Generating Lisp Program For Assembly Drawing In Autocad

T. Isaac Samuel Raja[1], D. Rabi Moses[2], T. Priscilla Dhava sundari[3]
[1, 2] Dept of Mechanical engineering
[3]Dept of Electronics and Communication engineering
[1, 2] Government College of Engineering, Tirunelveli.
[3]AlagappaChettair Government College of Engineering and Technology, Karaikudi

Abstract- *AutoLISP is a dialect of the LISP programming language built specially to use with AutoCAD and its derivatives. It is a subset of the LISP (List Processor) programming language, which is used in bids of artificial intelligence and expert systems. Many functions have been added to the LISP program in order to interface AutoLISP directly to AutoCAD, and you will see that some AutoCAD commands have been retained as AutoLISP functions. Flange Coupling is also a simple type of coupling than others. Here it consists of two flanges one keyed to the driving shaft and the other two the driven shaft. The two flanges are connected with the help of four or six bolts arranged in a concentric circle .In this thesis a flange coupling model is designed with simple programming language. Initially, transmitting power depending on the application is taken as the input for the generating various dimensions of the coupling.*

Keywords- AUTOLISP, AutoCAD, Drafting, Programming language, Flange coupling.

I. INTRODUCTION

After the development of AutoCAD, conscripting of engineering drawing becomes easier. In case of Mechanical Engineering drawing, drafting of machine element with small describing at any scale is made possible. Also main benefit of AutoCAD is to increase competence and reduce in time. But to achieve this benefits user must have thorough information of AutoCAD. Though one may have good command on CAD, he becomes bore in detailing in drawing elements and errors may be happens. To eradicate this problem, we can customize the AutoCAD. AutoLISP is the powerful old tool to automate the drawing. In AutoLISP, programmer assigns mathematical equations to get specific drawings by getting inputs from the user. After obtaining inputs or data from the user the application produces the image.

A. AutoCAD

AutoCAD a desktop application since 1982 is available as software application for copmputer aided designing and drafting software. Later 2010 as a mobile web- and cloud-based app marketed as AutoCAD 360. Established and promoted by

Autodesk, Inc., AutoCAD was first launched in December 1982, running on PCs with graphics managers. Mainframe computers were employed to integrate with CAD operations, they perform commercial CAD operations. Later they were modified to be used with minicomputers and as well as in modern computers.

B. AutoCAD VBA

Visual basic application is specially designed to modify the application programming interface (API). VBA or the visual basic for application is a normal or general programming language. VBA is an application of Microsoft but can aid to be used along with various working directories one such thing is AutoCAD. In AutoCAD, Visual basic application is allied to access objects through the ActiveX Automation interface.

C. AutoLISP

LIST Processing is an acronym of LISP. One of the main application and use of LISP program is Auto LISP, specially build by automation engineers for use with the AutoCAD versions and in some of its derivatives like AutoCAD Map 3D, AutoCAD Architecture and AutoCAD Mechanical. Neither the user interface design line nor the transcriber to effect AutoLISP code are involved in the AutoCAD product line.

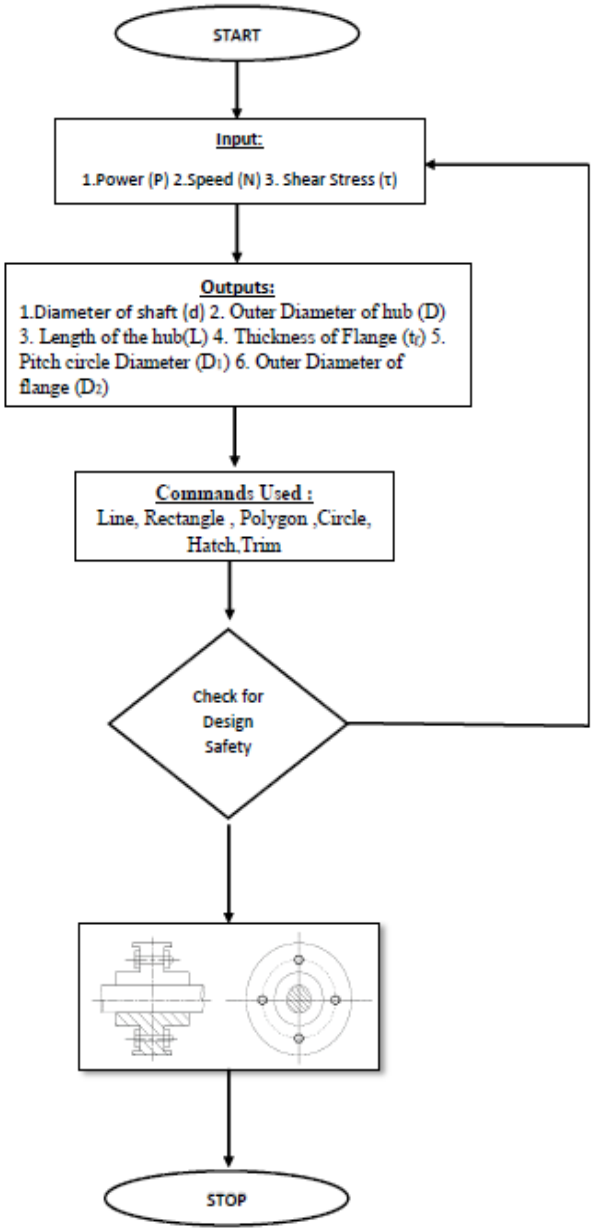
D. Benefits of AutoLISP

- All the manual drafting operations can be done individually by AutoLISP.
- LISP has certain set of tasks which allows us to perform many operations automatically in a very less time.

- They can even handle loops, mathematical operations, error spotting and corrections.
- Once the program for a machine assembly or component is done many other forms can be generated based on the inputs from the parent assembly drawing itself.

II. METHODOLOGY

Initially the inputs for the assembly drawing generation is given. From which the generated program will produce the output in terms of dimensdions. By using basic commands the entire assembly drawing is modeled which is given in the upcoming sections.



III. PROBLEM DESCRIPTION

The flange coupling with the basic dimensions and its relations are given in the Fig 1. This is the simplest type of the coupling. Flange coupling is usually made up of cast iron material, they are very easy to be modeled and manufactured. It consists of a hollow hole in which the shaft is connected for transmitting the power to the desired parts. Various types of keys are attached to it for improving the stability between the mated components.

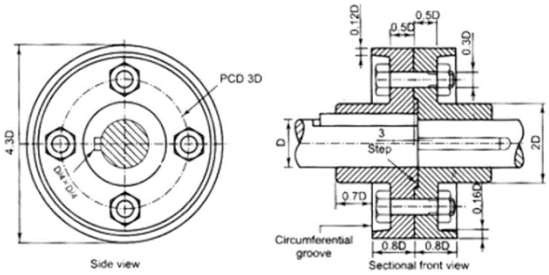


Fig 1. Layout of flange coupling

IV. ANALYTICAL CALCULATIONS

A flange coupling of protected type is designed to transmit power of 15hp which runs at 500 revolution/cycle.

A.Design of Hub

$$P = \frac{2\pi N T}{4500} \quad (1)$$

By substituting the values in Equation 1, Torque can be obtained as followed

T = 2150 Kgf-cm .

Transmitted by the shaft (T),

$$T = \frac{\pi}{16} \times \tau \times d^3 \quad (2)$$

d = 3.5 cm

Outer Diameter of hub, (D) = 2 d = 2 x 3.5 = 7 cm
Length of the hub, (L) = 1.5 d = 1.5 x 3.5 = 5.25 cm

B.Design of Key

Width of key, (w) = 1.2 cm
Thickness of Key (t) = w = 1.2 cm
Length of key (l)= L =5.25 cm

C.Design of flange

Thickness of Flange t_f= 0.5 d = 0.5 X 3.5 = 1.75 cm

D.Design of Bolts

Since the diameter of the shaft is 3.5cm, therefore let us take the number of bolts, n = 3

Pitch circle Diameter (D_1)= 3 d = 3 x 3.5
Outer Diameter of flange (D_2) = 4 d = 4 x 3.5 = 14 cm
Thickness of protective circumferential flange (tp) = 0.25 d = 0.25 x 3.5 = 0.875 say 1cm

V. AUTOLISP PROGRAMMING

The AutoLISP is a based lisp programming procedure .The various steps involved in AutoLISP programing are as follows

Step1 : Defining the drive where AUTOCAD is installed .
Step2 : Defining a name for our program it should be noted that ,no alphabets should be used which are to be used to define formula .For example "d" which is used to define the diameter. In turn this alphabet "d" should not be used to define our program .
Step3 : Defining the point on the working coordinate based on required diagram.
Step4 : By using commands such as line , hatch, circle, rectangle, triangle ,etc.

```
(defun c:xyz ()
(setq        n (getreal "enter the speed:")
        tou (getreal "enter the permissible shear stress in N/mm^2:")
        p (getreal "enter the magnitude of load in N:")
        pi (getreal "enter pi value")
        t (/ (* 45000 p) (* 2 pi n))
        d (expt (/ (* 16 t) (* pi tou)) (/ 1 3))
)
( setq pt1 '(0 0)
        pt2 (polar pt1 (/ pi 2) (* 0.16 d))
        pt3 (polar pt2 0 (* 0.3 d))
        pt4 (polar pt3 (/ pi 2) (* 0.99 d))
        pt5 (polar pt4 pi d )
        pt6 (polar pt5 (/ pi 2) (* 0.5 d))
        pt23 (polar pt6 0 (* 3 d))
        pt24 (polar pt23 (* 1.5 pi) (* 0.5 d))
        pt25 (polar pt24 pi d)
        pt26 (polar pt25 (* 1.5 pi) (* 0.99 d))
        pt27 (polar pt26 0 (* 0.3 d))
        pt28 (polar pt27 (* 1.5 pi) (* 0.16 d))
        pt29 (polar pt3 (/ pi 2) (* 0.145 d))
        pt30 (polar pt29  pi (* 0.25 d))
        pt31 (polar pt30 (/  pi 2) (* 0.7 d))
        pt32 (polar pt31 0 (* 0.25 d))
        pt33 (polar pt32 (* 1.5 pi) (* 0.2 d))
        pt35 (polar pt33 0 (* 1.5 d))
        pt36 (polar pt35 (* 1.5 pi) (* 0.3 d))
        pt34 (polar pt36 pi (* 1.5 d))
        pt37 (polar pt32 0 d )
        pt38 (polar pt37 0 (* 0.25 d))
        pt39 (polar pt38 (* 1.5 pi) (* 0.7 d))
        pt40 (polar pt39 pi (* 0.25 d))
        pt7 (polar pt6 pi (* 0.5 d))
        pt8 (polar pt7 (/ pi 2) d)
        pt21 (polar pt8 0 (* 4 d))
        pt22 (polar pt21 (* 1.5 pi) d)
        pt81 (polar pt3 (/ pi 2) (* 0.495 d))
        pt82 (polar pt81 pi (* 0.25 d))
        pt83 (polar pt82 (/ pi 2) (* 0.19 d))
        pt84 (polar pt83 0 (* 0.25 d))
        pt85 (polar pt83 (* 1.5 pi) (* 0.38 d))
        pt86 (polar pt85 0 (* 0.25 d))
        pt9 (polar pt8 0 (* 0.5 d))
        pt10 (polar pt9 (/ pi 2) (* 0.5 d))
        pt11 (polar pt10 0 d)
        pt61 (polar pt11 (/ pi 2 ) (* 0.345 d))
        pt62 (polar pt61 0 d )
        pt18 (polar pt62 (* 1.5 pi) (* 0.345 d))
        pt19 (polar pt18 0 d )
        pt20 (polar pt19 (* 1.5 pi) (* 0.5 d))
        pt63 (polar pt11 (/  pi 2) (* 0.145 d))
        pt64 (polar pt63 pi (* 0.25 d))
        pt65 (polar pt64 (/ pi 2) (* 0.7 d))
        pt66 (polar pt65 0 (* 0.25 d))
        pt67 (polar pt61 (/ pi 2) (* 0.3 d))
        pt68 (polar pt67 0 (* 1.5 d))
        pt69 (polar pt68 (* 1.5 pi) (* 0.3 d))
        pt12 (polar pt67 (/ pi 2) (* 0.345 d))
        pt13 (polar pt12  pi (* 0.3 d))
        pt14 (polar pt13 (/ pi 2) (* 0.16 d))
        pt15 (polar pt14 0 (* 1.6 d))
        pt16 (polar pt15 (* 1.5 pi) (* 0.16 d))
        pt17 (polar pt16 pi  (* 0.3 d))
        pt70 (polar pt17 (* 1.5 pi) (* 0.345 d))
        pt71 (polar pt66 0 d)
        pt72 (polar pt71 0 (* 0.25 d))
        pt73 (polar pt72 (* 1.5 pi) (* 0.7 d))
        pt74 (polar pt73 pi  (* 0.25 d))
        pt90 (polar pt21 (* 1.5 pi)  (* 0.5 d))
        pt91 (polar pt90 0 (* 3 d))
        pt92 (polar pt91 (/ pi 2) (* 1.5 d))
        pt93 (polar pt91 0 (* 1.5 d))
        pt94 (polar pt91 (* 1.5 pi) (* 1.5 d))
        pt95 (polar pt91 pi (* 1.5 d))
```

```
        pt100 (polar pt1 0 (* 0.8 d))
        pt101 (polar pt100 (/ pi 2) d )
        pt102 (polar pt101 0 (* 0.1 d))
        pt103 (polar pt102 (/ pi 2) (* 2.3 d))
        pt104 (polar pt103 pi (* 0.1 d))
        pt105 (polar pt104 (/ pi 2) d )
        pt106 (polar pt102 ( / pi 2) (* 1.57 d ))
        pt107 (polar pt106 (/ pi 2) (* 0.17 d ))
        pt108 (polar pt107 0 (* 1.7 d ))
        pt109 (polar pt108 ( * 1.5 pi ) (* 0.17 d ))
        pt110 (polar pt91 ( / pi 2 ) (* 0.4376 d ))
        pt111 (polar pt110 pi  (* 0.07 d ))
        pt112 (polar pt111 (/ pi 2)  (* 0.15 d ))
        pt113 (polar pt112 0  (* 0.14 d ))
        pt114 (polar pt113 (* 1.5 pi ) (* 0.15 d ))
)
( command "pline" pt1 pt2 pt3 pt4 pt5 pt6 pt23 pt24 pt25 pt26 pt27 pt28 "c")
( command "pline" pt29 pt30 pt31 pt32  "c")
( command "pline" pt33 pt35 pt36 pt34 "c")
( command "pline" pt37 pt38 pt39 pt40 "c")
( command "pline" pt7 pt8 pt21 pt22 "c")
(command "line" pt81 pt82 "")
(command "line" pt83 pt84 "")
(command "line" pt85 pt86 "")
( command "pline" pt9 pt10 pt11 pt61 pt62 pt18 pt19 pt20 "c")
(command "hatch" "solid" "last" "")
( command "pline" pt63 pt64 pt65 pt66 "c")
( command "pline" pt61 pt67 pt68 pt69 "c")
( command "pline" pt67 pt12 pt13 pt14 pt15 pt16 pt17 pt70 "c")
(command "hatch" "solid" "last" "")
( command "pline" pt71 pt72 pt73 pt74 "c")
( command "circle" pt91 (* 2.15 d )"")
( command "circle" pt91 (* 2 d )"")
( command "circle" pt91 (* 0.5 d )"")
(command "hatch" "solid" "last" "")
( command "circle" pt91 d "")
( command "polygon" 6 pt92 "I" (* 0.35 d) "")
( command "polygon" 6 pt93 "I" (* 0.35 d) "")
( command "polygon" 6 pt94 "I" (* 0.35 d) "")
( command "polygon" 6 pt95 "I" (* 0.35 d) "")
( command "pline" pt100 pt101 pt102 pt103 pt104 pt105 "")
( command "pline" pt106 pt107 pt108 pt109 "c")
(command "hatch" "solid" "last" "")
( command "pline" pt111 pt112 pt113 pt114 "c")
(command "hatch" "solid" "last" "")
(command "ucsicon" "no" "")
(command "zoom" "extents")
(command "regen" )
)
```

VI. CONCLUSION

AutoCAD is one of the most versatile 2D and 3D CAD software available in the market. In addition to that, the AUTOLISP Programming provides and Intuitive approach towards generating automatic diagrams with minimal user input. The Program can also be saved as a Macro or as an AutoCAD plug-in which can then be used in any type of systems and can be made commercial. Even a first time Novice user can be able to understand the level of programming in LISP as it uses a common representation of equations and command history which is easy to understand. Thus the flange coupling has been designed for transmitting the power and found the designed values within the allowable limits and the modelling of Flange coupling has been successfully done by using the AutoCAD LISP.

REFERENCES

[1] GuanZhi-chao and Fang Xiao-chu "A Generation Method of Mechanical Assembly Drawing in CAD" in Wuhan University Journal of Natural Sciences Vol. 5 No. 3 pp.,298~300, 2000.

[2] Muhammad Akilu1 and Hassan Ibrahim Ogiri "Autocad Programming For Reinforced Concrete Structural Detailing" in International Journal of Scientific Engineering and Technology

[3] Mohammed Amine, Dirhar, Essaid and El Kennassi "3D Autolisp Mesh Generation" in Journal Of Advanced Research In Mechanical Engineering And Technology Vol. 4 No. 1 , 2017

[4] Edelsbrunner H, "Geometry and Topology for Mesh Generation." in Cambridge, United Kingdom, Cambridge University Press, 2001

[5] The ABC's of AutoLISP by Omura George, 2001

[6] Gabriel, Richard P. "Performance and Evaluation of Lisp Systems" in MIT Press ,1985.

[7] George P.L , "Automatic mesh generation and finite element computation." In Handbook of Numerical Analysis Elsevier Science B.V, Vol. IV, 1996.

[8] Baker T. J., "Automatic Mesh Generation for Complex Three-Dimensional Regions Using a Constrained Delaunay Triangulation." In Engineering with Computers, vol. 5, pp. 161-175, 1989.