

CE USART

Generated by Doxygen 1.8.11

Contents

1	CE USART Project	1
2	CE USART Project	3
3	Data Structure Index	5
3.1	Data Structures	5
4	File Index	7
4.1	File List	7
5	Data Structure Documentation	9
5.1	LedInterface Struct Reference	9
5.1.1	Detailed Description	9
5.1.2	Field Documentation	9
5.1.2.1	Init	9
5.1.2.2	Off	9
5.1.2.3	On	9
5.1.2.4	Toggle	10
5.2	SerialInterface Struct Reference	10
5.2.1	Detailed Description	10
5.2.2	Field Documentation	10
5.2.2.1	Close	10
5.2.2.2	GetByte	10
5.2.2.3	IsOpen	10
5.2.2.4	Open	10
5.2.2.5	RxBufferHasData	11
5.2.2.6	SendArray	11
5.2.2.7	SendByte	11
5.2.2.8	SendString	11
5.3	TickType Struct Reference	11
5.3.1	Detailed Description	11
5.3.2	Field Documentation	11
5.3.2.1	DelayMs	11
5.3.2.2	StartMs	11

6 File Documentation	13
6.1 include/common.h File Reference	13
6.1.1 Macro Definition Documentation	13
6.1.1.1 COMPILED_DATA_TIME	13
6.1.1.2 EN_DEBUG_INTERFACE	13
6.1.1.3 FALSE	13
6.1.1.4 FIRMWARE_VERSION	14
6.1.1.5 HARDWARE_VERSION	14
6.1.1.6 TRUE	14
6.1.1.7 USART_OVER_SAMPLE_16	14
6.2 include/led_interface.h File Reference	14
6.3 include/MCU/device.h File Reference	14
6.3.1 Macro Definition Documentation	14
6.3.1.1 __FPU_PRESENT	14
6.4 include/MCU/LED/blue_led.h File Reference	14
6.4.1 Variable Documentation	15
6.4.1.1 RedLed	15
6.5 include/MCU/LED/green_led.h File Reference	15
6.5.1 Variable Documentation	15
6.5.1.1 GreenLed	15
6.6 include/MCU/LED/red_led.h File Reference	15
6.6.1 Variable Documentation	16
6.6.1.1 BlueLed	16
6.7 include/MCU/tick.h File Reference	16
6.7.1 Function Documentation	16
6.7.1.1 Tick_DelayMs(uint32_t delayMs)	16
6.7.1.2 Tick_DelayMs_NonBlocking(uint_fast8_t reset, TickType *config)	16
6.7.1.3 Tick_GetMs(void)	17
6.7.1.4 Tick_Init(void)	17
6.8 include/MCU/usart3.h File Reference	17

6.8.1	Variable Documentation	17
6.8.1.1	SerialPort3	17
6.9	include/serial_interface.h File Reference	17
6.10	Mainpage.md File Reference	18
6.11	README.md File Reference	18
6.12	src/main.c File Reference	18
6.12.1	Detailed Description	18
6.12.2	Function Documentation	18
6.12.2.1	main(void)	18
6.12.2.2	PrintHeader()	18
6.13	src/MCU/LED/blue_led.c File Reference	19
6.13.1	Detailed Description	19
6.13.2	Macro Definition Documentation	19
6.13.2.1	LED_PIN_BR	19
6.13.2.2	LED_PIN_BS	20
6.13.2.3	LED_PIN_ODR	20
6.13.3	Function Documentation	20
6.13.3.1	Led_Init(void)	20
6.13.3.2	Led_Off(void)	20
6.13.3.3	Led_On(void)	20
6.13.3.4	Led_Toggle(void)	20
6.13.4	Variable Documentation	20
6.13.4.1	BlueLed	20
6.14	src/MCU/LED/green_led.c File Reference	21
6.14.1	Detailed Description	21
6.14.2	Macro Definition Documentation	21
6.14.2.1	LED_PIN_BR	21
6.14.2.2	LED_PIN_BS	22
6.14.2.3	LED_PIN_ODR	22
6.14.3	Function Documentation	22

6.14.3.1	Led_Init(void)	22
6.14.3.2	Led_Off(void)	22
6.14.3.3	Led_On(void)	22
6.14.3.4	Led_Toggle(void)	22
6.14.4	Variable Documentation	22
6.14.4.1	GreenLed	22
6.15	src/MCU/LED/red_led.c File Reference	23
6.15.1	Detailed Description	23
6.15.2	Macro Definition Documentation	23
6.15.2.1	LED_PIN_BR	23
6.15.2.2	LED_PIN_BS	24
6.15.2.3	LED_PIN_ODR	24
6.15.3	Function Documentation	24
6.15.3.1	Led_Init(void)	24
6.15.3.2	Led_Off(void)	24
6.15.3.3	Led_On(void)	24
6.15.3.4	Led_Toggle(void)	24
6.15.4	Variable Documentation	24
6.15.4.1	RedLed	24
6.16	src/MCU/tick.c File Reference	25
6.16.1	Detailed Description	25
6.16.2	Macro Definition Documentation	25
6.16.2.1	TIMER_FREQUENCY_HZ	25
6.16.3	Function Documentation	26
6.16.3.1	SysTick_Handler(void)	26
6.16.3.2	Tick_DelayMs(uint32_t delayMs)	26
6.16.3.3	Tick_DelayMs_NonBlocking(uint_fast8_t reset, TickType *config)	26
6.16.3.4	Tick_GetMs(void)	26
6.16.3.5	Tick_Init(void)	26
6.16.4	Variable Documentation	26

6.16.4.1 TickCounter	26
6.17 src/MCU/usart3.c File Reference	26
6.17.1 Detailed Description	27
6.17.2 Macro Definition Documentation	28
6.17.2.1 UART_BRR_SAMPLING16	28
6.17.2.2 UART_BRR_SAMPLING8	28
6.17.2.3 UART_DIV_SAMPLING16	28
6.17.2.4 UART_DIV_SAMPLING8	28
6.17.2.5 UART_DIVFRAQ_SAMPLING16	28
6.17.2.6 UART_DIVFRAQ_SAMPLING8	28
6.17.2.7 UART_DIVMANT_SAMPLING16	29
6.17.2.8 UART_DIVMANT_SAMPLING8	29
6.17.3 Function Documentation	29
6.17.3.1 Close(void)	29
6.17.3.2 GetByte(uint8_t *destination)	29
6.17.3.3 IsOpen(void)	29
6.17.3.4 IsWriteBusy(void)	29
6.17.3.5 Open(uint32_t baudrate)	29
6.17.3.6 RxBufferHasData(void)	30
6.17.3.7 SendArray(const uint8_t *source, uint32_t length)	30
6.17.3.8 SendByte(uint8_t source)	31
6.17.3.9 SendString(const uint8_t *source)	31
6.17.4 Variable Documentation	31
6.17.4.1 IsOpenFlag	31
6.17.4.2 SerialPort3	32
Index	33

Chapter 1

CE USART Project

This project is based on Contextual Eelectornics course that brings up a the USART peripheral on STM32 micro.

In this specific example the board I have is a Nucleo F446ZE and this particular project implements a dead simple "serial interface".

The board is setup to run at a 115200 buadrate and is essentially an echo server with the following characters additionally toggling leds

- 1 -> Toggles Green Led on board
- 2 -> Toggles Blue Led on board
- 3 -> Toggles Red Led on board

On boot the board will will setup the Initialize all the device and then print a header out to the USART 3 device that includes the firmware and hardware version along with the date compiled.

It's basic but a start.

Repo this is based off of

Chapter 2

CE USART Project

This project is based on Contextual Eelectornics course that brings up a the USART peripheral on STM32 micro.

In this specific example the board I have is a Nucleo F446ZE and this particular project implements a dead simple "serial interface".

The board is setup to run at a 115200 buadrate and is essentially an echo server with the following characters additionally toggling leds

- 1 -> Toggles Green Led on board
- 2 -> Toggles Blue Led on board
- 3 -> Toggles Red Led on board

On boot the board will will setup the Initialize all the device and then print a header out to the USART 3 device that includes the firmware and hardware version along with the date compiled.

It's basic but a start.

Repo this is based off of

Chapter 3

Data Structure Index

3.1 Data Structures

Here are the data structures with brief descriptions:

LedInterface	9
SerialInterface	10
TickType	11

Chapter 4

File Index

4.1 File List

Here is a list of all files with brief descriptions:

include/common.h	13
include/led_interface.h	14
include/serial_interface.h	17
include/MCU/device.h	14
include/MCU/tick.h	16
include/MCU/usart3.h	17
include/MCU/LED/blue_led.h	14
include/MCU/LED/green_led.h	15
include/MCU/LED/red_led.h	15
src/main.c	
Entry point	18
src/MCU/tick.c	
Mili-Second systick implementation	25
src/MCU/usart3.c	
USART driver	26
src/MCU/LED/blue_led.c	
Led Driver	19
src/MCU/LED/green_led.c	
Led Driver	21
src/MCU/LED/red_led.c	
Led Driver	23

Chapter 5

Data Structure Documentation

5.1 LedInterface Struct Reference

```
#include <led_interface.h>
```

Data Fields

- void(* [On](#))(void)
- void(* [Off](#))(void)
- void(* [Toggle](#))(void)
- void(* [Init](#))(void)

5.1.1 Detailed Description

Definition at line 11 of file led_interface.h.

5.1.2 Field Documentation

5.1.2.1 void(* Init) (void)

Definition at line 15 of file led_interface.h.

5.1.2.2 void(* Off) (void)

Definition at line 13 of file led_interface.h.

5.1.2.3 void(* On) (void)

Definition at line 12 of file led_interface.h.

5.1.2.4 void(* Toggle) (void)

Definition at line 14 of file led_interface.h.

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

- include/[led_interface.h](#)

5.2 SerialInterface Struct Reference

```
#include <serial_interface.h>
```

Data Fields

- uint_fast8_t(* [IsOpen](#))(void)
- uint_fast8_t(* [Open](#))(uint32_t baudrate)
- void(* [Close](#))(void)
- uint_fast8_t(* [SendByte](#))(uint8_t)
- uint_fast8_t(* [SendString](#))(const uint8_t *source)
- uint_fast8_t(* [SendArray](#))(const uint8_t *source, uint32_t length)
- int_fast8_t(* [RxBufferHasData](#))(void)
- int_fast8_t(* [GetByte](#))(uint8_t *destination)

5.2.1 Detailed Description

Definition at line 6 of file serial_interface.h.

5.2.2 Field Documentation

5.2.2.1 void(* Close) (void)

Definition at line 9 of file serial_interface.h.

5.2.2.2 int_fast8_t(* GetByte) (uint8_t *destination)

Definition at line 14 of file serial_interface.h.

5.2.2.3 uint_fast8_t(* IsOpen) (void)

Definition at line 7 of file serial_interface.h.

5.2.2.4 uint_fast8_t(* Open) (uint32_t baudrate)

Definition at line 8 of file serial_interface.h.

5.2.2.5 int_fast8_t(* RxBufferHasData) (void)

Definition at line 13 of file serial_interface.h.

5.2.2.6 uint_fast8_t(* SendArray) (const uint8_t *source, uint32_t length)

Definition at line 12 of file serial_interface.h.

5.2.2.7 uint_fast8_t(* SendByte) (uint8_t)

Definition at line 10 of file serial_interface.h.

5.2.2.8 uint_fast8_t(* SendString) (const uint8_t *source)

Definition at line 11 of file serial_interface.h.

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

- [include/serial_interface.h](#)

5.3 TickType Struct Reference

```
#include <tick.h>
```

Data Fields

- uint32_t [StartMs](#)
- uint32_t [DelayMs](#)

5.3.1 Detailed Description

Definition at line 6 of file tick.h.

5.3.2 Field Documentation

5.3.2.1 uint32_t DelayMs

Definition at line 8 of file tick.h.

5.3.2.2 uint32_t StartMs

Definition at line 7 of file tick.h.

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

- [include/MCU/tick.h](#)

Chapter 6

File Documentation

6.1 include/common.h File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include "MCU/device.h"
```

Macros

- `#define TRUE 1`
- `#define FALSE 0`
- `#define FIRMWARE_VERSION "00.001D"`
- `#define HARDWARE_VERSION "01"`
- `#define COMPILED_DATA_TIME "[" __DATE__ " " __TIME__ "]"`
- `#define EN_DEBUG_INTERFACE`
- `#define USART_OVER_SAMPLE_16 1`

6.1.1 Macro Definition Documentation

6.1.1.1 `#define COMPILED_DATA_TIME "[" __DATE__ " " __TIME__ "]"`

Definition at line 13 of file common.h.

6.1.1.2 `#define EN_DEBUG_INTERFACE`

Definition at line 15 of file common.h.

6.1.1.3 `#define FALSE 0`

Definition at line 9 of file common.h.

6.1.1.4 `#define FIRMWARE_VERSION "00.001D"`

Definition at line 11 of file common.h.

6.1.1.5 `#define HARDWARE_VERSION "01"`

Definition at line 12 of file common.h.

6.1.1.6 `#define TRUE 1`

Definition at line 8 of file common.h.

6.1.1.7 `#define USART_OVER_SAMPLE_16 1`

Definition at line 16 of file common.h.

6.2 `include/led_interface.h` File Reference

Data Structures

- struct [LedInterface](#)

6.3 `include/MCU/device.h` File Reference

```
#include "stm32f4xx.h"
```

Macros

- `#define __FPU_PRESENT 0U`

6.3.1 Macro Definition Documentation

6.3.1.1 `#define __FPU_PRESENT 0U`

Definition at line 7 of file device.h.

6.4 `include/MCU/LED/blue_led.h` File Reference

```
#include "common.h"  
#include "led_interface.h"
```

Variables

- [LedInterface RedLed](#)

Setup [LedInterface](#) Interface.

6.4.1 Variable Documentation

6.4.1.1 LedInterface RedLed

Setup [LedInterface](#) Interface.

Definition at line 64 of file red_led.c.

6.5 include/MCU/LED/green_led.h File Reference

```
#include "common.h"
#include "led_interface.h"
```

Variables

- [LedInterface GreenLed](#)

Setup [LedInterface](#) Interface.

6.5.1 Variable Documentation

6.5.1.1 LedInterface GreenLed

Setup [LedInterface](#) Interface.

Definition at line 64 of file green_led.c.

6.6 include/MCU/LED/red_led.h File Reference

```
#include "common.h"
#include "led_interface.h"
```

Variables

- [LedInterface BlueLed](#)

Setup [LedInterface](#) Interface.

6.6.1 Variable Documentation

6.6.1.1 LedInterface BlueLed

Setup [LedInterface](#) Interface.

Definition at line 64 of file blue_led.c.

6.7 include/MCU/tick.h File Reference

```
#include "common.h"
```

Data Structures

- struct [TickType](#)

Functions

- void [Tick_Init](#) (void)
Initialize systick.
- uint32_t [Tick_GetMs](#) (void)
Get systick.
- int_fast8_t [Tick_DelayMs_NonBlocking](#) (uint_fast8_t reset, [TickType](#) *config)
Non-Blocking mili-second.
- void [Tick_DelayMs](#) (uint32_t delayMs)
Blocking mili-second resolution delay.

6.7.1 Function Documentation

6.7.1.1 void Tick_DelayMs (uint32_t delayMs)

Blocking mili-second resolution delay.

Definition at line 33 of file tick.c.

6.7.1.2 int_fast8_t Tick_DelayMs_NonBlocking (uint_fast8_t reset, TickType * config)

Non-Blocking mili-second.

< invalid pointer

Definition at line 43 of file tick.c.

6.7.1.3 uint32_t Tick_GetMs (void)

Get systick.

Definition at line 26 of file tick.c.

6.7.1.4 void Tick_Init (void)

Initialize systick.

Definition at line 19 of file tick.c.

6.8 include/MCU/usart3.h File Reference

```
#include "common.h"
#include "serial_interface.h"
```

Variables

- [SerialInterface SerialPort3](#)
Setup [SerialInterface](#) struct.

6.8.1 Variable Documentation

6.8.1.1 SerialInterface SerialPort3

Setup [SerialInterface](#) struct.

Definition at line 192 of file usart3.c.

6.9 include/serial_interface.h File Reference

```
#include "common.h"
```

Data Structures

- struct [SerialInterface](#)

6.10 Mainpage.md File Reference

6.11 README.md File Reference

6.12 src/main.c File Reference

Entry point.

```
#include "common.h"
#include <MCU/LED/green_led.h>
#include <MCU/LED/blue_led.h>
#include <MCU/LED/red_led.h>
#include "MCU/tick.h"
#include "MCU/usart3.h"
```

Functions

- static void `PrintHeader` ()
Print serial header.
- int `main` (void)
Main function.

6.12.1 Detailed Description

Entry point.

Author

Matthew Philyaw (matthew.philyaw@gmail.com)

6.12.2 Function Documentation

6.12.2.1 int main (void)

Main function.

Definition at line 21 of file main.c.

6.12.2.2 static void PrintHeader () [static]

Print serial header.

Prints a header out to the USART device that contains details about the firmware/hardware version

Definition at line 60 of file main.c.

6.13 src/MCU/LED/blue_led.c File Reference

Led Driver.

```
#include <MCU/LED/blue_led.h>
```

Macros

- `#define LED_PIN_BS GPIO_BSRR_BS_7`
GPIO defines for driving the output.
- `#define LED_PIN_BR GPIO_BSRR_BR_7`
- `#define LED_PIN_ODR GPIO_ODR_ODR_7`

Functions

- static void `Led_On` (void)
Turn Led on.
- static void `Led_Off` (void)
Turn Led off.
- static void `Led_Toggle` (void)
Toggle Led.
- static void `Led_Init` (void)
Init Led.

Variables

- `LedInterface BlueLed`
Setup `LedInterface` Interface.

6.13.1 Detailed Description

Led Driver.

Author

Matthew Philyaw Blue Led on the Nucleo F446ZE board

6.13.2 Macro Definition Documentation

6.13.2.1 `#define LED_PIN_BR GPIO_BSRR_BR_7`

Definition at line 15 of file `blue_led.c`.

6.13.2.2 `#define LED_PIN_BS GPIO_BSRR_BS_7`

GPIO defines for driving the output.

Definition at line 14 of file `blue_led.c`.

6.13.2.3 `#define LED_PIN_ODR GPIO_ODR_ODR_7`

Definition at line 16 of file `blue_led.c`.

6.13.3 Function Documentation

6.13.3.1 `static void Led_Init (void) [static]`

Init Led.

This will be specific to the Led in use

Definition at line 49 of file `blue_led.c`.

6.13.3.2 `static void Led_Off (void) [static]`

Turn Led off.

Definition at line 28 of file `blue_led.c`.

6.13.3.3 `static void Led_On (void) [static]`

Turn Led on.

Definition at line 21 of file `blue_led.c`.

6.13.3.4 `static void Led_Toggle (void) [static]`

Toggle Led.

Definition at line 35 of file `blue_led.c`.

6.13.4 Variable Documentation

6.13.4.1 `LedInterface BlueLed`

Initial value:

```
= {  
    Led_On,  
    Led_Off,  
    Led_Toggle,  
    Led_Init  
}
```

Setup `LedInterface` Interface.

Definition at line 64 of file `blue_led.c`.

6.14 src/MCU/LED/green_led.c File Reference

Led Driver.

```
#include <MCU/LED/blue_led.h>
```

Macros

- `#define LED_PIN_BS GPIO_BSRR_BS_0`
GPIO defines for driving the output.
- `#define LED_PIN_BR GPIO_BSRR_BR_0`
- `#define LED_PIN_ODR GPIO_ODR_ODR_0`

Functions

- static void `Led_On` (void)
Turn Led on.
- static void `Led_Off` (void)
Turn Led off.
- static void `Led_Toggle` (void)
Toggle Led.
- static void `Led_Init` (void)
Init Led.

Variables

- `LedInterface GreenLed`
Setup `LedInterface` Interface.

6.14.1 Detailed Description

Led Driver.

Author

Matthew Philyaw Green Led on the Nucleo F446ZE board

6.14.2 Macro Definition Documentation

6.14.2.1 `#define LED_PIN_BR GPIO_BSRR_BR_0`

Definition at line 15 of file `green_led.c`.

6.14.2.2 `#define LED_PIN_BS GPIO_BSRR_BS_0`

GPIO defines for driving the output.

Definition at line 14 of file `green_led.c`.

6.14.2.3 `#define LED_PIN_ODR GPIO_ODR_ODR_0`

Definition at line 16 of file `green_led.c`.

6.14.3 Function Documentation

6.14.3.1 `static void Led_Init (void) [static]`

Init Led.

This will be specific to the Led in use

Definition at line 49 of file `green_led.c`.

6.14.3.2 `static void Led_Off (void) [static]`

Turn Led off.

Definition at line 28 of file `green_led.c`.

6.14.3.3 `static void Led_On (void) [static]`

Turn Led on.

Definition at line 21 of file `green_led.c`.

6.14.3.4 `static void Led_Toggle (void) [static]`

Toggle Led.

Definition at line 35 of file `green_led.c`.

6.14.4 Variable Documentation

6.14.4.1 `LedInterface GreenLed`

Initial value:

```
= {  
    Led_On,  
    Led_Off,  
    Led_Toggle,  
    Led_Init  
}
```

Setup [LedInterface](#) Interface.

Definition at line 64 of file `green_led.c`.

6.15 src/MCU/LED/red_led.c File Reference

Led Driver.

```
#include <MCU/LED/blue_led.h>
```

Macros

- `#define LED_PIN_BS GPIO_BSRR_BS_14`
GPIO defines for driving the output.
- `#define LED_PIN_BR GPIO_BSRR_BR_14`
- `#define LED_PIN_ODR GPIO_ODR_ODR_14`

Functions

- static void `Led_On` (void)
Turn Led on.
- static void `Led_Off` (void)
Turn Led off.
- static void `Led_Toggle` (void)
Toggle Led.
- static void `Led_Init` (void)
Init Led.

Variables

- `LedInterface RedLed`
Setup `LedInterface` Interface.

6.15.1 Detailed Description

Led Driver.

Red Led on the Nucleo F446ZE board

Author

Matthew Philyaw

6.15.2 Macro Definition Documentation

6.15.2.1 `#define LED_PIN_BR GPIO_BSRR_BR_14`

Definition at line 15 of file red_led.c.

6.15.2.2 `#define LED_PIN_BS GPIO_BSRR_BS_14`

GPIO defines for driving the output.

Definition at line 14 of file `red_led.c`.

6.15.2.3 `#define LED_PIN_ODR GPIO_ODR_ODR_14`

Definition at line 16 of file `red_led.c`.

6.15.3 Function Documentation

6.15.3.1 `static void Led_Init (void) [static]`

Init Led.

This will be specific to the Led in use

Definition at line 49 of file `red_led.c`.

6.15.3.2 `static void Led_Off (void) [static]`

Turn Led off.

Definition at line 28 of file `red_led.c`.

6.15.3.3 `static void Led_On (void) [static]`

Turn Led on.

Definition at line 21 of file `red_led.c`.

6.15.3.4 `static void Led_Toggle (void) [static]`

Toggle Led.

Definition at line 35 of file `red_led.c`.

6.15.4 Variable Documentation

6.15.4.1 `LedInterface RedLed`

Initial value:

```
= {  
    Led_On,  
    Led_Off,  
    Led_Toggle,  
    Led_Init  
}
```

Setup `LedInterface` Interface.

Definition at line 64 of file `red_led.c`.

6.16 src/MCU/tick.c File Reference

Mili-Second systick implementation.

```
#include <MCU/tick.h>
```

Macros

- `#define` [TIMER_FREQUENCY_HZ](#) 1000
Define the mili-second resolution.

Functions

- void [Tick_Init](#) (void)
Initialize systick.
- uint32_t [Tick_GetMs](#) (void)
Get systick.
- void [Tick_DelayMs](#) (uint32_t delayMs)
Blocking mili-second resolution delay.
- int_fast8_t [Tick_DelayMs_NonBlocking](#) (uint_fast8_t reset, [TickType](#) *config)
Non-Blocking mili-second.
- void [SysTick_Handler](#) (void)
systick interrupt handler

Variables

- static volatile uint32_t [TickCounter](#)

6.16.1 Detailed Description

Mili-Second systick implementation.

Author

Matthew Philyaw (matthew.philyaw@gmail.com)

6.16.2 Macro Definition Documentation

6.16.2.1 `#define` [TIMER_FREQUENCY_HZ](#) 1000

Define the mili-second resolution.

Definition at line 12 of file tick.c.

6.16.3 Function Documentation

6.16.3.1 void SysTick_Handler (void)

systick interrupt handler

Definition at line 67 of file tick.c.

6.16.3.2 void Tick_DelayMs (uint32_t delayMs)

Blocking mili-second resolution delay.

Definition at line 33 of file tick.c.

6.16.3.3 int_fast8_t Tick_DelayMs_NonBlocking (uint_fast8_t reset, TickType * config)

Non-Blocking mili-second.

< invalid pointer

Definition at line 43 of file tick.c.

6.16.3.4 uint32_t Tick_GetMs (void)

Get systick.

Definition at line 26 of file tick.c.

6.16.3.5 void Tick_Init (void)

Initialize systick.

Definition at line 19 of file tick.c.

6.16.4 Variable Documentation

6.16.4.1 volatile uint32_t TickCounter [static]

Definition at line 14 of file tick.c.

6.17 src/MCU/usart3.c File Reference

USART driver.

```
#include <MCU/usart3.h>
```

Macros

- `#define UART_DIV_SAMPLING16(_PCLK_, _BAUD_) (((_PCLK_)*25U)/(4U*(_BAUD_)))`
Taken from the STM32 HAL.
- `#define UART_DIVMANT_SAMPLING16(_PCLK_, _BAUD_) (UART_DIV_SAMPLING16(_PCLK_, _BAUD_)/100U)`
- `#define UART_DIVFRAQ_SAMPLING16(_PCLK_, _BAUD_) (((UART_DIV_SAMPLING16(_PCLK_, _BAUD_) - (UART_DIVMANT_SAMPLING16(_PCLK_, _BAUD_) * 100U)) * 16U + 50U) / 100U)`
- `#define UART_BRR_SAMPLING16(_PCLK_, _BAUD_)`
- `#define UART_DIV_SAMPLING8(_PCLK_, _BAUD_) (((_PCLK_)*25U)/(2U*(_BAUD_)))`
- `#define UART_DIVMANT_SAMPLING8(_PCLK_, _BAUD_) (UART_DIV_SAMPLING8(_PCLK_, _BAUD_)/100U)`
- `#define UART_DIVFRAQ_SAMPLING8(_PCLK_, _BAUD_) (((UART_DIV_SAMPLING8(_PCLK_, _BAUD_) - (UART_DIVMANT_SAMPLING8(_PCLK_, _BAUD_) * 100U)) * 8U + 50U) / 100U)`
- `#define UART_BRR_SAMPLING8(_PCLK_, _BAUD_)`

Functions

- static uint_fast8_t `IsOpen` (void)
Exposes IsOpenFlag.
- static uint_fast8_t `IsWriteBusy` (void)
Checks to see if USART is writing.
- static uint_fast8_t `Open` (uint32_t baudrate)
Enables and initializes the USART.
- static void `Close` (void)
Close USART.
- static uint_fast8_t `SendByte` (uint8_t source)
Send one byte over the USART.
- static int_fast8_t `RxBufferHasData` (void)
Checks to see if receive buffer has data.
- static int_fast8_t `GetByte` (uint8_t *destination)
Checks to see if receive buffer has data.
- static uint_fast8_t `SendString` (const uint8_t *source)
Send string over the USART.
- static uint_fast8_t `SendArray` (const uint8_t *source, uint32_t length)
Send byte array over USART.

Variables

- static uint_fast8_t `IsOpenFlag` = `FALSE`
Flag to indicate the state of the USART.
- `SerialInterface SerialPort3`
Setup SerialInterface struct.

6.17.1 Detailed Description

USART driver.

Author

Matthew Philyaw (matthew.philyaw@gmail.com)

6.17.2 Macro Definition Documentation

6.17.2.1 #define UART_BRR_SAMPLING16(_PCLK_, _BAUD_)

Value:

```
(( (UART_DIVMANT_SAMPLING16( _PCLK_, _BAUD_ ) << 4U) + \
    _PCLK_, (_BAUD_) & 0xF0U) + \
    _PCLK_, (_BAUD_) & 0x0FU) \
    (UART_DIVFRAQ_SAMPLING16(
    (UART_DIVFRAQ_SAMPLING16(
```

Definition at line 18 of file usart3.c.

6.17.2.2 #define UART_BRR_SAMPLING8(_PCLK_, _BAUD_)

Value:

```
(( (UART_DIVMANT_SAMPLING8( _PCLK_, _BAUD_ ) << 4U) + \
    _PCLK_, (_BAUD_) & 0xF8U) << 1U) + \
    _PCLK_, (_BAUD_) & 0x07U) \
    (UART_DIVFRAQ_SAMPLING8(
    (UART_DIVFRAQ_SAMPLING8(
```

Definition at line 27 of file usart3.c.

6.17.2.3 #define UART_DIV_SAMPLING16(_PCLK_, _BAUD_) (((_PCLK_)*25U)/(4U*(_BAUD_)))

Taken from the STM32 HAL.

Set of macros to calculate fractional baud rate which differs for this M4 part compared to the M0 part in the course

Definition at line 13 of file usart3.c.

6.17.2.4 #define UART_DIV_SAMPLING8(_PCLK_, _BAUD_) (((_PCLK_)*25U)/(2U*(_BAUD_)))

Definition at line 22 of file usart3.c.

6.17.2.5 #define UART_DIVFRAQ_SAMPLING16(_PCLK_, _BAUD_) (((UART_DIV_SAMPLING16(_PCLK_, _BAUD_) - (UART_DIVMANT_SAMPLING16(_PCLK_, _BAUD_) * 100U)) * 16U + 50U) / 100U)

Definition at line 15 of file usart3.c.

6.17.2.6 #define UART_DIVFRAQ_SAMPLING8(_PCLK_, _BAUD_) (((UART_DIV_SAMPLING8(_PCLK_, _BAUD_) - (UART_DIVMANT_SAMPLING8(_PCLK_, _BAUD_) * 100U)) * 8U + 50U) / 100U)

Definition at line 24 of file usart3.c.

6.17.2.7 `#define UART_DIVMANT_SAMPLING16(_PCLK_, _BAUD_) (UART_DIV_SAMPLING16((_PCLK_), (_BAUD_))/100U)`

Definition at line 14 of file usart3.c.

6.17.2.8 `#define UART_DIVMANT_SAMPLING8(_PCLK_, _BAUD_) (UART_DIV_SAMPLING8((_PCLK_), (_BAUD_))/100U)`

Definition at line 23 of file usart3.c.

6.17.3 Function Documentation

6.17.3.1 `static void Close (void) [static]`

Close USART.

Disables the device

Definition at line 88 of file usart3.c.

6.17.3.2 `static int_fast8_t GetByte (uint8_t* destination) [static]`

Checks to see if receive buffer has data.

Returns

1 = success 0 = no data -1 = port closed

Definition at line 131 of file usart3.c.

6.17.3.3 `static uint_fast8_t IsOpen (void) [static]`

Exposes IsOpenFlag.

Definition at line 39 of file usart3.c.

6.17.3.4 `static uint_fast8_t IsWriteBusy (void) [static]`

Checks to see if USART is writing.

Definition at line 46 of file usart3.c.

6.17.3.5 `static uint_fast8_t Open (uint32_t baudrate) [static]`

Enables and initializes the USART.

Definition at line 53 of file usart3.c.

6.17.3.6 `static int_fast8_t RxBufferHasData (void) [static]`

Checks to see if receive buffer has data.

Returns

1 = have data 0 = no data -1 = port closed

Definition at line 117 of file usart3.c.

6.17.3.7 `static uint_fast8_t SendArray (const uint8_t* source, uint32_t length) [static]`

Send byte array over USART.

Parameters

<i>source</i>	pointer to byte array
<i>length</i>	of byte array

Returns

TRUE = success else port may not be open or invalid pointer

Definition at line 174 of file usart3.c.

6.17.3.8 `static uint_fast8_t SendByte (uint8_t source) [static]`

Send one byte over the USART.

Parameters

<i>source</i>	byte to send
---------------	--------------

Returns

TRUE = success otherwise port is closed or other error

Definition at line 98 of file usart3.c.

6.17.3.9 `static uint_fast8_t SendString (const uint8_t * source) [static]`

Send string over the USART.

Parameters

<i>source</i>	pointer to array holding string
---------------	---------------------------------

Returns

TRUE = success else port may be closed or invalid pointer

Definition at line 152 of file usart3.c.

6.17.4 Variable Documentation

6.17.4.1 `uint_fast8_t IsOpenFlag = FALSE [static]`

Flag to indicate the state of the USART.

Definition at line 34 of file usart3.c.

6.17.4.2 SerialInterface SerialPort3

Initial value:

```
= {  
    IsOpen,  
    Open,  
    Close,  
    SendByte,  
    SendString,  
    SendArray,  
    RxBufferHasData,  
    GetByte  
}
```

Setup [SerialInterface](#) struct.

Definition at line 192 of file usart3.c.

Index

- __FPU_PRESENT
 - device.h, 14
- blue_led.c
 - BlueLed, 20
 - LED_PIN_BR, 19
 - LED_PIN_BS, 19
 - LED_PIN_ODR, 20
 - Led_Init, 20
 - Led_Off, 20
 - Led_On, 20
 - Led_Toggle, 20
- blue_led.h
 - RedLed, 15
- BlueLed
 - blue_led.c, 20
 - red_led.h, 16
- COMPILED_DATA_TIME
 - common.h, 13
- Close
 - SerialInterface, 10
 - usart3.c, 29
- common.h
 - COMPILED_DATA_TIME, 13
 - EN_DEBUG_INTERFACE, 13
 - FALSE, 13
 - FIRMWARE_VERSION, 13
 - HARDWARE_VERSION, 14
 - TRUE, 14
 - USART_OVER_SAMPLE_16, 14
- DelayMs
 - TickType, 11
- device.h
 - __FPU_PRESENT, 14
- EN_DEBUG_INTERFACE
 - common.h, 13
- FALSE
 - common.h, 13
- FIRMWARE_VERSION
 - common.h, 13
- GetByte
 - SerialInterface, 10
 - usart3.c, 29
- green_led.c
 - GreenLed, 22
 - LED_PIN_BR, 21
- LED_PIN_BS, 21
- LED_PIN_ODR, 22
- Led_Init, 22
- Led_Off, 22
- Led_On, 22
- Led_Toggle, 22
- green_led.h
 - GreenLed, 15
- GreenLed
 - green_led.c, 22
 - green_led.h, 15
- HARDWARE_VERSION
 - common.h, 14
- include/MCU/LED/blue_led.h, 14
- include/MCU/LED/green_led.h, 15
- include/MCU/LED/red_led.h, 15
- include/MCU/device.h, 14
- include/MCU/tick.h, 16
- include/MCU/usart3.h, 17
- include/common.h, 13
- include/led_interface.h, 14
- include/serial_interface.h, 17
- Init
 - LedInterface, 9
- IsOpen
 - SerialInterface, 10
 - usart3.c, 29
- IsOpenFlag
 - usart3.c, 31
- IsWriteBusy
 - usart3.c, 29
- LED_PIN_BR
 - blue_led.c, 19
 - green_led.c, 21
 - red_led.c, 23
- LED_PIN_BS
 - blue_led.c, 19
 - green_led.c, 21
 - red_led.c, 23
- LED_PIN_ODR
 - blue_led.c, 20
 - green_led.c, 22
 - red_led.c, 24
- Led_Init
 - blue_led.c, 20
 - green_led.c, 22
 - red_led.c, 24

- Led_Off
 - blue_led.c, 20
 - green_led.c, 22
 - red_led.c, 24
- Led_On
 - blue_led.c, 20
 - green_led.c, 22
 - red_led.c, 24
- Led_Toggle
 - blue_led.c, 20
 - green_led.c, 22
 - red_led.c, 24
- LedInterface, 9
 - Init, 9
 - Off, 9
 - On, 9
 - Toggle, 9
- main
 - main.c, 18
- main.c
 - main, 18
 - PrintHeader, 18
- Mainpage.md, 18
- Off
 - LedInterface, 9
- On
 - LedInterface, 9
- Open
 - SerialInterface, 10
 - usart3.c, 29
- PrintHeader
 - main.c, 18
- README.md, 18
- red_led.c
 - LED_PIN_BR, 23
 - LED_PIN_BS, 23
 - LED_PIN_ODR, 24
 - Led_Init, 24
 - Led_Off, 24
 - Led_On, 24
 - Led_Toggle, 24
 - RedLed, 24
- red_led.h
 - BlueLed, 16
- RedLed
 - blue_led.h, 15
 - red_led.c, 24
- RxBufferHasData
 - SerialInterface, 10
 - usart3.c, 29
- SendArray
 - SerialInterface, 11
 - usart3.c, 30
- SendByte
 - SerialInterface, 11
 - usart3.c, 31
- SendString
 - SerialInterface, 11
 - usart3.c, 31
- SerialInterface, 10
 - Close, 10
 - GetByte, 10
 - IsOpen, 10
 - Open, 10
 - RxBufferHasData, 10
 - SendArray, 11
 - SendByte, 11
 - SendString, 11
- SerialPort3
 - usart3.c, 31
 - usart3.h, 17
- src/MCU/LED/blue_led.c, 19
- src/MCU/LED/green_led.c, 21
- src/MCU/LED/red_led.c, 23
- src/MCU/tick.c, 25
- src/MCU/usart3.c, 26
- src/main.c, 18
- StartMs
 - TickType, 11
- SysTick_Handler
 - tick.c, 26
- TIMER_FREQUENCY_HZ
 - tick.c, 25
- TRUE
 - common.h, 14
- tick.c
 - SysTick_Handler, 26
 - TIMER_FREQUENCY_HZ, 25
 - Tick_DelayMs, 26
 - Tick_DelayMs_NonBlocking, 26
 - Tick_GetMs, 26
 - Tick_Init, 26
 - TickCounter, 26
- tick.h
 - Tick_DelayMs, 16
 - Tick_DelayMs_NonBlocking, 16
 - Tick_GetMs, 16
 - Tick_Init, 17
- Tick_DelayMs
 - tick.c, 26
 - tick.h, 16
- Tick_DelayMs_NonBlocking
 - tick.c, 26
 - tick.h, 16
- Tick_GetMs
 - tick.c, 26
 - tick.h, 16
- Tick_Init
 - tick.c, 26
 - tick.h, 17
- TickCounter
 - tick.c, 26

- TickType, [11](#)
 - DelayMs, [11](#)
 - StartMs, [11](#)
- Toggle
 - LedInterface, [9](#)
- UART_BRR_SAMPLING16
 - usart3.c, [28](#)
- UART_BRR_SAMPLING8
 - usart3.c, [28](#)
- UART_DIV_SAMPLING16
 - usart3.c, [28](#)
- UART_DIV_SAMPLING8
 - usart3.c, [28](#)
- UART_DIVFRAQ_SAMPLING16
 - usart3.c, [28](#)
- UART_DIVFRAQ_SAMPLING8
 - usart3.c, [28](#)
- UART_DIVMANT_SAMPLING16
 - usart3.c, [28](#)
- UART_DIVMANT_SAMPLING8
 - usart3.c, [29](#)
- USART_OVER_SAMPLE_16
 - common.h, [14](#)
- usart3.c
 - Close, [29](#)
 - GetByte, [29](#)
 - IsOpen, [29](#)
 - IsOpenFlag, [31](#)
 - IsWriteBusy, [29](#)
 - Open, [29](#)
 - RxBufferHasData, [29](#)
 - SendArray, [30](#)
 - SendByte, [31](#)
 - SendString, [31](#)
 - SerialPort3, [31](#)
 - UART_BRR_SAMPLING16, [28](#)
 - UART_BRR_SAMPLING8, [28](#)
 - UART_DIV_SAMPLING16, [28](#)
 - UART_DIV_SAMPLING8, [28](#)
 - UART_DIVFRAQ_SAMPLING16, [28](#)
 - UART_DIVFRAQ_SAMPLING8, [28](#)
 - UART_DIVMANT_SAMPLING16, [28](#)
 - UART_DIVMANT_SAMPLING8, [29](#)
- usart3.h
 - SerialPort3, [17](#)