الجمـهوريـــة الجــزائريـة الديمـــــــــقراطيـة الشــــــــعبيـة

Democratic and Popular Republic of Algeria

وزارة التـــــعليـم العـــالي والبــحث العـــلمـــي

Ministry of Higher Education and Scientific Research

جـــــــــــامعة الشهيد العربـــــي التبســـي – تـبســـــــــة

University chahid Larbi Tébessi – Tébessa –

Faculty of Science and Technology

Department of Mechanical Engineering
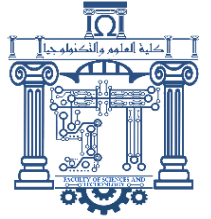
Master's Thesis

Submitted for Academic Master's degree

In: Mechanical Engineering

Specialty: Mechanical Construction

By: Houria Mohamed El Amine

Subject

Design and CNC machining of an impeller with non- twisted blades

Presented and publicly supported, on 07/ 06 /2023, in front of jury member:

Samir Daghbouj	Pr	President
Madani Maalim	MAA	Supervisor
Younes Ramdane	MCA	Examinator

Class : 2022/2023

ACKNOWLEDGEMENTS

DEDICATE

I dedicate this work to my parent.

To my brother & little sister.

To all my family and my friends.

Table of Contents

Figures List

Tables List

Abbreviations list

NC   Numerical Control

CNC   Computer Numerical Control

MIT   Massachusetts Institute of Technology

APT   Automatic Programed Tools

DNC   Direct Numerical Control

CAM   Computer Aided manufacturing

CAD   Computer Aided Design

CIM   Computer Integrated Manufacturing

ISO   International Standard Organization

B-Spline   Basis Spline

NURBS   Non-Uniform Rational Basis spline

P_i Control points

$B_i^n(t)$ Bernstein polynomial

w_i Weight

N_i Basis function

B_i Blending function

Abstract

In the mechanical sector, the design and manufacture of complex shaped parts such as automotive parts, boat shells, turbine blades and aerospace parts can be a long and costly process. The machining of such complex surface parts is generally carried out using 5-axis milling, so the generation of NC programs for the manufacture of these parts requires very high skill. For this, it is necessary to use CAM/CAD software.

In this work, a new method of manually programming with irregular cycle machining to get a G-code program for machining an impeller with non-twisted blade and comparing it with Camworks post process program.

Keywords: CNC; CAM/CAD; Freeform surfaces; NURBS; Sinumerik840D; G-code CAMWOKS

Résumé

Dans le secteur mécanique, la conception et fabrication des pièces en forme complexe telles que les pièces d'automobile, les coques de bateaux, les aubes de turbines et les pièces en aérospatial peut être un processus long et coûteux. L'usinage de telles pièce en surface complexes sont réalisé en général en utilisant le frisage 5 axes, donc la génération des programmes CN pour la fabrication de ces pièces exige une compétence très élevée. Pour cela il est nécessaire d'utiliser des systèmes d'aide informatique spécialise pour la conception et la fabrication assistées par ordinateur.

Dans ce travail, une nouvelle méthode de programmation manuelle avec l'usinage à cycle irrégulier est utilisée pour obtenir un programme G-code pour l'usinage d'une roue à aubes à pales non torsadées, et le comparer avec le programme de post-processus Camworks.

Mots-clés : MOCN ; CFAO ; Surfaces complexes ; NURBS ; SINUMERIK840D ; G-code CAMWORKS

General introduction

Machining is a technology that has undergone significant innovations in recent years. The joint improvement of tools, digital control directors and machine tools in general has expanded the range of applications of this technique.

Computer Numerical Control allows the manufacture of parts of complex shapes such as tooling, clogging matrices and plastic injection molds, which together constitute an industry with very high added value.

The freeform surfaces used in mechanical design and manufacturing are machined on CNC machine. The process of processing these surfaces go through three operations: roughing to remove the maximum material, semi-finishing to approach the final shape and finishing to obtain the final form with a good overhead condition and minimum processing time.

In this thesis a general study is proposed about design & machining compressor impeller on CNC machines using CAD/CAM software.

The first chapter present bibliographies research about Computer Numerical Control (CNC) citing architecture, classification and used programming language.

In the 2nd chapter a general overview about complexes form associated with curves and surfaces Based on polynomial models (BEZIER; B-SPLINE; NURBS) used on CAD/CAM.

The 3rd chapter present machining simulation using emco winnc sinumerik 840d software and their pre-programmed milling cycle with giving program example of some cycle.

Finally, in the last chapter a case study of designing a mold of a compressor impeller using CAD software (SOLIDWORKS) and machining on 5 axis using CAM software (CAMWORKS).

CHAPTER I

GENERALITIES ABOUT CNC

I.1 Introduction

Computer Numerical Control (CNC) machines have become a major part in almost all types of manufacturing processes and is used in a wide range of applications, from aerospace applications to the manufacturing of energy systems and medical robots, we see a major reliance on CNC machines [1].

CNC machining allows for higher precision, increased accuracy and greater repeatability than manual machining, making it possible to produce complex custom-designed parts. CNC machining also allows for the automation of multiple machine tools and the integration of complementary processes such as grinding and polishing.

In this chapter a general overview of CNC technology is presented, citing classification and architecture.

I.2 History

The work carried out by Falcon and Jacquard at the end of the seventeenth century showed that he was possible to control the movements of a machine from information transmitted by

perforated cardboard. Their 1805 loom was the first piece of equipment to be equipped with this from this point of view, it can be considered as the ancestor. However, it is necessary to

link NC's industrial operations to the development of electronics.

In 1947, the U.S. Air Force found that the complex designs and shapes of aircraft parts such as helicopter rotor blades and missile components were causing problems for manufacturers, who could not keep up to projected production schedules. At this time, John Parsons, of the Parsons Corporation, of Traverse City, Michigan, began experimenting with the idea of making a machine tool generate a "three-axis curve" by using numerical data to control the machine tool motions.[1]

But when the US Air Force entrusts to this same Parsons the realization of pieces of forms still more complex for its future supersonic aircraft, it realizes that its method is too approximate and that only a continuous machining in 3 dimensions will be able to give satisfaction.

In the spring of 1949, he entrusted the care to the Massachusetts Institute of Technology (MIT) Develop servos capable of driving a machine that will receive instructions from a card reader.

This machine, a prototype Cincinnati vertical spindle milling machine. Designed for perform

simultaneous displacements on 3 axes, is officially presented in September 1952 in the Servomechanisms laboratory of MIT. Mathematical information being the basis of the concept, it is given numerical control. It would just as well be called command symbolic.

- The different stages in NC development are as follows:

1954: Bendix acquires the Parsons patent and manufactures the first industrial NC.

1955: in Font de Lac (Wisconsin), the American manufacturer Giddins & Lewis sells the first NC.

1959: appearance of the NC in Europe (Hannover Fair). MIT announces the creation of the programming language APT (Automatic Programed Tools).

1960: appearance of the DNC (Direct Numerical Control) system

1964: In France, Electric Telemechanic launches the NC NUM 100 designed with Relay Teletatic.

1968: NC adopts integrated circuits; it becomes more compact and powerful.

The first machining center is on sale by Kearney & Trecker (USA).

1972: Minicomputers replace wired logic; NC becomes CNC.

1976: development of microprocessor NCs.

1984: appearance of advanced graphical functions and programming mode conversational, start of computer-aided manufacturing (CAM).

1986: NC integrated into communications networks, the beginning of the manufacturing era flexible (CIM: computer integrated manufacturing).

1990: development of 32-bit microprocessor NCs (Fig I.1).

Figure I- 1: 32-bit NC operator control

I.3 Definition and structure of a numerically controlled machine tool

I.3.1 Definition

Computer Numerical Control (CNC) is an automated manufacturing technology that uses numerical instructions stored in computer memory to control and coordinate the operation of machine tools such as lathes, mills, routers, and grinders. Written in a specific programming language, these instructions instruct the machine tool to produce parts with high precision and repeatability.

I.3.2 Structure of a numerically controlled machine tool

CNC system consists of the following 6 major elements [1]:

  a) Input Device
  b) Machine Control Unit
  c) Machine Tool
  d) Driving System
  e) Feedback Devices
  f) Display Unit

the (fig I.2) present CNC machine diagram structure:
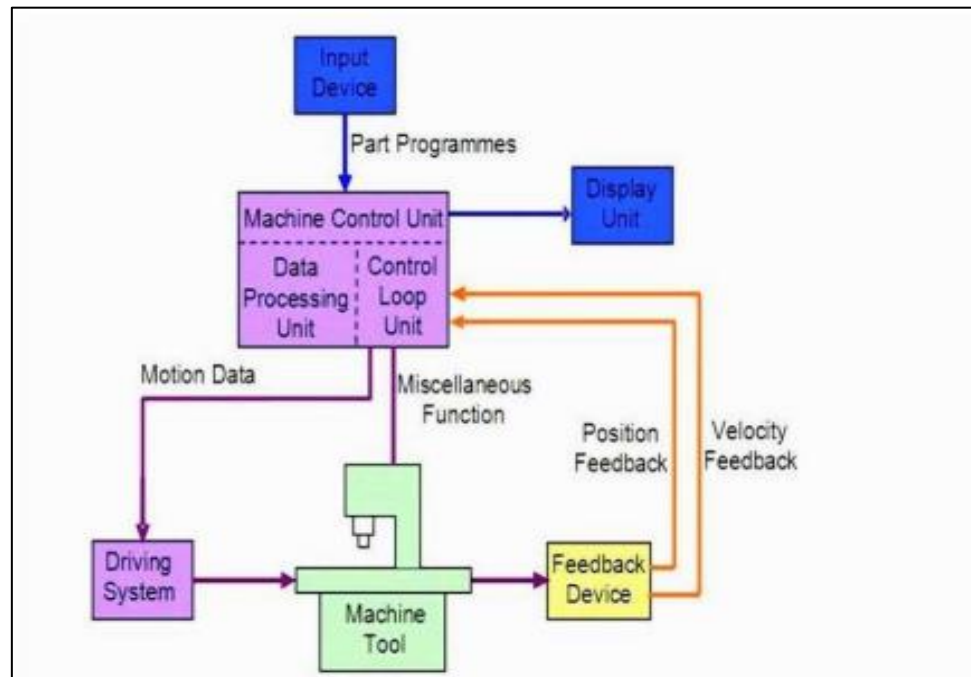
Figure I- 2: diagram of CNC structure [1]

a) Input device:

- Floppy Disk Drive
- USB Flash Drive
- Serial communication

b) Machine control unit (MCU)

It's the heart of CNC system. There are two sub-units in the machine control unit

- Data processing unit (DPU)
- Control loop unit (CLU)

c) Machine tool

There are many different types of tools for each operation like milling, turning, drilling…etc.

d) Driver system

The driving system is an important component of CNC machine as the accuracy and repeatability depend very much on the characteristics and performance of the driving system. The driving system is classified as (fig I-3):

- Spindle drives: used to provide angular motion to the workpiece or cutting tool
- Feed drives: used to drive the slide or the table
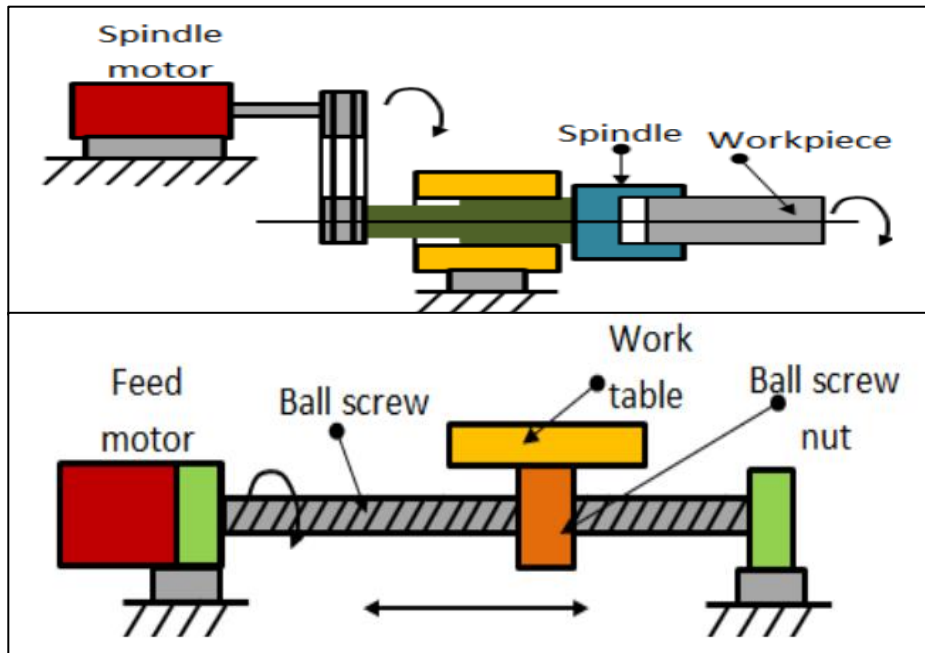
Figure I- 3: driver system [2]

e) Feedback device

A CNC machine takes the commanded position from the CNC program. The drive motor is rotated a corresponding amount, which in turn drives the ball screw, causing linear motion of the axis. A feedback device confirms that the proper amount of ball screw revolutions has occurred. (Fig I-4) define the feedback system.
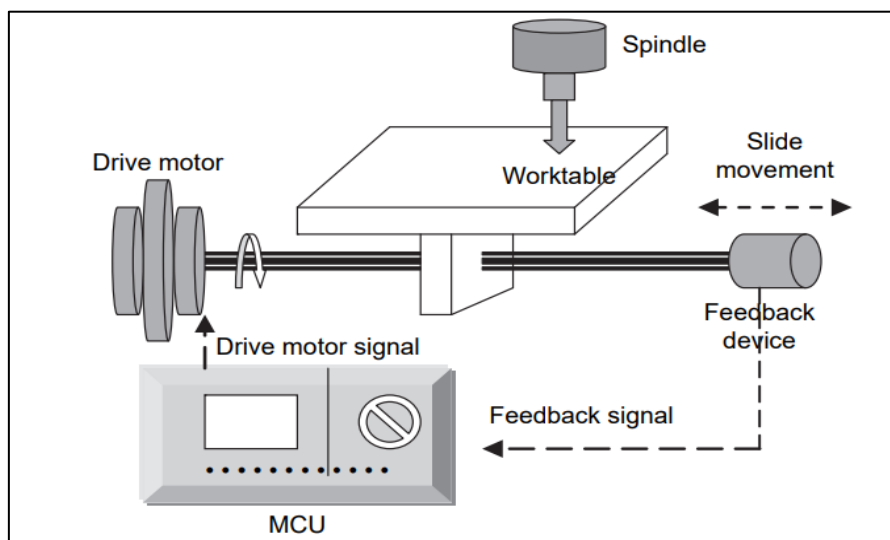
Figure I- 4: Feedback system [3]

g) Display Unit

The Display Unit serves as an interactive device between the machine and the operator. In an advanced CNC machine, the Display Unit can show the graphics simulation of the tool path so that part programs can be verified before the actually machining. When the machine is running, the Display Unit displays the present status such as the position of the machine slide, the spindle RPM, the feed rate, the part programs, etc..[1]

I.4 Classification of CNC

CNC machine tool systems are classified in several ways such us:

I.4.1 classification based on control system

- Point-to-point system

Point-to-point control equipment is the name given to the control equipment used with them. These are referred to as point-to-point machines. Some machine tools, such as drilling, boring, and tapping machines, need the cutter and work piece to be put in specific fixed relative locations that must be maintained while the cutter does its operation.[1]
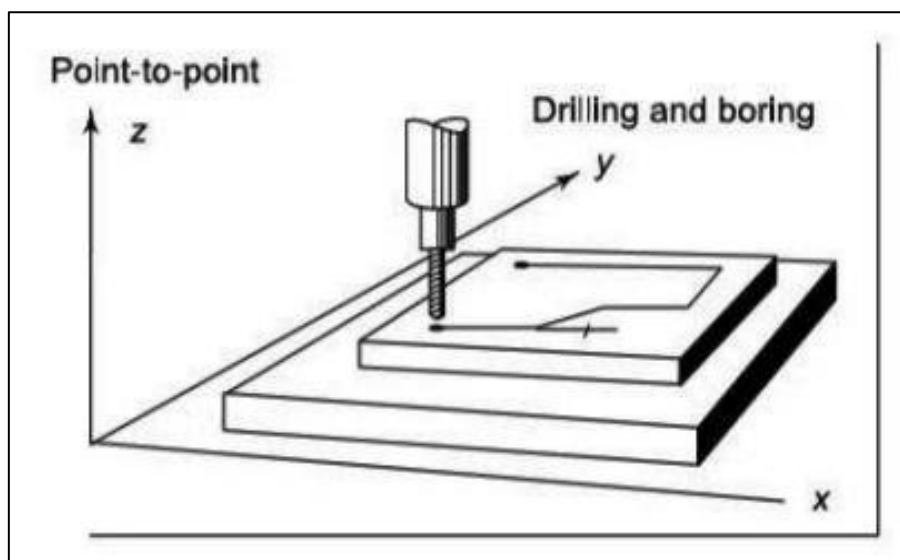
Figure I- 5: point to point system [1]

- Contouring system

These machine tools, which include milling and routing machines, are referred to as "contouring machines," and the controls required for their operation are referred to as "contouring controls." Contouring machines can also be utilized as point-to-point machines; however, they are not cost-

effective unless the work piece also requires a contouring operation. Other types of machine tools involve the work piece moving with regard to the cutter while the cutting operation is underway.[1]
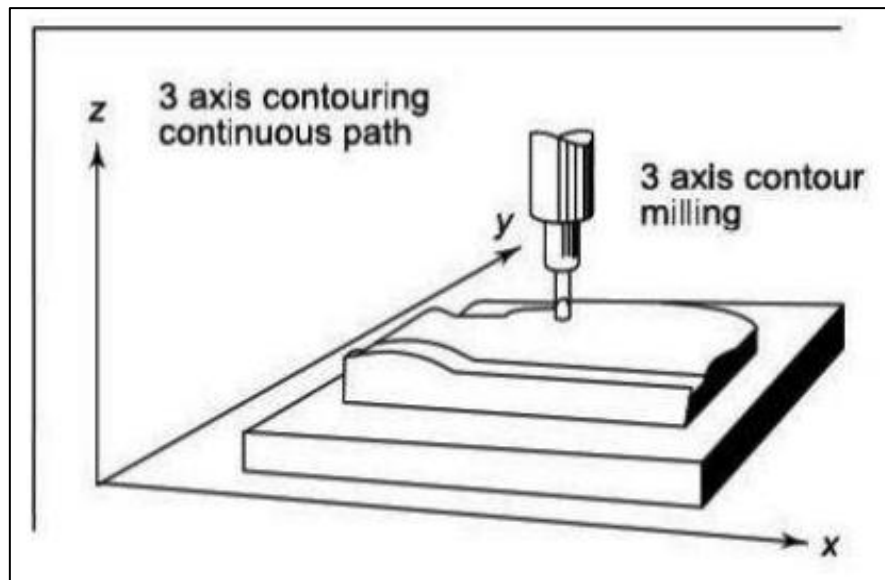
Figure I- 6: Contouring system [1]

I.4.2   Based on feedback control

- Open loop system

In an open-loop system, the machine slides are shifted based on the information loaded into the control system from the part program. As a result, there is no slide position measurement and no feedback signals to compare to the input signal. The ability of the driving systems to move the slide through the exact required distance is fully dependent on the correct movement of the slide. A stepper motor is the most popular technique for driving the lead screw. Stepper motors are the most straightforward way to translate precise electrical inputs into proportionate movement. Because the sliding position is not checked, the system's accuracy is dependent on the motor's ability to step through the exact number of steps provided at the input, as demonstrated in (fig I-7). [1]

- Closed loop system

A closed-loop system returns a signal from a measuring instrument known as a transducer to the control unit. The slides are attached to the transducer. The signal indicates the position and movement of the slides. The control unit adjusts the slide's position until it reaches its destination;

this mechanism has feedback. Although more expensive and sophisticated than open-loop systems, these systems provide more precise positioning. Servomotors are employed in this type of system. [1]
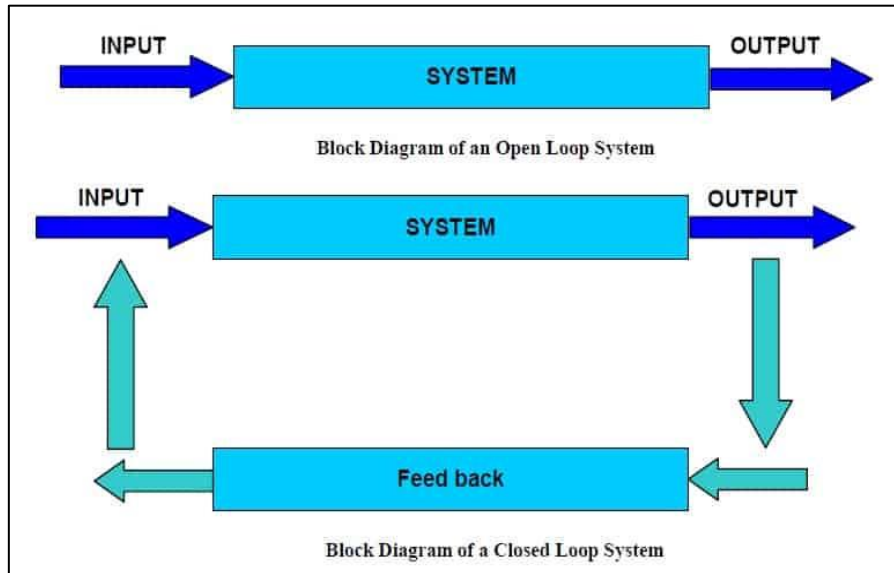
Figure I- 7: open and closed loop system [1]

I.4.3   Based according to the number of axes

We now classify the machine tools by the number of elementary movements that they can be implemented when moving the tool relative to the workpiece.

➢ 2& 3 axes CNC machines

CNC lathes will be coming under 2 axes machines. There will be two axes along which motion takes place. The saddle will be moving longitudinally on the bed (Z-axis) and the cross slide moves transversely on the saddle (along X-axis). In 3-axes machines, there will be one more axis, perpendicular to the above two axes. By the simultaneous control of all the 3 axes, complex surfaces can be machined.

➢ 4 & 5 axes CNC machines

4 and 5 axes CNC machines provide multi-axis machining capabilities beyond the standard 3-axis CNC tool path movements. A 5-axis milling center includes the three X, Y, Z axes, the A axis which is rotary tilting of the spindle and the B-axis, which can be a rotary index table.

The table below present the axis of possible operation of a CNC machine:

Table I- 1: Axis of the different possible movements

N° of axis	Movement	Possible operations
1	Z	Pinout, press
2	Z, X	Turning: all the shapes obtained have the same axis of symmetry
3	X, Y, Z	Milling: surfacing, drilling, milling of pockets grooves and freeform surfaces. The tool axis remains parallel to a fixed direction relative to the workpiece.
3	X, Z, C	Turning: with spindle servo control allows milling on lathes: general turning, turning tool milling, off-axis drilling. The spindle is guided into position.
4	X, Y, Z, B	Milling: surfacing, drilling, milling of pockets grooves and freeform surfaces. The tool axis remains contained in a fixed plane.
4	X, Y, Z, C	Milling (cf. X, Y, Z, B)
4	X, Y, Z, C	Turning
4	2x (X, Z)	Two-turret turning
5	2x (X, Z), C	Turning with two turrets and servo-control of the spindle, (cf X, Z, C)
5	X, Y, Z, A, C	milling of freeform surfaces: milling with the sidewall of the tool, milling with deburring, drilling in all directions
5	X, Y, Z, B, C	Milling of freeform (cf. X, Y, Z, A, C)

(Fig I-8) show the different primary and additional axis on CNC machine
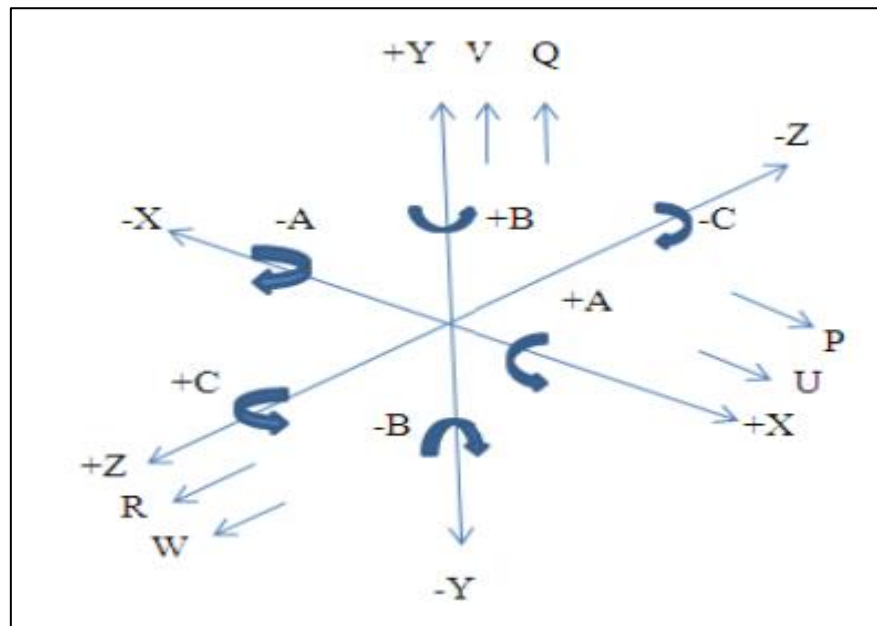
Figure I- 8: Primary Axes and Additional Axes

I.5   Part program

A part program is a set of instructions often referred to as blocks, each of which refers to a segment of the machining operation performed by the machine tool. Each block may contain several code words in sequence [1].

I.5.1   Basic Codes in CNC Machines

➢ Preparatory Codes (G) for initial machining setup and operating conditions.

➢ Miscellaneous Codes (M) for on/off of coolant flow, spindle rotation etc.

➢ Spindle Speed (S)

➢ Feed Rate (F)

➢ Tool number (T)
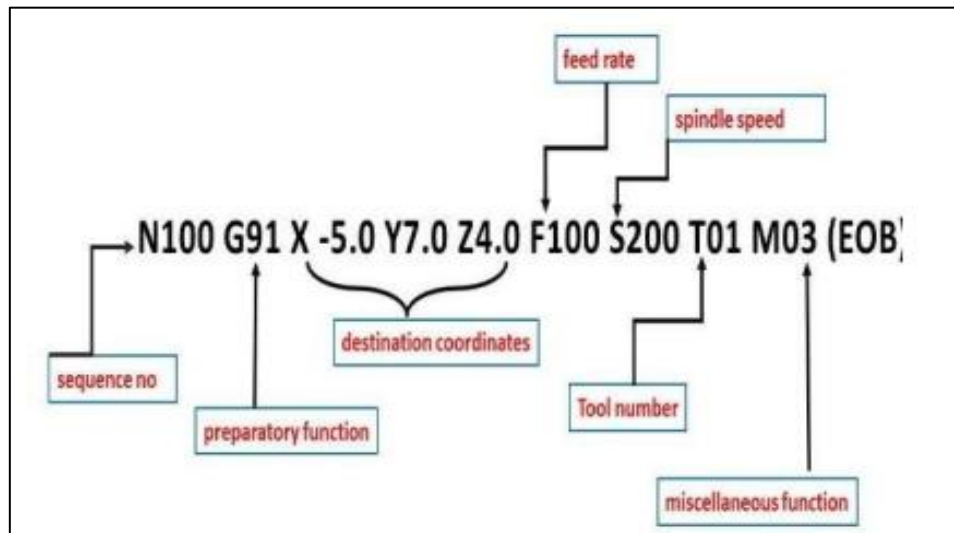
Figure I- 9: block program [1]

I.5.2   Programming languages

Originally, the programming language was G-code, developed by the EIA (Electronic Industries Alliance) in the early 1960s, and finally standardized by ISO in February 1980.

a) ISO language (International Standard Organization)

Developed by the EIA in the early 1960s the programming language was G-code, and finally standardized by ISO* in 1980 under the reference (ISO 6983). The ISO language is widely used and serves as the basis for many current languages.

b) FANUC language

The Fanuc language is based on the ISO language of 1980. It adds additional functions which makes it a unique language. The specifics of the language are:

- Parentheses for comments
- Calling sub-program with M98
- Semicolons at the end of blocks

c) SIEMENS language

The SIEMEMS language is based on the ISO language. It adds functions this is what makes it a unique language. The specifics of the language are**:**

- Semi colons for comments
- Cycle calls

d) NUM language

The NUM language is based on the ISO language. It adds additional functions this which makes it a unique language. The specifics of the language are:

- Parentheses for comments
- Calling sub-program with G77

e) HEIDENHAIN language

The HEIDENHAIN language is a language invented by the Germans to animate mainly conversational CNC robots. This language represents practically no elements of the ISO code.

f) MAZATROL language

Mazatrol uses its own conversational and intuitive programming language. The idea of

their language is to reduce the complexity

I.6   Main functions

I.6.1   The main functions of the G-code

The main functions of the G-code are summarized in the following table [4]:

Table I- 2: Main G-code function

G00	Rapid Positioning Motion
G01	Linear Interpolation Motion
G02	Circular Interpolation Motion CW
G03	Circular Interpolation Motion CCW
G04	Dwell (P) (P=Seconds″.″ Milliseconds)
G10/G11	Writing data/Erasing data
G17	XY plane
G18	XZ plane
G19	YZ plane

G20	Inch units
G21	Metric units
G28	Automatic return to reference point
G31	Skip function (used for sensors and tool length measurements)
G33	Constant pitch thread
G34	Variable pitch thread
G40	Cutter Compensation Cancel G41/G42/G141 (X , Y)
G41	2D Cutter Compensation, Left (X, Y, D)
G54	Select work coordinate system
G42	2D Cutter Compensation, right (X, Y, D)
G90	Absolute Positioning Command
G91	Incremental programming
G94/95	Inverse Time Feed OFF / Feed Per Minute ON
G98	Set initial plane default

I.6.2   The main functions of the M-code

The main functions of the M-code are summarized in the following table [4]:

Table I- 3: Main M-code function

M00	Program stop
M01	Optional stop
M02	Program end
M03	Spindle start
M04	Spindle start (anti-clockwise)
M05	Spindle stop
M06	Tool change

M07	Coolant ON (Within the spindle)
M08	Coolant ON
M09	Coolant OFF
M30	End program
M98	Call subroutine
M99	End subroutine

I.7   Advantages and inconveniences

I.7.1   The advantages of CNC machines

- Flexibility in operations is improved, as it has the ability to produce complex shapes with good dimensional accuracy
- Repeatability
- Reduced scrap loss
- Easier to program
- Easy store
- edit and change existing programs
- Stabilize manufacturing costs
- Avoids human errors
- Safer to operate
- Usually generates closer tolerances than manual machines
- High production rates

I.7.2   The advantages of CNC machines

There is no perfect invention in the world when it has advantages, it is meaning has disadvantages but the good invention it's the advantages is bigger than its disadvantage.

The CNC disadvantages [5]:

- The high cost of the machine
- Operator must be provided at the highest level of training to deal with this type of

machines

- Shortage of specialists in the maintenance of this type of machines

- Providing special Tools for remote control this machine and till working continually

I.8   application of CNC machine

CNC machines are widely used in the metal cutting industry and are best used to produce the following types of products [1][5]:

- Parts with complicated contours

- Parts requiring close tolerance and/or good repeatability

- Parts requiring expensive jigs and fixtures if produced on conventional machines

- Parts that may have several engineering changes, such as during the development stage of a prototype in cases where human errors could be extremely costly

- Parts that are needed in a hurry

- Small batch lots or short production runs.

- The applications of CNC include both for machine tool as well as non-machine tool areas. In the machine tool category, CNC is widely used for lathe, drill press, milling machine, grinding unit, laser, sheet-metal press working machine, tube bending machine etc.

I.9   CAD/CAM and CNC

The need for efficiency and accuracy in CNC programming has been major reason for

development of a variety of methods that use a computer to prepare part programs. Computer assisted CNC programming has been around for many years. First in the form of language-based programming, such as APT™ or Compact II™. Since the late 1970's, CAD/CAM has played a significant role by adding the visual aspect to the programming process. The acronym CAD/CAM means Computer Aided Design and Computer Aided Manufacturing. The first three letters (CAD) cover the area of engineering drafting.[6]

I.9.1   Computer-Aided Design (CAD)

The computer design system has been closely associated with the development of the concept of computer graphics. However, the concept of computer design goes beyond many computer graphics in terms of analysis and modeling. However, interactive computer graphics (ICG: Interactive Computer Graphics) is the necessary technical basis for the computer design system.

It is a software that allows us to draw the geometric pieces to be produced in all dimensions and then convert the two-dimensional shapes to three-dimensional if necessary. It also allows us to simulate geometric pieces in terms of movement, distribute stresses and also allows Draw the curves of forces and moment that affect the piece of engineering to be produced We then save the file in an appropriate format for understanding CAM software the most famous of these programs [5][7]:

- Autodesk Inventor
- SolidWorks
- CATIA

The (figure I-10) it explains how to process cad software: The sketch is beginning from start and adjust settings. if we need draw the 2 dimensions, we choice the 2D icon or if we need draw 3 dimensions, we choice the 3D icon. and we can transfer design 2D to 3D by option in software. After finish the design we export the design to program languages for reading the design by CAM.[5][7]
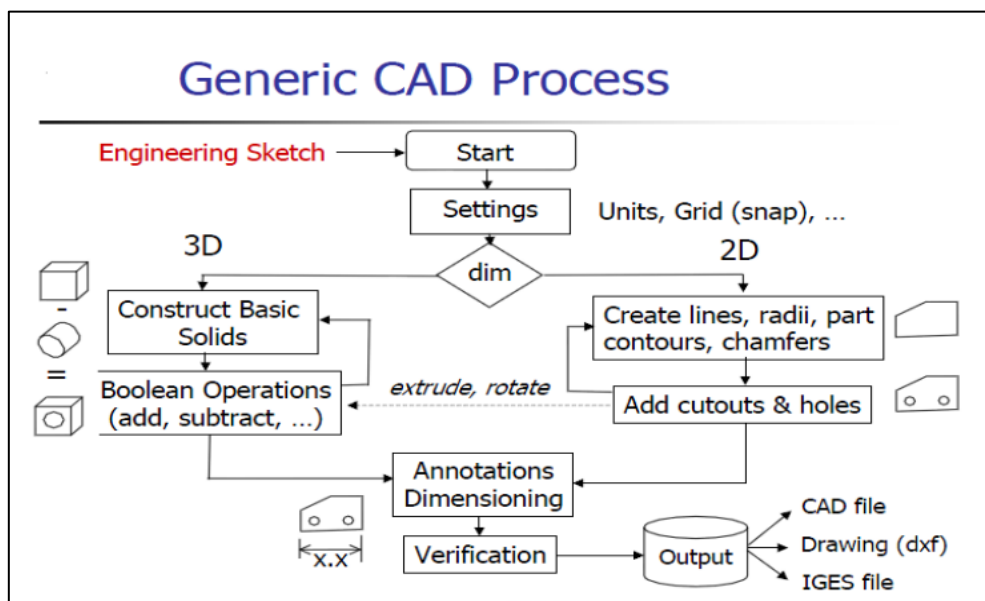
Figure I- 10: Generic CAD Process [5]

I.9.2   Computer-Aided Manufacturing (CAM):

Computer Aided Manufacturing can be defined as the use of computer systems to plan, manage and control manufacturing processes through direct or indirect interaction of computers with production sites in the factory. CAM is a software that converts a part of engineering designed on CAD software to G-CODE to enable the CNC machine to read the design. This G-CODE is

transferred to the CNC machine directly from the software via a network cable or indirectly via Flash memory and then determine the parameters of the CNC machine for the beginning of the manufacturing stage [7].

The most famous of these programs:

- Surfcam
- PowerMill
- camworks
- Solidcam

The (figure I-11) shows and summarized the typical workflow of a CAM system. The order of the definition can be different in different CAM systems, but the definition of the listed data is required every case for appropriate work. The most important application of the CAM systems is the milling technology, so hereafter I focus to this application [5].
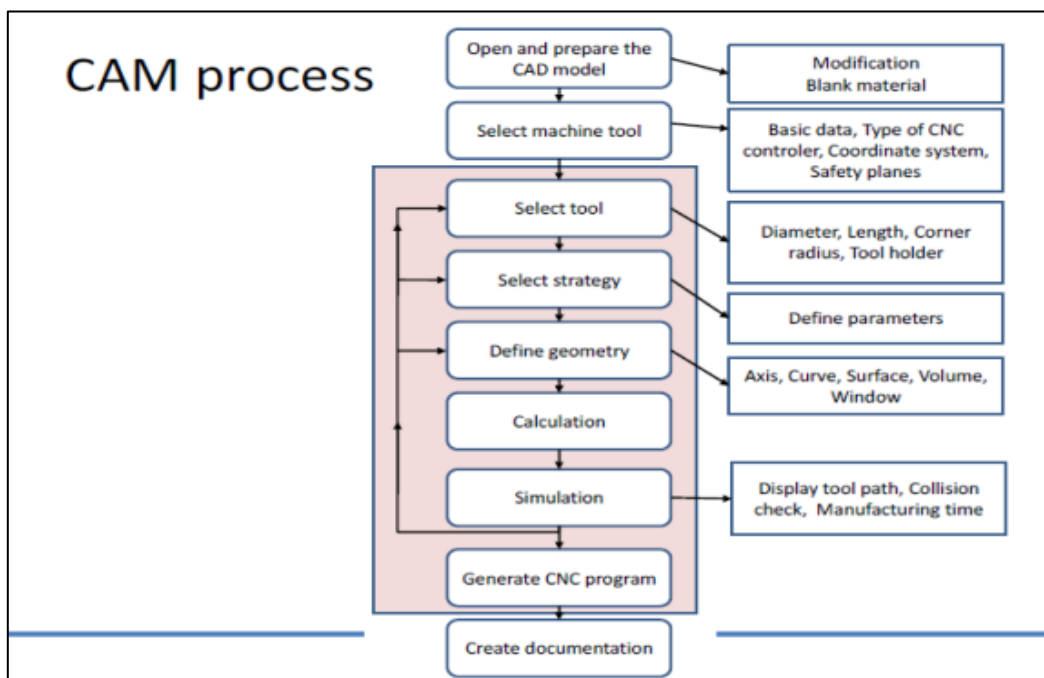
Figure I- 11: CAM Process [5]

I.10 Conclusion:

In this chapter, we have introduced computer numerical control (CNC). Its history, what are its components. How does it work and what are the pros and cons. We discuss computer-aided design (CAD) and computer-aided manufacturing (CAM) and how they work in CNC.

CHAPTER II

Freeform Curves and surfaces

II.1 Introduction

Mechanical manufacturing has employed a variety of CAD/CAM systems for many years, each offering unique interpolation techniques. The goal of this diversity is to achieve tool trajectories better suited to CAM tolerance when machining intricate surfaces. Common methods include linear, circular, and polynomial interpolation.

Freeform or synthetic curves and surfaces are necessary for products with specially crafted shapes, such as car bodies, ship hulls, impeller blades, and stylish consumer goods. The demand for these shapes arises when curves are represented through measured data points. Modern techniques have been developed to construct these curves. For example, in Computer-Aided Design (CAD) applications, popular CAD modeling software relies heavily on Bezier, B-spline, and NURBS representation. These representations provide easy interactive shape modification and efficient computational interrogations.

II.2 Linear interpolation and circular interpolation

When using modern CAD/CAM systems, designers adopt more free-form or contoured geometric shapes for the design and modeling of complex parts. Since conventional CNC (Computer Numerical Control) machines only provide linear (G01) and circular (G02, G03) interpolations, CAM systems have to create many linear and circular segments to approximate the contoured geometry under given tolerances before sending NC codes to CNC machines [8].

Figure II- 1: a) linear interpolation   b) circular interpolation

II.1.1 Linear interpolation

The term linear interpolation means that the control system is capable to calculate thousands of intermediate coordinate points between the start point and end point of the cut.

Result of this calculation is the shortest path between the two points. All calculations are automatic, the control system constantly coordinates and adjusts the feedrate for all cutting axis, normally.

Linear interpolation is closely related to fast positioning movement. While the rapid movement of the tool from one workspace position to another is intended for use without cutting, the linear interpolation mode is designed for practical material removal such as profiling, pocketing, facing milling and many other cutting movements.

It used in part programming to perform a straight-line cutting motion from the tool start position to the end position. It always uses the shortest distance the cutting toolpath can travel. The movement programmed in linear interpolation mode is always a straight line connecting the start and end points of the contour. In this mode, the tool moves from one location to another, moving the shortest distance between the endpoints. This is a very important programming function, mainly used for contouring and profiling. All angular movements (such as chamfers, chamfers, angles, tapers, etc.) must be programmed in this mode to be accurate.[24]

II.1.2 Circular interpolation

Programming a toolpath along an arc is similar to programming a toolpath along a straight line. The process of arc contour machining is called arc interpolation. Commonly used for profiling on vertical and horizontal CNC machining centers. Circular interpolation for programming in applications such as outer and inner diameters (collars and pitch circles), circular grooves, spherical or conical, radial grooves, grooves, broken corners, helical milling, and even large countersinks Arc or full circle. If the necessary information is provided in the program, the CNC unit interpolates the defined arc with very high precision.

Circular interpolation is defined by the codes G2 and G3 (according to movement, standard, etc.) in the CN file understandable by the HSM (high speed machining) machine controller.[24]

II.2 Curves and surfaces representation

II.2.1 History

Before the advent of computer graphics, engineers used splines to design airplane wings and car chassis. A spline is a long, flexible piece of wood or plastic, with a rectangular cross-section, held in various positions by heavy lead blocks with protrusions called ducks that hold the spline in fixed positions on the drawing board [9]. The splines then fit the natural shape between the ducks. By moving the duck, the designer can change the shape of the spline. The downside is obvious, and the mapping equipment needed to record the duck's location and repair the many complex parts takes up space in storage facilities, at the expense of consumers. A less obvious disadvantage is that there are no closed solutions in mathematical analysis [9].

In the 1960s, the landscape of mathematics and engineering underwent a transformation thanks to the innovation of Pierre Bezier. With the aid of a computer, Bezier developed the CAGD software UNISURF, which allowed designers to craft curves that appeared seamless on a monitor. This departure from traditional design methods requiring physical space sparked the evolution of CAD software. Further, Bezier's strides led to a novel mathematical construct referred to as a spline a curve defined by minimal points.

II.3 Bezier curve and surfaces

II.3.1 Bezier curve

Bezier curves are a type of parametric curve that is employed in computer graphics and related fields. These curves are delineated by a set of control points that serve as guides that define the curve's shape. The curve is subsequently defined by a mathematical algorithm, which is typically referred to as 'DeCasteljau's' algorithm. This algorithm determines the points on the curve based on the set of control points. The approach of 'DeCasteljau' being more intuitive and more used. Bezier's method is based on the deformation of space. We start from a simple curve (like a quarter circle) on a system of axes. In reality, the curve chosen by Bezier was a polynomial.

We start by letting the four points of a control polygon be the set P, where P = {P0, P1, P2, P3}. The position of these points in two or three dimensions determines the curvature of the curve. Now let Q(u) be a parametrically defined vector valued function where $0 \leq u \leq 1$. As u varies from 0 to 1, the vector values of Q(u) sweep out the curve [9].

Mathematically a parametric Bezier curve is defined by [10]:

$$p(u) = \sum_{i=0}^{n} B_{n,i}(u) P_i \quad \text{........ (1)}$$

Recall the Bernstein polynomials of degree n (we will use this in the general case of degree-n Bezier curves) [1]:

$$B_i^n(u) = c\binom{n}{i} u^i (1-u)^{n-i} \quad \text{...... (2)}$$

where i = 0, 1, 2, ..., n, and

$$c\binom{n}{i} = \frac{n!}{n!(i-n)!} \quad \text{....... (3)}$$

Now we can mathematically define a degree three Bezier curve [9-10]

$$p(u) = B_0^3(u)P_0 + B_1^3(u)P_1 + B_2^3(u)P_2 + B_3^3(u)P_3 \ \text{......} \ \textbf{(4)}$$

Finding expressions for each basis term is straight forward for the degree three case, but can be very tedious for splines of degree greater than three. Knowing that n = 3 and we have 4 control points, by using (1) we compute [2-4]:

$$\begin{aligned}
B_0^3(u) &= (1-u)^3 \\
B_1^3(u) &= 3u(1-u)^2 \\
B_2^3(u) &= 3u^2(1-u) \quad \text{......} \ \textbf{(5)} \\
B_3^3(u) &= u^3
\end{aligned}$$

Putting these terms back into (4) gives us our polynomial p(u):

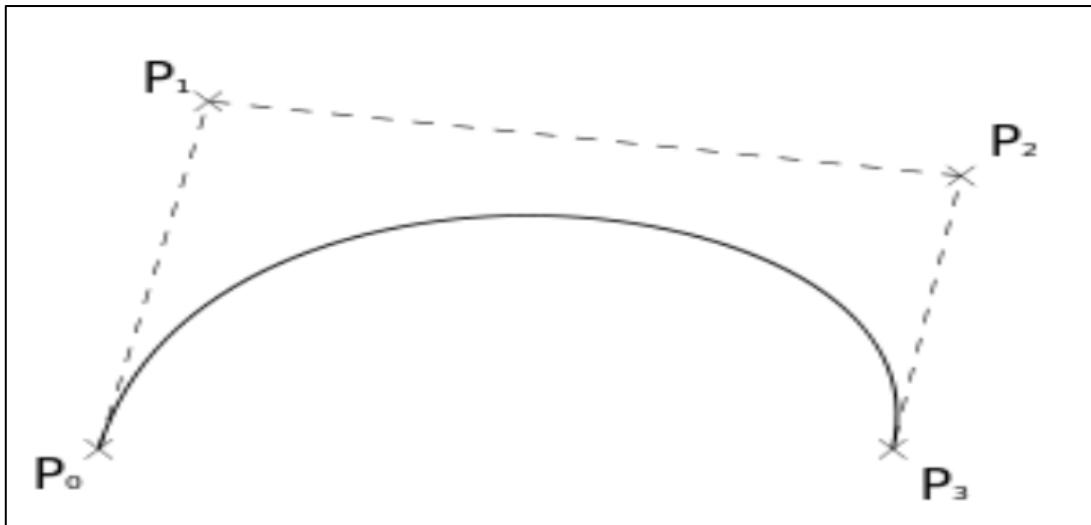$$p(u) = (1-u)^3 P_0 + 3u(1-u)^2 P_1 + u^2(1-u)P_2 + u^3 P_3 \ \text{.....} \ \textbf{(6)}$$

Figure II- 2: Bezier curve

Example 1:

Supposing that the control point of the desired curves are:

P0= (1,0), P1= (3,3), P2= (6,3), P3= (8,1).

And n=3, involves Bezier curves.

Let u: (0 0.2 0.5 0.7 1)

Employing the equation (6) to determine p(u), the plotting of the control points and the curve resulting from the equation (6) show it the (Fig 2), with $0 \leq u \leq 1$

For u=0

$$p(0) = [1 \ 0](1-0)^3 + [3 \ 3]3(0)(1-0)^2 + [6 \ 3]3(0)^2(1-0) + [8 \ 1](0)^3$$

$$p(0) = [1 \ 0]$$

For u=0,2

$$p(0.2) = [1 \ 0](1-0.2)^3 + [3 \ 3]3(0.2)(1-0.2)^2 + [6 \ 3]3(0.2)^2(1-0.2) + [8 \ 1](0.2)^3$$
$$p(0.2) = [2.30 \ 1.44]$$

For u=0.5

$$p(0.5) = [1 \ 0](1-0.5)^3 + [3 \ 3]3(0.5)(1-0.5)^2 + [6 \ 3]3(0.5)^2(1-0.5) + [8 \ 1](0.5)^3$$
$$p(0.2) = [4.5 \ 2.375]$$

For u=0.7

$$p(0.2) = [1 \ 0](1-0.7)^3 + [3 \ 3]3(0.7)(1-0.7)^2 + [6 \ 3]3(0.7)^2(1-0.7) + [8 \ 1](0.7)^3$$
$$p(0.2) = [5.98 \ 2.23]$$

For u=1

$$p(0.2) = [1 \ 0](1-1)^3 + [3 \ 3]3(1)(1-1)^2 + [6 \ 3]3(1)^2(1-1) + [8 \ 1](1)^3$$
$$p(0.2) = [8 \ 1]$$

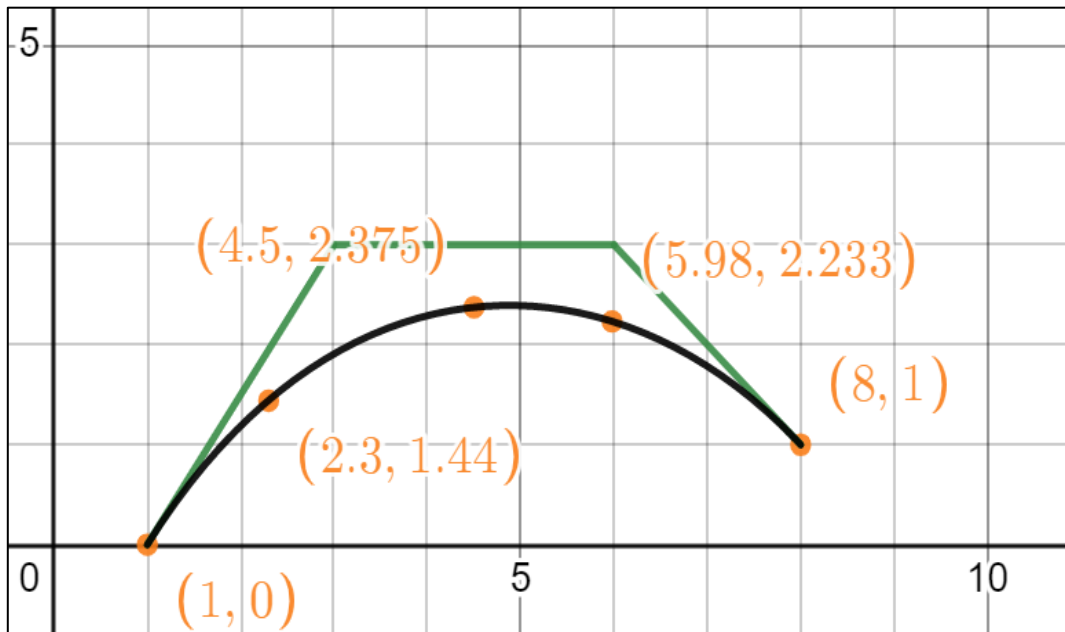We can now draw our curve (in this case we have used 'desmos')

Figure II- 3: Bezier curve with four control points n=3

II.3.2 Rational Bezier geometries

Polynomials can be used to describe a parabola, but not any of the remaining conic shapes. It is known from projective geometry, that any conic curve can be seen under the same structure and properties; the only difference is that homogeneous coordinates need to be considered [9].

With these homogeneous coordinates (x, y, z, ω), and making them equal to the classical ones (x', y', z'), the expression yields to $(x', y', z') = (\dfrac{x}{\omega}, \dfrac{y}{\omega}, \dfrac{z}{\omega})$ and $\mathbf{P} = \omega . \mathbf{P'}$.

The resulting geometry can be written as:

$$C(u) = \frac{\displaystyle\sum_{i=0}^{n} B_{i,n}(u) . w_i . P_i}{\displaystyle\sum_{i=0}^{n} B_{i,n}(u) . w_i} , \quad u \in [0,1] \qquad \text{....... (7)}$$

where w_i is the weight in each control point P_i.

The denominator of projective geometry is a non-negative polynomial, allowing the geometry to be defined as one polynomial divided by another. However, the number of control points is linked to the order of the geometry, leading to numerical instabilities and computational inefficiency.

II.3.3 Bezier surfaces

The Bezier surface is simply an extension of the Bezier curve in two parametric directions, a tensor product of two curves. It is evaluated through [9-10-11]

$$S(u,v) = \sum_{i=0}^{n}\sum_{j=0}^{m} P_{i,j} B_{i,n}(u) B_{j,m}(v) \quad \dots\dots \textbf{(8)}$$

Hence, although it is frequently the case that n = m, the parametric directions have degrees

n and m, respectively. This results in the control points (n + 1) *(m + 1).

The control points are supposed to be stored in a matrix, according to equation (8) however

Since the outcome is unaffected by the evaluation order, this must not necessarily be the case.

$$S(u,v) = \frac{\sum_{i=0}^{n}\sum_{j=0}^{m} w_{i,j} P_{i,j} B_{i,n}(u) B_{j,m}(u)}{\sum_{i=0}^{n}\sum_{j=0}^{m} w_{i,j} B_{i,n}(u) B_{j,m}(u)} \quad \dots\dots \textbf{(9)}$$

Although the intimidating appearance of Equation (9), it is only the evaluation of a rational surface, which is perfectly similar to the situation with one parametric direction. We frequently use the same limitations for surface concatenation, but this time in two parametric directions.

II.4 B-splines curve and surfaces

II.4.1 B-splines curve

A spline of degree n is a piecewise polynomial function of degree n that is continuous of class

C^{n-1} in every node.

A B-spline curve may be described in a few different ways. It resembles the Bezier formulation in its simplest form [10]:

$$p_i(u) = \sum_{i=0}^{n} P_i N_{i,k}(u) \quad \dots.. \textbf{(10)}$$

$$N_{i,1}(u) = \begin{cases} 1 \ \ if \ \ t_i \le u \le t_{i+1} \\ 0 \ \ \ \ otherwise \end{cases} \quad \dots\dots \textbf{(11)}$$

$$N_{i,k}(u) = \frac{(u - t_i)}{t_{i-k-1} - t_i} N_{i,k-i}(u) + \frac{(t_{i-k} - u)}{t_{i-k} - t_{i+1}} N_{i,k-i}(u) \quad \dots\dots \textbf{(12)}$$

Where ($P_0, P_1, ..., P_n$) are control points and $N_{i,k}$ are the basis functions defines using the Cox-de Boor recursion formula

The basis function can be written as triangular scheme [10]:

$$
\begin{array}{ccccccc}
N_{0,0}(u) & \rightarrow & N_{0,1}(u) & ... & N_{0,k-1}(u) & \rightarrow & N_{0,k}(u) \\
& \square & & & & \square & \\
N_{1,0}(u) & \rightarrow & N_{1,1}(u) & ... & N_{1,k-1}(u) & & \\
\vdots & & & & & & \\
N_{n-1,0}(u) & \rightarrow & N_{n-1,1}(u) & & & & \\
& \square & & & & & \\
N_{n,0}(u) & & & & & &
\end{array}
$$

.........(13)

Where k controls the degree (k−1) of the resulting polynomial in u and also the continuity of the curve.

$$m=n+k+1 \quad …. \textbf{(14)}$$

Where m the number of knots.

The final equation of the uniform B-Spline curve when k = 4 in the matrix form is [10-12]:

$$
p_i(u) = \frac{1}{6}[u^3 \ u^2 \ u \ 1]
\begin{bmatrix}
-1 & 3 & -3 & 1 \\
3 & -6 & 3 & 0 \\
-3 & 0 & 3 & 1 \\
1 & 4 & 1 & 0
\end{bmatrix}
\begin{bmatrix}
P_{i-1} \\
P_i \\
P_{i+1} \\
P_{i+2}
\end{bmatrix}
\quad ..... \textbf{(15)}
$$

i ∈ [1: n-2]

Example 2

a uniform quadratic B-spline is defined the control point ($P_0 \ P_1 \ P_2$) the

m=k+n+1=2+2+1=5

Knot vector T= { $t_1 \ t_2 \ t_3 \ t_4 \ t_5$ }={0 1 2 3 4 5}

So, the B-spline is defined by

$$p(u) = \sum_{n=0}^{2} N_{i,2}(u) P_i$$

Using Cox-de Boor formula

$$N_{0,0}(u) \rightarrow N_{0,1}(u) \rightarrow N_{0,2}(u)$$

$$\Box \qquad\qquad \Box$$

$$N_{0,1}(u) \rightarrow N_{1,1}(u)$$

$$\Box$$

$$N_{2,0}(u)$$

Evaluating the basic functions for $N_{2,0}(u)$ gives:

$$N_{0,2}(u) = \frac{t-0}{2-0} N_{0,1}(u) + \frac{3-t}{3-1} N_{1,1}(u)$$

$$N_{0,2}(u) = \left[\frac{t-0}{1-0} N_{0,0}(u) + \frac{2-t}{2-1} N_{1,0}(u)\right] + \frac{3-t}{2}\left[\frac{t-1}{2-1} N_{1,0}(u) + \frac{3-t}{3-2} N_{2,0}(u)\right]$$

$$N_{0,2}(u) = \frac{1}{2}\left[t^2 N_{0,0}(u) + t(2-t)N_{1,0}(u) + (3-t)(t-1)N_{1,0}(u) + (3-t)^2 N_{0,2}(u)\right]$$

- When $0 < t < 1$ then $N_{0,0}(u) = 1$; $N_{1,0}(u) = 0$; $N_{2,0}(u) = 0$

$$N_{0,2}(u) = \begin{cases} \dfrac{1}{2}t^2 & if\ 0 \le t \le 1 \\[2mm] \dfrac{1}{2}\left[-2(t-1)^2 + 2(t-1) + 1\right] & if\ 1 \le t \le 2 \\[2mm] \dfrac{1}{2}\left[(t-2)^2 - 2(t-2) + 1\right] & if\ 2 \le t \le 3 \\[2mm] 0 & otherwise \end{cases}$$

Doing similar for $N_{1,2}(u)$ $N_{2,2}(u)$

$$N_{1,2}(u) = \begin{cases} \dfrac{1}{2}(t-1)^2 & if\ 1 \le t \le 2 \\[2mm] \dfrac{1}{2}\left[-2(t-2)^2 + 2(t-2) + 1\right] & if\ 2 \le t \le 3 \\[2mm] \dfrac{1}{2}\left[(t-3)^2 - 2(t-3) + 1\right] & if\ 3 \le t \le 4 \\[2mm] 0 & otherwise \end{cases}$$

$$N_{2,2}(u) = \begin{cases} \dfrac{1}{2}(t-3)^2 & if\ 2 \le t \le 3 \\[2mm] \dfrac{1}{2}\left[-2(t-3)^2 + 2(t-3) + 1\right] & if\ 3 \le t \le 4 \\[2mm] \dfrac{1}{2}\left[(t-4)^2 - 2(t-4) + 1\right] & if\ 4 \le t \le 5 \\[2mm] 0 & otherwise \end{cases}$$
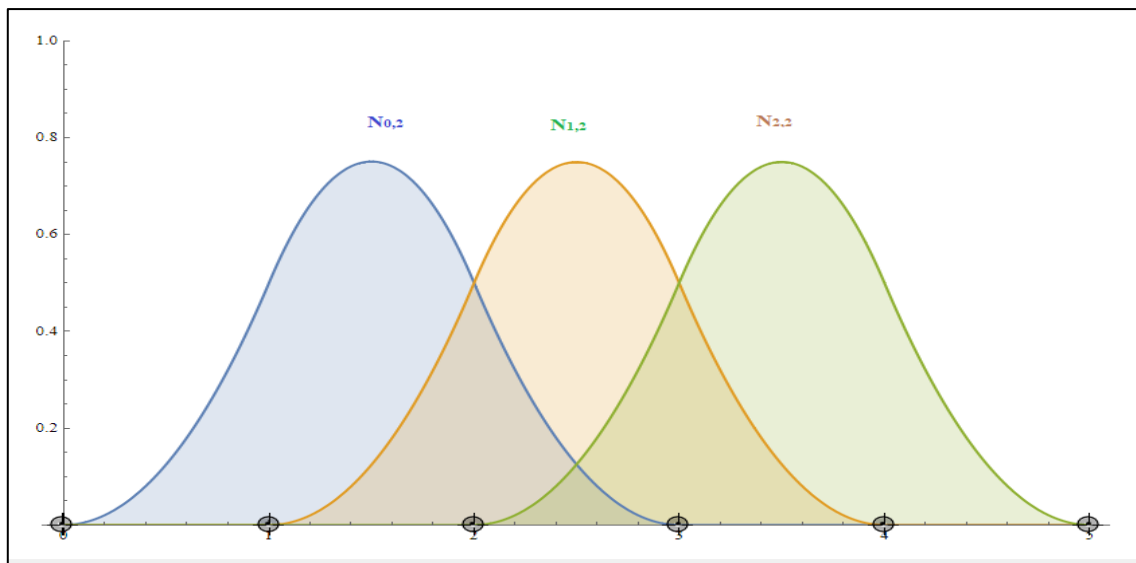
Figure II- 4: basis B-spline curve of degree 3

II.4.2 Geometric properties of B-spline Curves

- Interpolation at the extreme points
- Any B-spline curve starts with a point on the first edge of the control polygon is tangent to it (same for the last point).
- The intermediate points are control points external to the curve.

- The affine transform of a B-spline curve is the curve passing through the transform of the points.

- A B-spline curve belongs to the convex envelope of the points that control it. If it is of order n, it belongs to the convex hull of n-1 consecutive points.

II.4.3 Rational B-spline

Following the same procedure that took us from the Bezier geometries to the rational Bezier, a rational B-spline can be obtained, which in case of curves takes the form [10-11-12]:

$$C(u) = \frac{\sum\limits_{i=0}^{n} B_{i,n}(u) \, . \, w_{i,j} \, . \, P_{i,j}}{\sum\limits_{i=0}^{n} B_{i,p}(u) \, . \, w_{i,j}} \, , \quad 0 \le u \le 1 \quad \ldots\ldots(16)$$

where $B_{i,\,p}$ are the basis functions.

II.4.4 B-spline surfaces

A B-spline has a bi-directional control net, consisting of control points $P_{i,j}$, $0 \le i \le m$, $0 \le j \le n$ The surface has an additional knot vector V in the form:

$$V = \{v_0, \ldots, v_{n+q+1}\} \quad \ldots\ldots \textbf{(17)}$$

where the degree of the surface is p in the u-direction and q in the v-direction. The surface requires two basic functions in the form of (16) and is given by [14]:

$$S(u,v) = \sum_{i=0}^{m}\sum_{j=0}^{n} N_{i,p}(u) N_{j,q}(v) P_{i,j} \quad u_p \le u \le u_{m+1} \, , \, v_q \le v \le v_{n+1} \ \ldots\ldots(18)$$
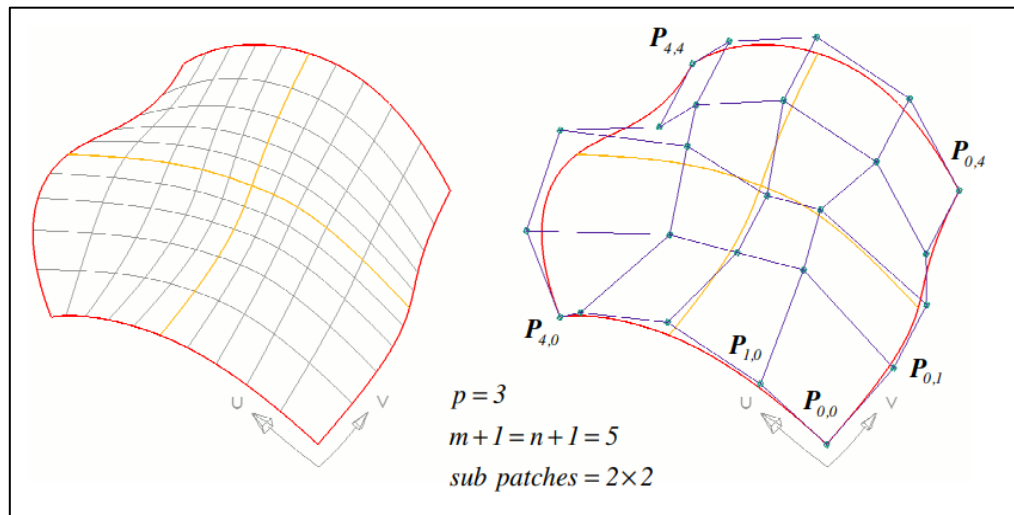
Figure II- 5: B-spline surface with 2x2 sub-patches, and control polygon [14]

II.5 NURBS curve and surfaces

A non-uniform, rational B-spline (NURBS) is defined by its order, a set of weighted control points, and a knot vector. NURBS curves and surfaces are generalizations of both B-splines and Bezier curves and surfaces, the primary difference being the weighting of the control points, which makes NURBS curves rational.

NURBS interpolator has great advantages for free-form surface machining compared to the conventional linear/circular interpolator. However, the existing NURBS interpolators can only handle the NURBS trajectory given in a customized NURBS G code.[15]

II.5.1 Proprieties of a NURBS curve

- Ordre: The order of a NURBS curve defines the number of nearby control points that influence any given point on the curve
- Control Points: The control points determine the shape of the curve.

The parametric equation of NURBS curve that is defined by (m +1) control points, Pi is given by [8-10-13-15]:

$$C(u) = \sum_{i=0}^{n} R_{i,j}(u)P_i \qquad \text{....... (19)}$$

Where the rational B-spline basis function $R_{i,j}$, is defined by:

$$R_{i,j}(u) = \frac{w_i N_{i,j}(u)}{\sum_{i=0}^{n} w_i N_{i,j}(u)} \qquad \text{.......(20)}$$

where k is the rank of the NURBS curve, $N_{i,j}(u)$ are the B-spline basis functions, P_i are the control points, and w_i are the weights. The basic function can be defined by a recursion equation:

$$\begin{cases} N_{i,0}(u) = \begin{cases} 1, & u_i \le u \le u_{i+1} \\ 0, & others \end{cases} \\ N_{i,j}(u) = \dfrac{u - u_i}{u_{i+j} - u_i} N_{i,j-1}(u) + \dfrac{u_{i+j+1} - u}{u_{i+j+1} - u_{i+1}} N_{i+1,j-1}(u) \end{cases} \qquad \text{...... (21)}$$

where k is the power, ui (0, 1, 2,..., m) are knot points, which form the knot vector U

$$u\left[u_0, u_1, \ldots., u_m\right]$$

Obviously, it can only be determined if the rank of the NURBS curve, its control points, knot vector, and weights are determined.

II.5.2 NURBS surfaces

If we extend equation (18) in two parametric directions, we obtain a surface with the same properties as the NURBS curve. This equation is not pleasant to the eye in its mathematical form, but in its rendered [17].

$$S(u,v) = \frac{\sum_{i=0}^{n} \sum_{j=0}^{m} w_{i,j} P_i \, N_{i,m}(u) N_{j,m}(v)}{\sum_{i=0}^{n} \sum_{j=0}^{m} w_{i,j} N_{i,m}(u) N_{j,m}(v)} \qquad 0 \le u, v \le 1 \text{......(22)}$$

By evaluating this equation, we obtain a surface with local control, one knot vector in each parametric direction and with rational properties; virtually anything one could wish for when modeling a surface. It goes without saying that $N_{i,m}$ (u) uses the knot vector u and $N_{j,m}$ (v) the knot vector v.
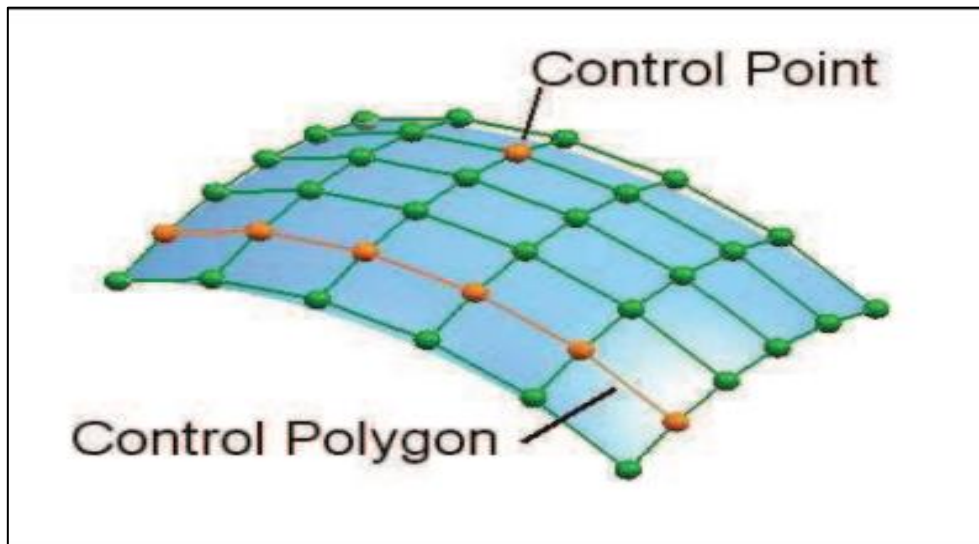
Figure II- 6: NURBS surface [17]

II.5.3 Importance of NURBS

NURBS are widely used in CAD/CAM systems for the following reasons as below [17]:

- The representation of both the standard analytical shapes and free-form shapes

  with the unified common mathematical form

- NURBS provides the flexibility to design the most complex intricate shapes by

  manipulating the control points as well as weights

- It is fast and computationally stable

- They have lucid geometrical representation that are useful for the designer

II.5.4 NURBS formulation

NURBS is used in CNC part programs for its flexibility in shape control. In this section we summarize the main characteristics of the NURBS formulation.[17]
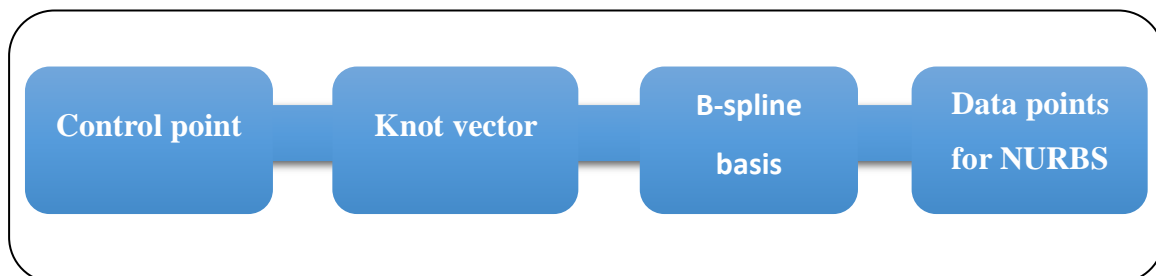
Figure II- 7: Block diagram of NURBS formulation

II.5.5 NURBS Interpolation

In NURBS interpolation, NURBS curve data (e.g., control points, weights, and knot vector) are considered as inputs parameters and they are directly sent to the CNC system where generates the interpolation points based on the NURBS curve data, instead of the small line segment data that are defined by the G01 command. The table II-1 shows the implementation of interpolation based on line segment and NURBS.[17]

Table II- 1: NUBRS interpolation formats by different CNC makers

	Fanuc	Siemens	Okuma	Mitsubishi	Toshiba
CNC model	15 series 16 series 18 series 30 series	840D	OSP700M (Spar H1 CNC)	M700	TOSNUC 888
CPU	64-bit RISC	RISC	RISC		
G-code	G06.2	Language type: BSPLINE	G132	G70.0 G70.1	

II.5.6 Example of NURBS surface in CAD

The (Fig II-8) shown below present an example of NURBS surface in cad using solidworks
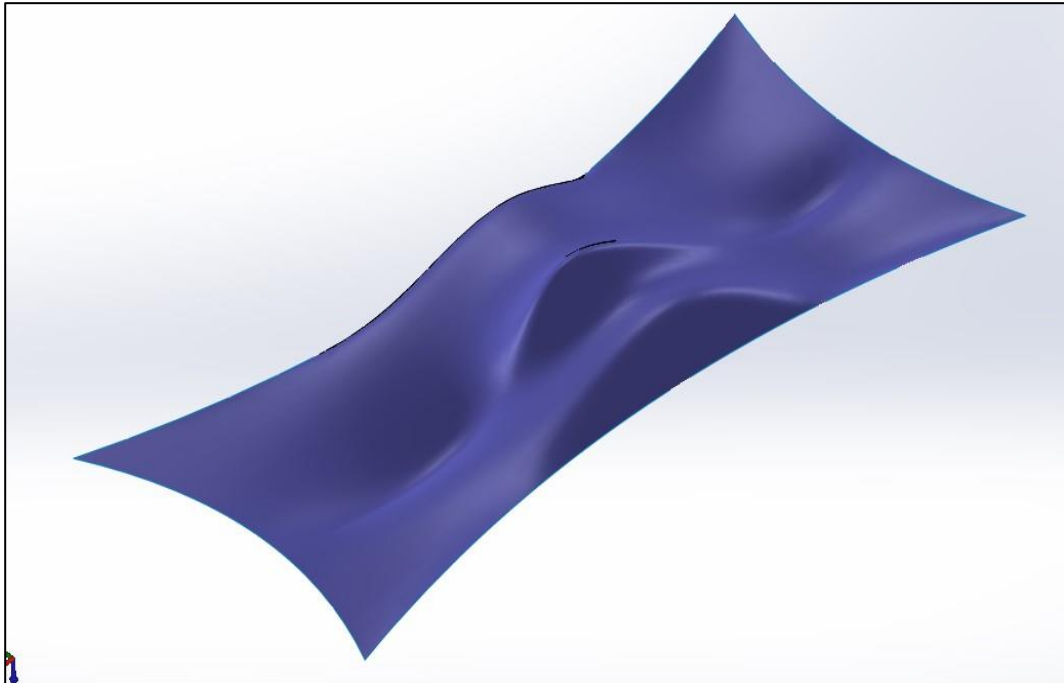
Figure II- 8: NURBS surface in CAD

II.6 Conclusion

In this chapter we have present a general overview of linear interpolation, circular interpolation, also we have presented the fundamental principles of construction of freeform surfaces using Bezier, B-spline, NURBS curves and their roles to improve machining precision on numerical control system.

The interpolator NURBS allow to define complex curves using a reduced number of parameters what also makes it possible to reduce considerably the length of the program of machining, to avoid the decomposition in segments of the courses of tool, and to have advances of the machine spindles without discontinuity.

CHAPTER III
SINUMERIK 840D CYCLES
PROGRAMMING

III.1 Introduction

There are currently several methods of creating NC programs. They are divided into several groups, manual programming, workshop programming and NC programming using CAD/CAM systems. Their main difference is the functions or features they provide.

The most fundamental of these is the control system itself, which allows the creation of NC programs by writing ISO code (G-code) directly or using predefined cycles. Another approach is control using shop floor programming, which simplifies the creation of NC programs through the use of cycles (blocks) and complex graphical assistance. The most complex method of creating NC programs is the CAD/CAM system. These systems support the creation of NC programs for complex parts containing complex 3D surfaces.[18]

In terms of the available cycles and functions, it can be said that the manual programmable control system contains the fewest number of cycles and functions available and can be used for simple components without complex surfaces, with this system making the most demands on the programmer, in addition to knowing the technology and the ISO code. Nevertheless, this method of creating NC programs is used in single or small-lot production of parts, as well as in teaching the basics of programming CNC machines.

III.1 WinNc Sinumerik 840D

III.1.1 definition

WinNC is a software developed by "EMCO Maier GmbH" that provides a simulation environment for CNC programming and training. Sinumerik 840D is a numerical control system developed by Siemens AG for CNC machines.

WinNC Sinumerik 840D is a version of the WinNC software that is specifically designed to work with CNC machines that use the Sinumerik 840D control system. It allows users to program, simulate, and optimize CNC machining processes for a wide range of industrial applications. The software includes a user-friendly interface that makes it easy for operators and technicians to interact with the machine and to fine-tune the programming to achieve the desired results.

Some of the features of WinNC Sinumerik 840D include:

- Simulation of CNC programs in a virtual environment before execution on the machine

- Support for multi-axis machining and complex geometries

- Real-time monitoring and analysis of machine data

- Integration with CAD/CAM software for seamless transfer of machining instructions

- Automatic error detection and correction

- Optimization of machining processes for improved efficiency and productivity.

III.1.2 Screen division

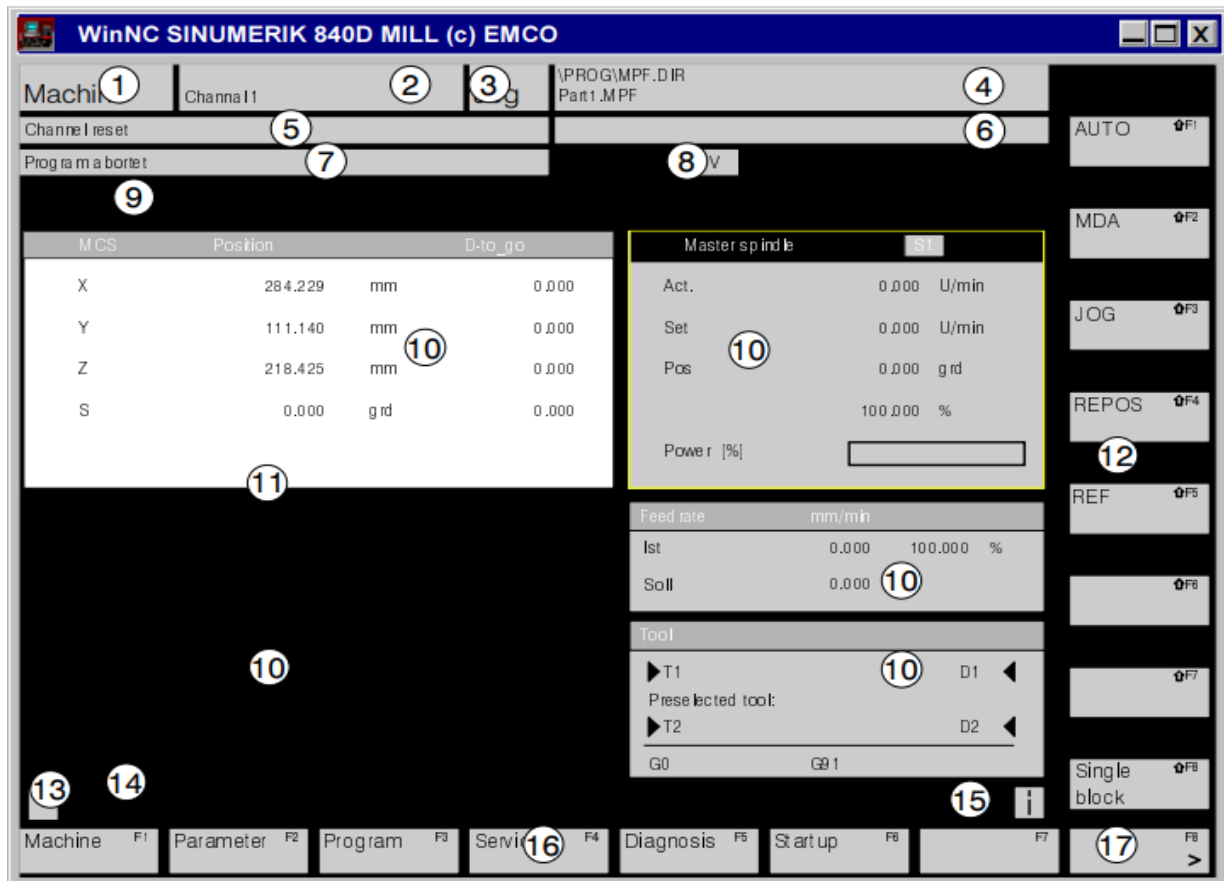(Fig III-1) show screen interface of WinNc sinumerik 840D:

Figure III- 1: WinNc screen division

1. Display of the active operating area

2. Display of the active channel

3. Operating mode

4. Program path and name of the selected program

5. Channel status

6. Channel operating messages

7. Program status

8. Channel status display (SKIP, DRY, SBL…)

9. Alarm and message line

10. Working window (program editor), NC displays (feed, tool) avalaible in the active operating area are displayed here

11. The selected window is marked with a border and the headline is displayed inverted

12. Vertical softkeys

13. When this symbol is displayed, the key $\boxed{\Lambda}$ is active (jump back to superior menu is possible)

14. Dialogue line with operator notes

15. When this symbol is displayed, the key $\boxed{\text{i}}$ is active (information available)

16. Horizontal soft keys

17. When this symbol is displayed, the key $\boxed{>}$ is active (more softkey function available in this line).

III.2 Pre-programmed cycles

Cycles are used to speed up and facilitate repetitive activities. A standard cycle is defined as a subroutine with its own name and parameter list. The pre-programmed cycles in the Sinumerik 840D also include a graphical interface. The graphical interface provides the programmer with sufficient information about the order and properties of the parameters that need to be added for the functionality of the cycle. Subroutines work with user-defined parameters, based on which they perform the required steps. On (Figure III-2) can be seen example of pre-programed cycle for contour roughing [18].
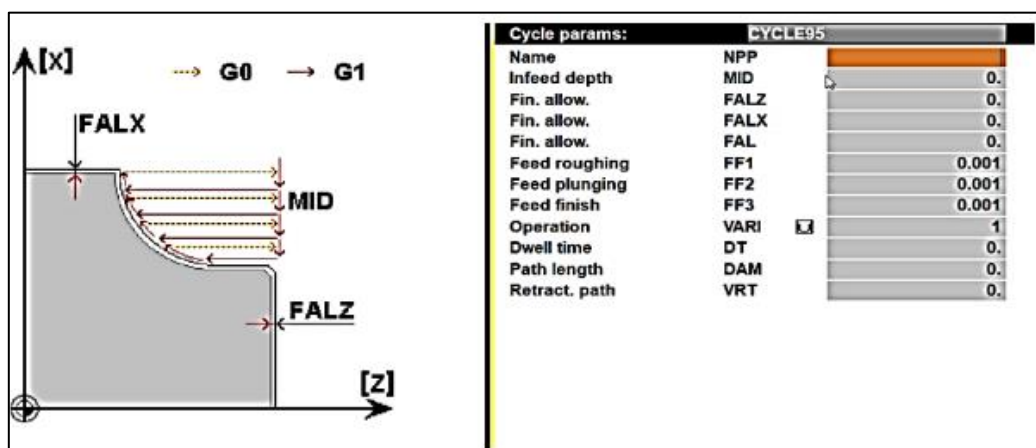
Figure III- 2: pre-programmed cycle example

III.2.1 Milling cycles

The table below present pre-programmed milling cycle

Table III- 1: Milling cycles [19]

Cycle 71	Face milling
Cycle 72	Contour milling
Cycle 90	Thread cutting
Longhole	Longhole on a circle
Slot 1	Slot on a circle
Slot 2	Circular slot
Pocket 1	Rectangular pocket
Pocket 2	Circular pocket
Pocket 3	Rectangular pocket
Pocket 4	Circular pocket
Cycle 76	Rectangular stud milling
Cycle 77	Circular stud milling

III.2.1.1 Cycle 71 face milling

CYCLE71 (RTP, RFP, SDIS, DP, PA, PO, LENG, WID, STA, MID, MIDA, FDP, FALD, FFP1, VARI)

RTP   Retraction plane (absolute)   **Re**traction **P**lane

After the cycle the tool stands at this height. RTP must be higher than the reference level.

RFP   Reference plane (absolute)   **R**eference **P**lane

Height of the workpiece surface, in most cases the workpiece zero point lies on the surface (RFP=0)

SDIS   Safety clearance without sign   **S**afety **Dis**tance

The tool moves in rapid motion up to the height SDIS across the reference level and then changes to working feed

DP   Depth (absolute)   **Dep**th

PA   Starting point, abscissa (absolute)   **P**oint **A**bscissa

PO   Starting point, ordinate (absolute)   **P**oint **O**rdinate

LENG Rectangle length along the 1st axis   **Length**

The angle from which dimensioning is carried out results from the sign.

WID   Rectangle length along the 2nd axis **Width**

The angle from which dimensioning is carried out results from the sign.

STA   Angle between the longitudinal axis of the rectangle and the 1st axis of the plane (enter abscissa without sign)

Value range: $0°$ _STA $< 180°$

MID   Maximum infeed depth (enter without sign) **M**aximal **i**nfeed **D**epth MIDA Maximum infeed width value for clearance in the plane

(Enter without sign)   **M**aximal **i**nfeed **D**epth **an**

FDP   Retraction travel in the plane (incremental, enter without sign)

FALD Final machining allowance in depth (incremental, enter without sign) in machining mode smoothing FALD stands for the residual material on the area.   **F**inishing **A**llowance **D**epth

FFP1 Feedrate for surface machining   **F**eed **F**or **P**lane

VARI Type of machining   **VARI**ant

UNITS DIGIT

Value: 1... Clear up to final machining allowance2... Finishing

TENS DIGIT

Value:

1... parallel to the abscissa, in one direction

2... parallel to the ordinate, in one direction

3... parallel to the abscissa, With changing direction

4... parallel to the ordinate, with changing direction
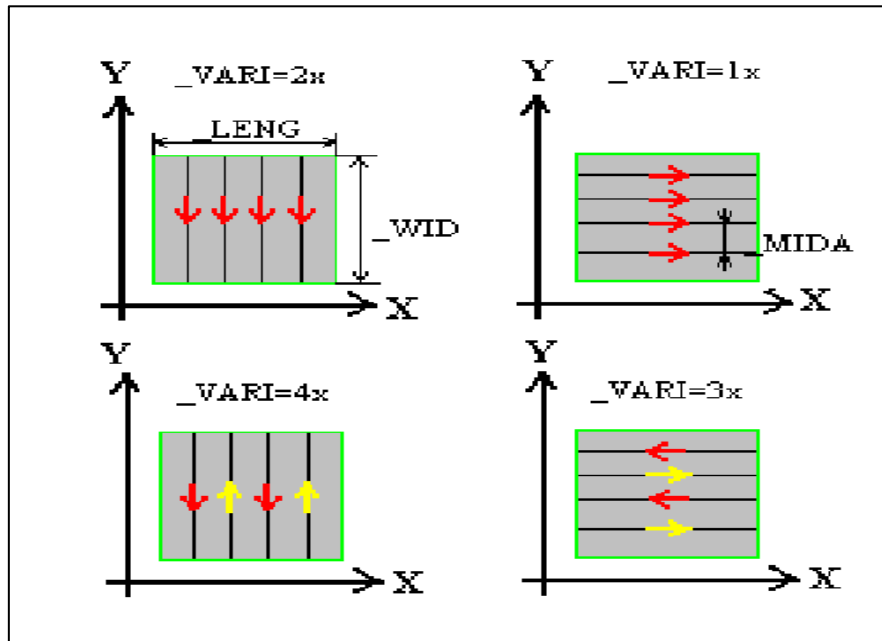
Figure III- 3: cycle 71 paramètres

Example cycle71
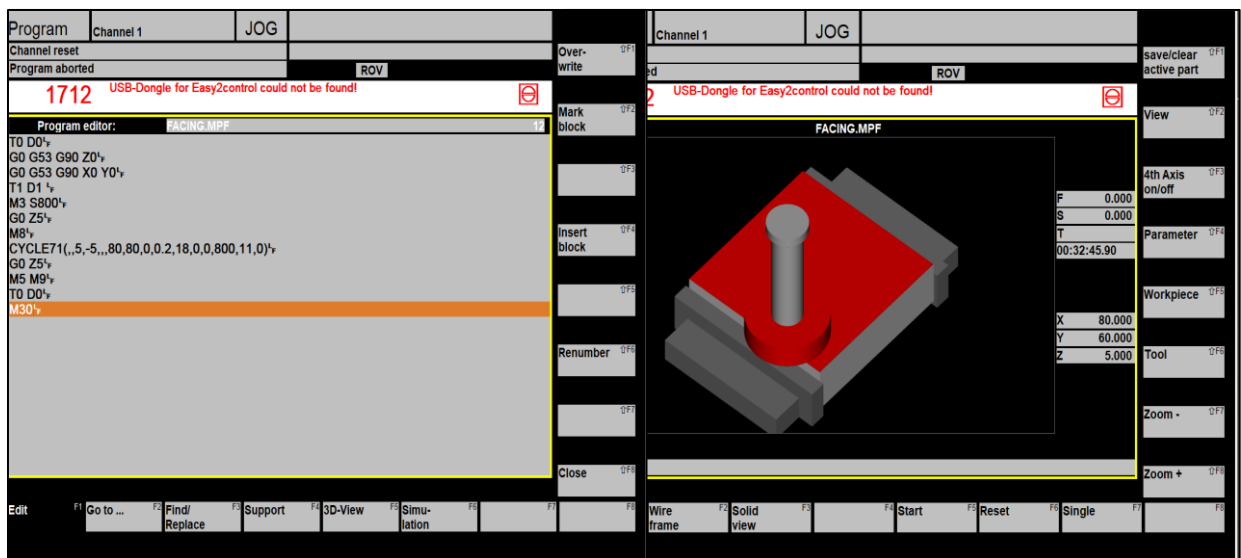
Figure III- 4: cycle 71 example

III.2.1.2 Cycle 72 contour milling

CYCLE72 (KNAME, RTP, RFP, SDIS, DP, MID, FAL, FALD, FFP1, FFD, VARI,RL,AS1, LP1, FF3, AS2, LP2)

Parameter additional to Cycle81:

KNAME Name of contour subroutine

The contour to be milled is completely programmed in a subroutine. With the parameter KNAME you can define the name of the contour subroutine.

FAL   Final machining allowance at the edge contour (enter without sign) FFD Feedrate   for depth infeed (enter without sign)

RL   Following the contour clockwise or counterclockwise with G40, G41 or G42

Value: 40…... G40 (approach and leaving only linear)

       41…... G41

       42…... G42

LP1/LP2 Length, radius

LP1: Length of approach travel (along a straight line) or radius of the millcenter path of the arc approach (along a circle)

LP2: Length of return travel (along a straight line) or radius of the millcenter path of the arc approach (along a circle)

These values must be programmed >0.

FF3 Return feedrate and feedrate for intermediate positioning in the plane. If no feedrate is programmed, the intermediate motions are carried out with surface feedforward for G01.

AS1/AS2 Specification of approach/retraction travel

AS1: approaching travel **AS2:** retraction travel

If AS2 is not programmed, the behavior of retraction is the same so that of the approach

UNITS DIGIT:

Value: 1…. Straight tangential line

       2…. Semi- circle

       3…. Quadrant

Figure III- 5: contouring milling parameter

Example cycle72

Figure III- 6: cycle 72 example

III.2.1.3 Rectangular pocket POCKET1/circular pocket POCKET2

POCKET1(RTP,RFP,SDIS,DP,DPR,LENG,WID,CRAD,CPA,CPO,STA1,FFD,FFP1,MID,CDIR,FAL,VARI,MIDF,FFP2,SSF)

POCKET2(RTP,RFP,SDIS,DP,DPR,PRAD,CPA,CPO,FFD,FFP1,MID,CDIR,FAL,VARI,MIDF,FFP2,SSF)

The tool traverses with rapid feed to the pocket center at level safety distance overreference plane and works off the pocket from center outwards.

Pocket length and width must be larger than the tool diameter, otherwise cycle abortwith alarm.

RTP retraction plane, **RFP** reference plane, **SDIS** safety distance, **DP, DPR** enddepth like previous cycles.

PRAD   pocket radius without sign   **P**ocket **R**adius

LENG   pocket length without sign   **L**ength

WID   pocket width without sign   **W**idth

CRAD   corner radius without sign   **C**orner **Rad**ius

CPA   center point in X   **C**entre **P**oint **A**bscissa

CPO   center point in Y   **C**entre **P**oint **O**rdinateSTA1 pocket angle related to X

FFD   feed for depth infeed **F**eed **F**or **D**epth

FFP1   feed for surface machining   **F**eed **F**or **P**lane

MID   max. infeed depth for cutting **M**aximum **I**nfeed **D**epthwithout sign

CDIR   machining direction   **C**utting **Dir**ection

      2: G2 (clockwise)   3: G3 (counterclockwise)

FAL   finishing offset without sign **F**inishing **A**llowance

VARI   machining variant   **Vari**ant

0: roughing and finishing to final dimension

1: roughing until finishing offset

2: finishing from finishing offset to final dimension
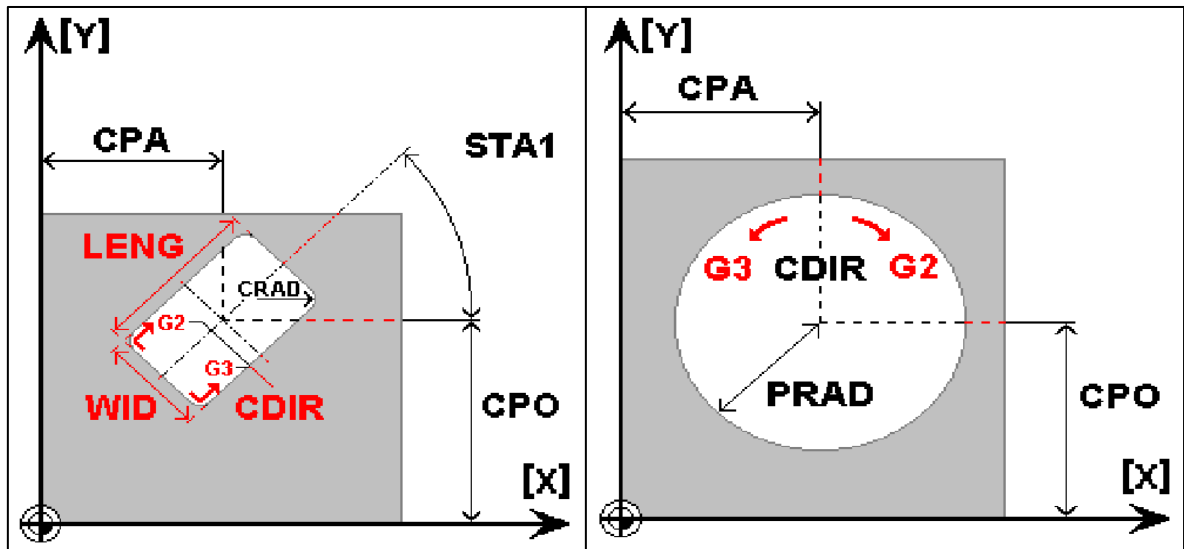
Figure III- 7: Rectangular pocket /circular pocket

Example POCKET2
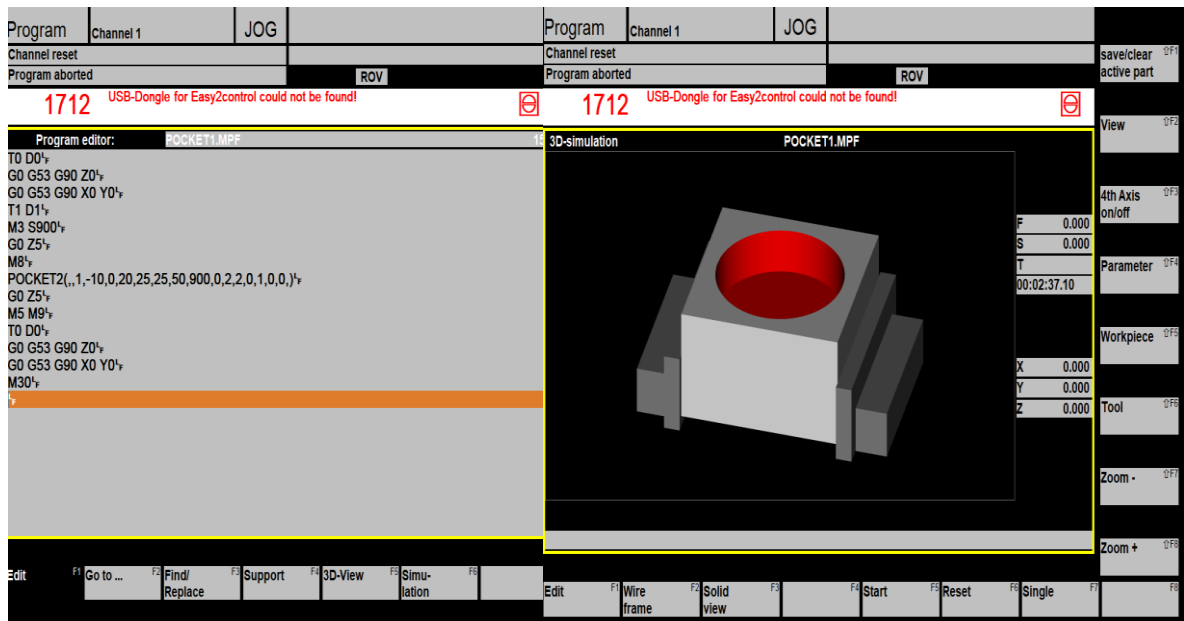
Figure III- 8: circular pocket POCKET2 example

III.1.1.1   Rectangular pocket POCKET3/circular pocket POCKET4

POCKET3(RTP,RFP,SDIS,DP,LENG,WID,CRAD,PA,PO,STA,MID,FAL,FALD,FFP1,FFD,CDIR,VARI,MIDA,AP1,AP2,AD,RAD1,DP1)

POCKET4(RTP,RFP,SDIS,DP,PRAD,PA,PO,MID,FAL,FALD,FFP1,FFD,CDIR,VARI,MIDA,AP1,AD,RAD1,DP1)

These cycles can be applied to roughing and finishing.For finishing is, a face cutter is needed.

The depth infeed will always start at the pocket center point and performed vertically from there¡ thus a pilot drill can be suitable performed in this position.

A tool offset must therefore be programmed before the cycle is called. Otherwise, the cycle is aborted and a alarm is output.

New functions from POCKET3 concerning POCKET 1:

- Milling direction can be defined with a G instruction (G2/G3) or as compared to the spindle direction.
- For solid machining, the maximum infeed width in the plane is programmable.
- Finishing allowance considering the pocket.
- Three different:

-vertically at the pocket center point.

-along the helicoidal path around the pocket center.

-oscillating around the pocket central axis.

- Shorter approach path around the pocket center.
- Consideration of a workpiece contour in the plane and a basic size at the base (optimum processing of pre-formed pockets possible.

New function from POCKET4 concerning POCKET 2:

- Milling direction can be defined with a G instruction (G2/G3) or as compared to the spindle direction.
- For solid machining, the maximum infeed width in the plane is programmable.
- Finishing allowance for the pocket base.
- Two different penetration strategies:

    -vertically from the pocket center point

    -along a helicoidal path around the pocket center.
- Shorter approach path in the plane for finishing.
- Consideration of a workpiece contour in the plane and a basic size at the base (optimum processing of pre-formed pockets possible)
- MIDA is newly calculated during edge machining.
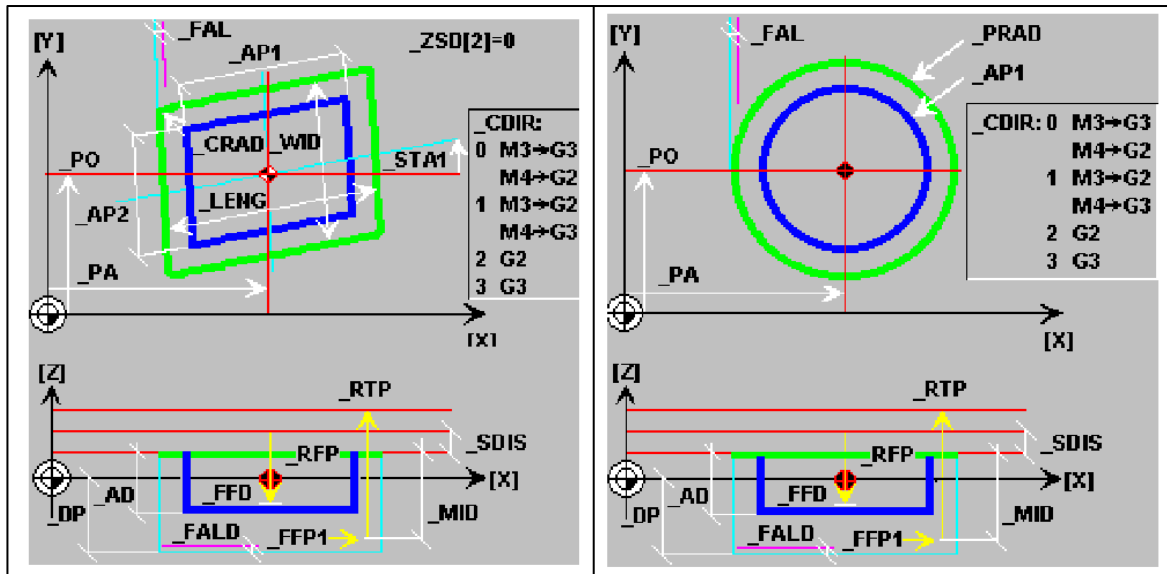
Figure III- 9: Rectangular pocket POCKET3/circular pocket POCKET4

Example POCKET3
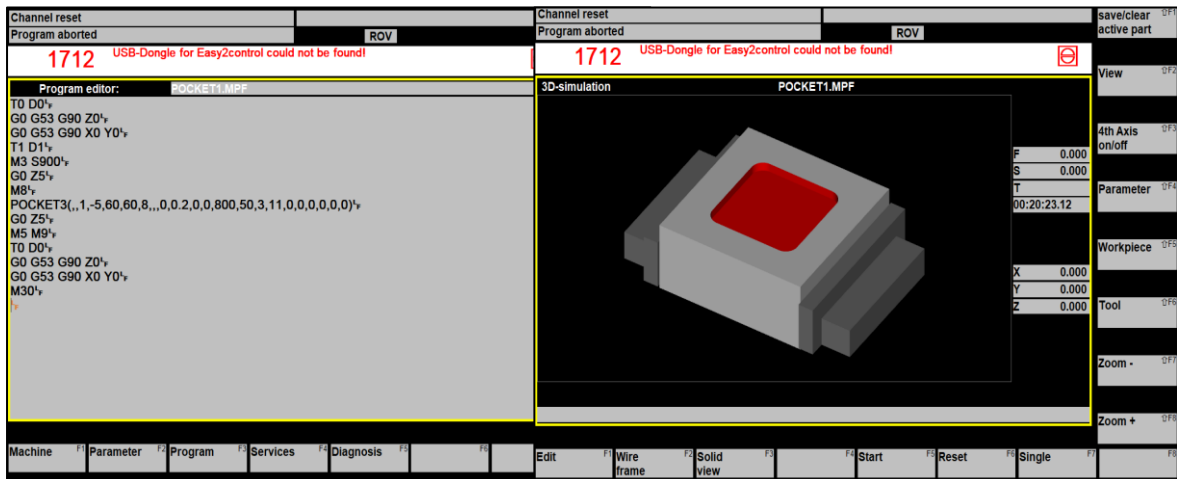
Figure III- 10: rectangular pocket POCKET3 example

III.2.2 Drilling cycle

III.2.2.1 Cycle 81 drilling centering/cycle 82counterboring

CYCLE81 (RTP, RFP, SDIS, DP, DPR)

CYCLE82 (RTP, RFP, SDIS, DP, DPR, DTB)

RTP   retraction plane absolute   **R**eTraction **P**lane

After the cycle the tool is placed on this height. RTP must be higher than the reference plane.

RFP   reference plane absolute   **R**e**F**erence **P**lane

Height of the workpiece surface, most time the workpiece reference point is on the surface (RFP=0)

SDIS   safety distance without sign   **S**afety **DIS**tance

The tool traverses with rapid feed until SDIS over reference plane and then changes to working feed

DP   end depth absolute   **D**e**P**th

Depth of the hole, related to the workpiece reference point.

DPR   end depth relative to reference plane   **D**e**P**th **R**elative

Depth of the hole, related to the reference plane, without sign.

Either DP or DPR can be programmed. If nevertheless both parameters are programmed, DPR will be valid.

DTB   dwell time at hole ground in [s]   **D**well **T**ime at **B**ottom

The tool retracts only after a dwell time to clean the hole ground (only CYCLE82).

Example cycle 81
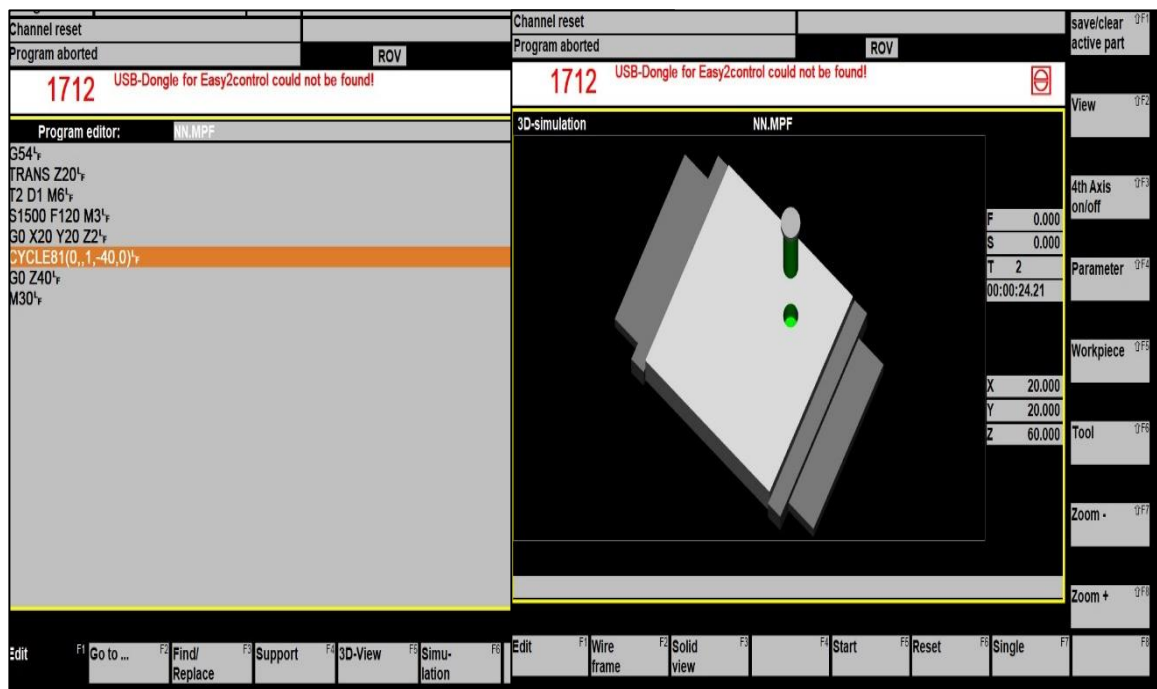
Figure III- 11: drilling cycle example

III.3 case study

in this case we'll try de create a G-code program of a non-twisted blade impeller using cycle milling
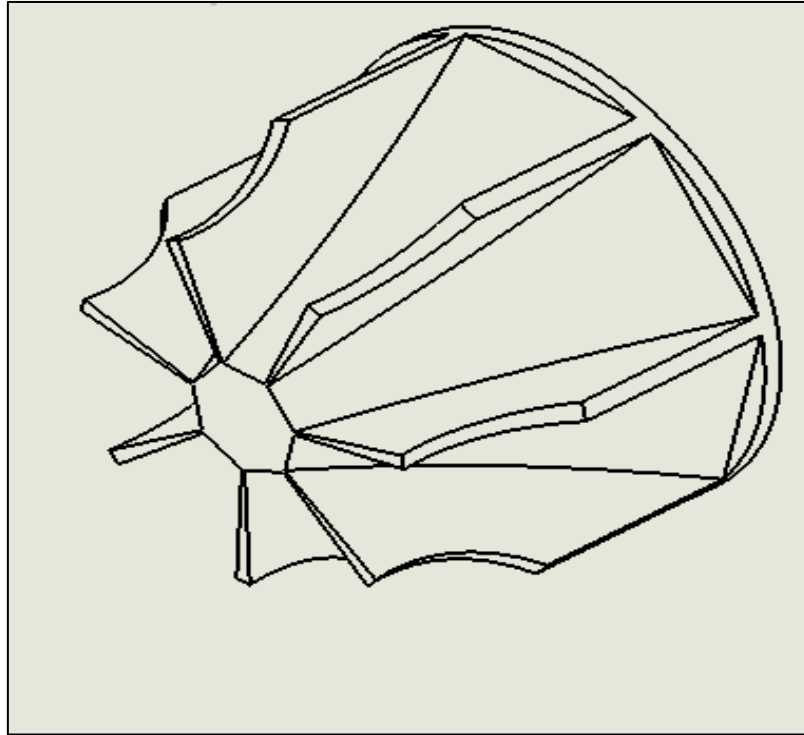
Figure III.12: drawing definition of blade

III.3.1 main and sub programs:

Figure III.13: main and sub program of blade

III.3.2 simulation
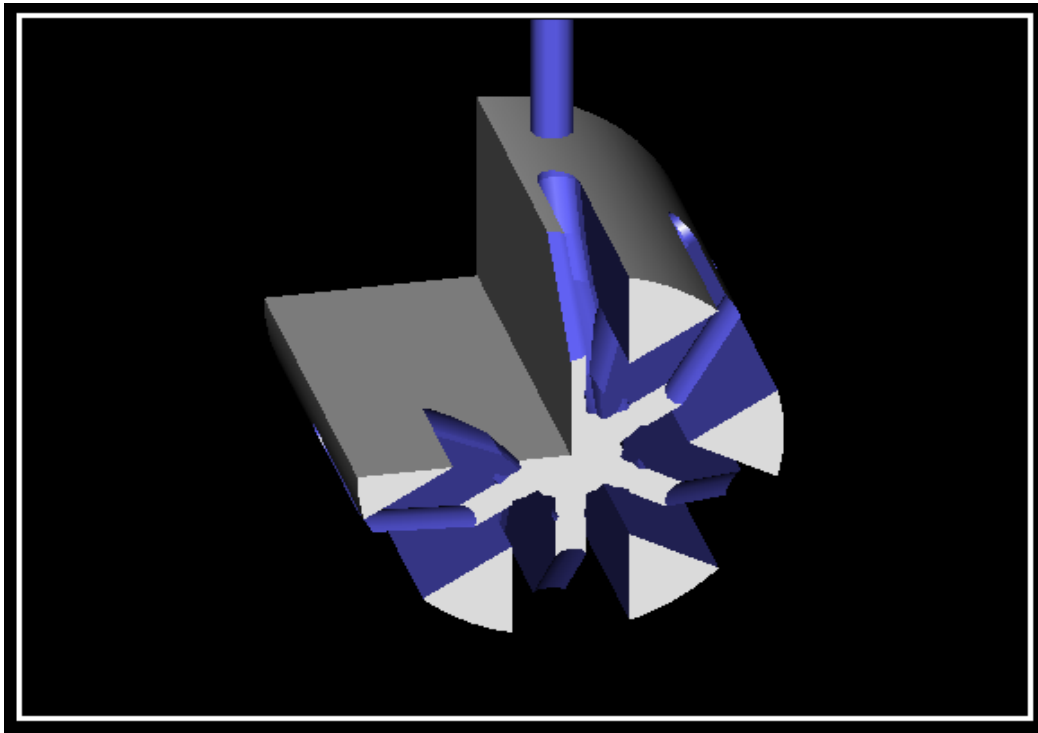
Figure III.14: program simulation

III.3.3 Calculate spindle speed and feedrate

Using table bellow we can detemine the spindle speed & feedrate

- **Spindle speed**

$$N = \frac{1000 V_c}{\pi D}$$

N : spindle speed (tr/min)

V_c : cutting speed (m/min)

D: diameter (mm)

$$N = \frac{1000 \times 70}{3.14 \times 10} = 2230 \, tr / \min$$

- **Feedrate**

$$V_f = f_z \times z \times N$$

f_z : feed per teeth (mm/tooth)

V_f : feedrate (mm/min)

z : number of flutes

$$V_f = 0.1 \times 2 \times 2230 = 446mm / \min$$

Table III- 2: cutting speed & feed per tooth for some materials

material	Cutting speed m/min	Feed per tooth (mm)		
		Face mill	Slab mill	End mill
Aluminum	70-100	0.2-0.8	0.2-0.6	0.1-0.4
brass	35-50	0.15-0.6	0.15-0.5	0.07-0.3
bronze	50-70	0.07-0.3	0.2-0.6	0.-0.4
Cast iron	20-35	0.1-0.4	0.1-0.3	0.05-0.2
Mild steel	30-40	0.1-0.4	0.1-0.3	0.05-0.2
copper	20-30	0.1-0.4	0.1-0.3	0.05-0.2

III.4 Conclusion

In this chapter a presentation of the software Sinumerik 840D by EMCO which's one of most software used on CNC programming with presentation of the pre-programming cycles used in this software with giving examples for it, the aim of these cycles is to optimize and increase effectiveness of NC program preparation, for milling and turning; but those cycles work for simple shapes so a new cycle has been created to machining a non-twisted blade impeller from a cylindrical piece.

CHAPTER IV

impeller
Design and machining

IV.1 Introduction

Automated CNC-machining has been in use for decades to produce plastic, metal, and wood objects. To achieve this, a 3D design program or Computer Assisted Design (CAD) is employed to design the desired object. SolidWorks and CAMWorks are two examples of such computer software. Once designed, the CAD files are transferred to NC code in a process known as computer assisted manufacturing. This results in more precise machining and reduces the need for manual control of automated machine tools.

This chapter is composed of two parts, the first consists the manufacturing of the compressor impeller in the molding phase in this part we will do the design of the mold with the help of SOLIDWORKS software, in the second part we shall do the machining of the piece on CNC machine using CAMWorks software.

IV.2 Mold phase

IV.2.1 SOLIDWORKS presentation

SolidWorks is 3D parametric design software that enables users to create three-dimensional objects ranging from complex multi-component designs to single objects. The term parametric means that the dimensions of the object to be designed can be modified later and therefore the modeled geometry changes.

SolidWorks has three basic functions: parts, assemblies, and drawings. The designer draws the geometry and determines the dimensions of the manufactured object in local section.

Assembly allows designers to assemble individual objects or components by linking them together using linking functionality. Drawings describe a 3D part or assembly as a 2D model and can use ANSI (inches) and ISO (millimeters) standards.

IV.2.2 Obtaining the model using SOLIDWORKS
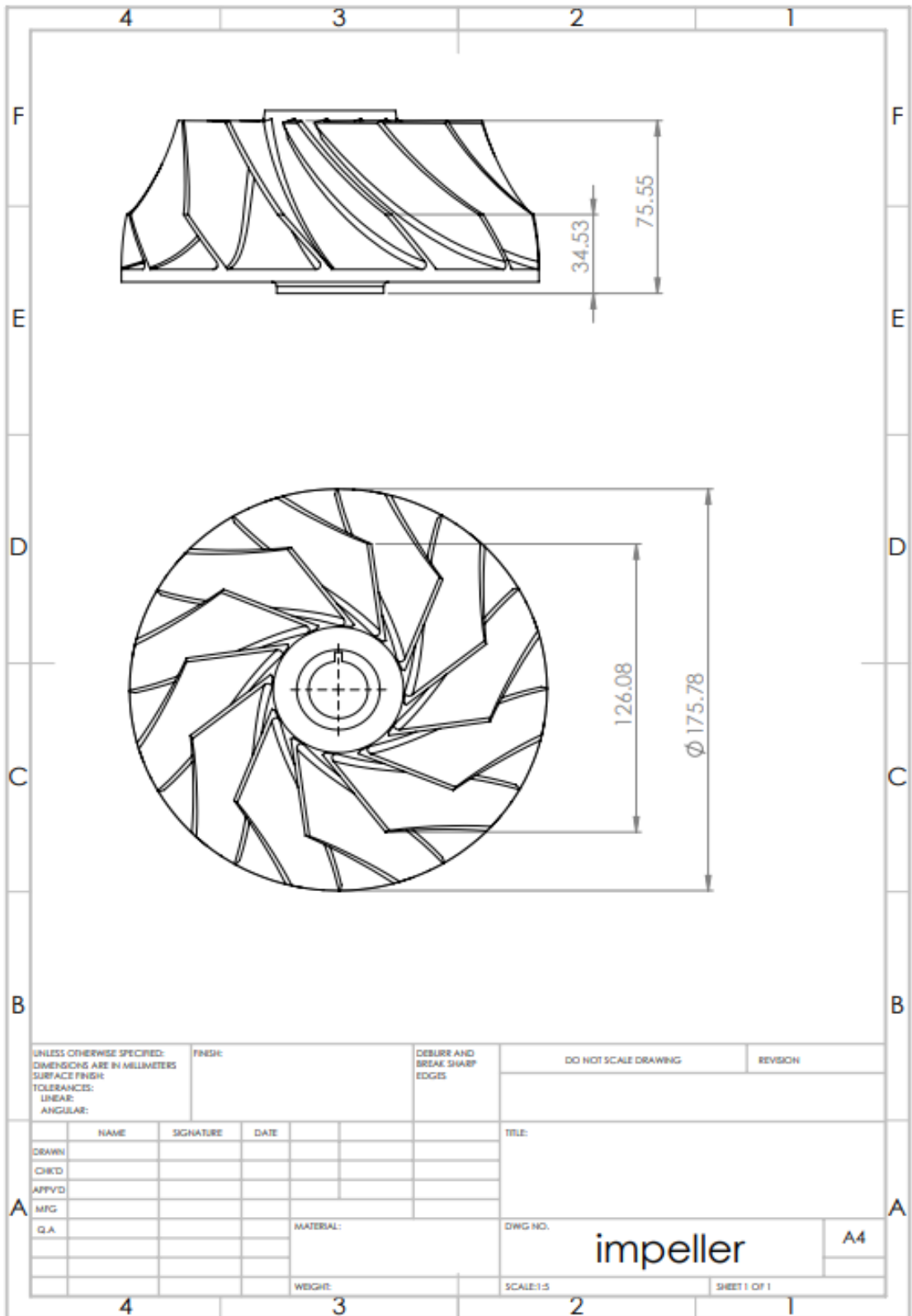
Figure IV- 1: Impeller drawing definition

Our piece is a turbo compressor impeller with 12 twisted blades.[20]

Table IV- 1: impeller characteristic dimensions [21]

Model name	Impeller							
assembly	Turbo-compressor							
material	Aluminum alloy 6061							
Dimensions	Blade number	Outlet diameter	Inlet diameter	Inlet Width	Outlet Width	Blade Angle, β1	Blade Angle, β2	Thick-ness
	12	175.78 (mm)	125.55 (mm)	34.5 (mm)	75.5 (mm)	33°	65°	2 (mm)

steps followed to design the model

1. Select front plane and create sketch
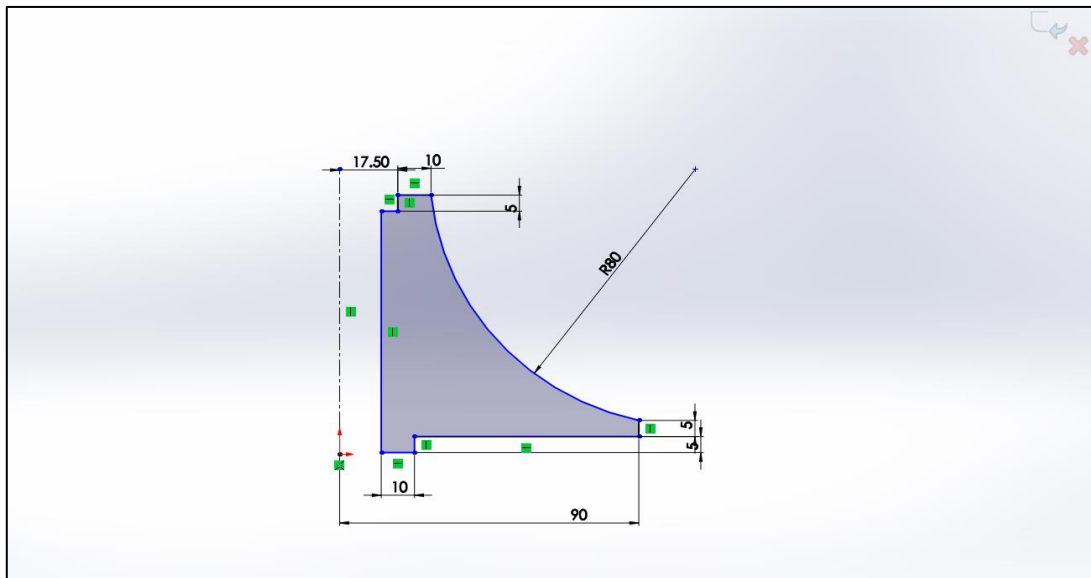
Figure IV- 2: drawing sketch

2. Revolved boss/base
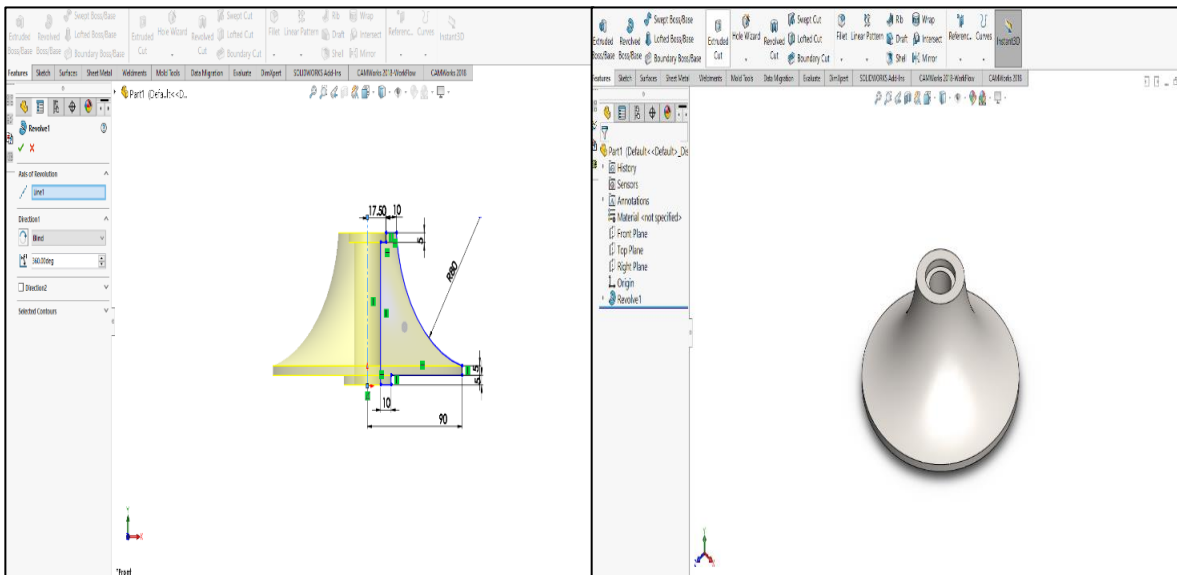
Figure IV- 3: revolved boss

3. Create new reference plane
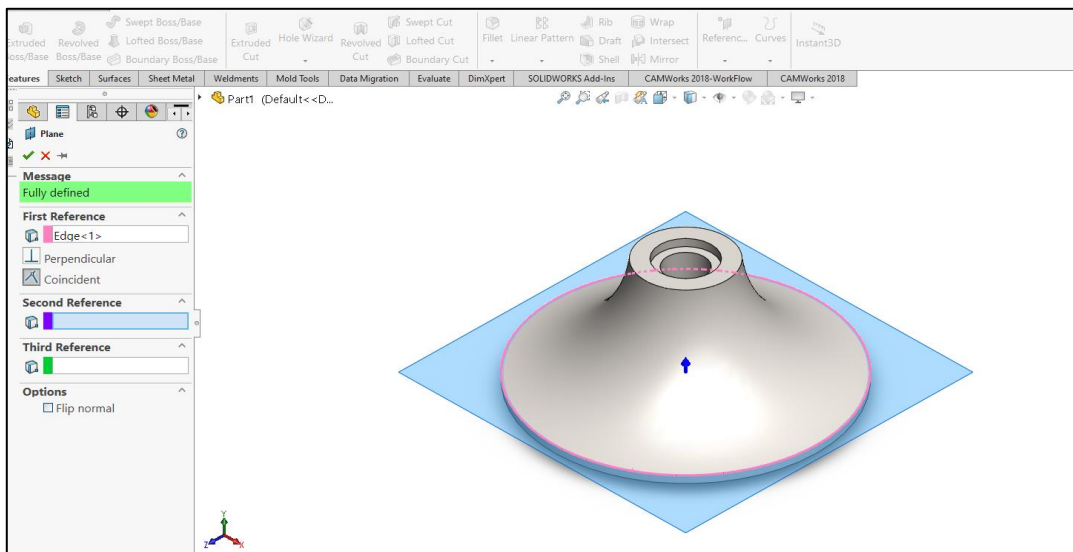
Figure IV- 4: reference plane

4. Select plane1 and create sketch with convert entities
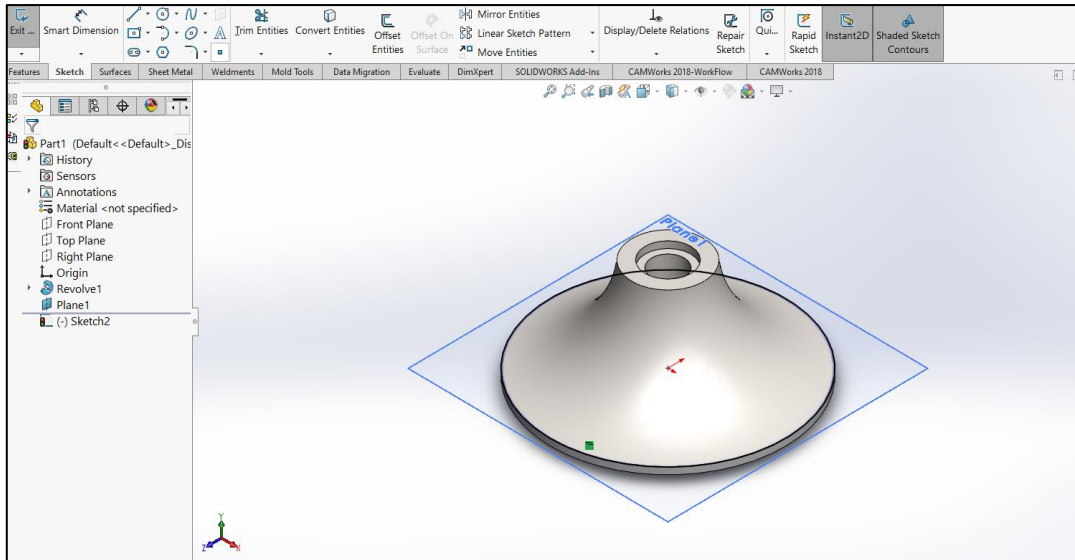
Figure IV- 5: create sketch with convert entities

5. Draw helix
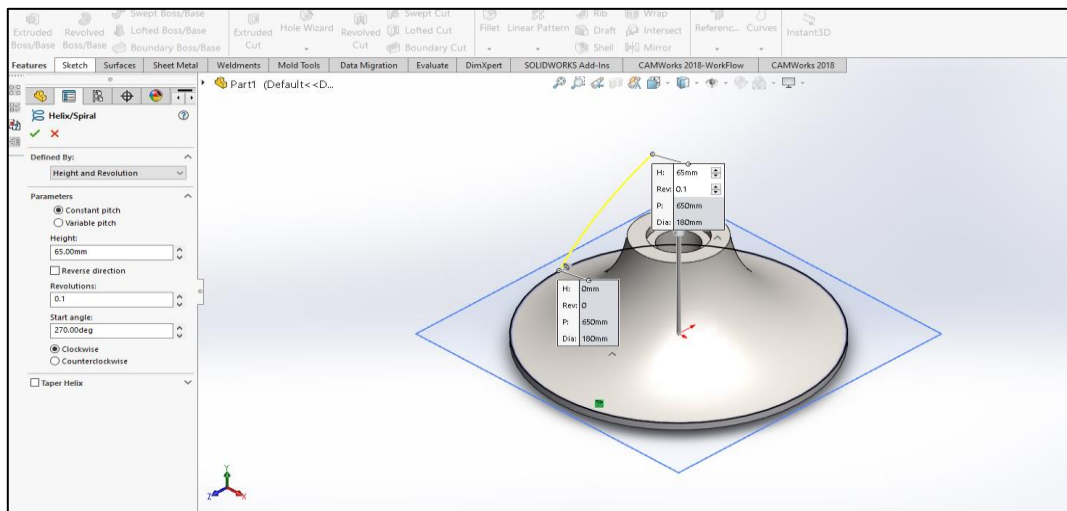
Figure IV- 6: drawing 1st helix

6. Do the same for next sketch
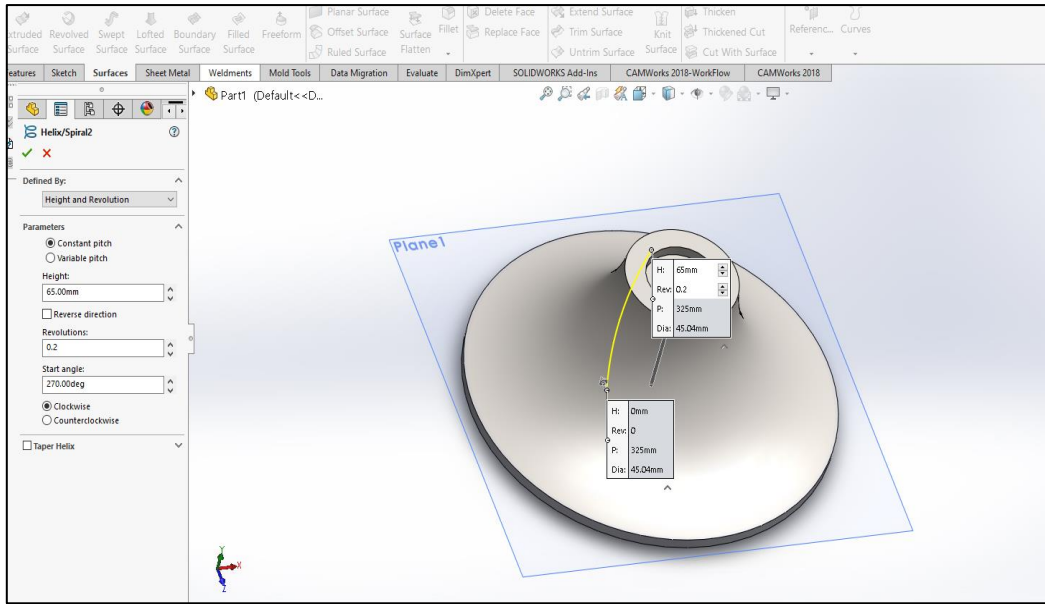
Figure IV- 7: drawing 2nd helix

7. Surface loft
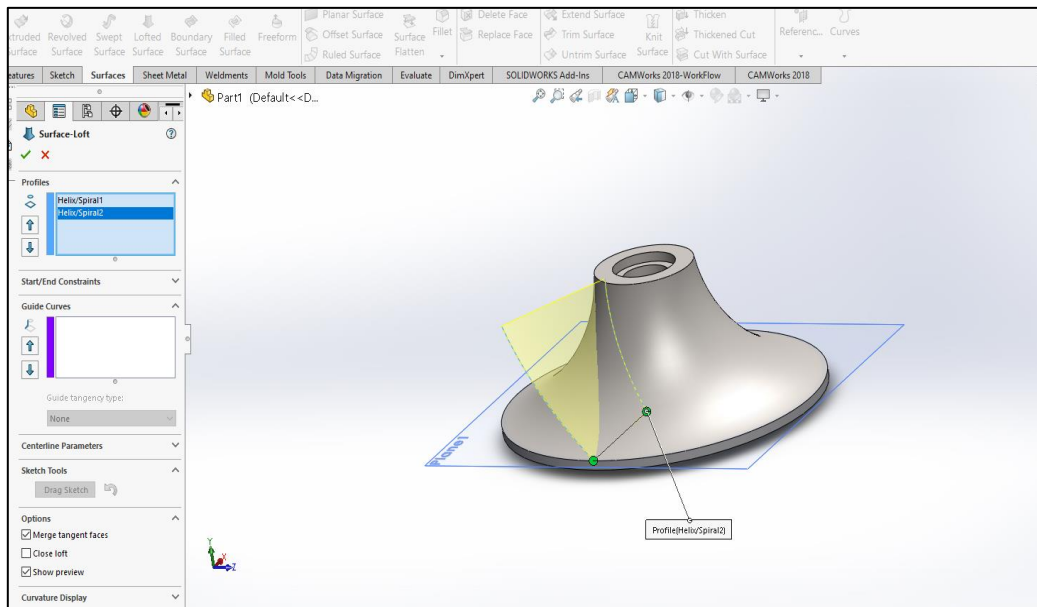
Figure IV- 8: surface loft

8. Thicken
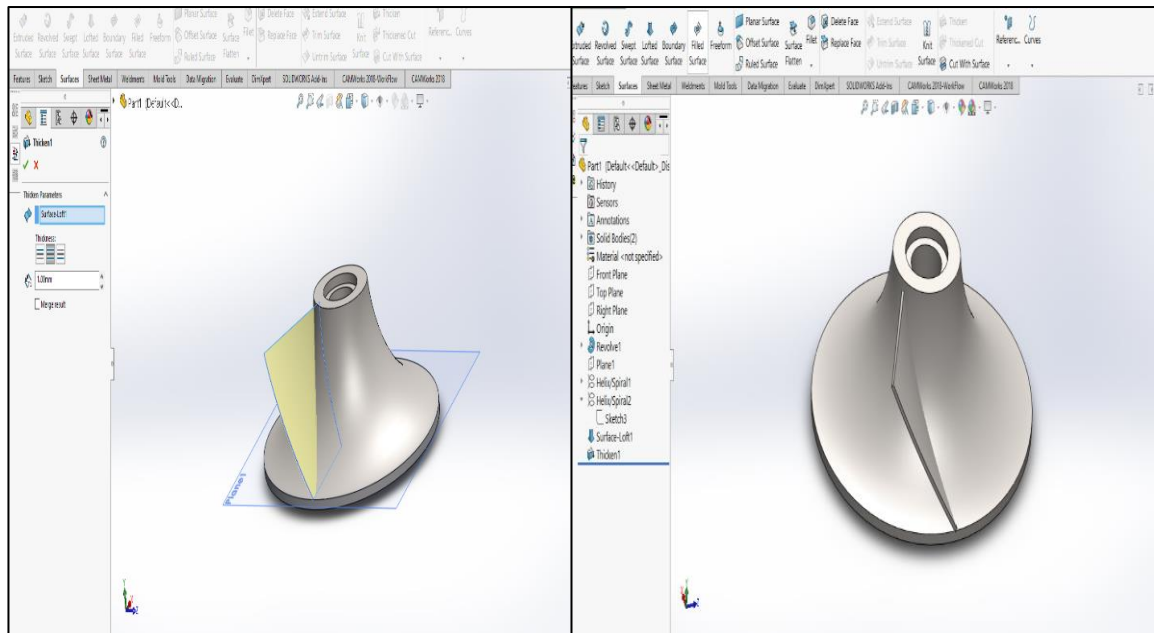
Figure IV- 9: thicken

9. Circular pattern
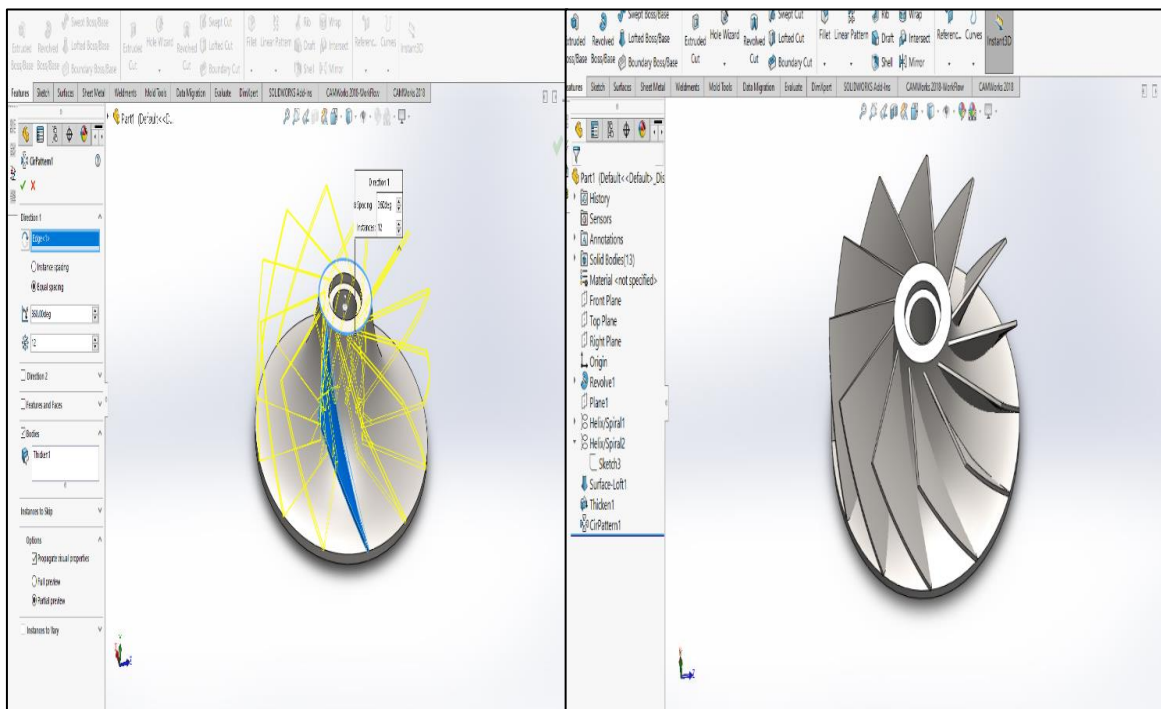
Figure IV- 10: circular pattern

10. Combine bodies
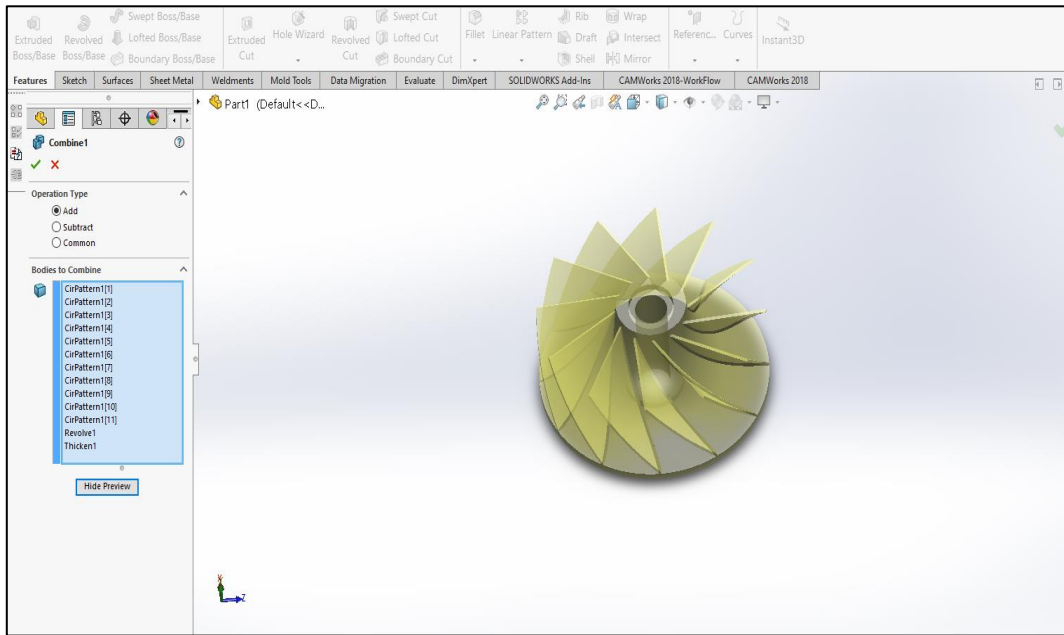
Figure IV- 11: combine bodies

11. Select front plane and draw the sketch
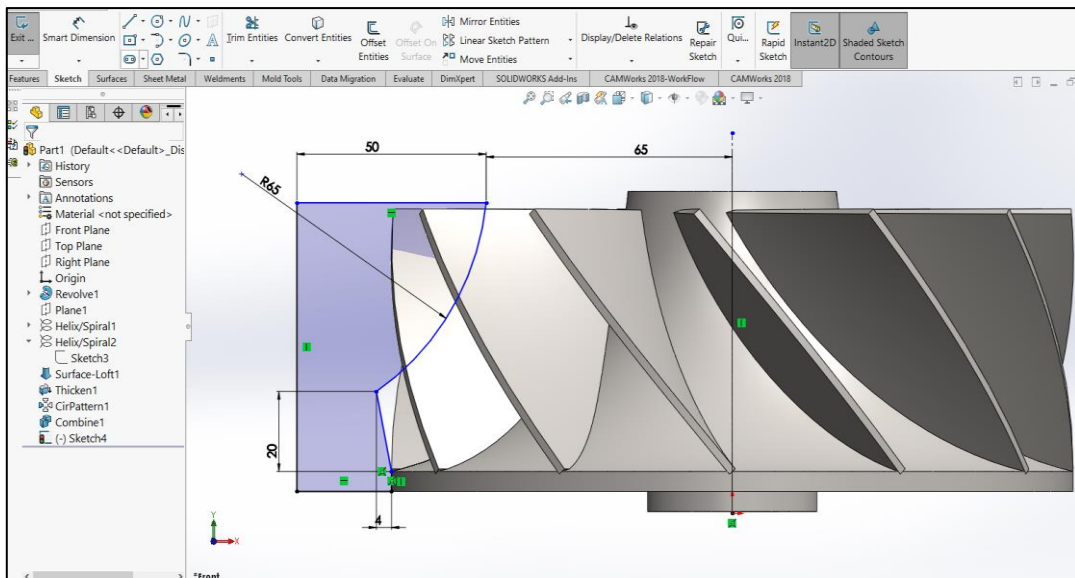
Figure IV- 12: create sketch on blades

12. Revolved cut
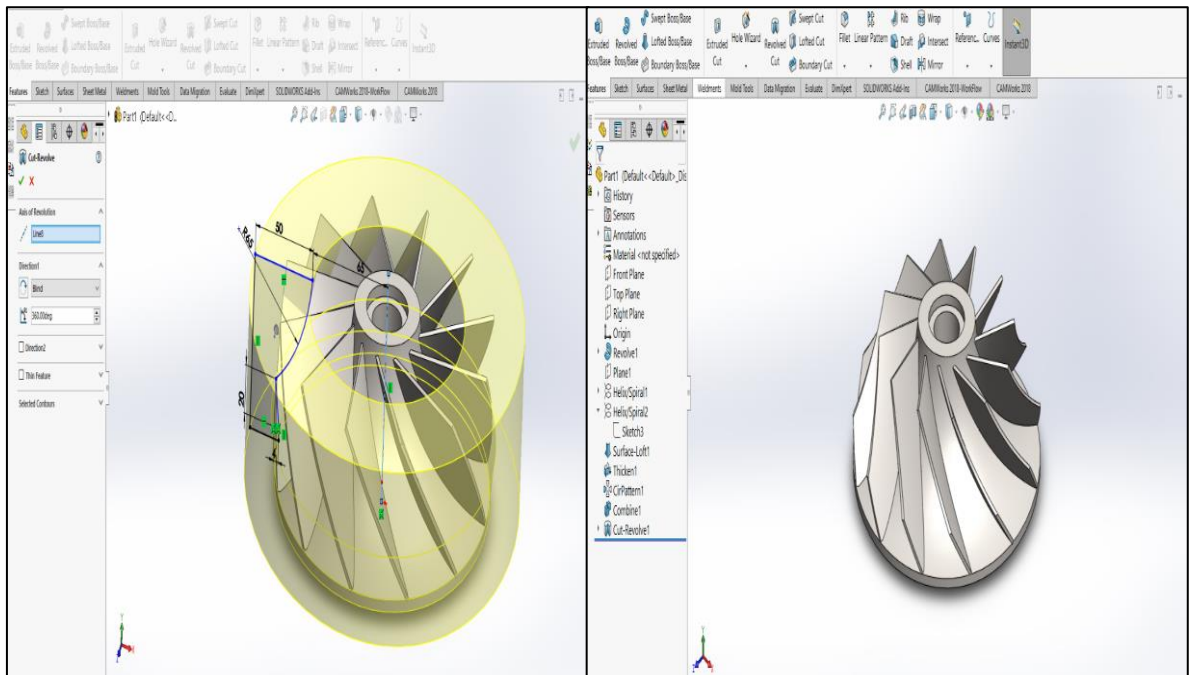
Figure IV- 13: revolved cut

13. Finished piece
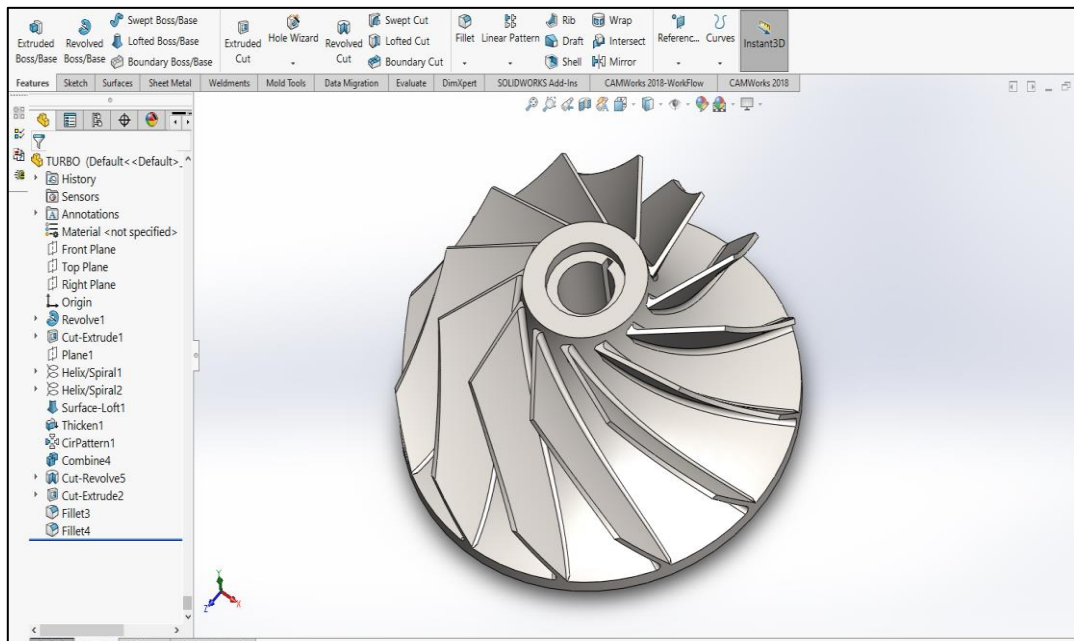
Figure IV- 14: finished piece

IV.2.3 Mold design using SOLIDWORKS mold tools

- Calculate Shrinkage rate

$$Shrink \ rate = \frac{(cavity \ \dim - part \ \dim)}{cavity \ \dim} \times 100\%$$

$$Shrink \ rate = \frac{(200 - 175.78)}{200} \times 100\% = 12.11\%$$

Mold design steps

1. define scale
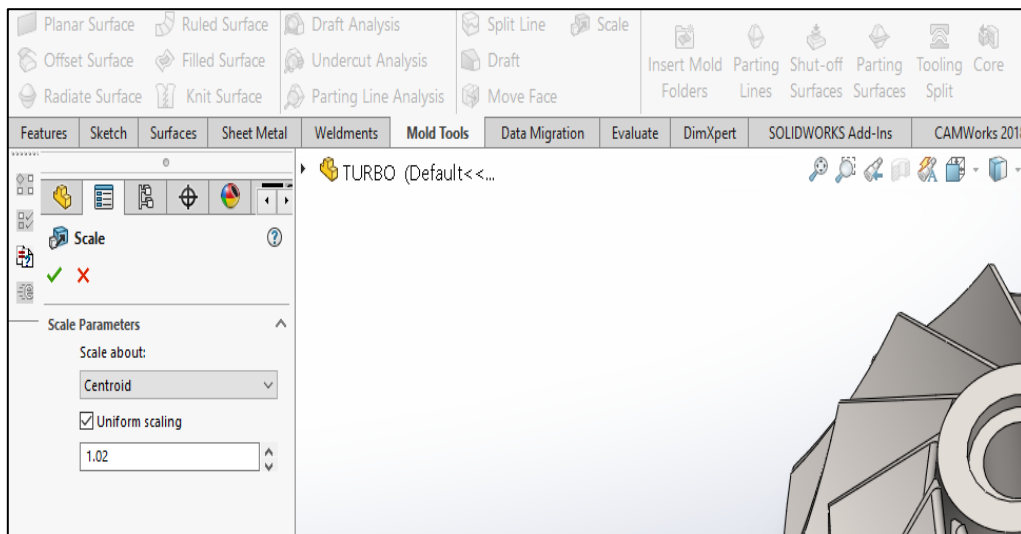
Figure IV- 15: define scale

2. Select parting line
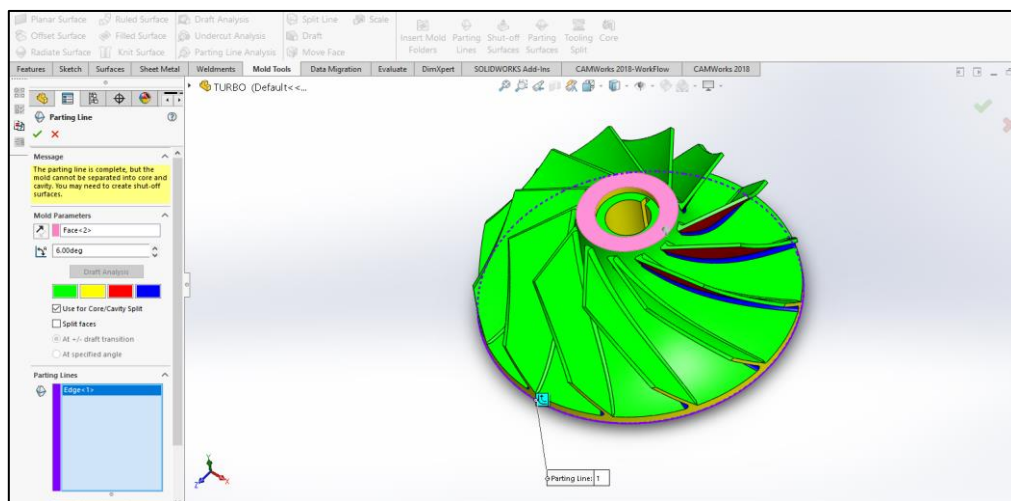
Figure IV- 16: select parting line

3. Shut off surface
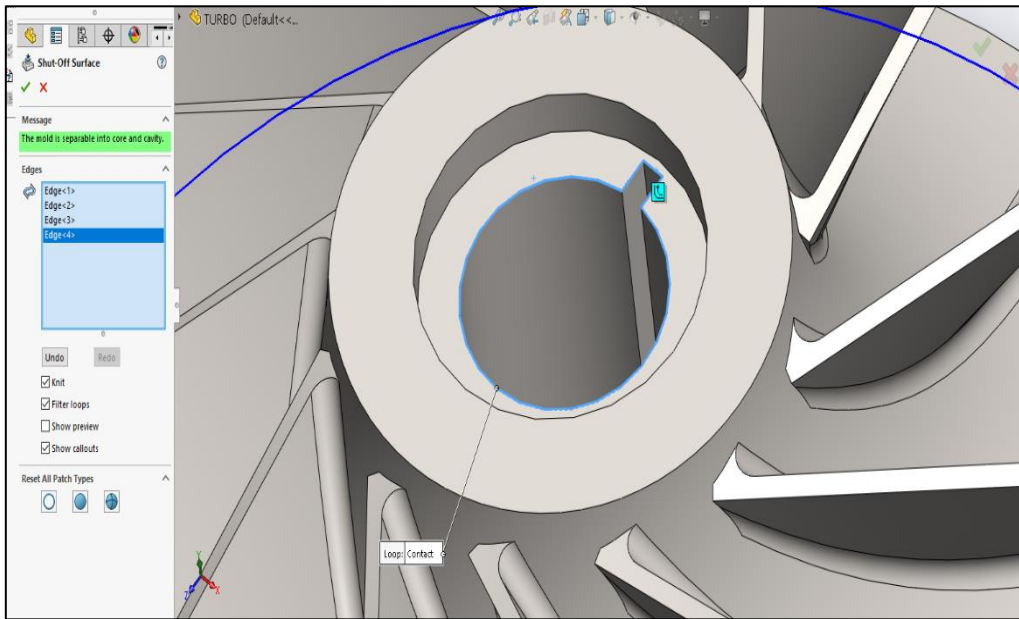
Figure IV- 17: shut-off surface

4. Create parting surface
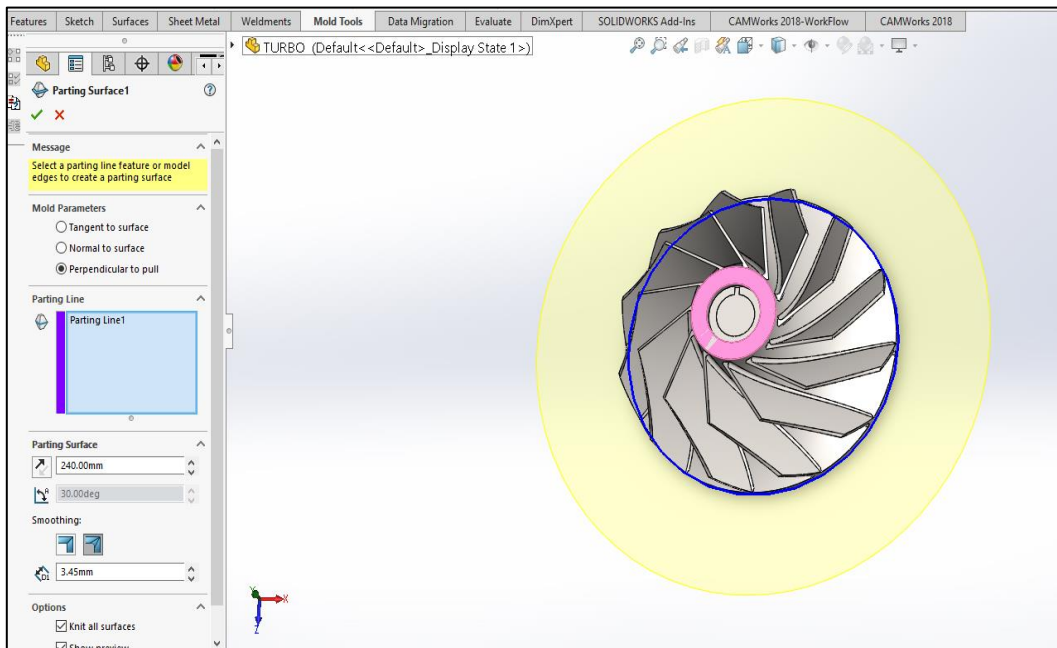
Figure IV- 18: parting surface

5. Create new sketch on parting surface
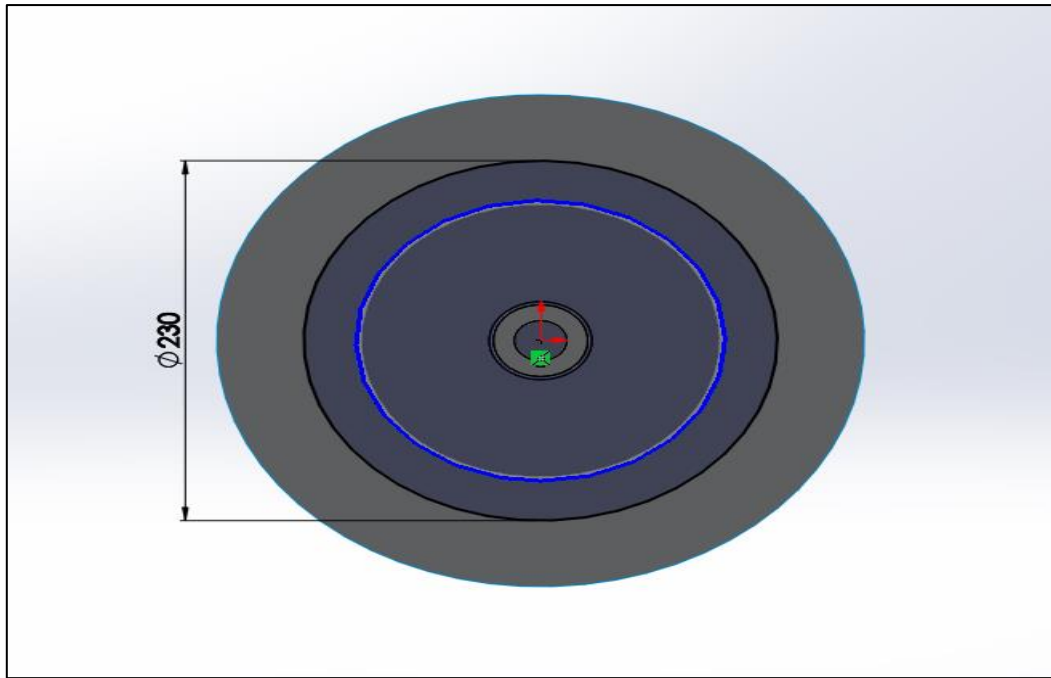
Figure IV- 19: sketch on parting surface

6. Tooling split
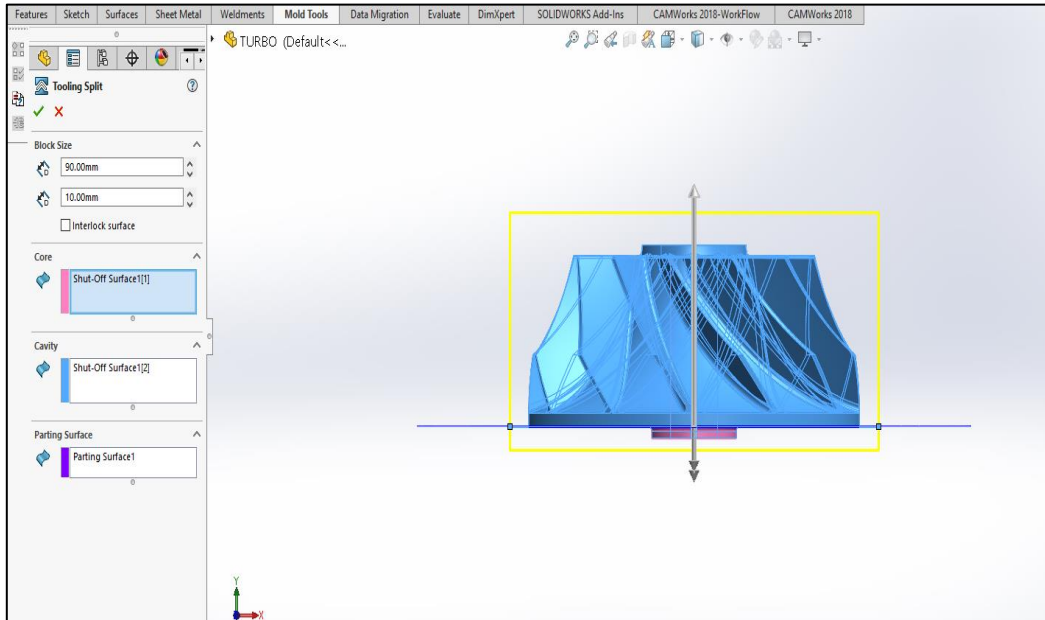
Figure IV- 20: tooling split

7. Obtaining mold core and cavity
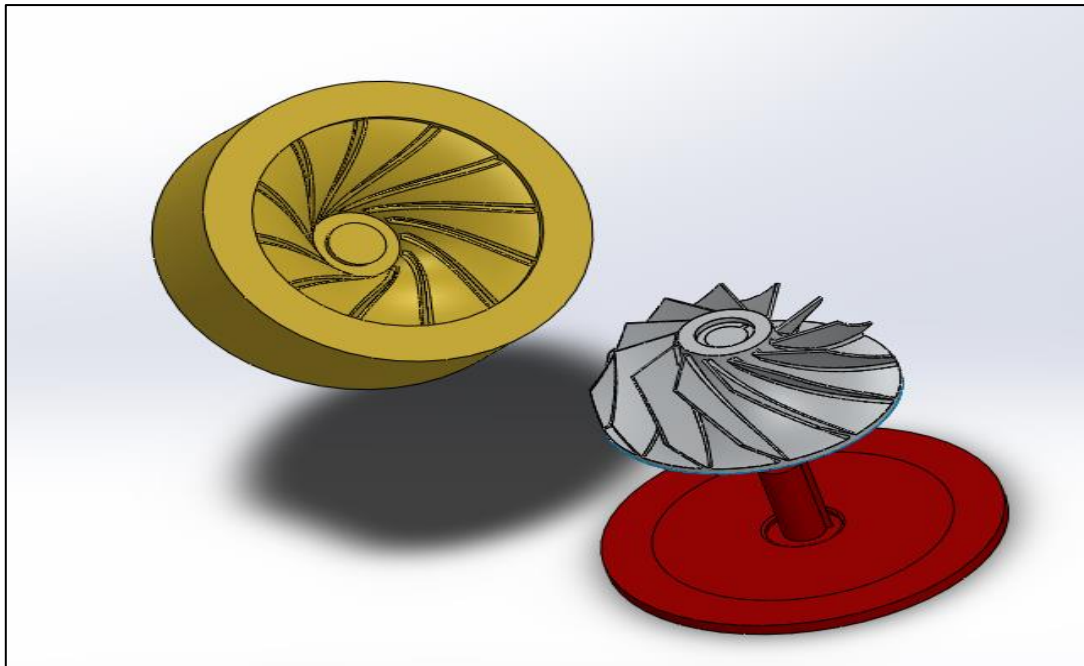
Figure IV- 21: mold core and cavity

8. Add pins to get finished mold
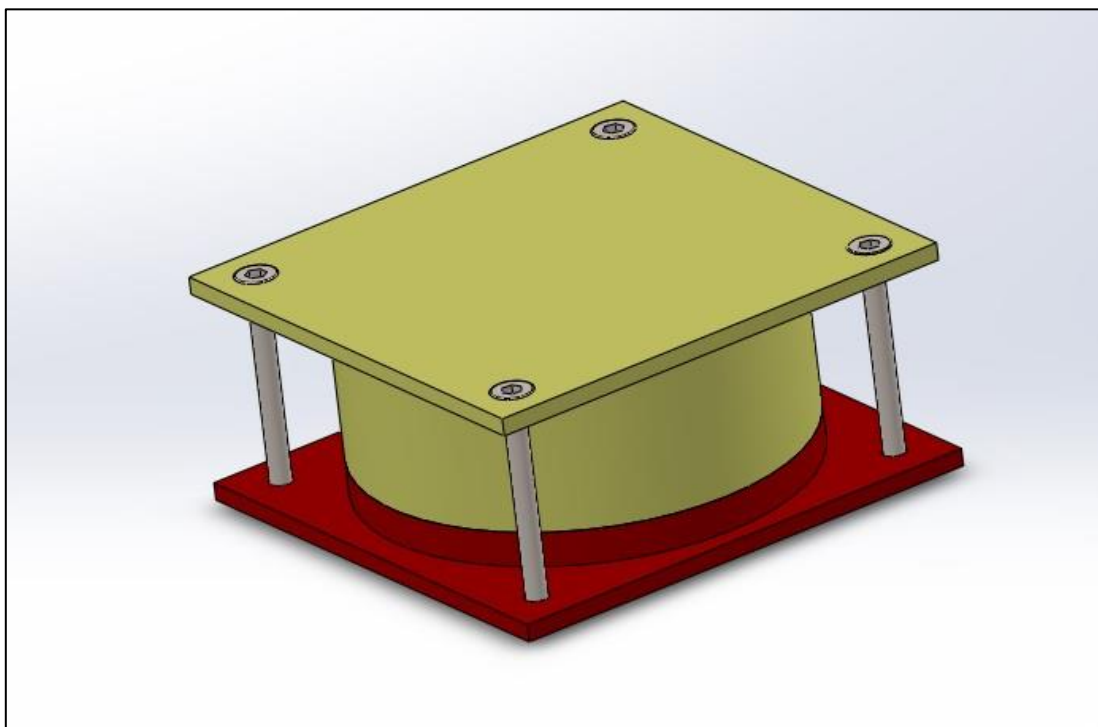
Figure IV- 22: finished mold

VI.2 CNC Machining phase using CAMWORKS

VI.2.1 CAMWORKS presentation

CAMWorks is a computer-aided manufacturing (CAM) software solution developed by Geometric Americas Inc., a subsidiary of HCL Technologies. It is a fully integrated CAM software solution that runs directly inside SOLIDWORKS, a popular 3D CAD software.

CAMWorks uses feature recognition technology to automatically recognize and extract features such as holes, pockets, and slots from a SOLIDWORKS 3D model. These features are then used to create toolpaths for machining operations such as milling, turning, and wire EDM.

CAMWorks also includes a range of additional features such as automatic toolpath verification, post-processing customization, and machine simulation. These features help to streamline the CNC programming process and improve the overall efficiency and quality of the manufacturing process. Overall, CAMWorks is a powerful CAM software solution that can help manufacturers of all sizes to streamline their machining operations, reduce errors, and increase productivity.

VI.2.2 process of using CAMWorks
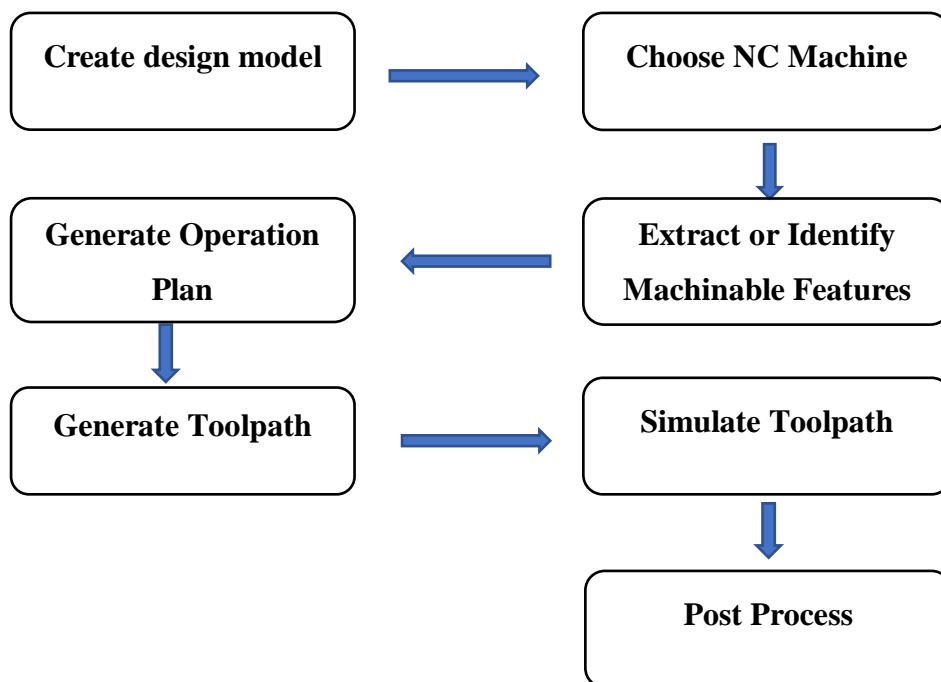
VI.2.3 CAMWorks Machining Modules

The machining modules included in CAMWorks represent a fairly complete set of capabilities in support of virtual machining and toolpath generations. These modules include:
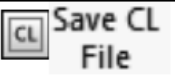
- 2½ axis mill: includes roughing, finishing, thread milling, face milling and single point cycles (drilling, boring, reaming, tapping) to machine prismatic features

- 3 axis mill: includes 2.5 axis capabilities plus strategies to machine complex,

- contoured surfaces encountered in mold making and aerospace applications

- 2 and 4 axis turning: includes roughing, finishing, grooving, threading, cutoff and single point cycles (drilling, boring, reaming and tapping)

- Mill-turn: includes milling and turning capabilities for multitasking machine centers

- Multiaxis machining: 4 axis and 5 axis machining, including high-performance automotive part finishing, impellers, turbine blades, cutting tools, 5 axis trimming, and undercut machining in mold and die making

- Wire EDM: 2.5 axis and 4 axis cutting operations automate the creation of rough, skim and tab cuts.

VI.2.4 The major command buttons in CAMWorks

Table below present the major commands used in Camworks:

Table IV- 2: major command buttons in CAMWorks

Button Symbol	Name	Function
	Define Machine	Allows you to define the machine tool that the part will be machined on, such as 3-axis mill.
	Define Coordinate System	Allows you to define a coordinate system and assign it as the Fixture Coordinate System for the active machine
	Stock Manger	Allows you to define the raw stock from a bounding box, an extruded sketch or an STL file.

Extract Machinable Features	Extract Machinable Features	Initiates automatic feature recognition (AFR) to automatically extract solid features that correspond to the machinable features defined in the technology database (TechDBTM). The types of machinable features recognized for mill and turn are different. CAMWorks determines the types of features to recognize based on the NC machine selected. The machinable features extracted are listed in the feature manager window under the CAMWorks feature tree tab
Generate Operation Plan	Generate Operation Plan	Generates operation plans automatically for the selected machinable features. The operation plans and associated machining strategy and machining parameters are selected based on rules defined in TechDBTM. An operation plan contains information on how the machinable features are to be machined. The operations generated are listed in the feature manager window under the CAMWorks operation tree tab
Generate Toolpath	Generate Toolpath	Creates toolpath for the selected operation plans and displays the toolpath on the part. A toolpath is a cutting entity (line, circle, arc, etc.) created by a cutting cycle that defines tool motion
Simulate Toolpath	Simulate Toolpath	Provides a visual verification of the machining process for the current part by simulating the tool motion and the material removal process
Step Thru Toolpath	Step Through Toolpath	Allows you to view toolpath movements either one movement at a time, a specified number of movements or all movements
Save CL File	Save CL File	Allows you to save the current operation and associated parameters in the technology database as CL (cutter location) data for future use

G1 Post Process	Post Process	Translates toolpath and operation information into G-code for a specific machine tool controller

VI.2.5 Impeller machining with Camworks

1. Define machine, stock, coordinate system and create Mill part setup
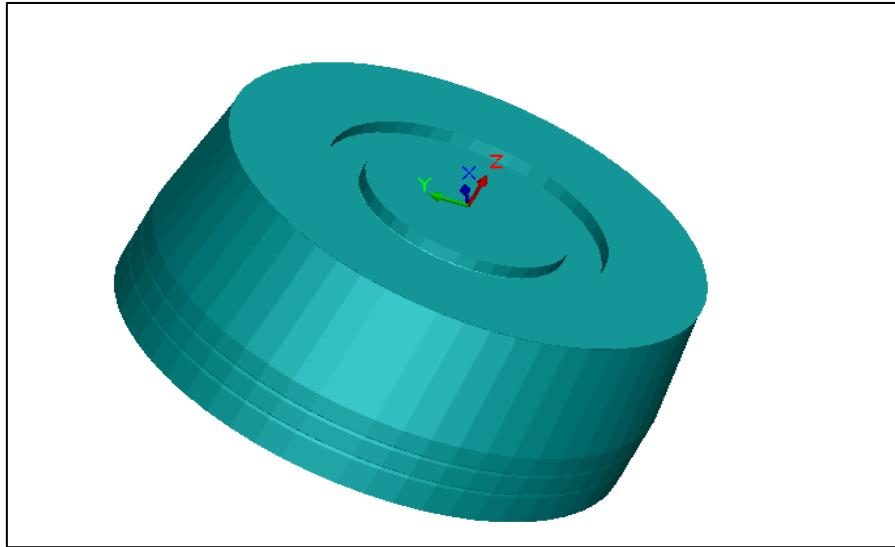
    - Stock manager obtained by Turning operation

Figure IV- 23: stock manager

2. Create new multi surface feature

    - under Mill part setup1, create a new multi surface feature

    - For the strategy, select 5 Axis Impeller
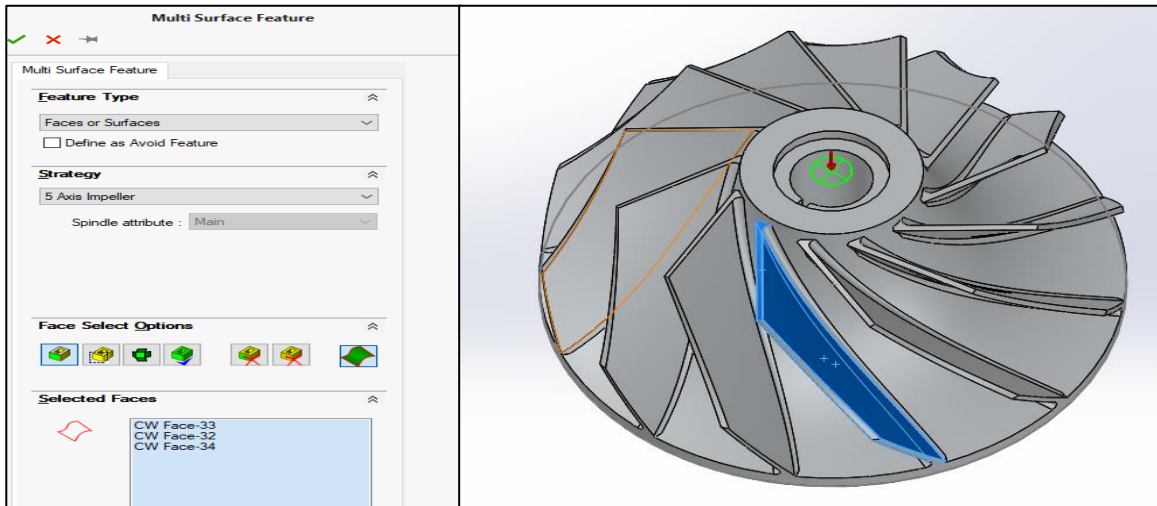
    - select the blade faces

Figure IV- 24: multi surface feature

3. Generate operation plan

4. Select tool

- Click on the CAMWorks tool tab

- Drag Multiaxis Mill1 from T09 – 12.00 Ball Nose to T08 – 10.00 Ball Nose

5. Edit Operation Parameters – Pattern Tab

- Edit the operation parameters of Multi Axis1 and edit the following parameters on the Pattern Tab

- Under Pattern, set the Pattern to Offset from Surface
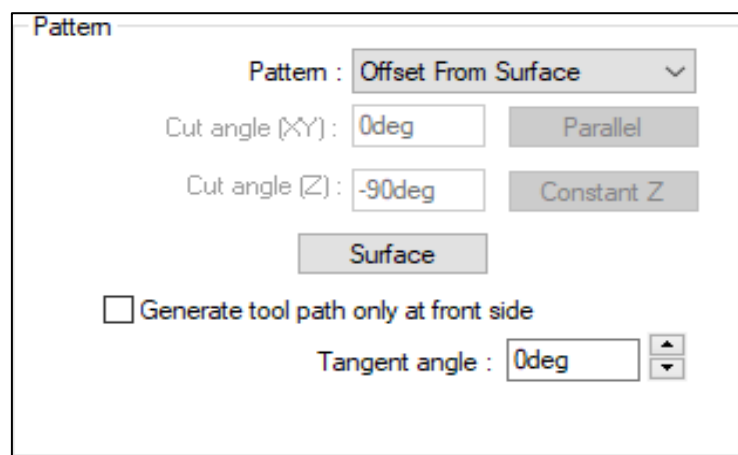
- Click on the Surface button

Figure IV- 25: pattern tab

- Create a new multi surface feature by clicking the Create Features… button

- Under the Selected Faces select the fillet faces between the blade and the floor as shown
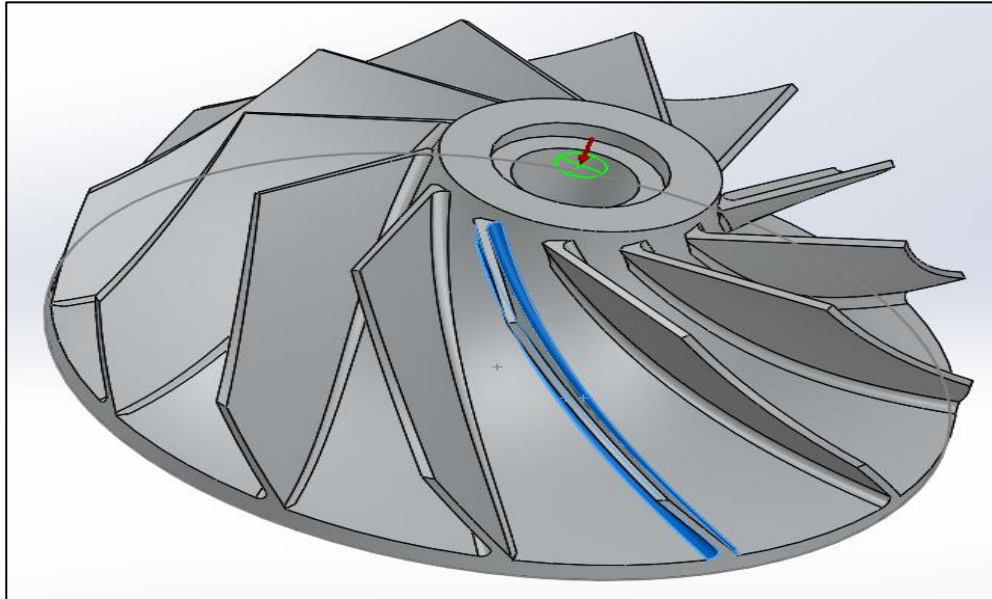

Figure IV- 26: avoided surfaces

- Under Direction, set the Pattern to Zig and set the Cut direction to Climb

- Under Limits, set the Method to Avoid Cuts at Exact Edges
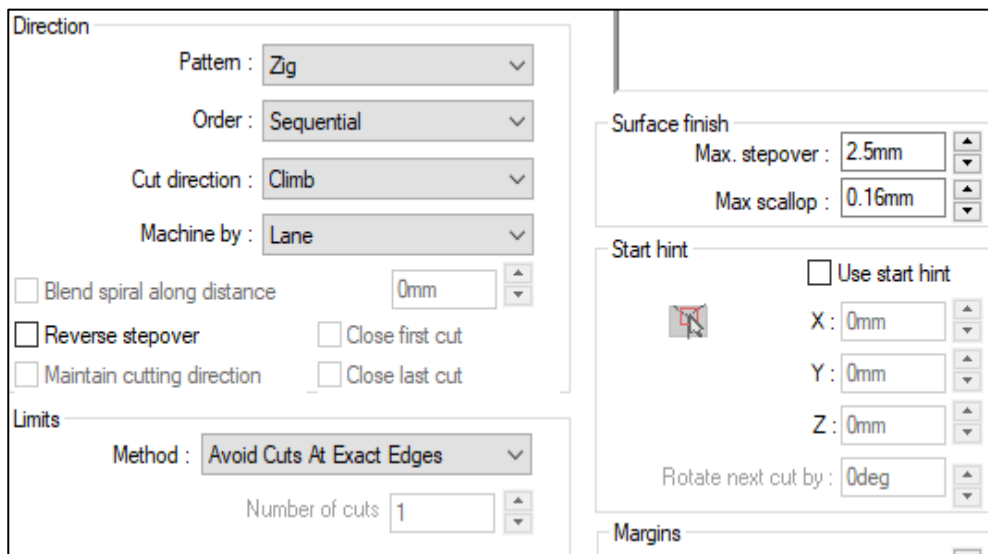
- Under Surface, set the Max. stepover to 2.5mm

Figure IV- 27: pattern tab

6. Edit Operation Parameters – Axis Control tab

- Under Cutting direction tilt options set the Tool axis will be to Tilted Relative To Cutting Direction

- Set the Side tilt angle to 90deg

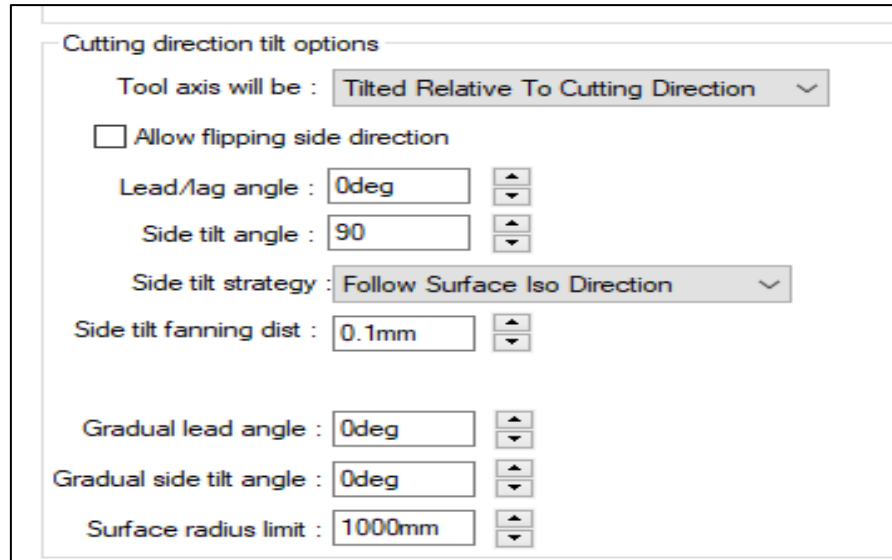- Set the Side Tilt Strategy to Follow Surface Iso Direction

Figure IV- 28: axis control tab

7. Edit Operation Parameters – Gouge Checking tab

- Edit the following parameters on the Gouge Checking Tab

- On the Group 1 tab, under Gouge checking, check on the Apply gouge checking to check box and check on the non-cutting position and Flute check boxes

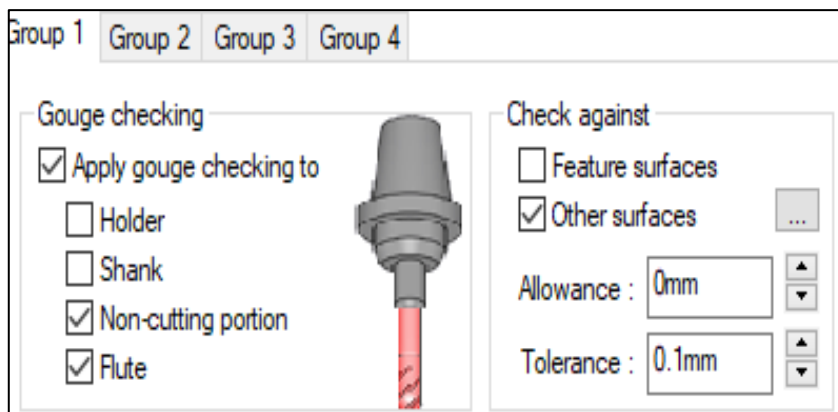- Under Check against, check on the other surfaces check box

Figure IV- 29: Gouge Checking tab

- Click on the selection box to define the other surfaces

- Click on the Create Features…button

- Select the floor surface of the impeller as the Selected Faces and click OK to insert the Multi-Surface Feature
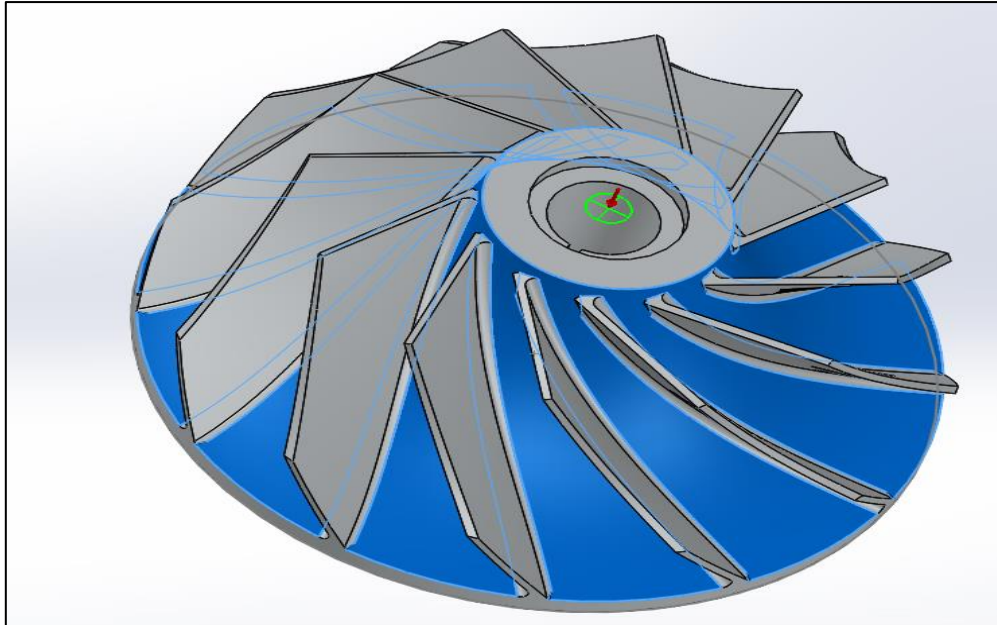
Figure IV- 30: avoid floor surface of impeller

- Under the Gouge check options set the Strategy to Retract Along Tool Axis
- On the Group 2 tab, under Gouge checking, check on the Apply gouge checking to check box and check on the Shank, Non-cutting position and Flute check boxes

- Under Check against, check on the Feature surfaces check box
- Under Gouge check options, set the Strategy to Tilt Tool and the Angle definition to Use Side Tilt Angle
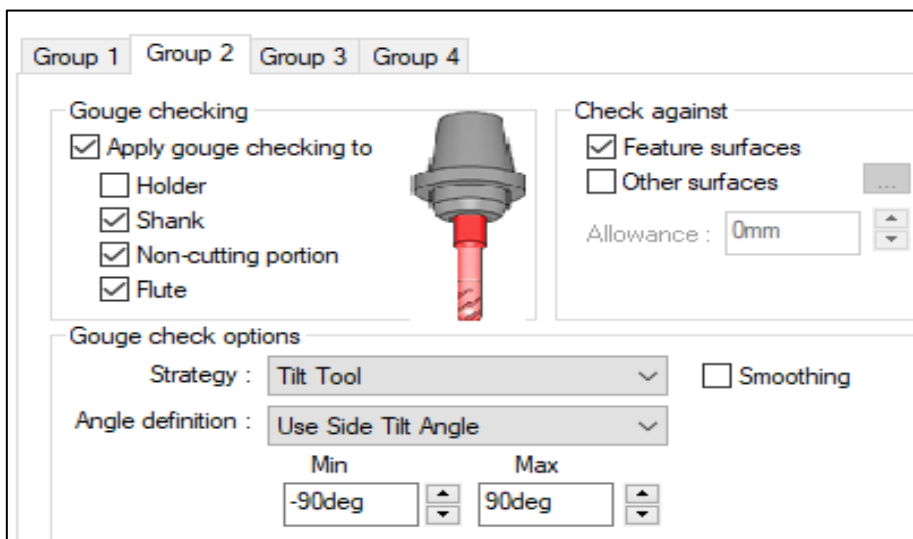

Figure IV- 31: Gouge checking 2nd group

8. Edit Operation Parameters – Entry/Retract tab

    - Under Clearance, set the Type to Cylinder About Z and set the Radius to 125mm

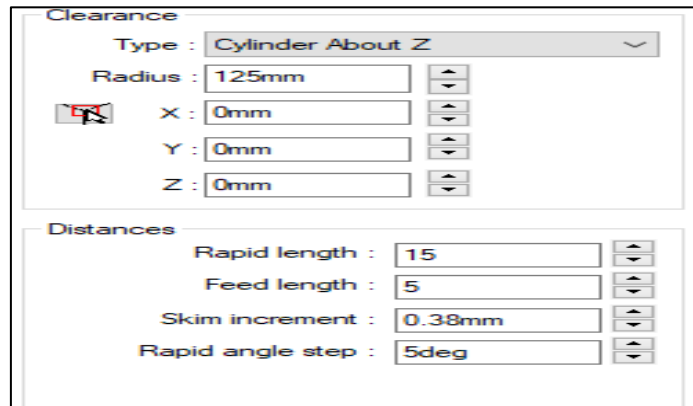   - Under the Distances, set the Rapid length to 15mm and the Feed length to 5mm

Figure IV- 32: Entry/Retract tab

9. Edit Operation Parameters – Links tab

    - Set the Links along cut, Gaps > link threshold, Entry/Exit method to None

    - Set the Links between cut, Stepover > link threshold, Link type to Retract to Rapid Distance and Entry/Exit method to None
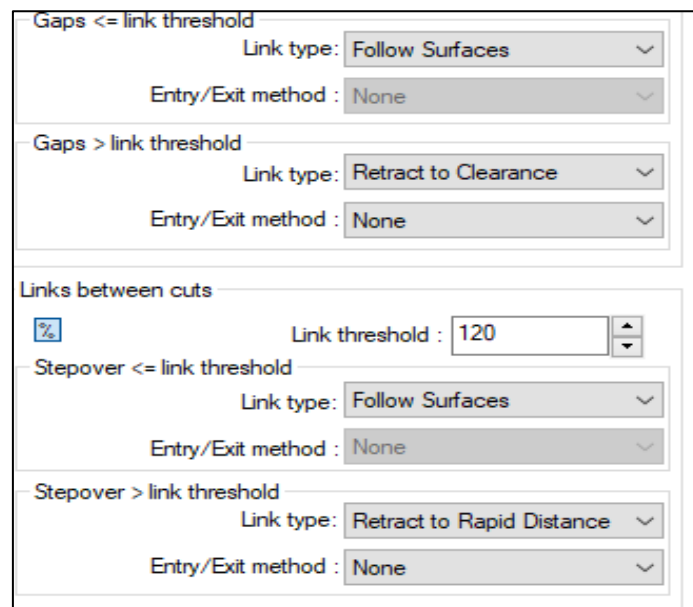

Figure IV- 33: Links tab

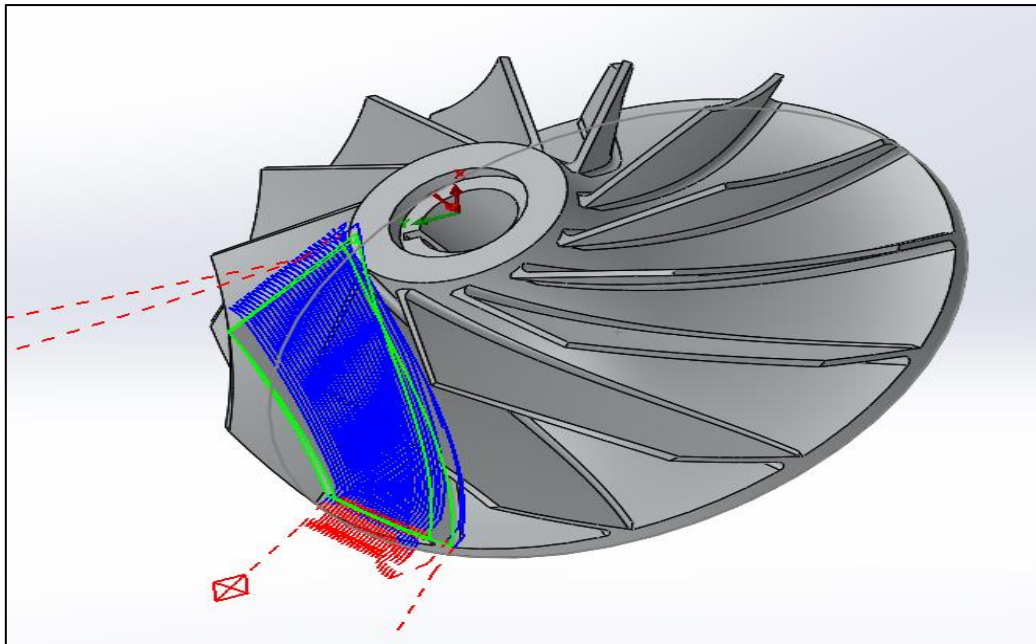   - Click OK

10. Generate toolpath

Figure IV- 34: obtained toolpath

The next step is to finish the floor of the part between the blades. To do this we will insert a new Multi Surface feature

1. Insert new multi surface feature

    - Insert new Multi Surface feature
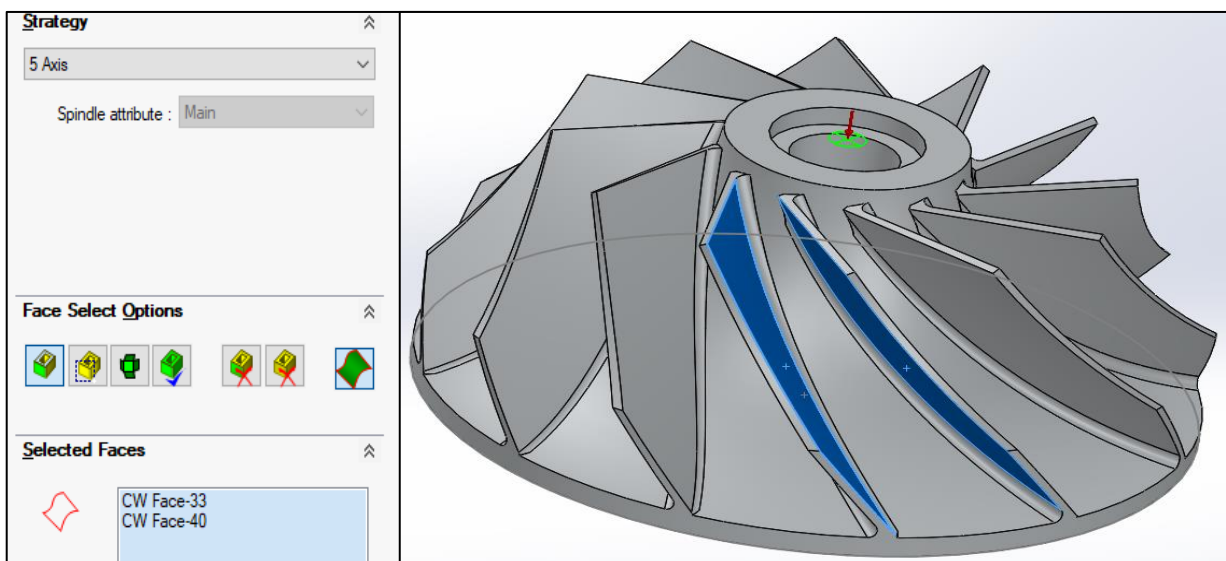
    - For Strategy, select 5 Axis

    - Select faces

Figure IV- 35: new multi-surface

2. Insert new multi surface

- Under Feature Type, check on the Define as Avoid Feature checkbox
- Select faces
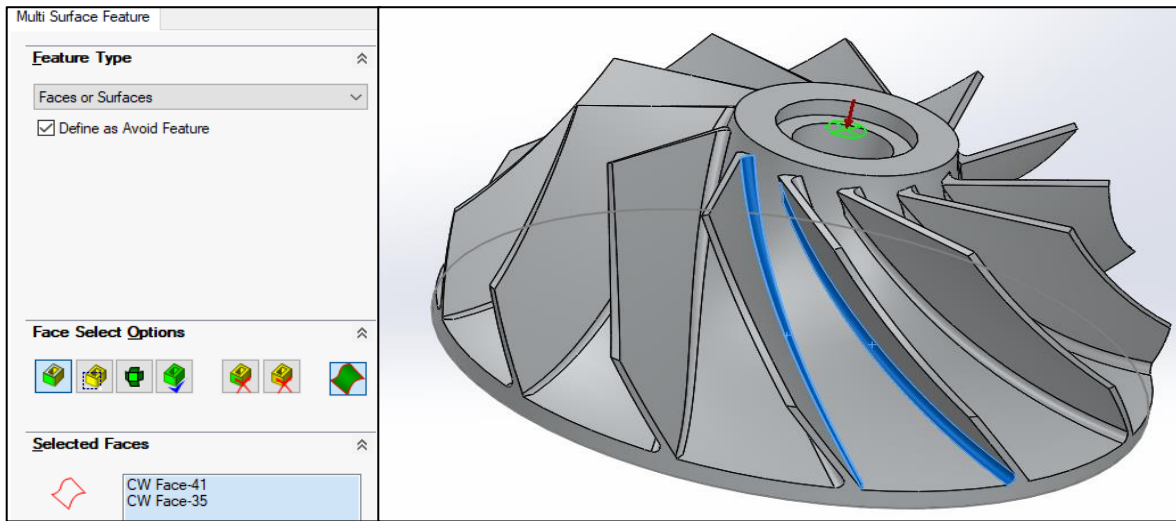
Figure IV- 36: avoided faces selected

3. Generate operation
4. Edit operation parameters
    - Change tool to T08 – 10 Ball Nose
    - Under Pattern, set the Pattern to Offset from Surface
    - Click on the Surface button
    - Click on the check box next to the Multi Surface feature for the bottom radius between the blades
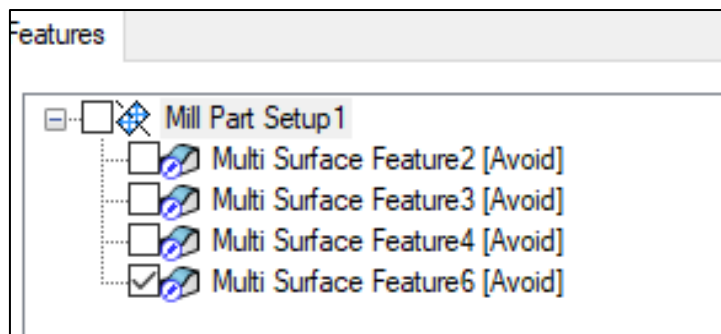

Figure IV- 37: multi surface tab

- Click OK
- Under Limits, set the Method to By Number of Cuts and set the Number of cuts to 1

5. Edit the Operation Parameters, Entry/Retract tab

- Click on the Entry/Retract tab

- Under Clearance, set the Type to Cylinder About Z

- Set the Radius to 125mm

- Under Distances, set the Rapid length to 15mm and the Feed length to 5mm

- Set the Leadin and Leadout to Parallel Line and set Length to 5mm

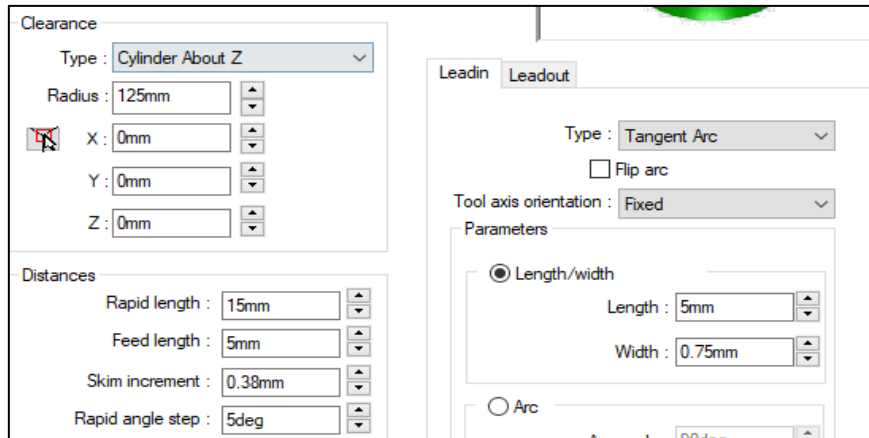**Figure IV- 38:** Entry/Retract tab

6. Edit the Operation, Axis Control tab

- Click on the Axis Control tab

- Under Cutting direction tilt options, set the Side tilt angle to 85deg

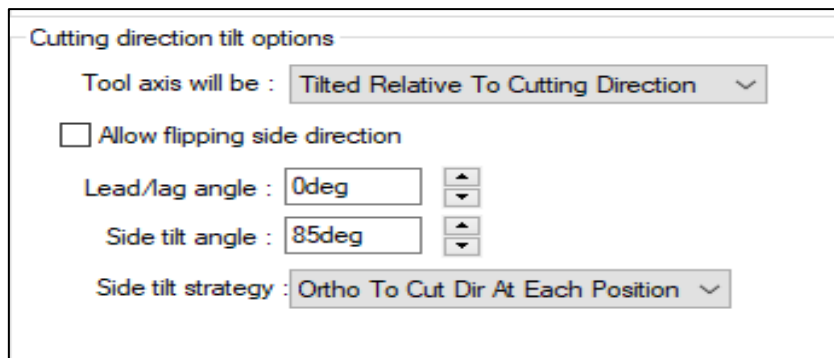- Set the Side tilt strategy to Ortho to Cut Dir at Each Position

Figure IV- 39: Axis Control tab

7. Edit the Operation, Finish tab

- Click on the Finish tab

- Under Surface finish, set the Depth along tool axis to -3mm

- Click OK

Figure IV- 40: surface finish tab

8. Edit Operation Parameters, Gouge Checking tab

- Click on the Gouge Checking tab
- Under Group 1, Gouge checking click on the Apply gouge checking to checkbox
- Click on Non-cutting portion and Flute checkboxes
- Under Check against, click on the other surfaces check box and click on the selection button
- In the Avoid Feature selection box select the floor surface and click OK
- Under Gouge check options, set the Strategy to Retract Along Tool Axis
- Click Preview to Generate toolpath

9. Edit Operation Parameters, Roughing tab

- Click on the Roughing tab
- Check on the Area roughing check box
- Click on the Area roughing options button
- For Calculation applied select After Gouge Checking
- Set the Max stepover to 1mm
- Set the Pattern to Zigzag
- Click OK
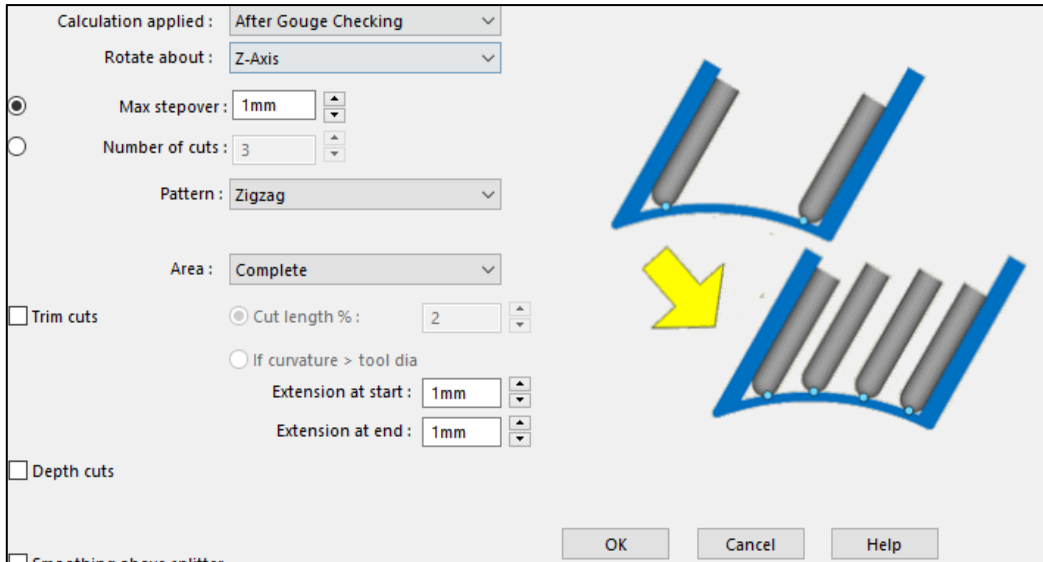
Figure IV- 41: Roughing tab

10. Generate toolpath
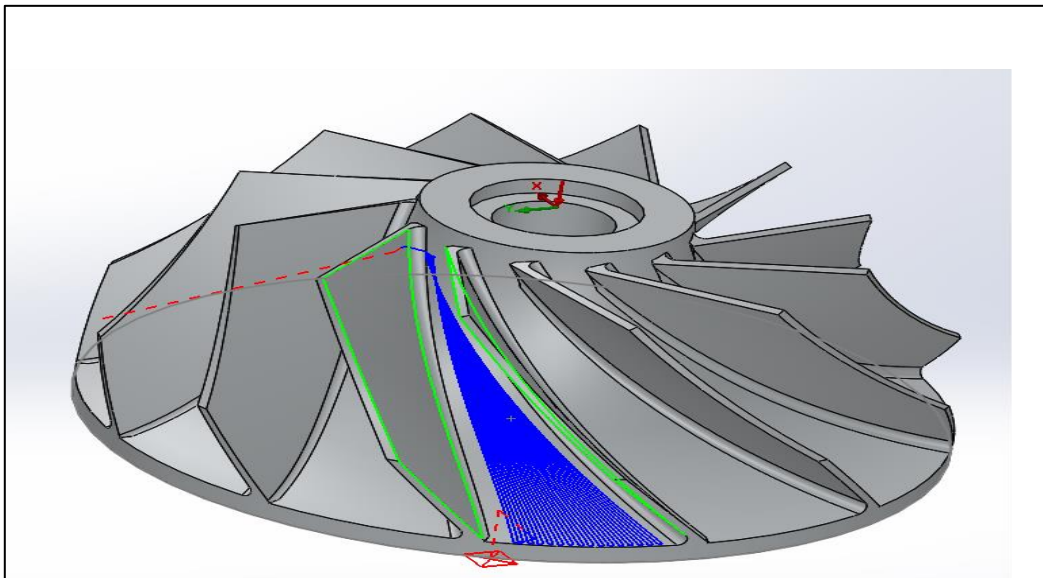
Figure IV- 42: toolpath between blades

We have just programmed the removal of material between two blades on an impeller part.

In the next steps, we will pattern the toolpath to the other blades

11. Edit Operation Parameters, Advanced tab

- Edit Multiaxis Mill2
- Click on the Advanced tab

- Under Rotate toolpath check on the Rotate toolpath check box

- Set the Rotate about to Z Axis

- Set the Number of steps to 12

- Set the Rotation angle to 30deg

- Click ok

- Do the same thing to Multiaxis Mill1
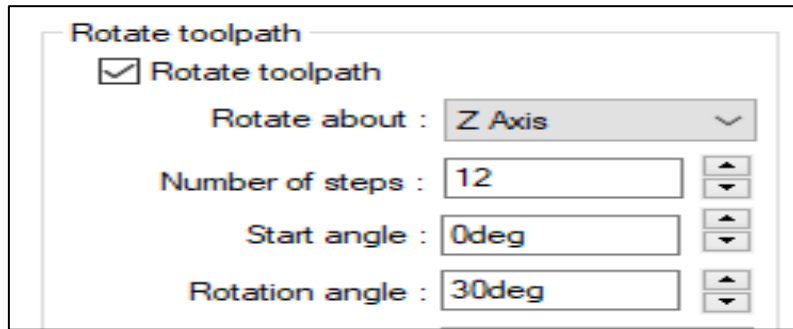
Figure IV- 43: Advanced tab

12. Generate toolpath

13. Simulate toolpath
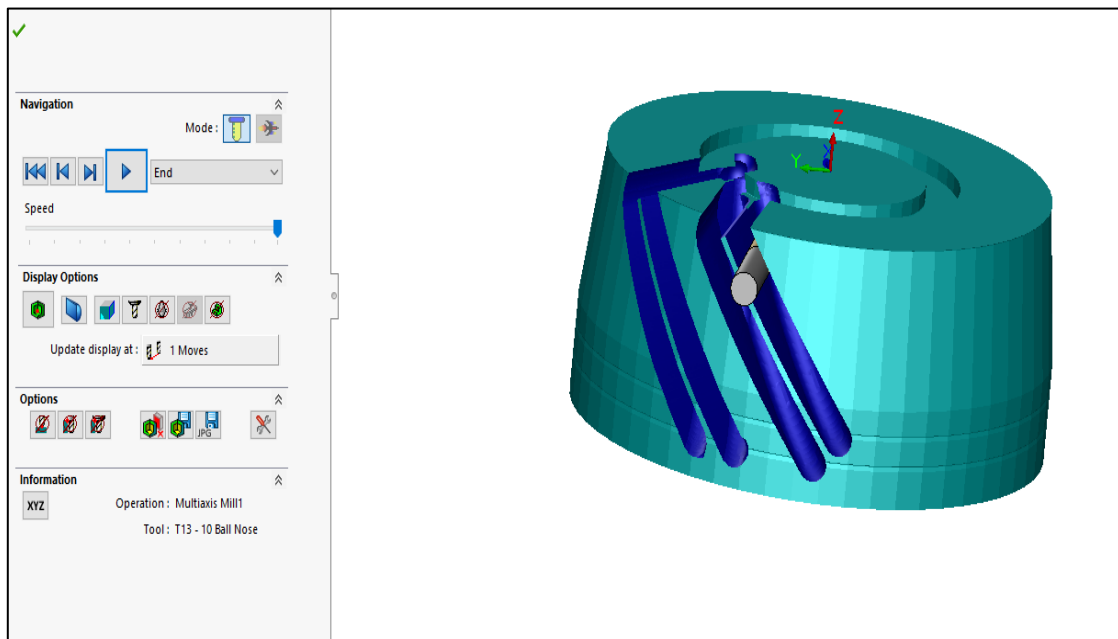
Figure IV- 44: simulation of toolpath

14. Post process

O0001

N1 G21

N2 (10MM CRB 4FL BM 22 LOC)

N3 G91 G28 X0 Y0 Z0

N4 T08 M06

N5 S7189 M03

N6 (Multiaxes Mill1)

N7 G90 G54 G00 X15.29 Y57.062 A-51.016 B88.1261

N8 G43 Z265.436 H08 M08

N9 X12.126 Y56.222 Z239.773

N10 Z229.773

N11 G01 Z224.773 F5.

…….

…….

…….

N3870 G00 Z172.711

N3871 X76.287 Y45.553 Z231.309

N3872 M09

N3873 G91 G28 Z0

N3874 G28 X0 Y0

N3875 M30

VI.3 Conclusion

There are many different methods of mechanical manufacturing, and each method has its own characteristics, which gives the designer the opportunity to realize any of his ideas, working on CNC machines is safer for the designer and works well in line with international standards.

In this part the design of impeller and mold was obtained using CAD software Solidworks then the generation of toolpath and machining simulation was realized using CAM software Camworks.

General conclusion

At the end of this thesis, the opportunity to acquire and improve knowledge in the field of CAD/CAM and address a current problem in the industrial environment has been highlighted. The thesis summarizes the final project of my university course within the department of mechanical engineering at the faculty of science and technology of "Larbi Tebessi university."

Furthermore, the chosen theme of my graduation project has facilitated my gradual integration into my future work framework, especially considering that Algerian companies have started utilizing CNC machines in the mechanical manufacturing sector in recent years.

The project provides a comprehensive understanding of computer numerical control (CNC) machines, as well as an exploration of computer-aided design (CAD) and computer-aided manufacturing (CAM) and their application in CNC.

Additionally, the fundamental principles of constructing freeform surfaces using Bezier, B-spline, NURBS curves have been presented, highlighting their roles in CAD, including the representation of sculptured surfaces, and in CAM, which involves reducing NC data and transfer time while ensuring high efficiency.

The thesis also delves into the manual programming of a milling program for machining a non-twisted blade impeller on a 4-axis CNC machine using the software "Emco WinNc Sinumerik 840D."by developing a new personalized milling cycles which enables time-saving and improves dimensional precision in machining processes.

Finally, the steps for designing a mold for a turbo-compressor impeller using Solidworks are presented, followed by an explanation of the multiaxis machining process using Camworks software. This software allows the creation of toolpaths for complex shapes that cannot be machined using traditional 3 axis machines. The resulting G-code is then used as input for the CNC machine.

References

1. Bulon Ch. Hazarika, 15-22-308, CAD CAM & Automation, NIT Silchar, Assam.

2. NPTEL – Mechanical – Mechatronics and Manufacturing Automation. (2017). Drives and mechanisms.

3. Digital Controllers for Industrial Control. (2008). In Industrial Control Technology (pp. 429–568).

4. Mach motion Specializing in CNC Automation and Motion Control G & M Code Copyright © 2016, Mach Motion All rights reserved.

5. youn, L. (2020, October 03). COMPUTER NUMERICAL CONTROL(CNC).

6. SMID, P. (2003). CNC PROGRAMMING HANDBOOK LIBRARY OF CONGRESS CATALOGING.

7. Prof. Olivier de Weck ©January 6, 2005 Engineering Design and Rapid Prototyping Lecture 4.

8. Zhao, Z., Li, M., Xiao, Y., & Liu, B. (2011). Interpolation Algorithm on Numerical Control Technology. Key Engineering Materials.

9. Barsky, B. A. (1990). Acm/Siggraph '90 course 25: Parametric Bernstein/Bezier curves and tensor product surfaces, Dallas, TX. Aug. 7th 1990.

10. Salomon, D. (2006). Curves and Surfaces for Computer Graphics. Springer.

11. Andersson, F. (2003). Bezier and B-spline Technology.

12. Ameddah, H., & Assas, M. (2009). Interpolation by Bezier curves and NURBS Strategies of Machining of Complex Surfaces in CNC Milling. AIP Conference Proceedings 1107.

13. Ameddah, H., & Assas, M. (2011). NURBS interpolation Strategies of Complex Surfaces in High-Speed Machining. International Journal of CAD/CAM, Vol. 11, No. 1.

14. Lockyer, P. S. (2006). Controlling the Interpolation of NURBS Curves and Surfaces [PhD thesis]. The University of Birmingham.

15. Piegl, L., & Tiller, W. (1997). The NURBS Book (2nd ed.). Springer.

16. Development of a Parametric 3D Turbomachinery Blade Modeler. (2015). [MA thesis]. Delft University of Technology.

17. OPTIMIZED NURBS CURVE BASED G-CODE PART PROGRAM FOR CNC SYSTEMS. (2018). [PhD thesis]. Purdue University.

18. Dodok, T. (2022). Development of advanced cycles for control system sinumerik 840D SL. Matec-conferences, 357, 2022, 01002.

19. EMCO WINNC SINUMERIK 810D/840D milling (Ref. No. En 1814 Edition G2007-06). Software description.

20. Boyuan Casting. (n.d.). Compressor Wheel Made by C355 Aluminum Alloy Low-Pressure Casting. Retrieved March 20, 2023, from https://fr.made-in-china.com/co_boyuancasting/product_Compressor-Wheel-Made-by-C355-Aluminum-Alloy-Low-Pressure-Casting_uorniuuuhy.html

21. Modal analysis of compressor impeller of turbocharger. (2019). IRE Journals, 3(1).

22. Computer aided manufacturing with SolidWorks CAM and Mastercam. (2020). [MA thesis]. Arcada.

23. Schroff, S. (2021). Introduction

24. Malim, Madani & Djamaa, Mohamed & Cherif, & Mohammed, Khechana. (2017). Méthode de Fabrication par Usinage des Aubes de forme NACA sur Machine Cnc.

25. Malim, M., Djamaa, M. C., & Mohammed, K. (2018). Four-axis machining of an impeller with non-twisted blades ResearchGate.