CIRCULAR ROULARD GENERATOR.

The purpose of this program is to generate roulards on the outside of a circle (epicycloids). The process is extended to allow the piont whose locus is being traced to be within or outside the rolling circle. The goimetry is shown in Fig. 1 below:-

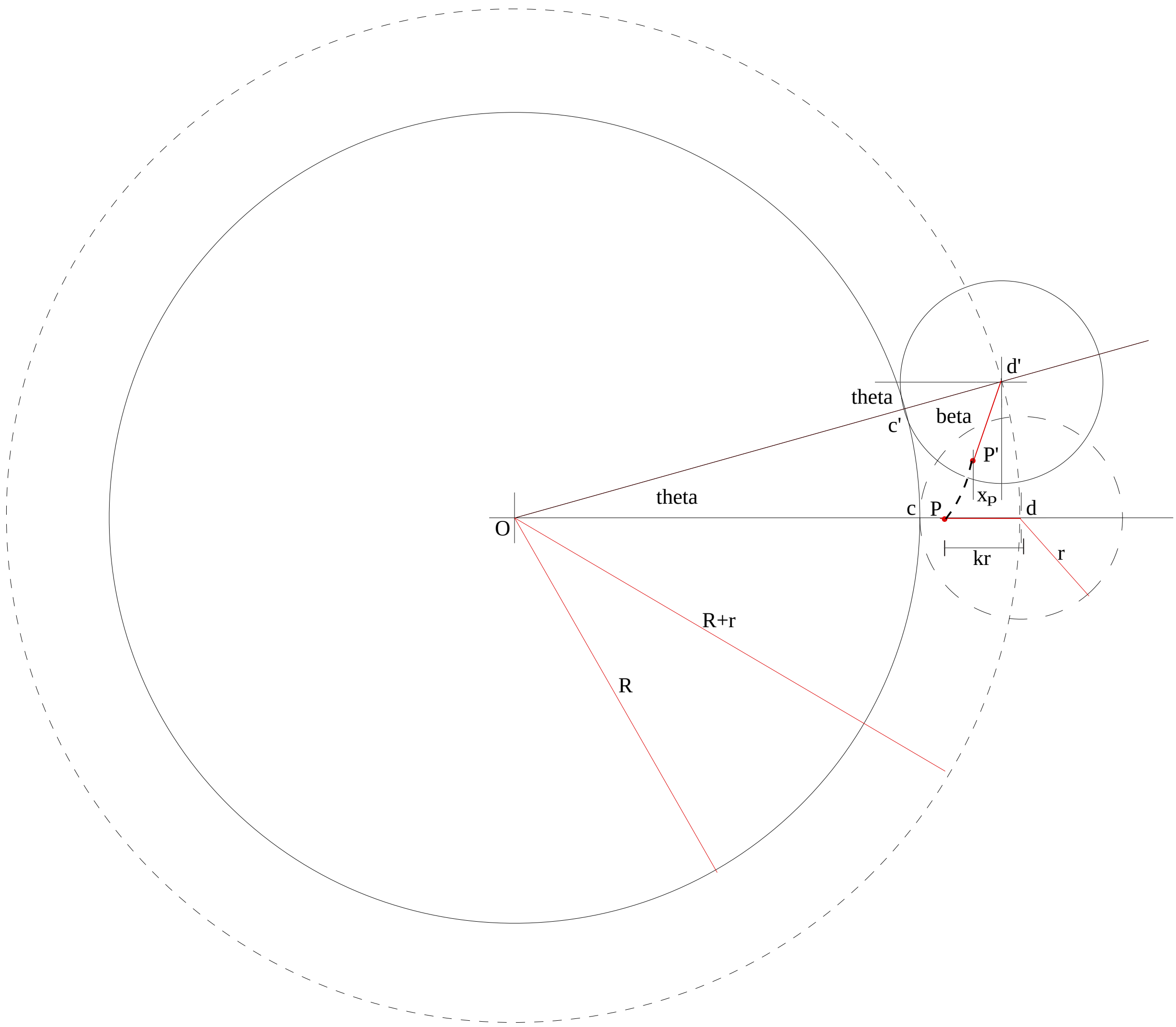

Fig. 1

O is the centre of the Large Circle around which the roulard is generated and of Radius R.
The Small (rolling) Circle has a Radius r and the "Tracing Point" P is set at a distance of kr from the centre of the Small Circle.. It is assumed that initially the point P lies on the line O - A . After the centre of the rolling circle has moved through an angle theta the tracing point has moved to P' .

Co-ordinates of the <u>centre</u> of the rolling circle (point d) [w.r.t. Centre O] are given by:

$$x_d = (R + r) \text{ Cos theta}, \quad y_d = (R + r) \text{ Sin theta} \quad \text{- - - -} \quad (1)$$

Co-ordinates of point P <u>w.r.t. point d</u> are :-

$$x_p = - kr \text{ Cos (beta + theta)} \quad , \quad y_p = - kr \text{ Sin (beta + theta)} \quad \text{- - - -} \quad (2)$$

Since the rolling circle is always in contact with the Large Circle as it rolls, the <u>circumferential</u>

distance a -- a' must be equal to the circumferential distance c -- c' .

Therefore Since theta = c c' / R and beta = a a' / r and c c' = a a'

we get R.theta = r.beta or theta = r.beta/R

[Note: beta will be used as the independant variable.]

This gives:- x_p = k.r.Cos (beta + r.beta / R) = k.r.Cos(1+r/R).beta - - - - - - (3a)

y_p = k.r.Sin (beta + r.beta / R) = k.r.Sin (1 + r/R). beta - - - - - - (3b)

Therefore the co-ordinates of P w.r.t centre O are :-

$X_p = x_a - x_p$ and $Y_p = y_a - y_p$

In order to finish at the starting point, and thus make a closed figure, the circumferences of the two circles need to be in certain ratios. In such a closed figure the small (generating) circle rotates n times while its centre rotates about the large circle m times. [n and m are Integers].
The ratio of n to m must be the same as that of the circumferences and hence of the radii.

n/m = R/r - - - - - (4)

Let a be the Integer Part of R/r . Then circumference of large circle = 2πR
circumference of large circle = 2πr

When the small circle has revolved a times its 'track' length will be 2πra so that it will be 'short' of the starting point on the large circle circumference by 2πR - 2πra = 2π(R - a.r) - - - (5)

When the difference is equal to the 'track length' of one revolution of the small circle (or zero) the Figure will be 'closed'.
When R/r is a integer a = R/r and a.r = R, so that the figure will 'close 'after one 'circuit' of the large circle.
When R/r is not an integer the small circle will have to make m 'circuits' before 'catching up' on the starting point.
On the first 'circuit' it will be 'short' by 2π(R - a.r) , on the next it will be 'shorter' still ie. by 4π(R - a.r) etc.
After m 'circuits' it will be 'short' by a whole revolution of the small circle ie. by 2πr so that -
2mπ(R - a.r) = 2πr ie. m(R - a.r) = r or m = r / (R - a.r) - - - - (6)
This can be written as 1 / (R/r - a) But R/r - a = b the 'fractional' part of R/r
Therefore m = 1 / b [<u>But m must be an Integer</u>]
The small circle then has to make one more revolution to reach the starting point.
No. of revolutions of the small circle for m 'circuits' is- m.a so that the total No. of revolutions is- m.a + 1 .

If 1/b is not a integer but we can multiply it by a integer such that the product is an integer , we can calculate the minimum number of 'circuits' necessary to make a 'closed' figure by multiplying m by the same number.
eg. If R/r were say 8.3 ie. a = 8, b = 0.3 we could multiply 1/b and m by 3 to give 10

'circuits' and get a closed figure.

EXTENSIONS

The purpose of the 'extended' algorithm is to enable the user to design a circular roulard to a wanted style and size, for use in cards etc. and framing photographs.

Refering to the diagram it will be seen that the 'Outer' radius of the 'Ring' will be $R + (k + 1)r$ and the 'Inner' radius $R - (k - 1)r$ [Note that if k is less than 1 the Inner radius will be greater than R].

If 'Outer' radius = R_O and 'Inner' radius = R_I then we can re-arrange the above to give:-

$r = (R_O - R_I) / 2k$ and $R = (R_O + R_I) / 2 - (R_O - R_I) / 2k$ - - - - (7)

This gives:- $R/r = k.(R_O + R_I) / (R_O - R_I) - 1 = a + b$ - - - - (8)

But m must be integer, so we start with the required sizes (R_O and R_I) and a suitable value of k and then calculate how these would change for a range of values of m from 1 to 10 [range could be extended]. The user can then select the best compromise, bearing in mind the changes in size and the total number of loops. If necessary another value of k can be tried. All values of m will give a closed figure but some will give one nearer to the original requires dimentsions than others and. of course, a different number of loops.

First we calculate 'initial' values of R and r from the input dimentions and from these 'initial' values of X [= R/r] , a and b .

For each of the (10) values of m we must calculate the modified values of R_O and R_I that are similar to the above but make m the integer required.

In order to get a balance and make the 'new' ring as neer to the original requested dimensions as possible we multiply the radius of the large circle by a factor (near unity) and divide that of the small circle by the same factor. (p)

$R_N = p.R$ $r_N = r/p$ Therefore $X_N = p^2.X$ or $p^2 = X_N / X$

For each value of m we make m.b an integer by rounding it and then calculate a 'new' value of X (= X_N) and from this p and hence the 'new' (modified) values of the outer and inner radii of the ring.
We also calculate the 'deviation' of the 'new' ring dimensions from the original (entered) values ie. the absolute value of $(X_N / X) - 1$. The user may choose a suitably low value.

All this selection is done, of course, before the figure is drawn.