

EQUIDISTANT CYLINDRICAL

10xxxxxx.1

LEGAL RESTRICTIONS:

This Reusable Software Component (RSC) contains data with Unlimited Government Rights.

DESCRIPTION:

EQUIDISTANT CYLINDRICAL is a C language code component that provides conversions between Geodetic coordinates (latitude and longitude) and Equidistant Cylindrical projection coordinates (easting and northing).

CERTIFICATION LEVEL:

This RSC has been certified at level 4. A level 4 component satisfies the criteria for reliability, testing, and documentation for the Army Reuse Center (ARC). The component comes with test materials and a Reuse Manual that aids in integrating the component into a software system.

LEVEL OF TESTING/ACCEPTANCE:

Unit and integration testing have been performed for the functions contained in this component.

PURPOSE/INTENDED USE:

The purpose of EQUIDISTANT CYLINDRICAL is to provide a reusable component that supports the following coordinate conversions:

- Geodetic coordinates (latitude and longitude in radians) to Equidistant Cylindrical projection coordinates (easting and northing in meters), and
- Equidistant Cylindrical projection coordinates (easting and northing in meters) to Geodetic coordinates (latitude and longitude in radians).

A particular ellipsoid is specified in terms of the following parameters:

- Semi-Major Axis (a): Radius (in meters) at the equator, and
- Semi-Minor Axis (b): Radius (in meters) at a pole.

A particular variation of the Equidistant Cylindrical projection is specified in terms of the following parameters:

- Central Meridian – Longitude (in radians) at the horizontal center of the projection,
- Standard Parallel – Latitude (in radians) at which the scale factor of the projection is 1.0,
- False Easting – A coordinate value (in meters) assigned to the central meridian of the projection to avoid the inconvenience of using negative coordinates, and
- False Northing – A coordinate value (in meters) assigned to the origin latitude of the projection (which is always the equator) to avoid the inconvenience of using negative coordinates.

HARDWARE/ENVIRONMENT CONSTRAINTS:

There are no hardware or environment constraints. There are no limitations.

FUNCTIONS:

`Set_Equidistant_Cyl_Parameters` – This function sets the ellipsoid parameters and Equidistant Cylindrical projection parameters for the particular variation of the Equidistant Cylindrical projection that is to be used in subsequent coordinate conversion operations, returning the scale factor for the specified projection.

`Get_Equidistant_Cyl_Parameters` – This function returns the current values of the ellipsoid parameters and Equidistant Cylindrical projection parameters.

`Convert_Geodetic_To_Equidistant_Cyl` – This function converts the specified geodetic coordinates (latitude and longitude in radians) to Equidistant Cylindrical projection coordinates (easting and northing in meters) using the current ellipsoid parameters and Equidistant Cylindrical projection parameters.

`Convert_Equidistant_Cyl_To_Geodetic` – This function converts the specified Equidistant Cylindrical projection coordinates (easting and northing in meters) to geodetic coordinates (latitude and longitude in radians) using the current ellipsoid parameters and Equidistant Cylindrical projection parameters.

EXAMPLE APPLICATIONS:

The following example illustrates how EQUIDISTANT CYLINDRICAL can be used to convert Geodetic coordinates to Equidistant Cylindrical projection coordinates and back again:

Function Call:

```
status = Set_Equidistant_Cyl_Parameters (a, b, Std_Parallel, Central_Meridian,  
False_Easting, False_Northing)
```

Inputs:

a	6371007.0 (Radius of sphere with same surface area as WGS84 Ellipsoid)
b	6371007.0 (Radius of sphere with same surface area as WGS84 Ellipsoid)
Std_Parallel	0.0
Central_Meridian	0.0
False_Easting	0.0
False_Northing	0.0

Outputs:

None

Function Call:

```
status = Convert_Geodetic_To_Equidistant_Cyl (Latitude, Longitude, Easting, Northing)
```

Inputs:

Latitude:	-35.0
Longitude:	75.0

Outputs:

Easting:	-2149467.29
Northing:	-3891826.82

Function Call:

```
status = Convert_Equidistant_Cyl_To_Geodetic (Easting, Northing, Latitude, Longitude)
```

Inputs:

Easting: -2149467.29

Northing: -3891826.82

Outputs:

Latitude: -35.0

Longitude: 75.0