Development Of Postprocessor For Milling Machine HAAS VOP - C

Thesis submitted in accordance with the requirements of the National Technical University College of Malaysia for the Degree of Bachelor of Engineering (Honours) Manufacturing (Process)

By

ZULKIFLEE BIN MAD ATARI

Faculty of Manufacturing Engineering
November 2005
KOLEJ UNIVERSITI TEKNIKAL KEBANGSAAN MALAYSIA

BORANG PENGESAHAN STATUS TESIS*

JUDUL: Development of postprocessor for milling machine HAAS VOP-C

SESU PENGAJIAN : 2/2005

Saya ZULKIFLEE BIN MAD ATARI

(HURUF BESAR)

mengaku membenarkan tesis (PSM/Sarjana/Doktor Falsafah) ini disimpan di Perpustakaan Kolej Universiti Teknikal Kebangsaan Malaysia (KUTKM) dengan syarat-syarat kegunaan seperti berikut:

1. Tesis adalah hak milik Kolej Universiti Teknikal Kebangsaan Malaysia.
2. Perpustakaan Kolej Universiti Teknikal Kebangsaan Malaysia dibenarkan membuat salinan untuk tujuan pengajian sahaja.
3. Perpustakaan dibenarkan membuat salinan tesis ini sebagai bahan pertukaran antara institusi pengajian tinggi.
4. **Sila tandakan (✓)

☐ SULIT

☐ TERHAĐ

☐ TIDAK TERHAĐ

(Mengandungi maklumat yang berdjarah keselamatan atau kepentingan Malaysia yang termaktub di dalam AKTA RAHSIA RASMI 1972)

☐ TERHAĐ

☐ TIDAK TERHAĐ

(Mengandungi maklumat TERHAĐ yang telah ditentukan oleh organisasi/badan di mana penyelidikan dijalankan)

Disahkan oleh:

(TANDATANGAN PENULIS)

Alamat Tetap:
NO.84 KG.TELUK BULOH,
36400 HUTAN MELINTANG,
PERAK DARUL RIDZUAN.

Tarikh: 25 November 2005

(TANDATANGAN PENYELIA)

Cop Rasmı:

ROSIDAH BINTI JAAFAR
Penyarah
Fakulti Kejuruteraan Pembuatan
Kolej Universiti Teknikal Kebangsaan Malaysia
Karung Berkunci 1200
75450 Ayer Keroh, Melaka.

Tarikh: 9/12/2005

* Tesis dimaksudkan sebagai tesis bagi Ijazah Doktor Falsafah dan Sarjana secara penyelidikan, atau disertasi bagi pengajian secara kerja kursus dan penyelidikan, atau Laporan Projek Sarjana Muda (PSM).
** Jika tesis ini SULIT atau TERHAĐ, sila lampirkan surat daripada pihak berkuasa/organisasi berkenaan dengan menyatakan sekali sebab dan tempoh tesis ini perlu dikelaskan sebagai SULIT atau TERHAĐ.
Pustakawan
Perpustakawan Kolej Universiti Teknikal Kebangsaan Malaysia
KUTKM, Ayer Keroh
MELAKA.

Saudara,

PENGKELASAN TESIS SEBAGAI SULIT/TERHAD
- TESIS SARJANA MUDA KEJURUTERAAN PEMBUATAN (PROSES PEMBUATAN):
ZULKIFLEE BIN MAD ATARI
TAJUK: DEVELOPMENT OF POSTPROCESSOR FOR MILLING MACHINE HAAS VOP-C

Sukacita dimaklumkan bahawa tesis yang tersebut di atas bertajuk “Development Of Postprocessor For Milling Machine HAAS VOP-C” mohon dikelaskan sebagai terhad untuk tempoh lima (5) tahun dari tarikh surat ini memandangkan ia mempunyai nilai dan potensi untuk dikomersialkan di masa hadapan.

Sekian dimaklumkan. Terima kasih.

“BERKHIDMAT UNTUK NEGARA KERANA ALLAH”

Yang benar,

WAN HASRULNIZZAM WAN MAHMOOD
Pensyarah,
Fakulti Kejuruteraan Pembuatan
(Penyelia Bersama)
06-2332122

s.k. Penyelia Utama:
Pn. Rosidah Jaafar
DECLARATION

I hereby, declare this thesis entitled “Development Of Postprocessor For Milling Machine-HAAS VOP-C” is the results of my own research except as cited in the reference.

Signature : ......................................................
Author’s Name : ZULKIFLEE BIN MAD ATARI.
Date : 25 NOVEMBER 2005
APPROVAL

This thesis submitted to the senate of KUTKM and has been accepted as fulfillment of the requirement for the degree of Bachelor of Engineering (Honours) Manufacturing (Process). The members of the supervisory committee are as follows:

ROSIDAH BINTI JAAFAR
Main supervisor
Faculty of Manufacturing Engineering
ABSTRAK

Di dalam laporan Projek Tahun Akhir (PTA) ini, ianya menjurus kepada perancangan kerja yang dijalankan serta cara perlaksanaannya bagi mencapai objektif yang dikehendaki. Oleh yang demikian bagi melaksanakan matlamat tersebut perancangan yang teratur dan sistematik perlu dilakukan bagi memastikan objektif yang ingin dicapai terlaksana mengikuti spesifikasi yang telah ditentukan. Oleh itu, kajian yang terperinci perlu dilakukan bagi merealisasikan matlamat tersebut Ini kerana ianya akan mempengaruhi hasil kerja yang bakal dijalankan nanti.

Oleh yang demikian, di dalam laporan ini ianya menerangkan tentang teori – teori yang berkaitan dengan ‘post processor’ yang terdapat di pasaran pada masa kini. Ini penting bagi mendapatkan segala maklumat yang dikehendaki yang berkaitan dengan ‘postprocessor’.

Manakala dari segi kaedah perlaksanaannya, beberapa jenis peralatan akan digunakan bagi melaksanakan projek yang ingin dihasilkan. Antara peralatan yang akan digunakan adalah perisian C++, CATIA (CAD/CAM) dan mesin CNC (Haas) bagi melaksanakan projek tersebut. Disamping itu, segala maklumat yang berkaitan dengan kajian telah ditunjukkan di dalam keputusan di dalam laporan ini.
ABSTRACT

For the final year project, the explanation and detail about the process planning for final year project and process will be determine so that the project objective or scope will be achieved. So the systematic planning is important to make sure that all the planning processes are according to the plan. Actually this can be achieved by performing some research or experiment and these is important in order to make the comparison for the final product.

From the research also it will give a lot of information about the postprocessor program and to upgrade the postprocessor at final year project. In additional, they are various types of equipments need to be use such as CAD/CAM software, Microsoft Studio Visual Basic C++ software and lastly is CNC (Haas) milling machine to test the program to get the result.
DEDICATION

For My Family:
Mr. Mad Atari Bin Muhamad Sanif
Mrs. Arbaayah Binti Esa
Nor'azean Binti Mad Atari
ACKNOWLEDGEMENT

Syukur alhamdulillah, with much guidance and support my final year thesis project is now completed. Firstly I would really like to express my gratefulness towards Allah S.W.T because giving me the opportunity to complete my final graduation project. Without faith towards the almighty, I would not have the confidence and strength and strength to finish this project.

Not to mention, I am in debt towards my beloved family especially my parents, Mr. Mad Atari bin Muhammad Sanif and Mrs. Arbaayah bt. Esa for showing love, support and advice when in truly need.

I would like to also express my greatest appreciations to my wise supervisor, Mrs. Rosidah Jaafar, truly without his guidance and wisdom, I would have been lost. Not to forget to thank to Dr. Ir. Prianggada I Tanaya as second supervisor and help me on how to develop the programs.

To my friends Nidzwan Bin Nosbi, Mohd Jamil Masyadar and Irwan Alfarid Zainol which helps me to find the information regarding my final year project. Gratitude is also dedicate to my friends, lectures and staffs who showed over flowing support and high sprit in helping finishing my thesis project. Thank you four your all.
# TABLE OF CONTENTS

Abstract .................................................. i-ii
Dedication ............................................... iii
Acknowledgement ........................................ iv
Table of Contents ...................................... v-vii
List of Figures .......................................... viii-x
List of Tables .......................................... xi
List of Symbols .......................................... xii

## 1. INTRODUCTION

1.1 Background ........................................... 1
1.2 Problem Statements ................................. 1
1.3 Objectives of the Research ........................ 2

## 2. LITERATURES REVIEW

2.1 Transmission program postprocessor and generalized 3
   2.1.1 Postprocessor .................................. 3-4
2.2 Advance In Nc Control Languages .................. 4
2.3 Advance In Computer Languages .................... 5
2.4 APT Languages statements .......................... 6-20

## 3. METHODOLOGY

3.1 Methodology .......................................... 21
3.2 Step To Develop The Postprocessor At PTA 1 .... 22
3.3 Process Plan PTA 1 ................................... 23
3.4 Step To Develop The Postprocessor At PTA 2 .... 24
3.5 Process Plan PTA 2 ................................... 25
3.6 Alternative Routed For Part Programming ........ 26-27
3.7 CL. Data ............................................. 28
3.8 Flow Chart For CAD Processing .................... 29

© Universiti Teknikal Malaysia Melaka
3.9 Developing and programming NC Codes
  3.12.1 Step 1: Sketch the Part
  3.12.2 Step 2: Determining The stock Size And Selecting The tool
  3.12.3 Step 3: Determine The Feed Rate And Depth Of Cut
  3.12.4 Step 4: Determine How To Cut The Part

4. RESULT
  4.1 Drawing the product shape using the CATIA software
    4.1.1 CAT.PART (Geometry)
    4.1.2 CAT.PROCESS (Tool path)
  4.2 Generated the program
    4.2.1 APT Languages
    4.2.2 NC.Code
  4.3 Flowchart programming
  4.4 All the table had shown the character value for each character.
  4.5 Generate Programming Using the Microsoft Studio Visual Basic
     (C++/C)
    4.5.1 Final result (input)
    4.5.2 Head File and Variable File
    4.5.3 Variable Address Function File
    4.5.4 Main Function
    4.5.5 Function Definition
  4.6 Full postprocessor programming (input)
  4.7 Result discussion
    4.7.1 Header and variable file
    4.7.2 Variable address function file.
    4.7.3 Main function
    4.7.4 Function definition
  4.8 Final result (output)
  4.9 Problem solving
    4.9.1 Programming Header and Tail File at the Machine Controller
    4.9.2 Programming Header and Tail File from the CATIA software
4.10 Programming Recommendation
  4.10.1(a) INPUT (if true input) 91
  4.10.1(b) OUTPUT 91
  4.10.1(c) INPUT (if wrong input) 92
  4.10.1(d) OUTPUT 92-93

5. DISCUSSION
  5.1 Discussion 94-95

6. CONCLUSION
  6.1 Conclusion 96

7. REFERENCE 97

8. APPENDICES
   Table A: The table shown ASCII Character Set 98
LIST OF FIGURES

2.1 Defining circle 6
2.2 Defining a circle using two intersecting lines 7
2.3 Cutter definition 8
2.4 Definition of INTOL 10
2.5 Defining a line using two points 11
2.6 Defining a line using a point and circle 11
2.7 Defining a line using a point and the x-axis or another line 11
2.8 Defining a line using a point and parallelism or perpendicularity to another line 12
2.9 Defining a line using its tangency to two circle 12
2.10 Tool movement direction 14
2.11 Definition of OUTTOL 15
2.12 Defining a plane using three points 16
2.13 Defining a plane using a point and parallelism to another plane 16
2.14 Defining a plane using two points and perpendicularity to another plane 17
2.15 Defining a point using its x,y and z coordinates 17
2.16 Defining a point using intersection of lines and circles 18
3.1 Flow Chart Step to analyze the PTA 1 22
3.2 Flow Chart Step to analyze the PTA 2 24
3.3 Flow Chart Routed For Part Programming 26
3.4 Flow Chart CAD processing process 29
4.1 Sketching drawing 32
4.2 Product Shape 33
4.3 Tool Movement 33
4.4 Solid Material 34
4.5 Final Product 34
4.6 Flow Chart APT languages L1 41
4.7 Flow Chart APT languages L2 42

© Universiti Teknikal Malaysia Melaka
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.8</td>
<td>Flow Chart APT languages L3</td>
<td>43</td>
</tr>
<tr>
<td>4.9</td>
<td>Flow Chart APT languages</td>
<td>44</td>
</tr>
<tr>
<td>4.10</td>
<td>Flow Chart APT languages</td>
<td>45</td>
</tr>
<tr>
<td>4.11</td>
<td>Flow Chart APT languages L6</td>
<td>46</td>
</tr>
<tr>
<td>4.12</td>
<td>Flow Chart APT languages L7</td>
<td>47</td>
</tr>
<tr>
<td>4.13</td>
<td>Flow Chart APT languages L8</td>
<td>48</td>
</tr>
<tr>
<td>4.14</td>
<td>Flow Chart APT languages L9</td>
<td>49</td>
</tr>
<tr>
<td>4.15</td>
<td>Flow Chart APT languages L10</td>
<td>50</td>
</tr>
<tr>
<td>4.16</td>
<td>Flow Chart APT languages</td>
<td>51</td>
</tr>
<tr>
<td>4.17</td>
<td>Flow Chart APT languages L12</td>
<td>52</td>
</tr>
<tr>
<td>4.18</td>
<td>Flow Chart APT languages L84</td>
<td>53</td>
</tr>
<tr>
<td>4.19</td>
<td>Flow Chart APT languages L85</td>
<td>54</td>
</tr>
<tr>
<td>4.20</td>
<td>Flow Chart APT languages L86</td>
<td>55</td>
</tr>
<tr>
<td>4.21</td>
<td>Header and variable file of programming</td>
<td>60</td>
</tr>
<tr>
<td>4.22</td>
<td>variable address function file of programming</td>
<td>60</td>
</tr>
<tr>
<td>4.23</td>
<td>Main function of programming</td>
<td>61</td>
</tr>
<tr>
<td>4.24</td>
<td>Main function of programming (con’t1)</td>
<td>61</td>
</tr>
<tr>
<td>4.25</td>
<td>Main function of programming (con’t2)</td>
<td>62</td>
</tr>
<tr>
<td>4.26</td>
<td>Main function of programming (con’t3)</td>
<td>62</td>
</tr>
<tr>
<td>4.27</td>
<td>Main function of programming (con’t4)</td>
<td>63</td>
</tr>
<tr>
<td>4.28</td>
<td>Main function of programming (con’t5)</td>
<td>63</td>
</tr>
<tr>
<td>4.29</td>
<td>Main function of programming (con’t6)</td>
<td>64</td>
</tr>
<tr>
<td>4.30</td>
<td>Main function of programming (con’t7)</td>
<td>64</td>
</tr>
<tr>
<td>4.31</td>
<td>Main function of programming (con’t8)</td>
<td>65</td>
</tr>
<tr>
<td>4.32</td>
<td>Main function of programming (con’t9)</td>
<td>65</td>
</tr>
<tr>
<td>4.33</td>
<td>Main function of programming (con’t10)</td>
<td>66</td>
</tr>
<tr>
<td>4.34</td>
<td>Main function of programming (con’t11)</td>
<td>66</td>
</tr>
<tr>
<td>4.35</td>
<td>Main function of programming (con’t12)</td>
<td>66</td>
</tr>
<tr>
<td>4.36</td>
<td>Definition 1 and 2 of programming</td>
<td>67</td>
</tr>
<tr>
<td>4.37</td>
<td>Definition 3 and 4 of programming</td>
<td>67</td>
</tr>
<tr>
<td>4.38</td>
<td>Definition 5 and 6 of programming</td>
<td>67</td>
</tr>
<tr>
<td>4.39</td>
<td>Definition 7 and 8 of programming</td>
<td>68</td>
</tr>
<tr>
<td>4.40</td>
<td>Definition 9 and 10 of programming</td>
<td>68</td>
</tr>
</tbody>
</table>
4.41  Definition 11 and 12 of programming  68
4.42  Definition 13 and 14 of programming  69
4.43  Definition 15 and 16 of programming  69
4.44  Definition 17 and 18 of programming  69
4.45  Definition 19 and 20 of programming  70
4.46  Output of programming  87
4.47  Header and Tail of programming  88
4.48  True input calculation sub function of programming  90
4.49  Output calculation sub function of programming  90
4.50  Wrong input calculation sub function of programming  91
4.51  Output calculation sub function of programming if wrong input  91
# LIST OF TABLES

3.1 DEVELOPMENT OF POSTPROCESSOR FOR MILLING MACHINE HAAS VOP-C PLANNING TABLE FOR PTA 2

<table>
<thead>
<tr>
<th>4.1</th>
<th>COOLNT value / codes number</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.2</td>
<td>CUTCOM value / codes number</td>
</tr>
<tr>
<td>4.3</td>
<td>TLAXIS value / codes number</td>
</tr>
<tr>
<td>4.4</td>
<td>RAPID value / codes number</td>
</tr>
<tr>
<td>4.5</td>
<td>GOTO value / codes number</td>
</tr>
<tr>
<td>4.6</td>
<td>CUTTER value / codes number</td>
</tr>
<tr>
<td>4.7</td>
<td>TOOLNO value / codes number</td>
</tr>
<tr>
<td>4.8</td>
<td>TPRINT value / codes number</td>
</tr>
<tr>
<td>4.9</td>
<td>LOADTL value / codes number</td>
</tr>
<tr>
<td>4.10</td>
<td>FEDRAT value / codes number</td>
</tr>
<tr>
<td>4.11</td>
<td>SPINDL value / codes number</td>
</tr>
<tr>
<td>4.12</td>
<td>REWIND value / codes number</td>
</tr>
<tr>
<td>4.13</td>
<td>END value / codes number</td>
</tr>
<tr>
<td>Symbol</td>
<td>Meaning</td>
</tr>
<tr>
<td>---------------</td>
<td>-----------------</td>
</tr>
<tr>
<td>1) CUTCOM</td>
<td>G40</td>
</tr>
<tr>
<td>2) RAPID</td>
<td>G00</td>
</tr>
<tr>
<td>3) FEDRAT</td>
<td>F</td>
</tr>
<tr>
<td>4) INCHES</td>
<td>G20</td>
</tr>
<tr>
<td>5) METRIC</td>
<td>G21</td>
</tr>
<tr>
<td>6) SPINDL</td>
<td>s</td>
</tr>
<tr>
<td>7) COOLNT/ON</td>
<td>M08</td>
</tr>
<tr>
<td>8) COOLNT/OFF</td>
<td>M09</td>
</tr>
<tr>
<td>9) PROGRAM END</td>
<td>M02</td>
</tr>
<tr>
<td>10) REWIND</td>
<td>M30</td>
</tr>
<tr>
<td>11) SPINDLE STOP</td>
<td>M05</td>
</tr>
<tr>
<td>12) TOOL CHANGE</td>
<td>M16</td>
</tr>
<tr>
<td>13) XY PLANE</td>
<td>G17</td>
</tr>
<tr>
<td>14) G43/G44/G143 cancel</td>
<td>G49</td>
</tr>
<tr>
<td>15) CANNED Cycle Cancel</td>
<td>G80</td>
</tr>
<tr>
<td>16) ABSOLUTE</td>
<td>G90</td>
</tr>
<tr>
<td>17) LINEAR</td>
<td>G01</td>
</tr>
<tr>
<td>18) SPINDLE FORWARD</td>
<td>M03</td>
</tr>
<tr>
<td>19) TOOL LENGTH</td>
<td>G43</td>
</tr>
<tr>
<td>20) FEED PERMINUTE</td>
<td>G94</td>
</tr>
</tbody>
</table>
CHAPTER 1
Introduction

1.1 Background

The Postprocessor is the part of the CAM software that translates the tool path data into the correct file format when saving (in fact an export filter). This functionality is the same as used in the (Windows) device driver that comes with any printer, to translate the word processor’s output to the format required by that printer. In many current CAM systems the postprocessor can be configured by the user, making it easy to connect to any new machine.

So for example a complete APT part program must include function not accomplished by geometry statements and motion commands. These additional functions are implemented by postprocessor statements and auxiliary statements. Postprocessor statements control the operation of the machine tool and play a supporting role in generating the tool path. Such statements are used to define cutter size, specify speeds and feeds, turn coolant flow on and off and control other features of the particular machine tool on which the machining job will performed.

1.2 Problem Statements

i. To upgrade the previous new post processor program for CNC (Haas) milling machine.

ii. Used the Microsoft studio visual basic C++ software program to create the postprocessor program as translations mediums.
1.3 Objective

i. Create drawing from CAD/CAM software.

ii. Generate the APT languages program.

iii. Generate the NC_codes.

iv. Find the problem at CATIA program when transfer it at CNC milling machine.

v. Develop the postprocessor program CNC milling machine (Haas VPO-C).
CHAPTER 2

Literatures Review

2.1 TRANSMISSION PROGRAM, POSTPROCESSORS AND GENERALIZED

2.1.1 Postprocessor

NC programs which have been generated software can be read-in to the control system of a CNC machine tool and subsequently be executed. Conversely, it is also possible provided certain requirements are met to transmit NC programs from the machine control system to the Simulators, e.g. for modification or test run purposes. The transmission program is designed to transmit NC programs without any changes of format or syntax. Certain control characters, such as for encoding program start and program end or line feed can be defined by the user. To facilitate the interface configuration and to ensure a perfect interconnection, the transmission program allows the link between the Simulator and the target machine control to be tested before it is actually established.

NC programs written programming code must be translated into the code of the machine control before they can be read by the control system. For these purpose a variety of postprocessors are available, each designed for a specific control system and for adjustment to the respective interfaces. The postprocessors change the coded programs into basic NC blocks which correspond to the DIN standard and whose syntax is compatible with the dialect of DIN used in the target control. At present more than seventy postprocessors are available. The most economic solution for the translation of NC programs is the Generalized Postprocessor, which enables the user to determine cross-references of commands in the translation. Provided that these features are supported by the target control system, the performance characteristics of the Generalized Postprocessor are the following:

• Addresses are changed if necessary,
• Formats of values of the addresses are adapted to the control,
• Subroutines are kept during the translation,
• Parameters are retained or added,
• Cycles are transformed into the respective cycles of the target control,
• Segment contour programming is kept while being translated into G-functions.

The capabilities of the target control system software are used to the full in the translation. With its user-defined command references between the source and the object code, the Generalized Postprocessor is a powerful and universal tool which can adapt the software to a wide variety of CNC control systems.

2.2 ADVANCES IN NC CONTROL LANGUAGES

The standard language read by most NC machines is defined G and M codes. Although the simple elements is (G1=linear move,G2=circular interpolation) there are wide variations in controllers, particularly with regard to the inclusion of “macros” and subroutines. A hole drilling macro allows the NC program to substitute one line for hundreds by letting the NC controller do more of the calculation of individual tool positions. Other macros exists for creating rectangular and circular pockets, bolt hole patterns, hole tapping operations, etc.

Controllers already exist which input a complete NURBS surface definition. The data is more compact and far richer in information content than the traditional G-codes. The EIA association is also creating a standard for a Basic Control which is an attempt to standardize CLDATA. Once standardized, machine controllers will read BCL directly with no need for post processing into G-codes. There has been a recent trend in the direction of “open architecture” controllers where the user has access to the functionality of the controller and the ability to customize it for their own particular needs. As NC machines become more “sensor-rich” the open-architecture controller offers an opportunity to use machines in more efficient and creative ways. Force control, acoustic based wear monitoring, in-process measurement and adjustment are just a few of the Possibilities.
2.3 ADVANCES IN COMPUTER LANGUAGES

The fundamental observation that a computer program is both data structures and algorithms, and that the data structure design is at least as important as the algorithm design has led to the development of object oriented languages like C++. Hierarchical data structures, inheritance, classes, derived classes, polymorphism and virtual functions are some of the powerful concepts now routinely used in modern software Engineering. NC programming systems have been primarily algorithm oriented with less emphasis placed on data structures. As a result, at each step in the process shown in Figure 1 the information associated with the previous step would generally “evaporate.” That is, the process plan was derived from the part geometry, but specific information about the part geometry was not retained for subsequent steps. At the last step where G codes are processed by the NC machine to make the part, the information content relevant to the original design intent, is negligible. Ideally, an improved system would add derived Information at each step, without any loss of information from the previous step. Conclusion: NC programming systems should have well designed data structures based on modern object oriented concepts.

[Robert B. Jerard, Copyright © 1998 by ASME]
2.4 APT LANGUAGES STATEMENTS.

1. ATANGL - At angle (descriptive data).
   The data that follows this APT word is an angle specified in degrees. See LINE

2. CENTER - Center (descriptive data).
   The data that follows this APT word specifies the location of the center of a circle
   or circular arc. See CIRCLE

3. CIRCLE - Circle (geometry type).
   Used to define a circle in the x-y plane. Methods of definitions include:
   - Using the coordinates of its center and its radius (See figure 6)
     \[ C1 = \text{CIRCLE/CENTER, 100, 50, 0, RADIUS, 32} \]
   - Using the point identifying its center and its radius (See figure 6)
     \[ C1 = \text{CIRCLE/CENTER, P1, RADIUS, 32} \]
   - Using the point identifying its center and a line to which it is tangent ((See
     figure 6)
     \[ C1 = \text{CIRCLE/CENTER, P1, TANTO, L1} \]

![Figure 2.1: Defining circle.[4]](image)

© Universiti Teknikal Malaysia Melaka
Figure 2.2: Defining a circle using two intersecting lines.[4]

- Using the three point on its circumference ((See figure 6)
  \[ C1 = \text{CIRCLE/P2, P3, P4} \]
- Using two intersecting lines and the radius of the circle ((See figure 7)
  \[ C2 = \text{CIRCLE/XSMALL, L2, YSMALL, L3, RADIUS, 25} \]
  \[ C3 = \text{CIRCLE/YLARGE, L2, YLARGE, L3, RADIUS, 25} \]
  \[ C4 = \text{CIRCLE/XLARGE, L2, YLARGE, L3, RADIUS, 25} \]
  \[ C5 = \text{CIRCLE/YSMALL, L2, YSMALL, L3, RADIUS, 25} \]

4. CLPRINT - Cutter location print
   Used to obtain a computer printout of the cutter location sequence.

5. COOLNT - Coolant (postprocessor statement).
   Actuates various coolant options that may be available on the machine tool

6. CUTTER - Cutter (postprocessor statements)
   Defines cutter diameter and other cutting tool dimensions required offset calculation.

7. DELAY - Delay (postprocessor command).
   Used to delay the machine tool operation by a certain period of time, specified in seconds.

8. END – End (postprocessor statement)
   Stop the programs at the end of a section, turning off spindle rotation and coolant, if applicable (corresponds to a M02 or M03)

9. FEDRAT
   Used to specify feed rate. Method of specification includes:
   - Feed rate given in units per minute (inches or mm, depending on units specification)
     FEDRAT/120, IPM (corresponds to G94 F120 or G98 F120)