

## Lời mở đầu

Trong thời buổi công nghệ phát triển như hiện nay, không khó để một cá nhân có thể sở hữu cho mình một chiếc máy tính đủ dùng cho việc thiết kế ra một sản phẩm cơ khí. Và trong rất nhiều sản phẩm nổi trội trên thị trường chuyên biệt cho việc thiết kế như Soliworks, Catia, Inventor, Autocad... thì có thể nói việc tiếp cận và sử dụng phù hợp nhất với người mới bắt đầu có lẽ sẽ là những sản phẩm từ hãng phần mềm Autodesk tiêu biểu là Autocad. Trong khuôn khổ cho báo cáo môn học kỳ này, chúng em xin được đưa ra cách thiết kế ( vẽ ) một sản phẩm với việc ứng dụng công cụ lập trình AUTOLISP được tích hợp trên phần mềm Autocad. Với đề tài là tạo dựng bản vẽ **“cụm trục ra hộp giảm tốc bánh răng côn một cấp”** chúng em hi vọng nó là minh họa thiết thực cho những ai đang chuẩn bị hay đã được biết đến sẽ có cái nhìn tổng quan hơn cũng như sự thiết thực trong ứng dụng của Autolisp đối với bài toán thiết kế hiện nay. Nếu xem xét một cách cẩn thận có thể nhìn thấy tính linh hoạt cũng như khả năng mở rộng của Autolisp là rất cao. Với việc liên kết với các ứng dụng văn phòng hay những công cụ lập trình khác sẽ là điểm mạnh cho công cụ này phát triển.

Qua báo cáo lần này, nhóm chúng em xin chân thành cảm ơn thầy **Trịnh Đồng Tính** – GV môn học Tự động hoá thiết kế đã tận tình chỉ dạy và hướng dẫn chúng em về kiến thức của môn học mang lại cũng như hoàn thiện bản báo cáo này!

Về cơ bản báo cáo đã hoàn thành nhưng không thể tránh khỏi thiếu sót hoặc chưa hợp lý, rất mong góp ý, phản hồi từ Thầy để báo cáo chúng em được hoàn chỉnh hơn.

Một lần nữa em cảm ơn !

Nhóm sinh viên thực hiện : Nhóm 9

1. Nguyễn Văn Thuận – Nhóm trưởng
2. Phạm Văn Quốc
3. Phan Văn Tuyên
4. Lê Văn Thiện
5. Hồ Việt Khánh
6. Đặng Đức Đạt

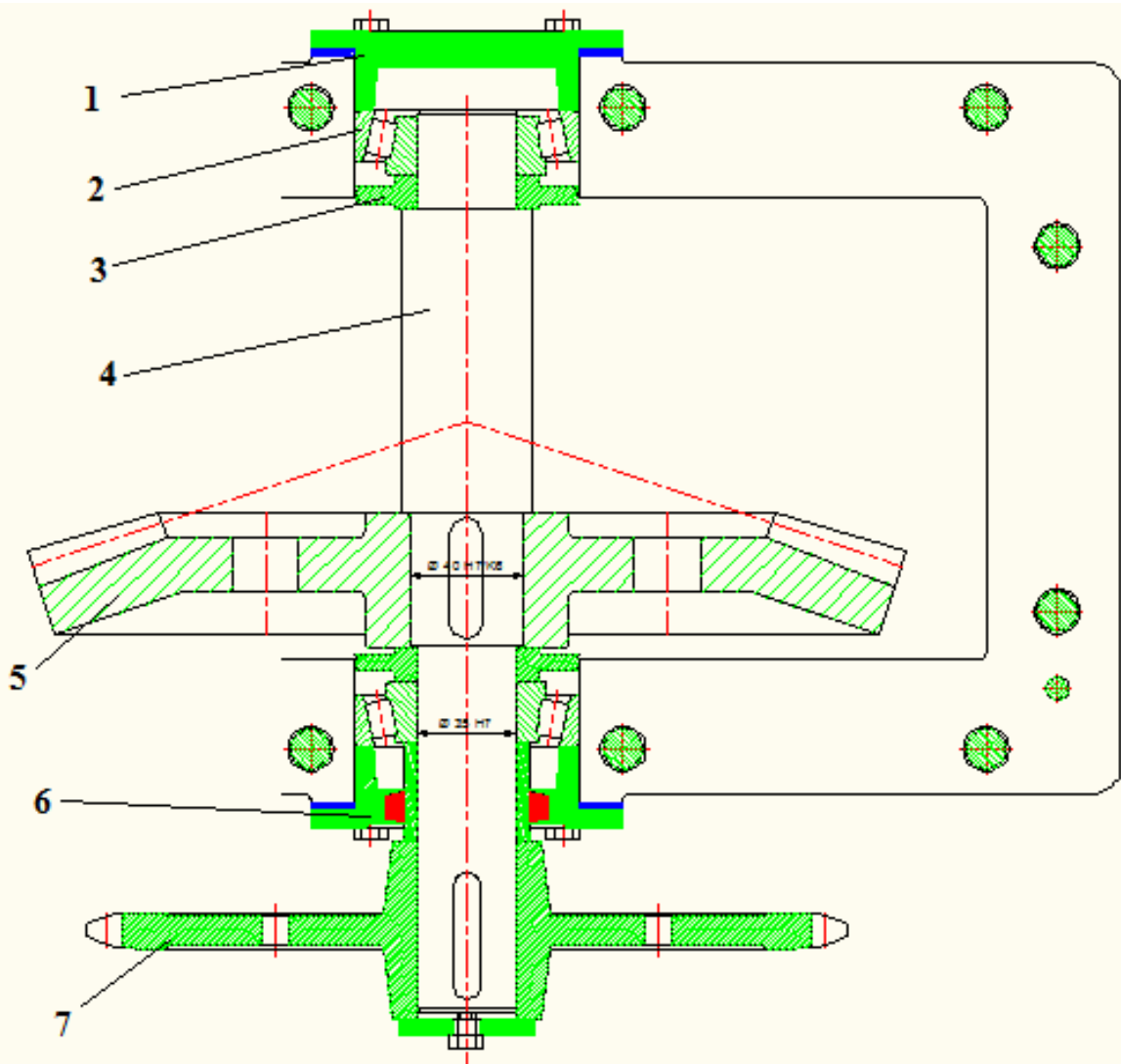
## MỤC LỤC

<b>Lời mở đầu</b> .....	1
<b>MỤC LỤC</b> .....	2
<b>YÊU CẦU ĐỀ TÀI</b> .....	3
<b>PHÂN CÔNG CÔNG VIỆC</b> .....	5
<b>CHƯƠNG 1. TỔNG QUAN VỀ AUTOLISP</b> .....	6
1.1. Giới thiệu về Autolisp.....	6
1.1.1 Sơ lược về LISP .....	6
1.1.2 Lịch sử phát triển của Autolisp .....	6
1.2. Ưu nhược điểm của Autolisp .....	6
1.2.1 Ưu điểm .....	6
1.2.2 Nhược điểm.....	6
1.3. Những khó khăn khi tiếp cận với Autolisp .....	6
1.4. Một số khái niệm và cú pháp lập trình .....	7
1.4.1 Giới thiệu .....	7
1.4.2 Biến .....	7
1.4.3 Hàm.....	7
1.4.4 Kiểu dữ liệu.....	8
1.4.5 Bảng mã DXF .....	8
1.4.6 Dữ liệu mở rộng.....	8
1.4.7 Điều kiện.....	9
1.4.8 Vòng lặp.....	9
1.4.9 Ngôn ngữ lập trình điều khiển hộp thoại DCL.....	9
1.4.10 Hướng đối tượng.....	9
<b>CHƯƠNG 2 . NỘI DUNG CHI TIẾT</b> .....	10
2.1. Các dữ liệu cần nhập vào bảng số liệu Demo .....	10
2.2. Cách thức xây dựng bản vẽ chi tiết và cụm chi tiết .....	11
2.2.1 Cách thức xây dựng bản vẽ chi tiết .....	11
2.2.2 Cách thức xây dựng cụm chi tiết.....	11
2.3. Các quan hệ kích thước sử dụng để thiết lập bản vẽ từ các số liệu ban đầu.....	12
2.3.1 Chi tiết trực.....	12
2.3.2 Ổ đĩa côn .....	13

---

2.3.3. Bánh răng côn .....	14
2.3.4. Đĩa xích.....	15
2.3.5. Nắp ổ 1 (ổ thông) .....	16
2.3.6. Nắp ổ 2 ( không thông ) .....	17
2.3.7. Vòng chắn dầu .....	17
2.4. Cách tiến hành code và kết quả chạy chương trình.....	19
2.4.1 Chi tiết trục .....	19
2.4.2 Ổđũa côn.....	20
2.4.3 Bánh răng côn .....	20
2.4.4 Đĩa xích.....	21
2.4.5 Nắp ổ 1 (ổ thông) .....	22
2.4.6 Nắp ổ 2 (không thông) .....	22
2.4.7 Vòng chắn dầu .....	23
2.4.8 Hoàn thiện.....	24
<i>Nguồn tham khảo</i> .....	24

## YÊU CẦU ĐỀ TÀI



### ĐỀ 8 : CỤM TRỤC RA HGT BÁNH RĂNG CÔN MỘT CẤP

1. Nắp ổ
2. Ố đĩa côn
3. Vòng chắn dầu
4. Trục
5. Bánh răng côn lớn
6. Nắp bánh răng côn lớn
7. Đĩa xích

Hình 1 : Yêu cầu của đề tài

**PHÂN CÔNG CÔNG VIỆC***Bảng 1 : Phân công trách nhiệm các thành viên*

<b>STT</b>	<b>TÊN</b>	<b>NHIỆM VỤ</b>	<b>VỊ TRÍ</b>
1	<b>NGUYỄN VĂN THUẬN</b>	+ Phân công công việc + Vẽ trục, tổng hợp và ghép các chi tiết, ghi kích thước + Hoàn thiện chương trình chính. + Soạn thảo Báo cáo	Nhóm Trưởng
2	<b>PHẠM VĂN QUỐC</b>	+ Tính toán kết cấu + Vẽ bánh răng côn và vòng dẫn dầu. + Kiểm tra chương trình chính + Hoàn thiện chương trình chính	Thành Viên
3	<b>PHAN VĂN TUYỀN</b>	+ Tính toán kết cấu + Vẽ đĩa xích và nắp ổ thông. + Kiểm tra chương trình chính	Thành Viên
4	<b>HỒ VIỆT KHÁNH</b>	+ Tính toán kết cấu + Vẽ nắp ổ 1 và ổ đĩa côn. + Kiểm tra chương trình chính	Thành Viên
5	<b>LÊ VĂN THIỆN</b>	+ Tính toán kết cấu + Vẽ vỏ hộp + Hoàn thiện bản báo cáo	Thành Viên
6	<b>ĐẶNG ĐỨC ĐẠT</b>	+ Tính toán kết cấu + Vẽ bạc lót + Hoàn thiện bản báo cáo	Thành Viên

## CHƯƠNG 1. TỔNG QUAN VỀ AUTOLISP

### 1.1. Giới thiệu về Autolisp

#### 1.1.1 Sơ lược về LISP

LISP – List Processing là một chuẩn ngôn ngữ lập trình được John McCarthy phát triển vào năm 1956 trong dự án nghiên cứu AI (Artificial Intelligence). Phiên bản đầu tiên LISP 1.5 được giới thiệu vào đầu thập niên 60 và phát triển với nhiều biến thể như: BBNLisp, Interlisp, MacLisp, NIL (New Implementation of Lisp), Franz Lisp... Vào thập niên 70 và đầu những năm 80 đã có máy tính chuyên dụng như LispMachines được thiết riêng để chạy những chương trình LISP. Đến năm 1981 để chuẩn hóa LISP các nhà lập trình đã tập hợp và chuẩn hóa thành chuẩn Common LISP. Năm 1984 Golden Common LISP trở thành chuẩn chính thức cho máy tính IBM và sau này phát triển thành XLISP- tiền thân của Autolisp ngày nay.

#### 1.1.2 Lịch sử phát triển của Autolisp

AutoLisp được phát triển từ XLISP là ngôn ngữ lập trình trên môi trường AutoCAD và được công bố phiên bản đầu tiên 2.18 vào tháng 01 năm 1986. Cùng với sự phát triển của AutoCAD các phiên bản của Autolisp ngày càng được hoàn thiện với nhiều tính năng mới, có thể kể đến một vài phiên bản tiêu biểu như sau:

Chính thức giới thiệu phiên bản 2.5 tích hợp vào AutoCAD R7 với một số tính tăng cơ bản về các tương tác với đối tượng trong bản vẽ.- Phiên bản 2.6 tích hợp vào AutoCAD R7 với chức năng 3D và một số hàm mới getcorner, getkword, và initget.- Phiên bản tích hợp vào AutoCAD R12 giới thiệu một số hàm GUI (Graphic User Interface) và ngôn ngữ điều khiển hộp thoại DCL (Dialog Control Language).- Phiên bản Visual LISP™ giới thiệu cùng với AutoCAD R14 là một môi trường phát triển Autolisp độc lập, trực quan với sự hỗ trợ của các công cụ gỡ rối.- Visual LISP™ được chính thức tích hợp vào AutoCAD 2000 và từ đó đến nay được bổ sung nhiều tính năng mới.

### 1.2. Ưu nhược điểm của Autolisp

#### 1.2.1 Ưu điểm

- Làm việc rất tốt và dễ dàng với điểm và các yếu tố hình học.
- Rất mềm dẻo, không khắt khe.
- Không cần trình dịch
- lập trình và thực hiện lệnh.
- Chạy được trên tất các các hệ điều hành với cùng 1 file Lisp.
- Quản lý đối tượng với List
- một kiểu dữ liệu với nhiều ưu điểm vượt trội trong quản lý tọa độ điểm.
- Mã nguồn mở và cộng đồng phát triển Autolisp rất rộng lớn.

#### 1.2.2 Nhược điểm

- Hình thức bên ngoài không hấp dẫn.
- Cú pháp khó hiểu.- Hạn chế, không có trình biên dịch.
- Ngôn ngữ trung gian nên thực thi chậm.
- Hầu như không thể tương tác với hệ thống.

### 1.3. Những khó khăn khi tiếp cận với Autolisp

Có thể khẳng định chắc chắn một điều là Autolisp là một ngôn ngữ rất dễ tiếp cận so với một số ngôn ngữ lập trình khác vì nó là ngôn ngữ lập trình theo kịch bản (Script). Tuy nhiên, để tiếp

cận được với Autolisp yêu cầu người học phải có kiến thức nền về lập trình và nắm vững về AutoCAD, đồng thời phải có kiến thức nhất định về hình học. Chương trình Autolisp là một tổ hợp những kịch bản được định trước nhằm điều khiển AutoCAD thực thi theo suy nghĩ của người thiết kế.

Đa số mọi người muốn học Autolisp là để giải quyết những bài toán trong lĩnh vực chuyên môn của mình. Để tiếp cận và ứng dụng tốt Autolisp trong công việc yêu cầu người lập trình phải có sự liên hệ với nhu cầu công việc thực tế, điều này phụ thuộc rất lớn vào sở trường của mỗi người. Bạn đang thực hiện một vài thao tác để hoàn thiện bản vẽ của mình và bạn chợt nhận ra nó cứ lặp lại liên tục. Một ý tưởng nảy ra là bạn cần thực hiện một đoạn chương trình Autolisp để tự động thực hiện các thao tác này và chương trình Autolisp được hoàn thành. Điều này có thể giải thích được vì sao một số người lại cảm thấy khó khăn khi tiếp cận với Autolisp mặc dù khả năng tư duy về lập trình của họ khá tốt.

## 1.4. Một số khái niệm và cú pháp lập trình

### 1.4.1 Giới thiệu

Một chương trình Autolisp luôn bắt đầu bằng dấu “(“ và kết thúc bằng dấu “)“.

Một chương trình Autolisp đơn giản như sau :

```
(defun myProg()
  (princ "Tecco 533")
  (princ)
)
```

Autolisp là ngôn ngữ trả về giá trị sau khi thực hiện lệnh. Bạn có thể kiểm tra điều này bằng cách mở AutoCad và gõ vào dòng lệnh (+ 1 2) trong mục command. Và ngay lập tức kết quả trả về là 3.

### 1.4.2 Biến

- Để gán giá trị trong Autolisp bạn cần sử dụng từ khoá “setq”, ví dụ với cú pháp : (setq a 1)

- Để kiểm tra giá trị của biến dùng từ khoá “!” với cú pháp : !a

- Giống một số ngôn ngữ lập trình khác Autolisp cũng quy định cách đặt tên biến như sau :

- + Không dùng các ký tự đặc biệt: \*, &, ^, \$ ..v.v..
- + Không dùng các từ khoá của AutoCad : LINE, PLINE, MIRROR..v.v..
- + Tên biến không phân biệt chữ hoa và chữ thường.

### 1.4.3 Hàm

Autolisp quy định từ khoá “defun” để định nghĩa hàm thực thi với cú pháp :

```
(defun myProg()
  (princ "Tecco 533")
  (princ)
)
```

Ngoài ra Autolisp còn sử dụng từ khoá C: sẽ khai báo với AutoCad là chương trình sẽ thực thi lệnh đầu nhắc lại lệnh Command với cú pháp :

```
(defun C:myProg() ;Command trong Cad myProg để chạy
(princ "Tecco 533")
(princ)
)
```

Với hàm đầu tiên để thực thi bạn phải gõ Command : (myProg) tại dòng nhắc lệnh còn với hàm thứ hai bạn chỉ cần gõ Command: myProg giống như một lệnh trong AutoCad.

#### 1.4.4 Kiểu dữ liệu

Một số kiểu dữ liệu thông dụng trong Autolisp như sau :

- String : Chuỗi gồm các ký tự và số
- Integers : Số tự nhiên
- Real : Số thực
- List : Kiểu dữ liệu đặc trưng và cũng là thế mạnh của LISP so với các ngôn ngữ lập trình khác.
- Associated List : Đây là kiểu dữ liệu định nghĩa các đối tượng trong AutoCad.

Dựa trên các kiểu dữ liệu trên Autolisp phân loại các nhóm hàm dựng sẵn như sau :

- Hàm xử lý chuỗi : substr, strlen, strcase, strcat,.
- Hàm xử lý số : abs, atof, atoi, fix, float, itoa.
- Hàm xử lý List : car, cdr, cadr, caddr, caar, caddr, foreach, list, cons, nth.
- Hàm chuyển đổi : fix, float, itoa, atoi, atof, rtos, angtos.
- Hàm toán học : +, -, \*, /, +1, -1, cos, atan, sin, sqrt, expt.
- Hàm lựa chọn thực thể entsel, ssget.
- Hàm xử lý tập chọn : ssadd, ssdel, sslength, ssname.
- Hàm xử lý đối tượng : entget, entlast, entnext, entdel, entmod, entupd.
- Hàm xử lý file : pen, close, read-line, write-line.

#### 1.4.5 Bảng mã DXF

AutoCad định nghĩa một đối tượng trên bản vẽ theo kiểu dữ liệu Associated List như sau :

```
((-1. ) (0. "LINE") (5. "22") (100 . "AcDbEntity") (67 . 0) (8 . "0") (62 . 4) (100 . "AcDbLine")
(10 3.39219 5.3243 0.0) (11 8.72878 3.10374 0.0) (210 0.0 0.0 1.0))
```

Đây là một tập hợp các cặp đôi (mã số . dữ liệu) được qui định trước. Tùy theo đối tượng và thuộc tính đối tượng mà Associated List sẽ có những tham số khác nhau. Các mã số này tuân theo một qui định trong bảng định nghĩa cho trước gọi là bảng mã DXF. Để có thể điều khiển được các đối tượng trong bản vẽ AutoCAD yêu cầu người lập trình phải hiểu rất rõ về bảng mã DXF này.

#### 1.4.6 Dữ liệu mở rộng

AutoCAD dùng các mã số từ 1000 đến 1042 để biểu diễn các dữ liệu mở rộng. Với dữ liệu mở rộng người lập trình có thể đánh dấu đối tượng trên AutoCAD để thực hiện các thao tác tiếp theo. Một ứng dụng điển hình trên AutoCAD sử dụng dữ liệu mở rộng này là chương trình Nova-TDN của



Công ty tin học Hải Hòa. Thông qua dữ liệu mở rộng chương trình có thể phân biệt được đâu là tìm tuyến, đâu là trắc dọc, cắt ngang... Toàn bộ dữ liệu mở rộng được định nghĩa trong Associated List với mã số-3.

Ví dụ :

Code :

```
((-3 ("TECCO533" (1000 . "Tim tuyen"))))
```

#### 1.4.7 Điều kiện

Cũng giống như một số ngôn ngữ lập trình khác Autolisp hỗ trợ người lập trình 2 cú pháp điều kiện là điều kiện xác định If và điều kiện lựa chọn Cond với cú pháp như sau :

```
(if<điều kiện>
)
(cond
)
...
)
)
```

#### 1.4.8 Vòng lặp

Autolisp không hỗ trợ vòng lặp For mà chỉ hỗ trợ 2 vòng lặp Repeat và While với cú pháp như sau :

Code :

```
(while <điều kiện>
)
(repeat
)
)
```

#### 1.4.9 Ngôn ngữ lập trình điều khiển hộp thoại DCL

Autolisp cung cấp cho người lập trình một ngôn ngữ điều khiển hộp thoại DCL để giải quyết về giao diện tương tác với người sử dụng. Thông qua ngôn ngữ DCL người lập trình có thể thiết kế các Form nhập liệu trực quan giúp cho chương trình trở nên thân thiện hơn.

#### 1.4.10 Hướng đối tượng

Bản thân Autolisp không phải là ngôn ngữ lập trình hướng đối tượng, nhưng cùng với xu hướng phát triển của lập trình hướng đối tượng bắt đầu từ Visual LISP™ cho AutoCAD R14 hãng AutoDesk đã tích hợp vào AutoCAD công nghệ ActiveX với kỹ thuật lập trình hướng đối tượng VLA (Visual LISP ActiveX). Thông qua công nghệ ActiveX người lập trình có thể điều khiển tất cả các đối tượng trên bản vẽ qua các thuộc tính và phương thức của nó.

## CHƯƠNG 2 . NỘI DUNG CHI TIẾT

Vậy ý tưởng để hoàn thành đề tài này là gì và thực hiện nó như thế nào?

Trên cơ sở là bản vẽ có sẵn chúng ta xác định từ bản vẽ các chi tiết tách rời và phân công trách nhiệm cho các thành viên tìm hiểu và xây dựng nó. Tuy nhiên không phải là các chi tiết này được thiết kế 1 cách rời rạc mà phải có sự thống nhất chung về bộ thông số. Thì trên cơ sở đó ta đi xác định các thông số nhập vào. Sau khi xác định được các thông số này, mỗi thành viên dựa trên các thông số này xem có sự ràng buộc nào với chi tiết mình sẽ làm hay ko và từ đó đưa nó vào bài làm. Có một số chi tiết có thể được chọn tùy ý ko bắt buộc thì những thông số này phải có sự thống nhất chung được xác định từ trước để tránh những sai sót nhằm lẫn dẫn tới lắp ghép sai, bởi vậy những thông số này phải có sự nắm bắt ngay từ lúc đầu.

Với mỗi chi tiết sẽ có thể có những ý tưởng khác nhau để vẽ, nhưng thông thường sẽ theo quy luật sau:

- ✓ Xác định khoảng cách và đặt tên cho biến khoảng cách
- ✓ Thiết lập sơ đồ điểm, xác định vị trí đầu tiên định vẽ sao cho thuận lợi nhất.
- ✓ Thực hiện lệnh vẽ đường nối điểm, fillet, chamfer... Để tạo đường bao hình
- ✓ Thực hiện lệnh đối xứng( nếu có )
- ✓ Thực hiện gạch vật liệu (nếu có )

Trên cơ sở là vậy, tuy nhiên để làm được điều đó đôi khi ngoài hàm chính ta còn phải bổ sung các hàm phụ để thực hiện một số thao tác như hàm lấy đối xứng, gạch vật liệu....

Trên cơ sở các chi tiết đã được dựng xong hoàn toàn thì đến bước quan trọng là lắp ghép các chi tiết thành cụm các chi tiết tạo nên một bộ phận máy, vậy để làm được điều này, chúng ta cần phải xác định các yêu cầu sau:

- ✓ Xác định tâm của cả cơ cấu
- ✓ Xác định các biến từ các chi tiết( thực ra là khoảng cách)
- ✓ Các thông số nhập, dữ liệu sẽ được gọi vào một hàm chung.
- ✓ Các hàm cho từng chi tiết sẽ được tách biệt từng phần cho dễ nhìn và lược bỏ những thứ mà phần thông số nhập và dữ liệu đã có
- ✓ Đừng quên gọi hàm của tất cả cho vào một hàm chính được thực thi bằng command (c: tên chương trình chính )

Như vậy là xong, bây giờ mọi thứ đã quá rõ ràng công việc tiếp theo sẽ là thực hiện từng bước như thế và hoàn thiện!

### 2.1. Các dữ liệu cần nhập vào bảng số liệu Demo

- Đường kính bánh răng côn :  $d_o$
- Số răng bánh răng côn to:  $Z_2$
- Số răng bánh răng côn nhỏ :  $Z_1$
- Mô đun :  $m$
- Bước xích :  $p$
- Số răng đĩa xích :  $z$

Bảng 2 . Bảng dữ liệu Demo

Thông số	Kích thước
dt	40
Z <sub>2</sub>	51
mte	6
Z <sub>1</sub>	17
P	19.05
z	41

## 2.2. Cách thức xây dựng bản vẽ chi tiết và cụm chi tiết

### 2.2.1 Cách thức xây dựng bản vẽ chi tiết

Cụm bản vẽ chi tiết được tách ra thành các chi tiết nhỏ như sau:

- *Nắp ổ*
- *Ổ đĩa côn*
- *Vòng chắn đầu*
- *Trục*
- *Bánh răng côn lớn*
- *Nắp ổ thông*
- *Đĩa xích*
- *Bạc lót*
- *Vít ổ trục*

### 2.2.2 Cách thức xây dựng cụm chi tiết

Chương trình được xây dựng bằng ngôn ngữ AutoLISP từ một chương trình chính và các chương trình con.

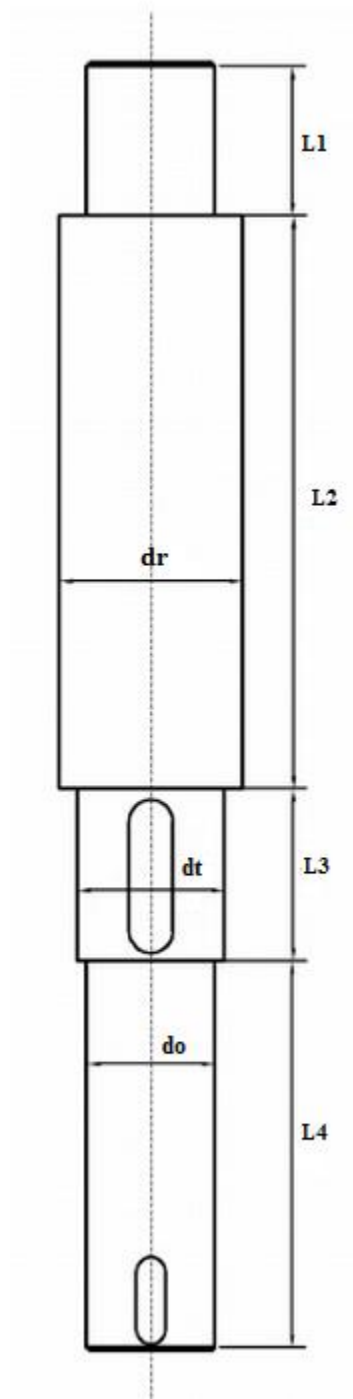
Chương trình chính: ở đây là chương trình yêu nhập số liệu, khai báo các biến hệ thống, các biến và các thông số cần thiết khác. Sau đó tiến hành vẽ trục nhờ vào các thông số đã biết, tính toán được nhờ công thức... Tiếp đó ta xác định các điểm trên trục cùng với những thông số đầu vào cần thiết gọi chương trình con vẽ các chi tiết ghép thành cụm chi tiết. Khi đã thành cụm chi tiết ta tiến hành vẽ nốt vỏ hộp.

Chương trình con ở đây là các chương trình vẽ các chi tiết trong cụm ra hộp giảm tốc bánh răng côn. VD: Chương trình con vẽ bánh răng côn, ổ đĩa côn, đĩa xích, v.v...

Phần cuối của toàn bộ chương trình là các hàm, chương trình con phục vụ cho quá trình vẽ các chi tiết và cụm chi tiết.

## 2.3. Các quan hệ kích thước sử dụng để thiết lập bản vẽ từ các số liệu ban đầu

## 2.3.1 Chi tiết trục



Hình 2 : Chi tiết trục

- ❖ Thông số đầu vào :
- ✓ Đường kính lỗ của bánh răng côn :  $d_t$
- ✓ Chiều dài moay – ơ bánh răng côn :  $l_m$
- ✓ Các thông số đầu vào của bánh răng côn :  $u, z_1, z_2, m_{te}$
- ✓ Chiều rộng ổ đĩa côn :  $B$
- ✓ Chiều rộng moay – ơ đĩa xích :  $B_0$

Thông số tính toán :

$d_o$ : đường kính lỗ của bánh răng côn.

$$L_1 = T + 12$$

$$L_2 = (R_e - b) \cos \delta_{a2} + R_e \sin \delta_{a1} + 20$$

$$L_3 = l_m - 1$$

$$L_4 = 20 + B + 35 + B_0 - 3$$

Trong đó :

$$+ R_e = 0,5 m_{te} \sqrt{z_1^2 + z_2^2} = 0,5 m_{te} \sqrt{z_1^2 + (u \cdot z_1)^2}$$

$$+ b = 0,3 R_e$$

$$+ \delta_1 = \arctan\left(\frac{z_1}{z_2}\right) \text{ và } \delta_2 = 90 - \delta_1$$

$$+ \delta_{a1} = \delta_1 + \theta_{a1} = \arctan\left(\frac{z_1}{z_2}\right) + \arctan\left(\frac{2}{\sqrt{z_1^2 + z_2^2}}\right)$$

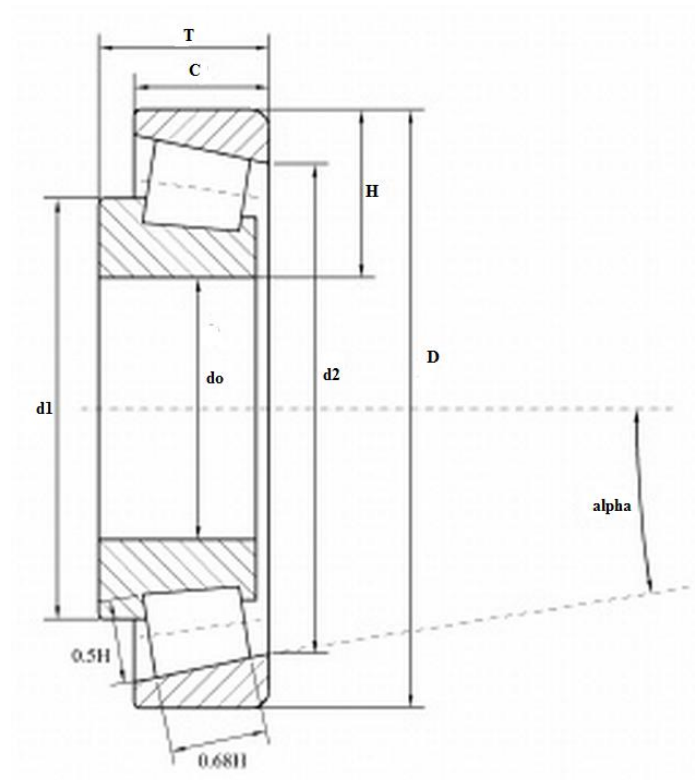
$$+ \delta_{a2} = 90 - \delta_1 + \theta_{a2} = 90 - \arctan\left(\frac{z_1}{z_2}\right) + \arctan\left(\frac{2}{\sqrt{z_1^2 + z_2^2}}\right) + l_m$$

$$+ B_0 = 1,8 \cdot d_o$$

$$+ l_m = 1,2 d_t$$

### 2.3.2 Ổ đĩa côn

Thông số đầu vào đường kính trục tại chỗ lắp ổ lăn :  $d_o = dt - 5$  (Với  $dt$  là đường kính trục lắp ổ đĩa côn).



Hình 3. Chi tiết ổ đĩa côn

Các thông số tra theo bảng :

**Ổ ĐŨA CÔN THEO GHOST 333-71-CỖ TRUNG BÌNH**

*Bảng 3 : Ổ đũa côn tiêu chuẩn GHOST 333-71*

Ký hiệu	d, m	D, m	D <sub>1</sub> , mm	d <sub>1</sub> , mm	B, m	C <sub>1</sub> , m	T, mm	r, m	r <sub>1</sub> , mm	a, (o)	C, kN	C <sub>0</sub> , kN
1	2	3	4	5	6	7	8	9	10	11	12	13
7304	20	52	42	35.4	15	13	16.25	2.0	0.8	11.17	25.0	16.6
7305	25	62	50.5	43.5	17	15	18.25	2.0	0.8	13.50	29.6	20.9
7306	30	72	58	50.6	19	17	20.75	2.0	0.8	13.50	40.0	29.9
7307	35	80	65.5	56.3	21	18	22.75	2.5	0.8	12.00	48.1	35.3
7308	40	90	74.5	62.5	23	20	25.25	2.5	0.8	10.50	61.0	46.0
7309	45	100	83.5	70.5	25	22	27.25	2.5	0.8	10.83	76.1	59.3
7310	50	110	92	76.5	27	23	29.25	3.0	1.0	11.67	96.6	75.9
7311	55	120	97.5	84	29	25	31.5	3.0	1.0	12.50	102.0	81.5
7312	60	130	108.5	94	31	27	33.5	3.5	1.2	11.50	118.0	96.3
7313	65	140	116.5	101	33	28	36	3.5	1.2	11.50	134.0	111.0
7314	70	150	126	104.5	35	30	38	3.5	1.2	11.67	168.0	137.0
7315	75	160	133	112.5	37	31	40	3.5	1.2	12.33	178.0	148.0
7317	85	180	150	130	41	35	44.5	4.0	1.5	11.83	221.0	195.0
7318	90	190	158	137	43	36	46.5	4.0	1.5	12.00	240.0	201.0

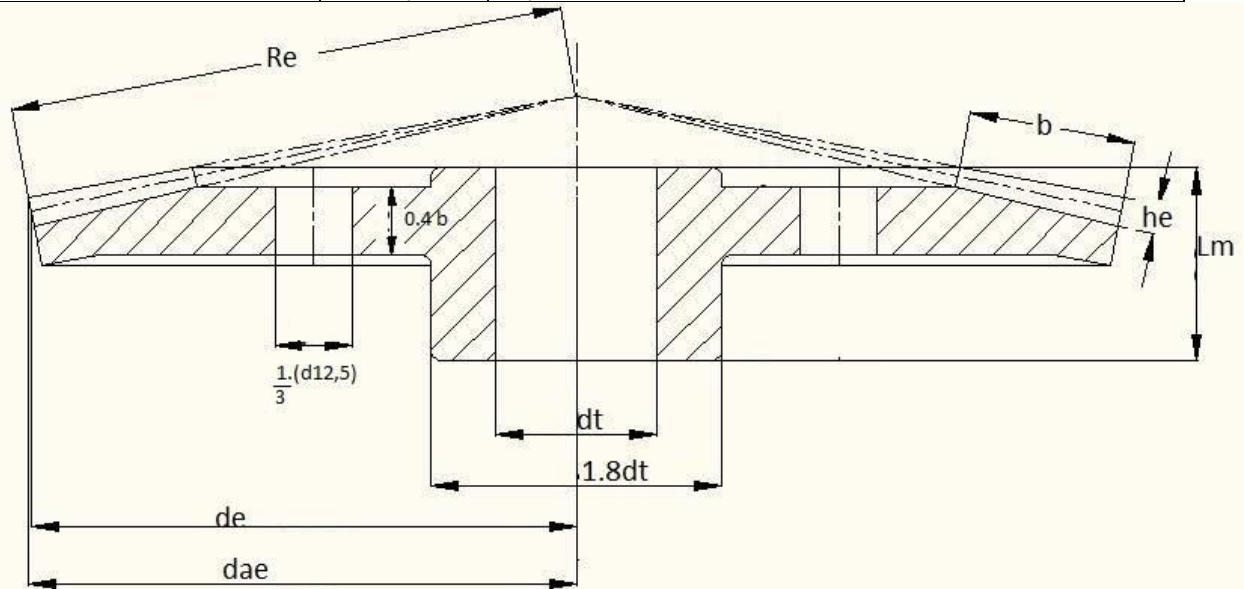
2.3.3. Bánh răng côn

Các công thức tính các thông số của bánh răng côn

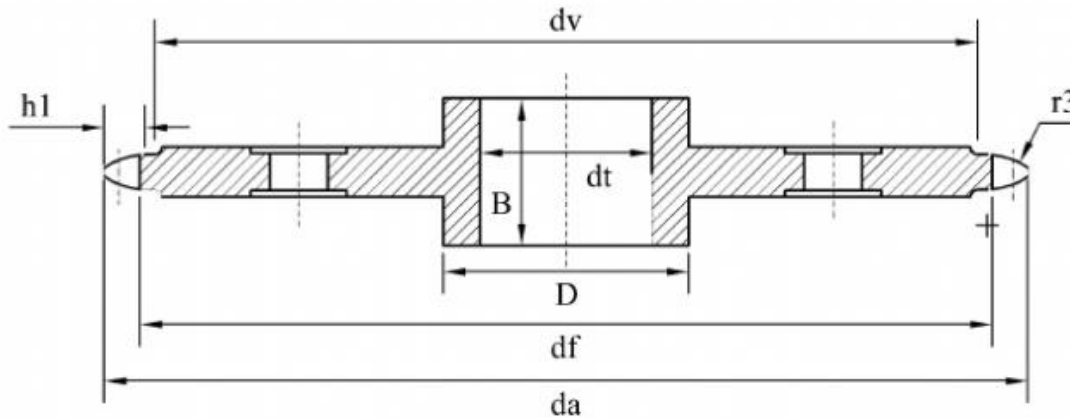
*Bảng 4 : Thông số của bánh răng côn*

Thông số	Kí hiệu	Công thức
Chiều dài côn ngoài	$R_e$	$R_e = 0,5m_{te}\sqrt{z_1^2 + z_2^2} = 0,5m_{te}\sqrt{z_1^2 + (uz_1)^2}$
Chiều rộng vành răng	b	$b = 0,3R_e$
Đường kính chia ngoài	$d_e$	$d_e = m_{te} \cdot z_1$
Góc côn chia	$\delta$	$\delta = \arctan(z_1/z_2)$
Chiều cao răng ngoài	$h_e$	$h_e = 2,2m_{te}$
Chiều cao đầu răng	$h_{ae}$	$h_{ae1} = m_{te}; h_{ae1} = 2m_{te} - h_{ae1} = m_{te}$
Đường kính đỉnh răng ngoài	$d_{ae}$	$d_{ae} = d_e + 2h_{ae}\cos \delta$
Chiều cao chân răng	$h_{fe}$	$h_{fe1} = h_{fe2} = 1,2m_{te}$
Đường kính trung bình	$d_m$	$d_m = (1 - 0,5 \frac{b}{R_e})d_e$
Góc đỉnh răng	$\theta_a$	$\theta_a = \arctan(\frac{h_{ae}}{R_e})$

Góc chân răng	$\theta_f$	$\theta_f = \arctan\left(\frac{h_{fe}}{R_e}\right)$
Góc côn đỉnh	$\delta_a$	$\delta_a = \delta + \theta_a$
Góc côn đáy	$\delta_f$	$\delta_f = \delta - \theta_f$
Chiều dài moay $\sigma$	$l_{m1}$	$l_{m1} = 1,2dt$ ; dt - đường kính trục lắp BR côn



2.3.4. Đĩa xích



Hình 5. Đĩa xích

- Đường kính vòng chia :  $d = \frac{p}{\sin(\frac{\pi}{z})}$  ;

Trong đó : p là bước xích chọn theo bảng sau :

Bảng 5 : Bước xích tiêu chuẩn

p								
8	9,525	12,7	15,875	19,05	25,4	31,7	38,1	44,45

- Z là số răng của đĩa xích
- Đường kính vòng đỉnh răng  $d_a = p \cdot [0,5 + \cotg(\frac{z}{\pi})]$
- Đường kính vòng đáy răng :  $d_f = d - 2r$

- Đường kính vành đĩa :  $d_v = p \cdot \cotg\left(\frac{\pi}{z}\right) - 1,2h$

Chiều cao h được tra theo bảng :

Bảng 6. Chiều cao h

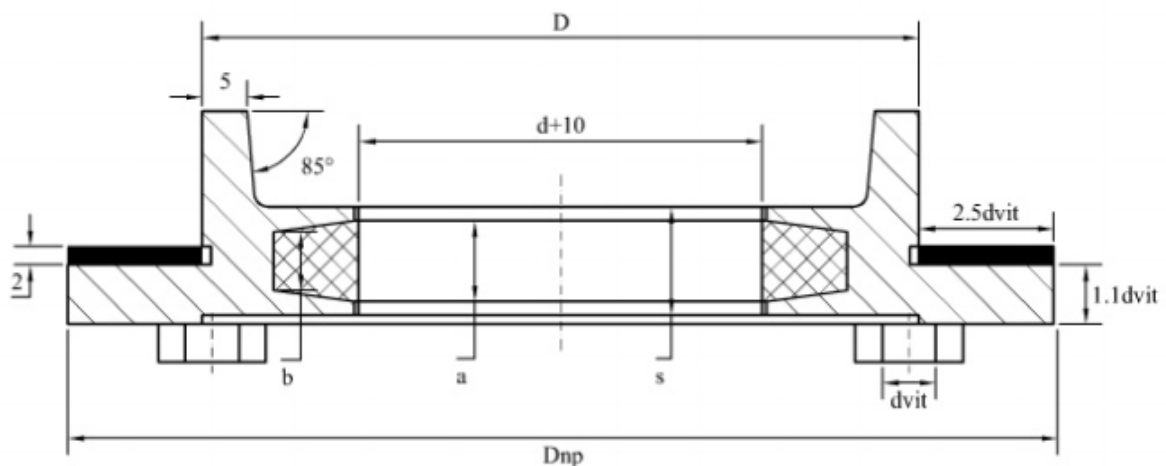
p		9,525	12,7	15,875	19,05	24,5	31,75
h	7,5	8,3	8,3	10,3	12,4	15,9	19,9

Các kích thước B, h,  $d_o$ ,  $d_1$  tra theo bảng sau :

Bảng 7: Các kích thước khác của đĩa xích theo tiêu chuẩn

Bước xích b	B	$d_o$	$d_1$	h
8	3	2.31	5	7.5
9.535	5.72	3.28	6.35	8.5
12.7	2.4	3.66	7.75	10
12.7	3.3	3.66	7.75	10
12.7	5.4	4.45	8.51	11.8
12.7	7.75	4.45	8.51	11.8
15.875	6.48	5.08	10.16	14.8
15.875	9.65	5.08	10.16	14.8
19.05	12.7	5.96	11.91	18.2
25.4	15.88	7.95	15.88	24.2
31.75	19.05	9.55	19.05	30.2

2.3.5. Nắp ổ 1 (ổ thông)



Hình 6. Nắp hộp 1



- Đường kính trong ổ đĩa côn :  $d$
- Đường kính ngoài ổ đĩa côn :  $D$
- Đường kính ngoài bạc lót :  $d+10$
- Đường kính vít :  $d_{vit} = 6mm$ .

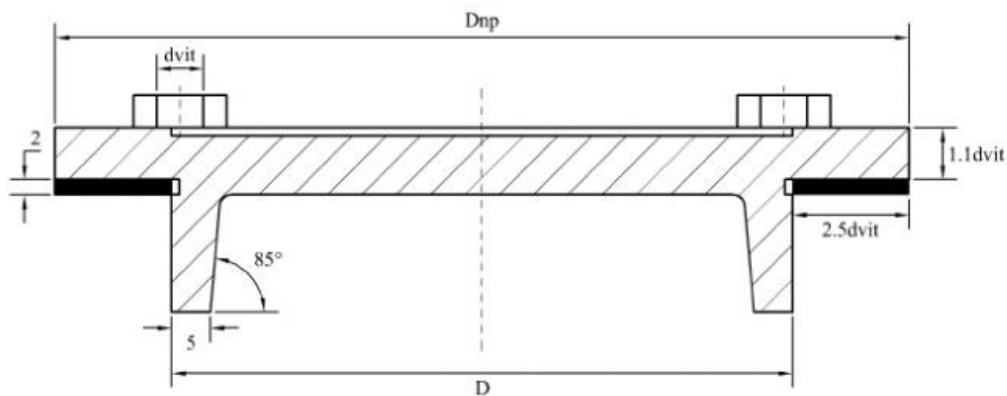
Ta có các thông số của vòng phốt như bảng sau :

Bảng 8: Các kích thước theo tiêu chuẩn của vòng phốt

d	$D_p$	a	b	$S_0$
20	33	6	4.3	9
25	38	6	4.3	9
30	43	6	4.3	9
35	48	9	6.5	12
40	59	9	6.5	12
45	64	9	6.5	12
50	69	9	6.5	12
55	74	9	6.5	12
60	79	9	6.5	12
65	84	9	6.5	12
70	89	9	6.5	12

### 2.3.6. Nắp ổ 2 ( không thông )

- Đường kính trong ổ đĩa côn :  $d$
- Đường kính ngoài ổ đĩa côn :  $D$
- Đường kính vít :  $d_{vit} = 6mm$ .

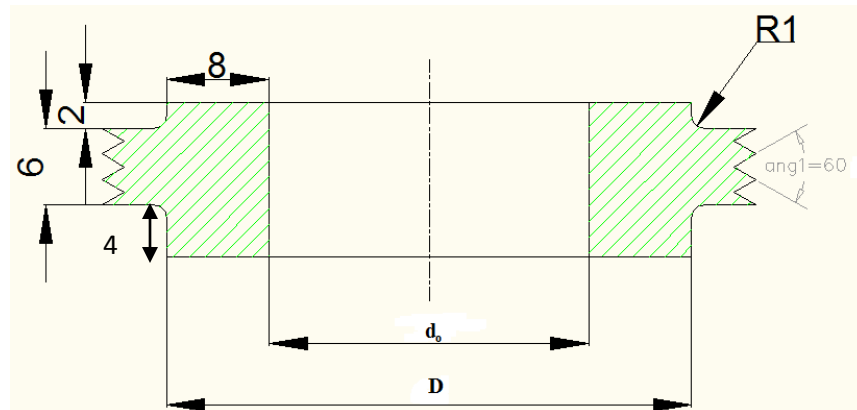


Hình 7. Nắp hộp 2

### 2.3.7. Vòng chắn dầu

- Đường kính trong  $d_o$  = đường kính trong ổ lăn
- Đường kính ngoài  $D$  = đường kính ngoài ổ lăn

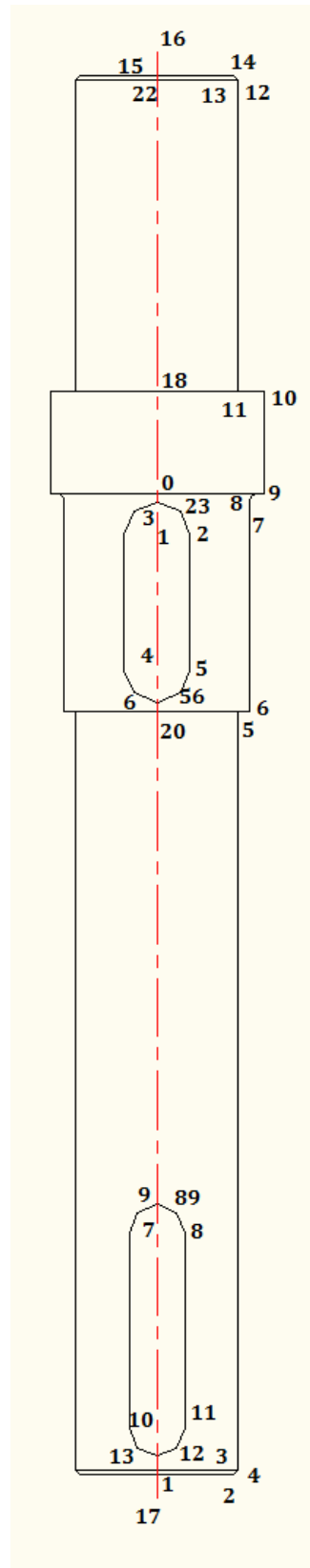
Các kích thước như trong hình vẽ (mặc định khi vẽ).



Hình 8 : Vòng chắn dầu

## 2.4. Cách tiến hành code và kết quả chạy chương trình

## 2.4.1 Chi tiết trục

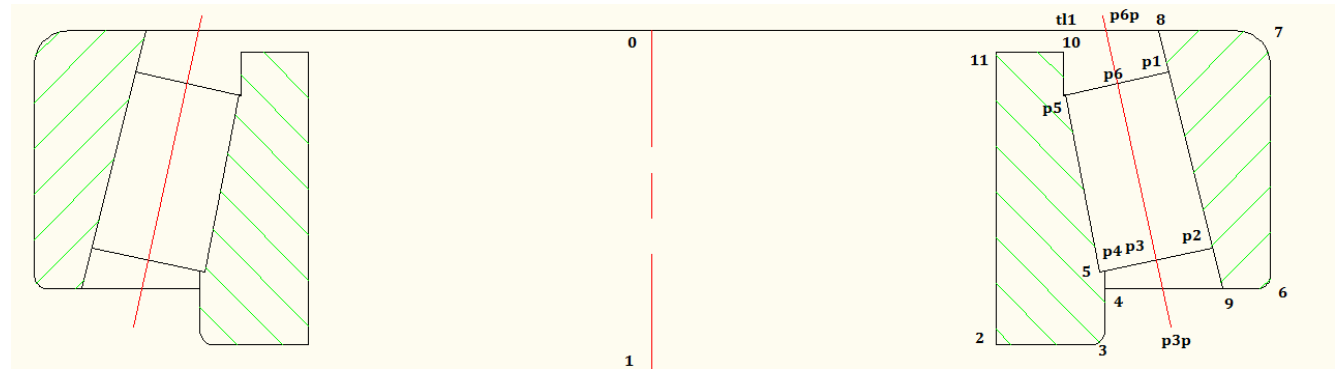


Cách tiến hành vẽ :

- Lưu các biến hệ thống
- Đặt lại các biến số hệ thống để tránh sai sót
- Đánh số các điểm (hình 9: Nằm trên trục là “tr\_số, trên phần then là th\_số)
- Vẽ đường tâm tr16-tr17
- Vẽ lần lượt các đường bao bằng pline(tr1-tr2-tr3-tr4-tr5-tr6-tr7-tr8-tr9-tr10-tr11-tr12-tr13-tr14-tr15;th3-th23-th2-th5-th56-th6) và fillet với bán kính R(tr2-tr3-tr4;tr7-tr8-tr9;th2-th23-th3;th5-th56-th6); đưa các đường này vào nhóm chọn ss(để lấy đối xứng).
- Lấy đối xứng nhóm chọn qua tr16-tr17

Hình 9 : Chi tiết trục

2.4.2 Ổ đũa côn

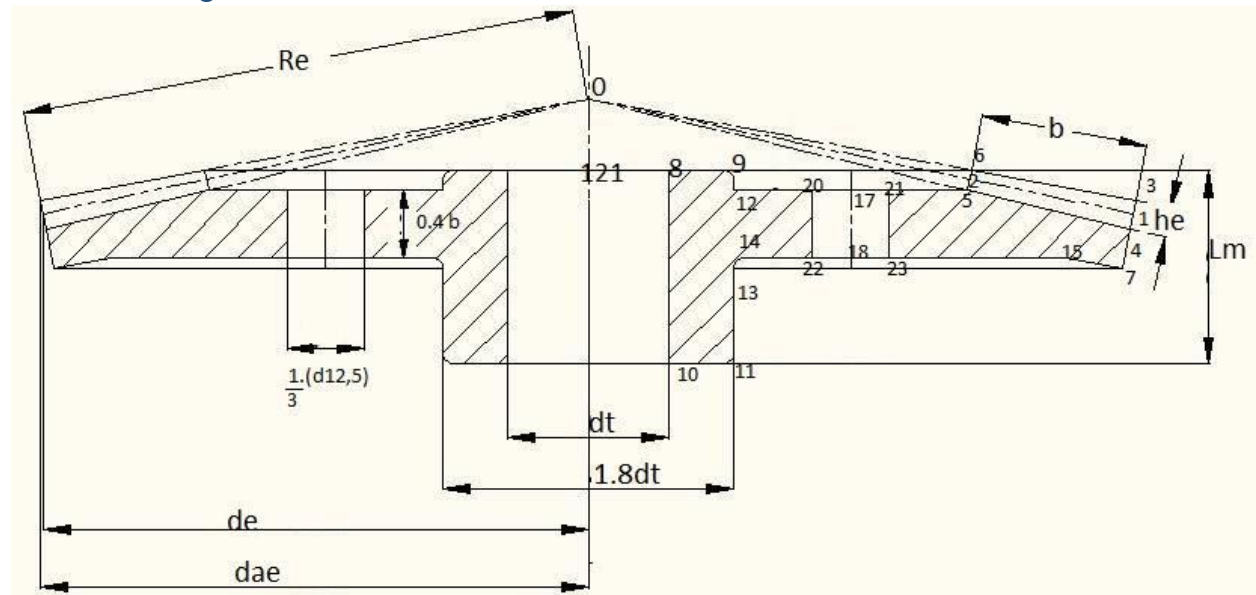


Hình 10 : Chi tiết ổ đũa côn

Cách tiến hành vẽ :

- Lưu các biến hệ thống
- Đặt lại các biến số hệ thống để tránh sai sót
- Đánh số các điểm (hình 10: Nằm phần gạch là “t\_số, phần không gạch ký hiệu như trên)
- Vẽ đường tâm tr0-tr1
- Vẽ lần lượt các đường bao bằng pline (t2-tr3-t4-t5-p5-t10- t11;t6-t7-t8-t9) và fillet với bán kính R(t2-t3-t4;t9-t6-t7;t6-t7-t8); các đường line (t0-t8;p6p-p3p;p1-p5;p2-p4) đưa các đường này vào nhóm chọn ss,ss1(để lấy đối xứng).
- Lấy đối xứng nhóm chọn qua t0-t1

2.4.3 Bánh răng côn

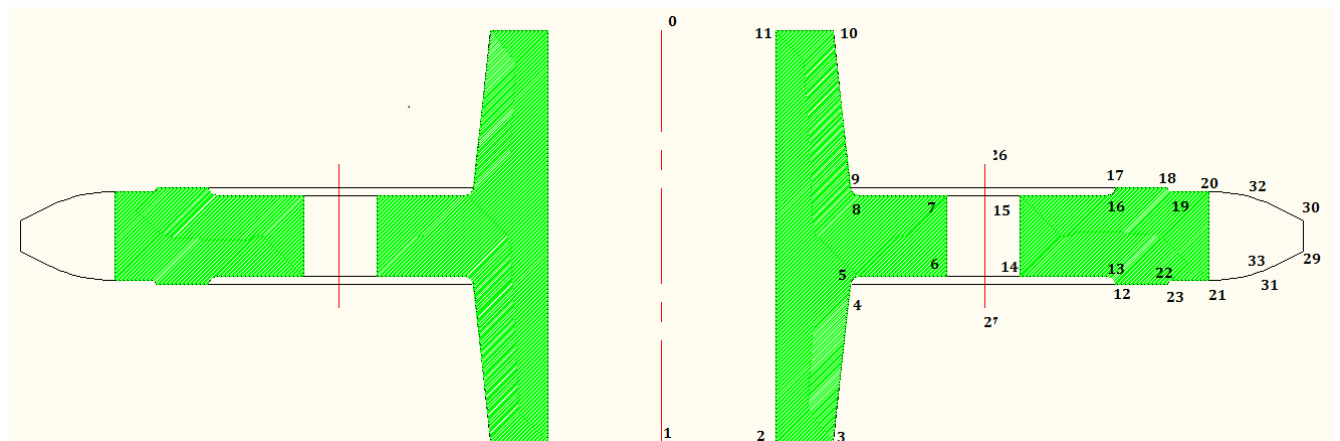


Hình 11 : Chi tiết bánh răng côn

Cách tiến hành vẽ :

- Lưu các biến hệ thống
- Đặt lại các biến số hệ thống để tránh sai sót
- Đánh số các điểm (hình 11: cách đánh số là "brc\_số, trên phần then là th\_số)
- Vẽ đường tâm để lấy đối xứng brc0-brc121, các đường tâm khác brc0-brc5, brc0-brc2, brc0-brc6.
- Vẽ lần lượt các đường bao bằng pline(brc8-brc9-brc12-brc20-brc22-brc14-brc13-brc11-brc10; brc21-brc23-brc15-brc7-brc4); đưa các đường này vào nhóm chọn ss, ss1 (để lấy đối xứng).
- Lấy đối xứng nhóm chọn qua brc0-brc121

#### 2.4.4 Đĩa xích



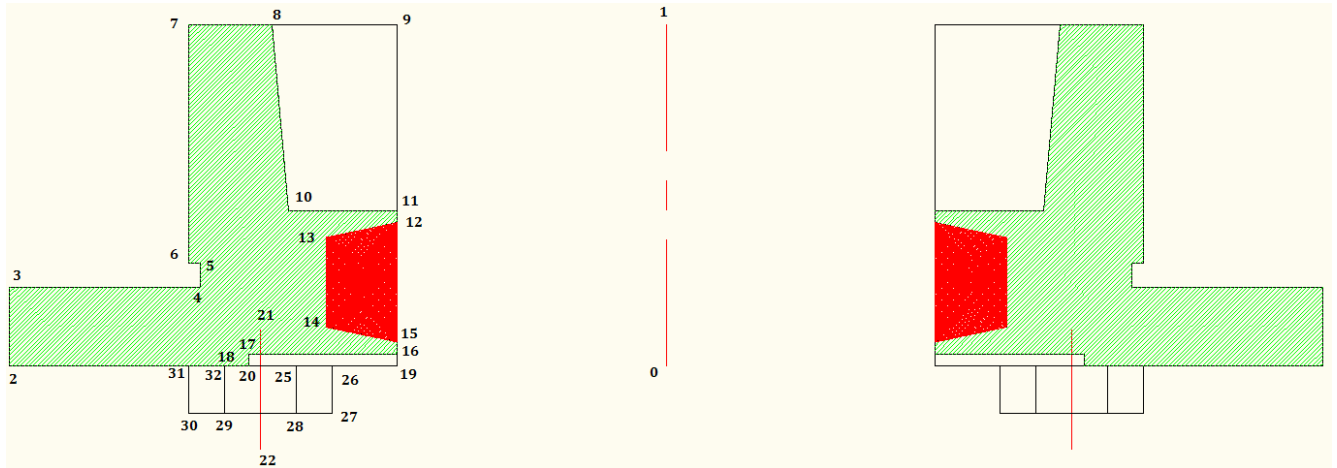
Hình 12 : Chi tiết đĩa xích

Cách tiến hành vẽ :

- Lưu các biến hệ thống
- Đặt lại các biến số hệ thống để tránh sai sót
- Đánh số các điểm (hình 12: các điểm đánh số theo kiểu *pt\_số*)
- Vẽ đường tâm pt0-pt1
- Vẽ lần lượt các đường bao bằng pline(pt2-pt3-pt4-pt5-pt6-pt7-pt8-pt9-pt10-pt11; pt12-pt13-pt14-pt15-pt16-pt17-pt18-pt19-pt20-pt21-pt22-pt23; pt21-pt31-pt29-pt30-pt32-pt20; pt9-pt17; pt4-pt12) và các fillet (pt4-pt5-pt6; pt7-pt8-pt9; pt15-pt16-pt17; pt18-pt19-pt20; pt21-pt22-pt23; pt21-pt31-pt29-pt30-pt32-pt20); đưa các đường này vào nhóm chọn ss, ss1, ss2 (để lấy đối xứng và gach vật liệu).

- Lấy đối xứng nhóm chọn qua pt0-pt1

2.4.5 Nắp ổ 1 (ổ thông)

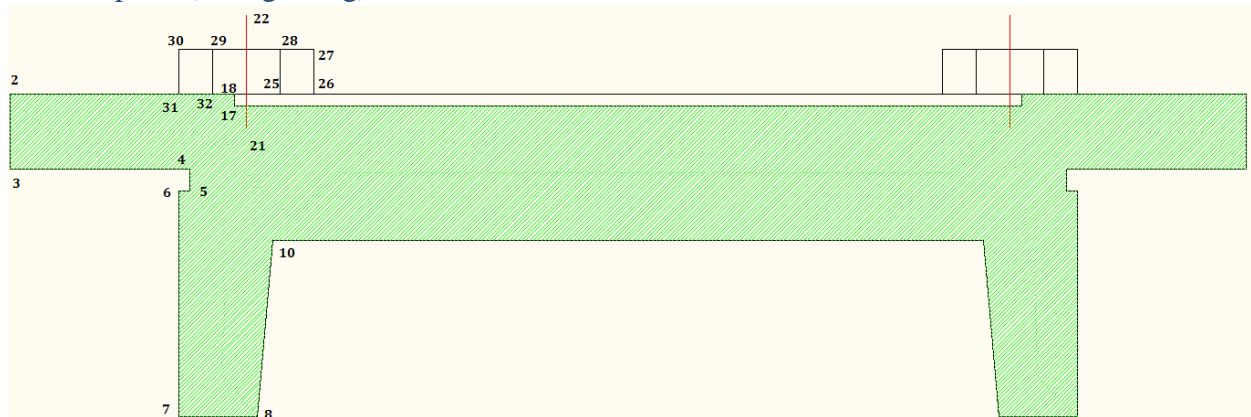


Hình 13 : Chi tiết nắp ổ thông

Cách tiến hành vẽ :

- Lưu các biến hệ thống
- Đặt lại các biến số hệ thống để tránh sai sót
- Đánh số các điểm (hình 13: các điểm đánh số theo kiểu *pt\_số*)
- Vẽ đường tâm pt00-pt01
- Vẽ lần lượt các đường bao bằng pline(pt19-pt16-pt17-pt18-pt2-pt3-pt04-pt05-pt06-pt07-pt08-10-pt11-pt12-pt13-pt14-pt15; pt18-pt19;pt25-pt26-pt27-pt28;pt29-pt30-pt31-pt32); đưa các đường này vào nhóm chọn ss,ss1,ss2(để lấy đối xứng và gạch vật liệu).
- Lấy đối xứng nhóm chọn qua pt00-pt01

2.4.6 Nắp ổ 2 (không thông)

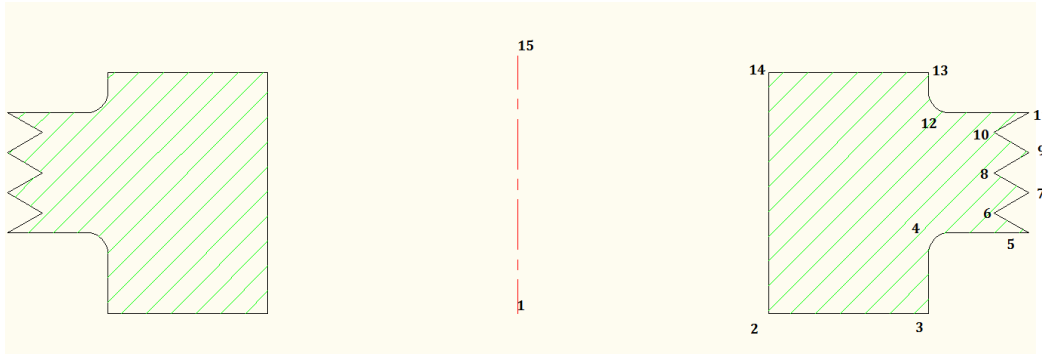


Hình 14 : Chi tiết nắp ổ không thông

Cách tiến hành vẽ :

- Lưu các biến hệ thống
- Đặt lại các biến số hệ thống để tránh sai sót
- Đánh số các điểm (hình 14: các điểm đánh số theo kiểu *pc\_số*)
- Vẽ đường tâm pc00-pc01
- Vẽ lần lượt các đường bao bằng pline (pc19-pc16-pc17-pc18-pc2-pc3-pc04-pc05-pc06-pc07-pc08-pc10-pc11-pc12-pc13-pc14-pc15; pc18-pc19;pc25-pc26-pc27-pc28;pc29-pc30-pc31-pc32); đưa các đường này vào nhóm chọn ss,ss1,ss2(để lấy đối xứng và gạch vật liệu).
- Lấy đối xứng nhóm chọn qua pc00-pc01

#### 2.4.7 Vòng chắn dầu

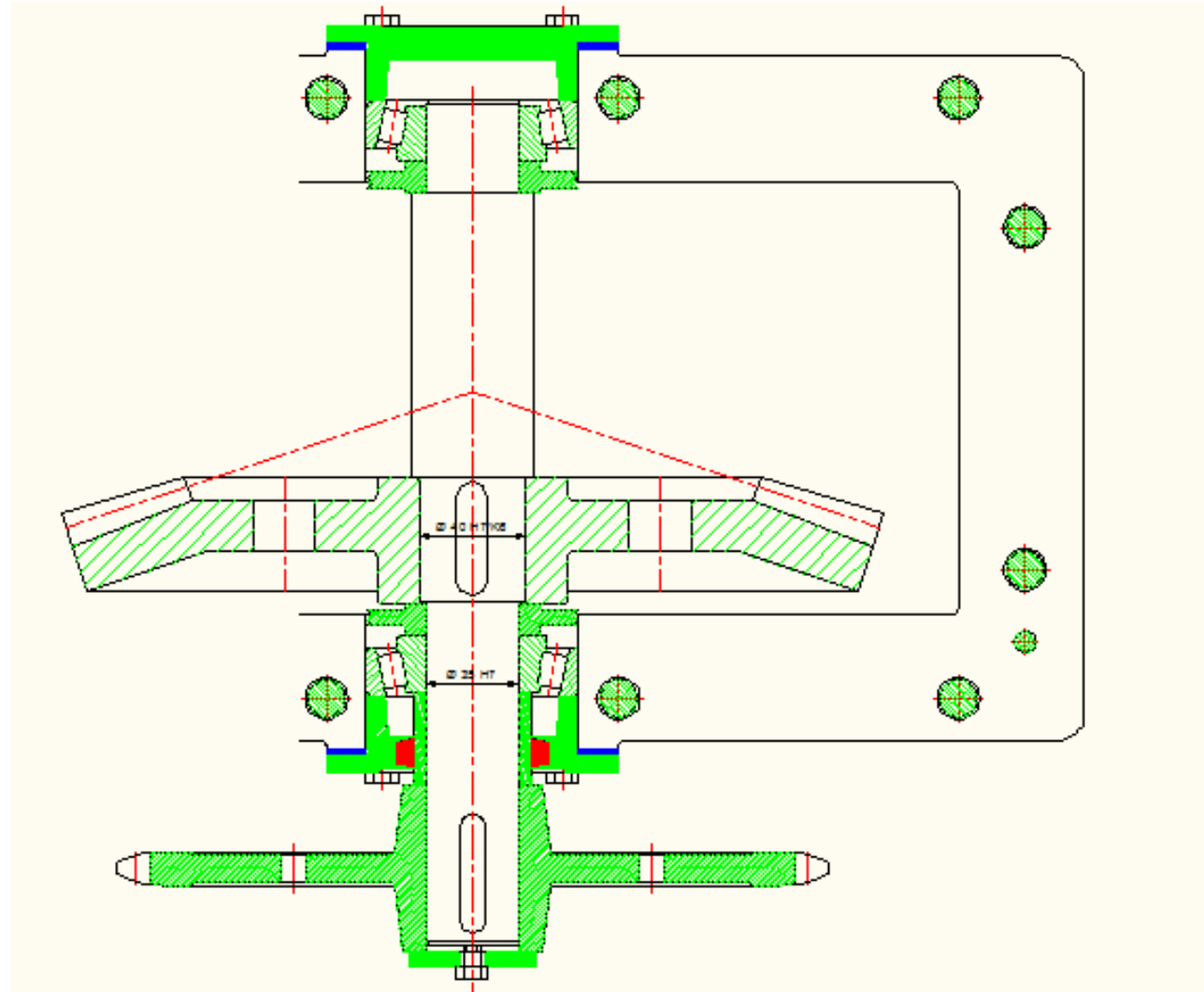


Hình 15 : Chi tiết vòng chắn dầu

Cách tiến hành vẽ :

- Lưu các biến hệ thống
- Đặt lại các biến số hệ thống để tránh sai sót
- Đánh số các điểm (hình 15: các điểm đánh số theo kiểu *uc\_số*)
- Vẽ đường tâm c1-c15
- Vẽ lần lượt các đường bao bằng pline(c2-c3-c4-c5-c6-c7-c8-c9-c10-c11-c12-c13-c14) và fillet bán kính R (c3-c4-c5;c11-c12-c13); đưa các đường này vào nhóm chọn ss (để lấy đối xứng và gạch vật liệu).
- Lấy đối xứng nhóm chọn qua c1-c15

## 2.4.8 Hoàn thiện



Hình 14 : Hoàn thành cụm trục ra hộp giảm tốc bánh răng côn một cấp

### Nguồn tham khảo

Bài giảng và ví dụ minh hoạ Autolisp – Thầy Trịnh Đồng Tính

Tự động hoá thiết kế Cơ khí – PGS Trịnh Chất, TS Trịnh Đồng Tính

Dung sai lắp ghép và kỹ thuật đo lường – PGS Ninh Đức Tôn, GVC Nguyễn Thị Xuân Bảy

Chi tiết máy – Nguyễn Trọng Hiệp

Tính toán hệ dẫn động cơ khí – PGS Trịnh Chất, TS Lê Văn Uyển

Nguồn tài liệu về tiêu chuẩn của các chi tiết và hướng dẫn sử dụng Autolisp trên Internet