

**LABORATORY WORK BOOK**  
**For The Course**  
**TC-102 Engineering Drawing & Workshop**

(Batch 2011-12)



Name: \_\_\_\_\_

Roll No. : \_\_\_\_\_

Batch: \_\_\_\_\_

Year: \_\_\_\_\_

Dept.: \_\_\_\_\_

**Department of Electronic Engineering**  
**N.E.D. University of Engineering & Technology,**  
**Karachi –75270, Pakistan**

**LABORATORY WORK BOOK**  
**for the course**

**Engineering Drawing & Workshop**  
**(TC-102)**

PREPARED BY:

**Mr. Danish Mahmood Khan (Sr. Lab. Engineer)**  
**Ms. Komal Masroor (Lecturer)**

REVIEWED BY:

**Dr. Syed Shoaib Hasan Zaidi (Chairman)**

APPROVED BY:

**The Board of Studies of Department of Electronic Engineering**

## **ACKNOWLEDGEMENT**

We owe thanks to a number of people who helped and supported us in the compilation of this work book.

We would like to pay our heartfelt gratitude to **Ms. Ayesha Danish**, IT Manager-Jr. (dept. of Telecommunications engineering) for helping us introduce AutoCAD; to give the students an understanding of drawings in a 3D view. Our deepest thanks to **Ms. Tooba Khan**, for structuring lab sessions based on LabView. Also, we sincerely appreciate **Mr. Zeeshan Nafees**, hardware technician (dept. of Electronic engineering), for extending his support to re-engineer practical work related to PCB designing using OrCAD.

We are also grateful to **Prof. Dr. Shoaib Hasan Zaidi** for being a wonderful guide to encourage implementation of problem based learning.

Above all, we would like to acknowledge resources from the INTERNET, cited in the bibliography section, that have been helpful in assembling this workbook.

# **PREFACE**

This document has been divided into four sections:

- Computer Aided Designing
- Soldering
- Data Visualization and Analysis
- Data Acquisition and Control

Each of these segments has been designed to enhance essential hands-on learning and application skills. Lectures and demonstrations are supplemented with project based learning. This process introduces critical thinking and problem solving skills.

- Computer Aided Design (CAD) software are now a very common method of producing technical drawings. OrCAD is one such tool which is used for designing Printed Circuit Boards (PCBs). The sequence of steps, starting from simulating an electronic circuit, converting this to a pattern for the PCB using the software, transferring this pattern to the PCB and subsequently processing the PCB are taught in this segment.

Note: Tasks covered in this manual may not cover all the features of a tool. The emphasis however, is on the steps that you will need to perform in each OrCAD tool so that your design works smoothly through the flow.

- Hands-on Soldering will allow students to make sophisticated and permanent Electronic circuits easily, efficiently and accurately which will help them in their hardware based Term Projects as well as in their Final Year projects.

- The availability of technical computing environment such as MATLAB is now reshaping the role and applications of computer laboratory projects.

It is a powerful and accessible tool for data visualization (both 2D and 3D graphing) and for data analysis. Also, it gives the students an opportunity to easily conduct numerical experiments and to tackle realistic and more complicated problems. The lab sessions have been designed to familiarize students with a computer software (e.g. MATLAB) to address real world problems.

- LabVIEW is a graphical programming language which is widely utilized for data acquisition and process control. This manual therefore, will help the students to get started with the graphical programming.



## CONTENTS

Lab #	List of Experiments	Remarks
	<b><u>Section 1 : Manual drawing</u></b>	
<b>1</b>	<i>To study the objectives of drawing To identify the drawing instruments and study their uses To make some drawings using the drawing equipment</i>	
<b>2</b>	<i>A Polygon in a circle A Pentagon when length of a side is given A Circle in a polygon A Circle in a Triangle</i>	
<b>3</b>	<i>Draw an ellipse Draw an ellipse in a circle Draw an ellipse in a rectangle</i>	
<b>4</b>	<i>To draw a cycloid To draw Epi Cycloid Hypo Cycloid Involute of a circle</i>	
<b>5</b>	<i>To draw Nut and bolts Rivets Threads</i>	
<b>6A</b>	<i>To draw orthographic views of the given machine parts by first and third angle projection methods</i>	
<b>6B</b>	<i>To draw orthographic views of given object using first angle method</i>	
<b>6C</b>	<i>To draw orthographic views of given object using first angle method</i>	
<b>6D</b>	<i>To draw orthographic views of the given object by first angle method</i>	
<b>7</b>	<i>To draw circles in oblique and isometric cubes</i>	
<b>8A</b>	<i>To draw isometric view from the given orthographic views</i>	
<b>8B</b>	<i>To draw isometric view from the given orthographic views</i>	

<b>8C</b>	<i>To draw isometric view from the given orthographic views</i>	
<b>9</b>	<i>To draw the development of lateral surfaces of: Pyramid Prism Cylinder</i>	
<b>10</b>	<i>To learn standard symbols used in civil or architectural drawings To make architectural drawing of different buildings</i>	
<b>11A</b>	<i>To learn standard electrical symbols</i>	
<b>11B</b>	<i>To control one or more bulbs from different places To design a tunnel lighting circuit</i>	
	<b><u>Section 2: Project Based Learning</u></b>	
<b>12</b>	<i>Introduction to Breadboard</i>	
<b>13</b>	<i>To learn how to solder To learn how to de- solder</i>	
	<i>Projects</i>	
	<b><u>Section 3 : PCB DESIGNING using OrCAD</u></b>	
<b>14</b>	<i>Introduction to Orcad Capture Generation of schematic Generation of Netlist</i>	
<b>15</b>	<i>Introduction to Layout Tool Creation of Layout Board</i>	
<b>16</b>	<i>Practical on plotting, fixing and developing a film for exposing of PCB</i>	

	<b><u>Section 4: DATA VISUALIZATION &amp; ANALYSIS</u></b>	
17	<i>Introduction to MATLAB and MATLAB GUI</i>	
18	<i>Introduction to basic operations</i>	
19	<i>Introduction to Mathematical functions, Variables and Matrix operations</i>	
20	<i>Array Operations And Programming in MATLAB</i>	
21	<i>Introduction to Graph plotting in MATLAB</i>	
22	<i>Introduction to Control Flow and Loops in MATLAB</i>	
	<b><u>Section 5: INTRODUCTION TO DATA ACQUISITION, CONTROL AND HARDWARE INTERFACE</u></b>	
23	<i>Introduction to LabView</i>	
24	<i>To learn how to create subVI and use it in another VI</i>	
25	<i>To learn how loops are designed and work in LabView</i> <i>To write a VI for the Multiplication of a random number with 10 and displaying the result continuously, until it is stopped.</i>	
26	<i>To learn how for loops are designed and work in LabView</i>	
27	<i>To learn different functionalities of arrays</i>	
28	<i>To display quantities on waveform charts and waveform graphs</i>	
29	<i>To learn the use of structures in LabView for decision-making</i>	
30	<i>To learn how to build clusters and their various functions.</i>	
31	<i>To learn the use of various functions to edit and manipulate strings</i>	

	<i>To learn the use of basic file I/O functions</i>	
	<b><u>Section 6: INTRODUCTION TO AUTOCAD</u></b>	
<b>32</b>	<i>To be familiar with AutoCAD (2D and 3D drawings)</i>	
<b>33</b>	<i>To learn to draw lines, rays and polylines</i>	
<b>34</b>	<i>To learn to draw objects (e.g. rectangle, polygon, circle etc.)</i>	
	<b><u>BIBLIOGRAPHY</u></b>	

**Note:** Various contents of this manual have been extracted from internet; the links have been cited where possible. If any of the references have been missed, we would be more than pleased to add it to the list of citations in the revisions of this manual.

*Thank you*

**Dr. Shoaib Zaidi**  
**Chairman**  
**Department of Electronic and Telecommunications Engineering**

# Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

## Lab Session 01

### OBJECTIVES:

*To study the objectives of drawing.*

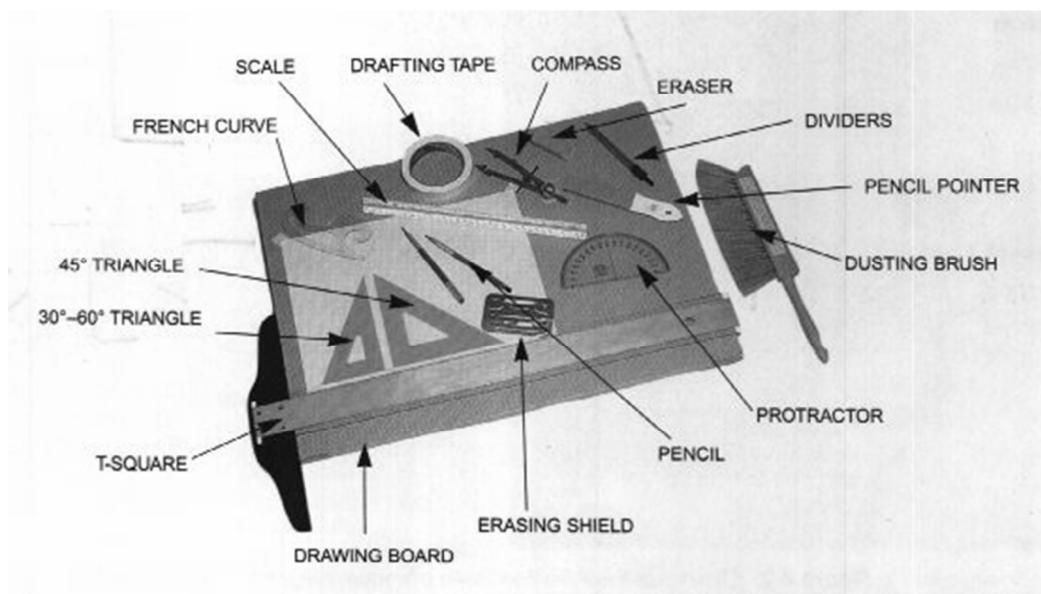
*To identify the drawing instruments and study their uses.*

*To make some drawings using the drawing equipment.*

### DRAWING OBJECTIVES:

The following are the drawing objectives that students should try to attain:

1. **Accuracy:** No drawing is useful if it is not accurate. Therefore, an engineer or designer must acquire the habit of accuracy to achieve success in professional employment.
2. **Speed:** Time is money and so there's no demand for a slow drafter, engineer or technician in industry; so, what one needs to focus on is speed. However, it cannot be attained by hurrying; rather it comes with study and practice.
3. **Legibility:** Drafters, technicians and engineers must remember that a drawing is a means of communication to others, and that it must be clear and legible to serve its purpose well. Care should be given to details, especially to lettering.
4. **Neatness:** if a drawing is to be accurate and legible, it must also be clean. Untidy drawings are the result of sloppy and careless methods and will be unacceptable to an instructor or employer.



### DRAWING INSTRUMENTS AND THEIR USES:

Drawing Instruments are used to prepare neat and accurate Drawings. To a greater extent, the accuracy of the Drawings depends on the quality of instruments used to prepare them. The following is the list of Drawing Instruments and other materials required.

a) Drawing Board

## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

- b) T-square or Drafter (Drafting machine)
- c) Set Squares
- d) Protractor
- e) Drawing Instrument Box
- f) Drawing Sheet
- g) Drawing Pencils
- h) Drawing Pins/Clips/Tape
- i) French curves

**Drawing Boards:** It is rectangular in shape and is made of strips of well-seasoned soft wood about 25mm thick. It is cleated at the back by two battens to prevent warping. One of the edges of the board is used as a *working edge*, on which the T-square is made to slide. It should, therefore be perfectly straight. For a right-hander the left-hand side of the board is the working edge whereas the opposite is true for the left-handers.

If the left edge of the drawing table top has a straight edge and if the surface is hard and smooth, a drawing board then becomes unnecessary and the same drawing table can be used as a board provided the drawings are fastened on it with a drafting tape and it is also recommended to have a back-up sheet placed between the drawing and the table top.

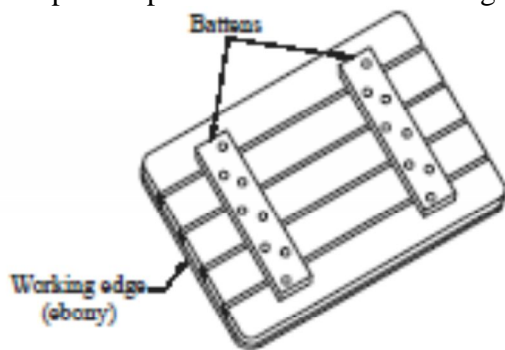


Fig. 1A.1

**T-Squares:** It is a technical drawing instrument used by draftsmen primarily as a guide for drawing horizontal lines on a drafting table. It is made of a long strip called the blade, fastened rigidly at right angles to a shorter piece called the head or the stock. The upper edge of the blade and the inner edge of the head are working edges and must be straight. The working edge of the head must not be convex or the T-square will rock when the head is placed against the board. The blade should have transparent edges and should be free of nicks along the working edge.

**Testing and correcting T-squares:** to test the working edge of the head, see if the T-square rocks when the head is placed against a straight edge, such as a drawing board working edge that has already been tested and found true. If the working edge is not straight, the T-square should be replaced. To test the working edge of the blade, draw a sharp line very carefully with a hard pencil along the entire length of the working edge; then turn the t-scale over and draw the line again along the same edge. If the edge is straight, the two lines will coincide; otherwise, the space between the lines will be twice the error of the blade.

it is difficult to correct a crooked T-square blade, and if the error is considerable, it may be necessary to discard the T-square and obtain another.

## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

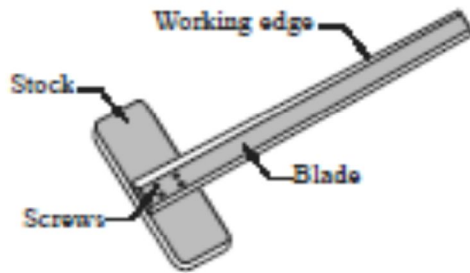


Fig. 1A.2

**Set squares:** Set squares are generally made from Plastic or celluloid material. They are triangular in shape with one corner, a right angle triangle. They are used for drawing all straight lines except the horizontal lines which are usually drawn with the T-square. Vertical lines can be drawn with T-square and the set square. In combination with the T-square, lines at  $30^\circ$  or  $60^\circ$  angle with vertical or horizontal lines can be drawn with  $30^\circ$ - $60^\circ$  set square and at  $45^\circ$  with  $45^\circ$  set square. The two set squares used simultaneously along with the T-square will produce lines making angles of  $15^\circ$ ,  $75^\circ$ ,  $105^\circ$  etc.

Parallel straight lines in any position, not very far apart, as well as lines perpendicular to any line from any given point within or outside it, can be drawn with the two squares.

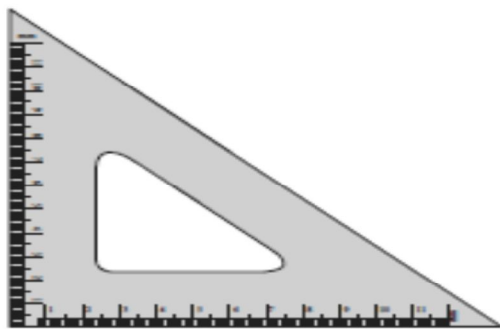


Fig. 1A.4.  $45^\circ$  Set Square

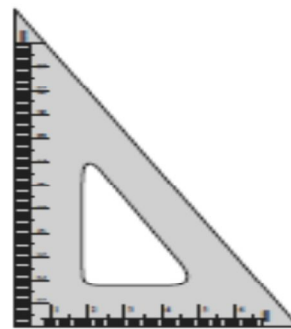


Fig. 1A.5.  $30^\circ$ - $60^\circ$  Set Square

**Protractor:** it is flat and circular or semi-circular in shape. The commonest type is of protractor is semi-circular and of about 100mm diameter. Its circumferential edge, graduated to  $1^\circ$  divisions, is numbered at every  $10^\circ$  interval and is readable from both the ends. The diameter of the semi-circle is called the base of the protractor and its center point is marked by a perpendicular line.

The protractor is used to draw or measure such angles as cannot be drawn using a set square. Also, a circle can be divided into any number of parts by means of a protractor.

## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

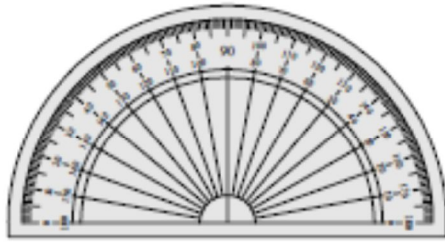


Fig. 1A.7

**Drawing Instrument Box:** It consists of the following:

- a) Large size compasses,
- b) Large size divider,
- c) Small size bow pen, bow divider, and
- d) Lengthening bar

### **Drawing sheet:**

They are available in many varieties and good quality paper with smooth surface should be selected for Drawings which are to be preserved for longer time. Recommended sizes of Drawing Sheets are given below:

#### **Standard sizes of drawing sheet**

Designation	Size (mm)	Designation	Size (mm)
A0	1189 × 841	A3	420 × 297
A1	841 × 594	A4	297 × 210
A2	594 × 420	A5	210 × 148

**Drawing Pencils:** The accuracy and appearance of a Drawing depends on the quality of Pencil used to make it. The grade of a Pencil lead is marked on the Pencil. HB denotes medium grade. Increase in hardness is shown by value put in front of H such as 2H, 3H etc., Softer pencils are marked as 2B, 3B, and 4B etc. A Pencil marked 3B is softer than 2B and Pencil marked 4B is softer than 3B and so on. Beginning of a Drawing may be made with H or 2H. For lettering and dimensioning, H and HB Pencils are used. The final fair work however, may be done with harder pencils like 3H and upwards. For freehand sketching, where considerable erasing is required to be done, soft-grade pencils such as HB should be used.

**Drawing Pins/clips/ tape:** These are used to fix the drawing sheet on the Drawing board.

**Compass:** It is used for drawing circles and arcs of circles. The compass has two legs hinged at one end. One of the legs has a pointed needle fitted at the lower end whereas the other end has provision for inserting pencil lead. Circles up to 120mm diameters are drawn by keeping the legs of compass straight. For drawing circles more than 150 mm radius, a lengthening bar is used. It is advisable to keep the needle end about 1mm long compared to that of pencil end so that while drawing circles, when the needle end is pressed it goes inside the drawing sheet by a small distance (approximately 1mm). Refer fig. 1A.8



## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

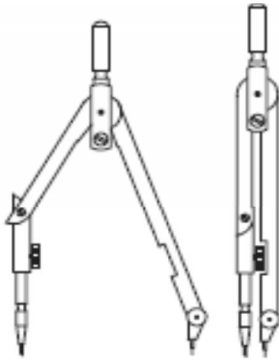


Fig. 1A.8 Compass

**French Curves:** Drawing mechanical curves other than circles or circular arcs generally requires the use of an irregular or French curve. An irregular or French curve is a device for the mechanical drawing of curved lines and should not be applied directly to the points or used for purpose of producing an initial curve. These are made in various shapes, some of which have been shown in the figure below. Some set squares also have these curves set in their middle.



### Exercise 1:

**Draw a straight horizontal line using T-square.**

- To draw a straight horizontal line using a T-square, hold firmly its stock with the left hand against the working edge of the board, and draw the line from left to right.

**Note:** The pencil should be held slightly inclined in the direction of the line (i.e. to the right) while the pencil point should be as close as possible to the working edge of the blade. Horizontal parallel lines are drawn by sliding the stock to the desired position.

### Exercise 2:

**Draw a line perpendicular to the horizontal line drawn in Exercise 1 from a given point within it.**

- Place the T-square a little below the given line.
- Arrange any one of the Set Squares with one of the edges containing the right angle touching the working edge of the T-square, and the other edge passing through the given point.
- Hold firmly the T-square and the set-square with the left hand in this position.
- With the right hand, draw the required line through the given point in the upward direction which is the required vertical line.

**Note:** The pencil point should always be in contact with the edge of the set-square. Vertical parallel lines may be drawn by sliding the set-square along the edge of the T-square to the required position.

## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

### Exercise 3:

Divide the given line segment AB into n-equal parts. (Here n=5 parts).

1. Draw a line segment AB.



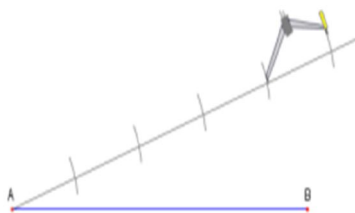
2. From point A, draw a line segment at an angle to the given line, and about the same length. The exact length is not important.



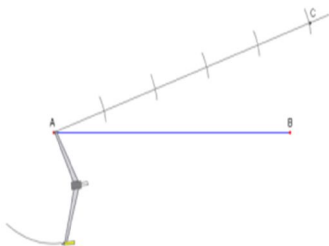
3. Set the compass on A, and set its width to a bit less than one fifth of the length of the new line.



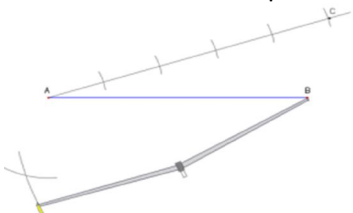
4. Step the compass along the line, marking off 5 arcs. Label the last one C.



5. With the compass width set to CB, draw an arc from A just below it.



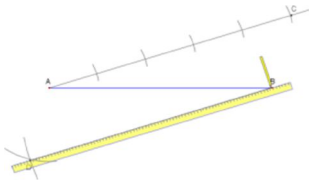
6. With the compass width set to AC, draw an arc from B crossing the one drawn in step 4. This intersection is point D.



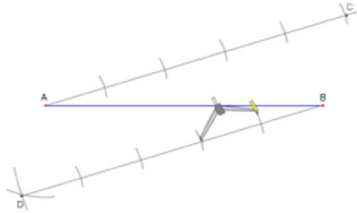
## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

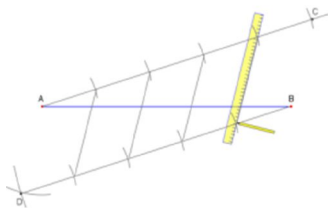
7. Draw a line from D to B.



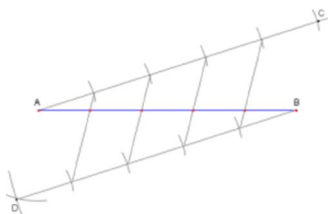
8. Using the same compass width as used to step along AC, step the compass from D along DB making 4 new arcs across the line



9. Draw lines between the corresponding points along AC and DB.

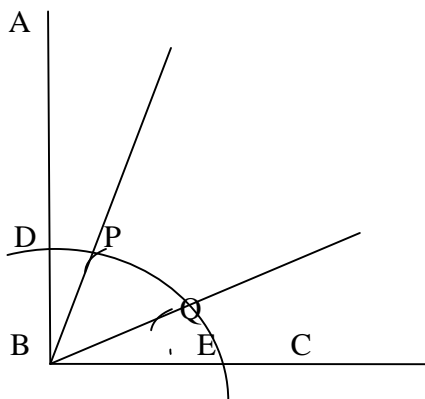


10. Done. The lines divide the given line segment AB in to 5 congruent parts.



### Exercise 4:

Divide a given right angle into three parts (trisect it).



## Engineering Drawing & Workshop

*NED University of Engineering and Technology- Department of Electronic Engineering*

- Let ABC be the given right angle.
- With center B and any radius, draw an arc cutting AB at D and BC at E
- With centers D and E having the same radius as BE or BD, draw arcs cutting the arc DE at points P and Q.
- Draw lines joining B with P and Q.
- BP and BQ trisect the right angle ABC.

Thus,  $\angle ABP + \angle PBQ + \angle QBC = \angle ABC$

**HOME WORK: Divide a line segment AB=15cm into 12 equal parts.**

## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

### Lab Session 02

## OBJECTIVES:

*To draw:*

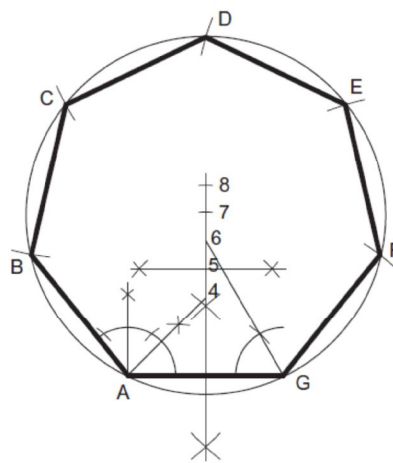
*A Polygon in a circle*

*A Pentagon when length of a side is given.*

*A Circle in a polygon.*

*A Circle in a Triangle.*

#### Exercise 1 : Polygon In A Circle:



#### Steps:

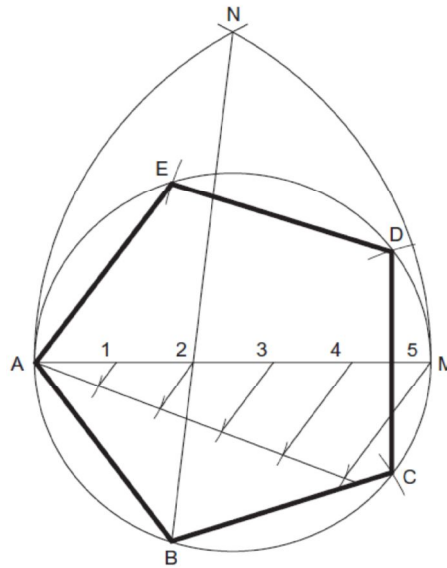
1. Draw a line GA equal in length to one of the sides.
  2. Bisect GA.
  3. From A construct an angle of  $45^\circ$  to intersect the bisector at point 4.
  4. From G construct an angle of  $60^\circ$  to intersect the bisector at point 6.
  5. Bisect between points 4 and 6 to give point 5.
- Point 4 is the center of a circle containing a square. Point 5 is the center of a circle containing pentagon. Point 6 is the center of a circle containing a hexagon. By marking off points at similar distances the centers of circles containing any regular polygon can be obtained.
6. Mark off point 7 so that  $6 \text{ to } 7 = 5 \text{ to } 6$  (  $= 4 \text{ to } 5$  ).
  7. With center at point 7 draw a circle, radius 7 to A (  $= 7 \text{ to } G$  ).
  8. Step off the sides of the figure from A to B, B to C, etc.

ABCDEFG is the required heptagon.

#### Exercise 2: To construct a regular polygon given a diagonal, i.e. within a given circle

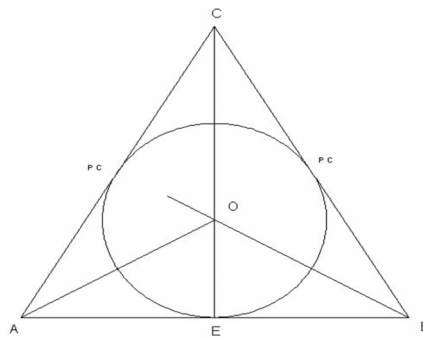
## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering



1. Draw the given circle and insert a diameter AM.
2. Divide the diameter into the same number of divisions as the polygon has sides.
3. With center M draw an arc, radius MA. With center A draw another arc of the same radius to intersect the first arc in N.
4. Draw N 2 and produce to intersect the circle in B (for any polygon).
5. AB is the first side of the polygon. Step out the other sides BC, CD, etc. ABCDE is the required polygon.

### Exercise3: To inscribe a circle in a given triangle



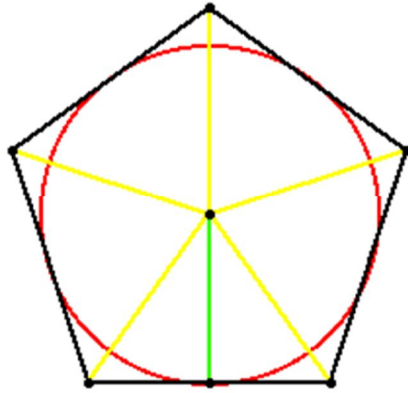
#### Steps:

1. Draw an equilateral triangle ABC.
2. Bisect any two angles by lines intersecting each other at 'O'.
3. Draw a perpendicular from O to any one side of the triangle meeting it at E.
4. With Center O and radius OP, describe the required circle.

## Engineering Drawing & Workshop

*NED University of Engineering and Technology- Department of Electronic Engineering*

**HOME WORK:** To inscribe a circle in a regular polygon (say Pentagon).



## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

### Lab Session 03

## OBJECTIVES:

*Draw an ellipse*

*Draw an ellipse in a circle*

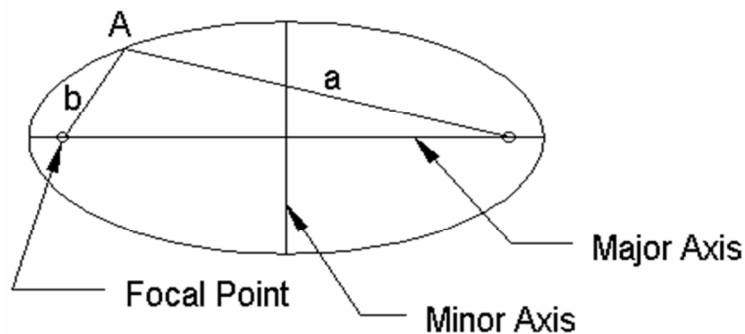
*Draw an ellipse in a rectangle*

*Draw an ellipse in a parallelogram*

## ELLIPSE

### Definition:

When a point moves so that the sum of its distances from two fixed points is a constant, then the locus of the point is an ellipse.



## DRAW AN ELLIPSE BY ARCS OF CIRCLES METHOD

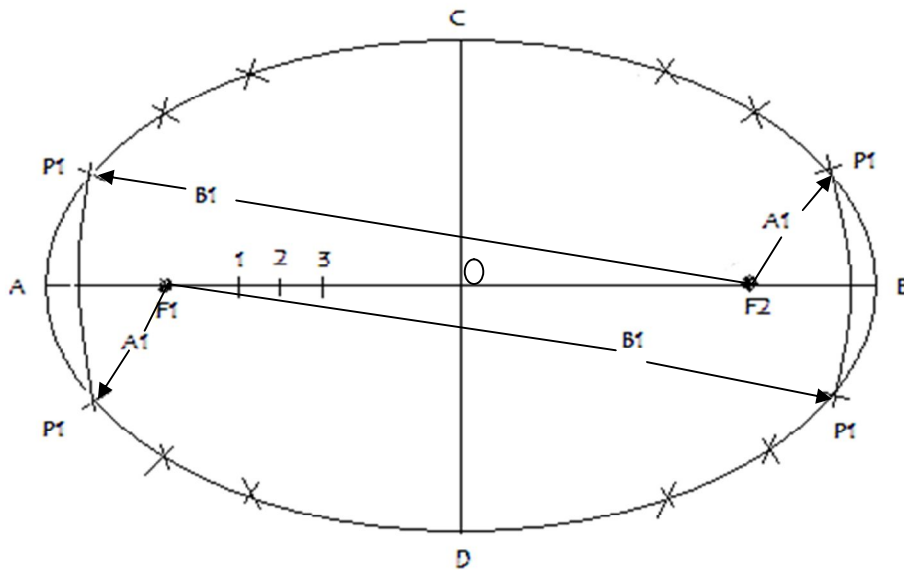
### Method of construction:

- i. Draw major axis and minor axis AB and CD respectively, bisecting each other at O.
- ii. With center C and radius equal to half AB (i.e. OA) draw arcs cutting AB at F1 and F2, the foci of the ellipse.
- iii. Take any points 1, 2, 3 etc. on AB.
- iv. With centers F1 and F2 and radius equal to A1, draw arcs on both sides of AB.
- v. With same centers and radius equal to B1, draw arcs intersecting the previous arcs at four points marked P1.
- vi. Similarly, with radii A2 and B2, A3 and B3 etc., obtain more points.
- vii. Draw smooth curve through these points. This curve is the required ellipse.



## Engineering Drawing & Workshop

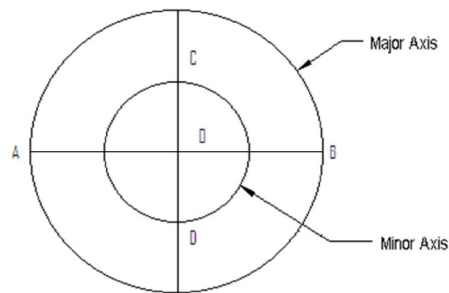
NED University of Engineering and Technology- Department of Electronic Engineering



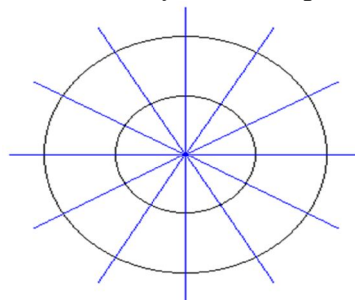
### DRAW AN ELLIPSE IN A CIRCLE USING “CONCENTRIC CIRCLES METHOD”

#### Method of construction:

- i. Draw the major and minor axes AB and CD intersecting at center O, and also draw circles with diameters AB and CD.



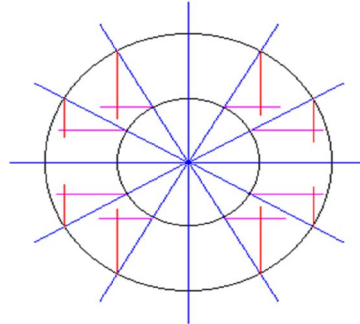
- ii. Draw radial lines out from the center at regular intervals. (The number of lines will affect the accuracy of the ellipse)



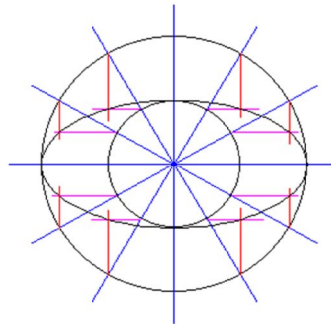
## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

- iii. Draw lines across from the points where the radial lines cross the minor axis circle.
- iv. Draw lines up or down from the points where the radial lines cross the major axis circle.



- v. Join the points where these lines cross with a smooth curve to draw the ellipse.



### DRAW AN ELLIPSE IN A RECTANGLE USING “OBLONG METHOD”

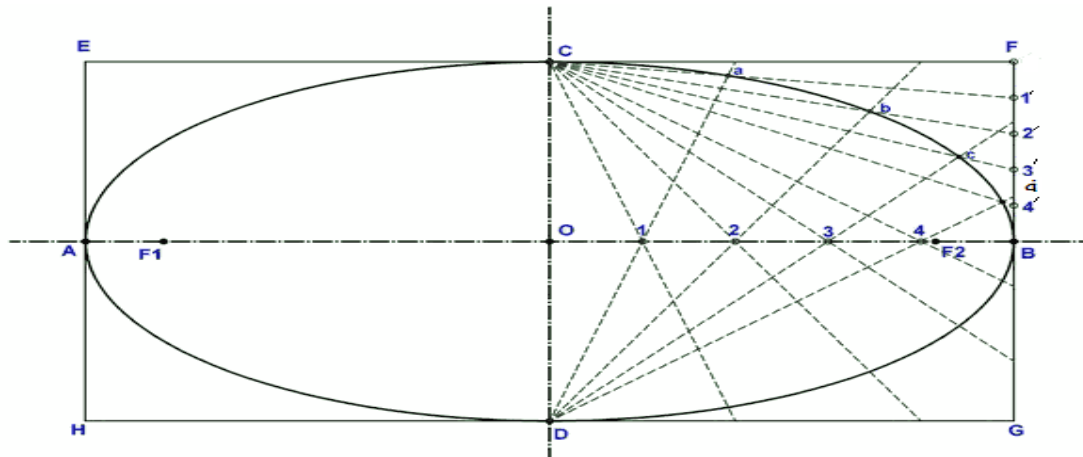
#### Method of construction

- i. Draw the major and minor axis AB and CD.
- ii. Draw the rectangle EFGH whose sides are equal to the length of the axes.
- iii. Divide the half of major axis OB into any equal number of parts, say 4, denoted by 1, 2, 3, 4 and divide FB into the same number of equal parts denoted by 1', 2', 3', 4'.
- iv. Draw radial lines from C through the points 1, 2, 3 and 4 on OB and 1', 2', 3', 4' on FB.
- v. Draw radial lines from D through the points 1, 2, 3 and 4 on OB, these will intersect the previously drawn radial lines at points a, b and c.
- vi. Draw the curve through C, a, b, c, d and B.
- vii. Complete the curve by repeating the same construction in each of the three quadrants.

(Note: As the curve is symmetric about both axes, the remaining curve may also be completed by drawing perpendiculars and horizontals from a, b, c and d.)

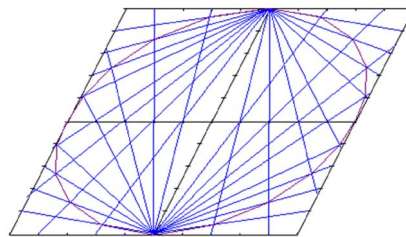
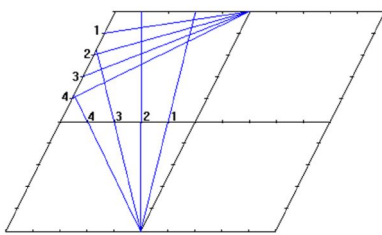
## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering



### HOMEWORK :

Draw an ellipse in parallelogram-“oblong method”



## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

### Lab Session 04

## OBJECTIVES:

To draw a cycloid

To draw an Epi-Cycloid

To draw a Hypo-Cycloid

To draw Involute of a circle

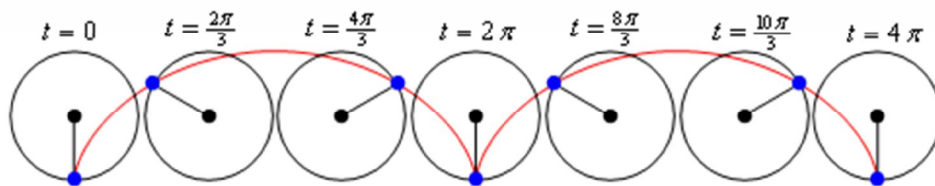
**Home work :**

Draw the involute of a square

### What is a Cycloid?

A **Cycloid** is the path or **Locus** followed by a point on a circle when it moves along a straight line without slipping.

Here is a cycloid sketched out with the wheel shown at various places. The dot is the point *P* on the wheel that we're using to trace out the curve.



From this sketch we can see that one arch of the cycloid is traced out in the range  $0 \leq \theta \leq 2\pi$ . This makes sense when you consider that the point *P* will be back on the ground after it has rotated through an angle of  $2\pi$ .

### Exercise 1:

#### Construction of a cycloid

- First draw the circle of given radius and a line equal to its circumference ( $2\pi r$ ) from its base *P* to the left or right. The line should be to the right if the circle is rotating clockwise, and to the left if the circle is rotating anti-clockwise.



- Now divide the circle into an equal number of parts. Here Use  $60^\circ$ - $30^\circ$  set square to have 12

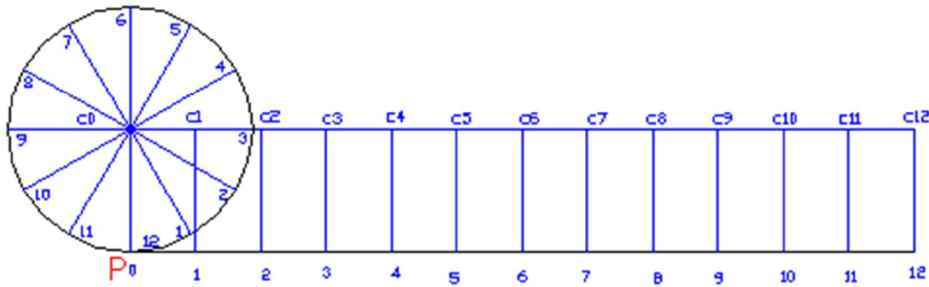
## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

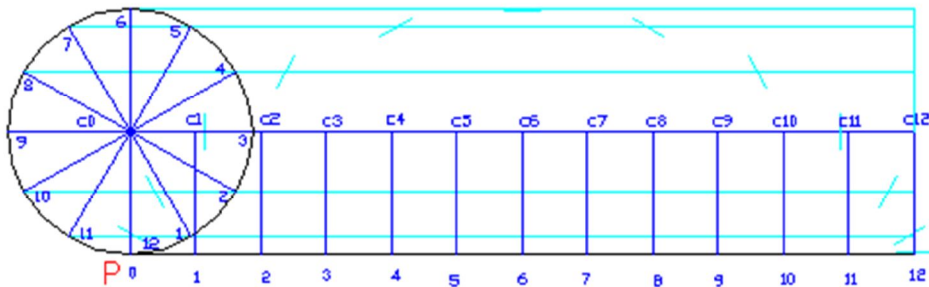
equal divisions. Label these sections 1 to 12.

- iii. Draw a line parallel to the base line from the center  $C_0$  of the circle, divide it into 12 equal parts and label them  $C_1$  to  $C_{12}$ . Now project these points on the base line labeling them 1 to 12.

(As the circle rotates along the line, point 1 on the circle will move to point 1 on the base line, and therefore the center of the circle will now be at point  $C_1$ .)



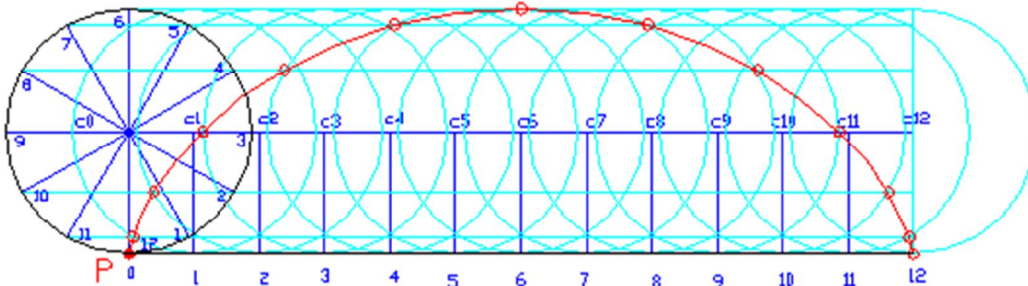
- iv. Next project lines from the equal divisions of the circle parallel to the base line. These are the height lines.
- v. With your compass set to the radius of the circle, place the point of the compass on the point  $C_1$  and cut the height line coming from point 1 on the circle. Then repeat this for point  $C_2$  and height line 2, etc. until you reach point  $C_{12}$  and line 12.



- vi. By joining these new points you will have created the locus of the point  $P$  for the circle as it rotates along the straight line without slipping. In other words, you have created a **Cycloid**.

Essentially what you have done is break up the continuous movement of the circle along the straight line into 12 sections and found the position of point  $P$  for each movement.

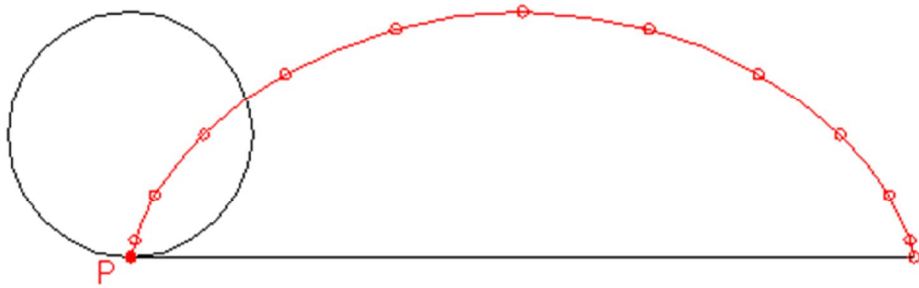
It would get much cluttered if you drew in every circle along the way.



And your final result is a **Cycloid**.

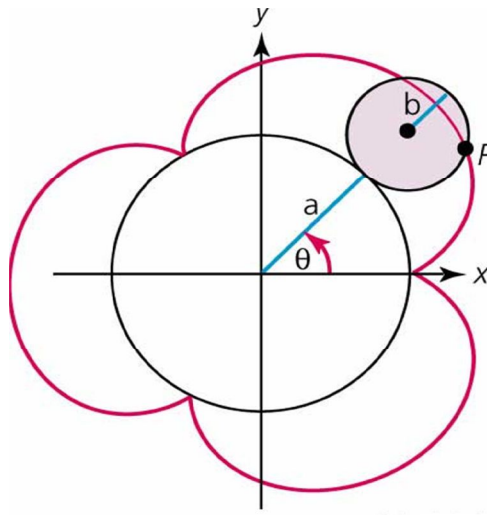
## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering



### EPICYCLOIDS

The cycloid is called the epicycloid when the generating circle rolls along outside another circle (directing circle). **OR** Locus followed by a point on a circle which rolls on the outside of a circular base surface.



### **Exercise 2:**

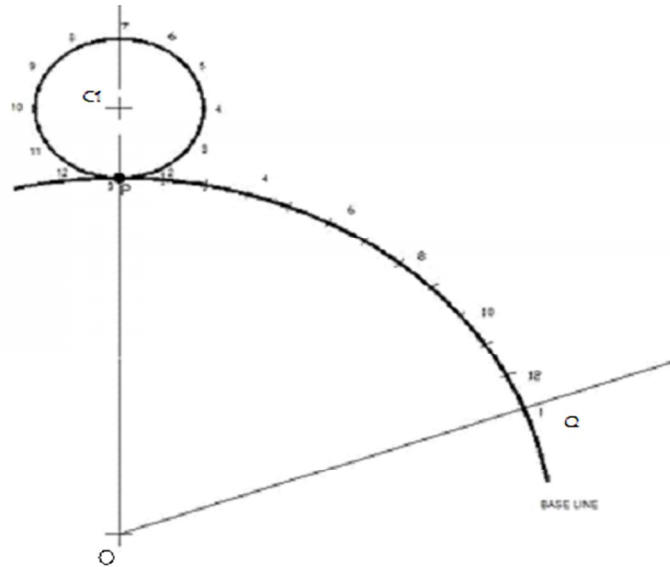
#### **Draw an Epicycloid**

- Draw a directing circle (only a part of it) with given radius  $R=OP=12\text{cm}$
- Draw a generating circle of given radius  $r=4\text{cm}$  touching the directing circle at point P. Divide it into 12 equal parts, labeling them 1 to 12.

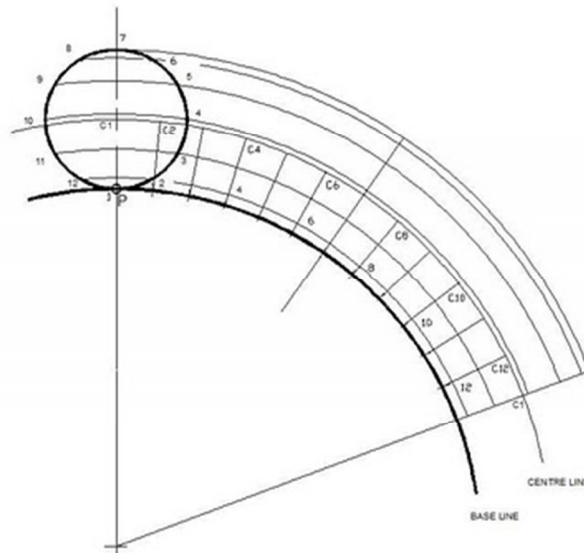
iii.  $\overline{R}$   $Q$ .

## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering



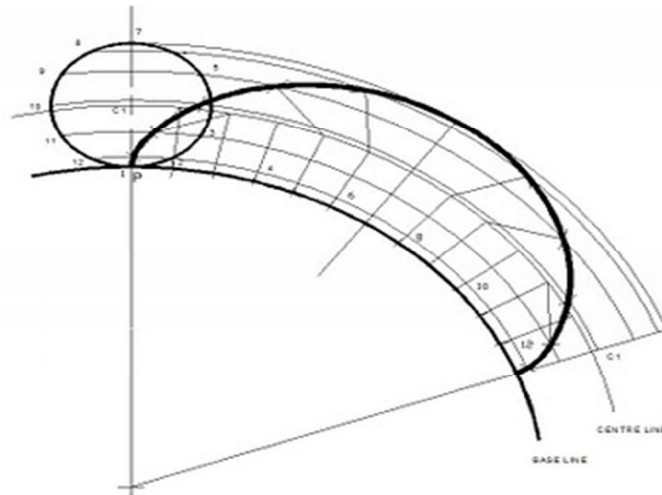
- iv. Draw an arc from the center  $C1$  of the generating circle concentric with the directing circle.
- v.  $Q$  in 12 equal parts) and mark them  $C1$  to  $C12$ .
- vi. With center  $O$  and radius  $O-1$  to  $O-12$ , draw arcs from each of the 12 points on the generating circle.
- vii. With your compass set to the radius of the generating circle and center  $C1$ , cut the height line coming from point 1 on the circle. Then repeat this for point  $C2$  and height line 2, etc. until you reach point  $C12$  and line 12.



- viii. By joining these new points you will have created the locus of the point  $P$  for the circle as it rolls outside the directing circle without slipping. In other words, you have created an **epicycloid**

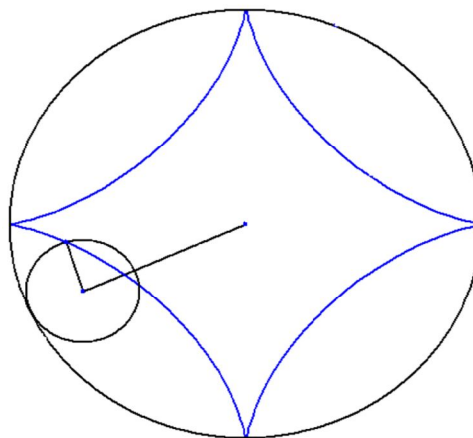
## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering



### HYPOCYCLOIDS

The curve traced by a point on a circle which rolls on the inside of a circular base surface.



#### Exercise 3

Draw a hypocycloid.

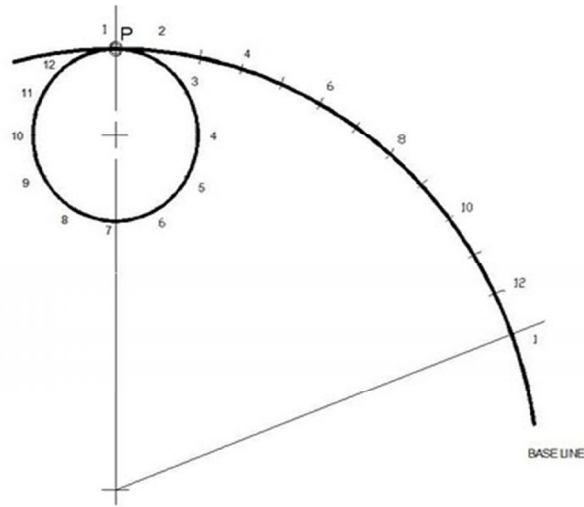
- i. Draw a directing circle (only a part of it) with given radius  $R=OP=12\text{cm}$
- ii. Draw a generating circle of given radius  $r=4\text{cm}$  touching the directing circle at point P from inside. Divide it into 12 equal parts, labeling them 1 to 12.
- iii. Q.

$\overline{R}$

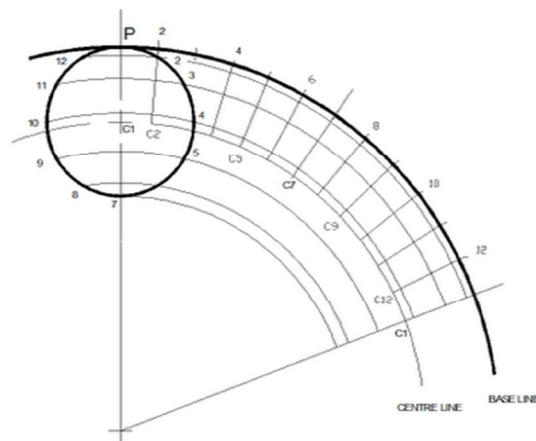


## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering



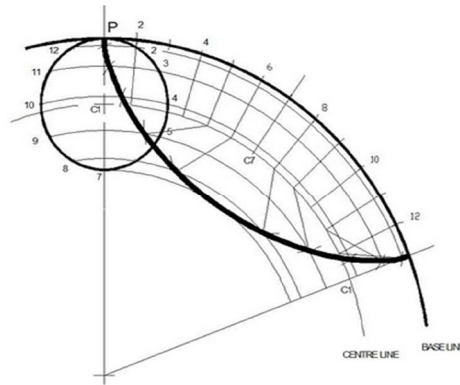
- iv. With center  $O$ , draw an arc from the center  $C1$  of the generating circle.
- v.  $Q$  in 12 equal parts) and mark them  $C1$  to  $C12$ .
- vi. With center  $O$  and radius  $O1$  to  $O12$ , draw arcs from each of the 12 points on the generating circle.



- vii. With your compass set to the radius of the generating circle and center  $C1$ , cut the height line coming from point 1 on the circle. Then repeat this for point  $C2$  and height line 2, etc. until you reach point  $C12$  and line 12.
- viii. By joining these new points you will have created the locus of the point  $P$  for the circle as it rolls outside the directing circle without slipping. In other words, you have created an **hypocycloid**

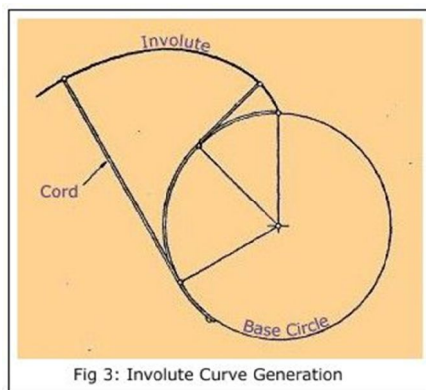
## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering



### INVOLUTE

The involute curve is generated by the locus of a point on a piece of string or cord (for example), when un-wound from a cylinder equivalent in diameter to the base circle, see figure 3. A part of this involute curve forms the gear tooth profile between its junction with the root radius and the gear outside diameter.



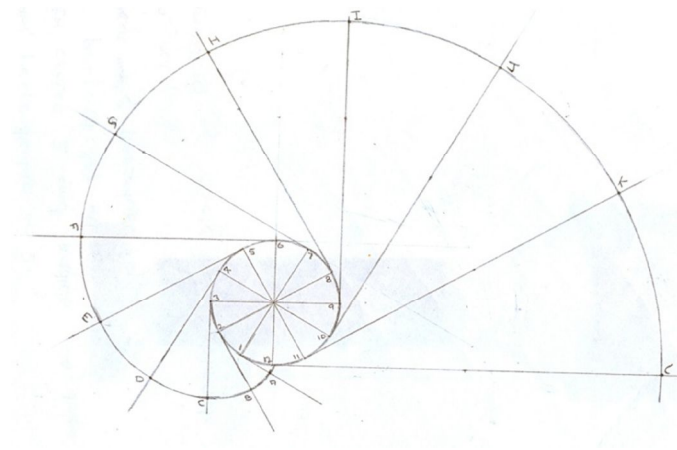
#### **Exercise 4:**

**Draw an involute of a circle.**

- i. Draw a circle of diameter  $D=3.5$  cm and divide it into 12 equal parts.
- ii. Draw a tangent at point 1 on the circle.
- iii. With center 1 and radius 1-12, draw an arc which cut the tangent at A.
- iv. Draw another tangent at point 2 on the circle.
- v. With center 2 and radius 2A, draw an arc which cut the second tangent at B.
- vi. Repeat the procedure taking each point from 3 to 12 as a center.
- vii. Involute of a circle is obtained.

## Engineering Drawing & Workshop

*NED University of Engineering and Technology- Department of Electronic Engineering*



### **HOMEWORK:**

*Draw an involute of a square.*

## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

### Lab Session 05

#### OBJECTIVE:

*To draw Nuts, bolts and screws*

*Rivets*

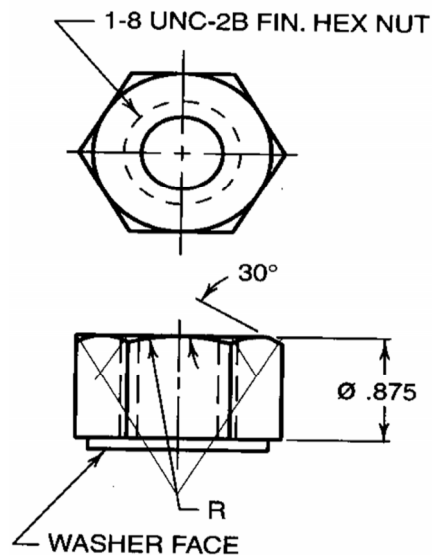
*Threads*

#### Nuts:

A nut is a mechanical threaded device used on the ends of bolts, studs and machine screws. Various types of nuts are used for different applications. The most common types are hex and square nuts.

#### Exercise 1

**To draw a Hex head Nut**



**Bolts:** A mechanical threaded device with a head on one end and threads on the other end. Bolts are paired with nuts.

**Studs:** A rod that is threaded on both ends and joins two mating parts. A nut may be used on one side.

**Cap screws:** A mechanical threaded device with a head on one end and threads on the other end. They join two mating parts and have longer threads than bolts. They can be made with slotted heads.

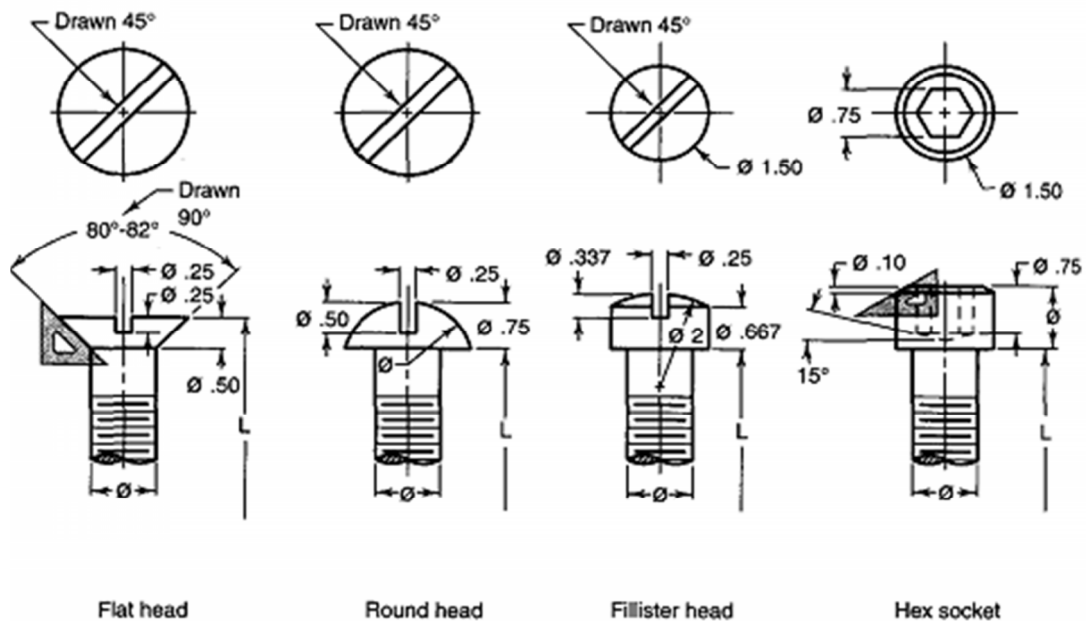
**Machine screws:** A mechanical threaded device with a head on one end and threads on the other end. The threaded end may screw into a mating part, or may be used with a nut. A machine screw is similar to cap screw, but it is normally smaller.

#### Exercise 2

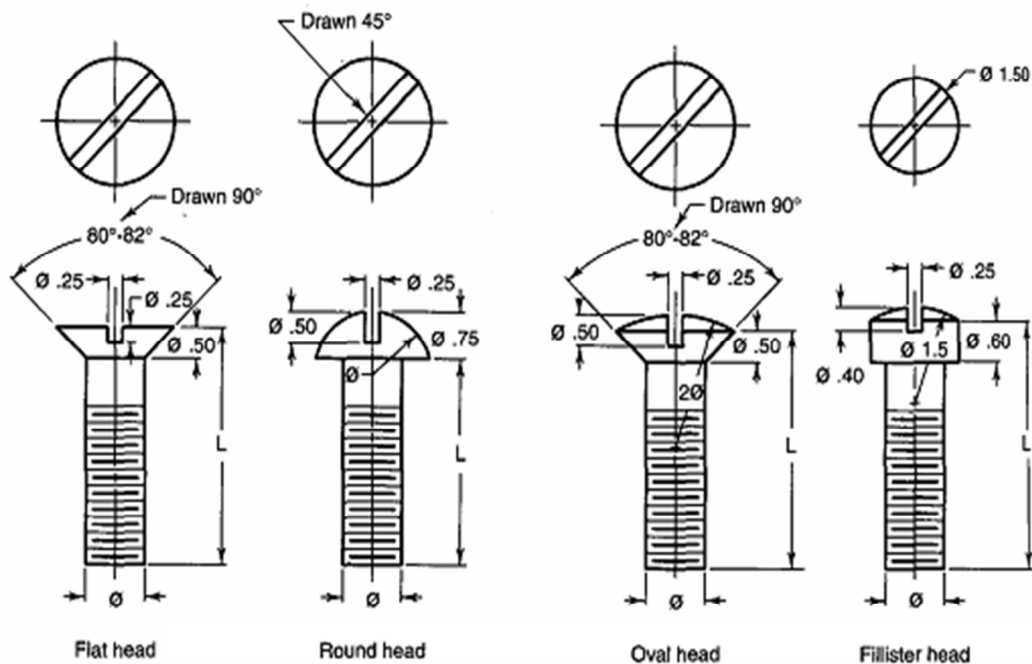
**To draw different types of cap screws**

## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering



### Exercise 3: To draw different types of machine screw



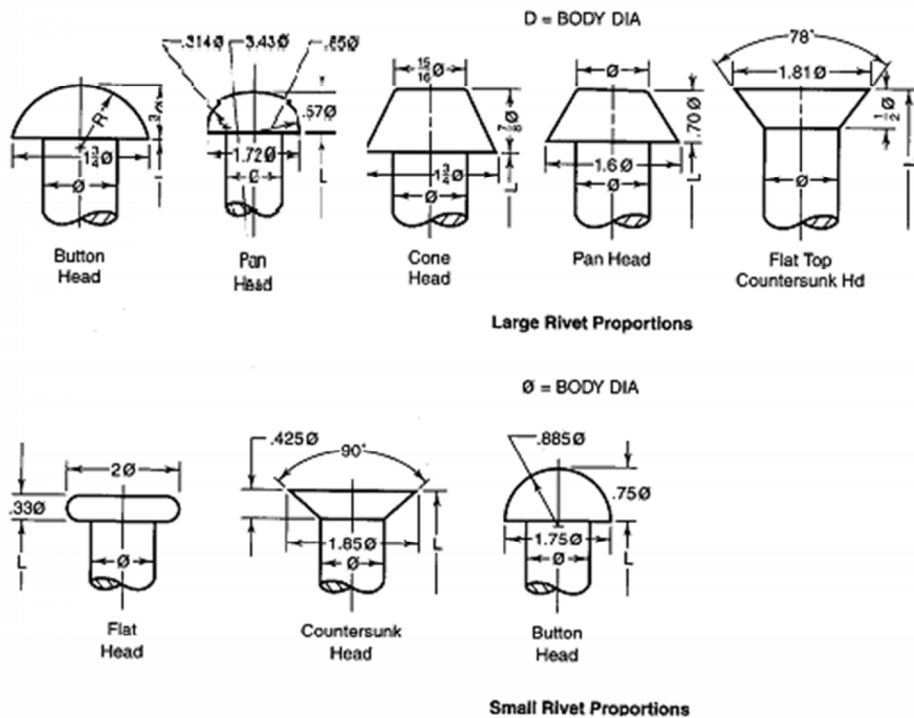
### Rivets

Rivets are metal pins with a head and are used to attach assembled parts permanently. Rivets are available in a variety of head styles. They are generally used on sheet metal, such as the skin of an aircraft attached to the frame, or ship parts. The hole for the rivets are drilled or punched, then rivet is held in place with a tool called dolly, while the other end of the rivet is hammered, pressed or forged in place.

## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

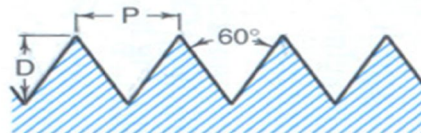
### Exercise 4: To draw different types of machine screw



### Threads:

#### Sharp V

The 60 degree Sharp-V thread was originally called the United States Standard thread, or the Sellers thread. For purposes of certain adjustments, the Sharp-V thread is useful with the increased friction resulting from the full thread face. It is also used on brass pipe work.



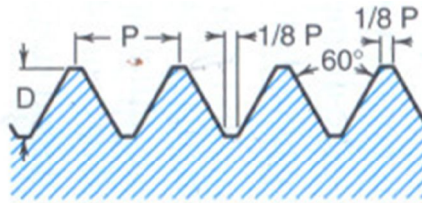
(A) Sharp V

#### American National

The American National thread with flattened roots and crests is a stronger thread. This form replaced the Sharp-V thread for general use and is still used for existing design.

## Engineering Drawing & Workshop

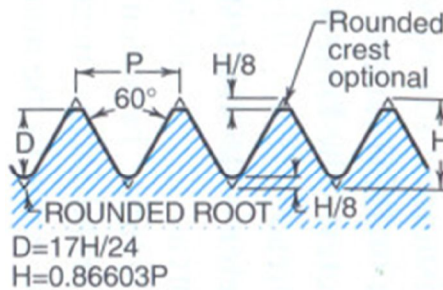
NED University of Engineering and Technology- Department of Electronic Engineering



(C) American National

### Unified (External)

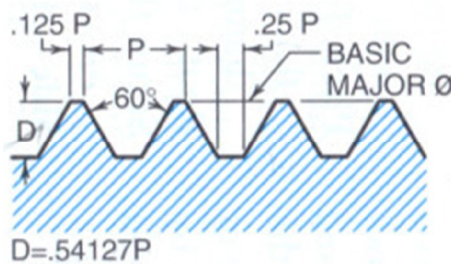
The Unified thread is the standard thread agreed upon by the United States, Canada, and Great Britain in 1948. The crest of the external thread may be flat or rounded, and the root is rounded; otherwise, the thread form is essentially the same as the American National.



(D) Unified (external)

### Metric

The metric thread is the new standard screw thread agreed upon for international screw thread fasteners. The crest and root are flat, but the external thread is often rounded when formed by the rolling process. The form is similar to the American National and the Unified threads but with less depth of thread.



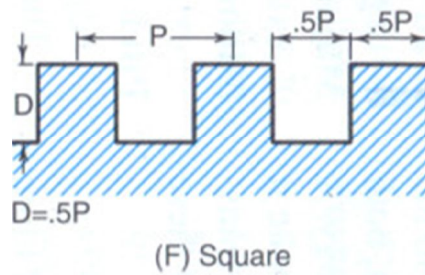
(E) Metric

### Square

The Square thread is theoretically the ideal thread for power transmission, since its face is nearly at right angles to the axis, but owing to the difficulty of cutting it with dies and because of other inherent disadvantages, such as the fact that split nuts will not readily disengage, the square thread has been replaced to a large extent by the Acme thread. The square thread is not standardized.

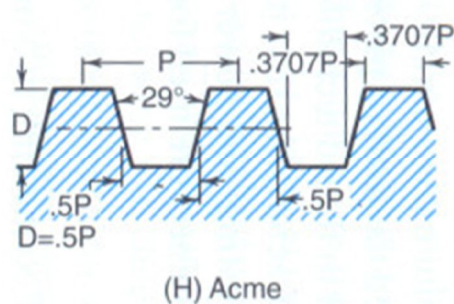
## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering



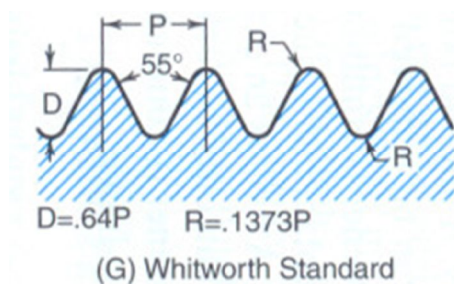
### Acme

The Acme thread is a modification of the square thread and has largely replaced it. It is stronger than the square thread, is easier to cut, and has the advantage of easy disengagement from a split nut, as on the lead screw of a lathe.



### Whitworth Standard

The Whitworth thread has been the British standard and is being replaced by the unified thread. Its uses correspond to those of the American National thread.



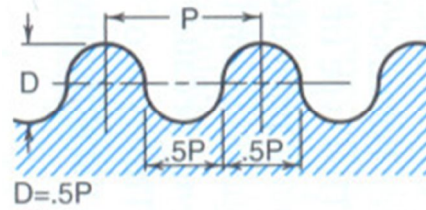
### Knuckle

The knuckle thread is usually rolled from sheet metal but is sometimes cast, and is used in modified forms in electric bulbs and sockets, bottle tops and the like.



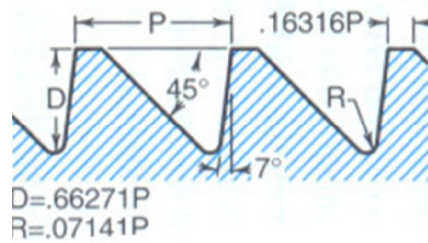
## Engineering Drawing & Workshop

*NED University of Engineering and Technology- Department of Electronic Engineering*



### Buttress

The buttress thread is designed to transmit power in one direction only and is used in the breech locks of large guns, in jacks, in airplane propeller hubs, and in other mechanisms



## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

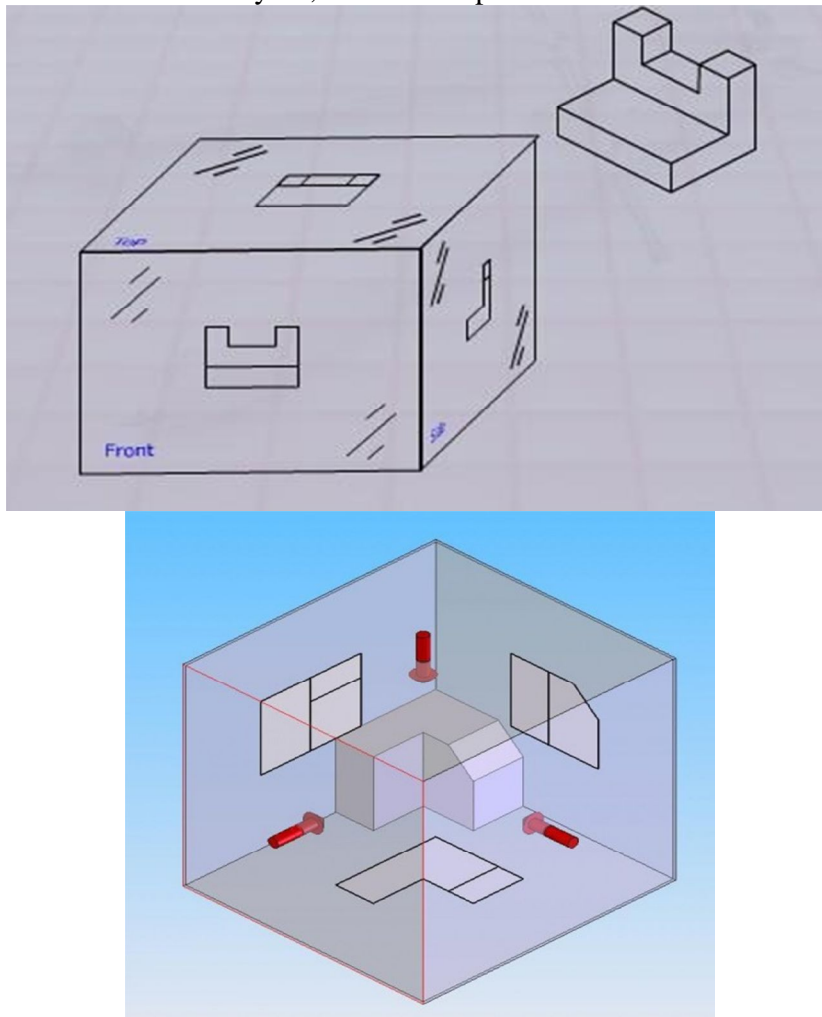
### Lab Session 06(A)

## OBJECTIVES:

*To draw orthographic views of the given machine parts by first and third angle projection methods*

### Orthographic projection

Orthographic projection refers to a 2-D representation of a 3-D object in a view that shows only one side at a time. Most orthographic drawings occur in multi-drawing sets in order to depict each side, top and bottom view. Professions in design and construction use these types of drawings to inform the viewer of layout, size and shape.

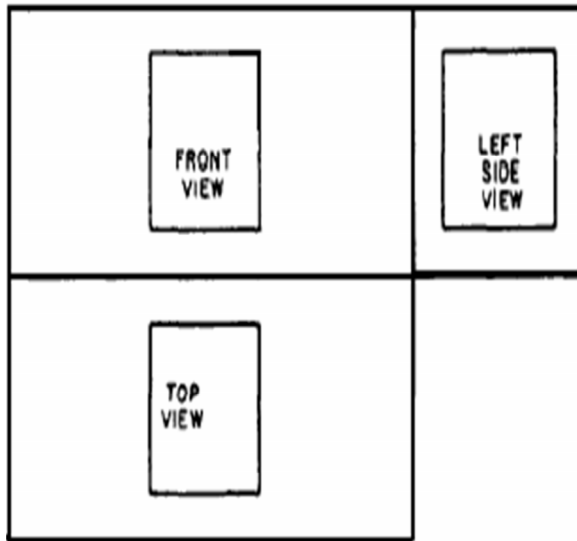


### Methods of Orthographic Projection

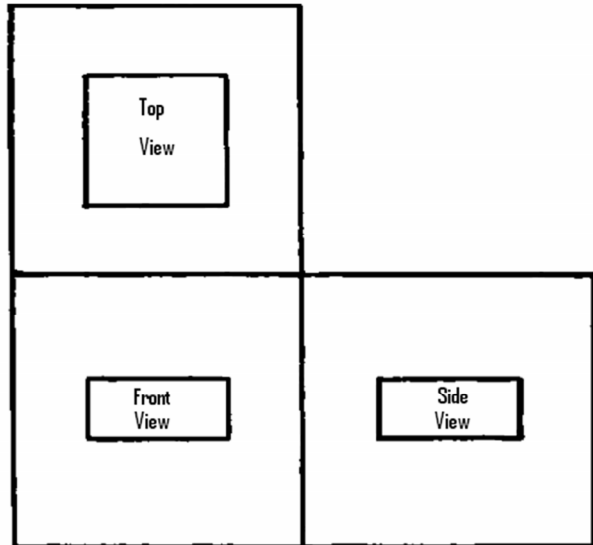
There are two ways of drawing in orthographic - **First Angle** and **Third Angle**. They differ only in the position of the top, front and side views.

## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

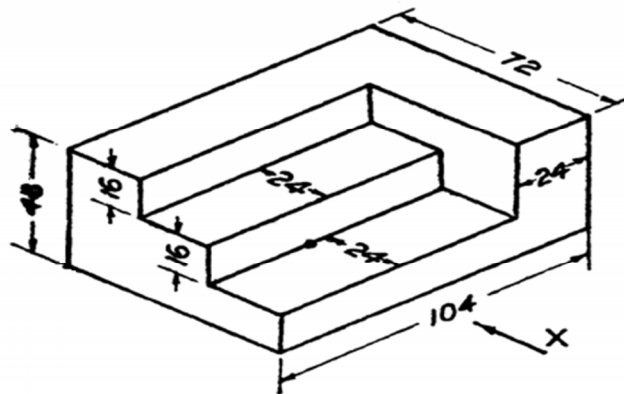


First angle orthographic projection



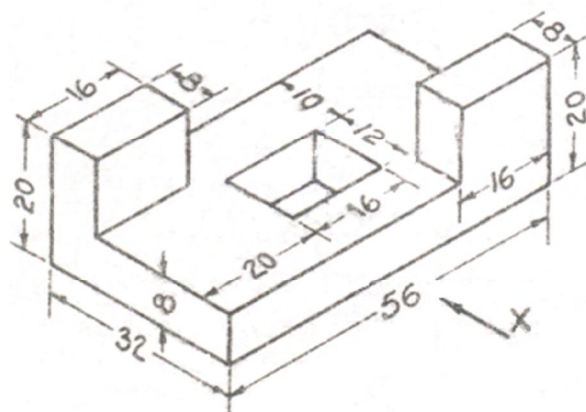
Third angle orthographic

**Exercise 1:** Draw orthographic projection of the given object by first angle projection method.



### HOMEWORK

Draw orthographic projection of the given object by first angle projection method.

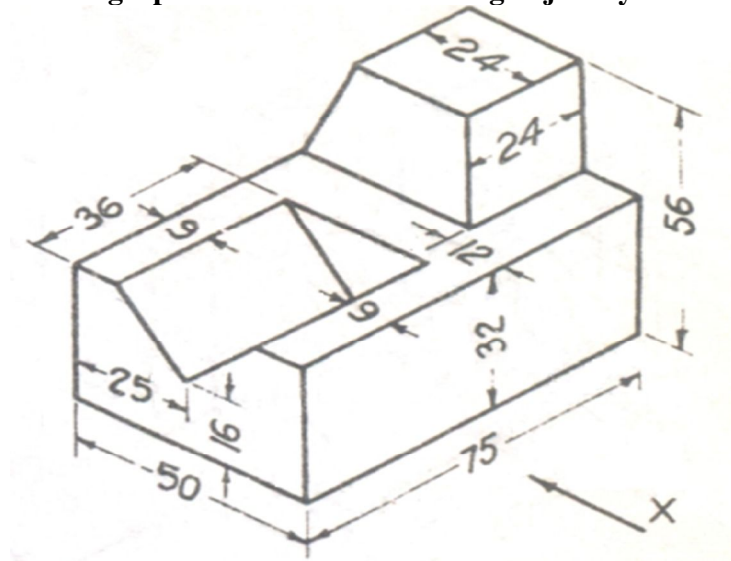


## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

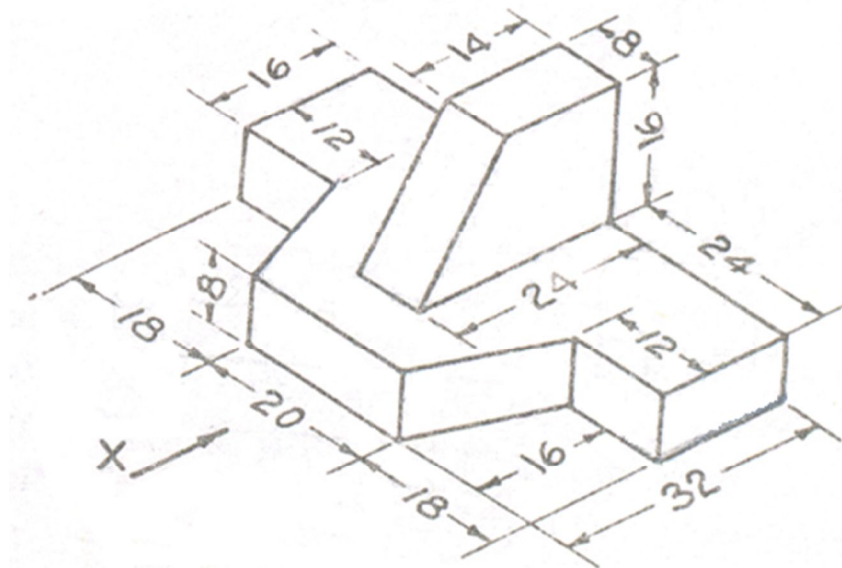
### Lab Session 06(B)

Exercise 1: To draw orthographic views of the following object by first angle method



### HOMEWORK

To draw orthographic views of the following object by first angle method

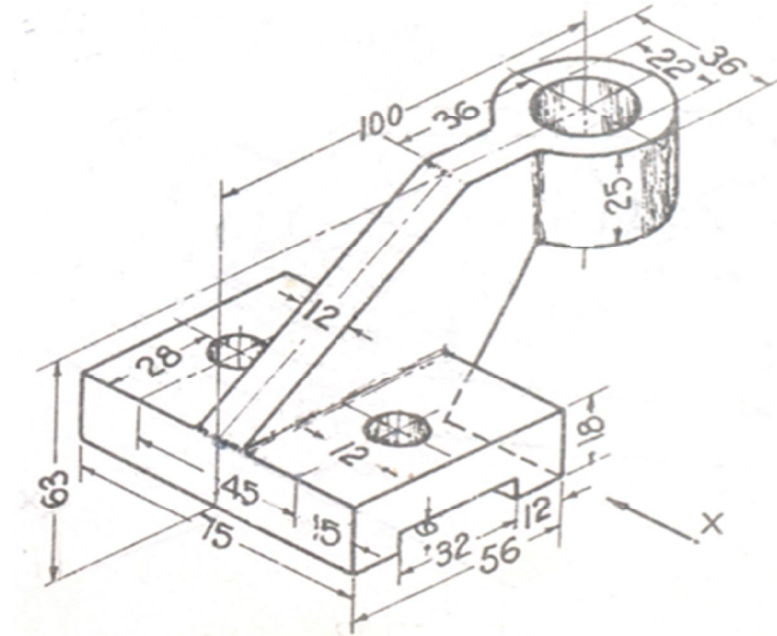


## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

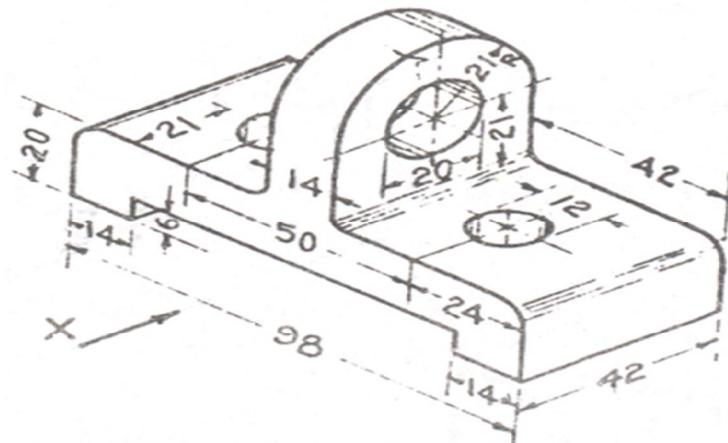
### Lab Session 06(C)

Exercise 1: To draw orthographic views of the following object by first angle method



### HOMEWORK

To draw orthographic views of the following object by first angle method

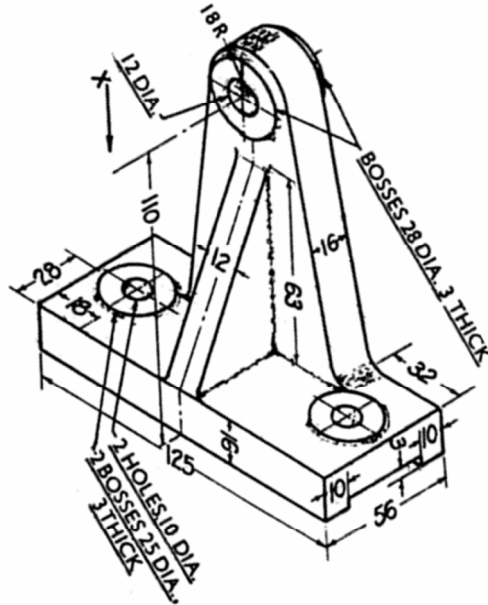


## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

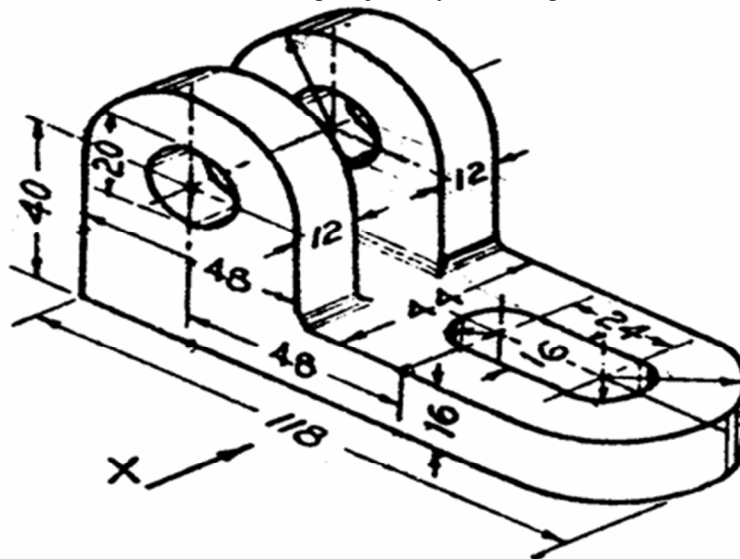
### Lab Session 06(D)

Exercise 1: To draw orthographic views of the following object by first angle method



### HOMEWORK

To draw orthographic views of the following object by first angle method



## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

### Lab Session 07

#### OBJECTIVE:

*To draw circles in oblique and isometric cubes*

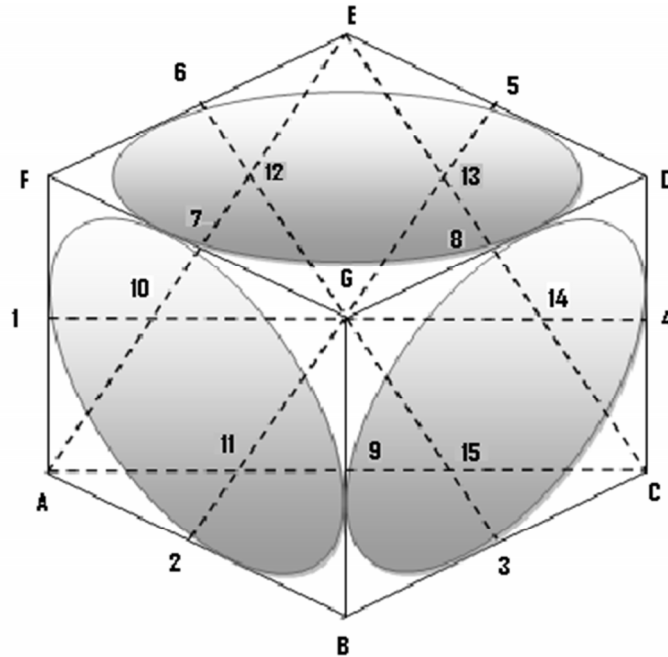
#### Exercise 1: To draw circles in an isometric cube

##### Steps of construction:

1. Draw an isometric cube ( $30^\circ$ ) each of whose sides measures 10 cm. Name all its vertices as A, B, C etc.
2. Bisect AF at point 1. Extend this bisector with the help of T-square, passing through G and bisecting CD at 4.
3. Join A to C bisecting BG at 9.
4. Similarly bisect AB at point 2. Extend this bisector with the help of T-square, passing through G and bisecting ED at 5.
5. Join A to E bisecting FG at 7.
6. Bisect BC at point 3. Extend this bisector with the help of T-square, passing through G bisecting EF at 6.
7. Join C to E bisecting DG at 8.
8. On face ABFG with center 10 and radius 1-10 draw an arc 1-7. Repeat this step with center 11.
9. With center A and radius A-7 draw an arc 7-9. Repeat this step with center G. An ellipse is obtained.
10. Complete the two ellipses on the remaining faces with the same method.

## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering



### Exercise 2: To draw circles in an isometric cube

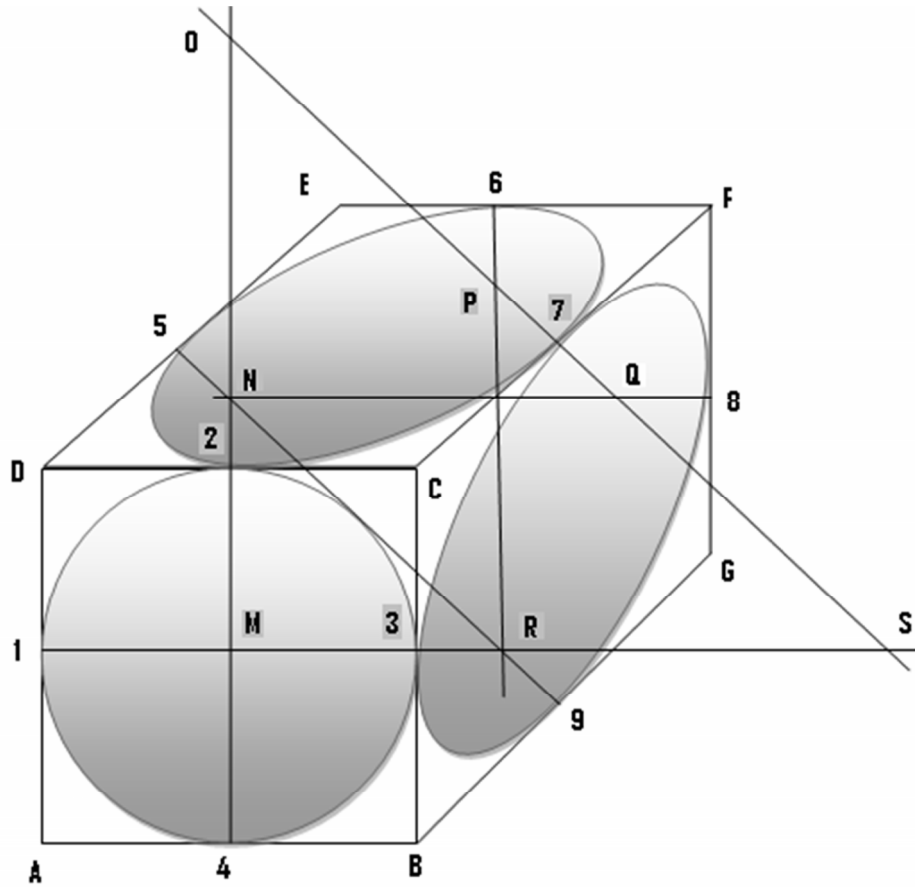
#### Steps of construction

1. Draw an oblique cube ( $45^\circ$ ) each of whose sides measures 10 cm. Name all its vertices as A, B, C etc.
2. Draw bisectors of all sides of the cube and mark the bisecting points as 1, 2, 3 etc.
3. Join 1-3 and 2-4 intersecting at M and extend them.
4. With center M and radius M-1, draw a circle.
5. Join 5-9 intersecting the extended lines at N and R.
6. With the help of a T-square, draw a horizontal line from 8 extending towards left meeting N.
7. With the help of a set-square, draw a perpendicular from 6 on R.
8. With center N and radius N-9 and N-5, draw arcs 8-9 and 2-5 respectively.
9. Similarly with center R and radius R-5 and R-3, draw arcs 5-6 and 3-9 respectively.
10. With the help of a  $45^\circ$  set-square, draw a line through 7 intersecting the extended lines at O, P, Q and S.
11. With center O and radius O-2 draw an arc 2-7. Similarly with center S and radius S-3 draw an arc 3-7.
12. With center P and radius P-6 draw an arc 6-7. Similarly with center Q and radius Q-7 draw an arc 7-8



## Engineering Drawing & Workshop

*NED University of Engineering and Technology- Department of Electronic Engineering*



## Engineering Drawing & Workshop

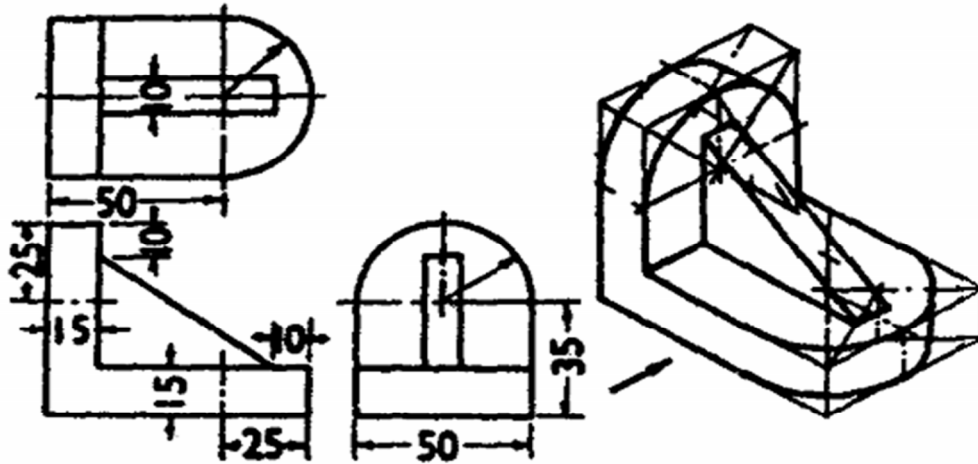
NED University of Engineering and Technology- Department of Electronic Engineering

### Lab Session 08(A)

#### OBJECTIVES:

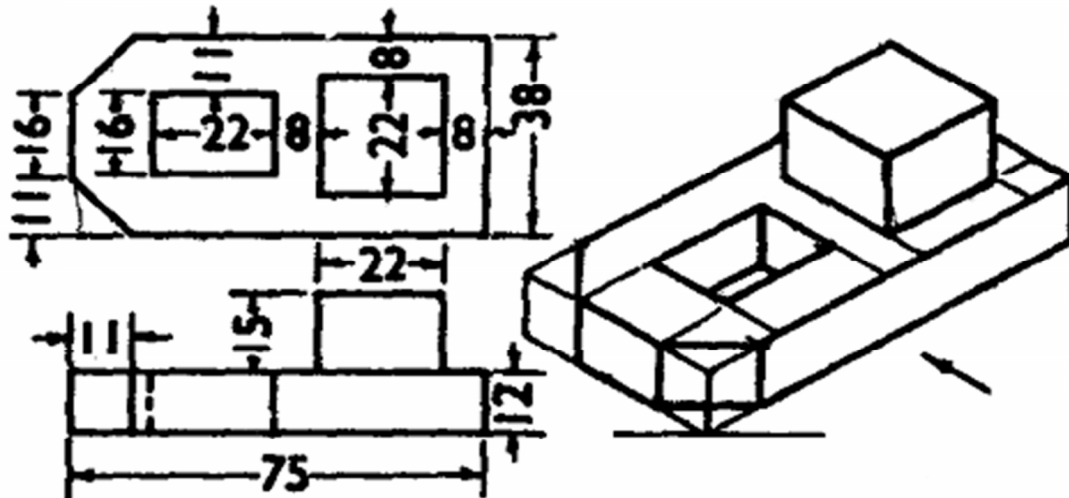
To draw isometric view from the given orthographic views

Exercise 1: To draw isometric view from the given orthographic views



#### HOMEWORK

To draw isometric view from the given orthographic views

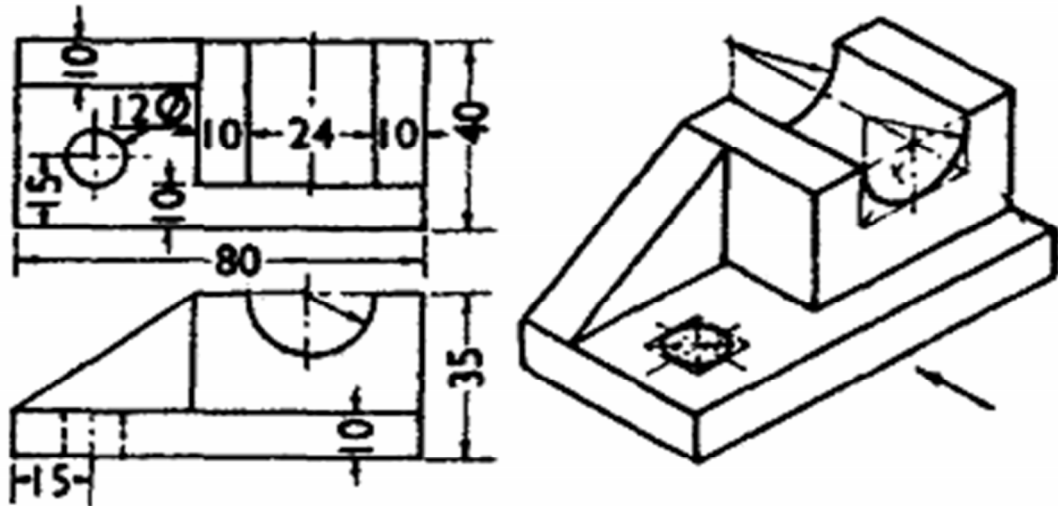


## **Lab Session 08(B)**

### **OBJECTIVES:**

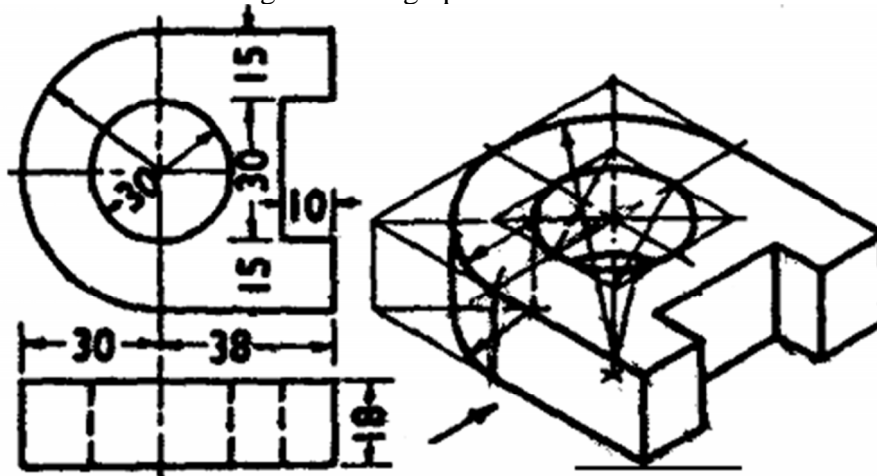
*To draw isometric view from the given orthographic views*

**Exercise 2: To draw isometric view from the given orthographic views**



### **HOMEWORK**

*To draw isometric view from the given orthographic views*



## Engineering Drawing & Workshop

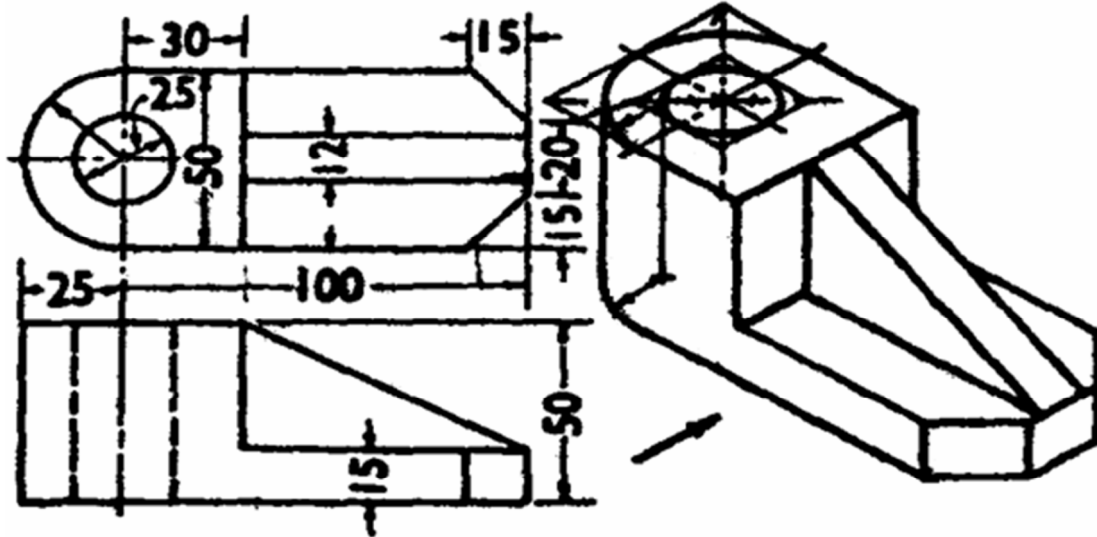
NED University of Engineering and Technology- Department of Electronic Engineering

### Lab Session 08(C)

#### OBJECTIVES:

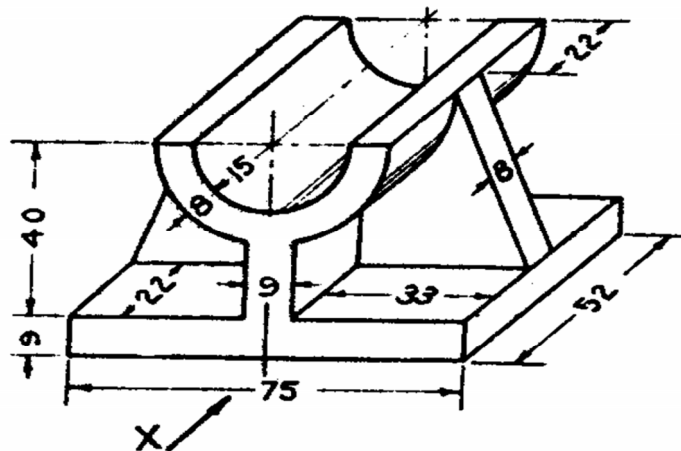
To draw isometric view from the given orthographic views

Exercise 3: To draw isometric view from the given orthographic views



#### HOMEWORK

To draw isometric view from the given orthographic views



## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

### Lab Session 09

#### OBJECTIVES:

*To draw the development of lateral surfaces of*

*Pyramid*

*Prism*

*Cylinder*

#### **PYRAMID**

A pyramid is a structure where the outer surfaces are triangular and converge at a point. The base of a pyramid can be trilateral, quadrilateral, or any polygon shape, meaning that a pyramid has at least three triangular surfaces (at least four faces including the base). The square pyramid, with square base and four triangular outer surfaces, is a common version.



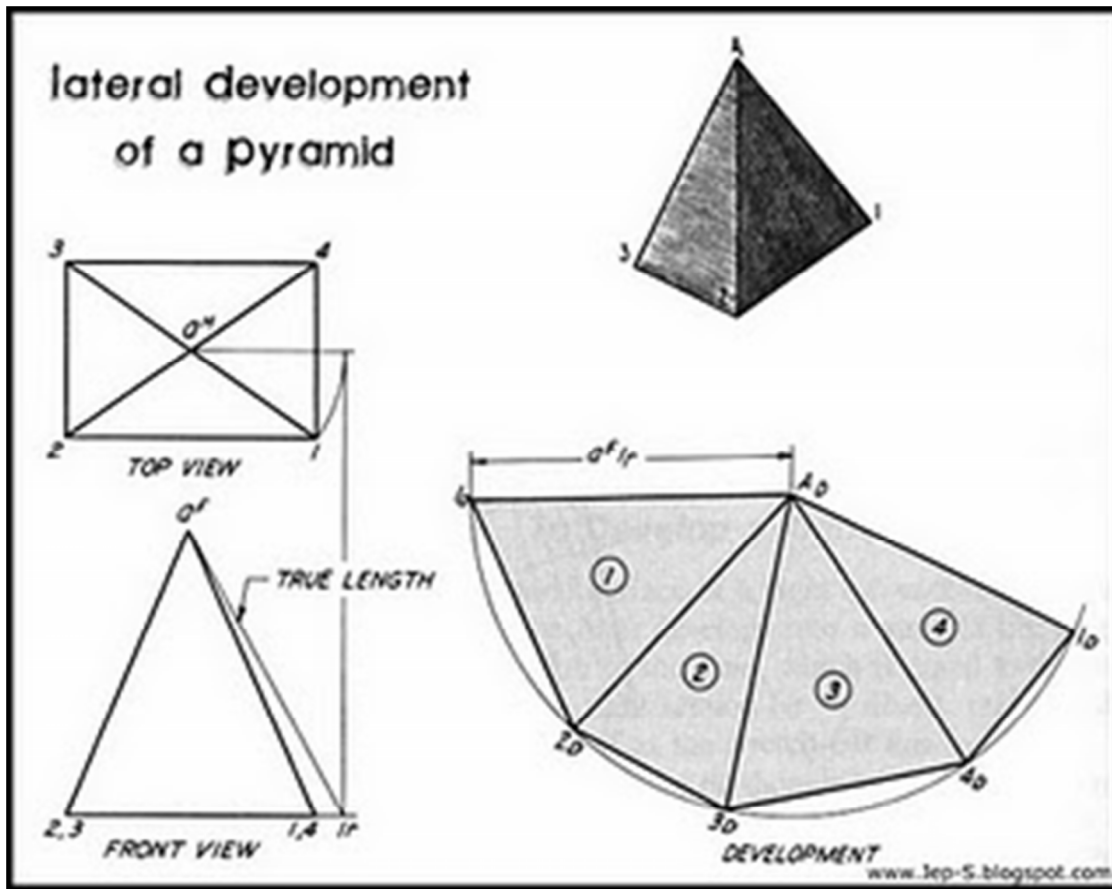
#### **Square pyramid**

##### **Steps for square pyramid**

1. With any point O as center and radius equal to the true length of the slant edge of the pyramid, draw an arc of the circle.
2. With radius equal to the true length of the side of the base, divide the arc into same number of divisions as the number of sides of the base.
3. Join these division-points with each other in correct order and also with the center of the arc.
4. The figure thus obtained is the development of the lateral surface of the pyramid.

## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering



### Exercise 1: Draw the development of the lateral surface of the truncated square pyramid.

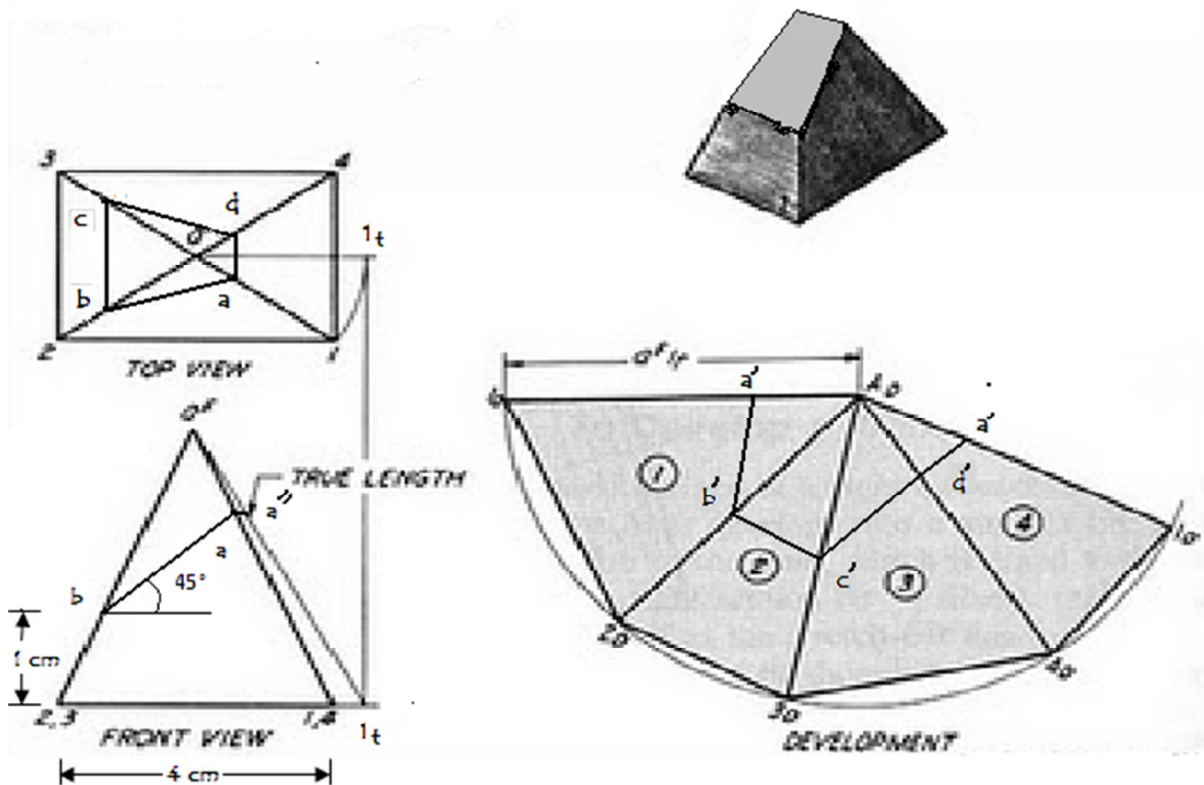
- Draw the top view of the truncated square pyramid and name all the corners as 1, 2, 3, 4 and center as  $O$ .

#### ○ TRUE LENGTH

- In top view, draw a line from  $O$  parallel to the base of the pyramid
- With center  $O$  and radius  $O1$ , draw an arc cutting the line drawn in step 2 at  $1t$ .
- Draw projection of  $1t$  base of the front view. Join  $a^F$  to  $1t$  to obtain the true length of the edge.

## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering



### DEVELOPMENT

1. With any point AD as center and radius equal to the true length of the slant edge of the pyramid  $a^F 1_t$ , draw an arc of the circle.
2. With radius equal to the true length of the side of the base, divide the arc into same number of divisions as the number of sides of the base and mark them as 1D, 2D etc.
3. Join these division-points with each other in correct order and also with the center of the arc.
4. Mark a point  $a'$  on AD1D and  $d'$  on AD4D such that  $a'AD = d'AD = a^F a''$ .
5. Mark a point  $b'$  on AD2D and  $c'$  on AD3D such that  $b'AD = c'AD = a^F b$ .
6. Join the points  $a'$ ,  $b'$ ,  $c'$  and  $d'$  and complete the development.

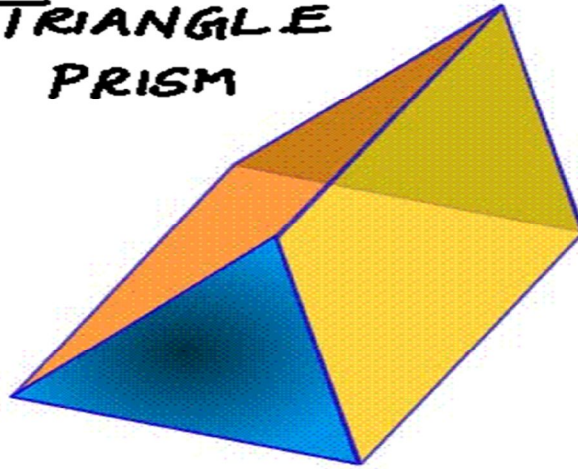
### PRISM

A polyhedron with two congruent and parallel faces (the bases) and whose lateral faces are parallelograms. The cross section is the same all along its length. The shape of the ends gives the prism a name, for example this is a "triangular prism" because it has triangular ends.

## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

### TRIANGLE PRISM



#### Exercise 2

**Draw the development of the lateral surface of the truncated Pentagonal prism**

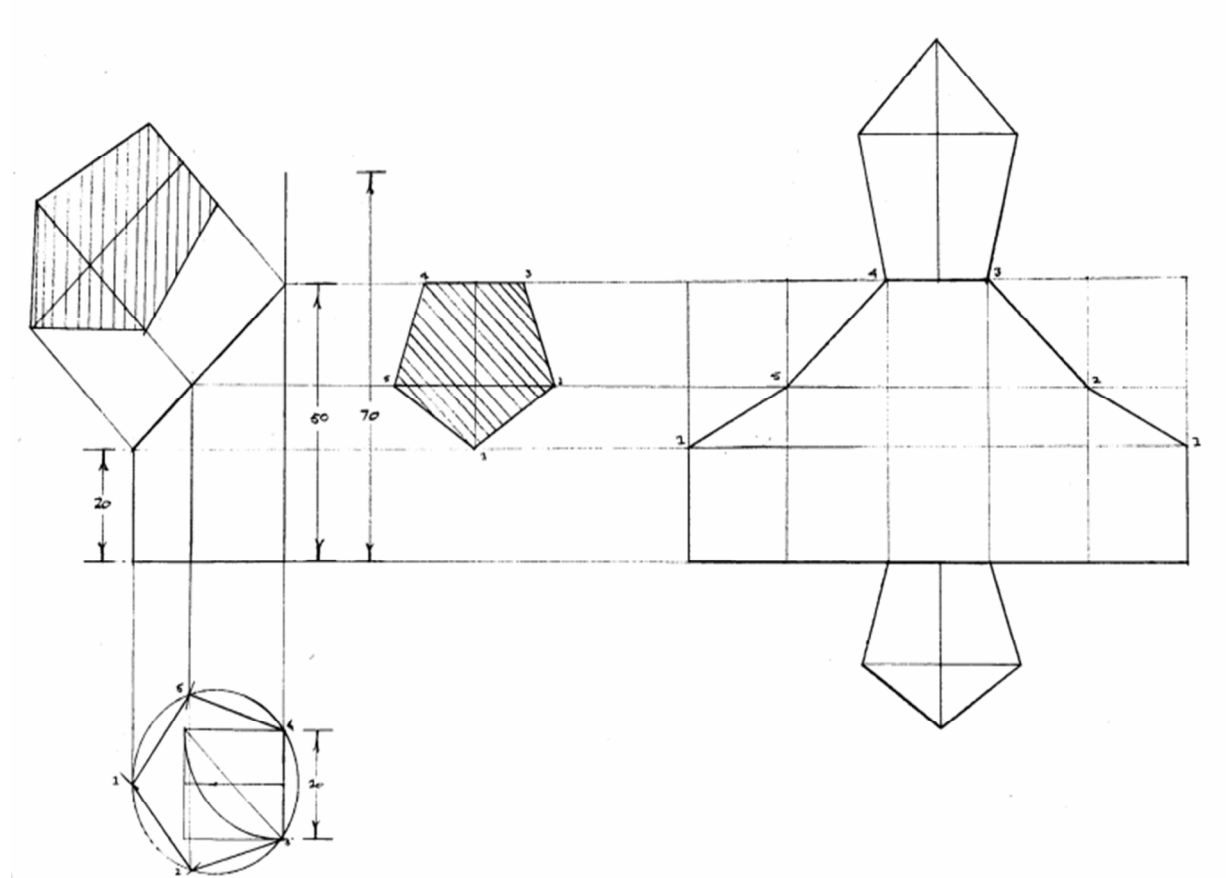
##### Steps

1. Name the corners of the prism and the points at which edges are cut.
2. Draw the development of the lateral surface of the prism, by drawing a rectangle whose width is equal to the height of the prism and length equal to the sum of lengths of the sides of its base.
3. Divide the rectangle into 5 equal rectangles vertically such that the horizontal side of each is equal to the side of the base of the prism.
4. Draw horizontal lines starting from points 1', 2' etc to cut the vertical lines for the corresponding edges in the development at points 1, 2 etc.
5. Join the points 1, 2 etc and complete the development.



## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering



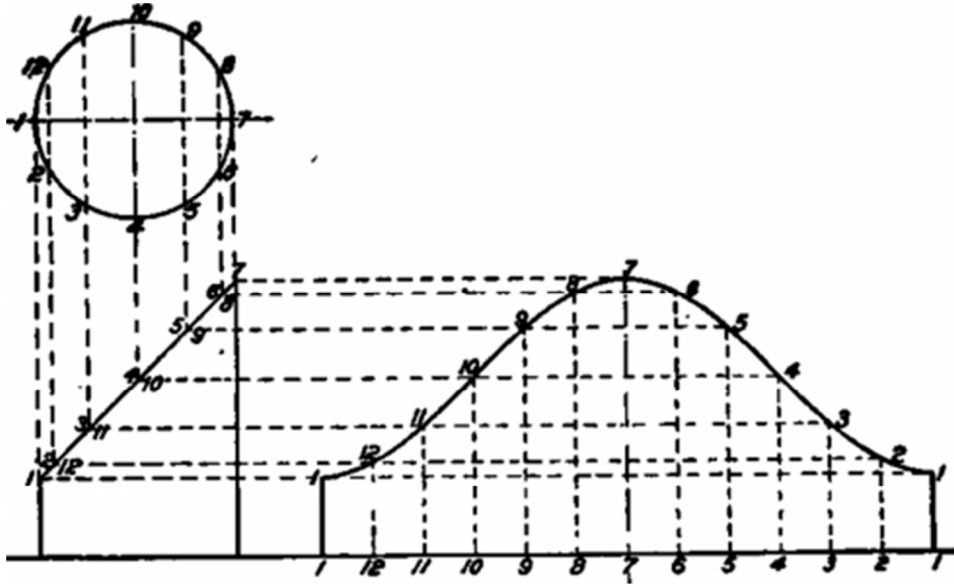
### CYLINDER

#### Exercise 3: Draw the development of the lateral surface of the truncated Circular Cylinder

1. Draw the base of the cylinder of radius 2 cm and divide it into 12 equal parts 1, 2, 3, etc.
2. Draw in elevation the elements corresponding to the various divisions of the base, and note the points where they intersect the oblique plane.
3. Draw the development of the lateral surface of the cylinder, by drawing a rectangle whose width is equal to the height of the cylinder and length equal to the circumference of the its base.
4. Divide the rectangle into 12 equal parts.
5. Project horizontal lines from each point on the oblique plane of the front view, to cut the corresponding lines in the development.

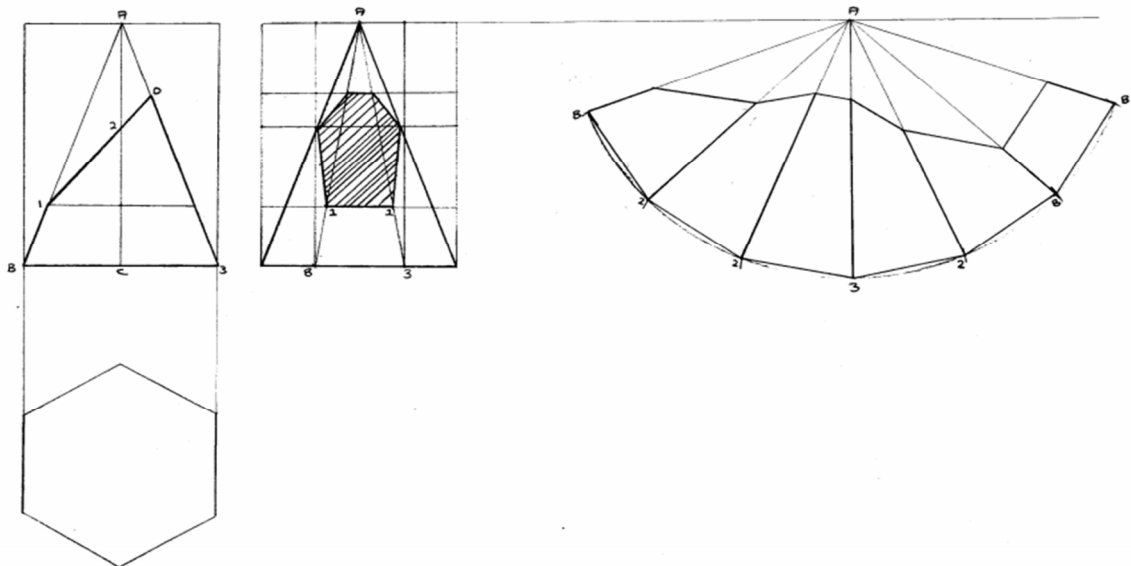
## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering



### HOMEWORK:

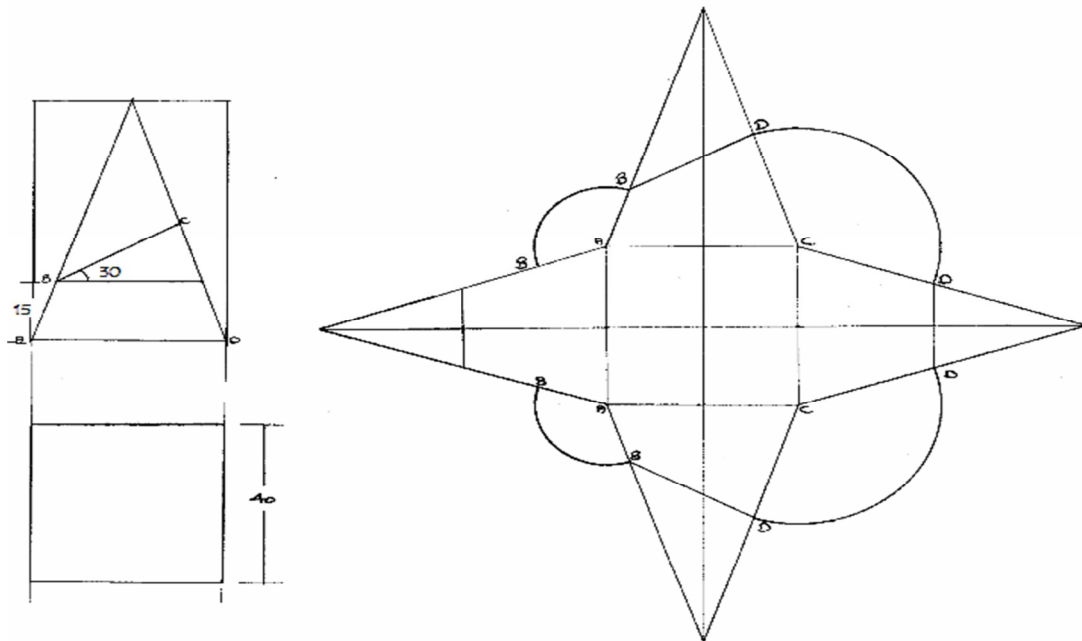
1. Draw the development of the lateral surface of the truncated hexagonal pyramid.



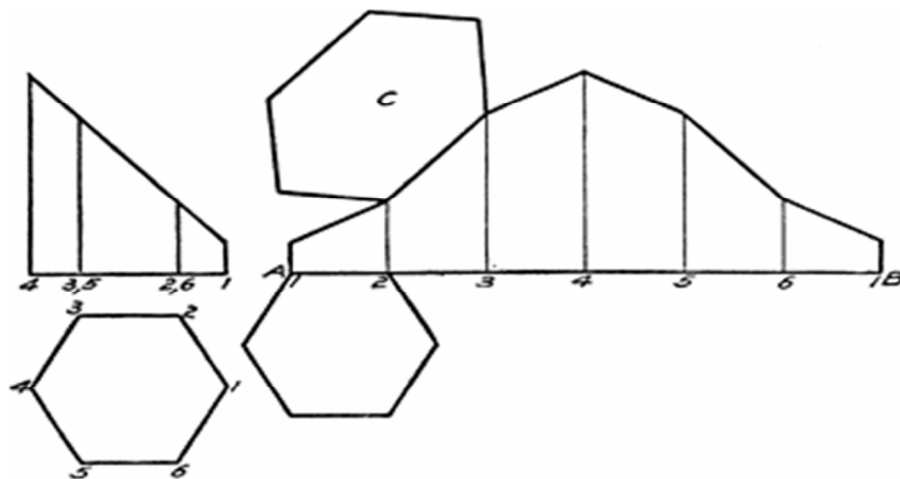
2. Draw the development of the lateral surface of the truncate square pyramid by star method.

## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering



3. Draw the development of the lateral surface of the truncated hexagonal prism.



## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

### Lab Session 10

#### OBJECTIVES:

To learn standard symbols used in civil or architectural drawings

To make architectural drawing of different buildings

#### Some Standard Symbols



SINGLE DOOR



BIFOLD DOOR



SLIDING DOOR



DOUBLE DOOR



POCKET DOOR



WINDOW



VENTILATOR



CURTAIN



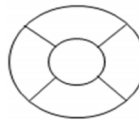
WINDOW WITH  
FINS



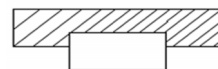
WASH BASIN



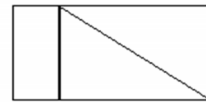
BATH TUB



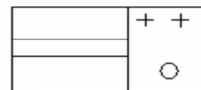
SHOWER



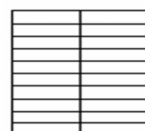
ALMIRAH



BED



SINK WITH  
DRAIN BOARD



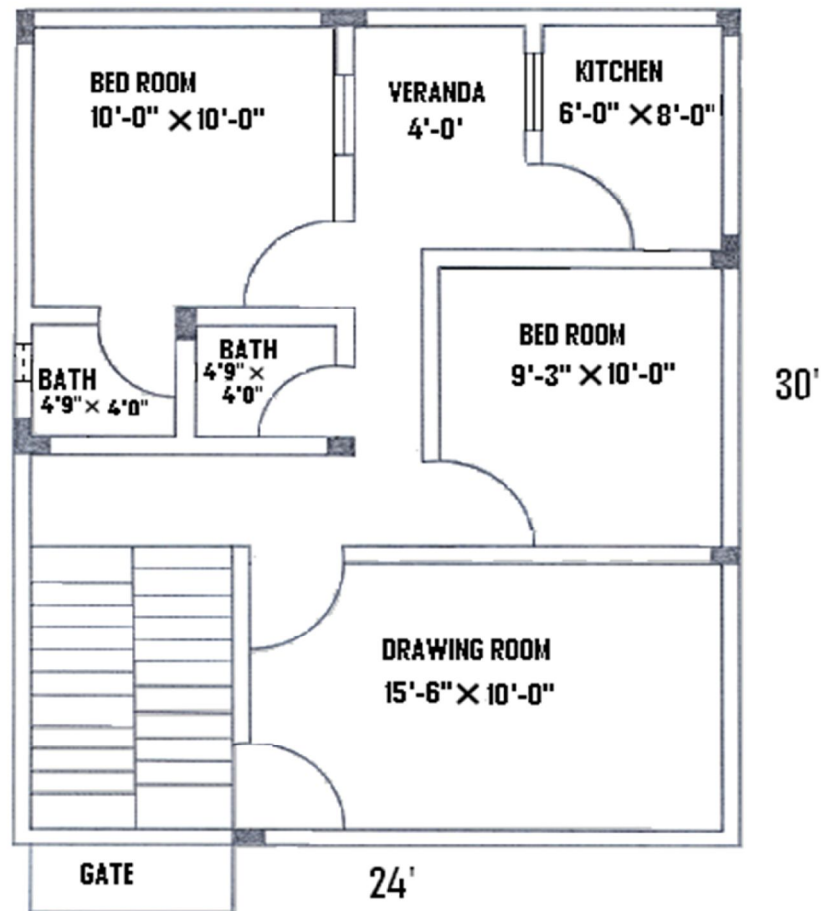
STAIRS

#### Exercise 1

Draw the following architectural plan of 80 sq-yards bungalow according to given measurements.

## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering



### HOMEWORK

1. Draw the architectural plan of your home.
2. Draw the architectural plan of a Telecommunication office.
3. Draw the architectural plan of the first floor of HPCC.

## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

### Lab Session 11(A)

#### OBJECTIVE:

- To learn standard electrical symbols

#### Standard Electrical Wiring Diagram Symbol

##### Symbols for Lighting



Surface Ceiling Light



Recessed Ceiling Light



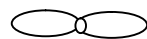
Drop Cord Light



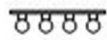
Landscape Light



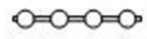
Wall Light



Ceiling Fan



Multi-light Bar



Track Light



Tube Light



Surface Fluorescent Light

##### Symbols for Switches



Switch



3-Way Switch



4-Way Switch

## Engineering Drawing & Workshop

*NED University of Engineering and Technology- Department of Electronic Engineering*

### Symbols for Outlets

 Outlet

 GFI Outlet

 Switched Outlet

 Waterproof Outlet

 Quad Outlet

 Floor Outlet


 240 Volt Outlet

### Symbols for Signals & Communications

 Smoke Detector

 Telecom Outlets






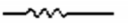

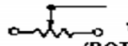







 Stereo Outlet

 Television Outlet

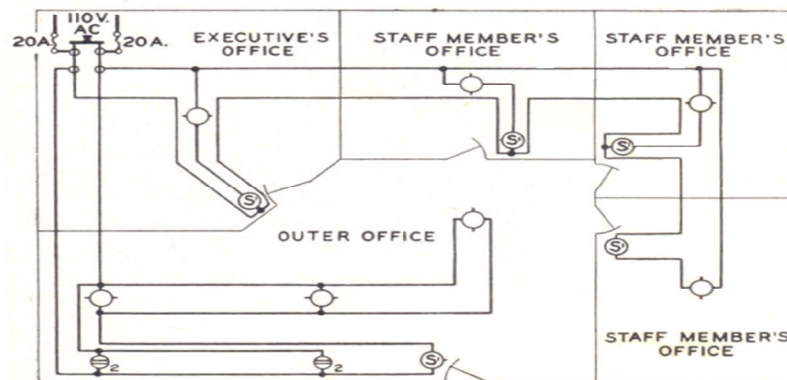
 Doorbell Button

## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

— WIRE	 LAMP INCANDESCENT
CONDUCTORS	 FUSE
 CONNECTED	RESISTORS
 CONNECTED	
 NOT CONNECTED	 FIXED
 GROUND	 VARIABLE (POTENTIOMETER)
	 RHEOSTAT
 CELL	 SWITCH
 BATTERY	 VOLTMETER
 OR	 AMMETER

**Exercise 1:** To draw detailed wiring plan for the lighting of an executive's office.





## **Lab Session 11(B)**

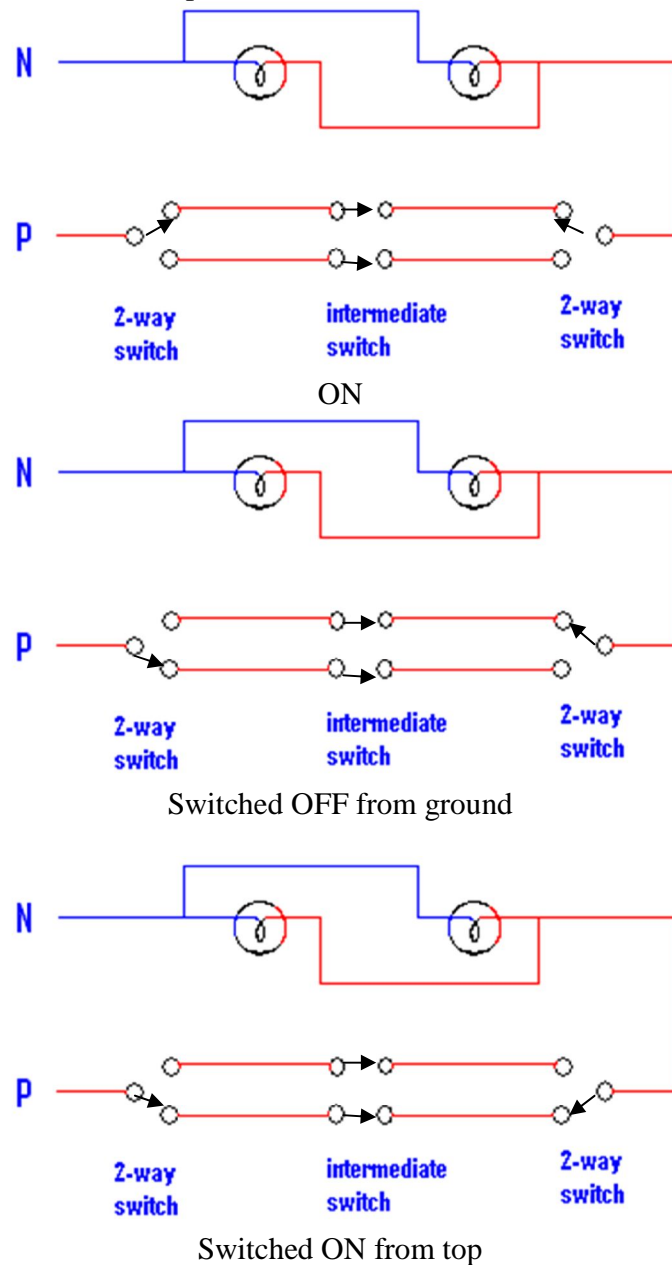
### **OBJECTIVE:**

*To control one or more bulbs from different places*

*To design a tunnel lighting circuit*

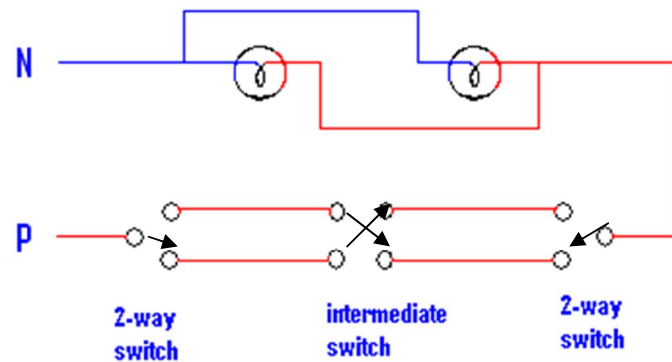
#### **Exercise 1**

To control two bulbs from three places

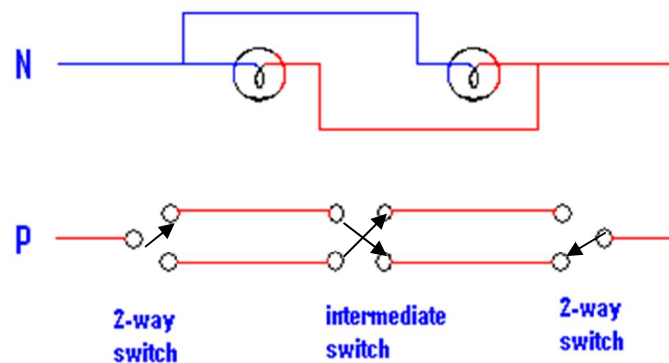


## Engineering Drawing & Workshop

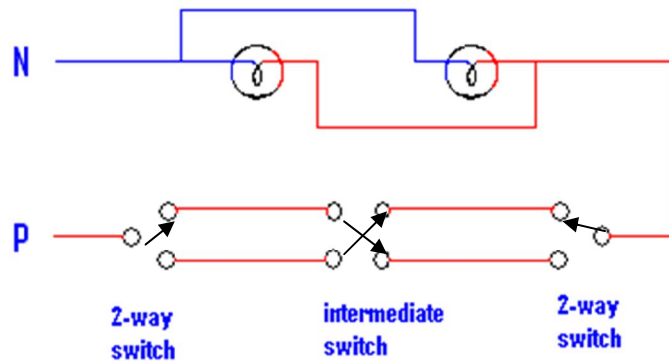
NED University of Engineering and Technology- Department of Electronic Engineering



Switched OFF from middle



Switched ON from ground



Switched OFF from top

**Exercise 2:** To install tunnel lighting, if we are walking down a tunnel. 5 switches and 4 lights are installed at 2m intervals. The first switch turns the first light on, the second switch turns the first light off but also switches the second light on and so on down the tunnel, so always the light behind goes off and the one in front comes on, and it will work no matter which direction you enter the tunnel. Show your circuit diagram (in five steps).

### HOMEWORK

To control one bulb from two places

## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

### Lab Session 12

#### **OBJECTIVE:**

Introduction to Breadboard

#### **Breadboard**

A breadboard is used to make up temporary circuits for testing or to try out an idea. No soldering is required so it is easy to change connections and replace components. Parts will not be damaged so they will be available to re-use afterwards. Thus, a breadboard is used to check whether the circuit works as intended.

The photograph shows a typical small breadboard which is suitable for beginners building simple circuits with one or two ICs (chips).



Small Breadboard  
Photograph © [Rapid Electronics](#)

#### **Connections on Breadboard**

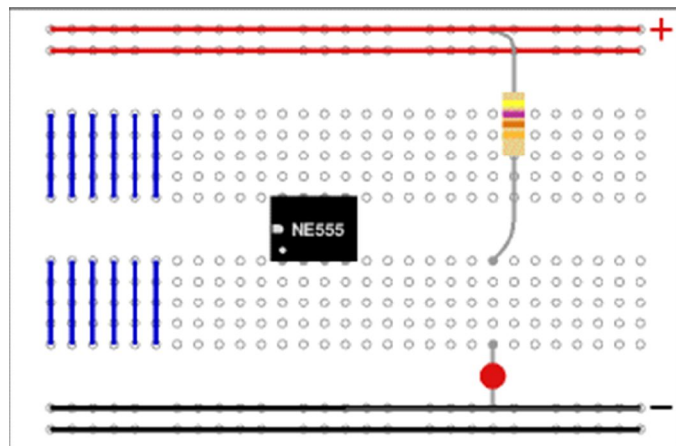
Breadboards have many tiny sockets (called 'holes') arranged on a 0.1" grid. The leads of most components can be pushed straight into the holes. ICs are inserted across the central gap with their notch or dot to the left.

Wire links can be made with single-core plastic-coated wire of 0.6mm diameter (the standard size). Stranded wire is not suitable because it will crumple when pushed into a hole and it may damage the board if strands break off.

The diagram shows how the breadboard holes are connected:

The top and bottom rows are linked horizontally all the way across as shown by the red and black lines on the diagram. The power supply is connected to these rows, + at the top and 0V (zero volts) at the bottom.

Use upper row of the bottom pair for 0V, then you can use the lower row for the negative supply with circuits requiring a dual supply (e.g. +9V, 0V, -9V).



The other holes are linked vertically in blocks of 5 with no link across the centre as shown by the blue lines on the diagram. Notices how there are separate blocks of connections to each pin of ICs.

## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

On larger breadboards there may be a break halfway along the top and bottom power supply rows. It is a good idea to link across the gap before you start to build a circuit, otherwise you may forget and part of your circuit will have no power!

### Building a Circuit on Breadboard

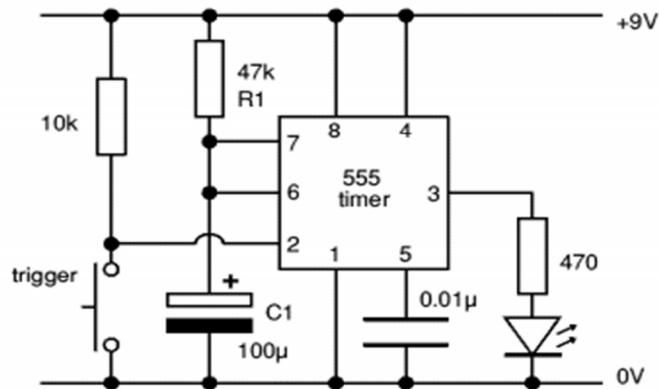
Converting a circuit diagram to a breadboard layout is not straightforward because the arrangement of components on breadboard will look quite different from the circuit diagram.

When putting parts on breadboard you must concentrate on their connections, not their positions on the circuit diagram. The IC (chip) is a good starting point so place it in the centre of the breadboard and work round it pin by pin, putting in all the connections and components for each pin in turn.

The best way to explain this is by example, so the process of building this 555 timer circuit on breadboard is listed step-by-step below.

The circuit is a monostable which means it will turn on the LED for about 5 seconds when the 'trigger' button is pressed. The time period is determined by R1 and C1 and you may wish to try changing their values. R1 should be in the range 1kΩ to 1MΩ

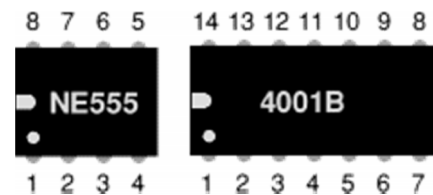
Time Period,  $T = 1.1 \times R1 \times C1$



Monostable Circuit Diagram

### IC pin numbers

IC pins are numbered anti-clockwise around the IC starting near the notch or dot. The diagram shows the numbering for 8-pin and 14-pin ICs, but the principle is the same for all sizes.



### Building the example circuit

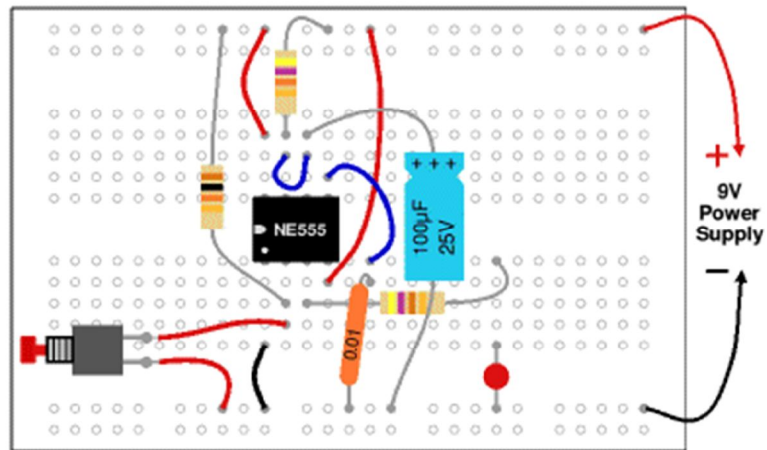
Begin by carefully insert the 555 IC in the centre of the breadboard with its notch or dot to the left.

Then deal with each pin of the 555:

## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

1. Connect a wire (black) to 0V.
2. Connect the 10k resistor to +9V.  
Connect a push switch to 0V (you will need to solder leads onto the switch)
3. Connect the 470 resistor to an unused block of 5 holes, then...  
Connect an LED (any



Monostable Circuit on Breadboard

- colour) from that block to 0V (short lead to 0V).
4. Connect a wire (red) to +9V.
5. Connect the 0.01µF capacitor to 0V.  
You will probably find that its leads are too short to connect directly, so put in a wire link to an unused block of holes and connect to that.
6. Connect the 100µF capacitor to 0V (+ lead to pin 6).  
Connect a wire (blue) to pin 7.
7. Connect 47k resistor to +9V.  
Check: there should be a wire already connected to pin 6.
8. Connect a wire (red) to +9V.

Finally,

- Check all the connections carefully.
- Check that parts are the correct way round (LED and 100µF capacitor).
- **Check that no leads are touching** (unless they connect to the same block).
- Connect the breadboard to a 9V supply and press the push switch to test the circuit.

If your circuit does not work disconnect (or switch off) the power supply and very carefully re-check every connection against the circuit diagram.

## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

### Lab Session 13(a)

**Objective:** To learn how to solder

#### **What is solder?**

Solder is an alloy (mixture) of tin and lead, typically 60% tin and 40% lead. It melts at a temperature of about 200°C. Coating a surface with solder is called 'tinning' because of the tin content of solder. Lead is poisonous and one should always wash his/her hands after using solder.



Solder for electronics use contains tiny cores of flux, like the wires inside a mains flex. The flux is corrosive, like an acid, and it cleans the metal surfaces as the solder melts. This is why solder must be melted actually on the joint, not on the iron tip. Without flux most joints would fail because metals quickly oxidize and the solder itself will not flow properly onto a dirty, oxidized, metal surface.

Reels of solder  
Photograph © [Rapid Electronics](#)

The best size of solder for electronics is 22swg (swg = standard wire gauge).

#### **How to Solder**

##### **Few safety precautions:**

- ⚡ **Never touch the element or tip of the soldering iron.**  
They are very hot (about 400°C) and will give a nasty burn.
- ⚡ **Take great care to avoid touching the mains flex with the tip of the iron**  
The iron should have a heatproof flex for extra protection. An ordinary plastic flex will melt immediately if touched by a hot iron and there is a serious risk of burns and electric shock.
- ⚡ **Always return the soldering iron to its stand when not in use**  
Never put it down on the workbench, even for a moment!
- ⚡ **Work in a well-ventilated area**  
The smoke formed when solder is melting mostly from the flux and quite irritating. Avoid breathing it by keeping head to the side of, not above, the work.
- ⚡ **Wash hands after using solder**  
Solder contains lead which is a poisonous metal.

##### **Preparing the soldering iron:**

- ⚡ **Place the soldering iron in its stand and plug in**

## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

The iron will take a few minutes to reach its operating temperature of about 400°C.

### ⚡ Dampen the sponge in the stand.

The best way to do this is to lift it out the stand and hold it under a cold tap for a moment, then squeeze to remove excess water. It should be damp, not dripping wet.

### ⚡ Wait a few minutes for the soldering iron to warm up.

Check if it is ready by trying to melt a little solder on the tip.

### ⚡ Wipe the tip of the iron on the damp sponge.

This will clean the tip.

### ⚡ Melt a little solder on the tip of the iron.

This is called 'tinning' and it will help the heat to flow from the iron's tip to the joint. It only needs to be done when the iron is plug in and occasionally while soldering if the tip is cleaned on the sponge.

## Start Soldering:

### ⚡ Hold the soldering iron like a pen, near the base of the handle.

Remember to never touch the hot element or tip.

### ⚡ Touch the soldering iron onto the joint to be made.

Make sure it touches both the component lead and the track. Hold the tip there for a few seconds and,

### ⚡ Feed a little solder onto the joint.

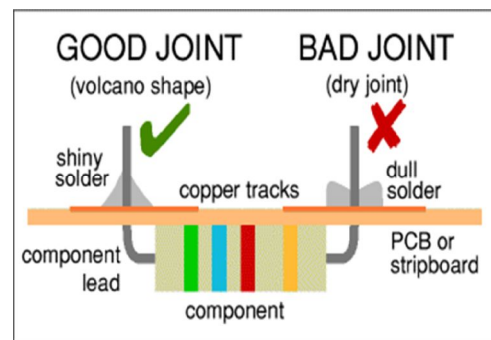
It should flow smoothly onto the lead and track to form a volcano shape as shown in the diagram. Apply the solder to the joint, not the iron.

### ⚡ Remove the solder, then the iron, while keeping the joint still.

Allow the joint a few seconds to cool before the circuit board will be moved.

### ⚡ Inspect the joint closely.

It should look shiny and have a 'volcano' shape. If not, reheat it and feed in a little more solder. This time ensure that **both** the lead and track are heated fully before applying solder.



## Using a heat sink

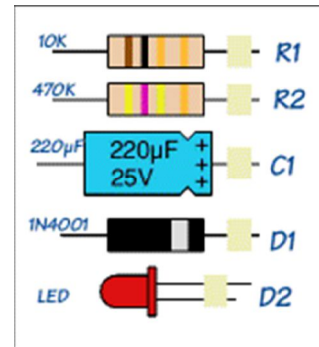
Some components, such as transistors, can be damaged by heat when soldering so it is wise to use a heat sink clipped to the lead between the joint and the component body. One can buy a special tool, but a standard crocodile clip works just as well and is cheaper.

## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

### Soldering Advice for Components

1. **Stick all the components onto a sheet of paper using sticky tape.**
2. **Identify each component** and write its name or value beside it.
3. **Add the code (R1, R2, C1 etc.) if necessary.** Many projects from books and magazines label the components with codes (R1, R2, C1, D1 etc.)
4. **Note down the values of Resistors and Capacitors by decoding schemes.**



Some components require special care when soldering.

Many must be placed the correct way round and a few are easily damaged by the heat from soldering. Appropriate warnings are given in the table below, together with other advice which may be useful when soldering.

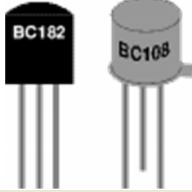



For most projects it is best to put the components onto the board in the order given below:

	Components	Pictures	Reminders and Warnings
1	<u>IC Holders</u> (DIL sockets)		<b>Connect the correct way round</b> by making sure the notch is at the correct end. Do NOT put the ICs (chips) in yet.
2	<u>Resistors</u>		No special precautions are needed with resistors.
3	<u>Small value capacitors</u> (usually less than 1µF)		These may be connected either way round. Take care with polystyrene capacitors because they are easily damaged by heat.
4	<u>Electrolytic capacitors</u> (1µF and greater)		<b>Connect the correct way round.</b> They will be marked with a + or - near one lead.
5	<u>Diodes</u>		<b>Connect the correct way round.</b> Take care with germanium diodes (e.g. OA91) because they are easily damaged by heat.
6	<u>LEDs</u>		<b>Connect the correct way round.</b> The diagram may be labelled <b>a</b> or <b>+</b> for anode and <b>k</b> or <b>-</b> for cathode; The cathode is the short lead and there may be a slight flat on the body of round LEDs.



## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

7	<b><u>Transistors</u></b>		<p><b>Connect the correct way round.</b>  Transistors have 3 'legs' (leads) so extra care is needed to ensure the connections are correct.  Easily damaged by heat.</p>
8	<b><u>Wire Links</u></b> between points on the circuit board.	 single core wire	<p>Use single core wire; this is one solid wire which is plastic-coated.  If there is no danger of touching other parts tinned copper wires can also be used, this has no plastic coating and looks just like solder but it is stiffer.</p>
9	<b><u>Battery clips</u></b> , buzzers and other parts with their own wires		<p><b>Connect the correct way round.</b></p>
10	<b><u>Wires</u></b> to parts off the circuit board, including <b><u>switches</u></b> , <b><u>relays</u></b> , <b><u>variable resistors</u></b> and <b><u>loudspeakers</u></b> .	 stranded wire	<p>use stranded wire which is flexible and plastic-coated.  Do not use single core wire because this will break when it is repeatedly flexed.</p>
11	<b><u>ICs (chips)</u></b>		<p><b>Connect the correct way round.</b>  <b>Many ICs are static sensitive.</b>  Leave ICs in their antistatic packaging until they are needed, and then earth hands by touching a metal water pipe or window frame before touching the ICs.  <b>Carefully insert ICs in their holders:</b>  make sure all the pins are lined up with the socket then push down firmly with thumb</p>

## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

### Lab Session 13(b)

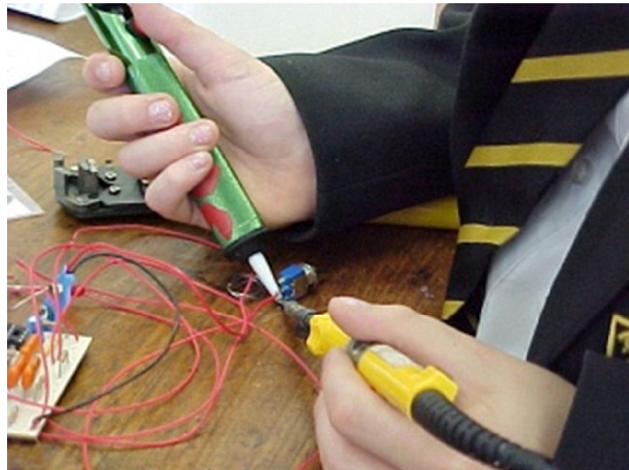
**Objective:** To learn how to de-solder

#### **De-soldering:**

In order to de-solder a joint to remove or re-position a wire or component, there are two ways to remove the solder:

##### **1. With a de-soldering pump (solder sucker)**

- ⚡ Set the pump by pushing the spring-loaded plunger down until it locks.
- ⚡ Apply both the pump nozzle and the tip of soldering iron to the joint.
- ⚡ Wait a second or two for the solder to melt.
- ⚡ Then press the button on the pump to release the plunger and suck the molten solder into the tool.
- ⚡ Repeat if necessary to remove as much solder as possible.
- ⚡ The pump will need emptying occasionally by unscrewing the nozzle.



Using a de-soldering pump (solder sucker)

##### **2. With solder remover wick (copper braid)**

- ⚡ Apply both the end of the wick and the tip of soldering iron to the joint.
- ⚡ As the solder melts most of it will flow onto the wick, away from the joint.
- ⚡ Remove the wick first, then the soldering iron.
- ⚡ Cut off and discard the end of the wick coated with solder.



Solder remover wick  
Photograph © [Rapid Electronics](#)

After removing most of the solder from the joint(s) one may be able to remove the wire or component lead straight away (allow a few seconds for it to cool). If the joint will not come apart easily apply soldering iron to melt the remaining traces of solder at the same time as pulling the joint apart, taking care to avoid burning .

## **Engineering Drawing & Workshop**

*NED University of Engineering and Technology- Department of Electronic Engineering*

### **Lab Session 14**

#### **OBJECTIVE:**

- **Introduction to OrCAD Capture**
- **Generation of schematic**
- **Generation of Netlist**

#### **OrCAD:**

This software is used for complete design and fabrication of PCB(Printed Circuit Board).

#### **OrCAD CAPTURE:**

It is a schematic design tool.

#### **SCHEMATIC:**

- It is a symbolic representation of electronic components which are connected with virtual wires
- Schematic provides input (Netlist) to the layout tool.

#### **NETLIST:**

A Netlist is a file, usually ASCII text, which defines connections between the components in your design.

#### **ASCII TEXT:**

The American Standard Code for Information Interchange (ASCII) represents text in computer & communication equipment.

#### **PSpice:**

It is OrCAD's simulation tool.

#### **OrCAD LAYOUT:**

This OrCAD tool is used for routing of tracks.

#### **CREATING PROJECT:**

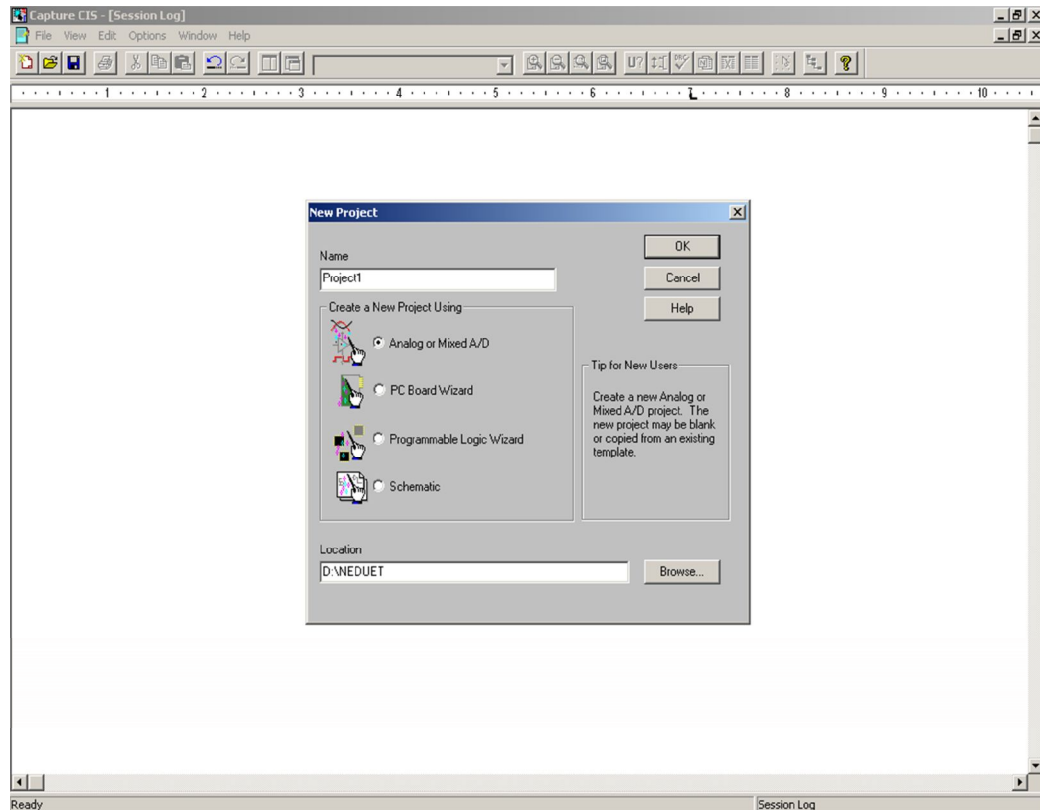
Creating Project Wizard provides framework for creating project.

##### **Follow these steps:**

1. Open OrCAD 16.0→Capture CIS
2. OrCAD\_Capture\_CIS\_Option\_with\_Capture→ok  
[CIS: Component Information System]
3. OrCAD Capture CIS – [Session Log]      \*// window will appear
4. From Menu bar select File→New→Project.
- 5.

# Engineering Drawing & Workshop

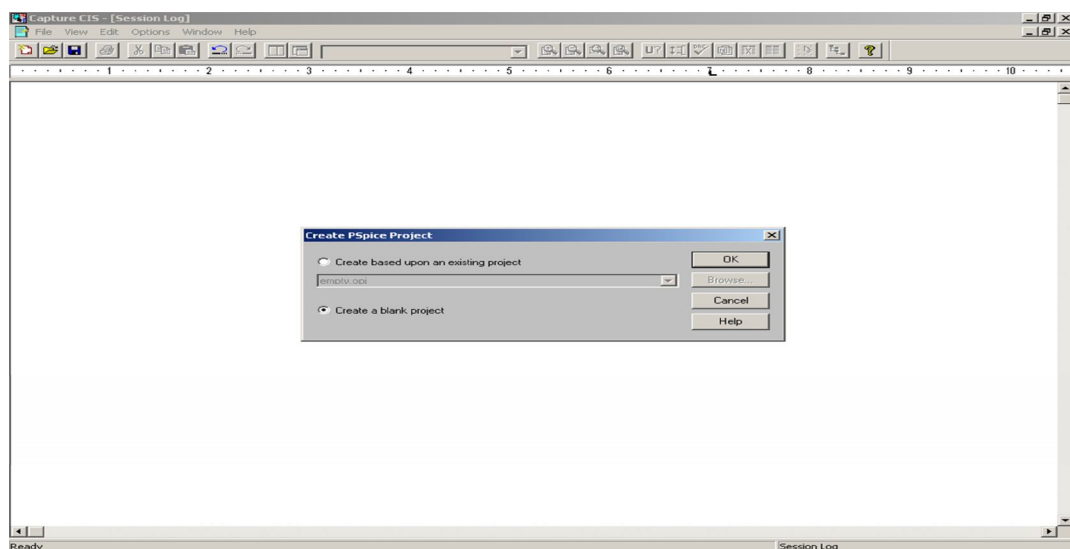
NED University of Engineering and Technology- Department of Electronic Engineering



- Give name to your project.
- Either select Analog or Mixed A/D or select Schematic.
- Select the location where you want to save your project  
[Note: *This folder must be unique.*]

*[\* If you have chosen Analog or Mixed A/D.]*

6.

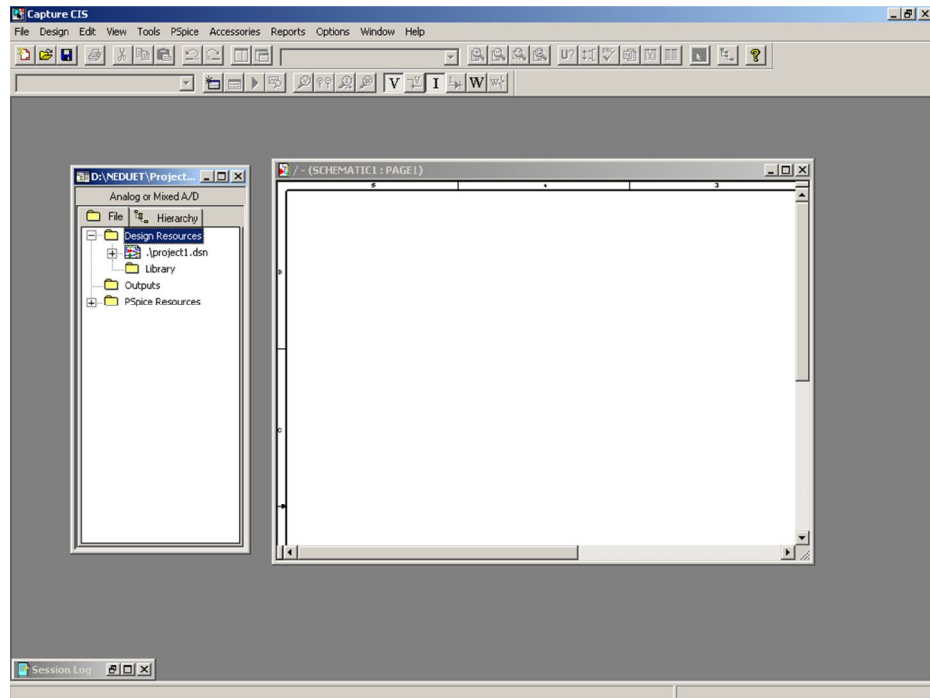


## Engineering Drawing & Workshop

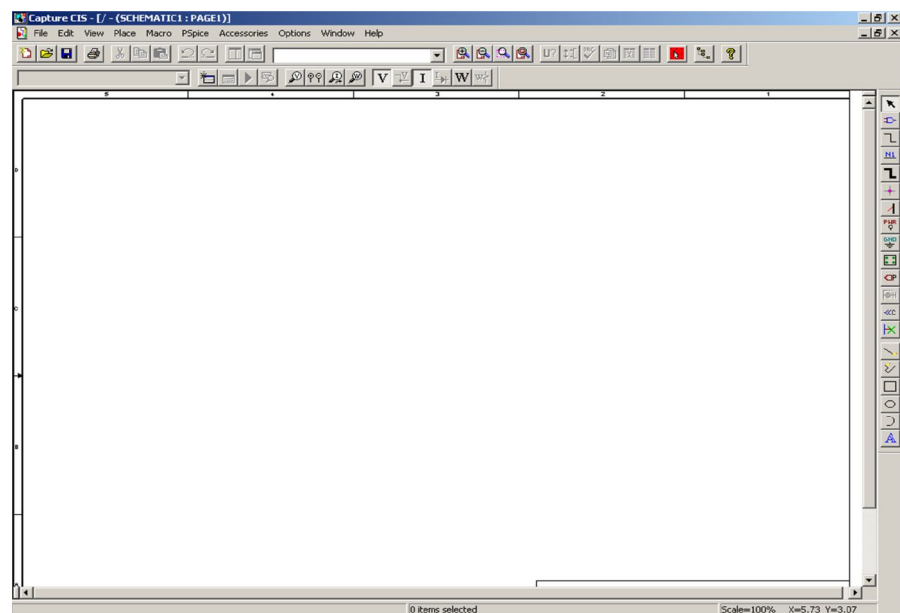
NED University of Engineering and Technology- Department of Electronic Engineering

7. Two frame-works will appear:

- A. **OrCAD Capture CIS-[/-(SCHEMATIC1:PAGE1)].**
- B. **OrCAD Capture CIS-[D:\NEDUET\Project1.opj].** \*// .opj is the project file



- A. **First we Select: OrCAD Capture CIS-[/-(SCHEMATIC1:PAGE1)] to work and making Schematic Circuit.**

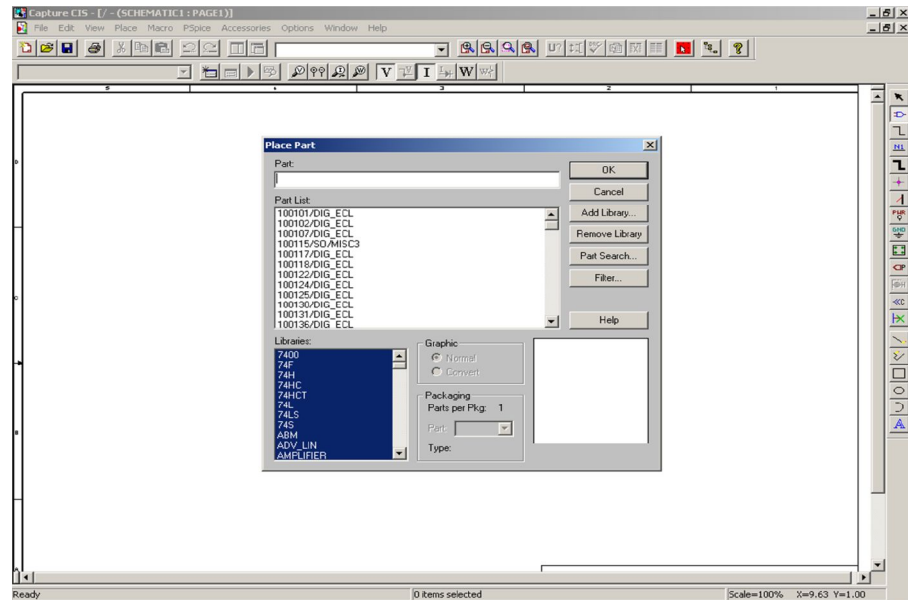


# Engineering Drawing & Workshop

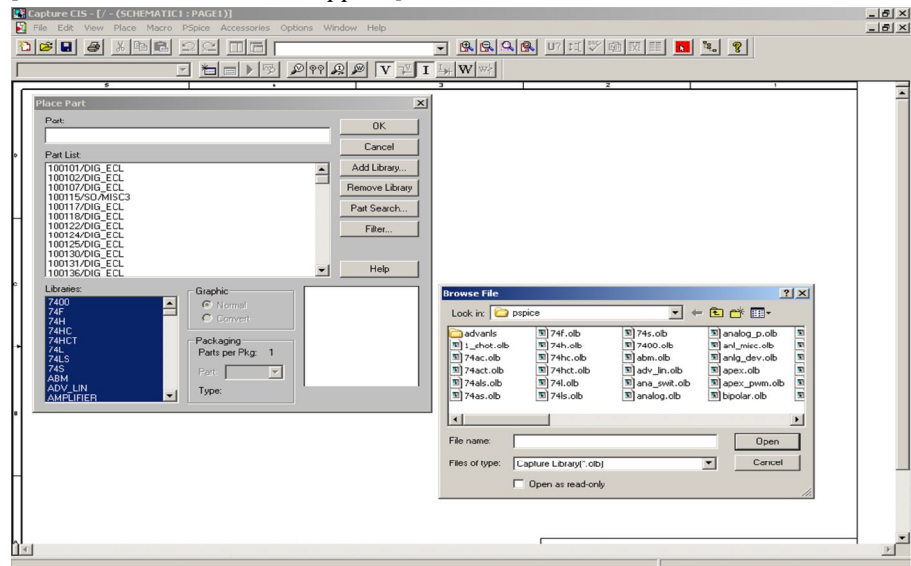
NED University of Engineering and Technology- Department of Electronic Engineering

## a) ADDING PARTS:

- Select Place Part tool from Menu bar.  
[Place Part window will appear.]



- If you are a beginner then click on Add Library.  
[Browse File window will appear]



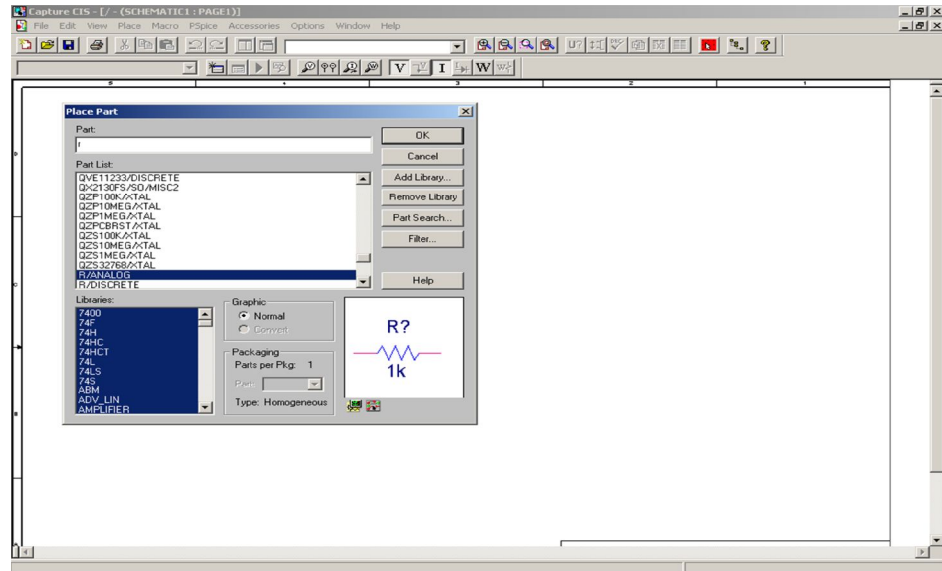
- Library file extension is .olb.

After adding library files .olb we can select parts (electronic) to make our schematic diagram.

## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

[Note: Select all the file of Libraries then we can easily search component by writing its name.]

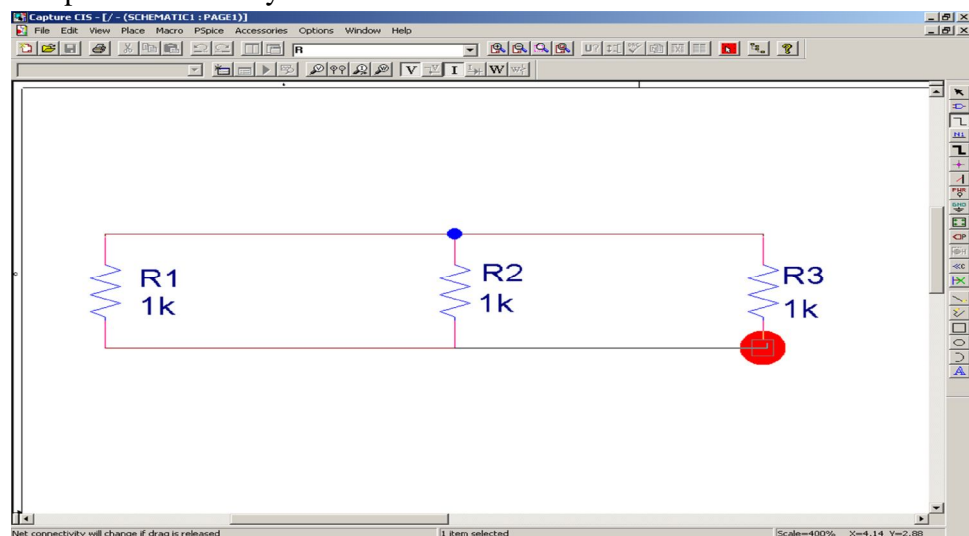


### b) CONNECTING PARTS:

- After adding required component on schematic page we join these components by virtual “wires”.
- Click Place Wire from place menu.

### c) ZOOM IN/OUT:

Use “I” and “O” keys to zoom in and out it will help you to connect the components correctly.



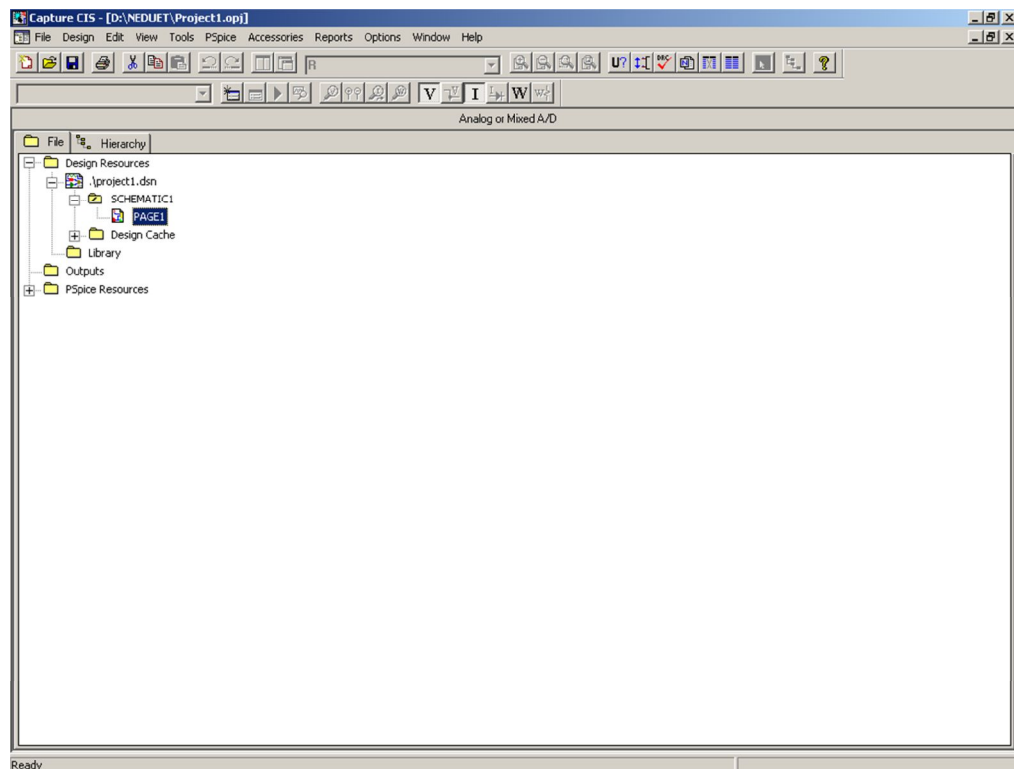
## Engineering Drawing & Workshop

*NED University of Engineering and Technology- Department of Electronic Engineering*

After finally making Schematic diagram on schematic page select File→Save→Close.

- We are now out from the Schematic page.
- Enter into the project page. \*// .opj is project extension
- .dsn is the schematic file format.

### **B. Now we have OrCAD Capture CIS-[D:\NEDUET\Project1.opj] framework window.**



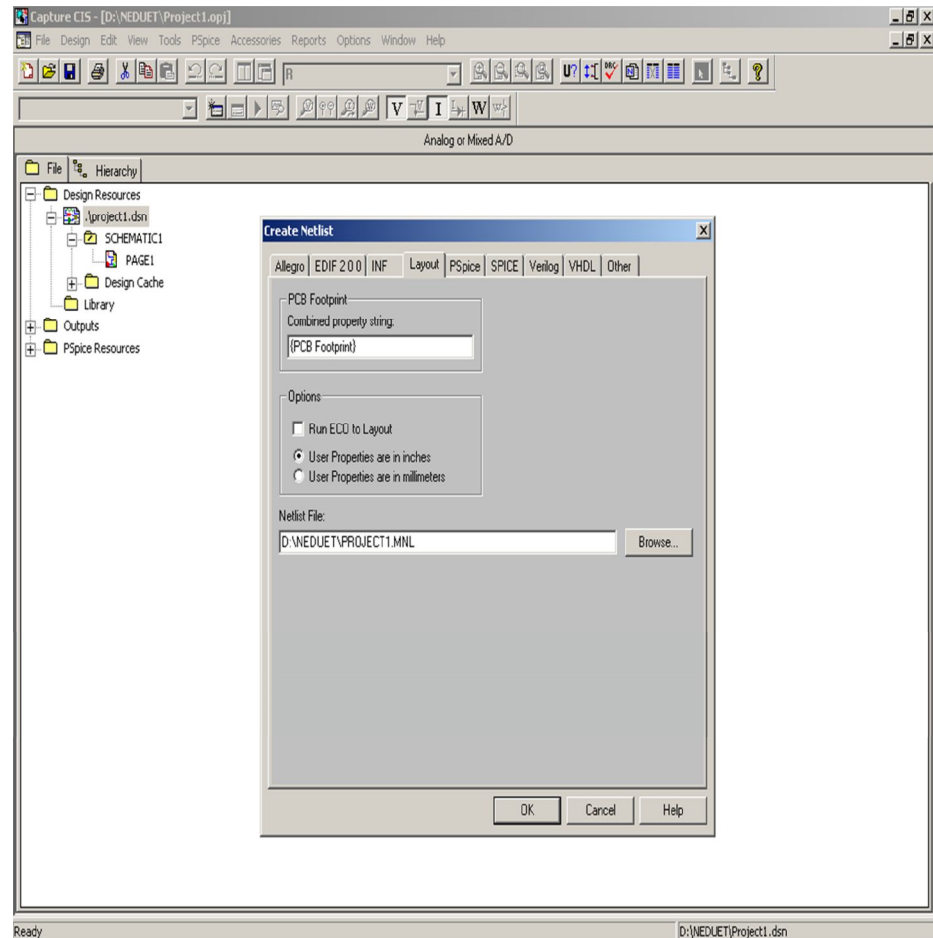
### **CREATING NETLIST:**

Select .dsn file by mouse cursor and then go to Tools→CreateNetlist.  
[Create Netlist Window will appear]



## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering



- Select Layout
- Select User properties are in inches.
- Put .MNL in a unique folder. \*// It will automatically generate  
[D:\NEDUET\Project1.MNL]
- Click OK.
- Go to File→Save→Close Project.
- The Netlist file .MNL is used as input for Layout.
- This .MNL file will be used to make .MAX file.

[Note: If we make any changes in our schematic then we always make update using Netlist option.]

# Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

## Lab Session 15

### OBJECTIVE:

- Introduction to Layout Tool.
- Creation of Layout Board.

### OrCAD LAYOUT:

OrCAD Layout tool is used for PCB routing and floor planning.

### LAYOUT:

Layout Tool accepts Netlist as input (created in Capture CIS) and generates output layout file that suitable for PCB fabrication.

### CREATING LAYOUT BOARD FILE:

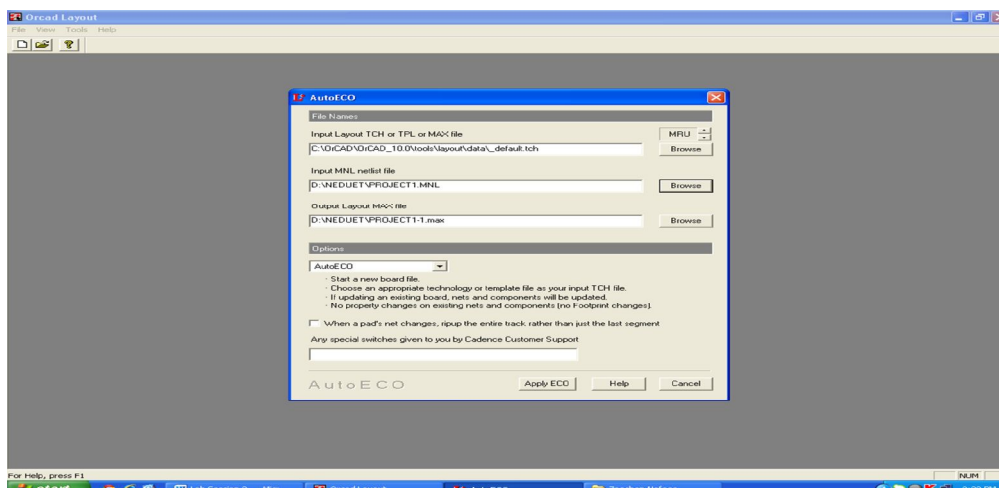
When we create a new board file in OrCAD Layout, we merge the electrical information from the Layout Netlist (.MNL) and physical information from template file (.TPL) or a technology file (.TCH) to create a new board design (.MAX). To be able to create a board file for a new design in Layout, we need to provide template file & netlist.

- A template (.TPL) file describes the characteristics of a physical board. A template includes information such as board outline, layer definition, grid settings, spacing & track width.
- A layout Netlist file (.MNL) describes the parts & interconnection in a schematic.
- A technology file (.TCH) can be considered as a subset of the template, but don't provide physical characteristic information of board.

1. Open OrCADLayout→File→New.

*[Auto ECO framework window will appear]*

2.



- In the input layout TCH or TPL or MAX file text box, specify the name & the location of the technology file to be used for your board.

## Engineering Drawing & Workshop

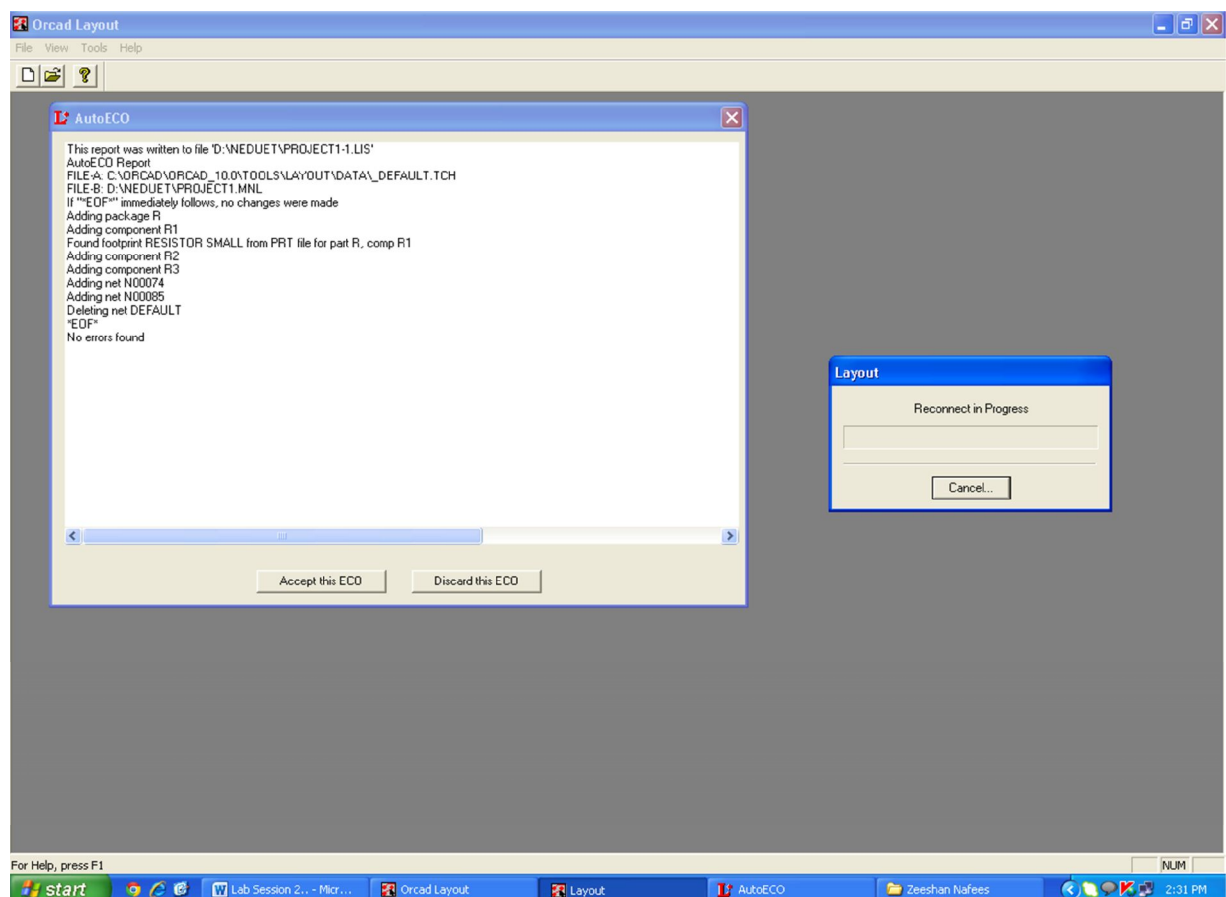
NED University of Engineering and Technology- Department of Electronic Engineering

- In the input MNL netlist file text box, specify the location of Project1.MNL created in layout netlist section.
- The output layout MAX file text box will automatically generate the name and location of file, Project1.MAX.

[Note: The Layout board file (.MAX) contains complete electrical & physical information of board.]

3. Apply AutoECO.
4. To create the Layout board file with the settings specified by you, click Apply ECO.
  - The Layout progress box appears indicating that the board file is being created.
  - The process of creating a board file will be completed only if the footprint information is available for all the components in the design.

*[At this stage beginner students may face some difficulty consult your INSTRUCTOR.]*

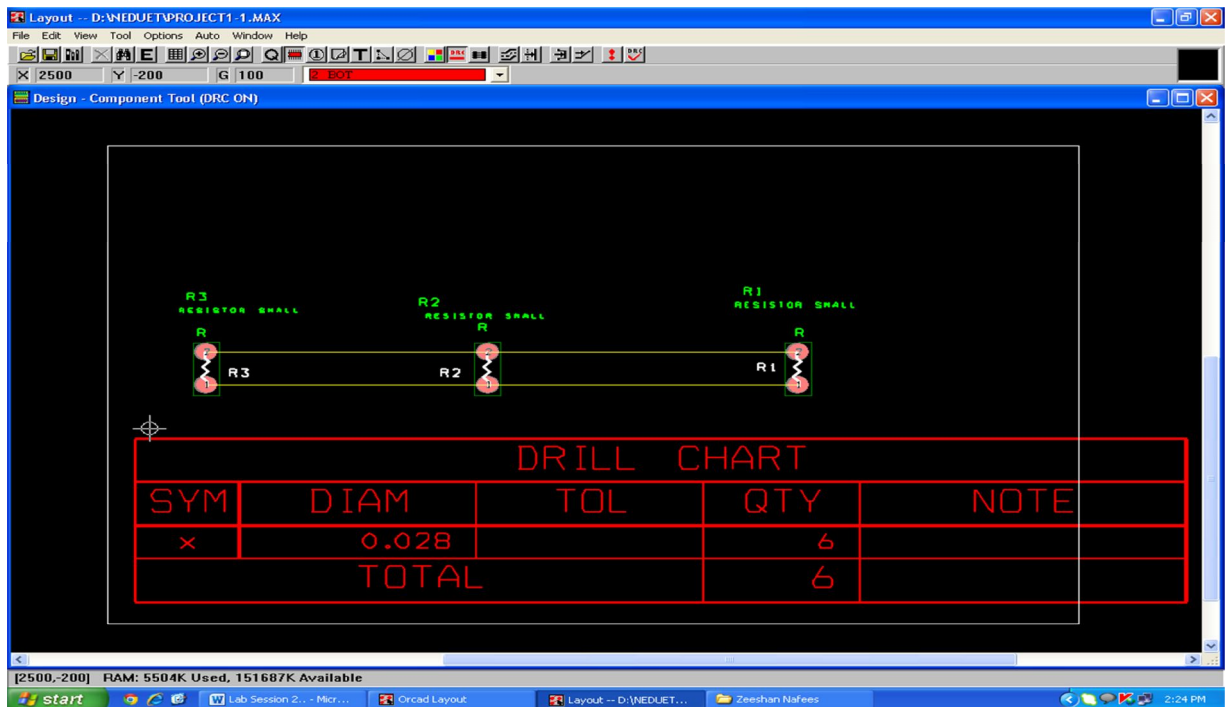


- Once AutoECO completed, AutoEco dialog box appears with the report, click Accept this ECO button.
- Click OK.

## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

- Now the framework window of .MAX file will be appeared.



## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

### Lab Session 16

#### **PRACTICAL ON PLOTTING, FIXING AND DEVELOPING A FILM FOR EXPOSING OF PCB'S**

#### **OBJECTIVES**

Upon the completion of this experiment we will be able to

1. Plot the film using photo plotter unit for exposing of PCB.
2. The fixing and development of the film.

#### **BACKGROUND**

For plotting of film a photo plotter unit is used. It exposes the film using a diode laser. The unit has an external power supply and is connected to PC via parallel port.

For photo plotting a dark room must be arranged, with a special green safe light illumination for standard development and fixing process. Therefore a unit must be installed for development and fixation of the film, consisting of three tanks one each for development, fixation and rinse.

The extent of delivery of photo plotter comprises of three software programs. The 1<sup>st</sup>, Gerb2Bitmap, serves for data preparation. This software exports the data in a proprietary FPF format for use on the plotter driver software. Further use of this software is i.e. o mount several layouts on one sheet of film, creation of step and repeat artwork. The second Run Plotter takes the so prepared data to control the plotter via the parallel port of your PC. It reads bitmap data in the above mentioned FPF format or at user's choice in windows BMP or EAGLE TIFF format. In Run Plotter, you may select the output resolution as well as positive or negative, direct or mirrored output.

The third application supplied is View Mate, a Gerber viewer and aperture converter tool. A built-in aperture converter helps transform the aperture tables of all CAD and CAM software into a standard format suitable for use with Gerb2Bitmap.

#### **OBSERVATION**

**Plotting:**

Material	Size( sq. mm)	Plot Area (sq. mm)
----------	---------------	--------------------

**Fixing and development:**

Chemical	Concentration
----------	---------------

#### **SUMMARY**

Enter the dark room mount the film in line with white arrow on the drum of the photo plotter, with some tape. The emulsion side must be facing the drum and sheet edges must be parallel to drum axis. Close the lid and leave the dark room.

## Engineering Drawing & Workshop

*NED University of Engineering and Technology- Department of Electronic Engineering*

Start the “Run\_ Plotter” software, set image quality and size, and run the photo plotter software. The drum starts rotating and the plotting starts. The computer shows the estimated time countdown i.e. how much time is left in completion of plotting. After plotting is complete, enter the dark room, open the lid and remove the film from the drum and start the development of the film.

The next part is the development and fixing of the film. Place the film removed from the photo plotter in the tank containing fixer. The film is then rinsed for 10 sec in tank containing water and after rinsing the film is dried. The film is then placed in developer solution for 30 seconds. After it rinse again and expose it to sunlight for few seconds.

### TESTS AND MEASUREMENTS:

- Light Source (Laser Diode): \_\_\_\_\_;  $\lambda$  = \_\_\_\_\_
- Resolution X: \_\_\_\_\_.
- Resolution Y: \_\_\_\_\_.
- Light Source (Dark room): \_\_\_\_\_.
- Speed of Plotting (per second): \_\_\_\_\_.

### PRELIMINARY INSTRUCTIONS:

1. The film must not be exposed to normal light before its development and fixing, because the film consist of crystals of silver halides, which when exposed to light form an image, and as normal light falls on the screen all the crystals are exposed to light and the film gets black.
2. The film must not be twisted or bent and must be parallel to drum axis otherwise the image will be distorted.
3. The parallel port connecting PC to photo plotter unit must not be loose.
4. Film from different manufacturers show different light intensity. Therefore it is necessary to adapt that light intensity.
5. Keep the drum surface clean. Do not use spray cleaners for removing residues of tape.
6. During plotting process the film is exposed to the laser light. The light can injure the human eye. Never open the plotter lid during operation.
7. The emulsion side must be facing the drum.

## PROCEDURE

### a) Plotting:

Enter the dark room turn the green safe light ON and make sure there is no extra light from outside dropping in take a sheet pf film from its box and mount it in line with white

## **Engineering Drawing & Workshop**

*NED University of Engineering and Technology- Department of Electronic Engineering*

arrow on the drum of the plotter, with some tape, which itself is in line with white arrow on the left of the drum housing.

The emulsion side as mentioned earlier must be facing the drum, under green light, the emulsion side of the film normally looks grey, and the opposite side is darker. After fixing upper edges turn the drum by hand and sweep over the sheet so it goes tightly on the drum, and is parallel to it and that the upper and the lower sheet corners are facing each other.

Close the lid and leave the dark room. Start the “Run Plotter” software, set image quality and size, (the output X resolution that can be set is 1016, 1355, 2032 and 4064, the Y resolution is fixed on 3000dpi) and run the photo plotter software. The drum starts rotating and after reaching normal speed the computer starts plotting with red diode laser. The red LED blinks according to the ON/OFF condition of the laser head. The compute shows the estimated time countdown i.e. how much time is left in completion of plotting. After plotting is complete, enter the dark room, open the lid and remove the film from the drum and start the development of the film.

### **b) Film Development and fixing:**

The next part is the development and fixing of the film. Place the film removed from the photo plotter in the tank containing fixer.(with a concentration of 1 part of fixer chemical and 2 parts of water) for 30 sec. The film is then rinsed for 10 sec in tank containing water. This step is taking because the developer will be destroyed if fixer drops in it.

After rinsing the film dry it, place it on the smooth surface; wipe the 1<sup>st</sup> side with a smooth fresh tissue and similarly the 2<sup>nd</sup> side. The film is then placed in developer solution (with concentration 1 part of developer and 3 part of water) for 30 sec. After it, rinse again and expose it to sunlight for few seconds.

## Engineering Drawing & Workshop

*NED University of Engineering and Technology- Department of Electronic Engineering*

### Lab Session 17

**Objective:** Introduction to MATLAB and MATLAB GUI

#### **Introduction:**

The name MATLAB stands for MATrixLABoratory. MATLAB was written originally to provide easy access to matrix software developed by the LINPACK (linear system package) and EISPACK (Eigen system package) projects.







MATLAB is a high-performance language for technical computing. It integrates Computation, visualization, and programming environment. Furthermore, MATLAB is a modern programming language environment: it has sophisticated data structures, contains built-in editing and debugging tools, and supports object-oriented programming. These factors make MATLAB an excellent tool for teaching and research.

MATLAB has many advantages compared to conventional computer languages (e.g., C, FORTRAN) for solving technical problems. MATLAB is an interactive system whose basic data element is an array that does not require dimensioning. The software package has been commercially available since 1984 and is now considered as a standard tool at most universities and industries worldwide.

It has powerful built-in routines that enable a very wide variety of computations. It also has easy to use graphics commands that make the visualization of results immediately available. Specific applications are collected in packages referred to as toolbox. There are toolboxes for signal processing, symbolic computation, control theory, simulation, optimization, and several other fields of applied science and engineering.

#### **Starting MATLAB**

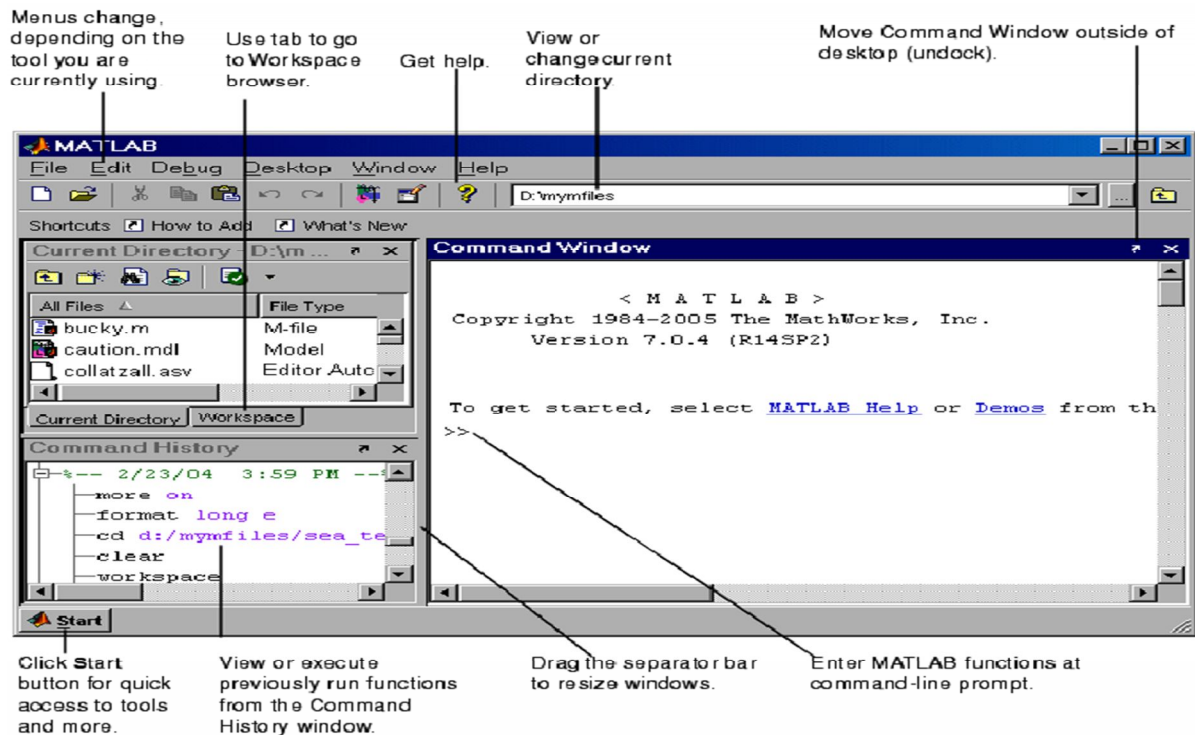
After logging into account, double-click on the MATLAB shortcut icon (MATLAB) on Windows desktop. A special window called the MATLAB desktop appears. The desktop is a window that contains other windows. The major tools within or accessible from the desktop are:

-  The Command Window
-  The Command History
-  The Workspace
-  The Current Directory
-  The Help Browser
-  The Start button



# Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering



## clc

Clear Command Window

**GUI Alternatives:** As an alternative to the `clc` function, select Edit > Clear Command Window in the MATLAB desktop.

**Syntax:** `clc`

**Description:** clears all input and output from the Command Window display, giving a "clean screen." After using `clc`, the scroll bar to see the history of functions cannot be used, but use of the up arrow to recall statements from the command history is still possible.

## clear

Remove items from workspace, freeing up system memory

**GUI Alternatives:** As an alternative to the `clear` function, use Edit > Clear Workspace in the MATLAB desktop.

**Syntax:**

- ▶ `clearall`
- ▶ `clear variable`

**Description:** `clear` removes all variables from the workspace. This frees up system memory.

## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

### Lab Session 18

**Objective:** Introduction to basic operations

#### **Basic Arithmetic Operators:**

Symbol	Operation	Example
+	Addition	2+3
-	Subtraction	2-3
*	Multiplication	2*3
/	Division	2/3
^	Power	2^3

#### **Creating MATLAB variables**

MATLAB variables are created with an assignment statement. The syntax of variable assignment is

Variable name = a value (or an expression)

For example,

>> x = expression

Where, expression is a combination of numerical values, mathematical operators, variables, and function calls. On other words, expression can involve:

- ✚ manual entry
- ✚ built-in functions
- ✚ user-defined functions

#### **Precedence**

Precedence is the order of performing arithmetic operations

Taerarchy of arithmetic operations	
PRECEDENCE	MATHEMATICAL OPERATIONS
First	The contents of all parentheses are evaluated first, starting from the innermost parentheses and working outward.
Second	All exponentials are evaluated, working from left to right
Third	All multiplications and divisions are evaluated, working from left to right
Fourth	All additions and subtractions are evaluated, starting from left to right

## Engineering Drawing & Workshop

*NED University of Engineering and Technology- Department of Electronic Engineering*

### **Exercise**

Q1. Calculate:

$$\frac{1}{2+3^2} + \frac{4}{5} \times \frac{6}{7}$$

- In MATLAB
  - Manually
- a. Compare the outputs.
  - b. If there is a difference in the two outputs, explain why it is so and how it can be solved?

Q2. Implement following in MATLAB

- a. Fahrenheit to Celsius formula
- b. Lens formula
- c. 3-Equations of motion

## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

### Lab Session 19

**Objective:** Introduction to Mathematical functions, Variables and Matrix operations

#### Mathematical functions

MATLAB offers many predefined mathematical functions for technical computing which contains a large set of mathematical functions. Typing `help elfun` and `help specfun` calls up full lists of elementary and special functions respectively. There is a long list of mathematical functions that are built into MATLAB. These functions are called built-ins. Many standard mathematical functions, such as  $\sin(x)$ ,  $\cos(x)$ ,  $\tan(x)$ ,  $e^x$ ,  $\ln(x)$ , are evaluated by the functions `sin`, `cos`, `tan`, `exp`, and `log` respectively in MATLAB.

Following are some commonly used functions, where variables  $x$  and  $y$  can be numbers, vectors, or matrices.

Table 2.1: Elementary functions

<code>cos(x)</code>	Cosine	<code>abs(x)</code>	Absolute value
<code>sin(x)</code>	Sine	<code>sign(x)</code>	Signum function
<code>tan(x)</code>	Tangent	<code>max(x)</code>	Maximum value
<code>acos(x)</code>	Arc cosine	<code>min(x)</code>	Minimum value
<code>asin(x)</code>	Arc sine	<code>ceil(x)</code>	Round towards $+\infty$
<code>atan(x)</code>	Arc tangent	<code>floor(x)</code>	Round towards $-\infty$
<code>exp(x)</code>	Exponential	<code>round(x)</code>	Round to nearest integer
<code>sqrt(x)</code>	Square root	<code>rem(x)</code>	Remainder after division
<code>log(x)</code>	Natural logarithm	<code>angle(x)</code>	Phase angle
<code>log10(x)</code>	Common logarithm	<code>conj(x)</code>	Complex conjugate

#### Types of Variables:

A MATLAB variable is essentially a tag that assign to a value while that value remains in memory. The tag gives a way to reference the value in memory so that programs can read it, operate on it with other data, and save it back to memory.

MATLAB provides three basic types of variables:

- 🔺 Local Variables
- 🔺 Global Variables

#### Local Variables

Each MATLAB function has its own local variables. These are separate from those of other functions (except for nested functions), and from those of the base workspace. Variables defined in a function do not remain in memory from one function call to the next, unless they are defined as global.

Scripts, on the other hand, do not have a separate workspace. They store their variables in a workspace that is shared with the caller of the script. When called from the command line, they share the base workspace. When called from a function, they share that function's workspace.

## Engineering Drawing & Workshop

*NED University of Engineering and Technology- Department of Electronic Engineering*

### Global Variables

If several functions, and possibly the base workspace, all declare a particular name as global, then they all share a single copy of that variable. Any assignment to that variable, in any function, is available to all the other functions declaring it global.

### Matrix Generation:

Matrices are the basic elements of the MATLAB environment. A matrix is a two-dimensional array consisting of m rows and n columns. Special cases are column vectors ( $n = 1$ ) and row vectors ( $m = 1$ ).

### Entering a vector

The elements of vectors in MATLAB are enclosed by square brackets and are separated by spaces or by commas. For example, to enter a row vector, v, type

```
>> v = [1 4 7 10 13]
v =
1 4 7 10 13
```

Column vectors are created in a similar way; however, semicolon (;) must separate the components of a column vector,

```
>> w = [1;4;7;10;13]
w =
1
4
7
10
13
```

### Entering a matrix

A matrix is an array of numbers. To type a matrix into MATLAB

- ▶ begin with a square bracket, [
- ▶ separate elements in a row with spaces or commas (,)
- ▶ use a semicolon (;) to separate rows
- ▶ End the matrix with another square bracket, ].

Here is a typical example. To enter a matrix A, such as,

```
A =
1 2 3
4 5 6
7 8 9
Type,
>> A = [1 2 3; 4 5 6; 7 8 9]
```

## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

### Matrix indexing

If  $i$  and  $j$  are two indices, then element of row  $i$  and column  $j$  of the matrix  $A$  is denoted by  $A(i,j)$ . Thus,  $A(i,j)$  in MATLAB refers to the element  $A_{ij}$  of matrix  $A$ . The first index is the row number and the second index is the column number.

For example,  $A(1,3)$  is an element of first row and third column.

Here,  $A(1,3)=3$ . Correcting any entry is easy through indexing. Here we substitute  $A(3,3)=9$  by  $A(3,3)=0$ . The result is

```
>>A(3,3) = 0
```

```
A =
```

```
1 2 3
```

```
4 5 6
```

```
7 8 0
```

### Matrix Operations:

#### Colon Operator:

To enter a vector  $k$  from say  $x$  to  $z$  having step size of  $y$  then colon operator is used instead of typing whole matrix manually.

```
>>k= x : y : z
```

The colon operator can also be used to pick out a certain row or column. For example, the statement  $A(m:n,k:l)$  specifies rows  $m$  to  $n$  and column  $k$  to  $l$ . Subscript expressions refer to portions of a matrix.

Only  $':'$  shows whole row or column to be extracted i.e.  $A(m:n,:)$  specifies rows from  $m$  to  $n$  with all columns.

#### Deleting row or column

To delete a row or column of a matrix, use the empty vector operator,  $[]$

```
>>A(3,:) = []
```

```
A =
```

```
1 2 3
```

```
4 5 6
```

Third row of matrix  $A$  is now deleted. To restore the third row, we use a technique for creating a matrix

```
>> A = [A(1,:);A(2,:);[7 8 0]]
```

```
A =
```

```
1 2 3
```

```
4 5 6
```

```
7 8 0
```

Matrix  $A$  is now restored to its original form.

#### Dimension (size)

To determine the dimensions of a matrix or vector, use the command `size`. For example,

```
>>size(A)
```

## Engineering Drawing & Workshop

*NED University of Engineering and Technology- Department of Electronic Engineering*

```
ans =  
    3 3  
means 3 rows and 3 columns. Or more explicitly with,  
>> [m,n]=size(A)
```

### **Transposing a matrix**

The transpose operation is denoted by an apostrophe or a single quote ('). It flips a matrix about its main diagonal and it turns a row vector into a column vector. Thus,

```
>> A'  
ans =  
    1 4 7  
    2 5 8  
    3 6 0
```

### **Exercise:**

1. Construct a matrix containing tables of numbers 2 to 10.
2. Solve the following system of Equations using MATLAB
  - a)  $x + 2y + 3z = 1$  ;  $3x + 3y + 4z = 1$  ;  $2x + 3y + 3z = 2$
  - b)  $3x - 4y + 5z = 6$  ;  $-4x + 9y + 8z = 2$  ;  $0.6x + 1.9y - 2.2z = -10$

## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

### Lab Session 20

#### **Objective:** Array Operations

Introduction to Programming in MATALB

#### **Array arithmetic operations**

Array arithmetic operations or array operations for short, are done element-by-element. The period character ' .' distinguishes the array operations from the matrix operations. However, since the matrix and array operations are the same for addition (+) and subtraction (-), the character pairs (:+) and (:-) are not used. The list of array operators is shown below in Table 3.2. If A and B are two matrices of the same size with elements  $A = [a_{ij}]$  and  $B = [b_{ij}]$ , then the command

```
>> C = A.*B
```

Produces another matrix C of the same size with elements  $c_{ij} = a_{ij}b_{ij}$ . For example, using the same 3 x 3 matrices,

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}, \quad B = \begin{bmatrix} 10 & 20 & 30 \\ 40 & 50 & 60 \\ 70 & 80 & 90 \end{bmatrix}$$

As

```
>> C = A.*B
```

```
C =  
    10    40    90  
   160   250   360  
   490   640   810
```

.*	Element-by-element multiplication
./	Element-by-element division
.^	Element-by-element exponentiation

Table 3.1: Array operators

#### **Programming In MATLAB:**

##### **M-File Scripts**

A script file is an external file that contains a sequence of MATLAB statements. Script files have a filename extension .m and are often called M-files. M-files can be scripts that simply execute a series of MATLAB statements, or they can be functions that can accept arguments and can produce one or more outputs.

##### **M-File functions**

Functions are programs (or routines) that accept input arguments and return output arguments. Each M-file function (or function or M-file for short) has its own area of workspace, separated from the MATLAB base workspace.



## Engineering Drawing & Workshop

*NED University of Engineering and Technology- Department of Electronic Engineering*

### **Input and output arguments**

The input arguments are listed inside parentheses following the function name. The output arguments are listed inside the brackets on the left side. They are used to transfer the output from the function file. The general form looks like this

function [outputs] = function\_name(inputs)

### **Input And Output to a script file**

**input** Taking input from keyboard

**Syntax:**

Variable = input('string') ;

**Description** The response to the input prompt can be any MATLAB expression, which is evaluated using the variables in the current workspace

**disp** Display text or array

**Syntax**

disp(X)

**Description** disp(X) displays an array, without printing the array name. If X contains a text string, the string is displayed.

### **Exercise:**

1. Write a MATLAB program to find the average of two numbers input by the user.
2. Write a program to prepare HSC marksheet.
3. Write a function to calculate geometric mean of three numbers input by the user in separate program and display the output.
4. Implement three equations of motion as separate MATLAB functions and write a program to calculate Vf, S1 and S2 through these functions.

## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

### Lab Session 21

**Objective:** Introduction to Graph plotting in MATLAB

#### **Basic Plotting**

MATLAB has an excellent set of graphic tools. Plotting a given data set or the results of computation is possible with very few commands

#### **Creating simple plots**

The basic MATLAB graphing procedure, for example in 2D, is to take a vector of x-coordinates,

$x = (x_1, \dots, x_N)$ , and a vector of y-coordinates,  $y = (y_1, \dots, y_N)$ , locate the points  $(x_i; y_i)$ , with  $i = 1, 2, \dots, n$  and then join them by straight lines. Prepare x and y in an identical array form; namely, x and y are both row arrays and column arrays of the same length.

The MATLAB command to plot a graph is `plot(x,y)`.

```
>> x = [1 2 3 4 5 6];  
>> y = [3 -1 2 4 5 1];  
>> plot(x,y)
```

#### **Plotting a Sine Wave:**

To plot the function  $\sin(x)$  on the interval  $[0; 2\pi]$ , first create a vector of x values ranging from 0 to  $2\pi$ , then compute the sine of these values, and finally plot the result

```
>> x = 0:pi/100:2*pi;  
>> y = sin(x);  
>> plot(x,y)
```

#### **Adding titles, axis labels, and annotations**

MATLAB enables to add axis labels and titles. For example, using the graph from the previous example, add x- and y-axis labels. Now label the axes and add a title. The character `\pi` creates the symbol  $\pi$

```
>> xlabel('x = 0:2\pi')  
>> ylabel('Sine of x')  
>> title('Plot of the Sine function')
```

#### **Specifying line styles and colors**

It is possible to specify line styles, colors, and markers (e.g., circles, plus signs . .) using the plot command:

```
plot(x,y,'style_color_marker')
```

where `style_color_marker` is a triplet of values from the Table.

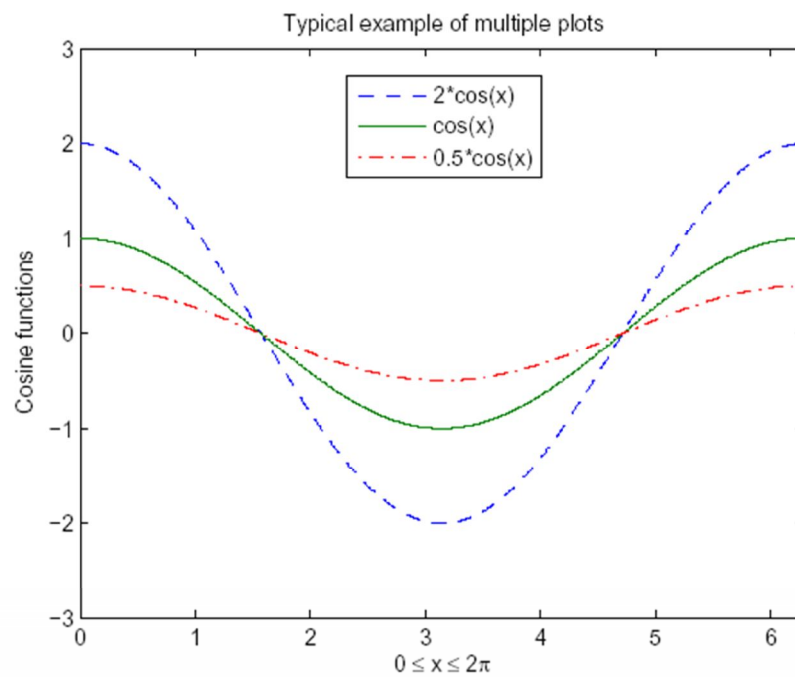
## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

SYMBOL	COLOR	SYMBOL	LINE STYLE	SYMBOL	MARKER
k	Black	—	Solid	+	Plus sign
r	Red	--	Dashed	o	Circle
b	Blue	:	Dotted	*	Asterisk
g	Green	—.	Dash-dot	.	Point
c	Cyan	none	No line	×	Cross
m	Magenta			s	Square
y	Yellow			d	Diamond

### Exercise:

Q. Write MATLAB code for following output.



## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

### Lab Session 22

**Objective:** Introduction to Control Flow and Loops in MATLAB

#### **Control flow and operators**

MATLAB is also a programming language. Like other computer programming languages, MATLAB has some decision making structures for control of command execution. These decision making or control flow structures include for loops, while loops, and if-else-end constructions. Control flow structures are often used in script M-files and function M-files. MATLAB provides several tools that can be used to control the flow of a program (script or function). In a simple program the commands are executed one after the other. But the flow control structure that make possible to skip commands or to execute specific group of commands.

#### **Relational and logical operators**

A relational operator compares two numbers by determining whether a comparison is true or false.

OPERATOR	DESCRIPTION
>	Greater than
<	Less than
>=	Greater than or equal to
<=	Less than or equal to
==	Equal to
~=	Not equal to
&	AND operator
	OR operator
~	NOT operator

#### **Loops:**

a) **For Loop**

Execute block of code specified number of times

**Syntax/General Format:**

```
for variable = initval:endval
statement
...
statement
end
```

**Description** for statement repeatedly executes one or more MATLAB statements in a loop. Loop counter variable x is initialized to value initval at the start of the first pass through the loop, and automatically increments by 1 each time through the loop. The program makes repeated passes through statements until either x has incremented to the value endval, or

## **Engineering Drawing & Workshop**

*NED University of Engineering and Technology- Department of Electronic Engineering*

MATLAB encounters a break, or return instruction, thus forcing an immediately exit of the loop

### **b) While Loop**

Repeatedly execute statements while condition is true

#### **Syntax**

while expression, statements, end

**Description** repeatedly executes one or more MATLAB statements in a loop, continuing until expression no longer holds true or until MATLAB encounters a break, or return instruction. thus forcing an immediately exit of the loop

## **Conditional Statements**

### **If else statement**

Execute statements if condition is true

#### **Syntax**

if expression, statements, end

**Description** evaluates expression and, if the evaluation yields logical 1 (true) or a nonzero result, executes one or more MATLAB commands denoted here as statements. expression is a MATLAB expression, usually consisting of variables or smaller expressions joined by relational operators or logical functions.

Simple expressions can be combined by logical operators (&&, ||, ~) .

## **Exercise:**

1. Write a program to generate counting from 1 to 20 and then 20 to 1.
2. Write a program which takes two numbers from the user and displays the relationship b/w them.
3. Write a program which takes three numbers input from user, and displays them in ascending order.
4. Write a program to take 10 numbers input from user using 'For Loop' and display the Sum of all numbers, even numbers and odd numbers separately.

## Engineering Drawing & Workshop

*NED University of Engineering and Technology- Department of Electronic Engineering*

### Lab Session 23

**Objective:** Introduction to LabView

#### **Introduction**

LabVIEW programs are called virtual instruments, or VIs, because they appear as physical instruments, such as oscilloscopes and multi-meters. LabVIEW VIs contain three components-the front panel, the block diagram, and the control and function palette.

You can use LabVIEW to communicate with hardware such as data acquisition, vision, and motion control devices

In this Lab, you will build simple VIs to incorporate basic programming structures in LabVIEW. This section will teach you fundamentals of LabVIEW front panel, block diagram and tool palette, and also explains data flow in the block diagram.

#### **Front Panel**

In LabVIEW, you build a user interface, or front panel, with controls and indicators.

**Controls** are knobs, push buttons, dials, and other input devices. **Indicators** are graphs, LEDs, and other displays.

#### **Block Diagram**

After you build the user interface, you add code using different functions to control the front panel objects. The block diagram contains this code. In some ways, the block diagram resembles a flowchart.

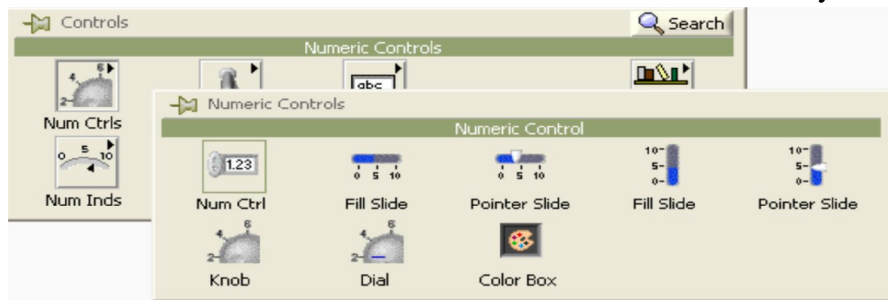
#### **Building a VI**

1. Start 'LabVIEW' program.
2. Open a 'blank VI'
3. You will have two screens available. One is the 'front panel' and the second is called 'block diagram'. Front Panel will always have the 'controls' and 'indicators' required by the program to show the input and output parameters. 'Block diagram' will have the detailed graphical representation of the actual program.
4. Go to the 'Front Panel'. The proposed program needs two numbers as inputs for 'addition' and 'multiplication'. Remember all inputs are 'controls' and all outputs are 'indicators'.
5. Right click your mouse keeping the cursor anywhere on the front panel. That opens the 'controls' palette. Then select 'Numerical controls' and a 'Numeric control', which brings a numeric control display onto the front panel. This can be used to input the required number as input 1.

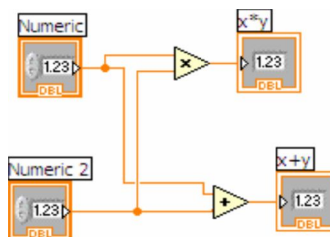
## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

6. These steps may be repeated to obtain a second control input, where you can input the second number. This will be your input2.



7. Now shift over to the 'Block diagram' by pressing CTRL+E characters on your keyboard.
8. You will see the icons representing the two numeric control displays provided on the front panel. They are called control terminals. Right click the mouse (keeping the cursor in the free space on the block diagram). That opens up the 'functions' palette.
9. Select Arith/compare palette. This opens up the Arithmetic/comparison palette. Select 'Numeric' palette. This opens up the 'express numeric' palette, which shows all the arithmetic functions. Select and drag the 'add' and 'multiply' functions on to the block diagram.
10. Right click on output side of the 'add' and 'multiply' functions and create indicator by selecting Create>>Indicator. This provides the display indicators for showing the results of the respective functions.
11. Get wiring tool and wire both the numeric controls to both the input terminals of the 'add' and 'multiply' functions. The block diagram will then show the following graphical program on the screen.



12. Go to the front panel. Your program is now ready for execution. Input the numbers you want to add and multiply in the respective numeric input boxes. Hit the run button on the tool bar and the output display boxes will show the results of addition and multiplication functions. Save the VI using the 'File' pull down menu as 'example1-add+multiply'.

## Engineering Drawing & Workshop

*NED University of Engineering and Technology- Department of Electronic Engineering*

**Exercise 1:** Build a VI to convert Rupees to Dollars. Take input from the user by using control.

**Exercise 2:** Solve the given equation  $x^2 - 9x + 20 = 0$  by using quadratic formula  $x =$   
\_\_\_\_\_

**Exercise 3:** Verify De Morgan's laws;

$$\overline{A + B} = \overline{A} \cdot \overline{B} \quad \overline{A \cdot B} = \overline{A} + \overline{B}$$



## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

### Lab Session 24



#### Objective

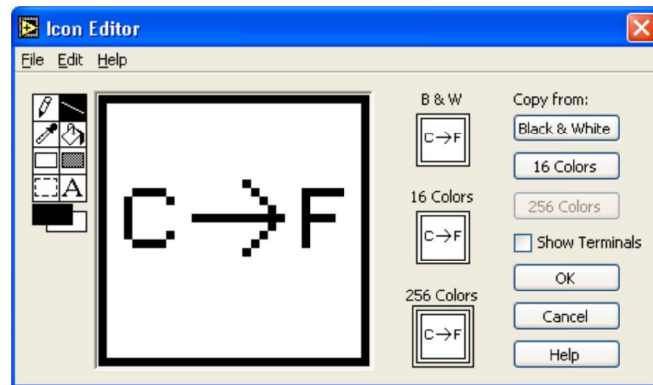
To learn how to create subVI and use it in another VI



#### Building subVI

Build a VI to convert temperature from degree centigrade to degree Fahrenheit.

#### Icon and Connector Pane


1. Right-click the icon in the upper right corner of the front panel window and select Edit Icon from the shortcut menu. The Icon Editor Dialog box appears.
2.  Double-click the Select tool, shown in pg 65, on the left side of the Icon Editor dialog box to select the default icon.
3. Press the <Delete> key to remove the default icon.
4.  Double-click the Rectangle tool, shown in pg 65, to redraw the border.
5. Create the icon in Figure.



- a)  Double-click the Text tool, shown in pg 66, and change the font to Small Fonts .
- b) Use the Text tool to click the editing area where you will begin typing.
- c) Type C and F. While the text is active, you can move the text by pressing the arrow keys.
- d)  Use the Pencil tool, shown in pg 66, to create the arrow.
- e) Note: To draw horizontal or vertical straight lines, press the <Shift> key while you use the Pencil tool to drag the cursor.
- f) Use the Select tool and the arrow keys to move the text and arrow you created.
- g) Select the B & W icon and click the 256 Colors button in the Copy from section to create a black and white icon, which LabVIEW uses for printing unless you have a color printer.
- h) Select the 16 Colors icon and click the 256 Colors button in the Copy from section.
- i) When you complete the icon, click the OK button to close the Icon Editor dialog box. The icon appears in the upper right corner of the front panel and block diagram.

## Engineering Drawing & Workshop

*NED University of Engineering and Technology- Department of Electronic Engineering*

6.  Right-click the icon on the front panel and select Show Connector from the shortcut menu to define the connector pane terminal pattern. LabVIEW selects a default connector pane pattern based on the number of controls and indicators on the front panel. For example, this front panel has two terminals, deg C and deg F, so LabVIEW selects a connector pane pattern with two terminals.
7. Assign the terminals to the numeric control and numeric indicator.
  - a) Select Help>>Show Context Help to display the Context Help window.
  - b) Click the left terminal in the connector pane. The tool automatically changes to the Wiring tool, and the terminal turns black.
  - c) Click the deg C control. A marquee highlights the control on the front panel.
  - d) Click an open space on the front panel. The marquee disappears, and the terminal changes to the data type color of the control to indicate that you connected the terminal.
  - e) Click the right terminal in the connector pane, and click the deg F indicator.
  - f) Click an open space on the front panel. Both terminals of the connector pane are orange.
  - g) Move the cursor over the connector pane. The Context Help window shows that both terminals are connected to double-precision, floating-point values.
8. Save and close the VI. You will use this VI later in the course.

## Using SubVIs

After you build a VI and create its icon and connector pane, you can use the VI as a subVI. To place a subVI on the block diagram, select Functions>>All Functions>>Select a VI. Navigate to the VI you want to use as a subVI and double-click to place it on the block diagram. You also can place an open VI on the block diagram of another open VI. Use the Positioning tool to click the icon in the upper right corner of the front panel or block diagram of the VI you want to use as a subVI and drag the icon to the block diagram of the other VI.

### Exercise 1

Use the VI built to calculate square root and convert it into subVI.

## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

### Lab Session 25

#### Objective

To learn how loops are designed and work in LabVIEW.

To write a VI for the Multiplication of a random number with 10 and displaying the result continuously, until it is stopped.

#### Introduction:


Structures are graphical representations of the loops. Loops use structures on the block diagram to repeat code and to execute code conditionally or in a specific order.


Like other nodes, structures have terminals that connect them to other block diagram nodes, execute automatically when input data are available, and supply data to output wires when execution completes.

- **For Loop** Executes a code a set number of times.
- **While Loop** Executes a code until a condition is met.
- **Case structure** executes depending on the input value passed to the structure.
- **Sequence structure** Contains one or more sequential code, which execute in sequential order.


In text base programming Do Loops repeat-until a condition is met. While loop iterates until the condition is true. The condition can be tested at the beginning or the end of the loop. In LabVIEW, the while Loop executes a graphical code while a condition is true or until a condition is met. A LabVIEW while loop always executes at least once.

Look at the Loop and identify the iteration index, conditional terminal and Boolean operator from the figure.

**Stop If True**  When a conditional terminal is in the Stop If True state, the While Loop executes its Block diagram until the conditional terminal receives a TRUE value from the front panel. The VI checks the conditional terminal at the end of each iteration.




**Continue If True** . When a conditional terminal is in the Continue If True state, the While Loop executes its sub-diagram until the Conditional Terminal receives false value.

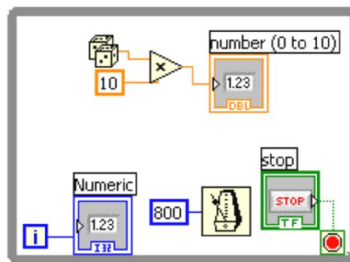
#### Building the VI:

1. Open LabVIEW and start a new VI. A blank VI opens the front panel and block diagram simultaneously. If you save one VI, second one will be saved automatically. Put a while loop from the function palette on the block diagram. This comes from Functions >> Structures >> While Loop
2. Right click on the conditional terminal and Select Create>>Control. This provides a True or False Control for the conditional terminal.
3. Get a random number function  by right clicking on blank space from All function » Functions » Numeric » Random number. Same way get multiply function from the numeric palette.

## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

4. Get wiring tool and wire the random number function to one input of 'multiply' function. [By default, you are in automatic tool selection mode, which provides you as soon as you bring the cursor near input/output terminal of any functional operator. In case, you can't get the wiring tool automatically, go to windows pull down menu and click on the 'show tools palette'. Then click on 'connect wire' tool on the palette.] Right click on 2nd input terminal of the multiply function and select Create>>Constant, and type 10 in the blank box.
5. Right click on the output side of the multiply function and create an indicator by selecting Create>>Indicator. Your block diagram will now show:  
This step will show the output value from the multiply function. Right click on the iteration  and create indicator to see number of iterations so far done. [Follow the same procedure you did for creating the indicator in the last step].
6. Double click on the stop button, this will show you location of the stop button in front panel. Identify the indicator of the number of iterations and Output number indicator and change the title of these indicators, if necessary.
7. Hit the run button on the tool bar and observe the running program. It will be too fast to see the numbers.
8. Stop the program by pressing the button and go to block diagram. Use CTRL-E whenever you want to move from front panel to block diagram and vice versa. Get a metronome  by right clicking Functions »All Function »Time & Dialog » Wait until next ms multiple. Right click on left hand side (at its input) of  and create a constant and type 800 in blue box.
9. Your block diagram should look like the diagram below



**Exercise 1:** Generate random numbers and multiply it with 10. If the answer is in between 4 and 6, then program should automatically abort. Also introduce a delay of 500ms after each iteration and display the number of iterations that it took till the condition is met. As soon as the condition is met, a dialogue box should appear indicating that your required instance has occurred.

## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

### Lab Session 26

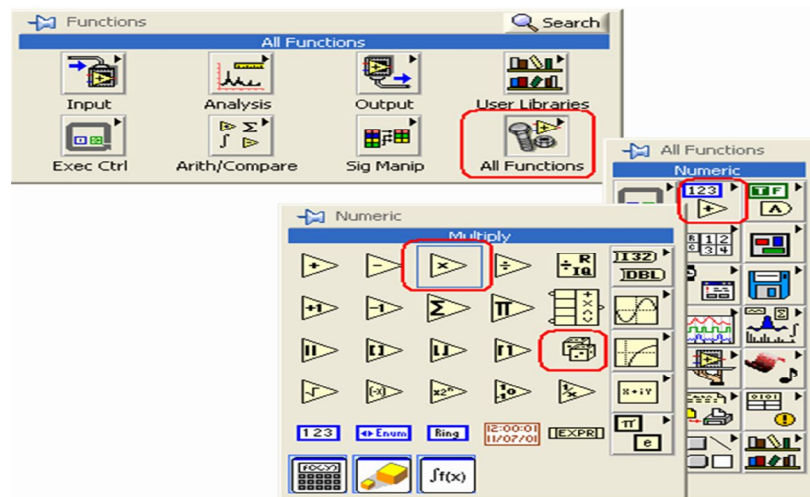
To learn how 'for' loops are designed and work in LabVIEW

#### Introduction:

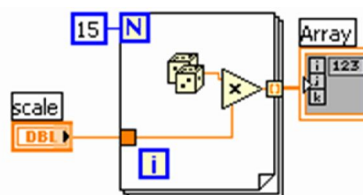
A For Loop executes a sub-diagram a set number of times specified by a constant or control at its count terminal (N).

#### Procedure to write 'generate Random values.vi'

1. Open LabVIEW and start a new VI. A blank VI opens the front panel and block diagram simultaneously. If you save one VI, second one will be saved automatically. Put a while loop from the function palette on the block diagram. This comes from Functions >> Structures >> For Loop
2. Get 'allfunctions' from the 'Functions' palette and connect all wire as given in the figure.



3. Connect all the wires properly and create the block diagram as shown.



4. Save this VI as 'GenerateRandomValues.VI'. Open the front panel. Run the program and see the results.

**Exercise 1:** Add 5 repeatedly in a number and display result in the form of an array after ten iterations.

## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

### Lab Session 27

#### Objective:

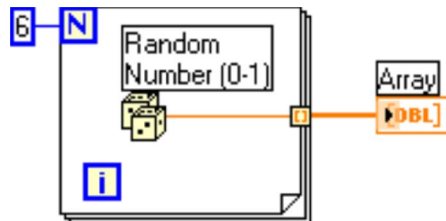
To learn different functionalities of arrays

#### Arrays

Arrays group data elements of the same type. An array consists of elements and dimensions. Elements are the data that make up the array. A dimension is the length, height, or depth of an array. You can build arrays of numeric, Boolean, path, string, waveform, and cluster data types. Consider using arrays when you work with a collection of similar data and when you perform repetitive computations. Arrays are ideal for storing data you collect from waveforms or data generated in loops, where each iteration of a loop produces one element of the array.

#### Auto-Indexing

If you wire an array to a For Loop or While Loop input tunnel, you can read and process every element in that array by enabling auto-indexing. When you auto-index an array output tunnel, the output array receives a new element from every iteration of the loop. The wire from the output tunnel to the array indicator becomes thicker as it changes to an array at the loop border, and the output tunnel contains square brackets representing an array, as shown in Figure



#### Array Functions

- **Array Size** - Returns the number of elements in each dimension of an array.
- **Initialize Array** - Creates an n-dimensional array in which every element is initialized to the value of element.
- **Array Subset** - Returns a portion of an array starting at index and containing length elements.
- **Build Array** - Concatenates multiple arrays or appends elements to an n-dimensional array.
- **Index Array** - Returns the element or sub-array of n-dimension array at index.

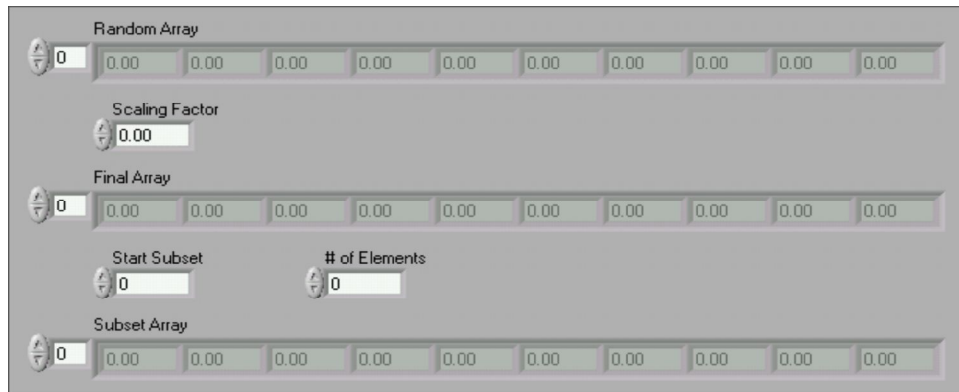
#### Building a VI

##### Front Panel

Build your front panel as shown in the following figure.

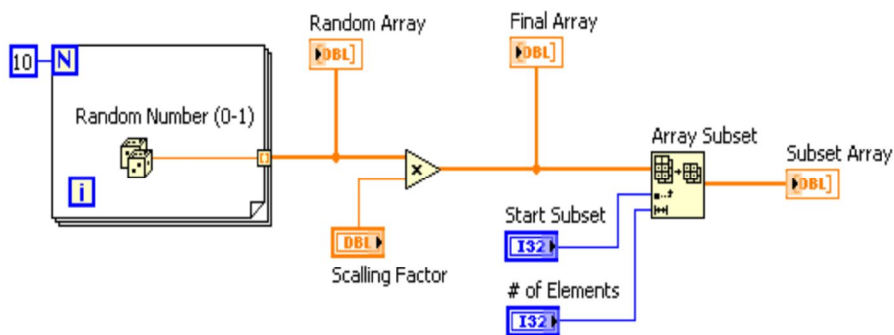
## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering



### Block Diagram

Build a block diagram as shown in figure.



**Exercise 1:** Create a two-dimensional array of the order 5x5..

**Exercise 2:** Verify that integral of constant is a ramp and integral of ramp is a square. Display the two results in separate arrays.

## Engineering Drawing & Workshop

*NED University of Engineering and Technology- Department of Electronic Engineering*

### Lab Session 28

#### **Objective:**

To display quantities on waveform charts and waveform graphs

#### **Introduction**

##### **Waveform Charts**

The waveform chart is a numeric indicator that displays one or more plots. The waveform chart is located on the Controls>>Graph Indicators palette. You can wire a scalar output directly to a waveform chart. The data type in the waveform chart terminal matches the input data type.

Waveform charts can display multiple plots. Bundle multiple plots together using the Bundle function located on the Cluster palette.

##### **Waveform graphs**

VI with graphs usually collects the data in an array and then plot the data to the graph. Waveform graph accept data in the form of array.

#### **Building a VI**

##### **Front Panel**

1. Put a waveform chart selecting from Control>>Graph indicators.

##### **Block Diagram**

1. Select a random number generator from numeric palette.
2. Multiply it with 10
3. Put this VI inside WHILE loop.
4. Join the output of the multiplier to the input of waveform chart terminal.
5. Click RUN

**Exercise 1:** Generate a sine wave and plot it on a wave form chart. Also double the amplitude of the generated sine wave and plot the two sine waves on the same waveform graph.



# Engineering Drawing & Workshop


NED University of Engineering and Technology- Department of Electronic Engineering

## Lab Session 29


### Objective:

To learn the use of structures in LabVIEW for decision-making

### Introduction:

**Select**  The Select function, located on the Functions>>Express>>Comparison palette, selects between two values dependent on a Boolean input.



**Case**  A Case structure has two or more subdiagrams, or cases. Only one subdiagram is visible at a time, and the structure executes only one case at a time. An input value determines which subdiagram executes. You must wire an integer, Boolean value, string, or enumerated type value to the selector terminal. You can specify a default case for the Case structure. You must specify a default case to handle out-of-range values or explicitly list every possible input value.

### Square Root VI

#### Front Panel

1. Open a blank VI and build the front panel shown in Figure 1.

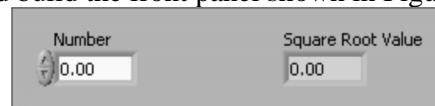


Figure 1

#### Block Diagram

1. Build the block diagram shown in Figure 2

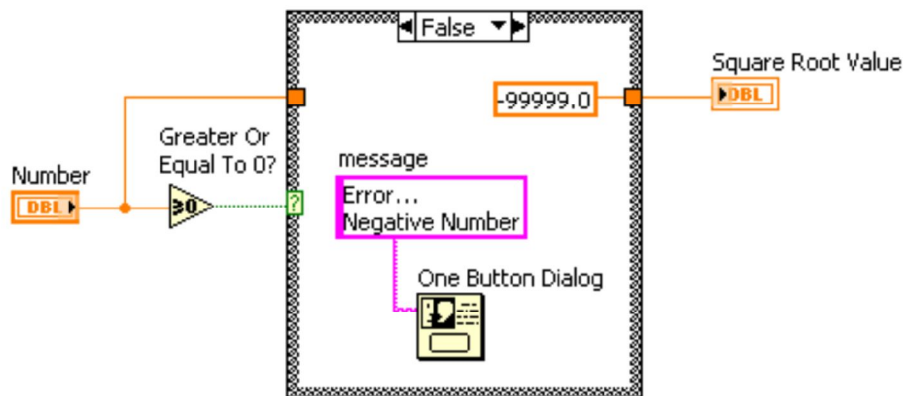




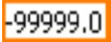
Figure 2



- a)  Place the Case structure, located on the Functions>>Execution Control palette, on the block diagram. Click the decrement or increment button to select the False case.
- b)  Place the Greater or Equal to 0? function, located on the Functions>>Arithmetic & Comparison>>Express Comparison palette, on the block diagram. This function returns True if Number is greater than or equal to 0.

## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

- c)  Right-click the numeric constant and select Properties from the shortcut menu. Select the Format and Precision tab. Set Digits of precision to 1, select Floating point notation, and click the OK button to ensure there is no data conversion between the constant and the numeric indicator outside the Casestructure.



- d) Place the One Button Dialog function, located on the Functions>>All Functions>>Time & Dialog palette, on the block diagram. This function displays a dialog box that contains the message “Error...Negative Number”.
- e) Right-click the message terminal of the One Button Dialog function, select Create>>Constant from the shortcut menu, type Error...Negative Number in the constant, and click the Enter button on the toolbar or click outside the control.
- f) Complete the diagram as shown in Figure 2.
2. Select the True case of the Case structure. Place the Square Root function, located on the Functions>>Arithmetic & Comparison>>Express Numeric palette, on the block diagram. This function returns the square root of Number. Wire the function as shown in Figure 3.

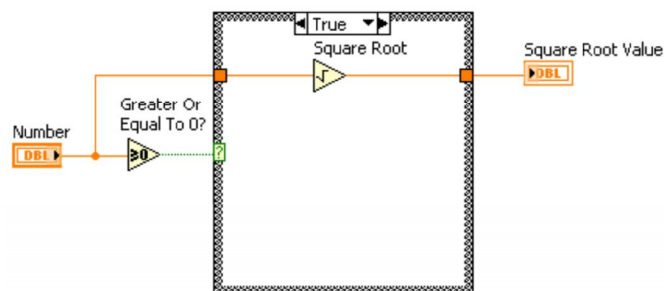


Figure 3

**Exercise 1:** Build a VI to convert temperature from °C to °F and °F to °C depending upon the user's choice and display the result using a single indicator for both conversions. Develop a mechanism to distinguish between the results by their respective units.

**Exercise 2:** Build a VI to perform addition, subtraction, multiplication and division on two numbers taken as input from the users. Only one of the functions can be performed at a time according to user's wish.

# Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

## Lab Session 30

### Objective:

To learn how to build clusters and their various functions.

### Clusters

Clusters group data elements of mixed types, such as a bundle of wires, as in a telephone cable, where each wire in the cable represents a different element of the cluster.

Bundling several data elements into clusters eliminates wire clutter on the block diagram and reduces the number of connector pane terminals that subVIs need.

### **Creating Cluster Controls and Indicators**

To create a cluster control or indicator, select a cluster on the Controls>>All Controls>>Array & Cluster palette, place it on the front panel, and drag controls or indicators into the cluster shell.



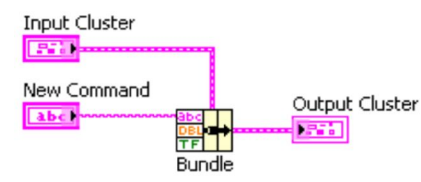
### **Cluster Order**

Cluster elements have a logical order unrelated to their position in the shell. The first object you place in the cluster is element 0; the second is element 1, and so on. If you delete an element, the order adjusts automatically.

### Cluster Functions

#### **Assembling Clusters**

Use the Bundle function to assemble a cluster from individual elements or to change the values of individual elements in an existing cluster without having to specify new values for all elements.

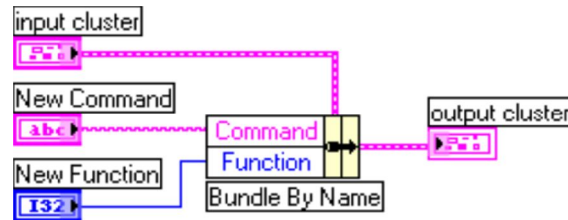


### **Replacing or Accessing Cluster Elements**

Use the Bundle by Name function to replace or access labeled elements of an existing cluster. Bundle by Name works similarly to the Bundle function, but instead of referencing cluster elements by their cluster order, it references them by their owned labels.

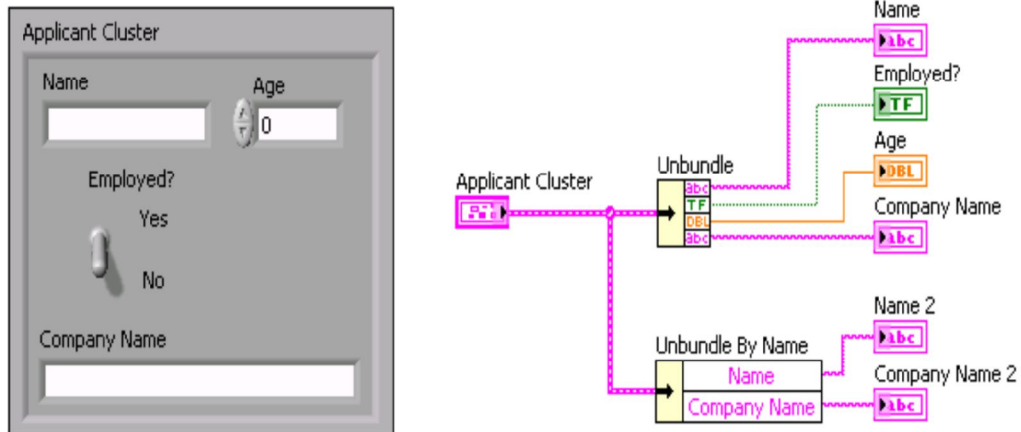
## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering



### Disassembling Clusters

Use the Unbundle function to split a cluster into its individual elements.



### Exercise 1

Make a cluster control containing one numeric control, two toggle buttons and one slider control. Increment numerical control by one and invert the logic of second toggle button and display the modified cluster. Also display the values of slider and first toggle button by in a separate cluster indicator.

## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

### Lab Session 31

#### Objective:

To learn the use of various functions to edit and manipulate strings

To learn the use of basic file I/O functions

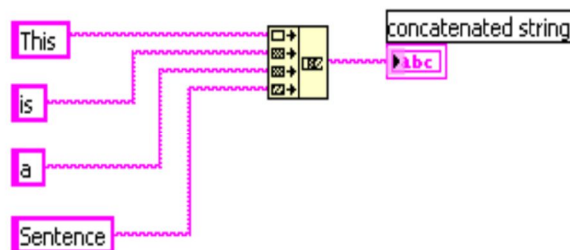
#### Introduction

##### Strings

A string is a sequence of displayable or non-displayable ASCII characters. Strings provide a platform independent format for information and data.

##### String Functions

- **String Length** - Returns in length the number of characters (bytes) string, including space characters.
- **Concatenate Strings** - Concatenates input strings and 1D array of strings into a single output string.



- **String Subset** - Returns the substring of the input string beginning at offset and containing length number of characters.
- **Match Pattern** - Searches for regular expression in string beginning at offset, and if it finds a match, splits string into three substrings.
- **Format into string-**



**Exercise 1:** Replace the string “brown” with “yellow” in the sentence “The quick brown fox jumps over the lazy dog”.

**Exercise 2:** Sort the characters in the string “The quick brown fox jumps over the lazy dog” alphabetically

#### File I/O VIs and Functions

File I/O operations pass data to and from files. Use the File I/O VIs and functions located on the Functions>>All Functions>>File I/O palette to handle all aspects of file I/O, including the following:

## Engineering Drawing & Workshop

*NED University of Engineering and Technology- Department of Electronic Engineering*

- Opening and closing data files
- Reading data from and writing data to files
- Reading from and writing to spreadsheet-formatted files
- Moving and renaming files and directories
- Changing file characteristics

### Basics of File I/O

A typical file I/O operation involves the following process:

1. Create or open a file. Indicate where an existing file resides or where you want to create a new file by specifying a path or responding to a dialog box to direct LabVIEW to the file location. After the file opens, a reference number represents the file. A reference number, or refnum, is a unique identifier for an object, such as a file, device, or network connection.
2. Read from or write to the file.
3. Close the file.

**Exercise 4:** Verify that integral of constant is a ramp and integral of ramp is a square as done in lab 5. Display the two results in separate arrays. Write results to a spread sheet and plot them on graphs.

## **Engineering Drawing & Workshop**

*NED University of Engineering and Technology- Department of Electronic Engineering*

### **Lab Session 32**

**OBJECTIVE:** To be familiar with AutoCAD

#### **Introduction**

AutoCAD is the first and most widely used design and drafting software. It has the capability to create complex and accurate drawings. Its position as the industry standard makes it an essential tool for anyone preparing for a career in engineering, design, or technology. CAD stands for Computer-Aided Design. An AutoCAD drawing is made up of objects. These can be either simple graphic objects (such as lines, arcs, circles, text, etc.) or blocks (which are groups of objects).

#### **AutoCAD systems offer**

- Increased accuracy
- Increased drawing speed
- Easy to revise
- Better design analysis
- Better presentation
- Libraries of drawing aids

#### **Starting AutoCAD**

From the **Start Menu** choose **Programs**. Then choose **AutoCAD 2011** from the menu  
OR

Double-click the icon available on the desktop.

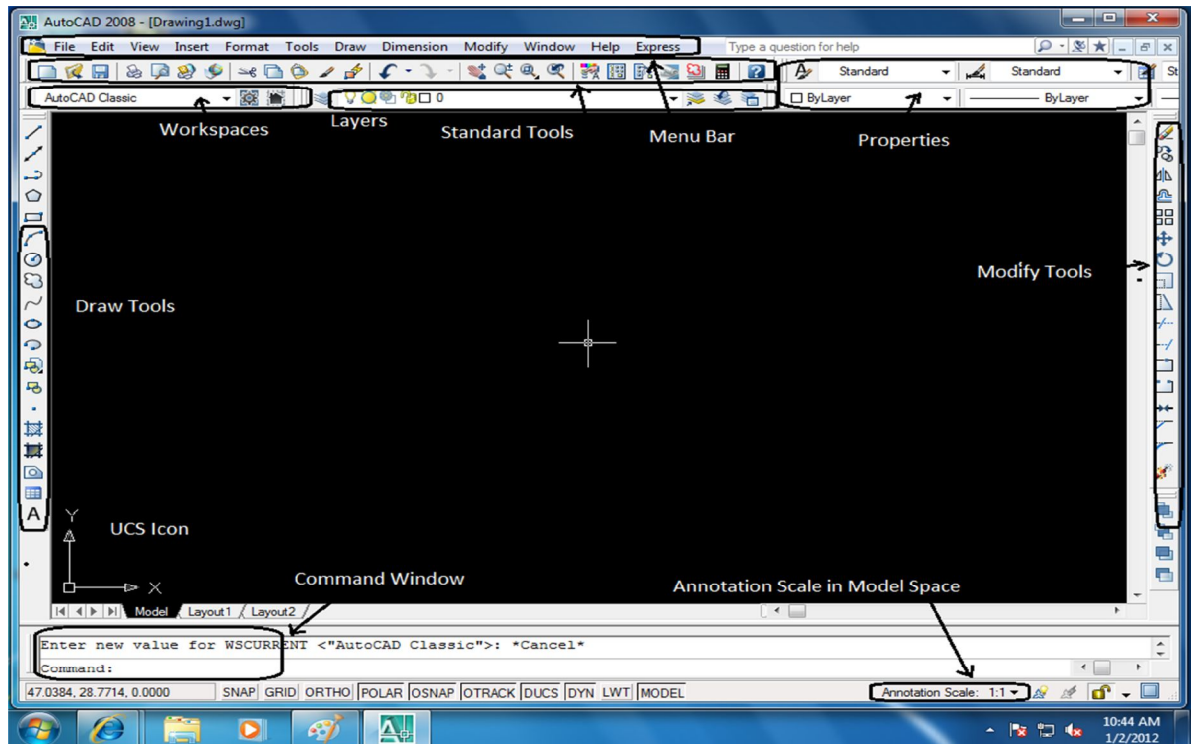
#### **The AutoCAD Interface**

##### **Drawing Area**

The black area is the Drawing Area. You can get an idea as to how large the drawing area is by moving the mouse. Notice the white crosshairs moving and look down in the lower left hand corner of the screen. A series of numbers are scrolling and changing. They are depicting the absolute position of the crosshairs in the drawing area. AutoCAD uses the Cartesian coordinate system. Notice the double arrowed icon at the lower left of the drawing area. This icon orients us to the coordinate system. "X" positioning is left to right, and "Y" positioning is bottom to top. The numerical readout at the lower left (called Cursor Tracking) is displaying decimal units with an accuracy of four (4) decimal places to the right in a format of X, Y. The lower left corner of the drawing area is the coordinate system's origin, 0, 0.

# Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering



**Fig: The AutoCAD Interface**

## Drawing Toolbar:

The Toolbar present at the Left side of Computer Screen. A Toolbar which contains basic drawing objects, like: line, polygon, polyline, circle, arc, ellipse etc.

## Modify Toolbar:

A Toolbar on Right side of Computer Screen. This Toolbar contains most important features, which are mostly required while drawing / drafting. The Toolbar contains: Erase, Move, Rotate, Cut etc.

## Axis:

Axis is the configuration / View of the Working Area, Normally it is in 2D (i.e. X-Axis & Y-Axis) but it can be rotated according to the view of object on the Screen. "3dorbit" command is used to rotate and to view the 3D image (i.e. X-Axis, Y-Axis & Z-Axis).

## Command Window:

Displays prompts and messages. In AutoCAD you can give the command in one of the three ways

- Choose an item from the menu or a shortcut menu
- Click a button on a toolbar
- Enter the command on the Command Line



## **Engineering Drawing & Workshop**

*NED University of Engineering and Technology- Department of Electronic Engineering*

### **Status Bar:**

Displays the cursor coordinates in the lower left corner.

### **User Coordinate System:**

In AutoCAD, there are two coordinate systems: A fixed system called the World Coordinate System (WCS) and a movable system called the User Coordinate System (UCS). In the WCS, the X axis is horizontal, the Y axis is vertical, and the Z axis is perpendicular to the XY plane. The origin is where the X and Y axes intersect (0, 0) in the lower-left corner of the drawing. You define a UCS in terms of the WCS.

Virtually all coordinate entry uses the current UCS.

### **EXERCISE:**

1. What are the advantages of AutoCAD?
2. List different types of 3-D shapes.
3. What type of coordinate systems is used in AutoCAD?

## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

### Lab Session 33

**OBJECTIVE:** To be familiar with drawing line, ray and polyline.

**THEORY:**

The Draw commands can be used to create new objects such as lines and circles. Most AutoCAD drawings are composed purely and simply from these basic components. A good understanding of the Draw commands is fundamental to the efficient use of AutoCAD.

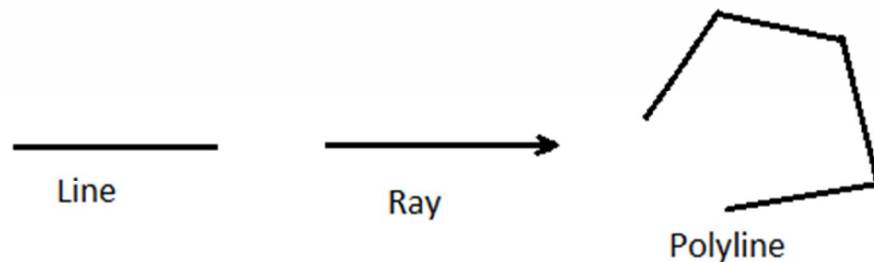


Fig: Line, Ray and Polyline

Drawing elements may be invoked by

- Using the pull-down menu
- Using the toolbars...
- Click on View... Toolbars

Lines are probably the most simple of AutoCAD objects. Using the Line command, a line can be drawn between any two points picked within the drawing area. Lines are usually the first objects you will want to draw when starting a new drawing because they can be used as "construction lines" upon which rest of your drawing will be based. Never forget that creating drawings with AutoCAD is not so dissimilar from creating drawings on a drawing board. Many of the basic drawing methods are the same.

Lines drawn between points are often called *vectors*. This terminology is used to describe the type of drawings that AutoCAD creates. AutoCAD drawings are generically referred to as "vector drawings".

**PROCEDURE:**

**The Line Command**

Toolbar Draw

Pull-down Draw Line

Keyboard LINE short-cut L

With the Line command you can draw a simple line from one point to another. When you *pick* the first point and move the *cross-hairs* to the location of the second point you will see a *rubber band line* which shows you where the line will be drawn when the second

## Engineering Drawing & Workshop

*NED University of Engineering and Technology- Department of Electronic Engineering*

point is picked. Line *objects* have two ends (the first point and the last point). You can continue picking points and AutoCAD will draw a straight line between each picked point and the previous point. Each line segment drawn is a separate *object* and can be moved or erased as required. To end this command, just hit the . key on the keyboard.

### Command Sequence

Command: LINE

Specify first point: (pick P1)

Specify next point or [Undo]: (pick P2)

Specify next point or [Undo]: . (to end)

You can also draw lines by entering the co-ordinates of their end points at the command prompt rather than picking their position from the screen.

### The Ray Command

Toolbar    custom

Pull-down    Draw.  
                 Ray

Keyboard    RAY

The Ray command creates a line similar to a construction line except that it extends infinitely in only one direction from the first pick point. The direction of the Ray is determined by the position of the second pick point.

### Command Sequence

Command: RAY

Specify start point: (pick the start point)

Specify through point: (pick a second point to determine direction)

Specify through point: . (to end or pick another point)

### The Polyline Family

Polylines differ from lines in that they are more complex objects. A single polyline can be composed of a number of straight-line or arc *segments*. Polylines can also be given line widths to make them appear solid. The illustration below shows a number of polylines to give you an idea of the flexibility of this type of line.

#### The Polyline Command

Toolbar    Draw

Pull-down    Draw.    Polyline

Keyboard    PLINE    short-cut PL

The Polyline or Pline command is similar to the line command except that the resulting object may be composed of a number of segments which form a single *object*. In addition

## Engineering Drawing & Workshop

*NED University of Engineering and Technology- Department of Electronic Engineering*

to the two ends a polyline is said to have *vertices* (singular *vertex*) where intermediate line segments join. In practice the Polyline command works in the same way as the Line command allowing you to pick as many points as you like. Again, just hit enter to end. As with the Line command, you also have the option to automatically *close* a polyline end to end. To do this, type C to use the close option instead of hitting enter. Follow the command sequence below to see how this works.

### Command Sequence

Command: PLINE

Specify start point: (pick P1)

Current line-width is 0.0000

Specify next point or [Arc/Halfwidth/Length/Undo/Width]: (pick P2)

Specify next point or [Arc/Close/Halfwidth/Length/Undo/Width]: (pick P3)

Specify next point or [Arc/Close/Halfwidth/Length/Undo/Width]: (pick P4)

Specify next point or [Arc/Close/Halfwidth/Length/Undo/Width]: (pick P5)

Specify next point or [Arc/Close/Halfwidth/Length/Undo/Width]: . (or C to close)

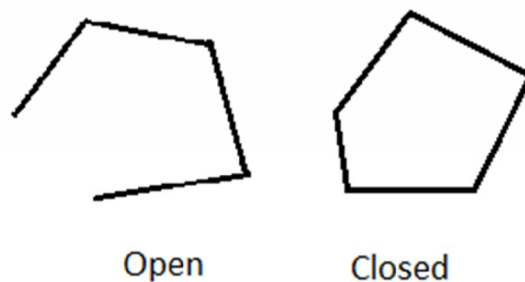


Fig: Open and Closed Polyline

In the illustration above, the figure on the left was created by hitting the Enter key after the fifth point was picked. The figure on the right demonstrates the effect of using the Close option.

Try experimenting with options such as Arc and Width and see if you can create polylines like the ones in the illustration above. The Undo option is particularly useful. This allows you to unpick polyline vertices, one at a time so that you can easily correct mistakes.

Polylines can be edited after they are created, for example, change their width. You can do this using the PEDIT command, Modify. Object. Polyline from the pull-down menu.

## Engineering Drawing & Workshop

*NED University of Engineering and Technology- Department of Electronic Engineering*

### **EXERCISES:**

1. In AutoCAD, a line can be drawn either by specifying distance from first point or using co-ordinates. Draw the lines by following the given command sequence.

Specifying distance from first point

Command: line

Specify first point: 10, 10

Specify next point or [Undo]: @10, 5

Specify next point or [Undo]: \*Cancel\*

Specifying co-ordinates

Command: line

Specify first point: 10, 10

Specify next point or [Undo]: 20, 15

Specify next point or [Undo]: \*Cancel\*

2. Draw a square of 10x10 by following the command sequence given below:

Command: line

Specify first point: 5, 5

Specify next point or [Undo]: 15, 5

Specify next point or [Undo]: 15, 15

Specify next point or [Close/Undo]: 5, 15

Specify next point or [Close/Undo]: 5, 5

Specify next point or [Close/Undo]: \*Cancel\*

3. Draw a rectangle by following the command sequence given below:

Command: rectangle

Specify first corner point or [Chamfer/Elevation/Fillet/Thickness/Width]: 5,5

Specify other corner point or [Dimensions]: d

Specify length for rectangles <0.0000>: 6

Specify width for rectangles <0.0000>: 7

Specify other corner point or [Dimensions]: 12,13

## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

### Lab Session 34

**OBJECTIVE:** To be familiar with drawing objects (Rectangle, Polygon, Circle).

#### **THEORY:**

The Draw commands can be used to create new objects such as lines and circles. Most AutoCAD drawings are composed purely and simply from these basic components. A good understanding of the Draw commands is fundamental to the efficient use of AutoCAD.

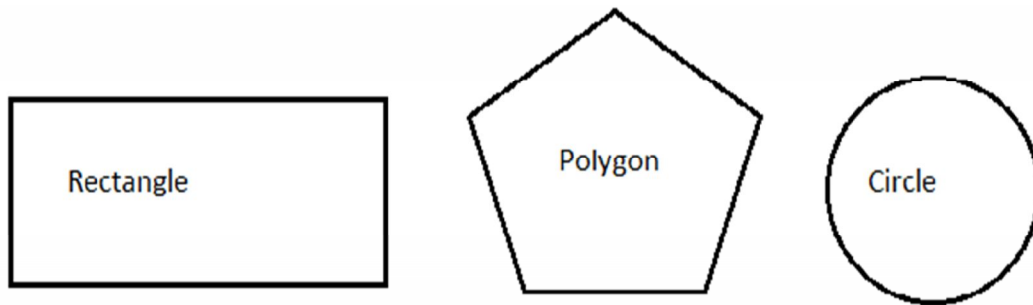


Fig: Rectangle, Polygon, Circle

Drawing elements may be invoked by

- Using the pull-down menu
- Using the toolbars...
- Click on View... Toolbars

#### **PROCEDURE:**

##### **The Rectangle Command**

Toolbar Draw

Pull-down Draw. Rectangle

Keyboard RECTANGLE short-cuts REC, RECTANG

The Rectangle command is used to draw a rectangle whose sides are vertical and horizontal. The position and size of the rectangle are defined by picking two diagonal corners. The rectangle isn't really an AutoCAD object at all. It is, in fact, just a closed polyline which is automatically drawn for you.

#### **Command Sequence**

Command: RECTANG

Specify first corner point or [Chamfer/Elevation/Fillet/Thickness/Width]: (pick P1)

Specify other corner point or [Dimensions]: (pick P2)

## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

The Rectangle command also has a number of options. Width works in the same way as for the Polyline command. The Chamfer and Fillet options have the same effect as the Chamfer and Fillet commands. Elevation and Thickness are 3D options.

Notice that, instead of picking a second point to draw the rectangle, you have the option of entering dimensions. Say you wanted to draw a rectangle 20 drawing units long and 10 drawing units wide. The command sequence would look like this:

### Command Sequence

Command: RECTANG

Specify first corner point or [Chamfer/Elevation/Fillet/Thickness/Width]: (pick a point)

Specify other corner point or [Dimensions]: D

Specify length for rectangles <0.0000>: 20

Specify width for rectangles <0.0000>: 10

Specify other corner point or [Dimensions]: (pick a point to fix the orientation)

### The Polygon Command

Toolbar Draw

Pull-down Draw Polygon

Keyboard POLYGON short-cut POL

The Polygon command can be used to draw any regular polygon from 3 sides up to 1024 sides. This command requires four inputs from the user, the number of sides, a pick point for the centre of the polygon, whether you want the polygon *inscribed* or *circumscribed* and then a pick point which determines both the radius of this imaginary circle and the orientation of the polygon. The polygon command creates a closed polyline in the shape of the required polygon.

This command also allows you to define the polygon by entering the length of a side using the Edge option. You can also control the size of the polygon by entering an exact radius for the circle. Follow the command sequence below to see how this command works.

### Command Sequence

Command: POLYGON

Enter number of sides <4>: 5

Specify center of polygon or [Edge]: (pick P1 or type E to define by edge length)

Enter an option [Inscribed in circle/Circumscribed about circle] <I>: . (to accept the inscribed default or type C for circumscribed)

Specify radius of circle: (pick P2 or enter exact radius)

In the illustration above, the polygon on the left is inscribed (inside the circle with the polygon vertexes touching it), the one in the middle is circumscribed (outside the circle with the polyline edges tangential to it) and the one on the right is defined by the length of an edge.

## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

### Circles, Arcs etc.

Along with Line and Polyline, the Circle command is probably one of the most frequently used. Fortunately it is also one of the simplest. However, in common with the other commands in this section there are a number of options that can help you construct just the circle you need. Most of these options are self-explanatory but in some cases it can be quite confusing. The Circle command, for example, offers 6 ways to create a circle, while the Arc command offers 10 different methods for drawing an arc. The sections below concentrate mainly on the *default* options but feel free to experiment.

### The Circle Command

Toolbar Draw 

Pull-down Draw ► Circle ► Center, Radius

Keyboard CIRCLE short-cut C

Circle command is used to draw circles. There are a number of ways you can define the circle. The default method is to pick the center point and then to either pick a second point on the circumference of the circle or enter the circle radius at the keyboard.

### Command Sequence

Command: CIRCLE

Specify center point for circle or [3P/2P/Ttr (tan radius)]: (pick P1)

Specify radius of circle or [Diameter] <50.0195>: (pick P2 or enter the exact radius)

As you can see from the command prompt above the default options are always indicated in triangular brackets like so <Default> and command options appear within square brackets like so [Option]. Each *option* is separated by a forward slash like this /. You can choose to use the alternative options by typing them at the prompt. For example, the circle command gives you three extra options to define a circle. 3P which uses any three points on the circumference, 2P which uses two points on the circumference to form a diameter and Ttr which stands for Tangent Radius. Obviously to use this last option you need to have drawn two lines which you can use as tangents to the circle. Try these options out to see how they work. Note that to invoke a command option; you need only type the upper-case part of the option name. For example, if you want to use the Ttr option, you need only enter "T". There are two more circle options on the pull-down menu that enable you to draw a circle by defining the center and diameter or by using 3 tangents.



## Engineering Drawing & Workshop

NED University of Engineering and Technology- Department of Electronic Engineering

### The Ellipse Command

Toolbar    Draw  
Pull-down    Draw.    Ellipse.    Axis,  
                  End  
Keyboard    ELLIPSE    short-cut EL

The Ellipse command gives you a number of different creation options. The default option is to pick the two end points of an axis and then a third point to define the eccentricity of the ellipse. After you have mastered the default option, try out the others.

### Command Sequence

Command: ELLIPSE

Specify axis endpoint of ellipse or [Arc/Center]: (pick P1)

Specify other endpoint of axis: (pick P2)

Specify distance to other axis or [Rotation]: (pick P3)

### EXERCISE:

1. Draw a rectangle with first corner point 5,5 length=6, width=7 and second corner point at 12,13
2. Draw a polygon inscribed in a circle by following the given command sequence  
Command: polygon  
Enter number of sides <5>: 5  
Specify center of polygon or [Edge]: 15, 12.5  
Enter an option [Inscribed in circle/Circumscribed about circle] <I>: i  
Specify radius of circle: 5
3. Draw a polygon circumscribed about a circle by specifying your own values.
4. Draw the circles by following the given command sequences

Command: CIRCLE

Specify center point for circle or [3P/2P/Ttr (tan tan radius)]: 3p

Specify first point on circle: 5, 2

Specify second point on circle: 5, 10

Specify third point on circle: 10, 5

Command: CIRCLE

Specify center point for circle or [3P/2P/Ttr (tan tan radius)]: 2p

Specify first end point of circle's diameter: 5, 5

Specify second end point of circle's diameter: 10, 10

**Note:** Take the screenshots of all the drawings and submit your lab before the next lab session.

## **Engineering Drawing & Workshop**

*NED University of Engineering and Technology- Department of Electronic Engineering*

### **Bibliography:**

- INTRODUCTION TO MATLAB FOR ENGINEERING STUDENTS  
By David Houcque (Northwestern University)
- <http://www.kpsec.freeuk.com/breadb.htm>
- [www.mathworks.com](http://www.mathworks.com)
- [www.rapidonline.com](http://www.rapidonline.com)
- <http://en.wikipedia.org/>
- <http://www.kpsec.freeuk.com/solder.htm>
- Lab View : Lab VIEW Graphical Programming Course  
Collection edited by: National Instruments