

STEVAL UART COMMUNICATION

Generated by Doxygen 1.8.13

Contents

1	Class Index	1
1.1	Class List	1
2	File Index	3
2.1	File List	3
3	Class Documentation	5
3.1	Frame Struct Reference	5
3.1.1	Detailed Description	5
3.1.2	Constructor & Destructor Documentation	6
3.1.2.1	Frame()	6
3.1.3	Member Function Documentation	6
3.1.3.1	getCommad()	6
3.1.3.2	reverse()	6
3.2	UART Struct Reference	7
3.2.1	Detailed Description	7

4 File Documentation	9
4.1 registers.h File Reference	9
4.2 uartSteval.cpp File Reference	9
4.2.1 Detailed Description	10
4.2.2 Function Documentation	10
4.2.2.1 GetRegistry()	10
4.2.2.2 receive()	10
4.2.2.3 send()	11
4.2.2.4 SetMotorRefSpeed()	11
4.2.2.5 SetRegistry()	11
4.2.2.6 StartMotor()	12
4.2.2.7 StopMotor()	12
4.3 uartSteval.h File Reference	13
4.3.1 Detailed Description	14
4.3.2 Function Documentation	14
4.3.2.1 GetRegistry()	14
4.3.2.2 receive()	14
4.3.2.3 send()	15
4.3.2.4 SetMotorRefSpeed()	15
4.3.2.5 SetRegistry()	15
4.3.2.6 StartMotor()	16
4.3.2.7 StopMotor()	16
Index	19

Chapter 1

Class Index

1.1 Class List

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

Frame	Uart frame obejct	5
UART	Uart connection object	7

Chapter 2

File Index

2.1 File List

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

registers.h	9
uartSteval.cpp	9
uartSteval.h	13

Chapter 3

Class Documentation

3.1 Frame Struct Reference

Uart frame obejct.

```
#include <uartSteval.h>
```

Public Member Functions

- [Frame](#) ()
Default construct a new [Frame](#) object.
- [Frame](#) (int motorId, [FRAME_CODES](#) frameCode, [STEVAL_REGISTERS](#) registerId, [STEVAL_REGISTER](#)←
[S_LEN](#) payload_len, int data)
Construct a new [Frame](#) object.
- int [getCommad](#) ()
Get the Commad object as int.
- int [reverse](#) (int b, int l)
Bitwise inversion of a number.

Public Attributes

- int **motorId**
- [FRAME_CODES](#) **frameCode**
- [STEVAL_REGISTERS](#) **reg**
- [STEVAL_REGISTERS_LEN](#) **regLen**
- int **data**
- int **CRC**

3.1.1 Detailed Description

Uart frame obejct.

3.1.2 Constructor & Destructor Documentation

3.1.2.1 Frame()

```
Frame::Frame (
    int motorid,
    FRAME_CODES framecode,
    STEVAL_REGISTERS registerId,
    STEVAL_REGISTERS_LEN payload_len,
    int data ) [inline]
```

Construct a new [Frame](#) object.

Parameters

<i>motorid</i>	Motor id
<i>framecode</i>	Frame code from FRAME_CODES enum
<i>registerId</i>	Motor control register id
<i>payload_len</i>	Motor control register payload length
<i>data</i>	Data to send, if no data put NO_DATA

3.1.3 Member Function Documentation

3.1.3.1 getCommad()

```
int Frame::getCommad ( ) [inline]
```

Get the Commad object as int.

Returns

int Command as single int

3.1.3.2 reverse()

```
int Frame::reverse (
    int b,
    int l ) [inline]
```

Bitwise inversion of a number.

Parameters

<i>b</i>	int number to reverse
<i>l</i>	Bitwise lenght of number

Returns

int inverst number

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

- [uartSteval.h](#)

3.2 UART Struct Reference

Uart connection object.

```
#include <uartSteval.h>
```

Public Attributes

- int **baud**
- char * **device**

3.2.1 Detailed Description

Uart connection object.

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

- [uartSteval.h](#)

Chapter 4

File Documentation

4.1 registers.h File Reference

This graph shows which files directly or indirectly include this file:

4.2 uartSteval.cpp File Reference

```
#include <ctype.h>
#include <stdio.h>
#include <wiringPi.h>
#include <iostream>
#include <unistd.h>
#include <wiringSerial.h>
#include "uartSteval.h"
```

Include dependency graph for uartSteval.cpp:

Functions

- **UART_STATUS receive** (**Frame** *cmd, int connection)
Receive uart frame.
- **UART_STATUS sendData** (int con, int data, int l, bool cmd_show=false)
- **UART_STATUS send** (**Frame** cmd, **UART** uart, bool cmd_show=false)
Send data in frame.
- **UART_STATUS StartMotor** (int motorId, **UART** uart)
Execute start motor command.
- **UART_STATUS StopMotor** (int motorId, **UART** uart)
Execute stop motor command.
- **UART_STATUS SetMotorRefSpeed** (int ref, int motorId, **UART** uart)
Set the motor ramp final speed registry.
- **UART_STATUS SetRegistry** (**STEVAL_REGISTERS** reg, **STEVAL_REGISTERS_LEN** regL, int motorId, **UART** uart, int data)
Set the registry value.
- **UART_STATUS GetRegistry** (**STEVAL_REGISTERS** reg, **STEVAL_REGISTERS_LEN** regL, int motorId, **UART** uart)
Get the registry value.

4.2.1 Detailed Description

Created on: Nov 27, 2020 Author: Jakub Walkowski

4.2.2 Function Documentation

4.2.2.1 GetRegistry()

```
UART_STATUS GetRegistry (
    STEVAL_REGISTERS reg,
    STEVAL_REGISTERS_LEN regL,
    int motorId,
    UART uart )
```

Get the registry value.

Parameters

<i>reg</i>	Motor control register id
<i>regL</i>	Montrol register payload length
<i>motorId</i>	Motor id
<i>uart</i>	Uart object

Returns

UART_STATUS Getting resoult

4.2.2.2 receive()

```
UART_STATUS receive (
    Frame * cmd,
    int connection )
```

Receive uart frame.

Parameters

<i>cmd</i>	Received frame
<i>connection</i>	Uart connection id

Returns

UART_STATUS Receiving status

4.2.2.3 send()

```
UART_STATUS send (
    Frame cmd,
    UART uart,
    bool cmd_show = false )
```

Send data in frame.

Parameters

<i>con</i>	Uart conneciton id
<i>data</i>	Data to send
<i>l</i>	Control register payload length
<i>cmd_show</i>	Optional param, if true show sending fame in console

Returns

UART_STATUS Sending resoult

4.2.2.4 SetMotorRefSpeed()

```
UART_STATUS SetMotorRefSpeed (
    int ref,
    int motorId,
    UART uart )
```

Set the motor ramp final speed registry.

Parameters

<i>ref</i>	Value to set
<i>motor</i> ↔ <i>Id</i>	Motor id
<i>uart</i>	Uart object

Returns

UART_STATUS Setting resoult

4.2.2.5 SetRegistry()

```
UART_STATUS SetRegistry (
    STEVAL_REGISTERS reg,
```

```
STEVAL_REGISTERS_LEN regL,  
int motorId,  
UART uart,  
int data )
```

Set the registry value.

Parameters

<i>reg</i>	Motor control register id
<i>regL</i>	Montrol register payload length
<i>motor↔ Id</i>	Motor id
<i>uart</i>	Uart object
<i>data</i>	Value to set

Returns

UART_STATUS Setting resoult

4.2.2.6 StartMotor()

```
UART_STATUS StartMotor (  
    int motorId,  
    UART uart )
```

Execute start motor command.

Parameters

<i>motor↔ Id</i>	Motor id
<i>uart</i>	Uart object

Returns

UART_STATUS Starting motor resoult

4.2.2.7 StopMotor()

```
UART_STATUS StopMotor (  
    int motorId,  
    UART uart )
```

Execute stop motor command.

Parameters

<i>motor↔ Id</i>	Motor id
<i>uart</i>	Uart object

Returns

UART_STATUS Stopping motor resoult

4.3 uartSteval.h File Reference

```
#include <climits>
```

```
#include "registers.h"
```

Include dependency graph for uartSteval.h: This graph shows which files directly or indirectly include this file:

Classes

- struct [UART](#)
Uart connection object.
- struct [Frame](#)
Uart frame obejct.

Macros

- #define **NO_DATA** INT_MAX

Functions

- [UART_STATUS receive](#) ([Frame](#) *cmd, int connection)
Receive uart frame.
- [UART_STATUS send](#) ([Frame](#) cmd, [UART](#) uart, bool cmd_show=false)
Send data in frame.
- [UART_STATUS StartMotor](#) (int motorId, [UART](#) uart)
Execute start motor command.
- [UART_STATUS StopMotor](#) (int motorId, [UART](#) uart)
Execute stop motor command.
- [UART_STATUS SetMotorRefSpeed](#) (int ref, int motorId, [UART](#) uart)
Set the motor ramp final speed registry.
- [UART_STATUS SetRegistry](#) ([STEVAL_REGISTERS](#) reg, [STEVAL_REGISTERS_LEN](#) regL, int motorId, [U↔
ART](#) uart, int data)
Set the registry value.
- [UART_STATUS GetRegistry](#) ([STEVAL_REGISTERS](#) reg, [STEVAL_REGISTERS_LEN](#) regL, int motorId, [UART](#) uart)
Get the registry value.

4.3.1 Detailed Description

Created on: Nov 27, 2020 Author: walko

4.3.2 Function Documentation

4.3.2.1 GetRegistry()

```
UART_STATUS GetRegistry (
    STEVAL_REGISTERS reg,
    STEVAL_REGISTERS_LEN regL,
    int motorId,
    UART uart )
```

Get the registry value.

Parameters

<i>reg</i>	Motor control register id
<i>regL</i>	Montrol register payload length
<i>motorId</i>	Motor id
<i>uart</i>	Uart object

Returns

UART_STATUS Getting resoult

4.3.2.2 receive()

```
UART_STATUS receive (
    Frame * cmd,
    int connection )
```

Receive uart frame.

Parameters

<i>cmd</i>	Received frame
<i>connection</i>	Uart connection id

Returns

UART_STATUS Receiving status

4.3.2.3 send()

```
UART_STATUS send (
    Frame cmd,
    UART uart,
    bool cmd_show = false )
```

Send data in frame.

Parameters

<i>con</i>	Uart conneciton id
<i>data</i>	Data to send
<i>l</i>	Control register payload length
<i>cmd_show</i>	Optional param, if true show sending fame in console

Returns

UART_STATUS Sending resoult

4.3.2.4 SetMotorRefSpeed()

```
UART_STATUS SetMotorRefSpeed (
    int ref,
    int motorId,
    UART uart )
```

Set the motor ramp final speed registry.

Parameters

<i>ref</i>	Value to set
<i>motor</i> ↔ <i>Id</i>	Motor id
<i>uart</i>	Uart object

Returns

UART_STATUS Setting resoult

4.3.2.5 SetRegistry()

```
UART_STATUS SetRegistry (
    STEVAL_REGISTERS reg,
```

```
STEVAL_REGISTERS_LEN regL,  
int motorId,  
UART uart,  
int data )
```

Set the registry value.

Parameters

<i>reg</i>	Motor control register id
<i>regL</i>	Montrol register payload length
<i>motor↔ Id</i>	Motor id
<i>uart</i>	Uart object
<i>data</i>	Value to set

Returns

UART_STATUS Setting resoult

4.3.2.6 StartMotor()

```
UART_STATUS StartMotor (  
    int motorId,  
    UART uart )
```

Execute start motor command.

Parameters

<i>motor↔ Id</i>	Motor id
<i>uart</i>	Uart object

Returns

UART_STATUS Starting motor resoult

4.3.2.7 StopMotor()

```
UART_STATUS StopMotor (  
    int motorId,  
    UART uart )
```

Execute stop motor command.

Parameters

<i>motor↔ id</i>	Motor id
<i>uart</i>	Uart object

Returns

UART_STATUS Stopping motor resoult

Index

- Frame, [5](#)
 - Frame, [6](#)
 - getCommad, [6](#)
 - reverse, [6](#)
- getCommad
 - Frame, [6](#)
- GetRegistry
 - uartSteval.cpp, [10](#)
 - uartSteval.h, [14](#)
- receive
 - uartSteval.cpp, [10](#)
 - uartSteval.h, [14](#)
- registers.h, [9](#)
- reverse
 - Frame, [6](#)
- send
 - uartSteval.cpp, [11](#)
 - uartSteval.h, [15](#)
- SetMotorRefSpeed
 - uartSteval.cpp, [11](#)
 - uartSteval.h, [15](#)
- SetRegistry
 - uartSteval.cpp, [11](#)
 - uartSteval.h, [15](#)
- StartMotor
 - uartSteval.cpp, [12](#)
 - uartSteval.h, [16](#)
- StopMotor
 - uartSteval.cpp, [12](#)
 - uartSteval.h, [16](#)
- UART, [7](#)
- uartSteval.cpp, [9](#)
 - GetRegistry, [10](#)
 - receive, [10](#)
 - send, [11](#)
 - SetMotorRefSpeed, [11](#)
 - SetRegistry, [11](#)
 - StartMotor, [12](#)
 - StopMotor, [12](#)
- uartSteval.h, [13](#)
 - GetRegistry, [14](#)
 - receive, [14](#)
 - send, [15](#)
 - SetMotorRefSpeed, [15](#)
 - SetRegistry, [15](#)
 - StartMotor, [16](#)
 - StopMotor, [16](#)