMP
      ENDIF

      IF(KCOUNT(I) .EQ. 2) THEN
      READ(14,11) TEMP
      WRITE(23,12) TEMP
      ENDIF

      IF(KCOUNT(I) .EQ. 3) THEN
      READ(17,11) TEMP
      WRITE(23,12) TEMP
      ENDIF

   11 FORMAT(A80)
   12 FORMAT(A80)
  120 CONTINUE
************************************************************************
C     THIS PART OF THE PROGRAM WRITES THE MOTION STATEMENTS TO AN
C     INTERNAL FILE
************************************************************************
************************************************************************
      READ(14,13) TEMP
   13 FORMAT(A80)
      WRITE(23,13) TEMP
      WRITE(23,14)'GO X',X(2)
   14 FORMAT(A4,1X,F7.4)
      WRITE(23,15)'GO Y',Y(2)
   15 FORMAT(A4,1X,F7.4)
      WRITE(23,16)'SPINDLE ON'
   16 FORMAT(A10)
      WRITE(23,17)'DWELL 04'
   17 FORMAT(A8)
      WRITE(23,18)'GO Z',Z(2)
   18 FORMAT(A4,1X,F7.4)
************************************************************************
      P=0
      DO 130 I=3,N-1

      DO 3020 J=1,5

      IF(KFLAG1(I) .EQ. J) THEN
      READ(19,19) TEMP
   19 FORMAT(A80)
      WRITE(23,20) TEMP
   20 FORMAT(A80)
```

```
      READ(19,19) TEMP
      WRITE(23,20) TEMP
      READ(19,19) TEMP
      WRITE(23,20) TEMP
      READ(19,19) TEMP
      WRITE(23,20) TEMP
      ENDIF

      IF(KFLAG2(I) .EQ. J) THEN
      READ(14,19) TEMP
      WRITE(23,20) TEMP
      ENDIF

      IF(KFLAG3(I) .EQ. J) THEN
      READ(15,21) TEMP
   21 FORMAT(A80)
      WRITE(23,22) TEMP
   22 FORMAT(A80)
      READ(15,21) TEMP
      WRITE(23,22) TEMP
      ENDIF

      IF(KFLAG4(I) .EQ. J) THEN
      READ(16,23) TEMP
   23 FORMAT(A80)
      WRITE(23,24) TEMP
   24 FORMAT(A80)
      ENDIF

      IF(KFLAG5(I) .EQ. J) THEN
      WRITE(23,114) 'ZERO AT'
  114 FORMAT(2X,A7)
      READ(28,115) XC,YC,ZC
  115 FORMAT(10X,F7.4,4X,F7.4,4X,F7.4)
      WRITE(23,116) 'X', XC
      WRITE(23,116) 'Y', YC
  116 FORMAT(2X,A1,1X,F7.4)
      READ(26,117) ANG
  117 FORMAT(1X,F10.4)
      WRITE(23,118) 'GR a', ANG
  118 FORMAT(1X,A4,1X,F10.4)
      WRITE(23,119) '> REF COODS'
  119 FORMAT(1X,A11)
      FLG=1
      P=P+1
      Q=I+D(P)-1
      ENDIF

 3020 CONTINUE

      R=I
      IF(FLG .EQ. 1) THEN
C     R=R+1
      IF(R .EQ. Q) THEN
      FLG=0
      ENDIF
      GO TO 130
      ENDIF

      IF(Z(I) .GT. 0.5) THEN
      WRITE(23,25)'GO Z', Z(I)
      GO TO 130
      ENDIF

      IF(Z(I-1) .GT. 0.5) THEN
      WRITE(23,25)'GO X', X(I)
      WRITE(23,25)'GO Y', Y(I)
```

```
      WRITE(23,25)'GO Z', Z(I)
      GO TO 130
      ENDIF

      IF(X(I) .EQ. X(I-1) .AND. Y(I) .EQ. Y(I-1) .AND.
     1    Z(I) .EQ. Z(I-1)) THEN
      GO TO 130
      ENDIF

      IF(X(I) .EQ. X(I-1) .AND. Y(I) .EQ. Y(I-1)) THEN
      WRITE(23,25)'GO Z',Z(I)
   25 FORMAT(A4,1X,F7.4)
      GO TO 130
      ENDIF

      IF(X(I) .EQ. X(I-1) .AND. Z(I) .EQ. Z(I-1)) THEN
      WRITE(23,26)'GO Y',Y(I)
   26 FORMAT(A4,1X,F7.4)
      GO TO 130
      ENDIF

      IF(Y(I) .EQ. Y(I-1) .AND. Z(I) .EQ. Z(I-1)) THEN
      WRITE(23,27)'GO X',X(I)
   27 FORMAT(A4,1X,F7.4)
      GO TO 130
      ENDIF

      IF(X(I) .EQ. X(I-1)) THEN
      WRITE(23,31)'GO Y',Y(I)
   31 FORMAT(A4,1X,F7.4)
      WRITE(23,32)'   Z',Z(I)
   32 FORMAT(A4,1X,F7.4)
      GO TO 130
      ENDIF

      IF(Y(I) .EQ. Y(I-1)) THEN
      WRITE(23,33)'GO X',X(I)
   33 FORMAT(A4,1X,F7.4)
      WRITE(23,34)'   Z',Z(I)
   34 FORMAT(A4,1X,F7.4)
      GO TO 130
      ENDIF

      IF(Z(I) .EQ. Z(I-1)) THEN
      WRITE(23,35)'GO X',X(I)
   35 FORMAT(A4,1X,F7.4)
      WRITE(23,36)'   Y',Y(I)
   36 FORMAT(A4,1X,F7.4)
      GO TO 130
      ENDIF

      WRITE(23,37)'GO X',X(I)
   37 FORMAT(A4,1X,F7.4)
      WRITE(23,38)'   Y',Y(I)
   38 FORMAT(A4,1X,F7.4)
      WRITE(23,39)'   Z',Z(I)
   39 FORMAT(A4,1X,F7.4)

  130 CONTINUE
**********************************************************************
      WRITE(23,40)'GO Z',Z(1)
   40 FORMAT(A4,1X,F7.4)
      WRITE(23,41)'GO X',X(N)
   41 FORMAT(A4,1X,F7.4)
      WRITE(23,42)'GO Y',Y(N)
   42 FORMAT(A4,1X,F7.4)
```

```
      WRITE(23,'(A11)')'SPINDLE OFF'
***********************************************************************
      READ(20,43) TEMP
   43 FORMAT(A80)
      WRITE(23,44) TEMP
   44 FORMAT(A80)
***********************************************************************
C     THIS PART OF THE PROGRAM DETERMINES THE LENGTH OF THE MOTION
C     STATEMENT FILE
***********************************************************************
      REWIND(23)
      K=1

      DO 140 I=1,2000

      READ(23,45) DUMMY
   45 FORMAT(A80)

      IF(DUMMY(1:3) .EQ. 'END') THEN
      GO TO 150
      ENDIF

      K=K+1
  140 CONTINUE
***********************************************************************
C     THIS PART OF THE PROGRAM READS THE FINAL INTERNAL FILE, WRITES
C     STATEMENT NUMBERS IN FRONT OF EACH STATEMENT AND WRITES IT TO
C     THE FINAL OUTPUT FILE
***********************************************************************
  150 REWIND(23)
      J=2

      DO 160 I=1,K

      READ(23,46) FINAL
   46 FORMAT(A20)

      IF(J .LT. 10) THEN
      WRITE(NUB,'(2H00,I1)')J
      ENDIF

      IF(J .GT. 9 .AND. J .LT. 100) THEN
      WRITE(NUB,'(1H0,I2)')J
      ENDIF

      IF(J .GE. 100 .AND. J .LT. 1000) THEN
      WRITE(NUB,'(I3)')J
      ENDIF

      WRITE(OUT,47) NUB,FINAL
   47 FORMAT(A3,1X,A20)
      J=J+1

  160 CONTINUE

 6000 END
***********************************************************************
      SUBROUTINE TOOL(IMP)
***********************************************************************
C     THIS SUBROUTINE CONVERTS THE CUTTER STATEMENT FROM THE INPUT FILE
C     TO THE TOOL DIAMETER STATEMENT IN DYNA
***********************************************************************
C     DIAM=TOOL DIAMETER READ FROM THE INPUT FILE
***********************************************************************
      CHARACTER*80 IMP
      REAL TD, DIAM
```

```
      READ(IMP,48) DIAM
   48 FORMAT(8X,F10.5)
      WRITE(13,49)'TD=', DIAM
   49 FORMAT(A4,1X,F6.4)
      RETURN
      END
************************************************************************
      SUBROUTINE FEED(IMP)
************************************************************************
C    THIS SUBROUTINE TAKES THE FEEDRATE STATEMENT FROM THE INPUT FILE
C    AND CHECKS WHETHER IT LIES IN THE RANGE OF FEEDRATES THAT CAN BE
C    SPECIFIED FOR THE DYNA MILLING MACHINE AND WRITES THE CORRESPONDING
C    DYNA FEEDRATE STATEMENT
************************************************************************
C     FER=FEEDRATE READ FROM THE INPUT FILE
************************************************************************
      CHARACTER*80 IMP
      REAL FER

      READ(IMP,50) FER
   50 FORMAT(8X,F10.5)

      IF(FER .GT. 0.10 .AND. FER .LT. 32.0) THEN
      GO TO 170
      ENDIF

 3030 WRITE(6,*)'THE FEEDRATE'
      WRITE(6,'(8X,F10.5)') FER
      WRITE(6,*)'CHOSEN IS OUT OF THE RANGE FOR THE DYNA'
      WRITE(6,*)'MILLING MACHINE, PLEASE ENTER A FEEDRATE WHICH LIES'
      WRITE(6,*)'IN THE RANGE OF 0.10 TO 32.0 IPM'
      READ(5,*) FER

      IF(FER .GT. 0.10 .AND. FER .LT. 32.0) THEN
      GO TO 180
      ENDIF

      GO TO 3030

  170 WRITE(14,51)'FR XYZ=', FER
   51 FORMAT(A7,1X,F8.4)
      GO TO 190
  180 WRITE(14,52)'FR XYZ=', FER
   52 FORMAT(A7,1X,F8.4)
  190 RETURN
      END
************************************************************************
      SUBROUTINE SPINON
************************************************************************
C     THIS SUBROUTINE TAKES THE SPINDLE ON STATEMENT FROM THE INPUT FILE
C     AND CONVERTS IT TO THE SPINDLE ON STATEMENT FOR THE DYNA MILLING
C     MACHINE
************************************************************************
      CHARACTER*10 BLOCK
      DATA BLOCK/'SPINDLE ON'/
      WRITE(15,53) BLOCK
   53 FORMAT(A10)
      WRITE(15,54)'DWELL 04'
   54 FORMAT(A8)
      RETURN
      END
************************************************************************
      SUBROUTINE SPIOFF
************************************************************************
C     THIS SUBROUTINE TAKES THE SPINDLE OFF STATEMENT FROM THE INPUT
C     FILE AND CONVERTS IT TO THE SPINDLE OFF STATEMENT FOR THE DYNA
```

```
C     MILLING MACHINE
************************************************************************
      CHARACTER*11 BLOCK
      DATA BLOCK/'SPINDLE OFF'/
      WRITE(16,54) BLOCK
   54 FORMAT(A11)
      RETURN
      END
************************************************************************
      SUBROUTINE SETUP(IMP)
************************************************************************
C     THIS SUBROUTINE TAKES THE FROM STATEMENT IN THE INPUT FILE AND
C     CONVERTS IT TO THE SETUP STATEMENT FOR THE DYNA MILLING MACHINE
************************************************************************
      CHARACTER*80 IMP,DUMMY
      INTEGER OUT,IN
      IN=9
      OUT=12
      WRITE(17,55)'SETUP > ZCXYU'
   55 FORMAT(A13)
      RETURN
      END
************************************************************************
      SUBROUTINE POINT(KFLAG,KFLAG1,KFLAG2,KFLAG3,KFLAG4,KFLAG5,D,N,L)
************************************************************************
C     THIS SUBROUTINE TAKES THE X, Y, AND Z COORDINATES AND WRITES THEM
C     TO A FILE
************************************************************************
C     KFLAG=FLAG TO DETERMINE THE NUMBER OF CUTTERS
C     KFLAG1=FLAG TO IDENTIFY CUTTER STATEMENT
C     KFLAG2=FLAG TO IDENTIFY FEEDRATE STATEMENT
C     KFLAG3=FLAG TO IDENTIFY SPINDLE ON STATEMENT
C     KFLAG4=FLAG TO IDENTIFY SPINDLE OFF STATEMENT
C     KFLAG5=FLAG TO IDENTIFY SURFACE STATEMENT
C     N=NUMBER OF POINTS
************************************************************************
      CHARACTER*80 INFIL
      CHARACTER*25 BLANK
      INTEGER N, M, KFLAG, IN, KFLAG1(0:5000),KFLAG2(0:5000)
      INTEGER L, D(100), KFLAG5(0:5000), E, H
      INTEGER KFLAG3(0:5000),KFLAG4(0:5000)
      REAL A,B,C
************************************************************************
      DATA BLANK/'                         '/
      N=0
      IN=9
      KFLAG=0
      M=1
      L=0
 2001 DO 210 I=1,5000
      READ(IN,56) INFIL
   56 FORMAT(A80)

      IF(INFIL(1:25) .EQ. BLANK) THEN
      GO TO 2001
      ENDIF

      IF(INFIL(3:6) .EQ. 'GOTO') THEN
      GO TO 2001
      ENDIF

      IF(INFIL(2:3) .EQ. 'GO') THEN
      GO TO 2001
      ENDIF

      IF(INFIL(1:4) .EQ. 'STOP') THEN
      GO TO 2001
```

```
      ENDIF

      IF(INFIL(1:6) .EQ. 'LEADER') THEN
      GO TO 2001
      ENDIF

      IF(INFIL(1:6) .EQ. 'TRACUT') THEN
      DO 700 H=1,3
      READ(IN,56) INFIL
  700 CONTINUE
      GO TO 2001
      ENDIF

      IF(INFIL(1:3) .EQ. 'END') THEN
      GO TO 2001
      ENDIF

      IF(INFIL(3:6) .EQ. 'FROM') THEN
      GO TO 2001
      ENDIF


      DO 3000 J=1,5

      IF(INFIL(1:6) .EQ. 'CUTTER') THEN
      CALL TOOL(INFIL)
      KFLAG1(N)=M
      KFLAG=KFLAG+1
      GO TO 200
      ENDIF

      IF(INFIL(1:6) .EQ. 'FEDRAT') THEN
      CALL FEED(INFIL)
      KFLAG2(N)=M
      GO TO 200
      ENDIF

      IF(INFIL(1:6) .EQ. 'SPINDL') THEN
      IF(INFIL(16:17) .EQ. 'ON') THEN
      CALL SPINON
      KFLAG3(N)=M
      GO TO 200
      ELSE
      CALL SPIOFF
      KFLAG4(N)=M
      GO TO 200
      ENDIF
      ENDIF
      IF(INFIL(1:7) .EQ. 'SURFACE') THEN
      L=L+1
      CALL CIRCLE(IMP,L,D)
      KFLAG5(N+1)=M
      N=N+D(L)
      GO TO 200
      ENDIF

      GO TO 3001
 3000 CONTINUE

  200 M=M+1
      GO TO 2001

 3001 IF(INFIL(1:4) .EQ. 'FINI') THEN
      CALL FINIS(INFIL)
      GO TO 220
      ENDIF
```

```
      READ(INFIL,57) A, B, C
   57 FORMAT(10X,F8.4,3X,F8.4,3X,F8.4)
      WRITE(18,58) A, B, C
   58 FORMAT(10X,F8.4,3X,F8.4,3X,F8.4)
      N=N+1
      M=1
      GO TO 2001

  210 CONTINUE
  220 RETURN
      END
*************************************************************************
      SUBROUTINE CIRCLE(IMP,L,D)
*************************************************************************
C     THIS SUBROUTINE TAKES THE CIRCULAR SURFACE DATA AND WRITES IT TO
C     THE OUTPUT FILE
*************************************************************************
      CHARACTER* 80 IMP, INFI
      CHARACTER* 25 BLANK
      INTEGER IN, OUT, L, D(100), K, E, FLG
      REAL XC,YC,ZC, X(200),Y(200),Z(200),DOT, RAD, XC1,YC1,ZC1,RAD1
      REAL X1,Y1,Z1,X2,Y2,Z2,XK,YK,ZK,MAG1,MAG2,MAGK,XFC,YFC,ZFC
      REAL XFC1,YFC1,ZFC1,MAGC,MAGC1,THETA,THETA1,DIV,PI
      REAL G,H,F,TEST1
      DATA BLANK/'                         '/
*************************************************************************
      IN=9
      OUT=12
      PI=3.1415926
      READ(IN,59) INFI
      READ(INFI,60) XC,YC,ZC
      WRITE(25,60) XC,YC,ZC
   59 FORMAT(A80)
      READ(IN,59) INFI
      READ(INFI,63) G,H,F,RAD
   63 FORMAT(10X,F7.4,4X,F7.4,4X,F7.4,4X,F7.4)
      K=0
      FLG=0
*************************************************************************
 3002 DO 201 I=1,200
      READ(IN,59) INFI

      IF(INFI(1:25) .EQ. BLANK) THEN
      READ(IN,59)INFI
      ENDIF

      IF(INFI(1:7) .EQ. 'SURFACE') THEN
      READ(IN,59)INFI
      READ(INFI,60)XC1,YC1,ZC1
      READ(IN,59)INFI
      READ(INFI,63)G,H,F,RAD1
      IF(XC1 .EQ. XC .AND. YC1 .EQ. YC .AND. ZC1 .EQ. ZC .AND.
     1   RAD1 .EQ. RAD) THEN
3004  READ(IN,59)INFI
      IF(INFI(3:6) .NE. 'GOTO') GO TO 3004
      READ(IN,59)INFI
      GO TO 3003
      ENDIF
      E=1
      ENDIF
      IF(INFI(3:6) .EQ. 'GOTO'.AND. FLG .NE. 0 ) THEN
      GO TO 3000
      ENDIF
      IF(INFI(3:6) .EQ. 'GOTO' .AND. K .EQ. 0 ) THEN
           GO TO 3002
      ENDIF
```

```
 3003 READ(INFI,60) X(I),Y(I),Z(I)
      WRITE(18,60) X(I), Y(I), Z(I)
   60 FORMAT(10X,F7.4,4X,F7.4,4X,F7.4)
      K=K+1
      FLG=1
  201 CONTINUE
*************************************************************************
 3000 D(L)=K
      X1=X(1) - XC
      Y1=Y(1) - YC
      X2=X(2) - XC
      Y2=Y(2) - YC
      XK=X(K) - XC
      YK=Y(K) - YC
      DOT= (X1*XK) + (Y1*YK)
      MAG1=SQRT(X1**2 + Y1**2)
      MAG2=SQRT(X2**2 + Y2**2)
      MAGK=SQRT(XK**2 + YK**2)
      ZFC=X1*Y2 - Y1*X2
      ZFC1=X1*YK - Y1*XK
      MAGC=SQRT(ZFC**2)
      MAGC1=SQRT(ZFC1**2)
      THETA=ACOS(DOT/(MAG1*MAGK))
      DIV=ZFC/ZFC1

      IF(DIV .LT. 0.) THEN
      THETA = 2.*PI - THETA
      ENDIF

      IF(ZFC .LT. 0.) THEN
      THETA1= - (180.0/PI) * THETA
      IF(ABS(THETA1) .GT. 335.0) THEN
      THETA1=-360.0
      ENDIF
      ENDIF

      IF(ZFC .GT. 0.) THEN
      THETA1=  (180.0/PI) * THETA
      IF(ABS(THETA1) .GT. 335.0) THEN
      THETA1=360.0
      ENDIF
      ENDIF

      WRITE(26,61) THETA1
   61 FORMAT(1X,F10.4)
 3001 RETURN
      END
*************************************************************************
      SUBROUTINE MOTION(KFLAG,KFLAG1,L,N,X,Y,Z)
*************************************************************************
C     THIS SUBROUTINE TAKES THE GOTO STATEMENT FROM THE INPUT FILE AND
C     CONVERTS IT TO THE MOTION STATEMENT FOR THE DYNA MILLING MACHINE
*************************************************************************
C     SCALX=SCALE FACTOR IN THE X DIRECTION
C     SCALY=SCALE FACTOR IN THE Y DIRECTION
C     SCALZ=SCALE FACTOR IN THE Z DIRECTION
C     DRANX=RANGE OF DYNA IN THE X DIRECTION
C     DRANY=RANGE OF DYNA IN THE Y DIRECTION
C     DRANZ=RANGE OF DYNA IN THE Z DIRECTION
C     TD1 TO TD8= EIGHT CUTTERS AVAILABLE ON DYNA MACHINE
C     FSCAL=FINAL SCALE FACTOR TO BE USED FOR SCALING THE PART DOWN
C     DIAM1 TO DIAM8= TOOL DIAMETERS READ FROM THE INPUT FILE
*************************************************************************
      DIMENSION A(5000),B(5000),C(5000)
      DIMENSION X(5000),Y(5000),Z(5000)
      INTEGER INT,N,J,KFLAG,KFLAG2,L
      REAL SCALX,SCALY,SCALZ,DRANX,DRANY,DRANZ,SCAL1,FSCAL
```

```
      REAL TD1,TD2,TD3,TD4,TD5,TD6,TD7,TD8,SDIAM
      REAL DIAM1, DIAM2, DIAM3, DIAM4
      REAL DIAM5, DIAM6, DIAM7, DIAM8
      REAL XRANGE,YRANGE,ZRANGE,FER,XZERO,YZERO
      REAL XC(100),YC(100),ZC(100),XNC(100),YNC(100),ZNC(100)
      DIMENSION Y1(10000)
      REAL XMIN,YMIN,XMAX,YMAX,ZMAX,ZMIN
      CHARACTER*1 ANS

      DRANX=5.5
      DRANY=4.5
      DRANZ=4.0
      TD1=0.03125
      TD2=0.0625
      TD3=0.09375
      TD4=0.1875
      TD5=0.25
      TD6=0.3125
      TD7=0.375
      TD8=0.5
************************************************************************
C     THIS PORTION OF THE SUBROUTINE READS IN THE X, Y, AND Z VALUES
C     FROM A FILE
************************************************************************
      DO 230 I=1,N

      READ(18,61) X(I),Y(I),Z(I)
   61 FORMAT(10X,F8.4,3X,F8.4,3X,F8.4)

  230 CONTINUE
************************************************************************
C     THIS PORTION OF THE SUBROUTINE PERFORMS THE COORDINATE
C     TRANSFORMATIONS BY MAKING THE SETUP X, Y, AND Z VALUES ZERO
************************************************************************
      XZERO=X(1)
      YZERO=Y(1)

      DO 240 I=1,N

      X(I)=X(I)-XZERO
      Y(I)=Y(I)-YZERO

  240 CONTINUE
************************************************************************
C     THIS PORTION OF THE SUBROUTINE CALCULATES THE MINIMUM X, Y, AND Z
C     VALUES
************************************************************************
      XMIN=X(1)
      YMIN=Y(1)
      ZMIN=Z(1)

      DO 250 I=2,N

      IF(X(I) .LT. XMIN) THEN
      XMIN=X(I)
      ENDIF

      IF(Y(I) .LT. YMIN) THEN
      YMIN=Y(I)
      ENDIF

      IF(Z(I) .LT. ZMIN) THEN
      ZMIN=Z(I)
      ENDIF

  250 CONTINUE
************************************************************************
```

```
C     THIS PORTION OF THE SUBROUTINE CALCULATES THE MAXIMUM X, Y, AND Z
C     VALUES
**********************************************************************
      XMAX=X(1)
      YMAX=Y(1)
      ZMAX=Z(1)

      DO 260 I=2,N

      IF(X(I) .GT. XMAX) THEN
      XMAX=X(I)
      ENDIF

      IF(Y(I) .GT. YMAX) THEN
      YMAX=Y(I)
      ENDIF

      IF(Z(I) .GT. ZMAX) THEN
      ZMAX=Z(I)
      ENDIF

  260 CONTINUE
**********************************************************************
C     THIS PORTION OF THE SUBROUTINE CALCULATES THE RANGE IN THE X, Y,
C     Z DIRECTION
**********************************************************************
      XRANGE=XMAX-XMIN
      YRANGE=YMAX-YMIN
      ZRANGE=ZMAX-ZMIN
**********************************************************************
C     THIS PORTION OF THE SUBROUTINE CALCULATES THE APPROXIMATE SCALE
C     VALUE IN THE X, Y, AND Z DIRECTION
**********************************************************************
      IF(XRANGE .GT. DRANX) THEN
      SCALX=XRANGE/DRANX
      ELSE
      SCALX=1.0
      ENDIF

      IF(YRANGE .GT. DRANY) THEN
      SCALY=YRANGE/DRANY
      ELSE
      SCALY=1.0
      ENDIF

      IF(ZRANGE .GT. DRANZ) THEN
      SCALZ=ZRANGE/DRANZ
      ELSE
      SCALZ=1.0
      ENDIF
**********************************************************************
C     THIS PORTION OF THE SUBROUTINE CALCULATES THE MAXIMUM OF THE
C     APPROXIMATE SCALE VALUES IN THE X, Y, AND Z DIRECTION
**********************************************************************
      IF(SCALX .GE. SCALY) THEN
      SCAL1=SCALX
      ENDIF

      IF(SCALY .GE. SCALX) THEN
      SCAL1=SCALY
      ENDIF

      IF(SCAL1 .LE. SCALZ) THEN
      SCAL1=SCALZ
      ENDIF
**********************************************************************
C     THIS PORTION OF THE SUBROUTINE READS IN THE TOOL DIAMETERS FROM A
```

```
C     INTERNAL FILE
*********************************************************************
      READ(13,62) DIAM1
      IF(KFLAG .GT. 0) THEN
      READ(13,62) DIAM2
      ENDIF

      IF(KFLAG .GT. 1) THEN
      READ(13,62) DIAM3
      ENDIF

      IF(KFLAG .GT. 2) THEN
      READ(13,62) DIAM4
      ENDIF

      IF(KFLAG .GT. 3) THEN
      READ(13,62) DIAM5
      ENDIF

      IF(KFLAG .GT. 4) THEN
      READ(13,62) DIAM6
      ENDIF

      IF(KFLAG .GT. 5) THEN
      READ(13,62) DIAM7
      ENDIF

      IF(KFLAG .GT. 6) THEN
      READ(13,62) DIAM8
      ENDIF

   62 FORMAT(5X,F6.4)
*********************************************************************
      IF(SCAL1 .EQ. 1.0) THEN
      WRITE(6,*)'******************************************'
      WRITE(6,*)'THE PART IS WITHIN THE RANGE OF DYNA '
      WRITE(6,*)'SO IT DOES NOT HAVE TO BE SCALED DOWN'
      WRITE(6,*)'******************************************'
      DO 410 I=1,L
      READ(25,112) XC(I),YC(I),ZC(I)
      WRITE(28,112)XC(I),YC(I),ZC(I)
  410 CONTINUE
  112 FORMAT(10X,F7.4,4X,F7.4,4X,F7.4)
      GO TO 270
      ENDIF
      GO TO 280
*********************************************************************
C     THIS PORTION OF THE SUBROUTINE WRITES OUT THE TOOL DIAMETERS IF
C     THE PART IS NOT TO BE SCALED DOWN
*********************************************************************
  270 WRITE(19,63)'TD=', DIAM1

      IF(KFLAG .GT.0) THEN
      WRITE(19,64)'TOOL 2'
      WRITE(19,65)'SPINDLE OFF'
      WRITE(19,63)'TD=', DIAM2
      WRITE(19,66)'SPINDLE ON'
      WRITE(19,103)'DWELL 04'
      ENDIF

      IF(KFLAG .GT.1) THEN
      WRITE(19,64)'TOOL 3'
      WRITE(19,65)'SPINDLE OFF'
      WRITE(19,63)'TD=', DIAM3
      WRITE(19,66)'SPINDLE ON'
      WRITE(19,103)'DWELL 04'
      ENDIF
```

```
      IF(KFLAG .GT.2) THEN
      WRITE(19,64)'TOOL 4'
      WRITE(19,65)'SPINDLE OFF'
      WRITE(19,63)'TD=', DIAM4
      WRITE(19,66)'SPINDLE ON'
      WRITE(19,103)'DWELL 04'
      ENDIF

      IF(KFLAG .GT.3) THEN
      WRITE(19,64)'TOOL 5'
      WRITE(19,65)'SPINDLE OFF'
      WRITE(19,63)'TD=', DIAM5
      WRITE(19,66)'SPINDLE ON'
      WRITE(19,103)'DWELL 04'
      ENDIF

      IF(KFLAG .GT.4) THEN
      WRITE(19,64)'TOOL 6'
      WRITE(19,65)'SPINDLE OFF'
      WRITE(19,63)'TD=', DIAM6
      WRITE(19,66)'SPINDLE ON'
      WRITE(19,103)'DWELL 04'
      ENDIF

      IF(KFLAG .GT.5) THEN
      WRITE(19,64)'TOOL 7'
      WRITE(19,65)'SPINDLE OFF'
      WRITE(19,63)'TD=', DIAM7
      WRITE(19,66)'SPINDLE ON'
      WRITE(19,103)'DWELL 04'
      ENDIF

      IF(KFLAG .GT.6) THEN
      WRITE(19,64)'TOOL 8'
      WRITE(19,65)'SPINDLE OFF'
      WRITE(19,63)'TD=', DIAM8
      WRITE(19,66)'SPINDLE ON'
      WRITE(19,103)'DWELL 04'
      ENDIF

   63 FORMAT(A4,1X,F6.4)
   64 FORMAT(A6)
   65 FORMAT(A11)
   66 FORMAT(A10)
  103 FORMAT(A8)

      FSCAL=1.0
      GO TO 310
*************************************************************************
C     THIS PORTION OF THE SUBROUTINE CALCULATES THE SCALE VALUE AND
C     CORRESPONDINGLY SCALES THE TOOL DIAMETER
*************************************************************************
  280 WRITE(6,*)'*******************************************'
      WRITE(6,*)'THE PART SHOULD BE SCALED DOWN'
      WRITE(6,67)SCAL1
   67 FORMAT(1X,F6.4)
      WRITE(6,*)'TIMES, SO THAT IT CAN BE MACHINED ON DYNA'
      WRITE(6,*)'*******************************************'
*************************************************************************
      DO 290 I=1,5

      KFLAG2=1
      CALL CUTTER(DIAM1,SCAL1,KFLAG2,SDIAM)
      SCAL1=DIAM1/SDIAM

      IF(KFLAG .GT. 0) THEN
```

```
      KFLAG2=2
      CALL CUTTER(DIAM2,SCAL1,KFLAG2,SDIAM)
      ENDIF

      IF(KFLAG .GT. 1) THEN
      KFLAG2=3
      CALL CUTTER(DIAM3,SCAL1,KFLAG2,SDIAM)
      ENDIF

      IF(KFLAG .GT. 2) THEN
      KFLAG2=4
      CALL CUTTER(DIAM4,SCAL1,KFLAG2,SDIAM)
      ENDIF

      IF(KFLAG .GT. 3) THEN
      KFLAG2=5
      CALL CUTTER(DIAM5,SCAL1,KFLAG2,SDIAM)
      ENDIF

      IF(KFLAG .GT. 4) THEN
      KFLAG2=6
      CALL CUTTER(DIAM6,SCAL1,KFLAG2,SDIAM)
      ENDIF

      IF(KFLAG .GT. 5) THEN
      KFLAG2=7
      CALL CUTTER(DIAM7,SCAL1,KFLAG2,SDIAM)
      ENDIF

      IF(KFLAG .GT. 6) THEN
      KFLAG2=8
      CALL CUTTER(DIAM8,SCAL1,KFLAG2,SDIAM)
      ENDIF
************************************************************************
      WRITE(6,*)'THE SCALE IS '
      WRITE(6,68)SCAL1
   68 FORMAT(1X,F6.4)
      WRITE(6,*)'DO YOU WISH TO USE THE SAME SCALE? (Y/N)'
      READ(5,69) ANS
   69 FORMAT(A1)

      IF(ANS .EQ. 'Y') THEN
      GO TO 300
      ENDIF

      IF(ANS .EQ. 'N') THEN
      WRITE(6,*)'PLEASE ENTER THE SCALE YOU WISH TO USE'
      READ(5,70) SCAL1
   70 FORMAT(F6.4)
      ENDIF

  290 CONTINUE
************************************************************************
C     THIS PORTION OF THE SUBROUTINE WRITES THE TOOL DIAMETERS TO AN
C     INTERNAL FILE
************************************************************************
  300 FSCAL=SCAL1
      DO 420 I=1,L
      READ(25,113) XC(I),YC(I),ZC(I)
      XNC(I)=XC(I)/FSCAL
      YNC(I)=YC(I)/FSCAL
      ZNC(I)=ZC(I)/FSCAL
      WRITE(28,113)XNC(I),YNC(I),ZNC(I)
  420 CONTINUE
  113 FORMAT(10X,F7.4,4X,F7.4,4X,F7.4)
      FDIAM=DIAM1/SCAL1
      WRITE(19,71)'TD=',FDIAM
```

```
      IF(KFLAG .GT. 0) THEN
      FDIAM=DIAM2/SCAL1
      WRITE(19,72)'TOOL 2'
      WRITE(19,73)'SPINDLE OFF'
      WRITE(19,71)'TD=',FDIAM
      WRITE(19,74)'SPINDLE ON'
      WRITE(19,104)'DWELL 04'
      ENDIF

      IF(KFLAG .GT. 1) THEN
      FDIAM=DIAM3/SCAL1
      WRITE(19,72)'TOOL 3'
      WRITE(19,73)'SPINDLE OFF'
      WRITE(19,71)'TD=',FDIAM
      WRITE(19,74)'SPINDLE ON'
      WRITE(19,104)'DWELL 04'
      ENDIF

      IF(KFLAG .GT. 2) THEN
      FDIAM=DIAM4/SCAL1
      WRITE(19,72)'TOOL 4'
      WRITE(19,73)'SPINDLE OFF'
      WRITE(19,71)'TD=',FDIAM
      WRITE(19,74)'SPINDLE ON'
      WRITE(19,104)'DWELL 04'
      ENDIF

      IF(KFLAG .GT. 3) THEN
      FDIAM=DIAM5/SCAL1
      WRITE(19,72)'TOOL 5'
      WRITE(19,73)'SPINDLE OFF'
      WRITE(19,71)'TD=',FDIAM
      WRITE(19,74)'SPINDLE ON'
      WRITE(19,104)'DWELL 04'
      ENDIF

      IF(KFLAG .GT. 4) THEN
      FDIAM=DIAM6/SCAL1
      WRITE(19,72)'TOOL 6'
      WRITE(19,73)'SPINDLE OFF'
      WRITE(19,71)'TD=',FDIAM
      WRITE(19,74)'SPINDLE ON'
      WRITE(19,104)'DWELL 04'
      ENDIF

      IF(KFLAG .GT. 5) THEN
      FDIAM=DIAM7/SCAL1
      WRITE(19,72)'TOOL 7'
      WRITE(19,73)'SPINDLE OFF'
      WRITE(19,71)'TD=',FDIAM
      WRITE(19,74)'SPINDLE ON'
      WRITE(19,104)'DWELL 04'
      ENDIF

      IF(KFLAG .GT. 6) THEN
      FDIAM=DIAM8/SCAL1
      WRITE(19,72)'TOOL 8'
      WRITE(19,73)'SPINDLE OFF'
      WRITE(19,71)'TD=',FDIAM
      WRITE(19,74)'SPINDLE ON'
      WRITE(19,104)'DWELL 04'
      ENDIF

   71 FORMAT(A4,1X,F6.4)
   72 FORMAT(A6)
   73 FORMAT(A11)
```

```
   74 FORMAT(A10)
  104 FORMAT(A8)
*************************************************************************
C     THIS PORTION OF THE SUBROUTINE CALCULATES THE NEW X, Y, AND Z
C     COORDINATE DATA
*************************************************************************
  310 DO 320 I=1,N

      X(I)=X(I)/FSCAL
      Y(I)=Y(I)/FSCAL
      Z(I)=Z(I)/FSCAL

  320 CONTINUE
*************************************************************************
C     THIS PORTION OF THE SUBROUTINE CALCULATES THE MAXIMUM VALUES OF THE
C     NEW X, Y, AND Z COORDINATES AND INFORMS THE USER OF THE
C     POSITION OF THE SETUP POINT
*************************************************************************
      XMAX=X(1)
      YMAX=Y(1)
      ZMAX=Z(1)

      DO 330 I=2,N

      IF(X(I) .GT. XMAX) THEN
      XMAX=X(I)
      ENDIF

      IF(Y(I) .GT. YMAX) THEN
      YMAX=Y(I)
      ENDIF

      IF(Z(I) .GT. ZMAX) THEN
      ZMAX=Z(I)
      ENDIF

  330 CONTINUE
      WRITE(6,*)'********************************************'
      WRITE(6,*)'THE MAXIMUM TRAVEL IN X DIRECTION IS'
      WRITE(6,105) 'XMAX=',XMAX,'INCHES'
      WRITE(6,*)'******************************************'
      WRITE(6,*)'THE MAXIMUM TRAVEL IN THE Y DIRECTION IS'
      WRITE(6,105) 'YMAX=',YMAX,'INCHES'
      WRITE(6,*)'********************************************'
      WRITE(6,*)'THE MAXIMUM TRAVEL IN THE Z DIRECTION IS'
      WRITE(6,105) 'ZMAX=',ZMAX,'INCHES'
      WRITE(6,*)'*********************************************'
  105 FORMAT(1X,A5,1X,F8.4,1X,A6)
      XMIN=X(1)
      YMIN=Y(1)
      ZMIN=Z(1)

      DO 340 I=2,N

      IF(X(I) .LT. XMIN) THEN
      XMIN=X(I)
      ENDIF

      IF(Y(I) .LT. YMIN) THEN
      YMIN=Y(I)
      ENDIF

      IF(Z(I) .LT. ZMIN) THEN
      ZMIN=Z(I)
      ENDIF

  340 CONTINUE
```

```
      WRITE(6,*)'THE MINIMUM X, Y, AND Z COORDINATES ARE'
      WRITE(6,106) 'XMIN=',XMIN,'INCHES'
      WRITE(6,106) 'YMIN=',YMIN,'INCHES'
      WRITE(6,106) 'ZMIN=',ZMIN,'INCHES'
  106 FORMAT(1X,A5,1X,F8.4,1X,A6)
      RETURN
      END
*********************************************************************
      SUBROUTINE CUTTER(TDIAM,SCAL1,KFLAG2,SDIAM)
*********************************************************************
C     THIS SUBROUTINE TAKES THE VARIOUS DIFFERENT CUTTERS, SCALES THEM
C     DOWN AND CHECKS WHETHER THE SCALED CUTTERS ARE AVAILABLE ON DYNA
*********************************************************************
      REAL TDIAM, SCAL1, SDIAM
      INTEGER KFLAG2
      CHARACTER*1 ANS
      REAL TD1,TD2,TD3,TD4,TD5,TD6,TD7,TD8


      TD1=0.03125
      TD2=0.0625
      TD3=0.09375
      TD4=0.1875
      TD5=0.25
      TD6=0.3125
      TD7=0.375
      TD8=0.5

      WRITE(6,*)'*****************************************'

      IF(KFLAG2 .EQ. 1) THEN
      WRITE(6,*)'THE FIRST CUTTER IS SCALED DOWN TO A SIZE'
      ENDIF

      IF(KFLAG2 .EQ. 2) THEN
      WRITE(6,*)'THE SECOND CUTTER IS SCALED DOWN TO A SIZE'
      ENDIF

      IF(KFLAG2 .EQ. 3) THEN
      WRITE(6,*)'THE THIRD CUTTER IS SCALED DOWN TO A SIZE'
      ENDIF

      IF(KFLAG2 .EQ. 4) THEN
      WRITE(6,*)'THE FOURTH CUTTER IS SCALED DOWN TO A SIZE'
      ENDIF

      IF(KFLAG2 .EQ. 5) THEN
      WRITE(6,*)'THE FIFTH CUTTER IS SCALED DOWN TO A SIZE'
      ENDIF

      IF(KFLAG2 .EQ. 6) THEN
      WRITE(6,*)'THE SIXTH CUTTER IS SCALED DOWN TO A SIZE'
      ENDIF

      IF(KFLAG2 .EQ. 7) THEN
      WRITE(6,*)'THE SEVENTH CUTTER IS SCALED DOWN TO A SIZE'
      ENDIF

      IF(KFLAG2 .EQ. 8) THEN
      WRITE(6,*)'THE EIGHTH CUTTER IS SCALED DOWN TO A SIZE'
      ENDIF

      SDIAM=TDIAM/SCAL1
      WRITE(6,75) SDIAM
   75 FORMAT(1X,F6.4)
```

```
      IF(SDIAM .EQ. TD1) THEN
      WRITE(6,*)'WHICH IS AVAILABLE ON DYNA'
      GO TO 350
      ENDIF

      IF(SDIAM .EQ. TD2) THEN
      WRITE(6,*)'WHICH IS AVAILABLE ON DYNA'
      GO TO 350
      ENDIF

      IF(SDIAM .EQ. TD3) THEN
      WRITE(6,*)'WHICH IS AVAILABLE ON DYNA'
      GO TO 350
      ENDIF

      IF(SDIAM .EQ. TD4) THEN
      WRITE(6,*)'WHICH IS AVAILABLE ON DYNA'
      GO TO 350
      ENDIF

      IF(SDIAM .EQ. TD5) THEN
      WRITE(6,*)'WHICH IS AVAILABLE ON DYNA'
      GO TO 350
      ENDIF

      IF(SDIAM .EQ.TD6) THEN
      WRITE(6,*)'WHICH IS AVAILBLE ON DYNA'
      GO TO 350
      ENDIF

      IF(SDIAM .EQ. TD7) THEN
      WRITE(6,*)'WHICH IS AVAILBLE ON DYNA'
      GO TO 350
      ENDIF

      IF(SDIAM .EQ. TD8) THEN
      WRITE(6,*)'WHICH IS AVAILABLE ON DYNA'
      GO TO 350
      ENDIF

      IF(SDIAM .LT. TD1) THEN
      WRITE(6,*)'WHICH IS CLOSE TO'
      WRITE(6,76) TD1
      WRITE(6,*)'AVAILABLE ON DYNA'
      WRITE(6,*)'DO YOU WISH TO USE THIS CUTTER? (Y/N)'
      READ(5,77) ANS

      IF(ANS .EQ. 'Y') THEN
      SDIAM=TD1
      ENDIF

      IF(ANS .EQ. 'N') THEN
      WRITE(6,*)'THE DATA WILL BE GENERATED ACCORDING TO'
      WRITE(6,*)'THE ORIGINAL SCALED CUTTER, BUT SINCE THIS'
      WRITE(6,*)'CUTTER IS NOT AVAILABLE ON DYNA, YOU WILL'
      WRITE(6,*)'HAVE TO MACHINE THE PART WITH A CUTTER'
      WRITE(6,*)'CLOSE TO THIS SIZE'
      ENDIF

      GO TO 350
      ENDIF

      IF(SDIAM .GT. TD1 .AND. SDIAM .LT. TD2) THEN
      WRITE(6,*)'WHICH IS CLOSE TO'
      WRITE(6,76) TD1
   76 FORMAT(1X,F6.4)
      WRITE(6,*)'AVAILABLE ON DYNA'
```

```
WRITE(6,*)'DO YOU WISH TO USE THIS CUTTER? (Y/N)'
READ(5,77) ANS

IF(ANS .EQ. 'Y') THEN
SDIAM=TD1
ENDIF

IF(ANS .EQ. 'N') THEN
WRITE(6,*)'THE DATA WILL BE GENERATED ACCORDING TO'
WRITE(6,*)'THE ORIGINAL SCALED CUTTER, BUT SINCE THIS'
WRITE(6,*)'CUTTER IS NOT AVAILABLE ON DYNA, YOU WILL'
WRITE(6,*)'HAVE TO MACHINE THE PART WITH A CUTTER'
WRITE(6,*)'CLOSE TO THIS SIZE'
ENDIF

GO TO 350
ENDIF

IF(SDIAM .GT. TD2 .AND. SDIAM .LT. TD3) THEN
WRITE(6,*)'WHICH IS CLOSE TO'
WRITE(6,76) TD2
WRITE(6,*)'AVAILABLE ON DYNA'
WRITE(6,*)'DO YOU WISH TO USE THIS CUTTER? (Y/N)'
READ(5,77) ANS

IF(ANS .EQ. 'Y') THEN
SDIAM=TD2
ENDIF

IF(ANS .EQ. 'N') THEN
WRITE(6,*)'THE DATA WILL BE GENERATED ACCORDING TO'
WRITE(6,*)'THE ORIGINAL SCALED CUTTER, BUT SINCE THIS'
WRITE(6,*)'CUTTER IS NOT AVAILABLE ON DYNA, YOU WILL'
WRITE(6,*)'HAVE TO MACHINE THE PART WITH A CUTTER'
WRITE(6,*)'CLOSE TO THIS SIZE'
ENDIF

GO TO 350
ENDIF

IF(SDIAM .GT. TD3 .AND. SDIAM .LT. TD4) THEN
WRITE(6,*)'WHICH IS CLOSE TO'
WRITE(6,76) TD3
WRITE(6,*)'AVAILABLE ON DYNA'
WRITE(6,*)'DO YOU WISH TO USE THIS CUTTER? (Y/N)'
READ(5,77) ANS

IF(ANS .EQ. 'Y') THEN
SDIAM=TD3
ENDIF

IF(ANS .EQ. 'N') THEN
WRITE(6,*)'THE DATA WILL BE GENERATED ACCORDING TO'
WRITE(6,*)'THE ORIGINAL SCALED CUTTER, BUT SINCE THIS'
WRITE(6,*)'CUTTER IS NOT AVAILABLE ON DYNA, YOU WILL'
WRITE(6,*)'HAVE TO MACHINE THE PART WITH A CUTTER'
WRITE(6,*)'CLOSE TO THIS SIZE'
ENDIF

GO TO 350
ENDIF

IF(SDIAM .GT. TD4 .AND. SDIAM .LT. TD5) THEN
WRITE(6,*)'WHICH IS CLOSE TO'
WRITE(6,76) TD4
WRITE(6,*)'AVAILABLE ON DYNA'
WRITE(6,*)'DO YOU WISH TO USE THIS CUTTER? (Y/N)'
```

```
READ(5,77) ANS

IF(ANS .EQ. 'Y') THEN
SDIAM=TD4
ENDIF

IF(ANS .EQ. 'N') THEN
WRITE(6,*)'THE DATA WILL BE GENERATED ACCORDING TO'
WRITE(6,*)'THE ORIGINAL SCALED CUTTER, BUT SINCE THIS'
WRITE(6,*)'CUTTER IS NOT AVAILABLE ON DYNA, YOU WILL'
WRITE(6,*)'HAVE TO MACHINE THE PART WITH A CUTTER'
WRITE(6,*)'CLOSE TO THIS SIZE'
ENDIF

GO TO 350
ENDIF

IF(SDIAM .GT. TD5 .AND. SDIAM .LT. TD6) THEN
WRITE(6,*)'WHICH IS CLOSE TO'
WRITE(6,76) TD5
WRITE(6,*)'AVAILABLE ON DYNA'
WRITE(6,*)'DO YOU WISH TO USE THIS CUTTER? (Y/N)'
READ(5,77) ANS

IF(ANS .EQ. 'Y') THEN
SDIAM=TD5
ENDIF

IF(ANS .EQ. 'N') THEN
WRITE(6,*)'THE DATA WILL BE GENERATED ACCORDING TO'
WRITE(6,*)'THE ORIGINAL SCALED CUTTER, BUT SINCE THIS'
WRITE(6,*)'CUTTER IS NOT AVAILABLE ON DYNA, YOU WILL'
WRITE(6,*)'HAVE TO MACHINE THE PART WITH A CUTTER'
WRITE(6,*)'CLOSE TO THIS SIZE'
ENDIF

GO TO 350
ENDIF

IF(SDIAM .GT. TD6 .AND. SDIAM .LT. TD7) THEN
WRITE(6,*)'WHICH IS CLOSE TO'
WRITE(6,76) TD6
WRITE(6,*)'AVAILABLE ON DYNA'
WRITE(6,*)'DO YOU WISH TO USE THIS CUTTER? (Y/N)'
READ(5,77) ANS

IF(ANS .EQ. 'Y') THEN
SDIAM=TD6
ENDIF

IF(ANS .EQ. 'N') THEN
WRITE(6,*)'THE DATA WILL BE GENERATED ACCORDING TO'
WRITE(6,*)'THE ORIGINAL SCALED CUTTER, BUT SINCE THIS'
WRITE(6,*)'CUTTER IS NOT AVAILABLE ON DYNA, YOU WILL'
WRITE(6,*)'HAVE TO MACHINE THE PART WITH A CUTTER'
WRITE(6,*)'CLOSE TO THIS SIZE'
ENDIF

GO TO 350
ENDIF

IF(SDIAM .GT. TD7 .AND. SDIAM .LT. TD8) THEN
WRITE(6,*)'WHICH IS CLOSE TO'
WRITE(6,76) TD7
WRITE(6,*)'AVAILABLE ON DYNA'
WRITE(6,*)'DO YOU WISH TO USE THIS CUTTER? (Y/N)'
READ(5,77) ANS
```

```
      IF(ANS .EQ. 'Y') THEN
      SDIAM=TD7
      ENDIF

      IF(ANS .EQ. 'N') THEN
      WRITE(6,*)'THE DATA WILL BE GENERATED ACCORDING TO'
      WRITE(6,*)'THE ORIGINAL SCALED CUTTER, BUT SINCE THIS'
      WRITE(6,*)'CUTTER IS NOT AVAILABLE ON DYNA, YOU WILL'
      WRITE(6,*)'HAVE TO MACHINE THE PART WITH A CUTTER'
      WRITE(6,*)'CLOSE TO THIS SIZE'
      ENDIF

      GO TO 350
      ENDIF

      IF(SDIAM .GT. TD8) THEN
      WRITE(6,*)'WHICH IS CLOSE TO'
      WRITE(6,76) TD8
      WRITE(6,*)'AVAILABLE ON DYNA'
      WRITE(6,*)'DO YOU WISH TO USE THIS CUTTER? (Y/N)'
      READ(5,77) ANS

      IF(ANS .EQ. 'Y') THEN
      SDIAM=TD8
      ENDIF

      IF(ANS .EQ. 'N') THEN
      WRITE(6,*)'THE DATA WILL BE GENERATED ACCORDING TO'
      WRITE(6,*)'THE ORIGINAL SCALED CUTTER, BUT SINCE THIS'
      WRITE(6,*)'CUTTER IS NOT AVAILABLE ON DYNA, YOU WILL'
      WRITE(6,*)'HAVE TO MACHINE THE PART WITH A CUTTER'
      WRITE(6,*)'CLOSE TO THIS SIZE'
      ENDIF

      GO TO 350
      ENDIF

   77 FORMAT(A1)
  350 WRITE(6,*)'********************************************'
      RETURN
      END
************************************************************************
      SUBROUTINE FINIS(IMP)
************************************************************************
C     THIS SUBROUTINE TAKES THE FINI STATEMENT FROM THE INPUT FILE AND
C     CONVERTS IT TO THE END STATEMENT FOR THE DYNA MILLING MACHINE
************************************************************************
      CHARACTER*80 IMP
      INTEGER OUT
      OUT=12
      WRITE(20,78)'END'
   78 FORMAT(A3)
      RETURN
      END
```

The vita has been removed from the scanned document