Laboratory Manual

CAD/CAM LAB

for

B. Tech.
Mechanical Engineering

Department of Mechanical Engineering

Rukmani & Sons

Brij Bhooshan Education Web Portal
Web: www.brijrbedu.org

LABORATORY MANUAL

CAD/CAM LAB

for

B. Tech.
Mechanical Engineering

Prepared by

A team of
Rukmani & Sons

E-mail: enquiry@brijrbedu.org
Website: http://www.brijrbedu.org

TABLE OF CONTENTS

LABORATORY OBJECTIVE

The general objectives of the CAD/CAM lab are to enable the students to:

- Model the 3-D geometric information of machine components including assemblies, and automatically generate 2-D production drawings
- Understand the basic analytical fundamentals that are used to create and manipulate geometric models in a computer program
- Improve visualization ability of machine components and assemblies before their actual fabrication through modeling, animation, shading, rendering, lighting and coloring
- Model complex shapes including freeform curves and surfaces
- Understand the possible applications of the CAD/CAM systems in motion analysis, structure analysis, optimization, rapid prototyping, reverse engineering and virtual engineering,
- Implement CNC programs for milling and turning machining operations
- Create a computer aided manufacturing (CAM) model and generate the machining codes automatically using the CAM system
- Integrate the CAD system and the CAM system by using the CAD system for modeling design information and converting the CAD model into a CAM model for modeling the manufacturing information,
- Use full-scale CAD/CAM software systems designed for geometric modeling of machine components and automatic generation of manufacturing information.

ABOUT THE LABORATORY

CAD/CAM laboratory presents the elements of solid modeling, creation of parts of increasing complexity and the assembly of parts to form a final design, along with mechanism simulation. The operation and programming of CNC machines is covered.

This laboratory contains the following setups and equipments:

1. CNC Milling Machine
2. CNC Lathe Machine
3. Robotic Arm
4. Computers for 2D and 3D modeling of mechanical components using modeling software and C language.

GUIDELINES FOR TEACHERS/TECHNICAL ASSISTANTS

1. Know the laboratory: The teacher is expected to understand the layout of laboratory, specifications of equipments/instruments/materials, procedure of experiments, method of working in groups, planning time etc.
2. Ensure that required equipments are in working condition before start of experiment and also keep the operating or instruction/user manuals of equipments/instruments and this laboratory manual available.
3. On the first day of the lab, inform the students about the importance of subject/laboratory, various equipments/instruments that will be used in the lab etc. Also instruct them how to make the practical record file for this lab.
4. Explain the theoretical concepts, relevant to the experiment, to the students before start of each practical.
5. Demonstrate the experiment(s) clearly to the students group-wise.
6. Instruct the students to perform the practical. While taking reading/observation, each student must get a chance to perform or observe the experiment.
7. If the experimental setup has variations in the specifications of the equipment, the teachers are advised to make the necessary changes.
8. Teacher shall assess the performance of students by observation or by asking viva related questions to the students to tap their achievements regarding related knowledge/skills so that students can prepare accordingly.
9. The teacher must check carefully and sign the practical record file of the students periodically.
10. Teacher shall ensure that the industrial/site/plant visits recommended as per the syllabus of laboratory are covered.
11. Teacher should ensure that the respective skills and competencies are developed in the students after the completion of the practical exercise.
12. Teacher may provide additional knowledge and skills to the students albeit not covered in the manual but are expected from students by the industries.
13. Teacher may suggest the students to refer additional related literature of the technical papers, reference books, seminar proceedings etc.
14. Teacher can organize group discussions/brain storming sessions/seminars to facilitate the exchange of practical knowledge amongst the students.

GENERAL PRECAUTIONS AND SAFETY PROCEDURES

1. Teacher/technical assistant must ensure that all the electrical equipments/ instruments are used and periodically performance tested as per manufacturer's recommendations (permissible electrical and ambient temperature ratings).
2. Before use, the electrical equipment, extension cords, power tools etc. must be inspected for any damage (worn insulation, bent/missing pins, etc.). Any equipment found to be damaged or otherwise unsafe must be removed from service.
3. The mains plug of equipments must only be inserted in a socket outlet provided with a protective earth contact.
4. **WARNING**: The protective earth connection inside or outside the equipments/instruments must NEVER be interrupted or tampered. **IT CAN MAKE THE EQUIPMENT DANGEROUS**.
5. If an instrument shows visible damage or fails to perform the intended measurements, it is likely that the protection has been impaired. In such case the instrument must be made inoperative and the necessary repairs should be carried out.
6. Extension cords or power strips must not be plugged into one another so as to increase the overall reach.
7. Report all problems with building electrical systems to the teacher/technical assistant/maintenance for corrective action.
8. In case of any electrical hazard/fire reach out for the nearest fire-extinguisher or sand and use it for putting out the fire. Report immediately to the teacher/ technical assistant nearby.
9. For reasons of safety, every student must come to the laboratory in shoes (covering the whole feet).
10. Avoid wearing garments with loose hanging parts. The students should also ensure that floor around the equipment/machine is clear and dry (not oily) to avoid slipping. Please report immediately to the lab staff on seeing any coolant/oil spillage.
11. The student should take the permission and guidance of the lab staff/teacher before operating any equipment/machine. Unauthorized usage of any machine without prior guidance may lead to fatal accidents and injury.
12. The student will not lean on the equipment/machine or take any kind of support of the machine at any point of time.

INSTRUCTIONS FOR STUDENTS

1. Listen carefully to the lecture and instructions given by the teacher about importance of subject/laboratory, curriculum structure, skills to be developed, information about equipment and instruments, procedure, method of continuous assessment, tentative plan of work in laboratory and total amount of work to be done in the semester/session.
2. Read and understand the theory of each experiment to be performed, before coming to the laboratory.
3. Understand the purpose of experiment and its practical implications. Observe carefully the demonstration of the experiment. When you perform it, organize the work in your group and make a record of all observations.
4. In case of absence, the student must perform the experiment(s) on the next turn or in his/her spare time with permission from the teacher/lab assistant.
5. Student should not hesitate to ask any difficulty faced during conduct of practical/exercise.
6. The student shall study all the questions given in the laboratory manual or asked by the teacher and know the answers to these questions properly.
7. The required instruments/tools will be issued from the laboratory store. They must be returned to the store on the same day at the end of lab hours.
8. Laboratory reports (practical file) should be submitted in a bound file or on A4 size sheets, properly filed, on the next turn completed in all respects i.e. with experiment(s) written, graphs attached (if applicable) and entries made in the list of contents of the file and get them checked from your laboratory teacher. Laboratory reports have associated grades/marks.
9. Student should not bring any food or drink item to the laboratory.
10. Student should develop habit of group discussion related to the experiments/exercises enabling exchange of knowledge/skills.
11. Student shall gain knowledge and develop required practical skills and competencies as expected by the industries.
12. Student shall develop the habit of evolving more ideas, innovations, skills etc. than included in the scope of the manual.
13. Student shall refer technical magazines, proceedings of the seminars; refer websites related to the scope of the subjects and update their knowledge and practical skills.

EXPERIMENT – 1

OBJECTIVE:

The main aim of this experiment is to generate a line of desired dimension and at given pixel on any output generating device.

PREREQUISITES:

Students are required to have understanding of the following basics to enable better understanding of the practical:

1. Knowledge of C-Language.
2. Basic understanding of C-Language commands.

PROCEDURE:

With the given dimensions, detail program in C language is written with the given algorithm stated by BRESENHAM.

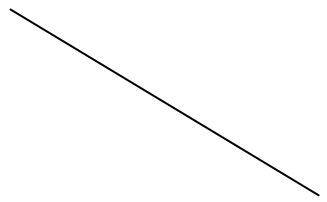
PROGRAM:

```
#include <stdio.h>
#include <conio.h>
#include <math.h>
#include <graphics.h>
void main()
{
int gd=DETECT,gm;
int xa,xb,ya,yb;
int dx,dy,x,y,xend,p;
initgraph(&gd,&gm,"c:\\tc\\bgi");
printf("Enter The Two Left Endpoints (xa,ya):\n");
scanf("%d%d",&xa,&ya);
printf("Enter The Two Right Endpoints(xb,yb):\n");
scanf("%d%d",&xb,&yb);
dx=abs(xa-xb);
```

```
dy=abs(ya-yb);
p=2*dy-dx;
if(xa>xb)
{
x=xb;
y=yb;
xend=xa;
}
else
{
x=xa;
y=ya;
xend=xb;
}
putpixel(x,y,6);
while(x<xend)
{
x=x+1;
if(p<0)
{
p=p+2*dy;
}
else
{
y=y+1;
p =p+2*(dy-dx);
}
putpixel(x,y,6);
}
getch();
return(0);
}
```

OUTPUT:

Enter The Two Left Endpoints (xa,ya) : 234 124
Enter The Two Right Endpoints (xb,yb) : 578 321



PRECAUTIONS:

1. Write the program correctly.
2. Compile the program before run.
3. Remove all the errors and warnings in the program.

EXPERIMENT – 2

OBJECTIVE:

The main aim of this experiment is to write a program for geometric transformation algorithm (translation/ rotation/ scaling).

PREREQUISITES:

Students are required to have understanding of the following basics to enable better understanding of the practical:

1. Knowledge of C-Language.
2. Basic understanding of C-Language commands.

PROCEDURE:

With the given dimensions, detail program in C language is written with the given algorithm stated by BRESENHAM.

PROGRAM:

```
#include <graphics.h>
#include <stdlib.h>
#include <stdio.h>
#include <conio.h>
#include<math.h>
void main()
{
        int gm;
        int gd=DETECT;
        int x1,x2,x3,y1,y2,y3,nx1,nx2,nx3,ny1,ny2,ny3,c;
        int sx,sy,xt,yt,r;
        float t;
        initgraph(&gd,&gm,"c:\tc\bg:");
        printf("\t Program for basic transactions");
        printf("\n\t Enter the points of triangle");
        setcolor(1);
        scanf("%d%d%d%d%d%d",&x1,&y1,&x2,&y2,&x3,&y3);
        line(x1,y1,x2,y2);
```

```
line(x2,y2,x3,y3);
line(x3,y3,x1,y1);
getch();
printf("\n 1.Transaction\n 2.Rotation\n 3.Scalling\n 4.exit");
printf("Enter your choice:");
scanf("%d",&c);
switch(c)
{
        case 1:
                printf("\n Enter the translation factor");
                scanf("%d%d",&xt,&yt);
                nx1=x1+xt;
                ny1=y1+yt;
                nx2=x2+xt;
                ny2=y2+yt;
                nx3=x3+xt;
                ny3=y3+yt;
                line(nx1,ny1,nx2,ny2);
                line(nx2,ny2,nx3,ny3);
                line(nx3,ny3,nx1,ny1);
                getch();

        case 2:
                printf("\n Enter the angle of rotation");
                scanf("%d",&r);
                t=3.14*r/180;
                nx1=abs(x1*cos(t)-y1*sin(t));
                ny1=abs(x1*sin(t)+y1*cos(t));
                nx2=abs(x2*cos(t)-y2*sin(t));
                ny2=abs(x2*sin(t)+y2*cos(t));
                nx3=abs(x3*cos(t)-y3*sin(t));
                ny3=abs(x3*sin(t)+y3*cos(t));
                line(nx1,ny1,nx2,ny2);
                line(nx2,ny2,nx3,ny3);
                line(nx3,ny3,nx1,ny1);
                getch();

        case 3:
                printf("\n Enter the scalling factor");
                scanf("%d%d",&sx,&sy);
                nx1=x1*sx;
```

```
            ny1=y2*sy;
            nx2=x2*sx;
            ny2=y2*sy;
            nx3=x3*sx;
            ny3=y3*sy;
            line(nx1,ny1,nx2,ny2);
            line(nx2,ny2,nx3,ny3);
            line(nx3,ny3,nx1,ny1);
            getch();

    case 4:
            break;
    default:
            printf("Enter the correct choice");
            }
            closegraph();
            }
```


PRECAUTIONS:

1. Write the program correctly.
2. Compile the program before run.
3. Remove all the errors and warnings in the program.

EXPERIMENT – 3

OBJECTIVE:

The main aim of this experiment to write a part programme for the given component of turning and thread cutting on CNC Lathe.

TOOLS REQUIRED:

1. Tool
2. Mild Steel shaft
3. Micro meter
4. Vernier

PROCEDURE:

1. For the given dimensions of the work piece to be machined write the program using G codes and M codes.
2. Using the simulation software or by running the machine in test mode check the program and if there is any error make the correction in the program.
3. Fix the work piece on the chucks.
4. Move the tool to the start point of the work piece by manual mode.
5. Reset the Machine.
6. Change the machine from manual mode to single block mode or auto mode.
7. Execute the program to get the required shape of the work piece.
8. Remove the machined work piece from the chuck.

PROGRAM:

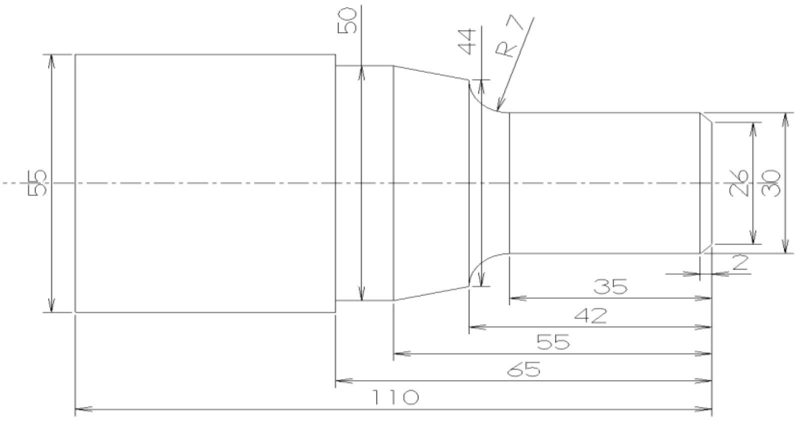
```
T0000
G21
(FACING)
G0T0801
G97S1200M04
G0X55.0Z0M07
G99G1X-1.0F0.2
G0Z2.0
```

```
(OD TURNING)
G92S1250M04
G96S210
X51.0
Z1.0
G71U1.0R2.0
G71P1Q3U0.2W0.12F0.15
N1G0X21.0
G01X21.0Z0
G1X25.0Z-2.0
Z-30.0
G2X35.0Z-35.0R5.0
G1X43.0Z-55.0
Z-65.0
N3X51.0
G97M09
T0000
G00X0Z0
(FINISHING)
T0402
G92S1200M04
G96S240
X55.0Z2.0M07
G70P1Q3F0.1
G97M09
T0000
G28U0W0
(THREADING)
T0304;
G00X0Z-100.0;
G97S100M04;
G00X25.0Z5.0;
```

```
G76P020060Q200R100;
G76X23.268Z-25.0P866Q400F1.0;
S0T0000;
G0X0Z-100.0M09;
M05
M30
```

RESULT:

The part program for the given model is written and the given component is machined to the given dimension.



PRECAUTIONS:

1. Write the program correctly.
2. Compile the program before run.
3. Remove all the errors and warnings in the program.

EXPERIMENT – 4

OBJECTIVE:

The main aim of this experiment to write a part programme for given component of milling operation on CNC Milling.

TOOLS REQUIRED:

1. Tools,
2. Aluminium shaft,
3. Vernier caliper

PROCEDURE:

1. For the given dimensions of the work piece to be machined write the program using G codes and M codes.
2. Using the simulation software or by running the machine in test mode check the program and if there is any error make the correction in the program.
3. Fix the work piece on the vice.
4. Move the tool to the start point of the work piece by manual mode.
5. Reset the Machine.
6. Change the machine from manual mode to single block mode or auto mode.
7. Execute the program to get the required shape of the work piece.
8. Remove the machined work piece from the vice.

PROGRAM:

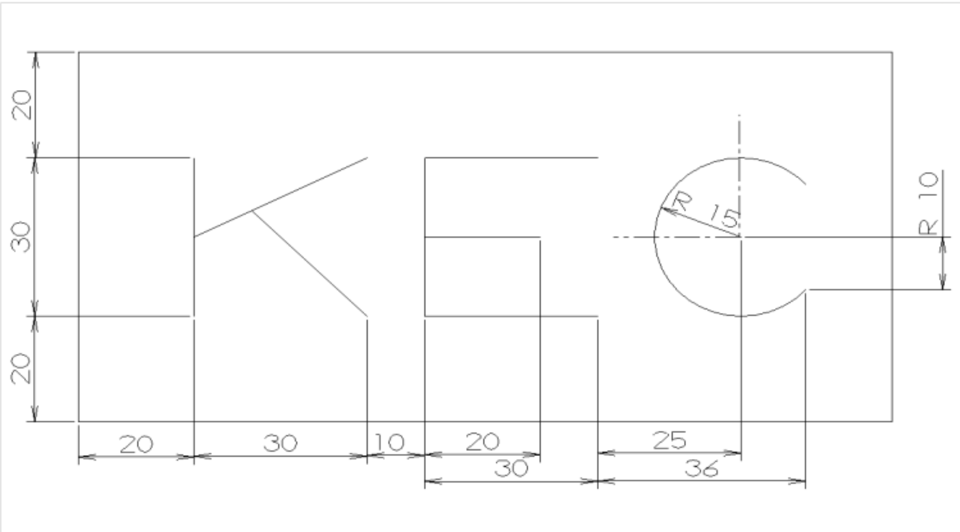
```
%
N01 G90
N02 M03S200
N03 G17
N04 G01 X2000 Y5000 F80
N05 G18
N06 G01Z-500
N07 G17
N08 G01 Y2000
N09 G18
N10 G01 Z500
```

```
N11 G17
N12 G01 Y3500
N13 G18
N14 G01 Z-500
N15 G17
N16 G01 X5000 Y5000
N17 G18
N18 G01 Z500
N19 G17
N20 G01 X3000 Y4000
N21 G18
N22 G01 Z-500
N23 G17
N24 G01 X5000 Y2000
N25 G18
N26 G01 Z500
N27 G17
N28 G01 X9000 Y5000
N29 G18
N30 G01 Z-500
N31 G17
N32 G01 X6000 Y5000
N33 G01 Y2000
N34 G01 X9000
N35 G18
N36 G01 Z500
N37 G17
N38 G01 X6000 Y3500
N39 G18
N40 G01 Z-500
N41 G17
N42 G01 X8000
N43 G18
N44 G01 Z500
N45 G17
N46 G01 X12600 Y2500
N47 G18
N48 G01 Z-500
N49 G17
N50 G02 X12600 Y4500 I-1100 J1000
N51 G18
```

```
N52 G01 Z500
N53 G17
N54 G01 X0Y0
N55 M30
%
```

RESULT:

The part program for producing the given model is written and the given aluminium work-piece is machined to the given dimension.



PRECAUTIONS:

1. Write the program correctly.
2. Compile the program before run.
3. Remove all the errors and warnings in the program.

EXPERIMENT – 5

OBJECTIVE:

The main aim of this experiment to write a part programme for given component of drilling and boring operation on CNC Milling.

TOOLS REQUIRED:

1. Tool
2. Mild Steel shaft
3. Micro meter
4. Vernier

PROCEDURE:

1. Study the drawing carefully to plan for the machining operations.
2. Use the man machine interface to programme for the given geometry.
3. Set the job and the offset for the given work piece.
4. Use the processor for the operation sequence and set the parameters of the operation.
5. Select all the operations for the simulation purpose and execute the program and verify the same in L MILL55 Vertical Machining Centre.
6. Execute the program and remove the work piece from the clamp.

PROGRAM:

```
O0020 (PECK DRILLING & BOARING);
N01
G0G91G28Z0
G28X0Y0
T04 (Centre drill)
M06
M03S800
G0G56G40G49X0Y0Z10.0
G01Z-10.0F5.0
G0Z0
M05
```

N02

G0G91G28Z0

G28X0Y0

T05(Drill dia 10.2mm)

M06

M03S800

G0G57G40G49X0Y0Z0

G95G98G83X0Y0Z-140.0R-100.Q5.0F5.0

G80

M05

G91G28Z0

N03

T08 (U Drill dia22mm)

M06

M03S600

G0G58G40G49X0Y0Z0

G95G98G83X0Y0Z-140.R-100.Q5.F5.0

G80

M05

G91G28X0Y0

N4

T10 (Boring bar dia23)

M06

M03S800

G0G90G59G40G49X0Y0Z0

G95G98G86X0Y0Z-140.R-100.F5.0

M05

G80

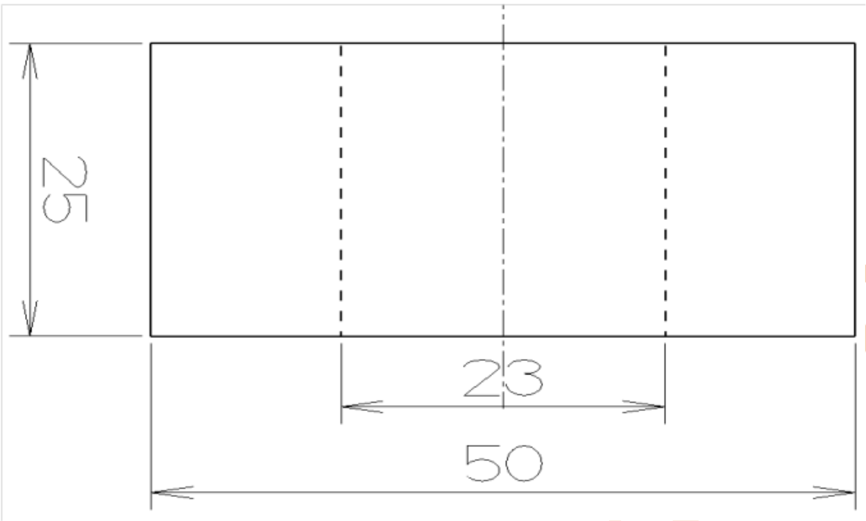
G91G28Z0

G28X0Y0

M30

RESULT:

The part program for the given model is written and the given component is machined to the given dimension.



PRECAUTIONS:

1. Write the program correctly.
2. Compile the program before run.
3. Remove all the errors and warnings in the program.

EXPERIMENT – 6

OBJECTIVE:

The main aim of this experiment is to write a program in C- language to differentiate given function**.**

PREREQUISITES:

Students are required to have understanding of the following basics to enable better understanding of the practical:

1. Knowledge of C-Language.
2. Basic understanding of C-Language commands.

PROCEDURE:

For a given function, detail program in C- language is written.

PROGRAM:

```
#include<iostream.h>
#include<conio.h>
#include<math.h>
float funct(float a);
int main()
{
   char choice='y';
   float f1,f2,x,h;
   clrscr();
   cout<<"X ? ";cin>>x;
   do{
      cout<<"Enter value of h ? ";cin>>h;
      cout<<endl<<"The derivative is: "<<endl;
      f1=(funct(x+h)-funct(x))/h;
      f2=(funct(x+h)-funct(x-h))/(2*h);
      cout<<"2 point derivative : "<<f1;
```

```
        cout<<endl<<"3 points derivative: "<<f2;
        cout<<endl<<"wanna continue (y/n) ? ";cin>>choice;
    }while(choice=='y');
    getch();
    return 0;
}
float funct(float x)
{
    return exp(x)*sin(x);
}
```

PRECAUTIONS:

1. Write the program correctly.
2. Compile the program before run.
3. Remove all the errors and warnings in the program.

EXPERIMENT – 7

OBJECTIVE:

The main aim of this experiment is to write a program for robot to pick a job from given position.

PREREQUISITES:

Students are required to have understanding of the following basics to enable better understanding of the practical:

1. Knowledge of VAL-Language.
2. Basic understanding of VAL-Language commands.

PROCEDURE:

With the given position, detail program in VAL- language is written.

PROGRAM:

1:	LBL[1:Begining];	(Label to jump to)
2:	UFRAME_NUM= 1;	(Frame number similar to work coordinates)
3:	UTOOL_NUM= 1;	(calls Tool number offset)
4:	J P[1:Perch] 50% CNT100 ;	(jog to a recorded position at 50%)(CNT=round off going to point)
5:	RO[1]= ON;	(turns on an output RO 1)
6:	RO[2]= OFF;	(turns off an output RO 2)
7:	RO[3]= ON;	(turns on an output RO 3)
8:	RO[4]= OFF;	(turns off output RO 4)
9:	J P[9:Perch 2] 50% CNT100 ;	(jog to a recorded position at 50%)(CNT= round off going through point /corner a certain amount)
10:	LBL[2:Pick Part];	(label to jump to from another part of program)
11:	$WAITTMOUT= 2000;	(If waiting to long timeout)
12:	;	
13:	L P[2:Approch] 500mm/sec CNT100 ;	(move in a straight line to the point at 500MM/sec rounding off corner)
14:	WAIT DI[2] = ON;	(wait for an input DI2 to come on)

15:	;	
16:	! R[15] is for the part style;	Utilizing register 15 for part style
17:	IF R[15] = 1 , JMP LBL[3];	If register 15 is a 1 (some part style) jump to label 3
18:	IF R[15] = 2 , JMP LBL[4];	If register 15 is a 2 (some part style) jump to label 4
19:	PAUSE;	
20:	ABORT;	If no part style selected abort program
21:	;	
22:	LBL[3:Pick Part#1];	Pick part style #1
23:	L P[11] 100mm/sec FINE ;	L=move in a straight line to a recorded position at a specific speed
24:	;	
25:	JMP LBL[5];	jump over part style #2 to label 5
26:	LBL[4:Pick Part#2];	Pick part style #2
27:	L P[12] 100mm/sec FINE ;	L=move in a straight line to a recorded position at a specific speed
28:	;	
29:	LBL[5:Continue];	label 5
30:	CALL CLSGRP1;	Call subprogram to close gripper
31:	L P[8:Depart Out] 100mm/sec CNT100	move in a straight line to depart position point
32:	IF DI[2] = ON , JMP LBL[98];	If the input Digital input #2(something with conveyor) is on jump to label 98 problem with conveyor
33:	;	
34:	LBL[6:Continue];	label 6
35:	J P[1:Perch] 50% CNT100 ;	move to recorded position the fastest possible way using joint moves
36:	JMP LBL[99];	jump to label 99
37:	;	
38:	;	
39:	LBL[98:Conveyor problem present];	conveyor problem was present
40:	macro_119;	run macro program
41:	DO[14]= OFF;	Turn digital output #14 off
42:	DO[16]= ON;	Turn digital output #14 on
43:	MESSAGE[Alarm indicating];	Message with the said text displayed
44:	MESSAGE[Dropped part];	Message with the said text displayed
45:	MESSAGE[Remedy- check sensor];	Message with the said text displayed

46:	PAUSE;	
47:	JMP LBL[6];	Jump to label #6
48:	;	
49:	LBL[99:End]	label 99
50:	J P[2:"Approach"]	joint move to recorded position #2 commented as an approach point
51:	J P[1:"Perch"]	joint move to recorded position #1 commented as perch point
	/END	End of program

PRECAUTIONS:

1. Write the program correctly.
2. Compile the program before run.
3. Remove all the errors and warnings in the program.

EXPERIMENT – 8

OBJECTIVE:

To study the characteristic features of CNC machine.

THEORY:

Features of CNC Computer NC systems include additional features beyond what is feasible with conventional hard-wired NC. These features, many of which are standard on most CNC Machine Control units (MCU), include the following:

Storage of more than one part program: With improvements in computer storage technology, newer CNC controllers have sufficient capacity to store multiple programs. Controller manufacturers generally offer one or more memory expansions as options to the MCU

Various forms of program input : Whereas conventional (hard-wired) MCUs are limited to punched tape as the input medium for entering part programs, CNC controllers generally possess multiple data entry capabilities, such as punched tape, magnetic tape, floppy diskettes, RS-232 communications with external computers, and manual data input (operator entry of program).

Program editing at the machine tool: CNC permits a part program to be edited while it resides in the MCU computer memory. Hence, a part program can be tested and corrected entirely at the machine site, rather than being returned to the programming office for corrections. In addition to part program corrections, editing also permits cutting conditions in the machining cycle to be optimized. After the program has been corrected and optimized, the revised version can be stored on punched tape or other media for future use.

Fixed cycles and programming subroutines: The increased memory capacity and the ability to program the control computer provide the opportunity to store frequently used machining cycles as macros that can be called by the part program. Instead of writing the full instructions for the particular cycle into every program, a programmer includes a call statement in the part program to indicate that the macro cycle should be executed. These cycles often require that certain parameters be defined, for example, a bolt hole circle, in which the diameter of the bolt circle, the spacing of the bolt holes, and other parameters must be specified.

Interpolation: Some of the interpolation schemes are normally executed only on a CNC system because of computational requirements. Linear and circular interpolation are sometimes hard-wired into the control unit, but helical, parabolic, and cubic interpolations are usually executed by a stored program algorithm.

Positioning features for setup: Setting up the machine tool for a given work part involves installing and aligning a fixture on the machine tool table. This must be accomplished so that the machine axes are established with respect to the work part. The alignment task can be facilitated using certain features made possible by software options in the CNC system. Position set is one of the features. With position set, the operator is not required to locate the fixture on the machine table with extreme accuracy. Instead, the machine tool axes are referenced to the location of the fixture using a target point or set of target points on the work or fixture.

Cutter length and size compensation: In older style controls, cutter dimensions had to be set precisely to agree with the tool path defined in the part program. Alternative methods for ensuring accurate tool path definition have been incorporated into the CNC controls. One method involves manually entering the actual tool dimensions into the MCU. These actual dimensions may differ from those originally programmed. Compensations are then automatically made in the computed tool path. Another method involves use of a tool length sensor built into the machine. In this technique, the cutter is mounted in the spindle and the sensor measures its length. This measured value is then used to correct the programmed tool path.

Acceleration and deceleration calculations: This feature is applicable when the cutter moves at high feed rates. It is designed to avoid tool marks on the work surface that would be generated due to machine tool dynamics when the cutter path changes abruptly. Instead, the feed rate is smoothly decelerated in anticipation of a tool path change and then accelerated back up to the programmed feed rate after the direction change.

Communications interface: With the trend toward interfacing and networking in plants today, most modern CNC controllers are equipped with a standard RS-232 or other communications interface to link the machine to other computers and computer driven devices. This is useful for various applications, such as (1)downloading part programs from a central data file; (2)collecting operational data such as work piece counts, cycle times, and machine utilization; and (3)interfacing with peripheral equipment, such as robots that unload and load parts.

Diagnostics: Many modern CNC systems possess a diagnostics capability that monitors certain aspects of the machine tool to detect malfunctions or signs of impending malfunctions or to diagnose system breakdowns

REFERENCES

1. Computer-Aided Design and Manufacturing, M. Groover, E. Zimmers, Pearson Education, 2009.
2. CAD/CAM/CIM, P. Radhakrishnan, S. Subramanyan, V. Raju, New Age International, 2008.
3. CAD/CAM: Principles and Applications, Posinasetti Nageswara Rao, Tata McGraw-Hill Education, 2004.
4. Computer Aided Manufacturing, Vikram Sharma, S.K. Kataria & Sons, 2013.

