autd3

3.1.0

Generated by Doxygen 1.8.11

Contents

Chapter 1

Introduction

Chapter 2

Namespace Index

2.1 Namespace List

Here is a list of all documented namespaces with brief descriptions:

Chapter 3

Hierarchical Index

3.1 Class Hierarchy

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

Chapter 4

Class Index

4.1 Class List

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

Chapter 5

Namespace Documentation

5.1 autd Namespace Reference

A base namespace of this autd library.

Classes

- class Controller

  A controller class.
- class FocalPointGain

  A gain with single focal point.
- class Gain

  A gain of phased array which describes amps and phases of each transducers.
- class Geometry
- class HoloGainSdp

  A gain with multiple focal points.
- class MatlabGain

  Import gain from matlab .mat file.
- class Modulation

  A sequence of amplitude gains which describes temporal modulation.
- class PlaneWaveGain
- class RawPCMModulation
- class SawModulation
- class SineModulation
- class TransducerTestGain

Typedefs

- typedef std::shared_ptr< Gain > **GainPtr**
- typedef std::shared_ptr< Geometry > **GeometryPtr**
- typedef std::shared_ptr< Modulation > **ModulationPtr**
- typedef Gain NullGain

  A gain which represents no ultrasound.
- typedef HoloGainSdp **HoloGain**

Enumerations

- enum LinkType { **ETHERCAT**, **ETHERNET**, **USB**, **SERIAL** }

  A connection type between autd board and PC.

5.1.1 Detailed Description

A base namespace of this autd library.

Chapter 6

Class Documentation

6.1 autd::Controller Class Reference

A controller class.

`#include <autd3.hpp>`

Public Member Functions

- void Open (LinkType type, std::string location="")

  Open device by link type and location. The scheme of location is as follows: ETHERCAT - <ams net="" id>=""> *or <ipv4 addr>="">:<ams net="" id>="">* *(ex. 192.168.1.2:192.168.1.3.1.1). The ipv4 addr will be extracted from leading 4 octets of ams net id if not specified. ETHERNET - ipv4 addr USB - ignored SERIAL - file discriptor.*
- bool **isOpen** ()
- void **Close** ()
- size_t remainingInBuffer ()

  Return the number of gains and modulations remaining in the buffer.
- GeometryPtr **geometry** ()
- void **SetGeometry** (const GeometryPtr &geometry)
- void **SetSilentMode** (bool silent)
- bool **silentMode** ()
- void AppendGain (const GainPtr &gain)

  [procedure style] append base gain
- void **AppendGainSync** (const GainPtr &gain)
- void **AppendModulation** (const ModulationPtr &modulation)
- void **AppendModulationSync** (const ModulationPtr &modulation)
- void **Flush** ()
- Controller & operator<< (const float coef)

  [stream style] append temporal coef
- Controller & operator<< (const GainPtr &gain)

  [stream style] append single gain
- Controller & operator<< (const ModulationPtr &mod)

  [stream style] append modulation

6.1.1 Detailed Description

A controller class.

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

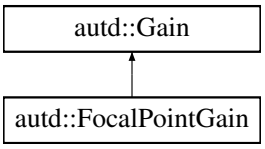
- include/autd3.hpp

6.2 autd::FocalPointGain Class Reference

A gain with single focal point.

`#include <autd3.hpp>`

Inheritance diagram for autd::FocalPointGain:



Public Member Functions

- void build ()

  Calculate a gain and put it into `_data`. Unless called explicitly by user, this method will be called internally on not-main thread. Be careful to manage critical sections if you extend this class.

Static Public Member Functions

- static GainPtr **Create** (Eigen::Vector3f point)

Additional Inherited Members

6.2.1 Detailed Description

A gain with single focal point.

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

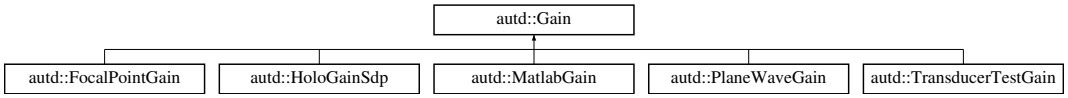
- include/autd3.hpp

6.3 autd::Gain Class Reference

A gain of phased array which describes amps and phases of each transducers.

`#include <autd3.hpp>`

Inheritance diagram for autd::Gain:



Public Member Functions

- virtual void build ()

  Calculate a gain and put it into `_data`. Unless called explicitly by user, this method will be called internally on not-main thread. Be careful to manage critical sections if you extend this class.
- void **SetGeometry** (const GeometryPtr &geometry)
- GeometryPtr **geometry** ()
- bool **built** ()

Static Public Member Functions

- static GainPtr **Create** ()

Protected Attributes

- std::mutex **_mtx**
- bool **_built**
- GeometryPtr **_geometry**
- std::map< int, std::vector< uint16_t > > **_data**

Friends

- class **Controller**
- class **Geometry**
- class **internal::Link**

6.3.1 Detailed Description

A gain of phased array which describes amps and phases of each transducers.

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

- include/autd3.hpp

6.4 autd::Geometry Class Reference

Public Member Functions

- int AddDevice (Eigen::Vector3f position, Eigen::Vector3f euler_angles)

  Add new device with position and rotation. Note that the transform is done with order: Translate -> Rotate.
- void DelDevice (int device_id)

  Remove device from the geometry.
- const int **numDevices** ()
- const int **numTransducers** ()
- const Eigen::Vector3f **position** (int transducer_idx)
- const Eigen::Vector3f & direction (int transducer_id)

  Normalized direction of a transducer specified by id.
- const int **deviceIdForTransIdx** (int transducer_idx)
- const int **deviceIdForDeviceIdx** (int device_index)
- float frequency ()

  Return a frquency of ultrasound.
- void SetFrequency (float freq)

  Set a frquency of ultrasound, which should be [33.4kHz $<$ freq $<$ 50.0kHz].

Static Public Member Functions

- static GeometryPtr **Create** ()

Friends

- class **Controller**

6.4.1 Member Function Documentation

6.4.1.1 int autd::Geometry::AddDevice (Eigen::Vector3f *position,* Eigen::Vector3f *euler_angles*)

Add new device with position and rotation. Note that the transform is done with order: Translate -> Rotate.

Parameters

position	Position of transducer #0, which is the one at the lower right corner.
euler_angles	ZYZ convention Euler angle of the device.

Returns

an id of added device, which is used to delete or do other device specific controls.

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

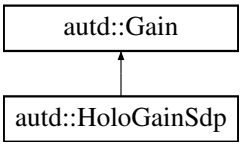
- include/autd3.hpp

6.5 autd::HoloGainSdp Class Reference

A gain with multiple focal points.

`#include <autd3.hpp>`

Inheritance diagram for autd::HoloGainSdp:



Public Member Functions

- virtual void build ()

  Calculate a gain and put it into `_data`. Unless called explicitly by user, this method will be called internally on not-main thread. Be careful to manage critical sections if you extend this class.

Static Public Member Functions

- static GainPtr **Create** (Eigen::MatrixX3f foci, Eigen::VectorXf amp)

Protected Attributes

- Eigen::MatrixX3f **_foci**
- Eigen::VectorXf **_amp**

6.5.1 Detailed Description

A gain with multiple focal points.

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

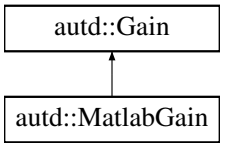
- include/autd3.hpp

6.6 autd::MatlabGain Class Reference

Import gain from matlab .mat file.

`#include <autd3.hpp>`

Inheritance diagram for autd::MatlabGain:



Public Member Functions

- void build ()

  Calculate a gain and put it into `_data`. Unless called explicitly by user, this method will be called internally on not-main thread. Be careful to manage critical sections if you extend this class.

Static Public Member Functions

- static GainPtr **Create** (std::string filename, std::string varname)

Protected Attributes

- std::string **_filename**
- std::string **_varname**

6.6.1 Detailed Description

Import gain from matlab .mat file.

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

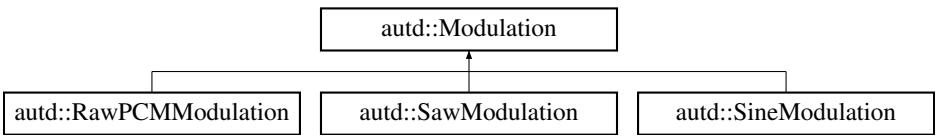
- include/autd3.hpp

6.7 autd::Modulation Class Reference

A sequence of amplitude gains which describes temporal modulation.

`#include <autd3.hpp>`

Inheritance diagram for autd::Modulation:



Public Member Functions

- const float **samplingFrequency** ()

Static Public Member Functions

- static ModulationPtr **Create** ()
- static ModulationPtr **Create** (uint8_t amp)

Public Attributes

- bool **loop**
- std::vector< uint8_t > **buffer**

Friends

- class **Controller**
- class **internal::Link**

6.7.1 Detailed Description

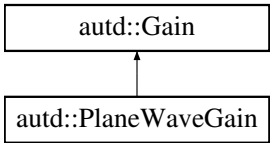
A sequence of amplitude gains which describes temporal modulation.

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

- include/autd3.hpp

6.8 autd::PlaneWaveGain Class Reference

Inheritance diagram for autd::PlaneWaveGain:



Public Member Functions

- void build ()

  Calculate a gain and put it into `_data`. Unless called explicitly by user, this method will be called internally on not-main thread. Be careful to manage critical sections if you extend this class.

Static Public Member Functions

- static GainPtr **Create** (Eigen::Vector3f direction)

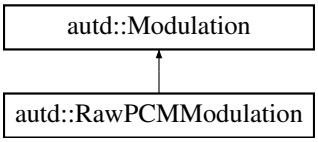
Additional Inherited Members

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

- include/autd3.hpp

6.9 autd::RawPCMModulation Class Reference

Inheritance diagram for autd::RawPCMModulation:



Static Public Member Functions

- static ModulationPtr **Create** (std::string filename, float samplingFreq=0.0f)

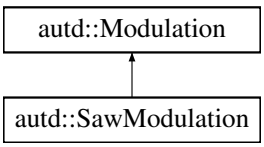
Additional Inherited Members

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

- include/autd3.hpp

6.10 autd::SawModulation Class Reference

Inheritance diagram for autd::SawModulation:



Static Public Member Functions

- static ModulationPtr **Create** (float freq)

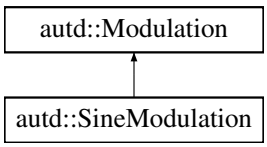
Additional Inherited Members

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

- include/autd3.hpp

6.11 autd::SineModulation Class Reference

Inheritance diagram for autd::SineModulation:



Static Public Member Functions

- static ModulationPtr **Create** (float freq, float amp=1.0f, float offset=0.5f)

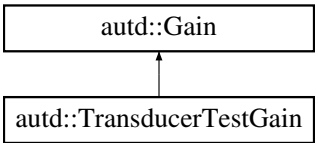
Additional Inherited Members

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

- include/autd3.hpp

6.12 autd::TransducerTestGain Class Reference

Inheritance diagram for autd::TransducerTestGain:



Public Member Functions

- void build ()

  Calculate a gain and put it into `_data`. Unless called explicitly by user, this method will be called internally on not-main thread. Be careful to manage critical sections if you extend this class.

Static Public Member Functions

- static GainPtr **Create** (int transducer_index, int amp, int phase)

Protected Attributes

- int **_xdcr_idx**
- int **_amp**
- int **_phase**

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

- include/autd3.hpp

Index