# LVLHFrameModel

5.3

Generated by Doxygen 1.8.14

# Contents

# Chapter 1

# Module Index

## 1.1 Modules

Here is a list of all modules:

# Chapter 2

# Namespace Index

## 2.1 Namespace List

Here is a list of all namespaces with brief descriptions:

# Chapter 3

# Data Structure Index

## 3.1 Data Structures

Here are the data structures with brief descriptions:

# Chapter 4

# File Index

## 4.1 File List

Here is a list of all files with brief descriptions:

# Chapter 5

# Module Documentation

## 5.1 Models

### Modules

- Utils

#### 5.1.1 Detailed Description

## 5.2 Utils

### Modules

- LvlhFrame

#### 5.2.1 Detailed Description

## 5.3 LvlhFrame

### Files

- file lvlh_frame.hh

  *Define the class LvlhFrame, the class used to represent a local-vertical, local-horizontal reference frame associated with a subject DynBody.*
- file lvlh_frame_messages.hh

  *Define the class LvlhFrameMessages, the class that specifies the message IDs used in the LvlhFrame model.*
- file lvlh_type.hh

  *Define the class LvlhType, which identifies the type of LVLH desired to be calculated.*
- file lvlh_frame.cc

  *Define methods for the LVLH reference frame class.*
- file lvlh_frame_messages.cc

  *Implement the class LvlhFrameMessages.*

### Namespaces

- jeod

  *Namespace jeod.*

#### 5.3.1 Detailed Description

# Chapter 6

# Namespace Documentation

## 6.1 jeod Namespace Reference

Namespace jeod.

### Data Structures

- class LvlhFrame

  *The class used to represent an LVLH reference frame associated with a subject DynBody.*
- class LvlhFrameMessages

  *The class that specifies the message IDs used in the LvlhFrame model.*
- class LvlhType

  *The class used to identify the type of LVLH desired.*

### 6.1.1 Detailed Description

Namespace jeod.

# Chapter 7

# Data Structure Documentation

## 7.1 jeod::LvlhFrame Class Reference

The class used to represent an LVLH reference frame associated with a subject DynBody.

`#include <lvlh_frame.hh>`

### Public Member Functions

- LvlhFrame ()=default
- ~LvlhFrame ()

  *Destruct an LvlhFrame object.*
- LvlhFrame (const LvlhFrame &)=delete
- LvlhFrame & operator= (const LvlhFrame &)=delete
- void initialize (DynManager &dyn_manager)

  *Begin initialization of an LvlhFrame.*
- void update ()

  *Update the state.*
- void set_subject_name (const std::string &new_name)

  *Set the subject_name to the supplied value.*
- void set_planet_name (const std::string &new_name)

  *Set the planet_name to the supplied value.*
- void set_subject_frame (RefFrame &new_frame)

  *Set the subject_frame to the supplied value.*
- void set_planet (BasePlanet &new_planet)

  *Set the planet whose PCI frame will be the reference for LVLH.*

### Data Fields

- RefFrame frame

  *The LVLH frame defined by the subject frame's motion with respect to the reference planet.*
- std::string subject_name {""}

  *The frame whose motion defines LVLH.*
- std::string planet_name {""}

  *The planet used as reference for the LVLH frame.*

### Protected Member Functions

- void compute_lvlh_frame (const RefFrameTrans &rel_trans)

  *Update the state of the LVLH frame wrt its parent.*

### Protected Attributes

- RefFrame * subject_frame {}

  *The (moving) frame specified with subject_name.*
- RefFrame * planet_centered_inertial {}

  *The inertial frame with origin at the center of the specified planet.*

### Private Attributes

- DynManager * local_dm {}

  *A local pointer to the dynamics manager needed for clean-up.*
- bool initialized {}

  *trick_units(–)*

### Friends

- class InputProcessor
- void init_attrjeod__LvlhFrame ()

## 7.1.1 Detailed Description

The class used to represent an LVLH reference frame associated with a subject DynBody.

Definition at line 82 of file lvlh_frame.hh.

## 7.1.2 Constructor & Destructor Documentation

#### 7.1.2.1 LvlhFrame() [1/2]

```
jeod::LvlhFrame::LvlhFrame ( )  [default]
```

#### 7.1.2.2 ~LvlhFrame()

```
jeod::LvlhFrame::~LvlhFrame ( )
```

Destruct an LvlhFrame object.

Definition at line 49 of file lvlh_frame.cc.

References frame, local_dm, planet_centered_inertial, and subject_frame.

#### 7.1.2.3 LvlhFrame() [2/2]

```
jeod::LvlhFrame::LvlhFrame (
            const LvlhFrame &  )  [delete]
```

### 7.1.3 Member Function Documentation

#### 7.1.3.1 compute_lvlh_frame()

```
void jeod::LvlhFrame::compute_lvlh_frame (
            const RefFrameTrans & rel_trans )  [protected]
```

Update the state of the LVLH frame wrt its parent.

**Parameters**

| | | |
|---|---|---|
| in | *rel_trans* | Planet relative state |

Definition at line 247 of file lvlh_frame.cc.

References frame.

Referenced by update().

#### 7.1.3.2 initialize()

```
void jeod::LvlhFrame::initialize (
            DynManager & dyn_manager )
```

Begin initialization of an LvlhFrame.

**Parameters**

| | | |
|---|---|---|
| in,out | *dyn_manager* | Dynamics manager |

Definition at line 75 of file lvlh_frame.cc.

References frame, initialized, jeod::LvlhFrameMessages::invalid_configuration, jeod::LvlhFrameMessages↩ ::invalid_name, local_dm, planet_centered_inertial, planet_name, subject_frame, and subject_name.

#### 7.1.3.3 operator=()

```
LvlhFrame& jeod::LvlhFrame::operator= (
             const LvlhFrame &  )  [delete]
```

#### 7.1.3.4 set_planet()

```
void jeod::LvlhFrame::set_planet (
             BasePlanet & new_planet )
```

Set the planet whose PCI frame will be the reference for LVLH.

**Parameters**

| | | |
|---|---|---|
| in | *new_planet* | new planet. |

Definition at line 238 of file lvlh_frame.cc.

References planet_centered_inertial.

#### 7.1.3.5 set_planet_name()

```
void jeod::LvlhFrame::set_planet_name (
             const std::string & new_name )
```

Set the planet_name to the supplied value.

**Parameters**

| | | |
|---|---|---|
| in | *new_name* | new name. |

Definition at line 229 of file lvlh_frame.cc.

References planet_name.

#### 7.1.3.6 set_subject_frame()

```
void jeod::LvlhFrame::set_subject_frame (
             RefFrame & new_frame )
```

Set the subject_frame to the supplied value.

**Parameters**

| | | |
|---|---|---|
| in | *new_frame* | new frame. |

Definition at line 220 of file lvlh_frame.cc.

References subject_frame.

#### 7.1.3.7 set_subject_name()

```
void jeod::LvlhFrame::set_subject_name (
             const std::string & new_name )
```

Set the subject_name to the supplied value.

**Parameters**

| | | |
|---|---|---|
| in | *new_name* | new name. |

Definition at line 211 of file lvlh_frame.cc.

References subject_name.

#### 7.1.3.8 update()

```
void jeod::LvlhFrame::update ( )
```

Update the state.

Definition at line 184 of file lvlh_frame.cc.

References compute_lvlh_frame(), frame, planet_centered_inertial, and subject_frame.

### 7.1.4 Friends And Related Function Documentation

#### 7.1.4.1 init_attrjeod__LvlhFrame

```
void init_attrjeod__LvlhFrame ( )  [friend]
```

#### 7.1.4.2 InputProcessor

```
friend class InputProcessor  [friend]
```

Definition at line 84 of file lvlh_frame.hh.

### 7.1.5 Field Documentation

#### 7.1.5.1 frame

```
RefFrame jeod::LvlhFrame::frame
```

The LVLH frame defined by the subject frame's motion with respect to the reference planet.

trick_units(–)

Definition at line 90 of file lvlh_frame.hh.

Referenced by compute_lvlh_frame(), initialize(), update(), and ∼LvlhFrame().

#### 7.1.5.2 initialized

```
bool jeod::LvlhFrame::initialized {}  [private]
```

trick_units(–)

Definition at line 122 of file lvlh_frame.hh.

Referenced by initialize().

#### 7.1.5.3 local_dm

```
DynManager* jeod::LvlhFrame::local_dm {}  [private]
```

A local pointer to the dynamics manager needed for clean-up.

trick_units(–)

Definition at line 117 of file lvlh_frame.hh.

Referenced by initialize(), and ∼LvlhFrame().

#### 7.1.5.4 planet_centered_inertial

```
RefFrame* jeod::LvlhFrame::planet_centered_inertial {}  [protected]
```

The inertial frame with origin at the center of the specified planet.

trick_units(–)

Definition at line 111 of file lvlh_frame.hh.

Referenced by initialize(), set_planet(), update(), and ~LvlhFrame().

#### 7.1.5.5 planet_name

```
std::string jeod::LvlhFrame::planet_name {""}
```

The planet used as reference for the LVLH frame.

trick_units(–)

Definition at line 100 of file lvlh_frame.hh.

Referenced by initialize(), and set_planet_name().

#### 7.1.5.6 subject_frame

```
RefFrame* jeod::LvlhFrame::subject_frame {}  [protected]
```

The (moving) frame specified with subject_name.

trick_units(–)

Definition at line 106 of file lvlh_frame.hh.

Referenced by initialize(), set_subject_frame(), update(), and ~LvlhFrame().

#### 7.1.5.7 subject_name

```
std::string jeod::LvlhFrame::subject_name {""}
```

The frame whose motion defines LVLH.

Can be on a vehicle or not.trick_units(–)

Definition at line 95 of file lvlh_frame.hh.

Referenced by initialize(), and set_subject_name().

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

- lvlh_frame.hh
- lvlh_frame.cc

## 7.2 jeod::LvlhFrameMessages Class Reference

The class that specifies the message IDs used in the LvlhFrame model.

`#include <lvlh_frame_messages.hh>`

### Public Member Functions

- LvlhFrameMessages ()=delete
- LvlhFrameMessages (const LvlhFrameMessages &)=delete
- LvlhFrameMessages & operator= (const LvlhFrameMessages &)=delete

### Static Public Attributes

- static const char * fatal_error = "utils/lvlh_frame/" "fatal_error"

  *Issued when performing an action results in an error return from the method performing the action.*
- static const char * illegal_value = "utils/lvlh_frame/" "illegal_value"

  *Issued when a simple type (e.g.*
- static const char * invalid_name = "utils/lvlh_frame/" "invalid_name"

  *Issued when a name is invalid (NULL, empty, or does not name an object of the specified type).*
- static const char * invalid_configuration = "utils/lvlh_frame/" "invalid_configuration"

  *Issued when insufficient information has been specified prior to initialization.*
- static const char * invalid_object = "utils/lvlh_frame/" "invalid_object"

  *Issued when a pointer points to an object of the wrong type.*
- static const char * null_pointer = "utils/lvlh_frame/" "null_pointer"

  *Error issued when a pointer is required but was not provided.*
- static const char * trace = "utils/lvlh_frame/" "trace"

  *Debug message issued to trace LvlhFrame actions.*
- static const char * divide_by_zero = "utils/lvlh_frame/" "divide_by_zero"

  *Fatal message when a divide by zero is encountered.*

### Friends

- class InputProcessor
- void init_attrjeod__LvlhFrameMessages ()

### 7.2.1 Detailed Description

The class that specifies the message IDs used in the LvlhFrame model.

Definition at line 81 of file lvlh_frame_messages.hh.

### 7.2.2 Constructor & Destructor Documentation

#### 7.2.2.1 LvlhFrameMessages() [1/2]

```
jeod::LvlhFrameMessages::LvlhFrameMessages ( )  [delete]
```

#### 7.2.2.2 LvlhFrameMessages() [2/2]

```
jeod::LvlhFrameMessages::LvlhFrameMessages (
            const LvlhFrameMessages &  )  [delete]
```

### 7.2.3 Member Function Documentation

#### 7.2.3.1 operator=()

```
LvlhFrameMessages& jeod::LvlhFrameMessages::operator= (
            const LvlhFrameMessages &  )  [delete]
```

### 7.2.4 Friends And Related Function Documentation

#### 7.2.4.1 init_attrjeod__LvlhFrameMessages

```
void init_attrjeod__LvlhFrameMessages ( )  [friend]
```

#### 7.2.4.2 InputProcessor

```
friend class InputProcessor  [friend]
```

Definition at line 83 of file lvlh_frame_messages.hh.

### 7.2.5 Field Documentation

#### 7.2.5.1 divide_by_zero

```
char const * jeod::LvlhFrameMessages::divide_by_zero = "utils/lvlh_frame/" "divide_by_zero"
[static]
```

Fatal message when a divide by zero is encountered.

trick_units(–)

Definition at line 126 of file lvlh_frame_messages.hh.

#### 7.2.5.2 fatal_error

```
char const * jeod::LvlhFrameMessages::fatal_error = "utils/lvlh_frame/" "fatal_error"  [static]
```

Issued when performing an action results in an error return from the method performing the action.

trick_units(–)

Definition at line 89 of file lvlh_frame_messages.hh.

#### 7.2.5.3 illegal_value

```
char const * jeod::LvlhFrameMessages::illegal_value = "utils/lvlh_frame/" "illegal_value"
[static]
```

Issued when a simple type (e.g.

an enum) has an illegal value.trick_units(–)

Definition at line 94 of file lvlh_frame_messages.hh.

#### 7.2.5.4 invalid_configuration

```
char const * jeod::LvlhFrameMessages::invalid_configuration = "utils/lvlh_frame/" "invalid_↩
configuration"  [static]
```

Issued when insufficient information has been specified prior to initialization.

trick_units(–)

Definition at line 106 of file lvlh_frame_messages.hh.

Referenced by jeod::LvlhFrame::initialize().

#### 7.2.5.5 invalid_name

```
char const * jeod::LvlhFrameMessages::invalid_name = "utils/lvlh_frame/" "invalid_name"  [static]
```

Issued when a name is invalid (NULL, empty, or does not name an object of the specified type).

trick_units(–)

Definition at line 100 of file lvlh_frame_messages.hh.

Referenced by jeod::LvlhFrame::initialize().

#### 7.2.5.6 invalid_object

```
char const * jeod::LvlhFrameMessages::invalid_object = "utils/lvlh_frame/" "invalid_object"
[static]
```

Issued when a pointer points to an object of the wrong type.

trick_units(–)

Definition at line 111 of file lvlh_frame_messages.hh.

#### 7.2.5.7 null_pointer

```
char const * jeod::LvlhFrameMessages::null_pointer = "utils/lvlh_frame/" "null_pointer"  [static]
```

Error issued when a pointer is required but was not provided.

trick_units(–)

Definition at line 116 of file lvlh_frame_messages.hh.

#### 7.2.5.8 trace

```
char const * jeod::LvlhFrameMessages::trace = "utils/lvlh_frame/" "trace"  [static]
```

Debug message issued to trace LvlhFrame actions.

trick_units(–)

Definition at line 121 of file lvlh_frame_messages.hh.

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

- lvlh_frame_messages.hh
- lvlh_frame_messages.cc

## 7.3 jeod::LvlhType Class Reference

The class used to identify the type of LVLH desired.

`#include <lvlh_type.hh>`

### Public Types

- enum Type { Rectilinear = 0, CircularCurvilinear = 1, EllipticalCurvilinear = 2 }

  *An enumeration to specify the type of LVLH coordinates to use, whether rectilinear, circular curvilinear, or elliptical curvilinear.*

### Public Member Functions

- LvlhType ()

  *Default constructor.*

### Data Fields

- Type value

  *Indicates type of LVLH coordinates desired.*

### Friends

- class InputProcessor
- void init_attrjeod__LvlhType ()

### 7.3.1 Detailed Description

The class used to identify the type of LVLH desired.

Definition at line 76 of file lvlh_type.hh.

### 7.3.2 Member Enumeration Documentation

#### 7.3.2.1 Type

`enum jeod::LvlhType::Type`

An enumeration to specify the type of LVLH coordinates to use, whether rectilinear, circular curvilinear, or elliptical curvilinear.

As of March 2015, elliptical is not implemented.

**Enumerator**

| | |
|---|---|
| Rectilinear | |
| CircularCurvilinear | |
| EllipticalCurvilinear | |

Definition at line 85 of file lvlh_type.hh.

### 7.3.3 Constructor & Destructor Documentation

#### 7.3.3.1 LvlhType()

```
jeod::LvlhType::LvlhType ( )  [inline]
```

Default constructor.

Definition at line 110 of file lvlh_type.hh.

### 7.3.4 Friends And Related Function Documentation

#### 7.3.4.1 init_attrjeod__LvlhType

```
void init_attrjeod__LvlhType ( )  [friend]
```

#### 7.3.4.2 InputProcessor

```
friend class InputProcessor  [friend]
```

Definition at line 78 of file lvlh_type.hh.

### 7.3.5 Field Documentation

#### 7.3.5.1 value

```
Type jeod::LvlhType::value
```

Indicates type of LVLH coordinates desired.

Default is rectilinear.trick_units(–)

Definition at line 103 of file lvlh_type.hh.

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

- lvlh_type.hh

# Chapter 8

# File Documentation

## 8.1 lvlh_frame.cc File Reference

Define methods for the LVLH reference frame class.

```
#include <cstddef>
#include "dynamics/dyn_manager/include/dyn_manager.hh"
#include "environment/planet/include/base_planet.hh"
#include "utils/math/include/vector3.hh"
#include "utils/message/include/message_handler.hh"
#include "utils/named_item/include/named_item.hh"
#include "../include/lvlh_frame.hh"
#include "../include/lvlh_frame_messages.hh"
```

### Namespaces

- jeod

  *Namespace jeod.*

### 8.1.1 Detailed Description

Define methods for the LVLH reference frame class.

## 8.2 lvlh_frame.hh File Reference

Define the class LvlhFrame, the class used to represent a local-vertical, local-horizontal reference frame associated with a subject DynBody.

```
#include <string>
#include "dynamics/dyn_manager/include/class_declarations.hh"
#include "environment/planet/include/class_declarations.hh"
#include "utils/ref_frames/include/ref_frame.hh"
#include "utils/sim_interface/include/jeod_class.hh"
```

### Data Structures

- class jeod::LvlhFrame

  *The class used to represent an LVLH reference frame associated with a subject DynBody.*

### Namespaces

- jeod

  *Namespace jeod.*

#### 8.2.1 Detailed Description

Define the class LvlhFrame, the class used to represent a local-vertical, local-horizontal reference frame associated with a subject DynBody.

## 8.3 lvlh_frame_messages.cc File Reference

Implement the class LvlhFrameMessages.

```
#include "utils/message/include/make_message_code.hh"
#include "../include/lvlh_frame_messages.hh"
```

### Namespaces

- jeod

  *Namespace jeod.*

### Macros

- #define MAKE_LVLHFRAME_MESSAGE_CODE(id) JEOD_MAKE_MESSAGE_CODE(LvlhFrame↩ Messages, "utils/lvlh_frame/", id)

#### 8.3.1 Detailed Description

Implement the class LvlhFrameMessages.

#### 8.3.2 Macro Definition Documentation

#### 8.3.2.1 MAKE_LVLHFRAME_MESSAGE_CODE

```
#define MAKE_LVLHFRAME_MESSAGE_CODE(
              id ) JEOD_MAKE_MESSAGE_CODE(LvlhFrameMessages, "utils/lvlh_frame/", id)
```

Definition at line 37 of file lvlh_frame_messages.cc.

## 8.4 lvlh_frame_messages.hh File Reference

Define the class LvlhFrameMessages, the class that specifies the message IDs used in the LvlhFrame model.

```
#include "utils/sim_interface/include/jeod_class.hh"
```

### Data Structures

- class jeod::LvlhFrameMessages
  *The class that specifies the message IDs used in the LvlhFrame model.*

### Namespaces

- jeod
  *Namespace jeod.*

### 8.4.1 Detailed Description

Define the class LvlhFrameMessages, the class that specifies the message IDs used in the LvlhFrame model.

## 8.5 lvlh_type.hh File Reference

Define the class LvlhType, which identifies the type of LVLH desired to be calculated.

```
#include "utils/sim_interface/include/jeod_class.hh"
```

### Data Structures

- class jeod::LvlhType
  *The class used to identify the type of LVLH desired.*

### Namespaces

- jeod
  *Namespace jeod.*

### 8.5.1 Detailed Description

Define the class LvlhType, which identifies the type of LVLH desired to be calculated.

# Index