DeltaRobot

v0.1

Generated by Doxygen 1.8.9.1

Mon Mar 9 2015 21:04:06

Contents

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

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

Chapter 2

Class Index

2.1 Class List

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

Chapter 3

Class Documentation

3.1 AX12 Class Reference

The AX12 class is used to control AX-12 motors from Dynamixel.

`#include <ax12.h>`

Inheritance diagram for AX12:

3.2 data Struct Reference

Public Attributes

- unsigned char **iID**
- unsigned int **iStartAddr**
- unsigned short **iLength**
- unsigned char **iError**
- unsigned char * **pucTable**

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

- dynamixel.h

3.3 dxl_hal Class Reference

Public Member Functions

- int **open** (QString &devName, int baudrate)
- void **close** (void)
- void **clear** (void)
- int **change_baudrate** (float baudrate)
- int **write** (unsigned char *pPacket, int numPacket)
- int **read** (unsigned char *pPacket, int numPacket)
- double **get_curr_time** ()
- bool **isOpen** ()

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

- dxl_hal.h
- dxl_hal.cpp

3.4 dynamixel Class Reference

Public Member Functions

- bool **isOpen** ()
- int **initialize** (QString port_num, int baud_rate)
- int **change_baudrate** (int baud_rate)
- int **terminate** (void)
- int **get_comm_result** (void)
- void **tx_packet** (void)
- void **rx_packet** (void)
- void **txrx_packet** (void)
- void **set_txpacket_id** (int id)
- void **set_txpacket_instruction** (int instruction)
- void **set_txpacket_parameter** (int index, int value)
- void **set_txpacket_length** (int length)
- int **get_rxpacket_error** (int error)
- int **get_rxpacket_error_byte** (void)
- int **get_rxpacket_parameter** (int index)
- int **get_rxpacket_length** ()
- void **ping** (int id)
- int **read_byte** (int id, int address)
- void **write_byte** (int id, int address, int value)
- int **read_word** (int id, int address)
- void **write_word** (int id, int address, int value)
- double **get_packet_time** ()
- void **set_packet_timeout** (int NumRcvByte)
- void **set_packet_timeout_ms** (int msec)
- int **is_packet_timeout** ()

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

- dynamixel.h
- dynamixel.cpp

3.5 dynamixel2 Class Reference

Public Member Functions

- bool **isOpen** ()
- int **initialize** (QString port_num, int baud_rate)
- int **change_baudrate** (int baud_rate)
- int **terminate** (void)
- int **get_comm_result** (void)
- void **tx_packet** (void)
- void **rx_packet** (void)
- void **txrx_packet** (void)
- void **set_txpacket_id** (unsigned char id)
- void **set_txpacket_instruction** (unsigned char instruction)
- void **set_txpacket_parameter** (unsigned short index, unsigned char value)
- void **set_txpacket_length** (unsigned short length)
- int **get_rxpacket_error_byte** (void)
- int **get_rxpacket_parameter** (int index)

- int **get_rxpacket_length** ()
- void **ping** (unsigned char id)
- int **get_ping_result** (unsigned char id, int info_num)
- void **broadcast_ping** ()
- void **reboot** (unsigned char id)
- void **factory_reset** (unsigned char id, int option)
- unsigned char **read_byte** (unsigned char id, int address)
- void **write_byte** (unsigned char id, int address, unsigned char value)
- unsigned short **read_word** (unsigned char id, int address)
- void **write_word** (unsigned char id, int address, unsigned short value)
- unsigned long **read_dword** (unsigned char id, int address)
- void **write_dword** (unsigned char id, int address, unsigned long value)
- unsigned char **get_bulk_read_data_byte** (unsigned char id, unsigned int start_address)
- unsigned short **get_bulk_read_data_word** (unsigned char id, unsigned int start_address)
- unsigned long **get_bulk_read_data_dword** (unsigned char id, unsigned int start_address)
- unsigned char **get_sync_read_data_byte** (unsigned char id, unsigned int start_address)
- unsigned short **get_sync_read_data_word** (unsigned char id, unsigned int start_address)
- unsigned long **get_sync_read_data_dword** (unsigned char id, unsigned int start_address)
- void **add_stuffing** ()
- void **remove_stuffing** ()
- double **get_packet_time** ()
- int **is_packet_timeout** ()
- void **set_packet_timeout** (int NumRcvByte)
- void **set_packet_timeout_ms** (int msec)

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

- dynamixel.h
- dynamixel.cpp

3.6 XJoystick::Info Struct Reference

Struct to handle the info.

`#include <xjoystick.h>`

Public Member Functions

- Info ()
  Default Constructor.

Public Attributes

- int ID
  Contains the Joystick's ID.
- QString name
  Contains all the data.

3.6.1 Detailed Description

Struct to handle the info.

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

- xjoystick.h

3.7 MainWindow Class Reference

Inheritance diagram for MainWindow:

Collaboration diagram for MainWindow:

Public Member Functions

- MainWindow (QWidget ∗parent=0)

  Default constructor.
- ∼MainWindow ()

  Default destructor.

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

- mainwindow.h
- mainwindow.cpp

3.8 OptionsServos Class Reference

Inheritance diagram for OptionsServos:

Collaboration diagram for OptionsServos:

Public Member Functions

- **OptionsServos** (QWidget ∗parent=0)

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

- optionsservos.h
- optionsservos.cpp

3.9 OptionsWindow Class Reference

Inheritance diagram for OptionsWindow:

Collaboration diagram for OptionsWindow:

Public Member Functions

- **OptionsWindow** (QWidget ∗parent=0)

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

- optionswindow.h
- optionswindow.cpp

3.10 ping_data Struct Reference

Public Attributes

- int **iID**
- int **iModelNo**
- int **iFirmVer**

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

- dynamixel.h

3.11 XJoystick Class Reference

The XJoystick's class is used to control the SFML Joystick's class with *signals and slots*

`#include <xjoystick.h>`

Inheritance diagram for XJoystick:

Collaboration diagram for XJoystick:

Classes

- struct Info

  Struct to handle the info.

Public Slots

- void update ()

  Updates all data.

Signals

- void changed ()

  Emmitted when a joystick is connected or disconnected.

Public Member Functions

- XJoystick (int ID=-1, float filter=0.001)

  Default constructor.
- ~XJoystick ()

  Default destructor.
- int amount ()

  Amount of joysticks connected.
- bool anyConnected ()

  True if there's any joystick connected.
- QVector< Info > available ()

  Returns the system available joysticks.
- bool button (int n)

  Returns the button state.

- int buttonCount ()

  Returns the number of buttons.
- int current ()
- QVector< P > getAxis ()

  Returns all the Joystick's axis and it's names.
- void axisPress (unsigned char a, float value=100)

  To mantain an axis in a certain value.
- void axisRelease (unsigned char a)

  To release an axis and restore it's value with the joystick.
- const float & operator[ ] (int n) const

  Returns the 'n' axis value.
- bool select (int s)

  Selects the especified joystick.

3.11.1 Detailed Description

The XJoystick's class is used to control the SFML Joystick's class with *signals and slots*

3.11.2 Member Function Documentation

3.11.2.1 int XJoystick::current ( ) [inline]

Returns the current selected joystick, -1 if there's no selected joystick

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

- xjoystick.h
- xjoystick.cpp

Index