SerialBasic

Generated by Doxygen 1.8.5

Sun Mar 2 2014 19:04:52

Contents

Chapter 1

Class Index

1.1 Class List

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

Chapter 2

File Index

2.1 File List

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

Chapter 3

Class Documentation

3.1 SerialBasic< Type > Class Template Reference

Writes and reads data to a serial port with 8N1, for a particular Type.

`#include <SerialBasic.hpp>`

Public Types

- typedef uint8_t **Byte**

Public Member Functions

- SerialBasic (uint16_t comPort, uint32_t baudRate)

  Attempt to open serial port with a given comPort and baudRate.
- ~SerialBasic ()

  Destroy SerialBasic object.
- boost::system::error_code & getErrorCode ()

  Get boost error code.
- template<class BeginIterator >
  std::size_t read (BeginIterator beginIterator, std::size_t size)

  Read serial data from the SerialBasic object's buffer (non-blocking)
- template<class BeginIterator >
  void write (BeginIterator beginIterator, std::size_t size)

  Write serial data to the serial port (blocking)

3.1.1 Detailed Description

template<class Type = uint8_t>class SerialBasic< Type >

Writes and reads data to a serial port with 8N1, for a particular Type.

Since SerialBasic was originally developed for communication with the Xbee transcievers, SerialBasic is generalized such that different data types can be serially transmitted. The default type of uint8_t is introduced with the assumption most developers will only transmit uint8_t arrays or other containers (e.g. vectors, lists, etc.).

Please note SerialBasic is developed with boost 1.55 and depends on boost threads (thus, ensure the boost compiled binaries are linked with the application utilizing SerialBasic). SerialBasic also depends on lambda functions from the C++0x standard. SerialBasic is also intended for Windows operating systems, however the constructor method can be modified for other operating systems supported by boost.

See Also

www.boost.org

3.1.2 Constructor & Destructor Documentation

3.1.2.1 template<class Type > SerialBasic< Type >::SerialBasic (uint16_t *comPort*, uint32_t *baudRate*)

Attempt to open serial port with a given comPort and baudRate.

Parameters

comPort	The COM port
baudRate	The baudrate

Exceptions

boost::system::system_error	Thrown if the attempt to open the serial port failed. Check boost error code to find out the reason of the failure.

3.1.3 Member Function Documentation

3.1.3.1 template<class Type > boost::system::error_code & SerialBasic< Type >::getErrorCode ( )

Get boost error code.

The error code can be used to verify whether a previously established connection is lost and why.

Returns

The boost error code.

3.1.3.2 template<class Type > template<class BeginIterator > std::size_t SerialBasic< Type >::read (BeginIterator *beginIterator*, std::size_t *size*)

Read serial data from the SerialBasic object's buffer (non-blocking)

Parameters

beginIterator	The starting location of where the data is saved. Can be a pointer to an array or an iterator of a container.
size	The maximum amount of data to copy from the SerialBasic object's buffer. For instance, if beginIterator is a pointer to an array of characters, size is the maxmimum amount of characters copied from the SerialBasic object's buffer. It is up to the developer to ensure the buffer to which beginIterator refers has the capacity to contain the maximum amount of requested data.

Returns

The actual amount of data saved.

3.1.3.3 template<class Type > template<class BeginIterator > void SerialBasic< Type >::write (BeginIterator *beginIterator*, std::size_t *size*)

Write serial data to the serial port (blocking)

Parameters

beginIterator	The starting location of where the data is taken. Can be a pointer to an array or an iterator of a container.
size	The maximum amount of data to write to the serial port. For instance, if beginIterator is a pointer to an array of characters, size is the maximum amount of characters written to the serial port associated with the SerialBasic object. It is up to the developer to ensure the buffer to which beginIterator refers is at least as large as the specified size.

Exceptions

boost::system::system_error	Thrown if there is a failure to write data to the serial port. Check boost error code to find out the reason of the failure.

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

- C:/Users/Andrew Powell/Documents/C/Learn Boost/SerialBasic/SerialBasic/SerialBasic.hpp

Chapter 4

File Documentation

4.1 C:/Users/Andrew Powell/Documents/C/Learn Boost/SerialBasic/SerialBasic/SerialBasic.hpp File Reference

```
#include <string>
#include <boost/thread.hpp>
#include <boost/bind.hpp>
#include <boost/asio.hpp>
#include <boost/system/error_code.hpp>
#include <cstdint>
#include <list>
#include <sstream>
#include <iterator>
#include <memory>
```

Classes

- class SerialBasic< Type >

  Writes and reads data to a serial port with 8N1, for a particular Type.
- class SerialBasic< Type >

  Writes and reads data to a serial port with 8N1, for a particular Type.

Typedefs

- typedef SerialBasic **Serial**

4.1.1 Detailed Description

Author

Andrew Powell andrew.powell@temple.edu

Version

1.0

4.1.2 LICENSE

Index