

ADVANCED COMPUTER AIDED
MANUFACTURING APPLICATIONS FOR
TURNING PROCESSES

TAN BENG HWANG

Report submitted in partial fulfillment of the requirements
for the award of the degree of
Bachelor of Engineering in Manufacturing Engineering

Faculty of Manufacturing Engineering

UNIVERSITI MALAYSIA PAHANG

June 2016

SUPERVISOR'S DECLARATION

I hereby declare that I have checked this thesis and in my opinion, this thesis is adequate in terms of scope and quality for the award of the degree of Bachelor of Engineering in manufacturing.

Signature :
Name of supervisor : DR. MUHAMMED NAFIS BIN OSMAN ZAHID
Position : SENIOR LECTURER
Date : 2 JUN 2016

STUDENT'S DECLARATION

I hereby declare that the work in this thesis is my own except for quotation and summaries which have been duly acknowledged. The thesis has not been accepted for any degree and is not concurrently submitted for award of other degree.

Signature :
Name : TAN BENG HWANG
ID Number : FA12054
Date : 2 JUN 2016

ACKNOWLEDGEMENTS

I am grateful and would like to express my sincere gratitude to my supervisor Dr. Nafis for his germinal ideas, invaluable guidance, continuous encouragement and constant support in making this research possible. He has always impressed me with his outstanding professional conduct, his strong conviction for science, and his belief that a Degree program is only a start of a life-long learning experience. I appreciate his consistent support from the first day I applied to graduate program to these concluding moments. I am truly grateful for his progressive vision about my training in science, his tolerance of my naïve mistakes, and his commitment to my future career. I also would like to express very special thanks to my supervisor for his suggestions and co-operation throughout the study. I also sincerely thanks for the time spent proofreading and correcting my many mistakes.

My sincere thanks go to all my friends, who helped me in many ways and made my stay at UMP pleasant and unforgettable. Many special thanks go to them for their excellent co-operation, inspirations and supports during this study.

I acknowledge my sincere indebtedness and gratitude to my parents for their love, dream and sacrifice throughout my life. I acknowledge the sincerity of my parents-in-law, who consistently encouraged me to carry on my higher studies in Malaysia. I am also grateful to my wife, for her sacrifice, patience, and understanding that were inevitable to make this work possible. I cannot find the appropriate words that could properly describe my appreciation for her devotion, support and faith in my ability to attain my goals, which was crucial for the successful completion of this study.

ABSTRACT

The application of computer technology have influenced numerous development in turning manufacturing process. Generally, a detailed variables have to be defined including cutting parameters, process parameters and machining operations. Nevertheless, process planning needed skilled process planners, longer process planning time spent and has low level of dependability. This paper aims to develop an user interface program controller where in order to support the decision making with large volume of data and to optimise the process planning task, the integration between intelligent systems for example Computer Aided Design (CAD), Computer Aided Process Planning (CAPP) and Computer Aided Manufacturing (CAM) have strengthened the relation between design and manufacturing stages. Through the integration of these intelligent systems, application of graphical user interface (GUI) for automated process planning in turning machining is developed to provide an effective machining process. The main purpose of the developed user interface program is to assist the process planning tasks in turning machining for construct machining program with minimum time and minimalize the need of skilled user in the process. Moreover, the graphical user interface (GUI) is able to improve the process planning in turning operations by integrating decision making tools and optimize process parameters such as spindle speed, depth of cut and cutting feed rate based on operation features with minimum human intervention and embed substantial levels of automation in the planning stage.

ABSTRAK

Penggunaan teknologi komputer telah mempengaruhi banyak pembangunan dalam proses pembuatan turning. Secara amnya, pembolehubah terperinci perlu ditakrifkan termasuk parameter pemotongan, parameter proses dan operasi pemesinan. Walau bagaimanapun, perancangan proses memerlukan perancang proses yang mahir, masa perancangan proses yang lebih lama dan mempunyai tahap kepercayaan yang rendah. Kerja kertas ini bertujuan untuk menciptakan satu program pengawal antaramuka dalam usaha untuk menyokong pemilihan proses pembuatan dengan jumlah data yang besar dan untuk mengoptimumkan tugas perancangan proses, integrasi antara sistem pintar contohnya Rekabentuk Bantuan Komputer (CAD), Perancangan Process Bantuan Komputer (CAPP) dan Pembuatan Bantuan Komputer (CAM) telah mengukuhkan hubungan antara reka bentuk dan pembuatan. Melalui integrasi sistem pintar ini, aplikasi antaramuka pengguna grafik (GUI) untuk perancangan proses automatik dalam pemesinan turning diciptakan untuk menyediakan proses pemesinan yang lebih berkesan. Tujuan utama program antaramuka pengguna yang diciptakan adalah untuk membantu tugas perancangan proses dalam pemesinan turning untuk membina program pemesinan dengan masa yang singkat dan menggurangkan keperluan pengguna yang mahir dalam proses perancangan. Selain itu, antaramuka pengguna grafik (GUI) mampu meningkatkan perancangan proses dalam operasi turning dengan mengintegrasikan alat pemilihan keputusan dan mengoptimumkan parameter proses seperti kelajuan spindle, kedalaman pemotongan dan kadar pemotongan berdasarkan proses operasi dengan campur tangan manusia yang minima dan membenamkan tahap automasi dalam peringkat perancangan proses.

TABLE OF CONTENTS

	Page
SUPERVISOR'S DECLARATION	iii
STUDENT'S DECLARATION	iv
ACKNOWLEDGEMENTS	v
ABSTRACT	vi
ABSTRAK	vii
TABLE OF CONTENTS	viii
LIST OF TABLES	ix
LIST OF FIGURES	x
LIST OF SYMBOLS	xi
LIST OF ABBREVIATIONS	xii
CHAPTER 1 INTRODUCTION	
1.1 Research Overview	1
1.2 Problem Statement	3
1.3 Objectives of the Research	4
1.4 Scope	4
CHAPTER 2 LITERATURE REVIEW	
2.1 Introduction	6
2.2 Process Planning	6

2.3	Computer Aided Process Planning (CAPP)	10
2.4	CAD / CAM	15
2.5	Optimization of Turning Machining Techniques	17
2.6	Programming Language	18
2.7	Turning Operations	19
	2.7.1 Tool Holder Styles	22
	2.7.2 Shapes of Turning Insert	23
	2.7.3 Operating Conditions	27
	2.7.4 Cutting Conditions	28
2.8	Summary	29

CHAPTER 3 RESEARCH DESIGN AND METHODOLOGY

3.1	Introduction	30
3.2	Flow Chart	31
3.3	NX Software	32
	3.3.1 Journaling	35
	3.3.2 Beginning of Journaling	36
	3.3.3 Model of Workpiece	37
3.4	CAM File Generation	38
	3.4.1 Facing Operation	46
	3.4.2 Roughing Operation	47
	3.4.3 Finishing Operation	49
	3.4.4 Grooving Operation	50
3.5	Visual Basic Programming	51
	3.5.1 Programming Environment	52
3.6	Fabrication of Model Workpiece	56
3.7	Summary	60

CHAPTER 4 RESULTS ANALYSIS AND DISCUSSIONS

4.1	Introduction	61
4.2	Process Planning in CNC Machining	62
4.3	Customisation of Recorded Codes	65
4.3.1	Create Appropriate Blank Size	65
4.3.2	Inheritance of Input Value	69
4.3.3	Cutting Region Selection	70
4.3.4	Materials Selection	73
4.4	Process Planning Simulation Time Duration	77
4.5	Program Verification	78
4.6	Summary	79

CHAPTER 5 CONCLUSIONS AND RECOMMENDATIONS

5.1	Introduction	81
5.2	Conclusions	81
5.3	Recommendations	82

REFERENCES 84**APPENDICES**

A1	Semester 1 Gantt chart	87
A2	Semester 2 Gantt chart	88
B	Functional codes for selecting point inferred	89
C	Machining program code generated for first model	91

LIST OF TABLES

Table No.	Title	Page
2.1	Operation list example	9
2.2	Speed and feeds calculation sheet example	10
2.3	Various research work based on rotational machining	14
2.4	Max-depth-of-cut for various tool cutters	25
2.5	Factors affecting choice of insert shape	27
4.1	Cutting parameters for aluminium and steel	74

LIST OF FIGURES

Figure No.	Title	Page
2.1	Part drawing sample for process plan given	8
2.2	Manual process plan (routing sheet) corresponding to part shown in Figure 2.1	9
2.3	Process flow in the CNC machining	13
2.4	CAPP generative approach	13
2.5	Basic function of CAD, CAPP and CAM	17
2.6	Turning operation	20
2.7	Variety of cutting operations in turning process	22
2.8	Turning Tool holders	23
2.9	Insert shapes with properties	24
2.10	Turning insert guide	26
2.11	Standard cutting tools	26
2.12	Cutting parameter calculation	29
3.1	NX software core	30
3.2	Flow chart	31
3.3	NX common API	32
3.4	Manufacturing process flow	33

3.5	G-code NC part program generation	33
3.6	Comparison of the process chain	34
3.7	Journal programming language selection	36
3.8	Model for NX software	38
3.9	The measurement of part length	39
3.10	MCS, work plane and spindle centreline	40
3.11	Part body selection	41
3.12	Part blank selection	41
3.13	Defining collision zones	42
3.14	Avoidance setup	43
3.15	Containment setup	44
3.16	Standard grooving tool data	45
3.17	Facing operation toolpath	46
3.18	As level reversal mode toolpath	47
3.19	Omitting reversal mode toolpath	48
3.20	Roughing operation toolpath	48
3.21	Finishing operation toolpath	49
3.22	Grooving cutting region	50
3.23	Grooving operation toolpath	51

3.24	Visual Basic window form	52
3.25	Window form command code	53
3.26	Comment function	54
3.27	Error warning window	55
3.28	Mazak Quick Turn Nexus 200 Turn Center	56
3.29	Technical specifications of Mazak Quick Turn Nexus 200 machine	57
3.30	60mm aluminium round bar rod	58
3.31	3-jaw chuck clamping	59
3.32	Tools holder	59
3.33	Control panel window	60
4.1	Graphical User Interface (GUI) that simplify process planning stage	62
4.2	CNC machining process flow	63
4.3	User interface to create the operation features	64
4.4	Blank size input programming	66
4.5	Blank diameter input	66
4.6	Blank length input	67
4.7	Translucent body selection	68
4.8	CAM model	68
4.9	Inheritance of blank length value	69

4.10	Avoidance value inherited	69
4.11	Containment value inherited	70
4.12	Part cutting region selection task in programming language	71
4.13	Function code added to prompt new selection	71
4.14	Cutting region prompt selection	72
4.15	Materials selection interface	73
4.16	Cutting parameter for aluminium	75
4.17	Cutting parameter for steel	75
4.18	Advance user interface	76
4.19	Time spend comparison for process planning	77
4.20	Machined parts	79

LIST OF SYMBOLS

ipm	Inches per minute
ipr	Inches per revolution
mm	Millimetres
mmpr	Millimetres per revolution
rpm	Revolutions per minute
smm	Surface metres per minute

LIST OF ABBREVIATIONS

AI	Artificial Intelligence
API	Application Programming Interface
BASIC	Beginner's All-purpose Symbolic Instruction Code
CAD	Computer Aided Design
CAM	Computer Aided Manufacturing
CAPP	Computer Aided Process Planning
CNC	Computer Numerical Control
GUI	Graphical User Interface
KBS	Knowledge Based System
MCS	Machine Coordinate System
NC	Numerical Control
RSM	Response Surface Methodology
UI	User Interface
WCS	Work Coordinate System

CHAPTER 1

INTRODUCTION

1.1 RESEARCH OVERVIEW

With the significant improvement of computer technology in Numerical Control (NC) for turning operation, the term of NC machining has evolved into Computer Numerically Controlled (CNC) machining. In recent trends, manufacturer demands for robust automated turning manufacturing process system to meet the goals of rapid production, lead time reduction and cost reduction, while manufacturing high quality products at minimum time [10]. The applications of intelligent systems for example Computer Aided Design (CAD), Computer Aided Process Planning (CAPP) and Computer Aided Manufacturing (CAM) have strengthened the relation between design and manufacturing stages.

These intelligent systems are integrated with the manufacturing processes in order to create the detailed information of part design, process planning and machining sequences in CNC turning manufacturing. Through these intelligent systems application, part conceptualization can be transformed into a real part production. CAD system

providing part design detailed information and CAPP channels these information for CAM to manufacture real part.

Process planning is an essential steps in transform CAD design into a real component. A process planning defines the required manufacturing process, strategies of operations, materials and machining tools needed based on CAD design to manufacture the product. A detailed information such as the machines specifications, tool specifications, setup procedure, operation time estimates have to be included in a process plan [23]. Generally, process planning was implemented manually with allocating sources information comprising machine performance, machinability data, existing stock availability, tooling inventories and production efficiency. Therefore, CAPP is a computerized form of process planning developed to support process planner through knowledge management capabilities and decision support tools.

CAD and CAM are the basic tools used in design and fabrication stages in manufacturing processes. Basically, CAD system is related with the design of component, modelling and simulation analysis of the virtual part. After the design is completed, the Computer Aided Process Planning (CAPP) system will transferred the geometry information which captured from the CAD design file with the allocation of machining process planning to the CAM program. The role of CAM program is to support the planning process, control of production, performance management in the manufacturing of the part by generating the machining code to manufacture the part on CNC machine [7]. The main purpose of constructing NC part programs is to instruct NC machine tool with the detailed set of machining commands.

This proposal discusses the development of machining instructions via CAM system particularly for turning operations. There are three important aspects to be considered in turning process planning task requirements. First, the manufacturing time

spent in process planning must be minimized. Next, the produced part quality attributes become a key concern to meet the part accuracy and surface precision. Lastly, production efficiency and machine performance must be improved through process management capabilities and decision making tools. In order to determine the process planning tools improvements, further investigations on the interaction medium between the operator and CAM software system need to be established.

1.2 PROBLEM STATEMENT

Generally during planning stage, many variables have to be defined including cutting parameters, process parameters and machining operations. An important consideration in planning tasks are usually time spend to construct a standard process plan. Additionally, manual process planning needed skilled process planners, longer process planning time spent and has low level of dependability. In order to support the decision making with large volume of data and to optimise the process planning task, the CAPP technology system is use as process simulator to provide an effective machining process plan.

In pursuance to develop an effective machining program for turning operation where experienced process planner is not necessary needed [23]. Therefore with the implementation of good CAPP system, the time require in process planning for machining parts could be minimized. Moreover manufacturing efficiency can be improved provided that CAPP systems were developed correctly.

1.3 OBJECTIVES OF THE RESEARCH

The aim of this research is:

1. To minimize time spend for process planning in turning operations by implementing programming instructions in CAM systems.
2. To improve the process planning in turning operations by integrating decision making tools and optimized cutting parameters.

Further investigation to improve process planning in turning operations will be discussed in the methodology section. The problems mentioned in section 1.2 have managed to clarify the gaps in the decision making of manual input.

Hence, this research objective was initiated to investigate the integration of user interface program controller that can be adopted in controlling automated machining operations.

1.4 SCOPE

The scope of this thesis is turning machining operations using CNC programming, with the emphasis on process planning and integrating with user interface programs which allows turning cutting parameters to be identified automatically and generating turning machining NC codes as the process output of CAM software. Thus, the process planning between CAM software and user interface program controller is the key body in this turning machining operations.

The primary purpose on behalf of integrating user interface program controller is to provide a more efficient tools in defining the command sequence for NC turning machine operations and to provide optimal parameters for instance feed rate or cutting speed, tool parameter and cutting depth which ultimately contribute to the lead time needed and quality of final product.

CHAPTER 2

LITERATURE REVIEW

2.1 INTRODUCTION

The main purpose of this chapter is to provide the review of knowledge relevant to this project by studying the literatures such as journals, articles, website, publications and any other reliable sources. This is to provide a clearer perspective in the research works done for turning processes. The process planning, Computer Aided Process Planning (CAPP), CAD/CAM, optimization of turning machining techniques, programming language and turning operations are being discussed and reviewed in this chapter.

2.2 PROCESS PLANNING

Process planning is to provide decision making of detailed machining processes of the parts or workpieces so that the part is machined competitively and economically from initial phase to finished phase [25]. As described by Chang et al. (1990), process planning is likewise a method designed for formulating completed machining operation commands to convert manufacturing part design into a real finish manufactured or

assembly part [5,6]. Manufacturing process planning is concerned with defining all execution steps in production stage which including properties, material, design and fabrication process of a part. The process planner have to consider various factors in order to clarify the product detailed planning, therefore to precede the planning task there is a need for simplified the design decision process. A complete process planning can achieve significant time reduction from design stage to manufacturing stage by:

- i) Provide support to the process planner in term of information transferring from database.
- ii) Develop automatic process planning system.
- iii) Automating part data transferring from CAD system to CAPP system.

Process planning is basically decided by the production engineer or process planner to determine the machining operation sequence and the most appropriate process parameters. The purpose of a process plan is to deliver clear and precise sequential route where the part is fabricated. In addition, process planning indicated the most optimal selection of tools, processes and parameters required for manufacturing. Decision making throughout the process planning will determine the production time, as well as the product quality and cost. The development of automatic process planning is proposed to assist production planning phase due to inconsistency planning, reduce time in manually generate a process plan and develop appropriate cutting toolpaths for CNC machining.

The function of process planning are as follow:

- i) To determine and select appropriate tools needed to manufacture the part

- ii) To determine the operations sequence to manufacture in term of operation toolpath routing and process details
- iii) To specify the process parameters for selected machining operations
- iv) To provide necessary documentation for the shop floor level

Figure 2.1 below shows the part drawing sample for wrench handle. A manual process plan is generated specifically for this part to assist the production planning phase. Figure 2.2 below displays that the manual process plan (routing sheet) for the part drawing given in figure 2.1. The routing sheet is listed all the machining operations and tools used for manufacturing this wrench handle.

Table 2.1 below illustrates the operation list example for process planning that describe the machining operations with their respectively cutting parameters and operation time required. Table 2.2 below shows the speeds and feeds calculation sheet example to determine the appropriate cutting parameters used for machining operations.

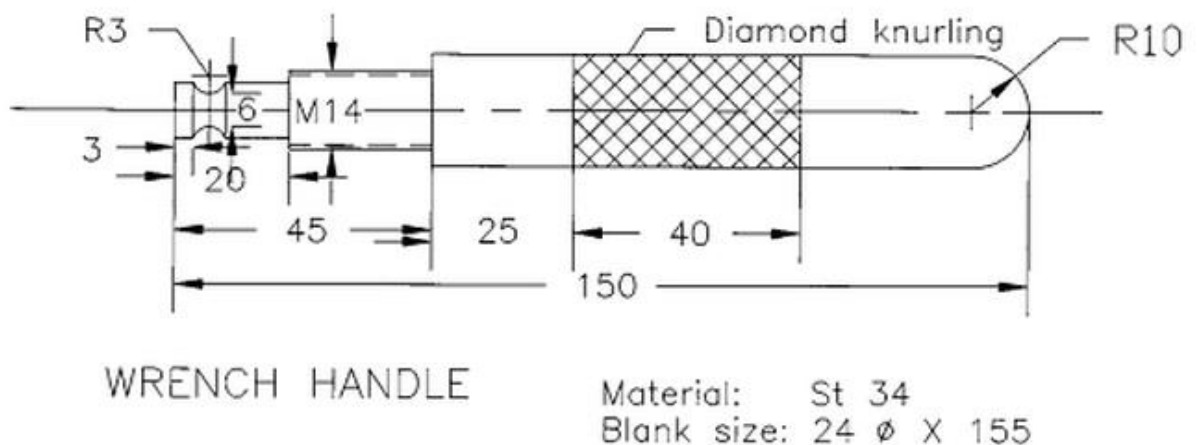


Figure 2.1: Part drawing sample for process plan given [17]

Op.No.	Operation	Tools Used
10	Turning to length and facing the ends	Facing tool, chuck
20	Make the center hole	Chuck, center drill
30	Rough turning of $\phi 20$ diameter	Rough turning tool
40	Finish turning of $\phi 20$ diameter	Finish turning tool
50	Forming the radius at one end	Form tool (radius turning)
60	Diamond knurling of the handle	Knurling tool
70	Reverse the part in the chuck and rough turn to $\phi 10$	Rough turning tool
80	Finish turning of $\phi 10$ size	Finish turning tool
90	Make the 3 mm radius groove	Form tool
100	Rough turning of $\phi 14$ diameter	Rough turning tool
110	Finish turning of $\phi 14$ diameter	Finish turning tool
120	Cutting the external threads M14	Thread cutting tool

Figure 2.2: Manual process plan (routing sheet) for part shown in Figure 2.1 [17]

Source: Leondes, C. T. (2000)

Table 2.1: Operation list example [17]

Operations list								
Part name:			Part no.: LH-0208-87			Drg. no.: CS-2003-88		
Revision no.:		Date: 27/02/01		Page 1 of 1		Planner: L.E. Hall		
Op. no.	Description	Machine tool	Tooling	Speed (rev/min)	Feed (mm/min)	Set-up time (min)	Op. time (min)	Remarks
10	Face end	Lathe No. 4	HSS LH copy	100	20		2.25	Use universal 3-jaw chuck
20	Face to length	Lathe No. 4	HSS LH copy	100	20		4.25	
30	Rough turn 10mm shoulder	Lathe No. 4	HSS LH copy	130	25		2.2	
40	Finish turn 100mm shoulder	Lathe No. 4	HSS LH copy	180	35		0.3	

Source: Leondes, C. T. (2000)

Table 2.2: Speed and feeds calculation sheet example [17]

Speeds and feeds sheet			
Page: 1/1	Planner: Peter Scallan	Date: 08/08/02	Part no.: PS/0904/02
Drg. no.: JH/2905/02	Part name: Machined component		Material: Aluminium alloy
Op. no.	Surface cutting speed	Spindle speed	Feed rate
1	$S = 300 \text{ m/min}$	$N = \frac{1000 \times S}{\pi \times d}$ $N = \frac{1000 \times 300}{\pi \times 100}$ $N = 956 \text{ rev/min}^*$	$\text{mm/rev} = \text{mm/t} \times \text{no. of t}$ $= 0.6 \times 10$ $= 6 \text{ mm/rev}$ $F = \text{mm/rev} \times N$ $= 6 \times 956$ $= 5736 \text{ mm/min}$ $= 5750 \text{ mm/min}^*$
2	$S = 300 \text{ m/min}$	$N = \frac{1000 \times S}{\pi \times d}$ $N = \frac{1000 \times 300}{\pi \times 20}$ $N = 4775 \text{ rev/min}^*$	$\text{mm/rev} = \text{mm/t} \times \text{No. of t}$ $= 0.25 \times 6$ $= 1.5$ $F = \text{mm/rev} \times N$ $= 1.5 \times 4775$ $= 7163 \text{ mm/min}$ $= 7200 \text{ mm/min} \#$

Source: Leondes, C. T. (2000)

2.3 COMPUTER AIDED PROCESS PLANNING (CAPP)

Computer Aided process planning (CAPP) is functions as a key element of a computer integrated manufacturing (CIM) system. Utilizing CAPP system is to generate a process plans consistently using automatic process simulator [4]. It involves the planning task for part feature selection, tool selection, process selection and sequence arrangement. Jahan and Kabir (2010) addressed that CAPP simplifies decision required to plan for the manufacturing of part and achieves the effective application of machining resources [12]. With the detailed inputs such as geometric features, size

dimension, tolerances and material selection, appropriate process sequence can be generated based upon the analysis and evaluation of these inputs.

However, a superior process reasoning appliance is needed as the basic fundamental of process planning to present the automated process planning machining system. There have been many commercial CAM software available, but it has the issue of compatibility requirements to adapt different nature of applications. Although there have been increasing engineers and researchers that attempted to further develop the CAPP system, but the implementation of most CAPP systems still needed experienced and skilled process planners.

Another issue is due to the fact of different manufacturing environment and industries need different inference mechanism and process knowledge representation in develop process planning. Therefore, the application that rely on automated process planning is still a long way to be realized. It is important to create a proper inference mechanism in selection of process parameters as well as to produce an efficient model of process knowledge representation, both quantitatively and qualitatively in process planning.

In general CAPP has two basic approaches, which are variant and generative. For variant approach CAPP follows the concept of similar parts apply similar plans. This concept is using group technology based on computerized planning retrieval techniques for classification and part coding. The processes are retrieve from the system library and apply standard procedure plans for similar works [14]. However experienced process operator is still required to identify the process plan input information for that particular part and make modification if necessary. In generative CAPP, process plan is generated from manufacturing database's available information with little intervention from human. Upon identifying the part design, it will create the required process plan

based on part information, decision logic, algorithms and process knowledge without reference to existing plans [19]. However, different generative decision making logics are needed to be customised for different manufacturer company.

Variant and generative approach CAPP is then evolving into semi-generative and expert CAPP system. In semi-generative system, the existing process plan can make modification same as variant system or generate a new process plan using generative system. While in expert approach, numerous Knowledge Based System (KBS) and Artificial Intelligence (AI) tools are implemented to automate the process plan [9]. Therefore, a successful CAPP system encompasses several tasks for example part feature identification, process selection, tool selection, machining parameters set up, process sequencing and lastly Computer Numerical Control (CNC) code generation that preparing detailed operation instructions to perform manufacturing.

CAPP system required the basic description input of geometry features, which including extrude, cut, slot, hole and etc. features used in part model. Xu et al (2011) described by using feature recognition approach, we can determine the geometry and features of a part design. Another approach used to determine feature is by using predefined features that stored in the feature library [24]. In the past decades, research work in extracting manufacturing features has been carried out. Verma and Rajotia (2010) studied numerous feature recognition techniques and then reviewed the advantages and disadvantages of the approach [22]. Vasmi Krishna et al. (2011) applied modelling design feature approach to obtain the feature description information [15]. Chu et al. (2012) implemented identification of machining features for mapping with cutting tool resources available [8].

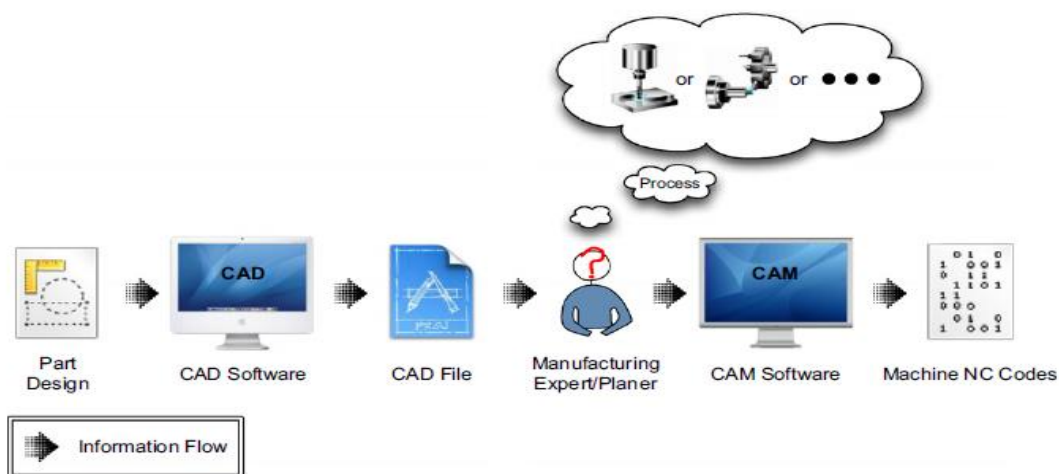


Figure 2.3: Process flow in the CNC machining [20]

Source: Safaieh M., (2013)

Figure 2.3 above shows the process flow in the CNC machining. Experienced process planer is still required to identify the process plan input information for that particular part and make modification if necessary.

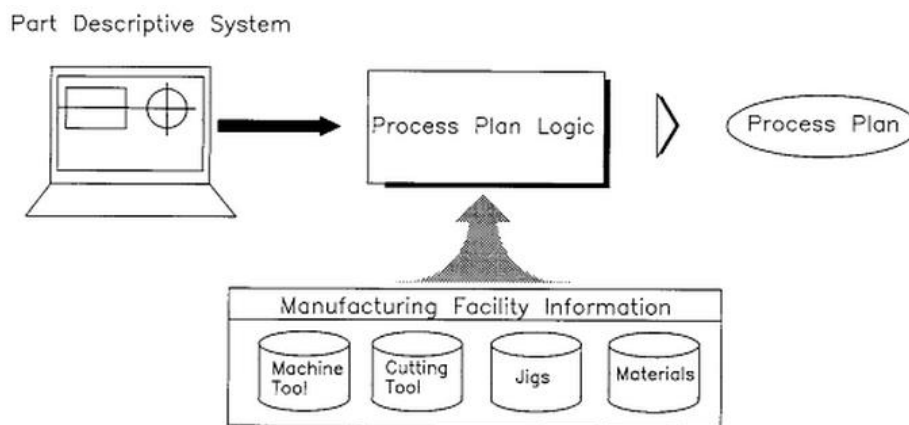


Figure 2.4: CAPP generative approach [17]

Source: Leondes, C. T. (2000)

Figure 2.4 above illustrates the CAPP generative approach in process planning where process plan is generated from manufacturing database's available information.

Table 2.3: Various research work based on rotational machining [8]

Parts	Authors	Viewpoint of CAPP	Machine tools	Reasoning techniques	Programming languages	Representation of geometrical and technological details
Rotational	Roa et al.	Entire process started from feature based modelling system (FBMS) until generating NC tool path	2-axis lathe	Expert system	-	Feature based recognition and modelling
	Yeo	Entire process started from feature recognition until generating NC tool path	2-axis lathe	The goal directed forward chaining technique	Goldworks & Common Lisp	Feature recognition
	Loh et al.	Automated feature recognition, operation planning and machinability data selection	2-axis lathe	Artificial intelligence (AI)	Goldworks & Common Lisp	Feature recognition
	Zhao et al.	Tool selection and cutting conditions	2-axis lathe	Expert system	Prolog	IGES with feature recognition Syntax descriptions
			2-axis lathe	Expert system	Prolog	
	Shunmugam et al.	Operation sequencing	2-axis lathe	Genetic algorithm	C++	Interactively entered by user
	Yildiz et al.	Tool path generation	2-axis lathe	Definition rule of features	Delphi 7	DFX with feature recognition

Source: Chu, X., Tang, C., Xue, D., Su, Y., & Sun, X. (2012).

Table 2.3 shows various research work based on rotational machining from previous research in CAPP. Different reasoning techniques and programming language is applied in the research work done.

The objective of CAPP system is fulfilled when CAD system successful to process the feature information of part and generates the process planning tasks for manufacturing. It is indeed a challenging task to develop improvement in CAPP system to support manufacturing process from feature extraction till the CNC code generation. Xu et al (2011) stated that various research work mainly for CAPP system development in rotational parts has been carried out [24]. Jahan and Kabir (2010) established a turning CAPP system that extracts feature information from data file and using few logic such as mathematic logic and algorithms to generate NC code [12].

2.4 CAD / CAM

Computer Aided Design (CAD) is functioning as the implementation of computer technology that creates two-dimensional (2D) or three-dimensional (3D) product design and documents the design's procedure as manufacturing database [16]. The basic CAD system aids in design process of creation, modification, and analysis. The difference between manual drawing and using CAD system is that CAD system provided automating designs function and modelling design with more precise and complete details.

Computer Aided Manufacturing (CAM) is an application that uses computer technology to manipulate manufacturing processes and related machinery to support all manufacturing operations. These operations including managing, planning, material tracking and transportation. It use to improve manufacturing process via precise tooling accuracy, material consistency and increased manufacturing speeds. CAM also applied advanced productivity tools such as simulation and optimization tools to leverage skills of manufacturing professionals [1].

Historically, contribution of skilled machinists in selection of tools, machining process and paths is necessary. Functioning knowledge of machining programming code like G-code and M-code is usually required in manual editing to ensure the program code run correctly. Over time, the development of CAM nowadays able to provide specific capabilities such as recording process wizards, automate featured based machining and machine tools user interface for ease of user [13]. Agrawal et al. (2013) make use of ALPHACAM software to perform process planning task automatically which minimize the manual interference of user [2]. The application of CAM software revealed that with algorithms or programming instructions, it is possible to achieve automated process planning and fulfil the objective to reduce dependency on manual planning.

In prior to that, CAD/CAM is the combination of CAD and CAM as an application software system that establish product design and manufacturing function into one system. The integration of CAD/CAM not only limited to automate design and manufacturing phases, furthermore, it provides automatic conversion from design to manufacturing [11]. Development of CAD/CAM system capable to extract the design details of product as in CAD database thus automatically convert it become process plan for manufacturing product. Considering the automated process planning permits the rapid machining processes, programming instructions has been developed within commercial CAD/CAM software as interaction medium with user. These commercial CAD/CAM software including NX Unigraphics, CATIA, Autodesk INVENTOR and MasterCAM allow integration of either programming languages such as C, C++, Java and Visual Basic. Figure 2.5 shows the basic function of CAD, CAPP and CAM in most commercial CAD/CAM software.

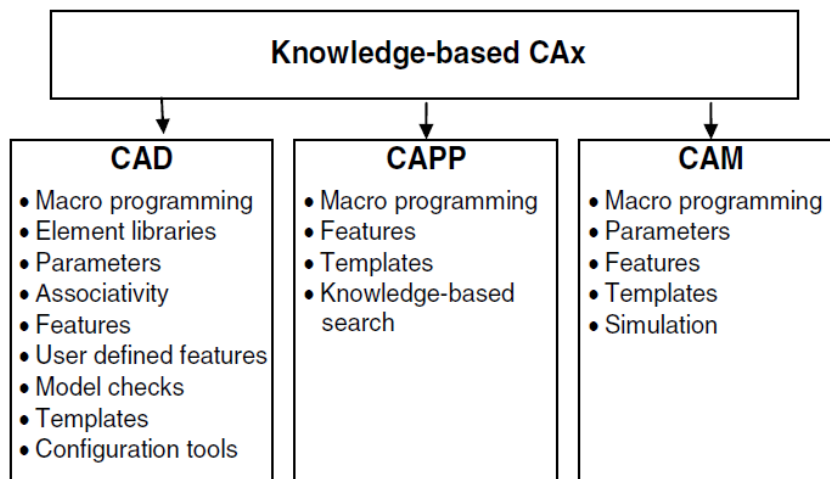


Figure 2.5: Basic function of CAD, CAPP and CAM

2.5 OPTIMIZATION OF TURNING MACHINING TECHNIQUES

Regarding optimal selection of cutting tools and cutting parameters, many industries still needed skilled and experienced machine operator to optimize economic cutting operations. Product quality of turned part will be determined by these cutting parameters and cutting tools selection. Selecting optimum cutting parameters for instance cutting speed, feed rate and cutting depth may slow down the machining progress as all process parameters have to be considered before machining. Traditionally, machine operator plays a major role in the cutting parameters selection, but it is even a difficult task for experienced machine operator to attain optimum values every single time. Therefore, various algorithms has been integrated with CAD/CAM to provide optimum process parameters for turning machining.

Most of these algorithms are computed automatically according to their programming input. Since Taylor (1907) pioneered the approach to investigate for the optimization method of machining parameters, increased study work has been

commenced in the area of optimization of machining parameters. Armarego & Brown (1969) examined unrestrained machine parameters optimization by means of differential calculus. Walvekar & Lambert (1970) investigated the application of geometric programming in machining variables selection. Petropoulos (1973) applied geometric programming in optimal selection of cutting speed, feed rate and machining rate variables. Prasad et al (1997) formulated process parameter optimization module for turning operations based on generative CAPP system [16].

Several old-style machining optimization methods for instance Lagrange's method, goal programming, dynamic programming and geometric programming are being applied effectively in turning operations for process variables optimization. The latest optimization techniques for process parameter optimal selection that widely used in manufacturing including scatter search technique, Taguchi technique, genetic algorithm, fuzzy logic, and response surface methodology (RSM) [16]. The implementation of algorithms manages to reduce setup and planning time for machining operations, therefore it is an ultimate goal to support machining processes at the planning stage.

2.6 PROGRAMMING LANGUAGE

Programming language has been developed within the CAD/CAM software and used to provide instructions that control the operations in CAD/CAM software. Various programming language such as BASIC (Beginner's All-purpose Symbolic Instruction Code), JAVA, C and C++ are being widely used in CAD/CAM system. These programming languages are necessary interpreters and compilers that act as translator for system. Interpreter is the execution of programme once the program language is

translated. While for compiler, the completed programme can be executed every times once it is translated into machining language [18].

Programming ability like Macro programming within a CAD/CAM software is a very convenient feature. The existence of Macro programming facility possible the user to insert minimum required information for design modelling or specific application available in the system. But macros can only be used in limited access to the geometry. One of the programming utilities for instance Application Programming Interface (API) allows regular standard access to software functional library and providing user to modify and develop their particular commercial application [18].

2.7 Turning Operations

Turning machining is described as process used to produce cylinder-shaped workpiece with the single point tool by using lathes machine for turning process. It is the material removal from the outer distance of a rotating cylinder-shaped surfaces. Normally workpiece will rotate on a lathe spindle then the cutting tool is moved into the workpiece linearly in the parallel direction, perpendicular direction or both directions at the same time to produce the complex rotational surfaces required. The primary motion of cutting in turning is the rotation of workpiece while the secondary motion of cutting is the tool feed motion [21].

The cutting behaviours for most turning process are similar, only one cutting tool is used for a particular surface and the feeding motion direction is primarily axial with respect to the spindle. The workpiece and the cutting tool are generally in contact once the cutting process is started, with the constant cutting speed and feed rate. Consideration of proper rotational speed and the cutting tool feed motion relative to the

workpiece is therefore necessary. Figure 2.6 below illustrates the turning operation with direction of tool feed motion and primary motion.

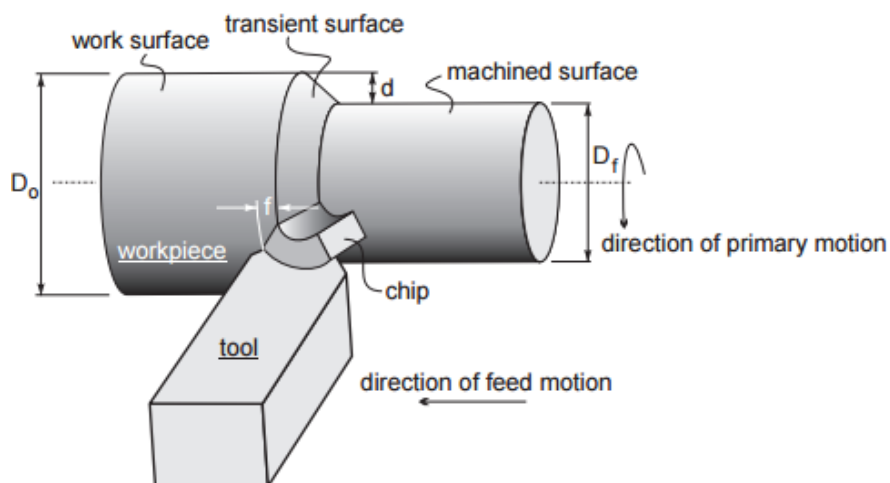


Figure 2.6: Turning operation

In general, single point tools are used in most turning machining tasks such as:

- Turning - Used to turn a cylindrical surface by removing the excess materials with the required depth
- Facing - used to produce a square surface at the end edges of the workpiece or to get a desired length
- Tapper – material removing at an angle create a conical geometry
- Chamfering - used to produce an angle on the corner of a surface
- Grooving - produces a groove moving along a straight line (axis Z)
- Threading - A pointed tool is fed straight through external or internal surface of rotating workpiece for creating threads line
- Contour profile – cutting tool follow profile contour to create shape

- Cut-off - The cutting tool is inserted radially to the rotating workpiece with a precise distance to part off a part
- Drilling - produce a hole by using the drill tools and fed along its axis into the rotating work
- Boring - Enlarging a hole made in previous process by fed a single-point tool linearly to the axis of rotation
- Reaming - Used to enlarge the existing holes to produce a good internal finish with accurate diameters.
- Knurling – Produce a regular cross-hatched pattern in surface of workpiece by mean of metal forming operation

Figure 2.7 below illustrates the most single point tools cutting operations used in turning process. The cutting operations are as described as above.

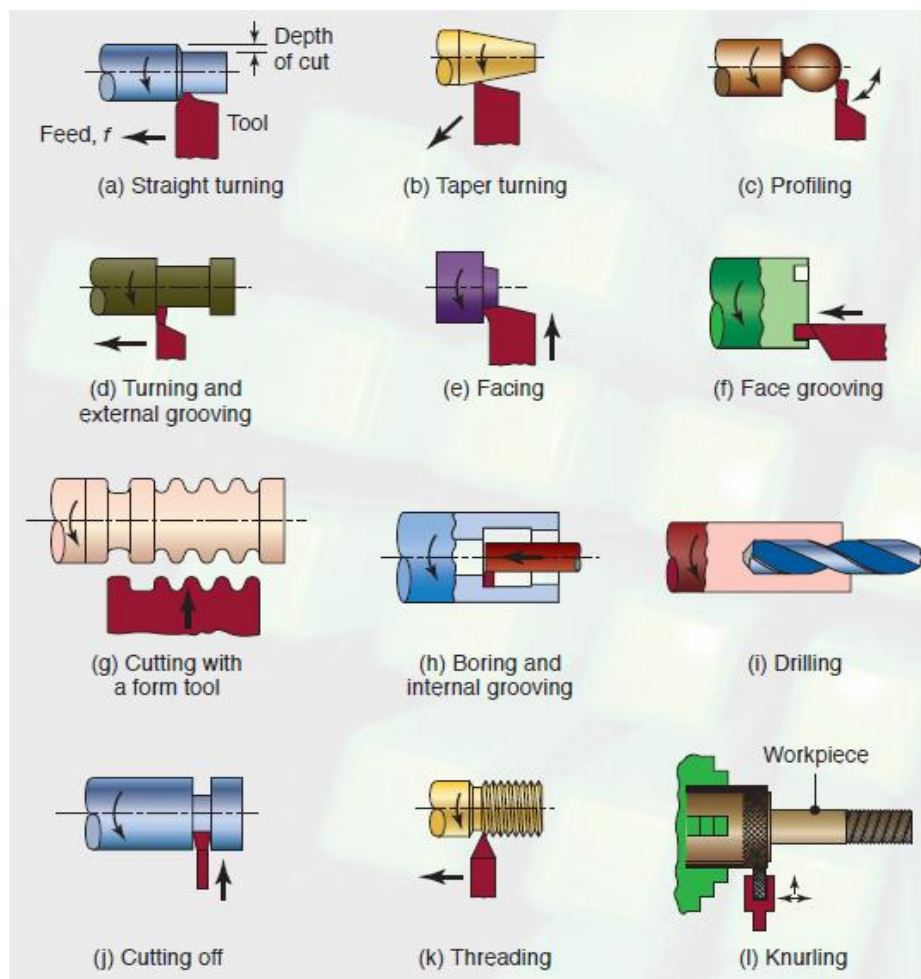


Figure 2.7: Variety of cutting operations in turning process

Source: Google image

2.7.1 Tool Holder Styles

The ANSI system outlined that there are 7 basic tool styles for lathe machining operations. The descriptions for the 7 major tool holder styles are A, B, C, D, E, F and G [21]. Figure 2.8 shows the turning tool holders used in lathe machine.

A Style: 0° side-cutting edge angle using straight shank for turning operations.

B Style: 15° side-cutting edge angle using straight shank for turning operations.

C Style: 0° end-cutting edge angle using straight shank for cut-off and grooving operations.

D Style: 45° side-cutting edge angle using straight shank for turning operations.

E Style: 30° side-cutting edge angle using straight shank for threading operations.

F Style: 0° end-cutting edge angle using offset shank for facing operations.

G Style: 0° side-cutting edge angle using offset shank, this G style is kind of 'A' style for turning operations close to the lathe chuck with additional clearance built in.



Figure 2.8: Turning Tool holders

Source: Google image

2.7.2 Shapes of Turning Insert

Turning inserts selection are depending to their shapes, sizes and thicknesses. The selection of the correct insert shape and chip breaker geometry, accompanied by the appropriate turning tool holder will eventually shows a significant impact on the tool

life and productivity during a particular turning process. Coated carbide inserts are best to use as they bring high productivity to the manufacturing process.

Another vital aspect in choosing the appropriate geometry for a hardness range of workpiece material is the insert strength. The most common shaped inserts predominantly used is triangle inserts because they are able to machining in a wide application. As mentioned previously, any of the seven basic turning holder styles can use a triangular insert for the machining operation. Squares-shaped inserts are regularly used on lead angle tools while diamond-shaped inserts are utilized in profile turning operations. Insert strength is basically rated according to the general rule based on shape of the insert, which the greater the included angle on the insert corner, the better the insert strength. Figure 2.9 displays the insert shapes with its properties regarding the cutting edge strength and vibration tendency.

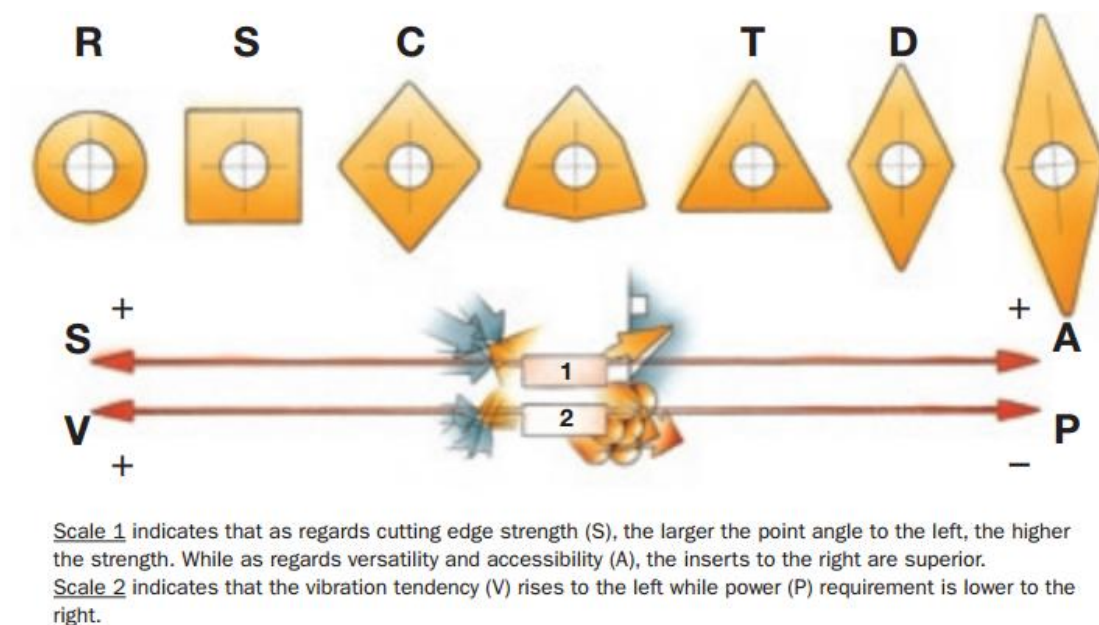









Figure 2.9: Insert shapes with properties

Source: Sandvik, Google Image

A large point angle is strong but needs more machine power and has a higher tendency to vibrate due to having a large cutting edge engaged in cut. The small point angle is weaker and has a smaller cutting edge engagement in cut, which can make it more sensitive to the effects of heat. Each insert shape has a set maximum effective cutting edge length which effects the depth of cut possible. The 80-degree point angle, rhombic-shaped insert is commonly used as it has effective compromise and appropriate for various operations. Table 2.4 below show the max depth of cut for various tool cutter shape with the calculation method. Figure 2.10 and figure 2.11 displays the turning insert guide and standard cutting tools respectively. The cutting tool inserts is selected according to the machining operations performed as shown in table 2.5.

Table 2.4: Max-depth-of-cut for various tool cutters

Shape		Max. cutting depth
S		0.66 x Cutting edge length
C		0.66 x Cutting edge length
T		0.5 x Cutting edge length
D		0.5 x Cutting edge length
V		0.25 x Cutting edge length
W		0.25 x Cutting edge length
R		0.4 x Insert dia.

Source: Kennametal, Google

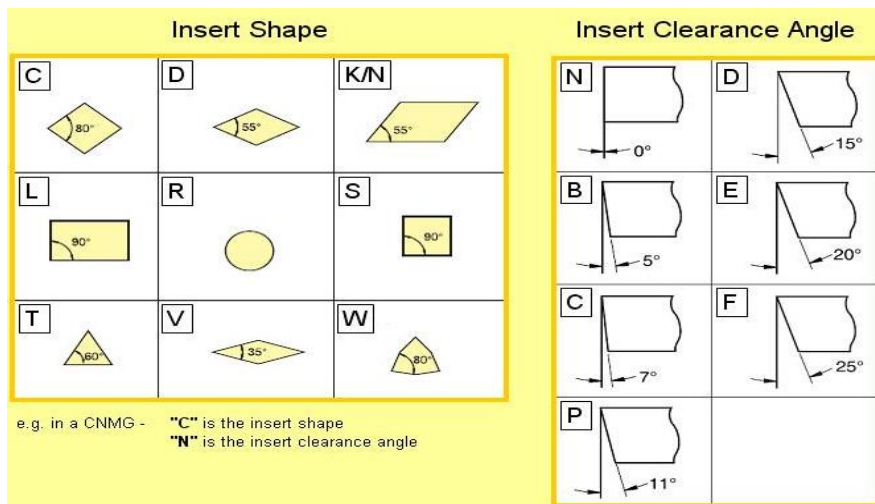


Figure 2.10: Turning insert guide

Source: Kennametal, Google

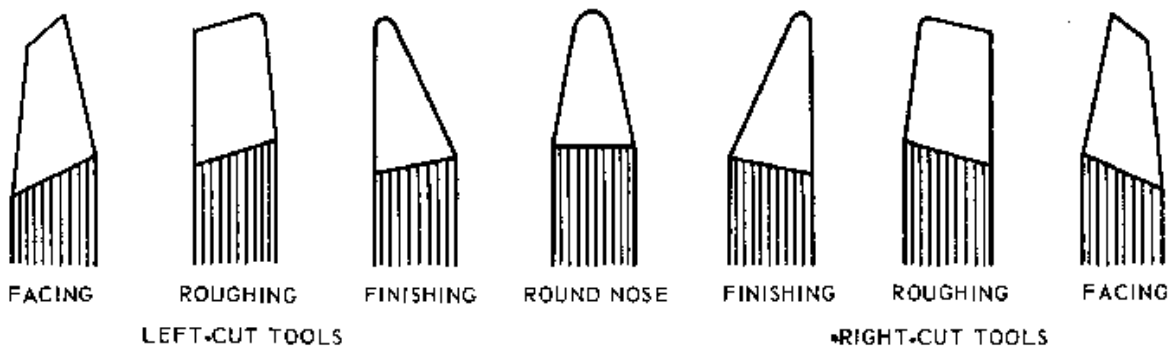


Figure 2.11: Standard cutting tools

Source: Google image

Table 2.5: Factors affecting choice of insert shape

Factors	R°	90°	80°	80°	60°	55°	35°
- Roughing (strength)	■	■	■	□	□		
- Light roughing/Semi finishing (No. of edges)		□	■	■	■	■	
- Finishing (No. of edges)			□	□	■	■	■
- Turning and facing (Feed directions)			■	□	□	■	■
- Profiling (Accessibility)			□	□	□	■	■
- Operational versatility	□		■	□	□	■	□
- Limited machine power			□	□	■	■	■
- Vibration tendencies (Reduction)				□	■	■	■
- Hard material	■	■					
- Intermittent machining	■	■	□	□	□		
- Small lead angle			■	■	■	■	■
- Large lead angle	■	■			■		

■ Most suitable □ Suitable

Source: Sandvik Catalogue, Google

2.7.3 Operating Conditions

Operating conditions manipulate three significant material removal aspects: metal cutting rate, surface finish and tool life. These three aspects have to be balanced and to attain least possible machining cost, better surface finish and maximum production rate, appropriate operating conditions must be selected whichever is desirable for a particular turning process. The setup of the workpiece alignment and the cutting tool will affected the performance rate of any machining operation [21].

The improper alignment of the workpiece, the cutting tool conditions and the machine performance is always existing minor inaccuracy and is typically so insignificant that shows no impact on machining operation. Whenever the unnoticed alignments causing in vibration, distortion or chatter then only it becomes a problem.

Thus it is very important to make sure that the setup is appropriate and suitable for the type of turning operation to be executed by taking the essential time and effort. This is particularly vital when machining heavy cuts or interrupted cuts.

2.7.4 Cutting Conditions

The following main cutting parameters are selected based on the tool material size and workpiece material [21]. The parameters are

- Cutting feed rate - The axial motion of the turning cutting tool along the workpiece for each revolution of the workpiece expressed as inches per revolution (IPR). The feed is also expressed as a distance travelled in a single minute or IPM (inches per minute).
- Cutting speed- The relative surface speed between tool and workpiece, expressed in surface feet per minute. Some machine tool is built to operate in revolutions per minute, then it requires converting surface speeds into revolutions per minute (RPM).
- Radial depth of cut and axial depth of cut - Relates to the depth of the tool cutting edge engages the workpiece. One linear dimension of the area of cut is determined by the depth of cut.
- Spindle speed – The manner of spindle to be rotating in minute, usually measure in revolutions per minute (RPM).

Figure 2.12 below illustrates the calculation method for cutting parameters in turning process.

Machining Calculations: Turning

<ul style="list-style-type: none"> Spindle Speed - N <ul style="list-style-type: none"> v = cutting speed D_o = outer diameter 	$N = \frac{v}{\pi D_o}$	(rpm)
<ul style="list-style-type: none"> Feed Rate - f_r <ul style="list-style-type: none"> f = feed per rev 	$f_r = N f$	(mm/min -or- in/min)
<ul style="list-style-type: none"> Depth of Cut - d <ul style="list-style-type: none"> D_o = outer diameter D_f = final diameter 	$d = \frac{D_o - D_f}{2}$	(mm -or- in)
<ul style="list-style-type: none"> Machining Time - T_m <ul style="list-style-type: none"> L = length of cut 	$T_m = \frac{L}{f_r}$	(min)
<ul style="list-style-type: none"> Mat'l Removal Rate - MRR 	$MRR = v f d$	(mm ³ /min -or- in ³ /min)

Figure 2.12: Cutting parameter calculation

Source: George Schneider Jr., (2010).

2.8 Summary

This chapter has addressed the development of the process planning, Computer Aided Process Planning (CAPP), CAD/CAM, optimization of turning machining techniques, programming language and turning operations. These applications is trying to implement into the field of turning machining to increase the ability for automation in the planning task and at the same time improving the machining approach.

CHAPTER 3

RESEARCH DESIGN AND METHODOLOGY

3.1 INTRODUCTION

NX software will be used as the program to assist turning machining process planning tasks. The process can be divided into design phase and manufacturing phase. During design process identified relevant design information and detailed design, then a feasibility study is performed and analyses are conducted. The manufacturing process begins with production planning includes tasks such as operations planning, material selection and machine procurement. During the production process there are varied tasks at different stages like NC programming, procurement of tools and quality checks for the processes used in product manufacturing. Figure 3.1 shows the NX software common API available.

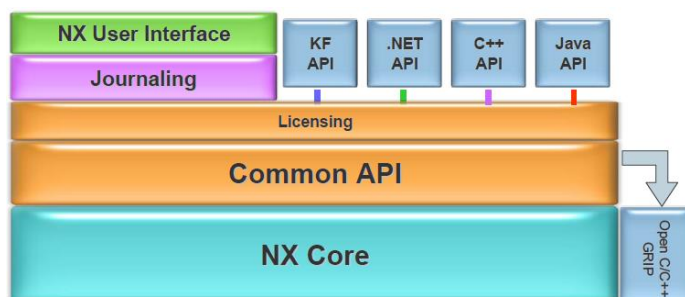


Figure 3.1: NX software core

3.2 FLOW CHART

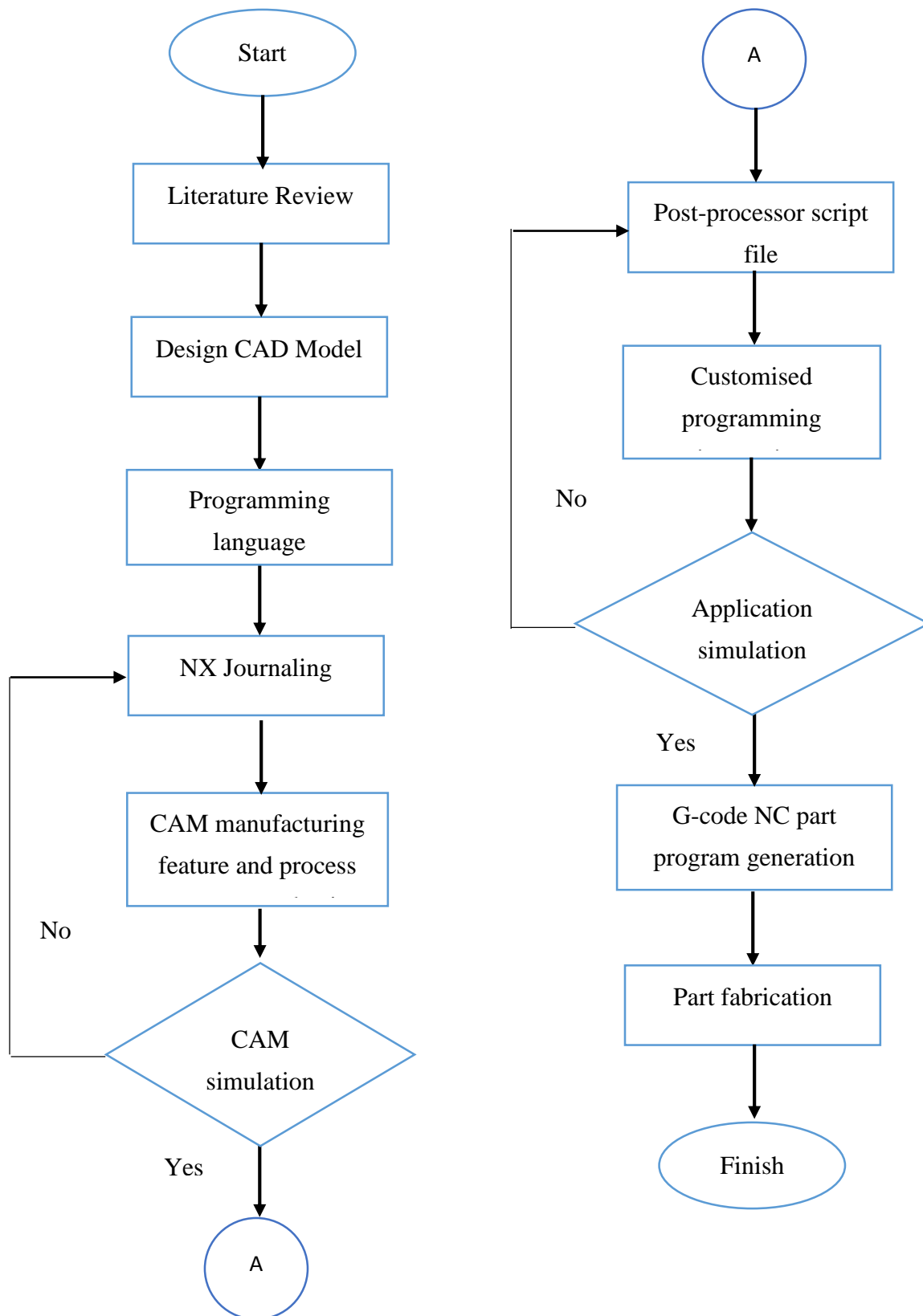


Figure 3.2: Flow chart

3.3 NX SOFTWARE

NX software is equipped with common API (Application Programming Interface) and programming tools which acts as automation foundation in creating machining operations. There is one tool called Journaling in NX software that is capable of program interactively and the ability to record using the common API. After the recording, the program can play or playback with the execution of previous recorded journal to reproduce the same action as recorded. The recorded action using journaling will be generated as a script file where can be translated into programming languages such as Visual Basic, C++ and Java as illustrated in figure 3.3.

Moreover, the script file can be edited to create particular functions and integrating with user interface to ease the commands. This journaling method is suitable to automating the repetitive workload and to automate process planning needed as the proposed approaches in this research study. Consequently, various programming steps can be avoided since by using this journaling tools to record and translated into script file that can be replay. The only challenging task for this journaling tools is the modification and editing of the programming language codes to perform specific function in turning processes.

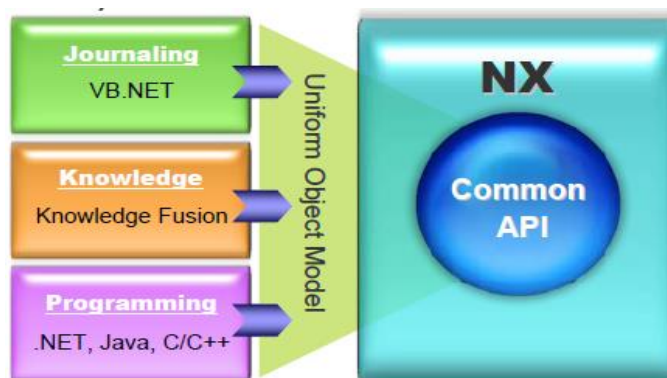


Figure 3.3: NX common API

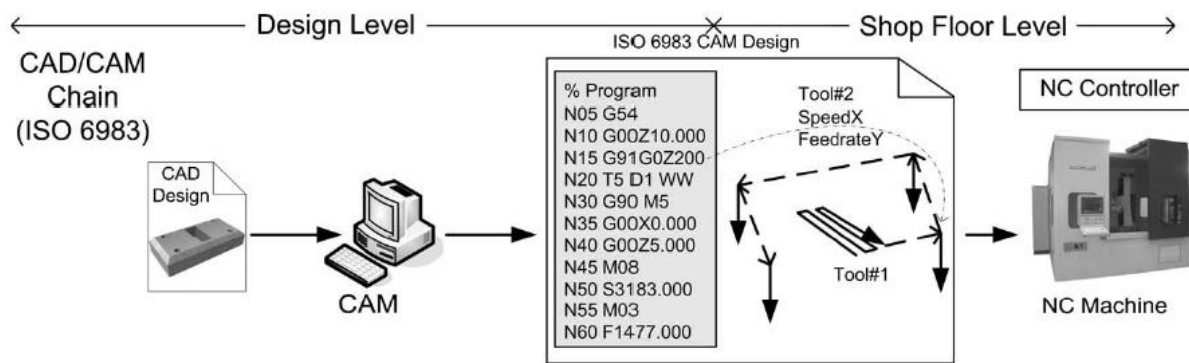


Figure 3.4: Manufacturing process flow [3]

Source: J. G. Campos (2011)

Figure 3.4 above illustrates the process flow of manufacturing process from initial stage of CAD design until the manufacturing stage using NC machine. While Figure 3.5 below displays the various machining code using to give instructions to NC machine to be operated.

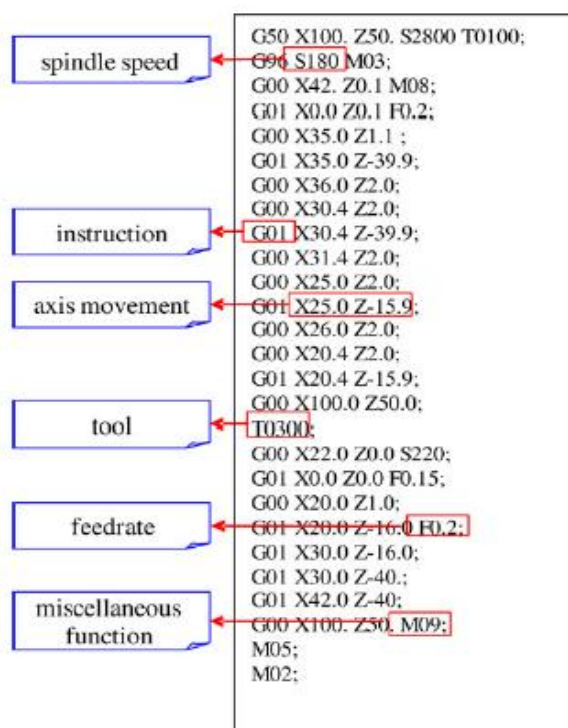


Figure 3.5: G-code NC part program generation

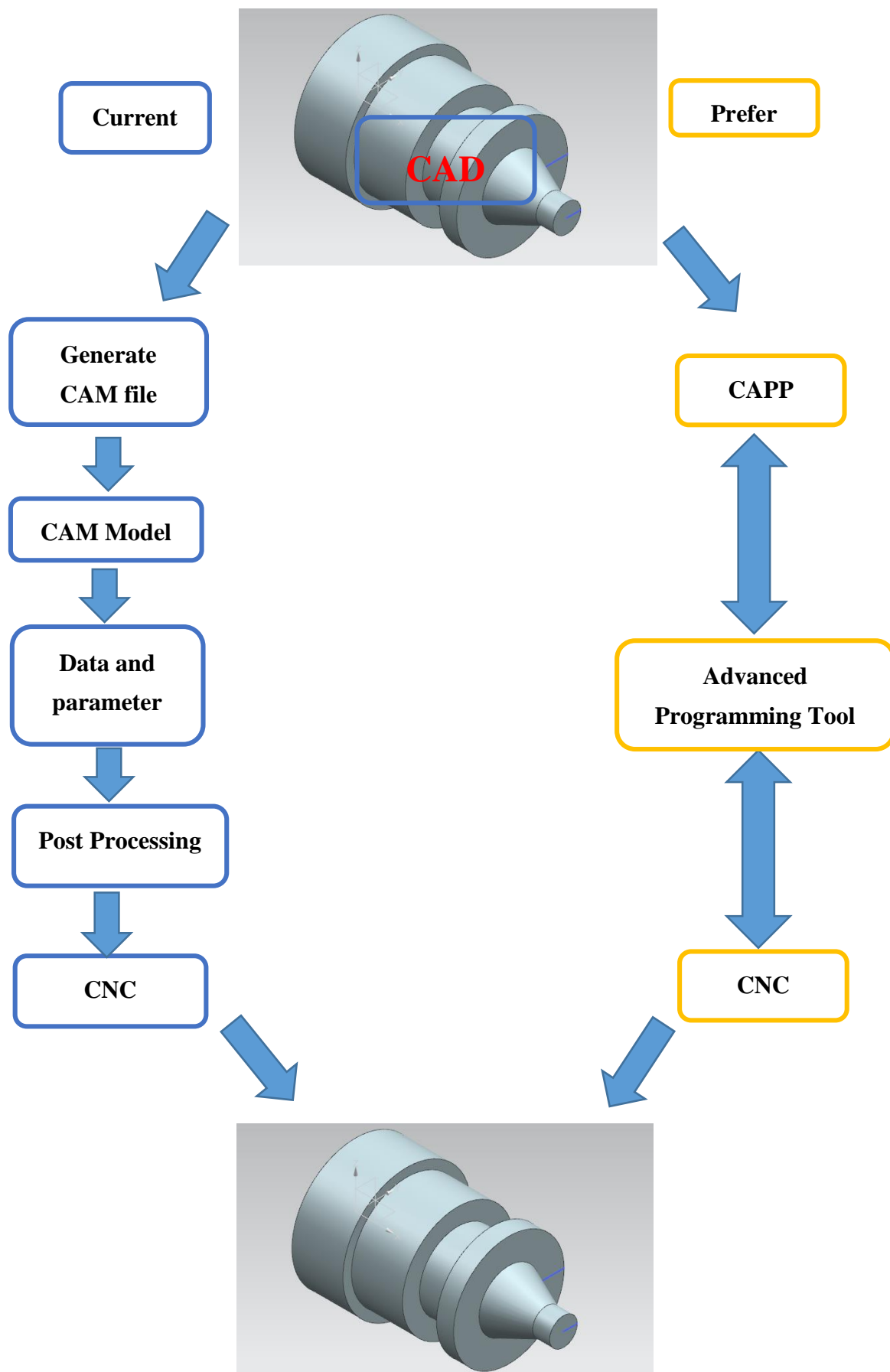


Figure 3.6: Comparison of the process chain

Generally, in the process flow of manufacturing process will begin from the initial stage of using CAD design, then transfer the CAD design to CAM model. The process is then continue with the inserting of operation features and their cutting parameters. After that the CAM file will generate the complete machining program to NC machine for machining purpose. In this research, utilization of CAPP system within the NX software allows to develop graphical user interface (GUI) as the controller to support the manufactruing process from feature recognition till the CNC code generation. Figure 3.6 illustrates the comparison of the process chain between current process flow and the utilization of user interface controller for automation process task.

3.3.1 Journaling

The Journaling is a rapid automation tool that has the capabilities to records, edits and playback interactive NX sessions. A script file is then produced based on the programming language Visual Basic and Open C++ and Java, which can be replay the session at a later time using the script file. Moreover, the generated script file can be edited with simple programming and enhanced with user interface components to produce a customized program to perform automation commands. Therefore, Journal files can be used as a template for automating repetitive workflow because it are based on the Common API and can be used as a basic technique in any of the supported automation languages for working automation code.

3.3.2 Beginning of Journaling

Before create a journal file, we have to specify what type of programming language for the journaling in NX to be recorded in. This is done by opening the NX software application and then going to the Preferences Menu, next selecting User Interface. Looking for Journaling tab and select the programming language you preferred and then hit OK. In this case, Visual Basic language will be used. Figure 3.7 shows the procedure in selecting the programming language.

Once the journal programming language is set, by selecting “Tools >> Journal >> Record”, Journaling can now begin. A prompt warning may appear to notify that when in the Journaling process, not all things could be recorded, by hitting OK to close this prompt. Next, specify name and the location where the journal file to be saved.

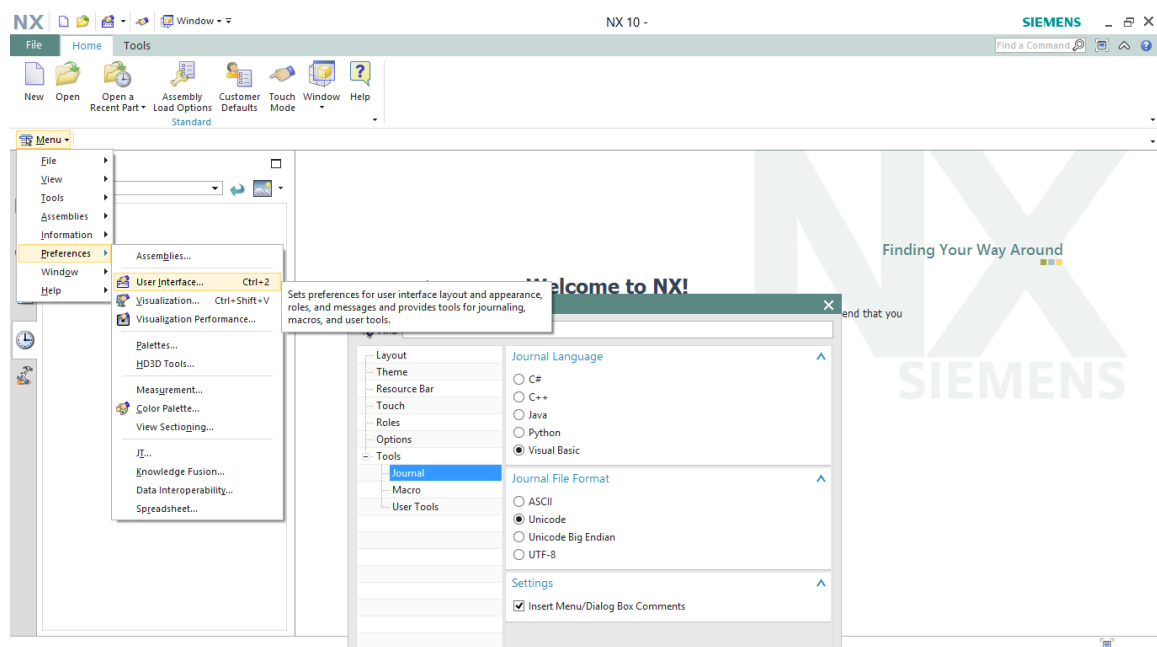


Figure 3.7: Journal programming language selection

After the journaling process is begin, there will be small green boxes appear next to the different commands to indicate which commands can be journaled. Perform the intended operations and once complete the operations go back to the tools menu, Select “Tools >> Journal >> Stop” in order to end the journal file recording. While journaling, try to avoid unnecessary actions such as zooming/panning/rotating otherwise the journaling file will be filled with a lot of unnecessary code.

Now the recorded operations can be playback with the journal file that has been created and use it as starting point to customize the programming instructions to assist the turning process planning. The programming instructions customised need to enable the user for defining the process operations and with their cutting regions for selected region with process parameters decision making tool.

3.3.3 Model of Workpiece

In this project, the model will be as displayed in figure 3.8 below. This model will be required few operation features to fabricate for example facing operation, roughing turning, finishing turning and lastly grooving operation. Considering the development of user interface has to include these few operation features, this model is suitable as the model design for development of user interface.

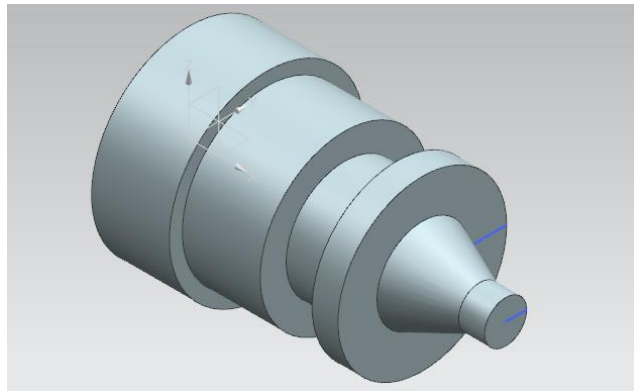


Figure 3.8: Model for NX software

3.4 CAM FILE GENERATION

The process of creating a single-spindle turning program consists of three parts which are setup, program and output. The workflow for setup tasks are to define conditions and parameters that are used commonly throughout the program, it involves analyse the part, choose the setup, define geometry, define collision zones and create tools. Analyse the part is the first task before setup which will give the idea of the overall size of the part. This allowed user to verify that the part was metric and to measure the overall dimensions so that user can determine appropriate cutter sizes and clearance distances. Figure 3.9 displays the part length measurement and it indicated the overall size of the blank.

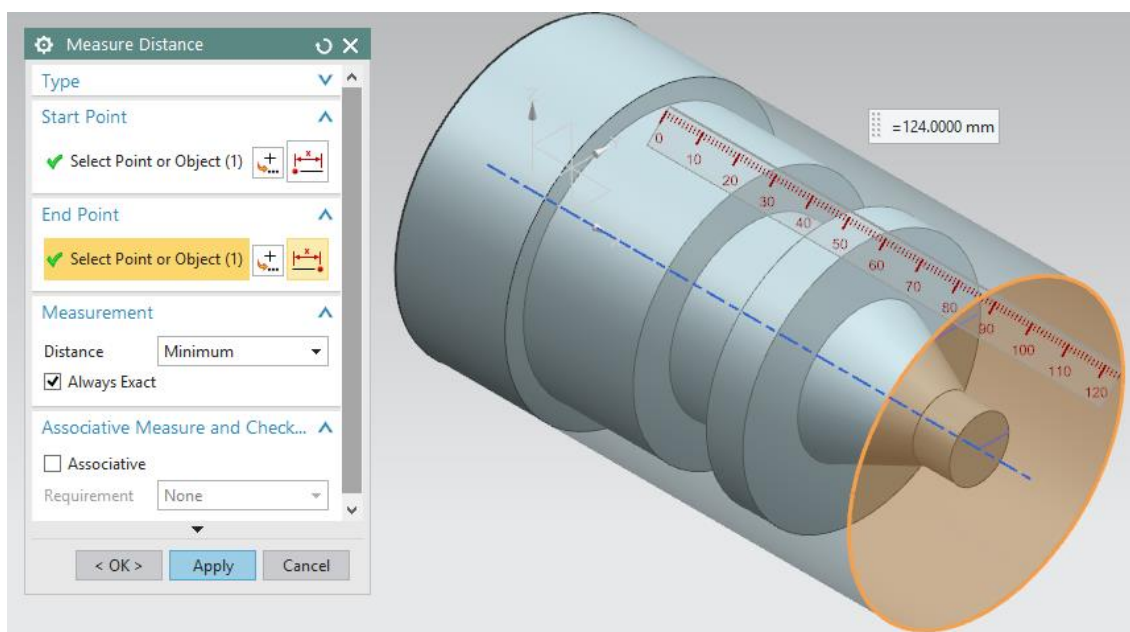


Figure 3.9: The measurement of part length

Choosing the setup is the second task that requires user to define a hierarchy of geometry groups including additional geometry and parameters that are used throughout the program. Defining the geometry is the task where user has to verify the coordinate system, define the blank and define part geometry for the part.

The Work Coordinate System (WCS) defines the coordinate system in which data is input, it should be oriented to the plane in which the cutter direction moving. The origin of the WCS should be on the centerline of rotation and XC direction should point to the right and YC direction should point up. Next, the Machine Coordinate System (MCS) defines the coordinate system in which the tool paths will be output. It represents program zero and should be oriented to the plane in which the cutter moves. In the MCS Spindle dialog box, be sure ZM-XM has been selected from the Specify Plane list. The MCS is oriented correctly for this part, the work plane and spindle centerline are also displayed in figure 3.10.

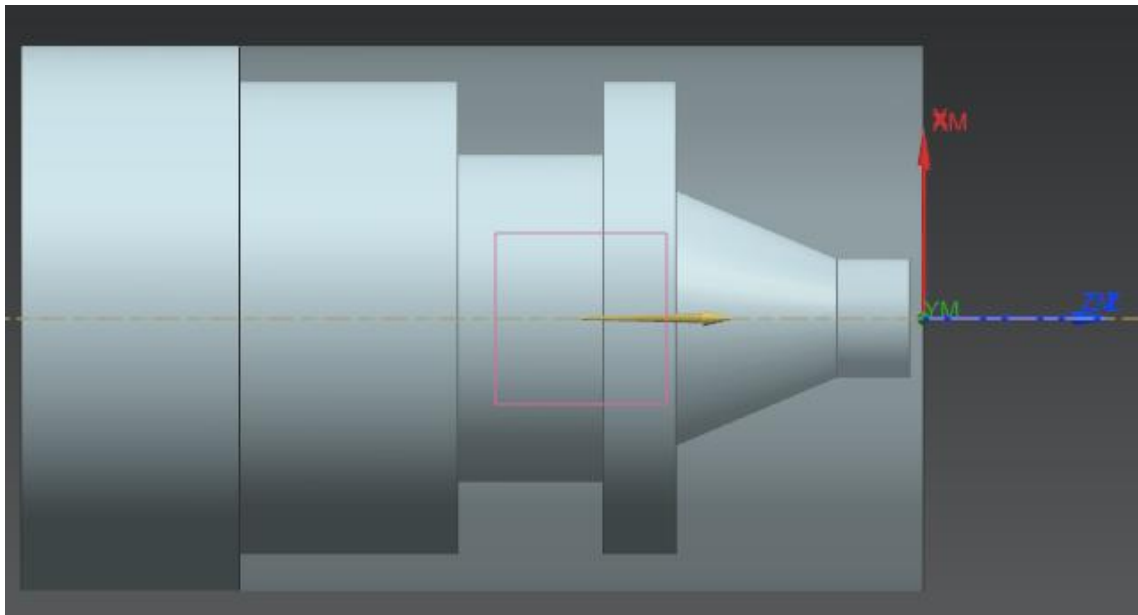


Figure 3.10: MCS, work plane and spindle centreline

Define part geometry is the task to specify the part, blank and part material. Figure 3.11 and figure 3.12 illustrate the part body and part blank selection. While for the part material selection, there are many pre-defined material in the database to be select. For the research, we only selected aluminium and steel as these materials are more commonly used in industry

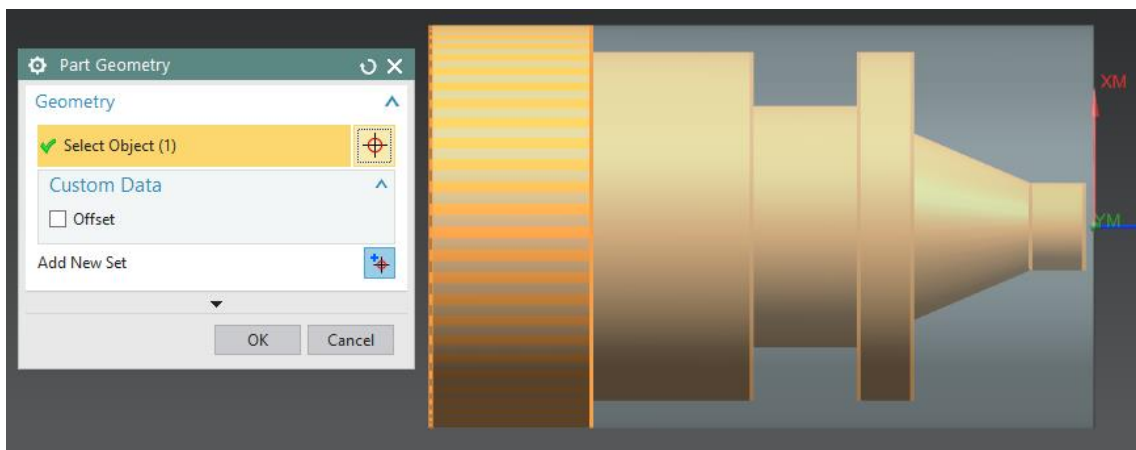


Figure 3.11: Part body selection

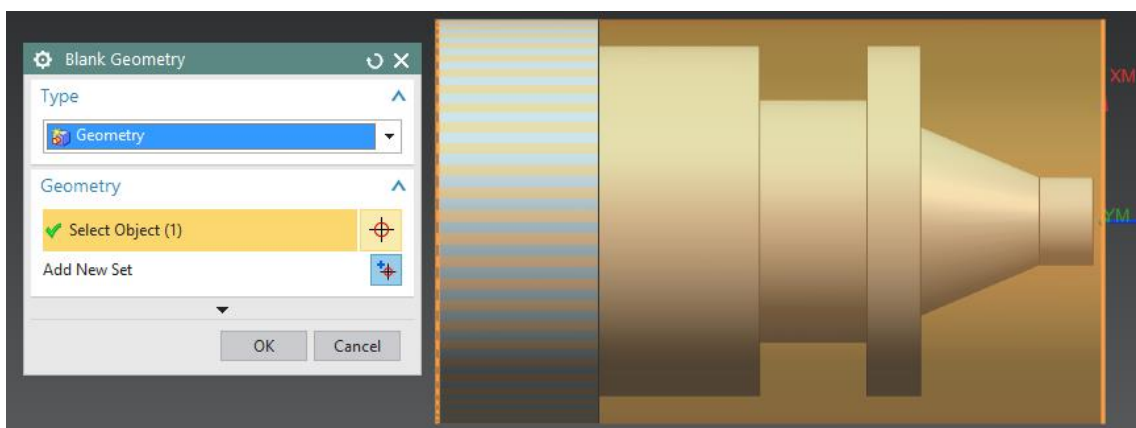


Figure 3.12: Part blank selection

Defining collision zones allow the cutter to avoid objects by using an axial containment plane (1), start and return points (2), and a clearance plane (3) as displays in figure 3.13. Start and return points define the start and return points that will be used to position the tool away from the part. Clearance plane define a clearance plane that can prevent the cutter from gouging the part when it enters and exits the inside diameter. Containment plane define a containment plane that can prevent tools from colliding

with the chuck jaws. The outside diameter roughing and finishing operations will need to use this containment group.

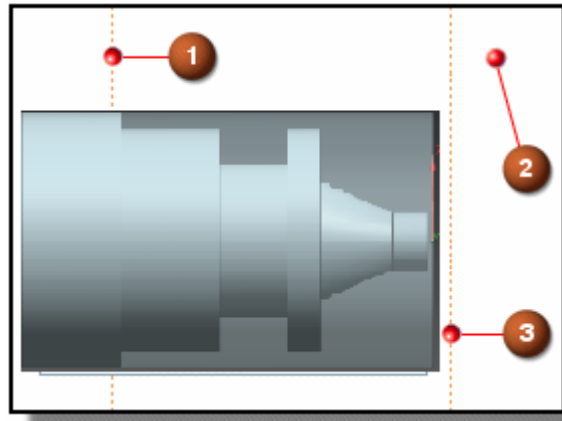


Figure 3.13: Defining collision zones

Figure 3.14 shows the avoidance setup for start and return points (2), and a clearance plane (3) as mentioned in figure 3.13. The step to define of the start and return points are to be used to position the tool away from the part. In the Geometry View of the Operation Navigator, double-click AVOIDANCE to edit the group. Then, in the Motion to Start Point (ST) section of the dialog box, select Direct from the Motion Type list and following with click a start point at the approximate screen position to confirm the coordinate point. While in the Motion to Return Point / Clearance (RT) section, select Direct from the Motion Type list. Next, in the Motion to Return Point / Clearance (RT) section, select Same as Start from the Point Option list. Expand the Radial Clearance Plane section of the dialog box and select Distance from the Axial Limit Option list, type the clearance value that larger than the blank length in the Axial ZM/XM box.



Avoidance	
Motion to Start Point (ST) ^	
Motion Type	Direct
Point Option	Point
Specify Point	<input checked="" type="checkbox"/>  
Approach (AP) v	
Motion to Start of Engage ^	
Motion Type	Direct
Departure (DP) v	
Motion to Return Point / Clearance (RT) ^	
Motion Type	Direct
Point Option	Same as Start
Motion to Gohome Point (GH) v	
Radial Clearance Plane ^	
Radial Limit Option	None
Axial Limit Option	Distance
Axial ZM/XM	134.0000

Figure 3.14: Avoidance setup

Figure 3.15 shows the containment geometry setup plane that can prevent tools from colliding with the chuck jaws. By choosing Home tab→Create Geometry, click CONTAINMENT and be sure that AVOIDANCE has been selected from the Geometry list. By specifying AVOIDANCE, the part and blank geometry defined in WORKPIECE and the parameters you specified in AVOIDANCE will be passed into the CONTAINMENT group. Next in the Axial Trim Plane 1 section of the dialog box,

select Distance from the Limit Option list and type negative value of distance between zero position and chuck jaws in the Axial ZM/XM

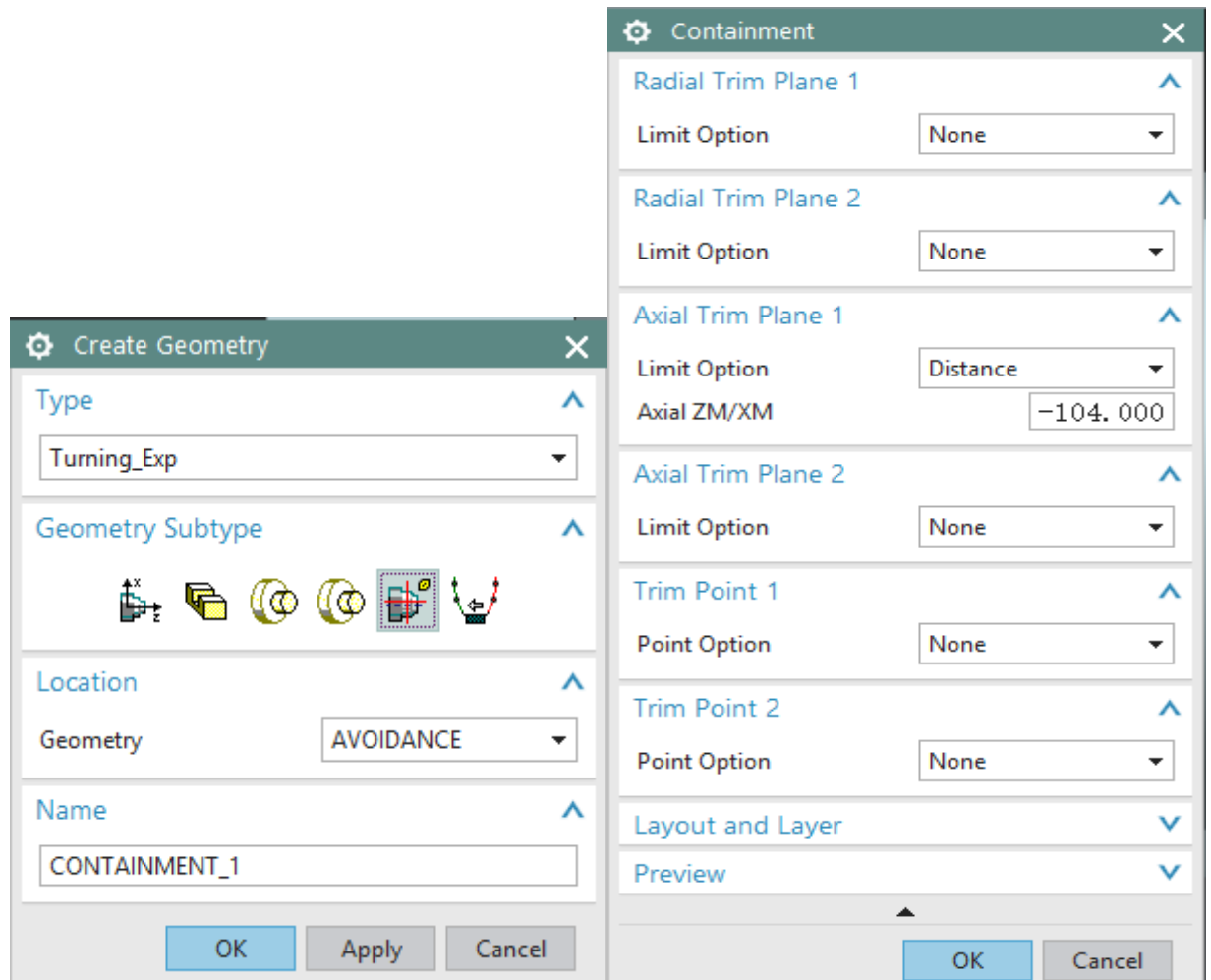


Figure 3.15: Containment setup

Meanwhile, creating tools allowed you to define cutting tools and assign them to appropriate stations. You can create tools during the setup or as you create operations. Once created, the tools are saved with the part and are available to be used when needed during program creation. Originally, the standard tools that are already predefined in the NX library are OD_55_L and OD_80_L which used in facing, roughing and finishing

operations. Therefore, user needs to define the other tools needed in machining operation for example in our model needs to use grooving tool.

Figure 3.16 shows the standard grooving tool OD_GROOVE_L that retrieved from the tool library that need to be used in the grooving operation. The dimension of the tool can be editing based on tool properties to increase the precision and accuracy.

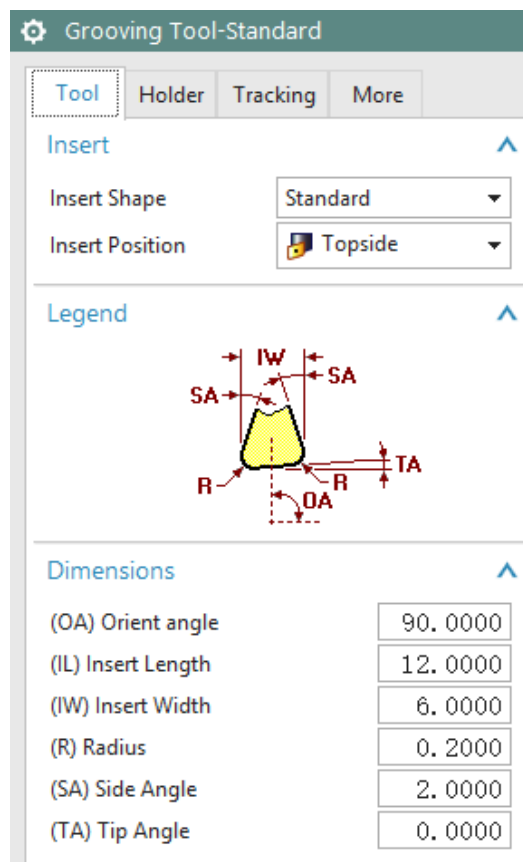


Figure 3.16: Standard grooving tool data

3.4.1 Facing Operation

The facing operation will be placed in the program, it will use the part and blank geometry that specified in `WORKPIECE` and the parameters defined in `AVOIDANCE`. It will use the `OD_80_L` tool defined by the setup and by using the `LATHE_FINISH` method removes all stock. These steps specify the operation type and groups that define essential information for facing operation.

In the Geometry section of the dialog box, click Edit next to Cut Regions. Then in the Axial Trim Plane 1 section of the dialog box, select Point from the Limit Option list. Select the end of the curve on the outside diameter of the part and click OK to accept the Cut Regions dialog box. Lastly click generate to complete the facing operation and generate the toolpath. Figure 3.17 shows the toolpath visualization for facing operation.

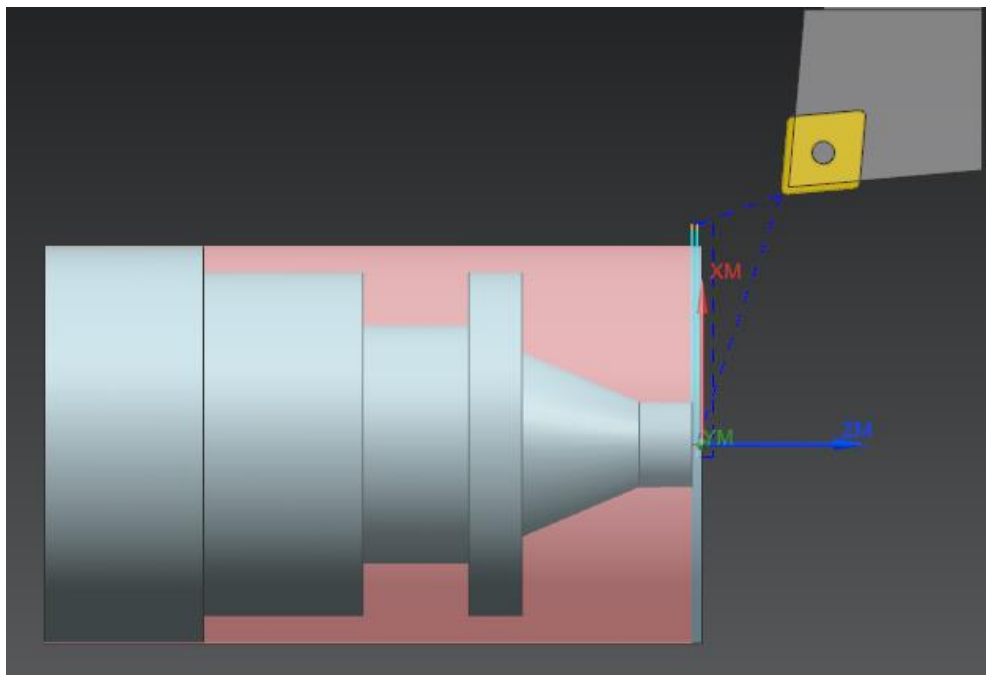


Figure 3.17: Facing operation toolpath

3.4.2 Roughing Operation

Roughing operation is to create an operation that roughs the outside diameter of the part. The operation uses the axial containment plane defined in CONTAINMENT to limit the cut region and avoid the chuck. Figure 3.18 displays the as level reversal mode for roughing operation that toolpath cuts down into the groove area. The red arrow on the groove indicates that the toolpath cuts down into the groove. This can be corrected by omitting the reversal mode, this will prevent the tool from descending into smaller diameter areas in the part. Figure 3.19 shows the reversal mode by omitting that prevent the toolpath cuts down into the groove.



Figure 3.18: As level reversal mode toolpath

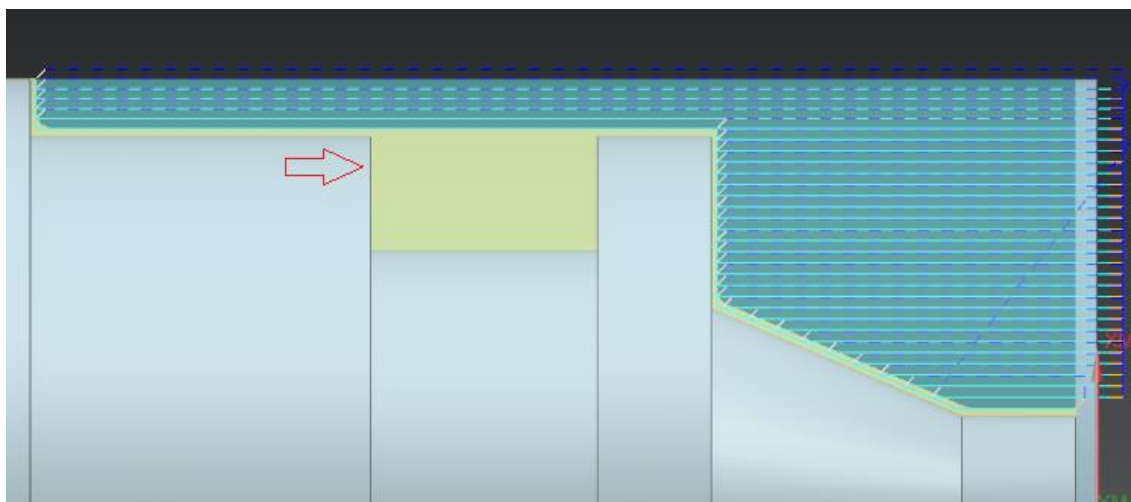


Figure 3.19: Omitting reversal mode toolpath

Then, click generate to complete the roughing operation and generate the toolpath. Figure 3.20 shows the toolpath visualization for roughing operation.

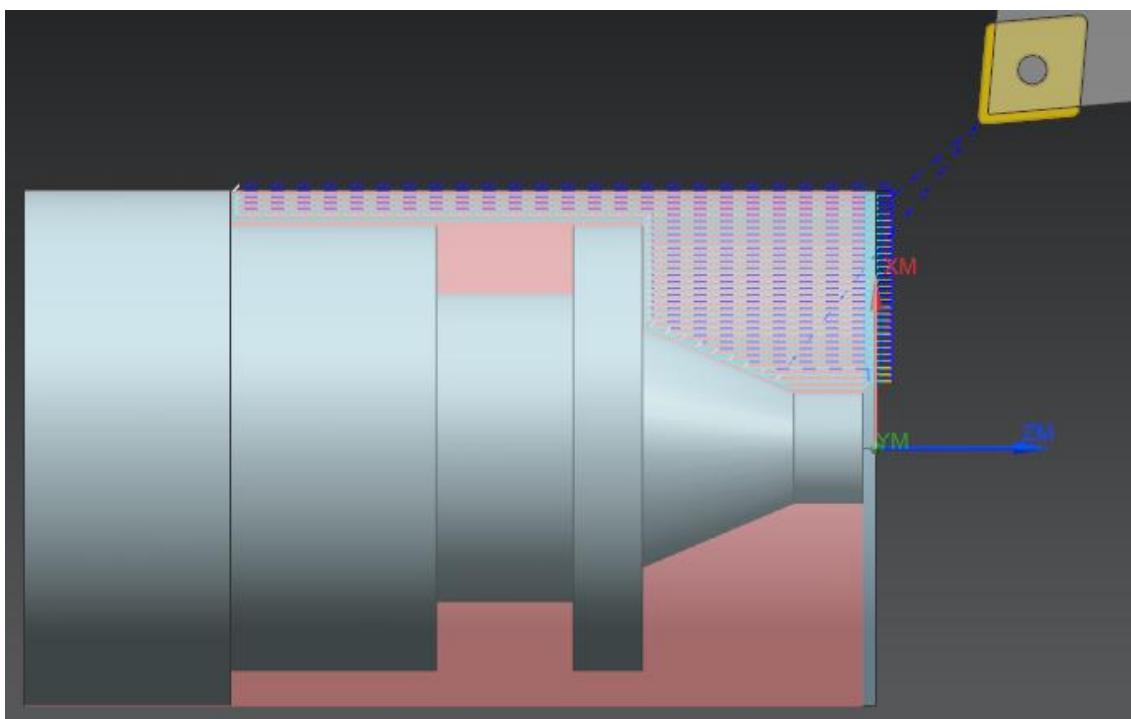


Figure 3.20: Roughing operation toolpath

3.4.3 Finishing Operation

Finishing operation is to create an operation that finishes the outside diameter of the part. In the Geometry section of the dialog box, click Edit next to Cut Regions. Then in the Trim Point 1 section of the dialog box, select Specify from the Point Option list and select the top of the vertical line at the end of the part. Next, in the Trim Point 2 section of the dialog box, select Specify from the Point Option list and select the top of the vertical line at the other end of the part. Following that, in the Region Selection section of the dialog box, select Multiple from the Region Machining list. Select Single Direction from the Region Sequence list. Click OK to complete the cut region. Lastly, click generate to complete the finishing operation and generate the toolpath. Figure 3.21 shows the toolpath visualization for finishing operation.

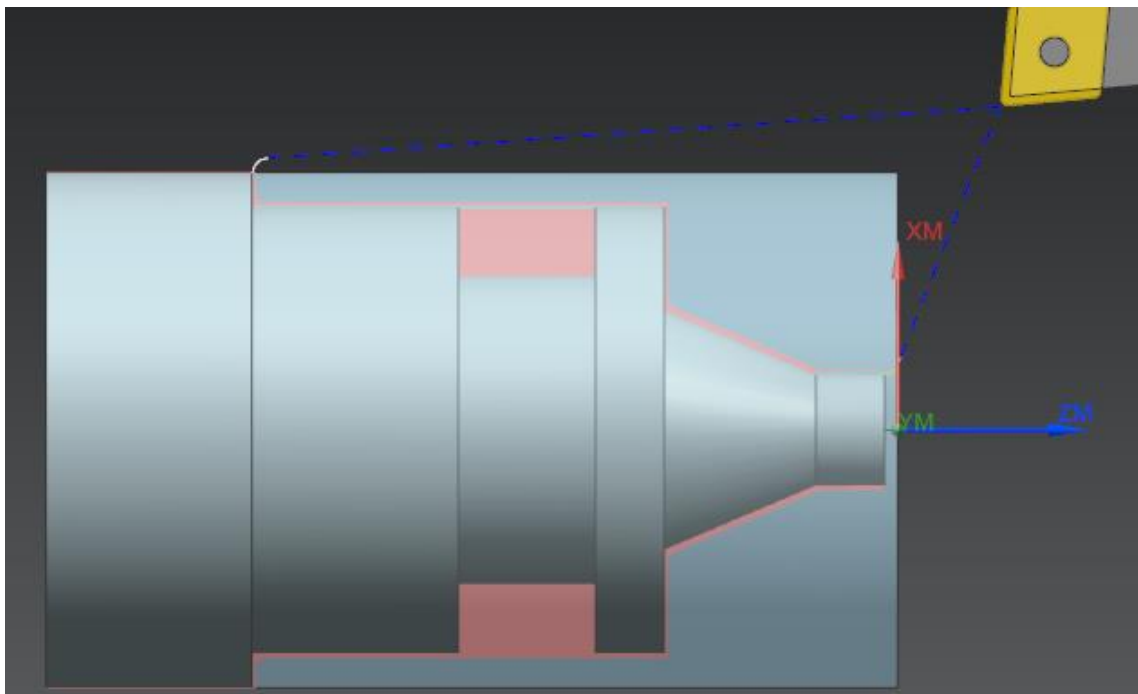


Figure 3.21: Finishing operation toolpath

3.4.4 Grooving Operation

Grooving operation requires to define two trim planes that constrain the cut region to the groove. In the Geometry section of the dialog box, click Edit next to Cut Regions. Then in the Axial Trim Plane 1 section of the dialog box, select Point from the Limit Option list and select the left end of the line defining the base of the groove. While in the Axial Trim Plane 2 section of the dialog box, select Point from the Limit Option list and select the right end of the line defining the base of the groove. Lastly, click OK to accept the Cut Regions dialog box and generate the toolpath. Figure 3.22 illustrates the cutting region selection constrain for grooving operation while figure 3.23 shows the toolpath visualization for grooving operation.

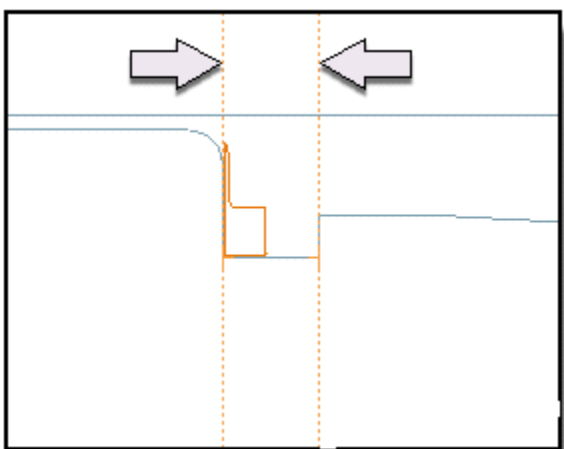


Figure 3.22: Grooving cutting region

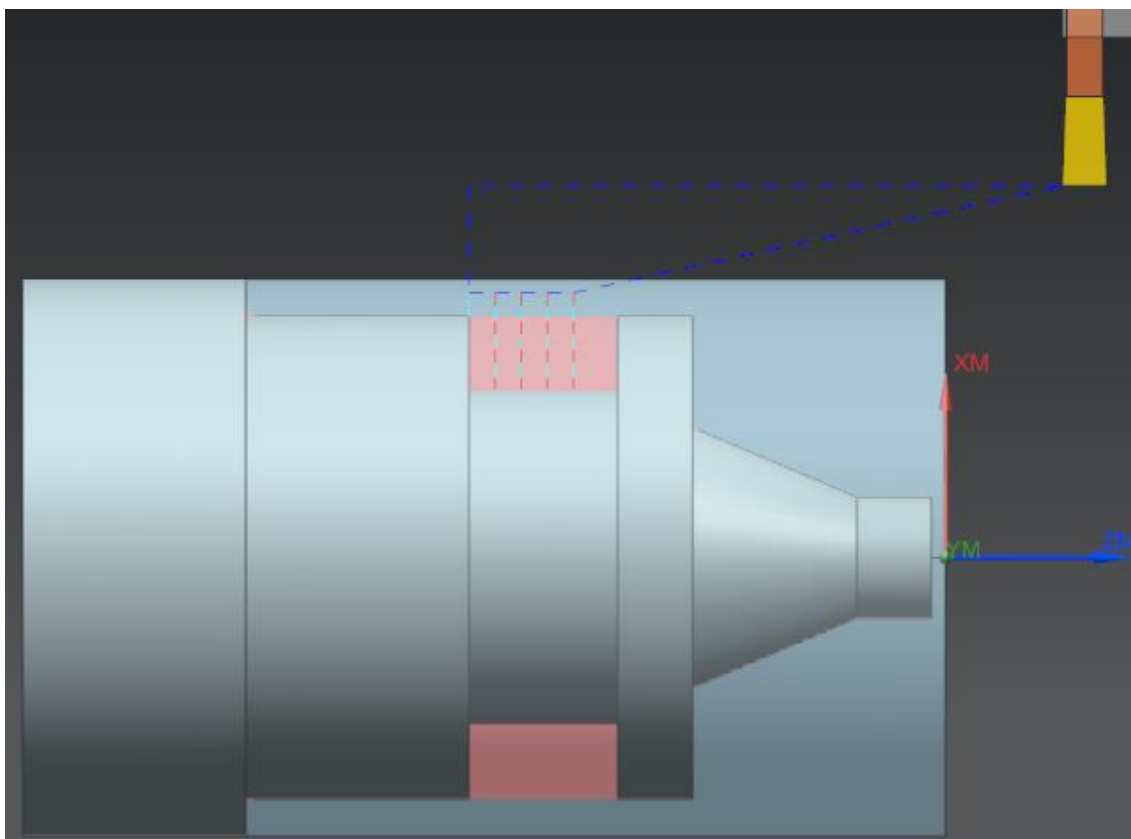


Figure 3.23: Grooving operation toolpath

3.5 VISUAL BASIC PROGRAMMING

The concept of basic programming is to strategize a set of commands into the computer system for performing certain tasks that are more efficient and much faster than human beings can do. From the CAM generation using NX software, the journaling tools able to record and translate all the instructions carried out into high level programming language which is visual basic program language.

The visual basic programming is ready in a text-only environment, window form format is inserts in order to show in a graphical environment. On the window forms, drag and drop the graphical object that wish to display and write program code for each

of the graphical object in order to carry out certain tasks in each of the events. These window forms can be linked together as one program and each form has their own graphical environment with its own program code that can be executed individually.

3.5.1 Programming Environment

The primary building block for visual basic is window form application as in figure 3.24. For the development of user interface, the label and the command button are the mainly tools used in the form application. The label tool is used as guides to the user and to provide instruction, while the command button is the most main controls used to execute commands.

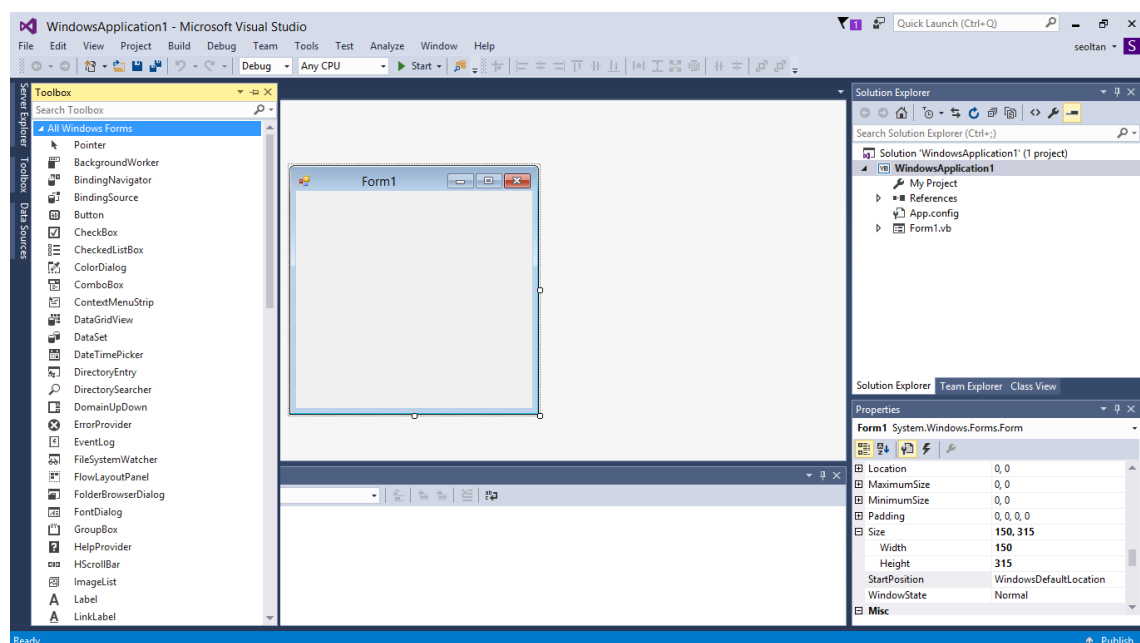


Figure 3.24: Visual Basic window form

Generally, most of the command button is associated with the Click event. When the user click the button, particular program code written in that command button will be executed. Figure 3.25 shows the command code that when button is click, current window form will close and then another window form will appear.

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click

    Dim frm1 As New Form2()

    frm1.Show()
    Me.Hide()

End Sub

Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click

    Dim frm2 As New Form3()

    frm2.Show()
    Me.Hide()

End Sub
```

Figure 3.25: Window form command code

The code can be explained by using a single apostrophe to serve as useful information about the input or output requirements of the program code. The compiler will ignore anything in the comment without have to delete the code so that can be undo by only delete the single apostrophe. Figure 3.26 shows the comment to indicate what process the code executes.

```

'-----
'   Menu: Insert->Sketch...
'-----
Dim markId1 As Session.UndoMarkId
markId1 = theSession.SetUndoMark(Session.MarkVisibility.Visible, "Start")

Dim nullNXOpen_Sketch As Sketch = Nothing

Dim sketchInPlaceBuilder1 As SketchInPlaceBuilder
sketchInPlaceBuilder1 = workPart.Sketches.CreateNewSketchInPlaceBuilder(nullNXOpen_Sketch)

Dim unit1 As Unit = CType(workPart.UnitCollection.FindObject("MilliMeter"), Unit)

Dim expression1 As Expression
expression1 = workPart.Expressions.CreateSystemExpressionWithUnits("0", unit1)

Dim expression2 As Expression
expression2 = workPart.Expressions.CreateSystemExpressionWithUnits("0", unit1)

```

Figure 3.26: Comment function

Next, the creation of undo marks, markId is to create undo marks that let any modifications made by the program to be undone if necessary. When the undo function is execute, the program will roll back before the undo mark was set. It is a good idea to roll back any errors to a known state.

The normal pattern to declare variables in visual basic is using Dim “variable name” as “variable type”, so the line “Dim unit1 As Unit” assigns block of memory for unit and named it as unit1. Then the unit declaration requires array of objects to work on therefore the code assign “FindObject” to assign the object and now the unit is set and specified.

The original journaling script file will end with error message if the journal trying to apply in different file as the code line is only looking for the object name that selected when it is recorded. This behaviour is referred as selection stickiness where the codes could not process with any other CAD models or components. The main reason to

remove the stickiness is to adapt the program to process different model and applicable to various parts.

The modification codes added need to be verified through multiple testing, thus it is time consuming and quite challenging. Even that the codes added is able to run, the correct result needs to be assured. Figure 3.27 shows the error warning window when the codes are not able to set specific object name as reference which represent the object. Therefore corrections have to be done for the added codes until it is able to execute completely.

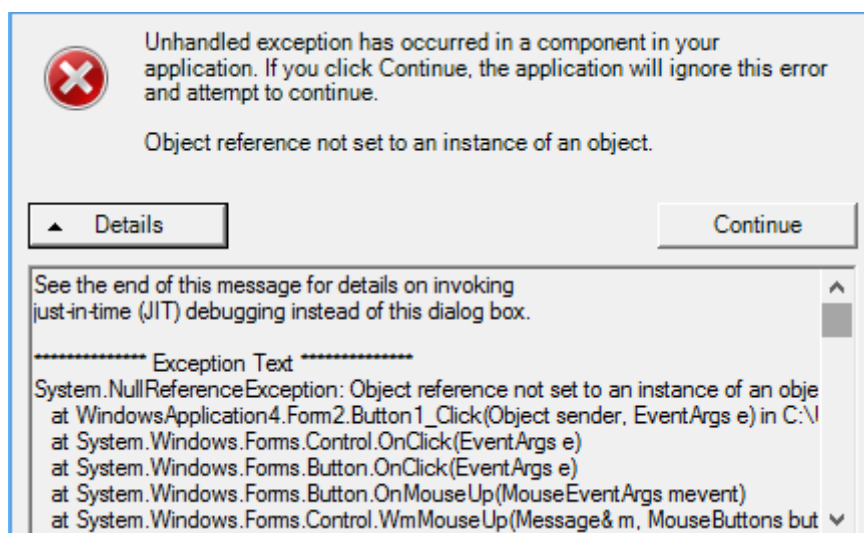


Figure 3.27: Error warning window

3.6 FABRICATION OF MODEL WORKPIECE

A lathe machine model Mazak Quick Turn Nexus 200 Turn Center is used to perform the workpiece fabrication by rotated against appropriate cutting tool for producing cylindrical shape by using the generated turning NC machining code. The Mazak Quick Turn Nexus 200 Turn Center offers high levels of precision machining and multiple means of programming and operating. This lathe machine also equipped with manual machining that added the capability which allows the machine to be operated manually without any type of programming. Figure 3.28 below shows the lathe machine that will be used for part fabrication. Then figure 3.29 displays the Mazak Quick Turn Nexus 200 Turn Center technical specifications.



Figure 3.28: Mazak Quick Turn Nexus 200 Turn Center

Source: Mazak Machine Catalogue

Specification		Bed Length - 500 in
Capacity	Maximum Swing	26.50 in / 695 mm
	Maximum Machining Diameter	14.750 in / 380 mm
	Maximum Bar Work Capacity	2.6 in / 65 mm
	Maximum Machining Length	22.701 in / 575 mm
Main Spindle	Chuck Size	8 in
	Maximum Speed	5000 rpm
	Motor Output (30 minute rating)	35.0 hp / 26 kw
Second Spindle	Chuck Size	6 in
	Maximum Speed	6000 rpm
	Motor Output (30 minute rating)	10.0 hp / 8 kw
Turret (Upper)	Number of Tools	12
	Maximum Speed	4500 rpm
	Motor Output (5 minute rating)	7.5 hp / 6 kw
Feed Axes	Travel (X Axis)	9.00 in / 230 mm
	Travel (Z Axis)	22.63 in / 575 mm
	Travel (W Axis)	23.00 in / 585 mm

Figure 3.29: Technical specifications of Mazak Quick Turn Nexus 200 machine

Source: Mazak Machine Catalogue

Aluminium round bar rods sized 60mm are used as a pre-shaped material for the part fabrication for part design. The round bar rod is clamped to the chuck of lathe machine and allowed to rotated at high speeds. The single point cutter is then feeds into the rotating workpiece and create the desired shape by removing materials in form of small chips. Figure 3.30 shows the material bar that used for part fabrication.



Figure 3.30: 60mm aluminium round bar rod

The aluminium bar stock is clamped and secured using the 3-jaw chuck as displayed in figure 3.31 below, then manually set the tool zero position by adjusting the tool cutter until it begin to touch the stock surface. Figure 3.32 shows the tools holder center of the machine where it can automatically change the tool used during the turning operation.

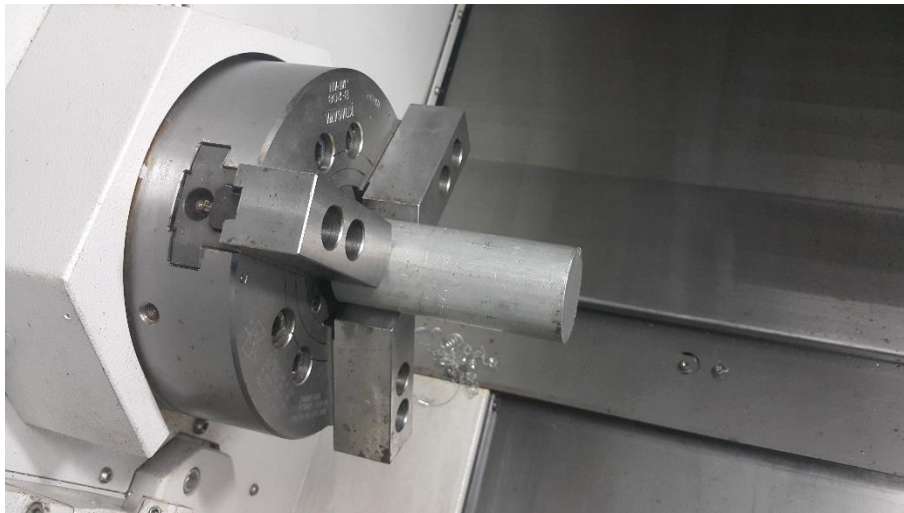


Figure 3.31: 3-jaw chuck clamping



Figure 3.32: Tools holder

Next, on the machine control panel set the approached tool position as the zero position as displayed in figure 3.33. The G-code program that generated from the NX software is loaded into the machine memory and executed the program code.



Figure 3.33: Control panel window

3.7 SUMMARY

This chapter has revealed the method starts by developing a part model which used in recording the programming code through journaling tools in NX software. Then, the part model is processed through from the starting of planning stage until completion of machining program. The procedures are clearly highlighted in the context where the CAD file is transferred to CAM file with operation features inserted. The journal script file generated is then modified using Visual Basic software as a platform to customise the programming code applicable to different models and universal. The part fabrication is used to validate the proposed user interface programming able to generate correct machining code and it is applicable.

CHAPTER 4

RESULTS ANALYSIS AND DISCUSSIONS

4.1 INTRODUCTION

In attempts to simplify the process planning tasks, the Graphical User Interface (GUI) is created to assist the NX software in the generation of machining program. For each command button developed as in figure 4.1, specific codes are incorporated within them so that the user interface can recall the NX software to execute particular instructions.

The graphical user interface (GUI) can be executed repetitively based on user input and as a result it could provide convenient procedures in generating complete machining program. Basically, the user interface will prompt user to key in essential input values for the required sections.



Figure 4.1: Graphical User Interface (GUI) that simplify process planning stage

4.2 PROCESS PLANNING IN CNC MACHINING

Despite the automation issues that having difficulty to remain competitive in manufacturing field by manufacture products in minimum time with good quality. Before the product can be fabricated, CNC machining needs extensive process planning in creating the machining program. Particularly in the planning stage, automation issue is still the main concern involved. Normally, CAM system is where the planning tasks are executed by transferred the 3D model of part product so that cutting toolpaths can be generated to achieve the desired surface finish and accuracy.

Prior to that, numerous cutting strategies and cutting parameters have to be defined and the skill of CAM operators are closely related to produce optimum machining operation program. Hence, implementing advanced programming tools to

automate the process planning tasks where the correct NC machining program can be generated without extensive tasks of manual process planning.

Figure 4.2 shows the typical steps of process flow to generate machining operation processes. The critical stage of the planning process in machining is represented by the red block. Several tasks such as the creating of operation feature, toolpath strategy and process parameter decisions making tend to slow down the entire planning process. With an appropriate application of user interface program, it can benefit to achieve rapid decision making in process planning stage.

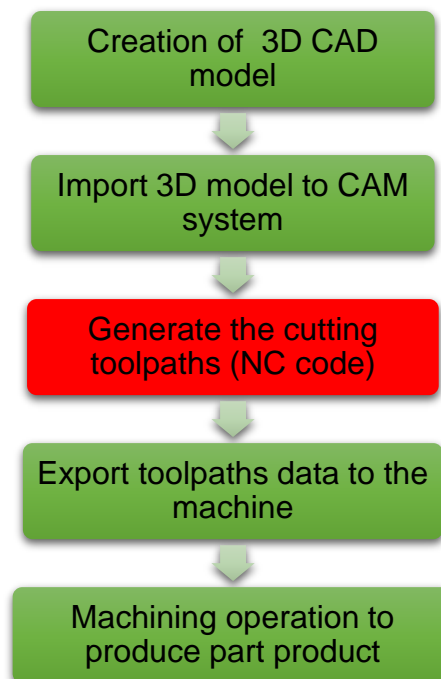


Figure 4.2: CNC machining process flow

With the customisation of NX journaling codes, an adequate automation level process planning have been adopted by simplify the planning. Therefore, time consuming activities can be reduced and minimize the need of manual intervention

during planning phase. Figure 4.3 displays the user interface created to implement the automation process planning tasks. For example, operation features such as facing, roughing, finishing and grooving are predetermined in the user interface created to automatically create the operation features with minimum user interaction.

On the other hand, the user interface are expected to support some flexibility during the planning phase. The user interface needs to be developed with the characteristics that will allow user to define the cutting regions for particular cutting operations. Moreover, the user interface also need to possess capability for customise various cutting parameters value such as depth of cut, feedrate and spindle speed for each operation feature inserted.

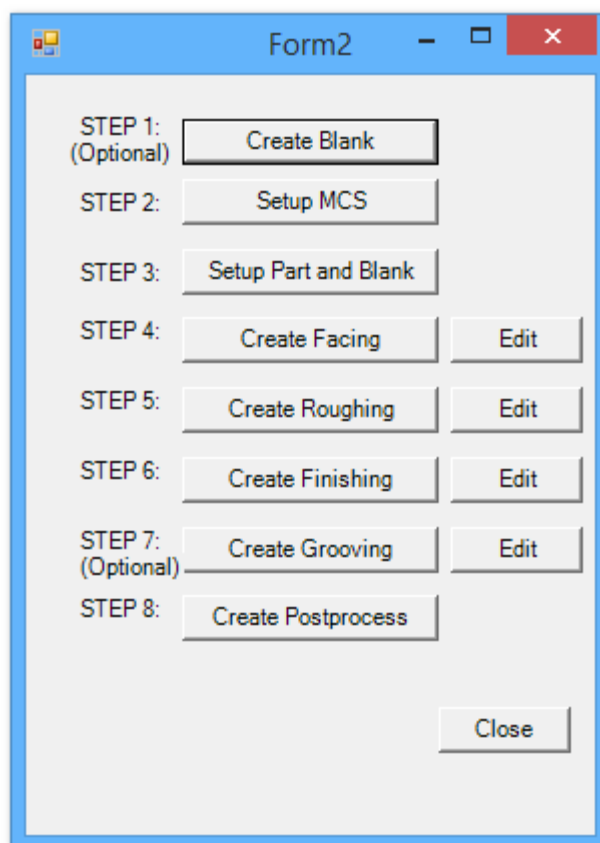


Figure 4.3: User interface to create the operation features

4.3 CUSTOMISATION OF RECORDED CODES

Within NX software, CAD file is transferred to CAM system and inserted appropriate operation features according to the CAD part design for instance operation facing, roughing, finishing and grooving which these operation features are regularly applying for standard turning operation. These step by step procedures are recorded through the journaling function and are converted to the programming language of Visual Basic. Since the original programming coding is only available to the fixed part model, the codes need to be reviewed and make modifications to generalize the programming coding so that it can be used for all type of part model.

4.3.1 Create Appropriate Blank Size

Generally, a CAD file only contains the model part to be fabricated. In order to generate machining program in CAM environment, a blank part which represent the raw material is needed. The user interface first step is to create an appropriate blank, in figure 4.4 shows the programming codes that automate the blank creation. It is programmed to create a blank with user input for the blank diameter and length, rather than manually sketch a circle and extrude it where it spend more time in searching the function and multiple steps involved.

Figure 4.5 illustrates the NX interface where the user interface programming is executed to create a blank, the prompt window request for blank diameter value input.

```

Dim expression3 As Double
expression3 = NXInputDialog.GetInputNumber("Set the blank diameter: ",
                                           "Blank Diameter",
                                           "")

theSession.SetUndoMarkVisibility(markId6, "Curve", Session.MarkVisibility.Visible)

Dim nXMatrix1 As NXMatrix
nXMatrix1 = theSession.ActiveSketch.Orientation

Dim center1 As Point3d = New Point3d(0.0, 0.0, 0.0)
Dim arc1 As Arc
arc1 = workPart.Curves.CreateArc(center1, nXMatrix1, expression3 / 2, 0.0, (360.0 * Math.PI / 180.0))

extrudeBuilder1.Limits.StartExtend.Value.RightHandSide = NXInputDialog.GetInputNumber("Set the blank length: ",
                                                                                       "Blank Length",
                                                                                       "")
length1 = extrudeBuilder1.Limits.StartExtend.Value.RightHandSide

extrudeBuilder1.Limits.EndExtend.Value.RightHandSide = "0"

extrudeBuilder1.BooleanOperation.Type = GeometricUtilities.BooleanOperation.BooleanType.Create

```

Figure 4.4: Blank size input programming

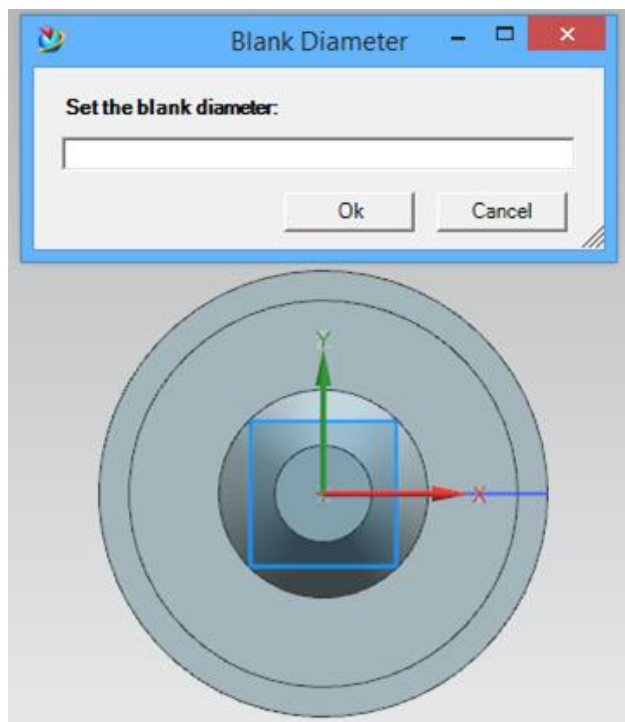


Figure 4.5: Blank diameter input

After the input of blank diameter, the program will following with another prompt window to request for blank diameter input value as in figure 4.6, the previous blank diameter input value has been allocated to form circle sketch.

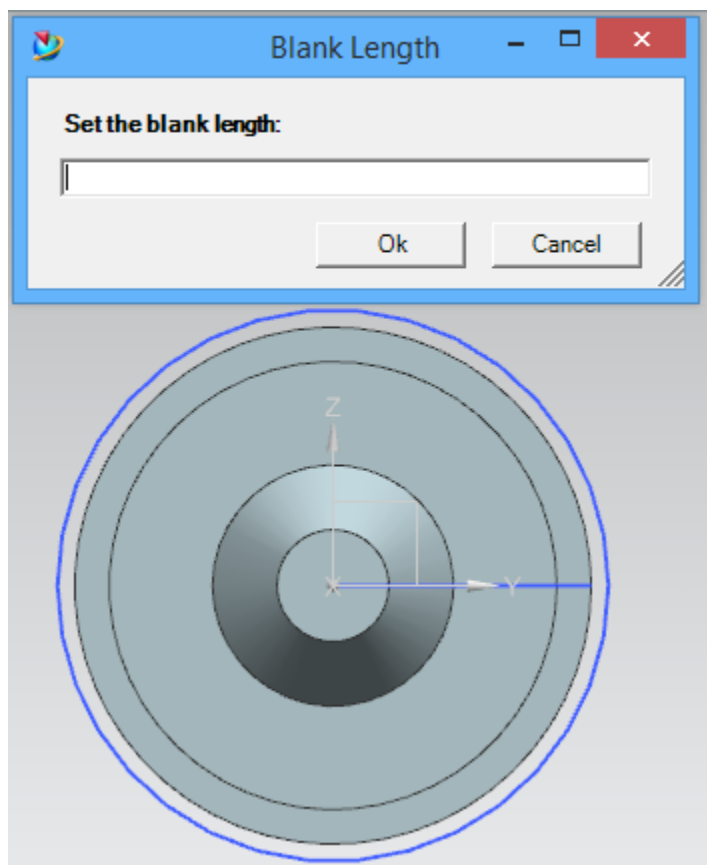


Figure 4.6: Blank length input

After the input value of blank length inserted, the NX interface will read the value and automatically generate the solid blank as in figure 4.7. Following that, prompt window will ask to choose the blank body to make it transparent so that it is easily to view the inner part model. Figure 4.8 displays the complete CAD model with blank that is ready to transfer to CAM session to insert machining operations.

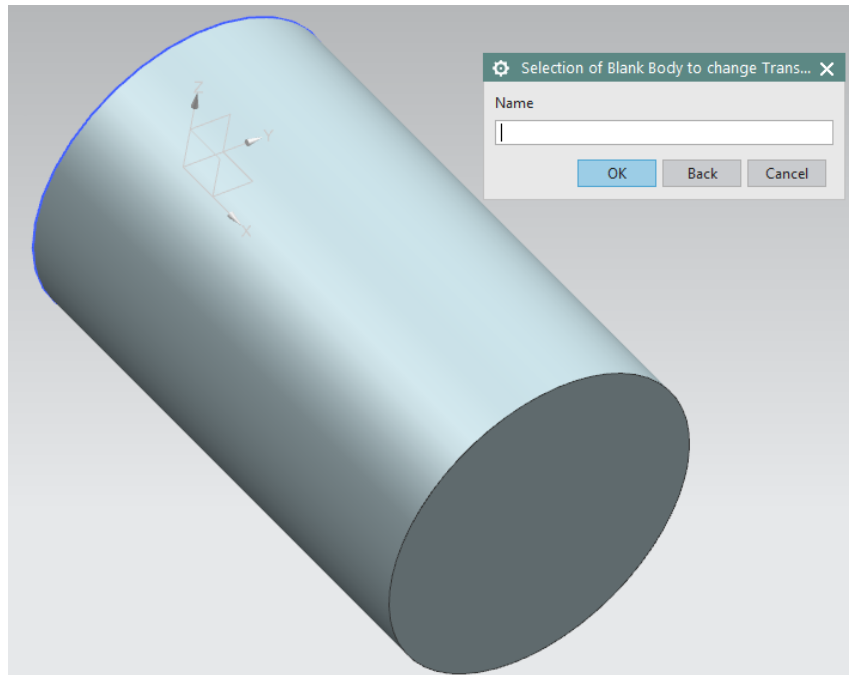


Figure 4.7: Translucent body selection

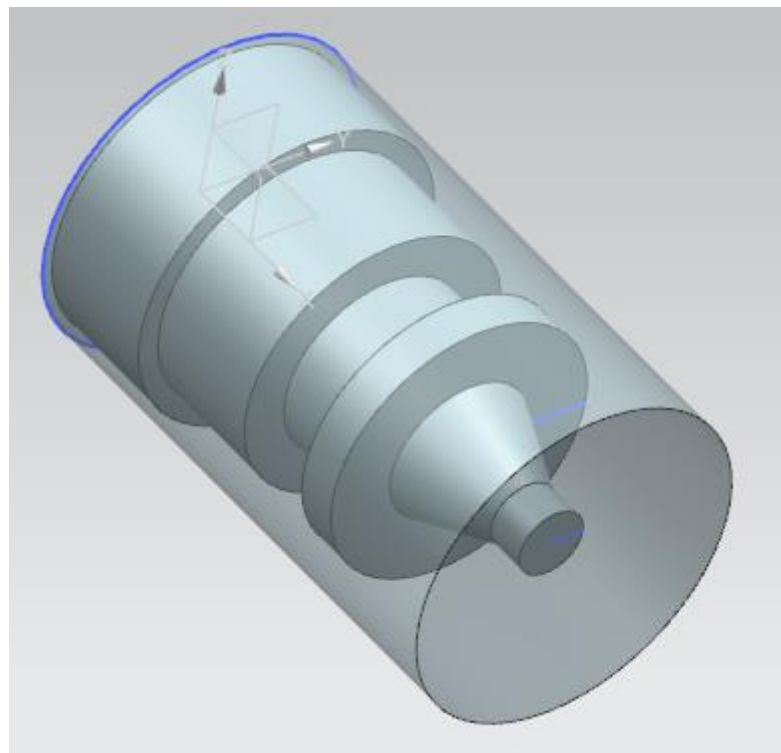


Figure 4.8: CAM model

4.3.2 Inheritance of Input Value

The inheritance of particular input value reduces the needed to insert same value again by derived from previous input as reference. In the user interface program, the blank length inserted will be inherited for the machine avoidance and containment limitation that prevent the cutting tool crashing with chuck. The programming code in figure 4.9 shows the input value of blank length will be defined as length1, then this value will be assigned to avoidance value as in figure 4.10 and containment value as in figure 4.11.

```
extrudeBuilder1.Limits.StartExtend.Value.RightHandSide = NXInputDialog.GetInputNumber("Set the blank length: ",
                                                "Blank Length",
                                                "")
length1 = extrudeBuilder1.Limits.StartExtend.Value.RightHandSide
```

Figure 4.9: Inheritance of blank length value

```
Dim expression27 As NXOpen.Expression
expression27 = workPart.Expressions.CreateSystemExpressionWithUnits("0", unit1)

turnAvoidanceReturnBuilder1.ReturnPointMode = NXOpen.CAM.TurnAvoidanceReturnBuilder.ReturnPointModes.SameAsStart

Dim turnAvoidanceClearanceBuilder1 As NXOpen.CAM.TurnAvoidanceClearanceBuilder
turnAvoidanceClearanceBuilder1 = turnAvoidanceGeomBuilder1.AvoidanceClearanceBuilder

turnAvoidanceClearanceBuilder1.AxialPlaneMode = NXOpen.CAM.TurnAvoidanceClearanceBuilder.AxialPlaneModes.Distance
' turnAvoidanceClearanceBuilder1.AxialPlaneValue = NXInputDialog.GetInputString("Set the Axial Avoidance clearance: ",
'                                             "Avoidance clearance",
'                                             "")
turnAvoidanceClearanceBuilder1.AxialPlaneValue = length1 + 10

Dim markId29 As NXOpen.Session.UndoMarkId
markId29 = theSession.SetUndoMark(NXOpen.Session.MarkVisibility.Invisible, "Avoidance")
```

Figure 4.10: Avoidance value inherited

```

theSession.SetUndoMarkName(markId32, "Containment Dialog")
'-----
' Dialog Begin Containment
'-----
turnFeatureGeomBuilder1.AxialTrimPlane1Builder.LimitOption = NXOpen.CAM.TrimPlane.LimitOptions.Distance
'turnFeatureGeomBuilder1.AxialTrimPlane1Builder.Radius.Value = NXInputDialog.GetInputString("Set the Axial Chuck Avoidance: ",
'
'                                     "Chuck Avoidance",
'                                     "-")
turnFeatureGeomBuilder1.AxialTrimPlane1Builder.Radius.Value = -length1 + 20
Dim markId33 As NXOpen.Session.UndoMarkId
markId33 = theSession.SetUndoMark(NXOpen.Session.MarkVisibility.Invisible, "Containment")
theSession.DeleteUndoMark(markId33, Nothing)
Dim markId34 As NXOpen.Session.UndoMarkId
markId34 = theSession.SetUndoMark(NXOpen.Session.MarkVisibility.Invisible, "Containment")

```

Figure 4.11: Containment value inherited

The correlation between blank length value input will always affected the avoidance and containment limitation value. With this programming customisation, the input value for one specific task able to inherit for other tasks where it will minimum the required prompt input from user, thus the process planning performance greatly improved.

4.3.3 Cutting Region Selection

A part of the original code as shown in figure 4.12 is the programming code where the cutting region selection of part performed for fixed part model. In attempts to generalize the programming code so that it can be used for all type of part model, the code where highlighted in red box is replaced with the function code as in green box in figure 4.13. By providing the new selection interface in the graphical user interface (GUI), the part cutting region can be decided according to user decision based on different CAD part file.

```

'-----
' Dialog Begin Part Geometry
'-----
'Dim part1 As NXOpen.Part = CType(theSession.Parts.FindObject("model 2"), NXOpen.Part)

' Dim partLoadStatus1 As NXOpen.PartLoadStatus
'partLoadStatus1 = part1.LoadThisPartFully()

' partLoadStatus1.Dispose()
Dim bodies1(0) As NXOpen.Body

'Dim component1 As NXOpen.Assemblies.Component = CType(workPart.ComponentAssembly.RootComponent.FindObject("COMPONENT model 2"), NXOpen.Assemblies.Component)

'Dim body1 As NXOpen.Body = CType(component1.FindObject("PROTO#.Bodies|REVOLVED(2)"), NXOpen.Body)

Dim body1 As NXOpen.Body = SelectBody1()
If body1 Is Nothing Then
    Return
End If

bodies1(0) = body1
Dim bodyDumbRule1 As BodyDumbRule
bodyDumbRule1 = workPart.ScRuleFactory.CreateRuleBodyDumb(bodies1)

```

Figure 4.12: Part cutting region selection task in programming language

```

Public Function SelectBody1() As Body

    Dim ui As UI = UI.GetUI
    Dim message As String = "Select Part Body"
    Dim title As String = "Selection of Part Body"

    Dim scope As Selection.SelectionScope = Selection.SelectionScope.AnyInAssembly
    Dim keepHighlighted As Boolean = False
    Dim includeFeatures As Boolean = False

    Dim selectionAction As Selection.SelectionAction =
        Selection.SelectionAction.ClearAndEnableSpecific

    Dim selectionMask_array(1) As Selection.MaskTriple
    With selectionMask_array(0)
        .Type = UFConstants.UF_solid_type
        .Subtype = 0
        .SolidBodySubtype = 0
    End With

    Dim selectedObject As TaggedObject = Nothing
    Dim cursor As Point3d

    ui.SelectionManager.SelectTaggedObject(message, title, scope,
        selectionAction, includeFeatures,
        keepHighlighted, selectionMask_array,
        selectedObject, cursor)

    Dim body1 As NXOpen.Body = CType(selectedObject, NXOpen.Body)

    If body1 Is Nothing Then
        Return Nothing
    End If

    Return body1

End Function

```

Figure 4.13: Function code added to prompt new selection

With one command button click require from the user interface, the particular operation feature selected will be executed. Next, prompt window of cutting region selection will appear and request to select the necessary cutting region for that operation feature. Figure 4.14 illustrates the selection of cutting region where the facing operation is executed. By selecting the cutting edge for facing operation approach to the exact location, then the machining program for facing operation is done.

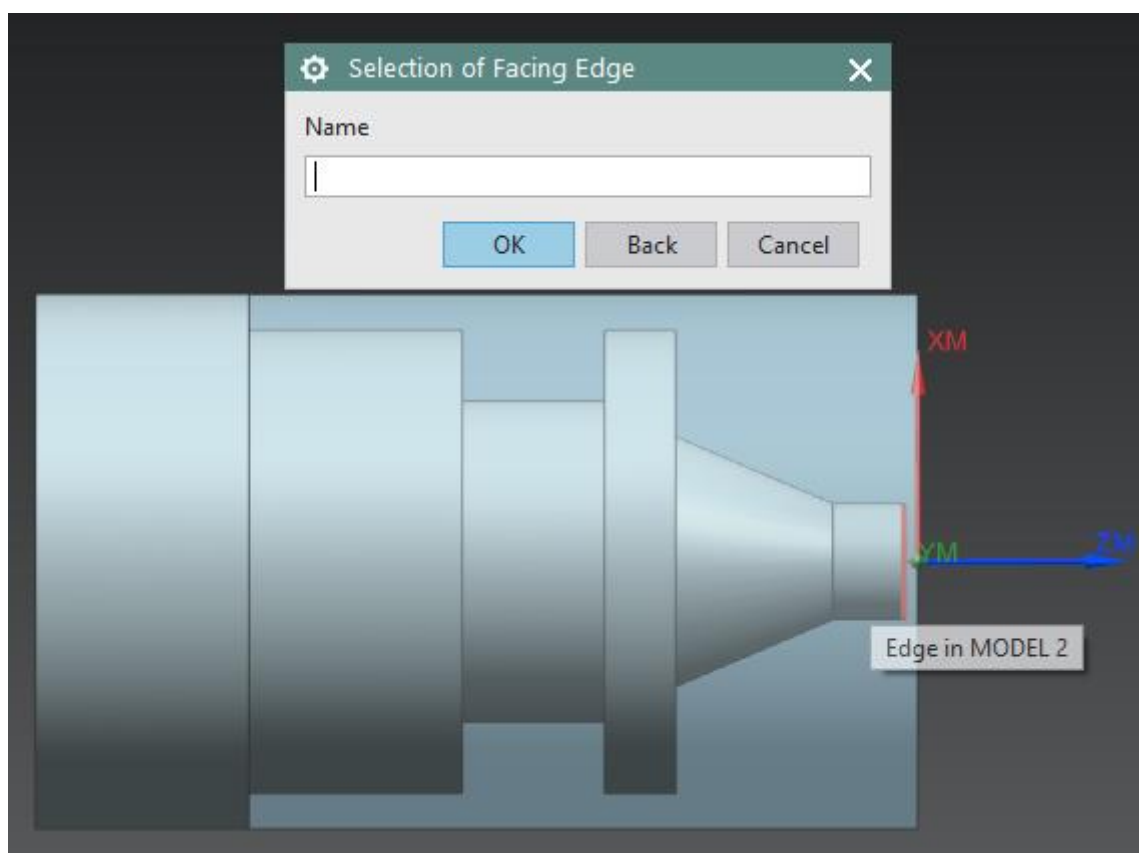


Figure 4.14: Cutting region prompt selection

4.3.4 Materials Selection

Another part modification is carried out by configuring the cutting parameters. The graphical user interface setup will allow user to setup cutting parameters based on part materials either in aluminium or steel as shown in figure 4.15. Therefore, the cutting parameters value used during machining are depend on the type of materials selected.

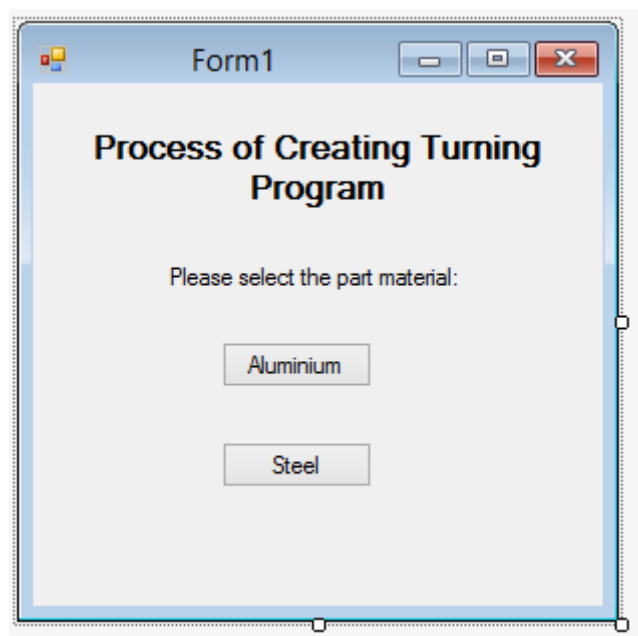


Figure 4.15: Materials selection interface

The cutting parameters comparison between aluminium and steel are as in table 4.1. The depth of cut mainly is 1.0mm for all operation features, while the feedrate and surface speed of aluminium are higher than steel. This is because aluminium is a soft metal if compared with steel, therefore when the machine remove material through steel material it need slow cutting rate to avoid cutting tool breakage.

Table 4.1: Cutting parameters for aluminium and steel

Operation	Parameters	Aluminium	Steel
Facing	Depth of Cut (mm)	1.0	1.0
	Feedrate (mmpr)	0.254	0.1778
	Surface Speed (SMM)	608.0	114.0
Roughing	Depth of Cut (mm)	1.0	1.0
	Feedrate (mmpr)	0.508	0.381
	Surface Speed (SMM)	547.2	83.6
Finishing	Depth of Cut (mm)	1.0	1.0
	Feedrate (mmpr)	0.254	0.1778
	Surface Speed (SMM)	608.0	114.0
Grooving	Depth of Cut (mm)	1.0	1.0
	Feedrate (mmpr)	0.254	0.1778
	Surface Speed (SMM)	608.0	114.0

The visual basic programming code that clearly stated the values used to setup cutting parameters for aluminium and steel are as figure 4.16 and figure 4.17. With the programming code, the cutting parameters can be directly assigned into the CAM system and automate the cutting parameters decision making.

```

' -----
'   Dialog Begin Feeds and Speeds
' -----
roughTurningBuilder1.FeedsBuilder.FeedCutBuilder.Value = 0.254

roughTurningBuilder1.FeedsBuilder.FeedCutBuilder.InheritanceStatus = False

roughTurningBuilder1.FeedsBuilder.FeedEngageBuilder.Value = 90.0

roughTurningBuilder1.FeedsBuilder.FeedEngageBuilder.InheritanceStatus = False

roughTurningBuilder1.FeedsBuilder.FeedRetractBuilder.Unit = NXOpen.CAM.FeedRateUnit.Rapid

roughTurningBuilder1.FeedsBuilder.FeedRetractBuilder.InheritanceStatus = False

roughTurningBuilder1.FeedsBuilder.FeedRetractBuilder.Value = 0.0

roughTurningBuilder1.FeedsBuilder.FeedFirstCutBuilder.Unit = NXOpen.CAM.FeedRateUnit.CutPercent

roughTurningBuilder1.FeedsBuilder.FeedFirstCutBuilder.InheritanceStatus = False

roughTurningBuilder1.FeedsBuilder.FeedFirstCutBuilder.Value = 60.0

roughTurningBuilder1.FeedsBuilder.FeedStepoverBuilder.Unit = NXOpen.CAM.FeedRateUnit.CutPercent

roughTurningBuilder1.FeedsBuilder.FeedStepoverBuilder.InheritanceStatus = False

roughTurningBuilder1.FeedsBuilder.FeedStepoverBuilder.Value = 60.0

roughTurningBuilder1.FeedsBuilder.SurfaceSpeedBuilder.Value = 608.0

```

Figure 4.16: Cutting parameter for aluminium

```

' -----
'   Dialog Begin Feeds and Speeds
' -----
roughTurningBuilder1.FeedsBuilder.FeedCutBuilder.Value = 0.1778

roughTurningBuilder1.FeedsBuilder.FeedCutBuilder.InheritanceStatus = False

roughTurningBuilder1.FeedsBuilder.FeedEngageBuilder.Value = 90.0

roughTurningBuilder1.FeedsBuilder.FeedEngageBuilder.InheritanceStatus = False

roughTurningBuilder1.FeedsBuilder.FeedRetractBuilder.Unit = NXOpen.CAM.FeedRateUnit.Rapid

roughTurningBuilder1.FeedsBuilder.FeedRetractBuilder.InheritanceStatus = False

roughTurningBuilder1.FeedsBuilder.FeedRetractBuilder.Value = 0.0

roughTurningBuilder1.FeedsBuilder.FeedFirstCutBuilder.Unit = NXOpen.CAM.FeedRateUnit.CutPercent

roughTurningBuilder1.FeedsBuilder.FeedFirstCutBuilder.InheritanceStatus = False

roughTurningBuilder1.FeedsBuilder.FeedFirstCutBuilder.Value = 60.0

roughTurningBuilder1.FeedsBuilder.FeedStepoverBuilder.Unit = NXOpen.CAM.FeedRateUnit.CutPercent

roughTurningBuilder1.FeedsBuilder.FeedStepoverBuilder.InheritanceStatus = False

roughTurningBuilder1.FeedsBuilder.FeedStepoverBuilder.Value = 60.0

roughTurningBuilder1.FeedsBuilder.SurfaceSpeedBuilder.Value = 114.0

```

Figure 4.17: Cutting parameters for steel

As the graphical user interface requirement is to minimize the planning tasks, the optimum cutting parameters are based on the pre-defined library data for aluminium and steel material rather than searching for optimum cutting parameters. Although searching for optimum parameters is not recommended in this process, but still the graphical user interface (GUI) created allow advance setting for cutting parameters value for various cutting operations. Figure 4.18 presented the advance user interface to modify the cutting parameters if the user prefer to determine calculated optimum cutting parameters.

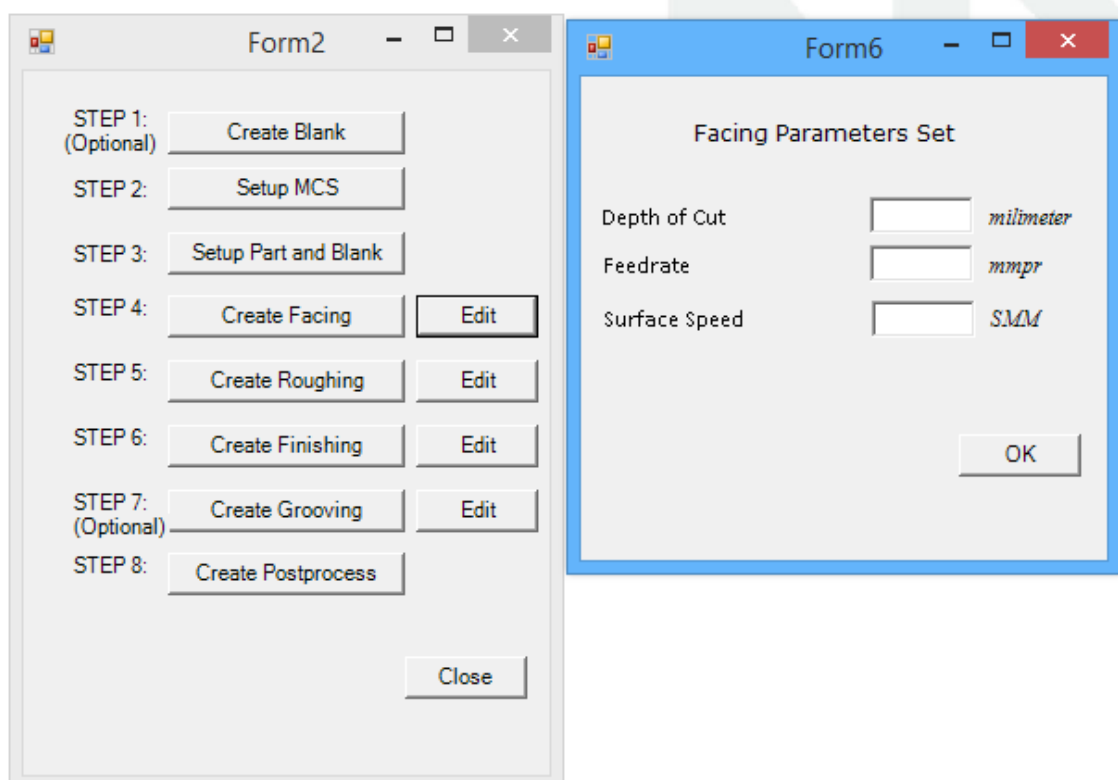


Figure 4.18: Advance user interface

Despite manually to insert the various cutting operations and cutting parameters, the graphical user interface created is selected as a platform to perform automation level

of process planning tasks which is equipped with foundation of programming and adaptation of customisation tools.

4.4 PROCESS PLANNING SIMULATION TIME DURATION

Simulation analyses are performed for both process planning tasks where the machining operations constructed manually and through implementing customised programming user interface. The estimated time duration to complete the whole process from CAD file till generated the machining G-code requires to operate the machine are recorded.

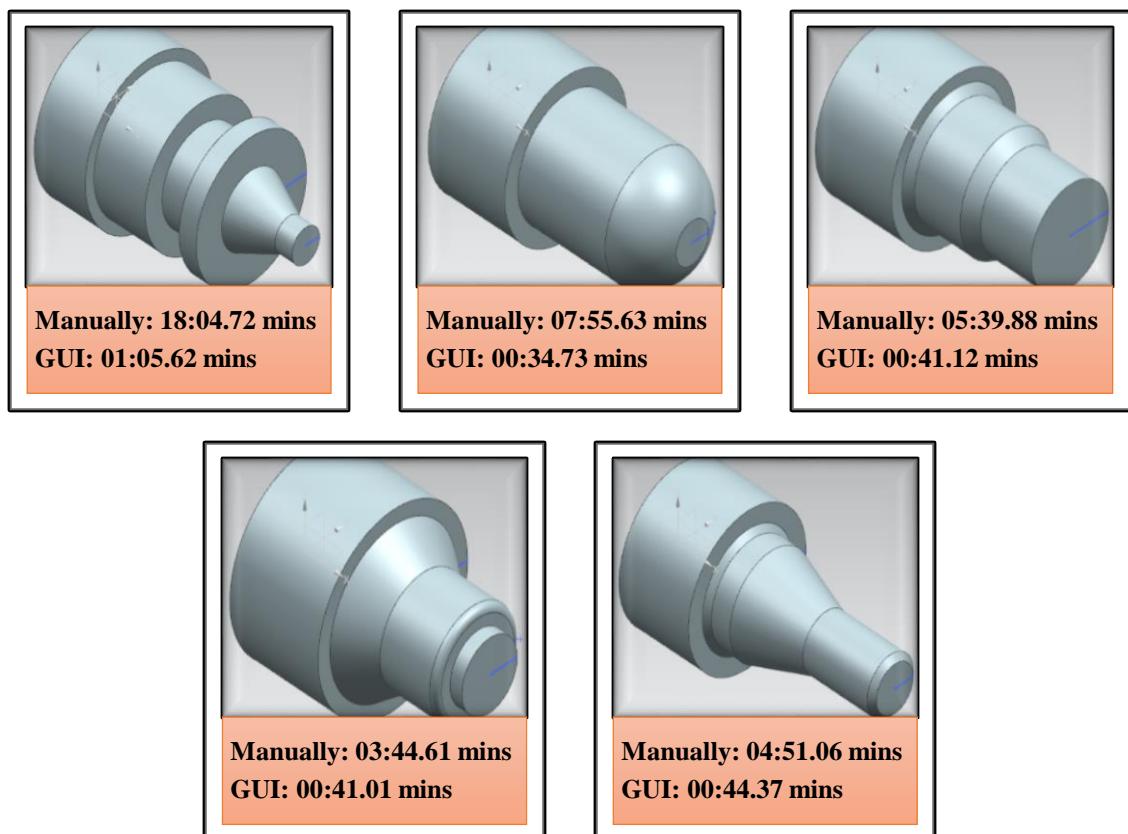


Figure 4.19: Time spend comparison for process planning

The simulation analyses that has been performed to few models are as shown in figure 4.19. Based on the simulation results, the time used for process planning using user interface is minimized when compared to process planning solely dependent on manual operation. The first model took 18 minutes 4 seconds to generate the machining program manually whereas using user interface managed to reduce the time to 1 minute 5 seconds. While for the other models, they took around 4 to 7 minutes to manually generate the machining program whereas using user interface managed to reduce the time to around 30 to 45 seconds.

The longer time spent in manually inserts the operations is due to having difficulties in finding the function button, the cutting region selection, insert cutting parameters and toolpath verification. With the application of user interface, automated machining processes tend to minimize the time spent to rapid generate correct operation features without extensive procedures.

The user interface with the customised programs managed to work effectively in develop process planning tasks and the generation of machining program code. It can be seen that repetitive work able to perform using the user interface for different CAD model to automate the process planning tasks and decision making criteria.

4.5 PROGRAM VERIFICATION

In order to validate the user interface created able to assist process planning tasks, few CAD models are used to undergo machining experiment with the intention of verifying the machining program generated using user interface can be implemented. Machining experiments were carried out and the results indicate the suggested approach

can be machined. Figure 4.20 illustrates the parts machined by using the machine program generated through the application of user interface developed.



Figure 4.20: Machined parts

The machining experiments have validated the developed user interface is now possible to assist in process planning tasks and fulfils the main objective in simplify the process planning tasks by allowing automated process planning to work in NX software.

4.6 SUMMARY

This chapter has described the developed user interface to be executed within NX software interface and generate machining operations effectively. The main objective of the user interface program is to assist the process planning tasks for

construct machining program with minimum time and minimalize the need of skilled user in the process. The user interface developed has shown that the machining operations can be controlled using the programming language code and performed effectively with few inputs required from user. The generation of machining programs have been successfully validated by fabricating few of the CAD model. However, current user interface development is not fully designed to cover all the machining operations involved in the planning stage. Therefore, considering the NX software flexibility, there might some possibility some of the machining operations are not able to connect between each other. More development is needed to fully integrate the user interface into the NX software to manage process planning tasks effectively.

CHAPTER 5

CONCLUSIONS AND RECOMMENDATIONS

5.1 INTRODUCTION

This chapter concluded the potential of advanced computer aided manufacturing application for turning processes. The research is conducted to enhance the capabilities of turning machining in process planning tasks through the programming methods in assisting the construction of machining program. The problems stated as the beginning of the thesis have been formulated with integration of user interface to assist the turning processes.

5.2 CONCLUSIONS

As a conclusion, the developments proposed in this research manage to minimize the time spend for process planning in turning operations by implementing programming instructions in NX CAM systems through the user interface program controller created. The details of the implementation have been discussed and the simulation analyses were performed, the results are compared based on time consumed

during process planning tasks to verify the application of graphical user interface programming controller could simplify the process planning tasks.

Secondly, the graphical user interface is able to improve the process planning in turning operations by integrating decision making tools and optimize process parameters such as spindle speed, depth of cut and cutting feed rate based on operation features with minimum human intervention and embed substantial levels of automation in the planning stage. However, optimization of process parameters need to be identified and improved through real machining performance analyses, then the process parameters can then be modified and customised in the programming language to further improving the processes outcome.

5.3 RECOMMENDATIONS

Current user interface development is not fully designed to cover all the machining operations involved in the planning stage. Therefore, considering the NX software flexibility, more developments can proceed to fully integrate the user interface into the NX software to manage process planning tasks completely. Further development can be extended by expanding the user interface system abilities to execute more kinds of operation feature for instance spot drill, drill, internal boring, internal grooving and threading. Moreover, integrating tools selection and selection of different toolpath strategies criteria might can be considered to further enhance the capabilities of the user interface program controller.

Various materials and cutting parameters is also necessary to make the system adaptable to the other materials selection and to produce effective machining operations. The establishment of materials database with their specific cutting parameters are

necessary for dealing with more different products to be manufactured. On top of that, optimization algorithms can be adopted into the programming system which indirectly control the cutting parameters automatically and propose optimum cutting parameters.

REFERENCES

This thesis is prepared based on the following references;

- [1] Aggarwal, A. and Singh, H., 2005. Optimization of machining techniques—a retrospective and literature review. *Sadhana*, 30(6), pp.699-711.
- [2] Agrawal, A., Soni, R.K. and Dwivedi, N., 2013. Development of integrated CNCRP system through CAD/CAM environment. *International Journal of Mechanical and Production Engineering Research and Development (IJMPERD)*, 3(5), pp. 1 - 10.
- [3] Campos, J.G. and Miguez, L.R., 2011. Standard process monitoring and traceability programming in collaborative CAD/CAM/CNC manufacturing scenarios. *Computers in Industry*, 62(3), pp.311-322.
- [4] Cay, F. and Chassapis, C., 1997. An IT view on perspectives of computer aided process planning research. *Computers in Industry*, 34(3), pp.307-337.
- [5] Chang, T.C. and Wysk, R.A., 1984. *An introduction to automated process planning systems*. Prentice Hall Professional Technical Reference.
- [6] Chang, T.C., 1990. *Expert process planning for manufacturing*. Addison-Wesley Longman.
- [7] Cheung, C.C., 2008. Semi-automated process planning and cost estimation of turned components based on CATIA V5 Machining.
- [8] Chu, X., Tang, C., Xue, D., Su, Y. and Sun, X., 2012. Identification of machining features based on available resources of cutting tools. *International journal of production research*, 50(15), pp.4141-4157.

- [9] Denkena, B., Shpitalni, M., Kowalski, P., Molcho, G. and Zipori, Y., 2007. Knowledge management in process planning. *CIRP Annals-Manufacturing Technology*, 56(1), pp.175-180.
- [10] Erdiđyi, F. and Hornyák, O., 2001, June. Simulation tools for supporting robust process planning in the field of NC turning. In *Proceedings 3rd Workshop on European Scientific and Industrial Collaboration* (pp. 27-29).
- [11] Groover, M.P., 2007. *Automation, production systems, and computer-integrated manufacturing*. Prentice Hall Press.
- [12] Jahan, M.D. and Kabir, G., 2010. Development of Computer Aided Process Planning (CAPP) for Rotational Parts. *Global Journal of Researches in Engineering*, 10(4), pp.16-24.
- [13] Knight, W.A. and Boothroyd, G., 2005. *Fundamentals of metal machining and machine tools* (Vol. 69). CRC Press. p. 401.
- [14] Koenig, D., 1990. *Computer-integrated manufacturing: theory and practice*. CRC Press.
- [15] Krishna, V., Shankar, N.V.S. and Babu, B.S., 2011. Feature based modelling and automated process plan generation for turning components. *Advances in production engineering and management*, 63, pp.153-162.
- [16] Lalit, N.K., Mallikarjuna, K. and Sarcar, M., 2008. *Computer Aided Design and Manufacturing*.
- [17] Leondes, C.T. ed., 2000. *Computer-Aided Design, Engineering, and Manufacturing: Systems Techniques and Applications, Volume V, The Design of Manufacturing Systems* (Vol. 5). CRC Press.
- [18] Rao, Posinasetti Nageswara. *CAD/CAM: principles and applications*. Tata McGraw-Hill Education, 2004.
- [19] Rho, H.M., Geelink, R., Van't Erve, A.H. and Kals, H.J.J., 1992. An integrated cutting tool selection and operation sequencing method. *CIRP Annals-Manufacturing Technology*, 41(1), pp.517-520.

- [20] Safaieh, M., Nassehi, A. and Newman, S.T., 2013. A novel methodology for cross-technology interoperability in CNC machining. *Robotics and Computer-Integrated Manufacturing*, 29(3), pp.79-87.
- [21] Schneider, G., 2010. Turning Tools and Operations. *American Machinist*.
- [22] Verma, A.K. and Rajotia, S., 2010. A review of machining feature recognition methodologies. *International Journal of Computer Integrated Manufacturing*, 23(4), pp.353-368.
- [23] Wang, H.P. and Li, J.K., 1991. Computer-Aided Process Planning, Advances in Industrial Engineering, Vol. 13.
- [24] Xu, X., Wang, L. and Newman, S.T., 2011. Computer-aided process planning—A critical review of recent developments and future trends. *International Journal of Computer Integrated Manufacturing*, 24(1), pp.1-31.
- [25] Zhang, H.C. and Altin, L., 1994. *Computerized manufacturing process planning systems*. Chapman & Hall, Ltd..

APPENDICES A1

Semester 1 Gantt chart

SEM 1 2015/2016		Week														
Task		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
		1	Project briefing by supervisor	P	A											
2	Determine project objective	P	A													
3	Determine problem statement	P	A													
4	Determine project scope	P	A													
5	Study about past research works and related articles	P		A	A	A	A									
6	Literature review	P					A	A	A							
7	Study and selection of software to be used	P					A									
8	Software installation and testing	P						A	A							
9	List bill of materials and budget plan	P						A	A							
10	Draft of proposal	P						A	A	A						
11	Proposal presentation	P									A	A				
12	Writing FYP 1 report	P				A	A	A	A	A	A	A	A			
13	Submission of Final Report FYP 1	P														A
		A														A

P=Plan
A=Actual

APPENDICES A2

Semester 2 Gantt chart

SEM 2 2015/2016		Week														
	Task															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	Update Gantt Chart	P	■													
		A	■													
2	NX software model part design	P	■													
		A	■													
3	Journaling of programming codes	P		■	■											
		A		■	■	■										
4	Programming codes customization	P			■	■										
		A			■	■	■									
5	Testing programming codes capabilities	P				■	■									
		A				■	■	■								
6	Improvise programming codes adaptation	P					■	■								
		A					■	■	■	■	■					
7	Testing improvised programming codes capabilities	P						■	■	■						
		A						■	■	■	■					
8	Graphical User Interface (GUI) design	P									■					
		A									■	■				
9	Improvise GUI design and functions	P									■	■				
		A									■	■	■			
10	Part fabrication to validate user interface adaptation	P										■	■			
		A										■	■	■		
11	Results and discussion	P											■	■		
		A											■	■	■	
12	Completion of thesis	P												■	■	
		A												■	■	■
13	Finalize of thesis	P													■	■
		A													■	■
14	Presentation of FYP 2 and submission of thesis	P														■
		A														■

P=Plan

A=Actual

APPENDICES B

Functional codes for selecting point inferred:

```
Imports System
Imports NXOpen
Imports NXOpen.UF
Imports NXOpenUI

Public Class select_point_with_point_constructor
    Public Shared theUFSession As UFSession
    Public Shared theSession As Session

    Public Shared Sub Main(ByVal args As String())
        theSession = Session.GetSession()
        theUFSession = UFSession.GetUFSession()

        Dim theUI As UI = UI.GetUI()
        'Insert code here
        Dim base_pt As Double() = New Double(2) {}
        While select_point("Select Point", base_pt) = UFConstants.UF_UI_OK
            theUI.NXMessageBox.Show("Selected Point",
NXMessageBox.DialogType.Information, (("X:" & base_pt(0).ToString() & vbLf &
"Y:") + base_pt(1).ToString() & vbLf & "Z:") + base_pt(2).ToString())

            End While
        End Sub

        Public Shared Function select_point(ByVal cue As String, ByRef base_pt As
Double()) As Integer
            Dim point_tag As NXOpen.Tag = NXOpen.Tag.Null
            Dim response As Integer = 0
            Dim base_method As UFUi.PointBaseMethod =
UFUi.PointBaseMethod.PointInferred
```

```
theUFSession.Ui.LockUgAccess(NXOpen.UF.UFConstants.UF_UI_FROM_CUSTOM)
    theUFSession.Ui.PointConstruct(cue, base_method, point_tag, base_pt, response)
```

```
theUFSession.Ui.UnlockUgAccess(NXOpen.UF.UFConstants.UF_UI_FROM_CUSTOM)
M)
```

```
    Return response
```

```
End Function
```

```
End Class
```

APPENDICES C**Machining program code generated for first model:**

```
%  
N0010 G94 G90 G20  
N0020 G50 X0.0 Z0.0  
N0030 T01 H00 M06  
N0040 G97 S2192 M03  
N0050 G94 G00 X2.0651 Z.6793  
N0060 X1.7384 Z-.0344  
N0070 G92 S0  
N0080 G96 S1995 M03  
N0090 G95 G01 X1.6911 F.009  
N0100 X-.0472 F.006  
N0110 G94 G00 X-.0945  
N0120 Z.0837  
N0130 X1.7402  
N0140 Z-.0689  
N0150 G95 G01 X1.6929 F.009  
N0160 X-.0472 F.01  
N0170 G94 G00 X-.0945  
N0180 Z.0492  
N0190 X1.7402  
N0200 Z-.1033  
N0210 G95 G01 X1.6929 F.009  
N0220 X-.0472 F.01  
N0230 G94 G00 X-.0945  
N0240 Z.0148  
N0250 X1.7402  
N0260 Z-.1378  
N0270 G95 G01 X1.6929 F.009  
N0280 X-.0472 F.01  
N0290 G94 G00 X-.0945  
N0300 Z-.0197
```

N0310 X1.7402
N0320 Z-.1722
N0330 G95 G01 X1.6929 F.009
N0340 X-.0472 F.01
N0350 G94 G00 X-.0945
N0360 Z-.0541
N0370 X1.7402
N0380 Z-.2067
N0390 G95 G01 X1.6929 F.009
N0400 X-.0472 F.01
N0410 G94 G00 X-.0945
N0420 Z-.0886
N0430 X1.7402
N0440 Z-.2411
N0450 G95 G01 X1.6929 F.009
N0460 X-.0472 F.01
N0470 G94 G00 X-.0945
N0480 Z-.123
N0490 X1.7402
N0500 Z-.2756
N0510 G95 G01 X1.6929 F.009
N0520 X-.0472 F.01
N0530 G94 G00 X-.0945
N0540 X2.0651 Z.6793
N0550 G97 S2230 M03
N0560 X1.5377 Z-.1113
N0570 G92 S0
N0580 G96 S1795 M03
N0590 G95 G01 Z-.1586 F.018
N0600 Z-3.1299 F.012
N0610 X1.5748 F.02
N0620 G94 G00 X1.6082 Z-3.0965
N0630 Z-.1102
N0640 X1.5007
N0650 G95 G01 Z-.1575 F.018
N0660 Z-3.1299 F.02
N0670 X1.5377
N0680 G94 G00 X1.5711 Z-3.0965

N0690 Z-.1102
N0700 X1.4636
N0710 G95 G01 Z-.1575 F.018
N0720 Z-3.1299 F.02
N0730 X1.5007
N0740 G94 G00 X1.5341 Z-3.0965
N0750 Z-.1102
N0760 X1.4265
N0770 G95 G01 Z-.1575 F.018
N0780 Z-3.1299 F.02
N0790 X1.4636
N0800 G94 G00 X1.497 Z-3.0965
N0810 Z-.1102
N0820 X1.3894
N0830 G95 G01 Z-.1575 F.018
N0840 Z-3.1299 F.02
N0850 X1.4265
N0860 G94 G00 X1.4599 Z-3.0965
N0870 Z-.1102
N0880 X1.3524
N0890 G95 G01 Z-.1575 F.018
N0900 Z-3.1299 F.02
N0910 X1.3894
N0920 G94 G00 X1.4228 Z-3.0965
N0930 Z-.1102
N0940 X1.3153
N0950 G95 G01 Z-.1575 F.018
N0960 Z-3.1299 F.02
N0970 X1.3524
N0980 G94 G00 X1.3858 Z-3.0965
N0990 Z-.1102
N1000 X1.2782
N1010 G95 G01 Z-.1575 F.018
N1020 Z-3.1299 F.02
N1030 X1.3153
N1040 G94 G00 X1.3487 Z-3.0965
N1050 Z-.1102
N1060 X1.2411

N1070 G95 G01 Z-.1575 F.018
N1080 Z-3.1299 F.02
N1090 X1.2782
N1100 G94 G00 X1.3116 Z-3.0965
N1110 Z-.1102
N1120 X1.2041
N1130 G95 G01 Z-.1575 F.018
N1140 Z-3.1299 F.02
N1150 X1.2411
N1160 G94 G00 X1.2745 Z-3.0965
N1170 Z-.1102
N1180 X1.167
N1190 G95 G01 Z-.1575 F.018
N1200 Z-3.1299 F.02
N1210 X1.2041
N1220 G94 G00 X1.2375 Z-3.0965
N1230 Z-.1102
N1240 X1.1299
N1250 G95 G01 Z-.1575 F.018
N1260 Z-3.1299 F.02
N1270 X1.167
N1280 G94 G00 X1.2004 Z-3.0965
N1290 Z-.1102
N1300 X1.0962
N1310 G95 G01 Z-.1575 F.018
N1320 Z-2.5394 F.02
N1330 X1.1299
N1340 G94 G00 X1.1633 Z-2.506
N1350 Z-.1102
N1360 X1.0624
N1370 G95 G01 Z-.1575 F.018
N1380 Z-2.5394 F.02
N1390 X1.0962
N1400 G94 G00 X1.1296 Z-2.506
N1410 Z-.1102
N1420 X1.0287
N1430 G95 G01 Z-.1575 F.018
N1440 Z-2.5394 F.02

N1450 X1.0624
N1460 G94 G00 X1.0958 Z-2.506
N1470 Z-.1102
N1480 X.9949
N1490 G95 G01 Z-.1575 F.018
N1500 Z-2.5394 F.02
N1510 X1.0287
N1520 G94 G00 X1.0621 Z-2.506
N1530 Z-.1102
N1540 X.9612
N1550 G95 G01 Z-.1575 F.018
N1560 Z-2.5394 F.02
N1570 X.9949
N1580 G94 G00 X1.0283 Z-2.506
N1590 Z-.1102
N1600 X.9274
N1610 G95 G01 Z-.1575 F.018
N1620 Z-2.5394 F.02
N1630 X.9612
N1640 G94 G00 X.9946 Z-2.506
N1650 Z-.1102
N1660 X.8937
N1670 G95 G01 Z-.1575 F.018
N1680 Z-2.5394 F.02
N1690 X.9274
N1700 G94 G00 X.9609 Z-2.506
N1710 Z-.1102
N1720 X.8565
N1730 G95 G01 Z-.1575 F.018
N1740 Z-.8399 F.02
N1750 G03 X.8933 Z-1.0059 I-.8694 K-.28
N1760 G01 X.8937 Z-1.0089
N1770 G94 G00 X.9271 Z-.9754
N1780 Z-.1102
N1790 X.8193
N1800 G95 G01 Z-.1575 F.018
N1810 Z-.7434 F.02
N1820 G03 X.8565 Z-.8399 I-.8322 K-.3764

N1830 G94 G00 X.8899 Z-.8065
N1840 Z-.1102
N1850 X.7821
N1860 G95 G01 Z-.1575 F.018
N1870 Z-.6701 F.02
N1880 G03 X.8193 Z-.7434 I-.795 K-.4497
N1890 G94 G00 X.8527 Z-.71
N1900 Z-.1102
N1910 X.7449
N1920 G95 G01 Z-.1575 F.018
N1930 Z-.6099 F.02
N1940 G03 X.7821 Z-.6701 I-.7578 K-.5099
N1950 G94 G00 X.8155 Z-.6367
N1960 Z-.1102
N1970 X.7077
N1980 G95 G01 Z-.1575 F.018
N1990 Z-.5586 F.02
N2000 G03 X.7449 Z-.6099 I-.7206 K-.5612
N2010 G94 G00 X.7783 Z-.5765
N2020 Z-.1102
N2030 X.6705
N2040 G95 G01 Z-.1575 F.018
N2050 Z-.5139 F.02
N2060 G03 X.7077 Z-.5586 I-.6834 K-.606
N2070 G94 G00 X.7411 Z-.5252
N2080 Z-.1102
N2090 X.6333
N2100 G95 G01 Z-.1575 F.018
N2110 Z-.4743 F.02
N2120 G03 X.6705 Z-.5139 I-.6462 K-.6455
N2130 G94 G00 X.7039 Z-.4804
N2140 Z-.1102
N2150 X.5961
N2160 G95 G01 Z-.1575 F.018
N2170 Z-.4391 F.02
N2180 G03 X.6333 Z-.4743 I-.609 K-.6807
N2190 G94 G00 X.6667 Z-.4409
N2200 Z-.1102

N2210 X.5589
N2220 G95 G01 Z-.1575 F.018
N2230 Z-.4076 F.02
N2240 G03 X.5961 Z-.4391 I-.5718 K-.7122
N2250 G94 G00 X.6295 Z-.4057
N2260 Z-.1102
N2270 X.5217
N2280 G95 G01 Z-.1575 F.018
N2290 Z-.3793 F.02
N2300 G03 X.5589 Z-.4076 I-.5346 K-.7406
N2310 G94 G00 X.5923 Z-.3742
N2320 Z-.1102
N2330 X.4845
N2340 G95 G01 Z-.1575 F.018
N2350 Z-.3538 F.02
N2360 G03 X.5217 Z-.3793 I-.4974 K-.7661
N2370 G94 G00 X.5551 Z-.3458
N2380 Z-.1102
N2390 X.4473
N2400 G95 G01 Z-.1575 F.018
N2410 Z-.3309 F.02
N2420 G03 X.4845 Z-.3538 I-.4602 K-.789
N2430 G94 G00 X.5179 Z-.3204
N2440 Z-.1102
N2450 X.4101
N2460 G95 G01 Z-.1575 F.018
N2470 Z-.3103 F.02
N2480 G03 X.4473 Z-.3309 I-.423 K-.8095
N2490 G94 G00 X.4807 Z-.2975
N2500 Z-.1102
N2510 X.3729
N2520 G95 G01 Z-.1575 F.018
N2530 Z-.2919 F.02
N2540 G03 X.4101 Z-.3103 I-.3858 K-.8279
N2550 G94 G00 X.4435 Z-.2769
N2560 Z-.2422
N2570 X.3691
N2580 G95 G01 X.3357 Z-.2756 F.018

N2590 G03 X.3729 Z-.2919 I-.3486 K-.8442 F.02
N2600 G94 G00 X.4063 Z-.2585
N2610 X2.0651 Z.6793
N2620 G97 S19191 M03
N2630 X.1985 Z-.155
N2640 G95 G02 X.2561 Z-.2756 I.0891 K-.0315 F.009
N2650 G92 S0
N2660 G96 S1995 M03
N2670 G01 X.2638 Z-.2783 F.01
N2680 G03 X.8657 Z-1.025 I-.3043 K-.8612
N2690 X.8661 Z-1.0321 I-.0566 K-.0072
N2700 G01 Z-2.5591
N2710 X1.1211
N2720 G02 X1.2155 Z-2.4646 I0.0 K.0945
N2730 G94 G00 X2.0651 Z.6793
N2740 M02
%