

## QMAST Source Code

Generated by Doxygen 1.7.5.1

Fri Sep 30 2011 14:56:53



# Contents

<b>1</b>	<b>The QMAST Alpha 6 Sailing Code</b>	<b>1</b>
<b>2</b>	<b>Data Structure Index</b>	<b>3</b>
2.1	Data Structures . . . . .	3
<b>3</b>	<b>File Index</b>	<b>5</b>
3.1	File List . . . . .	5
<b>4</b>	<b>Data Structure Documentation</b>	<b>7</b>
4.1	points Struct Reference . . . . .	7
4.1.1	Field Documentation . . . . .	7
4.1.1.1	latDeg . . . . .	7
4.1.1.2	latMin . . . . .	7
4.1.1.3	lonDeg . . . . .	7
4.1.1.4	lonMin . . . . .	7
<b>5</b>	<b>File Documentation</b>	<b>9</b>
5.1	/Users/allgood38/Desktop/qmast/sailcode_alpha6/DataAcquisition.pde File Reference . . . . .	9
5.1.1	Function Documentation . . . . .	9
5.1.1.1	checkErrorBit . . . . .	9
5.1.1.2	clearErrorBit . . . . .	9
5.1.1.3	sensorData . . . . .	9
5.1.1.4	setErrorBit . . . . .	9
5.2	/Users/allgood38/Desktop/qmast/sailcode_alpha6/LocationStruct.h File Reference . . . . .	9
5.2.1	Variable Documentation . . . . .	10

5.2.1.1	<a href="#">boatLocation</a>	10
5.2.1.2	<a href="#">clearPoints</a>	10
5.2.1.3	<a href="#">coursePoints</a>	10
5.2.1.4	<a href="#">floatingStationPoints</a>	10
5.2.1.5	<a href="#">stationPoints</a>	10
5.2.1.6	<a href="#">stayPoint</a>	10
5.2.1.7	<a href="#">waypoints</a>	10
5.3	<a href="#">/Users/allgood38/Desktop/qmast/sailcode_alpha6/Menu.pde</a> File - Reference	11
5.3.1	Function Documentation	11
5.3.1.1	<a href="#">displayMenu</a>	11
5.4	<a href="#">/Users/allgood38/Desktop/qmast/sailcode_alpha6/MotorControl- Functions.pde</a> File Reference	11
5.4.1	Function Documentation	11
5.4.1.1	<a href="#">servo_command</a>	11
5.4.1.2	<a href="#">setJib</a>	11
5.4.1.3	<a href="#">setMain</a>	11
5.4.1.4	<a href="#">setrudder</a>	11
5.4.1.5	<a href="#">setSails</a>	11
5.5	<a href="#">/Users/allgood38/Desktop/qmast/sailcode_alpha6/NavigationCode.pde</a> File Reference	11
5.5.1	Function Documentation	12
5.5.1.1	<a href="#">getCloseHauledDirn</a>	12
5.5.1.2	<a href="#">getOppositeCloseHauledDirn</a>	12
5.5.1.3	<a href="#">getWaypointDirn</a>	12
5.5.1.4	<a href="#">getWindDirn</a>	12
5.5.1.5	<a href="#">GPSdistance</a>	12
5.6	<a href="#">/Users/allgood38/Desktop/qmast/sailcode_alpha6/oldtack.pde</a> File - Reference	12
5.6.1	Function Documentation	12
5.6.1.1	<a href="#">getOutofIronsOld</a>	12
5.6.1.2	<a href="#">oldtack</a>	12
5.6.1.3	<a href="#">oldtack2</a>	12
5.7	<a href="#">/Users/allgood38/Desktop/qmast/sailcode_alpha6/ParsingFunctions.pde</a> File Reference	12

5.7.1	Function Documentation	13
5.7.1.1	DataValid	13
5.7.1.2	ParseGPGLL	13
5.7.1.3	Parser	13
5.8	/Users/allgood38/Desktop/qmast/sailcode_alpha6/sailcode_alpha6.pde File Reference	13
5.8.1	Define Documentation	15
5.8.1.1	ALL_IN	15
5.8.1.2	ALL_OUT	15
5.8.1.3	badCompassDataBit	15
5.8.1.4	badGpsData	15
5.8.1.5	badWindData	15
5.8.1.6	BUFF_MAX	15
5.8.1.7	checksumBadBit	15
5.8.1.8	DEGREE_TO_MINUTE	15
5.8.1.9	LATITUDE_TO_METER	15
5.8.1.10	LONGEST_NMEA	15
5.8.1.11	LONGITUDE_TO_METER	15
5.8.1.12	MARK_DISTANCE	15
5.8.1.13	noDataBit	15
5.8.1.14	oldDataBit	15
5.8.1.15	resetPin	15
5.8.1.16	rolloverDataBit	15
5.8.1.17	SHORTEST_NMEA	15
5.8.1.18	STATION_KEEPING_RADIUS	15
5.8.1.19	TACKING_ANGLE	15
5.8.1.20	tooMuchRollBit	15
5.8.1.21	twoCommasBit	15
5.8.1.22	txPin	16
5.8.1.23	WIND_CHANGE_THRESHOLD	16
5.8.2	Function Documentation	16
5.8.2.1	loop	16
5.8.2.2	setup	16
5.8.3	Variable Documentation	16

5.8.3.1	<a href="#">bspeed</a>	16
5.8.3.2	<a href="#">bspeedk</a>	16
5.8.3.3	<a href="#">currentPoint</a>	16
5.8.3.4	<a href="#">CurrentSelection</a>	16
5.8.3.5	<a href="#">deviation</a>	16
5.8.3.6	<a href="#">distanceVal</a>	16
5.8.3.7	<a href="#">errorCode</a>	16
5.8.3.8	<a href="#">extraCompassData</a>	16
5.8.3.9	<a href="#">extraCompassDataArray</a>	16
5.8.3.10	<a href="#">extraWindData</a>	16
5.8.3.11	<a href="#">extraWindDataArray</a>	17
5.8.3.12	<a href="#">heading</a>	17
5.8.3.13	<a href="#">heading_newest</a>	17
5.8.3.14	<a href="#">headingc</a>	17
5.8.3.15	<a href="#">headingVal</a>	17
5.8.3.16	<a href="#">ironTime</a>	17
5.8.3.17	<a href="#">jibVal</a>	17
5.8.3.18	<a href="#">mainVal</a>	17
5.8.3.19	<a href="#">pitch</a>	17
5.8.3.20	<a href="#">point</a>	17
5.8.3.21	<a href="#">points</a>	17
5.8.3.22	<a href="#">roll</a>	17
5.8.3.23	<a href="#">rudderDir</a>	17
5.8.3.24	<a href="#">rudderVal</a>	17
5.8.3.25	<a href="#">savedCompassChecksum</a>	17
5.8.3.26	<a href="#">savedCompassXorState</a>	17
5.8.3.27	<a href="#">savedWindChecksum</a>	17
5.8.3.28	<a href="#">savedWindXorState</a>	17
5.8.3.29	<a href="#">servo_ser</a>	17
5.8.3.30	<a href="#">startTime</a>	17
5.8.3.31	<a href="#">stationCounter</a>	17
5.8.3.32	<a href="#">StationKeepingTimeInBox</a>	17
5.8.3.33	<a href="#">StraightSailDirection</a>	17
5.8.3.34	<a href="#">tacking</a>	17

5.8.3.35	tackingSide	18
5.8.3.36	timesUp	18
5.8.3.37	trueWind	18
5.8.3.38	variance	18
5.8.3.39	wind_angl	18
5.8.3.40	wind_angl_newest	18
5.8.3.41	wind_velocity	18
5.9	/Users/allgood38/Desktop/qmast/sailcode_alpha6/SailingLogic.pde File Reference	18
5.9.1	Function Documentation	18
5.9.1.1	checkTack	18
5.9.1.2	rudderControl	18
5.9.1.3	sail	18
5.9.1.4	sailControl	18
5.9.1.5	sailCourse	18
5.9.1.6	sailToWaypoint	18
5.10	/Users/allgood38/Desktop/qmast/sailcode_alpha6/StationKeeping.pde - File Reference	18
5.10.1	Function Documentation	19
5.10.1.1	fillStationKeepingWaypoints	19
5.10.1.2	getStationKeepingCentre	19
5.10.1.3	stationKeep	19
5.10.1.4	stationKeepSinglePoint	19
5.11	/Users/allgood38/Desktop/qmast/sailcode_alpha6/Tack.pde File - Reference	19
5.11.1	Function Documentation	19
5.11.1.1	getOutofIrons	19
5.11.1.2	tack	19
5.12	/Users/allgood38/Desktop/qmast/sailcode_alpha6/Testing_Functions.pde File Reference	19
5.13	/Users/allgood38/Desktop/qmast/sailcode_alpha6/Transmit.pde File - Reference	19
5.13.1	Function Documentation	20
5.13.1.1	relayData	20
5.13.1.2	transmit	20

5.14	<a href="#">/Users/allgood38/Desktop/qmast/sailcode_alpha6/Utilities.pde</a>	<a href="#">File</a>	-	
	<a href="#">Reference</a>			20
5.14.1	<a href="#">Function Documentation</a>			20
5.14.1.1	<a href="#">between</a>			20
5.14.1.2	<a href="#">convertASCIItoHex</a>			20
5.14.1.3	<a href="#">degreesToRadians</a>			20
5.14.1.4	<a href="#">radiansToDegrees</a>			20



## Chapter 1

# The QMAST Alpha 6 Sailing Code

### **This Documentation**

Is the same style as the JavaDoc documentation. The same commands are used in the code, and generates these pages. For info on the program used check out a program called Doxygen

### **Revised by Laszlo 2011-05-13**

Ported to Arudino November 2010 by Christine and the supercool software team  
Created on: 2010-05-11 Author: Nader for MAST Software



## Chapter 2

# Data Structure Index

### 2.1 Data Structures

Here are the data structures with brief descriptions:

<a href="#">points</a>	7
------------------------	---



## Chapter 3

# File Index

### 3.1 File List

Here is a list of all files with brief descriptions:

/Users/allgood38/Desktop/qmast/sailcode_alpha6/ <a href="#">DataAcquisition.pde</a> . . . . .	9
/Users/allgood38/Desktop/qmast/sailcode_alpha6/ <a href="#">LocationStruct.h</a> . . . . .	9
/Users/allgood38/Desktop/qmast/sailcode_alpha6/ <a href="#">Menu.pde</a> . . . . .	11
/Users/allgood38/Desktop/qmast/sailcode_alpha6/ <a href="#">MotorControlFunctions.pde</a> . . . . .	11
/Users/allgood38/Desktop/qmast/sailcode_alpha6/ <a href="#">NavigationCode.pde</a> . . . . .	11
/Users/allgood38/Desktop/qmast/sailcode_alpha6/ <a href="#">oldtack.pde</a> . . . . .	12
/Users/allgood38/Desktop/qmast/sailcode_alpha6/ <a href="#">ParsingFunctions.pde</a> . . . . .	12
/Users/allgood38/Desktop/qmast/sailcode_alpha6/ <a href="#">sailcode_alpha6.pde</a> . . . . .	13
/Users/allgood38/Desktop/qmast/sailcode_alpha6/ <a href="#">SailingLogic.pde</a> . . . . .	18
/Users/allgood38/Desktop/qmast/sailcode_alpha6/ <a href="#">StationKeeping.pde</a> . . . . .	18
/Users/allgood38/Desktop/qmast/sailcode_alpha6/ <a href="#">Tack.pde</a> . . . . .	19
/Users/allgood38/Desktop/qmast/sailcode_alpha6/ <a href="#">Testing_Functions.pde</a> . . . . .	19
/Users/allgood38/Desktop/qmast/sailcode_alpha6/ <a href="#">Transmit.pde</a> . . . . .	19
/Users/allgood38/Desktop/qmast/sailcode_alpha6/ <a href="#">Utilities.pde</a> . . . . .	20



## Chapter 4

# Data Structure Documentation

### 4.1 points Struct Reference

```
#include <LocationStruct.h>
```

#### Data Fields

- double [latDeg](#)
- double [latMin](#)
- double [lonDeg](#)
- double [lonMin](#)

#### 4.1.1 Field Documentation

4.1.1.1 double [latDeg](#)

4.1.1.2 double [latMin](#)

4.1.1.3 double [lonDeg](#)

4.1.1.4 double [lonMin](#)

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

- [/Users/allgood38/Desktop/qmast/sailcode\\_alpha6/LocationStruct.h](#)





## Chapter 5

# File Documentation

### 5.1 /Users/allgood38/Desktop/qmast/sailcode\_alpha6/DataAcquisition.pde File Reference

#### Functions

- void [sensorData](#) (int *bufferLength*, char *device*)
- void [setErrorBit](#) (int *aBit*)
- void [clearErrorBit](#) (int *aBit*)
- int [checkErrorBit](#) (int *aBit*)

#### 5.1.1 Function Documentation

5.1.1.1 int [checkErrorBit](#) ( int *aBit* )

5.1.1.2 void [clearErrorBit](#) ( int *aBit* )

5.1.1.3 void [sensorData](#) ( int *bufferLength*, char *device* )

5.1.1.4 void [setErrorBit](#) ( int *aBit* )

### 5.2 /Users/allgood38/Desktop/qmast/sailcode\_alpha6/Location-Struct.h File Reference

#### Data Structures

- struct [points](#)

## Variables

- [points waypoints](#) [10]
- [points stationPoints](#) [4]
- [points floatingStationPoints](#) [4]
- [points coursePoints](#) [10]
- [points clearPoints](#)
- [points boatLocation](#)
- [points stayPoint](#)

### 5.2.1 Variable Documentation

#### 5.2.1.1 [points boatLocation](#)

#### 5.2.1.2 [points clearPoints](#)

**Initial value:**

```
{
  0, 0, 0, 0 }
```

#### 5.2.1.3 [points coursePoints](#)[10]

**Initial value:**

```
{
  { 0, 0, 0, 0 },
  { 0, 0, 0, 0 },
  { 0, 0, 0, 0 },
  { 0, 0, 0, 0 },
  { 0, 0, 0, 0 },
  { 0, 0, 0, 0 },
  { 0, 0, 0, 0 },
  { 0, 0, 0, 0 },
  { 0, 0, 0, 0 },
  { 0, 0, 0, 0 } }
```

#### 5.2.1.4 [points floatingStationPoints](#)[4]

#### 5.2.1.5 [points stationPoints](#)[4]

#### 5.2.1.6 [points stayPoint](#)

#### 5.2.1.7 [points waypoints](#)[10]

**Initial value:**

```
{
  { 44.0, 13.6927, -76.0, -29.5175 },
  { 38.0, 58.9443, -76.0, -28.7383 },
  { 38.0, 58.9515, -76.0, -28.7127 } }
```

### **5.3 /Users/allgood38/Desktop/qmast/sailcode\_alpha6/Menu.pde File Reference 11**

## **5.3 /Users/allgood38/Desktop/qmast/sailcode\_alpha6/Menu.pde - File Reference**

### **Functions**

- int [displayMenu](#) ()

#### **5.3.1 Function Documentation**

5.3.1.1 int [displayMenu](#) ( )

## **5.4 /Users/allgood38/Desktop/qmast/sailcode\_alpha6/MotorControl-Functions.pde File Reference**

### **Functions**

- void [servo\\_command](#) (int *whichservo*, int *position*, byte *longRange*)
- void [setrudder](#) (float *ang*)
- void [setSails](#) (float *ang*)
- void [setJib](#) (float *ang*)
- void [setMain](#) (float *ang*)

#### **5.4.1 Function Documentation**

5.4.1.1 void [servo\\_command](#) ( int *whichservo*, int *position*, byte *longRange* )

5.4.1.2 void [setJib](#) ( float *ang* )

5.4.1.3 void [setMain](#) ( float *ang* )

5.4.1.4 void [setrudder](#) ( float *ang* )

5.4.1.5 void [setSails](#) ( float *ang* )

## **5.5 /Users/allgood38/Desktop/qmast/sailcode\_alpha6/Navigation-Code.pde File Reference**

### **Functions**

- double [GPSdistance](#) (struct [points](#) *location1*, struct [points](#) *location2*)
- int [getWaypointDirn](#) (struct [points](#) *waypoint*)
- int [getCloseHauledDirn](#) ()
- int [getOppositeCloseHauledDirn](#) ()
- int [getWindDirn](#) ()

### 5.5.1 Function Documentation

5.5.1.1 `int getCloseHauledDirn ( )`

5.5.1.2 `int getOppositeCloseHauledDirn ( )`

5.5.1.3 `int getWaypointDirn ( struct points waypoint )`

5.5.1.4 `int getWindDirn ( )`

5.5.1.5 `double GPSdistance ( struct points location1, struct points location2 )`

## 5.6 /Users/allgood38/Desktop/qmast/sailcode\_alpha6/oldtack.pde - File Reference

### Functions

- void [oldtack](#) ()
- void [oldtack2](#) ()  
*newer revision*
- void [getOutofIronsOld](#) (int tackside)

### 5.6.1 Function Documentation

5.6.1.1 `void getOutofIronsOld ( int tackside )`

5.6.1.2 `void oldtack ( )`

5.6.1.3 `void oldtack2 ( )`

*newer revision*

## 5.7 /Users/allgood38/Desktop/qmast/sailcode\_alpha6/Parsing-Functions.pde File Reference

### Functions

- int [DataValid](#) (char \*val)
- void [ParseGPGLL](#) (char \*GPGLL\_string, double \*degree, double \*minute)
- int [Parser](#) (char \*val)

### 5.7.1 Function Documentation

5.7.1.1 int DataValid ( char \* *val* )

5.7.1.2 void ParseGPGLL ( char \* *GPGLL\_string*, double \* *degree*, double \* *minute* )

5.7.1.3 int Parser ( char \* *val* )

## 5.8 /Users/allgood38/Desktop/qmast/sailcode\_alpha6/sailcode\_alpha6.pde File Reference

```
#include "LocationStruct.h"    #include <SoftwareSerial.h>
#include <String.h> #include <stdio.h> #include <avr/io.-
h>
```

### Defines

- #define TACKING\_ANGLE 40  
*for parsing - necessary?*
- #define MARK\_DISTANCE 4
- #define STATION\_KEEPING\_RADIUS 15
- #define WIND\_CHANGE\_THRESHOLD 10
- #define BUFF\_MAX 511
- #define DEGREE\_TO\_MINUTE 60
- #define LATITUDE\_TO\_METER 1855
- #define LONGITUDE\_TO\_METER 1314
- #define noDataBit 0
- #define oldDataBit 1
- #define checksumBadBit 2
- #define twoCommasBit 3
- #define rolloverDataBit 4
- #define badCompassDataBit 5
- #define tooMuchRollBit 6
- #define badWindData 7
- #define badGpsData 8
- #define ALL\_IN 0
- #define ALL\_OUT 100
- #define resetPin 8
- #define txPin 9
- #define SHORTEST\_NMEA 5
- #define LONGEST\_NMEA 120

### Functions

- void setup ()
- void loop ()

## Variables

- int `extraWindData` = 0
  - when testing by sending strings through the serial monitor, you need to select "newline" ending from the dropdown beside the baud*
- int `extraCompassData` = 0
- int `savedWindChecksum` = 0
- int `savedWindXorState` = 0
- int `savedCompassChecksum` = 0
- int `savedCompassXorState` = 0
- char `extraWindDataArray` [LONGEST\_NMEA]
- char `extraCompassDataArray` [LONGEST\_NMEA]
- float `heading`
- float `deviation`
- float `variance`
- float `bspeed`
- float `bspeedk`
- float `wind_angl`
- float `wind_velocity`
- float `headingc`
- float `pitch`
- float `roll`
- float `trueWind`
- int `rudderVal`
- int `jibVal`
- int `mainVal`
- float `headingVal`
- float `distanceVal`
- float `heading_newest`
- float `wind_angl_newest`
- SoftwareSerial `servo_ser` = SoftwareSerial(7, txPin)
- int `rudderDir` = -1
- int `points`
- int `point`
- int `currentPoint` = 0
- int `StraightSailDirection`
- int `CurrentSelection`
- long `startTime`
- int `stationCounter`
- boolean `timesUp`
- int `StationKeepingTimeInBox` = 270000
- boolean `tacking`
- int `tackingSide`
- int `ironTime`
- int `errorCode`

## 5.8.1 Define Documentation

5.8.1.1 #define ALL\_IN 0

5.8.1.2 #define ALL\_OUT 100

5.8.1.3 #define badCompassDataBit 5

5.8.1.4 #define badGpsData 8

5.8.1.5 #define badWindData 7

5.8.1.6 #define BUFF\_MAX 511

5.8.1.7 #define checksumBadBit 2

5.8.1.8 #define DEGREE\_TO\_MINUTE 60

5.8.1.9 #define LATITUDE\_TO\_METER 1855

5.8.1.10 #define LONGEST\_NMEA 120

5.8.1.11 #define LONGITUDE\_TO\_METER 1314

5.8.1.12 #define MARK\_DISTANCE 4

5.8.1.13 #define noDataBit 0

5.8.1.14 #define oldDataBit 1

5.8.1.15 #define resetPin 8

5.8.1.16 #define rolloverDataBit 4

5.8.1.17 #define SHORTEST\_NMEA 5

5.8.1.18 #define STATION\_KEEPING\_RADIUS 15

5.8.1.19 #define TACKING\_ANGLE 40

for parsing - necessary?

for pololu non-buffering serial channel for parsing - necessary?

5.8.1.20 #define tooMuchRollBit 6

5.8.1.21 #define twoCommasBit 3

5.8.1.22 `#define txPin 9`

5.8.1.23 `#define WIND_CHANGE_THRESHOLD 10`

## 5.8.2 Function Documentation

5.8.2.1 `void loop ( )`

if menu returned 0, any updating happened in the menu function itself and we want the code to just keep doing what it was doing before (e.g. setting RC mode)

Straight Sail towards N,S,E,W as 0, 180, 90, 270. No sail control.

Straightsail can no longer be called in isolation, needs sailtoWaypoint which keeps track of when tacking is necessary

stationskeeps around a single spot in the middle of the square

5.8.2.2 `void setup ( )`

Change wind to send 5 times a second default for now, need to make sure we can get everything out of the buffer

## 5.8.3 Variable Documentation

5.8.3.1 `float bspeed`

5.8.3.2 `float bspeedk`

5.8.3.3 `int currentPoint = 0`

5.8.3.4 `int CurrentSelection`

5.8.3.5 `float deviation`

5.8.3.6 `float distanceVal`

5.8.3.7 `int errorCode`

5.8.3.8 `int extraCompassData = 0`

5.8.3.9 `char extraCompassDataArray[LONGEST_NMEA]`

5.8.3.10 `int extraWindData = 0`

when testing by sending strings through the serial monitor, you need to select "newline" ending from the dropdown beside the baud



5.8.3.11 char extraWindDataArray[LONGEST\_NMEA]

5.8.3.12 float heading

5.8.3.13 float heading\_newest

5.8.3.14 float headingc

5.8.3.15 float headingVal

5.8.3.16 int ironTime

5.8.3.17 int jibVal

5.8.3.18 int mainVal

5.8.3.19 float pitch

5.8.3.20 int point

5.8.3.21 int points

5.8.3.22 float roll

5.8.3.23 int rudderDir = -1

5.8.3.24 int rudderVal

5.8.3.25 int savedCompassChecksum = 0

5.8.3.26 int savedCompassXorState = 0

5.8.3.27 int savedWindChecksum = 0

5.8.3.28 int savedWindXorState = 0

5.8.3.29 SoftwareSerial servo\_ser = SoftwareSerial(7, txPin)

5.8.3.30 long startTime

5.8.3.31 int stationCounter

5.8.3.32 int StationKeepingTimeInBox = 270000

5.8.3.33 int StraightSailDirection

5.8.3.34 boolean tacking

5.8.3.35 int tackingSide

5.8.3.36 boolean timesUp

5.8.3.37 float trueWind

5.8.3.38 float variance

5.8.3.39 float wind\_angl

5.8.3.40 float wind\_angl\_newest

5.8.3.41 float wind\_velocity

## 5.9 /Users/allgood38/Desktop/qmast/sailcode\_alpha6/Sailing-Logic.pde File Reference

### Functions

- void [sailCourse](#) ()
- void [sailToWaypoint](#) (struct [points](#) waypoint)
- void [sail](#) (int waypointDirn)
- boolean [checkTack](#) (int corridorHalfWidth, struct [points](#) waypoint)
- void [sailControl](#) ()
- int [rudderControl](#) (int directionError)

### 5.9.1 Function Documentation

5.9.1.1 boolean [checkTack](#) ( int *corridorHalfWidth*, struct [points](#) *waypoint* )

5.9.1.2 int [rudderControl](#) ( int *directionError* )

5.9.1.3 void [sail](#) ( int *waypointDirn* )

5.9.1.4 void [sailControl](#) ( )

5.9.1.5 void [sailCourse](#) ( )

5.9.1.6 void [sailToWaypoint](#) ( struct [points](#) *waypoint* )

## 5.10 /Users/allgood38/Desktop/qmast/sailcode\_alpha6/Station-Keeping.pde File Reference

## 5.11 /Users/allgood38/Desktop/qmast/sailcode\_alpha6/Tack.pde File Reference19

### Functions

- void [getStationKeepingCentre](#) (double \*centreLatMin, double \*centreLonMin)
- void [fillStationKeepingWaypoints](#) (double centreLatMin, double centreLonMin, int windBearing)
- int [stationKeep](#) ()
- void [stationKeepSinglePoint](#) ()

#### 5.10.1 Function Documentation

5.10.1.1 void [fillStationKeepingWaypoints](#) ( double *centreLatMin*, double *centreLonMin*, int *windBearing* )

5.10.1.2 void [getStationKeepingCentre](#) ( double \* *centreLatMin*, double \* *centreLonMin* )

5.10.1.3 int [stationKeep](#) ( )

5.10.1.4 void [stationKeepSinglePoint](#) ( )

## 5.11 /Users/allgood38/Desktop/qmast/sailcode\_alpha6/Tack.pde - File Reference

### Functions

- void [tack](#) ()
- void [getOutoflrons](#) (int tackside)

#### 5.11.1 Function Documentation

5.11.1.1 void [getOutoflrons](#) ( int *tackside* )

5.11.1.2 void [tack](#) ( )

## 5.12 /Users/allgood38/Desktop/qmast/sailcode\_alpha6/Testing\_- Functions.pde File Reference

## 5.13 /Users/allgood38/Desktop/qmast/sailcode\_alpha6/Transmit.pde File Reference

### Functions

- void [transmit](#) (void)
- void [relayData](#) ()

### 5.13.1 Function Documentation

#### 5.13.1.1 void relayData ( )

Similar to the transmit function except the output contains information which is not read by LabView, so codes are not needed?

Latitude and longitude of boat's location, split into more precise degrees and minutes, to fit into a float

#### 5.13.1.2 void transmit ( void )

The transmit function is used to communicate with LabView? through a series of preset strings, which are represented by the graphical gauges?

Prints directly to the serial and takes input from global values

## 5.14 /Users/allgood38/Desktop/qmast/sailcode\_alpha6/Utilities.pde File Reference

### Functions

- float [degreesToRadians](#) (int angle)
- int [radiansToDegrees](#) (float angle)
- boolean [between](#) (int angle, int a, int b)
- char [convertASCIItoHex](#) (const char ch)

### 5.14.1 Function Documentation

#### 5.14.1.1 boolean between ( int *angle*, int *a*, int *b* )

#### 5.14.1.2 char convertASCIItoHex ( const char *ch* )

#### 5.14.1.3 float degreesToRadians ( int *angle* )

#### 5.14.1.4 int radiansToDegrees ( float *angle* )