

My Project

Generated by Doxygen 1.8.6

Tue Jul 28 2015 13:17:56

Contents

1	Namespace Index	1
1.1	Namespace List	1
2	Hierarchical Index	3
2.1	Class Hierarchy	3
3	Class Index	5
3.1	Class List	5
4	File Index	7
4.1	File List	7
5	Namespace Documentation	9
5.1	AutoTest Namespace Reference	9
5.1.1	Typedef Documentation	9
5.1.1.1	TestList	9
5.1.2	Function Documentation	9
5.1.2.1	addTest	9
5.1.2.2	findObject	10
5.1.2.3	run	10
5.1.2.4	testList	10
6	Class Documentation	11
6.1	Coords Class Reference	11
6.1.1	Detailed Description	11
6.1.2	Constructor & Destructor Documentation	11
6.1.2.1	Coords	11
6.1.2.2	Coords	12
6.1.3	Member Function Documentation	12
6.1.3.1	DegToDec	12
6.1.3.2	getDecLatitude	12
6.1.3.3	getDecLongitude	12
6.1.3.4	getLatitudeD	12

6.1.3.5	getLatitudeM	12
6.1.3.6	getLatitudeS	12
6.1.3.7	getLongitudeD	12
6.1.3.8	getLongitudeM	12
6.1.3.9	getLongitudeS	13
6.1.3.10	setDecLatitude	13
6.1.3.11	setDecLongitude	13
6.1.3.12	setLatitudeD	13
6.1.3.13	setLatitudeM	13
6.1.3.14	setLatitudeS	13
6.1.3.15	setLongitudeD	13
6.1.3.16	setLongitudeM	13
6.1.3.17	setLongitudeS	13
6.2	Radar Class Reference	13
6.2.1	Detailed Description	14
6.2.2	Constructor & Destructor Documentation	14
6.2.2.1	Radar	14
6.2.3	Member Function Documentation	14
6.2.3.1	getDistance	14
6.2.4	Member Data Documentation	14
6.2.4.1	directDistance	14
6.2.4.2	otherPoz	15
6.2.4.3	sac	15
6.2.4.4	selfPoz	15
6.3	Test< T > Class Template Reference	15
6.3.1	Constructor & Destructor Documentation	15
6.3.1.1	Test	15
6.3.2	Member Data Documentation	15
6.3.2.1	child	15
6.4	TestCoord Class Reference	16
6.4.1	Constructor & Destructor Documentation	16
6.4.1.1	TestCoord	16
6.5	TestRadar Class Reference	16
6.5.1	Constructor & Destructor Documentation	17
6.5.1.1	TestRadar	17
7	File Documentation	19
7.1	AutoTest.h File Reference	19
7.1.1	Macro Definition Documentation	20
7.1.1.1	DECLARE_TEST	20

7.1.1.2	TEST_MAIN	20
7.2	coords.cpp File Reference	21
7.2.1	Detailed Description	21
7.2.2	Function Documentation	21
7.2.2.1	sgn	21
7.3	coords.h File Reference	21
7.4	main.cpp File Reference	22
7.4.1	Function Documentation	23
7.4.1.1	main	23
7.5	radar.cpp File Reference	24
7.6	radar.h File Reference	24
7.7	test_coord.cpp File Reference	25
7.7.1	Function Documentation	26
7.7.1.1	cutNumber	26
7.7.2	Variable Documentation	26
7.7.2.1	cutDecLat	26
7.7.2.2	cutDecLong	26
7.7.2.3	mich_p37	26
7.7.2.4	mie_mor	26
7.7.2.5	mssr_bucen	26
7.7.2.6	mssr_veljav	26
7.7.2.7	po_mor5	26
7.8	test_coord.h File Reference	26
7.9	test_radar.cpp File Reference	27
7.9.1	Variable Documentation	27
7.9.1.1	mich__p37	27
7.9.1.2	mie__mor	28
7.9.1.3	mssr__bucen	28
7.9.1.4	mssr__veljav	28
7.9.1.5	po__mor5	28
7.9.1.6	r1	28
7.10	test_radar.h File Reference	28

Chapter 1

Namespace Index

1.1 Namespace List

Here is a list of all namespaces with brief descriptions:

AutoTest	9
------------------------------------	---

Chapter 2

Hierarchical Index

2.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

Coords	11
QObject	
TestCoord	16
TestRadar	16
Radar	13
Test< T >	15

Chapter 3

Class Index

3.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

Coords		
	Trieda Coords	11
Radar		
	Trieda Radar	13
Test< T >	15
TestCoord	16
TestRadar	16

Chapter 4

File Index

4.1 File List

Here is a list of all files with brief descriptions:

AutoTest.h	19
coords.cpp	
Tento subor obsahuje konštruktor a metódy triedy Coords predstavujúce súradnice radarového systému	
coords.h	21
main.cpp	22
radar.cpp	24
radar.h	24
test_coord.cpp	25
test_coord.h	26
test_radar.cpp	27
test_radar.h	28

Chapter 5

Namespace Documentation

5.1 AutoTest Namespace Reference

Typedefs

- typedef QList< QObject * > [TestList](#)

Functions

- [TestList](#) & [testList](#) ()
- bool [findObject](#) (QObject *object)
- void [addTest](#) (QObject *object)
- int [run](#) (int argc, char *argv[])

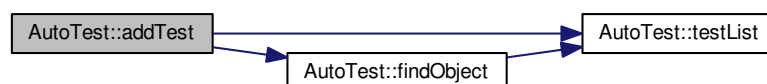
5.1.1 Typedef Documentation

5.1.1.1 typedef QList<QObject*> **AutoTest::TestList**

5.1.2 Function Documentation

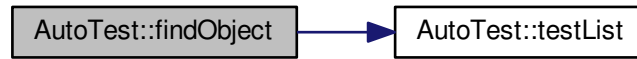
5.1.2.1 void AutoTest::addTest (QObject * *object*) [inline]

Here is the call graph for this function:



5.1.2.2 `bool AutoTest::findObject (QObject * object)` `[inline]`

Here is the call graph for this function:



5.1.2.3 `int AutoTest::run (int argc, char * argv[])` `[inline]`

Here is the call graph for this function:



5.1.2.4 `TestList& AutoTest::testList ()` `[inline]`

Chapter 6

Class Documentation

6.1 Coords Class Reference

Trieda [Coords](#).

```
#include <coords.h>
```

Public Member Functions

- [Coords](#) ()
- [Coords](#) (double, double, double, double, double, double)
- void [setLatitudeD](#) (double)
- void [setLatitudeM](#) (double)
- void [setLatitudeS](#) (double)
- void [setLongitudeD](#) (double)
- void [setLongitudeM](#) (double)
- void [setLongitudeS](#) (double)
- void [setDecLatitude](#) (double decLatitude)
- void [setDecLongitude](#) (double decLongitude)
- double [getLatitudeD](#) (void)
- double [getLatitudeM](#) (void)
- double [getLatitudeS](#) (void)
- double [getLongitudeD](#) (void)
- double [getLongitudeM](#) (void)
- double [getLongitudeS](#) (void)
- double [getDecLatitude](#) ()
- double [getDecLongitude](#) ()
- void [DegToDec](#) (void)

6.1.1 Detailed Description

Trieda [Coords](#).

Trieda [Coords](#) predstavuje suradnice radaroveho systemu a ich prevod do roznych typov.

6.1.2 Constructor & Destructor Documentation

6.1.2.1 Coords::Coords ()

Implicitny konštruktor

6.1.2.2 Coords::Coords (double *latitudeD*, double *latitudeM*, double *latitudeS*, double *longitudeD*, double *longitudeM*, double *longitudeS*)

Konstruktor triedy [Coords](#) obsahujuci vstupne suradnice.

Konstruktor ma ako vstupne argumenty sestticu cisel predstavujucich stupne minuty a sekundy zemepisnej sirky a dlzky.

Parameters

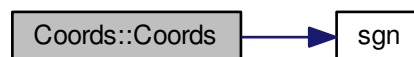
<i>arraySize</i>	size of dynamically allocated array
------------------	-------------------------------------

Returns

nothing

- < Informacia o stupni zemepisnej sirky
- < Informacia o minute zemepisnej sirky
- < Informacia o sekunde zemepisnej sirky
- < Informacia o stupni zemepisnej dlzky
- < Informacia o stupni zemepisnej dlzky
- < Informacia o stupni zemepisnej dlzky
- < Prevod zemepisnej sirky do dekadickych stupnov
- < Prevod zemepisnej dlzky do dekadickych stupnov

Here is the call graph for this function:



6.1.3 Member Function Documentation

6.1.3.1 void Coords::DegToDec (void)

6.1.3.2 double Coords::getDecLatitude ()

6.1.3.3 double Coords::getDecLongitude ()

6.1.3.4 double Coords::getLatitudeD (void)

6.1.3.5 double Coords::getLatitudeM (void)

6.1.3.6 double Coords::getLatitudeS (void)

6.1.3.7 double Coords::getLongitudeD (void)

6.1.3.8 double Coords::getLongitudeM (void)

- 6.1.3.9 double Coords::getLongitudeS (void)
- 6.1.3.10 void Coords::setDecLatitude (double *decLatitude*)
- 6.1.3.11 void Coords::setDecLongitude (double *decLongitude*)
- 6.1.3.12 void Coords::setLatitudeD (double *latitudeD*)
- 6.1.3.13 void Coords::setLatitudeM (double *latitudeM*)
- 6.1.3.14 void Coords::setLatitudeS (double *latitudeS*)
- 6.1.3.15 void Coords::setLongitudeD (double *longitudeD*)
- 6.1.3.16 void Coords::setLongitudeM (double *longitudeM*)
- 6.1.3.17 void Coords::setLongitudeS (double *longitudeS*)

The documentation for this class was generated from the following files:

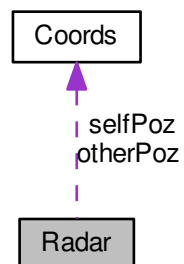
- [coords.h](#)
- [coords.cpp](#)

6.2 Radar Class Reference

Trieda [Radar](#).

```
#include <radar.h>
```

Collaboration diagram for Radar:



Public Member Functions

- [Radar](#) ([Coords](#), [Coords](#))
- double [getDistance](#) (void)

Public Attributes

- const double [sac](#) =48.0

- double [directDistance](#)
- [Coords](#) [selfPoz](#)
- [Coords](#) [otherPoz](#)

6.2.1 Detailed Description

Trieda [Radar](#).

Trieda predstavuje model radaru.

6.2.2 Constructor & Destructor Documentation

6.2.2.1 [Radar::Radar](#) ([Coords](#) [selfPoz](#), [Coords](#) [otherPoz](#))

Konstruktor triedy [Radar](#) obsahujuci vlastnu a vzdialenu suradnicu.

Konstruktor ma ako vstupne argumenty dva parametre typu [Coords](#).

Parameters

<i>arraySize</i>	size of dynamically allocated array
------------------	-------------------------------------

Returns

nothing

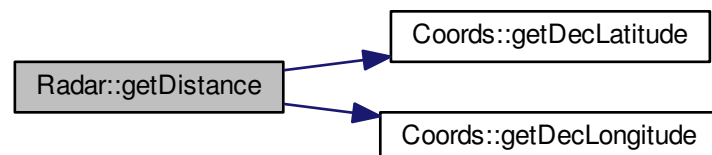
6.2.3 Member Function Documentation

6.2.3.1 double [Radar::getDistance](#) (void)

Funkcia [getDistance](#) triedy [Radar](#) pocitajuca vzdialenost.

Funkcia pocita vzdialenost pomocou kniznice GeographicLib vytvorenu pre geograficke vypocty.

Here is the call graph for this function:



6.2.4 Member Data Documentation

6.2.4.1 double [Radar::directDistance](#)

Vzdialenost od inej wgs suradnice

6.2.4.2 Coords Radar::otherPoz

Pozicia od ktorej sa pocita vzdialenost

6.2.4.3 const double Radar::sac =48.0

SAC identifikator

6.2.4.4 Coords Radar::selfPoz

Vlastna pozicia radaru

The documentation for this class was generated from the following files:

- [radar.h](#)
- [radar.cpp](#)

6.3 Test< T > Class Template Reference

```
#include <AutoTest.h>
```

Public Member Functions

- [Test](#) (const QString &name)

Public Attributes

- QSharedPointer< T > [child](#)

6.3.1 Constructor & Destructor Documentation

6.3.1.1 template<class T > Test< T >::Test (const QString & name) [inline]

Here is the call graph for this function:



6.3.2 Member Data Documentation

6.3.2.1 template<class T > QSharedPointer<T> Test< T >::child

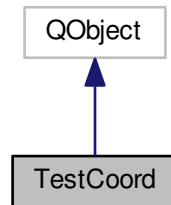
The documentation for this class was generated from the following file:

- [AutoTest.h](#)

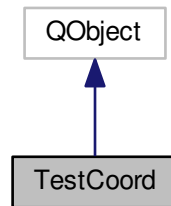
6.4 TestCoord Class Reference

```
#include <test_coord.h>
```

Inheritance diagram for TestCoord:



Collaboration diagram for TestCoord:



Public Member Functions

- [TestCoord\(\)](#)

6.4.1 Constructor & Destructor Documentation

6.4.1.1 TestCoord::TestCoord ()

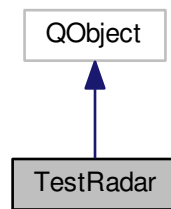
The documentation for this class was generated from the following files:

- [test_coord.h](#)
- [test_coord.cpp](#)

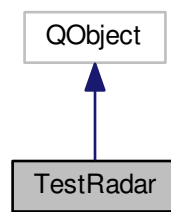
6.5 TestRadar Class Reference

```
#include <test_radar.h>
```

Inheritance diagram for TestRadar:



Collaboration diagram for TestRadar:



Public Member Functions

- [TestRadar](#) ()

6.5.1 Constructor & Destructor Documentation

6.5.1.1 TestRadar::TestRadar ()

The documentation for this class was generated from the following files:

- [test_radar.h](#)
- [test_radar.cpp](#)

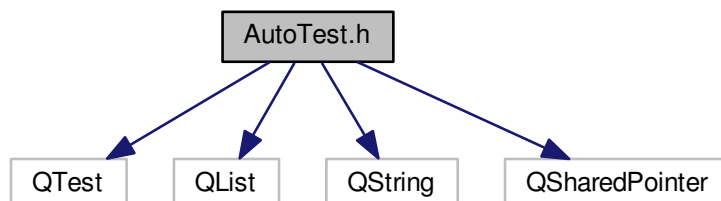
Chapter 7

File Documentation

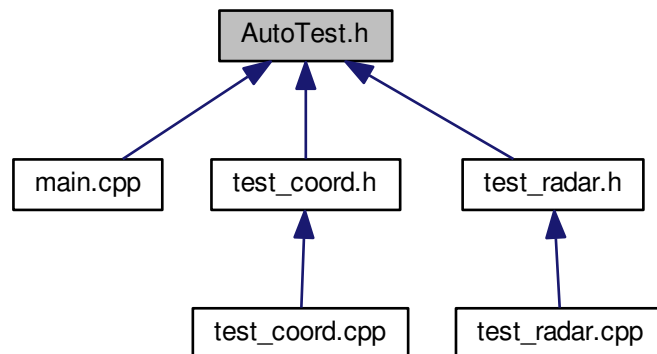
7.1 AutoTest.h File Reference

```
#include <QTest>
#include <QList>
#include <QString>
#include <QSharedPointer>
```

Include dependency graph for AutoTest.h:



This graph shows which files directly or indirectly include this file:



Classes

- class [Test< T >](#)

Namespaces

- [AutoTest](#)

Macros

- `#define DECLARE_TEST(className) static Test<className> t(#className);`
- `#define TEST_MAIN`

Typedefs

- `typedef QList< QObject * > AutoTest::TestList`

Functions

- `TestList & AutoTest::testList ()`
- `bool AutoTest::findObject (QObject *object)`
- `void AutoTest::addTest (QObject *object)`
- `int AutoTest::run (int argc, char *argv[])`

7.1.1 Macro Definition Documentation

7.1.1.1 `#define DECLARE_TEST(className) static Test<className> t(#className);`

7.1.1.2 `#define TEST_MAIN`

Value:

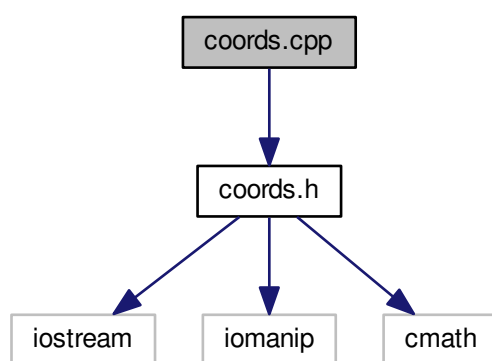
```
int main(int argc, char *argv[]) \
{ \
    return AutoTest::run(argc, argv); \
}
```

7.2 coords.cpp File Reference

Tento subor obsahuje konštruktor a metódy triedy [Coords](#) predstavujúce súradnice radarového systému.

```
#include "coords.h"
```

Include dependency graph for coords.cpp:



Functions

- `template<typename T >`
`T sgn (T n)`

7.2.1 Detailed Description

Tento subor obsahuje konštruktor a metódy triedy [Coords](#) predstavujúce súradnice radarového systému. Možno obsahlejší popis na viacerých radoch.

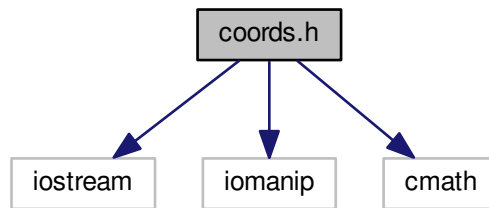
7.2.2 Function Documentation

7.2.2.1 `template<typename T > T sgn (T n)`

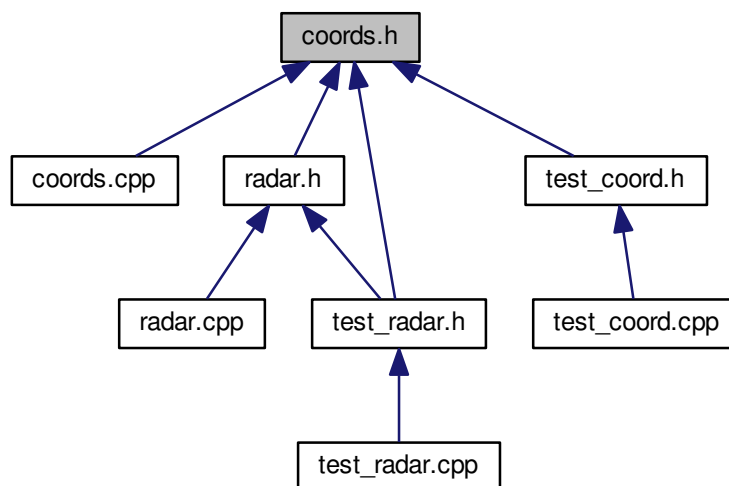
7.3 coords.h File Reference

```
#include <iostream>
#include <iomanip>
#include <cmath>
```

Include dependency graph for coords.h:



This graph shows which files directly or indirectly include this file:



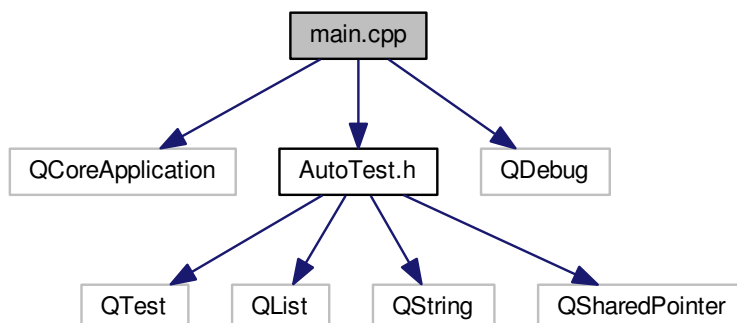
Classes

- class [Coords](#)
Trieda [Coords](#).

7.4 main.cpp File Reference

```
#include <QCoreApplication>
#include "AutoTest.h"
#include <QDebug>
```

Include dependency graph for main.cpp:



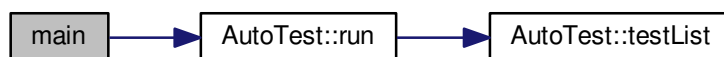
Functions

- int [main](#) (int argc, char *argv[])

7.4.1 Function Documentation

7.4.1.1 int main (int argc, char * argv[])

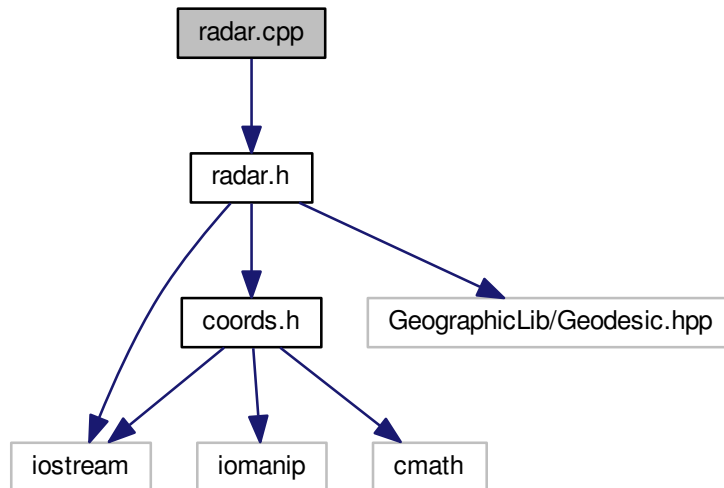
Here is the call graph for this function:



7.5 radar.cpp File Reference

```
#include "radar.h"
```

Include dependency graph for radar.cpp:



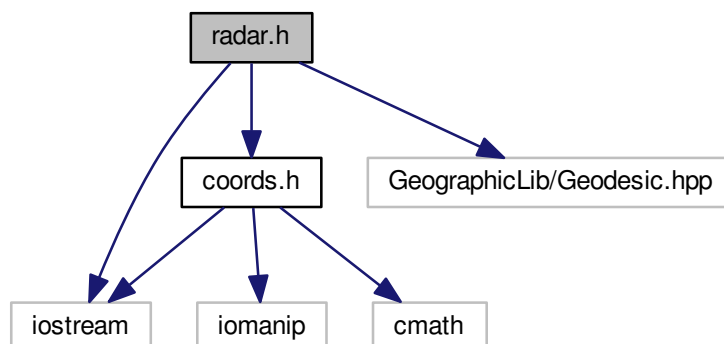
7.6 radar.h File Reference

```
#include "coords.h"
```

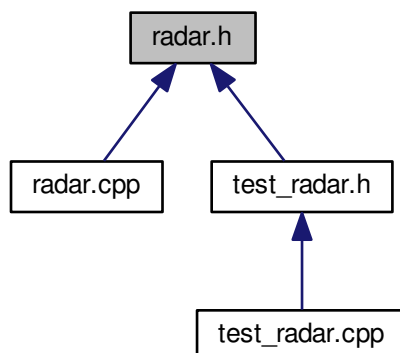
```
#include <iostream>
```

```
#include <GeographicLib/Geodesic.hpp>
```

Include dependency graph for radar.h:



This graph shows which files directly or indirectly include this file:



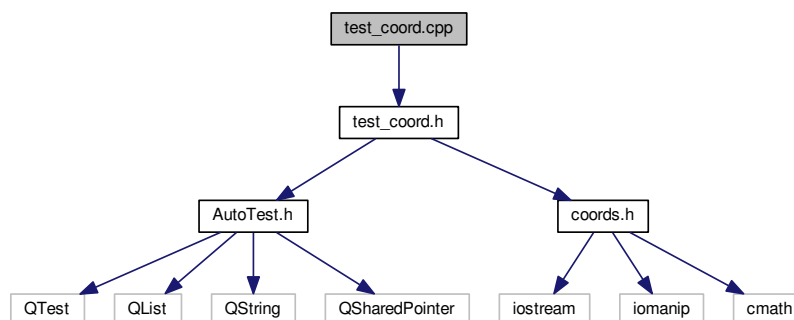
Classes

- class [Radar](#)
Trieda [Radar](#).

7.7 test_coord.cpp File Reference

```
#include "test_coord.h"
```

Include dependency graph for `test_coord.cpp`:



Functions

- double [cutNumber](#) (double number)

Variables

- double [cutDeclat](#)

- double [cutDecLong](#)
- [Coords mssr_veljav](#) (48, 15, 38.56, 17, 9, 47.83)
- [Coords mssr_bucen](#) (48, 18, 19.95, 19, 52, 14.61)
- [Coords mie_mor](#) (48, 3, 42.00, 17, 22, 35.00)
- [Coords mich_p37](#) (48, 43, 24.98, 21, 57, 4.39)
- [Coords po_mor5](#) (49, 1, 55.2701, 21, 18, 51.607)

7.7.1 Function Documentation

7.7.1.1 double `cutNumber` (double *number*)

7.7.2 Variable Documentation

7.7.2.1 double `cutDecLat`

7.7.2.2 double `cutDecLong`

7.7.2.3 `Coords mich_p37`(48, 43, 24.98, 21, 57, 4.39)

7.7.2.4 `Coords mie_mor`(48, 3, 42.00, 17, 22, 35.00)

7.7.2.5 `Coords mssr_bucen`(48, 18, 19.95, 19, 52, 14.61)

7.7.2.6 `Coords mssr_veljav`(48, 15, 38.56, 17, 9, 47.83)

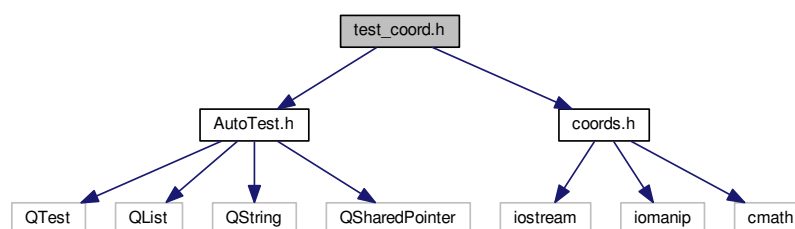
7.7.2.7 `Coords po_mor5`(49, 1, 55.2701, 21, 18, 51.607)

7.8 test_coord.h File Reference

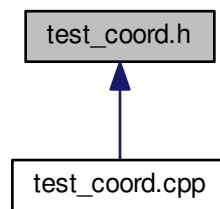
```
#include "AutoTest.h"
```

```
#include "coords.h"
```

Include dependency graph for test_coord.h:



This graph shows which files directly or indirectly include this file:



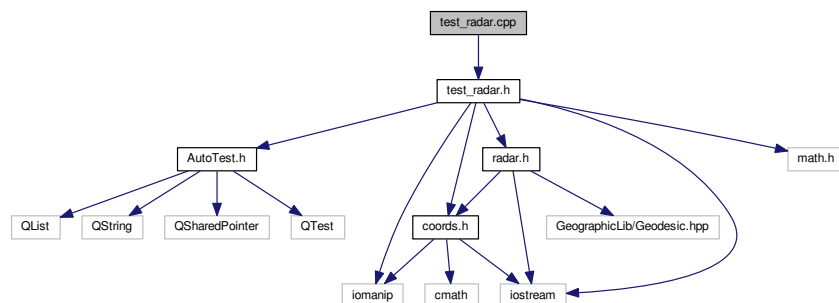
Classes

- class [TestCoord](#)

7.9 test_radar.cpp File Reference

```
#include "test_radar.h"
```

Include dependency graph for test_radar.cpp:



Variables

- [Coords mssr__veljav](#) (48, 15, 38.56, 17, 9, 47.83)
- [Coords mssr__bucen](#) (48, 18, 19.95, 19, 52, 14.61)
- [Coords mie__mor](#) (48, 3, 42.00, 17, 22, 35.00)
- [Coords mich__p37](#) (48, 43, 24.98, 21, 57, 4.39)
- [Coords po__mor5](#) (49, 1, 55.2701, 21, 18, 51.607)
- [Radar r1](#) ([mssr__veljav](#), [mssr__bucen](#))

7.9.1 Variable Documentation

7.9.1.1 Coords mich__p37(48, 43, 24.98, 21, 57, 4.39)

7.9.1.2 Coords mie__mor(48, 3, 42.00, 17, 22, 35.00)

7.9.1.3 Coords mssr__bucen(48, 18, 19.95, 19, 52, 14.61)

7.9.1.4 Coords mssr__veljav(48, 15, 38.56, 17, 9, 47.83)

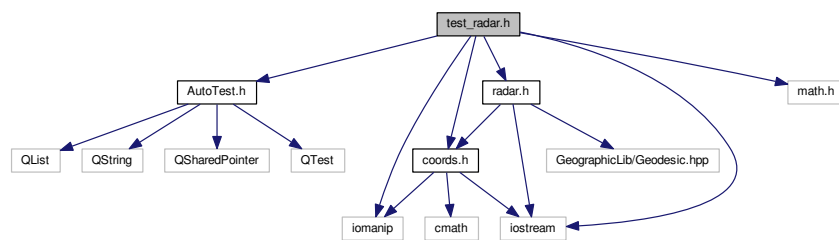
7.9.1.5 Coords po__mor5(49, 1, 55.2701, 21, 18, 51.607)

7.9.1.6 Radar r1(mssr__veljav, mssr__bucen)

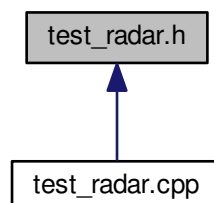
7.10 test_radar.h File Reference

```
#include "AutoTest.h"
#include "coords.h"
#include "radar.h"
#include <math.h>
#include <iomanip>
#include <iostream>
```

Include dependency graph for test_radar.h:



This graph shows which files directly or indirectly include this file:



Classes

- class [TestRadar](#)

Index

- addTest
 - AutoTest, 9
- AutoTest, 9
 - addTest, 9
 - findObject, 9
 - run, 10
 - TestList, 9
 - testList, 10
- AutoTest.h, 19
 - DECLARE_TEST, 20
 - TEST_MAIN, 20

- child
 - Test, 15
- Coords, 11
 - Coords, 11
 - DegToDec, 12
 - getDecLatitude, 12
 - getDecLongitude, 12
 - getLatitudeD, 12
 - getLatitudeM, 12
 - getLatitudeS, 12
 - getLongitudeD, 12
 - getLongitudeM, 12
 - getLongitudeS, 12
 - setDecLatitude, 13
 - setDecLongitude, 13
 - setLatitudeD, 13
 - setLatitudeM, 13
 - setLatitudeS, 13
 - setLongitudeD, 13
 - setLongitudeM, 13
 - setLongitudeS, 13
- coords.cpp, 21
 - sgn, 21
- coords.h, 21
- cutDecLat
 - test_coord.cpp, 26
- cutDecLong
 - test_coord.cpp, 26
- cutNumber
 - test_coord.cpp, 26

- DECLARE_TEST
 - AutoTest.h, 20
- DegToDec
 - Coords, 12
- directDistance
 - Radar, 14

- findObject
 - AutoTest, 9
- getDecLatitude
 - Coords, 12
- getDecLongitude
 - Coords, 12
- getDistance
 - Radar, 14
- getLatitudeD
 - Coords, 12
- getLatitudeM
 - Coords, 12
- getLatitudeS
 - Coords, 12
- getLongitudeD
 - Coords, 12
- getLongitudeM
 - Coords, 12
- getLongitudeS
 - Coords, 12
- main
 - main.cpp, 23
- main.cpp, 22
 - main, 23
- mich__p37
 - test_radar.cpp, 27
- mich_p37
 - test_coord.cpp, 26
- mie__mor
 - test_radar.cpp, 27
- mie_mor
 - test_coord.cpp, 26
- mssr__bucen
 - test_radar.cpp, 28
- mssr__veljav
 - test_radar.cpp, 28
- mssr_bucen
 - test_coord.cpp, 26
- mssr_veljav
 - test_coord.cpp, 26
- otherPoz
 - Radar, 14
- po__mor5
 - test_radar.cpp, 28
- po_mor5
 - test_coord.cpp, 26

- r1
 - test_radar.cpp, 28
- Radar, 13
 - directDistance, 14
 - getDistance, 14
 - otherPoz, 14
 - Radar, 14
 - sac, 15
 - selfPoz, 15
- radar.cpp, 24
- radar.h, 24
- run
 - AutoTest, 10
- sac
 - Radar, 15
- selfPoz
 - Radar, 15
- setDecLatitude
 - Coords, 13
- setDecLongitude
 - Coords, 13
- setLatitudeD
 - Coords, 13
- setLatitudeM
 - Coords, 13
- setLatitudeS
 - Coords, 13
- setLongitudeD
 - Coords, 13
- setLongitudeM
 - Coords, 13
- setLongitudeS
 - Coords, 13
- sgn
 - coords.cpp, 21
- TEST_MAIN
 - AutoTest.h, 20
- Test
 - child, 15
 - Test, 15
- Test< T >, 15
- test_coord.cpp, 25
 - cutDecLat, 26
 - cutDecLong, 26
 - cutNumber, 26
 - mich_p37, 26
 - mie_mor, 26
 - mssr_bucen, 26
 - mssr_veljav, 26
 - po_mor5, 26
- test_coord.h, 26
- test_radar.cpp, 27
 - mich__p37, 27
 - mie__mor, 27
 - mssr__bucen, 28
 - mssr__veljav, 28
 - po__mor5, 28
- r1, 28
 - test_radar.h, 28
 - TestCoord, 16
 - TestCoord, 16
 - TestCoord, 16
 - TestList
 - AutoTest, 9
 - testList
 - AutoTest, 10
 - TestRadar, 16
 - TestRadar, 17
 - TestRadar, 17