Flash Driver for STM32F103

Generated by Doxygen 1.8.18

1 Data Structure Index	1
1.1 Data Structures	1
2 File Index	3
2.1 File List	3
3 Data Structure Documentation	5
3.1 FLITF_t Struct Reference	5
4 File Documentation	7
4.1 FLASH/FLITF.c File Reference	7
4.1.1 Detailed Description	8
4.1.2 Macro Definition Documentation	8
4.1.2.1 PROGRAM_ENABLE	8
4.1.3 Function Documentation	9
4.1.3.1 Flash_ErasePage()	9
4.1.3.2 Flash_FullWord()	9
4.1.3.3 Flash_HalfWord()	9
4.1.3.4 Flash_Lock()	10
4.1.3.5 Flash_MassErase()	10
4.1.3.6 Flash_ProgramWrite()	10
4.1.3.7 Flash_Unlock()	11
Index	13

Data Structure Index

1	.1	Data	Stru	ctu	rΔe
	- 1	vala	OHU	G. U.	

Here are the data structures with brief descriptions:								
FLITF t	 			 				ļ

2 Data Structure Index

File Index

2.1 File List

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

FLASH/FLITF.c		
This file is the Implementation for Flash Driver Interface for STM32F103	 	 -
FLASH/ FLITF.h	 	 ?'

File Index

Data Structure Documentation

3.1 FLITF_t Struct Reference

Data Fields

- volatile uint_32t FLASH_ACR
- volatile uint_32t FLASH_KEYR
- volatile uint_32t FLASH_OPTKEYR
- volatile uint_32t FLASH_SR
- volatile uint_32t FLASH_CR
- volatile uint_32t FLASH_AR
- volatile uint_32t FLASH_RESERVED
- volatile uint_32t FLASH_OPR
- volatile uint_32t FLASH_WRPR

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

• FLASH/FLITF.c

File Documentation

4.1 FLASH/FLITF.c File Reference

This file is the Implementation for Flash Driver Interface for STM32F103.

```
#include "STD_TYPES.h"
#include "FLITF.h"
```

Data Structures

struct FLITF t

Macros

```
#define FPEC ((FLITF_t *)(0x40022000))
```

Casting Base Address of Flash Driver as Pointer to struct FLITF_t.

• #define HALF WORD LEN 2

HNo of bytes of Half Word.

#define KEY1 (0x45670123)

KEY1 to unlock Flash to write on it.

#define KEY2 (0xcdef89ab)

KEY2 to unlock Flash to write on it.

• #define LOCK 0x00000080

used to lock Flash

• #define START 0x00000040

used to trigger an ERASE operation when set

#define MASS_ERASE 0x00000004

used to earse all the Flash

• #define PAGE_ERASE 0x00000002

used to earse page

• #define PROGRAM ENABLE 0x00000001

used to enable falsh programming

• #define MER RESET 0x00001FFB

used to reset Mass Erase bit.

• #define EOP 0x00000020

used to indicate that a Flash operation is completed

#define BUSY 0x00000001

used to indicate that a Flash operation is in progress.

8 File Documentation

Functions

```
    void Flash_Lock (void)
```

Function to Lock The Flash.

void Flash_Unlock (void)

Function to Unlock The Flash.

void Flash_MassErase (void)

Function to erase all The Flash.

void Flash_ErasePage (uint_32t PageAddress)

Function to erase page from flash.

• void Flash_ProgramWrite (void *StartAddress, void *DataAddress, uint_32t DataLength)

Function to write Full Data.

void Flash_HalfWord (uint_16t *StartAddress, uint_16t Data)

Function to write Half Word Data.

void Flash_FullWord (uint_32t *StartAddress, uint_32t Data)

Function to write Full Word Data.

4.1.1 Detailed Description

This file is the Implementation for Flash Driver Interface for STM32F103.

This file is a user interface for Flash Driver Interface for STM32F103.

Author

```
Amr( Ibrahimamr222@gmail.com)
```

Version

0.1

Date

2020-06-05

Copyright

Copyright (c) 2020

4.1.2 Macro Definition Documentation

4.1.2.1 PROGRAM_ENABLE

```
#define PROGRAM_ENABLE 0x0000001
```

used to enable falsh programming

4.1.3 Function Documentation

4.1.3.1 Flash_ErasePage()

Function to erase page from flash.

Parameters



Returns

NA

4.1.3.2 Flash_FullWord()

Function to write Full Word Data.

Parameters

StartAddress	pointer to uint_32t , Start address to write
Data	variabe of uint 32t, Data to be written

Returns

NA

4.1.3.3 Flash_HalfWord()

Function to write Half Word Data.

10 File Documentation

Parameters

StartAddress	pointer to uint_16t , Start address to write
Data	variabe of uint_16t , Data to be written

Returns

NA

4.1.3.4 Flash_Lock()

```
void Flash_Lock (
     void )
```

Function to Lock The Flash.

Parameters



Returns

NA

4.1.3.5 Flash_MassErase()

```
void Flash_MassErase (
     void )
```

Function to erase all The Flash.

Parameters

NA

Returns

NA

4.1.3.6 Flash_ProgramWrite()

```
void * DataAddress,
uint_32t DataLength )
```

Function to write Full Data.

Parameters

StartAddress	pointer to void , Start address to write
DataAddress	pointer to void , Address of buffer of data
DataLength	variabe of uint_32t , Length of Data to be written

Returns

NA

4.1.3.7 Flash_Unlock()

```
void Flash_Unlock (
     void )
```

Function to Unlock The Flash.

Parameters

NA

Returns

NA

12 File Documentation

Index

```
FLASH/FLITF.c, 7
Flash_ErasePage
    FLITF.c, 9
Flash_FullWord
    FLITF.c, 9
Flash_HalfWord
    FLITF.c, 9
Flash_Lock
    FLITF.c, 10
Flash MassErase
    FLITF.c, 10
Flash_ProgramWrite
    FLITF.c, 10
Flash_Unlock
    FLITF.c, 11
FLITF.c
    Flash_ErasePage, 9
    Flash_FullWord, 9
    Flash_HalfWord, 9
    Flash_Lock, 10
    Flash_MassErase, 10
    Flash_ProgramWrite, 10
    Flash_Unlock, 11
    PROGRAM_ENABLE, 8
FLITF_t, 5
PROGRAM_ENABLE
    FLITF.c, 8
```