Proceedings of the World Congress on Engineering 2008 Vol II
WCE 2008, July 2 - 4, 2008, London, U.K.

Algorithm to Describe the Ideal Spur Gear Profile

O.Reyes, A.Rebolledo, G.Sanchez

***Abstract*—In this paper an algorithm to describe the ideal spur gear profile is proposed. More precisely, the goal is to describe the point to point movement to be used within a CNC machine. Three parameters are the required algorithm input data: the modulus, the number of teeth and the pressure angle. The algorithm is based upon the equations of the circular head and root thickness. The involute of the base circle is used to draw the tooth. The algorithm can be translated into any machine language. However, two codes are proposed in order to test it: A Visual Basic code (which can run as a macro in Excel) and a CNC Mitsubishi G code. Several tips for machining are finally given.**

***Index Terms*— Algorithm, spur gear profile, involute, modulus, CNC.**

I. Introduction

Any gear moves (or is moved by) other gear in order to transmit forces between different parts in a mechanism. Gears are manufactured by several methods. For example, a formed milling cutter is commonly used [1], [5]. For any deign project, it is always important to care about teeth geometry, in order to achieve minimal sliding. An ideal geometry is required to achieve circumferential contact. However, manufacturers often consider just rough approximations, resulting in friction and wear effects [10], [11].

In general, spur gear design involves three parameters: the modulus or diametral pitch, the pressure angle and the number of teeth. The most accepted profile is the involute of the base circle because, in theory, there is no undesirable sliding. Moreover, the uniform contact reduces mechanical losses.

Spur gear equations are well known. They can be found in many catalogs and manufacture books. See for example the references [2], [4] , [6], [9] and [12]. The present paper proposes an algorithm to describe the point to point movement to be used within a CNC machine: a subject which, to the best of our knowledge, has not been discussed in details in the open literature.

A general algorithm is given in order the interested reader could translate it to any programming language (ie. c/c++, java, etc.). Two codes are also given to test the algorithm: a Visual Basic code (which can run as a macro in Excel) and a CNC Mitsubishi G code. Several tips for machining and conclusions are finally given.

II. Nomenclature

θ: Actual angle.
r : Radius.
r_b : Base radius.
ψ : Pressure angle
r_e : External radius.
r_p : Pitch (or primitive) radius.
Ec: Circular thickness.
γ : Circular thickness angle.
m: Gear modulus
Pc : Picht circle.
Bc: Base circle.
Cl: Contact line
Ht_e : Circular head thickness.
ξ : Head angle.
Rc: Root circle.
r_{rc}: Root circle radius
σ : Mirrored curve rotation angle.
Z: Number of teeth
C_{rt} : Circular root thickness.

III. Profile over BC

A general procedure to draw the involute is described below.

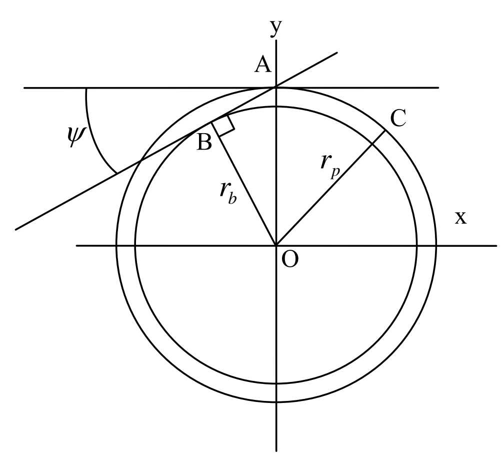

Fig. 1: drawing the base circle

A pitch circle centered at the coordinates origin is drawn first. Second, the contact line given by the pressure angle is drawn (see fig 1). An horizontal line parallel to the x axis is drawn tangent to Pc, obtaining point A. With the aid of ψ sketch Cl and the orthogonal line to Cl passing through the origin to obtain r_b. *Pc* may be regarded as a virtual contact circle to transmit the force to a pinion. In fact, *Pc* is previously calculated by the designer, depending on the torque and/or power to be transferred between mating gears. *Bc* is used then to sketch a curve portion of the tooth profile by means of *Bc*. Remember an involute is a curve traced by a point on a taut cord unwinding from it. The involute forms a spiral. In theory, this shape guarantees no sliding, because it follows the tangency along *Cl*. But the precision on the center between gears, the material elasticity and other effects do not allow to achieve that goal. The choice of the involute to describe the portion of contact allows however to avoid friction [13], [14] as much as possible.

Proceed to draw the involute once you have *Bc*. In vector form the curve is described by the following equations according to fig 2:

$$\begin{bmatrix} x \\ y \end{bmatrix} = r_b \begin{bmatrix} Cos(\theta) & Sin(\theta) \\ Sin(\theta) & -Cos(\theta) \end{bmatrix} \begin{bmatrix} 1 \\ \theta \end{bmatrix} \tag{1}$$

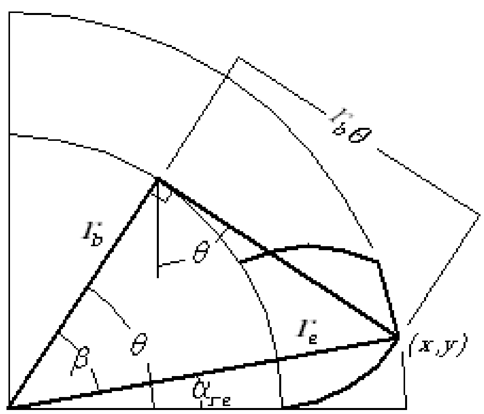

Fig. 2: A tooth portion

Next (1) is transformed in a polar way. Recall that:

$$x^2 + y^2 = r^2 \tag{2}$$

Putting (1) into (2) we get:

$$r^2 = r_b^2\left(1+\theta^2\right) \tag{3}$$

For a general radius is easy to get the angle and vice versa.

$$\theta = \sqrt{\frac{r^2 - r_b^2}{r_b^2}} \tag{4}$$

With the aid of fig. 2, we get the following relations, (5) ,(6) and (7):

$$tg(\beta) = \theta \tag{5}$$

$$\theta_{re} - \beta_{re} = \alpha_{re} \tag{6}$$

$$\alpha_{re} = tg(\beta_{re}) - \beta_{re} = \theta_{re} - atg(\theta_{re}) \tag{7}$$

The connection point between the involute and the external radius provides:

$$\theta_{re} = \sqrt{\frac{r_e^2 - r_b^2}{r_b^2}} \tag{8}$$

In (8) θ_{re} is the angle where the description of the involute stops, see fig. 2. Put (8) into (7) and the angle that contain the involute descriptions appears

$$\alpha_{re} = \theta_{re} - atg\left(\theta_{re}\right) \tag{9}$$

An arc concentric to *Bc* continues the profile. It stars at $\theta = \theta_{re}$ over the involute or starts at α_{re} over the external circle and finishes in $\alpha_{re} + \xi$, see fig. 3. Profile generation follows its way at the top of the tooth and the actual point would be controlled by the Boolean operation to stop the head arc:

$$\left(\alpha_{re} + \xi\right) < a\tan\left(\frac{y}{x}\right) \tag{10}$$

where

$$\begin{bmatrix} x \\ y \end{bmatrix} = r_e \begin{bmatrix} Cos(\alpha_{re} + step) \\ Sin(\alpha_{re} + step) \end{bmatrix} \tag{10.1}$$

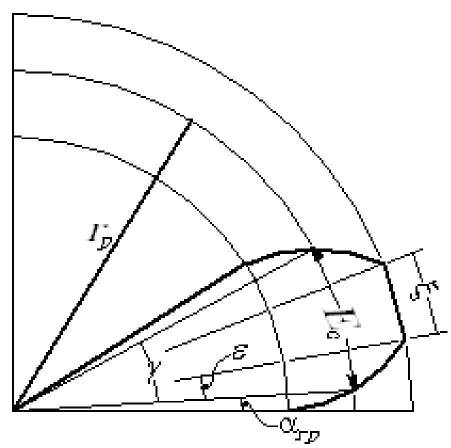

Fig. 3: Head thickness

As in (9), the angle that endows the beginning and the end of the involute can be expressed as:

$$\alpha_{rp} = \sqrt{\frac{r_p^2 - r_b^2}{r_b^2}} - atg\left(\sqrt{\frac{r_p^2 - r_b^2}{r_b^2}}\right) \tag{11}$$

Ec: is a well known data trough the gear module

$$Ec = \frac{\pi}{2} m = r_p \gamma \tag{12}$$

$$\gamma = \frac{\pi m}{2 r_p} \tag{13}$$

From fig. 3 the following relations are given (14), (15), (16)

$$\varepsilon = \alpha_{re} - \alpha_{rp} \tag{14}$$

$$\xi = \gamma - 2\varepsilon \tag{15}$$

Finally circular head thickness appears

$$Ht = \xi r_e \tag{16}$$

Now the mirrored involute is

$$\begin{aligned} x &= r_b \, Cos(\theta) + r_b \, \theta \, Sin(\theta) \\ y &= -r_b \, Sin(\theta) + r_b \, \theta \, Cos(\theta) \end{aligned} \tag{17}$$

But this curve must be rotated counterclockwise using the rotation matrix by the following angle, see fig. 4.

$$\sigma = \xi + 2\alpha_{re} \tag{18}$$

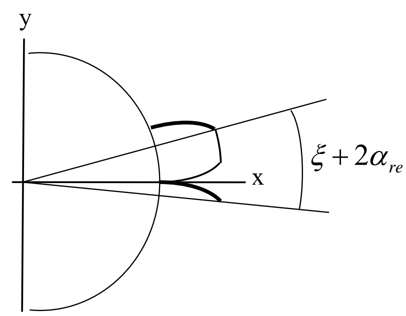

Fig. 4: Mirror curve by σ rotation angle

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = r_b \begin{bmatrix} Cos(\theta) & Sin(\theta) \\ -Sin(\theta) & +Cos(\theta) \end{bmatrix} \begin{bmatrix} 1 \\ \theta \end{bmatrix} \tag{19.0}$$

$$\begin{bmatrix} x \\ y \end{bmatrix} = r_b \begin{bmatrix} Cos(\sigma) & -Sin(\sigma) \\ Sin(\sigma) & +Cos(\sigma) \end{bmatrix} \begin{bmatrix} x' \\ y' \end{bmatrix} \tag{19.1}$$

This curve (19) starts from $\theta = \theta_{re}$ and decreases to $\theta = 0$.

IV. Complete profile

The tooth profile starts in Rc at the point p_1 of fig. 5.

$x = r_{rc}$; $y = 0$;

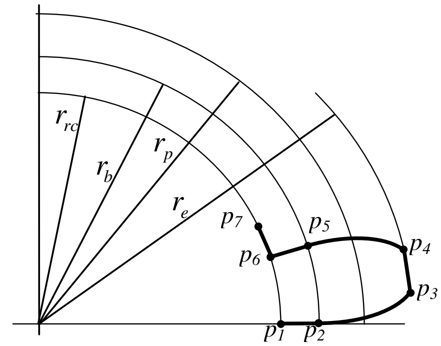

Fig. 5: Tooth profile

p_1 and p_2 are connected by a straight line. Coordinates of point p_2 are:

$x = r_b$; $y = 0$;

p_2 and p_3 are connected by the involute of the base circle. Using (1) coordinates of point p_3 are

$$\begin{aligned} x &= r_b \, Cos(\theta_{re}) + r_b \, \theta_{re} \, Sin(\theta_{re}) \\ y &= r_b \, Sin(\theta_{re}) - r_b \, \theta_{re} Cos(\theta_{re}) \end{aligned}$$

Where (8) give θ_{re}, p_3 and p_4 are connected by the arc of the external radius, coordinates of p_4 are:

$$\begin{aligned} x &= r_e \, Cos(\xi + \alpha_{re}) \\ y &= r_e \, Sin(\xi + \alpha_{re}) \end{aligned}$$

p_4 and p_5 are connected by the mirrored involute, coordinates of p_5 are:

$$\begin{aligned} x &= r_b \, Cos(\sigma) \\ y &= r_b \, Sin(\sigma) \end{aligned}$$

p_5 and p_6 are connected by a straight line. Coordinates of point p_6 are:

$$\begin{aligned} x &= r_r \, Cos(\sigma) \\ y &= r_r \, Sin(\sigma) \end{aligned}$$

σ is the angle that contains the tooth without root, the total amount of use depends of Z. The total pattern to be repeated is defined by the profile joining points p_i, i = 1,2,..,7. Define then τ as the sector angle of the pattern

$$\tau = \frac{360}{Z_0} \tag{20}$$

It is clear that the number of teeth is equal to the number of roots. Thus, p_6 and p_7 are connected by the arc of the root circle. Coordinates of point p_7 are:

ISBN:978-988-17012-3-7

$$x = r_r\, Cos(\tau)$$
$$y = r_r\, Sin(\tau)$$

root angle will be

$$\psi = \tau - \sigma \tag{21}$$

and the circular root thickness

$$C_{rt} = \psi\, r_{rc} \tag{22}$$

V. ALGORITMH BY SECTORS

Sector A: p_1 — p_2

step ← x_0;
$x \leftarrow r_{rc}$;
$y \leftarrow 0$;
do
    $x \leftarrow r_{rc} + s_0$;
    $y \leftarrow 0$;
while (x < r_b);

Sector B: p_2 — p_3

step ← θ_0;
$\theta \leftarrow 0$;
$x \leftarrow r_b$;
$y = 0$;
$\theta_{re} \leftarrow \sqrt{\dfrac{r_e^2 - r_b^2}{r_b^2}}$
do
    $x \leftarrow r_b\, Cos(\theta) + r_b\, \theta\, Sin(\theta)$
    $y \leftarrow r_b\, Sin(\theta) - r_b\, \theta\, Cos(\theta)$
    $\theta \leftarrow \theta + \theta_0$;
while ($a\tan\left(\dfrac{x}{y}\right) < \theta_{re}$)

Sector C: p_3 — p_4

step = θ_0;
$\theta \leftarrow \alpha_{re}$;
$x \leftarrow r_b\, Cos(\theta) + r_b\, \theta\, Sin(\theta)$
$y \leftarrow r_b\, Sin(\theta) - r_b\, \theta\, Cos(\theta)$
do
    $x \leftarrow r_e\, Cos(\theta)$
    $y \leftarrow r_e\, Sin(\theta)$
    $\theta \leftarrow \theta + \theta_0$;
while ($\theta < (\xi + \alpha_{re})$)

Sector D: p_4 — p_5

step ← θ_0;
$\theta \leftarrow \alpha_{re}$;
$\sigma \leftarrow \xi + 2\alpha_{re}$;
$x \leftarrow r_e\, Cos(\xi + \alpha_{re})$
$y \leftarrow r_e\, Sin(\xi + \alpha_{re})$
do
    $x \leftarrow (r_b\, Cos(\theta) + r_b\, \theta\, Sin(\theta))\cos(\sigma)$
    $\quad - (-r_b\, Sin(\theta) + r_b\, \theta\, Cos(\theta))\sin(\sigma)$
    $y \leftarrow (r_b\, Cos(\theta) + r_b\, \theta\, Sin(\theta))\sin(\sigma)$
    $\quad + (-r_b\, Sin(\theta) + r_b\, \theta\, Cos(\theta))\cos(\sigma)$
    $\theta \leftarrow \theta - \theta_0$;
while ($\theta > 0$)

Sector E: p_5 — p_6

step ← x_0;
$x \leftarrow r_b\, Cos(\sigma)$
$y \leftarrow r_b\, Sin(\sigma)$
do
    $x \leftarrow x - x_0$;
    $y \leftarrow x \tan(\sigma)$;
    $r \leftarrow \sqrt{x^2 + y^2}$
while (r > r_{rc});

Sector F: p_6 — p_7

step ← θ_0;
$\theta \leftarrow \sigma$;
$x \leftarrow r_r\, Cos(\sigma)$
$y \leftarrow r_r\, Sin(\sigma)$
do
    $x \leftarrow r_r\, Cos(\theta)$
    $y \leftarrow r_r\, Sin(\theta)$
    $\theta \leftarrow \theta + \theta_0$;
while ($\theta < (\tau)$);

VI. ASSEMBLY SECTORS

Until now, just one tooth has been described. Remember our goal is to describe the point to point movement to be used in a CNC machine. The comment now could be: "but a gear has Z_0 teeth". Rotation matrices are useful for repeating circular patterns. The proposed vba code is given in appendix A. It fills an Excel sheet of x and y points (see Table 1).

VII. MITSUBISHI G CODE

Three simple sectors are given in order to provide the reader with useful guidelines to implement the code in a cnc machine.

Sector A: $p_1 \rightarrow p_2$

#3 = 0.01;
#1 = r_{rc};
#2 = r_b;
#4 = 0;
WHILE [#1LT#2] DO1
G01 X#1 Y#4;
#1 = #1 + #3;
END DO1

Sector B: $p_2 \rightarrow p_3$

#3 = 0.01;
#5 = 0;
#1 = r_b;
#4 = 0.0001;
#6 = r_e;
#7 = SQRT[[#6*#6-#1*#1]/[#1*#1]];
#8 = #1;
WHILE [ATAN[#1/#4]LT#7] DO2
#1= #8*COS[#5]+#8*#3*SIN[#5];
#4 = #8*SIN[#5]-#8*#3*COS[#5];
G01 X#1 Y#4;
#5 = #5 + #3;
END DO2

Sector C: $p_3 \rightarrow p_4$

#3 = 0.01;
#5 = α_{re};
#6 = r_e;
#8 = r_b;
#1= #8*COS[#5]+#8*#3*SIN[#5];
#4 = #8*SIN[#5]-#8*#3*COS[#5];
#7 = $\xi + \alpha_{re}$;
WHILE [#5LT#7] DO3
G01 X#1 Y#4;
#1 = #6*COS[#5];
#4 = #6*SIN[#5];
#5 = #5 + #3;
END DO3

These subroutines must be saved with numbers. If the program has the number 17, modulus = 0.3, number of teeth = 10, pressure angle = 14.4, then invoke it as follows:

G65 P17 M0.3 Z10 P14.4;
Where the variables are: M #13; P #16; Z #26

Programmer should be careful about their names and meanings.

	A	B	C	D	E	F
1	modulus	m	0,300		pi	3,142
2	number of teeth	z	10,000			
3	pressure angle	alfa	14,400			
4						
5					Radius	
6	Primitive diameter	Dp	0,700		Primitive	0,350
7	External Diameter	De	0,900		External	0,450
8	Base diameter	Db	0,678		Base	0,339
9	Circular Pitch	Pc	0,314		Root	0,233
10	Addendum	Ac	0,100			
11	Circular thickness	Ec	0,157			
12	Clearance	J	0,017	0,166		
13	Tooth height	At	0,217			
14	Root diameter	Dr	0,467			

Table I: excel sheet for vba code

VIII. CNC MANUFACTURING

Consider someone is manufacturing the spur gear in a CNC vertical milling machine. The clue is putting a plate in the worktable and making a slot with our particular profile. Axis z of the tool must vary to penetrate the plate. Next, turn round the plate and planning surface will let the desired gear drop. Root radius is done selecting properly tool diameter. While machining is occurring one important parameter is the feed: it could be controlled modifying the steps for each sector. Moving point to point with predefined small steps allows describing the curve with a good approximation.

IX. CONCLUSION

Using the proposed algorithm allows to control the gear shape. Moreover user parameters can be used (i.- modulus, number of teeth and pressure angle, ii.- flexibility). Although many commercial packages generate profiles similar to this paper and send the data to CNC, the user has no control about the generated shape, since he does not have access to the code that generates it. In our case, no software is needed. Just write the code in CNC and it is done. Profiles more complicated could be generated joining small lines using the machine precision. Machining level curves aid to achieve surfaces. Authors recommend simulation before CNC.

Spur gears are usually parametric models that CAD systems build using curves approximations. Those models are commercial but commercial issues should change in favor of real model implementation, thus reducing friction in box gearing and getting more power transmission. Researchers should intensively work in this field.

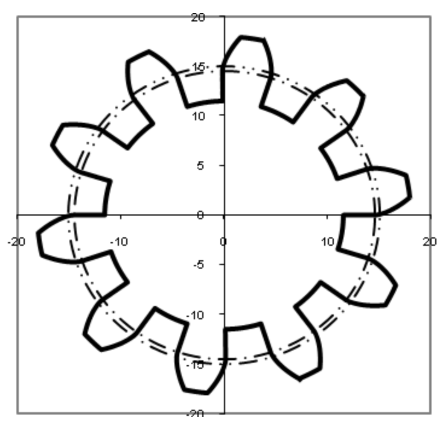

Fig 6: Profile generated
(Parameters used are in table 1)

APPENDIX A

VBA CODE TO DRAW SPUR GEAR

```
Private Sub DrawGear_Click()
Range("f20:i40000").Select
Selection.ClearContents
Range("j1:j40000").Select
Selection.ClearContents
Range("d4").Select
Dim pi, m, rp, gamma, rrc, rb, re, rr, are, tre, arp, epsilon, se,
sigma, Zo, tau, k, continue, taurot
```

```
pi = Range("F1")
m = Range("C1")
rp = Range("F6")
gamma = pi * m / (2 * rp)
rrc = Range("F9")
rb = Range("F8")
re = Range("F7")
rr = Range("F9")
are = (Sqr((re ^ 2 - rb ^ 2) / (rb ^ 2)) _
- Atn(Sqr((re ^ 2 - rb ^ 2) / (rb ^ 2))))
tre = Sqr((re ^ 2 - rb ^ 2) / (rb ^ 2))
arp = Sqr((rp ^ 2 - rb ^ 2) / (rb ^ 2)) _
- Atn(Sqr((rp ^ 2 - rb ^ 2) / (rb ^ 2)))
epsilon = are - arp
se = (gamma - 2 * epsilon)
sigma = se + 2 * are
Zo = Range("C2")
tau = 360 / Zo
taurot = 0
k = 20
Dim x, xi, y, yi, r, teta, follow
continue = True
Do While continue
   teta = 0
'========= Sector A ========='
      Dim stepA
      stepA = 0.1
      x = rrc
      y = 0
      follow = True
      Do While follow
         Cells(k, 7) = rotationx(taurot, x, y)
         Cells(k, 8) = rotationy(taurot, x, y)
         x = x + stepA
         k = k + 1
         If x > rb Then
            follow = False
         End If
      Loop
'======== Sector B ========='
      Dim stepB
      stepB = 0.1
      follow = True
      Do While follow
         x = rb * Cos(teta * pi / 180) + rb * teta * pi / 180 *
Sin(teta * pi / 180)
         y = rb * Sin(teta * pi / 180) - rb * teta * pi / 180 *
Cos(teta * pi / 180)
         Cells(k, 7) = rotationx(taurot, x, y)
         Cells(k, 8) = rotationy(taurot, x, y)
         teta = teta + stepB
          k = k + 1
         If Atn(y / x) > are Then
            follow = False
         End If
      Loop
'========= Sector C ========='
    teta = are * 180 / pi
      Dim stepC
      stepC = 0.1
      follow = True
      Do While follow
         x = re * Cos(teta * pi / 180)
         y = re * Sin(teta * pi / 180)
         Cells(k, 7) = rotationx(taurot, x, y)
         Cells(k, 8) = rotationy(taurot, x, y)
         teta = teta + stepC
         k = k + 1
         If teta > (are + se) * 180 / pi Then
            follow = False
         End If
     Loop
'========= Sector D ========='
   teta = tre * 180 / pi
   Dim stepD
    stepD = 0.1
    follow = True
    Do While follow
       xi = rb * Cos(teta * pi / 180) + rb * teta * pi / 180 *
Sin(teta * pi / 180)
       yi = -(rb * Sin(teta * pi / 180) - rb * teta * pi / 180 *
Cos(teta * pi / 180))
       x = rotationx(sigma * 180 / pi, xi, yi)
       y = rotationy(sigma * 180 / pi, xi, yi)
       Cells(k, 7) = rotationx(taurot, x, y)
       Cells(k, 8) = rotationy(taurot, x, y)
       teta = teta - stepD
       k = k + 1
       If teta < 0 Then
          follow = False
       End If
    Loop
'========= Sector E ========='
    stepE = 0.1
    follow = True
    Do While follow
       Cells(k, 7) = rotationx(taurot, x, y)
       Cells(k, 8) = rotationy(taurot, x, y)
       x = x - stepE
       y = x * Tan(sigma)
       k = k + 1
       r = Sqr(x ^ 2 + y ^ 2)
       If r < rrc Then
          follow = False
       End If
    Loop
'========= Sector F ========='
    stepF = 0.1
    teta = sigma * 180 / pi
    follow = True
    Do While follow
       x = (rr) * Cos(teta * pi / 180)
       y = (rr) * Sin(teta * pi / 180)
       Cells(k, 7) = rotationx(taurot, x, y)
       Cells(k, 8) = rotationy(taurot, x, y)
       k = k + 1
       teta = teta + stepF
       If teta > (tau) Then
          follow = False
       End If
    Loop
taurot = taurot + tau
If taurot > tau * (Zo) Then
   continue = False
End If
Cells(k, 9) = taurot
Loop
End Sub
```

APPENDIX B

FUNCTIONS

```
Function rotationx(rotang, x, y)
Dim pi
pi = Range("F1")
rotationx = x * Cos(rotang * pi / 180) - y * Sin(rotang * pi /
180)
End Function

Function rotationy(rotang, x, y)
Dim pi
pi = Range("F1")
rotationy = x * Sin(rotang * pi / 180) + y * Cos(rotang * pi /
180)
   End Function
```

ACKNOWLEDGMENT

We would like to acknowledge S. Scovino and J. López for their contributions to this work, especially for the development of the G code. They currently are at "Departamento de Tecnología Industrial, Universidad Simón Bolívar, Venezuela".

REFERENCES

[1] Phillip F. Ostwald, Jairo Muñoz ,1997, Manufacturing processes and systems, Processes for making gears pp. 524-532, JohnWiley & Sons, New York.
[2] BOSTON GEAR, Gear Catalog, Engineering information, pp. 137 -141. Available: http://www.bostongear.com/
[3] Giesecke, Mitchell, Spencer, Hill, Dygdon, Novak, 2000, Technical Drawing, Gearing and Cams, pp. 620-625, Prentice Hall.
[4] C. H. Jensen, 1999, Dibujo y diseño de ingenieria, Correas, Cadenas y engranajes, pp 409 – 411, Mc Graw Hill.
[5] Serope Kalpakjian, 1992, Gear manufacturing by machining, pp. 741 – 749, Addison-Wesley.
[6] Joseph E. Shigley, Charles R. Mischke, 2002, Engranes: descripción general, pp 840 – 856, Mc Graw Hill.
[7] Mitsubishi, Dynamite 4400 Programer manual, 2000.
[8] John Green, Stephen Bullen, Rob Bovey, Robert Rosenberg, 2001, Excel 2002 VBA, Wrox Press.
[9] Earle Buckingham,1887, Analytical mechanics of gears, Dover.
[10] Jesper Brauer,Sören Andersson, Simulation of wear in gears with flank interference—a mixed FE and analytical approach, Wear, Volume 254, Issue 11, October 2003, pp. 1216-1232.
[11] Jozef Wojnarowski and Valentin Onishchenko, Tooth wear effects on spur gear dynamics, Mechanism and Machine Theory, Volume 38, Issue 2, February 2003, Pages 161-178.
[12] G. Bonari, Static and Dynamic Modelling of Gear Transmission, Ph.D. Thesis, University of Modena and Reggio Emilia, Italy, 2006. (http: //www.vibrazioni.unimo.it/tesipdf/GiorgioBonoriPhDThesis.pdf)
[13] Song He, Sungmin Cho and Rajendra Singh, Prediction of dynamic friction forces in spur gears using alternate sliding friction formulations, Journal of Sound and Vibration Volume 309, Issues 3-5, 22 January 2008, Pages 843-85.
[14] S. He, R. Gunda and R. Singh, Effect of sliding friction on the dynamics of spur gear pair with realistic time-varying stiffness, Journal of Sound and Vibration 301 (3–5) (2007), pp. 927–949.