

LAMBERT

10xxxxxx.1

LEGAL RESTRICTIONS:

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

DESCRIPTION:

LAMBERT is a C language code component that provides conversions between Geodetic coordinates (latitude and longitude) and Lambert Conformal Conic projection coordinates (easting and northing) with two standard parallels.

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 LAMBERT is to provide a reusable component which supports the following coordinate conversions:

- Geodetic coordinates (latitude and longitude in radians) to Lambert Conformal Conic projection coordinates (easting and northing in meters) with two standard parallels,
- Lambert Conformal Conic projection coordinates (easting and northing in meters) with two standard parallels 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 Lambert Conformal Conic projection with two standard parallels is specified in terms of the following parameters:

- Central Meridian – Longitude (in radians) at the origin of the projection coordinate system,
- Origin Latitude – Latitude (in radians) at the origin of the projection coordinate system,
- 1st Standard Parallel – Latitude (in radians) of the first of the two standard parallels, where the point scale factor is 1.0,
- 2nd Standard Parallel – Latitude (in radians) of the second of the two standard parallels,
- False Easting – A coordinate value (in meters) assigned to the origin longitude 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 to avoid the inconvenience of using negative coordinates.

Note that when the two standard parallels are equal to each other, the result is equivalent to a Lambert Conformal Conic projection with one standard parallel.

HARDWARE/ENVIRONMENT CONSTRAINTS:

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

FUNCTIONS:

Set_Lambert_Parameters – This function sets the ellipsoid parameters and Lambert Conformal Conic projection parameters for the particular variation of the Lambert Conformal Conic projection that is to be used in subsequent coordinate conversion operations.

Get_Lambert_Parameters – This function returns the current values of the ellipsoid parameters and Lambert Conformal Conic projection parameters.

Convert_Geodetic_To_Lambert – This function converts the specified geodetic coordinates (latitude and longitude in radians) to Lambert Conformal Conic

projection coordinates (easting and northing in meters) using the current ellipsoid parameters and Lambert Conformal Conic projection parameters.

Convert_Lambert_To_Geodetic – This function converts the specified Lambert Conformal Conic projection coordinates (easting and northing in meters) to geodetic coordinates (latitude and longitude in radians) using the current ellipsoid parameters and Lambert Conformal Conic projection parameters.

EXAMPLE APPLICATIONS:

The following example illustrates how LAMBERT can be used to convert Geodetic coordinates to Lambert Conformal Conic projection coordinates and back again:

Function Call:

```
status = Set_Lambert_Parameters (a, b, Origin_Latitude, Central_Meridian,  
                                Std_Parallel_1, Std_Parallel_2, False_Easting, False_Northing)
```

Inputs:

a	6378137.0
b	6356752.3142
Origin_Latitude	45.0
Central_Meridian	0.0
Std_Parallel_1	40.0
Std_Parallel_2	50.0
False_Easting	0.0
False_Northing	0.0

Function Call:

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

Inputs:

Latitude:	35.0
-----------	------

Longitude: 75.0

Outputs:

Easting: 5972201

Northing: 1872576

Function Call:

status = Convert_Lambert_To_Geodetic (Easting, Northing, Latitude, Longitude)

Inputs:

Easting: 5972201

Northing: 1872576

Outputs:

Latitude: 35.0

Longitude: 75.0