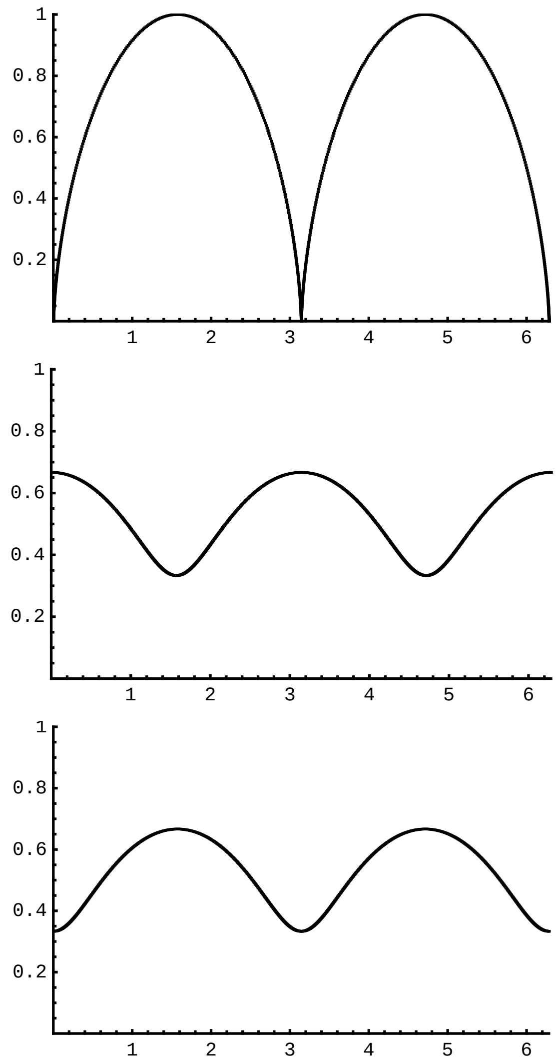
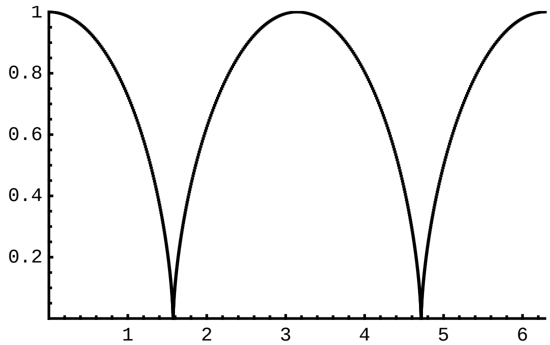
```
itb = Table[t, {t, 720}]; (* inner tip wing b *)
ita = Table[t, {t, 720}]; (* inner tip wing a *)
otb = Table[t, {t, 720}]; (* outer tip wing b *)
ota = Table[t, {t, 720}]; (* outer tip wing a *)

(* variable t is the 'ticking' angular degree *)

For[t = 1, t <= 720, t++,
  z = N[Pi * t / 180];
  itb[[t]] =
   {
    (z / 2) + (1 / 6) * Sin[z],
    (1 / 2) + (1 / 6) * Cos[z]
   };

  ita[[t]] =
   {
    (z / 2) + (1 / 6) * Sin[z + Pi],
    (1 / 2) + (1 / 6) * Cos[z + Pi]
   };

  otb[[t]] =
   {
    (z / 2) + (1 / 2) * Sin[z],
    (1 / 2) + (1 / 2) * Cos[z]
   };

  ota[[t]] =
   {
    (z / 2) + (1 / 2) * Sin[z + Pi],
    (1 / 2) + (1 / 2) * Cos[z + Pi]
   }
 ];

ListPlot[ota, PlotStyle → PointSize[0.001],
  Axes → Automatic, PlotRange → {{0, 2 Pi}, {0, 1}}];
ListPlot[itb, PlotStyle → PointSize[0.001], Axes → Automatic,
  PlotRange → {{0, 2 Pi}, {0, 1}}];
ListPlot[ita, PlotStyle → PointSize[0.001], Axes → Automatic,
  PlotRange → {{0, 2 Pi}, {0, 1}}];
ListPlot[otb, PlotStyle → PointSize[0.001], Axes → Automatic,
  PlotRange → {{0, 2 Pi}, {0, 1}}];

(* Savonius wave tracings *)

graymat =.
height = 100;
width = 720;
graymat = Table[{x, y}, {x, height}, {y, width}];
graymat[[All, All]] = {0, 0, 0};
```

```
(* ota[t] contains 720 x,y pairs, dimensions 720 x 2,
for the points lying on the path of the ota,
so find all points in graymat that match those points and color them red;
however, use regularly spaced x points generated
 by a loop rather than the actual data from above *)

For[t = 1, t <= 720, t++, graymat[[(Floor[99 * ota[[t, 2]] ] + 1), t ]] = {1, 0, 0}];

(* likewise for the other three waves *)

For[t = 1, t <= 720, t++, graymat[[(Floor[99 * otb[[t, 2]] ] + 1), t ]] = {0, 1, 0}];
For[t = 1, t <= 720, t++, graymat[[(Floor[99 * ita[[t, 2]] ] + 1), t ]] = {0, 0, 1}];
For[t = 1, t <= 720, t++, graymat[[(Floor[99 * itb[[t, 2]] ] + 1), t ]] = {1, 1, 0}];

(* Savonius circles centered at y=
 Pi*height of radius r at centers x= h/3 and 2*h/3 *)

rs = (height / 3)^2;  (*radius squared *)
tol = 100;
For[x = 1, x ≤ height, x++,
  For[y = 1, y ≤ width, y++,
   If[ ((-tol < (rs  -  ((y - 360)^2 + (x - 66.6)^2  )) < tol) && y ≥ 360),
    graymat[[x, y]] = {0.5, 0.5, 0.5}] ;
   If[(( -tol < (rs  -  ((y - 360)^2 + (x - 33.3)^2  )) < tol) && y ≤ 360),
    graymat[[x, y]] = {0.5, 0.5, 0.5}]
  ]];

gmat = Graphics[RasterArray[Apply[RGBColor, graymat, {2}]],
   ImageSize → {width, height}, AspectRatio → Automatic];
Show[gmat];



height = 100;
width = 720;
graymat = Table[{x, y}, {x, height}, {y, width}];
graymat[[All, All]] = {0, 0, 0};

(* N[720/(2*Pi)] =114.59155902616465` *)

For[t = 1, t <= 720, t++,
  graymat[[(Floor[99 * ota[[t, 2]] ] + 1), (Floor[114 * ota[[t, 1]] ] + 1) ]] = {1, 0, 0}];
For[t = 1, t <= 720, t++,
  graymat[[(Floor[99 * otb[[t, 2]] ] + 1), (Floor[114 * otb[[t, 1]] ] + 1) ]] = {0, 1, 0}];
For[t = 1, t <= 720, t++,
  graymat[[(Floor[99 * ita[[t, 2]] ] + 1), (Floor[114 * ita[[t, 1]] ] + 1) ]] = {0, 0, 1}];
For[t = 1, t <= 720, t++,
  graymat[[(Floor[99 * itb[[t, 2]] ] + 1), (Floor[114 * itb[[t, 1]] ] + 1) ]] = {1, 1, 0}];

(* Savonius circles centered at y=
 Pi*height of radius r at centers x= h/3 and 2*h/3 *)
```

```
rs = (height / 3)^2;  (*radius squared *)
tol = 100;
For[x = 1, x ≤ height, x++,
  For[y = 1, y ≤ width, y++,
   If[ ((-tol < (rs -  ((y - 360)^2 + (x - 66.6)^2 )) < tol) && y ≥ 360),
    graymat[[x, y]] = {0.5, 0.5, 0.5}] ;
   If[(( -tol < (rs -  ((y - 360)^2 + (x - 33.3)^2 )) < tol) && y ≤ 360),
    graymat[[x, y]] = {0.5, 0.5, 0.5}]
  ]];

gmat = Graphics[RasterArray[Apply[RGBColor, graymat, {2}]],
   ImageSize → {width, height}, AspectRatio → Automatic];
Show[
  gmat];
```




1
0.8
0.6
0.4
0.2
1 2 3 4 5 6