

MAXREFDES82# Code Documentation

V01.00

Generated by Doxygen 1.8.2

Mon Mar 21 2016 01:04:41

Contents

1	Main Page	1
1.1	Introduction	1
2	Module Index	3
2.1	Modules	3
3	File Index	5
3.1	File List	5
4	Module Documentation	7
4.1	STEMWIN_LIBRARY	7
4.1.1	Detailed Description	7
4.2	STEMWIN_LIBRARY_Private_TypesDefinitions	8
4.3	STEMWIN_LIBRARY_Private_Defines	9
4.3.1	Detailed Description	9
4.3.2	Macro Definition Documentation	9
4.3.2.1	BK_COLOR	9
4.3.2.2	COLOR_CONVERSION_0	9
4.3.2.3	DISPLAY_DRIVER_0	9
4.3.2.4	GUI_NUM_LAYERS	9
4.3.2.5	LCD_LAYER0_FRAME_BUFFER	9
4.3.2.6	LCD_LAYER1_FRAME_BUFFER	9
4.3.2.7	LCD_MIRROR_Y	9
4.3.2.8	LCD_SWAP_XY	10
4.3.2.9	NUM_BUFFERS	10
4.3.2.10	NUM_VSCREENS	10
4.3.2.11	XSIZE_PHYS	10
4.3.2.12	YSIZE_PHYS	10
4.4	STEMWIN_LIBRARY_Private_Macros	11
4.4.1	Detailed Description	11

4.4.2	Macro Definition Documentation	11
4.4.2.1	DEFINEDMA2D_COLORCONVERSION	11
4.5	STEMWIN_LIBRARY_Private_Variables	12
4.5.1	Detailed Description	12
4.5.2	Variable Documentation	12
4.5.2.1	hltdc	12
4.6	STEMWIN_LIBRARY_Private_FunctionPrototypes	13
4.6.1	Detailed Description	13
4.7	STEMWIN_LIBRARY_Private_Functions	14
4.7.1	Detailed Description	14
4.7.2	Function Documentation	14
4.7.2.1	HAL_DMA2D_MspDeInit	14
4.7.2.2	HAL_DMA2D_MspInit	14
4.7.2.3	HAL_LTDC_LineEvenCallback	15
4.7.2.4	HAL_LTDC_MspDeInit	15
4.7.2.5	HAL_LTDC_MspInit	15
4.7.2.6	LCD_X_Config	16
4.7.2.7	LCD_X_DisplayDriver	16
4.8	USBD_USR_MAIN	17
4.8.1	Detailed Description	17
4.8.2	Function Documentation	17
4.8.2.1	HAL_UART_MspDeInit	17
4.8.2.2	HAL_UART_MspInit	17
4.9	USBD_USER	19
4.9.1	Detailed Description	19
4.10	CMSIS	20
4.10.1	Detailed Description	20
4.11	Stm32f4xx_system	21
4.11.1	Detailed Description	21
4.12	STM32F4xx_System_Private_Includes	22
4.12.1	Detailed Description	22
4.12.2	Macro Definition Documentation	22
4.12.2.1	HSE_VALUE	22
4.12.2.2	HSI_VALUE	22
4.13	STM32F4xx_System_Private_TypesDefinitions	23
4.14	STM32F4xx_System_Private_Defines	24
4.14.1	Detailed Description	24

4.14.2 Macro Definition Documentation	24
4.14.2.1 VECT_TAB_OFFSET	24
4.15 STM32F4xx_System_Private_Macros	25
4.16 STM32F4xx_System_Private_Variables	26
4.16.1 Detailed Description	26
4.16.2 Variable Documentation	26
4.16.2.1 AHBPrescTable	26
4.16.2.2 SystemCoreClock	26
4.17 STM32F4xx_System_Private_FunctionPrototypes	27
4.18 STM32F4xx_System_Private_Functions	28
4.18.1 Detailed Description	28
4.18.2 Function Documentation	28
4.18.2.1 SystemCoreClockUpdate	28
4.18.2.2 SystemInit	29
5 File Documentation	31
5.1 Src/GUIConf.c File Reference	31
5.1.1 Macro Definition Documentation	31
5.1.1.1 GUI_NUMBYTES	31
5.1.2 Function Documentation	32
5.1.2.1 GUI_X_Config	32
5.2 Src/LCDConf_stm32f429i_disco_MB1075.c File Reference	32
5.3 Src/maintask.c File Reference	33
5.3.1 Detailed Description	34
5.3.2 Macro Definition Documentation	35
5.3.2.1 FACOTRY_CAL_NUM	35
5.3.2.2 MAJORVERSION	35
5.3.2.3 MINORVERSION	35
5.3.2.4 POWERUP_CAL_NUM	35
5.3.3 Function Documentation	35
5.3.3.1 Calibration	35
5.3.3.2 ConfigureMAX11254	35
5.3.3.3 ConfigureSPlandGPIO	35
5.3.3.4 MainTask	36
5.3.3.5 ReadMAX11254Results	36
5.3.4 Variable Documentation	36
5.3.4.1 cell0Grams	36

5.3.4.2	cell0Load	36
5.3.4.3	cell0Load0	36
5.3.4.4	cell0Load0Int	36
5.3.4.5	cell0Load0Sign	36
5.3.4.6	cell0Load700	36
5.3.4.7	cell0Load700Int	36
5.3.4.8	cell0Load700Sign	36
5.3.4.9	cell0Powerup	36
5.3.4.10	cell1Grams	37
5.3.4.11	cell1Load	37
5.3.4.12	cell1Load0	37
5.3.4.13	cell1Load0Int	37
5.3.4.14	cell1Load0Sign	37
5.3.4.15	cell1Load700	37
5.3.4.16	cell1Load700Int	37
5.3.4.17	cell1Load700Sign	37
5.3.4.18	cell1Powerup	37
5.3.4.19	cell2Grams	37
5.3.4.20	cell2Load	37
5.3.4.21	cell2Load0	38
5.3.4.22	cell2Load0Int	38
5.3.4.23	cell2Load0Sign	38
5.3.4.24	cell2Load700	38
5.3.4.25	cell2Load700Int	38
5.3.4.26	cell2Load700Sign	38
5.3.4.27	cell2Powerup	38
5.3.4.28	cell3Grams	38
5.3.4.29	cell3Load	38
5.3.4.30	cell3Load0	38
5.3.4.31	cell3Load0Int	38
5.3.4.32	cell3Load0Sign	39
5.3.4.33	cell3Load700	39
5.3.4.34	cell3Load700Int	39
5.3.4.35	cell3Load700Sign	39
5.3.4.36	cell3Powerup	39
5.3.4.37	data0	39
5.3.4.38	data0BO	39

5.3.4.39	data1	39
5.3.4.40	data1BO	39
5.3.4.41	data2	39
5.3.4.42	data2BO	39
5.3.4.43	data3	40
5.3.4.44	data3BO	40
5.3.4.45	r	40
5.3.4.46	Rect	40
5.3.4.47	request	40
5.3.4.48	SectorError	40
5.3.4.49	spiWritePacket	40
5.3.4.50	USBD_Device	40
5.3.4.51	usbTxBuf	40
5.3.4.52	x	40
5.3.4.53	y	40
5.4	Src/MAXREFDES82.c File Reference	41
5.4.1	Detailed Description	41
5.4.2	Function Documentation	42
5.4.2.1	main	42
5.4.2.2	MainTask	42
5.4.3	Variable Documentation	42
5.4.3.1	GUI_Initialized	42
5.4.3.2	USBD_Device	42
5.5	Src/stm32f4xx_hal_msp.c File Reference	42
5.6	Src/stm32f4xx_it.c File Reference	43
5.6.1	Function Documentation	44
5.6.1.1	BusFault_Handler	44
5.6.1.2	DebugMon_Handler	44
5.6.1.3	HardFault_Handler	44
5.6.1.4	LTDC_IRQHandler	44
5.6.1.5	MemManage_Handler	45
5.6.1.6	NMI_Handler	45
5.6.1.7	OTG_HS_IRQHandler	45
5.6.1.8	PendSV_Handler	45
5.6.1.9	SVC_Handler	46
5.6.1.10	SysTick_Handler	46
5.6.1.11	TIMx_IRQHandler	46

5.6.1.12	UsageFault_Handler	47
5.6.1.13	USARTx_DMA_TX_IRQHandler	47
5.6.1.14	USARTx_IRQHandler	47
5.6.2	Variable Documentation	47
5.6.2.1	hpcd	47
5.6.2.2	OS_TimeMS	48
5.6.2.3	TimHandle	48
5.6.2.4	UartHandle	48
5.7	Src/system_stm32f4xx.c File Reference	48
5.7.1	Detailed Description	48
5.8	Src/usbd_cdc_interface.c File Reference	49
5.8.1	Macro Definition Documentation	50
5.8.1.1	APP_RX_DATA_SIZE	50
5.8.1.2	APP_TX_DATA_SIZE	50
5.8.2	Function Documentation	50
5.8.2.1	HAL_TIM_PeriodElapsedCallback	50
5.8.2.2	HAL_UART_ErrorCallback	51
5.8.2.3	HAL_UART_RxCpltCallback	51
5.8.2.4	HAL_UART_TxCpltCallback	51
5.8.3	Variable Documentation	51
5.8.3.1	LineCoding	51
5.8.3.2	request	52
5.8.3.3	TimHandle	52
5.8.3.4	UartHandle	52
5.8.3.5	USBD_CDC_fops	52
5.8.3.6	USBD_Device	52
5.8.3.7	UserRxBuffer	52
5.8.3.8	UserTxBuffer	52
5.8.3.9	UserTxBufPtrIn	52
5.8.3.10	UserTxBufPtrOut	52
5.9	Src/usbd_conf.c File Reference	53
5.9.1	Function Documentation	54
5.9.1.1	HAL_PCD_ConnectCallback	54
5.9.1.2	HAL_PCD_DataInStageCallback	54
5.9.1.3	HAL_PCD_DataOutStageCallback	55
5.9.1.4	HAL_PCD_DisconnectCallback	55
5.9.1.5	HAL_PCD_ISOINIncompleteCallback	55

5.9.1.6	HAL_PCD_ISOOUTIncompleteCallback	56
5.9.1.7	HAL_PCD_MspDeInit	56
5.9.1.8	HAL_PCD_MspInit	56
5.9.1.9	HAL_PCD_ResetCallback	56
5.9.1.10	HAL_PCD_ResumeCallback	57
5.9.1.11	HAL_PCD_SetupStageCallback	57
5.9.1.12	HAL_PCD_SOFCallback	57
5.9.1.13	HAL_PCD_SuspendCallback	57
5.9.1.14	USBD_LL_ClearStallEP	58
5.9.1.15	USBD_LL_CloseEP	58
5.9.1.16	USBD_LL_DeInit	58
5.9.1.17	USBD_LL_Delay	59
5.9.1.18	USBD_LL_FlushEP	59
5.9.1.19	USBD_LL_GetRxDataSize	59
5.9.1.20	USBD_LL_Init	59
5.9.1.21	USBD_LL_IsStallEP	60
5.9.1.22	USBD_LL_OpenEP	60
5.9.1.23	USBD_LL_PrepareReceive	60
5.9.1.24	USBD_LL_SetUSBAddress	61
5.9.1.25	USBD_LL_StallEP	61
5.9.1.26	USBD_LL_Start	61
5.9.1.27	USBD_LL_Stop	62
5.9.1.28	USBD_LL_Transmit	62
5.9.2	Variable Documentation	62
5.9.2.1	hpcd	62
5.10	Src/usbd_desc.c File Reference	62
5.10.1	Macro Definition Documentation	63
5.10.1.1	USBD_CONFIGURATION_FS_STRING	63
5.10.1.2	USBD_CONFIGURATION_HS_STRING	63
5.10.1.3	USBD_INTERFACE_FS_STRING	64
5.10.1.4	USBD_INTERFACE_HS_STRING	64
5.10.1.5	USBD_LANGID_STRING	64
5.10.1.6	USBD_MANUFACTURER_STRING	64
5.10.1.7	USBD_PID	64
5.10.1.8	USBD_PRODUCT_FS_STRING	64
5.10.1.9	USBD_PRODUCT_HS_STRING	64
5.10.1.10	USBD_VID	64

5.10.2 Function Documentation	64
5.10.2.1 USBD_VCP_ConfigStrDescriptor	64
5.10.2.2 USBD_VCP_DeviceDescriptor	65
5.10.2.3 USBD_VCP_InterfaceStrDescriptor	65
5.10.2.4 USBD_VCP_LangIDStrDescriptor	65
5.10.2.5 USBD_VCP_ManufacturerStrDescriptor	65
5.10.2.6 USBD_VCP_ProductStrDescriptor	66
5.10.2.7 USBD_VCP_SerialStrDescriptor	66
5.10.3 Variable Documentation	66
5.10.3.1 __ALIGN_END	66
5.10.3.2 USBD_StringSerial	67
5.10.3.3 VCP_Desc	67

Chapter 1

Main Page

1.1 Introduction

This is the code documentation for the MAXREFDES82# reference design.

The Files page contains the File List page and the Globals page.

The Globals page contains the Functions, Variables, and Macros sub-pages.

Chapter 2

Module Index

2.1 Modules

Here is a list of all modules:

STEMWIN_LIBRARY	7
STEMWIN_LIBRARY_Private_TypesDefinitions	8
STEMWIN_LIBRARY_Private_Defines	9
STEMWIN_LIBRARY_Private_Macros	11
STEMWIN_LIBRARY_Private_Variables	12
STEMWIN_LIBRARY_Private_FunctionPrototypes	13
STEMWIN_LIBRARY_Private_Functions	14
USBD_USER	19
USBD_USR_MAIN	17
CMSIS	20
Stm32f4xx_system	21
STM32F4xx_System_Private_Includes	22
STM32F4xx_System_Private_TypesDefinitions	23
STM32F4xx_System_Private_Defines	24
STM32F4xx_System_Private_Macros	25
STM32F4xx_System_Private_Variables	26
STM32F4xx_System_Private_FunctionPrototypes	27
STM32F4xx_System_Private_Functions	28

Chapter 3

File Index

3.1 File List

Here is a list of all files with brief descriptions:

Src/ GUIConf.c	31
Src/ LCDConf_stm32f429i_disco_MB1075.c	32
Src/ maintask.c	33
Src/ MAXREFDES82.c	41
Src/ stm32f4xx_hal_msp.c	42
Src/ stm32f4xx_it.c	43
Src/ system_stm32f4xx.c	
CMSIS Cortex-M4 Device Peripheral Access Layer System Source File	48
Src/ usbd_cdc_interface.c	49
Src/ usbd_conf.c	53
Src/ usbd_desc.c	62

Chapter 4

Module Documentation

4.1 STEMWIN_LIBRARY

This file contains the USB Configuration.

Modules

- [STEMWIN_LIBRARY_Private_TypesDefinitions](#)
- [STEMWIN_LIBRARY_Private_Defines](#)
- [STEMWIN_LIBRARY_Private_Macros](#)
- [STEMWIN_LIBRARY_Private_Variables](#)
- [STEMWIN_LIBRARY_Private_FunctionPrototypes](#)
- [STEMWIN_LIBRARY_Private_Functions](#)

4.1.1 Detailed Description

This file contains the USB Configuration.

4.2 STEMWIN_LIBRARY_Private_TypesDefinitions

4.3 STEMWIN_LIBRARY_Private_Defines

Macros

- `#define LCD_SWAP_XY 1`
- `#define LCD_MIRROR_Y 1`
- `#define XSIZE_PHYS 320`
- `#define YSIZE_PHYS 240`
- `#define NUM_BUFFERS 1`
- `#define NUM_VSCREENS 1`
- `#define BK_COLOR GUI_DARKBLUE`
- `#define GUI_NUM_LAYERS 1`
- `#define COLOR_CONVERSION_0 GUICC_M8888I`
- `#define DISPLAY_DRIVER_0 GUIDRV_LIN_32`
- `#define LCD_LAYER0_FRAME_BUFFER ((uint32_t)0xD0400000)`
- `#define LCD_LAYER1_FRAME_BUFFER ((uint32_t)0xD0600000)`

4.3.1 Detailed Description

4.3.2 Macro Definition Documentation

4.3.2.1 `#define BK_COLOR GUI_DARKBLUE`

Definition at line 67 of file LCDConf_stm32f429i_disco_MB1075.c.

4.3.2.2 `#define COLOR_CONVERSION_0 GUICC_M8888I`

Definition at line 72 of file LCDConf_stm32f429i_disco_MB1075.c.

4.3.2.3 `#define DISPLAY_DRIVER_0 GUIDRV_LIN_32`

Definition at line 73 of file LCDConf_stm32f429i_disco_MB1075.c.

4.3.2.4 `#define GUI_NUM_LAYERS 1`

Definition at line 70 of file LCDConf_stm32f429i_disco_MB1075.c.

4.3.2.5 `#define LCD_LAYER0_FRAME_BUFFER ((uint32_t)0xD0400000)`

Definition at line 98 of file LCDConf_stm32f429i_disco_MB1075.c.

4.3.2.6 `#define LCD_LAYER1_FRAME_BUFFER ((uint32_t)0xD0600000)`

Definition at line 99 of file LCDConf_stm32f429i_disco_MB1075.c.

4.3.2.7 `#define LCD_MIRROR_Y 1`

Definition at line 56 of file LCDConf_stm32f429i_disco_MB1075.c.

4.3.2.8 #define LCD_SWAP_XY 1

Definition at line 55 of file LCDConf_stm32f429i_disco_MB1075.c.

4.3.2.9 #define NUM_BUFFERS 1

Definition at line 64 of file LCDConf_stm32f429i_disco_MB1075.c.

4.3.2.10 #define NUM_VSCREENS 1

Definition at line 65 of file LCDConf_stm32f429i_disco_MB1075.c.

4.3.2.11 #define XSIZE_PHYS 320

Definition at line 61 of file LCDConf_stm32f429i_disco_MB1075.c.

4.3.2.12 #define YSIZE_PHYS 240

Definition at line 62 of file LCDConf_stm32f429i_disco_MB1075.c.

4.4 STEMWIN_LIBRARY_Private_Macros

Macros

- #define `DEFINEDDMA2D_COLORCONVERSION`(*PFIX*, *PIXELFORMAT*)

4.4.1 Detailed Description

4.4.2 Macro Definition Documentation

4.4.2.1 #define `DEFINEDDMA2D_COLORCONVERSION`(*PFIX*, *PIXELFORMAT*)

Value:

```
static void Color2IndexBulk_##PFIX##DMA2D(LCD_COLOR * pColor, void * pIndex,
    U32 NumItems, U8 SizeOfIndex) { \
    DMA2D_Color2IndexBulk(pColor, pIndex, NumItems, SizeOfIndex, PIXELFORMAT);
}

static void Index2ColorBulk_##PFIX##DMA2D(void * pIndex, LCD_COLOR * pColor,
    U32 NumItems, U8 SizeOfIndex) { \
    DMA2D_Index2ColorBulk(pIndex, pColor, NumItems, SizeOfIndex, PIXELFORMAT); \
}
```

Definition at line 110 of file LCDConf_stm32f429i_disco_MB1075.c.

4.5 STEMWIN_LIBRARY_Private_Variables

Variables

- LTDC_HandleTypeDef [hltdc](#)

4.5.1 Detailed Description

4.5.2 Variable Documentation

4.5.2.1 LTDC_HandleTypeDef hltdc

Definition at line 126 of file LCDConf_stm32f429i_disco_MB1075.c.

4.6 STEMWIN_LIBRARY_Private_FunctionPrototypes

4.6.1 Detailed Description

4.7 STEMWIN_LIBRARY_Private_Functions

Functions

- void [HAL_DMA2D_MspInit](#) (DMA2D_HandleTypeDef *hdma2d)
DMA2D MSP Initialization This function configures the hardware resources used in This application:
- void [HAL_DMA2D_MspDeInit](#) (DMA2D_HandleTypeDef *hdma2d)
DMA2D MSP De-Initialization This function frees the hardware resources used in This application:
- void [HAL_LTDC_MspInit](#) (LTDC_HandleTypeDef *hltdc)
LTDC MSP Initialization This function configures the hardware resources used in This application:
- void [HAL_LTDC_MspDeInit](#) (LTDC_HandleTypeDef *hltdc)
LTDC MSP De-Initialization This function frees the hardware resources used in This application:
- void [HAL_LTDC_LineEvenCallback](#) (LTDC_HandleTypeDef *hltdc)
Line Event callback.
- void [LCD_X_Config](#) (void)
Called during the initialization process in order to set up the display driver configuration.
- int [LCD_X_DisplayDriver](#) (unsigned LayerIndex, unsigned Cmd, void *pData)
This function is called by the display driver for several purposes. To support the according task the routine needs to be adapted to the display controller. Please note that the commands marked with 'optional' are not cogently required and should only be adapted if the display controller supports these features.

4.7.1 Detailed Description

4.7.2 Function Documentation

4.7.2.1 void HAL_DMA2D_MspDeInit (DMA2D_HandleTypeDef * *hdma2d*)

DMA2D MSP De-Initialization This function frees the hardware resources used in This application:

- Disable the Peripheral's clock

Parameters

<i>hdma2d,:</i>	DMA2D handle pointer
-----------------	----------------------

Return values

<i>None</i>

Definition at line 210 of file LCDConf_stm32f429i_disco_MB1075.c.

4.7.2.2 void HAL_DMA2D_MspInit (DMA2D_HandleTypeDef * *hdma2d*)

DMA2D MSP Initialization This function configures the hardware resources used in This application:

- Peripheral's clock enable
- Peripheral's GPIO Configuration

Parameters

<i>hdma2d,:</i>	DMA2D handle pointer
-----------------	----------------------

Return values

<i>None</i>

Definition at line 197 of file LCDConf_stm32f429i_disco_MB1075.c.

4.7.2.3 void HAL_LTDC_LineEvenCallback (LTDC_HandleTypeDef * *hltdc*)

Line Event callback.

Parameters

<i>hltdc,:</i>	pointer to a LTDC_HandleTypeDef structure that contains the configuration information for the specified LTDC.
----------------	---

Return values

<i>None</i>

Definition at line 332 of file LCDConf_stm32f429i_disco_MB1075.c.

4.7.2.4 void HAL_LTDC_MspDeInit (LTDC_HandleTypeDef * *hltdc*)

LTDC MSP De-Initialization This function frees the hardware resources used in This application:

- Disable the Peripheral's clock

Parameters

<i>hltdc,:</i>	LTDC handle pointer
----------------	---------------------

Return values

<i>None</i>

Definition at line 316 of file LCDConf_stm32f429i_disco_MB1075.c.

4.7.2.5 void HAL_LTDC_MspInit (LTDC_HandleTypeDef * *hltdc*)

LTDC MSP Initialization This function configures the hardware resources used in This application:

- Peripheral's clock enable
- Peripheral's GPIO Configuration

Parameters

<i>hltdc,:</i>	LTDC handle pointer
----------------	---------------------

Return values

<i>None</i>	
-------------	--

Definition at line 227 of file LCDConf_stm32f429i_disco_MB1075.c.

4.7.2.6 void LCD_X_Config (void)

Called during the initialization process in order to set up the display driver configuration.

Parameters

<i>None</i>	
-------------	--

Return values

<i>None</i>	
-------------	--

Definition at line 367 of file LCDConf_stm32f429i_disco_MB1075.c.

4.7.2.7 int LCD_X_DisplayDriver (unsigned *LayerIndex*, unsigned *Cmd*, void * *pData*)

This function is called by the display driver for several purposes. To support the according task the routine needs to be adapted to the display controller. Please note that the commands marked with 'optional' are not cogently required and should only be adapted if the display controller supports these features.

Parameters

<i>LayerIndex</i> ,:	Index of layer to be configured
<i>Cmd</i>	:Please refer to the details in the switch statement below
<i>pData</i>	:Pointer to a LCD_X_DATA structure

Return values

<i>Status</i>	(-1 : Error, 0 : Ok)
---------------	----------------------

Definition at line 474 of file LCDConf_stm32f429i_disco_MB1075.c.

4.8 USB_D_USR_MAIN

This file is the CDC application main file.

Functions

- void [HAL_UART_MspInit](#) (UART_HandleTypeDef *huart)
UART MSP Initialization This function configures the hardware resources used in this application:
- void [HAL_UART_MspDeInit](#) (UART_HandleTypeDef *huart)
UART MSP De-Initialization This function frees the hardware resources used in this application:

4.8.1 Detailed Description

This file is the CDC application main file.

4.8.2 Function Documentation

4.8.2.1 void HAL_UART_MspDeInit (UART_HandleTypeDef * *huart*)

UART MSP De-Initialization This function frees the hardware resources used in this application:

- Disable the Peripheral's clock
- Revert GPIO, and NVIC configuration to their default state

Parameters

<i>huart</i> ,:	UART handle pointer
-----------------	---------------------

Return values

<i>None</i>

Definition at line 148 of file stm32f4xx_hal_msp.c.

4.8.2.2 void HAL_UART_MspInit (UART_HandleTypeDef * *huart*)

UART MSP Initialization This function configures the hardware resources used in this application:

- Peripheral's clock enable
- Peripheral's GPIO Configuration
- NVIC configuration for UART interrupt request enable

Parameters

<i>huart</i> ,:	UART handle pointer
-----------------	---------------------

Return values

<i>None</i>	
-------------	--

Definition at line 68 of file stm32f4xx_hal_msp.c.

4.9 USBD_USER

Modules

- [USB_USR_MAIN](#)

This file is the CDC application main file.

4.9.1 Detailed Description

4.10 CMSIS

Modules

- [Stm32f4xx_system](#)

4.10.1 Detailed Description

4.11 **Stm32f4xx_system**

Modules

- [STM32F4xx_System_Private_Includes](#)
- [STM32F4xx_System_Private_TypesDefinitions](#)
- [STM32F4xx_System_Private_Defines](#)
- [STM32F4xx_System_Private_Macros](#)
- [STM32F4xx_System_Private_Variables](#)
- [STM32F4xx_System_Private_FunctionPrototypes](#)
- [STM32F4xx_System_Private_Functions](#)

4.11.1 Detailed Description

4.12 STM32F4xx_System_Private_Includes

Macros

- #define [HSE_VALUE](#) ((uint32_t)8000000)
- #define [HSI_VALUE](#) ((uint32_t)16000000)

4.12.1 Detailed Description

4.12.2 Macro Definition Documentation

4.12.2.1 #define HSE_VALUE ((uint32_t)8000000)

Default value of the External oscillator in Hz

Definition at line 69 of file system_stm32f4xx.c.

4.12.2.2 #define HSI_VALUE ((uint32_t)16000000)

Value of the Internal oscillator in Hz

Definition at line 73 of file system_stm32f4xx.c.

4.13 STM32F4xx_System_Private_TypesDefinitions

4.14 STM32F4xx_System_Private_Defines

Macros

- #define [VECT_TAB_OFFSET](#) 0x00

4.14.1 Detailed Description

4.14.2 Macro Definition Documentation

4.14.2.1 #define VECT_TAB_OFFSET 0x00

< Uncomment the following line if you need to relocate your vector Table in Internal SRAM. Vector Table base offset field. This value must be a multiple of 0x200.

Definition at line 97 of file system_stm32f4xx.c.

4.15 STM32F4xx_System_Private_Macros

4.16 STM32F4xx_System_Private_Variables

Variables

- uint32_t [SystemCoreClock](#) = 16000000
- __I uint8_t [AHBPrescTable](#) [16] = {0, 0, 0, 0, 0, 0, 0, 0, 1, 2, 3, 4, 6, 7, 8, 9}

4.16.1 Detailed Description

4.16.2 Variable Documentation

4.16.2.1 __I uint8_t AHBPrescTable[16] = {0, 0, 0, 0, 0, 0, 0, 0, 1, 2, 3, 4, 6, 7, 8, 9}

Definition at line 125 of file system_stm32f4xx.c.

4.16.2.2 uint32_t SystemCoreClock = 16000000

Definition at line 124 of file system_stm32f4xx.c.

4.17 STM32F4xx_System_Private_FunctionPrototypes

4.18 STM32F4xx_System_Private_Functions

Functions

- void [SystemInit](#) (void)
Setup the microcontroller system Initialize the FPU setting, vector table location and External memory configuration.
- void [SystemCoreClockUpdate](#) (void)
Update SystemCoreClock variable according to Clock Register Values. The SystemCoreClock variable contains the core clock (HCLK), it can be used by the user application to setup the SysTick timer or configure other parameters.

4.18.1 Detailed Description

4.18.2 Function Documentation

4.18.2.1 void SystemCoreClockUpdate (void)

Update SystemCoreClock variable according to Clock Register Values. The SystemCoreClock variable contains the core clock (HCLK), it can be used by the user application to setup the SysTick timer or configure other parameters.

Note

Each time the core clock (HCLK) changes, this function must be called to update SystemCoreClock variable value. Otherwise, any configuration based on this variable will be incorrect.

- The system frequency computed by this function is not the real frequency in the chip. It is calculated based on the predefined constant and the selected clock source:

- If SYSCLK source is HSI, SystemCoreClock will contain the [HSI_VALUE\(*\)](#)
- If SYSCLK source is HSE, SystemCoreClock will contain the [HSE_VALUE\(**\)](#)
- If SYSCLK source is PLL, SystemCoreClock will contain the [HSE_VALUE\(**\)](#) or [HSI_VALUE\(*\)](#) multiplied/divided by the PLL factors.

(*) HSI_VALUE is a constant defined in stm32f4xx_hal_conf.h file (default value 16 MHz) but the real value may vary depending on the variations in voltage and temperature.

(**) HSE_VALUE is a constant defined in stm32f4xx_hal_conf.h file (its value depends on the application requirements), user has to ensure that HSE_VALUE is same as the real frequency of the crystal used. Otherwise, this function may have wrong result.

- The result of this function could be not correct when using fractional value for HSE crystal.

Parameters

<i>None</i>	
-------------	--

Return values

<i>None</i>	
-------------	--

Definition at line 219 of file system_stm32f4xx.c.

4.18.2.2 void SystemInit (void)

Setup the microcontroller system Initialize the FPU setting, vector table location and External memory configuration.

Parameters

<i>None</i>	
-------------	--

Return values

<i>None</i>	
-------------	--

Definition at line 150 of file system_stm32f4xx.c.

Chapter 5

File Documentation

5.1 Src/GUIConf.c File Reference

```
#include "GUI.h"
```

Macros

- #define `GUI_NUMBYTES` (1024) * 150

Functions

- void `GUI_X_Config` (void)

5.1.1 Macro Definition Documentation

5.1.1.1 #define GUI_NUMBYTES (1024) * 150

Attention

Licensed under MCD-ST Liberty SW License Agreement V2, (the "License"); You may not use this file except in compliance with the License. You may obtain a copy of the License at:

http://www.st.com/software_license_agreement_liberty_v2

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Definition at line 65 of file GUIConf.c.

5.1.2 Function Documentation

5.1.2.1 void GUI_X_Config (void)

Definition at line 82 of file GUIConf.c.

5.2 Src/LCDConf_stm32f429i_disco_MB1075.c File Reference

```
#include "LCDConf.h"
#include "GUI_Private.h"
#include "ili9341.h"
```

Macros

- #define [LCD_SWAP_XY](#) 1
- #define [LCD_MIRROR_Y](#) 1
- #define [XSIZE_PHYS](#) 320
- #define [YSIZE_PHYS](#) 240
- #define [NUM_BUFFERS](#) 1
- #define [NUM_VSCREENS](#) 1
- #define [BK_COLOR](#) GUI_DARKBLUE
- #define [GUI_NUM_LAYERS](#) 1
- #define [COLOR_CONVERSION_0](#) GUICC_M8888I
- #define [DISPLAY_DRIVER_0](#) GUIDRV_LIN_32
- #define [LCD_LAYER0_FRAME_BUFFER](#) ((uint32_t)0xD0400000)
- #define [LCD_LAYER1_FRAME_BUFFER](#) ((uint32_t)0xD0600000)
- #define [DEFINEDMA2D_COLORCONVERSION](#)(PFX, PIXELFORMAT)

Functions

- void [HAL_DMA2D_MspInit](#) (DMA2D_HandleTypeDef *hdma2d)
DMA2D MSP Initialization This function configures the hardware resources used in This application:
- void [HAL_DMA2D_MspDeInit](#) (DMA2D_HandleTypeDef *hdma2d)
DMA2D MSP De-Initialization This function frees the hardware resources used in This application:
- void [HAL_LTDC_MspInit](#) (LTDC_HandleTypeDef *hltdc)
LTDC MSP Initialization This function configures the hardware resources used in This application:
- void [HAL_LTDC_MspDeInit](#) (LTDC_HandleTypeDef *hltdc)
LTDC MSP De-Initialization This function frees the hardware resources used in This application:
- void [HAL_LTDC_LineEvenCallback](#) (LTDC_HandleTypeDef *hltdc)
Line Event callback.
- void [LCD_X_Config](#) (void)
Called during the initialization process in order to set up the display driver configuration.
- int [LCD_X_DisplayDriver](#) (unsigned LayerIndex, unsigned Cmd, void *pData)
This function is called by the display driver for several purposes. To support the according task the routine needs to be adapted to the display controller. Please note that the commands marked with 'optional' are not cogently required and should only be adapted if the display controller supports these features.

Variables

- LTDC_HandleTypeDef [hltdc](#)

5.3 Src/maintask.c File Reference

```
#include "main.h"
#include "GUI.h"
#include "string.h"
#include "stm32f429i_discovery_lcd.h"
#include "stm32f4xx_hal.h"
#include "stm32f429i_discovery.h"
#include "math.h"
```

Macros

- #define [POWERUP_CAL_NUM](#) 200
- #define [FACOTRY_CAL_NUM](#) 200
- #define [MAJORVERSION](#) 3
- #define [MINORVERSION](#) 5

Functions

- void [ConfigureSPlandGPIO](#) (void)
- void [ConfigureMAX11254](#) (void)
- void [ReadMAX11254Results](#) (void)
- void [Calibration](#) (int8_t type, int32_t conversions)
- void [MainTask](#) (void)

Variables

- uint8_t [usbTxBuf](#) [32] = {0}
- uint32_t [SectorError](#) = 0
- USB_D_HandleTypeDef [USB_D_Device](#)
- uint8_t [request](#) = 0
- GUI_RECT [Rect](#) = {0, 0, 320, 240}
- uint8_t [spiWritePacket](#) [4] = {0}
- uint8_t [data0](#) [4]
- uint8_t [data1](#) [4]
- uint8_t [data2](#) [4]
- uint8_t [data3](#) [4]
- uint32_t [data0BO](#)
- uint32_t [data1BO](#)
- uint32_t [data2BO](#)
- uint32_t [data3BO](#)
- int [cell0Load](#)
- int [cell1Load](#)

- int [cell2Load](#)
- int [cell3Load](#)
- double [cell0Load0](#) =80606.899
- double [cell1Load0](#) =2316.475
- double [cell2Load0](#) =-4303.219
- double [cell3Load0](#) =48106.286
- uint32_t [cell0Load0Int](#)
- uint32_t [cell1Load0Int](#)
- uint32_t [cell2Load0Int](#)
- uint32_t [cell3Load0Int](#)
- uint32_t [cell0Load0Sign](#)
- uint32_t [cell1Load0Sign](#)
- uint32_t [cell2Load0Sign](#)
- uint32_t [cell3Load0Sign](#)
- double [cell0Load700](#) =1386219.167
- double [cell1Load700](#) =1306319.674
- double [cell2Load700](#) =1238527.327
- double [cell3Load700](#) =1278776.285
- uint32_t [cell0Load700Int](#)
- uint32_t [cell1Load700Int](#)
- uint32_t [cell2Load700Int](#)
- uint32_t [cell3Load700Int](#)
- uint32_t [cell0Load700Sign](#)
- uint32_t [cell1Load700Sign](#)
- uint32_t [cell2Load700Sign](#)
- uint32_t [cell3Load700Sign](#)
- double [cell0Powerup](#)
- double [cell1Powerup](#)
- double [cell2Powerup](#)
- double [cell3Powerup](#)
- double [cell0Grams](#)
- double [cell1Grams](#)
- double [cell2Grams](#)
- double [cell3Grams](#)
- double [x](#) =0
- double [y](#) =0
- double [r](#) =0

5.3.1 Detailed Description

```
Project: MAXREFDES82
Filename: maintask.c
Description: This module contains the major functions for the
             implementation of the example program for the MAXREFDES82.
```

Revision History:

03-02-2016 Rev 03.50 MG Initial release.

Copyright (C) 2016 Maxim Integrated Products, Inc., All Rights Reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy,

modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software. THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL MAXIM INTEGRATED BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of Maxim Integrated Products, Inc. shall not be used except as stated in the Maxim Integrated Products, Inc. Branding Policy.

The mere transfer of this software does not imply any licenses of trade secrets, proprietary technology, copyrights, patents, trademarks, maskwork rights, or any other form of intellectual property whatsoever. Maxim Integrated Products, Inc. retains all ownership rights.

Definition in file [maintask.c](#).

5.3.2 Macro Definition Documentation

5.3.2.1 `#define FACOTRY_CAL_NUM 200`

Definition at line 52 of file maintask.c.

5.3.2.2 `#define MAJORVERSION 3`

Definition at line 54 of file maintask.c.

5.3.2.3 `#define MINORVERSION 5`

Definition at line 55 of file maintask.c.

5.3.2.4 `#define POWERUP_CAL_NUM 200`

Definition at line 51 of file maintask.c.

5.3.3 Function Documentation

5.3.3.1 `void Calibration ( int8_t type, int32_t conversions )`

Definition at line 510 of file maintask.c.

5.3.3.2 `void ConfigureMAX11254 ( void )`

Definition at line 408 of file maintask.c.

5.3.3.3 `void ConfigureSPlandGPIO ( void )`

Definition at line 370 of file maintask.c.

5.3.3.4 void MainTask (void)

Definition at line 112 of file maintask.c.

5.3.3.5 void ReadMAX11254Results (void)

Definition at line 469 of file maintask.c.

5.3.4 Variable Documentation

5.3.4.1 double cell0Grams

Definition at line 93 of file maintask.c.

5.3.4.2 int cell0Load

Definition at line 77 of file maintask.c.

5.3.4.3 double cell0Load0 =80606.899

Definition at line 80 of file maintask.c.

5.3.4.4 uint32_t cell0Load0Int

Definition at line 81 of file maintask.c.

5.3.4.5 uint32_t cell0Load0Sign

Definition at line 82 of file maintask.c.

5.3.4.6 double cell0Load700 =1386219.167

Definition at line 85 of file maintask.c.

5.3.4.7 uint32_t cell0Load700Int

Definition at line 86 of file maintask.c.

5.3.4.8 uint32_t cell0Load700Sign

Definition at line 87 of file maintask.c.

5.3.4.9 double cell0Powerup

Definition at line 90 of file maintask.c.

5.3.4.10 **double cell1Grams**

Definition at line 93 of file maintask.c.

5.3.4.11 **int cell1Load**

Definition at line 77 of file maintask.c.

5.3.4.12 **double cell1Load0 =2316.475**

Definition at line 80 of file maintask.c.

5.3.4.13 **uint32_t cell1Load0Int**

Definition at line 81 of file maintask.c.

5.3.4.14 **uint32_t cell1Load0Sign**

Definition at line 82 of file maintask.c.

5.3.4.15 **double cell1Load700 =1306319.674**

Definition at line 85 of file maintask.c.

5.3.4.16 **uint32_t cell1Load700Int**

Definition at line 86 of file maintask.c.

5.3.4.17 **uint32_t cell1Load700Sign**

Definition at line 87 of file maintask.c.

5.3.4.18 **double cell1Powerup**

Definition at line 90 of file maintask.c.

5.3.4.19 **double cell2Grams**

Definition at line 93 of file maintask.c.

5.3.4.20 **int cell2Load**

Definition at line 77 of file maintask.c.

5.3.4.21 double cell2Load0 =-4303.219

Definition at line 80 of file maintask.c.

5.3.4.22 uint32_t cell2Load0Int

Definition at line 81 of file maintask.c.

5.3.4.23 uint32_t cell2Load0Sign

Definition at line 82 of file maintask.c.

5.3.4.24 double cell2Load700 =1238527.327

Definition at line 85 of file maintask.c.

5.3.4.25 uint32_t cell2Load700Int

Definition at line 86 of file maintask.c.

5.3.4.26 uint32_t cell2Load700Sign

Definition at line 87 of file maintask.c.

5.3.4.27 double cell2Powerup

Definition at line 90 of file maintask.c.

5.3.4.28 double cell3Grams

Definition at line 93 of file maintask.c.

5.3.4.29 int cell3Load

Definition at line 77 of file maintask.c.

5.3.4.30 double cell3Load0 =48106.286

Definition at line 80 of file maintask.c.

5.3.4.31 uint32_t cell3Load0Int

Definition at line 81 of file maintask.c.

5.3.4.32 uint32_t cell3Load0Sign

Definition at line 82 of file maintask.c.

5.3.4.33 double cell3Load700 =1278776.285

Definition at line 85 of file maintask.c.

5.3.4.34 uint32_t cell3Load700Int

Definition at line 86 of file maintask.c.

5.3.4.35 uint32_t cell3Load700Sign

Definition at line 87 of file maintask.c.

5.3.4.36 double cell3Powerup

Definition at line 90 of file maintask.c.

5.3.4.37 uint8_t data0[4]

Definition at line 71 of file maintask.c.

5.3.4.38 uint32_t data0BO

Definition at line 74 of file maintask.c.

5.3.4.39 uint8_t data1[4]

Definition at line 71 of file maintask.c.

5.3.4.40 uint32_t data1BO

Definition at line 74 of file maintask.c.

5.3.4.41 uint8_t data2[4]

Definition at line 71 of file maintask.c.

5.3.4.42 uint32_t data2BO

Definition at line 74 of file maintask.c.

5.3.4.43 uint8_t data3[4]

Definition at line 71 of file maintask.c.

5.3.4.44 uint32_t data3BO

Definition at line 74 of file maintask.c.

5.3.4.45 double r =0

Definition at line 100 of file maintask.c.

5.3.4.46 GUI_RECT Rect = {0, 0, 320, 240}

Definition at line 66 of file maintask.c.

5.3.4.47 uint8_t request = 0

Definition at line 64 of file maintask.c.

5.3.4.48 uint32_t SectorError = 0

Definition at line 60 of file maintask.c.

5.3.4.49 uint8_t spiWritePacket[4] = {0}

Definition at line 68 of file maintask.c.

5.3.4.50 USB_D_HandleTypeDef USB_D_Device

Definition at line 68 of file MAXREFDES82.c.

5.3.4.51 uint8_t usbTxBuf[32] ={0}

Definition at line 56 of file maintask.c.

5.3.4.52 double x =0

Definition at line 100 of file maintask.c.

5.3.4.53 double y =0

Definition at line 100 of file maintask.c.

5.4 Src/MAXREFDES82.c File Reference

```
#include "main.h"
#include "WM.h"
#include "stm32f429i_discovery_lcd.h"
```

Functions

- void [MainTask](#) (void)
- int [main](#) (void)

Main program.

Variables

- uint8_t [GUI_Initialized](#) = 0
- USB_D_HandleTypeDef [USB_Device](#)

5.4.1 Detailed Description

```
Project: MAXREFDES82
Filename: MAXREFDES82.c
Description: This module contains the Main application for the
             implementation of the example program for the MAXREFDES82.
```

Revision History:

03-02-2016 Rev 03.50 MG Initial release.

Copyright (C) 2016 Maxim Integrated Products, Inc., All Rights Reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software. THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL MAXIM INTEGRATED BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of Maxim Integrated Products, Inc. shall not be used except as stated in the Maxim Integrated Products, Inc. Branding Policy.

The mere transfer of this software does not imply any licenses of trade secrets, proprietary technology, copyrights, patents, trademarks, maskwork rights, or any other form of intellectual property whatsoever. Maxim Integrated Products, Inc. retains all ownership rights.

Definition in file [MAXREFDES82.c](#).

5.4.2 Function Documentation

5.4.2.1 int main (void)

Main program.

Parameters

None	
------	--

Return values

None	
------	--

Definition at line 82 of file MAXREFDES82.c.

5.4.2.2 void MainTask (void)

Definition at line 112 of file maintask.c.

5.4.3 Variable Documentation

5.4.3.1 uint8_t GUI_Initialized = 0

Definition at line 67 of file MAXREFDES82.c.

5.4.3.2 USBD_HandleTypeDef USBD_Device

Definition at line 68 of file MAXREFDES82.c.

5.5 Src/stm32f4xx_hal_msp.c File Reference

```
#include "stm32f4xx_hal.h"
#include "main.h"
```

Functions

- void [HAL_UART_MspInit](#) (UART_HandleTypeDef *huart)

UART MSP Initialization This function configures the hardware resources used in this application:

- void [HAL_UART_MspDeInit](#) (UART_HandleTypeDef *huart)

UART MSP De-Initialization This function frees the hardware resources used in this application:

5.6 Src/stm32f4xx_it.c File Reference

```
#include "main.h"
#include "stm32f4xx_it.h"
#include "GUI.h"
```

Functions

- void [NMI_Handler](#) (void)
This function handles NMI exception.
- void [HardFault_Handler](#) (void)
This function handles Hard Fault exception.
- void [MemManage_Handler](#) (void)
This function handles Memory Manage exception.
- void [BusFault_Handler](#) (void)
This function handles Bus Fault exception.
- void [UsageFault_Handler](#) (void)
This function handles Usage Fault exception.
- void [SVC_Handler](#) (void)
This function handles SVC call exception.
- void [DebugMon_Handler](#) (void)
This function handles Debug Monitor exception.
- void [PendSV_Handler](#) (void)
This function handles PendSV exception.
- void [SysTick_Handler](#) (void)
This function handles SysTick Handler.
- void [OTG_HS_IRQHandler](#) (void)
This function handles USB-On-The-Go FS global interrupt request.
- void [USARTx_DMA_TX_IRQHandler](#) (void)
This function handles DMA interrupt request.
- void [USARTx_IRQHandler](#) (void)
This function handles UART interrupt request.
- void [TIMx_IRQHandler](#) (void)
This function handles TIM interrupt request.
- void [LTDC_IRQHandler](#) (void)
This function handles LTDC global interrupt request.

Variables

- PCD_HandleTypeDef [hpcd](#)
- UART_HandleTypeDef [UartHandle](#)
- TIM_HandleTypeDef [TimHandle](#)
- volatile GUI_TIMER_TIME [OS_TimeMS](#)
- LTDC_HandleTypeDef [hltdc](#)

5.6.1 Function Documentation

5.6.1.1 void BusFault_Handler (void)

This function handles Bus Fault exception.

Parameters

<i>None</i>	
-------------	--

Return values

<i>None</i>	
-------------	--

Definition at line 101 of file stm32f4xx_it.c.

5.6.1.2 void DebugMon_Handler (void)

This function handles Debug Monitor exception.

Parameters

<i>None</i>	
-------------	--

Return values

<i>None</i>	
-------------	--

Definition at line 136 of file stm32f4xx_it.c.

5.6.1.3 void HardFault_Handler (void)

This function handles Hard Fault exception.

Parameters

<i>None</i>	
-------------	--

Return values

<i>None</i>	
-------------	--

Definition at line 75 of file stm32f4xx_it.c.

5.6.1.4 void LTDC_IRQHandler (void)

This function handles LTDC global interrupt request.

Parameters

<i>None</i>	
-------------	--

Return values

<i>None</i>	
-------------	--

Definition at line 218 of file stm32f4xx_it.c.

5.6.1.5 void MemManage_Handler (void)

This function handles Memory Manage exception.

Parameters

<i>None</i>	
-------------	--

Return values

<i>None</i>	
-------------	--

Definition at line 88 of file stm32f4xx_it.c.

5.6.1.6 void NMI_Handler (void)

This function handles NMI exception.

Parameters

<i>None</i>	
-------------	--

Return values

<i>None</i>	
-------------	--

Definition at line 66 of file stm32f4xx_it.c.

5.6.1.7 void OTG_HS_IRQHandler (void)

This function handles USB-On-The-Go FS global interrupt request.

Parameters

<i>None</i>	
-------------	--

Return values

<i>None</i>	
-------------	--

Definition at line 177 of file stm32f4xx_it.c.

5.6.1.8 void PendSV_Handler (void)

This function handles PendSVC exception.

Parameters

<i>None</i>	
-------------	--

Return values

<i>None</i>	
-------------	--

Definition at line 145 of file stm32f4xx_it.c.

5.6.1.9 void SVC_Handler (void)

This function handles SVCcall exception.

Parameters

<i>None</i>	
-------------	--

Return values

<i>None</i>	
-------------	--

Definition at line 127 of file stm32f4xx_it.c.

5.6.1.10 void SysTick_Handler (void)

This function handles SysTick Handler.

Parameters

<i>None</i>	
-------------	--

Return values

<i>None</i>	
-------------	--

Definition at line 154 of file stm32f4xx_it.c.

5.6.1.11 void TIMx_IRQHandler (void)

This function handles TIM interrupt request.

Parameters

<i>None</i>	
-------------	--

Return values

<i>None</i>	
-------------	--

Definition at line 208 of file stm32f4xx_it.c.

5.6.1.12 void UsageFault_Handler (void)

This function handles Usage Fault exception.

Parameters

None	
------	--

Return values

None	
------	--

Definition at line 114 of file stm32f4xx_it.c.

5.6.1.13 void USARTx_DMA_TX_IRQHandler (void)

This function handles DMA interrupt request.

Parameters

None	
------	--

Return values

None	
------	--

Definition at line 188 of file stm32f4xx_it.c.

5.6.1.14 void USARTx_IRQHandler (void)

This function handles UART interrupt request.

Parameters

None	
------	--

Return values

None	
------	--

Definition at line 198 of file stm32f4xx_it.c.

5.6.2 Variable Documentation

5.6.2.1 PCD_HandleTypeDef hpccd

Definition at line 36 of file usbd_conf.c.

5.6.2.2 volatile GUI_TIMER_TIME OS_TimeMS

5.6.2.3 TIM_HandleTypeDef TimHandle

Definition at line 58 of file usbd_cdc_interface.c.

5.6.2.4 UART_HandleTypeDef UartHandle

Definition at line 56 of file usbd_cdc_interface.c.

5.7 Src/system_stm32f4xx.c File Reference

CMSIS Cortex-M4 Device Peripheral Access Layer System Source File.

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

Macros

- #define [HSE_VALUE](#) ((uint32_t)8000000)
- #define [HSI_VALUE](#) ((uint32_t)16000000)
- #define [VECT_TAB_OFFSET](#) 0x00

Functions

- void [SystemInit](#) (void)
Setup the microcontroller system Initialize the FPU setting, vector table location and External memory configuration.
- void [SystemCoreClockUpdate](#) (void)
Update SystemCoreClock variable according to Clock Register Values. The SystemCoreClock variable contains the core clock (HCLK), it can be used by the user application to setup the SysTick timer or configure other parameters.

Variables

- uint32_t [SystemCoreClock](#) = 16000000
- __I uint8_t [AHBPrescTable](#) [16] = {0, 0, 0, 0, 0, 0, 0, 0, 1, 2, 3, 4, 6, 7, 8, 9}

5.7.1 Detailed Description

CMSIS Cortex-M4 Device Peripheral Access Layer System Source File.

Author

MCD Application Team

Version

V1.3.2

Date

13-November-2015 This file provides two functions and one global variable to be called from user application:

- [SystemInit\(\)](#): This function is called at startup just after reset and before branch to main program. This call is made inside the "startup_stm32f4xx.s" file.
- SystemCoreClock variable: Contains the core clock (HCLK), it can be used by the user application to setup the SysTick timer or configure other parameters.
- [SystemCoreClockUpdate\(\)](#): Updates the variable SystemCoreClock and must be called whenever the core clock is changed during program execution.

Attention

©COPYRIGHT 2015 STMicroelectronics

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of STMicroelectronics nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Definition in file [system_stm32f4xx.c](#).

5.8 Src/usbd_cdc_interface.c File Reference

```
#include "main.h"
```

Macros

- [#define APP_RX_DATA_SIZE 32](#)
- [#define APP_TX_DATA_SIZE 32](#)

Functions

- void [HAL_TIM_PeriodElapsedCallback](#) (TIM_HandleTypeDef *htim)
TIM period elapsed callback.
- void [HAL_UART_RxCpltCallback](#) (UART_HandleTypeDef *huart)
Rx Transfer completed callback.
- void [HAL_UART_TxCpltCallback](#) (UART_HandleTypeDef *huart)
Tx Transfer completed callback.
- void [HAL_UART_ErrorCallback](#) (UART_HandleTypeDef *UartHandle)
UART error callbacks.

Variables

- uint8_t [request](#)
- USBDCDC_LineCodingTypeDef [LineCoding](#)
- uint8_t [UserRxBuffer](#) [APP_RX_DATA_SIZE]
- uint8_t [UserTxBuffer](#) [APP_TX_DATA_SIZE]
- uint32_t [UserTxBufPtrIn](#) = 0
- uint32_t [UserTxBufPtrOut](#) = 0
- UART_HandleTypeDef [UartHandle](#)
- TIM_HandleTypeDef [TimHandle](#)
- USBDCDC_HandleTypeDef [USBDCDC_Device](#)
- USBDCDC_ItfTypeDef [USBDCDC_fops](#)

5.8.1 Macro Definition Documentation

5.8.1.1 #define APP_RX_DATA_SIZE 32

Definition at line 33 of file usbd_cdc_interface.c.

5.8.1.2 #define APP_TX_DATA_SIZE 32

Definition at line 34 of file usbd_cdc_interface.c.

5.8.2 Function Documentation

5.8.2.1 void HAL_TIM_PeriodElapsedCallback (TIM_HandleTypeDef * htim)

TIM period elapsed callback.

Parameters

<i>htim</i> ,:	TIM handle
----------------	------------

Return values

<i>None</i>

Definition at line 226 of file usbd_cdc_interface.c.

5.8.2.2 void HAL_UART_ErrorCallback (UART_HandleTypeDef * *UartHandle*)

UART error callbacks.

Parameters

<i>UartHandle</i> ,:	UART handle
----------------------	-------------

Return values

<i>None</i>	
-------------	--

Definition at line 452 of file usbd_cdc_interface.c.

5.8.2.3 void HAL_UART_RxCpltCallback (UART_HandleTypeDef * *huart*)

Rx Transfer completed callback.

Parameters

<i>huart</i> ,:	UART handle
-----------------	-------------

Return values

<i>None</i>	
-------------	--

Definition at line 262 of file usbd_cdc_interface.c.

5.8.2.4 void HAL_UART_TxCpltCallback (UART_HandleTypeDef * *huart*)

Tx Transfer completed callback.

Parameters

<i>huart</i> ,:	UART handle
-----------------	-------------

Return values

<i>None</i>	
-------------	--

Definition at line 298 of file usbd_cdc_interface.c.

5.8.3 Variable Documentation**5.8.3.1 USBDCDC_LineCodingTypeDef LineCoding**

Initial value:

```
= {
    115200,
    0x00,
    0x00,
    0x08
}
```

```
}
```

Definition at line 40 of file usbd_cdc_interface.c.

5.8.3.2 uint8_t request

Definition at line 64 of file maintask.c.

5.8.3.3 TIM_HandleTypeDef TimHandle

Definition at line 58 of file usbd_cdc_interface.c.

5.8.3.4 UART_HandleTypeDef UartHandle

Definition at line 56 of file usbd_cdc_interface.c.

5.8.3.5 USB_D_CDC_ItfTypeDef USB_D_CDC_fops

Initial value:

```
=
{
    CDC_Itf_Init,
    CDC_Itf_DeInit,
    CDC_Itf_Control,
    CDC_Itf_Receive
}
```

Definition at line 71 of file usbd_cdc_interface.c.

5.8.3.6 USB_D_HandleTypeDef USB_D_Device

Definition at line 68 of file MAXREFDES82.c.

5.8.3.7 uint8_t UserRxBuffer[APP_RX_DATA_SIZE]

Definition at line 47 of file usbd_cdc_interface.c.

5.8.3.8 uint8_t UserTxBuffer[APP_TX_DATA_SIZE]

Definition at line 48 of file usbd_cdc_interface.c.

5.8.3.9 uint32_t UserTxBufPtrIn = 0

Definition at line 50 of file usbd_cdc_interface.c.

5.8.3.10 uint32_t UserTxBufPtrOut = 0

Definition at line 52 of file usbd_cdc_interface.c.

5.9 Src/usbd_conf.c File Reference

```
#include "stm32f4xx_hal.h"
#include "usbd_core.h"
```

Functions

- void [HAL_PCD_MspInit](#) (PCD_HandleTypeDef *hpcd)
Initializes the PCD MSP.
- void [HAL_PCD_MspDeInit](#) (PCD_HandleTypeDef *hpcd)
DeInitializes the PCD MSP.
- void [HAL_PCD_SetupStageCallback](#) (PCD_HandleTypeDef *hpcd)
Setup stage callback.
- void [HAL_PCD_DataOutStageCallback](#) (PCD_HandleTypeDef *hpcd, uint8_t epnum)
Data Out stage callback.
- void [HAL_PCD_DataInStageCallback](#) (PCD_HandleTypeDef *hpcd, uint8_t epnum)
Data In stage callback.
- void [HAL_PCD_SOFCallback](#) (PCD_HandleTypeDef *hpcd)
SOF callback.
- void [HAL_PCD_ResetCallback](#) (PCD_HandleTypeDef *hpcd)
Reset callback.
- void [HAL_PCD_SuspendCallback](#) (PCD_HandleTypeDef *hpcd)
Suspend callback.
- void [HAL_PCD_ResumeCallback](#) (PCD_HandleTypeDef *hpcd)
Resume callback.
- void [HAL_PCD_ISOOUTIncompleteCallback](#) (PCD_HandleTypeDef *hpcd, uint8_t epnum)
ISOC Out Incomplete callback.
- void [HAL_PCD_ISOINIncompleteCallback](#) (PCD_HandleTypeDef *hpcd, uint8_t epnum)
ISOC In Incomplete callback.
- void [HAL_PCD_ConnectCallback](#) (PCD_HandleTypeDef *hpcd)
Connect callback.
- void [HAL_PCD_DisconnectCallback](#) (PCD_HandleTypeDef *hpcd)
Disconnect callback.
- USB_StatusTypeDef [USB_LL_Init](#) (USB_HandleTypeDef *pdev)
Initializes the Low Level portion of the Device driver.
- USB_StatusTypeDef [USB_LL_DeInit](#) (USB_HandleTypeDef *pdev)
De-Initializes the Low Level portion of the Device driver.
- USB_StatusTypeDef [USB_LL_Start](#) (USB_HandleTypeDef *pdev)
Starts the Low Level portion of the Device driver.
- USB_StatusTypeDef [USB_LL_Stop](#) (USB_HandleTypeDef *pdev)
Stops the Low Level portion of the Device driver.
- USB_StatusTypeDef [USB_LL_OpenEP](#) (USB_HandleTypeDef *pdev, uint8_t ep_addr, uint8_t ep_type, uint16_t ep_mps)
Opens an endpoint of the Low Level Driver.
- USB_StatusTypeDef [USB_LL_CloseEP](#) (USB_HandleTypeDef *pdev, uint8_t ep_addr)
Closes an endpoint of the Low Level Driver.

- USB_StatusTypeDef [USB_LL_FlushEP](#) (USB_HandleTypeDef *pdev, uint8_t ep_addr)
Flushes an endpoint of the Low Level Driver.
- USB_StatusTypeDef [USB_LL_StallEP](#) (USB_HandleTypeDef *pdev, uint8_t ep_addr)
Sets a Stall condition on an endpoint of the Low Level Driver.
- USB_StatusTypeDef [USB_LL_ClearStallEP](#) (USB_HandleTypeDef *pdev, uint8_t ep_addr)
Clears a Stall condition on an endpoint of the Low Level Driver.
- uint8_t [USB_LL_IsStallEP](#) (USB_HandleTypeDef *pdev, uint8_t ep_addr)
Returns Stall condition.
- USB_StatusTypeDef [USB_LL_SetUSBAddress](#) (USB_HandleTypeDef *pdev, uint8_t dev_addr)
Assigns an USB address to the device.
- USB_StatusTypeDef [USB_LL_Transmit](#) (USB_HandleTypeDef *pdev, uint8_t ep_addr, uint8_t *pbuf, uint16_t size)
Transmits data over an endpoint.
- USB_StatusTypeDef [USB_LL_PrepareReceive](#) (USB_HandleTypeDef *pdev, uint8_t ep_addr, uint8_t *pbuf, uint16_t size)
Prepares an endpoint for reception.
- uint32_t [USB_LL_GetRxDataSize](#) (USB_HandleTypeDef *pdev, uint8_t ep_addr)
Returns the last transfered packet size.
- void [USB_LL_Delay](#) (uint32_t Delay)
Delay routine for the USB Device Library.

Variables

- PCD_HandleTypeDef [hpcd](#)

5.9.1 Function Documentation

5.9.1.1 void HAL_PCD_ConnectCallback (PCD_HandleTypeDef * *hpcd*)

Connect callback.

Parameters

<i>hpcd</i> ,:	PCD handle
----------------	------------

Return values

<i>None</i>

Definition at line 325 of file usbd_conf.c.

5.9.1.2 void HAL_PCD_DataInStageCallback (PCD_HandleTypeDef * *hpcd*, uint8_t *epnum*)

Data In stage callback.

Parameters

<i>hpcd</i> ,:	PCD handle
<i>epnum</i> ,:	Endpoint Number

Return values

<i>None</i>	
-------------	--

Definition at line 233 of file usbd_conf.c.

5.9.1.3 void HAL_PCD_DataOutStageCallback (PCD_HandleTypeDef * *hpcd*, uint8_t *epnum*)

Data Out stage callback.

Parameters

<i>hpcd</i> ,:	PCD handle
<i>epnum</i> ,:	Endpoint Number

Return values

<i>None</i>	
-------------	--

Definition at line 222 of file usbd_conf.c.

5.9.1.4 void HAL_PCD_DisconnectCallback (PCD_HandleTypeDef * *hpcd*)

Disconnect callback.

Parameters

<i>hpcd</i> ,:	PCD handle
----------------	------------

Return values

<i>None</i>	
-------------	--

Definition at line 335 of file usbd_conf.c.

5.9.1.5 void HAL_PCD_ISOINIncompleteCallback (PCD_HandleTypeDef * *hpcd*, uint8_t *epnum*)

ISOC In Incomplete callback.

Parameters

<i>hpcd</i> ,:	PCD handle
<i>epnum</i> ,:	Endpoint Number

Return values

<i>None</i>	
-------------	--

Definition at line 315 of file usbd_conf.c.

5.9.1.6 void HAL_PCD_ISOOUTIncompleteCallback (PCD_HandleTypeDef * *hpcd*, uint8_t *epnum*)

ISOC Out Incomplete callback.

Parameters

<i>hpcd</i> ,:	PCD handle
<i>epnum</i> ,:	Endpoint Number

Return values

<i>None</i>	
-------------	--

Definition at line 304 of file usbd_conf.c.

5.9.1.7 void HAL_PCD_MspDeInit (PCD_HandleTypeDef * *hpcd*)

DeInitializes the PCD MSP.

Parameters

<i>hpcd</i> ,:	PCD handle
----------------	------------

Return values

<i>None</i>	
-------------	--

Definition at line 185 of file usbd_conf.c.

5.9.1.8 void HAL_PCD_MspInit (PCD_HandleTypeDef * *hpcd*)

Initializes the PCD MSP.

Parameters

<i>hpcd</i> ,:	PCD handle
----------------	------------

Return values

<i>None</i>	
-------------	--

Definition at line 49 of file usbd_conf.c.

5.9.1.9 void HAL_PCD_ResetCallback (PCD_HandleTypeDef * *hpcd*)

Reset callback.

Parameters

<i>hpcd</i> ,:	PCD handle
----------------	------------

Return values

<i>None</i>	
-------------	--

Definition at line 253 of file usbd_conf.c.

5.9.1.10 void HAL_PCD_ResumeCallback (PCD_HandleTypeDef * *hpcd*)

Resume callback.

Parameters

<i>hpcd</i> ,:	PCD handle
----------------	------------

Return values

<i>None</i>	
-------------	--

Definition at line 293 of file usbd_conf.c.

5.9.1.11 void HAL_PCD_SetupStageCallback (PCD_HandleTypeDef * *hpcd*)

Setup stage callback.

Parameters

<i>hpcd</i> ,:	PCD handle
----------------	------------

Return values

<i>None</i>	
-------------	--

Definition at line 211 of file usbd_conf.c.

5.9.1.12 void HAL_PCD_SOFCallback (PCD_HandleTypeDef * *hpcd*)

SOF callback.

Parameters

<i>hpcd</i> ,:	PCD handle
----------------	------------

Return values

<i>None</i>	
-------------	--

Definition at line 243 of file usbd_conf.c.

5.9.1.13 void HAL_PCD_SuspendCallback (PCD_HandleTypeDef * *hpcd*)

Suspend callback.

Parameters

<i>hpcd,:</i>	PCD handle
---------------	------------

Return values

<i>None</i>	
-------------	--

Definition at line 283 of file usbd_conf.c.

5.9.1.14 USBD_StatusTypeDef USBD_LL_ClearStallEP (USBD_HandleTypeDef * *pdev*, uint8_t *ep_addr*)

Clears a Stall condition on an endpoint of the Low Level Driver.

Parameters

<i>pdev,:</i>	Device handle
<i>ep_addr,:</i>	Endpoint Number

Return values

<i>USB</i>	Status
------------	--------

Definition at line 509 of file usbd_conf.c.

5.9.1.15 USBD_StatusTypeDef USBD_LL_CloseEP (USBD_HandleTypeDef * *pdev*, uint8_t *ep_addr*)

Closes an endpoint of the Low Level Driver.

Parameters

<i>pdev,:</i>	Device handle
<i>ep_addr,:</i>	Endpoint Number

Return values

<i>USB</i>	Status
------------	--------

Definition at line 473 of file usbd_conf.c.

5.9.1.16 USBD_StatusTypeDef USBD_LL_DeInit (USBD_HandleTypeDef * *pdev*)

De-Initializes the Low Level portion of the Device driver.

Parameters

<i>pdev,:</i>	Device handle
---------------	---------------

Return values

<i>USB</i>	Status
------------	--------

Definition at line 422 of file usbd_conf.c.

5.9.1.17 void USBD_LL_Delay (uint32_t Delay)

Delay routine for the USB Device Library.

Parameters

<i>Delay,:</i>	Delay in ms
----------------	-------------

Return values

<i>None</i>

Definition at line 596 of file usbd_conf.c.

5.9.1.18 USBD_StatusTypeDef USBD_LL_FlushEP (USBD_HandleTypeDef * pdev, uint8_t ep_addr)

Flushes an endpoint of the Low Level Driver.

Parameters

<i>pdev,:</i>	Device handle
<i>ep_addr,:</i>	Endpoint Number

Return values

<i>USB</i>	Status
------------	--------

Definition at line 485 of file usbd_conf.c.

5.9.1.19 uint32_t USBD_LL_GetRxDataSize (USBD_HandleTypeDef * pdev, uint8_t ep_addr)

Returns the last transfered packet size.

Parameters

<i>pdev,:</i>	Device handle
<i>ep_addr,:</i>	Endpoint Number

Return values

<i>Recived</i>	Data Size
----------------	-----------

Definition at line 586 of file usbd_conf.c.

5.9.1.20 USBD_StatusTypeDef USBD_LL_Init (USBD_HandleTypeDef * pdev)

Initializes the Low Level portion of the Device driver.

Parameters

<i>pdev</i> ,:	Device handle
----------------	---------------

Return values

<i>USB</i> D	Status
--------------	--------

Definition at line 348 of file usbd_conf.c.

5.9.1.21 uint8_t USBD_LL_IsStallEP (USBD_HandleTypeDef * *pdev*, uint8_t *ep_addr*)

Returns Stall condition.

Parameters

<i>pdev</i> ,:	Device handle
<i>ep_addr</i> ,:	Endpoint Number

Return values

<i>Stall</i>	(1: yes, 0: No)
--------------	-----------------

Definition at line 521 of file usbd_conf.c.

5.9.1.22 USBD_StatusTypeDef USBD_LL_OpenEP (USBD_HandleTypeDef * *pdev*, uint8_t *ep_addr*, uint8_t *ep_type*, uint16_t *ep_mps*)

Opens an endpoint of the Low Level Driver.

Parameters

<i>pdev</i> ,:	Device handle
<i>ep_addr</i> ,:	Endpoint Number
<i>ep_type</i> ,:	Endpoint Type
<i>ep_mps</i> ,:	Endpoint Max Packet Size

Return values

<i>USB</i> D	Status
--------------	--------

Definition at line 458 of file usbd_conf.c.

5.9.1.23 USBD_StatusTypeDef USBD_LL_PrepareReceive (USBD_HandleTypeDef * *pdev*, uint8_t *ep_addr*, uint8_t * *pbuf*, uint16_t *size*)

Prepares an endpoint for reception.

Parameters

<i>pdev</i> ,:	Device handle
<i>ep_addr</i> ,:	Endpoint Number

<i>pbuf:pointer</i>	to data to be received
<i>size,:</i>	data size

Return values

<i>USB</i>	Status
------------	--------

Definition at line 572 of file usbd_conf.c.

5.9.1.24 USB_StatusTypeDef USB_LL_SetUSBAddress (USB_HandleTypeDef * *pdev*, uint8_t *dev_addr*)

Assigns an USB address to the device.

Parameters

<i>pdev,:</i>	Device handle
<i>dev_addr,:</i>	USB address

Return values

<i>USB</i>	Status
------------	--------

Definition at line 541 of file usbd_conf.c.

5.9.1.25 USB_StatusTypeDef USB_LL_StallEP (USB_HandleTypeDef * *pdev*, uint8_t *ep_addr*)

Sets a Stall condition on an endpoint of the Low Level Driver.

Parameters

<i>pdev,:</i>	Device handle
<i>ep_addr,:</i>	Endpoint Number

Return values

<i>USB</i>	Status
------------	--------

Definition at line 497 of file usbd_conf.c.

5.9.1.26 USB_StatusTypeDef USB_LL_Start (USB_HandleTypeDef * *pdev*)

Starts the Low Level portion of the Device driver.

Parameters

<i>pdev,:</i>	Device handle
---------------	---------------

Return values

<i>USB</i>	Status
------------	--------

Definition at line 433 of file usbd_conf.c.

5.9.1.27 USB_StatusTypeDef USB_LL_Stop (USB_HandleTypeDef * *pdev*)

Stops the Low Level portion of the Device driver.

Parameters

<i>pdev</i> ,:	Device handle
----------------	---------------

Return values

<i>USB</i>	Status
------------	--------

Definition at line 444 of file usbd_conf.c.

5.9.1.28 USB_StatusTypeDef USB_LL_Transmit (USB_HandleTypeDef * *pdev*, uint8_t *ep_addr*, uint8_t * *pbuf*, uint16_t *size*)

Transmits data over an endpoint.

Parameters

<i>pdev</i> ,:	Device handle
<i>ep_addr</i> ,:	Endpoint Number
<i>pbuf</i> ,:	Pointer to data to be sent
<i>size</i> ,:	Data size

Return values

<i>USB</i>	Status
------------	--------

Definition at line 555 of file usbd_conf.c.

5.9.2 Variable Documentation**5.9.2.1 PCD_HandleTypeDef *hpcd***

Definition at line 36 of file usbd_conf.c.

5.10 Src/usbd_desc.c File Reference

```
#include "usbd_core.h"
#include "usbd_desc.h"
#include "usbd_conf.h"
```


Macros

- #define [USBD_VID](#) 0x0483
- #define [USBD_PID](#) 0x5740
- #define [USBD_LANGID_STRING](#) 0x409
- #define [USBD_MANUFACTURER_STRING](#) "STMicroelectronics"
- #define [USBD_PRODUCT_HS_STRING](#) "STM32 Virtual ComPort in HS Mode"
- #define [USBD_PRODUCT_FS_STRING](#) "STM32 Virtual ComPort in FS Mode"
- #define [USBD_CONFIGURATION_HS_STRING](#) "VCP Config"
- #define [USBD_INTERFACE_HS_STRING](#) "VCP Interface"
- #define [USBD_CONFIGURATION_FS_STRING](#) "VCP Config"
- #define [USBD_INTERFACE_FS_STRING](#) "VCP Interface"

Functions

- uint8_t * [USBD_VCP_DeviceDescriptor](#) (USBD_SpeedTypeDef speed, uint16_t *length)
Returns the device descriptor.
- uint8_t * [USBD_VCP_LangIDStrDescriptor](#) (USBD_SpeedTypeDef speed, uint16_t *length)
Returns the LangID string descriptor.
- uint8_t * [USBD_VCP_ManufacturerStrDescriptor](#) (USBD_SpeedTypeDef speed, uint16_t *length)
Returns the manufacturer string descriptor.
- uint8_t * [USBD_VCP_ProductStrDescriptor](#) (USBD_SpeedTypeDef speed, uint16_t *length)
Returns the product string descriptor.
- uint8_t * [USBD_VCP_SerialStrDescriptor](#) (USBD_SpeedTypeDef speed, uint16_t *length)
Returns the serial number string descriptor.
- uint8_t * [USBD_VCP_ConfigStrDescriptor](#) (USBD_SpeedTypeDef speed, uint16_t *length)
Returns the configuration string descriptor.
- uint8_t * [USBD_VCP_InterfaceStrDescriptor](#) (USBD_SpeedTypeDef speed, uint16_t *length)
Returns the interface string descriptor.

Variables

- USBD_DescriptorsTypeDef [VCP_Desc](#)
- __ALIGN_BEGIN uint8_t
USBD_DeviceDesc[USB_LEN_DEV_DESC] [__ALIGN_END](#)
- uint8_t [USBD_StringSerial](#) [USB_SIZ_STRING_SERIAL]

5.10.1 Macro Definition Documentation

5.10.1.1 #define USBD_CONFIGURATION_FS_STRING "VCP Config"

Definition at line 43 of file usbd_desc.c.

5.10.1.2 #define USBD_CONFIGURATION_HS_STRING "VCP Config"

Definition at line 41 of file usbd_desc.c.

5.10.1.3 #define USBD_INTERFACE_FS_STRING "VCP Interface"

Definition at line 44 of file usbd_desc.c.

5.10.1.4 #define USBD_INTERFACE_HS_STRING "VCP Interface"

Definition at line 42 of file usbd_desc.c.

5.10.1.5 #define USBD_LANGID_STRING 0x409

Definition at line 37 of file usbd_desc.c.

5.10.1.6 #define USBD_MANUFACTURER_STRING "STMicroelectronics"

Definition at line 38 of file usbd_desc.c.

5.10.1.7 #define USBD_PID 0x5740

Definition at line 36 of file usbd_desc.c.

5.10.1.8 #define USBD_PRODUCT_FS_STRING "STM32 Virtual ComPort in FS Mode"

Definition at line 40 of file usbd_desc.c.

5.10.1.9 #define USBD_PRODUCT_HS_STRING "STM32 Virtual ComPort in HS Mode"

Definition at line 39 of file usbd_desc.c.

5.10.1.10 #define USBD_VID 0x0483

Definition at line 35 of file usbd_desc.c.

5.10.2 Function Documentation**5.10.2.1 uint8_t * USBD_VCP_ConfigStrDescriptor (USBD_SpeedTypeDef *speed*, uint16_t * *length*)**

Returns the configuration string descriptor.

Parameters

<i>speed</i> ,:	Current device speed
<i>length</i> ,:	Pointer to data length variable

Return values

<i>Pointer</i>	to descriptor buffer
----------------	----------------------

Definition at line 198 of file usbd_desc.c.

5.10.2.2 uint8_t * USB_D_VCP_DeviceDescriptor (USB_SpeedTypeDef *speed*, uint16_t * *length*)

Returns the device descriptor.

Parameters

<i>speed</i> ,:	Current device speed
<i>length</i> ,:	Pointer to data length variable

Return values

<i>Pointer</i>	to descriptor buffer
----------------	----------------------

Definition at line 127 of file usbd_desc.c.

5.10.2.3 uint8_t * USB_D_VCP_InterfaceStrDescriptor (USB_SpeedTypeDef *speed*, uint16_t * *length*)

Returns the interface string descriptor.

Parameters

<i>speed</i> ,:	Current device speed
<i>length</i> ,:	Pointer to data length variable

Return values

<i>Pointer</i>	to descriptor buffer
----------------	----------------------

Definition at line 217 of file usbd_desc.c.

5.10.2.4 uint8_t * USB_D_VCP_LangIDStrDescriptor (USB_SpeedTypeDef *speed*, uint16_t * *length*)

Returns the LangID string descriptor.

Parameters

<i>speed</i> ,:	Current device speed
<i>length</i> ,:	Pointer to data length variable

Return values

<i>Pointer</i>	to descriptor buffer
----------------	----------------------

Definition at line 139 of file usbd_desc.c.

5.10.2.5 uint8_t * USB_D_VCP_ManufacturerStrDescriptor (USB_SpeedTypeDef *speed*, uint16_t * *length*)

Returns the manufacturer string descriptor.

Parameters

<i>speed</i> ,:	Current device speed
<i>length</i> ,:	Pointer to data length variable

Return values

<i>Pointer</i>	to descriptor buffer
----------------	----------------------

Definition at line 170 of file usbd_desc.c.

5.10.2.6 uint8_t * USB_D_VCP_ProductStrDescriptor (USB_SpeedTypeDef *speed*, uint16_t * *length*)

Returns the product string descriptor.

Parameters

<i>speed</i> ,:	Current device speed
<i>length</i> ,:	Pointer to data length variable

Return values

<i>Pointer</i>	to descriptor buffer
----------------	----------------------

Definition at line 151 of file usbd_desc.c.

5.10.2.7 uint8_t * USB_D_VCP_SerialStrDescriptor (USB_SpeedTypeDef *speed*, uint16_t * *length*)

Returns the serial number string descriptor.

Parameters

<i>speed</i> ,:	Current device speed
<i>length</i> ,:	Pointer to data length variable

Return values

<i>Pointer</i>	to descriptor buffer
----------------	----------------------

Definition at line 182 of file usbd_desc.c.

5.10.3 Variable Documentation

5.10.3.1 __ALIGN_BEGIN uint8_t USB_D_StrDesc [USB_D_MAX_STR_DESC_SIZ] __ALIGN_END

Initial value:

```
= {
    0x12,
    USB_DESC_TYPE_DEVICE,
    0x00,
    0x02,
    0x00,
    0x00,
```

```

0x00,
USB_MAX_EP0_SIZE,
LOBYTE(USBD_VID),
HIBYTE(USBD_VID),
LOBYTE(USBD_PID),
HIBYTE(USBD_PID),
0x00,
0x02,
USBD_IDX_MFC_STR,
USBD_IDX_PRODUCT_STR,
USBD_IDX_SERIAL_STR,
USBD_MAX_NUM_CONFIGURATION
}

```

< IAR Compiler

Definition at line 74 of file usbd_desc.c.

5.10.3.2 uint8_t USBD_StringSerial[USB_SIZ_STRING_SERIAL]

Initial value:

```

=
{
    USB_SIZ_STRING_SERIAL,
    USB_DESC_TYPE_STRING,
}

```

IAR Compiler

Definition at line 106 of file usbd_desc.c.

5.10.3.3 USBD_DescriptorsTypeDef VCP_Desc

Initial value:

```

= {
    USBD_VCP_DeviceDescriptor,
    USBD_VCP_LangIDStrDescriptor,
    USBD_VCP_ManufacturerStrDescriptor,
    USBD_VCP_ProductStrDescriptor,
    USBD_VCP_SerialStrDescriptor,
    USBD_VCP_ConfigStrDescriptor,
    USBD_VCP_InterfaceStrDescriptor,
}

```

Definition at line 60 of file usbd_desc.c.