

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/27253833>

# Converting GPS coordinates [phi, lambda, h] to navigation coordinates (ENU)

ARTICLE · APRIL 2002

Source: OAI

---

CITATIONS

5

---

READS

814

1 AUTHOR:



[Samuel Picton Drake](#)

University of Adelaide

47 PUBLICATIONS 256 CITATIONS

SEE PROFILE



# Converting GPS Coordinates ( $\phi\lambda h$ ) to Navigation Coordinates ( $ENU$ )

*S. P. Drake*

Surveillance Systems Division  
Electronics and Surveillance Research Laboratory

DSTO-TN-0432

## ABSTRACT

This technical note outlines a more efficient procedure for converting latitude, longitude and height in the GPS coordinates to local east, north, up coordinates in the navigation frame.

APPROVED FOR PUBLIC RELEASE

*Published by*

*DSTO Electronics and Surveillance Research Laboratory*

*PO Box 1500*

*Edinburgh, South Australia, Australia 5111*

*Telephone: (08) 8259 5555*

*Facsimile: (08) 8259 6567*

*© Commonwealth of Australia 2002*

*AR No. AR 012-335*

*April, 2002*

***APPROVED FOR PUBLIC RELEASE***

# Converting GPS Coordinates ( $\phi\lambda h$ ) to Navigation Coordinates ( $ENU$ )

## EXECUTIVE SUMMARY

In many applications relevant to the Australian Defence Force (ADF) it is necessary to convert the Global Positioning System (GPS) coordinates of latitude, longitude and height to a local navigation frame with coordinates, east, north and up. For example when testing navigation instruments, such as the inertial navigation system (INS), it is often helpful to compare these measurements with those obtained from an independent GPS receiver. An INS records the east, north and up displacement from its point of origin. To compare INS and GPS measurements we need to transform GPS coordinates to navigation coordinates. Furthermore, east, north, up coordinates are essential in determining the line of sight for terrain data given as latitude, longitude and height, such as digital terrain elevation data (DTED). For large amounts of data, e.g, trial data, this process may be very computationally intensive.

Means for converting GPS data to navigation frame coordinates already exist. However, the method presented in this report is roughly three times faster than a commonly employed one. The coordinate transformation routine outlined in this report is accurate to within 10m over a range of 60km.

The coordinate transformation method outlined here will be used in the Navwar simulation package currently under development in DSTO Edinburgh. It will have the effect of significantly reducing the processing time of line of sight calculations in jamming models.



## Author

**Samuel Picton Drake**

*Surveillance System Division*

Sam Drake is a research scientist with the Surveillance Systems Division of DSTO. He obtained his honors degree in physics at the University of Melbourne, and went on to do a PhD in mathematical physics at the University of Adelaide. Following a post-doctoral position at the University of Padua, Italy, he joined the Navigation Systems group in 1999 working on the operational analysis of the use of GPS systems in the Australian Defence Force.

---



# Contents

<b>Glossary</b>	<b>ix</b>
<b>1 Introduction</b>	<b>1</b>
<b>2 Converting from WGS84 to navigation coordinates</b>	<b>1</b>
<b>3 Matlab Program: dllh2denu</b>	<b>3</b>
<b>4 Accuracy of Method</b>	<b>5</b>
<b>5 Efficiency of Method</b>	<b>5</b>
<b>6 Conclusion</b>	<b>6</b>

# Appendices

<b>A Matlab Code</b>	<b>7</b>
A.1 dllh2denu.m . . . . .	7
A.2 comparison.m . . . . .	8
<b>B Glossary, Physical Constants and Mathematical notation</b>	<b>9</b>





## Glossary

**ADF** Australian Defence Force

**DSTO** Defence Science Technology Organisation

**GPS** Global Positioning System

**ECEF** Earth Centered Earth Fixed

**ENU** East North Up

**INS** Inertial Navigation System

**RMS** Root Mean Square

**WGS84** World Geodetic System 1984



# 1 Introduction

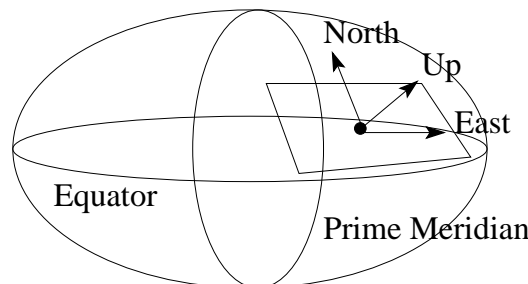
The aim of this report is to provide a useful and efficient means of using Global Positioning System (GPS) data to determine local range and bearing measurements. This is particularly pertinent owing to the ubiquity of GPS receivers.

The output of a GPS receiver is latitude, longitude and height in the World Geodetic System 1984 (WGS84) coordinate frame. However in many applications we are interested in the range and bearing of these coordinates with respect to a reference point, for example the range and bearing of an aeroplane, with respect to the control tower. When testing other navigation instruments, such as the inertial navigation system (INS), it is often helpful to compare these measurements with those obtained from an independent GPS receiver. An INS records the east, north and up displacement from its point of origin. To compare INS and GPS measurements we need to transform WGS84 coordinates to navigation coordinates. Furthermore, (east, north, up) coordinates are essential in determining the line of sight for terrain data given as latitude, longitude and height, such as digital terrain elevation data (DTED).

At this point the reader may ask, “hasn’t this already been done?” Yes, it has been done before, for example GPS<sup>TM</sup> have Matlab<sup>TM</sup> toolboxes which convert GPS coordinates into local navigation coordinates. In this report we derive a method which does this conversion three times faster than GPS<sup>TM</sup>. This difference is significant when dealing with large amounts of data.

## 2 Converting from WGS84 to navigation coordinates

Navigation coordinates are determined by the fitting of a tangent plane to a fixed point on the surface of the Earth. The  $\hat{e}$  axis points east, the  $\hat{n}$  axis north, and the  $\hat{u}$  axis points perpendicular to the tangent plane and away from the centre of the Earth.



*Figure 1: Navigation and ECEF frames.*

What follows below is a derivation of the transformation from latitude, longitude and height ( $\phi\lambda h$ ) to east, north and up (ENU). It has been included for completeness but the reader may safely jump to equation (4) if they are only interested in the final result.

The transformation from  $\phi\lambda h$  to  $ENU$  is a three stage process:

1. Determine latitude, longitude and height of reference point, e.g. location of radar.  
In this report it is the arbitrary point  $(\phi, \lambda, h)$ .

2. Express small changes in latitude, longitude and height in Earth Centred Earth Fixed (ECEF) coordinates.

GPS coordinates can be converted into ECEF coordinates using the following formulae:

$$\begin{aligned} x &= \left(\frac{a}{\chi} + h\right) \cos \phi \cos \lambda \\ y &= \left(\frac{a}{\chi} + h\right) \cos \phi \sin \lambda \\ z &= \left(\frac{a(1-e^2)}{\chi} + h\right) \sin \phi \end{aligned} \quad (1)$$

Where

$$\chi = \sqrt{1 - e^2 \sin^2 \phi},$$

$a$  and  $e^2$  are the semi-major axis and the first numerical eccentricity of the Earth respectively.

To convert small changes in latitude, longitude and height into ECEF coordinates we need to Taylor expand equation (1) about  $\phi \rightarrow \phi + d\phi$ ,  $\lambda \rightarrow \lambda + d\lambda$ , and  $h \rightarrow h + dh$ .

$$\begin{aligned} dx &= \left(\frac{-a \cos \lambda \sin \phi (1-e^2)}{\chi^3} - h \cos \lambda \sin \phi\right) d\phi - \left(\frac{a \sin \lambda \cos \phi}{\chi} + h \sin \lambda \cos \phi\right) d\lambda \\ &+ \cos \phi \cos \lambda dh + \left(\frac{1}{4}a \cos \phi \cos \lambda (-2 - 7e^2 + 9e^2 \cos^2 \phi) - \frac{1}{2}h \cos \lambda \cos \phi\right) d\phi^2 \\ &+ \left(\frac{a \sin \lambda \sin \phi (1-e^2)}{\chi^3} + h \sin \lambda \sin \phi\right) d\phi d\lambda - \cos \lambda \sin \phi dh d\phi \\ &+ \left(\frac{-a \cos \lambda \cos \phi}{2\chi} - \frac{1}{2}h \cos \lambda \cos \phi\right) d\lambda^2 - \sin \lambda \cos \phi dh d\lambda + \mathcal{O}(d\theta^3) + \mathcal{O}(dh d\theta^2) \\ dy &= \left(\frac{-a \sin \lambda \sin \phi (1-e^2)}{\chi^3} - h \sin \lambda \sin \phi\right) d\phi + \left(\frac{a \cos \lambda \cos \phi}{\chi} + h \cos \lambda \cos \phi\right) d\lambda \\ &+ \sin \phi \cos \lambda dh + \left(\frac{1}{4}a \cos \phi \sin \lambda (-2 - 7e^2 + 9e^2 \cos^2 \phi) - \frac{1}{2}h \sin \lambda \cos \phi\right) d\phi^2 \\ &+ \left(\frac{-a \cos \lambda \sin \phi (1-e^2)}{\chi^3} - h \cos \lambda \sin \phi\right) d\phi d\lambda - \sin \lambda \sin \phi dh d\phi \\ &+ \left(\frac{-a \sin \lambda \cos \phi}{2\chi} - \frac{1}{2}h \sin \lambda \cos \phi\right) d\lambda^2 + \cos \lambda \cos \phi dh d\lambda + \mathcal{O}(d\theta^3) + \mathcal{O}(dh d\theta^2) \\ dz &= \left(\frac{a(1-e^2) \cos \phi}{\chi^3} + h \cos \phi\right) d\phi + \sin \phi dh + \cos \phi dh d\phi \\ &+ \left(\frac{1}{4}a \sin \phi (-2 - e^2 + 9e^2 \cos^2 \phi) - \frac{1}{2}h \sin \phi\right) d\phi^2 + \mathcal{O}(dh d\phi^2), \end{aligned} \quad (2)$$

where  $d\theta$  is either  $d\phi$  or  $d\lambda$ .

3. By means of a rotation, displacements in ECEF coordinates are transformed to ENU coordinates.

The ECEF coordinates  $(dx, dy, dz)$  are orientated in such a way that from the centre of the Earth the  $\hat{z}$  points in the direction of true north,  $\hat{x}$  points in the direction of the prime meridian and the direction of  $\hat{y}$  is  $90^\circ$  from the prime meridian, see figure 1. The orientation of ENU coordinates is determined by rotating the ECEF coordinates; firstly about the  $\hat{z}$  axis by  $\lambda$  degrees and then the new  $\hat{y}$  axis by  $\phi$

degrees,

$$\begin{pmatrix} de \\ dn \\ du \end{pmatrix} = \begin{pmatrix} -\sin \lambda & \cos \lambda & 0 \\ -\sin \phi \cos \lambda & -\sin \phi \sin \lambda & \cos \phi \\ \cos \phi \cos \lambda & \cos \phi \sin \lambda & \sin \phi \end{pmatrix} \begin{pmatrix} dx \\ dy \\ dz \end{pmatrix}. \quad (3)$$

Substituting equation (2) into equation (3), and ignoring terms of  $\mathcal{O}(d\theta^3)$  and  $\mathcal{O}(dh d\theta^2)$  and higher, we get

$$\begin{aligned} de &= \left( \frac{a}{\chi} + h \right) \cos \phi d\lambda - \left( \frac{a(1-e^2)}{\chi^3} + h \right) \sin \phi d\phi d\lambda + \cos \phi d\lambda dh \\ dn &= \left( \frac{a(1-e^2)}{\chi^3} + h \right) d\phi + \frac{3}{2} a \cos \phi \sin \phi e^2 d\phi^2 + dh d\phi \\ &\quad + \frac{1}{2} \sin \phi \cos \phi \left( \frac{a}{\chi} + h \right) d\lambda^2 \\ du &= dh - \frac{1}{2} a \left( 1 - \frac{3}{2} e^2 \cos \phi + \frac{1}{2} e^2 + \frac{h}{a} \right) d\phi^2 \\ &\quad - \frac{1}{2} \left( \frac{a \cos^2 \phi}{\chi} - h \cos^2 \phi \right) d\lambda^2. \end{aligned} \quad (4)$$

### Note:

The coordinates  $(\phi, \lambda, h)$  are ellipsoidal (WGS84), not spheroidal (geocentric). The difference is most easily explained by figure 2.

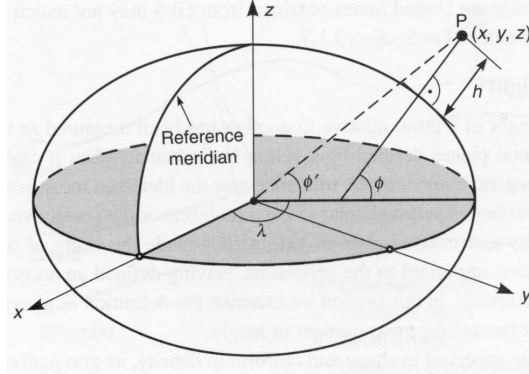


Figure 2: Ellipsoidal and spheroidal coordinates

## 3 Matlab Program: dllh2denu

Armed with equation (4) we are now in a position to write a Matlab program. The code for our program *dllh2denu.m* is given in appendix A. The routine *dllh2denu* is simple to use, it requires two inputs; the first is a vector indicating the location of the reference point (e.g. radar) in latitude, longitude in degrees and height in metres. The first input is a  $1 \times 3$  vector of the form

$$\text{llh0} = [\phi, \lambda, h].$$

The second input is a  $n \times 3$  matrix indicating the location of the points of interest (e.g. the location of the aeroplane). The second input is of the form

$$\text{llh} = \begin{bmatrix} \phi_1 & \lambda_1 & h_1 \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \phi_n & \lambda_n & h_n \end{bmatrix}.$$

The resulting output is a  $n \times 3$  matrix with the first, second, and third columns, representing the east, north and up displacement respectively in metres

$$\text{denu} = \begin{bmatrix} e_1 & n_1 & u_1 \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ e_n & n_n & u_n \end{bmatrix}.$$

As an example suppose the location of the reference point is

$$[\phi = 39^\circ, \lambda = -132^\circ, h = 0]$$

and three points of interested are located at

$\phi$	$\lambda$	$h$
$39^\circ + 0.5^\circ$	$-132^\circ$	0
$39^\circ + 0.5^\circ$	$-132^\circ + 0.5^\circ$	0
$39^\circ + 0.5^\circ$	$-132^\circ + 0.5^\circ$	60000

To determine the location of these points in ENU coordinates we would run *dllh2denu* in the following way:

```
>> format short g
>> llh0 = [39, -132, 0]

llh0 =

    39   -132     0

>> llh = [39 + 0.5, -132, 0;
39 + 0.5, -132+0.5, 0;
39 + 0.5, -132+0.5, 60000]

llh =

    39.5    -132         0
    39.5   -131.5         0
    39.5   -131.5    60000
```

```
>> denu = dllh2denu(llh0,llh)
```

```
denu =
```

```

      0      55510      -242.2
43008      55629      -389.07
43415      56153      59611
```

## 4 Accuracy of Method

The displacements as determined by Eq. (4) are approximations, since the Taylor expansion of Eq. (1) was truncated at second order. To estimate the error of this approximation it suffices to take an upper estimate on the effect of third order terms  $\mathcal{O}(d\theta^3)$  and  $\mathcal{O}(dh d\theta^2)$ , where  $d\theta$  is either  $d\phi$  or  $d\lambda$ . The greatest possible third order effect is at the equator. Third order terms at this point are of the order  $ad\theta^3$  and  $dh d\theta^2$ . If we limit  $d\theta$  to  $0.5^\circ$  and  $dh$  to 60km then this effect is less than 7m. Near the equator an angular offset of latitude or longitude of  $0.05^\circ$  equates to roughly 60km. Hence as a rule of thumb

Equation (4), and hence the program in appendix A, may be used if an error of less than 10m is acceptable and the range of measurements is less than 60km.

Should the range exceed 60km, or an accuracy much less than 10m be required, it is necessary to calculate  $dx, dy, dz$  using the method employed by GPSoft<sup>TM</sup>. This method firstly converts each point  $\phi_i, \lambda_i, h_i$  to  $x_i, y_i, z_i$  using Eq. (1). The displacements in ENU coordinates are then calculated by using Eq. (3), and the fact that  $dx_i = x_i - x_0$ , where  $x_0$  is the location of the reference point in ECEF coordinates.

To validate our code let us compare ENU displacements as determined by GPSoft<sup>TM</sup>'s *llh2xyz.m* and *xyz2enu.m* ( $\text{ENU}_{\text{GPSoft}}$ ) with those determined by *dllh2denu.m* ( $\text{ENU}_{\text{sam}}$ ). Taking the same reference and data points as before we find that

$\phi, \lambda, h$	$\text{ENU}_{\text{GPSoft}}$	$\text{ENU}_{\text{sam}}$	RMS difference
$39^\circ, -132^\circ, 0$	[0, 0, 0]	[0, 0, 0]	0m
$39+0.5^\circ, -132^\circ, 0$	[0, 55509.42, -242.21]	[0, 55510.13, -242.20]	0.7m
$39+0.5^\circ, -132 + 0.5^\circ, 0$	[43006.16, 55627.52, -388.04]	[43008.36, 55629.06, -389.07]	2.88m
$39+0.5^\circ, -132 + 0.5^\circ, 60000$	[43410.18, 56152.22, 59608.30]	[43415.27, 56152.66, 59610.93]	5.75m

## 5 Efficiency of Method

As already mentioned in the introduction, the main motivation behind this work is to speed up the processing of data. In this section a simple time comparison experiment



shows that data can be processed roughly three times faster using the method outlined in this report.

To perform this comparison fairly the programs *llh2xyz.m* and *xyz2enu.m* had to be modified so that they could process more than one coordinate at a time. The program *comparison.m* was used to compare the CPU time taken to calculate ENU using these two methods, the code is in appendix A. The results are shown below.

```
>> comparison
t_sat = 0.935 +- 0.01354
t_sam = 0.344 +- 0.005164
```

We can see from these results that the method presented in this report is roughly three times faster than that used by GPS<sup>TM</sup>Soft.

## 6 Conclusion

The aim of this note is to report work in DSTO to establish an efficient method by which to transform GPS coordinates to local navigation coordinates east, north and up. To this end equation (4) was derived and a Matlab program *dllh2denu.m* was created to evaluate it for any number of data points.

Equation (4), and consequently the program *dllh2denu.m*, are valid only if

- a) The range is less than 60km, and
- b) an error of a few meters is acceptable.

The accuracy and efficiency was compared with the method of GPS<sup>TM</sup>Soft and a simple test showed that they agreed to within a metre, but our method was three times faster than theirs.

## Appendix A: Matlab Code

### A.1 dllh2denu.m

```
function denu = dllh2denu(llh0,llh)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%CONSTANTS
a = 6378137;
b = 6356752.3142;
e2 = 1 - (b/a)^2;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%Location of reference point in radians
phi = llh0(1)*pi/180;
lam = llh0(2)*pi/180;
h    = llh0(3)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%Location of data points in radians
dphi= llh(:,1)*pi/180 - phi;
dlam= llh(:,2)*pi/180 - lam;
dh   = llh(:,3) - h;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%Some useful definitions
tmp1 = sqrt(1-e2*sin(phi)^2);
c1 = cos(lam);
s1 = sin(lam);
cp = cos(phi);
sp = sin(phi);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%Transformations
de = (a/tmp1+h)*cp*dlam - (a*(1-e2)/(tmp1^3)+h)*sp.*dphi.*dlam +cp.*dlam.*dh;

dn = (a*(1-e2)/tmp1^3 + h)*dphi + 1.5*cp*sp*a*e2*dphi.^2 + sp^2.*dh.*dphi + ...
      0.5*sp*cp*(a/tmp1 +h)*dlam.^2;

du = dh - 0.5*(a-1.5*a*e2*cp^2+0.5*a*e2+h)*dphi.^2 - ...
      0.5*cp^2*(a/tmp1 -h)*dlam.^2;

denu = [de, dn, du];
```

## A.2 comparison.m

```

clear all
N=100000; % Number data points
loops = 10;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%Initialization
phi0 = 39;
lam0 = -132;
h0 = 0;
orgllh = [phi0,lam0,0];
phi = zeros(N,1);
lam = zeros(N,1);
h = zeros(N,1);
enu = zeros(N,3);
xyz = zeros(N,3);
denu_sat = zeros(N,3);
denu_sam = zeros(N,3);
t1 = zeros(loops,1);
t2 = zeros(loops,1);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%Coordinate data
for i = 1:N;
    phi(i) = phi0 + 0.5*i/N;
    lam(i) = lam0 + 0.5*i/N;
    h(i) = h0 +i;
end
llh = [phi, lam, h];
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Comparison loop
for j = 1:loops
    t = cputime;
    orgxyz = llh2xyzmodi(orgllh);
    xyz = llh2xyzmodi(llh);
    denu_sat = xyz2enumodi(xyz,orgxyz);
    t1(j) = cputime -t;
    t = cputime;
    denu_sam = dllh2denu(orgllh,llh);
    t2(j) = cputime -t;
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Statistics
t1m = mean(t1);
t1std = std(t1);
t2m = mean(t2);
t2std = std(t2);
disp(['t_sat = ',num2str(t1m), ' +- ', num2str(t1std)]);
disp(['t_sam = ',num2str(t2m), ' +- ', num2str(t2std)]);

```

## Appendix B: Glossary, Physical Constants and Mathematical notation

Symbol	Value	Meaning
$a$	6378137	length of Earth's semi-major axis in metres
$b$	6356752.3142	length of Earth's semi-minor axis in metres
$e^2$	$6.69437999013 \times 10^{-3}$	first numerical eccentricity
$\phi$		latitude
$\lambda$		longitude
$h$		height
$e$		displacement in the east direction of the navigation frame
$n$		displacement in the north direction of the navigation frame
$u$		up displacement in the navigation frame
$x$		displacement in the $\hat{x}$ direction of the ECEF frame
$y$		displacement in the $\hat{y}$ direction of the ECEF frame
$z$		displacement in the $\hat{z}$ direction of the ECEF frame



## DISTRIBUTION LIST

Converting GPS Coordinates ( $\phi\lambda h$ ) to Navigation Coordinates ( $ENU$ )

S. P. Drake

Number of Copies

### DEFENCE ORGANISATION

#### Task Sponsor

Director General Aerospace Development 1

#### S&T Program

Chief Defence Scientist  
FAS Science Policy  
AS Science Corporate Management  
Director General Science Policy Development } 1

Counsellor, Defence Science, London Doc Data Sht

Counsellor, Defence Science, Washington Doc Data Sht

Scientific Adviser to MRDC, Thailand Doc Data Sht

Scientific Adviser Joint 1

Navy Scientific Adviser Doc Data Sht

Scientific Adviser, Army Doc Data Sht

Air Force Scientific Adviser 1

Director Trials 1

#### Aeronautical and Maritime Research Laboratory

Director, Aeronautical and Maritime Research Laboratory 1

#### Electronics and Surveillance Research Laboratory

Chief of Surveillance Systems Division 1

Research Leader, RLTSR 1

Head Navigation Systems 1

Task Manager: Chris Baker 1

Author: Sam Drake 1

#### DSTO Research Library and Archives

Library Edinburgh 1

Australian Archives 1

US Defense Technical Information Center 2

UK Defence Research Information Centre 2

Canada Defence Scientific Information Service 1

NZ Defence Information Centre 1

National Library of Australia 1

### **Capability Systems Staff**

Director General Maritime Development	Doc Data Sht
Director General Land Development	1

### **Navy**

SOSCI, Surface Combatants FEG Wartime Division, Maritime HQ, Garden Island	Doc Data Sht
--	--------------

### **Army**

ABCA Standardisation Officer, Puckapunyal	4
SO(Science), DJFHQ(L), MILPO, Enoggera, Queensland 4057	Doc Data Sht
NPOC QWG Engineer NBCD Combat Development Wing, Puckapunyal	1

### **Knowledge Staff**

Director General Command, Control, Communications and Computers	Doc Data Sht
---	--------------

### **Intelligence Program**

DGSTA, Defence Intelligence Organisation	1
Manager, Information Centre, Defence Intelligence Organisation	1

### **Corporate Support Program**

Library Manager, DLS-Canberra	1
Ms S. Doran, DLS, Sydney Wes	1

## **UNIVERSITIES AND COLLEGES**

Australian Defence Force Academy Library	1
Head of Aerospace and Mechanical Engineering, ADFA	1
Hargrave Library, Monash University	Doc Data Sht
Librarian, Flinders University	1

## **OTHER ORGANISATIONS**

NASA (Canberra)	1
AusInfo	1
State Library of South Australia	1

## **ABSTRACTING AND INFORMATION ORGANISATIONS**

Library, Chemical Abstracts Reference Service	1
Engineering Societies Library, US	1
Materials Information, Cambridge Scientific Abstracts, US	1
Documents Librarian, The Center for Research Libraries, US	1

## **INFORMATION EXCHANGE AGREEMENT PARTNERS**

Acquisitions Unit, Science Reference and Information Service, UK	1
Library – Exchange Desk, National Institute of Standards and Technology, US	1
National Aerospace Laboratory, Japan	1
National Aerospace Laboratory, Netherlands	1

## **SPARES**

DSTO Edinburgh Research Library	5
---------------------------------	---

<b>Total number of copies:</b>	<b>49</b>
--------------------------------	-----------



<b>DEFENCE SCIENCE AND TECHNOLOGY ORGANISATION DOCUMENT CONTROL DATA</b>						1. CAVEAT/PRIVACY MARKING	
2. TITLE  Converting GPS Coordinates ( $\phi\lambda h$ ) to Navigation Coordinates ( <i>ENU</i> )				3. SECURITY CLASSIFICATION  Document           (U) Title               (U) Abstract          (U)			
4. AUTHOR  S. P. Drake				5. CORPORATE AUTHOR  Electronics and Surveillance Research Laboratory PO Box 1500 Edinburgh, South Australia, Australia 5111			
6a. DSTO NUMBER DSTO–TN–0432		6b. AR NUMBER AR 012-335		6c. TYPE OF REPORT Technical Note		7. DOCUMENT DATE April, 2002	
8. FILE NUMBER 9505/19/203		9. TASK NUMBER JNT 00/086		10. SPONSOR DGAD		11. No OF PAGES 9	
						12. No OF REFS 0	
13. URL OF ELECTRONIC VERSION  <a href="http://www.dsto.defence.gov.au/corporate/reports/DSTO–TN–0432.pdf">http://www.dsto.defence.gov.au/corporate/ reports/DSTO–TN–0432.pdf</a>				14. RELEASE AUTHORITY  Chief, Surveillance Systems Division			
15. SECONDARY RELEASE STATEMENT OF THIS DOCUMENT  <i>Approved For Public Release</i>  <small>OVERSEAS ENQUIRIES OUTSIDE STATED LIMITATIONS SHOULD BE REFERRED THROUGH DOCUMENT EXCHANGE, PO BOX 1500, EDINBURGH, SOUTH AUSTRALIA 5111</small>							
16. DELIBERATE ANNOUNCEMENT  Australian Department of Defence and Defence Force personnel, and their equivalent in the US, UK and Canada.							
17. CITATION IN OTHER DOCUMENTS  No Limitations							
18. DEFTEST DESCRIPTORS  Geodetic coordinates                      Conversion Inertial navigation                         Global positioning system							
19. ABSTRACT  This technical note outlines a more efficient procedure for converting latitude, longitude and height in the GPS coordinates to local east, north, up coordinates in the navigation frame.							