# REUSE MANUAL

# GEOID

# 10xxxxxx.1

# Implementation

## TABLE OF CONTENTS

# SECTION 1. INTRODUCTION

## 1.1 PURPOSE OF THE REUSE MANUAL

This document describes the characteristics of the GEOID reusable software component and provides instructions on its installation and operation. The manual is a self-contained reference for the software engineer intending to incorporate the component in another software system. This manual was written with the assumption that the user has a basic working knowledge of C and is familiar with fundamental C concepts and terminology.

## 1.2 PURPOSE OF THE REUSABLE SOFTWARE COMPONENT

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

- WGS84 Ellipsoid height (meters) to Geoid height (meters) at the specified geodetic coordinates (latitude and longitude in radians), using the EGM96 gravity model, and

- WGS84 Geoid height (meters) to Ellipsoid height (meters) at the specified geodetic coordinates (latitude and longitude in radians), using the EGM96 gravity model.

A particular variation of the Geoid conversion is specified in terms of the following parameters:

- Longitude – Longitude (in radians) at the height conversion point,

- Latitude – Latitude (in radians) at the height conversion point.

## 1.3 GENERAL INFORMATION

### 1.3.1 POINT OF CONTACT

U.S. Army Topographic Engineering Center (USATEC)
Geospatial Informaiton Division (GID)
ATTN: CETEC-GD-A (Dan Specht)
7701 Telegraph Road
Alexandria, VA 22315-3864
Dan Specht (703) 428 - 6761 Project Manager

### 1.3.2 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.

### 1.3.3 LEGAL RESTRICTIONS

## SECTION 2. INSTALLATION

The following is a list of the files that make up the GEOID component:

Source Code Files:

```
geoid.c
```

Header Files :

```
geoid.h
```

Data Files :

```
none
```

The compilation instructions for the GEOID component are as follows:

DOS Makefile (Uses Microsoft C):

```
cl /nologo /W3 /FR /G2 /DNDEBUG /Gs /Ox /AM /D_DOS /c geoid.c
```

UNIX Makefile (Uses gcc compiler):

```
cc -g -O -ansi -Wall -c geoid.c
```

The compilation order of the GEOID component relative to other components is unconstrained.

### 2.1 PARTIAL REUSE

The GEOID component does not allow for partial reuse.

### 2.2 MODIFICATIONS

The GEOID component does not permit modifications.

# SECTION 3. ENVIRONMENT

This section provides details on the environment under which GEOID was developed, tested, and executed.

## 3.1 HARDWARE

### 3.1.1 DEVELOPMENT

The following is a list of hardware configurations under which GEOID was developed and tested.

- SUN SparcStation 20
- IBM compatible Pentium PC

### 3.1.2 TARGET

The following is a list of hardware configurations under which GEOID was executed.

- SUN SparcStation 20
- IBM compatible Pentium PC

## 3.2 SOFTWARE

### 3.2.1 OPERATING SYSTEM

The following is a list of operating systems under which GEOID was executed and tested.

- Solaris 2.5
- Windows 95

### 3.2.2 COMPILERS

The following is a list of compilers on which GEOID was compiled successfully.

- GCC version 2.8.1

- Microsoft Visual C++ version 6

## 3.3 ASSUMPTIONS AND PERFORMANCE LIMITATIONS

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

This RSC is written in ANSI C.

## SECTION 4. GLOBAL RSC ENVIRONMENT

### 4.1 TYPES

Not applicable.

### 4.2 CONSTANTS

The following is a list of significant visible constants declared globally in GEOID with their descriptions.

```
GEOID_NO_ERROR                  : No errors occurred in function
GEOID_FILE_OPEN_ERROR           : Geoid file opening error
GEOID_INITIALIZE_ERROR          : Geoid separation database can not
                                  initialize
GEOID_LAT_ERROR                 : Latitude out of valid range (-90 to 90
                                  degrees)
GEOID_LON_ERROR                 : Longitude out of valid range (-180 to 360
                                  degrees)
```

### 4.3 VARIABLES

The following is a list of significant global variables declared in GEOID with their descriptions.

```
static float GeoidHeightBuffer[NumbGeoidElevs] : buffer of Geoid elevations
```

### 4.4 INCLUDE FILES

```
math.h                          : Used to call standard math functions
string.h                        : Used for file parsing
geoid.h                         : Error codes and prototype error checking
```

### 4.5 DEPENDENCIES

None, other than the standard ANSI C math library.

# SECTION 5. FUNCTIONS

## 5.1 INITIALIZE_GEOID

### 5.1.1 DESCRIPTION

This function reads Geoid separation data from a file in the current directory and builds the Geoid separation table from it. If the separation file can not be found or accessed, an error code of GEOID_FILE_OPEN_ERROR is returned, If the separation file is incomplete or improperly formatted, an error code of GEOID_INITIALIZE_ERROR is returned, otherwise GEOID_NO_ERROR is returned.

### 5.1.2 INTERFACES AND EXAMPLES

The following is a list of the formal arguments required to use this function.

```
long Initialize_Geoid( )
```

Example:

```
status = Initialize_Geoid ()
```

Inputs:

None

Outputs:

None

### 5.1.3 DECLARATIONS

#### 5.1.3.1 TYPES

Not applicable.

#### 5.1.3.2 CONSTANTS

Not applicable.

5.1.3.3 VARIABLES

```
FILE *GeoidHeightFile   : A file of Geoid separation data
```

5.1.4 DEPENDENCIES

None.

5.1.5 ERROR HANDLING

This function returns the following status codes:

```
GEOID_FILE_OPEN_ERROR            : Geoid file opening error
GEOID_INITIALIZE_ERROR           : Geoid separation database can not
                                   initialize
```

5.2 GET_GEOID_HEIGHT

5.2.1 DESCRIPTION

This private function returns the height of the WGS84 Geoid above or below the WGS84 ellipsoid, at the specified geodetic coordinates, using a grid of height adjustments from the EGM96 gravity model.

5.2.2 INTERFACES AND EXAMPLES

The following is a list of the formal arguments required to use this function.

```
long Get_Geoid_Height ( double Latitude,
                        double Longitude,
                        double *DeltaHeight )
```

| | |
|---|---|
| Longitude | Longitude (in radians) at the height conversion point, |
| Latitude | Latitude (in radians) at the height conversion point, |
| DeltaHeight | Height difference between the WGS84 Ellipsoid and the Geoid at the conversion point. |

### 5.2.3 DECLARATIONS

#### 5.2.3.1 TYPES

Not applicable.

#### 5.2.3.2 CONSTANTS

Not applicable.

#### 5.2.3.3 VARIABLES

Not applicable.

### 5.2.4 DEPENDENCIES

None.

### 5.2.5 ERROR HANDLING

```
GEOID_LAT_ERROR                 : Latitude out of valid range (-90 to 90
                                  degrees)
GEOID_LON_ERROR                 : Longitude out of valid range (-180 to 360
                                  degrees)
```

## 5.3 CONVERT_ELLIPSOID_TO_GEOID_HEIGHT

### 5.3.1 DESCRIPTION

This function converts the specified WGS84 ellipsoid height at the specified geodetic coordinates to the equivalent Geoid height, using the EGM96 gravity model.

### 5.3.2 INTERFACES AND EXAMPLES

The following is a list of the formal arguments required to use this function.

```
long Convert_Ellipsoid_To_Geoid_Height ( double Latitude,
                                         double Longitude,
                                         double Ellipsoid_Height,
                                         double *Geoid_Height )
```

Latitude Latitude in radians (input),

Longitude Longitude in radians (input),

Ellipsoid_Height Height above the Ellipsoid in meters (input),

Geoid_Height Height above the Geoid in meters (output).

Example:

```
status = Convert_Ellipsoid_To_Geoid_Height (Latitude, Longitude,
           Ellipsoid_Height, Geoid_Height)
```

Inputs:

Latitude: 35.0

Longitude: -75.0

Ellipsoid_Height:

Outputs:

Geoid_Height:

### 5.3.3 DECLARATIONS

#### 5.3.3.1 TYPES

Not applicable.

#### 5.3.3.2 CONSTANTS

Not applicable.

#### 5.3.3.3 VARIABLES

double DeltaHeight height between the ellipsoid and Geoid

long Error_Code holds error code when generated

### 5.3.4 DEPENDENCIES

Get_Geoid_Height – Used to determine Geoid Height at the input geodetic coordinates.

### 5.3.5 ERROR HANDLING

This function returns the following status codes:

```
GEOID_LAT_ERROR                 : Latitude out of valid range (-90 to 90
                                  degrees)
GEOID_LON_ERROR                 : Longitude out of valid range (-180 to 360
                                  degrees)
```

## 5.4 CONVERT_GEOID_TO_ELLIPSOID_HEIGHT

### 5.4.1 DESCRIPTION

This function converts the specified WGS84 Geoid height at the specified geodetic coordinates to the equivalent ellipsoid height, using the EGM96 gravity model.

### 5.4.2 INTERFACES AND EXAMPLES

The following is a list of the formal arguments required to use this function.

```
long Convert_Geoid_To_Ellipsoid_Height ( double Latitude,
                                         double Longitude,
                                         double Geoid_Height,
                                         double *Ellipsoid_Height )
```

| | |
|---|---|
| Latitude | Latitude in radians (input), |
| Longitude | Longitude in radians (input), |
| Geoid_Height | Height above the Geoid in meters (output). |
| Ellipsoid_Height | Height above the Ellipsoid in meters (input), |

Example:

```
status = Convert_Geoid_To_Ellipsoid_Height (Latitude, Longitude, Geoid_Height,
           *Ellipsoid_Height)
```

Inputs:

| | |
|---|---|
| Latitude: | 35.0 |
| Longitude: | -75.0 |

Geoid_Height:

Outputs:

Ellipsoid_Height:

### 5.4.3 DECLARATIONS

#### 5.4.3.1 TYPES

Not applicable.

#### 5.4.3.2 CONSTANTS

Not applicable.

#### 5.4.3.3 VARIABLES

| | |
|---|---|
| double DeltaHeight | height between the ellipsoid and Geoid |
| long Error_Code | holds error code when generated |

### 5.4.4 DEPENDENCIES

Get_Geoid_Height – Used to determine Geoid Height at the input geodetic coordinates.

### 5.4.5 ERROR HANDLING

This function returns the following status codes:

This function returns the following status codes:

```
GEOID_LAT_ERROR                 : Latitude out of valid range (-90 to 90
                                  degrees)
GEOID_LON_ERROR                 : Longitude out of valid range (-180 to 360
                                  degrees)
```

## APPENDIX A STRUCTURE/DEPENDENCY DIAGRAMS

This component consists of a single compilation unit and depends only on the ANSI C standard math library.

## APPENDIX B DEFINITIONS/GLOSSARY

**Coordinate** – Linear or angular quantities that designate the position that a point occupies in a given reference frame or system. Also used as a general term to designate the particular kind of reference frame or system, such as Cartesian or spherical coordinates.

**Ellipsoid** – The surface generated by an ellipse rotating about one of its axes and often used to represent the surface of the Earth.

**Ellipsoid Height** – The height as referenced relative to the Earth's ellipsoid surface at a given geodetic position. Positions above the ellipsoid have positive values and those below have negative values.

**Geodetic Coordinates** – The quantities of latitude and longitude that define the position of a point on the surface of the earth with respect to the reference ellipsoid. Also, imprecisely called geographic coordinates.

**Geodetic Latitude** – The angle between the plane of the equator and the normal to the ellipsoid through the computation point. Geodetic latitude is positive north of the equator and negative south of the equator.

**Geodetic Longitude** – The angle between the plane of a meridian and the plane of the prime meridian. A longitude can be measured from the angle formed between the local and prime meridians at the pole of rotation of the reference ellipsoid, or by the arc along the equator intercepted by these meridians.

**Geoid** – A theoretical surface over the Earth representing sea level that ungulates due to variations in the Earth's mass distribution, and is everywhere perpendicular to the direction of gravity.

**Geoid Height** – The height as referenced relative to the Earth's Geoid surface at a given geodetic position. Positions above the Geoid have positive values and those below have negative values.

**Meridian** – A north-south reference line, particularly a great circle through the geographical poles of the earth, from which longitudes and azimuths are determined; or the intersection of a plane forming a great circle that contains both geographic poles of the earth, and the ellipsoid.

**Parallel** – A line on the earth, or a representation thereof, that represents the same latitude at every point.

## APPENDIX C REFERENCES

(1) Topographic Engineering Center, TEC-SR-7,.**Handbook for transformation of DATUMS, PROJECTIONS, GRIDS, AND COMMON COORDINATE SYSTEMS**, January 1996.