PROGRAM GEOCONV

About this document

This document is the description and users manual of program GeoConv.

Author

Program GeoConv is written by Eino Uikkanen (mail: eino.uikkanen@iki.fi) for free use of anyone.

Briefly about GeoConv

Short description

Program GeoConv is a coordinate conversion program, which converts between different coordinate file formats, coordinate types and datums. GeoConv also filters coordinates, reduces track-files to routes and creates unique Waypoint-ID's based on waypoint description and/or old ID.

Geoconv is coded using Visual Basic Framework 3.5 - user must have this framework installed (most Windows users have)

GeoConv if designed to be used in batch mode only. Therefore the start-up of GeoConv-usage may require some effort especially of a person, who normally uses only interactive programs. It's anyway worth it, if you convert a lot or if you want to do many kinds of conversions. It's especially worth it, if you want to convert groups of files at the same time instead of file by file conversions or if you want to automate your conversions.

Errors

I have tried to minimize the amount of programming errors in GeoConv by building it on carefully designed and tested program primitives. However, because of the large number of different input- and output-variants it has been impossible to test all combinations.

I am not a professional in geodesy, but I have tried to minimize the amount of errors in applied methods by carefully examining and studying the methods and by thoroughly checking the results of implemented methods.

However, I don't guarantee anything and I don't take any responsibility of correctness or accuracy of GeoConv's functioning or results.

The list of identified and corrected errors and new features is maintained in document GeoVers.htm. Your error reports and suggestions for improvement are greatly appreciated.

Version numbering

GeoConv has no version number, but you can check the version date by command `GEOCONV /?`.

Overall description of the functionalities

GeoConv reads the input-file(s), makes desired conversions and filterings, and writes the output-file. In basic use the user only needs to give the names of the input- and output-files and -formats (parameters INFILE, OUTFILE, INFORM and OUTFORM) and then run the program.

GeoConv recognizes following input- and output formats (INFORM and OUTFORM):

Format	Description
CRDLIST	User defined coordinate list
PCX5	Garmin PCX5, version 2.09, track, waypoint and route
WAYPOINT	B. Hildebrand, Waypoint+, version 1.7.17 , track, waypoint and route
OZIWPT	OziExplorer, waypoint-file
OZITRK	OziExplorer, track-file
OZIRTE	OziExplorer, route-file
MAGELLAN	Magellan PMGNWPL and PMGNTRK-formats, track, waypoint
MPSEXP	MapSource text export, track, waypoint (input only)
GPX	Topografix GPS eXchange format, waypoint, track, route
NMEA0183	NMEA0183-standard
GREK2KWPT	Genimap Reittikartta Plus, own places (Finnish)
GREK2KTRK	Genimap Reittikartta Plus, routes (Finnish)
GREK2KGPS	Genimap Reittikartta Plus, GPS-logs (Finnish)
NONE	No output-file, used for testing

In some file formats the input data must be categorized to one of three coordinate data types:

- Waypoint - named waypoint
- Track - track log, e.g. GPS-log
- Route - route given as series of named waypoints

If the value of the parameter INFORM is PCX5 or WAYPOINT, GeoConv recognizes the input-type by the content of the input-file. If the value of the parameter INFORM is something else, GeoConv recognizes the input-type by the value of the parameter INFORM. Therefore the user never needs to tell the input type.

There is normally no need to tell the output-type either, because GeoConv by default writes out the same type as it reads in (output-type=input-type). If the input-type is undefined, GeoConv writes track type by default. However, it is possible to explicitly define the output-type. For example, when reducing a track file to a route, you might want the output-type to be Route, not Track as the input-file. Output-type is defined by parameter OUTTYPE.

Description of the file formats supported by GeoConv s (INFORM, OUTFORM)

General

The input data can be either user defined or one of the predefined formats. The input- and output formats are set using GeoConv's parameters INFORM and OUTFORM. The default value for both is user defined format, CRDLIST. That means, that it is not mandatory to define the input- and/or output-format, if the format is CRDLIST. This section defines the main characteristics and use of the file formats supported by GeoConv.

If your application is not listed below, check if you can describe its file-format using user defined format (CRDLIST). If your application reads or writes NMEA-format, you can use input-/output-format NMEA0183.

N/B, that information may be lost in consecutive conversions, because the data content of the file formats vary.

Some applications expect the file extension to be in lowercase or in uppercase.

PCX5, Garmin PCX5, version 2.09

Writes version number 2.09 to the output-file. Change version number in the output-file manually, if you have an older or newer version. You can define the version-line in the output-file also with parameter geoconv.pcx5ver. The default value of this parameter is "`I  PCX5 2.09`".

When reading PCX5-file, make sure that the input-datum is in the datum list. The name of the input-datum is read from the input-file, but the values are fetched from the datum list.

GeoConv always writes PCX5-files in datum WGS84.

Waypoint	.WPT
Track	.TRK
Route	.RTE

WAYPOINT, Brent Hildebrand, Waypoint+, version 1.7.17

Only text-formats can be used.

When reading Waypoint+ -file, make sure that the input-datum is in the datum list. The name of the input-datum is read from the input-file, but the values are fetched from the datum list.

The output-datum of Waypoint+ -file is definable by parameter OUTDATUM. However, it is recommended to use datum WGS84, because the successful usage of other datums requires, that the datums are identically defined in Geoconv and Waypoint+.

Waypoint	.TXT
Track	.TXT
Route	.TXT

OZIWPT, OziExplorer, waypoint-file
OZITRK, OziExplorer, track-file
OZIRTE, OziExplorer, route-file

When reading OziExplorer -file, make sure that the input-datum is in the datum list. The name of the input-datum is read from the input-file, but the values are fetched from the datum list.

The output-datum of OZI -file is definable by parameter OUTDATUM. However, in order to avoid errors it is recommended to use datum WGS84.

You may optionally define default values for OziExplorer's track-line's properties with parameter OZITRKDEF.

Waypoint	.WPT	OZIWTP
Track	.PLT	OZITRK
Route	.RTE	OZIRTE

MAGELLAN, PMGNTRK and PMGNWPL-formats

Because the datum is fixed in this format, the values of the parameters INDATUM and OUTDATUM have no effect.

If the value of the parameter OUTNMEACS is Y, the optional NMEA0183 checksum is included in the output-sentence.

MPSEXP, MapSource text-export format (INFORM only)

When reading MapSource -file, make sure that the input-datum is in the datum list. The name of the input-datum is read from the input-file, but the values are fetched from the datum list.

GeoConv requires, that the coordinate format is one of the lat/lon –formats or UTM:

Supported MapSource position formats:
- Lat/Lon h ddd°mm's.s"
- Lat/Lon hddd°mm.mmm
- Lat/Lon hddd.ddddd°
- UTM

MapSource exports this format, but it can't be imported back to MapSource. Therefore you can use MPSEXP as INFORM only, not as OUTFORM. If you need to import tracks or waypoints to MapSource, use PCX5-format.

Only the fields listed below are read from the MapSource-text-export:

Waypoint:
- Waypoint-ID and description
- Latitude, longitude and height
- Date and time
- Symbol

Track:
- Latitude, longitude and height
- Date and time

GPX, Topografix GPS eXchange format

Because the datum is fixed to WGS84 in this format, the values of the parameters INDATUM and OUTDATUM have no effect.

GPX-format does not transfer height-coordinate (height above ellipsoid). Therefore GeoConv assumes in both input and output, that height above ellipsoid equals to the sum of the values of elements <ele> and <geoidheight>.

If input-file contains track-points, waypoints and route-points, but the order of appearance does not agree with GPX-standard, GeoConv does not change the order to agree with standard.

GeoConv's XML-parser recognizes and handles properly program instructions, comments and cdata-blocks, but program instructions and comments are not transferred to the output file.

```
<!--     -->        Comments read properly, but not written
<?name pidata?>     Prog. inst. read properly, but not written
<![CDATA[   ]]>     CDATA used in read and write
```

The values to parameters below are fetched from respective GeoConv-parameters. Values below are default values set by initialization run GeoConv.ini.

```
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.topografix.com/GPX/1/0"
xsi:schemaLocation="http://www.topografix.com/GPX/1/0 gpx.xsd">
```

GeoConv reads and writes the fields represented in the sample below:

```
<gpx version="1.0" creator="GeoConv"
   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
   xmlns="http://www.topografix.com/GPX/1/0"
   xsi:schemaLocation="http://www.topografix.com/GPX/1/0 gpx.xsd">
   <desc>From C:\GPSDATA\PCX5\SOUR\PORKNIEM.GRM</desc>

   <trk>
      <name>Track 01</name>
      <desc>Track 1</desc>
      <number>1</number>
      <trkseg>
         <trkpt lat="+059.975567" lon="+024.401246">
            <ele>0</ele>
            <time>1997-10-20T12:26:14Z</time>
            <geoidheight>0</geoidheight>
            <name>BRIDGE</name>
            <desc>Bridge to Varo</desc>
         </trkpt>
      </trkseg>
   </trk>

   <wpt lat="+060.008006" lon="+024.455481">
      <ele>0</ele>
      <time>1962-03-27T00:00:00Z</time>
      <geoidheight>0</geoidheight>
      <name>BRIDGE</name>
      <desc>Bridge to Varo</desc>
   </wpt>
   <rte>
     <name>Main Route</name>
     <desc><![CDATA[Main route to Abo]]></desc>
     <number>1</number>
     <rtept lat="+060.008006" lon="+024.455481">
       <ele>0</ele>
       <time>1962-03-27T00:00:00Z</time>
       <geoidheight>0</geoidheight>
       <name>BRIDGE</name>
       <desc>Bridge to Varo</desc>
     </rtept>
   </rte>
</gpx>
```

GREK2KWPT, Genimap Reittikartta Suomi Plus, omat paikat (Finnish)

GeoConv reads and writes text-files, which can be exported or imported to and from the application (Vie... / Tuo ....)

Because the datum is fixed in this format, the values of the parameters INDATUM and OUTDATUM have no effect.

Waypoint	.TXT	Vie.../Tuo... omat paikat (own places)

GREK2KTRK, Genimap Reittikartta Suomi Plus, reitit (Finnish)

GeoConv reads and writes text-files, which can be exported or imported to and from the application (Vie... / Tuo ....)

Because the datum is fixed in this format, the values of the parameters INDATUM and OUTDATUM have no effect.

Route	.TXT	Vie.../Tuo... reitti (Export/import route)

GREK2KGPS, Genimap Reittikartta Suomi Plus, GPS-file (Finnish)

GPS-file has to be put to the directory defined in the application.

Because the datum is fixed in this format, the values of the parameters INDATUM and OUTDATUM have no effect.

Track .GPS Check which is the directory defined by the application

NMEA0183, NMEA0183 standard

GeoConv reads sentences GGA, GLL, WPL, RMC, ZDA and PGRMM and writes sentences GGA, GLL and WPL.

Reading sentences GGA, GLL and WPL launch writing a point to the output-file if and only if the input-sentence is listed in the parameter INNMEA. The default value of INNMEA is GLL,WPL. N/B: GGA is missing from the default value of INNMEA for reason; if you include both GGA and GLL in INNMEA, and the input-file contains both GGA and GLL, every point is written twice in the output-file. This is because GGA and GLL contain the same coordinate information.

If input- or output-datum is not WGS84, which is the default value, it has to be defined by parameter INDATUM and/or OUTDATUM. Input-datum is also set by sentence PGRMM.

Reading GGA

If GGA is listed in the parameter INNMEA, reading GGA launches writing a track-line to the output-file.

The date from latest RMC or ZDA sentence is included when writing GGA-information.

When reading GGA-sentences the data below is derived from the input-data:

ID	FixTime (HHMMSS)
Description	Text-field containing following fields: - Fix quality (0=invalid, 1=GPS fix, 2=DGPS fix) - Number of satellites, - Horizontal dilution of position - Altitude above mean sea level, meters - Height of geoid (mean see level) above WGS84 ellipsoid, meters
Height	Height above the ellipsoid = sum of: - Altitude above mean sea level, meters - Height of geoid (mean sea level) above WGS84 ellipsoid, meters
Hours, minutes seconds	From the FixTime-field

Reading GLL

If GLL is listed in the parameter INNMEA, reading GLL launches writing a track-line to the output-file.

The date from latest RMC or ZDA sentence is included when writing GLL-information

If the latest GGA-sentence has the same fix-time as GLL, the GGA-information is included when writing GLL-information.

Reading WPL

If WPL is listed in the parameter INNMEA, reading WPL launches writing a waypoint-line to the output-file.

The date from latest RMC or ZDA sentence is included when writing GGA-information

Reading RMC or ZDA

GeoConv reads RMC and ZDA only to get the date, which is not included in GLL or GGA.

Reading RMC or ZDA doesn't launch any writing to the output-file.

Reading PGRMM

Sets input-datum. PGRMM is Garmin proprietary-sentence, and therefore it is never written to the output-file.

N/B: If the input datum is set by PGRMM is not WGS84, the height information in GGA is still in WGS84 datum and is not converted to other datums.

Writing NMEA

If both input- and output-formats are NMEA0183, GeoConv writes out the same sentences as it reads in (GGA of GGA, GLL of GLL and WPL of WPL).

If input- or output-format is NMEA0183 and the other format is something else, GeoConv interprets sentences GGA and GLL as track-type and sentence WPL waypoint-type.

If the value of the parameter OUTNMEACS is Y, the optional NMEA0183 checksum is included in the output-sentence.

Tables of the NMEA-actions

This table describes actions launched by input-NMEA-data. N/B: the actions "Write track-point" and "Write waypoint" are defaults, which can be changed with parameter OUTTYPE.

Input-sentence	Action
GGA	Write track-point (or waypoint), if GGA in parameter INNMEA
GLL	Write track-point (or waypoint), if GLL in parameter INNMEA
WPL	Write waypoint (or track-point), if WPL in parameter INNMEA
RMC	Read and store the NMEA-date (included in GGA, GLL and WPL information)
ZDA	Read and store the NMEA-date (included in GGA, GLL and WPL information)
PGRMM	Read and apply input datum (Garmin proprietary)

This table describes handling of NMEA-data in writing actions.

Procedure	INFORM	OUTFORM	Action
Write track-point	NMEA	NMEA	Write GGA if input is GGA and GLL if input is GLL
	OTHER	NMEA	Write both GGA and GLL
	ANY	OTHER	Write track-point according to the output-format
Write waypoint	NMEA	NMEA	Write WPL
	OTHER	NMEA	Write WPL
	ANY	OTHER	Write waypoint according to the output-format

CRDLIST, user defined coordinate-list

If the value of the parameter INFORM or OUTFORM is CRDLIST, GeoConv reads or writes a user defined list of coordinates.

The CRDLIST input-line may contain following information:

Field	Column definition	Remark
ID	INIDCOL	
Description	INDESCCOL	
Zone	INZONECOL	Used e.g. for UTM
Latitude	INLATCOL	
Longitude	INLONCOL	
Height	INHCOL	height above ellipsoid
Date	INDATECOL	
Time	INTIMECOL	

It is not mandatory, that input-line contains all fields listed above. In addition to the fields listed above, input-line can also contain other information, which can be transferred to the output-file as is.

The properties of the user defined coordinate-file and its fields are described by a set of parameters described in part "Parameters of the user defined format CRDLIST".

Parameter COMMENTSTR can be used to define, which lines should be regarded as comment-lines. Comment-lines are either ignored or written as is to the output-file depending on the value of parameter WRITECMT (Y/N). GeoConv interprets comment-lines as breaks in the track and start new track-block after every comment.

The content and format of the output-line of file-format CRDLIST is defined by parameter OUTLINEFORM, which lets you write a wide set of other information too.

Examples of lines in different user defined coordinate lists:

```
ILK263,65.123,28.543              ID, latitude, longitude, no height
ILK263,65.123,28.543,78           height included
6612345,3212345                   x, y
LAAKIO;64,124;24,2345             Column separator is semicolon
                                  (default is colon)
```

An example of a user defined coordinate list

```
rem    EUREF coordinates of the first class triangular points
rem    Source: Announcement 24 of Finnish Geodetic Institute
ILK042, Kymi (Ky),          60 31 15.86470, 26 54 25.76350, 71.214
ILK063, Rokkala (Rk),       61 59 27.40569, 30 07 34.63359, 189.685
ILK069, Koli (Kl),          63 05 39.80772, 29 48 31.81134, 364.704
```

NONE, no output-file

If the value of the parameter OUTFORM is NONE, GeoConv writes no output-file. Used for testing.

Passing the parameters to GeoConv and calling GeoConv

GeoConv is designed to be used in batch mode only - there is no way to give parameters to GeoConv in interactive mode.

Parameters can be set in two ways. The first way is to give the parameters in the command line. Command line parameters are given in format <parameter name>=<parameter value>. If you omit the parameter name, GeoConv assumes the first command line parameter to be the name of the input file (INFILE=) and the second command line parameter to be the name of the output file (OUTFILE=), e.g.:

```
GEOCONV suomi.wpt suomi.txt inform=pcx5 outform=waypoint
```

The other way is to use program FVALUE to set the parameters before calling GeoConv, e.g.:

```
fvalue geoconv.infile=myplaces.wpt
fvalue geoconv.outfile=myplaces.txt
fvalue geoconv.inform=PCX5
fvalue geoconv.outform=waypoint
GEOCONV
```

These two ways can be mixed, e.g.:

```
fvalue geoconv.latrange=59,70
fvalue geoconv.lonrange=20,30
fvalue geoconv.inform=PCX5
fvalue geoconv.outform=waypoint
GEOCONV myplaces.wpt myplaces.txt
```

If same parameter is set both in the command line and with program FVALUE, GeoConv uses the value given in the command line.

N/B: Parameters, which include special characters or spaces should be given with FVALUE only.

The idea of the program FVALUE is to provide a mean to edit parameter values in batch mode, "in fly". This makes it possible to have parameter settings and GeoConv-call in one batch file. It also makes it possible to include several CeoConv-calls in one batch-run and to edit parameter values in the batch file between the calls.

FVALUE-commands

Below the most important Fvalue-commands. Read more from Fvaluegb.doc

You can set the value of a GeoConvs parameter by command fvalue geoconv.<parameter name>=<parameter value> e.g.:

```
fvalue geoconv.inform=ozitrk
```

You can initialize the value of a GeoConvs setting the value to blank by command fvalue geoconv.<parameter name>= e.g.:

```
fvalue geoconv.inform=
```

You can check the value of a GeoConvs parameter by command fvalue geoconv.<parameter name>, e.g.:

```
fvalue geoconv.maxdist
```

You can see the values of all parameters set for GeoConv by command fvalue geoconv., e.g.:

```
fvalue geoconv.
```

General parameters

INFILE=name of the input-file(s)

The name of the input-file can be given as the first command-line parameter of with GeoConv-parameter INFILE. You can use wild chars (*,?) in the name of the input-file. If you use wild chars, GeoConv converts all files matching with the name.

E.g.:

```
GEOCONV c:\waypoint\nauvo1.gps
GEOCONV c:\pcx5\2001*.wpt
```

OUTFILE=name of the output-file

The name of the output-file can be given as the second command-line parameter or with GeoConv-parameter OUTFILE. By default, if the name of the output-file is not given, GeoConv prints on the screen.

If the name of the output-file is not complete, that is, e.g. the base or the extension of the name is missing, GeoConv takes the missing parts from the input-file-name. You need this functionality if you convert a set of files with one command. Then you can (or you have to) let GeoConv name the output-files according to the input-file name, e.g.:

`INFILE=\GREK\*.GPS`
`OUTFILE=\OZI\*.PLT` (you can omit the asterisk, e.g.: \OZI\.PLT)

input-file name	output-file name
`\GREK\MUONIO.GPS`	`\OZI\MUONIO.PLT`
`\GREK\INARI.GPS`	`\OZI\INARI.PLT`
`\GREK\KILPISJA.GPS`	`\OZI\KILPISJA.PLT`
etc.	

Examples:

`OutFile=c:\temp\tulos.txt`	Complete filename
`OutFile=d:\publish\.txt`	Filename missing and taken from the input-file name
`OutFile=d:\gps\turku.`	Extension missing and taken from the input-file name

INFORM=Format of the Input-file

Format of the Input-file: PCX5, OZIWPT, OZITRK, OZIRTE, etc... Default value is CRDLIST..

E.g.: Input-file format is GPX

```
INFORM=GPX
```

OUTFORM=Format of the Output-file

Format of the Output-file:: PCX5, OZIWPT, OZITRK, OZIRTE, etc.. Default value is CRDLIST.

E.g.: Output-file format is Waypoint+

```
OUTFORM=WAYPOINT
```

INNMEA, list of the NMEA0183-sentences launching a write to the output-file

GeoConv reads sentences GGA, GLL, WPL and RMC and writes sentences GGA, GLL and WPL.

Reading sentence GGA, GLL or WPL launches a write to the output-file if and only if the sentence is listed in parameter INNMEA.

`INNMEA=GGA,GLL,WPL`	
`INNMEA=GGA,WPL`	
`INNMEA=`	Default value, GLL and WPL

In most cases it is not recommended to let both GGA and GLL to launch a write to the output-file and let GGA to launch writing a point if and only if there are no GGL-sentences in the input-file.

OUTNMEACS=Y/N

If the value of OUTNMEACS is Y(es), program prints the optional NMEA0183 checksum in the output-formats NMEA0183 and MAGELLAN.

OUTTYPE=output-type (Waypoint,Track or Route)

The output-type is by default defined by GeoConv. However, you can explicitly define the output-type with parameter OUTTYPE:

`OUTTYPE=`	Output-type defined by GeoConv
`OUTTYPE=WAYPOINT`	Output-type is Waypoint
`OUTTYPE=TRACK`	Output-type is Track
`OUTTYPE=ROUTE`	Output-type is Route

INDATUM=Input-datum

Input-datum is in most cases defined by the program or input-file. In such cases the value of this parameter is ignored by GeoConv. Indatum is user-defined only if input-format is CRDLIST (user defined) and coordinate-type is D, DEG, DM or DMS.

GeoConv reads input-datum parameters from parameter-file DATUM.VAL, see section "Parameter Files".

OUTDATUM=Output-datum

Output-datum is in most cases defined by the program. In such cases the value of this parameter is ignored by GeoConv. Outdatum is user-defined only if:

- output-format is any of Waypoint+ or OziExplorer-formats or
- output-format is CRDLIST (user defined) and coordinate-type is DEG,DM or DMS.

GeoConv reads output-datum parameters from parameter-file DATUM.VAL, see section "Parameter Files".

MaxTBP, Maximum Time Between Points (seconds)

Some file-formats (e.g. NMEA0183) don't indicate a break in the track-line, but some formats indicate the start of a new track leg with a special field (e.g. Ozi, Waypoint+), blank line (e.g. PCX5) or similar. Parameter MaxTBP can be used to substitute the missing leg start information in the input-file, providing, that lines in the input-file have time-stamp information. If the value of MaxTBP is greater than zero, and the time difference between two consecutive points in a track is greater than MaxTBP, GeoConv assumes, that there is a break in the track line.

MaxTBP=	Default value 0 = MaxTPB has no effect
MaxTBP=60	Time difference greater than 60 seconds is interpreted as break

PCX5VER, version line format in PCX5-format

Some programs expect PCX5-files to have a certain version-number, even when the actual data format is the same. Parameter PCX5VER defines the version text, which is written to the PCX5-file.

```
geoconv.pcx5ver=I  PCX5 2.09          Default value
geoconv.pcx5ver=I  PCX5 2.08          Version 2.08
```

OZITRKDEF, default values for OziExplorer's track-line's properties

This parameter is used to define OziExplorer's track-line's properties , when output format is OziExplorer track (.PLT), but input format is something else. If input format is also OziExplorer track, the track-line properties are read from the input-file and this parameter is ignored. It is not mandatory to use this parameter, if the default values are OK (see below).

The parameter value is comma-separated list of following properties, e.g.

```
OZITRKDEF=1,16777215,0,2,8421376
```

Below the description of the items in the list. For more detailed inform, see OziExplorer's Help.

Item	Description	Default value	Typical/recommended values	Position in the file line
1	Width of the track plot	1	1 or 2	2
2	Track color	16777215	RGB code,	3
3	Track type	0	0,10,20	6
4	Track fill type	2	0–7	7
5	Track fill color	8421376	RGB code	8

WPIDFORM, default waypoint-ID
WPDESCFORM,default waypoint-description
TRACKDESCDEF, default track description

If the input file doesn't give waypoint ID, waypoint description or track description, this parameter can be used to define default values for waypoint ID, waypoint description or track description. Embedded printable fields can be used as in parameters OUTLINEFORM, OUTHEADERFORM and OUTFOOTERFORM.

E.g.
TrackDescDef=Stanford Lake Route
TrackDescDef=track @cdate@

The default values for these parameters are.

WPIDFORM=@NBR@
WPDESCFORM=@TOTDIST@
TrackDescDef=@fDate@

CREATEID=Y|N, create unique waypoint-ID's, default N (=don't create)
CREATEIDBYID=Y|N, create the unique ID primarily by old ID, default N (=create by Description)
CREATEIDLEN=length of the ID

If you set the parameter CREATEID (CREATEID=Y), GeoConv creates unique ID's to waypoints. If you set also parameter CREATEIDBYID (CREATEIDBYID=Y), creates GeoConv the new ID primarily by old ID and secondarily by the Description. If you don't set parameter CREATEIDBYID (which is default), creates GeoConv the new ID primarily by the Description and secondarily by the old ID.

If GeoConv can't create new unique ID by Description or old ID, it uses line numbers. If even this fails, returns GeoConv the old ID. Creation of unique ID's may increase the program execution time.

In the example below a six character long unique ID is created based on the Description:

```
Aspila              ASPILA
Karhujoki           KARHUJ
Karhujuomalampi     KARHUU
Karhujuomarotko     KARHUO
Karhuvaara          KARHUV
Karhuvuori          KARHUR
Nuoksa              NUOKSA
Tavastila           TAVAST
Tuorila             TUORIL
Tuorilampi          TUORIA
```

IDSEP=character string to be used to concatenate waypoint ID and Description.

In most file formats supported by GeoConv there is a separate field for both waypoint ID and waypoint Description. In some file formats there is only one field for waypoint ID or Description. That means, that either ID or Description will be lost, if conversion is done to or through such file format.

GeoConv can override this problem by concatenating ID and Description with IDSEP, when ID and Description must be put in one field.

The default IDSEP is two colons (::). Below an example of the concatenation:

```
KEIMIO + Top of Keimio = KEIMIO::Top of Keimio
```

GeoConv makes the concatenation when the output-format (OUTFORM=) is GREK2KWPT or GREK2KRTE. If either ID or Description is blank, GeoConv does not add IDSEP.

GeoConv splits the concatenated field correspondingly whenever it recognizes IDSEP.

To avoid mix-ups and confusion it is recommended to use only the default value of IDSEP. If you anyway use some other value, try to choose a value which doesn't appear in ID's and Descriptions otherwise.

You can set IDSEP-mechanism off by setting the value of IDSEP to NONE.

Examples:

`IDSEP=`	two colons (::, default value)
`IDSEP=/`	slash
`IDSEP=none`	IDSEP- mechanism off

FIXTRACK=substitute track breaks with straight lines Y/N.

The input track may consist of several separate track blocks and breaks between the blocks. Sometimes, e.g. when those breaks are short and caused by temporary loss of satellite coverage, you might want GeoConv to substitute the track breaks with straight lines. This can be done by setting the values of this parameter to Y. The default value is N.

Filtering-parameters

With these parameters you can select which lines of the input-file will be written to the output-file.

INTYPES=list of selected input-point-types (WAYPOINT, TRACK,ROUTE)

With this parameter you can choose, which point types in the input-file are written to the output-file. If INTYPES is empty, all point-types are written into the output-file. If INTYPES is not empty, only those point-types, which are listed in this parameter are written into the output-file. The point-types are WAYPOINT,TRACK,ROUTE

E.g.:

INTYPES=WAYPOINT	Only waypoints are written into the output-file
INTYPES=WAYPOINT,TRACK	Waypoints and trackpoints are written into the output-file

N/B: Parameter OUTTYPE you however can force the output-type. E.g. this parameter setting reads only track-points but forces them to waypoints.
INTYPES=TRACK
OUTTYPE=WAYPOINT

SKIPZEROCRDS=skip records with lat=0 and lon=0

If this parameter is set (SkipZeroCrds=Y, default value), GeoConv doesn't write out records, where both latitude and longitude are zero. Can be used in formats, where zero coordinate values indicate missing data.

LATRANGE=lower limit in degrees, higher limit in degrees
LONRANGE=lower limit in degrees, higher limit in degrees

If you set LATRANGE and/or LONRANGE, an input-line is excluded from the output-file if latitude or longitude is not between the set limits. You may leave the other end open set only lower or higher limit, e.g. LONRANGE=,70. GeoConv assumes, that limits are in input-datum.

E.g.:

```
latrange=59,70
lonrange=,30
```

RNGCRDTYPE=coordinate type of the coordinate limits

The coordinate type of the coordinate limits are set by parameters LATRANGE and LONRANGE. Default type is degrees (DEG). See parameter INCRDTYPE or OUTCRDTYPE for information about coordinate types.

INLINESTR=list of character strings which should match with the input-line
INIDSTR=list of character strings which should match with the Waypoint-ID
INDESCSTR=list of character strings which should match with the Waypoint-description

With parameters INIDSTR and INDESCSTR you can set a list of character strings which should match with the Input-line, Waypoint-ID or Waypoint-description respectively. The input-line is selected, if at least one of the character strings is match with the Input.line, Waypoint-ID or Waypoint-description.

Asterisk (*) in the character string matches with any string. Character comparison is not case sensitive.

E.g.:

`INLINESTR=Track*,Waypoint*`	Input-line must start with Track or Waypoint
`INIDSTR=LAM*`	ID must begin with (or lam,Lam etc.)

`INIDSTR=pol*,*tun`	ID must start with POL or end with TUN.
`INDESCSTR=*FISH*,*GUAY`	Description must contain FISH or end with GUAY

MAXDIST=maximum distance in kilometers

Input-line is excluded from the output-file if the distance from the point given with parameter FROMPOINT is greater than the maximum distance.

E.g.: `MAXDIST=30`

FROMPOINT=latitude,longitude

Sets the coordinates used in accordance with parameter MAXDIST. GeoConv assumes, that latitude and longitude are in input-datum.

E.g.: `FROMPOINT=62.142,24.543`

FROMCRDTYPE=coordinate type of FROMPOINT

Sets the coordinate type of latitude and longitude set by parameter FROMPOINT. Default type is degrees (DEG). See parameter INCRDTYPE or OUTCRDTYPE for information about coordinate-types.

Special input-file-parameters

These parameters need to be defined (=changed from their default values) only in exceptional situations, when the input file is a text file, but it's line length is greater then 2000 bytes or it's line delimiter is not carriage return + line feed (ASCII 13+10).

MaxLineLen=maximum length of the input line in bytes

If the maximum length of the input line is greater than 2000 bytes, use parameter MaxLineLen to set greater line length:

```
E.g.:

MaxLineLen=5012
```

LineEndStr=character string indicating the end of the input line

If the character string indicating the end of the input line is not the standard carriage return + line feed combination, use parameter LineEndStr to set different character string. You can use string @nnn to indicate any ASCII code.

```
E.g.

LineEndStr=@10
LineEndStr=@13@0
LineEndStr=***@10
```

Reduction-parameters

If either parameter REDUCEPOINTCOUNT or parameter REDUCEMAXERROR is non-zero, GeoConv reduces the input-file by removing points of the track until the filtering condition is fulfilled, still maintaining the shape of the track as close to the original shape as possible. Reduction is made after filtering.

If the type of the input-file is Track, but you want the type of the output-file to be Route, set the value of the parameter OUTTYPE to ROUTE.

Reducing of a track log to route is needed e.g., if you want upload the route to GPS, where the number of waypoints in a route is limited.

If the track to be reduced consists of several blocks (there are breaks in the track) and reduction is made using REDUCEMAXERROR, every block is reduced separately. If the track consists of several blocks and reduction is made using REDUCEPOINTCOUNT, the entire track is handled as one block. In both cases the breaks are maintained and the start- and end-points of the blocks are always selected to the reduced track regardless of the parameter settings. If you want GeoConv to disregard the breaks in the track, you might want to use parameter FIXTRACK.

The reduction algorithm performs optimal reduction result, but requires quite a lot of processing capacity. This is because all the points not yet selected are "competing" to be the next to be selected. We could say, that reduction "scope" or reduction "view" is the entire track. By limiting the reduction view we can get much faster reduction without substantially compromising in the result. This is done with parameter REDUCEPOINTVIEW, which defines the maximum number of points the reduction algorithm sees at a time. You might want to use this parameter is the size of the track-files if e.g. >10000 points.

REDUCEPOINTCOUNT=number of waypoints in the reduced track/route

Maximum number of waypoints in the reduced track/route

REDUCEMAXERROR=maximum deviation in meters

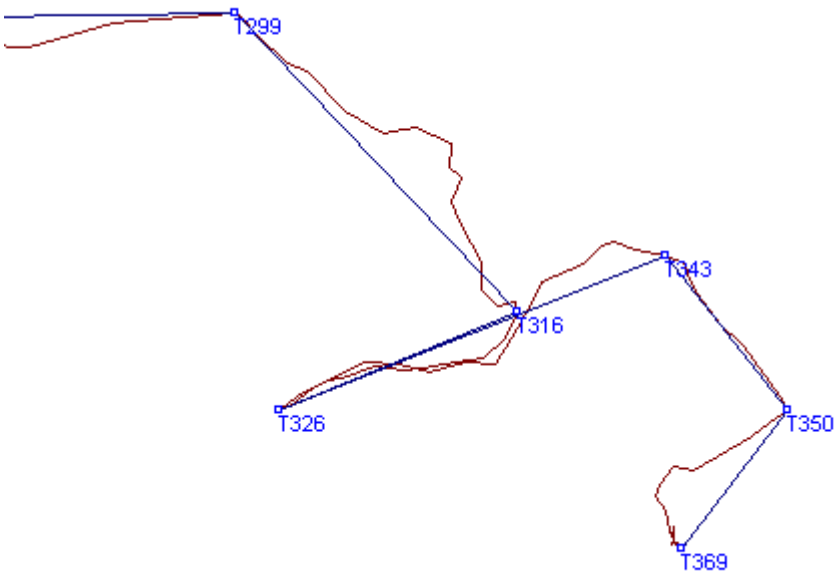
Maximum deviation between original and reduced track/route in meters.

REDUCEPOINTVIEW=size of reduction view in points

Maximum number of points the reduction algorithm sees at a time.

An example of a reduced track

In this example you can see the end of a track of 369 points (track, brown line) and the end of the



route reduced to 26 points (Route, blue line). The numbers refer to the original track-points.

Parameters used in format CRDLIST only

These parameters have a meaning only when associated with input- or output-format CRDLIST, that is, parameter INFORM or OUTFORM has value CRDLIST.

COMMENTSTR, string indicating the comment-line

If the input-file contains comment-lines, parameter COMMENTSTR defines how the comment-lines are recognised. CommentStr is comma-separated list of strings, which should match with the comment line. Asterisk (*) in the CommentStr matches with any text.

`REM*`	Line starting with REM is a comment-line (default value)
`CC:*,*BB:`	Line starting with CC: or ending with BB: is a comment-line
`##*##`	Line starting and ending with ## is a comment-line
` *`	Line starting with space is a comment-line

WRITECMT, write comment-line to the output-file

By default comment-lines are not written in the output-file, but if the parameter WRITECMT is set (WRITECMT=Y), GeoConv writes comment-lines as is to the output-file.

INHEADERLEN=length of the header block in lines

This parameter can be used to indicate the length of the header block in lines. These lines are skipped (read without writing anything to the output file)

INSEP=item separator of the input-line

Character, which separates items in the input-line. Default value is comma (,). You can give the separator character also as an ASCII-code preceded by @-character.

Separator is comma - no need to set parameter INSEP
```
ILK004, Geta (Ge),          60 23 06.38474, 19 50 53.29237, 118.309
```

Separator is semicolon - INSEP has to be set with command `FVALUE INSEP=;`
```
ILK009; Kökar (KÖ);         59 55 21,90272; 20 55 35,77597; 50.477
```

`FVALUE GEOCONV.INSEP=`	Separator character is comma (default value)
`FVALUE GEOCONV.INSEP=,`	Separator character is comma
`FVALUE GEOCONV.INSEP=;`	Separator character is semicolon
`FVALUE GEOCONV.INSEP=@9`	Separator character is tabulator character (code 9)

In most cases the character separator should be one character. This means, that every occurrence of that character is a separator. If the separator (value of INSEP) is longer than one character, the first character is used as separator, but two or more consecutive separator characters are interpreted as one separator.

E.g. interpretation of line A,,B,C,,D would depend on the input separator. Below the item numbers when the input separator is one character long and two character long:

INSEP=,
```
1  3 4  6
A,,B,C,,D
```

INSEP=,,
```
1  2 3  4
A,,B,C,,D
```

INIDCOL = column number of the ID-column
INDESCCOL = column number of the Description-column
INZONECOL = column number of the zone-column
INLATCOL = column number of the column, where is latitude, x or special coordinate type
INLONCOL = column number of the column, where is longitude or y
INHCOL = column number of the column, where is height

These parameters tell the column-numbers of the input-fields. The default value for all columns is zero, which means, that the field does not exist in the input-file.

Example 1:

```
FVALUE GEOCONV.INIDCOL=1,
FVALUE GEOCONV.INDESCCOL=2
FVALUE GEOCONV.INLATCOL=3
FVALUE GEOCONV.INLONCOL=4
FVALUE GEOCONV.INHCOL=5

ILK004, Geta (Ge),       60 23 06.38474, 19 50 53.29237, 118.309
ILK009, Kökar (KÖ),      59 55 21.90272, 20 55 35.77597, 50.477
ILK017, Dragsfjärd (Dr), 60 02 30.00060, 22 25 42.11416, 80.866
ILK021, Perniö (Pe),     60 14 28.38025, 23 03 01.74531, 103.973
```

Example 2:

```
FVALUE GEOCONV.INIDCOL=1
FVALUE GEOCONV.INDZONECOL=2
FVALUE GEOCONV.INLATCOL=4
FVALUE GEOCONV.INLONCOL=3

000009,35V, 364697.719, 6666187.578
000010,35V, 364703.049, 6665048.491
000011,35V, 364714.715, 6665062.999
000012,35V, 364704.939, 6665048.422
000013,35V, 364704.939, 6665048.422
```

INHFACTOR, input height factor

Input height is multiplied by this value. The default value is 1 (=no effect).

This can be used e.g. to change the sign of positive depth values, if depth values are used as heights.

INDATECOL, date format = column number of the date-column

This parameter tells the number of the input date column and optionally the respective date formats.

```
E.g.

INDATECOL=column, format
INDATECOL=5
INDATECOL=6,YYYY-MMM-DD
```

The format can be one of the following. In types 2 and 4 GeoConv reads also the time information from the date column. However, if INTIMECOL is defined too, the time information read from that column overrides the time information (but not date information) read from INDATECOL.

1. DATE (=Default format)

If the format is omitted or if it is DATE, GeoConv searches for one of the following date forms from the input string; if found, used as date information

DD.MM.YYYY
YYYY-MM-DD

MM/DD/YYYY
DD-MMM-YY (MMM=JAN,FEB,MAR,APR,..)
DD-MMM-YYYY (MMM=JAN,FEB,MAR,APR,..)

N/B, that the date string can be embedded. E.g. following strings would give same date.

```
15-NOV-2005
Created 10/15/2005
2005-10-15 SHORE
```

2. DAYS, SECONDS or MSECONDS

DAYS	Julian days = days from 1.1.-4713 12:00:00
SECONDS	Seconds from 1.1.-4713 12:00:00
MSECONDS	Milliseconds from 1.1.-4713 12:00:00

In this format you may define third parameter, which defines time shift between the zero point of the Julian date and some other date system. The unit of the shift parameter is always days. Instead of the numeric parameter you can use following parameters:

NOW	Time is calculated from the start of the program
INFILE	Time is calculated from the time stamp of the input file
DELPHI	Time is Delphi type (e.g. OziExplorer)

E.g.

```
indatecol=8,SECONDS,2440587.5
indatecol=6,MSECONDS,INFILE
```

3. Fixed length pattern (String containing YY)

A format pattern, where the date elements are represented fixed length strings. This format is recognized from two Y's (YY) in it.

YYYY	Year with four digits
YY	Year with two digits
MM	Number of the month (1-12)
DD	Day of the month (1-31)
MMM	Name of the month, three characters, e.g. SEP

E.g.

```
YYMMDD
YYYY-MM-DD
DD-MMM-YYYY
```

4. Other = free text = variable length pattern

The fourth way is to use free string, where you mix separators and following time elements. The string representing time elements are one character long and case sensitive. The lengths of the time elements may vary.

An E or e (=End) in the pattern string means, that the input string is right justified.

String	Time element
Y	Year
M	Month
D	Day
h	hours
m	minutes
s	seconds
e or E	Input string is right justified

E.g.

```
5/10/2005           Y/M/D
15-NOV-05           D-M-Y
3.2.2005 16:15      D.M.Y h:m
other text 2005.8.1 Y.M.De
```

INTIMECOL, time format = column number of the time-column

This parameter tells the number of the input time column and optionally the respective time format.

1. Fixed length pattern (String containing HH)

A format pattern, where the time elements are represented by following fixed length strings. This format is recognized and from two H's (HH) in it.

HH	Hours with two digits
MM	Minutes with two digits
SS.S	Seconds, number of decimals may vary

E.g.

```
HH:MM:SS
HH:MM:SS.SSS
```

2. Other = free text = variable length pattern

The second way is to use free string, where you mix separators and following time elements. The string representing time elements are one character long and case sensitive. The lengths of the time elements may vary.

An E or e (=End) in the pattern string means, that the input string is right justified.

String	Time element
H	hours
M	minutes
S	seconds
e or E	input string is right justified.

IN2ZONECOL = column number of the column, where is the comparison zone
IN2LATCOL = column number of the column, where is the comparison latitude
IN2LONCOL = column number of the column, where is comparison longitude
IN2HCOL = column number of the column, where is comparison height

Use these parameters to tell the column-numbers of coordinate values used for comparison purposes. The idea is to compare these values to the converted values of coordinates in columns INZONECO, INLATCOL, INLONCOL and INHCOL.

Values in columns IN2ZONECOL, IN2LATCON, IN2LONCOL and IN2HCOL must be given in output-formats, that is, datum in OUTDATUM, coordinate type in OUTCRDTYPE etc.

This mechanism can be used e.g. for testing coordinate-conversion parameters, see printable fields LAT2, LON2, H2, IN2DIST, MAXIN2DIST and AVEIN2DIST.

INCRDTYPE=type of the input-coordinates

See also the table 'Supported coordinate types' - crdtypes.htm – in GeoConv's home page.

D = General degree-input-format
D,separator character= General degree-input-format + separator character definition

In this format space (default) or 'separator character' is used as the separator between degrees, minutes and seconds. This format is most general and can be applied to most cases

In this format you can present

- degrees
- degrees and minutes
- or degrees, minutes an seconds

providing, that one or more spaces are used as a separator between degrees, minutes and seconds.

E.g.:

```
-17,8726448         Right (degrees)
+16.34324234        Right (degrees)
60 23 06.38474      Right (degrees, minutes and seconds)
+602306.38474       Wrong – separators missing
60 23   6.38474     Right – separator can be more than one space
```

DEG = Degrees and decimals of degrees
DM = Degrees, minutes and decimals of minutes
DMS = Degrees, minutes, seconds and decimals of seconds

In types DEG, DM and DMS spaces don't serve as separators between degrees, minutes and seconds, but spaces can be used to increase readability. Therefore you can freely use spaces, but the integer parts of minutes and seconds must have two numbers, that is, numbers less than 10 must be preceded by zero.

E.g.:

```
60 23 06.38474      Correct
+602306.38474       Correct – separators aren't mandatory
60 23  6.38474      Wrong - seconds should be preceded by zero
```

In all formats minus-character or characters S or W (South and West) in any part of the coordinate-value indicates negative value.

E.g.:

```
-20 13 14.5         Negative
20 13 15.5 W        Negative
```

In all formats you can use both period and comma as decimal point, except when used in a parameter list, where comma is used as a field separator.

E.g.:

```
-17,8726448         Right (comma)
+16.34324234        Right (period)
```

GON = Gon, grad

Degree-format, where full circle is 400 degrees. Alternative is to include character G in the expression, e.g. 40.1523G. In that case the expression is regarded as gons regardless of the INCRDTYPE.

```
200G                180 degrees
```

UTM = Universal Transverse Mercator

When INCRDTYPE is UTM, GeoConv reads the zone information from INZONECOL. The zone information consists of the Zone-Number (1-60) and the Band-Letter (CDEFGHJKLMNPQRSTUVWX), e.g. 14V. GeoConv also checks, if the zone field also contains the easting and northing values as in standard UTM-coordinate format, e.g. in the second column below. N/B, that the easting value comes in standard UTM format before the northing value.

```
000001,32U 579522.770 5551699.893
000002,34V 388455.958 6653097.435
000003,29U 420477.230 5551699.893
000004,27V 611544.042 6653097.435
000005,32F 579522.770 4448300.107
```

If the zone-field contains the zone information only without easting and northing, GeoConv reads the northing value from INLATCOL and the easting value from INLONCOL, e.g.

```
INIDCOL=1
INZONECOL=2
INLATCOL=3
INLONCOL=4

000001, 32U, 5551699.893, 579522.770
000002, 34V, 6653097.435, 388455.958
000003, 29U, 5551699.893, 420477.230
000004, 27V, 6653097.435, 611544.042
```

GeoConv assumes UTM to be defined between latitudes –80 and +84.

GeoConv gets the standard UTM parameters internally. Therefore there is no need to define UTM-grid parameters in the parameter file XYPARA.VAL. The standard parameters are:

Parameter	Value
Datum	From parameter UTMDATUM or if missing, WGS84
Central Meridian	6*Zonenumber –183
Latitude of Origin	0
False Easting	500000
False Northing	0 in the northern hemisphere and 10000000 in the southern hemisphere
Scale Factor at Central Meridian	0.9996 (exactly)

Following zones use exceptional parameters (according to the UTM standard). If the input-data contains references to the missing zones 32X, 34X or 36X, these zones are handled like standard zones.

Zone	Central Meridian	Width	Comment
31X	4,5	9	
33X	15	12	
35X	27	12	
37X	37,5	9	
31V	1,5	3	
32V	7,5	9	
32X			Missing, because of the expanded zones 31X and 33X
34X			Missing, because of the expanded zones 33X and 35X
36X			Missing, because of the expanded zones 35X and 37X

However, if there is a need to use other parameter values than the internal standard values in association with some UTM-zone (number and band), these values can be defined in the parameter-file XYPARA.VAL with grid-name UTM+ZoneNumber+BandLetter, e.g. INXYID=UTM04V. Only the exceptional values need to be defined and the rest of the fields may be left empty. In the example below the datum for UTM-zone 14V is changed to NAD27. The extra commas after NAD27 are not mandatory.

```
Name=Datum,CEM,LAO,FAE,FAN,SCF,Width, Free text
UTM14V=NAD27,,,,,
OSGB36=OSGB36,-2,49,400000,-100000,0.9996012717
```

TM = Transverse Mercator / Gauss-Krüger projected coordinates

User defined Transverse Mercator projected coordinate format. When this coordinate format is used, conversion parameters between the geodetic and grid-coordinates must be defined in the parameter file XYPARA.VAL and the parameter ID must be set using the parameter INXYID.

Fore more details about the needed parameters see section 'Parameter Files", XYPARA.VAL.

LCC1SP = Lambert Conic Conformal projected coordinates, one standard parallel
LCC2SP = Lambert Conic Conformal projected coordinates, two standard parallels

User defined Lambert Conic Conformal projected coordinate format. When this coordinate format is used, conversion parameters between the geodetic and grid-coordinates must be defined in the parameter file XYPARA.VAL and the parameter ID must be set using the parameter INXYID.

Fore more details about the needed parameters see section 'Parameter Files", XYPARA.VAL.

MERC1SP = Mercator projected coordinates, one standard parallel
MERC2SP = Mercator projected coordinates, two standard parallels

User defined Mercator projected coordinate format.

MERC1SP is the normal Mercator projection, where the only standard parallel is equator. MERC2SP is rarely used Mercator projection with two standard parallels.

When this coordinate format is used, conversion parameters between the geodetic and grid-coordinates must be defined in the parameter file XYPARA.VAL and the parameter ID must be set using the parameter INXYID.

Fore more details about the needed parameters see section 'Parameter Files", XYPARA.VAL.

KKJ = Finnish National Grid-coordinates

The zone is defined by the first number of y-coordinate. The conversion is three dimensional seven parameter similarity transformation using datum KKJ. See parameter INDATUM for the transformation parameters. Conversion parameters between KKJ-xy-coordinates and geographical coordinates are described in parameters file XYPARA.VAL, see section "Parameter Files".

YKJ = Finnish National Grid-coordinates, countrywide zone

Definition is exactly same as KKJ zone 3, but YKJ is used countrywide.

ERTS-TM35FIN, Finnish new National countrywide grid

As standard UTM, but used in countrywide zone.

ETRS-GKnn, Finnish new National grid for local area use

Used in 1 degree zones for local area works. 'nn' is the central meridian, e.g. central meridian of ETRS-GK27 is 27 degrees. The zone number (central meridian) is mandatory, because the zone can't be derived from the grid coordinate values (as in KKJ).

INXYID=ID referring to conversion parameters between input-xy-coordinates and geographical coordinates

INXYID is used to define the ID of the grid-conversion parameter-set when the selected coordinate type (INCRDTYPE) requires user defined grid-parameters (e.g. TM or LCC2SP).

When INXYID is defined, GeoConv reads conversion parameters between input-xy-coordinates and geographical coordinates from parameter-file XYPARA.VAL with ID given in XYID, see section "Parameter Files".

Examples of INXYID-values:

```
INXYID=OSGB36
INXYID=RT90
INXYID=KKJ1
```

OUTCRDTYPE=type of the output-coordinates

See also the table 'Supported coordinate types' - crdtypes.htm – in GeoConv's home page.

DEG = Degrees and decimals of degrees
DM = Degrees, minutes and decimals of minutes
DMS = Degrees, minutes, seconds and decimals of seconds
GON = Degrees; full circle is 400 degrees

OUTCRDTYPEs DEG, DM, DMS and GON can be given in simple mode or refined with formatting instructions, which are given as a text-string after the coordinate type. The coordinate type (e.g. DEG) and the formatting instruction must be separated by a comma, e.g.

`OUTCRDTYPE=DEG`	Simple mode (default formatting)
`OUTCRDTYPE=DM`	Simple mode (default formatting)
`OUTCRDTYPE=DEG,+S`	Refined with formatting instruction "+S"

The formatting instruction is a string of characters having following meanings:

Character	Meaning	Default
S	Put Spaces between degrees, minutes and seconds	No spaces
N	Use characters N, S, E and W	Use + and – to indicate the sign
+	Use + character to indicate positive values	Only negative values have sign
C	Use comma (,) as decimal-separator	Period (.) used as decimal-separator
*	Use NMEA-style: coordinate,X, where X=N,S,E or W (N/B comma after coordinate value)	
W	Decimal separator always after integer degrees (also in DM and DMS formats)	Decimal separator after integer degrees, minutes or seconds
T	Trailing sign (+,-, N,S,E or W). * Overrides this instruction	Leading sign
Number	Number of decimals (may be overridden in OUTLINEFORM)	3

Some examples, where `OUTLINEFORM=@LAT@ @LON@`

OUTCRDTYPE	Output
DEG	62.909 023.913
DM	6254.545 02354.782
DMS	625432.700 0235446.920
DMS,SN	N62 54 32.700 E023 54 46.920
DMS,SNT	62 54 32.700N 023 54 46.920E
DMS,SNC	N62 54 32,700 E023 54 46,920
DEG,*	6254.545,N 02354.782,E
DMS,W	62.5432700 023.5446920
DEG,6+T	62.909083+ 023.913033+

UTM = Universal Transverse Mercator

If the OUTCRDTYPE is UTM, GeoConv writes the zone-information (Zone-number and Band-letter) in the Zone-field, northing in the latitude-field and easting in the longitude-field. E.g. the OUTLINEFORM below would write standard UTM-coordinate format. Note the order (longitude=easting first, then latitude=northing) and spaces between Zone, easting and northing.

```
@ID@,@Zone@ @Lon,.3@ @Lan,.3@
```

GeoConv selects automatically the correct zone (zone and band) based on the coordinate values, calculates automatically the standard UTM-parameter-values for each zone and optionally reads the exceptional parameter-values from the parameter-file XYPARA.VAL. Read more about this and other information regarding UTM-handling from the description of INCRDTYPE.

TM = Transverse Mercator / Gauss-Krüger projected coordinates

User defined Transverse Mercator projected coordinate format. When this coordinate format is used, conversion parameters between the geodetic and grid-coordinates must be defined in the parameter file XYPARA.VAL and the parameter ID must be set using the parameter OUTXYID.

Fore more details about the needed parameters see section 'Parameter Files", XYPARA.VAL.

LCC1SP = Lambert Conic Conformal projected coordinates, one standard parallel
LCC2SP = Lambert Conic Conformal projected coordinates, two standard parallels

User defined Lambert Conic Conformal projected coordinate format. When this coordinate format is used, conversion parameters between the geodetic and grid-coordinates must be defined in the parameter file XYPARA.VAL and the parameter ID must be set using the parameter OUTXYID.

Fore more details about the needed parameters see section 'Parameter Files", XYPARA.VAL

MERC1SP = Mercator projected coordinates, one standard parallel
MERC2SP = Mercator projected coordinates, two standard parallels

User defined Mercator projected coordinate format.

MERC1SP is the normal Mercator projection, where the only standard parallel is equator. MERC2SP is rarely used Mercator projection with two standard parallels.

When this coordinate format is used, conversion parameters between the geodetic and grid-coordinates must be defined in the parameter file XYPARA.VAL and the parameter ID must be set using the parameter INXYID.

KKJn = x- and y- coordinates of KKJ-zone 0-5
KKJ = KKJ-zone is defined by longitude

The conversion is made by three dimensional seven parameter similarity transformation using datum KKJ. If the zone number is not given, it is calculated automatically from the longitude.

YKJ = Finnish National Grid-coordinates, countrywide zone

Definition is exactly same as KKJ zone 3, but YKJ is used countrywide.

ERTS-TM35FIN, Finnish new National countrywide grid

As standard UTM, but used in countrywide zone.

ETRS-GKnn, Finnish new National grid, Zones (nn)
ETRS-GK, Finnish new National grid, zone is defined by longitude

Used in 1 degree zones for local area works. 'nn' is the central meridian, e.g. central meridian of ETRS-GK27 is 27 degrees. If the zone number is not given, it is calculated automatically from the longitude.

OUTXYID=ID referring to conversion parameters between output-xy- and geographical coordinates

When OUTXYID is defined, GeoConv reads conversion parameters between output-xy-coordinates and geographical coordinates from parameter-file XYPARA.VAL with ID given in XYID, see section "Parameter Files". OUTXYID needs to be defined, when OUTCRDTYPE (below) is xy-coordinate type.

Examples of OUTXYID-values:

```
OUTXYID=OSGB36
OUTXYID=RT90
OUTXYID=KKJ1
OUTXYID=UTM14N
```

OUTLINEFORM=output line template

A character string or file, which defines the content and format of the output-line. The template can contain literal text and names of printable fields surrounded by @-characters. The names and the properties of the printable fields are listed in part "Printable fields". The names of the printable fields are replaced with the corresponding values at printing time.

With OUTLINEFORM value in the example below GeoConv would print first literal RN and then values of ID, latitude and longitude.

```
OUTLINEFORM=RN, @id@, @lat@, @lon@
```

If you want to use a file as the template instead of a character string, the value of the parameter OUTLINEFORM must have following format: `file:<file name>`, e.g.:

```
OUTLINEFORM=file:C:\GPS\LINEFORM.TXT
```

After the name of the printable field you can put a formatting instruction. The name and the formatting instruction have to be separated by comma.

The formatting instruction of a numeric field is in format K.D, where K is the length of the integer part and D is the number of decimals. If the actual length of the integer part is greater than desired length K, the entire integer part is printed anyway (number is never cut from the left). You can omit then length of the integer part (K), if you don't want to effect it anyway. In the example below latitude and longitude would be printed with four decimals and height would be printed with two decimals and minimum length of the integer part would be three.

```
OUTLINEFORM=RN, @id@, @lat,.4@, @lon,.4@, @h3.2@, @bearing,3.1@
```

The formatting instruction of a text field is in format K, where K is the length of the printed string. If the actual length of the text-string is greater than the desired length K, the entire text-string is printed anyway.

```
OUTLINEFORM=RN, @id@, @desc,40@, @lat,.4@, @lon,.4@, @h3.2@
```

Default value of the parameter OUTLINEFORM is:

```
@id@, @desc@, @lat,.6@, @lon,.6@, @h,3.3@
```

OUTHEADERFORM= output header template
OUTFOOTERFORM=output footer template

These parameters make it possible to print a header and/or footer to the output file in addition to the line information. The mechanism is the same as with parameter OUTLINEFORM. The default value for header and footer if null, that is, header or footer is not printed.

Below an example of a header- or footer template, which is saved as file:

```
Maximum speed = @maxspeed@
Minimum speed = @minspeed@
Average speed = @avespeed@
Total time = @trkhours@
Total length = @trkdist@
```

OUTPRLINE=Y/N, do print lines Y/N (OUTLINEFORM)

If you want to print only header- and/or footer without lines, you can set the value of this parameter to N (No). The default value of this parameter is Y (Yes, do print lines).

E.g..

`OUTPRLINE=`	Default value Yes, do print lines
`OUTPRLINE=Y`	Do print lines
`OUTPRLINE=N`	Do not print lines

INHEADERMASK, binary mask for the input file header
INLINEMASK, binary mask for the input line

With these parameters you can define the structure of a binary input file., e.g.

```
inlinemask=LNG,LNG,DBL,DBL,DBL,DBL,DBL,LNG,LNG
```

GeoConv first uses these masks to convert the input file to "visible" fields and then performs the rest of the conversion as with text files.

If there is no header in the binary input-file, you can omit the parameter INHEADERMASK and define only INLINEMASK.

Below the table of the recognized field types, which can be used in parameters INHEADERMASK and INLINEMASK.

Field value	Bytes	Description	Comment
BYTE	1	Byte	0-255
INT	2	Integer	Two's complement
USI	2	Unsigned integer	0-65535
LNG	4	Long integer	Two's complement
SGL	4	Single precision floating point	IEEE-standard
DBL	8	Double precision floating point	IEEE-standard
TEString	Variable	Reads up to and including 'String', but strips String away from the returned value. (T=To, E=Exclude)	String' may contain @nnn, where nnn is ASCII code of a character, e.g. `TE@255A`
TIString	Variable	As TEString, but the 'String' is included in the returned value (T=To, I=Include)	E.g. TI@
LINE	Variable	Up to and excluding CR+LF	Same as `TE@13@10`
COMMA	Variable	Up to and excluding comma (,)	Same as TE,
SPACE	Variable	Up to and excluding space	Same as `TE@32`
Number	Number	Fixed length text	

Unit conversion parameters (CRDLIST)

Units of measures of printable fields listed below and input-height in form CRDLIST can be converted using parameters below.

InHeightUnit=KM,M,FT,MI,NMI

Unit of the height in INHCOL and IN2HCOL. Default value M=meters

Units:

KM	Kilometer
M	Meter
FT	Feet
MI	Mile (statute)
NMI	Nautical mile

OutDistanceUnit=KM,M,FT,MI,NMI

Unit of the distance of printable fields of type 'distance' . Default value KM=kilometers

OutDurationUnit=H,S

Unit of the distance of printable fields of type 'duration' . Default value H=hours

Units:

H	Hour
S	Second

OutHeightUnit=KM,M,FT,MI,NMI

Unit of the height of printable fields of type 'height' . Default value M=Meters

OutSpeedUnit=KMH,MPS,MIPH,MPH,KNOT

Unit of the speed of printable fields of type 'speed' . Default value KMH=kilometers/hours

Units:

KMH	Kilometers/hour
MPS	Meters/second
MIPH or MPH	Miles/hour
KNOT	Knots, Nautical miles/hours

OutTimeDiff=hours

Time difference applied to printable fields of type 'time' . Default value is 0.

Eg. if time in the input-file is 15.12.2002 22:15 and OutTimeDiff=10, then output time is 16.12.2003 08:15.

Table of field-types for unit conversion

Field	Description	Field type
DIST	Distance from the previous point, including breaks	Distance
TRKDIST	Length of track line, breaks excluded	Distance
TOTDIST	Length of track line, breaks included	Distance
FROMDIST	Distance from FROMPOINT	Distance
REDUCEMAXDIST	Maximum deviation after reduction	Distance
HOURS	Time difference in hours between this and previous time stamp, breaks included	Duration
TRKHOURS	Accumulated time (HOURS), breaks excluded	Duration
TOTHOURS	Accumulated time (HOURS), breaks included	Duration
H	Height above ellipsoid	Height
IN2H	Height above ellipsoid / IN2	Height
ALTI	Altitude, height above geoid	Height
WGS84H	Height above WGS84-ellipsoid	Height
SPEED	Speed (DIST/HOURS)	Speed
MINSPEED	Minimum SPEED	Speed
MAXSPEED	Maximum SPEED	Speed
AVESPEED	Average SPEED	Speed
DATE	Date	Time
JULIAN	Julian date	Time
TIME	Time	Time

File selection parameters (FMK-parameters)

If you refer to a group of files by using characters *or ? in the name of the input-file, you may want to use file selection parameters. With file selection parameters you can select converted files based on files name, age, content, size etc.

Because file selection options are used in many programs, the naming of these parameter differs from the naming of the GeoConv-specific parameters. In file selection parameters 'GeoConv' must be substituted by 'FMK' (File Management Kit).

`Geoconv.datum=WGS84`	Setting GeoConv-specific parameter
`Fmk.age=30`	Setting file selection parameter (age<=30 days)

In the command line the parameter is always given without the "group-name", geoconv or fmk.

```
Geoconv inform=pcx5 size=-10000 age=14
```

File selection parameters are not initialized in the batch run GeocoIni.bat. You can initialize the file selection parameters with the command fvalue fmk.*=. If you use file selection parameters, it is recommended to give the initialization command also at the end of the batch run.

`fvalue fmk.*=`	Initialize file selection parameters

FMK.AGE=days - selection by the age of the file

Checks the time stamp (created or modified) of the file and selects the file if the age of the file is less than or equal to age (days).

`Fvalue fmk.age=30`	Select, if the age of the file is less than or equal to 30 days
`Fvalue fmk.age=0`	Select, if created or modified today

Selected time range can also be given as date-range in format dd.mm.yyyy-dd.mm.yyyy

`Fvalue fmk.age=1.9.2004-`	File is created or modified 1.9.2004 or later

FMK.SIZE=min-max, selection by the size of the file in bytes

Selects the file, if the size of the file in bytes is between the given limits.

`size=-10000`	Size is less than or equal to 10000 bytes
`size=2000-`	Size is at least 2000 bytes
`size=2000`	Size is at least 2000 bytes (without dash)

FMK.NAME=namestring, selection by the name of the file

Selects the file, if the namestring matches with the file-name. Asterisk in the namestring matches with any character string. By default the comparison test is not case sensitive.

`Name=*fish*`	Name must contain 'FISH'
`Name=01*`	Name must start with '01'
`Name=*5`	Name must end with '5'

Several namestrings can be used in one selection.

Name=fish*,*lake	Name must start with 'Fish' or end with 'Lake'

You can use characters * and ? which match with any character(s) in both geoconv.infile and fmk.name, but there are some differences in using these characters in these parameters:

- NAME accepts only *, but in INFILE you can use both * and ?
- In NAME you can use * anywhere, but in INFILE only in the end (everything after * in INFILE is disregarded)
- Handling characters * and ? in NAME is part of GeoConv-program, but handling characters * and ? in INFILE is based on operating system services.

FMK.CONTAINS= string1,sting2,string3,…, selection by the content of the file

Selects the file, if the file contains any of the strings. By default the comparison test is not case sensitive

```
fmk.contains=pcx5 2.09
fmk.contains=map project abe,calibration data
```

This parameter can also be given in the command line between quotation marks.

```
Geoconv ″OziExplorer track″
Geoconv ″map project abe,calibration data″
```

Parameter files

General information about parameter-files

GeoConv reads parameters of ellipsoids, datums and xy-conversions from respective parameter-files:

ELLIPS.VAL	Ellipsoidal parameters
DATUM.VAL	Datum definition parameters
XYPARA.VAL	Parameters describing conversions between geographical coordinates and metric xy-coordinates

Parameter-files must reside in the directory defined by the system parameter FVALUE, see section 'installation instructions'.

Parameter-files are text-files which can be edited by normal text editor. It is also possible to set values in the parameter-files using program FVALUE, which is used also to set program parameters.

An example of parameters file as is (text file):

```
Name=Datum,CEM,LAO,FAE,FAN,SCF,Width, Free text
KKJ=KKJ, 27, 0, 3500000, 0, 1, 3, Finnish National Grid
OSGB36=OSGB36,-2,49,400000,-100000,0.9996012717
```

An example of setting parameter values with program FVALUE. (in the command-line or batch-file)

```
FVALUE XYPARA.KKJ=KKJ, 27, 0, 3500000, 0, 1, 3, Finnish National Grid
FVALUE XYPARA.OSGB36=OSGB36,-2,49,400000,-100000,0.9996012717
```

ELLIPS.VAL, ellipsoidal parameters

Ellipsoidal parameters define the conversion between 3D-XYZ-parameters and geodetic parameters (latitude, longitude and height above ellipsoid)

The parameters in ELLIPS.VAL are semi major axis and flattening. Flattening is calculated (a-b)/a, where a=semi major axis and b=semi minor axis. Flattening can be expressed as a division, e.g. 1/298.257223563665

An example of ELLIPS.VAL as is (text file):

```
Name= equatorial radius, flattening, comment
WGS84= 6378137, 1/298.257223563665
WGS72= 6378135, 0.003352779454
GRS80= 6378137, 1/298.257222101
```

```
GRS67=6378160,1/298.247167
International= 6378388, 1/297
HAYFORD= 6378388, 1/297
Airy= 6377563.396, 0.003340850522
Bessel_1841= 6377397.155, 1/299.15281285
Clarke_1866= 6378206.400, 1/294.97870
Clarke_1880= 6378249.145, 0.003407561378, DoD
```

An example of setting ellipsoidal parameter values with program FVALUE.

```
FVALUE ELLIPS.GRS80= 6378137, 1/298.257222101
FVALUE ELLIPS.Clarke_1866= 6378206.400, 1/294.97870
```

DATUM.VAL, datum definition parameters

Datum-parameters define the transformation between two 3D-XYZ-coordinate systems, e.g. WGS84 <> NAD27.

The datum-file DATUM.VAL contains parameters for the 3D-similarity transformation between cartesian X/Y/Z-coordinates and the name of the ellipsoid used in conversion between cartesian X/Y/Z-coordinates and geodetic coordinates (φ, λ, h). The 3D-similarity transformation parameters can be given to or from an ECEF-datum, e.g. to or from WGS84 or ETRS89.

The 3D-similarity transformation is done by 7 Parameter Helmert Transformation formula:

```
|X2|   |DX|                 |  1   Rz -Ry |   |X1|
|Y2| = |DY| + (1+m/10^6) * | -Rz  1   Rx | * |Y1|
|Z2|   |DZ|                 |  Ry -Rx  1  |   |Z1|
```

Parameters in the file DATUM.VAL are:

1	1=parameters are given to ECEF-datum, e.g. to WGS84 0=parameters are given from ECEF-datum, e.g. from WGS84
2	Name of the ellipsoid
3-5	DX, DX, DZ = X/Y/Z shifts in meters
6-8	Rx, Ry, Rz = rotation angles around X/Y/Z axis in seconds
9	m = scale factor correction = (scale -1) * 10^6

If you add new datums to the file DATUM.VAL, try to avoid potential errors:
- check, that conversion direction between datum and ECEF-datum, e.g. WGS84, is right (parameter 1)
- check, that ellipsoid is defined in ELLIPS.VAL
- check, that the signs of the rotation angles are consistent with the formula - sometimes angles are given for a formula, where signs are opposite to what is represented above.
- check, that rotation angles are given in seconds
- check, that scale factor parameter is given in ppm (1/million), not ppb (1/billion)
- check, that scale factor parameter is given as scale-1, not scale itself

Conversions between datums are based on three coordinates: latitude, longitude and height above ellipsoid. Most file-formats supported by GeoConv don't store height. In conversions missing height causes a minor error, which however can be totally ignored in navigational usage.

An example of DATUM.VAL as is (text file):

```
Name=ToECEF,Ellipsoid,DX,DY,DZ,rx,ry,rz,m
WGS84=1,WGS84,0,0,0,0,0,0,0.0
KKJ=1,HAYFORD,-96.062,-82.428,-121.754,-4.801,-0.345,+1.376,1.496
Finnish Nautical Chart=1,International,-78,-231,-97,0,0,0,0.0
RT90=0,Bessel_1841,-414.0978567149,-41.3381489658,-603.0627177516,-
0.8550434314,+2.1413465185,-7.0227209516,0.0
OSGB36=1,Airy,+446.4,-125.2,+542.1,-0.150,-0.247,-0.842,-20.49
NAD27=1,Clarke_1866,-8,+160,+176,0,0,0,0
```

```
ITRF90=1,GRS80,0.060,-0.517,-0.223,-0.0183,03,0.0070,-0.011
Pulkovo=1,Krassovsky,28,-130,-95,0,0,0,0.0
```

An example of setting datum-parameter values with program FVALUE.

```
FVALUE DATUM.NAD27=1,Clarke_1866,-8,+160,+176,0,0,0,0
FVALUE DATUM.Pulkovo=1,Krassovsky,28,-130,-95,0,0,0,0.0
```

XYPARA.VAL, conversion parameters between geographical and metric xy-coordinates

Parameters in XYPARA.VAL define conversions between geographical coordinates and metric xy-coordinates, e.g. UTM- and Gauss-Krüger- or Lambert Conic Conformal -conversions with different parameters.

Selection of the desired grid-parameter set, a line in the file XYPARA.VAL, is done by setting the value of the parameter INXYID or OUTXYID. E.g. setting `INXYID= SPC CA 5` would select the line below.

```
SPC CA 5=NAD83,W118 00,N33 30,2000000,500000,,M,N34 02,N35 28
```

The order of the data lines or the titles in XYPARA.VAL has no meaning and does not affect the reading of the parameter-values.

The parameter values must be given comma separated in the order described below. If the value of a field is not used, it can be left blank, but the separating commas must still be in place. However, the end of the line after last meaningful value may be left empty.

In the table X indicates a mandatory field.

False Northing and False Easting must be given in the unit given in field UNIT. This unit is also the output-unit of the grid.

Degrees must be given as space separated DEG, DM or DMS format. This format is described in detail in section INCRDTYPE, type 'D'.

Period must be used as the decimal separator, because comma is used as field separator.

No	ID	Name	TM	LCC 1SP	LCC 2SP	MERC 1SP	MERC 2SP
ID	Name	Parameter set ID	X	X	X	X	X
1	Datum	Datum	X	X	X	X	X
2	CEM	Central Meridian = Longitude of Origin	X	X	X	X	X
3	LAO	Latitude of Origin	X		X		
4	FAE	False Easting (in unit defined by UNIT)	X	X	X	X	X
5	FAN	False Northing (in unit defined by UNIT)	X	X	X	X	X
6	SCF	Scale Factor at Central Meridian	X	X	X	X	
7	UNIT	Grid unit, e.g. M or FTUS		X	X	X	X
8	SP1	Standard Parallel 1			X		X
9	SP2	Standard Parallel 2		X	X		
10-	Free text	Free text					

Parameter ID'd starting with UTM are reserved for UTM-handling and can't be used for other purposes.

If the coordinate type is UTM, the parameter values are set automatically by the program, and therefore there is not need to define UTM-parameters in XYPARA.VAL, unless the user wants to use exceptional parameters (other than UTM-standard). In that case the exceptional parameter (e.g. datum), must be defined in XYPARA.VAL, and the parameter ID must be UTM+Zone Number+Band Letter, e.g. UTM03X. E.g. the line below would define, that in UTM-zone 21, band F, Sapper Hill 1943 is used as datum. N/B; that the end of the line after last meaningful field can be left empty.

```
UTM21F=Sapper Hill 1943
```

In coordinate type KKJ all parameters are defined by the program and the parameter setting have no effect.

Datum ID in XYPARA.VAL is referring to parameter-file DATUM.VAL. Therefore it is important to check, that this datum is defined in DATUM.VAL, if you manipulate XYPARA.VAL.

An example of XYPARA.VAL as is (text file):

```
Name=Datum,CEM,LAO,FAE,FAN,SCF,Width, Free text
RT90=RT90, 15.80827777777778, 0, 1500000, 0, 1, M
OSGB36=OSGB36,-2,49,400000,-100000,0.9996012717, M
DE_DHDN-2=DE_DHDN,6,0,2500000,0,1, M
DE_DHDN-3=DE_DHDN,9,0,3500000,0,1, M
DE_DHDN-4=DE_DHDN,12,0,4500000,0,1, M
DE_DHDN-5=DE_DHDN,15,0,5500000,0,1, M
SPC CA 1=NAD83,W122 00,N39 20,2000000,500000,,M,N40 00,N41 40
SPC CA 2=NAD83,W122 00,N37 40,2000000,500000,,M,N38 20,N39 50
SPC CA 3=NAD83,W120 30,N36 30,2000000,500000,,M,N37 04,N38 26
SPC CA 4=NAD83,W119 00,N35 20,2000000,500000,,M,N36 00,N37 15
SPC CA 5=NAD83,W118 00,N33 30,2000000,500000,,M,N34 02,N35 28
NAD27TEXAS=NAD27,99 00 00W,27 50 00N,2000000,0,1,FTUS,28 23 00N,30 17 00N
Morocco1SP=FR_ED50,6GW,,500000,300000,0.999615596,M,33GN
```

An example of setting xy-conversion-parameter values with program FVALUE. Can be used e.g. to set desired datums in fly in the batch run.

```
FVALUE XYPARA.OSGB36=OSGB36,-2,49,400000,-100000,0.9996012717
FVALUE XYPARA.SPC CA 5=NAD83,W118 00,N33 30,2000000,500000,,M,N34 02,N35 28
```

Printable fields

List of the names and properties of the printable fields used in printing templates, that is, in parameters OUTLINEFORM, OUTHEADERFORM and OUTFOOTERFORM.

INDATUM, input-datum
OUTDATUM, output-datum

Input- and output-datum IDs.

INXYID, input-xy-parameter definition ID
OUTXYID, output-xy-parameter definition ID

Input- and output-xy-ID's referring to parameter-file XYPARA.VAL

RTEID, Route/Track-file ID
RTEDESC, Route/Track name / Description

Route/track heading information

ID, Waypoint-ID
DESC, description

ID and description of the point.

NBR, output-line number

Prints the output-line number.

LEGSTART, is start of a track line

1 = point is the first point of a track line block
0 = point is in the middle of a track line block

DATE, date of the point (waypoint, trackpoint, route-point)

In formatting instruction you can use character strings representing different date-elements listed in the table below, e.g.:

```
@date,YYYY-MM-DD@
```

The default date format is YYYY-MM-DD.

YYYY	Year with four digits
YY	Year with two digits
MM	Number of the month (1-12)
DD	Day of the month (1-31)
MMM	Name of the month, three characters, e.g. SEP
DDD	DDD= Day of the year (1-366)
WWWW	WWWW = Year of the week
WW	WW = Week number (1-53)
W	W = Number of the weekday, 1=Mon, 2=Tue,...
WWW	Weekday: Mon, Tue, Wed, ...
Q	Year Quarter (1, 2, 3 or 4)
H	Year Half (1 or 2)

JULIAN, julian date of the point

```
@JULIAN,6.5@
```

DELPHITIME, Date and time in Delphi's TDateTime-format

Date and time in Delphi's TdateTime-format. E.g. OziExplorer uses this format.

TIME, time of the point

In the formatting instruction you can use character strings representing different elements of the time, e.g.:

```
@time,HH:MM:SS.SSS@
```

The default time format is HH:MM.

Elements of time:

`HH`	Hours
`MM`	Minutes
`SS.SSS`	Seconds (number of S-characters may vary)

CDATE, CTIME

Current date and time, usage as in DATE and TIME.

FDATE,FTIME

Date and time from input files timestamp, usage as in DATE and TIME.

INFILE, base part of the input file name

Base part of the input file name. This could be used e.g. in default track-description.

SYMBOL, waypoint symbol or symbol number

ZONE, coordinate zone

Prints the zone-information of the grid-coordinate format's, which use zones.

LAT, NORTHING or X

Prints latitude , northing or X-coordinate

In formats DEG, DM and DMS GeoConv ignores the length of the integer part in formatting instruction.

```
@lat,.4@
```

LON, EASTING or Y

Prints longitude, easting or Y-coordinate

In formats DEG, DM and DMS GeoConv ignores the length of the integer part in formatting instruction.

H, height above ellipsoid, meters

```
@H,3.2@
```

IN2ZONE,IN2LAT, IN2LON, IN2H

Comparison coordinate values from columns IN2LATCOL, IN2LONCOL and IN2HCOL, IN2H in meters.

CX, CY, CZ, 3D-cartesian coordinates, meters

3D-cartesian coordinates X, Y and Z of the output-coordinates.

IN2CX, IN2CY, IN2CZ, 3D-cartesian coordinates from the coordinates in IN2-columns, meters

3D-cartesian coordinates calculated from the coordinates in IN2-columns.

IN2DCX, IN2CDY, IN2CDZ, differences in values of 3D-cartesian coordinates, meters

Differences between the 3D-cartesian coordinates of the IN2-coordinates and converted coordinates. E.g. IN2DCX=IN2CX-CX.

AVEIN2DCX, AVEIN2CDY, AVEIN2CDZ, average differences in values of 3D-cartesian coordinates

Average differences between the 3D-cartesian coordinates of the IN2-coordinates and converted coordinates. E.g. AVEIN2DCX is the average of IN2DCX-values.

IN2DIST (km), IN2BEARING

Distance and bearing between the coordinate values in INLATCOL, INLONCOL, INHCOL and coordinate values in IN2LATCOL, IN2LONCOL, IN2HCOL.

MAXIN2DIST, AVEIN2DIST, km

Maximum and average value of IN2DIST.

ALTI, Altitude in meters

Height above the geoid. Not same as height above ellipsoid but gets often the same value due to poor or limited information got from applications and receivers.

WGS84h, height above WGS84-ellipsoid, meters

Part of NMEA0183-information. Often missing or inaccurate.

DIST, distance from the previous point, breaks included, km

Calculated regardless of the legstart, that is, breaks included (see field LEGSTART).

TRKDIST, accumulated length of the track line, breaks excluded, km

Accumulated length of the track line, only continuous parts included and breaks excluded.

TOTDIST, accumulated length of the track line, breaks included, km

Accumulated length of the track line, also breaks included as straight lines.

HOURS, time difference in hours between this and previous time stamp, breaks included

Time in hours, also breaks included

TRKHOURS, accumulated time, breaks excluded

Accumulated time, breaks excluded

TOTHOURS, accumulated time, also breaks included

Accumulated time, also breaks included

SPEED, calculated , km/h

Speed km/h calculated by the distance and time from previous point

MAXSPEED, MINSPEED, AVESPEED, km/h

Maximum-, minimum- and average speed. Breaks excluded.

BEARING, bearing from previous point, calculated

Calculated regardless of the legstart, that is, breaks included (see field LEGSTART).

HDOP, Horizontal Dilution Of Position

FIXQ, GPS Fix-quality according to NMEA-standard

0 = no fix / invalid
1 = GPS-fix

2 = DGPS-fix

SATNUM, number of fixed satellites

FROMLAT, latitude of FROMPOINT
FROMLON, longitude of FROMPOINT

FROMDIST, distance from FROMPOINT
FROMBEARING, bearing from FROMPOINT

REDUCEMAXDIST, final deviation

Final maximum deviation after reduction.

ITEM, item of the input-line

E.g. if the line is
`KARI,NAUVO,60.2,24.5`
gives `ITEM,2`
`NAUVO` as result

INHEIGHTUNIT
OUTSPEEDUNIT
OUTHEIGHTUNIT
OUTDURATIONUNIT
OUTDISTANCEUNIT

Units of measures set with corresponding parameters. See section "Unit conversion parameters".

OUTTIMEDIFF

Time difference set with corresponding parameter. See section "Unit conversion parameters".