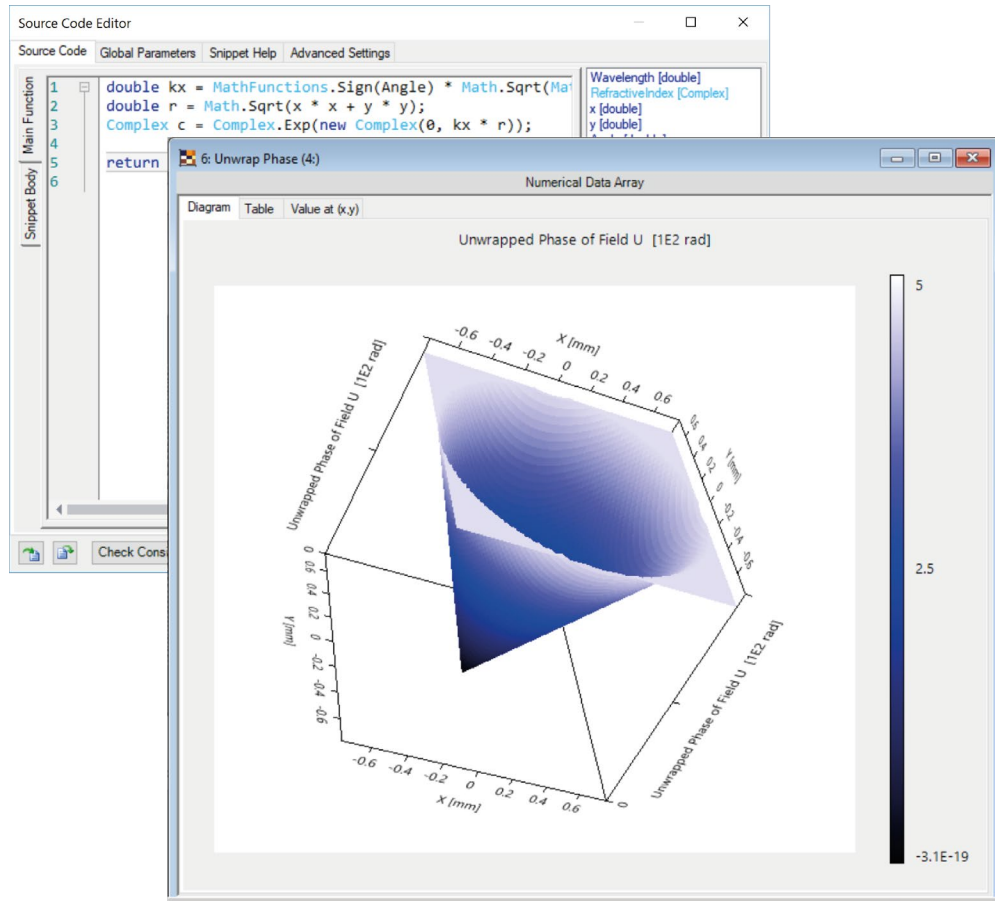


# Programming an Axicon Transmission Function

# Abstract



Axicons are routinely employed in optics for the purpose of transforming more or less homogeneous input light distributions into ring-shaped ones. This functionality is useful across many fields. In this programming example we show you how to programme an idealised component (transmission function) that performs like an axicon. You can then extrapolate these notions to programme your own ideal components in VirtualLab Fusion, using the Programmable Function feature.

# Task Description & Sample Code

## Task:

Use the Programmable Function feature to generate an ideal component that replicates the function of an axicon. The angle  $\alpha$  of the axicon should be a user-defined parameter.

$$\check{f}(x, y) = \exp \left\{ i \left[ \sqrt{\frac{\left(\frac{2\pi}{\lambda} \tan(\alpha)\right)^2}{1 + \tan^2(\alpha)}} \sqrt{x^2 + y^2} \right] \right\}$$

## Main Function

```
double kx = MathFunctions.Sign(Angle) *
    Math.Sqrt(Math.Pow(Math.Tan(Angle) *
    MathFunctions.TwoPi / Wavelength, 2) /
    (1 + Math.Pow(Math.Tan(Angle), 2)));
double r = Math.Sqrt(x * x + y * y);
Complex c = Complex.Exp(new Complex(0, kx * r));

return c;
```

### Global Parameters (User Defined)

Variable	Value	Allowed range
double Angle	3 deg	-90 deg - 90 deg

**Hint:** Note that the snippet of a Programmable Function can be used as-is in a Programmable Source as well!

# Document Information

title	Programming an Axicon Transmission Function
document code	CZT.0027
version	1.0
toolbox(es)	Starter Toolbox
VL version used for simulations	7.4.0.49
category	Feature Use Case
further reading	<ul style="list-style-type: none"><li>- <a href="#">Customizable Help for Programmable Elements</a></li><li>- <a href="#">How to Work with the Programmable Function &amp; Example (Cylindrical Lens)</a></li></ul>